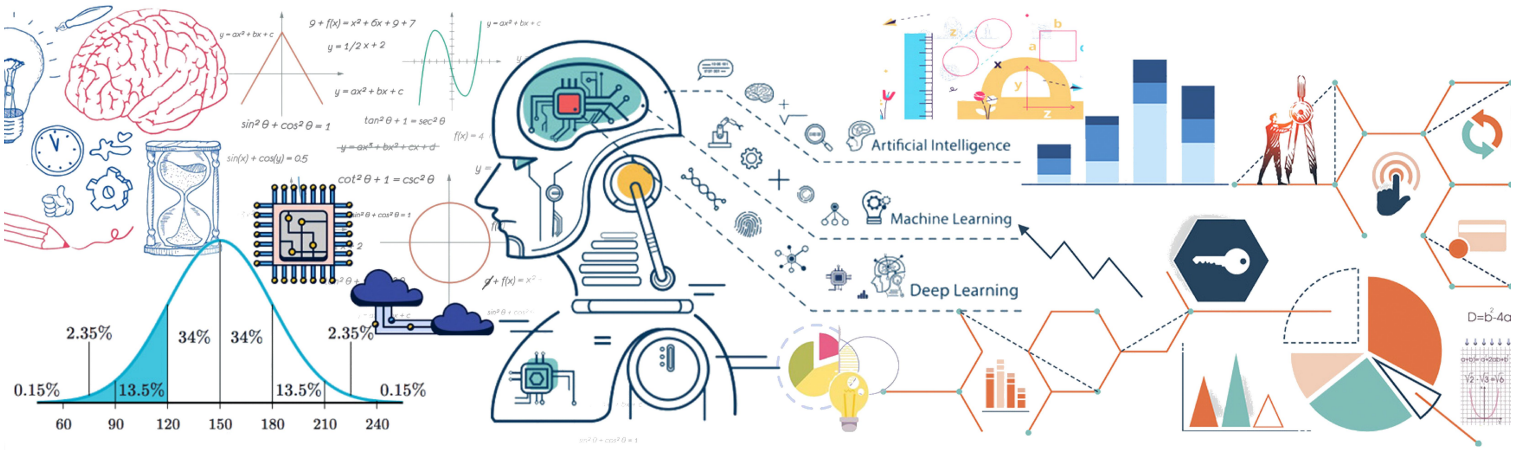
CAAST-NAHEP के अंतर्गत रिसर्च स्कॉलर्स के लिए प्रशिक्षण

Training for Research Scholars of NARES Under CAAST-NAHEP

प्रयोगात्मक डेटा प्रबंधन और विश्लेषण में सांख्यिकी और सूचना विज्ञान

Statistics and Informatics in Experimental Data Management & Analysis

23 फरवरी - 04 मार्च, 2021
February 23 – March 04, 2021



संदर्भ पुस्तिका
Reference Manual

पाठ्यक्रम निदेशक
Course Director

डॉ. सीमा जग्गी
Dr. Seema Jaggi

डॉ. सुदीप
Dr. Sudeep

पाठ्यक्रम समन्वयक
Course Coordinator

डॉ. अनिंदिता दत्ता
Dr. Anindita Datta

डॉ. सौमेन पाल
Dr. Soumen Pal

श्री संजीव कुमार
Mr. Sanjeev Kumar

भा. कृ. अनु. प. - भारतीय कृषि सांख्यिकी अनुसंधान संस्थान
लाइब्रेरी एवेन्यू, पूसा, नई दिल्ली - 110012
ICAR - Indian Agricultural Statistics Research Institute
Library Avenue, Pusa, New Delhi - 110012

CONTENTS

S. NO.	TOPIC	PAGE
1	Descriptive Statistics - *Seema Jaggi and Anindita Datta*	1
2	Exploratory Data Analysis - *Cini Varghese and Anindita Datta*	12
3	Correlation and Regression - *B.N. Mandal*	19
4	Tests of Significance - *Seema Jaggi*	44
5	Diagnostics and Remedial Measures - *Rajender Parsad and Lalmohan Bhar*	62
6	Planning of Experiments and Basic Experimental Designs - *Seema Jaggi and Anindita Datta*	79
7	Incomplete Block Designs - *Cini Varghese*	98
8	ANCOVA - *Sukanta Dash*	107
9	Analysis Of Groups Of Experiments - *Susheel Sarkar*	112
10	Experimental Data Analysis using R Software - *B.N. Mandal*	113
11	Factorial Experiments - *Seema Jaggi*	123
12	Split and Strip Plot Designs - *Susheel Kumar Sarkar and Anindita Datta*	137
13	Response Surface Designs - *Eldho Varghese*	149
14	Probit Analysis of Dose-Response Data - *Arpan Bhowmik and Lalmohan Bhar*	165
15	Logistic Regression - *Arpan Bhowmik*	180
16	Principal Components Analysis - *Rahul Banerjee*	185
17	Factor Analysis - *Rahul Banerjee*	197
18	Cluster Analysis - *Arpan Bhowmik*	208
19	Stability Analysis: Part I - *A.R. Rao*	222
20	Stability Analysis: Part II Additive Main Effects and Multiplicative Interaction (AMMI) Model in Crop Improvement - *A.R. Rao*	238
21	*Data Mining: An Overview* - *Shashi Dahiya*	248
22	Knowledge Representation using Ontologies - *Sudeep*	261

23	Association Rules Mining using R *- Anshu Bharadwaj*	281
24	Text Mining and Natural Language Processing (NLP) *- Sanchita Naha*	293
25	Machine Learning Techniques – ANN and SVM *- Chandan Kumar Deb*	298
26	Overview of Python *- Ashraful Haque*	310
27	*Database Management System Using MS Access* *-Soumen Pal*	321
28	SEQUENCE ALIGNMENT *-SB Lal*	331
29	Transcriptome Analysis *- Md Sameer Farooqi*	349
30	ASHOKA: The Supercomputing Hub *-K.K. Chaturvedi*	356
31	Genome Wide Association Study and Genomic Selection *-Neeraj Budhlakoti, D C Mishra and Anil Rai*	363
32	Computational Metagenomics *-Anu Sharma*	380
33	Introduction to Proteomics *-Sudhir Srivastava, Dwijesh Chandra Mishra and K. K. Chaturvedi*	387
34	Deep Learning: An Introduction *-Kanchan Sinha*	393

DESCRIPTIVE STATISTICS

Seema Jaggi and Anindita Datta
ICAR-IASRI, Library Avenue, New Delhi – 110 012
seema.jaggi@icar.gov.in, anindita.datta@icar.gov.in

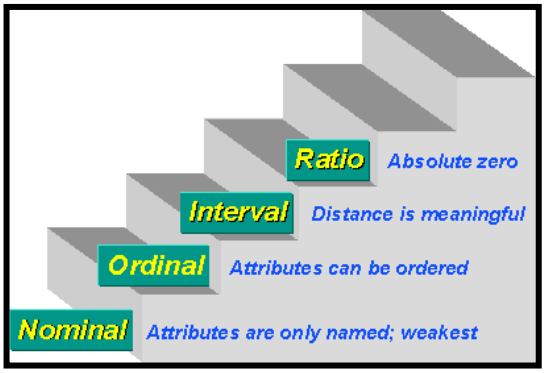
1. Descriptive Statistics

Descriptive statistics are used to describe the basic features of the data in a study. They provide simple summaries about the sample and the measures. Together with simple graphics analysis, they form the basis of virtually every quantitative analysis of data. Descriptive Statistics are used to present quantitative descriptions in a manageable form. In a research study, there may be lots of measures or we may measure a large number of people on any measure. Descriptive statistics help us to simplify large amounts of data in a sensible way. Each descriptive statistic reduces lots of data into a simpler summary. There are two basic methods: numerical and graphical. Using the numerical approach one might compute statistics such as the mean and standard deviation. These statistics convey information about the average degree of shyness and the degree to which people differ in shyness. Graphical methods are better suited than numerical methods for identifying patterns in the data. Numerical approaches are more precise and objective. Since the numerical and graphical approaches complement each other, it is wise to use both.

The raw data consist of measurements of some attribute on a collection of individuals. The measurement would have been made in one of the following scales *viz*., nominal, ordinal, interval or ratio scale.

2. Levels of Measurement

- **Nominal scale** refers to measurement at its weakest level when number or other symbols are used simply to classify an object, person or characteristic, *e.g.*, state of health (healthy, diseased).
- **Ordinal scale** is one wherein given a group of equivalence classes, the relation greater than holds for all pairs of classes so that a complete rank ordering of classes is possible, *e.g*., socio-economic status.
- When a scale has all the characteristics of an ordinal scale, and when in addition, the distances between any two numbers on the scale are of known size, **interval scale** is achieved, e.*g*., temperature scales like centigrade or Fahrenheit.
- An interval scale with a true zero point as its origin forms a ratio scale. In a **ratio scale**, the ratio of any two scale points is independent of the unit of measurement, e.g., height of trees.



3. Types of Descriptive Statistics

- Graphs and Frequency Distribution
  These represent the data enabling the researcher to see what the distribution of scores look like.
- Measures of Central Tendency
  These measures are the indices that enable to determine the average score of a group of scores.
- Measures of Variability
  These measures are indices that enable to indicate how spread out a group of scores are.

4. Frequency Distribution

The frequency distribution is a summary of the frequency of individual values or ranges of values for a variable. Preparation of frequency distribution is an often-used technique in statistical works when summarizing large masses of raw data, which leads to information on the pattern of occurrence of predefined classes of events.

Ungrouped Data: The simplest distribution would list every value of a variable and the number of persons who had each value.

Grouped Data: A way to summarize data is to distribute it into **classes** or **categories** and to determine the number of individuals belonging to each class, called the **class frequency**. It is easier to see patterns in the data, but there is loss of information about individual scores.

A tabular arrangement of data by classes together with the corresponding class frequencies is called a **frequency distribution** or **frequency table**. Following is the raw data on some measurements and its frequency distribution:

86	77	91	60	55
76	92	47	88	67
23	59	72	75	83
77	68	82	97	89
81	75	74	39	67
79	83	70	78	91
68	49	56	94	81

Table 1: Grouped frequency distribution

Class Interval	Frequency	Proportion	Cumulative Frequency
20-under 30	1	0.028	1
30-under 40	1	0.028	2
40-under 50	2	0.057	4
50-under 60	3	0.086	7
60-under 70	5	0.143	12
70-under 80	10	0.287	22
80-under 90	8	0.228	30

90-under 100	5	0.143	35

Following is a frequency distribution of Diameter at Breast Height (DBH) recorded to the nearest cm, of 80 teak trees in a sample plot.

Table 2: Frequency distribution of DBH of teak trees in a plot

DBH class (cm)	Frequency (Number of trees)
11 - 13	11
14 - 16	20
17 - 19	30
20 - 22	15
23 - 25	4
Total	80

5. Graphical Representation of Data

Frequency distributions are often graphically represented by a **histogram** or **frequency polygon**. A histogram consists of a set of rectangles having bases on a horizontal axis (the *x* axis) with centres at the class marks and lengths equal to the class interval sizes and areas proportional to class frequencies. If the class intervals all have equal size, the heights of the rectangles are proportional to the class frequencies and it is then customary to take the heights numerically equal to the class frequencies. If class intervals do not have equal size, these heights must be adjusted. A frequency polygon is a line graph of class frequency plotted against class mark. It can be obtained by connecting midpoints of the tops of the rectangles in the histogram.

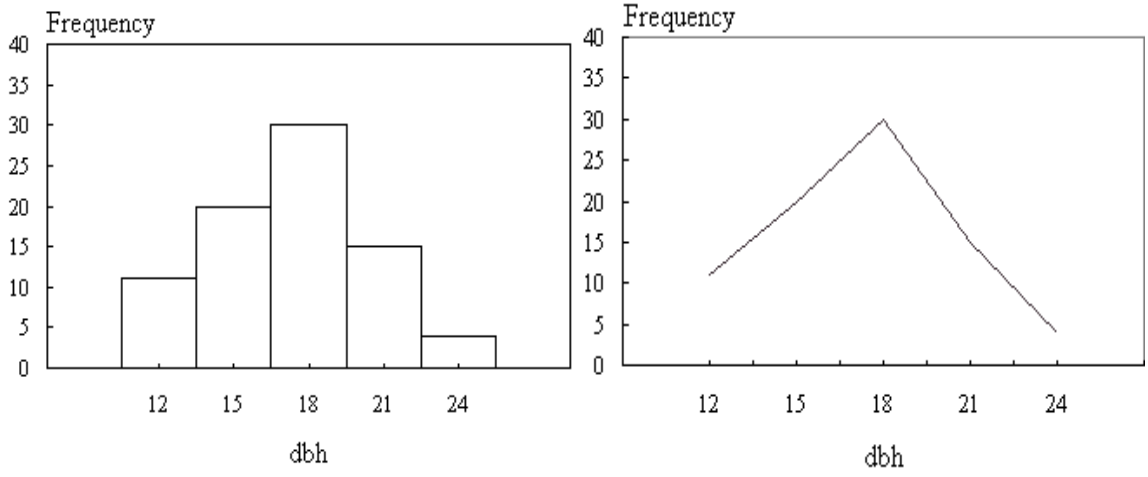


Fig. 1: Histogram and frequency curce showing the frequency distribution of DBH

The qualitative data is summarized in a frequency, relative frequency, or percent frequency distribution using **bar chart**. On the horizontal axis we specify the labels that are used for each of the classes. A frequency, relative frequency, or percent frequency scale is used for the vertical axis. Using a bar of fixed width drawn above each class label,

the height can be extended appropriately. The bars are separated to emphasize the fact that each class is a separate category.

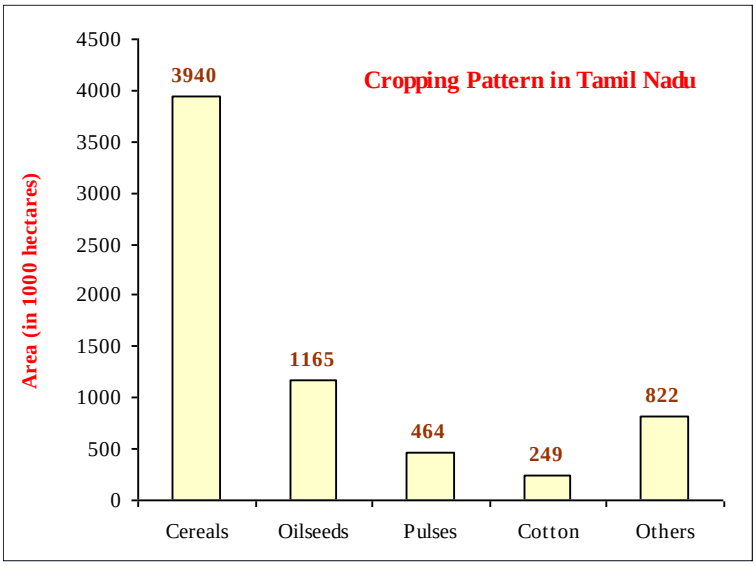


Fig. 2: Bar chart of cropping pattern

Pie chart is commonly used graphical device for presenting relative frequency distributions for qualitative data. Draw a circle; then use the **relative frequencies** to subdivide the circle into sectors that correspond to the relative frequency for each class. Since there are 360 degrees in a circle, a class with a relative frequency of .25 would consume .25(360) = 90 degrees of the circle. The above given cropping pattern is displayed in pie chart as follows:

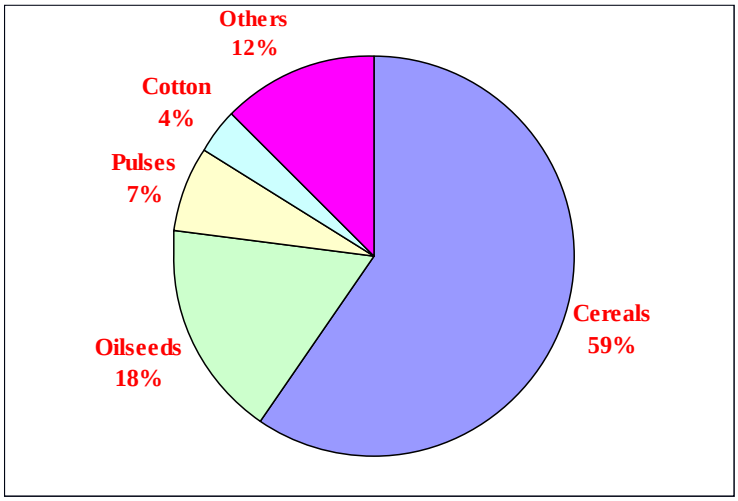


Fig. 3: Pie chart of cropping pattern

Having prepared a frequency distribution, a number of measures can be generated out of it, which leads to further condensation of the data. These are measures of location or central tendency, dispersion, skewness and kurtosis.

6. Measures of Central Tendency

The central tendency of a distribution is an estimate of the "center" of a distribution of values. There are three major types of estimates of central tendency:

- Mean
- Median
- Mode

The **Mean** or average is probably the most commonly used method of describing central tendency. To compute the mean add up all the values and divide by the number of values. The arithmetic mean $(\bar{x})$ or the mean of a set of N numbers $x_1, x_2, x_3, \ldots, x_N$ is

$$\text{Mean} = \frac{x_1 + x_2 + \ldots + x_N}{N}$$

If the numbers $x_1, x_2, \ldots, x_k$ occur $f_1, f_2, \ldots, f_k$ times respectively i.e., occur with frequencies $f_1, f_2, \ldots, f_k$, the arithmetic mean is

$$\text{Mean} = \frac{f_1 x_1 + f_2 x_2 + \ldots + f_k x_k}{f_1 + f_2 + \ldots + f_k}$$

Consider the data given in Table 2,

DBH class (cm)	Frequency (f) (Number of trees)	x	xf
11 - 13	11	12	132
14 - 16	20	15	300
17 - 19	30	18	540
20 - 22	15	21	315
23 - 25	4	24	96
Total	80	80	1383

$$\text{Mean} = \frac{1383}{80} = 17.29\,\text{cm}.$$

The **Median** is the score found at the exact middle of the set of values. One way to compute the median is to list all scores in numerical order, and then locate the score in the center of the sample. For example, let 8 scores be ordered as 15, 15, 15, 20, 20, 21, 25, 36. Score number 4 and 5 represent the halfway point. Since both of these scores are 20, the median is 20. If the two middle scores had different values, then average of two would determine the median.

For grouped data, the median is obtained using following:

$$\text{Median} = L + \left(\frac{\frac{N}{2} - (\sum f)_1}{f_m} \right) c,$$

where L is lower class limit of the median class (i.e., the class containing the median), $(\sum f)_1$ is sum of frequencies of all classes lower than the median class, f_m is the frequency of median class and c is the class interval.
Geometrically, the median is the value of x (abscissa) corresponding to that vertical line which divides a histogram into two parts having equal areas.

For Table 2,

DBH class (cm)	x	Frequency (f) (Number of trees)	Cumulative Frequency
11 - 13	12	11	11
14 - 16	15	20	31
17 - 19	18	30	61
20 - 22	21	15	76
23 - 25	24	4	80
Total	80	80	

N / 2 = 40 which falls in the class 17-19 and is thus the median class.

$$\text{Median} = 16.5 + \left(\frac{\frac{80}{2} - 31}{30} \right) 3 = 17.4 \text{ cm.}$$

The **Mode** is the most frequently occurring value in the set of scores. To determine the mode, order the scores and then count each one. The most frequently occurring value is the mode. In the example 15, 15, 15, 20, 20, 21, 25, 36, the value 15 occurs three times and is the mode. In some distributions there is more than one modal value. For instance, in a bimodal distribution there are two values that occur most frequently. The set 2, 3, 4, 4, 4, 5, 5, 7, 7, 7, 9 has two modes 4 and 7 and is called bimodal.

In case of grouped data, the mode will be the value (or values) of x corresponding to the maximum point (or points) on the curve. From a frequency distribution or histogram, the mode can be obtained from the formula,

$$\text{Mode} = L + \left(\frac{f_2}{f_1 + f_2} \right) c ,$$

where L is the lower class limit of modal class (the class containing the mode), f_1 is the frequency of the class previous to the modal class, f_2 is frequency of the class just after the modal class and c is the size of modal class.

From Table 2, the maximum frequency is 30 and hence the modal class is 17-19.

$$\text{Mode} = 16.5 + \left(\frac{15}{15 + 20} \right) 3 = 17.79 \text{ cm.}$$

Notice that for the same set of scores, we may get different values for the mean, median and mode. If the distribution is truly normal (i.e., bell-shaped), the mean, median and mode are all equal to each other. With three different measures of central tendency, how to know which one to use? The answer depends a lot on the data and what is to be communicated.

While the mean is the most frequently used measure of central tendency, it does suffer from one major drawback. Unlike other measures of central tendency, the mean can be influenced profoundly by one extreme data point (referred to as an "outlier"). The median and mode clearly do not suffer from this problem. There are certainly occasions where the mode or median might be appropriate. For qualitative and categorical data, the mode makes sense, but the mean and median do not. For example, when we are interested in knowing the typical soil type in a locality or the typical cropping pattern in a region we can use mode. On the other hand, if the data is quantitative one, we can use any one of the averages.

If the data is quantitative, then one has to consider the nature of the frequency distribution. When the frequency distribution is skewed (not symmetrical) the median or mode will be proper average. In case of raw data in which extreme values, either small or large, are present, the median or mode is the proper average. In case of a symmetrical distribution either mean or median or mode can be used. However, as seen already, the mean is preferred over the other two. The mean, median, and mode can be related (approximately) to the histogram: the mode is the highest bump, the median is where half the area is to the right and half is to the left, and the mean is where the histogram would balance.

The **Harmonic mean** H of the positive real numbers x_1, x_2, ..., x_n is defined to be

$$H = \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + ... + \frac{1}{x_n}}$$

Equivalently, the harmonic mean is the reciprocal of the arithmetic mean of the reciprocals. If a set of weights $w_1,...,w_n$ is associated to the dataset $x_1,...,x_n$, the weighted harmonic mean is defined by

$$H = \frac{\sum_{i=1}^{n} w_i}{\sum_{i=1}^{n} w_i / x_i}$$

The **geometric mean** of a data set x_1, ..., x_n is given by

$$G = (x_1 x_2 ... x_n)^{1/n}$$

When dealing with rates, speed and prices, harmonic mean may be used. If interested in relative change, as in the case of bacterial growth, cell division etc., geometric mean is the most appropriate average.

7. Measures of Dispersion

Averages are representatives of a frequency distribution but they fail to give a complete picture of the distribution. They do not tell anything about the scatterness of observations within the distribution.

Suppose that we have the distribution of the yields (kg per plot) of two paddy varieties from 5 plots each. The distribution may be as follows:

Variety I	45	42	42	41	40
Variety II	54	48	42	33	30

It can be seen that the mean yield for both varieties is 42 kg. But we can not say that the performance of the two varieties are same. There is greater uniformity of yields in the first variety whereas there is more variability in the yields of the second variety. The first variety may be preferred since it is more consistent in yield performance. From the above example, it is obvious that a measure of central tendency alone is not sufficient to describe a frequency distribution. In addition to it, a measure of **scatterness** of observations should be there. The scatterness or variation of observations from their average is called the **dispersion**. There are different measures of dispersion like the range, the quartile deviation, the mean deviation and the standard deviation.

The **Range** is simply the highest value minus the lowest value. The **Standard Deviation** (S.D) is a more accurate and detailed estimate of dispersion because an outlier can greatly exaggerate the range. The Standard Deviation shows the relation that set of scores has to the mean of the sample. The standard deviation is the square root of the sum of the squared deviations from the mean divided by the number of scores.

$$\text{Standard Deviation} = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{N}}$$

If $x_1, x_2,\ldots,x_k$ occur with frequencies $f_1, f_2,\ldots,f_k$ respectively, the standard deviation can be computed as

$$\text{Standard Deviation} = \sqrt{\frac{\sum_{i=1}^{k} f_i(x_i - \bar{x})^2}{N}}, \quad = \sqrt{\frac{\sum_{i=1}^{k} f_i x_i^2}{N} - \left(\frac{\sum_{i=1}^{k} f_i x_i}{N}\right)^2}, \quad N = \sum_{i=1}^{k} f_i$$

Consider the data given in Table 2.

DBH class (cm)	**Frequency (f) (Number of trees)**	**x**	**fx**	**fx^2**
11 - 13	11	12	132	1584
14 - 16	20	15	300	4500
17 - 19	30	18	540	9720
20 - 22	15	21	315	6615
23 - 25	4	24	96	2304
Total	80	80	1383	24723

$$\text{Standard Deviation} = \sqrt{\frac{24723}{80} - \left(\frac{1383}{80}\right)^2} = 3.19 \text{ cm.}$$

The **variance** of a set of data is defined as the square of the standard deviation. **Mean deviation** is the mean of the deviations of individual values from their average. The average may be either mean or median. For raw data the mean deviation from the median is the least.

Measures of Relative Dispersion

Suppose that the two distributions to be compared are expressed in the same units and their means are equal or nearly equal. Then their variability can be compared directly by using their standard deviations. However, if their means are widely different or if they are expressed in different units of measurement, we can not use the standard deviations as such for comparing their variability. We have to use the relative measures of dispersion in such situations.

There are relative dispersions in relation to range, the quartile deviation, the mean deviation, and the standard deviation. Of these, the coefficient of variation which is related to the standard deviation is important. The ratio of standard deviation (S.D) to mean expressed in percentage is called **coefficient of variation**,

C.V. = (S.D. / Mean) x 100

The C.V. is a unit-free measure. It is always expressed as percentage. The C.V. will be small if the variation is small. Of the two groups, the one with less C.V. is said to be more consistent.

The coefficient of variation is unreliable if the mean is near zero. Also it is unstable if the measurement scale used is not ratio scale. The C.V. is informative if it is given along with the mean and standard deviation. Otherwise, it may be misleading.

Suppose that the variation in height of seedlings and that of older trees of a species are to be compared. Let the mean height of seedlings be 50 cm and standard deviation of height of seedlings be 10 cm. Further let the mean height of trees be 500 cm with standard deviation of height of seedlings as 100 cm. By the absolute value of the standard deviation, one may tend to judge that variation is more in the case of trees but the relative variation, as indicated by the coefficient of variation (20%), is the same in both the sets.

Consider the measurements on yield and plant height of a paddy variety. The mean and standard deviation for yield are 50 kg and 10 kg respectively. The mean and standard deviation for plant height are 55 cm and 5 cm, respectively.

Here the measurements for yield and plant height are in different units. Hence, the variability can be compared only by using coefficient of variation. For yield,

C.V. = (10 / 50) x 100 = 20 %

For plant height,

C.V. = (5 / 55) x 100 = 9.1 %

The yield is subject to more variation than the plant height.

8. Shape of the Distribution

An important aspect of the "description" of a variable is the shape of its distribution, which tells the frequency of values from different ranges of the variable. A researcher is interested in how well the distribution can be approximated by the normal distribution Simple descriptive statistics can provide some information relevant to this issue. For example, if the **skewness** (which measures the deviation of the distribution from symmetry) is clearly different from 0, then that distribution is **asymmetrical**, while normal distributions are perfectly symmetrical. If the **kurtosis** (which measures "peakedness" of the distribution) is clearly different from 0, then the distribution is either flatter or more peaked than normal; the kurtosis of the normal distribution is 0.

Skewness is the degree of asymmetry, or departure from symmetry, of a distribution. If the frequency curve (smoothed frequency polygon) of a distribution has a longer 'tail' to the right of the central maximum than to the left, the distribution is said to be skewed to the right or to have positive skewness. If the reverse is true, it is said to be skewed to the left or to have negative skewness. An important measure of skewness expressed in dimensionless form is given by

$$\text{Cefficient of skewness} = \beta_1 = \frac{\mu_3^2}{\mu_2^3},$$

where μ_2 and μ_3 are the second and third central moments defined using the formula,

$$\mu_r = \frac{\sum_{i=1}^{N} (x_i - \bar{x})^r}{N}.$$

For grouped data, the above moments are given by

$$\mu_r = \frac{\sum_{i=1}^{N} f_i (x_i - \bar{x})^r}{N}.$$

For a symmetrical distribution, $\beta_1 = 0$. Skewness is positive or negative depending upon whether β_1 is positive or negative.

Kurtosis is the degree of peakedness of a distribution, usually taken relative to a normal distribution. A distribution having a relatively high peak is called leptokurtic, while the curve which is flat-topped is called platykurtic. A bell shaped curve which is not very peaked or very flat-topped is called mesokurtic. The measure of kurtosis, expressed in dimensionless form, is given by

$$\text{Cefficient of kurtosis} = \beta_2 = \frac{\mu_4}{\mu_2^2},$$

where μ_4 and μ_2 can be obtained from the formula as given above. The distribution is called normal if $\beta_2 = 3$. When β_2 is more than 3, the distribution is said to be leptokurtic. If β_2 is less than 3, the distribution is said to be platykurtic.

EXPLORATORY DATA ANALYSIS

Cini Varghese and Anindita Datta
I.A.S.R.I., Library Avenue, New Delhi – 110 012
cini.varghese@icar.gov.in;anindita.datta@icar.gov.in

Exploratory Data Analysis (EDA) is an attitude/philosophy about how a data analysis should be carried out. EDA heavily uses the collection of techniques that are usually called "statistical graphics", but it is not identical to statistical graphics. EDA is a philosophy as to how we dissect a data set; what we look for; how we look; and how we interpret. Simply put, in EDA we "visualize our data first", then talk about models / inferences, *etc*. A surprisingly small percentage of data analysts use EDA. Of the ones who do are often surprised by how this seemingly "unnecessary step" may profoundly change the way the data is modeled. Thus, EDA is an approach for data analysis that employs a variety of techniques to

- maximize insight into a data set,
- uncover underlying structure,
- extract important variables,
- detect outliers,
- test underlying assumptions,
- develop parsimonious models and
- determine optimal factor settings.

For classical analysis, the sequence of procedures is

Problem $\rightarrow$ Data $\rightarrow$ Model $\rightarrow$ Analysis $\rightarrow$ Conclusions

For EDA, the sequence is

Problem $\rightarrow$ Data $\rightarrow$ Analysis $\rightarrow$ Model $\rightarrow$ Conclusions

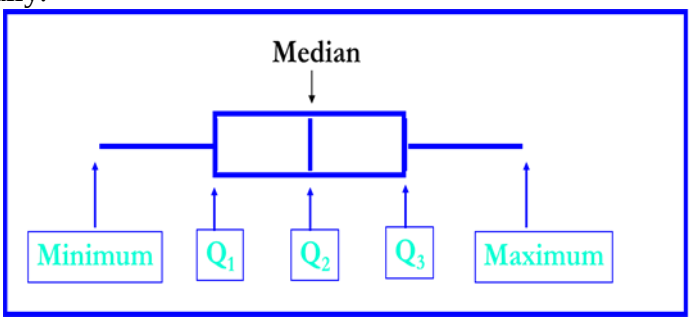
Many EDA techniques make little or no assumptions. They present and show the data, all of the data as is, with fewer encumbering assumptions. The techniques of EDA consist of simple arithmetic and easy-to-draw pictures that can be used to summarize data quickly. One such technique is Box-Plot.

Box Plot

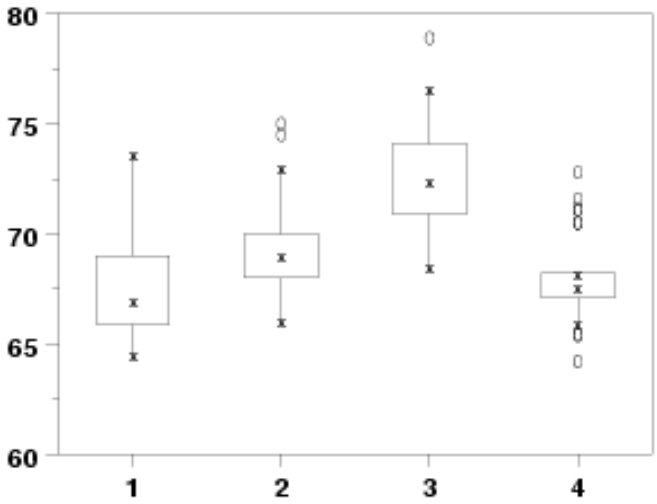
A box-plot is a graph of the five–number summary. Five-number system gives a good identification of center and spread of the data:

- Maximum
- Minimum
- Median = 50th percentile
- Lower quartile $Q_1 = 25^{th}$ percentile
- Upper quartile $Q_3 = 75^{th}$ percentile

In a box-plot, a central box spans the quartiles. A line in the box marks the median. Lines extend from the box out to the smallest and largest observations. Box-plots can be drawn either horizontally or vertically.



A box-plot may also indicate which observations, if any, might be considered outliers. Box- plots are an excellent tool f
or conveying location and variation information in data sets, particularly for detecting and illustrating location and variation changes between different groups of data.



The box plot can provide answers to the following questions:

- Is a factor significant?
- Does the location differ between groups?
- Does the variation differ between groups?
- Are there any outliers?

Probability Plot

The probability plot is a graphical technique for assessing whether or not a data set follows a given distribution such as the normal. The data are plotted against a theoretical distribution in such a way that the points should form approximately a straight line. Departures from this straight line indicate departures from the specified distribution. The correlation coefficient associated with the linear fit to the data in the probability plot is a measure of the goodness of the fit. Estimates of the location and scale parameters of the distribution are given by the intercept and slope. Probability plots can be generated for several competing distributions to see which

provides the best fit, and the probability plot generating the highest correlation coefficient is the best choice since it generates the straightest probability plot.

Histogram

The histogram can be used to answer the following questions:

- What kind of population distribution do the data come from?
- Where are the data located?
- How spread out are the data?
- Are the data symmetric or skewed?
- Are there outliers in the data?

Another method of displaying a set of data is with a stem-and-leaf plot. A stem-and-leaf plot is a display that organizes data to show its shape and distribution.

Stem-and-Leaf Display

A stem-and-leaf plot does resemble a histogram turned sideways. The stem values could represent the intervals of a histogram, and the leaf values could represent the frequency for each interval. The first digits of each data item are arranged to the left of a vertical line. To the right of the vertical line we record the last digit for each item in rank order. Each line in the display is referred to as a stem. Each digit on a stem is a leaf. One advantage to the stem-and-leaf plot over the histogram is that the stem-and-leaf plot displays not only the frequency for each interval, but also displays all of the individual values within that interval.

Example 1

If we have the following data:

8.6 11.7 9.4 9.1 10.2 11.0 8.8

A stem-and-leaf display of these data will be

Stem		Leaf
8	|	6 8
9	|	1 4
10	|	2
11	|	0 7

Here, leaf unit = 0.1. Leaf units may be 100, 10, 1, 0.1, and so on. Where the leaf unit is not shown, it is assumed to be = 1.

Example 2

Data:

44 46 47 49 63 64 66 68 68 72 72 75 76 81 84 88 106

Stem-and-Leaf Display:

Stem		Leaf				
4	|	4	6	7	9	
5	|					
6	|	3	4	6	8	8
7	|	2	2	5	6	
8	|	1	4	8		

9	|	
10	|	6

Key: 6 | 3 = 63, Leaf unit: 1.0, Stem unit: 1.0

Example 3

Data: 1806 1717 1974 1791 1682 1910 1838

Stem-and-Leaf Display:

Stem		**Leaf**
16	|	8
17	|	1 9
18	|	0 3
19	|	1 7

Here, the leaf unit = 10.

Example 4

The manager of an Auto company would like to get a better picture of the distribution of costs for engine tune-up parts. A sample of 50 customer invoices has been taken and the costs of parts, rounded to the nearest dollar, are listed below:

91	78	93	57	75	52	99	80	97	62
71	69	72	89	66	75	79	75	72	76
104	74	62	68	97	105	77	65	80	109
85	97	88	68	83	68	71	69	67	74
62	82	98	101	79	105	79	69	62	73

Stem-and-Leaf Display:

Stem		**Leaf**
5	|	2 7
6	|	2 2 2 2 5 6 7 8 8 8 9 9 9
7	|	1 1 2 2 3 4 4 5 5 5 6 7 8 9 9 9
8	|	0 0 2 3 5 8 9
9	|	1 3 7 7 7 8 9
10	|	1 4 5 5 9

Stretched Stem-and-Leaf Display

If we believe the original stem-and-leaf display has condensed the data too much, we can stretch the display by using two more stems for each leading digit(s). Whenever a stem value is stated twice, the first value corresponds to leaf values of 0-4, and the second values correspond to values of 5-9.

Example 5

If we consider data of Example 4, stretched stem-and-leaf display can be presented:

Stem		**Leaf**
5	|	2
5	|	7
6	|	2 2 2 2

6	|	5 6 7 8 8 8 9 9 9
7	|	1 1 2 2 3 4 4
7	|	5 5 5 6 7 8 9 9 9
8	|	0 0 2 3
8	|	5 8 9
9	|	1 3
9	|	7 7 7 8 9
10	|	1 4
10	|	5 5 9

Crosstabulation and Scatter Diagram

Often a manager is interested in tabular and graphical methods that will help understand the relationship between two variables. Crosstabulation and a scatter diagram are two methods for summarizing the data for two (or more) variables simultaneously.

Crosstabulation

Crosstabulation is a tabular method for summarizing the data for two variables simultaneously. Crosstabulation can be used when:

- One variable is qualitative and the other is quantitative
- Both variables are qualitative
- Both variables are quantitative

The left and top margin labels define the classes for the two variables.

Example 6

The number of houses sold for each style and price for two years is shown below:

Price Range	**Home Style**				
	Colonial	Ranch	Split	A-Frame	Total
$\leq$ $99,000	18	6	19	12	55
> $99,000	12	14	16	3	45
Total	30	20	35	15	100

From the preceding crosstabulation, it can be concluded that the greatest number of houses in the sample (19) are a split-level style and priced at less than or equal to $99,000. Only three homes in the sample are an A-Frame style and priced at more than $99,000.

Crosstabulation: Row or Column Percentages

Converting the entries in the table into row percentages or column percentages can provide additional insight about the relationship between the two variables.

Row Percentages

Price	**Home Style**				
Range	Colonial	Ranch	Split	A-Frame	Total

≤ $99,000	32.73	10.91	34.55	21.82	100
> $99,000	26.67	31.11	35.56	6.67	100

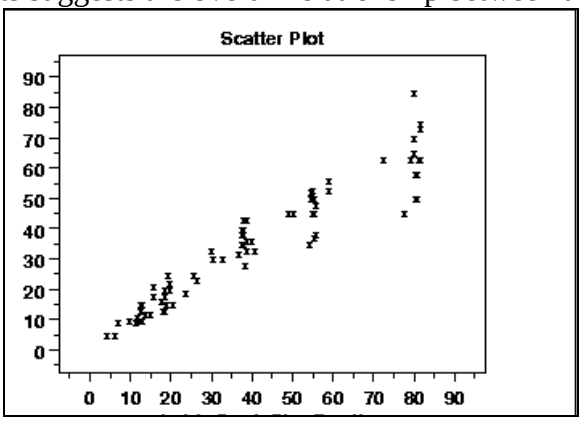
Note: row totals are actually 100.01 due to rounding.

Column Percentages

Price Range	**Home Style** Colonial	Ranch	Split	A-Frame
≤ $99,000	60.00	30.00	54.29	80.00
> $99,000	40.00	70.00	45.71	20.00
Total	100	100	100	100

Scatter Plot

It is a graphical presentation of the relationship between two quantitative variables. One variable is shown on the horizontal axis and the other variable is shown on the vertical axis. The general pattern of the plotted points suggests the overall relationship between the variables.



This above scatter plot reveals a linear relationship between the two variables indicating that a linear regression model might be appropriate. Scatter plots can provide answers to the following questions:

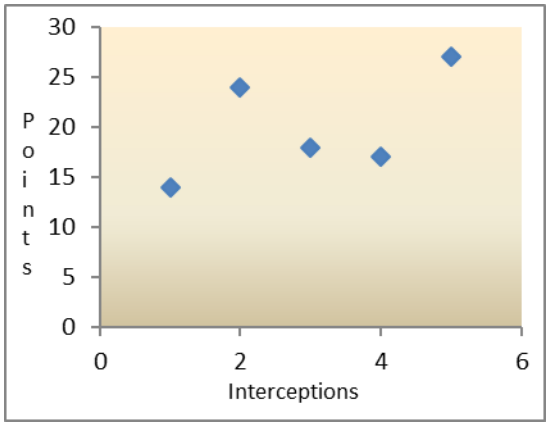
- Are variables X and Y related?
- Are variables X and Y linearly related?
- Are variables X and Y non-linearly related?
- Does the variation in Y change depending on X?
- Are there outliers?

Example 7

A football team is interested in investigating the relationship, if any, between interceptions made and points scored.

x = Number of Interceptions	*y* = Number of Points Scored
1	14

3	24
2	18
1	17
3	27



The preceding scatter diagram indicates a positive relationship between the number of interceptions and the number of points scored. Higher points scored are associated with a higher number of interceptions. The relationship is not perfect; all plotted points in the scatter diagram are not on a straight line.

CORRELATION AND REGRESSION

B.N. Mandal
ICAR-IASRI, Library Avenue, New Delhi – 110 012
bn.mandal@icar.gov.in

1. Introduction

In order to study the relationship between two or more variables through correlation and or regression, it is important to visualize the relation between them graphically. Scatter Plot is the simplest way of the diagrammatic representation of a bivariate data. It gives the idea of the distribution of the data like well defined positive or negative linear relationships, non-linear relationships or no apparent relationship.

2. Scatter Plots

It is the simplest way of the diagrammatic representation of a bivariate data. It gives the idea of the distribution of the data like well defined positive or negative linear relationships, non-linear relationships or no apparent relationship. The chart can be created using the Graph menu. **To Obtain Scatterplots**: from the menus, choose: Graphs→ Scatter. SPSS 10.0 gives three types of scatter plots viz. *simple, overlay, matrix, or 3-D*. For getting the desired scatter plot click the icon and then Select Define → Select variables and options for the chart.

To Obtain Simple Scatterplots: From the menus, choose: Graphs→ Legacy dialogs → Scatter/dot → Select the icon for Simple → Select Define → select a variable for the Y-axis and a variable for the X-axis. (Caution: These variables must be numeric, but should not be in date format). → if desired, select a variable and move it into *the Set Markers by* box. Each value of this variable is marked by a different symbol on the scatterplot. This variable may be numeric or string. → If desied, one can select a numeric or a string variable and move it into the *Label Cases* by box. You can label points on the plot with this variable.

- If selected, the value labels (or values if no labels are defined) of this variable are used as point labels.
- If we do not select a variable to label Cases by, case numbers can be used to label outliers and extremes.
- Select *Options* to specify the treatment of missing values in the data and control whether labels are to be displayed for points on the plot.
- Select *Titles* to define lines of text to be placed at the top or bottom of the plot.

To Obtain Overlay Scatterplots: This option is used to obtain plots for two or more variable pairs.
Select Graphs→ Legacy dialogs → Scatter→ Select the icon for Overlay→ Select Define → Select at least two pairs of variables, Select each variable separately. The first variable will appear in the Current Selections list box as Variable 1, and the second variable will appear as Variable 2. To deselect a variable, select it again in the source variable list. Once a pair of variables is selected, move the pair into the Y-X box (Caution: Variables should be numeric,

but should not be in date format.). As in case of simple scatter plots, select a numeric or a string variable and move it into the Label Cases by box. Points on the plot are labeled with the selected variable.

To reverse the order of the Y and X variables within a selected pair, select Swap Pair. For the specification of the treatment of the missing values and case labels display and for titles, follow the steps as in simple scatter plot.

To Obtain a Scatterplot Matrix: This option plots all possible combinations of two or more numeric variables against one another. For obtaining a Scatterplot matrix, select the icon for Matrix → Select Define → Select at least two Matrix numeric Variables. Rest options are similar to the earlier ones.

To Obtain 3-D Scatterplots: This option plots three numeric variables in three dimensions. Select the icon for 3-D→ Select Define → Select one variable for the Y-axis, one for the X-axis and one for the Z-axis. These variables must be numeric, but should not be in date format.

3. Bivariate and Partial Correlation

A correlation coefficient measures the strength of a linear association between two quantitative variables. The most commonly used measure of linear correlation between two variables is called the *Pearson-product- moment correlation coefficient* or simply the *sample correlation coefficient* and is denoted by *r*. The values of the correlation coefficient is not expressed in units of the data, but range from –1 to +1. While scatterplot provide a picture of the relation, the value of the correlation is the same if you switch the *Y* (vertical) and *X* (horizontal) measures. The sample correlation coefficient *r* is estimated by the formula

$$r = \frac{Cov(X,Y)}{\sqrt{Var(X)Var(Y)}}$$

For a sample of size *n*, the above expression can be written as

$$r = \sum_{i=1}^{n} \frac{(x_i - \bar{x})(y_i - \bar{y})}{(n-1)s_x s_y}$$

where s_x and s_y are the sample standard deviations of the two variables. The formula can be simplified to

$$r = \frac{n\sum_{i=1}^{n} x_i y_i - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)}{\sqrt{\left[n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2\right]\left[n\sum_{i=1}^{n} y_i^2 - \left(\sum_{i=1}^{n} y_i\right)^2\right]}}.$$

Test of significance of correlation coefficient

Case I: Let the population correlation coefficient of X and Y is denoted by ρ, then it is often of interest to test whether ρ is zero or different from zero, on the basis of observed correlation coefficient, r. Thus, if r is the sample correlation coefficient based on a sample of n observations, then the appropriate test procedure for testing the null hypothesis $H_0 : \rho = 0$ against the alternative $H_1 : \rho \neq 0$ is:

1. Compute the quantity $t = \dfrac{r\sqrt{n-2}}{\sqrt{1-r^2}}$.
2. Compare the computed value of $|t|$, with the table value of *t*-distribution with $(n-2)$ degrees of freedom, and at a given level of significance, say 5 % .
3. If the computed value of $|t|$ exceeds the table value (as in (ii) above), then $H_0 : \rho = 0$ is rejected against the alternative $H_1 : \rho \neq 0$.

Case II: One may be interested in testing $H_0 : \rho = \rho_0$ against the alternative $H_1 : \rho \neq \rho_0$. This sample correlation coefficient based on *n* pairs of observations is based on the following quantity

$$\frac{1}{2}\log_e\left(\frac{1+r}{1-r}\right)$$

which is a value of a random variable that follows approximately the normal distribution with mean $\dfrac{1}{2}\log_e\left(\dfrac{1+\rho}{1-\rho}\right)$ and variance $1/(n-3)$. Thus the test procedure is to compute

$$Z = \frac{\sqrt{n-3}}{2}\left(\log_e\left(\frac{1+r}{1-r}\right) - \log_e\left(\frac{1+\rho_0}{1-\rho_0}\right)\right)$$
$$= \frac{\sqrt{n-3}}{2}\log_e\left[\frac{(1+r)(1-\rho_0)}{(1-r)(1+\rho_0}\right]$$

and compare to the critical points of the standard normal distribution. For example, if the absolute value of Z, $|Z| > 1.96$, then the null hypothesis $H_0 : \rho = \rho_0$ against the alternative $H_1 : \rho \neq \rho_0$ is rejected at 5% level of significance. The alternative hypotheses $\rho < \rho_0$ or $\rho > \rho_0$ can also be tested using one tailed critical points.

Rank Correlation

In some cases, it is not possible to measure the data and only ranking is done. In such situations, the rank correlation is worked out which is nothing but the Pearson's Product moment correlation coefficient and is defined as the correlation between ranks of individuals with respect to two characters. This is also known as *Spearman's Rank correlation coefficient*

and lies between –1 and +1. If d_i denotes the difference between the ranks of ith individual and *n* denotes the number of individuals, then the Spearman's Rank Correlation Coefficient is given by

$$r = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)}.$$

If there is a tie in the ranks, then the ranks assigned is the average of the ranks assigned to these individuals had there been no tie. In case of ties, the rank correlation coefficient is given by

$r = 1 - \dfrac{6\left(\sum_{i=1}^{n} d_i^2 + \sum (m^3 - m)/12\right)}{n(n^2-1)}$, where *m* is the number of individuals having the same rank.

If in a group the data on more than two variables is collected and one is interested in obtaining the measure of linear association between all pairs of variables, then one can obtain the sample correlation coefficient for all possible pairs of variables. The probability of significance of each of these correlation coefficients can be obtained using any standard statistical software. However, if one scans the results for more than one pair of variables, the probabilities of significance are *pseudo probabilities* because they are designed to test one and only one correlation for significance and do not reflect the number of correlations tested. As a result some of the correlations may appear significant when they are not. The Bonferroni method may be used to adjust the stated significance levels. In this method, we divide the desired level of significance by *m* the number of correlation coefficients and if the probability is less than or equal to this ratio, then the correlation coefficient is significant at that level of significance. Alternatively, we multiply the probabilities of significance by *m* the number of correlation coefficients and if the probability is less than or equal to the desired level of significance, then the correlation coefficient is significant at that level of significance.

In the situations, when the number of observations or each pair of variables is not constant, one has to be cautious in scanning these *m* values to get a sense of the size of one correlation relative to another.

Partial Correlation

Sometimes the correlation between two variables Y and X_1 may be partly due to the correlation of a third variable, X_2 with both Y and X_1. The true correlation between Y and X_1 can only be observed once the effect of X_2 has been eliminated. We accomplish this by means of the *sample partial correlation coefficient*. Thus, partial correlation measures the linear association between two variables after the effects of one or more variables are removed. Partial correlation can reveal variables that enhance or suppress the relation between two particular variables. For example, if each Sunday for a year, one counts the number of

ants in the kitchen at a beach cabin and the number of cars passing the house in a five-minute interval, the correlation may be close to 1. Are the cars bringing the ants? Does this sound silly? A third variable, temperature is ignored. When the weather is hot, the ants flourish and lots of people flock to the beach; when it is cool, the numbers of both the cars and ants diminish. If the linear effect of temperature is controlled, the relationship between ants and cars disappears.

The *partial correlation* of variables Y and X_1 after removing the effect of variable X_2 (or "controlling" for variable X_2) is estimated as follows:

- Regress variable Y on X_2.
- Regress variable X_1 on X_2.
- For each case, compute the residuals for each of the regression equations.
- Compute the usual Pearson correlation between the two sets of residuals.

The residuals represent variables Y and X_1 with the effect of variable X_2 removed. The partial correlation coefficient between Y and X_1 after eliminating the effect of variable X_2 is denoted by the symbol $r_{Y1.2}$. If we write the ordinary correlation coefficients for Y and X_1, Y and X_2, and X_1 and X_2 as r_{Y1}, r_{Y2}, and r_{12}, respectively, the sample partial correlation coefficient for Y and X_1 with X_2, held fixed is given by the following definition.

Partial Correlation Coefficient: The measure of linear relationship between the variable Y and X_1 after making allowance for their association with X_2, is estimated by the sample partial correlation coefficient $r_{y1.2}$, where

$$r_{Y1.2} = \frac{r_{Y1} - r_{Y2}r_{12}}{\sqrt{\left(1-r_{Y2}^2\right)\left(1-r_{12}^2\right)}}.$$

A similar definition applies to $r_{Y2.1}$ which measures the correlation between Y and X_2 after eliminating the linear effect of X_1.

The partial correlation coefficients obtained after removing the effect of one variable as discussed above are called partial correlation coefficients of order one. In some situations, however, we may have to obtain the partial correlation coefficients after eliminating the effects of two or more variables. The number of variables that are used for eliminating the effects is known as the order of the sample partial correlation coefficient.

Test of Significance of Partial Correlation Coefficient

To test $H_0 : \rho_{ij.12\ldots} = 0$ against $H_1 : \rho_{ij.12\ldots} \neq 0$ compute

$$t = \frac{r_{ij.12\ldots}}{\sqrt{1-r^2_{ij.12\ldots}}}\sqrt{n-\theta-2}$$

where θ is the order of the coefficient. This statistic follows t-distribution with $n-\theta-2$ degrees of freedom. Reject H_0 if $|t| > t_{\alpha/2,n-\theta-2}$.

Steps to obtain bivariate correlations using SPSS

From the menus choose: Analyze → Correlate → Bivariate → Select two or more numeric variables. The following options are also available:

Steps to obtain partial correlations using SPSS

From the menus choose: Analyze → Correlate → Partial → Select two or more numeric variables for which partial correlation are to be computed → Select one or more numeric control variables.

One can compute correlation coefficient by using Correl function in MS-EXCEL as: CORREL(array1,array2), where Array1 is a cell range of values and Array2 is a second cell range of values. One can also obtain bivariate correlations by using Tools → Data Analysis→ Correlation and then choosing the input and output range. For testing of significance or working out the exact probability level of significance one may use the following:

Probability level of significance can be obtained by TDIST(x,degrees_freedom,tails), x is the numeric value at which to evaluate the distribution, Degrees_freedom is an integer indicating the number of degrees of freedom and Tails specifies the number of distribution tails to return. If tails = 1, TDIST returns the one-tailed distribution. If tails = 2, TDIST returns the two-tailed distribution.

Alternatively, we can get the t-value of the Student's t-distribution as a function of the probability and the degrees of freedom by using TINV (probability, degrees_freedom). Here, probability is the probability associated with the two-tailed Student's t-distribution.

3. Regression

In many statistical studies, the goal is to establish a relationship, expressed via an equation, for predicting typical values of one variable given the value of another variable(s). In such situations, regression analysis can be of help to us. The term *regression* is derived from the original heredity studies made by Sir Francis Galton (1822-1911) in which he compared the heights of sons to the heights of fathers. Galton showed that the heights of the sons of tall fathers over successive generations regressed towards the mean height of the population. In other words, sons of usually tall fathers tend to be shorter than their fathers and sons of usually short fathers tend to be taller than their fathers. Now-a-days, the term regression is most of the prediction problems and does not necessarily imply a regression towards the population mean. In this section, we deal with the problem of estimating or predicting the value of a dependent variable given a set of independent variables. We begin with the case of single independent variable.

Simple Linear Regression

Let the variation in response variable (y) is explained by independent variable (x) called *regressor*. Simple regression of y on x or equation of a straight line as a statistical model, add a term for random error (ε) because the points do not fall on the line:

$$y = \beta_0 + \beta_1 x + \varepsilon$$

The *slope* (β_1) is the ratio between the vertical change and the horizontal change along the line. A test $\beta_1 = 0$ is same as that of a test that correlation coefficient (*r)* is zero as $r = \hat{\beta}_1 s_x / s_y$.

The *intercept* (β_0 or *constant* as it is often called) is where the line intercepts the vertical axis at $x = 0$.

To represent the errors (ε) in the model, draw a short vertical line from each point to the line. The lengths of these line segments between the line and the plot points are called *residuals* and are estimates for the true errors.

In the above equation, y is the *dependent* or *outcome* or *predicted* variable, the one you are trying to predict; x is the *independent* or *predictor* variable; and the intercept (β_0) and slope (β_1) are *coefficients*. If the model is a good descriptor of the relation between the variables, one can use the estimates of the coefficients to predict the value of the dependent variable for new cases.

Fitting of simple regression

Suppose n observations are made on y and x. Then, for each observation we have unobserved error term ε_i. We make the following assumptions regarding the $\varepsilon_i's$, which are random variables (i) errors are independent (ii) errors have zero mean and constant variance $\left(\sigma^2\right)$. These assumptions can also be written as

$$E(\varepsilon_i) = 0, \; Var(\varepsilon_i) = \sigma^2 \quad \text{for all } i = 1,2,\cdots,n.$$
$$Cov(\varepsilon_i, \varepsilon_{i'}) = 0 \qquad \text{for all } i \neq i' = 1,2,\ldots,n$$

Estimation of Parameters

The method of least squares for estimating the parameters β_0, β_1 as also σ^2, requires the minimization of the error sum of squares, *i.e.*, the sum of the squares of the vertical line segments, given by

$$S = \sum_{i=1}^{n} \varepsilon_i^2 = \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_i\right)^2$$

Differentiating S w.r.t. β_0 and β_1, and equating the derivatives to zero, we get a set of two equations as

$$\sum_{i=1}^{n} y_i = n\beta_0 + \beta_1 \sum_{i=1}^{n} x_i$$

$$\sum_{i=1}^{n} x_i y_i = \beta_0 \sum_{i=1}^{n} x_i + \beta_1 \sum_{i=1}^{n} x_i^2$$

These equations are called normal equations. The solution of these equations gives us the least squares estimates of β_0 and β_1 as b_0 and b_1

$$b_1 = Sxy / Sxx$$

$$b_0 = \bar{y} - b_1 \bar{x}$$

where $S_{xy} = \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) = \dfrac{n\sum_{i=1}^{n} x_i y_i - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)}{n}$

and $S_{xx} = \sum_{i=1}^{n} (x_i - \bar{x})^2 = \dfrac{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2}{n}$. Further, let $s_{xy} = S_{xy}/(n-1)$ and $s_{xx} = S_{xx}/(n-1)$.

It should be noted that these estimates do minimize the error sum of squares, S. The fitted regression equation is thus

$$\hat{y} = b_0 + b_1 x$$

or $$\hat{y} = \bar{y} + b_1 (x - \bar{x})$$

The 'hat' over y indicates that if were substitute for x a value that is within the observed range of the predictor x, but has not necessarily been observed, then the regression equation gives us the predicated y for that given value of x. Note that if we set $x = \bar{x}$ in the fitted regression equation, then $y = \bar{y}$, meaning thereby that the point $(\bar{x}, \bar{y})$ lies on the regression line.

Estimation of σ^2

In addition to estimating β_0 and β_1, an estimate of σ^2 is required to test hypotheses and construct interval estimates pertinent to the regression model. Ideally, we would like this estimate not to depend on the adequacy of the fitted model. This is only possible when there are several observations on y for at least one value of x, or when prior information concerning σ^2 is available. When this approach cannot be used, the estimate of σ^2 is obtained from the residual or error sum of squares.

$$SSE = \sum_{i=1}^{n} \hat{\varepsilon}_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

A convenient computing formula for SSE may be found by substituting $\hat{y} = b_0 + b_1 x_1$ and simplifying, yielding

$$SSE = \sum_{i=1}^{n} y_i^2 - n\bar{y}^2 - b_1 S_{xy} = S_{yy} - b_1 S_{xy}$$

The residual sum of square has $(n-2)$ degrees of freedom, two degrees are associated with the estimates b_0 and b_1, involved in obtaining $\hat{y}_i$. Now the expected value of SSE is $E(SSE) = (n-2)\sigma^2$ so an unbiased estimator of σ^2 is

$$\hat{\sigma}^2 = s^2 = \frac{SSE}{n-2} = MSE$$

The quantity MSE is called the error mean square or the residual mean square. The square root of s^2 is sometimes called the standard error of regression, and it has the same units as the response variable y. Because σ^2 depends on the residual sum of squares, any violation of the assumption on the model errors or any misspecification of the model form may seriously damage the usefulness of s^2 as an estimate of σ^2.

The above splitting of the total sum of squares due to $y's$ into two components can be formally put in an Analysis of variance table, as below:

Analysis of Variance: Simple Linear Regression

Source of Variation	**d.f.**	**S.S.**	**M.S.**
Regression	1	$b_1 S_{xy}$	MSR
Deviation form Regression (Residual)	$n-2$	SSE	$s^2 = SSE/(n-2) = MSE$
Total (corrected mean)	$n-1$	S_{yy}	

We have partitioned total variation in y in two parts as variations due to regression and deviation from regression. The test statistic is

$$F_0 = \frac{MSR}{MSE}$$

F_0 follows F_1, n_{-2} distribution of $H_0: \beta_1 = 0$ if $F_0 > F_{\alpha,1,} n_{-2}$.

We have seen above that $SSE = \sum_{i=1}^{n} y_i^2 - n\bar{y}^2 - b_1 S_{xy} = S_{yy} - b_1 S_{xy}$. It can further be simplified to

$$SSE = (n-1)(s_{yy} - b_1 s_{xy})$$

Now dividing both sides by $(n-1)s_{yy}$, we obtain

$$\frac{SSE}{(n-1)s_{yy}} = 1 - \frac{b_1 s_{xy}}{s_{yy}}$$

$$r^2 = 1 - \frac{SSE}{(n-1)s_{yy}}.$$

From the above, it can be concluded that r^2 measures the proportion of the total variation in the values of Y that can be accounted for or explained by the linear relationship with the values of X. Thus a correlation of $r = 0.6$ means that 0.36 or 36% of the total variation of the values of Y in our sample is accounted for by a linear relationship with the values of X. r^2 (Square of the correlation coefficient) is also known the coefficient of determination.

Remark 1: On the similar lines as above, the square of the sample partial correlation coefficient is called as sample coefficient of partial determination, which represents the ratio of the unexplained variation to the previously unexplained variation. That is $r_{Y1.2}^2$ gives us the proportion of the variation in the values of Y that was unexplained by a regression line involving only X_2 that can now be explained by including X_1 in the model along with X_2.

Precision of estimates

We derive the variances of b_1, b_0 and $\hat{y}_i$ for obtaining precision of estimates.

$$b_1 = S_{xy} / S_{xx} = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

$$Var(b_1) = \frac{\sigma^2}{S_{xx}}$$

$$Var(b_0) = Var(\bar{y} - b_1\bar{x}) = Var(\bar{y}) + \bar{x}^2 \; v(b_1)$$

$$= \frac{\sigma^2}{n} + \frac{\bar{x}^2.\sigma^2}{S_{xx}} = \left(\frac{1}{n} + \frac{\bar{x}^2}{S_{xx}}\right)\sigma^2$$

Test of Significance of $\beta's$

We are often interested in testing hypothesis about model parameters. The tests are valid, if the assumption of normality of error terms is satisfied. One may be interested in testing the hypotheses $H_0 : \beta_0 = a$ against $H_1 : \beta_0 \neq a$. The appropriate test statistic for testing this is

$$t = \frac{|b_0 - a|}{SE(b_0)} = \frac{|b_0 - a|}{\sqrt{s^2\left(\frac{1}{n} + \frac{\bar{x}^2}{S_{xx}}\right)}}$$

Which follows, a t-distribution with $(n-2)$ d.f. If H_0 is true. Reject H_0 if $t > t_{\alpha/2,n-2}$.

We may be further interested in knowing, whether x is contributing significantly towards variability in y. This can be known by testing $H_0 : \beta_1 = 0$ against $H_1 : \beta_1 \neq 0$. We use the statistic

$$t = \frac{|b_1|}{SE(b_1)} = \frac{|b_1|}{\sqrt{\frac{s^2}{S_{xx}}}}$$

Which is distributed as t with $(n-2)$ d.f. If H_0 is true. Reject H_0 if $t > t_{\alpha/2,n_{-2}}$. Alternatively, the analysis of variance can also be used for testing $H_0 : \beta_1 = 0$ against $H_1 : \beta_1 \neq 0$.

Variance of estimated mean and variance of prediction

The variance of $\hat{y}_i$ will be derived for the two situations where $\hat{y}_i$ is used as an estimate of the mean and where it is used as a prediction. Variance when $\hat{y}_i$ is used as the estimate of true mean of y at the specific value of x.

$$\hat{y}_i = b_0 + b_1 x_i = \bar{y} - b_1(x_i - \bar{x})$$

$$V(\hat{y}_i) = \frac{\sigma^2}{n} + \frac{(x_i - \bar{x})^2 \sigma^2}{S_{xx}} = \left[\frac{1}{n} + \frac{(x_i - \bar{x})^2}{S_{xx}}\right]\sigma^2$$

The variance of the fitted value attains its minimum of σ^2 / n when the regression equation is evaluated at $x_i = \bar{x}$ and increases as the value of x moves away from $\bar{x}$.

When $\hat{y}_i$ is used as a predictor for some future observation, the variance for prediction must take into account the fact that the quantity being predicated is itself a random variable. Therefore, variance for prediction, $Var(y_{i(pred)})$ is the variance of the difference between $\hat{y}_i$ and the future observation y_f

$$Var(y_{i(pred)}) = V(\hat{y}_i - y_f)$$

$$= V(\hat{y}_i) + \sigma^2$$
$$= \left[1 + \frac{1}{n} + \frac{(x_i - \bar{x})^2}{S_{xx}}\right]\sigma^2$$

Note that the variance for prediction is the variance for estimation plus the variance of the quantity being predicted.

Example 3.1:

Data of illustration

Observation No.	y	x
1	78.5	7
2	74.3	1
3	104.3	11
4	87.6	11
5	95.9	7
6	109.2	11
7	102.7	3
8	72.5	1
9	93.1	2
10	15.9	21
11	83.8	1
12	113.3	11
13	119.4	10

Model to be fitted is $y = \beta_0 + \beta_1 x + \varepsilon$

Normal equations for estimation of parameters are

$$13b_0 + 97b_1 = 1250.5$$
$$97b_0 + 1139b_1 = 10132.0$$

These can also be written as

$$\begin{bmatrix} 13 & 97 \\ 97 & 1139 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \end{bmatrix} = \begin{bmatrix} 1250.5 \\ 10132.0 \end{bmatrix}$$

Parameter estimates are

$$b_0 = 81.792, \qquad b_0 = 1.930, \qquad s^2 = 140.01$$
$$(5.437) \qquad (0.581)$$

The figures in the parenthesis denote the SE of the estimated parameter.

Fitted model is $y = 81.792 + 1.930x \qquad (r^2 = 0.501)$

Test of significance of $\beta's$

(a) $H_0 : \beta_0 = 0, \qquad H_1 : \beta_0 \neq 0,$

$$t = \frac{81.792}{5.437} = 15.043 \qquad (t_{.05'11} = 2.201)$$

(b) $H_0 : \beta_0 = 0, \qquad H_1 : \beta_0 \neq 0,$

$$t = \frac{1.93}{.581} = 3.322$$

Estimated mean response at $x = 4$

$$\hat{y}_0 = 81.792 + 1.930(4) = 89.512$$

$$\hat{V}(\hat{y}_0) = 14.829 \qquad \text{Table value of } t_{\alpha/2,11} = 2.201$$

95% confidence interval for mean response y_0 is

$$89.512 - 2.201\sqrt{14.829} \leq y_0 \leq 89.512 + 2.201\sqrt{14.829}$$

$$89.512 \leq y_0 \leq 95.988$$

If $\hat{y}_0$ is used for prediction of future observation, then

$$\hat{V}(\hat{y}_0) = 140.101 + 14.829 = 154.930$$

95% confidence interval of prediction $\hat{y}_0$ is

$89.512 \pm 2.201\sqrt{154.930}$ i.e. 62.116 to 116.108.

Test for Linearity of Regression

For any given problem we assume the regression is linear and proceed with the estimation of parameters as discussed above. This assumption is made to avoid laborious calculations. A linear regression equation is always preferred over a nonlinear regression curve if the assumption of linearity can be justified. Therefore, the linearity of regression must be tested using the following test.

Let us select a random sample of n observations using k distinct values of x, says $x_1, x_2, \cdots, x_k$, such that the sample contains n_1 observed values of the random variable y_1 corresponding to x_1, n_2 observed values of y_2 corresponding to x_2, $\cdots$, n_k observed value of y_k corresponding to x_k, $n = \sum_{i=1}^{n} n_i$. We define

y_{ij} = jth value of the random variable y_i,

$y_{i.}$ = sum of the value of y_i in our sample.

Hence, if $n_4 = 3$ measurements of y are made corresponding of $x = x_4$, we could indicate these observations by y_{41}, y_{42}, and y_{43}. Then $y_{4.} = y_{41} + y_{42} + y_{43}$. Now the computed value

$$f = \frac{\chi_1^2/(k-2)}{\chi_2^2/(n-k)},$$

where $\chi_1^2 = \sum \frac{y_{i.}^2}{n_i} - \frac{\left(\sum y_{ij}\right)^2}{n} - b^2(n-1)s_x^2$

$$\chi_2^2 = \sum y_{ij}^2 - \sum \frac{y_{i.}^2}{n_i}$$

is a value of the random variable F, having an F distribution with $k-2$ and $n-k$ degree of freedom under the null hypothesis that the relationship is linear and therefore may be used to test the hypothesis H_0 for linearity of regression.

When H_0 is true, $\chi_1^2/(k-2)$ and $\chi_2^2/(n-k)$ are independent estimates of σ^2. However, if H_0 is false, $\chi_1^2/(k-2)$ overestimates σ^2. Hence, we reject the hypothesis of linearity of regression at the α level of significance when our f value falls in a critical region of size α located in the upper tail of the F distribution.

Multiple Regression

For the situations with more than one independent variables, $X_1, X_2, \ldots, X_p$, say that are the causes of variation in *Y*, we fit multiple regression of y on $x's$ to account for this variation. Multiple regression of y on $x's$ is denoted as

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \varepsilon$$

where β_0 denotes intercept and $\beta_i's$ $(i = 1,2,\ldots,p)$ are called partial regression coefficients. ε is random error. β_i gives average change in y per unit change in x_i keeping other $x's$ constant.

Fitting of multiple regression model

Suppose n observation are made on y and $x's$. Then for each observation we have our unobserved error term ε_i. We make the following assumptions regarding the random variables $\varepsilon_i's$ same as those in simple linear regression case.

In order to estimate the unknown parameters $\beta_0, \beta_1, \beta_{2,\ldots}, \beta_p$, we use the method of least squares which requires minimization of the error sum of squares, given by

$$S = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_{1i} - \beta_2 x_{2i} - \cdots - \beta_p x_{pi}\right)^2$$

Differentiating S w.r.t. $\beta_0, \beta_1, \beta_{2,\ldots}, \beta_p$ and equating the derivatives to zero, we get a set of $p+1$ equations in $p+1$ unknown as

$$\sum_{i=1}^{n} y_i = n\beta_0 + \beta_1 \sum_{i=1}^{n} x_{1i} + \beta_2 \sum_{i=1}^{n} x_{2i} + \ldots + \beta_p \sum_{i=1}^{n} x_{pi}$$

$$\sum_{i=1}^{n} x_{1i} y_i = \beta_0 \sum_{i=1}^{n} x_{1i} + \beta_1 \sum_{i=1}^{n} x_{1i}^2 + \beta_2 \sum_{i=1}^{n} x_{1i} x_{2i} + \ldots + \beta_p \sum_{i=1}^{n} x_{1i} x_{pi}$$

$$\sum_{i=1}^{n} x_{2i} y_i = \beta_0 \sum_{i=1}^{n} x_{2i} + \beta_1 \sum_{i=1}^{n} x_{1i} x_{2i} + \beta_2 \sum_{i=1}^{n} x_{2i}^2 + \ldots + \beta_p \sum_{i=1}^{n} x_{2i} x_{pi}$$

$$\vdots$$

$$\sum_{i=1}^{n} x_{pi} y_i = \beta_0 \sum_{i=1}^{n} x_{pi} + \beta_1 \sum_{i=1}^{n} x_{1i} x_{pi} + \beta_2 \sum_{i=1}^{n} x_{2i} x_{pi} + \ldots + \beta_p \sum_{i=1}^{n} x_{pi}^2$$

These normal equations can be solved simultaneously to get $p+1$ unknowns. However, it is better to solve these equations by inverting the matrix of coefficients of right hand side as this enables us to test significance of $\beta's$ in a straightforward manner. The above equations can be written as

$$\begin{bmatrix} n & \sum x_{1i} & \sum x_{2i} & \cdots & \sum x_{pi} \\ \sum x_{1i} & \sum x_{1i}^2 & \sum x_{1i}x_{2i} & \cdots & \sum x_{1i}x_{pi} \\ \sum x_{2i} & \sum x_{1i}x_{2i} & \sum x_{2i}^2 & \cdots & \sum x_{2i}x_{pi} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \sum x_{pi} & \sum x_{1i}x_{pi} & \sum x_{1i}x_{pi} & \cdots & \sum x_{pi}^2 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum x_{1i} y_i \\ \sum x_{2i} y_i \\ \vdots \\ \sum x_{pi} y_i \end{bmatrix}$$

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_p \end{bmatrix} = \begin{bmatrix} n & \sum x_{1i} & \sum x_{2i} & \cdots & \sum x_{pi} \\ \sum x_{1i} & \sum x_{1i}^2 & \sum x_{1i}x_{2i} & \cdots & \sum x_{1i}x_{pi} \\ \sum x_{2i} & \sum x_{1i}x_{2i} & \sum x_{2i}^2 & \cdots & \sum x_{2i}x_{pi} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \sum x_{pi} & \sum x_{1i}x_{pi} & \sum x_{1i}x_{pi} & \cdots & \sum x_{pi}^2 \end{bmatrix}^{-1} \begin{bmatrix} \sum y_i \\ \sum x_{1i} y_i \\ \sum x_{2i} y_i \\ \vdots \\ \sum x_{pi} y_i \end{bmatrix}$$

Let the inverse of the matrix be denoted by

$$c = \begin{bmatrix} c_{00} & c_{01} & c_{02} & \cdots & c_{0p} \\ & c_{11} & c_{12} & \cdots & c_{1p} \\ & & c_{22} & \cdots & c_{2p} \\ & & & \cdots & \vdots \\ & & & & c_{pp} \end{bmatrix}$$

Then , $Var(b_i) = c_{ii}\sigma^2$, $i = 0,1,2,\cdots p$ and $Cov(b_i, b_j) = c_{ij}\sigma^2$.

Estimation of σ^2

In addition to estimating $\beta's$ an estimate of σ^2 is required to test hypotheses and construct interval estimates pertinent to the regression model. Ideally, we would like this estimate not to depend on the adequacy of the fitted model. This is only possible when there are several observations on y for at least one value of x, or when prior information concerning σ^2 is available. When this approach cannot be used, the estimate of σ^2 is obtained from the residual or error sum of squares.

$$SSE = \sum_{i=1}^{n} \hat{\varepsilon}_i^2 \quad = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

The residual sum of square has $(n-p-1)$ degrees of freedom, because $p+1$ degrees are associated with the estimates $\beta's$ involved in obtaining $\hat{y}_i$. Now the expected value of SSE is $E(SSE) = (n-p-1)\sigma^2$ so an unbiased estimator of σ^2 is

$$\hat{\sigma}^2 = s^2 = \frac{SSE}{n-p-1} = MSE .$$

The quantity MSE is called the error mean square or the residual mean square. The square root of s^2 is sometimes called the standard error of regression, and it has the same units as the response variable y.

The above splitting of the total sum of squares due to $y's$ into two components can be formally put in an Analysis of variance table, as below:

Analysis of Variance for Multiple Linear Regression

Source of Variation	**d.f.**	**S.S.**	**M.S.**
Regression	p	$\sum b_i S_{xiy}$	MSR
Deviation form Regression (Residual)	$n-p-1$	SSE	$s^2 = SSE/(n-p-1) = MSE$

Total (corrected mean)	$n-1$	S_{yy}	

Estimate of σ^2 is this case works out to be

$$s^2 = \frac{S_{yy} - \sum_{i=1}^{n} b_1 S x_i y}{n-p-1}$$

Test of Significance of β's

One may be interested in testing the hypotheses $H_0 : \beta_i = 0$ against $H_1 : \beta_i \neq 0$ for some i. The appropriate test statistic for testing this is

$$t = \frac{|b_i|}{SE(b_1)} = \frac{|b_i|}{\sqrt{c_{ii} s^2}} \text{ as } Var(b_i) = c_{ii} s^2$$

Which follows, under the hypothesis a *t*-distribution with $(n-p-1)$ d.f, if H_0 is true. Reject H_0 if $t > t_{\alpha/2, n-p-1}$.

Multiple correlation coefficient (R)

The correlation coefficient between the observed values y_i and predicted values $\hat{y}_i$ is termed as multiple correlation coefficient (R). Note that $0 \le R \le 1$. R is obtained as

$$R = \sqrt{\frac{Sum\ of\ Squares\ due\ to\ regression/\beta_0}{Total\ corrected\ sum\ of\ squares\ of\ y}}$$

$$= \sqrt{\frac{b_0 \sum_{i=1}^{n} y_i + b_1 \sum_{i=1}^{n} x_{1i} y_i + b_2 \sum_{i=1}^{n} x_{2i} y_i + b_p \sum_{i=1}^{n} x_{pi} y_i - \left(\sum_{i=1}^{n} y_i\right)^2 \frac{1}{n}}{\sum_{i=1}^{n} y_i^2 - \left(\sum_{i=1}^{n} y\right)^2 \frac{1}{n}}}$$

$$= \sqrt{\frac{\sum_{i=1}^{p} b_i S_{xiy}}{S_{yy}}}$$

Test of Significance of R

The test of the null hypothesis that multiple correlation coefficient in the population is zero is identical to the F-test of the null hypothesis that $\beta_1 = \beta_2 = \cdots = \beta_p = 0$. The relation is

$F = \dfrac{R^2}{1-R^2} \quad \dfrac{n-p-1}{p}$. This F follows F-distribution with p and $(n-p-1)$ d.f. Reject H_0 if $F > F_{\alpha, p, n-p-1}$.

Coefficient of determination $\left(R^2\right)$

The sample coefficient of multiple determination, denoted by $R^2_{Y.12\ldots p}$, is given by

$$R^2_{Y.12} = 1 - \frac{SSE}{(n-1)s_y^{2'}}$$

where $SSE = S_{yy} - b_1 Sx_1 y - b_2 Sx_2 y - \cdots - b_p Sx_p y$

One can easily see that the coefficient of multiple determination is the square of multiple correlation coefficient and is denoted by $\left(R^2\right)$. This concept is very important as $R^2 \times 100$ gives percentage of variation in y explained by regressors. Obviously R^2 must lie between 0 and 1. Thus R^2 is an indicator of fitness of the fitted model. However, a large value of R^2 should not alone be taken as a measure of goodness of fitted regression model.

4. Discussion

In addition to predicting the outcome variable for a new sample of data, regression analysis serves other purposes:

- To assess how well the dependent variable can be explained by knowing the value of the independent variable (or a set of independent variables).
- To identify which subset from many measures is most effective for estimating the dependent variable.

For this, one should first explore the variables graphically in scatterplots to ascertain if a linear model is appropriate for describing the relationship and to identify any possible rogue values (outliers) that might distort results. Ideally, in an observational study, the configuration of plot points should form the shape of an American football, for there are fewer points at the low and high ends of the independent variable than in the middle. In an experimental study, the values of *x* are fixed or set at specified levels, so the configuration may not exhibit such a clear pattern.

In assessing the suitability of the data for a regression, it helps to think of the fixed *x* situation. Visually scan the distribution of *y* values for each *x* (or each small range of *x*'s) – that is, look at vertical strips or bands of points extending up from the x axis. Do the y values within each strip look like a sample from a normal distribution? Is the spread (variance) within each strip roughly the same across the strips? Or is it considerably greater at one side of the plot than at the other? If you guess an average value of y for each strip, do these averages fall along a straight line?

More formally, normality is not required for the estimates of the coefficients. To make tests and estimate confidence intervals, however, these assumptions are required:

- The errors are normally distributed with mean 0.
- The errors have constant variance.
- The errors are independent of each other.

These assumptions are checked by studying the residuals from the model. The **Durbin-Watson** statistic can be used to test for the serial correlation of adjacent error terms.

To identify problems, always look at plots of y versus x before the regression and plots of residuals and diagnostics after the analysis. Non-linearity, Outliers and the presence of sub-populations can distort the results of regression analysis. Relationships among the dependent and independent variables may be *masked or falsely enhanced* if your sample contains subpopulations (that is, the sample is not homogeneous).

In summary, to help identify problems, always look at plots of y versus x before the regression and plots of residuals and diagnostics after the analysis.

For fitting a regression equation using MS-EXCEL: Prepare your data in a Worksheet. Now choose Tools → Data Analysis → Regression. Then give the range for dependent variable, independent variables and output range. If a regression equation without intercept is required then check on Intercept zero.

5. Steps for Linear Regression using SPSS

From the menus choose: Analyze → Regression → Linear... → Click Reset to restore dialog box defaults, and then select: In the Linear Regression dialog box, select a numeric dependent variable → Select one or more numeric independent variables. Optionally, one can:

- Group independent variables into blocks and specify different entry methods for different subsets of variables.
- Choose a selection variable to limit the analysis to a subset of cases having a particular value(s) for this variable.
- Select a case identification variable for identifying points on plots.
- Click WLS for a weighted least-squares analysis and move a numeric weighting variable into the WLS Weight box.

Linear Regression Statistics: The following statistics are available

Regression Coefficients. **Estimates** displays Regression coefficient B, standard error of B, standardized coefficient beta, t value for B, and two-tailed significance level of t. **Confidence intervals** displays 95% confidence intervals for each regression coefficient, or a covariance matrix. **Covariance matrix** displays a variance-covariance matrix of regression coefficients with covariances off the diagonal and variances on the diagonal. A correlation matrix is also displayed.

Model fit. The variables entered and removed from the model are listed, and the following goodness-of-fit statistics are displayed: multiple R, R^2 and adjusted R^2, standard error of the estimate, and an analysis-of-variance table.

R-squared change. Displays changes in R^2 change, F-change, and the significance of F change.

Descriptives. Provides the number of valid cases, the mean, and the standard deviation for each variable in the analysis. A correlation matrix with a one-tailed significance level and the number of cases for each correlation are also displayed.

Part and partial correlations. Displays zero-order, part, and partial correlations.
Collinearity diagnostics. Eigenvalues of the scaled and uncentered cross-products matrix, condition indices, and variance-decomposition proportions are displayed along with variance inflation factors (VIF) and tolerances for individual variables.

Residuals. Displays the Durbin-Watson test for serial correlation of the residuals and casewise diagnostics for the cases meeting the selection criterion (outliers above n standard deviations). casewise diagnostics.

Linear Regression Plots
Plots can aid in the validation of the assumptions of normality, linearity, and equality of variances. Plots are also useful for detecting outliers, unusual observations, and influential cases. After saving them as new variables, predicted values, residuals, and other diagnostics are available in the Data Editor for constructing plots with the independent variables. The following plots are available:

Scatterplots. One can plot any two of the following: the dependent variable, standardized predicted values, standardized residuals, deleted residuals, adjusted predicted values, Studentized residuals, or Studentized deleted residuals. Plot the standardized residuals against the standardized predicted values to check for linearity and equality of variances.

Produce all partial plots. Displays scatterplots of residuals of each independent variable and the residuals of the dependent variable when both variables are regressed separately on the rest of the independent variables. At least two independent variables must be in the equation for a partial plot to be produced.

Standardized Residual Plots. One can obtain histograms of standardized residuals and normal probability plots comparing the distribution of standardized residuals to a normal distribution**.**

If any plots are requested, summary statistics are displayed for standardized predicted values and standardized residuals (*ZPRED and *ZRESID).

To Obtain Plots With a Linear Regression
In the Linear Regression dialog box, click Plots.
- *For scatterplots, select one variable for the vertical (y) axis and one variable for the horizontal (x) axis. To request additional scatterplots, click Next.*

Linear Regression Save
One can save predicted values, residuals, and other statistics useful for diagnostics. Each selection adds one or more new variables to your active data file.

Predicted Values. Values that the regression model predicts for each case.

Distances. Measures to identify cases with unusual combinations of values for the independent variables and cases that may have a large impact on the regression model.

Prediction Intervals. The upper and lower bounds for both mean and individual prediction intervals.

Residuals. The actual value of the dependent variable minus the value predicted by the regression equation.

Influence Statistics. The change in the regression coefficients (DfBeta(s)) and predicted values (DfFit) that results from the exclusion of a particular case. Standardized DfBetas and DfFit values are also available along with the covariance ratio, which is the ratio of the determinant of the covariance matrix with a particular case excluded to the determinant of the covariance matrix with all cases included.

Save to New File. Saves regression coefficients to a file that you specify.

In the Linear Regression dialog box, click Save → Select the values or desired statistics.

Linear Regression Options
The following options are available:
Stepping Method Criteria. These options apply when either the forward, backward, or stepwise variable selection method has been specified. Variables can be entered or removed from the model depending on either the significance (probability) of the F value or the F value itself.

Include constant in equation. By default, the regression model includes a constant term. Deselecting this option forces regression through the origin, which is rarely done. Some results of regression through the origin are not comparable to results of regression that do include a constant. For example, R^2 cannot be interpreted in the usual way**.**

Missing Values. One can choose one of the following**:**
Exclude cases listwise. Only cases with valid values for all variables are included in the analyses.
Exclude cases pairwise. Cases with complete data for the pair of variables being correlated are used to compute the correlation coefficient on which the regression analysis is based. Degrees of freedom are based on the minimum pairwise N.
Replace with mean. All cases are used for computations, with the mean of the variable substituted for missing observations**.**

To Specify Options for a Linear Regression
In the Linear Regression dialog box, click Options → Select the desired options.
In General Steps for Linear Regression
From the menus choose: Analyze → Regression → Linear... → Click Reset to restore dialog box defaults, and then select:

- Dependent:
- Independent:
- Case Labels:

Statistics…
Descriptives
Residuals
Casewise diagonostics
Outliers outside 1.0 standard deviations
Plots

- Y: SDRESID
- X: ZPRED

Click Next
- Y: ZPRED
- X: DEPENDENT

Standardized Residual Plots
Normal probability plot
Save…
Predicted Values
Standardized
Distances
Cook's Leverage values
Residuals
Studentized deleted
Influence Statistics
Standardized DfBeta(s) DfFit

On the Correlation table select and hide the Sig. (1-tailed) and N rows.
Options…
Include constant in equation…

To hide footnotes, select them and then choose Hide from the View menu.

Note: A multiple linear regression equation can also be fitted using the following options:

From the menu choose: Analyze→ General Linear Model → Univariate …. → Dependent Variable → Covariate(s) → Model…→ Custom → Model main effects (same as covariates) →options…→Display→ observed power, parameter estimates, and residual plot.

6. Exercises

Exercise 6.1: The following data are taken from Berenson and Levine (1992). Fifteen similar homes built by one developer in various locations around the United States were evaluated in the study. The builders recorded the amount of oil consumed in January, the average outside temperature (in degree Fahrenheit), and the number of inches of attic insulation in each home.

Case	Avg.Temp.	Insulation(Inches)	Oil Consumed in January
1	40	3	275.3
2	27	3	363.8
3	40	10	164.3
4	73	6	40.8
5	64	6	94.3
6	34	6	230.9
7	9	6	366.7
8	8	10	300.6
9	23	10	237.8
10	63	3	121.4
11	65	10	31.4
12	41	6	203.5
13	21	3	441.1

14	38	3	323.0
15	58	10	52.5

1. Draw a scatter plot matrix using insulation, Oil consumed and temperature.
2. Fit a multiple linear regression equation using oil consumed as dependent variable and insulation and average temperature as independent variable

Exercise 6.2: The follwing data was collected through a pilot sample survey on Hybrid Jowar crop on yield and biometrical characters. The biometrical characters were average Plant Population (PP), average Plant Height(PH), average Number of Green Leaves(NGL) and Yield (Kg./plot).

1. Plot a simple scatter diagram between (i) yield and PP (ii) yield and PH (iii) yield and NGL.
2. Plot a scatter diagram using matrix option using the variables yield, PP, PH and NGL.
3. Compute bivariate and partial correlations among yield, PP, PH and NGL.
4. Fit a simple linear regression by taking yield as dependent variable and NGL as independent variable.
5. Fit a multiple linear regression equation by taking yield as dependent variable and biometrical characters as explanatory variables.
6. Compute various statistics viz. Estimates, Confidence intervals, Covariance matrix , Model fit, R-squared change, Descriptives, Part and partial correlations, Collinearity diagnostics and Residuals
7. Check for multicollinearity between PP, PH and NGL.
8. Identify the outliers in the data set.
9. Fit the model without intercept.

No.	PP	PH	NGL	Yield
1	142.00	0.5250	8.20	2.470
2	143.00	0.6400	9.50	4.760
3	107.00	0.6600	9.30	3.310
4	78.00	0.6600	7.50	1.970
5	100.00	0.4600	5.90	1.340
6	86.50	0.3450	6.40	1.140
7	103.50	0.8600	6.40	1.500
8	155.99	0.3300	7.50	2.030
9	80.88	0.2850	8.40	2.540
10	109.77	0.5900	10.60	4.900
11	61.77	0.2650	8.30	2.910
12	79.11	0.6600	11.60	2.760
13	155.99	0.4200	8.10	0.590
14	61.81	0.3400	9.40	0.840
15	74.50	0.6300	8.40	3.870
16	97.00	0.7050	7.20	4.470
17	93.14	0.6800	6.40	3.310

18	37.43	0.6650	8.40	1.570
19	36.44	0.2750	7.40	0.530
20	51.00	0.2800	7.40	1.150
21	104.00	0.2800	9.80	1.080
22	49.00	0.4900	4.80	1.830
23	54.66	0.3850	5.50	0.760
24	55.55	0.2650	5.00	0.430
25	88.44	0.9800	5.00	4.080
26	99.55	0.6450	9.60	2.830
27	63.99	0.6350	5.60	2.570
28	101.77	0.2900	8.20	7.420
29	138.66	0.7200	9.90	2.620
30	90.22	0.6300	8.40	2.000
31	76.92	1.2500	7.30	1.990
32	126.22	0.5800	6.90	1.360
33	80.36	0.6050	6.80	0.680
34	150.23	1.1900	8.80	5.360
35	56.50	0.3550	9.70	2.120
36	136.00	0.5900	10.20	4.160
37	144.50	0.6100	9.80	3.120
38	157.33	0.6050	8.80	2.070
39	91.99	0.3800	7.70	1.170
40	121.50	0.5500	7.70	3.620
41	64.50	0.3200	5.70	0.670
42	116.00	0.4550	6.80	3.050
43	77.50	0.7200	11.80	1.700
44	70.43	0.6250	10.00	1.550
45	133.77	0.5350	9.30	3.280
46	89.99	0.4900	9.80	2.690

Exercise 6.3: The following data in on 6 variables.

X	Y1	Y2	Y3	Y4	X1
4.00	4.26	3.10	5.39	6.58	8.00
5.00	5.68	4.74	5.73	5.76	8.00
6.00	7.24	6.13	6.08	7.71	8.00
7.00	4.82	7.26	6.42	8.84	8.00
8.00	6.95	8.14	6.77	8.47	8.00
9.00	8.81	8.77	7.11	7.04	8.00
10.00	8.04	9.14	7.46	5.25	8.00
11.00	8.33	9.26	7.81	5.56	8.00
12.00	10.84	9.13	8.15	7.91	8.00
13.00	7.58	8.74	12.74	6.89	8.00
14.00	9.96	8.10	8.84	12.50	19.00

(a) Plot simple scatter plots between (i) y1 and x; (ii) y2 and x; (iii) y3 and x and (iv) y4 and x1.

(b) Obtain bivariate correlation coefficients between (i) y1 and x; (ii) y2 and x; (iii) y3 and x and (iv) y4 and x1.

(c) Fir simple linear regression equations for each of the four pairs by taking y-variables as dependent and x-variables as independent.

TESTS OF SIGNIFICANCE

Seema Jaggi
ICAR-IASRI, Library Avenue, New Delhi – 110 012
seema.jaggi@icar.gov.in

1. Introduction

In applied investigations, one is often interested in comparing some characteristic (such as mean or variance) of a group with a specified value, or in comparing two or more groups with regard to the characteristic. For instance, one may want to know whether mean timber yield obtained from recently felled plantations of a particular age in a particular management unit is some specifid value, one may wish to know whether average yield of a crop in a certain district is equal to a specified value, one may wish to compare two species of trees with regard to mean height, to know if genetic fraction of total variation in a strain is more than a given value. In making such comparisons, one can not rely on mere numerical magnitudes of index of comparison such as mean and variance. This is because each group is represented only by a sample of observations and if another sample were drawn, the numerical value would change. This variation between samples from the same population can at best be reduced in a well-designed controlled experiment but can never be eliminated. One is forced to draw inferences in presence of sampling fluctuations which affect observed differences between groups, clouding real differences. Statistical science provides an objective procedure for distinguishing whether observed difference connotes any real difference among groups. Such a procedure is called **testing of hypothesis**. Thus, in short, testing of hypothesis is a method of making due allowance for sampling fluctuation affecting results of experiments or observations. These tests have wide applications in agriculture, forestry, medicine, industry, social sciences, etc.

1.1 Definitions

Statistical Hypothesis: It is an assumption either about the form or about the parameters of a distribution. For example, average height of a particular species of tree is 50 feet, normal distribution has mean 20.

If all the parameters are completely specified, hypothesis is called a **simple hypothesis**, otherwise it is a **composite hypothesis**. For example, average height of tree is 50 feet is a simple hypothesis and average height of tree is greater than 50 feet is a composite hypothesis.

Null Hypothesis (H_0): The hypothesis under test for a sample study is called Null hypothesis (H_0). It represents a theory that has been put forward, either because it is believed to be true or because it is to be used as a basis for argument, but has not been proved. For example, in a clinical trial of a new drug, null hypothesis might be that the new drug is, on average, as effective as the current drug i.e. H_0: Effect of the two drugs, on the average, is same.

Alternative Hypothesis (H_1): Any null hypothesis is tested against a rival, which is called Alternative hypothesis (H_1). For example, mean height (μ) of trees of a particular species in a region is some specified value μ_0, i.e.

H_0: $\mu = \mu_0$.

Alternative hypothesis could be any of the following:

H_1: $\mu \neq \mu_0$ (Two-tailed)
$\mu < \mu_0$ (Left-tailed)
$\mu > \mu_0$ (Right-tailed)

For framing a suitable H_0 and H_1, four possibilities in order of preference are the following:

Possibilities	**H_0**	**H_1**
(i)	Simple	Simple
(ii)	Simple	Composite
(iii)	Composite	Simple
(iv)	Composite	Composite

The first one when both are simple is of little practical importance. As Possibility (ii) is preferred over Possibility (iii), therefore hypotheses should always be structured in such a way that H_0 is simple and H_1 is composite.

Two Types of Errors

True Situation → Decision Made ↓	**H_0 is True**	**H_0 is False**
Reject H_0	Type I error	Correct decision
Accept H_0	Correct decision	Type II error

Probabilities of these types of error are respectively denoted by α and β, i.e.
Probability of Type I error = α
and Probability of Type II error = β.

The ideal procedure of hypothesis testing is to minimize both α and β. However, this is not possible in practice because a test which minimizes one type of error, maximizes the other type of error. As Type I error is considered to be more serious than Type II error, therefore probability of Type I error is fixed and probability of Type II error is minimized. Generally, α is taken to be 5% or 1%.

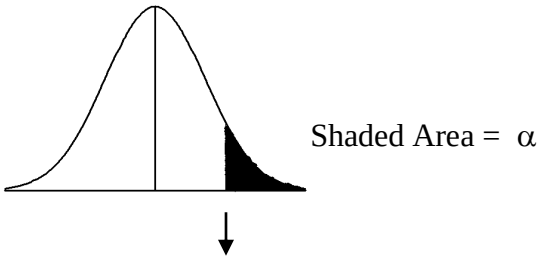
Level of Significance (α): It is the size of Type I error. The higher the value of α, less precise is the result.

Confidence Interval: The confidence interval of a parameter with confidence coefficient 100(1-α)% is the interval (a, b) such that it is expected to lie in this interval in 100(1-α)% cases.

Test Statistic: A test statistic is a quantity calculated from data. Its value is used to decide whether or not the null hypothesis should be rejected.

Critical Value(s): The critical value(s) is that value with which value of test statistic in a sample is compared to determine whether or not the null hypothesis is rejected. The critical value for any

hypothesis test depends on significance level α at which the test is carried out, and whether the test is one-sided or two-sided.



Power of a Test: It is defined as the probability of rejecting H_0 when it is false. Thus,

Power = 1 - β

Among a given set of tests, best test is one having maximum power.

Steps in Hypothesis Testing

- State statistical hypotheses
- Check assumptions
- Calculate test statistic
- Set the test criteria
- Interpret the results

We now discuss some tests of hypothesis that are based on normal, t, F and chi-square distributions.

2. Test of Significance for Large Samples

For large n (sample size), almost all the distributions can be approximated very closely by a normal probability curve, we therefore use the **normal test** of significance for large samples. If t is any statistic (function of sample values), then for large sample

$$Z = \frac{t - E(t)}{\sqrt{V(t)}} = N(0.1)$$

Thus if the discrepancy between the observed and the expected (hypothetical) value of a statistic is greater than Z_α times the standard error (S.E), hypothesis is rejected at α level of significance. Similarly if

$$| t - E(t) | \leq Z_\alpha \times S.E(t),$$

the deviation is not regarded significant at 5% level of significance. In other words the deviation t - E(t), could have arisen due to fluctuations of sampling and the data do not provide any evidence against the null hypothesis which may, therefore be accepted at α level of significance. If $|Z| \leq 1.96$, then the hypothesis H_0 is accepted at 5% level of significance. Thus the steps to be used in the normal test are as follows:

i) Compute the test statistic Z under H_0.

ii) If $|Z| > 3$, H_0 is always rejected
iii) If $|Z| < 3$, we test its significance at certain level of significance

The table below gives some critical values of Z:

Level of Significance	Critical Value (Z_α) of Z	
	Two-tailed test	Single tailed test
10%	1.645	1.280
5%	1.960	1.645
1%	2.580	2.330

2.1 Test for Single Mean

A very important assumption underlying the tests of significance for variables is that the sample mean is asymptotically normally distributed even if the parent population from which the sample is drawn is not normal.

If x_i (i =1,…,n) is a random sample of size n from a normal population with mean μ and variance σ^2, then the sample mean is distributed normally with mean μ and variance $\frac{\sigma^2}{n}$. Based on this random sample, our aim is to test that mean of the population has a specified value μ_0, i.e.

H_0: $\mu = \mu_0$

The alternative hypothesis could be any of the following:

H_1: $\mu \neq \mu_0$ (two tailed)
$\mu < \mu_0$ (left tailed)
$\mu > \mu_0$ (right tailed)

Test Statistic:

$$Z = \frac{\bar{x} - \mu_0}{\sigma / \sqrt{n}}$$

follows a standard normal distribution.

Test Criteria: Depending on the alternative hypothesis selected, the test criteria are as follows:

H_1	Test	Reject H_0 at level of significance α if
$\mu \neq \mu_0$	Two-tailed	$\|Z\| > Z_{\alpha/2}$
$\mu < \mu_0$	Left-tailed	$Z < -Z_\alpha$
$\mu > \mu_0$	Right-tailed	$Z > Z_\alpha$

Z_α is the table value of Z at level of significance α. If σ^2 is unknown, then it is estimated by sample variance s^2 (for large n), where $s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$

Example 2.1: The mean timber yield obtained from 30 recently felled plantations at the age of 50 years in a particular management unit is 93 m^3/ha with a standard deviation of 10 m^3/ha. Test whether the mean timber yield is 100 m^3/ha based on past records.

Solution: H_0 : $\mu = 100$ m^3/ha, H_1: $\mu \neq 100$ m^3/ha (two tailed test).

Here, $\bar{x} = 93$ m^3/ha., $n = 30$, $\mu = 100$ m^3/ha and $\sigma = 10$ m^3/ha.

Thus,

$$Z = \frac{93-100}{10/\sqrt{30}} = -3.834$$

Since $|Z| > 1.96$, we conclude that the data does not provide any evidence in favour of the null hypothesis H_0 may therefore be rejected at 5% level of significance. Hence the decision would be to accept the alternative hypothesis that there has been significant decline in the productivity of the management unit with respect to the plantations of the species considered.

Note: The value of sample mean is an acceptable value of population mean if the statistic Z lies between $-Z_{\alpha/2}$ to $Z_{\alpha/2}$, i.e.

$$-Z_{\alpha/2} \leq \frac{\bar{x}-\mu}{\sigma/\sqrt{n}} \leq Z_{\alpha/2}.$$

Thus, 100(1-α)% confidence-interval for μ is

$$(\bar{x} - Z_{\alpha/2}\,\sigma/\sqrt{n},\ \bar{x} + Z_{\alpha/2}\,\sigma/\sqrt{n}).$$

2.2 Test for Difference of Means

Let $\bar{x}_1\,(\bar{x}_2)$ be the mean of a sample of size n_1 (n_2) from a population with mean μ_1 (μ_2) and variance $\sigma_1^2\,(\sigma_2^2)$. Our aim is to test

H_0 : $\mu_1 = \mu_2$

against H_1 : $\mu_1 \neq \mu_2$

$\mu_1 > \mu_2$

$\mu_1 < \mu_2$

Test Statistic:

$$Z = \frac{(\bar{x}_1 - \bar{x}_2)}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}} \sim N(0,1)$$

follows a standard normal distribution

Test Criteria:

H_1	Test	Reject H_0 at level of

		significance α if
$\mu_1 \neq \mu_2$	Two-tailed	$\|Z\| > Z_{\alpha/2}$
$\mu_1 < \mu_2$	Left-tailed	$Z < -Z_{\alpha}$
$\mu_1 > \mu_2$	Right-tailed	$Z > Z_{\alpha}$

$$Z = \frac{(\bar{x}_1 - \bar{x}_2)}{\sigma\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}} \quad \text{, If } \sigma_1^2 = \sigma_2^2 = \sigma^2$$

If σ is not known, then its estimate is used

$$\hat{\sigma}^2 = s^2 = \frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2}$$

2.3 Test for Single Proportion

Suppose in a sample of size n (>30), x be the number of successes. Then observed proportion of successes = $x/n = p$. Let P be the population proportion. The hypothesis to be tested is that population proportion is some specified value P_0, i.e.

H_0: $P = P_0$
H_1: $P \neq P_0$
$P > P_0$
$P < P_0$

Test Statistic:

$$Z = \frac{p - P_0}{\sqrt{P_0(1 - P_0)/n}}$$

follows approximately a standard normal distribution.

Test Criteria:

H_1	**Test**	**Reject H_0 at level of significance α if**
$P \neq P_0$	Two-tailed	$\|Z\| > Z_{\alpha/2}$
$P < P_0$	Left-tailed	$Z < -Z_{\alpha}$
$P > P_0$	Right-tailed	$Z > Z_{\alpha}$

Example 2.2: In a sample of 1000 people, 540 are rice eaters and the rest are wheat eaters. Can we assume that both rice and wheat are equally popular at 1% level of significance?

Solution: It is given that n = 1000, x = Number of rice eaters = 540, p = sample proportion of rice eaters $= 540/1000 = 0.54$.

H_0 : Both rice and wheat are equally popular, i.e. P = 0.5

H_1 : $P \neq 0.5$

$$Z = \frac{p - P_0}{\sqrt{P_0(1 - P_0)/n}} = \frac{0.54 - 0.5}{\sqrt{0.5 \text{ x } 0.5/1000}} = 2.532$$

Tabulated value of Z at 1% level of significance is 2.575. Since $|Z| < 2.575$, therefore H_0 is not rejected and we conclude that rice and wheat are equally popular.

2.4 Test for Difference of Proportions

Suppose we want to compare two populations with respect to the prevalence of a certain attribute A. Let x_1 (x_2) be the number of persons possessing the given attribute A in random sample of size n_1 (n_2) from 1st (2nd) population. Then sample proportions will be

$$p_1 = \frac{x_1}{n_1},\ p_2 = \frac{x_2}{n_2}$$

Let P_1 and P_2 be the population proportions. Our aim here is to test that there is no significant difference between population proportions, i.e.

H_0: $P_1 = P_2$
H_1: $P_1 \neq P_2$
$P_1 > P_2$
$P_1 < P_2$

Test Statistic:

$$Z = \frac{p_1 - p_2}{\sqrt{\left(\frac{P_1Q_1}{n_1} + \frac{P_2Q_2}{n_2}\right)}}$$

follows approximately a standard normal distribution. In case $P_1 = P_2 = P$ (say) and P is not known, it is estimated as follows:

$$\hat{P} = \frac{n_1p_1 + n_2p_2}{n_1 + n_2}$$

Test Criteria:

H_1	Test	Reject H_0 at level of significance α if
$P_1 \neq P_2$	Two-tailed	$\|Z\| > Z_{\alpha/2}$
$P_1 < P_2$	Left-tailed	$Z < -Z_{\alpha}$
$P_1 > P_2$	Right-tailed	$Z > Z_{\alpha}$

Consider an experiment on rooting of stem cuttings of *Casuarina equisetifolia* wherein the effect of dipping the cuttings in solutions of IBA at two different concentrations was observed. Two batches of 30 cuttings each, were subjected dipping treatment at concentrations of 50 and 100 ppm of IBA solutions respectively. Based on the observations on number of cuttings rooted in each batch of 30 cuttings, the following proportions of rooted cuttings under each concentration were obtained. At 50 ppm, the proportion of rooted cuttings was 0.5 and at 100 ppm, the proportion was 0.37. Test whether the observed proportions are indicative of significant differences in the effect of IBA at the two concentrations.

Here, $p_1 = 0.5$ and $p_2 = 0.37$. Then $q_1 = 0.5$, $q_2 = 0.63$. The value of $n_1 = n_2 = 30$. Thus,

$$Z = \frac{0.5 - 0.37}{\sqrt{\frac{(0.5)(0.5)}{30} + \frac{(0.37)(0.63)}{30}}} = 1.024$$

Since the calculated value of Z (1.024) is less than the table value (1.96) at 5% level of significance, we can conclude that there is no significant difference between proportion rooted cuttings under the two concentration levels.

3. Test of Significance for Small Samples

In this section, the statistical tests based on t, χ^2 and F are given.

3.1 Tests Based on t-Distribution

3.1.1 Test for an Assumed Population Mean

Suppose a random sample $x_1,..,x_n$ of size n ($n \geq 2$) has been drawn from a normal population whose variance σ^2 is unknown. On the basis of this random sample the aim is to test

H_0 : $\mu = \mu_0$

H_0 : $\mu \neq \mu_0$

$\mu > \mu_0$

$\mu < \mu_0$

Test statistic:

$$t = \frac{\bar{x} - \mu_0}{s/\sqrt{n}} \sim t_{n-1},$$

where $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ and $s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$

The table giving the value of t required for significance at various levels of probability and for different degrees of freedom are called the t – tables which are given in Statistical Tables by Fisher and Yates. The computed value is compared with the tabulated value at α percent level of significance and at (n-1) degrees of freedom and accordingly the null hypothesis is accepted or rejected.

3.1.2 Test for the Difference of Two Population Means

Let $\bar{x}_1 (\bar{x}_2)$ be the sample mean of a sample of size n_1 (n_2) from a population with mean μ_1 (μ_2) and variance of the two population be same σ^2, which is unknown. Our aim is to test

H_0 : $\mu_1 = \mu_2$

H_1 : $\mu_1 \neq \mu_2$ or $\mu_1 > \mu_2$ or $\mu_1 < \mu_2$

Let s_i^2, i =1, 2 be sample variances of the two samples. Then common unknown population variance σ^2 is estimated as

$$s^2 = \frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2}$$

Test Statistic:

$$t = \frac{\bar{x}_1 - \bar{x}_2}{s\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$$

which follows a t-distribution with $n_1 + n_2 - 2$ d.f.

Test Criteria:

H_1	Test	Reject H_0 at level of significance α if
$\mu_1 \neq \mu_2$	Two-tailed	$\lvert t \rvert > t_{n_1+n_2-2}(\alpha/2)$
$\mu_1 < \mu_2$	Left-tailed	$t < -t_{n_1+n_2-2}(\alpha)$
$\mu_1 > \mu_2$	Right-tailed	$t > t_{n_1+n_2-2}(\alpha)$

This test statistic is used under certain assumptions *viz*., (i) The variables involved are continuous (ii) The population from which the samples are drawn follow normal distribution (iii) The samples are drawn independently (iv) The variances of the two populations from which the samples are drawn are homogeneous (equal). The homogeneity of two variances can be tested by using F-test.

Example 3.1: A group of 5 plots treated with nitrogen at 20 kg/ha. yielded 42, 39, 48, 60 and 41 kg whereas second group of 7 plots treated with nitrogen at 40 kg/ha. yielded 38, 42, 56, 64, 68, 69 and 62 kg. Can it be concluded that nitrogen at level 40 kg/ha. increases the yield significantly?

Solution: H_0: $\mu_1 = \mu_2$, H_1: $\mu_1 < \mu_2$

Here, $\bar{x}_1 = 46$, $\bar{x}_2 = 57$, $s^2 = 121.6$

$$t = \frac{46 - 57}{\sqrt{121.6(\frac{1}{5} + \frac{1}{7})}} = -1.7 \sim t_{10}$$

Since $|t| < 1.81$ (value of t at 5% and 10 d.f), the yield from two doses of nitrogen do not differ significantly.

3.1.3 Paired t-test for Difference of Means

When the two samples are not independent but the sample observations are paired together, then this test is applied. The paired observations are on the same unit or matching units. For example, to know the impact of a new teaching method on the performance of students, the observations, in terms of marks, are collected before and after the new teaching method is implemented. Let (x_i, y_i), $i = 1,\ldots,n$ be the pairs of observations and let $d_i = x_i - y_i$. Our aim is to test

H_0 : $\mu_1 = \mu_2$

H_1 : $\mu_1 \neq \mu_2$

$\mu_1 > \mu_2$

$\mu_1 < \mu_2$

Test Statistic:

$$t = \frac{\bar{d}}{s_d/\sqrt{n}}$$

follows t distribution with n-1 d.f., where $\bar{d} = \frac{1}{n}\sum_{i=1}^{n} d_i$ and $s_d^2 = \frac{1}{n-1}\sum_{i=1}^{n}(d_i - \bar{d})^2$.

Test Criteria:

H_1	Test	Reject H_0 at level of significance α if
$\mu_1 \neq \mu_2$	Two-tailed	$\lvert t \rvert > t_{n-1}(\alpha/2)$
$\mu_1 < \mu_2$	Left-tailed	$t < -t_{n-1}(\alpha)$
$\mu_1 > \mu_2$	Right-tailed	$t > t_{n-1}(\alpha)$

3.1.4 Test for Significance of Observed Correlation Coefficient

Given a random sample (x_i, y_i) , i = 1,…, n from a bivariate normal population. We want to test the null hypothesis that the population correlation coefficient is zero i.e.

$H_0 : \rho = 0$

$H_1 : \rho \neq 0$

Test Statistic:

$$t = \frac{r\sqrt{n-2}}{\sqrt{1-r^2}} \sim t_{n-2},$$

where r is the sample correlation coefficient. H_0 is rejected at level α if

$\lvert t \rvert > t_{n-2}(\alpha/2)$

This test can also be used for testing the significance of rank correlation coefficient.

3.2 Test of Significance Based on Chi-Square Distribution

3.2.1 Test for the Variance of a Normal Population

Let $x_1, x_2,\ldots,x_n$ $(n\geq 2)$ be a random sample from a normal population with mean μ and variance σ^2. On the basis of this sample our aim is to test

$H_0 : \sigma^2 = \sigma_0^2$

against $H_1 : \sigma^2 \neq \sigma_0^2$

$\sigma^2 < \sigma_0^2$

$\sigma^2 > \sigma_0^2$

Test Statistic:

$$\chi^2 = \sum_{i=1}^{n} \left(\frac{x_i - \mu}{\sigma_0} \right)^2$$

follows a chi-square distribution with n d.f. when μ is known, and

$$\chi^2 = \sum_{i=1}^{n} \left(\frac{x_i - \bar{x}}{\sigma_0} \right)^2 = \frac{(n-1)s^2}{\sigma_0^2}$$

follows a chi-square distribution with n-1 d.f. when μ is not known.

Test Criteria:

H_1	Test	Reject H_0 at level of significance α if	
		μ is known	μ is not known
$\sigma^2 \neq \sigma_0^2$	Two-tailed	$\chi^2 < \chi^2_n(1-\alpha/2)$ or $\chi^2 > \chi^2_n(\alpha/2)$	$\chi^2 < \chi^2_{n-1}(1-\alpha/2)$ or $\chi^2 > \chi^2_{n-1}(\alpha/2)$
$\sigma^2 < \sigma_0^2$	Left-tailed	$\chi^2 < \chi^2_n(1-\alpha)$	$\chi^2 < \chi^2_{n-1}(1-\alpha)$
$\sigma^2 > \sigma_0^2$	Right-tailed	$\chi^2 > \chi^2_n(\alpha)$	$\chi^2 > \chi^2_{n-1}(\alpha)$

Tables are available for χ^2 at different levels of significance and with different degrees of freedom.

3.2.2 Test for Goodness of Fit

A test of wide applicability to numerous problems of significance in frequency data is the χ^2 test of goodness of fit. It is primarily used for testing the discrepancy between the expected and the observed frequency, For instance, one may be interested in testing whether a variable like the height of trees follows normal distribution. A tree breeder may be interested to know whether the observed segregation ratios for a character deviate significantly from the Mendelian ratios. In such situations, we want to test the agreement between the observed and theoretical frequencies. Such a test is called a test of goodness of fit.

H_0 : the fitted distribution is a good fit to the given data

H_1 : not a good fit.

Test statistic: If O_i and E_i, i =1,…,n are respectively the observed and expected frequency of ith class, then the statistic

$$\chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \sim \chi^2_{n\text{-}r\text{-}1}$$

where r is the number of parameters estimated from the sample, n is the number of classes after pooling. H_0 is rejected at level α if calculated χ^2 > tabulated $\chi^2_{n\text{-}r\text{-}1}(\alpha)$.

Example 3.2: In an F_2 population of chillies, 831 plants with purple and 269 with non-purple chillies were observed. Is this ratio consistent with a single factor ratio of 3:1?

Solution: On the hypothesis of a ratio of 3:1, the frequencies expected in the purple and non-purple classes are 825 and 275 respectively.

	Frequency		
	Observed (O_i)	Expected (E_i)	$O_i - E_i$
Purpose	831	825	6
Non-purple	269	275	-6

$$\chi^2 = \sum_{i=1}^{2} \frac{(O_i - E_i)^2}{E_i} = 0.17 .$$

Here χ^2 is based on one degree of freedom. It is seen from the table that the value of 0.17 for χ^2 with 1 d.f corresponds to a level of probability which lies between 0.5 and 0.7. It is concluded that the result is non-significant.

3.2.3 Test of Independence

Another common use of the χ^2 test is in testing independence of classifications in what are known as contingency tables. When a group of individuals can be classified in two ways, the result of the classification in two ways the results of the classification can be set out as follows:

Contingency table

Class	**A_1**	**A_2**	**A_3**
B_1	n_{11}	n_{21}	n_{31}
B_2	n_{12}	n_{22}	n_{32}
B_3	n_{13}	n_{23}	n_{33}

Such a table giving the simultaneous classification of a body of data in two different ways is called contingency table. If there are r rows and c columns the table is said to be an r x c table.

H_0: the attributes are independent

H_1: they are not independent

Test statistic:

$$\chi^2 = \sum_{j=1}^{c} \sum_{i=1}^{r} \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \sim \chi^2_{(r-1)(c-1)}$$

H_0 is rejected at level α if $\chi^2 > \chi^2_{(r-1)(c-1)}$

3.3 Test of Significance Based on F-Distribution

3.3.1 Test for the Comparison of Two Population Variances

Let x_i , i = 1,…,n_1 and x_j, j=1,…,n_2 be the two random samples of sizes n_1 and n_2 drawn from two independent normal populations $N(\mu_1, \sigma_1^2)$ and $N(\mu_1, \sigma_2^2)$ respectively. s_1^2 and s_2^2 are the sample variances of the two samples.

$$s_1^2 = \frac{1}{n_1 - 1} \sum_{i=1}^{n1} (x_i - \bar{x}_1)^2 \text{ and } s_2^2 = \frac{1}{n_2 - 1} \sum_{j=1}^{n2} (x_j - \bar{x}_2)^2$$

$$\bar{x}_1 = \frac{1}{n_1} \sum_{i=1}^{n1} x_i, \quad \bar{x}_2 = \frac{1}{n_2} \sum_{j=1}^{n2} x_j$$

$H_0 : \sigma_1^2 = \sigma_2^2$

Test statistic: Assuming $s_1^2 > s_2^2$

$$F = \frac{s_1^2}{s_2^2} \sim F_{n_1-1,\, n_2-1}$$

Tables are available giving the values of F required for significance at different levels of probability and for different degrees of freedom. The computed value of F is compared with the tabulated value and the inference is drawn accordingly.

3.3.2 Test for Homogeneity of Several Population Means

The test of significance based on t-distribution is an adequate procedure only for testing the significance of the difference between two sample means. In a situation when we have three or more samples to consider at a time, an alternative procedure is needed for testing the hypothesis that all the samples are drawn from the same population i.e. they have the same mean. For Example, 5 fertilizers are applied to four plots each of wheat and yield of wheat on each of the plot is obtained. The interest is to find whether effects of these fertilizers on the yields is significantly different or in other words, whether the samples have come from the same normal population. This is done through F-test that uses the technique of Analysis of Variance (ANOVA).

ANOVA is the technique of partitioning the total variability into different known components. It consist in the estimation of the amount of variation due to each of the independent factors (causes) separately and then comparing these estimates due to assignable factors with the estimate due to chance factor or experimental error. The F statistic used for testing the hypothesis H_0: $\mu_1 = \mu_2 = \ldots = \mu_k$ $(k>2)$ is

$$F = \frac{\text{Variation among the sample means}}{\text{Variation within the samples}}$$

Practical on Testing of Hypothesis

1. Independent Samples t-Test

An experiment was conducted to evaluate the effect of inoculation with mycorrhiza on the height growth of seedlings of *Pinus kesiya*. In the experiment, 10 seedlings (Group I) were inoculated with mycorrhiza while another 10 seedlings (Group II) were left without inoculation with the microorganism. Following table gives the height of seedlings obtained under the two groups of seedlings:

Plot	Group I	Group II
1	23.0	8.5
2	17.4	9.6
3	17.0	7.7
4	20.5	10.1
5	22.7	9.7
6	24.0	13.2
7	22.5	10.3
8	22.7	9.1
9	19.4	10.5
10	18.8	7.4

Test whether inoculated and uninoculated seedlings are significantly different.
Solution: H_0: Mean of Group I (μ_1) = Mean of Group II (μ_2) and H_1: $\mu_1 \neq \mu_2$
From the given data $\bar{x}_1 = 20.8$, $\bar{x}_2 = 9.61$,

$$s_1^2 = \frac{(23.0)^2 + (17.4)^2 + ... + (18.8)^2 - \frac{(208)^2}{10}}{10-1} = \frac{57.24}{9} = 6.36$$

$$s_2^2 = \frac{(8.5)^2 + (9.6)^2 + ... + (7.4)^2 - \frac{(96.1)^2}{10}}{10-1} = \frac{24.3}{9} = 2.7$$

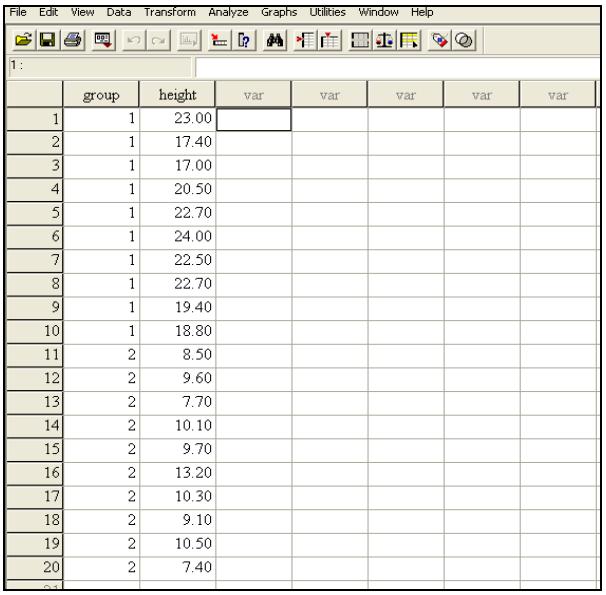
$$s^2 = \frac{(10-1)(6.36) + (10-1)(2.7)^2}{10+10-2} = \frac{57.24 + 24.43}{18} = 4.537$$

$$t = \frac{20.8 - 9.61}{\sqrt{4.737(\frac{1}{10} + \frac{1}{10})}} = 11.75$$

The computed value of t is compared with the tabular value of t (2.10) at $n_1 + n_2 - 2 = 18$ degrees of freedom. Since the computed value is greater than 2.10 and it is concluded that the populations of inoculated and uninoculated seedlings are significantly different with respect to their mean height.

The procedure for independent samples t-test using SPSS software is given below:

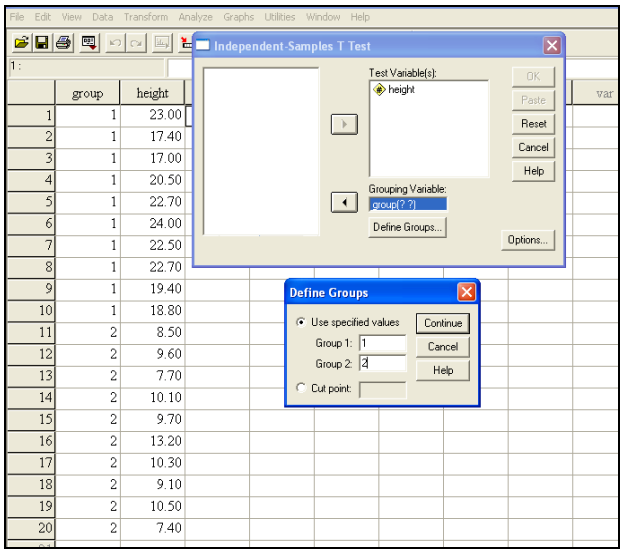
Data entry in SPSS



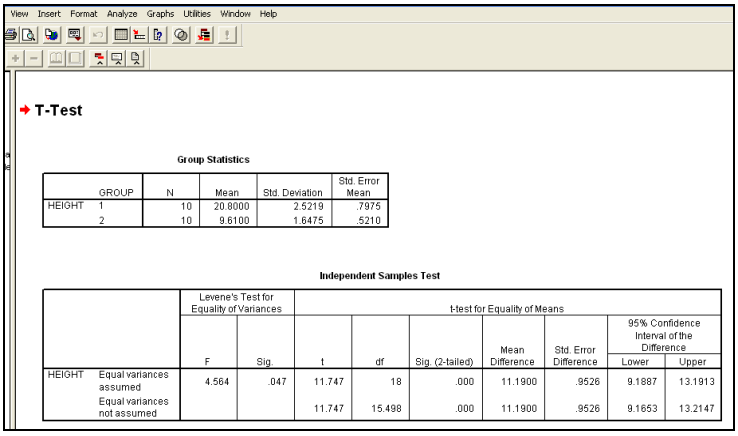
	group	height	var	var	var	var	var
1	1	23.00					
2	1	17.40					
3	1	17.00					
4	1	20.50					
5	1	22.70					
6	1	24.00					
7	1	22.50					
8	1	22.70					
9	1	19.40					
10	1	18.80					
11	2	8.50					
12	2	9.60					
13	2	7.70					
14	2	10.10					
15	2	9.70					
16	2	13.20					
17	2	10.30					
18	2	9.10					
19	2	10.50					
20	2	7.40					

Analyze → Compare Means → Independent Samples t-test

Selection of Variables



Output



T-Test

Group Statistics

	GROUP	N	Mean	Std. Deviation	Std. Error Mean
HEIGHT	1	10	20.8000	2.5219	.7975
	2	10	9.6100	1.6475	.5210

Independent Samples Test

		Levene's Test for Equality of Variances		t-test for Equality of Means						
									95% Confidence Interval of the Difference	
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
HEIGHT	Equal variances assumed	4.564	.047	11.747	18	.000	11.1900	.9526	9.1887	13.1913
	Equal variances not assumed			11.747	15.498	.000	11.1900	.9526	9.1653	13.2147

2. Paired t-Test

The following data pertain to organic carbon content measured at two different layers of a number of soil pits. Test whether the mean carbon content from two layers of soil pit differ or not.

Soil pit	Organic Carbon (%)		
	Layer 1 (x)	Layer 2 (y)	Difference (d)
1	1.59	1.21	0.38
2	1.39	0.92	0.47
3	1.64	1.31	0.33
4	1.17	1.52	-0.35
5	1.27	1.62	-0.35
6	1.58	0.91	0.67

7	1.64	1.23	0.41
8	1.53	1.21	0.32
9	1.21	1.58	-0.37
10	1.48	1.18	0.30

The observations are paired by soil pits. The paired t-test can be used in this case to compare the organic carbon status of soil at the two depth levels.

Solution: Mean of Layer 1 (μ_1) = Mean of Layer 2 (μ_2) and H_1: $\mu_1 \neq \mu_2$

$$\bar{d} = \frac{\sum_{i=1}^{n} d_i}{n} = \frac{1.81}{10} = 0.181$$

$$s_d^2 = \frac{1}{10-1}\left([(0.38)^2 + (0.47)^2 + ... + (0.30)^2] - \frac{(1.81)^2}{10} \right) = \frac{1.3379}{9} = 0.1486$$

Thus,

$$t = \frac{0.181}{\sqrt{\frac{0.1486}{10}}} = 1.485$$

The value of t (1.485) is less than the tabular value, 2.262, for 9 degrees of freedom at the 5% level of significance. It may therefore be concluded that there is no significant difference between the mean organic carbon content of the two layers of soil.

3. Equality of Several Means (Analysis of Variance)

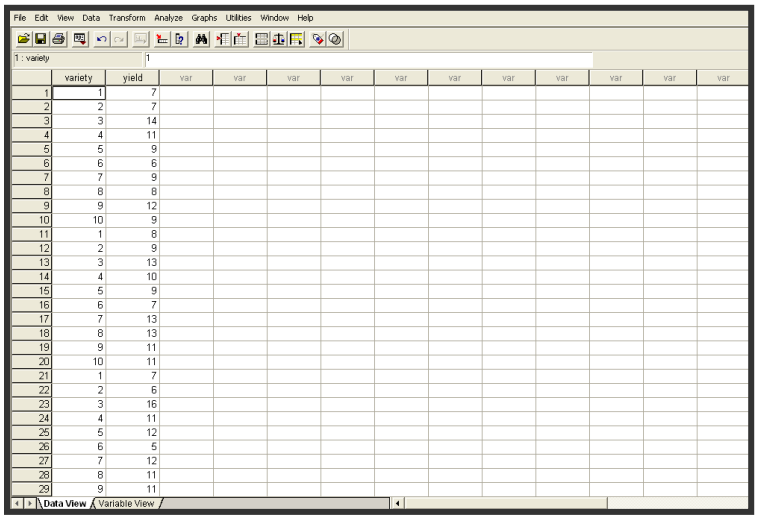
Ten varieties of wheat are grown in 3 plots each and the following yields in kg per hectare are obtained:

Variety → Plots ↓	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	7	7	14	11	9	6	9	8	12	9
2	8	9	13	10	9	7	13	13	11	11
3	7	6	16	11	12	5	11	11	11	11

Test the significance between mean variety yields.

SPSS Procedure

Data Entry

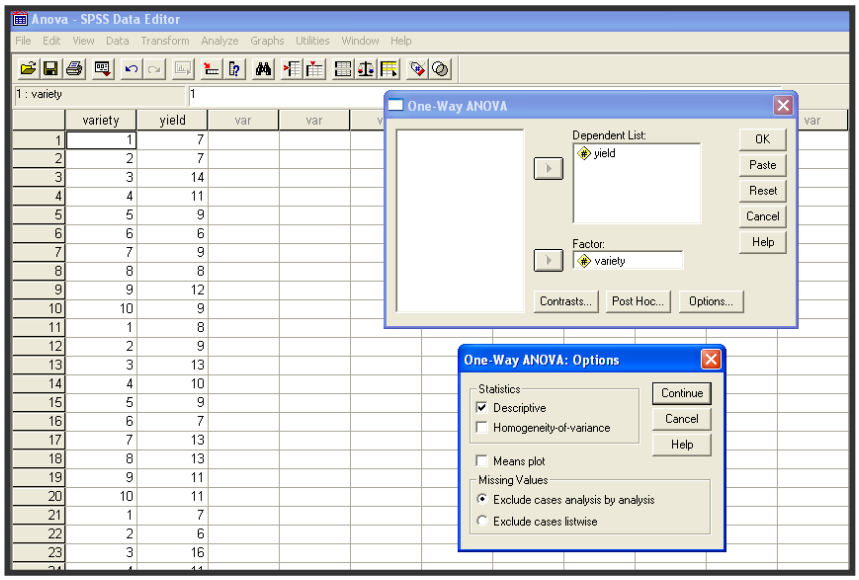


Analyze

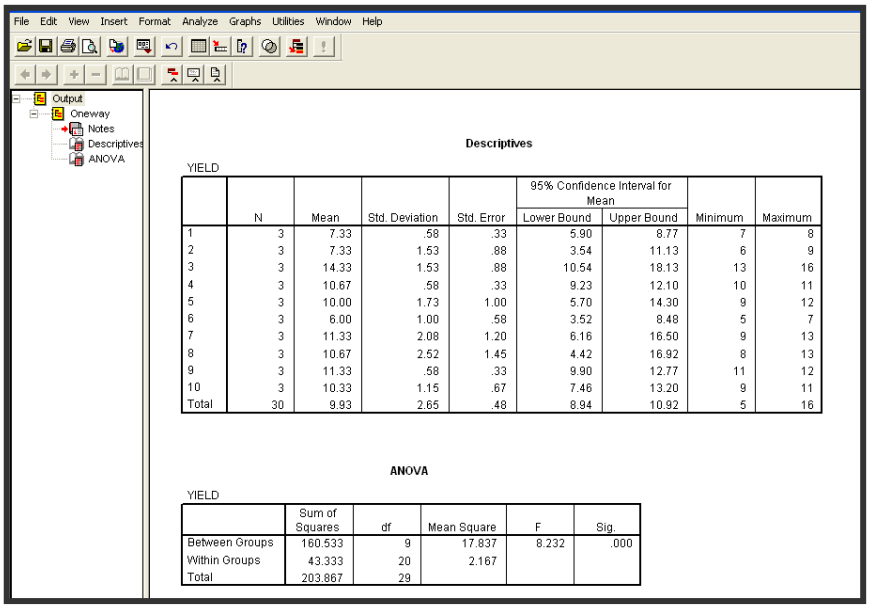
Compare Means

One-Way ANOVA….

- Select one or more dependent variables.
- Select a single independent factor variable



Output



Descriptives

YIELD

	N	Mean	Std. Deviation	Std. Error	95% Confidence Interval for Mean		Minimum	Maximum
					Lower Bound	Upper Bound		
1	3	7.33	.58	.33	5.90	8.77	7	8
2	3	7.33	1.53	.88	3.54	11.13	6	9
3	3	14.33	1.53	.88	10.54	18.13	13	16
4	3	10.67	.58	.33	9.23	12.10	10	11
5	3	10.00	1.73	1.00	5.70	14.30	9	12
6	3	6.00	1.00	.58	3.52	8.48	5	7
7	3	11.33	2.08	1.20	6.16	16.50	9	13
8	3	10.67	2.52	1.45	4.42	16.92	8	13
9	3	11.33	.58	.33	9.90	12.77	11	12
10	3	10.33	1.15	.67	7.46	13.20	9	11
Total	30	9.93	2.65	.48	8.94	10.92	5	16

ANOVA

YIELD

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	160.533	9	17.837	8.232	.000
Within Groups	43.333	20	2.167		
Total	203.867	29			

DIAGNOSTICS AND REMEDIAL MEASURES

Rajender Parsad and Lalmohan Bhar
ICAR-IASRI, Library Avenue, New Delhi – 110 012
rajender.parsad@icar.gov.in; lm.bhar@icar.gov.in

The interpretation of data based on analysis of variance (ANOVA) is valid only when the following assumptions are satisfied:

1. **Additive Effects:** Treatment effects and block (environmental) effects are additive.
2. **Independence of errors:** Experimental errors are independent.
3. **Homogeneity of Variances:** Errors have common variance.
4. **Normal Distribution:** Errors follow a normal distribution.

Also the statistical tests t, F, z, etc. are valid under the assumption of independence of errors and normality of errors. The departures from these assumptions make the interpretation based on these statistical techniques invalid. Therefore, it is necessary to detect the deviations and apply the appropriate remedial measures.

- The assumption of independence of errors, *i.e.*, error of an observation is not related to or depends upon that of another. This assumption is usually assured with the use of proper randomization procedure. However, if there is any systematic pattern in the arrangement of treatments from one replication to another, errors may be non-independent. This may be handled by using nearest neighbour methods in the analysis of experimental data.
- The assumption of additive effects can be defined and detected in the following manner:

Additive Effects

The effects of two factors, say, treatment and replication, are said to be additive if the effect of one-factor remains constant over all the levels of other factors. A hypothetical set of data from a randomized complete block (RCB) design, with 2 treatments and 2 replications, with additive effects is given in Table 1.

Table 1

Treatment	Replication		Replication Effect
	I	II	I - II
A	190	125	*65*
B	170	105	*65*
Treatment Effect (A-B)	*20*	*20*	

Here, the treatment effect is equal to 20 for both replications and replication effect is 65 for both treatments.

When the effect of one factor is not constant at all the levels of other factor, the effects are said to be non-additive. A common departure from the assumption of additivity in biological experiments is one where the effects are multiplicative. Two factors are said to have multiplicative effects if their effects are additive only when expressed in terms of percentages. Table 2 illustrates a hypothetical set of data with multiplicative effects.

Table 2

Treatment	**Replication**		**Replication Effect**	
	I	**II**	**I - II**	**100(I - II)/II**
A	200 (2.30103)	125 (2.09691)	*75* (0.20412)	*60*
B	160 (2.20412)	100 (2.0000)	*60* (0.20412)	*60*
Treatment Effect (A-B)	*40* (0.09691)	*25* (0.09691)		
100 (A - B)/B	*25*	*25*		

In this case, the treatment effect is not constant over replications and the replication effect is not constant over treatments. However, when both treatment effect and replication effect are expressed in terms of percentages, an entirely different pattern emerges. For such violations of assumptions, Logarithmic transformation is quite suitable. For illustration, the Logarithmic transformation of data in Table 2 is given in brackets.

This is, however a crude method for testing the additivity. Tukey (1949) gave a statistical test for testing the additivity in a RCB design. This test is known as one degree of freedom test for non-additivity. In this test, one degree of freedom is isolated from error and this degree of freedom is called as the degree of freedom for non-additivity. In the sequel, we describe the procedure in brief.

Suppose that an experiment has been conducted to compare v treatments using RCB design with r replications. Let y_{ij} denote the observed value of the response variable for i^{th} treatment in j^{th} replication; $i = 1,2, \ldots, v$; $j = 1,2, \ldots, r$. Arrange the data in a $v \times r$ table as given below:

Treatment	1	2	…	*j*	…	*r*	Treatment Total	Treatment Mean	Deviations from Grand Mean	Sum of Cross Product
1	y_{11}	y_{12}	$\cdots$	y_{1j}	$\cdots$	y_{1r}	$T_{1.}$	$\bar{y}_{1.}$	$d_{1.}$	C_1
2	y_{21}	y_{22}	$\cdots$	y_{2j}	$\cdots$	y_{2r}	$T_{2.}$	$\bar{y}_{2.}$	$d_{2.}$	C_2
⋮	⋮	⋮	…	⋮	…	⋮	⋮	⋮	⋮	⋮
i	y_{i1}	y_{i2}	$\cdots$	y_{ij}	$\cdots$	y_{ir}	$T_{i.}$	$\bar{y}_{i.}$	$d_{i.}$	C_i
⋮	⋮	⋮	…	⋮	…	⋮	⋮	⋮	⋮	⋮
v	y_{v1}	y_{v2}	$\cdots$	y_{vj}	$\cdots$	y_{vr}	$T_{v.}$	$\bar{y}_{v.}$	$d_{v.}$	C_v
Replication Total	$R_{.1}$	$R_{.2}$	$\cdots$	$R_{.j}$	$\cdots$	$R_{.r}$	*G* (Grand total)			
Replication Mean	$\bar{y}_{.1}$	$\bar{y}_{.2}$	$\cdots$	$\bar{y}_{.j}$	$\cdots$	$\bar{y}_{.r}$		$GM = \frac{G}{vr}$		
Deviation from Grand Mean	$d_{.1}$	$d_{.2}$	$\cdots$	$d_{.j}$	$\cdots$	$d_{.r}$				

where $T_{i.} = \sum_{j=1}^{r} y_{ij}$; $\bar{y}_{i.} = T_{i.} / r$; $R_{.j} = \sum_{i=1}^{v} y_{ij}$; $\bar{y}_{.j} = R_{.j} / v$; $d_{i.} = \bar{y}_{i.} - GM$

$d_{.j} = \bar{y}_{.j} - GM$; $C_i = \sum_{j=1}^{r} y_{ij} \times d_{.j}$

Obtain $L = \sum_{i=1}^{v} C_i d_{i.}$; $D_1 = \sum_{i=1}^{v} d_{i.}^2$; $D_2 = \sum_{j=1}^{r} d_{.j}^2$

Sum of squares due to non-additivity (SSNA)= $\frac{L^2}{D_1 \times D_2}$

The sum of squares due to treatments, replications and total sum of squares are given by

Sum of squares due to treatments (SST) = $\sum_{i=1}^{v} \frac{T_{i.}^2}{r} - \frac{G^2}{vr}$

Sum of squares due to replications (SSR) = $\sum_{j=1}^{r} \frac{R_{.j}^2}{v} - \frac{G^2}{vr}$

Total sum of squares (TSS) = $\sum_{i=1}^{v} \sum_{j}^{r} y_{ij}^2 - \frac{G^2}{vr}$

Sum of squares due to Error (SSE) = TSS – SST-SSR-SSNA

Then the outline of ANOVA table is

Source	**df**	**SS**	**MS**
Treatments	*v*-1	SST	MST
Replications	*r*-1	SSR	MSR
Non-additivity	1	SSNA	MSNA
Error	(*v*-1)(*r*-1)-1	SSE	MSE
Total	***vr*-1**	**TSS**	

The mean squares (MS) are obtained by dividing sum of squares (SS) by corresponding degrees of freedom (df). The non-additivity is tested by F-statistic with 1 and (*v*-1)(*r*-1)-1 degree of freedom calculated value of F = $\frac{\text{MSNA}}{\text{MSE}}$.

Normality of Errors

The assumptions of homogeneity of variances and normality are generally violated together. To test the validity of normality of errors for the character under study, one can take help of Normal Probability Plot, Anderson-Darling Test, D'Augstino's Test, Shapiro - Wilk's Test, Ryan-Joiner test, Kolmogrov-Smirnov test, etc. In general moderate departures from

normality are of little concern in the fixed effects ANOVA as F - test is slightly affected but in case of random effects, it is more severely impacted by non-normality. The significant deviations of errors from normality, makes the inferences invalid. So before analyzing the data, it is necessary to convert the data to a scale that it follows a normal distribution. In the data from designed field experiments, we do not directly use the original data for testing of normality or homogeneity of observations because this is embedded with the treatment effects and some of other effects like block, row, column, etc. So there is need to eliminate these effects from the data before testing the assumptions of normality and homogeneity of variances. For eliminating the treatment effects and other effects we fit the model corresponding to the design adopted and estimate the residuals. These residuals are then used for testing the normality of the observations. In other words, we want to test the null hypothesis H_0: errors are normally distributed against alternative hypothesis H_1: errors are not normally distributed. For details on these tests one may refer to D'Agostino and Stephens (1986). Most of the standard statistical packages available in the market are capable of testing the normality of the data. In SAS and SPSS commonly used tests are Shapiro-Wilk test and Kolmogrov-Smirnov test. MINITAB uses three tests viz. Anderson-Darling, Ryan-Joiner, Kolmogrov-Smirnov for testing the normality of data.

Homogeneity of Error Variances

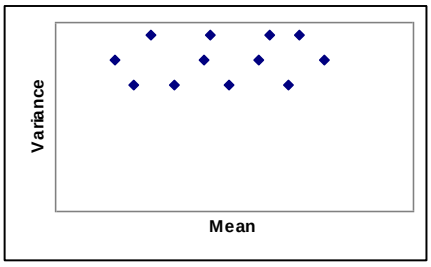
A crude method for detecting the heterogeneity of variances is based on scatter plots of means and variance or range of observations or errors, residual vs fitted values, etc. To be clearer, let Y_{ij} be the observation pertaining to i^{th} treatment $(i = 1(1)v)$ in the jth replication $(j = 1(1)r_i)$. Compute the mean and variance for each treatment across the replications (the range can be used in place of variance) as

$$\text{Mean} = \bar{Y}_{i.} = \frac{1}{r_i}\sum_{j=1}^{r_i} Y_{ij}\ ; \qquad \text{Variance} = S_i^2 = \frac{1}{r_i - 1}\sum_{j=1}^{r_i}\left(Y_{ij} - \bar{Y}_{i.}\right)^2$$

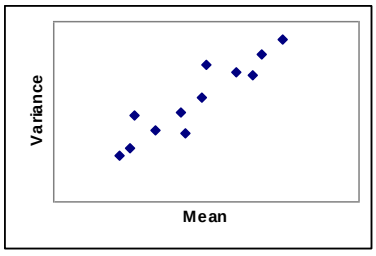
Draw the scatter plot of mean vs variance (or range). If $S_{i.}^2$'s $(i = 1(1)v)$ are equal (constant) or nearly equal, then the variances are homogeneous. Based on these scatter plots, the heterogeneity of variances can be classified into two types:

1. Where the variance is functionally related to mean.
2. Where there is no functional relationship between the variance and the mean.

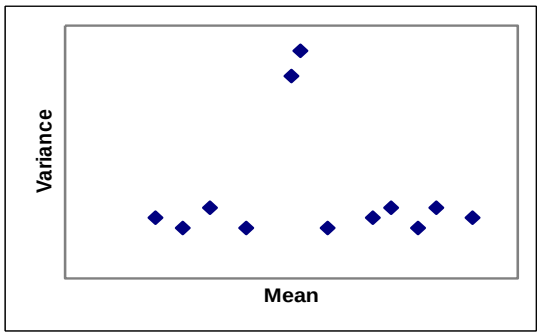
For illustration some scatter - diagrams of mean and variances (or range) are given as:



(a) Homogeneous variance



(b) Heterogeneous variance where variance is proportional to mean



(c) Heterogeneous variance without any functional relationship between variance and mean

The first kind of variance heterogeneity (figure b) is usually associated with the data whose distribution is non-normal *viz.*, negative binomial, Poisson, binomial, etc. The second kind of variance heterogeneity usually occurs in experiments, where, due to the nature of treatments tested, some treatments have errors that are substantially higher (lower) than others. For example, in varietal trials, where various types of breeding material are being compared, the size of variance between plots of a particular variety will depend on the degree of genetic homogeneity of material being tested. The variance of F_2 generation, for example, can be expected to be higher than that of F_1 generation because genetic variability in F_2 is much higher than that in F_1. The variances of varieties that are highly tolerant of or highly susceptible to, the stress being tested are expected to be smaller than those of having moderate degree of tolerance. Also in testing yield response to a chemical treatment, such as, fertilizer, insecticide or herbicide, the non-uniform application of chemical treatments may result in a higher variability in the treated plots than that in the untreated plots.

The scatter-diagram of means and variances of observations for each treatment across the replications gives only a preliminary idea about homogeneity of error variances. Statistically the homogeneity of error variances is tested using Bartlett's test for normally distributed errors and Levene test for non-normal errors. These tests are described in the sequel.

Bartlett's Test for Homogeneity of Variances

Let there are *v*- independent samples drawn from same population and i^{th} sample is of size r_i and $(r_1 + r_2 + ... + r_v) = N$. In the present case, the independent samples are the residuals of the observations pertaining to *v* treatments and i^{th} sample size is the number of replications of the treatment *i*. One wants to test the null hypothesis $H_0 : \sigma_1^2 = \sigma_2^2 = ... = \sigma_v^2$ against the alternative hypothesis H_1 : at least two of the σ_i^2'*s* are not equal, where σ_i^2 is the error variance for treatment *i*.

Let e_{ij} denotes the residual pertaining to the observation of treatment *i* from replication *j*, then it can easily be shown that the sum of residuals pertaining to a given treatment is zero. In

this test $S_i^2 = \frac{1}{r_i - 1}\sum_{j=1}^{r_i}(e_{ij} - \bar{e}_{i.})^2 = \frac{1}{r_i - 1}\sum_{j=1}^{r_i} e_{ij}^2$ is taken as unbiased estimate of σ_i^2. The procedure involves computing a statistic whose sampling distribution is closely approximated by the χ^2 distribution with *v - 1* degrees of freedom. The test statistic is

$$\chi_0^2 = 2.3026\frac{q}{c}$$

and null hypothesis is rejected when $\chi_0^2 > \chi_{\alpha,v-1}^2$, where $\chi_{\alpha,v-1}^2$ is the upper α percentage point of χ^2 distribution with *v - 1* degrees of freedom.

To compute χ_0^2, follow the steps:

Step 1: Compute mean and variance of all *v*-samples.

Step 2: Obtain pooled variance $S_p^2 = \frac{\sum_{i=1}^{v}(r_i - 1)S_i^2}{N - v}$

Step 3: Compute $q = (N - v)\log_{10} S_p^2 - \sum_{i=1}^{v}(r_i - 1)\log_{10} S_i^2$

Step 4: Compute $c = 1 + \frac{1}{3(v-1)}\left(\sum_{i=1}^{v}(r_i - 1)^{-1} - (N - v)^{-1}\right)$

Step 5: Compute χ_0^2.

Bartlett's χ^2 test for homogeneity of variances is a modification of the normal-theory likelihood ratio test. While Bartlett's test has accurate Type I error rates and optimal power when the underlying distribution of the data is normal, it can be very inaccurate if that distribution is even slightly non-normal (Box 1953). Therefore, Bartlett's test is not recommended for routine use.

An approach that leads to tests that are much more robust to the underlying distribution is to transform the original values of the dependent variable to derive a *dispersion variable* and then to perform analysis of variance on this variable. The significance level for the test of homogeneity of variance is the *p*-value for the ANOVA *F*-test on the dispersion variable. Commonly used test for testing the homogeneity of variance using a dispersion variable is Levene Test given by Levene (1960). The procedure is described in the sequel.

Levene Test for homogeneity of Variances

The test is based on the variability of the residuals. The larger the error variance, the larger the variability of the residuals will tend to be. To conduct the Levene test, we divide the data into different groups based on the number of treatments if the error variance is either increasing or

decreasing with the treatments, the residuals in the one treatment will tend to be more variable than those in others treatments. The Levene test than consists simply F – statistic based on one way ANOVA used to determine whether the mean of absolute/ Square root deviation from mean are significantly different or not. The residuals are obtained from the usual analysis of variance. The test statistic is given as

$$F = \frac{\left\{\sum_{i=1}^{v}(r_i - 1)\right\}}{v-1} \frac{\left\{\sum_{i=1}^{v} r_i(\bar{d}_{i.} - \bar{d}_{..})^2\right\}}{\sum_{i=1}^{v}\sum_{j=1}^{r_i}(d_{ij} - \bar{d}_{i.})^2} \sim F((v-1), \sum_{i=1}^{v}(r_i - 1)$$

where $d_{ij} = \left|e_{ij} - \bar{e}_i\right|$; $\bar{d}_{i.} = \frac{\sum_{j=1}^{r_i} d_{ij}}{r_i}$; $\bar{d}_{..} = \frac{\sum_{i=1}^{v}\sum_{j=1}^{r_i} d_{ij}}{\sum_{j=1}^{r_i} r_i}$ and e_{ij} is the jth residual for the ith plot, $\bar{e}_i$ is the mean of the residuals of the ith treatment.

This test was modified by Brown and Forsythe (1974). In the modified test, the absolute deviation is taken from the median instead of mean in order to make the test more robust.

In the present investigation, the Bartlett's χ^2-test has been used for testing the homogeneity of error variances when the distribution of errors is normal and Levene test for non-normal errors.

Remark 1: In a block design**,** it can easily be shown that the sum of residuals within a given block is zero. Therefore, the residuals in a block of size 2 will be same with their sign reverse in order. As a consequence, *q* in Bartlett's test and numerator in Levene test statistic becomes zero for the data generated from experiments conducted to compare only two treatments in a RCB design. Hence, the tests for homogeneity of error variances cannot be used for the experiments conducted to compare only two treatments in a RCB design. Inferences from such experiments may be drawn using Fisher-Behren t-test. Further, Bartlett's test cannot be used for the experimental situations where some of the treatments are singly replicated.

Remark 2: In a RCB design, it can easily be shown that the sum of residuals from a particular treatment is zero. As a consequence, the denominator of Levene test statistic is zero for the data generated from RCB designs with two replications. Therefore, Levene test cannot be used for testing the homogeneity of error variances for the data generated from RCB designs with two replications.

Remedial Measures

In this section, we shall discuss the remedial measures for non-normal and/or heterogeneous data in greater details.

Data transformation is the most appropriate remedial measure, in the situation where the variances are heterogeneous and are some functions of means. With this technique, the original data are converted to a new scale resulting into a new data set that is expected to satisfy the homogeneity of variances. Because a common transformation scale is applied to all observations, the comparative values between treatments are not altered and comparison between them remains valid.

Error partitioning is the remedial measure of heterogeneity that usually occurs in experiments, where, due to the nature of treatments tested some treatments have errors that are substantially higher (lower) than others.

Here, we shall concentrate on those situations where character under study is non-normal and variances are heterogeneous. Depending upon the functional relationship between variances and means, suitable transformation is adopted. The transformed variate should satisfy the following:

1. The variances of the transformed variate should be unaffected by changes in the means. This is also called the variance stabilizing transformation.
2. It should be normally distributed.
3. It should be one for which effects are linear and additive.
4. The transformed scale should be such for which an arithmetic average from the sample is an efficient estimate of true mean.

The following are the three transformations, which are being used most commonly, in biological research.

a) Logarithmic Transformation
b) Square root Transformation
c) Arc Sine or Angular Transformation

a) Logarithmic Transformation

This transformation is suitable for the data where the variance is proportional to square of the mean or the coefficient of variation (S.D./mean) is constant or where effects are multiplicative. These conditions are generally found in the data that are whole numbers and cover a wide range of values. This is usually the case when analyzing growth measurements such as trunk girth, length of extension growth, weight of tree or number of insects per plot, number of eggmass per plant or per unit area etc.

For such situations, it is appropriate to analyze *log X* instead of actual data, *X*. When data set involves small values or zeros, *log (X+1), log(2X +1)* or $log\left(X+\frac{3}{8}\right)$ should be used instead of *log X*. This transformation would make errors normal, when observations follow negative binomial distribution like in the case of insect counts.

b) Square-Root Transformation

This transformation is appropriate for the data sets where the variance is proportional to the mean. Here, the data consists of small whole numbers, for example, data obtained in counting rare events, such as the number of infested plants in a plot, the number of insects caught in

traps, number of weeds per plot, parthenocarpy in some varieties of mango. This data set generally follows the Poisson distribution and square root transformation approximates Poisson to normal distribution.

For these situations, it is better to analyze $\sqrt{X}$ than that of *X*, the actual data. If X is confirmed to small whole numbers then, $\sqrt{X+\frac{1}{2}}$ or $\sqrt{X+\frac{3}{8}}$ should be used instead of $\sqrt{X}$.

This transformation is also appropriate for the percentage data, where, the range is between 0 to 30% or between 70 and 100%.

c) Arc Sine Transformation

This transformation is appropriate for the data on proportions, *i.e.*, data obtained from a count and the data expressed as decimal fractions and percentages. The distribution of percentages is binomial and this transformation makes the distribution normal. Since the role of this transformation is not properly understood, there is a tendency to transform any percentage using arc sine transformation. But only that percentage data that are derived from count data, such as % barren tillers (which is derived from the ratio of the number of non-bearing tillers to the total number of tillers) should be transformed and not the percentage data such as % protein or % carbohydrates, %nitrogen, etc. which are not derived from count data. For these situations, it is better to analyze $\sin^{-1}(\sqrt{X})$ than that of *X*, the actual data. If the value of X is 0%, it should be substituted by $\left(\frac{1}{4n}\right)$ and the value of 100% by $\left(100-\frac{1}{4n}\right)$, where *n* is the number of units upon which the percentage data is based.

It is interesting to note here that not all percentage data need to be transformed and even if they do, arc sine transformation is not the only transformation possible. The following rules may be useful in choosing the proper transformation scale for percentage data derived from count data.

Rule 1: The percentage data lying within the range 30 to 70% is homogeneous and no transformation is needed.

Rule 2: For percentage data lying within the range of either 0 to 30% or 70 to 100%, but not both, the square root transformation should be used.

Rule 3: For percentage that do not follow the ranges specified in Rule 1 or Rule 2, the Arc Sine transformation should be used.

The other transformations used are reciprocal square root [$\frac{1}{\sqrt{X}}$, when variance is proportional to cube of mean], reciprocal [$\frac{1}{X}$, when variance is proportional to fourth power of mean] and tangent hyperbolic transformation.

The transformation discussed above are a particular case of the general family of transformations known as Box-Cox transformation.

d) Box-Cox Transformation

By now we know that if the relation between the variance of observations and the mean is known then this information can be utilized in selecting the form of the transformation. We now elaborate on this point and show how it is possible to estimate the form of the required transformation from the data. The transformation suggested by Box and Cox (1964) is a power transformation of the original data. Let y_{ut} be the observation pertaining to the uth plot; then the power transformation implies that we use y_{ut}*'s* as

$$y_{ut}^{*} = y_{ut}^{\lambda}.$$

The transformation parameter λ in $y_{ut}^{*} = y_{ut}^{\lambda}$ may be estimated simultaneously with the other model parameters (overall mean and treatment effects) using the method of maximum likelihood. The procedure consists of performing, for the various values of λ, a standard analysis of variance on

$$y_{ut}^{(\lambda)} = \begin{cases} \dfrac{y_{ut}^{\lambda} - 1}{\lambda \dot{y}_{ut}^{\lambda-1}} & \lambda \neq 0 \\ \dot{y}_{ut} \ln y_{ut} & \lambda = 0 \end{cases} \qquad \text{(A)}$$

where $\dot{y}_{ut} = \ln^{-1}\left[(1/n) \sum_{u=1}^{N} \sum_{t=1}^{n_u} \ln y_{ut} \right]$.

$\dot{y}_{ut}$ is the geometric mean of the observations. The maximum likelihood estimate of λ is the value for which the error sum of squares, say SSE (λ), is minimum. Notice that we cannot select the value of λ by directly comparing the error sum of squares from analysis of variance on y^{λ} because for each value of λ the error sum of squares is measured on a different scale. Equation (A) rescales the responses so that the error sums of squares are directly comparable. This is a very general transformation and the commonly used transformations follow as particular cases. The particular cases for different values of λ are given below.

λ	Transformation
1	No Transformation
½	Square Root
0	Log
-1/2	Reciprocal Square Root
-1	Reciprocal

Remark 3: If any one of the observations is zero then the geometric mean is undefined. In the expression (A), geometric mean is in denominator so it is not possible to compute that expression. For solving this problem, we add a small quantity to each of the observations.

Note: It should be emphasized that transformation, if needed, must take place right at the beginning of the analysis, all fitting of missing plot values, all adjustments by covariance etc. being done with the transformed variate and not with the original data. At the end, when the conclusions have been reached, it is permissible to 're-transform' the results so as to present them in the original units of measurement, but this is done only to render them more intelligible.

As a result of this transformation followed by back transformation, the means will rather be different from those that would have been obtained from the original data. A simple example is that without transformation, the mean of the numbers 1, 4, 9, 16 and 25 is 11. Suppose a square root transformation is used to give 1, 2, 3, 4 and 5, the mean is now 3, which after back- transformation gives 9. Usually the difference will not be so great because data do not usually vary as much as those given, but logarithmic and square root transformation always lead to a reduction of the mean, just as angles of equal formation usually lead to its moving away from the central value of 50%.

However, in practice, computing treatment means from original data is more frequently used because of its simplicity, but this may change the order of ranking of converted means for comparison. Although transformations make possible a valid analysis, they can be very awkward. For example, although a significant difference can be worked out in the usual way for means of the transformed data, none can be worked out for the treatment means after back transformation.

Non-parametric tests in the Analysis of Experimental Data

When the data remains non-normal and/or heterogeneous even after transformation, a recourse is made to non-parametric test procedures. A lot of attention is being paid to develop non-parametric tests for analysis of experimental data. Most of these non-parametric test procedures are based on rank statistic. The rank statistic has been used in development of these tests as the statistic based on ranks is

1. distribution free
2. easy to calculate and
3. simple to explain and understand.

Another reason for use of rank statistic is due to the well known result that the average rank approaches normality quickly as *n* (number of observations) increases, under the rather general conditions, while the same might not be true for the original data {see e.g. Conover and Iman (1976, 1981)}. The non-parametric test procedures available in literature cover completely randomized designs, randomized complete block designs, balanced incomplete block designs, design for bioassays, split plot designs, cross-over designs and so on. For an excellent and elaborate discussions on non-parametric tests in the analysis of experimental data, one may refer to Siegel and Castellan Jr. (1988), Deshpande, Gore and Shanubhogue (1995), Sen (1996), and Hollander and Wolfe (1999).

Kruskal-Wallis Test can be used for the analysis of data from completely randomized designs. Skillings and Mack Test helps in analyzing the data from a general block design. Friedman Test and Durbin Test are particular cases of this test. Friedman Test is used for the analysis of data from randomized complete block designs and Durbin test for the analysis of data from balanced incomplete block designs.

Some examples of testing the assumptions of normality and homogeneity of errors and remedial measures are discussed in the Appendix.

Some Useful References

Anderson,V.L. and McLean,R.A.(1974). *Design of Experiments: A realistic approach*. Marcel Dekker Inc., New York.

Bartlett,M.S.(1947). The use of transformation. *Biometrics*, **3**, 39-52.

Brown, M. B. and Forsythe, A.B. (1974). Robust tests for the equality of variances. *Journal of the American Statistical Association*, **69**, 364-367.

Box, G.E.P., (1953). Non-normality and tests on variances. *Biometrika*, **40** (**1** & **2**), 318-335.

Box,G.E.P. and Cox,D.R.(1964). An analysis of transformation. *J. Roy. Stastist. Soc*. **B**, **26**, 211-252.

Conover, W.J. and Iman, R.L. (1976). In Some Alternative Procedure Using Ranks for the Analysis of Experimental Designs. *Comm. Statist.: Theory & Methods*, **A5(14)**, 1349-1368.

Conover,W.J. and Iman,R.L.(1981). Rank transfomations as a bridge between parametric and non-parametric statistics (with discussion). *American Statistician*, **35**, 124-133.

D'Agostino, R.B. and Stephens, M. A.(1986). *Goodness-of-fit Techniques*. Marcel Dekkar, Inc., New York.

Deshpande, J.V., Gore, A.P. and Shanubhogue, A. (1995). *Statistical Analysis of Non-normal Data*. Wiley Eastern Limited, New Delhi.

Dean, A and Voss, D (1999). *Design and Analysis of Experiments*. Springer, New York.

Dolby,J.L.(1963). A quick method for choosing a transformation. *Technometrics*, **5**, 317-326.

Draper,N.R. and Hunter,W.G.(1969). Transformations: Some examples revisited. *Technometrics*, **11**, 23-40.

Hollander, M. and Wolfe, D.A. (1999). *Nonparametric Statistical Methods*, 2nd Ed, John Wiley, New York.

Levene, H., (1960). Robust Tests for Equality of Variances. **I** Olkin, ed., *Contributions to Probability and Statistics*, Palo Alto, Calif.: Stanford University Press, 278-292.

Royston, P. (1992), Approximating the Shapiro-Wilk W-Test for non-normality. *Statistics and Computing*, **2**, 117 -119.

Shapiro, S. S. and Wilk, M. B.. (1965). An analysis of variances test for normality (Complete Samples). *Biometrika*, **52**, 591-611

Siegel, S. and Catellan, N.J. Jr. (1988*). Non parametric statistics for the Behavioural Sciences*, 2nd Ed., McGraw Hill, New York.

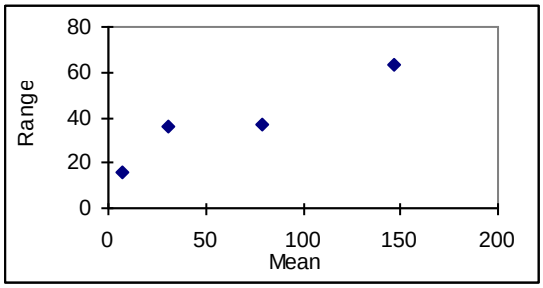
Tukey, J.W.(1949). One degree of freedom for non-additivity. *Biometrics*, **5**, 232-242.

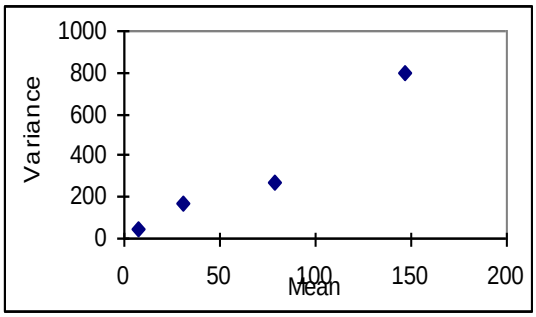
Appendix

Example 1: Suppose an entomologist is interested in determining whether four different kinds of traps caught equivalent insects when applied to same field. Each of the traps is used six times on the field and resulting data (number of insects per hour) are as shown below alongwith mean, variance and range.

Treatment	Replication						Mean	Variance	Range
	I	II	III	IV	V	VI	$\overline{Y}_i$	S_i^2	
A	3	1	12	7	17	2	7	40.4	16
B	9	29	21	24	28	45	31	138.4	36
C	63	84	97	61	98	71	79	270.8	37
D	172	118	109	172	143	168	147	798.4	63

A scatter plot of mean and variance and mean versus range are given as follows:





Both plots indicate that variances are heterogeneous and variance is proportional to mean.

Obtain the residuals for testing the normality and homogeneity of error terms. The residuals obtained are given below:

Treatment	Replication						Mean	Variance
	I	II	III	IV	V	VI		S_i^2
A	-1.00	0.75	10.00	-1.25	3.25	-11.75	0	50.35
B	-14.00	9.75	0.00	-3.25	-4.75	12.25	0	94.85
C	-13.00	11.75	23.00	-19.25	12.25	-14.75	0	314.85
D	28.00	-22.25	-33.00	23.75	-10.75	14.25	0	650.20

Normality of error terms in SPSS: Analyze → Descriptive Statistics→ Explore→ (Enter the variable whose normality is to be tested in Dependent List) → Click on the option Plots → Check the option Normality Plots with tests

Shapiro-Wilk Test		**Kolmogrov-Smirnov Test**	
Statistic (SW)	**p-value**	**Statistic (KS)**	**p-value**
0.980	0.882	0.110	0.200

The errors were found to be normally distributed. Therefore, homogeneity of error variances was tested using Bartlett's test. It is described in the sequel.

Pooled Variance $\left(S_p^2\right) = \frac{5(50.35+94.85+314.85+650.20)}{20} = 277.5625$

$$q = 20\log_{10} 277.5625 - 5[\log_{10} 50.35 + \log_{10} 64.85 + \log_{10} 314.85 + \log_{10} 650.20]$$
$$= 3.916278$$

$$c = 1 + \frac{1}{9}\left(\frac{4}{5} - \frac{1}{20}\right) = 1.08333$$

$\chi_0^2 = 8.324$.

Since $\chi^2_{0.05,3} = 7.81$, therefore, we reject the null hypothesis and conclude that the variances are unequal.

The $\frac{S_i^2}{\bar{Y}_{i.}}$ are 5.77, 5.32, 3.43 and 5.43, indicating that variance is proportional to mean. Therefore, square root transformation should be used. After application of square root transformation, the residuals are

Treatment	Replication						Variance
	I	II	III	IV	V	VI	S_i^2
A	-0.03614	-0.92542	1.05800	0.20614	0.98287	-1.28544	0.928
B	-1.34939	0.87854	-0.40473	-0.12183	-0.42993	1.42735	0.999
C	-0.28226	0.78841	0.99143	-1.08068	0.30794	-0.72483	0.694
D	1.66779	-0.74153	-1.64469	0.99637	-0.86087	0.58293	1.622

Normality of error terms on the transformed data:

Shapiro-Wilk Test		**Kolmogrov-Smirnov Test**	
Statistic (SW)	**p-value**	**Statistic (KS)**	**p-value**
0.956	0.414	0.127	0.200

The errors remain normally distributed after transformation. The results of homogeneity of error variances using Bartlett's test are

Bartlett's Test (normal distribution): Test statistic = 0.89, p-value = 0.828
Hence, we conclude that the errors are normally distributed and have a constant variance after transformation.

The results of analysis of variance with original and transformed data are given in the sequel.

ANOVA: Original Data

Source	DF	Seq SS	Adj. SS	Mean Square	F (F-calc)	p(Pr>F)
Replication	5	689.0	689.0	137.8	0.37	0.86
Treatment	3	70828.5	70828.5	23609.5	63.80	0.00
Error	15	5551.0	5551.0	370.1		
Total	23	77068.5				

R-Square	Root MSE
92.80%	19.2371

Tukey Simultaneous Tests for All Pairwise Treatment Comparisons

	1	2	3	4
1	.			
2	0.3525	.		
3	0.0001	0.0013	.	
4	0.0000	0.0000	0.0001	.

ANOVA: Transformed Data

Source	DF	Seq SS	Adj. SS	Mean Square	F (F-calc)	p(Pr>F)
Replication	5	5.055	5.055	1.011	0.71	0.622
Treatment	3	326.603	326.603	108.868	76.98	0.000
Error	15	21.214	21.214	1.414		
Total	23	352.872				

R-Square	Root MSE
93.99%	1.18922

Tukey Simultaneous Tests for All Pairwise Treatment Comparisons

	1	2	3	4
1	.			
2	0.0091			
3	0.0000	0.0003		
4	0.0000	0.0000	0.0015	.

With transformed data treatments 1 and 2 are significantly different whereas with original data, they were not.

Example 2: A varietal trial on Rapeseed-Mustard was conducted at Faizabad with 11 varieties using a randomized complete block design with 3 replications. The experimental data (Yield in kg/ha) obtained from the above experiment is

Treatments ↓	**Replications→**		
	R1	**R2**	**R3**
MCN-157	952.380	1058.200	1079.364
MCN-158	846.560	634.920	687.830
MCN-159	529.100	687.830	687.830
MCN-160	1058.200	1005.290	952.380
MCN-161	1111.110	888.888	846.560
MCN-162	899.470	634.920	1005.290
MCN-163	1058.200	1164.020	952.380
MCN-164	687.830	740.740	529.100
MCN-165	952.380	952.380	867.724
MCN-166	1058.200	1058.200	529.100
MCN-167	1269.840	1164.020	1216.930

The analysis of variance of the original data is given as

ANOVA: Original Data

Sources	**DF**	**SS**	**MS**	**F**	**Prob. >F**
Replication	2	52534.9880	26267.4940	1.46	0.2563
Treatment	10	967055.0471	96705.5047	5.37	0.0007
Error	20	360218.589	18010.929		
Total	**32**	1379808.624			

R-Square	**CV**	**RMSE**	**Yld Mean**
0.738936	14.878	134.2048	902.035

Normality of error terms was tested, the results are given as

Shapiro-Wilk Test		**Kolmogrov-Smirnov Test**	
Statistic (SW)	**p-value**	**Statistic (KS)**	**p-value**
0.9679	0.4249	0.1018	>0.1500

Since the data is normal, therefore, Bartlett's test is used for testing the homogeneity of error variances. The results are given as

Bartlett's Test
Test Statistic : 20.177
P-Value : 0.0276

The errors were found to be heterogeneous.

Therefore, we can conclude that the data is heterogeneous and normal.

Therefore, Box-Cox transformation was used as a remedial measure. In the sequel we describe the results of the Box-Cox transformation.

For this we transform the data by varying λ from -10 to +10 with an increment of 0.01. The error sum of squares are computed for each value of λ. The value of λ with minimum error sum of squares is used for transformation given in (A). The minimum value SSE is obtained for λ = 2.38. Therefore, reciprocal transformation was used.

The assumptions of normality and homogeneity of errors are again tested using the transformed data.

Normality of error terms was tested, the results are given as

Shapiro-Wilk Test		**Kolmogrov-Smirnov Test**	
Statistic (SW)	**p-value**	**Statistic (KS)**	**p-value**
0.984	0.8885	0.0867	>0.1500

Since the data is normal, therefore, Bartlett's Test is used for testing the homogeneity of error variances. The results are given as

Bartlett's Test (normal distribution)
Test Statistic : 15.725
P-Value : 0.107757

The transformed observations were found to be normal and homogeneous Therefore, ANOVA was performed on the transformed data. The results obtained are:

ANOVA: Transformed Data

Sources	DF	SS	MS	F	Prob. >F
Replication	2	3.865471E13	1.93273335E13	1.62	0.2238
Treatment	10	7.8841391E14	7.8841391E13	6.59	0.0002
Error	20	2.3934391E14	1.1967195E13		
Total	**32**	**1.0664125E15**			

R-Square	CV	RMSE	Transformed Yld Mean
0.7756	29.563	3459363	11701777

We can see that there is no change in the results of significance of treatment and replication effects. However, the transformed data satisfied the assumptions of ANOVA.

PLANNING OF EXPERIMENTS AND BASIC EXPERIMENTAL DESIGNS

SEEMA JAGGI and ANINDITA DATTA
ICAR-Indian Agricultural Statistics Research Institute
seema@iasri.res.in, anindita.datta@icar.gov.in

An experiment is usually associated with a scientific method for testing certain phenomena. An experiment facilitates the study of such phenomena under controlled conditions and thus creating controlled condition is an essential component. Scientists in the biological fields who are involved in research constantly face problems associated with planning, designing and conducting experiments. Basic familiarity and understanding of statistical methods that deal with issues of concern would be helpful in many ways. Researchers who collect data and then look for a statistical technique that would provide valid results will find that there may not be solutions to the problem and that the problem could have been avoided first by a properly designed experiment. Obviously it is important to keep in mind that we cannot draw valid conclusions from poorly planned experiments. Second, the time and cost involved in many experiments are enormous and a poorly designed experiment increases such costs in time and resources. For example, an agronomist who carries out fertilizer experiment knows the time limitation of the experiment. He knows that when seeds are to be planted and harvested. The experimenter plot must include all components of a complete design. Otherwise what is omitted from the experiment will have to be carried out in subsequent trials in the next cropping season or next year. The additional time and expenditure could be minimized by a properly planned experiment that will produce valid results as efficiently as possible. Good experimental designs are products of the technical knowledge of one's field, an understanding of statistical techniques and skill in designing experiments.

Any research endeavor may entail the phases of Conception, Design, Data collection, Analysis and Dissemination. Statistical methodologies can be used to conduct better scientific experiments if they are incorporated into entire scientific process, i.e., From inception of the problem to experimental design, data analysis and interpretation. When planning experiments we must keep in mind that large uncontrolled variations are common occurrences. Experiments are generally undertaken by researchers to compare effects of several conditions on some phenomena or in discovering an unknown effect of particular process. An experiment facilitates the study of such phenomena under controlled conditions. Therefore the creation of controlled condition is the most essential characteristic of experimentation. How we formulate our questions and hypotheses are critical to the experimental procedure that will follow. For example, a crop scientist who plants the same variety of a crop in a field may find variations in yield that are due to periodic variations across a field or to some other factors that the experimenter has no control over. The methodologies used in designing experiments will separate with confidence and accuracy a varietal difference of crops from the uncontrolled variations.

The different concepts in planning of experiment can be well explained through chapati tasting

experiment.

Consider an experiment to detect the taste difference in chapati made of wheat flour of c306 and pv 18 varieties. The null hypothesis we can assume here is that there is no taste difference in chapatis made of c306 or pv18 wheat flours. After the null hypothesis is set, we have to fix the level of significance at which we can operate. The pv18 is a much higher yielding variety than c306. Hence a false rejection may not help the country to grow more pv18 and the wheat production may decrease while a false acceptance may give more production of pv18 wheat and the consumption may be less or practically nil. Thus the false acceptance or false rejection are of practically equal consequence and we agree to choose the level of significance at $\alpha = 0.05$. Now to execute the experiment, a subject is to be found with extrasensory powers who can detect the taste differences. The colours of c306 and pv18 are different and anyone, even without tasting the chapatis, can distinguish the chapatis of either kind by a mere glance. Thus the taster of the chapatis has to be blindfolded before the chapatis are given for tasting. Afterwards, the method is to be decided in which the experiment will be conducted. The experiment can be conducted in many ways and of them three methods are discussed here:

- Give the taster equal number of chapatis of either kind informing the taster about it.
- Give the taster pairs of chapatis of each kind informing the taster about it.
- Give the taster chapatis of either kind without providing him with any information. Let us use 6 chapatis in each of these methods.

Under first method of experimentation, if the null hypothesis is true, then the experimenter cannot distinguish the two kinds of chapaties and he will randomly select 3 chapatiS out of 6 chapaties given to him, as made of pvl8 wheat. In that case, all correct guesses are made if selection exactly coincides with the exactly used wheat variety and the probability for such an occurrence is:

$$1/\binom{6}{3} = 1/20 = 0.05$$

Under second method,the pv18 wheat variety chapaties are selected from each pair given if the null hypothesis is true. Furthermore, independent choices are made of pv18 variety chapaties from each pair. Thus the probability of making all correct guesses is

$$1/(2)^3 = 1/8 = 0.125.$$

In third method the experimenter has to make the choice for each chapati and the situation is analogous at calling heads or tails in a coin tossing experiment. The probability of making all correct guesses would then be:

$$1/2^6 = 1/64 = .016.$$

If the experimenter makes all correct guesses in third method as its probability is smaller than the selected $\alpha = 0.05$, we can reject the null hypothesis and conclude that the two wheat varieties give different tastes at chapaties. In other methods the probability of making all correct guesses

does not exceed α = 0.05 and hence with either method, we cannot reject the null hypothesis even if all correct guesses are made.
However, if 8 chapaties are used by first method and if the taster guesses all of them, we can reject the null hypothesis, at 0.05 level of significance, as the probability of making all correct guesses would then be $1/\binom{8}{3} = 1/56$ which is smaller than 0.05. 8 chapaties will not enable us to reject the null hypothesis even if all correct guesses are made by second method as the probability of making all correct guesses is $\left(\frac{1}{4}\right)^4 = \frac{1}{16} = 0.06$ it is easy to see that if 10 chapaties are given by second method and if all correct guesses are made, then we can reject the null hypothesis at 0.05 level of significance. Not to unduly influence the taster in making guesses, we should also present the chapaties in a random order rather than systematically presenting them for tasting.

The above discussed chapati tasting experiment brings home the following salient features of experimentation:

- All the extraneous variations in the data should be eliminated or controlled excepting the variations due to the treatments under study. One should not artificially provide circumstances for one treatment to show better results than others.
- Far a given size of the experiment, though the experiment can be done in many ways, even the best results may not turn out to be significant with some designs, while some other design can detect the treatment differences. Thus there is an imperative need the choose the right type of design, before the commencement of the experiment, lest the results may be useless.
- If for some specific reasons related to the nature .of the experiment, a particular method has to be used in experimentation, then adequate number of replications of each treatment have to be provided in order to get valid inferences.
- The treatments have to be randomly allocated to the experimental units.

The terminologies often used in planning and designing of experiments are listed below.

Treatment
Treatment refers to controllable quantitative or qualitative factors imposed at a certain level by the experimenter. For an agronomist several fertilizer concentrations applied to a particular crop or a variety of crop is a treatment. Similarly, an animal scientist looks upon several concentrations of a drug given to animal species as a treatment. In agribusiness we may look upon impact of advertising strategy on sales a treatment. To an agricultural engineer, different levels of irrigation may constitute a treatment.

Experimental Unit
An experimental unit is an entity that receives a treatment e.g., for an agronomist or horticulturist it may be a plot of a land or batch of seed, for an animal scientist it may be a group of pigs or

sheep, for a scientist engaged in forestry research it may be different tree species occurring in an area, and for an agricultural engineer it may be manufactured item. Thus, an experimental unit maybe looked upon as a small subdivision of the experimental material, which receives the treatment.

Experimental Error

Differences in yields arising out of experimental units treated alike are called Experimental Error.

Controllable conditions in an experiment or experimental variable are terms as a factor. For example, a fertilizer, a new feed ration, and a fungicide are all considered as factors. Factors may be qualitative or quantitative and may take a finite number of values or type. Quantitative factors are those described by numerical values on some scale. The rates of application of fertilizer, the quantity of seed sown are examples of quantitative factors. Qualitative factors are those factors that can be distinguished from each other, but not on numerical scale e.g., type of protein in a diet, sex of an animal, genetic make up of plant etc. While choosing factors for any experiment researcher should ask the following questions, like What treatments in the experiment should be related directly to the objectives of the study? Does the experimental technique adopted require the use of additional factors? Can the experimental unit be divided naturally into groups such that the main treatment effects are different for the different groups? What additional factors should one include in the experiment to interact with the main factors and shed light on the factors of direct interest? How desirable is it to deliberately choose experimental units of different types?

Basic Principles of Design of Experiments

Given a set of treatments which can provide information regarding the objective of an experiment, a design for the experiment, defines the size and number of experimental units, the manner in which the treatments are allotted to the units and also appropriate type and grouping of the experimental units. These requirements of a design ensure validity, interpretability and accuracy of the results obtainable from an analysis of the observations.

These purposes are served by the principles of:

- Randomization
- Replication
- Local (Error) control

Randomization

After the treatments and the experimental units are decided the treatments are allotted to the experimental units at random to avoid any type of personal or subjective bias, which may be conscious or unconscious. This ensures validity of the results. It helps to have an objective comparison among the treatments. It also ensures independence of the observations, which is necessary for drawing valid inference from the observations by applying appropriate statistical techniques.

Depending on the nature of the experiment and the experimental units, there are various

experimental designs and each design has its own way of randomization. Various speakers while discussing specific designs in the lectures to follow shall discuss the procedure of random allocation separately.

Replication

If a treatment is allotted to r experimental units in an experiment, it is said to be replicated r times. If in a design each of the treatments is replicated r times, the design is said to have r replications. Replication is necessary to

- Provide an estimate of the error variance which is a function of the differences among observations from experimental units under identical treatments.
- Increase the accuracy of estimates of the treatment effects.

Though, more the number of replications the better it is, so far as precision of estimates is concerned, it cannot be increased infinitely as it increases the cost of experimentation. Moreover, due to limited availability of experimental resources too many replications cannot be taken.

The number of replications is, therefore, decided keeping in view the permissible expenditure and the required degree of precision. Sensitivity of statistical methods for drawing inference also depends on the number of replications. Sometimes this criterion is used to decide the number of replications in specific experiments.

Error variance provides a measure of precision of an experiment, the less the error variance the more precision. Once a measure of error variance is available for a set of experimental units, the number of replications needed for a desired level of sensitivity can be obtained as below.

Given a set of treatments an experimenter may not be interested to know if two treatment differ in their effects by less than a certain quantity, say, d. In other words, he wants an experiment that should be able to differentiate two treatments when they differ by d or more.

The significance of the difference between two treatments is tested by t-test where

$$t = \frac{\bar{y}_i - \bar{y}_j}{\sqrt{2s^2/r}},$$

Here, $\bar{y}_i$, and $\bar{y}_j$ are the arithmetic means of two treatment effects each based on r replications, s^2 is measure of error variation.

Given a difference d, between two treatment effects such that any difference greater than d should be brought out as significant by using a design with r replications, the following equation provides a solution of r.

$$t = \frac{|d|}{\sqrt{2s^2/r}},$$

$$r = \frac{t_0^2}{d^2} x 2s^2 \qquad \ldots(1)$$

where t_0 is the critical value of the t-distribution at the desired level of significance, that is, the value of t at 5 or 1 per cent level of significance read from the t-table. If s^2 is known or based on a very large number of observations, made available from some pilot pre-experiment investigation, then t is taken as the normal variate. If s^2 is estimated with n degree of freedom (d.f.) then t_0 corresponds to n d.f.

When the number of replication is r or more as obtained above, then all differences greater than d are expected to be brought out as significant by an experiment when it is conducted on a set of experimental units which has variability of the order of s^2. For example, in an experiment on wheat crop conducted in a seed farm in Bhopal, to study the effect of application of nitrogen and phosphorous on yield a randomized block design with three replications was adopted. There were 11 treatments two of which were (i) 60 Kg/ha of nitrogen (ii) 120 Kg/ha of nitrogen. The average yield figures for these two application of the fertilizer were 1438 and 1592 Kg/ha respectively and it is required that differences of the order of 150 Kg/ha should be brought out significant. The error mean square (s^2) was 12134.88. Assuming that the experimental error will be of the same order in future experiments and t_0 is of the order of 2.00, which is likely as the error d.f. is likely to be more than 30 as there are 11 treatments; Substituting in (1), we get:

$$r = \frac{2t_0^2 s^2}{d^2} = \frac{2x2^2 x2134.88}{150^2} = 4\,(\text{approx.})$$

Thus, an experiment with 4 replications is likely to bring out differences of the order of 150 Kg/ha as significant.

Another criterion for determining r is to take a number of replications which ensures at least 10 d.f. for the estimate of error variance in the analysis of variance of the design concerned since the sensitivity of the experiment will be very much low as the F test (which is used to draw inference in such experiments) is very much unstable below 10 d.f.

Local Control

The consideration in regard to the choice of number of replications ensure reduction of standard error of the estimates of the treatment effect because the standard error of the estimate of a treatment effect is $\sqrt{s^2/r}$, but it cannot reduce the error variance itself. It is, however, possible to devise methods for reducing the error variance. Such measures are called *error control* or local control. One such measure is to make the experimental units homogenous. Another method is to form the units into several homogenous groups, usually called blocks, allowing variation between the groups.

A considerable amount of research work has been done to divide the treatments into suitable groups of experimental units so that the treatment effect can be estimated more precisely Extensive use of combinatorial mathematics has been made for formation of such group

treatments. This grouping of experiment units into different groups has led to the development of various designs useful to the experimenter. We now briefly describe the various term used in designing of an experiment

Blocking

It refers to methodologies that form the units into homogeneous or pre-experimental subject-similarity groups. It is a method to reduce the effect of variation in the experimental material on the Error of Treatment of Comparisons. For example, animal scientist may decide to group animals on age, sex, breed or some other factors that he may believe has an influence on characteristic being measured. Effective blocking removes considerable measure of variation nom the experimental error. The selection of source of variability to be used as basis of blocking, block size, block shape and orientation are crucial for blocking. The blocking factor is introduced in the experiment to increase the power of design to detect treatment effects.

The importance of good designing is inseparable from good research (results). The following examples point out the necessity for a good design that will yield good research. First, a nutrition specialist in developing country is interested in determining whether mother's milk is better than powdered milk for children under age one. The nutritionist has compared the growth of children in village A, who are all on mother's milk against the children in village B, who use powdered milk. Obviously, such a comparison ignores the health of the mothers, the sanitary-conditions of the villages, and other factors that may have contributed to the differences observed without any connection to the advantages of mother's milk or the powdered milk on the children. A proper design would require that both mother's milk and the powdered milk be alternatively used in both villages, or some other methodology to make certain that the differences observed are attributable to the type of milk consumed and not to some uncontrollable factor. Second, a crop scientist who is comparing 2 varieties of maize, for instance, would not assign one variety to a location where such factors as sun, shade, unidirectional fertility gradient, and uneven distribution of water would either favor or handicap it over the other. If such a design were to be adopted, the researcher would have difficulty in determining whether the apparent difference in yield was due to variety differences or resulted from such factors as sun, shade, soil fertility of the field, or the distribution of water. These two examples illustrate the type of poorly designed experiments that are to be avoided.

Analysis of Variance

Analysis of Variance (ANOVA) is a technique of partitioning the overall variation in the responses into different assignable sources of variation, some of which are specifiable and others unknown. Total variance in the sample data is partitioned and is expressed as the sum of its non-negative components is a measure of the variation due to some specific independent source or factor or cause. ANOVA consists in estimation of the amount of variation due to each of the independent factors (causes) separately and then comparing these estimates due to ascribable factors (causes) with the estimate due to chance factor the latter being known as experimental error or simply the error.

Total variation present in a set of observable quantities may, under certain circumstances, be partitioned into a number of components associated with the nature of classification of the data.

The systematic procedure for achieving this is called *Analysis of Variance*. The initial techniques of the analysis of variance were developed by the statistician and geneticist R. A. Fisher in the 1920s and 1930s, and is sometimes known as Fisher's analysis of variance, due to the use of Fisher's F-distribution as part of the test of statistical significance.

Thus, ANOVA is a statistical technique that can be used to evaluate whether there are differences between the average value, or mean, across several population groups. With this model, the *response variable is continuous* in nature, whereas the *predictor variables are categorical*. For example, in a clinical trial of hypertensive patients, ANOVA methods could be used to compare the effectiveness of three different drugs in lowering blood pressure. Alternatively, ANOVA could be used to determine whether infant birth weight is significantly different among mothers who smoked during pregnancy relative to those who did not. In a particular case, where two population means are being compared, ANOVA is equivalent to the independent two-sample *t*-test.

The fixed-effects model of ANOVA applies to situations in which the experimenter applies several treatments to the subjects of the experiment to see if the response variable values change. This allows the experimenter to estimate the ranges of response variable values that the treatment would generate in the population as a whole. In it factors are fixed and are attributable to a finite set of levels of factor eg. Sex, year, variety, fertilizer etc.

Consider for example a clinical trial where three drugs are administered on a group of men and women some of whom are married and some are unmarried. The three classifications of sex, drug and marital status that identify the source of each datum are known as factors. The individual classification of each factor is known as levels of the factors. Thus, in this example there are 3 levels of factor drug, 2 levels of factor sex and 2 levels of marital status. Here all the effects are fixed. Random effects models are used when the treatments are not fixed. This occurs when the various treatments (also known as factor levels) are sampled from a larger population. When factors are random, these are generally attributable to infinite set of levels of a factor of which a random sample are deemed to occur *eg.* research stations, clinics in Delhi, sire, etc. Suppose new inject-able insulin is to be tested using 15 different clinics of Delhi state. It is reasonable to assume that these clinics are random sample from a population of clinics from Delhi. It describe the situations where both fixed and random effects are present.

In any ANOVA model, general mean is always taken as fixed effect and error is always taken as random effect. Thus class of model can be classified on the basis of factors, other than these two factors. ANOVA can be viewed as a generalization of *t*-tests: a comparison of differences of means across more than two groups.
The ANOVA is valid under certain assumptions. These assumptions are:

- Samples have been drawn from the populations that are normally distributed.
- Observations are independent and are distributed normally with mean zero and variance σ^2.
- Effects are additive in nature.

The ANOVA is performed as one-way, two-way, three-way, etc. ANOVA when the number of factors is one, two or three respectively. In general if the number of factors is more, it is termed as multi-way ANOVA.

Completely Randomized Design

Designs are usually characterized by the nature of grouping of experimental units and the procedure of random allocation of treatments to the experimental units. In a completely randomized design the units are taken in a single group. As far as possible the units forming the group are homogeneous. This is a design in which only randomization and replication are used. There is no use of local control here.

Let there be *v* treatments in an experiment and *n* homogeneous experimental units. Let the ith treatment be replicated r_i times *(i* = 1,2,..., *v)* such that $\sum_{i=1}^{v} r_i = n$. The treatments are allotted at random to the units.

Normally the number of replications for different treatments should be equal as it ensures equal precision of estimates of the treatment effects. The actual number of replications is, however, determined by the availability of experimental resources and the requirement of precision and sensitivity of comparisons. If the experimental material for some treatments is available in limited quantities, the numbers of their replication are reduced. If the estimates of certain treatment effects are required with more precision, the numbers of their replication are increased.

Randomization

There are several methods of random allocation of treatments to the experimental units. The *v* treatments are first numbered in any order from *1* to *v*. The *n* experimental units are also numbered suitably. One of the methods uses the random number tables. Any page of a random number table is taken. If *v* is a one-digit number, then the table is consulted digit by digit. If *v* is a two-digit number, then two-digit random numbers are consulted. All numbers greater than *v* including zero are ignored.

Let the first number chosen be n_1; then the treatment numbered n_1 is allotted to the first unit. If the second number is n_2 which may or may not be equal to n_1 then the treatment numbered n_2 is allotted to the second unit. This procedure is continued. When the ith treatment number has occurred r_i times, $(i = 1,2,...,v)$ this treatment is ignored subsequently. This process terminates when all the units are exhausted.

One drawback of the above procedure is that sometimes a very large number of random numbers may have to be ignored because they are greater than *v*. It may even happen that the random number table is exhausted before the allocation is complete. To avoid this difficulty the following procedure is adopted. We have described the procedure by taking *v* to be a two-digit number.

Let *P* be the highest two-digit number divisible by *v*. Then all numbers greater than *P* and zero are ignored. If a selected random number is less than *v*, then it is used as such. If it is greater than or equal to *v*, then it is divided by *v* and the remainder is taken to the random number. When a number is completely divisible by *v*, then the random number is *v*. If *v* is an *n*-digit number, then *P* is taken to be the highest *n*-digit number divisible by *v*. The rest of the procedure is the same as above.

Analysis

This design provides a one-way classified data according to levels of a single factor. For its analysis the following model is taken:

$$y_{ij} = \mu + t_i + e_{ij}\ , \qquad i = 1,\cdots,v;\ j = 1,\cdots r_i,$$

where y_{ij} is the random variable corresponding to the observation y_{ij} obtained from the jth replicate of the ith treatment, μ is the general mean, t_i is the fixed effect of the ith treatment and e_{ij} is the error component which is a random variable assumed to be normally and independently distributed with zero means and a constant variance σ^2.

Let $\sum_j y_{ij} = T_i \quad (i = 1,2,...,v)$ be the total of observations from i^{th} treatment. Let further $\sum_i T_i = G$. Correction factor ($C.F.$) $= G^2/n$.

Sum of squares due to treatments $= \sum_{i=1}^{v} \frac{T_i^2}{r_i} - C.F.$

Total sum of squares $= \sum_{i=1}^{v}\sum_{j=1}^{r_i} y_{ij}^2 - C.F.$

ANALYSIS OF VARIANCE

Sources of variation	**Degrees of freedom (D.F.)**	**Sum of squares (S.S.)**	**Mean squares (M.S.)**	**F**
Treatments	$v - 1$	$SST = \sum_{i=1}^{v} \frac{T_i^2}{r_i} - C.F.$	$MST = SST / (v - 1)$	MST/MSE
Error	$n - v$	SSE = by subtraction	$MSE = SSE / (n - v)$	
Total	$n - 1$	$\sum_{ij} y_{ij}^2 - C.F.$		

The hypothesis that the treatments have equal effects is tested by F-test where F is the ratio MST / MSE with $(v - 1)$ and $(n - v)$ degrees of freedom.

3. Randomized Complete Block Design

It has been seen that when the experimental units are homogeneous then a CRD should be adopted. In any experiment, however, besides treatments the experimental material is a major source of variability in the data. When experiments require a large number of experimental units, the experimental units may not be homogeneous, and in such situations CRD can not be recommended. When the experimental units are heterogeneous, a part of the variability can be accounted for by grouping the experimental units in such a way that experimental units within each group are as homogeneous as possible. The treatments are then allotted randomly to the experimental units within each group (or blocks). The principle of first forming homogeneous

groups of the experimental units and then allotting at random each treatment once in each group is known as local control. This results in an increase in precision of estimates of the treatment contrasts, due to the fact that error variance that is a function of comparisons within blocks, is smaller because of homogeneous blocks. This type of allocation makes it possible to eliminate from error variance a portion of variation attributable to block differences. If, however, variation between the blocks is not significantly large, this type of grouping of the units does not lead to any advantage; rather some degrees of freedom of the error variance is lost without any consequent decrease in the error variance. In such situations it is not desirable to adopt randomized complete block designs in preference to completely randomized designs.

If the number of experimental units within each group is same as the number of treatments and if every treatment appears precisely once in each group then such an arrangement is called a ***randomized complete block design.***

Suppose the experimenter wants to study *v* treatments. Each of the treatments is replicated *r* times (the number of blocks) in the design. The total number of experimental units is, therefore, *vr*. These units are arranged into *r* groups of size *v* each. The error control measure in this design consists of making the units in each of these groups homogeneous.

The number of blocks in the design is the same as the number of replications. The *v* treatments are allotted at random to the *v* plots in each block. This type of homogeneous grouping of the experimental units and the random allocation of the treatments separately in each block are the two main characteristic features of randomized block designs. The availability of resources and considerations of cost and precision determine actual number of replications in the design.

Analysis

The data collected from experiments with randomized block designs form a two-way classification, that is, classified according to the levels of two factors, *viz.,* blocks and treatments. There are *vr* cells in the two-way table with one observation in each cell. The data are orthogonal and therefore the design is called an *orthogonal design.* We take the following model:

$$y_{ij} = \mu + t_i + b_j + e_{ij}\ , \qquad \begin{pmatrix} i = 1,2,...,v; \\ j = 1,2,...,r \end{pmatrix},$$

where y_{ij} denotes the observation from i^{th} treatment in j^{th} block. The fixed effects μ, t_i, b_j denote respectively the general mean, effect of the ith treatment and effect of the jth block. The random variable e_{ij} is the error component associated with y_{ij}. These are assumed to be normally and independently distributed with zero means and a constant variance σ^2.

Following the method of analysis of variance for finding sums of squares due to blocks, treatments and error for the two-way classification, the different sums of squares are obtained as follows: Let $\sum_j y_{ij} = T_i \ \ (i = 1,2,...,v)$ = total of observations from i^{th} treatment and

$\sum_j y_{ij} = B_j \quad j = 1, \cdots, r$ = total of observations from j^{th} block. These are the marginal totals of the two-way data table. Let further, $\sum_i T_i = \sum_j B_j = G.$

Correction factor (*C.F.*) = G^2/rv, Sum of squares due to treatments $= \sum_i \frac{T_i^2}{r} - C.F.,$

Sum of squares due to blocks $= \sum_j \frac{B_j^2}{v} - C.F.$, Total sum of squares $= \sum_{ij} y_{ij}^2 - C.F.$

ANALYSIS OF VARIANCE

Sources of variation	**Degrees of freedom (D.F.)**	**Sum of squares (S.S.)**	**Mean squares (M.S.)**	**F**
Blocks	*r - 1*	$SSB = \sum_j \frac{B_j^2}{v} - C.F.$	*MSB = SSB / (r - 1)*	*MSB/MSE*
Treatments	*v - 1*	$SST = \sum_i \frac{T_i^2}{r} - C.F.$	*MST = SST / (v - 1)*	*MST/MSE*
Error	*(r - 1)(v - 1)*	*SSE = by subtraction*	*MSE = SSE / (v - 1)(r - 1)*	
Total	*vr - 1*	$\sum_{ij} y_{ij}^2 - C.F.$		

The hypothesis that the treatments have equal effects is tested by F-test, where F is the ratio *MST / MSE* with *(v - 1)* and *(v - 1)(r - 1)* degrees of freedom. We may then be interested to either compare the treatments in pairs or evaluate special contrasts depending upon the objectives of the experiment. This is done as follows:

The critical difference for testing the significance of the difference of two treatment effects, say $t_i - t_j$ is $C.D. = t_{(v-1)(r-1),\alpha/2}\sqrt{2MSE/r}$, where $t_{(v-1)(r-1),\alpha/2}$ is the value of Student's *t* at the level of significance α and degree of freedom *(v - 1)(r - 1).* If the difference of any two-treatment means is greater than the C.D. value, the corresponding treatment effects are significantly different.

4. Latin Square Design

Latin square designs are normally used in experiments where it is required to remove the heterogeneity of experimental material in two directions. These designs require that the number of replications equal the number of ***treatments*** or ***varieties***.

Definition 1. A Latin square arrangement is an arrangement of *v* symbols in v^2 cells arranged in *v* rows and *v* columns, such that every symbol occurs precisely once in each row and precisely once in each column. The term *v* is known as the **order** of the Latin square.

If the symbols are taken as *A, B, C, D,* a Latin square arrangement of order 4 is as follows:

A	*B*	*C*	*D*
B	*C*	*D*	*A*
C	*D*	*A*	*B*
D	*A*	*B*	*C*

A Latin square is said to be in the ***standard form*** if the symbols in the first row and first column are in natural order, and it is said to be in the ***semi-standard form*** if the symbols of the first row are in natural order. Some authors denote both of these concepts by the term ***standard form***. However, there is a need to distinguish between these two concepts. The standard form is used for randomizing the Latin-square designs, and the semi-standard form is needed for studying the properties of the orthogonal Latin squares.

Definition 2. If in two Latin squares of the same order, when superimposed on one another, every ordered pair of symbols occurs exactly once, the two Latin squares are said to be **orthogonal**. If the symbols of one Latin square are denoted by Latin letters and the symbols of the other are denoted by Greek letters, the pair of orthogonal Latin squares is also called a **graeco-latin square**.

Definition 3. If in a set of Latin squares every pair is orthogonal, the set is called a set of **mutually orthogonal latin squares (MOLS)**. It is also called a **hypergraeco latin square.**

The following is an example of graeco latin square:

A	B	C	D
B	A	D	C
C	D	A	B
D	C	B	A

α	γ	δ	β
β	δ	γ	α
γ	α	β	δ
δ	β	α	γ

$A\alpha$	$B\gamma$	$C\delta$	$D\beta$
$B\beta$	$A\delta$	$D\gamma$	$C\alpha$
$C\gamma$	$D\alpha$	$A\beta$	$B\delta$
$D\delta$	$C\beta$	$B\alpha$	$A\gamma$

We can verify that in the above arrangement every pair of ordered Latin and Greek symbols occurs exactly once, and hence the two latin squares under consideration constitute a graecolatin square.

It is well known that the maximum number of MOLS possible of order v is $v - 1$. A set of $v - 1$ MOLS is known as a complete set of MOLS. Complete sets of MOLS of order v exist when v is a ***prime or prime power.***

Randomization

According to the definition of a Latin square design, treatments can be allocated to the v^2 experimental units (may be animal or plots) in a number of ways. There are, therefore, a number of Latin squares of a given order. The purpose of randomization is to select one of these squares at random. The following is one of the methods of random selection of Latin squares.

Let a $v \times v$ Latin square arrangement be first written by denoting treatments by Latin letters *A, B, C, etc.* or by numbers *1, 2, 3, etc.* Such arrangements are readily available in the ***Tables for Statisticians and Biometricians*** (Fisher and Yates, 1974). One of these squares of any order can be written systematically as shown below for a 5×5 Latin square:

A	*B*	*C*	*D*	*E*
B	*C*	*D*	*E*	*A*
C	*D*	*E*	*A*	*B*
D	*E*	*A*	*B*	*C*
E	*A*	*B*	*C*	*D*

For the purpose of randomization rows and columns of the Latin square are rearranged randomly. There is no randomization possible within the rows and/or columns. For example, the following is a row randomized square of the above 5×5 Latin square;

A	*B*	*C*	*D*	*E*
B	*C*	*D*	*E*	*A*
E	*A*	*B*	*C*	*D*
D	*E*	*A*	*B*	*C*
C	*D*	*E*	*A*	*B*

Next, the columns of the above row randomized square have been rearranged randomly to give the following random square:

E	*B*	*C*	*A*	*D*
A	*C*	*D*	*B*	*E*
D	*A*	*B*	*E*	*C*
C	*E*	*A*	*D*	*B*
B	*D*	*E*	*C*	*A*

As a result of row and column randomization, but not the randomization of the individual units, the whole arrangement remains a Latin square.

Analysis of Latin Square Designs

In Latin square designs there are three factors. These are the factors *P, Q,* and treatments. The data collected from this design are, therefore, analyzed as a three-way classified data. Actually, there should have been v^3 observations as there are three factors each at *v* levels. But because of the particular allocation of treatments to the cells, there is only one observation per cell instead of *v* in the usual three way classified orthogonal data. As a result we can obtain only the sums of squares due to each of the three factors and error sum of squares. None of the interaction sums of squares of the factors can be obtained. Accordingly, we take the model

$$Y_{ijs} = \mu + r_i + c_j + t_s + e_{ijs}$$

where y_{ijs} denotes the observation in the ith row, j^{th} column and under the sth treatment; $\mu, r_i, c_j, t_s (i, j, s = 1,2,...,v)$ are fixed effects denoting in order the general mean, the row, the column and the treatment effects. The e_{ijs} is the error component, assumed to be independently and normally distributed with zero mean and a constant variance, σ^2.

The analysis is conducted by following a similar procedure as described for the analysis of two-way classified data. The different sums of squares are obtained as below: Let the data be

arranged first in a *row* × *column* table such that y_{ij} denotes the observation of (*i*, *j)*th cell of table.

Let $R_i = \sum_j y_{ij} = i^{th}\ row\ total\ (i = 1,2,...,v)$, $C_j = \sum_i y_{ij} = j^{th}\ column\ total\ (j = 1,2,...,v)$, T_s = sum of those observations which come from s^{th} treatment *(s= 1,2,…,v)*, $G = \sum_i R_i = grand\ total$. Correction factor, $C.F. = \frac{G^2}{v^2}$. Treatment sum of squares = $\sum_s \frac{T_s^2}{v} - C.F.$, Row sum of squares = $\sum_i \frac{R_i^2}{v} - C.F.$, Column sum of squares = $\sum_j \frac{C_j^2}{v} - C.F.$

Analysis of Variance of *v* × *v* Latin Square Design

Sources of Variation	D.F.	S.S.	M.S.	F
Rows	*v -1*	$\sum_i \frac{R_i^2}{v} - C.F.$		
Columns	*v - 1*	$\sum_j \frac{C_j^2}{v} - C.F.$		
Treatments	*v - 1*	$\sum_s \frac{T_s^2}{v} - C.F.$	s_t^2	s_t^2 / s_e^2
Error	*(v - 1)(v - 2)*	By subtraction	s_e^2	
Total	v^2-1	$\sum_{ij} y_{ij}^2 - C.F.$		

The hypothesis of equal treatment effects is tested by *F*-test, where *F* is the ratio of treatment mean squares to error mean squares. If *F* is not significant, treatment effects do not differ significantly among themselves. If *F* is significant, further studies to test the significance of any treatment contrast can be made in exactly the same way as discussed for randomized block designs.

SAS Code

Analysis of data obtained from experiment conducted under CRD setup

```
data crd;
input treatment yield;
cards;
1       850.5
1       453.6
1       878.85
1       623.7
1       510.3
```

1	765.45
1	680.4
1	595.35
1	538.65
1	850.5
1	850.5
1	793.8
1	1020.6
1	708.75
1	652.05
1	623.7
1	396.9
1	822.15
1	680.4
1	652.05
1	538.65
1	850.5
1	680.4
1	.
1	.
2	510.3
2	963.9
2	652.05
2	1020.6
2	878.85
2	567
2	680.4
2	538.65
2	567
2	510.3
2	425.25
2	567
2	623.7
2	538.65
2	737.1
2	453.6
2	481.95
2	368.55
2	567
2	595.35
2	567
2	595.35
2	.
2	.
2	.
3	992.25

```
3	850.5
3	1474.2
3	510.3
3	850.5
3	793.8
3	453.6
3	935.55
3	1190.7
3	481.95
3	623.7
3	878.85
3	1077.3
3	850.5
3	680.4
3	737.1
3	737.1
3	708.75
3	708.75
3	652.05
3	567
3	453.6
3	652.05
3	567
3	.
;
```

```
proc glm;
class treatment;
model yield = treatment;
means  treatment;
means treatment/lsd;
run;
```

Analysis of data obtained from experiment conducted under RBD setup

```
data rbd;
input block  treatment yield;
cards;
1	1	6.9
1	2	6.48
1	3	6.52
1	4	6.9
1	5	6
1	6	7.9
2	1	4.6
2	2	5.57
```

```
2	3	7.6
2	4	6.65
2	5	6.18
2	6	7.57
3	1	4.4
3	2	4.28
3	3	5.3
3	4	6.75
3	5	5.5
3	6	6.8
4	1	4.81
4	2	4.45
4	3	5.3
4	4	7.75
4	5	5.5
4	6	6.62
;
proc glm;
class  block treatment;
model yield = block treatment;
means  treatment;
means treatment/tukey;
run;
```

Analysis of data obtained from experiment conducted under LSD setup

```
data lsd;
input row column trt yld;
cards;
1	1	3	3.1
1	2	6	5.95
1	3	1	1.75
1	4	5	6.4
1	5	2	3.85
1	6	4	5.3
2	1	2	4.8
2	2	1	2.7
2	3	3	3.3
2	4	6	5.95
2	5	4	3.7
2	6	5	5.4
3	1	1	3
3	2	2	2.95
3	3	5	6.7
```

```
3       4       4       5.95
3       5       6       7.75
3       6       3       7.1
4       1       5       6.4
4       2       4       5.8
4       3       2       3.8
4       4       3       6.55
4       5       1       4.8
4       6       6       9.4
5       1       6       5.2
5       2       3       4.85
5       3       4       6.6
5       4       2       4.6
5       5       5       7
5       6       1       5
6       1       4       4.25
6       2       5       6.65
6       3       6       9.3
6       4       1       4.95
6       5       3       9.3
6       6       2       8.4
;
proc glm;
class  blk trt;
model yld = blk trt;
means  trt;
means trt/tukey;
run;
```

INCOMPLETE BLOCK DESIGNS

Cini Varghese
ICAR-IASRI, Library Avenue, New Delhi – 110 012
cini.varghese@icar.gov.in

1. Introduction

Incomplete block designs are desirable when number of treatments to be tested is large and / or complete blocks are unavailable or inappropriate. These designs were introduced by Yates in order to eliminate heterogeneity to a greater extent as compared to a complete block design, when the number of treatments is large. The precision of the estimate of a treatment effect depends on the number of replications of the treatment - the larger the number of replications, the more is the precision. Similar is the case for the precision of estimate of the difference between two treatment effects. If a pair of treatment occurs together more number of times in the design, the difference between these two treatment effects can be estimated with more precision. To ensure equal or nearly equal precision of comparisons of different pairs of treatment effects, the treatments are so allocated to the experimental units in different blocks of equal sizes such that each treatment occurs at most once in a block and it has an equal number of replications and each pair of treatments has the same or nearly the same number of replications. When the number of replications of all pairs of treatments in a design is the same, then we have an important class of designs called **Balanced Incomplete Block** (BIB) designs and when there are unequal number of replications for different pairs of treatments, then the designs are called as **Partially Balanced Incomplete Block** (PBIB) designs. Another important class of incomplete block designs is lattice designs. Some of these are Balanced Incomplete Block (BIB) designs while others are Partially Balanced Incomplete Block (PBIB) designs.

2. Balanced Incomplete Block (BIB) Designs

A BIB design is an arrangement of v treatments in b blocks each of size k (<v) such that

(i) Each treatment occurs at most once in a block
(ii) Each treatment occurs in exactly r blocks
(iii) Each pair of treatments occurs together in exactly λ blocks.

Example 2.1: A BIB design for v = b = 5, r = k = 4 and λ = 3 in the following:

Blocks	
1	(1,2,3,4)
2	(1,2,3,5)
3	(1,2,4,5)
4	(1,3,4,5)
5	(2,3,4,5)

The symbols v, b, r, k, λ are called the parameters of the design. These parameters satisfy the relations

$$vr = bk \quad \ldots(2.1)$$

and $$\lambda(v-1) = r(k-1) \quad \ldots(2.2)$$

A BIB design cannot exist unless (2.1) and (2.2) are satisfied. For instance, no design exists for v = b = 6 and r = k = 3 since, from (2.2) λ=6/5 is not an integer. However, these

conditions are not sufficient for the existence of a BIB design. Even if both (2.1) and (2.2) are satisfied, it does not follow that such a design exists. For example, no BIB design exits for v = 15, b = 21, r = 7, k = 5, and $\lambda = 2$ even though both conditions are satisfied. In search of a criterion for the availability of a BIB design, Fisher proved that no design with b<v is possible.

Construction of BIB Designs

There is no single method of constructing all BIB designs. Solutions of many designs are still **unknown**. We describe below a few well known series of BIB designs.

2.1 Unreduced BIB Designs

These designs are obtained by taking all combinations of the v treatments k at a time. Therefore, the parameters of all unreduced BIB designs are:

$$v, k, b = {}^{v}C_{k}, r = {}^{v-1}C_{k-1}, \lambda = {}^{v-2}C_{k-2}$$

The BIB design for v = 5 treatments given in the previous section is an example of an unreduced BIB design in blocks of size 4.

Example 2.1: Let v = 5, k = 3, then $b = {}^{5}C_{3} = 10$, $r = {}^{4}C_{2} = 6$ and $\lambda = {}^{3}C_{1}$. The 10 blocks are:

	Blocks
1	(1,2,3)
2	(1,2,4)
3	(1,2,5)
4	(1,3,4)
5	(1,3,5)
6	(1,4,5)
7	(2,3,4)
8	(2,3,5)
9	(2,4,5)
10	(3,4,5)

These unreduced designs usually require a large number of blocks and replications so that the resulting designs will often be too large for practical purposes.

2.2 BIB Designs using MOLS

Before we describe the method, we explain the concept of mutually orthogonal Latin squares (MOLS) which will be used in the construction of BIB designs.

A Latin square of order s is an arrangement of s symbols in an s × s array such that each symbol occurs once in each row and once in each column of the array. For example, the following are 4 × 4 Latin squares of order 4 in symbols A, B, C, and D:

A B C D	A B C D	A B C D
B A D C	C D A B	D C B A
C D A B	D C B A	B A D C
D C B A	B A D C	C D A B

Two Latin squares are pairwise orthogonal if, when one square is superimposed on the other, each symbol of one Latin square occurs once with each symbol of the other square. Three or

more squares are mutually orthogonal if they are pair-wise orthogonal. The three 4×4 Latin squares above are mutually orthogonal.

A complete set of s-1 mutually orthogonal Latin squares is known to exist for any $s = p^n$, where p is a prime number. Tables can be found in Fisher and Yates (1963). Now we describe the methods of constructing BIB designs using MOLS.

Suppose $v= s^2$ treatments are set out in an $s \times s$ array. A group of s blocks each of size s is obtained by letting the rows of the array represent blocks. Another group of s blocks is given by taking the columns of the array as blocks. Now suppose one of the orthogonal Latin squares is superimposed on to the array of treatments. A further group of s blocks is obtained if all treatments common to a particular symbol in the square are placed in a block. Each of the s-1 orthogonal squares produces a set of s blocks in this manner. The resulting design is a BIB design with parameters $v = s^2$, $b = s^2 + s$, $k = s$, $r = s + 1$, $\lambda = 1$.

Example 2.2: For $v = 3^2 = 9$ treatments a 3×3 array and a complete set of mutually orthogonal Latin squares of order 3×3 are :

1 2 3	A B C	A B C
4 5 6	C A B	B C A
7 8 9	B C A	C A B

Four groups of 3 blocks are obtained from the rows, columns and the symbols of the two squares, as follows:

	Blocks		
	(1, 2, 3)		(1, 5, 9)
Rows	(4, 5, 6)	**First square**	(2, 6, 7)
	(7, 8, 9)		(3, 4, 8)
	(1, 4, 7)		(1, 6, 8)
Columns	(2, 5, 8)	**Second square**	(2, 4, 9)
	(3, 6, 9)		(3, 5, 7)

It can be checked that this is a BIB design with parameters $v = 9$, $b = 12$, $r = 4$, $k = 3$, and $\lambda = 1$.

2.3 Randomization Procedure

(i) Allot the treatment symbols (1,2,...,v) to the v treatments at random.
(ii) Allot the groups of k treatments to the b blocks at random.
(iii) Randomize the positions of the treatment numbers within each block.

2.4 Statistical Analysis

Consider the following model:

Observation = General mean + treatment effect + block effect + random error.

Random errors are assumed to be independently and identically distributed normally with mean zero and constant variance σ^2. On minimising the error sum of squares with respect to

the parameters, we get a set of normal equations which can be solved to get the estimates of different contrasts of various treatment and block effects.

Now we compute

G = Grand total of observations
$\bar{y}$ = grand mean = G/n, where n= vr = bk = total number of observations
T_i = Sum of obervations for treatment i, (i=1,2,..., v)
B_j = Sum of observations in block j, (j=1,2,..., b)
$CF= G^2/ n$,
Q_i = adjusted i^{th} treatment total
= T_i - (Sum of block totals in which treatment i occurs) / Block size (k)

A solution for the ith treatment effect is,

$$\hat{\tau} = (k Q_i) / (\lambda v) \qquad (i = 1,2, ..., v)$$

Adjusted treatment mean for treatment i= i^{th} treatment effects ($\hat{\tau}_i$) + grand mean ($\bar{y}$).

Various sums of squares can be obtained as follows:

(i) Total Sum of Squares (TSS) = Σ (observations)^2 - CF
(ii) Treatment Sum of Squares unadjusted (SST_u) = [Σ T_i^2] /r - CF
(iii) Block Sum of Squares unadjusted (SSB_U) = [Σ B_j^2] / k - CF
(iv) Treatments Sum of Squares adjusted (SST_A) = Σ $\hat{\tau}_i$ Q_i
(v) Error SS (SSE) = TSS - SSB_U - SST_A
(vi) Blocks sum of squares adjusted (SSB_A) = SST_A + SSB_U - SST_U

The analysis of variance for a BIB design is given below:

Table 2.1: ANOVA for a BIB (v, b, r, k, λ) Design

Source	DF	SS	MS	F
Treatment (unadj.)	v-1	SST_u		
Blocks (unadjusted)	b-1	SSB_u		
Treatments (adjusted)	v-1	SST_A	MST	MST/MSE
Blocks (adjusted)	b-1	SSB_A	MSB	MSB/MSE
Error	n-b-v+1	SSE	MSE	
Total	n-1	TSS		

Note: MST = SST_A / (v-1), MSB = SSB_A / (b-1) and MSE = SSE / (n -b- v + 1)
Coefficient of Variation = ($\sqrt{MSE}$ / $\bar{y}$) × 100

Standard error of difference between two adjusted treatment means = $[2k MSE/ (\lambda v)]^{1/2}$.

C.D. = $t_{0.05} \times [2k MSE/ (\lambda v)]^{1/2}$

3. Partially Balanced Incomplete Block (PBIB) Designs

BIB designs may not fit well to many experimental situations as these designs may not be available for all numbers of treatments and block sizes or may require a large number of replications. To overcome these difficulties PBIB designs were introduced. In these designs the variance of every estimated elementary contrast among treatment effects is not the same. The definition of PBIB designs is based on the association scheme.

Association Scheme

Given v treatment symbols 1,2,..,v, a relation satisfying the following conditions is called an m-class association scheme (m ≥2):

(i) Any two symbols are either 1st, 2nd,..., or m^{th} associates; the relation of association being symmetric, *i.e.*, if the symbol α is the ith associate of β, then β is the ith associate of α.

(ii) Each symbol α has n_i i^{th} associates, the number n_i being independent of α,

(iii) If any two symbols α and β are i^{th} associates, then the number of symbols that are j^{th} associates of α and k^{th} associate of β is p^i_{jk} and is independent of the pair of ith associates α and β.

The numbers v, n_i and p^i_{jk} (i,j,k = 1,2,...,m) are called the parameters of the association scheme and satisfy the following relations:

$$\sum_{i=1}^{m} n_i = v - 1$$

$$\sum_{k=1}^{m} p^i_{jk} = n_j - 1, \quad \text{if } i = j$$

$$= n_j, \quad \text{if } i \neq j$$

$$n_i p^i_{jk} = n_j p^j_{ik}$$

Example 3.1: Consider v=12 treatments denoted by numbers 1 to 12. Form 3 groups of 4 symbols each as follows: (1,2,3,4), (5,6,7,8), (9,10,11,12). We now define any two treatments as first associates if they belong to the same group, and second associates if they belong to the different groups. Here, $n_1 = 3$, $n_2 = 8$.

Definition: Given an association scheme with m classes (m ≥2) we have a PBIB design with m associate classes based on the association scheme, if the v treatment symbols can be arranged into b blocks, such that

(i) Every symbol occurs at most once in a block.

(ii) Every symbol occurs in exactly r blocks.

(iii) If two symbols are i^{th} associates, then they occur together in λ_i blocks, the number λ_i being independent of the particular pair of ith associates α and β.

The numbers v, b, r, k, λ_i (i =1,2,...,m) are called the parameters of the design. It can be easily seen that

$$vr = bk \text{ and } \sum_{i=1}^{m} n_i \lambda_i = r(k-1).$$

It may be mentioned that as in the case of BIB designs, the complementary design of a PBIB with parameters v,b,r,k,λ_i is also a PBIB design having the same association scheme with the parameters $v^*=v$, $b^*=b$, $r^*=b-r$, $k^*=v-k$, $\lambda_i^*=b-2r+\lambda_i$.

PBIB designs can be broadly classified into (i) two-associate class PBIB designs (ii) three-associate class PBIB designs and (iii) higher associate class PBIB designs. Two-class association schemes and the two-associate PBIB designs have been extensively studied in the literature and are simple to use. As an illustration, we describe Group Divisible (GD) association scheme and the designs based on it.

3.1 GD Association Scheme

Let v = mn symbols be arranged into m groups of n symbols each. A pair of symbols belonging to the same group is first associates [n_1 = n-1] and a pair of symbols belonging to different groups is second associates [n_2 = n(m-1)]. A PBIB (2) design based on a GD scheme is called a GD design.

Method of Construction of Some GD Designs

Let D be a BIB design with parameters v = m, b, r, k, λ. Obtain a design D^* from D by replacing the ith treatment (i=1,2,...,v) in D by n new treatment symbols $i_1,i_2,...,i_n$. D^* is a group divisible design with the following parameters v^*= mn, b^*= b, r^*= r, k^*= nk, m, n, λ_1 =r , λ_2=λ.

Example 3.1: Consider the following BIB design with parameters (4, 4, 3, 3, 2):

(1, 2, 3)
(1, 2, 4)
(1, 3, 4)
(2, 3, 4)

Replacing 1 by a, b; 2 by c, d; 3 by e, f and 4 by g, h, the following GD design with parameters v = 8, b = 4, r = 3, k = 6, λ_1 =3 , λ_2 = 2. is obtained:

(a, b, c, d, e, f)
(a, b, c, d, g, h)
(a, b, e, f, g, h)
(c, d, e, f, g, h)

3.2 Triangular association scheme and Design

3.2.1 Association scheme: Let there be n(n-1)/2 treatments arranged in a square array of size n such that the positions of the principal diagonal of the array are left blank, the n(n-1)/2 positions above the principal diagonal are filled up by the v treatment symbols and the positions below the principal diagonal are filled up by the v symbols in such a manner that the resultant arrangement is symmetrical about the principal diagonal.

Two treatments are first associates if they belong to same row or same column of the array and second associates, otherwise.
Triangular scheme exists when n≥5**.**
v=n(n-1)/2, n≥5, n_1=2(n-2), n_2=(n-2)(n-3)/2

$$P_1 = \begin{bmatrix} (n-2) & (n-3) \\ (n-3) & [(n-3)(n-4)]/2 \end{bmatrix} \quad P_2 = \begin{bmatrix} 4 & 2(n-4) \\ 2(n-4) & [(n-4)(n-5)]/2 \end{bmatrix}$$

Example 3.2.1: For n=5

*	1	2	3	4
1	*	5	6	7
2	5	*	8	9
3	6	8	*	10
4	7	9	10	*

Table 2.2 shows the various associates of all the treatments.

Table 2.2

Treatment	**1st Associates**	**2nd Associates**
1	2, 3,4,5, 6,7	8,9,10,
2	1,3,4,5, 8, 9	6,7, 10
3	1, 2,4, 6,8,10	5,7, 9
4	1,2,3,7,9,10,	5,6,8
5	1,2, 6,7, 8, 9	3,4,10
6	1,3,5,7, 8,10	2,4,9
7	1,4,5, 6, 9,10,	2,3,8
8	2,3,5, 6, 9,10	1,4,7
9	2,4,5,7, 8,10	1,3,6
10	3,4,6,7,8,9	1,2,5

3.2.2 Method of construction of Triangular designs: A two class association scheme is called triangular design if it is based on triangular association scheme**.** In a triangular association scheme, if we take each row as a block then the resultant design is triangular design with parameters v = n (n-1)/2, b=n, r=2, k=n-1, λ_1=1, λ_2= 0.

Example 3.2.2: Suppose n=5, giving rise to v=10 treatments as follows:

*	1	2	3	4
1	*	5	6	7
2	5	*	8	9
3	6	8	*	10
4	7	9	10	*

Taking each row as block, the following triangular design is obtained:

1 2 3 4
1 5 6 7
2 5 8 9
3 6 8 10

4 7 9 10

Here, v = 10, b=5, r=2, k=4, λ_1=1, λ_2= 0.

Analysis of data obtained from experiment conducted under BIBD setup

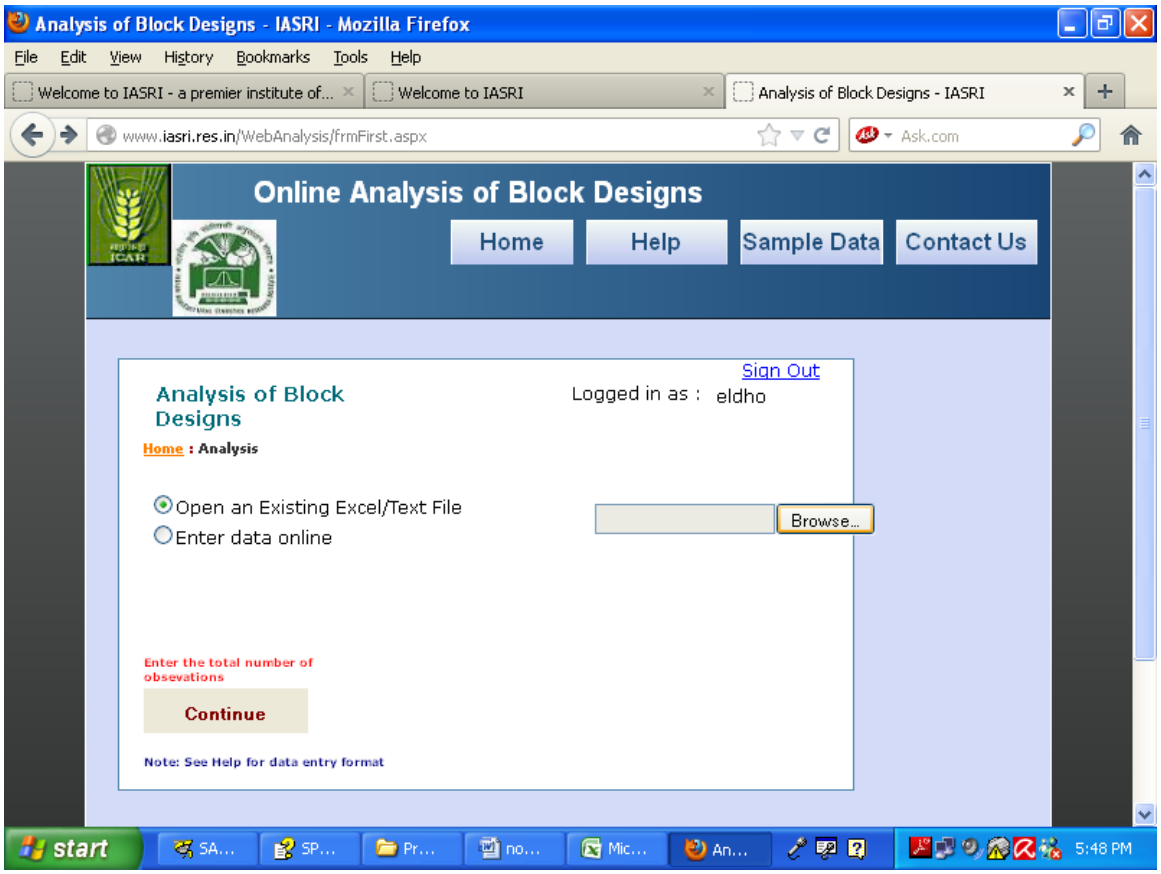
```
data bibd;
input blk trt yld;
cards;
1       1       77
1       2       65
1       3       75
2       7       47
2       9       61
2       8       60
3       6       54
3       4       60
3       5       65
4       1       70
4       7       62
4       4       62
5       8       72
5       5       55
5       2       55
6       3       50
6       6       40
6       9       60
7       1       63
7       8       67
7       6       54
8       4       62
8       2       53
8       9       57
9       3       68
9       5       67
9       7       66
10      1       69
10      9       62
10      5       52
11      2       61
11      6       63
11      7       79
12      8       65
12      3       65
12      4       38
;
proc glm;
class  blk trt;
```

```
model yld = blk trt/ss1; /*treatment effects adjusted for block effects*/
means  trt;
lsmeans trt/tukey;
run;

proc glm;
class  blk trt;
model yld = trt blk /ss1; /* block effects adjusted for treatment effects*/
means  trt;
lsmeans trt/tukey;
run;
```

Analysis of data obtained from experiment conducted under PBIBD setup

Go to the website http://www.iasri.res.in/WebAnalysis



References and Suggested Reading

Cochran, W.G. and Cox, G.M. (1957). Experimental designs. John Wiley & Sons, Inc., New York.

Fisher, R.A. and Yates, F. (1963). Statistical Tables. Hafner Pub. Co., New York.

Harshbarger, B. (1947). Rectangular Latices. *Va. Agr. Exp. Sta. Mem. 1.*

Harshbarger,B. (1949). Triple Rectangular Lattices. *Biometrics,* **5**, 1-13.

Yates, F.(1936). A new method of arranging variety trials involving a large number of varieties. *J. Agric. Sci.*, **26**, 424-455.

ANCOVA

Sukanta Dash
ICAR-IASRI, Library Avenue, New Delhi – 110 012
sukanta.dash@icar.gov.in

Introduction

The meaning of ANVOVA is Analysis of Covariance. It is a general linear model with one continuous outcome variable (quantitative) and one or more factor variables (qualitative). ANCOVA is a merger of ANOVA and regression for continuous variables. ANCOVA tests whether certain factors have an effect on the outcome variable after removing the variance for which quantitative predictors (covariates) account. The inclusion of covariates can increase statistical power because it accounts for some of the variability.

It is well known that in designed experiments the ability to detect existing differences among treatments increases as the size of the experimental error decreases, a good experiment attempts to incorporate all possible means of minimizing the experimental error. Besides proper experimentation, a proper data analysis also helps in controlling experimental error. In situations where blocking alone may not be able to achieve adequate control of experimental error, proper choice of data analysis may help a great deal. By measuring one or more *covariates* - the characters whose functional relationships to the character of primary interest are known - the Analysis of Covariance (ANCOVA) can reduce the variability among experimental units by adjusting their values to a common value of the covariates. For example, in an animal feeding trial, the initial body weight of the animals usually differs. Using this initial body weight as a covariate, the final weights recorded after the animals have been subjected to various physiological feeds (treatments) can be adjusted to the values that would have been obtained had there been no variation in the initial body weights of the animals at the start of the experiment. An another example, in a field experiment where rodents have (partially) damaged some of the plots, covariance analysis with rodent damage as a covariate could be useful in adjusting plot yields to the levels that they should have been had there been no rodent damage in any plot.

ANCOVA requires measurement of the character of primary interest plus the measurement of one or more variables known as *covariates*. It also requires that the functional relationship of the covariates with the character of primary interest is known beforehand. Generally a linear relationship is assumed, though other type of relationships could also be assumed.

Consider the case of a variety trial in which weed incidence is used as a covariate. With a known functional relationship between weed incidence and grain yield, the character of primary interest, the covariance analysis can adjust grain yield in each plot to a common level of weed incidence. With this adjustment, the variation in yield due to weed incidence is quantified and effectively separated from that due to varietal difference.

ANCOVA can be applied to any number of covariates and to any type of functional relationship between variables *viz.* quadratic, inverse polynomial, etc. Here we illustrate the use of covariance analysis with the help of a single covariate that is linearly related with the character of primary interest. It is expected that this simplification shall not unduly reduce

the applicability of the technique, as a single covariate that is linearly related with the primary variable is adequate for most of the experimental situations in agricultural research.

Uses of Covariance Analysis in Agricultural Research

There are several important uses of covariance analysis in agricultural research. Some of the most important ones are:

1. To control experimental error and to adjust treatment means.
2. To aid in the interpretation of experimental results.
3. To estimate missing data.

Error Control and Adjustment of Treatment Means

It is now well realized that the size of experimental error is closely related to the variability between experimental units. It is also known that proper blocking can reduce experimental error by maximizing the differences between the blocks and thus minimizing differences within blocks. Blocking, however, can not cope with certain types of variability such as spotty soil heterogeneity and unpredictable insect incidence. In both instances, heterogeneity between experimental plots does not follow a definite pattern, which causes difficulty in getting maximum differences between blocks. Indeed, blocking is ineffective in the case of nonuniform insect incidences because blocking must be done before the incidence occurs. Furthermore, even though it is true that a researcher may have some information on the probable path or direction of insect movement, unless the direction of insect movement coincides with the soil fertility gradient, the choice of whether soil heterogeneity or insect incidence should be the criterion for blocking is difficult. The choice is especially difficult if both sources of variation have about the same importance.

Use of covariance analysis should be considered in experiments in which blocking couldn't adequately reduce the experimental error. By measuring an additional variable (*e.g.*, covariate X) that is known to be linearly related to the primary variable Y, the source of variation associated with the covariate can be deducted from experimental error. This adjusts the primary variable Y linearly upward or downward, depending on the relative size of its respective covariate. The adjustment accomplishes two important improvements:

1. The treatment mean is adjusted to a value that it would have had; had there been no differences in the values of the covariate.
2. The experimental error is reduced and the precision for comparing treatment means is increased.

Although blocking and covariance techniques are both used to reduce experimental error, the differences between the two techniques are such that they are usually not interchangeable. The ANCOVA can be used only when the covariate representing the heterogeneity among the experimental units can be measured quantitatively. However, that is not a necessary condition for blocking. In addition, because blocking is done before the start of the experiment, it can be used only to cope with sources of variation that are known or predictable. ANCOVA, on the other hand, can take care of unexpected sources of variation that occur during the experiment. Thus, ANCOVA is useful, as a supplementary procedure to take care of sources of variation that cannot be accounted for by blocking.

When covariance analysis is used for error control and adjustment of treatment means, the covariate must not be affected by the treatments being tested. Otherwise, the adjustment removes both the variation due to experimental error and that due to treatment effects. A good example of covariates that are free of treatment effects are those that are measured before the treatments are applied, such as soil analysis and residual effects of treatments applied in the past experiments. In other cases, care must be exercised to ensure that the covariates defined are not affected by the treatments being tested. This technique can be illustrated through the following example:

Example 1: A trial was designed to evaluate 15 rice varieties grown in soil with a toxic level of iron. The experiment was in a RCB design with three replications. Guard rows of a susceptible check variety were planted on two sides of each experimental plot. Scores for tolerance for iron toxicity were collected from each experimental plot as well as from guard rows. For each experimental plot, the score of susceptible check (averaged over two guard rows) constitutes the value of the covariate for that plot. Data on the tolerance score of each variety (Y variable) and on the score of the corresponding susceptible check (X variable) are shown below:

Scores of tolerance for iron toxicity (Y) of 15 rice varieties and those of the corresponding guard rows of a susceptible check variety (X) in a RCB trial

Variety Number	Replication-I		Replication-II		Replication-III	
	X	Y	X	Y	X	Y
1.	15	22	16	13	16	14
2.	16	14	15	23	15	23
3.	15	24	15	24	15	23
4.	16	13	15	23	15	23
5.	17	17	17	16	16	16
6.	16	14	15	23	15	23
7.	16	13	15	23	16	13
8.	16	16	17	17	16	16
9.	17	14	15	23	15	24
10.	17	17	17	17	15	26
11.	16	15	15	24	15	25
12.	16	15	15	23	15	23
13.	15	24	15	24	16	15
14.	15	25	15	24	15	23
15.	15	24	15	25	16	16

The usual analysis of variance without using the covariate (X variable) is as follows:

Source	DF	SS	Mean Square	F Value	Pr > F
Replication	2	104.0444	52.0222	2.85	0.0745
Treatment	14	265.9111	18.9937	1.04	0.4448
Error	28	510.6222	18.2365		
Total	**44**	**880.5778**			

R-Square **C.V.** **Root MSE** **Y - Mean**

0.4201	21.5436	4.2704	19.82222

Using the covariate, the analysis is the following:

Source	DF	S.S.	M.S.	F-Value	Pr > F
Replication	2	22.4802	11.2402	2.71	0.0844
Treatment	14	152.5606	10.8972	2.63	0.0151
Covariate X	1	398.7516	398.7516	96.24	0.0001
Error	27	111.8707	4.1434		

R-Square	C.V.	Root MSE	Y Mean
0.8730	10.2689	2.0355	19.8222

It is interesting to note that the use of a covariate has resulted into a considerable reduction in the error mean square and hence the CV has also reduced drastically. This has helped in catching the small differences among the treatment effects as significant. This was not possible when the covariate was not used. The covariance analysis will thus result into a more precise comparison of treatment effects.

The probability of significance of pairwise comparisons among the least square estimates of the treatment effects are given below:

Pr > |T| H0: LSMEAN(i)=LSMEAN(j)

i/j	1	2	3	4	5	6	7	8	9
1	.	0.3370	0.0666	0.4431	0.0019	0.3370	1.0000	0.0252	0.0232
2	0.3370	.	0.3370	0.8425	0.0237	1.0000	0.3370	0.1834	0.1697
3	0.0666	0.3370	.	0.2497	0.1620	0.3370	0.0666	0.6757	0.6751
4	0.4431	0.8425	0.2497	.	0.0157	0.8425	0.4431	0.1320	0.1191
5	0.0019	0.0237	0.1620	0.0157	.	0.0237	0.0019	0.2361	0.2493
6	0.3370	1.0000	0.3370	0.8425	0.0237	.	0.3370	0.1834	0.1697
7	1.0000	0.3370	0.0666	0.4431	0.0019	0.3370	.	0.0252	0.0232
8	0.0252	0.1834	0.6757	0.1320	0.2361	0.1834	0.0252	.	0.9727
9	0.0232	0.1697	0.6751	0.1191	0.2493	0.1697	0.0232	0.9727	.
10	0.0001	0.0019	0.0237	0.0012	0.3370	0.0019	0.0001	0.0361	0.0385
11	0.0874	0.4294	0.8575	0.3249	0.1046	0.4294	0.0874	0.5445	0.5439
12	0.2497	0.8425	0.4431	0.6915	0.0351	0.8425	0.2497	0.2493	0.2361
13	0.1270	0.5524	0.7066	0.4294	0.0739	0.5524	0.1270	0.4298	0.4229
14	0.0446	0.2497	0.8425	0.1803	0.2158	0.2497	0.0446	0.8096	0.8204
15	0.0589	0.3249	0.9860	0.2393	0.1452	0.3249	0.0589	0.6736	0.6809

Pr > |T| H0: LSMEAN(i)=LSMEAN(j)

i/j	10	11	12	13	14	15
1	0.0001	0.0874	0.2497	0.1270	0.0446	0.0589
2	0.0019	0.4294	0.8425	0.5524	0.2497	0.3249
3	0.0237	0.8575	0.4431	0.7066	0.8425	0.9860
4	0.0012	0.3249	0.6915	0.4294	0.1803	0.2393
5	0.3370	0.1046	0.0351	0.0739	0.2158	0.1452
6	0.0019	0.4294	0.8425	0.5524	0.2497	0.3249

7	0.0001	0.0874	0.2497	0.1270	0.0446	0.0589
8	0.0361	0.5445	0.2493	0.4298	0.8096	0.6736
9	0.0385	0.5439	0.2361	0.4229	0.8204	0.6809
10	.	0.0124	0.0031	0.0079	0.0351	0.0191
11	0.0124	.	0.5524	0.8425	0.7066	0.8425
12	0.0031	0.5524	.	0.6915	0.3370	0.4294
13	0.0079	0.8425	0.6915	.	0.5671	0.6915
14	0.0351	0.7066	0.3370	0.5671	.	0.8575
15	0.0191	0.8425	0.4294	0.6915	0.8575	.

References

Cochran, W. G., and Cox, G. M. (1957). *Experimental Design,* 2nd edition. New York: Wiley.

Searle, S. R. (1971). *Linear Models*. John Wiley & Sons, New York.

ANALYSIS OF GROUPS OF EXPERIMENTS

Susheel Sarkar
ICAR-IASRI, Library Avenue, New Delhi – 110 012
susheel.sarkar@icar.gov.in

1. Introduction

In large-scale experimental programmes it is necessary to repeat the trial of a set of treatments like varieties or manures at a number of places or in a number of seasons. The places where the trial is repeated are usually experimental stations located in the tract. The aim of repetition is to study the susceptibility of treatment effects to place variation. More generally, the aim of repetition is to find out treatments suitable for particular tracts in which case the trials are carried out simultaneous on a representative selection of sites.

Further, the purpose of the research carried out at experimental stations is to formulate the recommendations for the practitioners which consist of a population quite extensive either in space or time or both. Therefore, it becomes necessary to ensure that the results obtained from researches are valid for at least several places in the future and over a reasonably heterogeneous space.

A single experiment will precisely furnish information about only one place where the experiment is conducted and about the season in which the experiment is conducted. It has, thus, become a common practice to repeat an experiment at different places or over a number of occasions to obtain valid recommendations taking into account place to place variation or variation over time or both. In such cases of repeated experiments appropriate statistical procedures for a combined analysis of data would have to be followed by the analysis of individual experiments varying with their objectives. In combined analysis of data, the main points of interest would be

i) to estimate the average response to given treatments and
ii) to test consistency of the responses from place to place or occasion to occasion *i. e.* interaction of the treatment effects with places or years.

The utility and the significance of the estimates of average response depend on whether the response is consistence from place to place or changes with it, in other words on the absence or the presence of interaction.

The results of a set of trials may, therefore, be considered as belonging to one of the following four types:

i) the experimental errors are homogeneous and the interaction is absent,
ii) the experimental errors are homogeneous and the interaction is present,
iii) the experimental errors are heterogeneous and the interaction is absent, and
iv) the experimental errors are heterogeneous and the interaction is present.

The meaningfulness of average estimates of treatment responses would therefore, depend largely upon the absence of presence of this interaction analysis.

2. Analysis Procedure

For combined analysis or analysis for groups of Experiments following steps are to be followed

Step I: Construct an out line of combined analysis of variance over years or for places or environment, based on the basic design used. For example, the data of grain yield for four places, four treatments each treatment replicated five times is given in Table-1.

Step II: Perform usual Analysis of variance for the given data. Here the experiment conducted is in randomized complete block design. So perform analysis of four places separately for the four places. This may be done either in SAS, SPSS or EXCEL software.

Step III: We have *p* error mean squares that belongs to *p* RBD conducted and we have to test the homogeneity of variances. Now we have following two situations:

Situation I: When *p* = 2

In this situation, we apply *F*-test for testing the homogeneity of variances. Here null and alternate hypothesis are H_0: $\sigma_1^2 = \sigma_2^2$ and H_1: $\sigma_1^2 \neq \sigma_2^2$. Let Se_1^2 and Se_2^2 are the mean square errors (mse) for the two places. Then the value of *F* statistics will be Se_1^2 / Se_2^2 and this value will be tested against the Table *F* value at $n_1$and n_2 degrees of freedom at 5 % level of significance, where $n_1$and n_2 are degrees of freedom (df) for error for the two places, respectively. If the calculated value of *F* is greater than tabulated *F* value then the null hypothesis of homogeneity of variance is rejected and the data is heterogeneous in different places, otherwise it is homogeneous.

Situation II: When p > 2

In this situation, we apply Bartlett's Chi-square test. Here null and alternate hypothesis are $H_0: \sigma_1^2 = \sigma_2^2 = \cdots = \sigma_p^2$ against the alternative hypothesis

H_1 : at least two of the σ_i^2's are not equal, where σ_i^2 is the error variance for i th place/ location.

Let Se_1^2, Se_2^2, ..., Se_p^2 are the mse of *p* locations respectively and n_1, n_2, …, n_p are the df for p locations. The test statistics

$$\chi^2_{p-1} = \frac{\sum n_i \log \bar{s}_e^2 - \sum n_i \log s_{e_i}^2}{1 + \frac{1}{3(p-1)}\left(\sum \frac{1}{n_i} - \frac{1}{\sum n_i}\right)}, \qquad \text{where} \quad \bar{s}_e^2 = \frac{\sum n_i s_{e_i}^2}{\sum n_i}$$

and if $n_i = n$

$$\chi^2_{p-1} = \frac{n[p \log \bar{s}_e^2 - \sum \log s_{e_i}^2]}{1 + \frac{(p+1)}{3np}}.$$

where χ^2_{p-1} follows χ^2 distribution with *p* - 1 degree of freedom.

If the calculated value of χ^2_{p-1} is greater than tabulated χ^2_{p-1} value at *p*-1df then the null hypothesis of homogeneity of variance is rejected and the data is heterogeneous in different places, otherwise it is homogeneous.

Step IV: If error variances are not homogeneous, then for performing the combined analysis of weighted least square is required, the weight being the reciprocals of the root mean square error. The weighted analysis is carried out by defining a new variable as newres = res/ root mean square. This transformation is similar to Aitken's transformation. This new variable is thus homogeneous and thus combined analysis of variance can performed on this new variable. If error variance variances are homogeneous then there is no need to transform the data.

Step V: Now one can view the groups of experiments as a nested design with several factors nested within one another. The places/ locations are treated as big blocks, with the experiments nested within these. The combined analysis of data, therefore, can be done as that a nested design. For doing the analysis, the replication wise data of treatments at each place/ location provide useful information. An advantage of this analysis is that there is a further reduction in error sum of squares because one more source of variability is taken out from the experimental error thus reducing the experimental error. This may also lead to the reduction in the value of CV.

Step VI: Next step in the analysis is to test for the significance of place × treatment interaction. It can be seen that the question whether the interaction place × treatment is significant, that is whether the difference between treatments tend to vary from place to place can be settled by comparing the mean square for place × treatment with the estimate of error variance by the *F*-test. If the mean square is found to be non-significant it means interaction is absent. If this interaction is assumed to be non-existence, sum of squares for treatments × places and the error sum of squares can be pooled and a more precise estimate of error can be obtained for testing the significance of treatment differences. If, however, interaction is significant *i. e.* treatment effects are varying with places, then the appropriate mean square for testing the significance of treatments is the mean square due to place × treatment.

3. Illustration

Now we shall proceed for the combined analysis of a set of data.

Table-1: Data for grain yield (kg/ plot) with four treatments in five replications

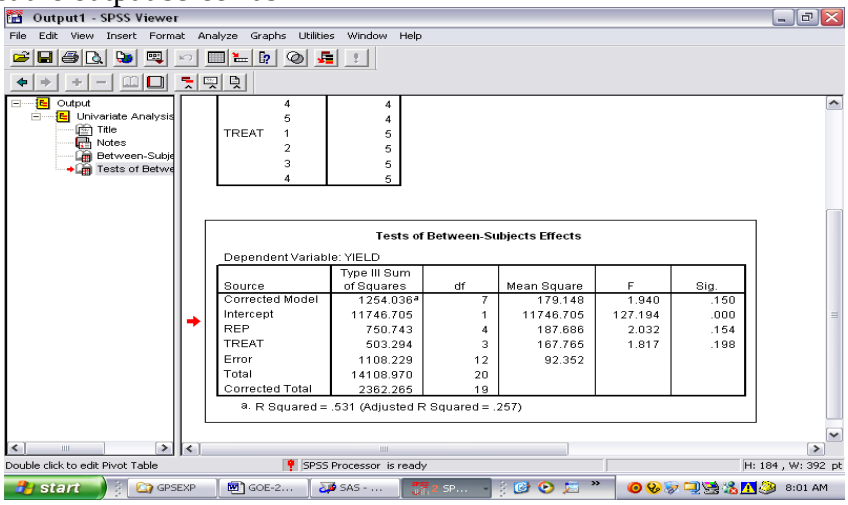
Place	Replication					
	Treatment	I	II	III	IV	V
1	1	33.6	33.7	30.9	33.3	15.0
	2	34.0	27.2	46.2	36.7	11.6
	3	30.5	33.2	15.1	33.3	29.7
	4	30.8	14.4	14.2	9.5	12.0
2	1	28.8	28.8	35.2	41.6	43.2
	2	46.4	43.2	38.4	54.4	57.6
	3	35.2	32.0	32.0	25.6	33.6
	4	51.2	40.0	49.6	51.2	49.6
3	1	30.1	38.1	21.4	17.6	14.3
	2	36.1	18.3	38.0	31.0	26.6
	3	27.2	40.7	15.5	18.1	12.3
	4	37.8	54.5	13.2	18.1	7.3
4	1	23.8	48.8	19.5	28.8	34.4
	2	15.2	39.0	39.8	52.0	31.2
	3	40.2	52.0	33.0	41.2	35.0
	4	43.2	46.8	34.5	44.5	38.0

A) By using SPSS

- First we analyze the data for each place separately by using SPSS. Make four files in SPSS Data Editor with some name say gpsexpt-1; the first column of which is replication, second column as treatment, third column as yield. Now the SPSS commands are

Analyze → General Linear model → Univariate → yield → button [Put yield under Dependent variable] → rep → button [Put rep under Fixed Factor(s)] → treat → button [Put treat under Fixed Factor(s)] → Model → Custom → rep → Build term[s] → treat → Build term[s] → Main effects → Continue → OK

- We will get the output screen as



Here error mean square is 92.352 with df 12. Similarly we analyze the data of three other places and get the error mean square. Now the error mean squares of four places along with degrees of freedom are

place	Degrees of freedom	Error mean square
1	12	78.234
2	12	28.309
3	12	108.466
4	12	67.903

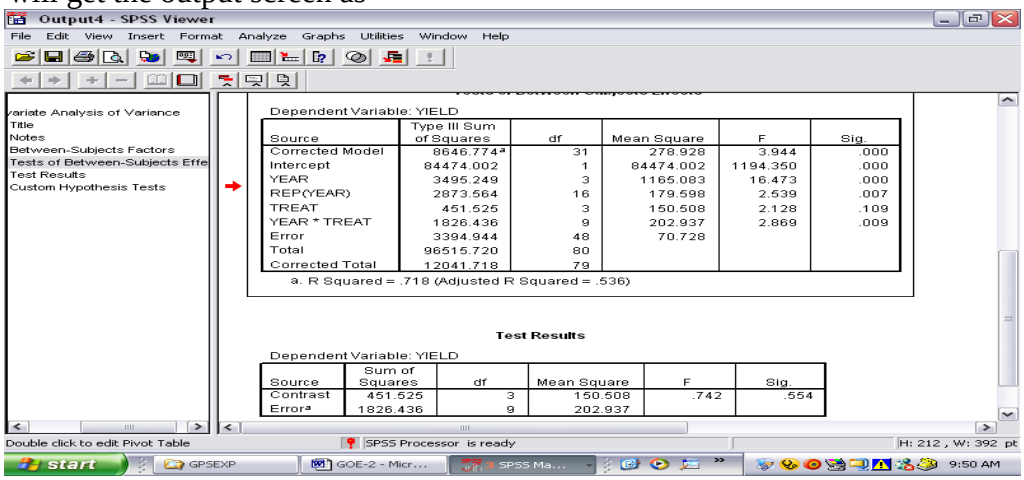
Now we test the homogeneity of error variances using analyze Bartlett's Chi-square test as described in Step III.

In our case value of χ^2 is non-significant. So we can perform the combined analysis. For combined analysis of data using SPSS we proceed as follows

- Make a combined file in SPSS Data Editor with some name say gpsexpt-comb; the first column of which is place, second column as replication, third column as replication and fourth column as yield. The SPSS commands are

```
UNIANOVA
   yield  BY place rep treat
    /METHOD = SSTYPE(3)
   /INTERCEPT = INCLUDE
   /CRITERIA = ALPHA(.05)
   /DESIGN = place rep(place) treat place*treat
   /test treat vs place*treat.
```

- We will get the output screen as



In the above analysis place × treatment interaction is significant. Therefore, treatment is tested against the place × treatment interaction.

B) By using SAS

Now we give the steps used while analyzing the data using SAS software.
First we analyze the data for four places separately using **proc glm**.

```
data klkrbd1;
input yr rep trt yld;
cards;
. . . .
. . . .
. . . .
;
proc glm;
class  rep trt;
model yld  =  rep trt/ss3;
run;
```

We test the homogeneity of error variances using analyze Bartlett's Chi-square test as described in Step III. Since value of χ^2 is non-significant. So we can perform the combined analysis. For combined analysis of data using SAS we proceed as follows

```
data klkgps1;
input yr rep trt yld;
cards;
. . . .
. . . .
. . . .
;
proc glm;
class  yr rep trt;
model yld  =  yr rep(yr) trt yr*trt/ss3;
test h=trt  e=trt*yr;
run;
```

```
proc glm;
class  yr rep trt;
model yld  =  yr rep(yr) trt/ss3;
run;
```

The different sites or places are natural environments. The natural environments are generally considered as random. All other effects in the model involve the environment either as nested or as crossed classification are considered as random. The assumption of these random effects helps in identifying the proper error terms for testing the significance of various effects. The combined analysis of data can easily be carried out using PROC GLM of SAS with Random statement with TEST option. The RANDOM statement produces a table of expected mean squares which can be used to determine appropriate denominators of *F*-statistics for all terms in the MODEL statement. These tests are produced by the TEST option at the end of the RANDOM statement. The steps to be followed are given below:

```
data klkgps1;
input yr rep trt yld;
cards;
. . . .
. . . .
. . . .
;
proc glm;
class  yr rep trt;
model yld  =  yr rep(yr) trt yr*trt/ss3;
random yr yr*trt rep(yr) /test;
run;
proc glm;
class  yr rep trt;
model yld  =  yr rep(yr) trt/ss3;
random yr rep(yr) /test;
run;
```

A complete procedure of groups of experiments analysis using SAS is described in Parsad et al. (2004) and also complete steps are given on the link "Analysis of Data" in Design Resources Server (www.iasri.res.in/design).

References

Gomez, K.A. and Gomez, A.A.(1984). *Statistical Procedures for Agricultural Research, 2nd Edition.* John Wiley and Sons, New York.

Parsad, R., Gupta, V.K., Batra, P.K., Srivastava, R., Kaur, R., Kaur, A. and Arya, P. (2004). *A Diagnostic Study of Design and Analysis of Field Experiments*. Technical Report, IASRI, New Delhi.

Panse, V.G. and Sukhatme, P.V.(1978). *Statistical Merthods for Agricultural Research Workers.* 3rd Edition., ICAR, New Delhi.

Nigam, A. K. and Gupta, V. K. 91979). *Handbook on Analysis of Agricultural Experiments.* IASRI, New Delhi.

Experimental Data Analysis using R Software

B N Mandal
ICAR-IASRI, Library Avenue, New Delhi – 110 012
bn.mandal@icar.gov.in

Introduction

R is open source software for statistical analysis and graphical display. It has several in-built facilities namely
i) an data handling and storage facility,
ii) a collection of operators for calculations on arrays and matrices,
iii) a large, integrated collection of intermediate tools for data analysis,
iv) graphical facilities for data analysis and display
v) a well developed, simple and effective programming language (called 'R' or 'S') which includes conditionals, loops, user defined recursive functions and input and output facilities.

R is very much a vehicle for newly developing methods of interactive data analysis. It has developed rapidly, and has been extended by a large collection of packages.

Origin

R can be regarded as an implementation of the S language which was developed at Bell Laboratories by Rick Becker, John Chambers and Allan Wilks, and also forms the basis of the S-Plus systems. Robert Gentleman and Ross Ihaka of the Statistics Department of the University of Auckland started the project on R in 1995 and hence the name software has been named as 'R'.

R was introduced as an environment within which many classical and modern statistical techniques can be implemented. A few of these are built into the base R environment, but many are supplied as packages. There are about 25 packages supplied with R (called "standard" and "recommended" packages) and many more are available through the CRAN family of Internet sites (via http://CRAN.R-project.org) and elsewhere.

Availability

Since R is an open source project, it can be obtained freely from the website www.r-project.org. One can download R from any CRAN mirror out of several CRAN (Comprehensive R Archive Network) mirrors. Latest available version of R is *R version 3.2.3* and it has been released on 12-10-2015.

Installation

To install R in windows operating system, simply double click on the setup file. It will automatically install the software in the system.

Usage

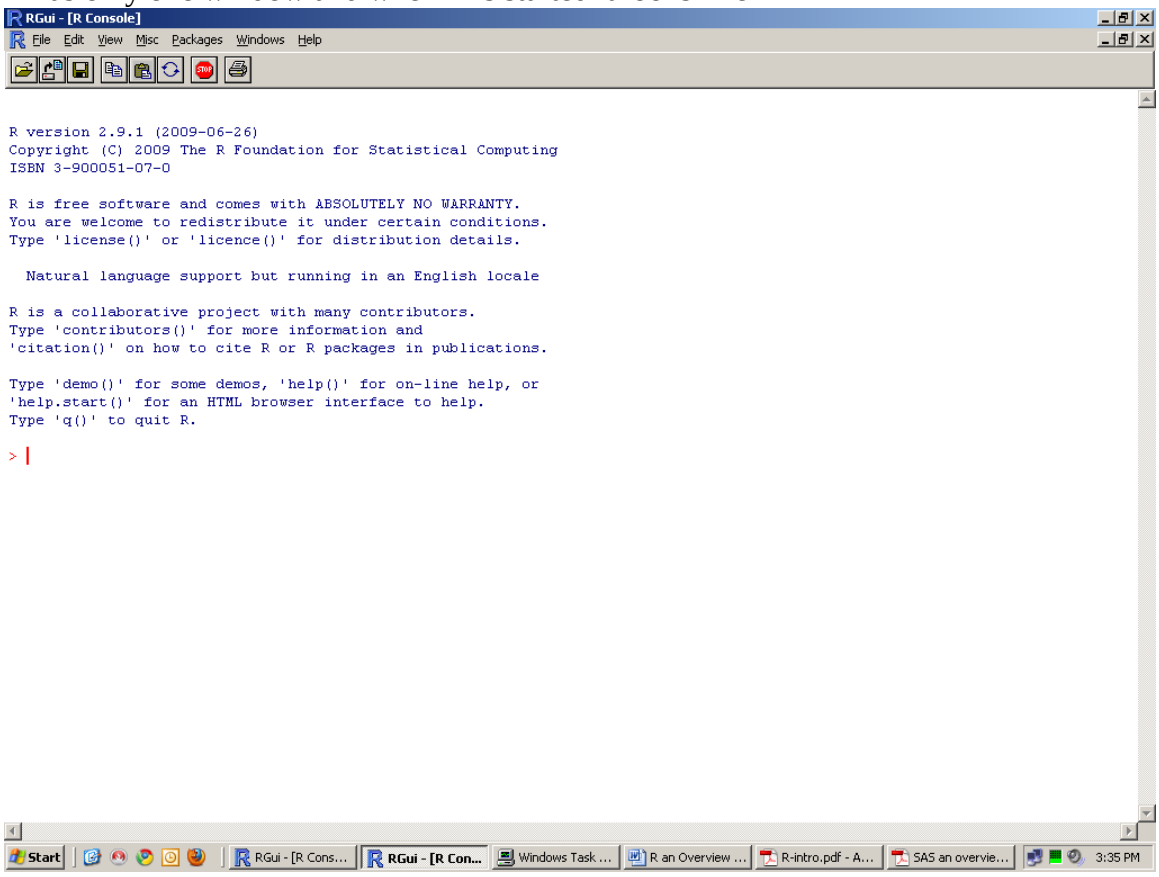
R can work under Windows, UNIX and Mac OS. In this note, we consider usage of R in Windows set up only.

Difference with other packages

There is an important difference between R and the other statistical packages. In R, a statistical analysis is normally done as a series of steps, with intermediate results being stored in objects. Thus whereas SAS and SPSS will give large amount of output from a given analysis, R will give minimal output and store the results in an object for subsequent interrogation by further R functions.

Windows of R

R has only one window and when R is started it looks like



R commands

i. R commands are case sensitive, so X and x are different symbols and would refer to different objects.
ii. Elementary commands consist of either expressions or assignments.
iii. If an expression is given as a command, it is evaluated and printed at the console.
iv. An assignment also evaluates an expression and passes the value to a variable but the result is not automatically printed.
v. Commands are separated either by a semi-colon (‘;’), or by a newline.
vi. Elementary commands can be grouped together into one compound expression by braces ‘{‘ and ‘}’.
vii. Comments can be put almost anywhere, starting with a hashmark (‘#’). Anything written after # marks to the end of the line is considered as a comment.
viii. Window can be cleared of lines by pressing Ctrl + L keys.

Executing commands from or diverting output to a file
If commands are stored in an external file, say "d:/commands.txt" they may be executed at any time in an R session with the command

> source("d:/commands.txt")

For Windows Source is also available on the File menu.

The function *sink()*,

> sink("d:/record.txt")

will divert all subsequent output from the console to an external file, 'record.txt' in D drive. The command

> sink()

restores it to the console once again.

Analysis of Experimental Data using R

1. Analysis of Data from Completely Randomized Design
Example: A feeding trial with 3 feeds namely (i) Feed without supplement, (ii) Feed with 5% supplement and (iii) Feed with 10% supplement was conducted to study their effect on weight gain of Tilapia. For this purpose twenty-five fishes were allotted at random to each of the three treatments and the three treatments and the weight gain (gms) records of the were obtained. The details of the experiment are given below:

trt	Weight gain	trt	Weight gain	trt	Weight gain
1	850.5	2	510.3	3	850.5
1	453.6	2	963.9	3	1474.2
1	878.85	2	652.05	3	510.3
1	623.7	2	1020.6	3	850.5
1	510.3	2	878.85	3	793.8
1	765.45	2	567	3	453.6
1	680.4	2	680.4	3	935.55
1	595.35	2	538.65	3	1190.7
1	538.65	2	567	3	481.95
1	850.5	2	510.3	3	623.7
1	850.5	2	425.25	3	878.85
1	793.8	2	567	3	1077.3
1	1020.6	2	623.7	3	850.5
1	708.75	2	538.65	3	680.4
1	652.05	2	737.1	3	737.1

1	623.7	2	453.6	3	737.1
1	396.9	2	481.95	3	708.75
1	822.15	2	368.55	3	708.75
1	680.4	2	567	3	652.05
1	652.05	2	595.35	3	567
1	538.65	2	567	3	453.6
1	850.5	2	595.35	3	652.05
1	680.4	3	992.25	3	567

where Feed 1- Feed without supplement,
Feed 2- Feed with 5% supplement
Feed 3- Feed with 10% supplement

1. Perform the analysis of variance of the data to test whether there is any difference between treatment effects.
2. Perform all possible pair wise treatment comparisons and identify the best treatment i.e. the treatment giving highest weight gain.

The steps of performing the above analysis using R are given below.

i) Create the data file.

The first line of the file should have a name for each variable. Each additional line of the file has as its first item a row label and the values for each variable.

Trt	weightgain
1	850.5
1	453.6
1	878.85
.	.
.	.
.	.
3	453.6
3	652.05
3	567

ii) Read the data file in R

The function *read.table()* can then be used to read the data frame directly

```
> data1 <- read.table("d:/crd.txt", header = TRUE)
```

By default numeric items (except row labels) are read as numeric variables and non-numeric variables as factors. This can be changed if necessary.

If data file is created in comma separated format, one can use read.csv() function.

iii) Store variable names in R

Through *read.csv()* or *read.table()* functions, data along with variable labels is read into R memory. However, to read the variables' names directly into R, one should use *attach(dataset)* function. For example,

>attach (data1)

causes R to directly read all the variables names eg. trt and yld . it is a good practice to use the *attach(datafile)* function immediately after reading the *datafile* into R.

iv) Identify the factors/treatments.

In this case 'trt' is factor affecting yields. To check whether the 'trt' variable is numeric or factor, use the following command.

> is.factor(trt)

[1] FALSE

> is.numeric(trt)

[1] TRUE

This shows that 'trt' is a numeric variable and hence we need to convert this to a factor variable. Otherwise, analysis will be wrong. To convert the variable 'trt' from numeric to a factor variable, use *as.factor()* function.

> trt=as.factor(trt)

> trt

[1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3

[61] 3 3 3 3 3 3 3 3 3

Levels: 1 2 3

> is.factor(trt)

[1] TRUE

v) Perform analysis of variance

Now, use *aov()* function with model specification *y~Group* and store the results in anova_output.

> anova_output=aov(weightgain~trt)

```
> anova_output
Call:
   aov(formula = weightgain ~ trt)

Terms:
                    trt Residuals
Sum of Squares   287872.4 2460182.9
Deg. of Freedom        2        66

Residual standard error: 193.0686
Estimated effects may be unbalanced
```

To get more information on the analysis done, use *summary()* function.

```
> summary(anova_output)

            Df  Sum Sq Mean Sq F value  Pr(>F)
trt          2  287872  143936  3.8614 0.02595 *
Residuals   66 2460183   37275
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

This shows that treatment effects are significant at 5% level.

vi) Perform post hoc tests

To conduct post-hoc tests, R provides a simple function to carry out the Tukey HSD test.

```
> TukeyHSD(anova_output)
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = weightgain ~ trt)

$trt
        diff        lwr       upr     p adj
2-1 -86.89891 -224.94944  51.15161 0.2931780
3-1  71.38859  -63.68998 206.46716 0.4186377
3-2 158.28750   21.65028 294.92472 0.0192552
```

From above we can see that feed3 is significantly better than feed2.

To get all the entries stored in ‘parameters’ data frame, use *names()* function.

```
> names(anova_output)
[1] "coefficients"  "residuals"     "effects"       "rank"          "fitted.values" "assign"        "qr"
[8] "df.residual"   "contrasts"     "xlevels"       "call"          "terms"         "model"
```

2. Analysis of data from Randomized Complete Block Designs

Example: An experiment was conducted to see the effects of different fertilization regime on growth of silver carp. The fertilization regimes are (A) PROSHIKA fertilization regime: weekly application of 1,250 kg cow manure, 15 kg urea and 22.5 kg TSP per hectare (B) BRAC fertilization regime: weekly application of 156 kg cow manure, 27.5 kg urea and 13.1 kg TSP per hectare (personal communication). (C) CARITAS fertilization regimes: fortnight application of 1,500 kg cow manure per hectare. (D) BAU fertilization regime: fortnight application of 5,000 kg cow manure, 50 kg urea and 50 kg TSP per hectare. (E) CRSP organic fertilization regime developed from Nile tilapia ponds: 250 kg cow dung (DM) ha^{-1} wk^{-1} supplemented with urea and TSP to give 28 kg N and 7 kg P ha^{-1} wk^{-1}. The experiment was conducted using a Randomized complete Block Design (RCB) design with 3 replications. The yield from each pond was recorded. The details of the experiment are given below:

trt	trtn	rep	yield
PROSHIKA	1	1	1539.69
BRAC	2	1	1261.85
CARITAS	3	1	1389.19
BAU	4	1	1192.39
CRSP	5	1	1250.27
PROSHIKA	1	2	1296.58
BRAC	2	2	1227.12
CARITAS	3	2	1273.43
BAU	4	2	1180.82
CRSP	5	2	1296.58
PROSHIKA	1	3	1122.93
BRAC	2	3	1250.27
CARITAS	3	3	1180.82
BAU	4	3	1146.09
CRSP	5	3	1574.42

Note: Strains of mustard in bold are the four checks.

i) Perform the analysis of variance of the data to test whether there is any difference between treatment effects.

ii) Perform all possible pair wise treatment comparisons and identify the best treatment i.e. the treatment giving highest yield. Also identify the other treatments which are non-significantly different from this treatment.

The steps of analysis are

i) Create the data file.

ii) Read and attach the data file in R.

```
>data2=read.table("d:/data2.txt",header=TRUE)
>attach(data2)
```

Make sure 'trtn' and 'rep' is factor variable using trtn=as.factor(trtn) and rep=as.factor(rep).

iv) Obtain treatment wise mean yield

To obtain mean yield under first treatment, use following command

>mean(yield[data2$trtn==1]).

Similarly for other treatments, mean can be obtained.

v) Perform analysis of variance

>out2=aov(yield~trtn+rep)
> summary(out2)

This gives following output

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
trtn	4	68758	17189.4	0.8841	0.5148
rep	2	17171	8585.4	0.4416	0.6578
Residuals	8	155534	19441.7		

This shows that the treatment effects are not significantly different.

To carry out two-way and other complex analysis of variance, the model may be modified as per situations.

Model	Interpreation
$y \sim x_1 * x_2$	y is explained by two factors x_1 and x_2 as well as their interactions (suitable for analysis of data from factorial experiments)
$y \sim x_1 + x_2 + x_1:x_2$	-Do-

Packages

All R functions and datasets are stored in packages. The contents of a package are available only when the package is loaded. This is done to run the codes efficiently without much memory usage. To see which packages are installed at your machine, use the command

> library()

Before using any package in R, it has to be downloaded from R website. The install that package from packages drop down menu in R. Then to load the package in a session, use a command like

> library(fishmethods)

Users connected to the Internet can use the install.packages() and update.packages() functions to install and update packages. Use *search()* to display the list of packages that are loaded.

Standard packages

The standard (or base) packages are considered part of the R source code. They contain the basic functions that allow R to work and the datasets and standard statistical and graphical functions that are described in this manual. They should be automatically available in any R installation.

Contributed packages and CRAN

There are a number of contributed packages for R, written by many authors. Various packages deal with various analyses. Most of the packages are available for download from CRAN (http://CRAN.R-project.org/ and its mirrors), and other repositories such as Bioconductor (http://www.bioconductor.org/). The R FAQ contains a list that was current at the time of release, but the collection of available packages changes frequently. As on March 08, 2010, the CRAN package repository contains 2239 available packages.

Getting Help

Complete help files in HTML and PDF forms are available in R. To get help on a particular command/function etc., type *help (command name)*. For example, to get help on function 'mean, type *help(mean)* as shown below

> help (mean)

This will open the help file with the page containing the description of the function mean. Another way to get help is to use "?" followed by function name. For example,

>?mean

will open the same window again.

In this lecture note, all R commands and corresponding outputs are given as shaded text to differentiate the normal texts. Since R is case-sensitive, i.e. typing *Help(mean)*, would generate an error message,

> Help (mean)
Error: could not find function "Help"

Further Readings

Various documents are available in www.r-project.org from beginners' level to most advanced level. Also, along with R, the following manuals are available in pdf form:

1. An Introduction to R
2. R Reference Manual
3. R Import/Export
4. R Language Definition
5. Writing R Extensions
6. R internals
7. R installations and Administration

References

1. Design Resources Server. Indian Agricultural Statistics Research Institute *(*ICAR*)*, New Delhi 110 012, India. www.iasri.res.in/design (accessed on 12.05.2011)

2. R Development Core Team (2011). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

FACTORIAL EXPERIMENTS

SEEMA JAGGI
ICAR-IASRI, Library Avenue, New Delhi – 110 012
seema.jaggi@icar.gov.in

1. Introduction

Factorial Experiments are experiments that investigate the effects of two or more factors or input parameters on the output response of a process. Factorial experiment design, or simply factorial design, is a systematic method for formulating the steps needed to successfully implement a factorial experiment. Estimating the effects of various factors on the output of a process with a minimal number of observations is crucial to being able to optimize the output of the process.

In a factorial experiment, the effects of varying the levels of the various factors affecting the process output are investigated. Each complete trial or replication of the experiment takes into account all the possible combinations of the varying levels of these factors. Effective factorial design ensures that the least number of experiment runs are conducted to generate the maximum amount of information about how input variables affect the output of a process.

For example, an experiment on rooting of cuttings involving two factors, each at two levels, such as two hormones at two doses, is referred to as a 2 x 2 or a 2^2 factorial experiment. Its treatments consist of the following four possible combinations of the two levels in each of the two factors.

Treatment number	Treatment Combination	
	Hormone	Dose (ppm)
1	NAA	10
2	NAA	20
3	IBA	10
4	IBA	20

The total number of treatments in a factorial experiment is the product of the number of levels of each factor; in the 2^2 factorial example, the number of treatments is 2 x 2 = 4, in the 2^3 factorial, the number of treatments is 2 x 2 x 2 = 8. The number of treatments increases rapidly with an increase in the number of factors or an increase in the levels in each factor. For a factorial experiment involving 5 clones, 4 espacements, and 3 weed-control methods, the total number of treatments would be 5 x 4 x 3 = 60. Thus, indiscriminate use of factorial experiments has to be avoided because of their large size, complexity, and cost. Furthermore, it is not wise to commit oneself to a large experiment at the beginning of the investigation when several small preliminary experiments may offer promising results. For example, a tree breeder has collected 30 new clones from a

neighbouring country and wants to assess their reaction to the local environment. Because the environment is expected to vary in terms of soil fertility, moisture levels, and so on, the ideal experiment would be one that tests the 30 clones in a factorial experiment involving such other variable factors as fertilizer, moisture level, and population density. Such an experiment, however, becomes extremely large as factors other than clones are added. Even if only one factor, say nitrogen or fertilizer with three levels were included, the number of treatments would increase from 30 to 90. Such a large experiment would mean difficulties in financing, in obtaining an adequate experimental area, in controlling soil heterogeneity, and so on. Thus, the more practical approach would be to test the 30 clones first in a single-factor experiment, and then use the results to select a few clones for further studies in more detail. For example, the initial single-factor experiment may show that only five clones are outstanding enough to warrant further testing. These five clones could then be put into a factorial experiment with three levels of nitrogen, resulting in an experiment with 15 treatments rather than the 90 treatments needed with a factorial experiment with 30 clones.

The amount of change produced in the process output for a change in the 'level' of a given factor is referred to as the 'main effect' of that factor. Table 1 shows an example of a simple factorial experiment involving two factors with two levels each. The two levels of each factor may be denoted as 'low' and 'high', which are usually symbolized by '-' and '+' in factorial designs, respectively.

Table 1. A Simple 2-Factorial Experiment

	A (-)	**A (+)**
B (-)	20	40
B (+)	30	52

The main effect of a factor is basically the 'average' change in the output response as that factor goes from '-' to '+'. Mathematically, this is the average of two numbers: 1) the change in output when the factor goes from low to high level as the other factor stays low, and 2) the change in output when the factor goes from low to high level as the other factor stays high.

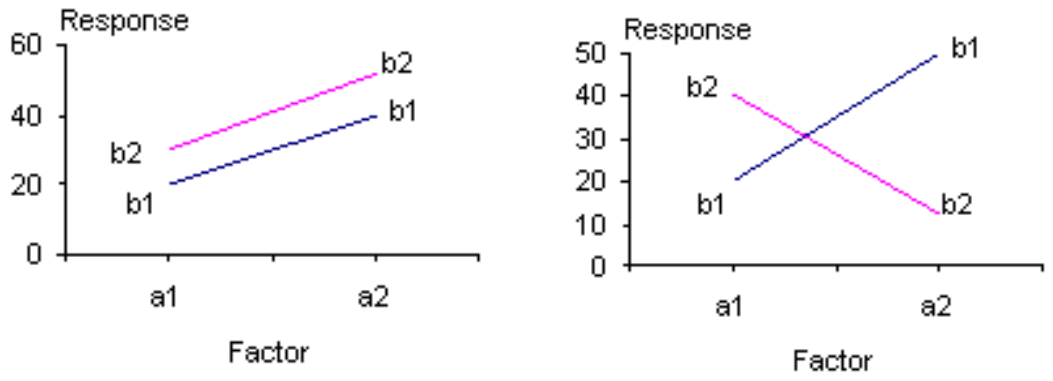
In the example in Table 1, the output of the process is just 20 (lowest output) when both A and B are at their '-' level, while the output is maximum at 52 when both A and B are at their '+' level. The main effect of A is the average of the change in output response when B stays '-' as A goes from '-' to '+', or (40-20) = 20, and the change in output response when B stays '+' as A goes from '-' to '+', or (52-30) = 22. The main effect of A, therefore, is equal to 21.

Similarly, the main effect of B is the average change in output as it goes from '-' to '+' , i.e., the average of 10 and 12, or 11. Thus, the main effect of B in this process is 11. Here, one can see that the factor A exerts a greater influence on the output of process, having a main effect of 21 versus factor B's main effect of only 11. It must be noted that aside from 'main effects', factors can likewise result in 'interaction effects.' Interaction effects are changes in the process output caused by two or more factors that are interacting with each

other. Large interactive effects can make the main effects insignificant, such that it becomes more important to pay attention to the interaction of the involved factors than to investigate them individually. In Table 1, as effects of A (B) is not same at all the levels of B (A) hence, A and B are interacting.

Thus, **interaction** is the failure of the differences in response to changes in levels of one factor, to retain the same order and magnitude of performance through out all the levels of other factors OR the factors are said to interact if the effect of one factor changes as the levels of other factor(s) changes.

Graphical representation of lack of interaction between factors and interaction between factors are shown below. In case of two parallel lines, the factors are non-interacting.



If interactions exist which is fairly common, we should plan our experiments in such a way that they can be estimated and tested. It is clear that we cannot do this if we vary only one factor at a time. For this purpose, we must use multilevel, multifactor experiments.

The running of factorial combinations and the mathematical interpretation of the output responses of the process to such combinations is the essence of factorial experiments. It allows to understand which factors affect the process most so that improvements (or corrective actions) may be geared towards these.

We may define factorial experiments as experiments in which the effects (main effects and interactions) of more then one factor are studied together. In general if there are 'n' factors, say, F_1, F_2,..., F_n and i^{th} factor has s_i levels, i=1,...,n, then total number of treatment combinations is $\prod_{i=1}^{n} s_i$. Factorial experiments are of two types.

Experiments in which the number of levels of all the factors are same i.e all s_i's are equal are called **symmetrical factorial experiments** and the experiments in at least two of the s_i's are different are called as **asymmetrical factorial experiments**. Factorial experiments provide an opportunity to study not only the individual effects of each factor but also there interactions. They have the further advantage of economising on experimental resources. When the experiments are conducted factor by factor much more resources are required for the same precision than when they are tried in factorial experiments.

2. Experiments with Factors Each at Two Levels

The simplest of the symmetrical factorial experiments are the experiments with each of the factors at 2 levels. If there are 'n' factors each at 2 levels, it is called as a 2^n factorial where the power stands for the number of factors and the base the level of each factor. Simplest of the symmetrical factorial experiments is the 2^2 factorial experiment i.e. 2 factors say A and B each at two levels say 0 (low) and 1 (high). There will be 4 treatment combinations which can be written as

00 = $a_0 b_0$ = 1; A and B both at first (low) levels
10 = $a_1 b_0$ = a ; A at second (high) level and B at first (low) level
01 = $a_0 b_1$ = b ; A at first level (low) and B at second (high) level
11 = $a_1 b_1$ = ab; A and B both at second (high) level.

In a 2^2 factorial experiment wherein r replicates were run for each combination treatment, the main and interactive effects of A and B on the output may be mathematically expressed as follows:

A = [ab + a - b - (1)] / 2r; (main effect of factor A)
B = [ab + b - a - (1)] / 2r; (main effect of factor B)
AB = [ab + (1) - a - b] / 2r; (interactive effect of factors A and B)

where r is the number of replicates per treatment combination; a is the total of the outputs of each of the r replicates of the treatment combination a (A is 'high and B is 'low); b is the total output for the n replicates of the treatment combination b (B is 'high' and A is 'low); ab is the total output for the r replicates of the treatment combination ab (both A and B are 'high'); and (1) is the total output for the r replicates of the treatment combination (1) (both A and B are 'low').

Had the two factors been independent, then [ab + (1) - a - b] / 2n will be of the order of zero. If not then this will give an estimate of interdependence of the two factors and it is called the interaction between A and B. It is easy to verify that the interaction of the factor B with factor A is BA which will be same as the interaction AB and hence the interaction does not depend on the order of the factors. It is also easy to verify that the main effect of factor B, a contrast of the treatment totals is orthogonal to each of A and AB.

Table 2. Two-level 2-Factor Full-Factorial

RUN	Comb.	M	A	B	AB
1	(1)	+	-	-	+
2	a	+	+	-	-
3	b	+	-	+	-
$4 = 2^2$	ab	+	+	+	+

Consider the case of 3 factors A, B, C each at two levels (0 and 1) i.e. 2^3 factorial experiment. There will be 8 treatment combinations which are written as

000 = $a_0 b_0 c_0$ = (1); A, B and C all three at first level
100 = $a_1 b_0 c_0$ = a ; A at second level and B and C at first level
010 = $a_0 b_1 c_0$ = b ; A and C both at first level and B at second level
110 = $a_1 b_1 c_0$ = ab; A and B both at second level and C is at first level.
001 = $a_0 b_0 c_1$ = c ; A and B both at first level and C at second level.
101 = $a_1 b_0 c_1$ = ac; A and C at second level, B at first level
011 = $a_0 b_1 c_1$ = bc; A is at first level and B and C both at second level
111 = $a_1 b_1 c_1$ = abc; A, B and C all the three at second level

In a three factor experiment there are three main effects A, B, C; 3 first order or two factor interactions AB, AC, BC; and one second order or three factor interaction ABC.

Table 3. Two-level 3-Factor Full-Factorial Experiment Pattern

RUN	Comb.	M	A	B	AB	C	AC	BC	ABC
1	(1)	+	-	-	+	-	+	+	-
2	a	+	+	-	-	-	-	+	+
3	b	+	-	+	-	-	+	-	+
4	ab	+	+	+	+	-	-	-	-
5	c	+	-	-	+	+	-	-	+
6	ac	+	+	-	-	+	+	-	-
7	bc	+	-	+	-	+	-	+	-
$8 = 2^3$	abc	+	+	+	+	+	+	+	+

Main effect $A = \frac{1}{4}\{[abc] - [bc] + [ac] - [c] + [ab] - [b] + [a] - [1]\}$

$$= \frac{1}{4}(a-1)(b+1)(c+1)$$

$$AB = \frac{1}{4}[(abc)-(bc)-(ac)+c) - (ab) - (b) - (a)+ (1)]$$

$$ABC = \frac{1}{4}[(abc) - (bc) - (ac) + (c) - (ab) + (b) + (a) - (1)]$$

or equivalently,

$$AB = \frac{1}{4}(a-1)(b-1)(c+1)$$

$$ABC = \frac{1}{4}(a-1)(b-1)(c-1)$$

The method of representing the main effect or interaction as above is due to Yates and is very useful and quite straightforward. For example, if the design is 2^4 then

$$A = (1/2^3)[(a-1)(b+1)(c+1)(d+1)]$$
$$ABC = (1/2^3)[(a-1)(b-1)(c-1)(d+1)]$$

In case of a 2^n factorial experiment, there will be 2^n (=v) treatment combinations with 'n' main effects, $\binom{n}{2}$ first order or two factor interactions, $\binom{n}{3}$ second order or three factor interactions, $\binom{n}{4}$ third order or four factor interactions and so on , $\binom{n}{r}$, $(r-1)^{th}$ order or r factor interactions and $\binom{n}{n}$ $(n-1)^{th}$ order or n factor interaction. Using these v treatment

combinations, the experiment may be laid out using any of the suitable experimental designs viz. completely randomised design or block designs or row-column designs, etc.

Steps for Analysis

1. The Sum of Squares (S.S.) due to treatments, replications [in case randomised block design is used], due to rows and columns (in case a row-column design has been used), total S.S. and error S.S. is obtained as per established procedures. No replication S.S. is required in case of a completely randomised design.
2. The treatment sum of squares is divided into different components viz. main effects and interactions each with single d.f. The S.S. due to these factorial effects is obtained by dividing the squares of the factorial effect total by $r.2^n$. For obtaining 2^n-1 factorial effects in a 2^n factorial experiment, the 'n' main effects is obtained by giving the positive signs to those treatment totals where the particular factor is at second level and minus to others and dividing the value so obtained by $r.2^{n-1}$, where r is the number of replications of the treatment combinations. All interactions can be obtained by multiplying the corresponding coefficients of main effects.

   For a 2^2 factorial experiment, the S.S. due to a main effect or the interaction effect is obtained by dividing the square of the effect total by 4r. Thus,

   S.S. due to main effect of A $= [A]^2/ 4r$, with 1 d.f.

   S.S. due to main effect of B $= [B]^2/ 4r$, with 1 d.f

   S.S. due to interaction AB $= [AB]^2/ 4r$, with 1 d.f.

3. Mean squares (M.S) is obtained by dividing each S.S. by corresponding degrees of freedom.
4. After obtaining the different S.S.'s, the usual Analysis of variance (ANOVA) table is prepared and the different effects are tested against error mean square and conclusions drawn.
5. Standard errors (S.E.'s) for main effects and two factor interactions:

   S.E of difference between main effect means $= \sqrt{\frac{2MSE}{r.2^{n-1}}}$

   S.E of difference between A means at same level of B=S.E of difference between B means at same level of A= $\sqrt{\frac{2MSE}{r.2^{n-2}}}$

   In general,

   S.E. for difference between means in case of a r-factor interaction = $\sqrt{\frac{2MSE}{r.2^{n-r}}}$

   The critical differences are obtained by multiplying the S.E. by the student's t value at α% level of significance at error degrees of freedom.

The ANOVA for a 2^2 factorial experiment with r replications conducted using a RCBD is as follows:

ANOVA

Sources of Variation	DF	S.S.	M.S.	F
Between Replications	r-1	SSR	MSR=SSR/(r-1)	MSR/MSE
Between treatments	2^2-1=3	SST	MST=SST/3	MST/MSE
A	1	SSA=$[A]^2$/4r	MSA=SSA	MSA/MSE
B	1	SSB=$[B]^2$/4r	MSB=SSB	MSB/MSE
AB	1	SSAB=$[AB]^2$/4r	MSAB=SSAB	MSAB/MSE
Error	(r-1)(2^2-1) =3(r-1)	SSE	MSE=SSE/3(r-1)	
Total	r.2^2-1=4r-1	TSS		

ANOVA for a 2^3-factorial experiment conducted in RCBD with r replications is given by

ANOVA

Sources of Variation	DF	SS	MS	F
Between Replications	r-1	SSR	MSR=SSR/(r-1)	MSR/MSE
Between treatments	2^3 -1=7	SST	MST=SST/7	MST/MSE
A	1	SSA	MSA=SSA	MSA/MSE
B	1	SSB	MSB=SSB	MSB/MSE
C	1	SSC	MSC=SSC	MSC/MSE
AB	1	SSAB	MSAB=SSAB	MSAB/MSE
AC	1	SSAC	MSAC=SSAC	MSAC/MSE
BC	1	SSBC	MSBC=SSBC	MSBC/MSE
ABC	1	SSABC	MSABC=SSABC	MSABC/MSE
Error	(r-1)(2^3-1) =7(r-1)	SSE	MSE=SSE/7(r-1)	
Total	r.2^3-1=8r-1	TSS		

Similarly ANOVA table for a 2^n factorial experiment can be made.

3. Experiments with Factors Each at Three Levels

When factors are taken at three levels instead of two, the scope of an experiment increases. It becomes more informative. A study to investigate if the change is linear or quadratic is possible when the factors are at three levels. The more the number of levels, the better, yet the number of the levels of the factors cannot be increased too much as the size of the experiment increases too rapidly with them. Consider two factors A and B,

each at three levels say 0, 1 and 2 (3^2-factorial experiment). The treatment combinations are

00 $= a_0b_0 = 1$; A and B both at first levels
10 $= a_1b_0 = a$; A is at second level and B is at first level
20 $= a_2b_0 = a^2$; A is at third level and b is at first level
01 $= a_0b_1 = b$; A is at first level and B is at second level
11 $= a_1b_1 = ab$; A and B both at second level
21 $= a_2b_1 = a^2b$; A is at third level and B is at second level
02 $= a_0b_2 = b^2$; A is at first level and B is at third level
12 $= a_1b_2 = ab^2$; A is at second level and B is at third level
22 $= a_2b_2 = a^2b^2$; A and B both at third level

Any standard design can be adopted for the experiment.

The main effects A, B can respectively be divided into linear and quadratic components each with 1 d.f. as A_L, A_Q, B_L and B_Q. Accordingly AB can be partitioned into four components as $A_L B_L$, $A_L B_Q$, $A_Q B_L$, $A_Q B_Q$.

The coefficients of the treatment combinations to obtain the above effects are given as

Treatment Totals→ Factorial Effects ↓	**[1]**	**[a]**	**[a^2]**	**[b]**	**[ab]**	**[a^2b]**	**[b^2]**	**[ab^2]**	**[a^2b^2]**	**Divisor**
M	+1	+1	+1	+1	+1	+1	+1	+1	+1	$9r=r\times3^2$
A_L	-1	0	+1	-1	0	+1	-1	0	+1	$6r=r\times2\times 3$
A_Q	+1	-2	+1	+1	-2	+1	+1	-2	+1	$18r=6\times3$
B_L	-1	-1	-1	0	0	0	+1	+1	+1	$6r=r\times2\times3$
$A_L B_L$	+1	0	-1	0	0	0	-1	0	+1	$4r=r\times2\times2$
A_Q BL	-1	+2	-1	0	0	0	+1	-2	+1	$12r=r\times6\times2$
B_Q	+1	+1	+1	-2	-2	-2	+1	+1	+1	$18r=r\times3\times6$
$A_L B_Q$	-1	0	+1	+2	0	-2	-1	0	+1	$12r=r\times2\times6$
$A_Q B_Q$	+1	-2	+1	-2	+4	-2	+1	-2	+1	$36r=r\times6\times6$

The rule to write down the coefficients of the linear (quadratic) main effects is to give a coefficient as +1 (+1) to those treatment combinations containing the third level of the corresponding factor, coefficient as 0(-2) to the treatment combinations containing the second level of the corresponding factor and coefficient as -1(+1) to those treatment combinations containing the first level of the corresponding factor. The coefficients of the treatment combinations for two factor interactions are obtained by multiplying the corresponding coefficients of two main effects. The various factorial effect totals are given as

$[A_L] = +1[a^2b^2]+0[ab^2] -1[b^2]+1[a^2b]+0[ab] -1[b]+1[a^2]+0[a] -1[1]$

$[A_Q] = +1[a^2b^2] -2[ab^2]+1[b^2]+1[a^2b] -2[ab]+1[b]+1[a^2] -2[a]+1[1]$

$[B_L] = +1[a^2b^2]+1[ab^2]+1[b^2]+0[a^2b]+0[ab]+0[b] -1[a^2] -1[a] -1[1]$

$[A_LB_L] = +1[a^2b^2]+0[ab^2] -1[b^2]+0[a^2b]+0[ab]+0[b] -1[a^2]+0[a] -1[1]$

$[A_QB_L] = +1[a^2b^2] -2[ab^2]+1[b^2]+0[a^2b]+0[ab]+0[b] -1[a^2]+2[a] -1[1]$

$[B_Q] = +1[a^2b^2]+1[ab^2]+1[b^2] -2[a^2b] -2[ab] -2[b] -1[a^2] -1[a] -1[1]$

$[A_LB_Q] = +1[a^2b^2]+0[ab^2] -1[b^2] -2[a^2b]+0[ab]+2[b]+1[a^2]+0[a] -1[1]$

$[A_QB_Q] = +1[a^2b^2] -2[ab^2]+1[b^2] -2[a^2b]+4[ab] -2[b]+1[a^2] -2[a]+1[1]$

Factorial effects are given by

$A_L = [A_L]/r.3$ $A_Q = [A_Q]/r.3$ $B_L = [B_L]/r.3$ $A_LB_L = [A_LB_L]/r.3$

$A_QB_L = [A_QB_L]/r.3$ $B_Q = [B_Q]/r.3$ $A_LB_Q = [A_LB_Q]/r.3$ $A_QB_Q = [A_QB_Q]/r.3$

The sum of squares due to various factorial effects is given by

$$SSA_L = \frac{[A_L]^2}{r.2.3}; \quad SSA_q = \frac{[A_Q]^2}{r.6.3}; \quad SSB_L = \frac{[B_L]^2}{r.3.2}; \quad SSA_LB_L = \frac{[A_LB_L]^2}{r.2.2};$$

$$SSA_QB_L = \frac{[A_QB_L]^2}{r.6.2}; \quad SSB_Q = \frac{[B_Q]^2}{r.3.6}; \quad SSA_LB_Q = \frac{[A_LB_Q]^2}{r..2.6}; \quad SSA_QB_Q = \frac{[A_QB_Q]^2}{r.6.6};$$

If a RBD is used with r-replications then the outline of analysis of variance is

ANOVA

Sources of Variation	**D.f**	**SS**	**MS**
Between Replications	r-1	SSR	MSR=SSR/(r-1)
Between treatments	3^2-1=8	SST	MST=SST/8
A	2	SSA	MSA=SSA/2
A_L	1	SSA_L	MSA_L= SSA_L
A_Q	1	SSA_Q	MSA_Q=SSA_Q
B	2	SSB	MSB=SSB/2
B_L	1	SSB_L	MSB_L= SSB_L
B_Q	1	SSB_Q	MSB_Q=SSB_Q
AB	4	SSAB	MSAB=SSAB/2
A_LB_L	1	SSA_LB_L	MSA_LB_L=SSA_LB_L
A_QB_L	1	SSA_QB_L	MSA_QB_L=SSA_QB_L
A_LB_Q	1	SSA_LB_Q	MSA_LB_Q=SSA_LB_Q
A_QB_Q	1	SSA_QB_Q	MSA_QB_Q=SSA_QB_Q
Error	(r-1)(3^2-`1) =8(r-1)	SSE	MSE=SSE/8(r-1)
Total	r.3^2-1=9r-1	TSS	

In general, for n factors each at 3 levels, the sum of squares due to any linear (quadratic) main effect is obtained by dividing the square of the linear (quadratic) main effect total by $r.2.3^{n-1}$($r.6.3^{n-1}$). Sum of squares due to a 'p' factor interaction is given by taking the square of the total of the particular interaction component divided by $r.(a_1\ a_2\ ...a_p).\ 3^{n-p}$, where $a_1, a_2,...,a_p$ are taken as 2 or 6 depending upon the linear or quadratic effect of particular factor.

4. Confounding in Factorial Experiments

When the number of factors and/or levels of the factors increase, the number of treatment combinations increase very rapidly and it is not possible to accommodate all these treatment combinations in a single homogeneous block. For example, a 2^5 factorial would have 32 treatment combinations and blocks of 32 plots are quite big to ensure homogeneity within them. A new technique is therefore necessary for designing experiments with a large number of treatments. One such device is to take blocks of size less than the number of treatments and have more than one block per replication. The treatment combinations are then divided into as many groups as the number of blocks per replication. The different groups of treatments are allocated to the blocks.

There are many ways of grouping the treatments into as many groups as the number of blocks per replication. It is known that for obtaining the interaction contrast in a factorial experiment where each factor is at two levels, the treatment combinations are divided into two groups. Such two groups representing a suitable interaction can be taken to form the contrasts of two blocks each containing half the total number of treatments. In such case the contrast of the interaction and the contrast between the two block totals are given by the same function. They are, therefore, mixed up and can not be separated. In other words, the interaction has been confounded with the blocks. Evidently the interaction confounded has been lost but the other interactions and main effects can now be estimated with better precision because of reduced block size. This device of reducing the block size by taking one or more interaction contrasts identical with block contrasts is known as **confounding**. Preferably only higher order interactions, that is, interactions with three or more factors are confounded, because their loss is immaterial. As an experimenter is generally interested in main effects and two factor interactions, these should not be confounded as far as possible.

When there are two or more replications, if the same set of interactions are confounded in all the replications, confounding is called **complete** and if different sets of interaction are confounded in different replications, confounding is called **partial**. In complete confounding all the information on confounded interactions are lost. But in partial confounding, the confounded interactions can be recovered from those replications in which they are not confounded.

Advantages of Confounding

It reduces the experimental error considerably by stratifying the experimental material into homogeneous subsets or subgroups. The removal of the variation among incomplete blocks (freed from treatments) within replicates results in smaller error mean square as compared with a RBD, thus making the comparisons among some treatment effects more precise.

Disadvantages of Confounding

- In the confounding scheme, the increased precision is obtained at the cost of sacrifice of information (partial or complete) on certain relatively unimportant interactions.
- The confounded contrasts are replicated fewer times than are the other contrasts and as such there is loss of information on them and they can be estimated with a lower degree of precision as the number of replications for them is reduced.

- An indiscriminate use of confounding may result is complete or partial loss of information on the contrasts or comparisons of greatest importance. As such the experimenter should confound only those treatment combinations or contrasts which are of relatively less or of importance at all.
- The algebraic calculations are usually more difficult and the statistical analysis is complex, especially when some of the units (observations) are missing.

Confounding in 2^3 Experiment

Although 2^3 is a factorial with small number of treatment combinations but for illustration purpose, this example has been considered. Let the three factors be A, B, C each at two levels.

Factorial Effects → **Treat. Combinations ↓**	A	B	C	AB	AC	BC	ABC
(1)	-	-	-	+	+	+	-
(a)	+	-	-	-	-	+	+
(b)	-	+	-	-	+	-	-
(ab)	+	+	-	+	-	-	-
(c)	-	-	+	+	-	-	+
(ac)	+	-	+	-	+	-	-
(bc)	-	+	+	-	-	+	-
(abc)	+	+	+	+	+	+	+

The various factorial effects are as follows:

A = (abc) + (ac) + (ab) + (a) - (bc) - (c) - (b) - (1)
B = (abc) + (bc) + (ab) + (b) - (ac) - (c) - (a) - (1)
C = (abc) + (bc) + (ac) + (c) - (ab) - (b) - (a) - (1)
AB = (abc) + (c) + (ab) + (1) - (bc) - (ac) - (b) - (a)
AC = (abc) + (ac) + (b) + (1) - (bc) - (c) - (ab) - (a)
BC = (abc) + (bc) + (a) + (1) - (ac) - (c) - (ab) - (b)
ABC = (abc) + (c) + (b) + (a) - (bc) - (ac) - (ab) - (1)

Let the highest order interaction ABC be confounded and we decide to use two blocks of 4 units (plots) each per replicate.

Thus in order to confound the interaction ABC with blocks all the treatment combinations with positive sign are allocated at random in one block and those with negative signs in the other block. Thus the following arrangement gives ABC confounded with blocks and hence we loose information on ABC.

Replication I

Block 1:	(1)	(ab)	(ac)	(bc)
Block 2 :	(a)	(b)	(c)	(abc)

It can be observed that the contrast estimating ABC is identical to the contrast estimating block effects.

The other six factorial effects viz. A, B, C, AB, AC, BC each contain two treatments in block 1 (or 2) with the positive signs and two with negative sign so that they are orthogonal with block totals and hence these differences are not influenced among blocks

and can thus be estimated and tested as usual without any difficulty. Whereas for confounded interaction, all the treatments in one group are with positive sign and in the other with negative signs.

Similarly if AB is to be confounded, then the two blocks will consists of

Block 1	(abc)	(c)	(ab)	(1)
Block 2	(bc)	(ac)	(b)	(a)

Here AB is confounded with block effects and cannot be estimated independently whereas all other effects A, B, C, AC, Bc and ABC can be estimated independently.

When an interaction is confounded in one replicate and not in another, the experiment is said to be partially confounded. Consider again 2^3 experiment with each replicate divided into two blocks of 4 units each. It is not necessary to confound the same interaction in all the replicates and several factorial effects may be confounded in one single experiment. For example, the following plan confounds the interaction ABC, AB, BC and AC in replications I, II, III and IV respectively.

Rep. I		**Rep. II**		**Rep. III**		**Rep. IV**	
Block 1	**Block 2**	**Block 3**	**Block 4**	**Block 5**	**Block 6**	**Block 7**	**Block 8**
(abc)	(ab)	(abc)	(ac)	(abc)	(ab)	(abc)	(ab)
(a)	(ac)	(c)	(bc)	(bc)	(ac)	(ac)	(bc)
(b)	(bc)	(ab)	(a)	(a)	(b)	(b)	(a)
(c)	(1)	(1)	(b)	(1)	(c)	(1)	(c)

In the above arrangement, the main effects A, B and C are orthogonal with block totals and are entirely free from block effects. The interaction ABC is completely confounded with blocks in replicate 1, but in the other three replications the ABC is orthogonal with blocks and consequently an estimate of ABC may be obtained from replicates II, III and IV. Similarly it is possible to recover information on the other confounded interactions AB (from I, III, IV), BC (from I, II, IV) and AC (from I, II, III). Since the partially confounded interactions are estimated from only a portion of the observations, they are determined with a lower degree of precision than the other effects.

For carrying out the statistical analysis, the various factorial effects and their S.S. are estimated in the usual manner with the modification that for **completely confounded** interactions neither the S.S due to confounded interaction is computed nor it is included in the ANOVA table. The confounded component is contained in the (2p-1) degrees of freedom (D.f.) (in case of p replicates) due to blocks. The partitioning of the d.f for a 2^3 completely confounded factorial is as follows.

Source of Variation	**D.f**
Blocks	2p-1
A	1
B	1
C	1
AB	1
AC	1
BC	1
Error	6(p-1)
Total	8p-1

In general for a 2^n completely confounded factorial in p replications, the different d.f's are given as follows

Source of Variation	D.f
Replication	p-1
Blocks within replication	$p(2^{n-r}-1)$
Treatments	$2^n-1-(2^{n-r}-1)$
Error	By subtraction
Total	$p2^n-1$

The treatment d.f has been reduced by $2^{n-r}-1$ as this is the total d.f confounded per block.

In case of partial confounding, we can estimate the effects confounded in one replication from the other replication in which it is not confounded. In $(2^n, 2^r)$ factorial experiment with p replications, following is the splitting of d.f's.

Source of Variation	D.f
Replication	p-1
Blocks within replication	$p(2^{n-r}-1)$
Treatments	2^n-1
Error	By subtraction
Total	$p2^n-1$

The S.S. for confounded effects are to be obtained from those replications only in which the given effect is not confounded.

5. Fractional Factorial

In a factorial experiment, as the number of factors to be tested increases, the complete set of factorial treatments may become too large to be tested simultaneously in a single experiment. A logical alternative is an experimental design that allows testing of only a fraction of the total number of treatments. A design uniquely suited for experiments involving large number of factors is the fractional factorial. It provides a systematic way of selecting and testing only a fraction of the complete set of factorial treatment combinations. In exchange, however, there is loss of information on some pre-selected effects. Although this information loss may be serious in experiments with one or two factors, such a loss becomes more tolerable with large number of factors. The number of interaction effects increases rapidly with the number of factors involved, which allows flexibility in the choice of the particular effects to be sacrificed. In fact, in cases where some specific effects are known beforehand to be small or unimportant, use of the fractional factorial results in minimal loss of information.

In practice, the effects that are most commonly sacrificed by use of the fractional factorial are high order interactions - the four-factor or five-factor interactions and at times, even the three-factor interaction. In almost all cases, unless the researcher has prior information to indicate otherwise one should select a set of treatments to be tested so that all main effects and two-factor interactions can be estimated.

In forestry research, the fractional factorial is to be used in exploratory trials where the main objective is to examine the interactions between factors. For such trials, the most appropriate fractional factorials are those that sacrifice only those interactions that involve more than two factors.

With the fractional factorial, the number of effects that can be measured decreases rapidly with the reduction in the number of treatments to be tested. Thus, when the number of effects to be measured is large, the number of treatments to be tested, even with the use of fractional factorial, may still be too large. In such cases, further reduction in the size of the experiment can be achieved by reducing the number of replications. Although the use of fractional factorial without replication is uncommon in forestry experiments, when fractional factorial is applied to exploratory trials, the number of replications required can be reduced to the minimum.

Another desirable feature of fractional factorial is that it allows reduced block size by not requiring a block to contain all treatments to be tested. In this way, the homogeneity of experimental units within the same block can be improved. A reduction in block size is, however, accompanied by loss of information in addition to that already lost through the reduction in number of treatments.

SPLIT AND STRIP PLOT DESIGNS

Susheel Kumar Sarkar and Anindita Datta
ICAR-IASRI, Library Avenue, New Delhi - 110 012
susheel.sarkar@icar.gov.in; anandita.datta@icar.gov.in

1. Split Plot Design

1.1 Introduction

In conducting experiments, sometimes some factors have to be applied in larger experimental units while some other factors can be applied in comparatively smaller experimental units. Further some experimental materials may be rare while the other experimental materials may be available in large quantity or when the levels of one (or more) treatment factors are easy to change, while the alteration of levels of other treatment factors are costly, or time-consuming. One more point may be that although two or more different factors are to be tested in the experiment, one factor may require to be tested with higher precision than the others. In all such situations, a design called the split plot design is adopted.

A split plot design is a design with at least one blocking factor where the experimental units within each block are assigned to the treatment factor levels as usual, and in addition, the blocks are assigned at random to the levels of a further treatment factor. The designs have a nested blocking structure. In a block design, the experimental units are nested within the blocks, and a separate random assignment of units to treatments is made within each block. In a split plot design, the experimental units are called split-plots (or sub-plots), and are nested within whole plots (or main plots).

In split plot design, plot size and precision of measurement of effects are not the same for both factors, the assignment of a particular factor to either the main plot or the sub-plot is extremely important. To make such a choice, the following guidelines are suggested:

Degree of Precision- For a greater degree of precision for factor B than for factor A, assign factor B to the sub-plot and factor A to the main plot e.g. a plant breeder who plans to evaluate ten promising rice varieties with three levels of fertilization, would probably wish to have greater precision for varietal comparison than for fertilizer response. Thus, he would designate variety as the sub-plot factor and fertilizer as the main plot factor. Or, an agronomist would assign variety to main plot and fertilizer to sub-plot if he wants greater precision for fertilizer response than variety effect.

Relative Size of the Main effects- If the main effect of one factor (A) is expected to be much larger and easier to detect than that of the other factor (B), factor A can be assigned to the main plot and factor B to the sub-plot. This increases the chance of detecting the difference among levels of factor B which has a smaller effect.

Management Practices- The common type of situation when the split plot design is automatically suggestive is the difficulties in the execution of other designs, i.e. practical execution of plans. The cultural practices required by a factor may dictate the use of large

plots. For practical expediency, such a factor may be assigned to the main plot e.g. in an experiment to evaluate water management and variety, it may be desirable to assign water mangement to the main plot to minimize water movement between adjacent plots, facilitate the simulation of the water level required, and reduce border effects. Or, if ploughing is one of the factors of interest, then one cannot have different depths of ploughing in different plots scattered randomly apart.

1.2 Randomization and Layout

There are two separate randomization processes in a split plot design – one for the main plot and another for the sub-plot. In each replication, main plot treatments are first randomly assigned to the main plots followed by a random assignment of the sub-plot treatments within each main plot. This procedure is followed for all replications. A possible layout of a split plot experiment with four main plot treatments (a=4), three sub-plot treatments(b=3), and four replications(r=4) is given below:

Rep. I			
b_1	b_3	b_2	b_2
b_3	b_2	b_1	b_3
b_2	b_1	b_3	b_1
a_4	a_2	a_1	a_3

Rep. II			
b_3	b_1	b_2	b_1
b_1	b_2	b_1	b_3
b_2	b_3	b_3	b_2
a_1	a_4	a_2	a_3

Rep. III			
b_3	b_1	b_2	b_3
b_2	b_3	b_3	b_2
b_1	b_2	b_1	b_1
a_3	a_2	a_4	a_1

Rep. IV			
b_2	b_3	b_3	b_1
b_1	b_2	b_1	b_2
b_3	b_1	b_2	b_3
a_1	a_4	a_3	a_2

The above layout has the following important features – • The size of the main plot is *b* times the size of the sub-plot, • Each main plot treatment is tested *r* times whereas each sub-plot treatment is tested *ar* times, thus the number of times a sub-plot treatment is tested will always be larger than that for the main plot and is the primary reason for more precision for the sub-plot treatments relative to the main plot treatments.

This concept of splitting each plot may be extended further to accommodate the application of additional factors. An extension of this design is called the split-split plot design where the sub-plot is further divided to include a third factor in the experiment. The design allows for 3 different levels of precision associated with the 3 factors. That is, the degree of precision associated with the main factor is lowest, while the degree of precision associated with the sub-sub plot is the highest.

1.3 Model

The model for simple split plot design is

$$Y_{ijk} = \mu + \rho_i + \tau_j + \delta_{ij} + \beta_k + (\tau\beta)_{jk} + \varepsilon_{ijk}$$

for i = 1,2, …,r, j = 1,2, …,a, k = 1,2, …b,

where,

Y_{ijk} : observation corresponding to k^{th} level of sub-plot factor(B), j^{th} level of main plot factor(A) and the ith replication.

μ : general mean
ρ_i : i^{th} block effect
τ_j : j^{th} main plot treatment effect
β_k : k^{th} sub-plot treatment effect
$(\tau\beta)_{jk}$: interaction between j^{th} level of main-plot treatment and the kth level of sub-plot treatment

The error components δ_{ij} and ε_{ijk} are independently and normally distributed with means zero and respective variances σ^2_δ and σ^2_ε.

1.4 Analysis

Whole-Plot analysis:

This part of the analysis is based on comparisons of whole-plot totals:

- The levels of A are assigned to the whole plots within blocks according to a randomized complete block design, and so the sum of squares for A needs no block adjustment. There are *a* –1 degrees of freedom for A, so the sum of squares is given by
  $$ssA = \sum_j y^2_{\cdot j \cdot} / rb - y^2_{\cdots} / rab$$
  [The "dot" notation means "add over all values of the subscript replaced with a dot"]

- There are *r* –1 degrees of freedom for blocks, giving a block sum of squares of
  $$ssR = \sum_i y^2_{i \cdot \cdot} / ab - y^2_{\cdots} / rab$$
- There are *a* whole plots nested within each of the *r* blocks, so there are, in total, *r*(*a* -1) whole-plot degrees of freedom. Of these, *a* –1 are used to measure the effects of A leaving (*r* –1)(*a* –1) degrees of freedom for whole-plot error. Equivalently, this can be obtained by the subtraction of the block and A degrees of freedom from the whole-plot total degrees of freedom i.e. (*ra* –1) –(*r* –1) –(*a* –1) = (*r* –1)(*a* –1).
  So, the whole plot error sum of squares, is obtained as
  $$ssE_1 = \sum_i \sum_j y^2_{i\,j\,\cdot} / b - y^2_{\cdots} / rab - ssR - ssA$$
- The whole plot error mean square $msE_1 = ssE_1 / (r-1)(a-1)$, is used as the error estimate to test the significance of whole plot factor(A).

Sub-plot analysis:

This part of the analysis is based on the observations arising from the split-plots within whole plots:

- There are *rab* –1 total degrees of freedom, and the total sum of squares is
  $$sstot = \sum_i \sum_j \sum_k y^2_{i\,j\,k} - y^2_{\cdots} / rab$$
- Due to the fact that all levels of B are observed in every whole plot as in a randomized complete block design, the sum of squares for B needs no adjustment for whole plots, and is given by -
  $ssB = \sum_k y^2_{\cdot\cdot k} / ra - y^2_{\cdots} / rab$, corresponding to *b* –1 degrees of freedom.

- The interaction between the factors A and B is also calculated as part of the split-plot analysis. Again, due to the complete block structure of both the whole-plot design and the split-plot design, the interaction sum of squares needs no adjustment for blocks. The number of interaction degrees of freedom is $(a-1)(b-1)$, and the sum of squares is
$$ss(AB) = \sum_j \sum_k y_{.jk}^2 / r - y_{...}^2 / rab - ssA - ssB$$

- Since there are b split plots nested within the ra whole plots, there are, in total, $ra(b-1)$ split-plot degrees of freedom. Of these, $b-1$ are used to measure the main effect of B, and $(a-1)(b-1)$ are used to measure the AB interaction, leaving $ra(b-1) - (b-1) - (a-1)(b-1) = a(r-1)(b-1)$ degrees of freedom for error. Equivalently, this can be obtained by subtraction of the whole plot, B, and AB degrees of freedom from the total i.e. $(rab-1) - (ra-1) - (b-1) - (a-1)(b-1) = a(r-1)(b-1)$.
The split-plot error sum of squares can be calculated by subtraction:
$ssE_2 = sstot - ssR - ssA - ssE_1 - ssB - ss(AB)$.

- The split-plot error mean square $msE_2 = ssE_2 / a(r-1)(b-1)$ is used as the error estimate in testing the significance of split-plot factor(B) and interaction(AB).

- The analysis of variance table is outlined as follows:

ANOVA

Source of Variation	Degrees of Freedom	Sum of Squares	Mean Square	F
Whole plot analysis				
Replication	r-1	ssR	-	-
Main plot treatment(A)	a-1	ssA	msA	msA/msE_1
Main plot error(E_1)	(r-1)(a-1)	ssE_1	$msE_1 = E_a$	
Sub-plot analysis				
Sub-plot treatment(B)	b-1	ssB	msB	msB/msE_2
Interaction (AxB)	(a-1)(b-1)	ss(AB)	ms(AB)	$ms(AB)/msE_2$
Sub-plot error(E_2)	a(r-1)(b-1)	ssE_2	$msE_2 = E_b$	
Total	rab-1	sstot		

1.5 Standard Errors and Critical Differences:

Estimate of S.E. of difference between two main plot treatment means = $\sqrt{\frac{2E_a}{rb}}$

Estimate of S.E. of difference between two sub-plot treatment means = $\sqrt{\frac{2E_b}{ra}}$

Estimate of S.E. of difference between two sub-plot treatment means at the same level of main plot treatment = $\sqrt{\frac{2E_b}{r}}$

Estimate of S.E. of difference between two main plot treatment means at the same or different levels of sub-plot treatment $= \sqrt{\frac{2[(b-1)E_b + E_a]}{rb}}$

Critical difference is obtained by multiplying the $S.E_d$ by $t_{5\%}$ table value for respective error d.f. for (i), (ii) & (iii). For (iv), as the standard error of mean difference involves two error terms, we use the following equation to compute the weighted t values:

$$t = \frac{(b-1)E_b t_b + E_a t_a}{(b-1)E_b + E_a}$$

where t_a and t_b are t-values at error d.f. (E_a) and error d.f.(E_b) respectively.

Example 1:

In a study carried by agronomists to determine if major differences in yield response to N fertilization exist among different varieties of jowar, the main plot treatments were three varieties of jowar (V_1: CO-18, V_2: CO-19 and V_3: C0-22), and the sub-plot treatments were N rates of 0, 30, and 60 Kg/ha. The study was replicated four times, and the data gathered for the experiment are shown in Table 1.

Table 1: Replication-wise yield data.

		N rate, Kg/ha		
Replication	Variety	0	30	60
		Yield, kg per plot		
I	V_1	15.5	17.5	20.8
	V_2	20.5	24.5	30.2
	V_3	15.6	18.2	18.5
II	V_1	18.9	20.2	24.5
	V_2	15.0	20.5	18.9
	V_3	16.0	15.8	18.3
III	V_1	12.9	14.5	13.5
	V_2	20.2	18.5	25.4
	V_3	15.9	20.5	22.5
IV	V_1	12.9	13.5	18.5
	V_2	13.5	17.5	14.9
	V_3	12.5	11.9	10.5

Analyze the data and draw conclusions.

Steps of analysis:

- Calculate the replication totals(R), and the grand total(G) by first constructing a table for the replication × variety totals shown in Table 1.1, and then a second table for the variety × nitrogen totals as shown in Table 1.2.

Table 1.1 Replication × variety (RA) - table of yield totals.

	Variety			
Replication	V_1	V_2	V_3	Rep.Total(R)
I	53.8	75.2	52.3	181.3
II	63.6	54.4	50.1	168.1
III	40.9	64.1	58.9	163.9
IV	44.9	45.9	34.9	125.7
Variety Total(A)	203.2	239.6	196.2	
Grand Total(G)				639.0

Table 1.2 Variety × Nitrogen (AB) - table of yield totals.

	Variety			
Nitrogen	V_1	V_2	V_3	Nitrogen Total(B)
N_0	60.2	69.2	60.0	189.4
N_1	65.7	81.0	66.4	213.1
N_2	77.3	89.4	69.8	236.5

- Compute the various sums of squares for the main plot analysis by first computing the correction factor:

  C.F. = $\frac{G^2}{rab} = \frac{(639)^2}{4\times 3\times 3}$ = 11342.25

  Total S.S. (sstot) = $[\,(15.5)^2 + (20.5)^2 + \ \ldots\ + (10.5)^2\,]$ - C.F.

  = 637.97

  Replication S.S. (ssR) = $\frac{\sum R^2}{ab} - \text{C.F.}$

  $= \frac{(181.3)^2 + (168.1)^2 + (163.9)^2 + (125.7)^2}{3\times 3} - 11342.25$

  = 190.08

  S.S. due to Variety (ssA) = $\frac{\sum A^2}{rb} - \text{C.F.}$

  $= \frac{(203.2)^2 + (239.6)^2 + (196.2)^2}{4\times 3} - 11342.25$

  = 90.487

  Main plot error S.S. (ssE_1) = $\frac{\sum (RA)^2}{b} - \text{C.F.} - \text{ssR} - \text{ssA}$

  $= \frac{(53.8)^2 + (63.6)^2 + \ldots + (34.9)^2}{3} - 11342.25 - 190.08 - 90.487$

  = 174.103

- Compute the various sums of squares for sub-plot analysis

S.S. due to Nitrogen (ssB) = $\frac{\sum B^2}{ra} - C.F.$

$= \frac{(189.4)^2 + (213.1)^2 + (236.5)^2}{4 \times 3} - 11342.25$

$= 92.435$

S.S. due to Interaction (A × B) = $\frac{\sum (AB)^2}{r} - C.F. - ssA - ssB$

$= \frac{(60.2)^2 + (65.7)^2 + ... + (69.8)^2}{4} - 11342.25 - 90.487 - 92.435$

$= 9.533$

Sub-plot error S.S. (ssE_2) = Total S.S. – All other sum of squares

= 637.97 – (190.08 + 90.487 + 174.103 + 92.435 +9.533)

= 81.332

- Calculate the mean square for each source of variation by dividing the S.S. by its corresponding degrees of freedom and compute the F value for each effect that needs to be tested, by dividing each mean square by the corresponding error mean square, as shown in Table 1.3.

Table 1.3 ANOVA results.

Source of variation	Degrees of freedom	Sum of Squares	Mean Square	F
Replication	3	190.08	63.360	
Variety(A)	2	90.487	45.243	1.56^{ns}
Error(a)	6	174.103	29.017(E_a)	
Nitrogen(B)	2	92.435	46.218	10.23^{**}
Variety×Nitrogen (A×B)	4	9.533	2.383	<1
Error(b)	18	81.332	4.518 (E_b)	
Total	35	637.97		

^{ns} – not significant, ** - significant at 1% level.

- Compute the coefficient of variation for the main plot and sub-plot as:

cv(a) = $\frac{\sqrt{E_a}}{G.M.} \times 100$, and cv(b) = $\frac{\sqrt{E_b}}{G.M.} \times 100$ respectively.

- Compute standard errors and to make specific comparisons among treatment means compute respective critical differences only when F-tests show significance differences and interpret.

- Conclusion: There was no significant difference among variety means. Yield was significantly affected by nitrogen. However, the interaction between N rate and variety was not significant. All the varieties gave significant response to 30 kg N/ha as well as to 60 kg N/ha.

2. Strip Plot Design

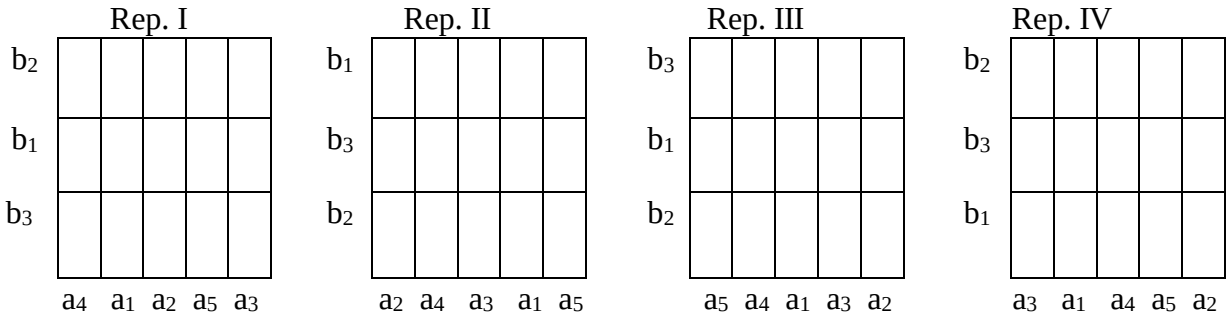
2.1 Introduction

Sometimes situation arises when two factors each requiring larger experimental units are to be tested in the same experiment, e.g., suppose four levels of spacing and three levels of methods of ploughing are to be tested in the same experiment. Here both the factors require large experimental units. If the combinations of the two factors at all possible levels are allotted in a R.B.D. in the normal way, the experimental plots shall have to be very large thereby bringing heterogeneity. So, it will not be appropriate. On the other hand if one factor (spacing) is taken in main plots and other factor (methods of ploughing) is taken in sub-plots within main plots, the sub-plots shall have to be large enough. Hence split plot design also will not be appropriate. In such situations a design called Strip plot design is adopted.

The strip plot is a 2-factor design that allows for greater precision in the measurement of the interaction effect while sacrificing the degree of precision on the main effects. The experimental area is divided into three plots, namely the vertical-strip plot, the horizontal-strip plot, and the intersection plot. We allocate factors A and B, respectively, to the vertical and horizontal-strip plots, and allow the intersection plot to accommodate the interaction between these two factors. As in the split plot design, the vertical and the horizontal plots are perpendicular to each other. However, in the strip plot design the relationship between the vertical and horizontal plot sizes is not as distinct as the main and sub-plots were in the split plot design. The sub-plot treatments instead of being randomized independently within each main plot as in the case of split plot design are arranged in strips across each replication. The intersection plot, which is one of the characteristics of the design, is the smallest in size.

2.2 Randomization and Layout:

In this design each block is divided into number of vertical and horizontal strips depending on the levels of the respective factors. Let A represent the vertical factor with *a* levels, B represent the horizontal factor with *b* levels and *r* represent the number of replications. To layout the experiment, the experimental area is divided into *r* blocks. Each block is divided into *b* horizontal strips and *b* treatments are randomly assigned to these strips in each of the *r* blocks separately and independently. Then each block is divided into *a* vertical strips and *a* treatments are randomly assigned to these strips in each of the *r* blocks separately and independently. A possible layout of a strip plot experiment with $a = 5$ (a_1, a_2, a_3, a_4, and a_5), $b = 3$ (b_1, b_2, and b_3) and four replications is given below:



The strip plot design sacrifices precision on the main effects of both the factors in order to provide higher precision on the interaction which will generally be more accurately determined than in either randomised blocks or simple split plot design. Consequently this design is not recommended unless practical considerations necessitate its use or unless the interaction is the principle object of study.

2.3 Model

The model for strip plot design is

$$Y_{ijk} = \mu + \rho_i + \alpha_j + (\rho\alpha)_{ij} + \beta_k + (\rho\beta)_{ik} + (\alpha\beta)_{jk} + \varepsilon_{ijk}$$

for i = 1,2, …,r, j = 1,2, …,a, k = 1,2, …b,

where,

Y_{ijk} : observation corresponding to j^{th} level of factor A, k^{th} level of factor B and i^{th} replication

μ : general mean

ρ_i : i^{th} block effect

α_j : effect of jth level of factor A

β_k : effect of kth level of factor B

$(\alpha\beta)_{jk}$: interaction between j^{th} level of factor A and the kth level of factor B

The error components $(\rho\alpha)_{ij}$, $(\rho\beta)_{ik}$ and ε_{ijk} are independently and normally distributed with means zero and respective variances σ^2_a, σ^2_b, and σ^2_ε.

2.4 Analysis

In statistical analysis separate estimates of error are obtained for main effects of the factor, A and B and for their interaction AB. Thus there will be three error mean squares applicable for testing the significance of main effects of the factors and their interaction separately.

Suppose 4 levels of spacings(A) and 3 levels of methods(B) of ploughing are to be tested in the same experiment. Each replication is divided into 4 strips vertically and into 3 strips horizontally. In the vertical strips the four different levels of spacings are allotted randomly and in the horizontal strips three methods of ploughing are allotted randomly. Let there be 4 replications(R). The analysis of variance is carried out in three parts viz. vertical strip analysis, horizontal strip analysis and interaction analysis as follows:

- Form spacing × replication (A × R) table of yield totals and from this table compute the S.S. due to replication, S.S. due to spacings and S.S. due to interaction - Replication × Spacing i.e. error(a).
- Form method × replication (B × R) table of yield totals and from this table compute the S.S. due to methods and S.S. due to interaction - Replication × Method i.e. error(b).
- Form spacing × method (A × B) table of yield totals and from this table compute the S.S. due to interaction - Spacing × Method.

- Total S.S. will be obtained as usual by considering all the observations of the experiment and the error S.S. i.e. error(c) will be obtained by subtracting from total S.S. all the S.S. for various sources.

- Now, calculate the mean square for each source of variation by dividing each sum of squares by its respective degrees of freedom.

- Compute the F-value for each source of variation by dividing each mean square by the corresponding error term.

- The analysis of variance table is outlined as follows:

ANOVA

Source of Variation	**Degrees of Freedom**	**Sum of Squares**	**Mean Square**	**F**
Replication(R)	(r-1)= 3	ssR	-	-
Spacing(A)	(a-1)= 3	ssA	msA	msA/msE$_1$
Error(a)	(r-1)(a-1)= 9	ssE$_1$	msE$_1$ =E$_a$	
Method(B)	(b-1)= 2	ssB	msB	msB/msE$_2$
Error(b)	(r-1)(b-1)= 6	ssE$_2$	msE$_2$ =E$_b$	
Spacing×Method (A×B)	(a-1)(b-1)= 6	ss(AB)	ms(AB)	ms(AB)/msE$_3$
Error(c)	(r-1)(a-1)(b-1)=18	ssE$_3$	msE$_3$ =E$_c$	
Total	(rab-1)= 47	sstot		

2.5 Standard Errors and Critical Differences:

Estimate of S.E. of difference between two A level means = $\sqrt{\frac{2E_a}{rb}}$

Estimate of S.E. of difference between two B level means = $\sqrt{\frac{2E_b}{ra}}$

Estimate of S.E. of difference between two A level means at the same level of B means = $\sqrt{\frac{2[(b-1)E_c + E_a]}{rb}}$

Estimate of S.E. of difference between two B level means at the same level of A means = $\sqrt{\frac{2[(a-1)E_c + E_b]}{ra}}$

Critical difference is obtained by multiplying the S.E$_d$ by $t_{5\%}$ table value for respective error d.f. for (i) & (ii). For (iii) & (iv), as the standard error of mean difference involves two error terms, we use the following equation to compute the weighted t values:

$$t = \frac{(b-1)E_c t_c + E_a t_a}{(b-1)E_c + E_a} \text{ , and } t = \frac{(a-1)E_c t_c + E_b t_b}{(a-1)E_c + E_b} \text{ respectively,}$$

where t_a, t_b, and t_c are t-values at error d.f. (E_a), error d.f.(E_b) and error d.f.(E_c) respectively.

SAS input statements for the split plot experiment

```
data split plot;
input rep var nit yield;
cards;
 1  1  0  15.5
 1  1  1  17.5
 1  1  2  20.8
 1  2  0  20.5
 1  2  1  24.5
 1  2  2  30.2
 1  3  0  15.6
 1  3  1  18.2
 1  3  2  18.5
 2  1  0  18.9
 2  1  1  20.2
 2  1  2  24.5
 2  2  0  15.0
 2  2  1  20.5
 2  2  2  18.9
 2  3  0  16.0
 2  3  1  15.8
 2  3  2  18.3
 3  1  0  12.9
 3  1  1  14.5
 3  1  2  13.5
 3  2  0  20.2
 3  2  1  18.5
 3  2  2  25.4
 3  3  0  15.9
 3  3  1  20.5
 3  3  2  22.5
 4  1  0  12.9
 4  1  1  13.5
 4  1  2  18.5
 4  2  0  13.5
 4  2  1  17.5
 4  2  2  14.9
 4  3  0  12.5
 4  3  1  11.9
 4  3  2  10.5
;
proc print;
proc glm;
class rep var nit;
```

```
model yield = rep var rep* var nit var*nit;
test h = var e = rep*var;
mean var nit var*nit;
run;
```

RESPONSE SURFACE DESIGNS

Eldho Varghese
ICAR-CMFRI, Kochi, Kerala
eldho.varghese@icar.gov.in

1. Introduction

The subject of Design of Experiments deals with the statistical methodology needed for making inferences about the treatment effects on the basis of responses (univariate or multivariate) collected through the planned experiments. To deal with the evolution and analysis of methods for probing into mechanism of a system of variables, the experiments involving several factors simultaneously are being conducted in agricultural, horticultural and allied sciences. Data from experiments with levels or level combinations of one or more factors as treatments are normally investigated to compare level effects of the factors and also their interactions. Though such investigations are useful to have objective assessment of the effects of levels actually tried in the experiment, this seems to have inadequate, especially when the factors are quantitative in nature. The above analysis cannot give any information regarding the possible effects of the intervening levels of the factors or their combinations, *i.e.*, one is not able to interpolate the responses at the treatment combinations not tried in the experiment. In such cases, it is more realistic and informative to carry out investigations with the twin purposes:

a) To determine and to quantify the relationship between the response and the settings of a group of experimental factors.
b) To find the settings of the experimental factors that produces the best value or the best set of values of the response(s).

If all the factors are quantitative in nature, it is natural to think the response as a function of the factor levels and data from quantitative factorial experiments can be used to fit the response surfaces over the region of interest. Response surfaces besides inferring about the twin purposes can provide information about the rate of change of a response variable. They can also indicate the interactions between the quantitative treatment factors. The special class of designed experiments for fitting response surfaces is called response surface designs. A good response surface design should possess the properties *viz.*, detectability of lack of fit, the ability to sequentially build up designs of increasing order and the use of a relatively modest, if not minimum, number of design points. Before formulating the problem mathematically, we shall give examples of some experimental situations, where response surface methodology can be usefully employed.

Example 1: The over-use of nitrogen (N) relative to Phosphorus (P) and Potassium (K) concerns both the agronomic and environmental perspective. Phosphatic and Potassic fertilizers have been in short supply and farmers have been more steadily adopting the use of nitrogenous fertilizers because of the impressive virtual response. There is evidence that soil P and K levels are declining. The technique of obtaining individual optimum doses for the N, P and K through separate response curves may also be responsible for unbalanced fertilizer use. Hence, determining the optimum and balanced dose of N, P and K for different crops has

been an important issue. This optimum and balanced dose should be recommended to farmers in terms of doses from the different sources and not in terms of the values of N, P and K alone, as the optimum combination may vary from source to source. However, in actual practice the values of N, P and K are given in terms of kg/ha rather than the combined doses alongwith the source of the fertilizers.

Example 2: For value addition to the agriculture produce, food-processing experiments are being conducted. In these experiments, the major objective of the experimenter is to obtain the optimum combination of levels of several factors that are required for the product. To be specific, suppose that an experiment related to osmotic dehydration of the banana slices is to be conducted to obtain the optimum combination of levels of concentration of sugar solution, solution to sample ratio and temperature of osmosis. The levels of the various factors are the following

	Factors	**Levels**
1.	Concentration of sugar solution	40%, 50%, 60%, 70% and 80%
2.	Solution to sample ratio	1:1, 3:1, 5:1, 7:1and 9:1
3.	Temperature of osmosis	25^0C, 35^0C, 45^0C, 55^0C and 65^0C

In this situation, response surface designs for 3 factors each at five equispaced levels can be used.

Example 3: Yardsticks (a measure of the average increase in production per unit input of a given improvement measure) of many fertilizers, manures, irrigation, pesticides for various crops are being obtained and used by planners and administrators in the formulation of policies relating to manufacture/import/subsidy of fertilizers, pesticides, development of irrigation projects etc.

The yardsticks have been obtained from the various factorial experiments. However, these will be more reliable and satisfy more statistical properties, if response surface designs for slope estimation are used.

In general response surface methodology is useful for all the factorial experiments in agricultural experimental programme that are under taken so as to determine the level at which each of these factors must be set in order to optimize the response in some sense and factors are quantitative in nature. To achieve this we postulate that the response is a function of input variables, *i.e.*

$$y_u = \varphi(x_{1u}, x_{2u}, ..., x_{vu}) + e_u \qquad (1.1)$$

where $u = 1,2,..., N$ represents the N observations and x_{iu} is the level of the ith factor in the uth observation. The function φ describes the form in which the response and the input variables are related and e_u is the experimental error associated with the uth observation such that E (e_u) = 0 and Var(e_u) = σ^2. Knowledge of function φ gives a complete summary of the results of the experiment and also enables us to predict the response for values of the x_{iu} that are not included in the experiment. If the function φ is known then using methods of calculus, one may obtain the values of $x_1, x_2, ..., x_v$ which give the optimum (say, maximum) response.

In practice the mathematical form of φ is not known; we, therefore, often approximate it, within the experimental region, by a polynomial of suitable degree in variables x_{iu}. The adequacy of the fitted polynomial is tested through the usual analysis of variance. Polynomials which adequately represent the true dose-response relationship are called **Response Surfaces** and the designs that allow the fitting of response surfaces and provide a measure for testing their adequacy are called **response surface designs**. If the function φ in (1.1) is of degree one in $x_{iu}'s$ *i.e.*

$$y_u = \beta_0 + \beta_1 x_{1u} + \beta_2 x_{2u} + ... + \beta_v x_{vu} + e_u \tag{1.2}$$

we call it a first-order response surface in $x_1, x_2, ..., x_v$. If (1.1) takes the form

$$y_u = \beta_0 + \sum_{i=1}^{v} \beta_i x_{iu} + \sum_{i=1}^{v} \beta_{ii} x_{iu}^2 + \sum_{i=1}^{v-1} \sum_{i'=i+1}^{v} \beta_{ii'} x_{iu} x_{i'u} + e_u \tag{1.3}$$

We call it a second-order (quadratic) response surface. Henceforth, we shall concentrate on the second order response surface which is more useful in agricultural experiments.

2. The Quadratic Response Surface

The general form of a second-degree (quadratic) surface is

$$y_u = \beta_0 + \beta_1 x_{1u} + \beta_2 x_{2u} + ... + \beta_v x_{vu} + \beta_{11} x_{1u}^2 + \beta_{22} x_{2u}^2 + ... + \beta_{vv} x_{vu}^2 + \beta_{12} x_{1u} x_{2u} + \beta_{13} x_{1u} x_{3u} + ... + \beta_{v-1,v} x_{v-1,u} x_{vu} + e_u$$

Let us assume that $x_{iu}'s$ satisfy the following conditions:

(A) $\sum_{u=1}^{N} \left\{ \prod_{u=1}^{v} x_{iu}^{\alpha_i} \right\} = 0$, if any α_i is odd, for $\alpha_i = 0,1,2$ *or* 3 and $\sum \alpha_i \le 4$.

(B) $\sum_{u=1}^{N} x_{iu}^2 = \text{constant (for all } i) = N\lambda_2$ (say)

(C) $\sum_{u=1}^{N} x_{iu}^4 = \text{constant (for all } i) = CN\lambda_4$ (say) (2.1)

(D) $\sum_{u=1}^{N} x_{iu}^2 x_{i'u}^2 = \text{constant} = N\lambda_4$ (say), for all $i \neq i'$

We shall estimate the parameters $\beta_i's$ through the method of least squares. Let $b_0, b_i's, b_{ii}'s, b_{ii'}'s$ denote the best linear unbiased estimate of $\beta_0, \beta_i's, \beta_{ii}'s, \beta_{ii'}'s$ respectively. Under the above restrictions on $x_{iu}'s$, the normal equations are found to be:

$$\sum_{u=1}^{N} y_u = Nb_0 + N\lambda_2 \sum_{i=1}^{v} b_{ii}$$

$$\sum_{u=1}^{N} x_{iu} y_u = N\lambda_2 b_i$$

$$\sum_{u=1}^{N} x_{iu} x_{i'u} y_u = N\lambda_4 b_{ii'} \qquad (2.2)$$

$$\sum_{u=1}^{N} x_{iu}^2 y_u = N\lambda_2 b_0 + CN\lambda_4 b_{ii} + N\lambda_4 \sum_{i' \neq i} b_{i'i'}$$

$$= N\lambda_2 b_0 + (C-1)N\lambda_4 b_{ii} + N\lambda_4 \sum_{i=1}^{v} b_{ii}$$

Solving the above normal equations, we obtain the estimates b_i's as

$$b_0 = \left[\lambda_4 (C + v - 1) \sum_{u=1}^{N} y_u - \lambda_2 \sum_{i=1}^{v} \sum_{u=1}^{N} x_{iu}^2 y_u \right] \Big/ N\Delta$$

$$b_i = \sum_{u=1}^{N} x_{iu} y_u \Big/ N\lambda_2 \qquad (2.3)$$

$$b_{ii'} = \sum_{u=1}^{N} x_{iu} x_{i'u} y_u \Big/ N\lambda_4$$

$$b_{ii} = \left[\sum_{u=1}^{N} x_{iu}^2 y_u - \left\{ (\lambda_2^2 - \lambda_4) \sum_{i=1}^{v} \sum_{u=1}^{N} x_{iu}^2 y_u - (C-1)\lambda_2 \lambda_4 \sum_{u=1}^{N} y_u \right\} \Big/ \Delta \right] \Big/ [(C-1)N\lambda_4]$$

where $\Delta = (C + v - 1)\lambda_4 - v\lambda_2^2$.

The variances of and covariances between the estimated parameters are as follows:

$$V(b_0) = \lambda_4 (C + v - 1)\sigma^2 / N\Delta$$

$$V(b_i) = \sigma^2 / N\lambda_2$$

$$V(b_{ii'}) = \sigma^2 / N\lambda_4 \qquad (2.4)$$

$$V(b_{ii}) = \sigma^2 \left[1 + (\lambda_2^2 - \lambda_4) \right] / [(C-1)N\lambda_4]$$

$$Cov(b_0, b_{ii}) = -\lambda_2 \sigma^2 / N\Delta$$

$$Cov(b_{ii}, b_{i'i'}) = (\lambda_2^2 - \lambda_4)\sigma^2 / [(C-1)N\lambda_4 \Delta]$$

Other covariances are zero. From the above expressions it is clear that a necessary condition for the design to exist is that $\Delta > 0$. Thus, a necessary condition for a Second Order Design to exist is that

(E) $\lambda_4 / \lambda_2^2 > v/(C + v - 1)$ (2.5)

If $\hat{y}$ is the estimated response at any given experimental point $(x_{10}, x_{20},...,x_{v0})$, then the variance of $\hat{y}$ is given by

$$V(\hat{y}) = V(b_0) + V(b_i)\left(\sum_{i=1}^{v} x_{i0}^2\right) + V(b_{ii})\left(\sum_{i=1}^{v} x_{i0}^4\right) + V(b_{ii'})\left(\sum_{i=1}^{v-1}\sum_{i'=i+1}^{v} x_{i0}^2 x_{i'0}^2\right) + 2\,Cov(b_0, b_{ii})\left(\sum_{i=1}^{v} x_{i0}^2\right) + 2\,Cov(b_{ii}, b_{i'i'})\left(\sum_{i=1}^{v-1}\sum_{i'=1+1}^{v} x_{i0}^2 x_{i'0}^2\right) \tag{2.6}$$

If $\sum_{i=1}^{v} x_{i0}^2 = d^2$, where d is the distance of the point $(x_{10}, x_{20},...,x_{v0})$ from the origin, then we may write

$$V(\hat{y}) = V(b_0) + d^2[V(b_i) + 2\,Cov(b_0, b_{ii})] + d^4 V(b_{ii}) + \sum_{i=1}^{v-1}\sum_{i'=i+1}^{v} x_{i0}^2 x_{i'0}^2 [V(b_{ii'}) + 2\,Cov(b_{ii}, b_{i'i'}) - 2V(b_{ii})] \tag{2.7}$$

From the above expression, it is clear that if the coefficient of $\sum_{i=1}^{v-1}\sum_{i'=i+1}^{v} x_{i0}^2 x_{i'0}^2$ is made equal to zero, the variance of the estimated response at $(x_{10}, x_{20},...,x_{v0})$ will be a function of d, the distance of the point $(x_{10}, x_{20},...,x_{v0})$ from the origin. Now, the coefficient of $\sum_{i=1}^{v-1}\sum_{i'=i+1}^{v} x_{i0}^2 x_{i'0}^2$ is

$$\begin{aligned} & V(b_{ii'}) + 2\ Cov(b_{ii}, b_{i'i'}) - 2V(b_{ii}) \\ &= \frac{\sigma^2}{N\lambda_4}\left[1 + \frac{2\left(\lambda_2^2 - \lambda_4\right)}{\Delta(C-1)} - \frac{2}{(C-1)}\left\{1 + \frac{\lambda_2^2 - \lambda_4}{\Delta}\right\}\right] \\ &= \frac{\sigma^2}{N\lambda_4}\left[1 - \frac{2}{(C-1)}\right] \end{aligned} \tag{2.8}$$

Obviously, this is zero, if and only if $C = 3$. Thus, when $C = 3$, the variance of the estimated response at a given point, the response being estimated through a design satisfying (A), (B), (C), (D), (E) becomes a function of the distance of that point from the origin. Such designs are called as Second Order Rotatable Designs (SORD). We may now formally define a SORD:

Let us consider N treatment combinations (points) $\{x_{iu}\}$, $i = 1,2,...,v, u = 1,2,...,N$ to form a design in v factors, through which a Second-degree surface can be fitted. This design is said to be a SORD if the variance of the estimated response at any given point is a function of the distance of that point from the origin. The necessary and sufficient conditions for a set of points $\{x_{iu}\}$, $i = 1,2,...,v, u = 1,2,...,N$ to form a SORD are

(A') $\sum_{u=1}^{N}\left\{\prod_{u=1}^{v} x_{iu}^{\alpha_i}\right\} = 0$, if any α_i is odd, for $\alpha_i = 0,1,2$ *or* 3 and $\sum \alpha_i \le 4.$

(B') $\sum_{u=1}^{N} x_{iu}^2 = N\lambda_2$

(C') $\sum_{u} x_{iu}^4 =$ constant $= 3N\lambda_4$ $\qquad i = 1,2,...,v$

(D') $\sum_{u} x_{iu}^2 x_{i'u}^2 = N\lambda_4$; $\qquad i \ne i'$

(E') $\lambda_4 / \lambda_2^2 > v/(v+2)$ (2.9)

The conditions (A'), (B') and (D') are same as conditions (A), (B) and (D) in (2.1).

We now prove the following.

Lemma: If a set of points $\{x_{iu},\ i = 1,2,...,v, u = 1,2,...,N\}$, satisfying (A'), (B'), (C') and (D') are such that every point is equidistant from the origin, then

$$\lambda_4 / \lambda_2^2 = v/(v+2) \tag{2.10}$$

Proof: Let d be the distance of any point from the origin. Then, since all the points are equidistant from the origin, we have

$$d^2 = \frac{1}{N}\sum_{u=1}^{N}\left(\sum_{i=1}^{v} x_{iu}^2\right) = v\lambda_2$$

$$d^4 = \frac{1}{N}\sum_{u=1}^{N}\left(\sum_{i=1}^{v} x_{iu}^2\right)^2$$

and $$= \frac{1}{N}\sum_{u=1}^{N}\left[\sum_{i=1}^{v} x_{iu}^4 + 2\sum_{i=1}^{v}\sum_{i'=i+1}^{v} x_{iu}^2 x_{i'u}^2\right]$$

$$= 3v\lambda_4 + v(v-1)\lambda_4$$

Thus, $v^2\lambda_2^2 = 3v\lambda_4 + v(v-1)\lambda_4$

or, $\lambda_4(v+2) - v\lambda_2^2 = 0$

An arrangement of points satisfying (A'), (B'), (C') and (D') but not (E') is called a Second Order Rotatable Arrangement (SORA). A SORA can always be converted to an SORD by adding at least one central point.

A near stationary region is defined as a region where the surface slopes along the v variable axes are small compared to the estimate of experimental error. The stationary point of a near

stationary region is the point at which the slope of the response surface is zero when taken in all the directions. The coordinates of the stationary point $\mathbf{x_0} = (x_{10}, x_{20}, ..., x_{v0})'$ are obtained by differentiating the following estimated response equation with respect to each x_i and equating the derivatives to zero and solving the resulting equations

$$\hat{Y}(x) = b_0 + \sum_{i=1}^{v} b_i x_i + \sum_{i=1}^{v} b_{ii} x_i^2 + \sum_{i=1}^{v-1} \sum_{i'=i+1}^{v} b_{ii'} x_i x_{i'} \qquad (2.11)$$

In matrix notation (2.11) can be written as

$$\hat{Y}(x) = b_0 + \mathbf{x'b} + \mathbf{x'Bx} \qquad (2.12)$$

where $\mathbf{x} = (x_1, x_2, ..., x_v)'$, $\mathbf{b} = (b_1, b_2, ..., b_v)'$

and

$$\mathbf{B} = \begin{bmatrix} b_{11} & b_{12}/2 & \dots & b_{1v}/2 \\ b_{12}/2 & b_{22} & \dots & b_{2v}/2 \\ \dots & \dots & \dots & \dots \\ b_{1v}/2 & b_{2v}/2 & \dots & b_{vv} \end{bmatrix}.$$

From equation (2.12)

$$\frac{\partial \hat{Y}(x)}{\partial x} = \mathbf{b} + 2\mathbf{Bx} \qquad (2.13)$$

The stationary point $\mathbf{x_0}$ is obtained by equating (2.13) to zero and solving for $\mathbf{x}$, *i.e.*

$$\mathbf{x_0} = -\frac{1}{2}\mathbf{B}^{-1}\mathbf{b} \qquad (2.14)$$

To find the nature of the surface at the stationary point we examine the second derivative of $\hat{Y}(x)$. From (2.13)

$$\frac{\partial^2 \hat{Y}(x)}{\partial x^2} = 2\mathbf{B} \qquad \text{(since } \mathbf{B} \text{ is symmetric).}$$

The stationary point is a maximum, minimum or a saddle point according as **B** is negative definite, positive definite or indefinite matrix. If $\lambda_1, \lambda_2, ..., \lambda_v$ represent the *v* eigenvalues of **B**. Then it is easy to see that if $\lambda_1, \lambda_2, ..., \lambda_v$ are

(i) All negative, then at $\mathbf{x}_0$ the surface is a maximum

(ii) All positive, then at $\mathbf{x}_0$ the surface is a minimum

(iii) of mixed signs, i.e. some are positive and others are negative, then x_0 is a saddle point of the fitted surface.

Furthermore, if λ_i is zero (or very close to zero), then the response does not change in value in the direction of the axis associated with x_i variable. The magnitude of λ_i indicates how quickly the response changes in the direction of axis associated with x_i variable.

The conditions in (2.1) and (2.9) help in fitting of the response surfaces and define some statistical properties of the design like rotatability. However, these conditions need not necessary be satisfied before fitting a response surface. This can be achieved by using the software packages like the Statistical Analysis System (SAS). PROC RSREG fits a second order response surface design and locates the coordinates of the stationary point, predict the response at the stationary point and give the eigenvalues $\lambda_1, \lambda_2, ..., \lambda_v$ and the corresponding eigen vectors. It also helps in determining whether the stationery point is a point of maxima, minima or is a saddle point. The lack of fit of a second order response surface can also be tested using LACKFIT option under model statement in PROC RSREG. The lack of fit is tested using the statistic

$$F = \frac{SS_{LOF}/(N'-p)}{SS_{PE}/(N-N')} \tag{2.15}$$

where N is the total number of observations, N' is the number of distinct treatments and p is the number of terms included in the model. SS_{PE} (sum of squares due to pure error) has been calculated in the following manner: denote the lth observation at the uth design point by y_{lu}, where $l = 1, \ldots, r_u\ (\geq 1),\ u = 1, \ldots, N'$. Define $\bar{y}_u$ to be average of r_u observations at the uth design point. Then, the sum of squares for pure error is

$$SS_{PE} = \sum_{u=1}^{N'} \sum_{l=1}^{r_u} (y_{lu} - \bar{y}_u)^2 \tag{2.16}$$

Then sum of squares due to lack of fit (SS_{LOF}) = sum of squares due to error - SS_{PE}

The analysis of variance table for a second order response surface design is given below.

Table 1. Analysis of variance for second order response surface

Source	d.f.	S.S.
Due to regression coefficients	$2v + \binom{v}{2}$	$\hat{b}_0 \sum_{u=1}^{N} y_u + \sum_i \hat{b}_i \left(\sum_{u=1}^{N} x_{iu} y_u \right) + \sum_i \hat{b}_{ii} \left(\sum_{u=1}^{N} x_{iu}^2 y_u \right) + \sum_{i \neq i'} \sum \hat{b}_{ii'} \left(\sum_{u=1}^{N} x_{iu} x_{i'u} y_u \right) - CF$
Error	$N - 2v - \binom{v}{2} - 1$	*By subtraction = SSE*
Total	$N-1$	$\sum_{u=1}^{N} y_u^2 - CF$

In the above table CF = correction factor = $\frac{(Grand\ Total)^2}{N}$. For testing the lack of fit the sum of squares is obtained using (2.16) and then sum of squares is obtained by subtracting the sum of squares due to pure error from sum of squares due to error. The sum of squares due to

lack of fit and sum of squares due to pure error are based on $N'-2v-\binom{v}{2}-1$ and $N-N'$ degrees of freedom respectively.

It is suggested that in the experiments conducted to find a optimum combination of levels of several quantitative input factors, at least one level of each of the factors should be higher than the expected optimum. It is also suggested that the optimum combination should be determined from response surface fitting rather than response curve fitting, if the experiment involves two or more than two factors.

3. Construction of Second Order Rotatable designs

A second order response surface design is at least resolution V fractional factorial design. Here

3.1 Central Composite Rotatable Designs

Let there be *v* factors in the design. A class of SORD for *v* factors can be constructed in the following manner. Construct a factorial *v*-factors with levels $\pm\alpha$ containing 2^p combinations, where 2^p is the smallest fraction of 2^v without confounding any interaction of third order or less. Next, another $2v$ points of the following type are considered: $(\pm\beta\ 0\ 0\ \ldots\ 0), (0\ \pm\beta\ 0\ \ldots\ 0), (0\ 0\ \ldots\ \pm\beta)$. These $N=2^p+2v$ points, give rise to a SORD in *v* factors with levels $\pm\alpha, \pm\beta, 0$. We have for this design,

$$\sum_{u=1}^{N} x_{iu}^2 = 2^p\alpha^2 + 2\beta^2$$

$$\sum_{u=1}^{N} x_{iu}^4 = 2^p\alpha^4 + 2\beta^4$$

$$\sum_{u=1}^{N} x_{i0}^2 x_{i'0}^2 = 2^p\alpha^4.$$

On applying the condition of rotatability, we have

$$3.2^p\alpha^4 = 2^p\alpha^4 + 2\beta^4$$

$$\Rightarrow \beta^4 = \alpha^4 2^p$$

$$or\ \ \beta^2/\alpha^2 = 2^{p/2}.$$

This equation gives a relationship between β and α. For determining α and β uniquely, we either fix $\alpha=1$ or $\lambda_2=1$. For $\alpha=1, \Rightarrow \beta^2=2^{p/2}$.

Example. Let $v = 4$. Then the points of the SORD are

$$
\begin{array}{rrrr}
-\alpha & -\alpha & -\alpha & -\alpha \\
-\alpha & -\alpha & -\alpha & \alpha \\
-\alpha & -\alpha & \alpha & -\alpha \\
-\alpha & -\alpha & \alpha & \alpha \\
-\alpha & \alpha & -\alpha & -\alpha \\
-\alpha & \alpha & -\alpha & \alpha \\
-\alpha & \alpha & \alpha & -\alpha \\
-\alpha & \alpha & \alpha & \alpha \\
\alpha & -\alpha & -\alpha & -\alpha \\
\alpha & -\alpha & -\alpha & \alpha \\
\alpha & -\alpha & \alpha & -\alpha \\
\alpha & -\alpha & \alpha & \alpha \\
\alpha & \alpha & -\alpha & -\alpha \\
\alpha & \alpha & -\alpha & \alpha \\
\alpha & \alpha & \alpha & -\alpha \\
\alpha & \alpha & \alpha & \alpha \\
\beta & 0 & 0 & 0 \\
-\beta & 0 & 0 & 0 \\
0 & \beta & 0 & 0 \\
0 & -\beta & 0 & 0 \\
0 & 0 & \beta & 0 \\
0 & 0 & -\beta & 0 \\
0 & 0 & 0 & \beta \\
0 & 0 & 0 & -\beta \\
0 & 0 & 0 & 0
\end{array}
$$

There are *25* points – a central point has been added because, all the non-central points are equidistant from the origin, as $\beta = 2\alpha$, here.

3.2 Construction of SORD using BIB Designs

If there exists a BIB design D with parameters v^*, b^*, r^*, k^*, λ^* such that $r^* = 3\lambda^*$, then a SORD with each factor at *3* levels can be constructed.

Let $\mathbf{N}^*$ be the $v^* \times b^*$ incidence matrix of D. Then $\mathbf{N}^{*\prime}$ is a matrix of order $b^* \times v^*$, every row of which contains exactly k^* unities and every column contains exactly r^* unities, rest positions being filled up by zeros. In $\mathbf{N}^{*\prime}$, replace the unity by α. Then, we get b^* combinations involving α and zero. Next, each of these combinations are 'multiplied' with those of a 2^{k^*} factorial with levels ± 1 where, the term 'multiplication' means the multiplication of the corresponding entries in the two combinations, zero entries remaining unaltered. Thus, if $(\alpha\ \alpha\ 0)$ is multiplied by $(-1\ -1)$ we get $(-\alpha\ \ -\alpha\ \ 0)$. The procedure of multiplication gives rise to $b^*\ 2^{k^*}$ points each of v^*-dimension. These points evidently satisfy all the conditions (A'), (B'), (C') and (D'); however, since each point in the

arrangement is at the same distance from the origin, we have to take at least one central point to get a SORD in $v = v^*$ factors. The levels of the factors are $\pm\alpha, 0$. The value of α can be determined by fixing $\lambda_2 = 1$.

SORD's can be constructed using BIB designs, even when $r^* \neq 3\lambda^*$. In the case, where $r^* < 3\lambda^*$ the set of $b^* 2^{k^*}$ points obtained using $\mathbf{N}^{*\prime}$ is to be augmented with further $2v^*$ points of the type

$$(\pm\beta\ 0\ 0 \ldots 0), (0 \pm\beta\ 0 \ldots 0), (0\ 0 \ldots \pm\beta)$$

For the N points ($N = b^* 2^{k^*} + 2v^*$), we have

$$\sum_u x_{iu}^4 = r^* 2^{k^*} \alpha^4 + 2\beta^4$$

$$\sum_u x_{iu}^2 x_{i'u}^2 = \lambda^* 2^{k^*} \alpha^4.$$

Thus $2\beta^4 + r^* 2^{k^*} \alpha^4 = 3\lambda^* 2^{k^*} \alpha^4$

or, $\beta^2 / \alpha^2 = (3\lambda^* - r^*)^{1/2} . 2^{\frac{k^*-1}{2}}$

When $r^* > 3\lambda^*$, the points augmented are of type $(\pm\beta \pm\beta \ldots\ldots \pm\beta)$ and 2^p in number, where 2^p is the smallest fraction of 2^{v^*} factorial with levels $\pm\beta$, such that no interaction of order three or less is confounded. In this case,

$$\sum_u x_{iu}^4 = r^* 2^{k^*} \alpha^4 + 2^p \beta^4$$

$$\sum_u x_{iu}^2 x_{i'u}^2 = \lambda^* 2^{k^*} \alpha^4 + 2^p \beta^4.$$

Thus, $3\lambda^* 2^{k^*} \alpha^4 + 3.2^p \beta^4 = r^* 2^{k^*} \alpha^4 + 2^p \beta^4$

or, $2^{p+1} \beta^4 = (r^* - 3\lambda^*) 2^{k^*} \alpha^4$,

which gives $\beta^2 / \alpha^2 = (r^* - 3\lambda^*)^{1/2} . 2^{(k^* - p - 1)/2}$.

In both the cases, we get v^*-factor SORD with each factor at five levels

4. Practical Exercise

Exercise 1: Consider an experiment that was conducted to investigate the effects of three fertilizer ingredients on the yield of a crop under fields conditions using a second order rotatable design. The fertilizer ingredients and actual amount applied were nitrogen (N), from 0.89 to 2.83 kg/plot; phosphoric acid (P_2O_5) from 0.265 to 1.336 kg/plot; and potash (K_2O), from 0.27 to 1.89 kg/plot. The response of interest is the average yield in kg per plot. The levels of nitrogen, phosphoric acid and potash are coded, and the coded variables are defined as

X_1=(N-1.629)/0.716, X_2=(P_2O_5-0.796)/0.311, X_3=(K_2O -1.089)/0.482

The values 1.629, 0.796 and 1.089 kg/plot represent the centres of the values for nitrogen, phosphoric acid and potash, respectively. Five levels of each variable are used in the experimental design. The coded and measured levels for the variables are listed as

	Levels of x_I				
	-1.682	**-1.000**	**0.000**	**+1.000**	**+1.682**
N	0.425	0.913	1.629	2.345	2.833
P_2O_5	0.266	0.481	0.796	1.111	1.326
K_2O	0.278	0.607	1.089	1.571	1.899

Six center point replications were run in order to obtain an estimate of the experimental error variance. The complete second order model to be fitted to yield values is

$$Y = \beta_0 + \sum_{i=1}^{3} \beta_i x_i + \sum_{i=1}^{3} \beta_{ii} x_i^2 + \sum_{i=1}^{2} \sum_{i'=2}^{3} \beta_{ii'} x_i x_{i'} + e$$

The following table list the design settings of x_1, x_2 and x_3 and the observed values at 15 design points N, P_2O_5, K_2O and yield are in kg.

Table 2: Central Composite Rotatable Design Settings in the Coded Variables x_1, x_2 and x_3, the original variables N, P_2O_5, K_2O and the Average Yield of a Crop at Each Setting

x_1	x_2	x_3	**N**	**P_2O_5**	**K_2O**	**Yield**
-1	-1	-1	0.913	0.481	0.607	5.076
1	-1	-1	2.345	0.481	0.607	3.798
-1	1	-1	0.913	1.111	0.607	3.798
1	1	-1	2.345	1.111	0.607	3.469
-1	-1	1	0.913	0.481	1.571	4.023
1	-1	1	2.345	0.481	1.571	4.905
-1	1	1	0.913	1.111	1.571	5.287
1	1	1	2.345	1.111	1.571	4.963
-1.682	0	0	0.425	0.796	1.089	3.541
1.682	0	0	2.833	0.796	1.089	3.541
0	-1.682	0	1.629	0.266	1.089	5.436
0	1.682	0	1.629	1.326	1.089	4.977
0	0	-1.682	1.629	0.796	0.278	3.591
0	0	1.682	1.629	0.796	1.899	4.693
0	0	0	1.629	0.796	1.089	4.563
0	0	0	1.629	0.796	1.089	4.599
0	0	0	1.629	0.796	1.089	4.599
0	0	0	1.629	0.796	1.089	4.275
0	0	0	1.629	0.796	1.089	5.188
0	0	0	1.629	0.796	1.089	4.959

```
OPTIONS LINESIZE = 72;
DATA RP;
INPUT N P K YIELD;
CARDS;
….
….
….
;

PROC RSREG;
MODEL YIELD = N P K /LACKFIT NOCODE;
RUN;
```

Response Surface for Variable YIELD

Response Mean	4.464050
Root MSE	0.356424
R-Square	0.8440
Coef. of Variation	7.9843

Regression	**d.f.**	**Sum of Squares**	**R-Square**	**F-Ratio**	**Prob > F**
Linear	3	1.914067	0.2350	5.022	0.0223
Quadratic	3	3.293541	0.4044	8.642	0.0040
Crossproduct	3	1.666539	0.2046	4.373	0.0327
Total Regression	9	6.874147	0.8440	6.012	0.0049

Regression	**d.f.**	**Sum of Squares**	**R-Square**	**F-Ratio**	**Prob > F**
Lack of Fit	5	0.745407	0.149081	1.420	0.3549
Pure Error	5	0.524973	0.104995		
Total Error	10	1.270380	0.127038		

Parameter	**d.f**	**Estimate**	**Std Error**	**T-ratio**	**Prob > |T|**
INTERCEPT	1	6.084180	1.543975	3.941	0.0028
N	1	1.558870	0.854546	1.824	0.0981
P	1	-6.009301	2.001253	-3.003	0.0133
K	1	-0.897830	1.266909	-0.709	0.4947
N*N	1	-0.738716	0.183184	-4.033	0.0024
P*N	1	-0.142436	0.558725	-0.255	0.8039
P*P	1	2.116594	0.945550	2.238	0.0491
K*N	1	0.784166	0.365142	2.148	0.0573
K*P	1	2.411414	0.829973	2.905	0.0157
K*K	1	-0.714584	0.404233	-1.768	0.1075

Factor	**d.f.**	**Sum of Squares**	**Mean Squares**	**F-Ratio**	**Prob > F**
N	4	2.740664	0.685166	5.393	0.0141
P	4	1.799019	0.449755	3.540	0.0477
K	4	3.807069	0.951767	7.492	0.0047

Canonical Analysis of Response Surface

Factor	Critical Value
N	1.758160
P	0.656278
K	1.443790

Predicted value at stationary point 4.834526 kg

	Eigenvectors		
Eigenvalues	**N**	**P**	**K**
2.561918	0.021051	0.937448	0.347487
-0.504592	0.857206	-0.195800	0.476298
-1.394032	-0.514543	-0.287842	0.807708

Stationary point is a saddle point.

The eigenvalues obtained are λ_1, λ_2 and λ_3 as 2.561918, -0.504592, -1.394032. As λ_2 and λ_3 are negative, therefore, take $W_2 = W_3 = 0$. Let

M = {0.021051 0.857206 -0.514543,
0.937448 -0.195800 -0.287842,
0.34787 0.476298 0.807708};

denotes the matrix of eigenvectors. The estimated response at the stationary points be 4.834526 kg/plot. Let the desired response be Y_{des} =5.0 kg/plot. Therefore, let W_1, obtained from the equation is sqrt (difference/2.561918)=AX1, say. To obtain various different sets of many values of W_1, generate a random variable, u, which follows uniform distribution and multiply this value with $2u-1$ such that W_1 lies within the interval, (-AX1, AX1). Now to get a combination of $x_i's$ that produces the desired response obtain $\mathbf{x} = \mathbf{M} * \mathbf{W} + \mathbf{x_0}$.

```
PROC IML;
W=J(3,1,0);
Ydes=5.0;
W2=0;
W3=0;
Dif=Ydes - 4.834526;
Ax1=Sqrt(dif/2.561918);
u= uniform(0);
W1= ax1*(2*u-1); print w1;
w[1,] = w1;
w[2,] = 0;
w[3,] = 0;
m = {0.021051  0.857206    -0.514543,
     0.937448  -0.195800   -0.287842,
     0.34787    0.476298   0.807708};
xest = {1.758160, 0.656278, 1.443790};
x = m*W+xest;
print x;
run;
```

Combinations of N, P, K estimated to produce 5.0 kg/plot of Beans.

Y	N	P	K
5.0	1.760	0.730	1.471
	1.762	0.815	1.503
	1.754	0.460	1.371

One can select a practically feasible combination of N, P and K.

5. Response Surface Designs for Slope Estimation

The above discussion relates to the response surface designs for response optimization. In many practical situations, however, the experimenter is interested in estimation of the rate of change of response for given value of independent variable(s) rather than optimization of response. This problem is frequently encountered *e.g.*, in estimating rates of reaction in chemical experiments; rates of growth of biological populations; rates of changes in response of a human being or an animal to a drug dosage, rate of change of yield per unit of fertilizer dose. Efforts have been made in the literature for obtaining efficient designs for the estimation of differences in responses *i.e.*, for estimating the slope of a response surface.

Many researchers with different approaches have taken up the problem of designs for estimating the slope of a response surface. We confine ourselves to two main approaches, namely

- Slope Rotatability
- Minimax Designs

The designs possessing the property that the estimate of derivative is equal for all points equidistant from the origin are known as **slope rotatable designs**. For a second order response surface, the rate of change of response due to i^{th} independent variable is given by

$$\frac{\partial \hat{y}(x)}{\partial x_i} = b_i + 2b_{ii}x_i + \sum_{i' \neq i}^{v} b_{ii'}x_{i'}$$

For second order design satisfying (2.1) we have

$$Cov(b_i, b_{ii}) = Cov(b_i, b_{ij}) = Cov(b_{ii}, b_{ii'})$$

Thus variance of $\frac{\partial \hat{y}(x)}{\partial x_i}$ is given by

$$Var\left(\frac{\partial \hat{y}(x)}{\partial x_i}\right) = Var(b_i) + \rho^2 Var(b_{ii}) + x_i^2[4Var(b_{ii}) - Var(b_{ii'})]$$

Thus in order to obtain slope rotatable design, the design must satisfy

- Conditions of symmetry (2.1)
- $\frac{\lambda_4}{\lambda_2^2} > v/(c+v-1)$
- $4Var(b_{ii}) = Var(b_{ii'})$.

It is important to note here that no rotatable design can be slope rotatable.

A minimax design is one that minimizes the variance of the estimated slope maximized over all points in the design.

6. Web Resources on Response Surface designs

Response surface fitting can also be done using SAS and SPSS steps given on the link Response Surface Designs at Design Resources Server using the link http://www.iasri.res.in/design/Analysis%20of%20data/response_surface.html. Response surface fitting can also be performed from IP Authenticated Indian NARS Statistical Computing Portal (http://stat.iasri.res.in/sscnarsportal) using the link response surface designs.

Some Useful References

Box, G.E.P. and Draper, N.R. (1987). *Empirical model building and response surfaces*. New York, Wiley.

Das, M.N., Rajender Parsad and Manocha, V.P. (1999). Response Surface Designs. *Statistics and Applications*. **1**, 17-34.

Khuri, A.I. and Cornell, J.A. (1987). *Response Surfaces: Designs and Analysis*. New York: Marcel Dekker.

Myers, R.H. and Montgomery, D.C. (1995). *Response Surfaces Methodology: Process and product optimization using designated experiments.* John Wiley and Sons.

Parsad, R., Srivastava, R. and Batra, P.K. (2004). Designs for Fitting Response Surfaces in Agricultural Experiments. IASRI, Publication.

PROBIT ANALYSIS OF DOSE-RESPONSE DATA

ARPAN BHOWMIK AND LALMOHAN BHAR
ICAR-IASRI, Library Avenue, New Delhi – 110 012
Arpan.bhowmik@icar.gov.in, lm.bhar@icar.gov.in

1. Introduction

Probit analysis is widely used in various fields where the response variable is qualitative. The main application of probit analysis is observed in the field of toxicological studies, where it transforms the sigmoid dose-response curve to a straight line that can then be easily analyzed by regression either through least squares or maximum likelihood. In other words, probit analysis is a methodology which transforms the complex percentage affected vs. dose response into a linear relation of probit vs. dose response. The probit can then be translated into percentages. The method is appropriate because of the typical shape found in the dose response curve. The method is approximate but allows quantification of consequence due to exposure.

"Probit" is an abbreviation of the term "probability unit" (the term is attributed to Bliss) and was the first such model developed and studied to treat data such as the percentage of pest killed by a pesticide. Bliss(1934) proposed transforming the percentage killed into "probit" (he defined it arbitrarily as equal to 0 for 0.0001 and 10 for 0.9999) and included a table to aid other researchers to convert kill-percentage to probit, which then could be plotted against the logarithm of the dose i.e. dosage. The table introduced by Bliss was carried forward in an important text on toxicological application by Finney (1952). Values tabulated by Bliss can be derived from probit as defined here by adding a value of 5. Using Bliss's idea, Leslie *et al*. (1945) were able to discuss the distribution of body–weight at which female rats in the wild reach maturity through probit analysis.

Mainly Probit analysis is used to analyze data from bioassays [most commonly refers to assessment of vitamins, hormones, toxicants and drugs of all kinds by means of response produced when doses are given to experimental animals (Finney 1952)] experiments, such as proportion of insect killed by several concentrations of an insecticide or at several time intervals at one or more concentration of an insecticide (Throne *et al*., 1995). One type of assay which has been found valuable in many different fields, but especially in toxicological studies, is that dependent upon quantal or all-or-nothing response. Though quantitative measurement of a response is almost always to be preferred when practicable, there are certain responses which permit no graduation and which can only be expressed as 'occurring' or 'not-occurring'. The most common example is mortality such as in many insecticidal studies the interest lies in whether or not a test insect is dead, or whether the insect has reached a certain degree of inactivation. In fungicidal investigations, failure of a spore to germinate is a quantal response of similar importance.

2. Probit Model

In probability theory and statistics, the probit function is the inverse cumulative distribution function (CDF), associated with the standard normal distribution. An alternative distribution could be the logistic distribution, which leads to the logit or logistic model. Both logistic and probit curves are so similar that they yield almost identical results. In practice they give

estimated probabilities that differs very little (Aldrich and Nelson, 1984). The choice between logistic and probit is a matter of practical preference and experience.

For the standard normal distribution N (0, 1), the CDF is commonly denoted by Φ (*z*) (continuous, monotone increasing sigmoid function) given by,

$$\Phi(z) = \mathrm{P}(Z \le z) = \int_{-\infty}^{z} \varphi(u)\mathrm{du} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} e^{-\frac{u^2}{2}} \mathrm{du} \qquad \ldots (2.1)$$

As an example, considering the familiar fact that the N (0, 1) distribution places 95% of probability between -1.96 and 1.96, and is symmetric about zero, it follows that

$$\Phi(-1.96) = 0.025 = 1 - \Phi(1.96) \qquad \ldots (2.2)$$

The probit function gives the 'inverse' computation, generating a value of an N (0, 1) random variable, associated with specified cumulative probability. Formally, the probit function is the inverse of Φ (*z*), denoted by $\Phi^{-1}(\mathrm{p})$. Continuing the example,

$$\Phi^{-1}(0.025) = -1.96 = -\Phi^{-1}(0.975) \qquad \ldots (2.3)$$

In general,

$$\Phi(\mathrm{probit(p)}) = \mathrm{p} \text{ and } \mathrm{probit}(\Phi(\mathrm{z})) = \mathrm{z} \qquad \ldots (2.4)$$

In statistics, a probit model is a popular specification of a generalized linear model. If Y be a binary response variable, and let X be the single predictor variable, then the probit model assumes that,

$$\mathrm{P}(\mathrm{Y_i} = 1 | \mathrm{X_i} = \mathrm{x}) = \Phi(\alpha + \beta \mathrm{x_i})$$
$$= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\alpha+\beta \mathrm{x_i}} e^{-\frac{1}{2}z^2} \mathrm{dz} \qquad \ldots (2.5)$$

where Φ is the CDF of the standard normal distribution. The parameters β are estimated by maximum likelihood.

3. Quantal Response

3.1 Frequency Distribution ofTolerance

Two major components in any dose-response situation are the *stimulus* (e.g. a vitamin, a drug, a mental test or a physical force) and the *subject* (e.g. an animal, a plant, a human volunteer etc.). A stimulus is applied to the subject at a specified dose, intensity specified in units of concentration, weight, time or other appropriate measure, under controlled environmental condition. As a result subject manifests a response.

The response is quantal, occurrence or non-occurrence will depend upon the intensity of the stimulus. For any subject under controlled conditions, response occurs above a certain level of intensity, such a value is generally known as *threshold* or *limen*, but *tolerance* is now widely accepted. The tolerance value will vary from one subject to another in the population used. For quantal response data it is therefore necessary to consider distribution of tolerance over the population studied. If the dose or intensity of stimulus is measured by z, the distribution of tolerance may be expressed by:

$$dP = f(z)dz \qquad \ldots (3.1)$$

This equation states the proportion, dP, of the whole population of subject whose tolerance lie between z and z+dz at the time of testing, where dz represents a small interval on dose scale; the factor relating dP to the length of this interval is the frequency function, $f(z)$, uniquely determined for each possible value of z.

If a dose z_0 were given to the whole population, every individual whose tolerance was less than z_0 would respond. The proportion of these is P,
where

$$P = \int_0^{z_0} f(z)dz \qquad \ldots (3.2)$$

The measure of dose is here assumed to be a quantity that can conceivably range from zero to $+\infty$, response being certain for very high doses so that

$$\int_0^{\infty} f(z)dz = 1 \qquad \ldots (3.3)$$

3.2 The Dose Metameter

The frequency distribution of tolerances, as measured on the natural scale, is usually markedly skewed, but often a simple transformation of the scale of measurement will convert it to a distribution approximately of normal form. The transformed scale of dose on which tolerances are normally distributed is known as *metametric* scale, and the measure of dose is the *dose* metameter.

The transformation

$$x = \log_{10} z \qquad \ldots (3.4)$$

generally brings normality in the response variable, however for some fungicide a better transformation may be

$x = z^i$, where usually $i \leq 1$.

3.3 The Median Effective Dose

The effectiveness of a stimulus in relation to a quantal response is referred to as the *minimal effective dose,* or, for a more restricted class of stimuli as the *minimal lethal dose.* However it does not take into account the variation in tolerance within a population. The logical weakness of such concepts is the assumption that there is a dose for any given chemical, which is only just sufficient to kill all or most of the animals of a given species, and that doses a bit lesser would not kill any animal of that species. However, in toxicological studies such assumptions do not always hold good.

It might be thought that the minimal lethal dose of a poison could instead be defined as the dose just sufficient to kill a member of the species with the least possible tolerance, and also a *maximal non-lethal* dose as the dose, which will just fail to kill the most resistant member. Some doses will be so low that no test subject will succumb to them and others so high as to prove fatal at all and difficulties arise in determination of the end-points of these ranges. The problem is that of determining the dose at which the dose response curve for the whole population needs the 0% or 100% levels of kill and even a very large experiment could scarcely estimate these points with any accuracy.

Alternatively, a *median lethal dose*, or, as a more general term to include response other than mortality, a *median effective dose* is preferred. This is the dose that will produce a response in half the population. The median effective dose is commonly referred to as the ED_{50}, the more restricted concept of median lethal dose as the LD_{50}. With a fixed total number of subjects, effective doses in the neighborhood of ED_{50} can usually be estimated more precisely than those for more extreme percentage levels and this is, therefore, particularly favoured in expressing the effectiveness of the stimulus. The ED_{50} can be regarded as the median of the tolerance distribution and thus it is the level of tolerance such that exactly half the subject lies on either side of it.

For any distribution of tolerance, the ED_{50} is the value of z_0, such that

$$\int_0^{z_0} f(z)dz = 0.5 \qquad \ldots (3.5)$$

When a simple normalizing transformation for the doses is available, so that x, the normalizing measure of dose (commonly known as dosage), has a normally distributed tolerance, equation (3.1) is transformable to

$$dP = \frac{1}{\sigma\sqrt{2\Pi}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2} dx . \qquad \ldots (3.6)$$

where μ is the center of the distribution and σ^2, its variance. The μ is the population value of the mean dosage tolerance, or median effective dosage, and efforts must be directed at estimating it from the observational data. The $\log_{10}ED_{50}$ is the value of x_0 for which

$$\int_{-\infty}^{x_0} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2} dx = 0.5 \qquad \ldots (3.7)$$

The solution of equation (3.7) is μ, so that the ED_{50} is 10^{μ}.

Any two insecticides may require the same rate of application in order to be effective to half the population, but, if the distribution of tolerances has a lesser 'spread' for one than for the other, any increase or decrease from this rate will produce a greater change in mortality for the first than for the second. This spread is measured by the variance σ^2. This measure along with the ED_{50} fully describes the effectiveness of the stimulus. The smaller the value of σ^2, the greater is the effect on mortality of any change in dose.

4. Estimation of the Median Effective Dose

4.1 The N.E.D. and Probit Transformation

Initially the measure of the probability of response was proposed on a transformed scale i.e. the normal equivalent deviate (or N.E.D.). This response metameter is Y, defined by:

$$P = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{Y} e^{-\frac{1}{2}u^2} du \qquad \ldots (4.1)$$

Thus the N.E.D. of any value of P between 0 and 1 is defined as the abscissa corresponding to a probability P in a normal distribution with mean 0 and variance 1.

Equation (4.1) determines either of P and Y uniquely from the other. From integration of equation (3.6), if P is the probability of response at a dosage whose metameter is a particular value X, then

$$P = \int_{-\infty}^{X} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2} dx \qquad \ldots (4.2)$$

which by writing $x = \mu+\sigma u$

becomes $$P = \int_{-\infty}^{\frac{(X-\mu)}{\sigma}} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2}\, du \qquad \ldots (4.3)$$

Comparison of equation (4.3) with equation (4.1) shows that

$$Y = \frac{(X - \mu)}{\sigma} \qquad \ldots (4.4)$$

Thus, the relation between the dose metameter (x) and the N.E.D. of the probability of response at that dosage is a straight line.

Bliss (1934) suggested a slightly different response metameter. Bliss defined the probit of the proportion P as Y, where

$$P = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{Y-5} e^{-\frac{1}{2}u^2}\, du \qquad \ldots (4.5)$$

For any P, the probit is simply the N.E.D. increased by 5. All subsequent theory is essentially same for the two metameters. The N.E.D, however, is negative if P is less than 50%, whereas the probit is generally positive unless P is exceedingly small.

Comparison with equation (4.1) shows that the probit of the expected proportion responded is related by the linear equation

$$Y = 5 + \frac{1}{\sigma}(x-\mu) \qquad \ldots (4.6)$$

In particular, the median effective dosage is estimated as that value of *x* which gives Y = 5.

4.2 The Probit Regression Line

When experimental data on the relationship between dosage and response have been obtained, either a graphical or an arithmetical approach can be used to estimate the parameters. Both approaches depend on the probit transformation. The graphical approach is much more rapid and is sufficiently good for many purposes, but for some, more complex problems, or when an accurate assessment of the precision of estimates is wanted, the more detailed arithmetical analysis is necessary. Here graphical approach is discussed.

To start with, the percentage response observed for each dose are calculated and converted to probits by means of the following table (Finney, 1971):

Table 4.1: Transformation of percentages to probits

%	0	1	2	3	4	5	6	7	8	9
0	—	2.67	2.95	3.12	3.25	3.36	3.45	3.52	3.59	3.66
10	3.72	3.77	3.82	3.87	3.92	3.96	4.01	4.05	4.08	4.12
20	4.16	4.19	4.23	4.26	4.29	4.33	4.36	4.39	4.42	4.45
30	4.48	4.50	4.53	4.56	4.59	4.61	4.64	4.67	4.69	4.72
40	4.75	4.77	4.80	4.82	4.85	4.87	4.90	4.92	4.95	4.97
50	5.00	5.03	5.05	5.08	5.10	5.13	5.15	5.18	5.20	5.23
60	5.25	5.28	5.31	5.33	5.36	5.39	5.41	5.44	5.47	5.50
70	5.52	5.55	5.58	5.61	5.64	5.67	5.71	5.74	5.77	5.81
80	5.84	5.88	5.92	5.95	5.99	6.04	6.08	6.13	6.18	6.23
90	6.28	6.34	6.41	6.48	6.55	6.64	6.75	6.88	7.05	7.33
—	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
99	7.33	7.37	7.41	7.46	7.51	7.58	7.65	7.75	7.88	8.09

For example, for a 17% response, the corresponding probit would be 4.05. Additionally, for a 50% response (LC50), the corresponding probit would be 5.00.

The probits are then plotted against the dose metameter i.e. the logarithm (base 10) of the dose. Very extreme probits, say outside the range 2.5-7.5, carry little weight and should be disregarded. A straight line is drawn to fit the points as satisfactorily as possible. The line is nothing but the weighted regression line of the mortality probit on x. By visual inspection, the $\log_{10}ED_{50}$ is estimated from the line as m, the dosage at which Y = 5. This can be taken as estimate of μ.The estimated slope of the line (b) is an estimate of $1/\sigma$, can be obtained as the increase in Y for a unit increase in x. These two estimates are then substituted for the parameters in equation (4.6) to give the estimated relation between dosage and response. To test the hypothesis that the line is an adequate representation of the data, a χ^2 test of the form

$$\chi^2=\sum\frac{(r-np)^2}{np(1-p)} \sim \chi^2_{k-2} \qquad \ldots (4.7)$$

may be used. Here n is the number of subjects exposed to a specific concentration, r is the observed number of units respond out of n number of unit, $p = \frac{r}{n}$ is the estimated proportion of response for that particular concentration. Here k level of concentration is applied over the test subject and summation is taken over all the level of concentration tested. A value of χ^2 within the limits of random variation indicates satisfactory agreement theory (the line) and observation (the data).

Example 4.1: Table 4.2 contains the data on effect of a series of concentrations of the pesticide Rotenone when spraying on *Macrosiphoniella sanborni*, the chrysanthemum aphis, in batches of about fifty (Finney, 1971).

Table 4.2: Toxicity of Rotenone to *Macrosiphoniella sanborni*

Concentration (mg. /1.)	No. of insects (n)	No. of affected (r)	% kill (p)	Log concentration (x)	Empirical probit
10.2	50	44	88	1.01	6.18
7.7	49	42	86	0.89	6.08
5.1	46	24	52	0.71	5.05
3.8	48	16	33	0.58	4.56
2.6	50	6	12	0.41	3.82
0	49	0	0	-	-

Table 4.2 summarizes the dose metameter, percentage kill, and empirical probit values for the experiment. Over the range of concentrations tested, the relation between percentage kills and log concentration is apparently sigmoid. The percentages are plotted against the logarithm of doses and fitted with the normal sigmoid curve in Fig. 4.1.

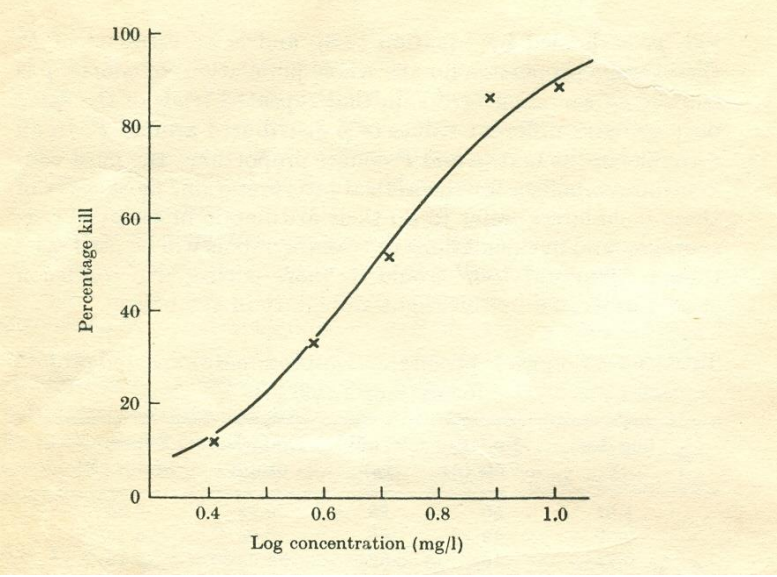


Fig.4.1: Relation between percentage kill of ***Macrosiphoniella sanborni*** and logarithm of dose of Rotenone.

In order to fit a straight line, percentages of kill have been converted into probits using Table 4.1 and are given in the last column of Table 4.3. When probits are plotted against dosages (logarithm to the base 10 of doses); they lie nearly on a straight line. Fig 4.2 gives the plot of probits vs. dosages. From this line, probits corresponding to many different values of x can be found out and converted back to percentages by using Table 4.1 inversely.

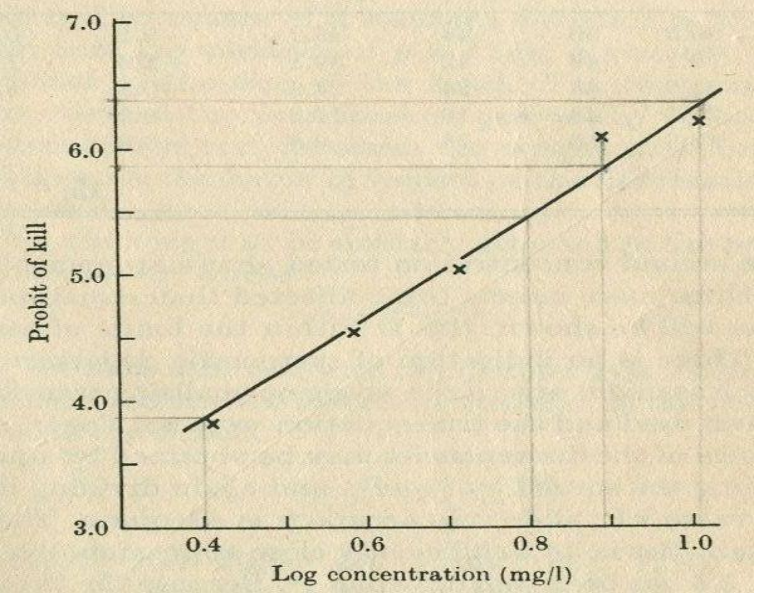


Fig. 4.2: Relation between probit of kill of ***Macrosiphoniella sanborni*** and logarithm of dose of Rotenone.

In Fig. 4.2 of the present example, a probit value of 5 is given by a dosage of m = 0.687; this therefore is the estimate of $\log_{10}ED_{50}$, and the ED_{50} is estimated as a concentration of 4.86mg/l. Similarly the $\log_{10}ED_{90}$ corresponds to a probit of 6.28 and is therefore 1.006; the ED_{90} is thus estimated as 10.14 mg/l.
Thus Fig. 5.2 can also be used to give the slope of the line: an increase of 0.319 in x corresponds with an increase of 1.28 in probit. Hence the estimated regression coefficient of probit on dosage, or the rate of increase of probit value per unit increase in x, is

$$b = 4.01 \quad \ldots (4.8)$$

Thus equation (4.6) becomes

$$Y = 5 + 4.01 (x - 0.687), \text{ or } Y = 2.25 + 4.01x \quad \ldots (4.9)$$

Equation (4.9) may be used to calculate expected numbers of insects killed at each concentration. By substitution of the values of x used in the experiment, the equation gives the values of Y which are given in column 2 of Table 4.3 as expected probits. Thus a probit of 6.30 corresponds to a percentage of between 90 and 91, or, more accurately, 90 + 2/6%. If the expected proportion for any concentration is multiplied by n, the number of insects tested at that concentration, the result is the expected number of responded insects, or the average number which would be affected in a batch of size n if equation (4.9) represents the true relationship between dosage and response. These numbers, np, may then be compared with the actual numbers affected, r, in order to judge the adequacy of the equation.

Table 4.3: Comparison of Observed and Expected Mortality.

Log concentration **(x)**	**Expected probit** **(Y)**	**% kill** **(p)**	**No. of insects** **(n)**	**No. affected** Observed (r)	**No. affected** Expected (np)	**Discrepancy** **(r-np)**	$\frac{(r-np)^2}{np(1-p)}$
1.01	6.30	90.3	50	44	45.2	-1.2	0.33
0.89	5.83	79.7	40	42	39.1	2.9	1.06
0.71	5.10	54.0	46	24	24.8	-0.8	0.06
0.58	4.58	33.7	48	16	16.2	-0.2	0.00
0.41	3.90	13.6	50	6	6.8	-0.8	0.11
							$\chi^2_{[3]}=1.56$

Since proportion of response has been estimated from the data, the degree of freedom of χ^2 (=3) is two less than the number of concentrations tested. From Fisher and Yates Table (1964, Table IV), the tabulated value of $\chi^2_{[3]}$ at 5% level of significance is 7.815. Thus the calculated value of $\chi^2_{[3]}$ (1.56) is much smaller than the tabulated value of $\chi^2_{[3]}$ at 5% level of significance. Hence, the probit regression line is very satisfactory representation of the results of the experiment.

4.4. Probit Analysis Using Statistical Software

Various statistical packages can be used for probit analysis. The easiest by far is to use a statistical package such as SPSS or SAS. Considering example (4.1), how SPSS and SAS can be used in probit analysis is described below.

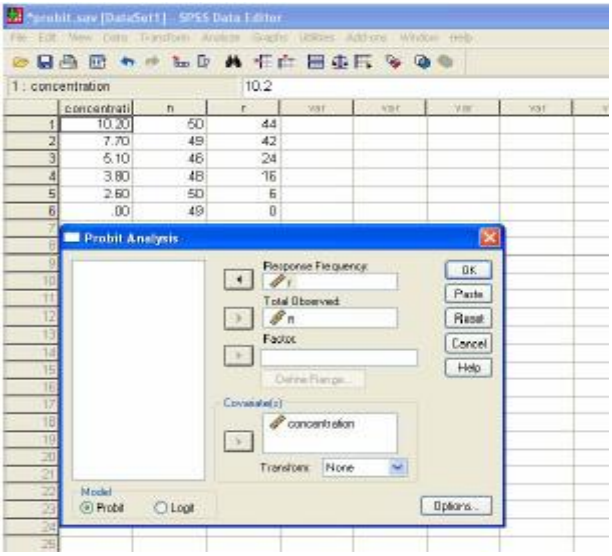
4.1 Probit Analysis through SPSS

In order to perform a probit analysis through SPSS the following steps are necessary:

Step-I: Enter data. Here con is the concentration, n is the total number of insects exposed and r is the total number of insects affected.

Step-II: Go to Analyze⇒ Regression ⇒ Probit…

Step-III: Put the variables in their respective boxes as indicated below ⇒ Select log base 10 in transform box ⇒ Choose Probit⇒ Click OK.



Step-IV: See the out put and interpret the result.

4.2 Probit Analysis through SAS

Alternatively, we can perform the same probit analysis using SAS by writing the following syntax in programme editor of SAS (Since there is very little difference in the output of SPSS and SAS, only SAS output is presented here).

```
data arpan;
input con n r;
datalines;
10.2   50     44
7.7    49     42
5.1    46     24
3.8    48     16
2.6    50     6
0      49     0

;
ods html;
Proc Probit log10 ;
Model r/n=con/lackfit inversecl;
title ('output of probit analysis');
run;
ods html close;
```

The output can be viewed as:

(output of probit analysis)

Probit Procedure

Model Information	
Data Set	WORK.ARPAN

Model Information	
Events Variable	r
Trials Variable	n
Number of Observations	5
Number of Events	132
Number of Trials	243
Name of Distribution	Normal
Log Likelihood	-120.0516414

Number of Observations Read	6
Number of Observations Used	5
Number of Events	132
Number of Trials	243

Goodness-of-Fit Tests			
Statistic	Value	DF	Pr > Chi Sq
Pearson Chi-Square	1.7289	3	0.6305

Probit Model in Terms of Tolerance Distribution	
MU	SIGMA
0.68533786	0.23734947

Estimated Covariance Matrix for Tolerance Parameters		
	MU	SIGMA
MU	0.000488	-0.000063
SIGMA	-0.000063	0.000726

(output of probit analysis)

Probit Procedure

Probit Analysis on Log10(con)			
Probability	**Log10(con)**	**95% Fiducial Limits**	
0.01	0.13318	-0.03783	0.24452
0.02	0.19788	0.04453	0.29830
0.03	0.23893	0.09668	0.33253
0.04	0.26981	0.13584	0.35834
0.05	0.29493	0.16764	0.37940
0.06	0.31631	0.19466	0.39737
0.07	0.33506	0.21832	0.41316
0.08	0.35184	0.23946	0.42733
0.09	0.36711	0.25866	0.44026
0.10	0.38116	0.27631	0.45218
0.15	0.43934	0.34898	0.50192
0.20	0.48558	0.40618	0.54202
0.25	0.52525	0.45467	0.57700
0.30	0.56087	0.49759	0.60904
0.35	0.59388	0.53666	0.63942
0.40	0.62521	0.57295	0.66905
0.45	0.65551	0.60716	0.69861

Probit Analysis on Log10(con)			
Probability	**Log10(con)**	**95% Fiducial Limits**	
0.50	0.68534	0.63983	0.72870
0.55	0.71516	0.67142	0.75986
0.60	0.74547	0.70240	0.79265
0.65	0.77679	0.73330	0.82766
0.70	0.80980	0.76480	0.86563
0.75	0.84543	0.79777	0.90761
0.80	0.88510	0.83352	0.95533
0.85	0.93133	0.87427	1.01188
0.90	0.98951	0.92456	1.08401
0.91	1.00357	0.93658	1.10155
0.92	1.01883	0.94960	1.12065
0.93	1.03562	0.96387	1.14170
0.94	1.05436	0.97976	1.16526
0.95	1.07574	0.99783	1.19218
0.96	1.10086	1.01898	1.22388
0.97	1.13174	1.04490	1.26294

Probit Analysis on Log10(con)			
Probability	**Log10(con)**	**95% Fiducial Limits**	
0.98	1.17279	1.07924	1.31498
0.99	1.23750	1.13315	1.39721

Here we only consider the metametric scale. From the output obtained through SAS, we can estimate $\log_{10}ED_{50}$ as 0.68534 mg. /l. with its 95% fiducial limits are 0.72870 and 0.63983 mg. /l. respectively.

6. Conclusions

Probit analysis has been widely used in diverse fields wherein the response variable is qualitative. Probit analysis for dose-response studies under regression framework is commonly done. In such studies, the estimation of the median effective dose (ED_{50}) i.e. the dose that will produce a response in half the population along with its variance can be chiefly done. This can be easily be achieved by using any standard statistical software.

References

Aldrich, J. H. and Nelson, F. D. (1984). *Linear probability, logit and probit models*, Sage Publications, New Delhi.

Bliss, C.I. (1934). The method of probits. *Science*, **79,** 38-39.

Bondari, K. (2002). Analysis of dosage-response data in agricultural research, *Proceedings of the South East SAS Users Group (SESUG) Annual Meeting,* Savannah, Georgia. In: http://analytics.ncsu.edu/sesug/2002/ST01.pdf

Finney, D.J. (1952).*Probit analysis: A statistical treatment of the sigmoid response curve*, 2nd ed. Cambridge University Press, London, England.

Finney, D.J. (1971). *Probit Analysis* (3rd edition). Cambridge University Press, Cambridge, England.

Fisher, R.A. and Yates, F. (1964). *Statistical Tables for Biological, Agricultural and Medical Resarch* (6th ed.). Edinburgh: Oliver and Boyd.

Fox, J. (1984). *Linear statistical models and related methods with application to social research*, Wiley, New York.

Leslie, P.H., Perry, J.S. and Watson, J.S. (1945). The determination of the median body-weight at which female rats reach maturity. *Proc. Zool. Soc*., **115,** 473-88.

Parson, S.D., Hunter, G.L. and Rayner, A.A. (1967). Use of probit analysis in a study of the effect of the ram on time of ovulation in the ewe. *J.Reprod.Fert*., **14,** 71-80.

Throne, J.E., Weaver, D.K. and Baker, J.E. (1995). Probit analysis: Assessing goodness-of-fit based on back transformation and residuals. *J. Econ. Entomol.*, **88**(5), 1513-1516.

LOGISTIC REGRESSION USING SPSS

Arpan Bhowmik
ICAR-IASRI, Library Avenue, New Delhi – 110 012
Arpan.bhowmik@icar.gov.in

1. Introduction

Regression analysis is a widely used method for obtaining a functional relationship between the response or dependent variable and one or more explanatory or predictor variables. In all the regression models, we implicitly assumed that the response variable is quantitative in nature whereas the explanatory variables are either quantitative, qualitative or a mixture thereof. In case of qualitative or non-metric response variable usual assumptions of regression models are violated, hence, it is better to look for alternative models. In practice, situations involving categorical outcomes are quite common. Suppose we want to study the labour force participation (LFP) decision of adult males. Since an adult is either in the labour force or not, LFP is a yes or no decision. Similarly, in the setting of evaluating an extension program, for example, predictions may be made for the dichotomous outcome of success/failure or improved/not-improved. An economist may be interested in determining the probability that an agro-based industry will fail given a number of financial ratios and the size of the firm (i.e. large or small) etc.

Usually discriminant analysis could be used for addressing each of the above problems. However, because the independent variables are mixture of categorical and continuous variables, the multivariate normality assumption may not hold. In these cases the most preferable technique is the logistic regression analysis as it does not make any assumptions about the distribution of the independent variables.

2. Violation of Assumptions of Linear Regression Model when Response is Qualitative

Linear regression is considered in order to explain the constraints in using such model when the response variable is qualitative. Consider the following simple linear regression model with single predictor variable and a binary response variable:

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i \quad , i = 1, 2, \ldots, n$$

where the outcome Y_i is binary (taking values 0,1), $\varepsilon_i \sim N(0, \sigma_\varepsilon^2)$, and are independent and n is the number of observations.

Let π_i denote the probability that $Y_i = 1$ when $X_i = x$, i.e.

$$\pi_i = P(Y_i = 1 | X_i = x) = P(Y_i = 1)$$

thus $P(Y_i = 0) = 1 - \pi_i$.

Under the assumption $E(\varepsilon_i) = 0$, the expected value of the response variable is

$E(Y_i) = 1.(\pi_i) + 0.(1-\pi_i) = \pi_i$

If the response is binary, then the error terms can take on two values, namely,

$$\varepsilon_i = 1 - \pi_i \quad \text{when } Y_i = 1$$

$$\varepsilon_i = -\pi_i \quad \text{when } Y_i = 0$$

Because the error is dichotomous (discrete), so normality assumption is violated. Moreover, the error variance is given by:

$$V(\varepsilon_i) = \pi_i(1-\pi_i)^2 + (1-\pi_i)(-\pi_i)^2$$

$$= \pi_i(1-\pi_i)$$

It can be seen that variance is a function of π_i's and it is not constant. Therefore the assumption of homoscadasticity (equal variance) does not hold.

3. Logistic regression

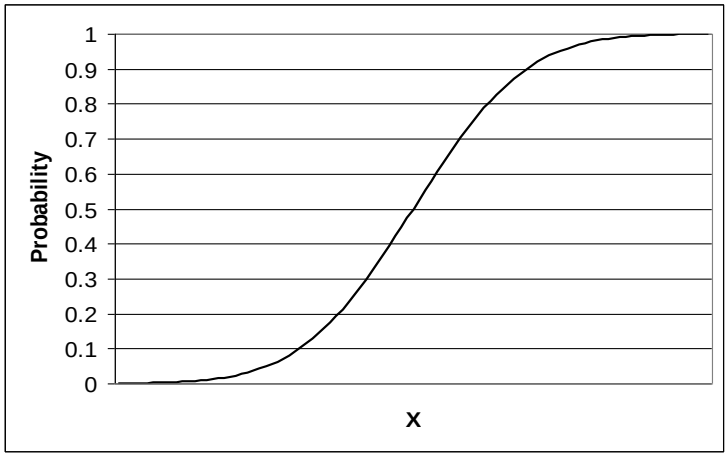
Logistic regression is normally recommended when the independent variables do not satisfy the multivariate normality assumption and at the same time the response variable is qualitative. Situations where the response variable is qualitative and independent variables are mixture of categorical and continuous variables, are quite common and occur extensively in statistical applications in agriculture, medical science etc. The statistical model preferred for the analysis of such binary (dichotomous) responses is the binary logistic regression model, developed primarily by a researcher named Cox during the late 1950s. Processes producing sigmoidal or elongated S-shaped curves are quite common in agricultural data. Logistic regression models are more appropriate when response variable is qualitative and a non-linear relationship can be established between the response variable and the qualitative and quantitative factors affecting it. It addresses the same questions that discriminant function analysis and multiple regression do but with no distributional assumptions on the predictors. In logistic regression model, the predictors need not have to be normally distributed, the relationship between response and predictors need not be linear or the observations need not have equal variance in each group etc. A good account on logistic regression can be found in Fox (1984) and Kleinbaum (1994).

The problem of non-normality and heteroscadasticity leads to the non applicability of least square estimation for the linear probability model. Weighted least square estimation, when used as an alternative, can cause the fitted values not constrained to the interval (0, 1) and therefore cannot be interpreted as probabilities. Moreover, some of the error variance may come out to be negative. One solution to this problem is simply to constrain the probability of outcome to the unit interval while retaining the linear relation between probability of outcome and regressor within the interval. However, this constrained linear probability model has certain unattractive features such as abrupt changes in slope at the extremes 0 and 1 making it hard for fitting the same on data. A smoother relation between the probability of outcome and regressor is generally more sensible. To correct this problem, a positive monotone (i.e. non-decreasing) function is required to transform linear combination of regressor to unit interval. Any cumulative probability

distribution function (CDF) meets this requirement. That is, re-specify the model as $\pi_i = P(\beta_0 + \beta_1 x_i)$. where, π_i is the probability of outcome and P is the cumulative distribution function. Moreover, it is advantageous if P is strictly increasing, for then, the transformation is one-to-one, so that model can be rewritten as $P^{-1}(\pi_i) = (\beta_0 + \beta_1 x_i)$, where P^{-1} is the inverse of the CDF P. Thus the non-linear model for itself will become both smooth and symmetric, approaching $\pi = 0$ and $\pi = 1$ as asymptotes. Thereafter maximum likelihood method of estimation can be employed for model fitting.

3.1 Properties of Logistic Regression Model

The Logistic response function resembles an S-shape curve, a sketch of which is given in the following figure. Here the probability π initially increases slowly with increase in X, and then the increase accelerates, finally stabilizes, but does not increase beyond 1.



The shape of the S-curve can be reproduced if the probabilities can be modeled with only one predictor variable as follows:

$\pi = P(Y=1|X=x) = 1/(1+e^{-z})$

where $z = \beta_0 + \beta_1 x$, and e is the base of the natural logarithm. Thus for more than one (say r) explanatory variables, the probability π is modeled as

$\pi = P(Y=1|X_1=x_1 ... X_r=x_r)$

$=1/(1+e^{-z})$

where $z = \beta_0+\beta_1 x_1+...+\beta_r x_r$.

This equation is called the logistic regression equation. It is nonlinear in the parameters β_0, $\beta_1 \ldots \beta_r$. Modeling the response probabilities by the logistic distribution and estimating the parameters of the model constitutes fitting a logistic regression. The method of estimation generally used is the maximum likelihood estimation method.

To explain the popularity of logistic regression, let us consider the mathematical form on which the logistic model is based. This function, called f (z), is given by

$$f(z) = 1/(1+e^{-z}),\ -\infty < z < \infty$$

Now when $z = -\infty$, $f(z) = 0$ and when $z = \infty$, $f(z) = 1$. Thus the range of f (z) is 0 to1. So the logistic model is popular because the logistic function, on which the model is based, provides

- Estimates that lie in the range between zero and one.
- An appealing S-shaped description of the combined effect of several explanatory variables on the probability of an event.

3.2. Maximum Likelihood Method of Estimation of Logistic Regression

Generally, the maximum likelihood method is used for estimating the parameters of the logistic regression model. The maximum likelihood estimates β_0 and β_1 in the simple logistic regression model are those values of β_0 and β_1 that maximize the log-likelihood function. No closed-form solution exists for the values of β_0 and β_1 that maximize the log-likelihood function. Computer intensive numerical search procedures are therefore required to find the maximum likelihood estimates $\hat{\beta}_0$ and $\hat{\beta}_1$. Standard statistical software such as SPSS (Analyze- Regression-Binary Logistic) provide maximum likelihood estimates for logistic regression. Once these estimates $\hat{\beta}_0$ and $\hat{\beta}_1$ are found, by substituting these values into the response function the fitted response function, say, $\hat{\pi}_i$, can be obtained. The fitted response function is as follows:

$$\hat{\pi}_i = \left(\frac{1}{1+e^{-\left(\hat{\beta}_0+\hat{\beta}_1 X_i\right)}} \right)$$

When log of the odds of occurrence of any event is considered using a logistic regression model, it becomes a case of logit analysis. Thus formed logit model will have its right hand side as a linear regression equation.

4. Practical Exercise using SPSS

Sometimes quantitative information on adoption of a technology is not available but is available in qualitative form such as adopted / non-adopted, low / high adoption etc. The statistical model preferred for the analysis of such binary (dichotomous) responses is the binary logistic regression model. It can be used to describe the relationship of several independent variables to the binary (say, named 0 & 1) dependent variable. The logistic regression is used for obtaining probabilities of occurrence, say E, of the different categories when the model is of the form: $P(E=1) = \frac{1}{1+\exp(-z)}$ where z is a function of associated variables, if $P(E=1) \geq 0.5$ then there is more chance of occurrence of an event and if $P(E=1) < 0.5$ then probability of occurrence of the

event is minimum. If the experimenter wants to be more stringent, then the cutoff value of 0.5 could be increased to, say, 0.7.

For performing a Logistic Regression Analysis using SPSS
From the menus choose:
Analyze
Regression
Binary Logistic...
Select one dichotomous dependent variable. This variable may be numeric or string.
Select one or more covariates.
To include interaction terms, select all of the variables involved in the interaction and then select a*b.
A detailed practical exercise with the real data involving both linear probability model and logistic regression model will be carried out using SPSS in the class.

References:

Hosmer, D.W. and Lemeshow, S. (2000). Applied Logistic Regression, New York: Wiley.
Fox, J. (1984). *Linear statistical models and related methods with application to social research*, Wiley, New York.
Kleinbaum, D.G. (1994). Logistic regression: A self learning text, New York: Springer.

PRINCIPAL COMPONENT ANALYSIS

Rahul Banerjee
ICAR-IASRI, Library Avenue, New Delhi – 110 012
Rahul.banerjee@icar.gov.in

Introduction

Principal component analysis (PCA) is one among techniques for taking high dimensional data, and using the dependencies between the variables to represent it in a more tractable, lower-dimensional form, without losing too much information. It was invented by Pearson (1901) and Hotelling (1933) and first applied in ecology by Goodall (1954) under the name "Factor Analysis". During 1970 PCA was considered as the ordination method of choice for community data. Further, simulation studies made by Swan (1970), Austin and Noy-Meir (1971) demonstrated the horseshoe effect and showed that the linear assumption of PCA was not compatible with the non-linear structure of community data. Recently, it has stimulated the search for more appropriate ordination method and is most widely used as well as well-known of the "standard" multivariate methods. The purpose of principal component analysis is to derive a small number of linear combinations (principal components) of a set of variables that retain as much information in the original variables as possible. Often a small number of principal components can be used in place of the original variables for plotting, regression, clustering and so on. Principal component analysis can also be viewed as a technique to remove multicollinearity in the data. PCA is a way of identifying patterns in the data; data is expressed in such a way that the similarities and differences are highlighted. Once the patterns are found in the data, it can be compressed (reduce the number of dimensions) without losing much information.

PCA is one of the simplest and most robust ways of dimensionality reduction. It is also one of the oldest methods, and has been rediscovered many times in many fields, so it is also known as the Karhunen-Loève transformation, the Hotelling transformation, the method of empirical orthogonal functions, and singular value decomposition. PCA is concerned with explaining the variance covariance structure of a set of variables through a few linear combinations of these variables. Mathematically it is orthogonal linear transformation of data to a new coordinate system such that the greatest variance by some projection of the data comes to lie on the first coordinate (called the first principal component (PC)), the second greatest variance on the second coordinate, and so on. PCA is an intermediate step for further treatment of data that includes regression analysis, indexing, assigning weights, etc.

In this technique, we transform the original set of variables to a new set of uncorrelated random variables. These new variables are linear combinations of the originals variables and are derived in decreasing order of importance so that the first principal component accounts for as much as possible of the variation in the original data.

PCA basically tries to explain the total data variability with the help of a fewer number of linear combinations of the original data called the Principal Components. The broad objective of PCA is the reduction in the data dimensions.

"The information content of the new variables is as much as the information content of the original variables."

Mathematically:

Let $\mathbf{x} = (x_1, x_2, \ldots, x_p)'$ be a px1 random vector with the dispersion matrix as

$\boldsymbol{\Sigma} = \left((\sigma_{ij})\right) = \begin{bmatrix} \sigma_{11} & \cdots & \sigma_{1p} \\ \vdots & \ddots & \vdots \\ \sigma_{p1} & \cdots & \sigma_{pp} \end{bmatrix}$; $\boldsymbol{\Sigma}$ is assumed to be a positive definite matrix

The total variability in **x** is given by the trace ($\boldsymbol{\Sigma}$).

PCA aims to replace **x** with some $\mathbf{y} = (y_1, y_2, \ldots, y_p)'$; where y_i's are the linear combinations of x_i such that.

1. y_i's are uncorrelated i.e. Cov(y_i,y_j)=0$\forall i \neq j$
2. The total variation in x = The total variation in y
3. Total variation of $(y_1, y_2, \ldots, y_k)' \approx$ Total variation of $(y_1, y_2, \ldots, y_p)'$; where k<<p

Definition: The Principal Components (PCAs) are the uncorrelated linear combinations $(y_1, y_2, \ldots, y_p)'$ whose variance are in the decreasing order *i.e.* y_1 explaining the maximum variability followed by y_2 and so on.

The first principal component may be defined as the linear combination:

$y_1 = \boldsymbol{l_1'}\mathbf{x}$ *i.e.* $y_1 = a_{11}x_1 + a_{12}x_2 + \cdots + a_{1p}x_p$

that maximizes Var($\boldsymbol{l_1'}\mathbf{x}$) subject to $\boldsymbol{l_1'}\boldsymbol{l_1} = 1$; *i.e.* $a_{11}^2 + a_{12}^2 + \cdots + a_{1p}^2 = 1$

This constraint is introduced because if this is not done Var(y_1) can be increased simply by multiplying any a s a_{1j}'s by a constant factor.

The second principal component may be defined as the linear combination:

$y_2 = \boldsymbol{l_2'}\mathbf{x}$; *i.e.* $y_2 = a_{21}x_1 + a_{22}x_2 + \cdots + a_{2p}x_p$

that maximizes Var($\boldsymbol{l_2'}\mathbf{x}$) subject to $\boldsymbol{l_2'}\boldsymbol{l_2} = 1; a_{21}^2 + a_{22}^2 + \cdots + a_{2p}^2 = 1;\ Cov(\boldsymbol{l_1'}\mathbf{x}, \boldsymbol{l_2'}\mathbf{x}) = 0$

Similarly,

The ith principal component may be defined as the linear combination:

$y_i = \boldsymbol{l_i'}\mathbf{x}$ that maximizes Var($\boldsymbol{l_i'}\mathbf{x}$) subject to $\boldsymbol{l_i'}\boldsymbol{l_i} = 1;\ Cov(\boldsymbol{l_i'}\mathbf{x}, \boldsymbol{l_k'}\mathbf{x}) = 0 \forall k < i$

It is quite likely that first few principal components account for most of the variability in the original data. If so, these few principal components can then replace the initial p variables in subsequent analysis, thus reducing the effective dimensionality of the problem. An analysis of principal components often reveals relationships that were not previously suspected and thereby allows interpretation that would not ordinarily result. However, Principal Components Analysis is more of a mean to an end rather than end in itself because this frequently serves as intermediate steps in much larger investigations by reducing the dimensionality of the problem and providing easier interpretation. It is a mathematical technique, which does not require user to specify the statistical model or assumption about distribution of original variates. It may also be mentioned that principal components are artificial variables and often it is not possible to assign physical meaning to them. Further, since Principal Components Analysis transforms

original set of variables to new set of uncorrelated variables. It is worth stressing that if the original variables are uncorrelated, then there is no point in carrying out the Principal Components Analysis. It is important to note here that the principal components depend on the scale of measurement. Conventional way of getting rid of this problem is to use the standardized variables with unit variances.

Major Uses of PCA:

- Data Dimension Reduction.
- Data Projection & Visualization.
- Idea about Data Clusters.
- Multidimensional Outlier detection.
- Ranking of multidimensional data.
- Checking for multidimensional normality.

Assumptions & Limitations:

Sample size: Correlation coefficients tend to be reliable when estimated from large samples. With increase in sample sizes the sampling distribution become narrower implying that in normal run we get more precise estimates. Therefore, it is important that sample size is large so enough that correlations area could be reliably estimated. Though there is no scientific answer to exact sample size for running of PCA, some arbitrary rules of thumbs say that there should be at least 10 observations for each variable. Hutcheson and Sofroniou (1999) recommend that the minimum sample size should be at least 150- 300. For a few highly correlated variables sample size closer to 150 is sufficient. Some suggest that number of variable determines the size of sample. Bryant and Yarnold (1995), Nunnaly (1978) and Gorsuch (1983) recommend that the ideal sample to variable ratio should be at least five. The rule of significance states that there should be 51 extra observations than the number of variables, to support chisquare testing (Lawley and Maxwell, 1971). These rules are not mutually exclusive.

Normality: PCA is a generally a non-parametric analysis. If variables are normally distributed, the solution is enhanced. To the extent normality fails, the solution is degraded by may still be worthwhile.

Linearity: The analysis is degraded when linearity fails, because correlation measures linear relationship and does not reflect non-linear relationship

Orthogonality: Principal components (PCs) are orthogonal. That means the second PC will be perpendicular to first PC and subsequently the third PC will be perpendicular to second PC.

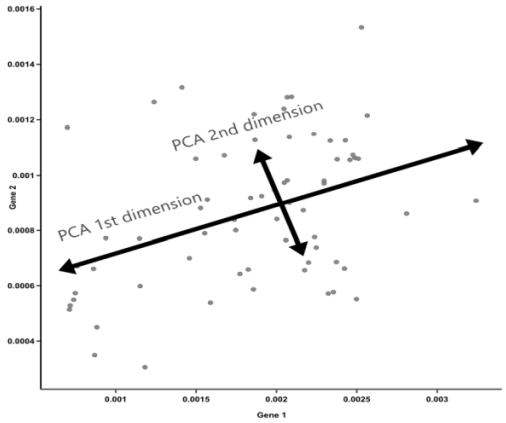


Fig 1: Orthogonality in the principal components

Steps in Principal Component Analysis:

There are five major steps in doing PCA analysis as listed below.

Step 1: Preparation of data: The first and foremost step is preparation of data set. The preparation of data includes finding correlation and, centring and normalizing data. It is worth mentioning that PCA cannot always reduce a large number of original variables to a small number of transformed variables. If the original variables are uncorrelated, the PCA is of no use. Contrary to this, a significant reduction is possible if the original variables are highly correlated. If the variables are significantly correlated, we go for further analysis, i.e. centering and scaling the data. The centring of data produces a data set whose mean is zero. The centered data matrix also called as 'X' data matrix is done by subtracting the mean from each variable. If the variances of the variables in the data are significantly different, it is better to scale the data to unit variance. This is achieved by dividing each variable by its standard deviation.

Step 2: Computation of sample variance/covariance matrix C: The formula for calculation of covariance is as follows.

$$\mathbf{C} = \frac{1}{N-1}(X - \overline{X'})'(X - \overline{X'})$$

Where, C is covariance matrix, N is sample size, X' is transpose of centred data matric X. Since we have centred the dataset to zero mean, the mean vector $\overline{X'}$ will be zero. Therefore, covariance formula will be as follow.

$$\mathbf{C} = \frac{1}{N-1}X'X$$

In order to prevent undue influence of any variables of different scale or unit on the PCs, it is common to standardise these so as to have zero means and unit variances. The co-variance matrix C then takes the form of the correlation matrix.

Step 3: Finding eigenvalues and eigenvectors: The third step in PCA is computation of eigenvalues (λ_i) or latent root/unit or normalised eigenvectors (e_i) of A. Here A denotes covariance matrix. An eigenvalue of a matrix A is a scalar (λ) if there is a non-zero vector x satisfies that $Ax = \lambda x$. The eigenvalues of matrix A can be found by solving the characteristics equation, $\det(A - \lambda I) = 0$, where, det is determinant and I is an identity matrix.

However, this is possible only if the number of variables is small. If there are too many variables, solving for λ is non-trivial and we use other methods (Gentle et al., 2004; Golub and van der Vorst, 2000). An important property of the eigenvalues is that they add up to the sum of the diagonal elements of A, i.e., sum of variances of PCs is equal to sum of variances of original variables. Once the eigenvalues of a matrix (A) have been found, we can find the eigenvectors by Gaussian Elimination.

After computing eigenvalues, arrange the variable in descending order based on per cent variation explained by them. Eigenvalues determine the radius of "ellipse". By this process, we will be able to extract lines that characterise the data. The first eigenvector will go through the middle of the data point, as if it is the line of best fit. The second eigenvector will give us the other, less important, pattern of the data and so on. The number of chosen eigenvectors will be the number of dimensions of the new data set. The components of lesser significance can be ignored, so as to reduce the dimensions of the data set.

Step 4: Selecting PCs: There are different methodologies for choosing the number of PCs; including both heuristic and statistical. The common methodologies used for selecting PCAs are explained below.

- Selection based on proportion of variance that components explain: The cumulative proportion can be used to determine the amount of variance that PCs explain. The PCs are retained that explain an acceptable level of variance. As mentioned above the acceptable level depends on the application. For example, in evocative purposes, you may only need 80% of the variance explained while in other analysis one may want to have at least 90% of the variance explained by PCs.
- Selection based on eigenvalues: Size of the eigenvalue can also be used to determine the number of PCs. Generally, the PCs with eigenvalue greater than one are retained for further analysis.
- Scree plot method of PC selection: The scree plot orders the eigenvalues from largest to smallest. The ideal pattern is a steep curve, followed by a bend, and then a straight line. The "elbow" location (PC 4 in Figure 2) in a scree plot might indicate a good number of PCs to retain. A more precise method to "detect the elbow" is to start at the right-hand side of the scree plot and look at the points that roughly lie on a straight line. The leftmost point along this line indicates the number of components to be retained. For example, in Figure 2, components 4–7 are almost forming trend line, therefore components 1–4 would sufficiently explain the total variance. This method was proposed by Cattell (1966) and later revised by Cattell and Jaspers (1967).

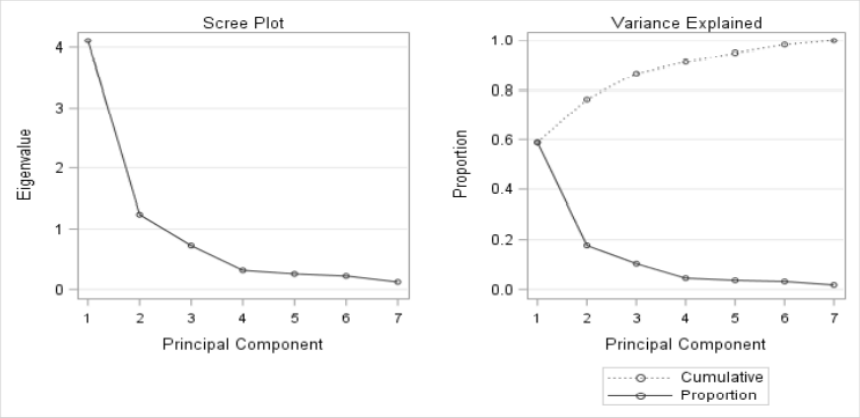


Fig 2: Scree plot to determine number of PCs

Step 5: Deriving new dataset: After computation of matrix of eigenvectors and deciding PCs, the final step in PCA is deriving new dataset. The new dataset is derived by taking Y = XV, where Y, matrix of transformed data also called as matrix of PC scores. X is original dataset; and V, transformation matrix (eigenvector matrix). Basically we have transformed our dataset so that it is expressed in terms of pattern between them, where the pattern is the lines that most closely describe the relationship between the data.

Rotation of principal components:

The interpretation of the components (which is governed by the loadings–the correlations of the original variables with the newly created components) can be enhanced by "rotation" which could be thought of a set of coordinated adjustments of the vectors on a bi-plot. There is no single optimal way of doing rotations, but probably the most common approach is "varimax" rotation in which the components are adjusted in a way that makes the loadings either high positive (or negative) or zero, while keeping the components uncorrelated or orthogonal. Varimax rotation assumes that there is no intercorrelations between components. Varimax rotation (also called Kaiser-Varimax rotation) maximizes the sum of the variance of the squared loadings, where 'loadings' means correlations between variables and factors. This usually results in high factor loadings for a smaller number of variables and low factor loadings for the rest. Remaining components all have eigenvalues of more than one. In simple terms, the result is a small number of important variables are highlighted, which makes it easier to interpret your results. One side-product of rotation is that the first, or principal components is no longer optimal or the most efficient single-variable summary of the data set, but losing that property is often worth the increase in interpretability.

Example 4: Let us consider the following data on average minimum temperature (x_1), average relative humidity at 8 hrs. (x_2), average relative humidity at 14 hrs. (x_3) and total rainfall in cm. (x_4) pertaining to Raipur district from 1970 to 1986 for kharif season from 21st May to 7th Oct.

	x_1	x_2	x_3	x_4
	25.0	86	66	186.49
	24.9	84	66	124.34
	25.4	77	55	98.79
	24.4	82	62	118.88

	22.9	79	53	71.88
	7.7	86	60	111.96
	25.1	82	58	99.74
	24.9	83	63	115.20
	24.9	82	63	100.16
	24.9	78	56	62.38
	24.3	85	67	154.40
	24.6	79	61	112.71
	24.3	81	58	79.63
	24.6	81	61	125.59
	24.1	85	64	99.87
	24.5	84	63	143.56
	24.0	81	61	114.97
Mean	23.56	82.06	61.00	112.97
S.D.	4.13	2.75	3.97	30.06

with the variance co-variance matrix:

$$\boldsymbol{\Sigma} = \begin{bmatrix} 17.02 & -4.12 & 1.54 & 5.14 \\ & 7.56 & 8.50 & 54.82 \\ & & 15.75 & 92.95 \\ & & & 903.87 \end{bmatrix}$$

Find the eigenvalues and eigenvectors of the above matrix. Arrange the eigenvalues in decreasing order. Let, the eigenvalues in decreasing order and corresponding eigenvectors are:

Eigen Values	Eigen vector
λ_1=916.902	$\mathbf{a_1}$=(0.006, 0.061, 0.103, 0.993)
λ_2=18.375	$\mathbf{a_2}$ = (0.955, -0.296, 0.011, 0.012)
λ_3=7.87	$\mathbf{a_3}$=(0.141, 0.485, 0.855, -0.119)
λ_4=1.056	$\mathbf{a_4}$=(0.260,0.820, -0.509,0.001)

The principal components for this data will be

$$y_1 = 0.006x_1 + 0.061x_2 + 0.103x_3 + 0.993x_4$$

$$y_2 = 0.955x_1 - 0.296x_2 + 0.011x_3 + 0.012x_4$$

$$y_3 = 0.141x_1 + 0.485x_2 + 0.855x_3 - 0.119x_4$$

$$y_4 = 0.26x_1 + 0.82x_2 - 0.509x_3 + 0.001x_4$$

The variance of principal components will be eigenvalues i.e.

$\text{Var}(y_1) = 916.902$, $\text{Var}(y_2) = 18.375$, $\text{Var}(y_3) = 7.87$, $\text{Var}(y_4) = 1.056$

The total variation explained by principal components is

$\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 916.902 + 18.375 + 7.87 + 1.056 = 944.20$

As such, it can be seen that the total variation explained by principal components is same as that explained by original variables. It could also be proved mathematically as well as empirically that the principal components are uncorrelated.

The proportion of total variation accounted for by the principal components is

The proportion of total variation accounted for by the principal components is:

$$\frac{\lambda_1}{\lambda_1+\lambda_2+\lambda_3+\lambda_4}=\frac{916.902}{944.203}=0.97$$

Continuing, the first two components account for a proportion

$$\frac{\lambda_1+\lambda_2}{\lambda_1+\lambda_2+\lambda_3+\lambda_4}=\frac{935.277}{944.203}=0.99$$

of the total variance.

Hence, in further analysis, the first or first two principal components y_1 and y_2 could replace four variables by sacrificing negligible information about the total variation in the system. The scores of principal components can be obtained by substituting the values of x_i's in the equations of y_i's. For above data, the first two principal components for first observation i.e. for year 1970 can be worked out as

$$y_1=0.006x25.0+0.061x86+0.103x66+0.993x186.49=197.380$$

$$y_2=0.955x25.0-0.296x86+0.011x66+0.012x186.49=1.383$$

Similarly, for the year 1971

$$y_1=0.006x24.9+0.061x84+0.103x66+0.993x124.34=135.54$$

$$y_2=0.955x24.9-0.296x84+0.011x66+0.012x124.34=1.134$$

Thus, the whole data with four variables can be converted to a new data set with two principal components.

Following steps of SAS may be used for performing the principal component analysis.

The PROC PRINCOMP can be used for performing principal component analysis. Raw data, a correlation matrix, a covariance matrix or a sum of squares and cross products (SSCP) matrix can be used as input data. The data sets containing eigenvalues, eigenvectors, and standardized or unstandardized principal component scores can be created as output. The basic syntax of PROC PRINCOMP is as follows:

PROC PRINCOMP Cov ;

BY variables; FREQ Variable;

PARTIAL Variables; VAR Variables;

WEIGHT Variable;

RUN;

The PROC PRINCOMP and RUN are must. However, the VAR statement listing the numeric variables to be analysed is usually used alongwith PROC PRINCOMP statement. If the DATA= data set is TYPE=SSCP, the default set of variables does not include intercept.

Therefore, INTERCEPT may also be included in the VAR statement. The following options are available in PROC PRINCOMP.

A. DATA SETS SPECIFICATION

1. DATA= SAS-data-set: names the SAS data set to be analysed. This data set can be ordinary data set or a TYPE = CORR, COV, FACTOR, UCORR or UCOV data set.

2. OUT = SAS-data-set: creates an output data set containing original data alongwith principal component scores.

3. OUTSTAT-SAS-data-set: creates an output data set containing means, standard deviations, number of observations, correlations or covariances, eigenvalues and eigenvectors.

B. ANALYTICAL DETAILS SPECIFICATION

1. COV: computes the principal components from the covariance matrix. The default option is computation of principal components using a correlation matrix.

2. N=: the non-negative integer equal to the number of principal components to be computed.

3. NOINT : omits the intercept from the model

4. PREFIX= name: specifies a prefix for naming the principal components. The default option is PRIN1, PRIN2, ...

5. STANDARD (STD): standardizes the principal component scores to unit variance from the variance equal to corresponding eigenvalue.

6. VARDEF=DF | N | WDF | WEIGHT: specifies the divisor (error degree of freedom | number of observations | sun of weights | sum of weights-1) in calculating variances and standard deviations. The default option is DF.

Besides these options NOPRINT option suppresses the output.

The other statements in PROC PRINCOMP are:

By variables: obtains the separate analysis on observations in groups defined by variables.

FREQ statement: It names a variable that provides frequencies of each observation in the data set. Specifically, if n is the value of the FREQ variable for a given observation, then that observation is used ‘n’ times.

PARTIAL Statement: used to analyze for a partial correlation or covariance matrix. VAR statement: Lists the numeric variables to be analysed.

WEIGHT Statement: If we want to use relative weights for each observation in the input data set, place the weights in a variable in the data set and specify the name in a weight statement. This is often done when the variance associated with each observation is different and the values of the weight variable are proportional to reciprocals of the variances.

The observation is used in the analysis only if the value of the WEIGHT statement variable is non-missing and greater than zero.

The other closely related procedures with PROC PRINCOMP are PROC PRINQUAL: It performs a principal component analysis of a qualitative data. PROC CORRESP: performs correspondence analysis, which is a weighted principal component analysis of contingency tables.

For detailed steps for performing principal component analysis using SAS, a reference may be made to link "Analysis of Data" at Design Resources Server. SAS codes can be obtained from https://drs.icar.gov.in/Analysis of data/principal_component.html

REFERENCES

Austin, M. P. and I. Noy-Meir (1971), The problem of non-linearity in ordination: experiments with two gradient models. Journal of Ecology, 59: 763–773.

Bose, A. and A. Dey (2011), The wonderful world of eigenvalues. In: R. Sujatha, H. N. Ramaswamy, C. S. Yogananda. (Eds.) Math Unlimited: Essays in Mathematics. CRC Press, Boca Raton.

Bryant, F. B. and P. R. Yarnold (1995), Principal components analysis and exploratory and confirmatory factor analysis. In Grimm and Yarnold, Reading and understanding multivariate analysis. American Psychological Association Books.

Cattell, R. B. (1966), The scree test for the number of factors. Multivariate behavioral Research, 1(2): 245-276.

Cattell, R. B. and J. Jaspers (1967), A general plasmode (No. 30-10-5-2) for factor analytic exercises and research. Multivariate Behavioral Research Monographs.

Cangelosi, R. and A. Goriely (2007), Component retention in principal component analysis with application to cDNA microarray data. Biology Direct, 2 (2): 1-21.

Gentle, J. E., Härdle W. and Mori, Yuichi (eds.) (2004), Handbook of Computational Statistics: Concepts and Methods, Springer.

Golub, G. H. and H.A. van der Vorst (2000), Eigenvalue computation in the 20th century, Journal of Computational and Applied Mathematics, 123 (1-2): 35-65.

Goodall, D. W. (1954). Objective methods for the classification of vegetation: An essay in the use of factor analysis. Australian Journal of Botany, 2: 304-324.

Gorsuch, R. L. (1983), Factor Analysis. Hillsdale, NJ: Lawrence Erlbaum. Orig. ed. 1974.

Hotelling, H. (1933), Analysis of a complex of statistical variables into principal components, Journal of Educational Psychology 24: 417-441, 498-520.

Hutcheson, G. and N. Sofroniou (1999), The multivariate social scientist: Introductory statistics using generalized linear models, Thousand Oaks, CA: Sage Publications.

Jackson, D. A. (1993), Stopping rules in principal components analysis: a comparison of heuristical and statistical approaches, Ecology, 74(8).

Lawley, D. N. and A. E. Maxwell (1971), Factor analysis as a statistical method, London: Butterworth and Co.

Nunnaly, J. (1978), Psychometric theory, New York: McGraw-Hill.

Pearson, K. (1901), On lines and planes of closest fit to systems of points in space. Philosophical Magazine, 2(11):559–572.

Swan, J. M. A. (1970), An examination of some ordination problems by the use of simulated vegetational data. Ecology, 51: 89–102.

FACTOR ANALYSIS

Rahul Banerjee
ICAR-IASRI, Library Avenue, New Delhi – 110 012
Rahul.banerjee@icar.gov.in

Introduction

Factor analysis is a statistical method used to describe variability among observed, correlated variables in terms of a potentially lower number of unobserved variables called **factors**.

For example, it is possible that variations in six observed variables mainly reflect the variations in two unobserved (underlying) variables. Factor analysis searches for such joint variations in response to unobserved latent variables. The observed variables are modelled as linear combinations of the potential factors, plus "error" terms. The factors typically are viewed as broad concepts or ideas that may describe an observed phenomenon. For example, a basic desire of obtaining a certain social level might explain most consumption behaviour. These unobserved factors are more interesting to the social scientist than the observed quantitative measurements.

Factor analysis is generally an exploratory/descriptive method that requires many subjective judgments. It is a widely used tool and often controversial because the models, methods, and subjectivity are so flexible that debates about interpretations can occur. The method is similar to principal components although, as the textbook points out, factor analysis is more elaborate. In one sense, factor analysis is an inversion of principal components. In factor analysis we model the observed variables as linear functions of the "factors." In principal components, we create new variables that are linear combinations of the observed variables. In both PCA and FA, the dimension of the data is reduced. Examples of fields where factor analysis is involved include physiology, health, intelligence, sociology, and sometimes ecology among others.

A common rationale behind factor analytic methods is that the information gained about the interdependencies between observed variables can be used later to reduce the set of variables in a dataset. It may help to deal with data sets where there are large numbers of observed variables that are thought to reflect a smaller number of underlying/latent variables. It is one of the most commonly used inter-dependency techniques and is used when the relevant set of variables shows a systematic inter-dependence and the objective is to find out the latent factors that create a commonality.

Notations and Terminology:

Collect all of the variables X's into a vector $\mathbf{X}$ for each individual subject. Let $\mathbf{X}_i$ denote observable trait i. These are the data from each subject, and are collected into a vector of traits.

$$\mathbf{X} = \left(X_1, X_2, \ldots, X_p\right)' = \text{vector of traits}$$

This is a random vector, with a population mean. Assume that vector of traits $\mathbf{X}$ is sampled from a population with population mean vector:

$\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_p)'$ = population mean vector

Here, $E(X_i)=\mu_i$ denotes the population mean of variable i.

Consider m unobservable common factors $f_1, f_2, \ldots, f_m$. The ith common factor is fi.

Generally, m is going to be substantially less than p.

The common factors are also collected into a vector,

$\mathbf{f} = (f_1, f_2, \ldots, f_p)'$ = vector of common factors

Model:

Our factor model can be thought of as a series of multiple regressions, predicting each of the observable variables X_i from the values of the unobservable common factors f_i:

$$X_1 = \mu_1 + l_{11}f_1 + l_{12}f_2 + \cdots + l_{1m}f_m + \epsilon_1$$

$$X_2 = \mu_2 + l_{21}f_1 + l_{22}f_2 + \cdots + l_{2m}f_m + \epsilon_2$$

$$\cdots$$

$$X_p = \mu_p + l_{p1}f_1 + l_{p2}f_2 + \cdots + l_{pm}f_m + \epsilon_p$$

Here, the variable means μ_1 through μ_p can be regarded as the intercept terms for the multiple regression models.

The regression coefficients l_{ij} (the partial slopes) for all of these multiple regressions are called factor loadings. Here, l_{ij} = *loading* of the ith variable on the jth factor. These are collected into a matrix as shown here:

$$\mathbf{L} = \begin{pmatrix} l_{11} & \cdots & l_{1m} \\ \vdots & \ddots & \vdots \\ l_{p1} & \cdots & l_{pm} \end{pmatrix}$$ = Matrix of factor loadings

And finally, the errors ε_i are called the specific factors. Here, ε_i = *specific factor* for variable i. The specific factors are also collected into a vector:

$\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_p)'$ = vector of specific factors

In summary, the basic model is like a regression model. Each of our response variables X is predicted as a linear function of the unobserved common factors f_1, f_2 through f_m. Thus, our explanatory variables are f_1, f_2 through f_m. We have m unobserved factors that control the variation among our data.

We will generally reduce this into matrix notation as shown in this form here:

$$\mathbf{X} = \boldsymbol{\mu} + \mathbf{Lf} + \boldsymbol{\epsilon}$$ (**Note!** In general, we want $m << p$.)

Model Assumptions:

Mean:

1. The specific factors or random errors all have mean zero: $E(\epsilon_i)=0$; $i = 1, 2, \ldots, p$
2. The common factors, the f's, also have mean zero: $E(f_i)=0$; $i = 1, 2, \ldots, m$
   A consequence of these assumptions is that the mean response of the ith trait is μ_i. *i.e.* $E(X_i)=\mu_i$

Variance:

1. The common factors have variance one: $var(f_i)=1$; $i = 1, 2, \ldots, m$
2. i is ψ_i: $var(\epsilon_i)=\psi_i$; $i = 1, 2, \ldots, p$ Here, ψ_i is called the *specific variance*

Correlations:

1. The common factors are uncorrelated with one another: $cov(f_i,f_j)=0$ for $i \neq j$
2. The specific factors are uncorrelated with one another: $cov(\epsilon_i,\epsilon_j)=0$ for $i \neq j$
3. The specific factors are uncorrelated with the common factors: $cov(\epsilon_i,f_j)=0$; $i = 1, 2, \ldots, p$; $j = 1, 2, \ldots, m$

These assumptions are necessary to uniquely estimate the parameters. An infinite number of equally well-fitting models with different values for the parameters may be obtained unless these assumptions are made.

Under this model the variance for the *i*th observed variable is equal to the sum of the squared loadings for that variable and specific variance:

The variance of trait *i* is: $\sigma_i^2 = var(X_i) = \sum_{j=1}^{m} l_{ij}^2 + \psi_i$

This derivation is based on the previous assumptions. $\sum_{j=1}^{m} l_{ij}^2$ is called the *Communality* for variable *i.* Later on, we will see how this is a measure for how well the model performs for that particular variable. The larger the communality, the better the model performance for the *i*th variable.

The covariance between pairs of traits *i* and *j* is: $\boldsymbol{\sigma_{ij}} = \boldsymbol{Cov}(\boldsymbol{X_i}, \boldsymbol{X_j}) = \sum_{k=1}^{m} \boldsymbol{l_{ik} l_{jk}}$

The covariance between trait *i* and factor *j* is: $cov(X_i,f_j)=l_{ij}$

In matrix notation, our model for the variance-covariance matrix is expressed as shown below:

$$\boldsymbol{\Sigma} = \mathbf{L}\mathbf{L}' + \boldsymbol{\psi}$$

This is the matrix of factor loadings times its transpose, plus a diagonal matrix containing the specific variances.

Here Ψ equals:

$$\boldsymbol{\psi} = \begin{pmatrix} \psi_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \psi_p \end{pmatrix}$$

A parsimonious (simplified) model for the variance-covariance matrix is obtained and used for estimation.

Note:

1. The model assumes that the data is a linear function of the common factors. However, because the common factors are not observable, we cannot check for linearity.
2. The variance-covariance matrix is a symmetric matrix, that is the variance between variables *i* and *j* is the same as the variance between *j* and *i*. For this model:

   $$\mathbf{\Sigma=LL'+\Psi}$$

   The variance-covariance matrix is going to have $p(p+1)/2$ unique elements of Σ approximated by:

   - *mp* factor loadings in the matrix **L**, and
   - *p* specific variances

   This means that there are *mp* plus *p* parameters in the variance-covariance matrix. Ideally, $mp + p$ is substantially smaller than $p(p+1)/2$. However, if *mp* is too small, the $mp + p$ parameters may not be adequate to describe Σ. There may always be the case that this is not the right model and you cannot reduce the data to a linear combination of factors.

3. If we have more than one variable in our analysis, that is if $p > 1$, the model is inherently ambiguous. To explain that, let **T** be any *m* x *m* *orthogonal matrix*. **A** matrix is orthogonal if its inverse is equal to the transpose of the original matrix.

   $$\mathbf{T'T=TT'=I}$$

   We can write our factor model in matrix notation:

   $$\mathbf{X=\mu+Lf+\epsilon=\mu+LTT'f+\epsilon=\mu+L^*f^*+\epsilon}$$

   Note that This does not change the calculation because the identity matrix times any matrix is the original matrix. This results in an alternative factor

model, where the relationship between the new factor loadings and the original factor loadings is:

$$\mathbf{L^*=LT}$$

and the relationship between the new common factors and the original common factors is:

$$\mathbf{f^*=T'f}$$

This gives a model that fits equally well. Moreover, because there is an infinite number of orthogonal matrices, then there is an infinite number of alternative models. This model, as it turns out, satisfies all of the assumptions discussed earlier.

Note...

$$E(\mathbf{f^*})=E(\mathbf{T'f})=\mathbf{T}'E(\mathbf{f})=\mathbf{T'0}=0,$$

$$var(\mathbf{f^*})=var(\mathbf{T'f})=\mathbf{T}'var(\mathbf{f})\mathbf{T}=\mathbf{T'IT}=\mathbf{T'T}=\mathbf{I}$$

and

$$cov(\mathbf{f^*,\epsilon})=cov(\mathbf{T'f,\epsilon})=\mathbf{T'}cov(\mathbf{f,\epsilon})=\mathbf{T'0}=0$$

So **f*** satisfies all of the assumptions, and hence **f*** is an equally valid collection of common factors. There is a certain apparent ambiguity to these models. This ambiguity is later used to justify a factor rotation to obtain a more parsimonious description of the data.

Methods to estimate the parameters of a factor model:

- Principal Component Model.
- Maximum Likelihood Method.
- Principal Factor Method.
- To understand the role of Factor Analysis, consider the following examples

Example 7: What underlying attitudes lead people to respond to the questions on a political survey as they do? Examining the correlations among the survey items reveals that there is significant overlap among various subgroups of items--questions about taxes tend to correlate with each other, questions about military issues correlate with each other, and so on. With factor analysis, you can investigate the number of underlying factors and, in many cases, you can identify what the factors represent conceptually. Additionally, you can compute factor scores for each respondent, which can then be used in subsequent analyses. For example, you might build a logistic regression model to predict voting behavior based on factor scores.

Example 8: A manufacturer of fabricating parts is interested in identifying the determinants of a successful salesperson. The manufacturer has on file the information shown in the following table. He is wondering whether he could reduce these seven variables to two or three factors, for a meaningful appreciation of the problem.

Data Matrix for Factor Analysis of seven variables (14 sales people)

Sales Person	Height (x_1)	Weight (x_2)	Education (x_3)	Age (x_4)	No. of Children (x_5)	Size of Household (x_6)	IQ (x_7)
1	67	155	12	27	0	2	102
2	69	175	11	35	3	6	92
3	71	170	14	32	1	3	111
4	70	160	16	25	0	1	115
5	72	180	12	30	2	4	108
6	69	170	11	41	3	5	90
7	74	195	13	36	1	2	114
8	68	160	16	32	1	3	118
9	70	175	12	45	4	6	121
10	71	180	13	24	0	2	92
11	66	145	10	39	2	4	100
12	75	210	16	26	0	1	109
13	70	160	12	31	0	3	102
14	71	175	13	43	3	5	112

Can we now collapse the seven variables into three factors? Intuition might suggest the presence of three primary factors: maturity revealed in age/children/size of household, physical size as shown by height and weight, and intelligence or training as revealed by education and IQ.

The sales people data have been analyzed by the SAS program. This program accepts data in the original units, automatically transforming them into standard scores. The three factors derived from the sales people data by principal component analysis (SAS program) are presented below:

Three-factor results with seven variables sales people characteristics

Variable	Factor I	Factor II	Factor III	Communality
Height	0.59038	0.72170	-0.30331	0.96140 (sumsq I,II and III)
Weight	0.45256	0.75932	-0.44273	0.97738
Education	0.80252	0.18513	0.42631	0.86006
Age	-0.86689	0.41116	0.18733	0.95564
No. of Children	-0.84930	0.49247	0.05883	0.96730
Size of Household	-0.92582	0.30007	-0.01953	0.94756
IQ	0.28761	0.46696	0.80524	0.94918
Sum of squares	3.61007	1.85136	1.15709	
Variance summarized	0.51572	0.26448	0.16530	Average=0.94550

Factor Loadings

The coefficients in the factor equations are called "factor loadings". They appear above in each factor column, corresponding to each variable. The equations are:

$$\mathbf{F}_1 = 0.59038\ x_1 + 0.45256\ x_2 + 0.80252\ x_3 - 0.86689\ x_4 - 0.84930\ x_5 - 0.92582\ x_6 + 0.28761 x_7$$

$$\mathbf{F}_2 = 0.72170 x_1 + 0.75932 x_2 + 0.18513 x_3 + 0.41116 x_4 + 0.49247 x_5 + 0.30007 x_6 + 0.46696 x_7$$

$$\mathbf{F}_3 = -0.30331 x_1 - 0.44273 x_2 + 0.80252 x_3 + 0.18733 x_4 + 0.58830 x_5 - 0.01953 x_6 + 0.80524 x_7$$

The factor loadings depict the relative importance of each variable with respect to a particular factor. In all the three equations, education (x_3) and IQ (x_7) have got positive loading factor indicating that they are variables of importance in determining the success of sales person.

Variance summarized

Factor analysis employs the criterion of maximum reduction of variance - variance found in the initial set of variables. Each factor contributes to reduction. In our example Factor I accounts for 51.6% of the total variance. Factor II for 26.4% and Factor III for 16.5%. Together the three factors "explain" almost 95% of the variance.

Communality

In the ideal solution the factors derived will explain 100% of the variance in each of the original variables, "Communality" measures the percentage of the variance in the original variables that is captured by the combinations of factors in the solution. Thus communality is computed for each of the original variables. Each variables communality might be thought of as showing the extent to which it is revealed by the system of factors. In our example the communality is over 85% for every variable. Thus the three factors seem to capture the underlying dimensions involved in these variables.

There is yet another analysis called varimax rotation, after we get the initial results. This could be employed if needed by the analyst. We do not intend to dwell on this and those who want to go into this aspect can use SAS program for varimax rotation.

Factor Analysis in SAS

The model for common factor analysis posits one unique factor for each observed variable. The PROC FACTOR can be used for several types of common factor and component analysis. Both orthogonal and oblique rotations are available. We can compute scoring coefficients by the regression method. All major statistics computed by PROC FACTOR can also be saved in an output DATA SET. The PROC FACTOR can be invoked by the following statements:

```
PROC FACTOR <options>;
VAR variables;
PRIORS
Communalities;
PARTIAL Variables;
FREQ Variable;
WEIGHT Variable;
BY
variables;
RUN;
```

Usually only the VAR statement is needed in addition to the PROC FACTOR statement. The some of the important options available with PROC FACTOR are:

METHOD=NAME : specifies the method of extracting factors. The default option is METHOD = PRINCIPAL, which yields principal component analysis if no PRIORS is used or if PRIORS = ONE is specified; if a PRIORS = value other than one is specified, a principal factor anlaysis is performed.

METHOD= PRINT : yields iterated principal factor analysis.
METHOD=ML : performs maximum- likelihood factor analysis.
METHOD = ALPHA : produced alpha factor analysis.
METHOD =ULS: produced unweighted least squares factor analysis.

NFACTORS=n | NFACT=n | N=n specifies the maximum number of factors to be extracted.

PRIORS =name: (ASMC | INPUT | MAX | ONE | RANDOM | SMC): specifies a method for computing prior communality estimates

ROTATE=name: gives the rotation method. The default is ROTATE=NONE. FACTOR performs the following orthogonal rotation methods:

- EQUAMAX
- ORTHOMAX
- QUARTIMAX
- PARSIMAX
- VARIMAX

After the initial factor extraction, the common factors are uncorrelated with each other. If the factors are rotated by an orthogonal transformation, the rotated factors are uncorrelated. If the factors are rotated by an oblique transformation, the rotated factors become correlated. Oblique rotations often produce more useful patterns than do orthogonal rotations. However, a consequence of correlated factors is that there is no single unambiguous measure of the importance of a factor in explaining a variable. Thus, for oblique rotations, the pattern matrix doesn't provide all the necessary information for interpreting the factors.

References:

Bryant, F. B., & Yarnold, P. R. (1995). Principal components analysis and exploratory and confirmatory factor analysis. In L. G. Grimm & P. R. Yarnold (Eds.), *Reading and understanding multivariate analysis*. Washington, DC: American Psychological Association.

Dunteman, G. H. (1989). *Principal components analysis*. Newbury Park, CA: Sage Publications.

Fabrigar, L. R., Wegener, D. T., MacCallum, R. C., & Strahan, E. J. (1999). Evaluating the use of exploratory factor analysis in psychological research. *Psychological Methods*, *4*(3), 272-299.

Gorsuch, R. L. (1983). *Factor Analysis*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Hair, J. F., Jr., Anderson, R. E., Tatham, R. L., & Black, W. C. (1995). *Multivariate data analysis with readings* (4th ed.). Upper Saddle River, NJ: Prentice-Hall.

Hatcher, L. (1994). *A step-by-step approach to using the SAS system for factor analysis and structural equation modeling*. Cary, NC: SAS Institute.

Hutcheson, G., & Sofroniou, N. (1999). *The multivariate social scientist: Introductory statistics using generalized linear models*. Thousand Oaks, CA: Sage Publications.

Kim, J. -O., & Mueller, C. W. (1978a). *Introduction to factor analysis: What it is and how to do it*. Newbury Park, CA: Sage Publications.

Kim, J. -O., & Mueller, C. W. (1978b). *Factor Analysis: Statistical methods and practical issues*. Newbury Park, CA: Sage Publications.

Lawley, D. N., & Maxwell, A. E. (1962). Factor analysis as a statistical method. *The Statistician, 12*(3), 209-229.

Levine, M. S. (1977). *Canonical analysis and factor comparison*. Newbury Park, CA: Sage Publications.

Pett, M. A., Lackey, N. R., & Sullivan, J. J. (2003). *Making sense of factor analysis: The use of factor analysis for instrument development in health care research*. Thousand Oaks, CA: Sage Publications.

Shapiro, S. E., Lasarev, M. R., & McCauley, L. (2002). Factor analysis of Gulf War illness: What does it add to our understanding of possible health effects of deployment, *American Journal of Epidemiology, 156*, 578-585.

Velicer, W. F., Eaton, C. A., & Fava, J. L. (2000). Construct explication through factor or component analysis: A review and evaluation of alternative procedures for determining the number of factors or components. In R. D. Goffin & E. Helmes (Eds.), *Problems and solutions in human assessment: Honoring Douglas Jackson at seventy.* Boston, MA: Kluwer.

Widaman, K. F. (1993). Common factor analysis versus principal component analysis: Differential bias in representing model parameters, *Multivariate Behavioral Research, 28*, 263-311.

CLUSTER ANALYSIS

ARPAN BHOWMIK
ICAR-IASRI, Library Avenue, New Delhi – 110 012
Arpan.bhowmik@icar.gov.in

1. Introduction

Cluster analysis is usually done in an attempt to combine cases into groups when the group membership is not known prior to the analysis. Cluster analysis is a technique for grouping individual or objects into unknown groups. It differs from other methods of classification such as Discriminant analysis, in that in cluster analysis the number and characteristics of the groups are to be derived from the data and are not usually known prior to the analysis.

In biology, cluster analysis has been used for decades in the area of taxonomy, where living things are classified into arbitrary groups on the basis of their characteristics group. The classification proceeds from the most general to the most specific in steps. The most general classification is kingdom followed by phylum, subphylum, and class etc. Cluster analysis has been used in medicine to assign patient to specific diagnostic categories on the basis of their presenting symptoms and signs. Cluster analysis is also an important tool for investigation in data mining. For example consumers can be clustered on the basis of their purchases in marketing research. Here the emphasis may be on the methods that can be used for large data sets. In short it is possible to find application of cluster analysis in virtually any field of research. It is also possible to cluster the variables rather than the cases. Clustering of variables is sometimes used in analyzing the items in a scale to determine which items tends to be close together in terms of individual response to them.

There are 4 basic steps to conduct cluster analysis for any experiments. Those are given below:

1. Select a distance measure.
2. Select a clustering algorithm.
3. Determine the number of clusters.
4. Validate the analysis.

2. Clustering Methods (Johnson and Wichern, 2006)

The commonly used methods of clustering fall into two general categories.

(i) Hierarchical and
(ii) Non hierarchical.

2.1 Hierarchical cluster Analysis:

Hierarchical clustering techniques proceed by either a series of mergers or a series of successive divisions. Agglomerative hierarchical method starts with the individual objects, thus there are as many clusters as objects. The most similar objects are first

grouped and these initial groups are merged according to their similarities. Eventually, as the similarity decreases, all subgroups are fused into a single cluster.

Divisive hierarchical methods work in the opposite direction. An initial single group of objects is divided into two sub groups such that the objects in one sub group are far from the objects in the others. These subgroups are then further divided into dissimilar subgroups. The process continues until there are as many subgroups as objects i.e., until each object form a group. The results of both agglomerative and divisive method may be displayed in the form of a two dimensional diagram known as Dendrogram. It can be seen that the Dendrogram illustrate the mergers or divisions that have been made at successive levels.

Linkage methods are suitable for clustering items, as well as variables. This is not true for all hierarchical agglomerative procedure. The following types of linkage are now discussed:

(i) Single linkage (minimum distance or nearest neighbour),
(ii) Complete linkage (maximum distance or farthest neighbour) and
(iii) Average linkage (average distances).

Also other methods of hierarchical clustering techniques like Ward's method and Centroid method are available in the literature.

2.1.1 Steps of Agglomeration in Hierarchical Cluster analysis:

The following are the steps in the agglomerative hierarchical clustering algorithm for groups of N objects (items or variables).

i. Start with N clusters, each containing a single entity and an N×N symmetric matrix of distance (or similarities) $\mathbf{D} = \{d_{ik}\}$.
ii. Search the distance matrix for the nearest (most similar) pair of clusters. Let the distance between most similar clusters U and V be d_{uv}.
iii. Merge clusters U and V. Label the newly formed cluster (UV). Update the entries in the distance matrix by (a) deleting the rows and columns corresponding to clusters U and V and (b) adding a row and column giving the distances between cluster (UV) and the remaining clusters.
iv. Repeat steps (ii) and (iii) a total of N-1 times (All objects will be in a single cluster after the algorithm terminates). Record the identity of clusters that are merged and the levels (distances or similarities) at which the mergers take place.

The basic ideas behind the cluster analysis are now shown by presenting the algorithm components of linkage methods.

2.2 Non Hierarchical Clustering Method

Non-hierarchical clustering techniques are designed to group items, rather than variables, into a collection of K clusters. The number of clusters, K, may either be specified in advance or determined as part of the clustering procedure. Because a matrix of distance does not have to be determined and the basic data do not have to be stored during the computer run. Non hierarchical methods can be applied to much larger data sets than can hierarchical techniques. Non hierarchical methods start from either (1) an initial partition

of items into groups or (2) an initial set of seed points which will form nuclei of the cluster.

2.2.1 K means Clustering (Afifi, Clark and Marg, 2004)

The K means clustering is a popular non hierarchical clustering technique. For a specified number of clusters K the basic algorithm proceeds in the following steps.

i. Divide the data into K initial cluster. The number of these clusters may be specified by the user or may be selected by the program according to an arbitrary procedure.

ii. Calculate the means or centroid of the K clusters.

iii. For a given case, calculate its distance to each centroid. If the case is closest to the centroid of its own cluster, leave it in that cluster; otherwise, reassign it to the cluster whose centroid is closest to it.

iv. Repeat step (iii) for each case.

v. Repeat steps (ii), (iii), and (iv) until no cases are reassigned.

3. Dendrogram

Dendrogram is also called hierarchical tree diagram or plot, and shows the relative size of the proximity coefficients at which cases are combined. The bigger the distance coefficient or the smaller the similarity coefficient, the more clustering involved combining unlike entities, which may be undesirable. Trees are usually depicted horizontally, not vertically, with each row representing a case on the Y axis, while the X axis is a rescaled version of the proximity coefficients. Cases with low distance/high similarity are close together. Cases showing low distance are close, with a line linking them a short distance from the left of the Dendrogram, indicating that they are agglomerated into a cluster at a low distance coefficient, indicating alikeness. When, on the other hand, the linking line is to the right of the Dendrogram the linkage occurs at a high distance coefficient, indicating the cases/clusters were agglomerated even though much less alike. If a similarity measure is used rather than a distance measure, the rescaling of the X axis still produces a diagram with linkages involving high alikeness to the left and low alikeness to the right.

4. Distance Measures

Given two objects X and Y in a ‘p’ dimensional space, a dissimilarity measure satisfies the following conditions:

1. $d(X,Y) \geq 0$ for all objects X and Y.
2. $d(X,Y) = 0$ iff $X = Y$.
3. $d(X,Y) = d(Y,X)$.

Condition (3) implies that the measure is symmetric so that the dissimilarity measure that compares X and Y is same as the comparison for object Y verses X. Condition (2)

requires the measures to be zero, when ever object X equals to object Y. The objects are identical if d(X, Y) = 0. Finally, Condition (1) implies that the measure is never negative. Some dissimilarity measures are as follows.

4.1 Euclidian Distance

This is probably the most commonly chosen type of distance. It simply is the geometric distance in the multidimensional space. It is computed as,

$$d(X,Y)= \{\sum_{i=1}^{p}(X_i - Y_i)^2\}^{1/2} \quad \text{or}$$

in matrix form

$$d(X,Y)= \sqrt{(X-Y)'(X-Y)}$$

where $\mathbf{X}' = (X_1, X_2, \ldots, X_p)$

$\mathbf{Y}' = (Y_1, Y_2, \ldots, Y_p)$

The statistical distance between the same two observations is of the form

$$d(X,Y) = \sqrt{(X-Y)'A(X-Y)},$$

where $\mathbf{A} = \mathbf{S}^{-1}$ and S contains the sample variances and covariances.

Euclidian and square Euclidian distances are usually computed from raw data and not from standardized data.

4.2 Square Euclidean Distance

Square the standard Euclidean distance in order to place progressively greater weight on objects that are further apart. This distance is computed as:

$$d^2(X,Y) = \sum_{i=1}^{p}(X_i - Y_i)^2$$

or in matrix form

$$d^2(X,Y) = (\mathbf{X} - \mathbf{Y})' (\mathbf{X} - \mathbf{Y})$$

4.3 Minkowski Metric

When there is no idea about prior knowledge of the distance group then one goes for minkowski metric. This can be computed as given below:

$$d(X,Y) = \{\sum_{i=1}^{p}|X_i - Y_i|^m\}^{\frac{1}{m}}$$

For m = 1, d(X,Y) measures the city block distance between two points in p dimensions. For m = 2, d(X,Y) becomes the Euclidean distance. In general, varying m changes the weight given to larger and smaller differences.

4.4 City-Block (Manhattan) Distance

This distance is simply the average difference across dimensions. In most cases, this distance measure yields result similar to the simple Euclidean distance. This can be computed as :

$$d(X,Y) = \sum_{i=1}^{p} |X_i - Y_i|$$

4.5 Chebychev Distance

This distance measure may be appropriate in case when we want to define the objects as different if they are different on any one of the dimensions. The chebychev distance is computed as:

$$d(X,Y) = \text{maximum} |X_i - Y_i|$$

Two additional popular measures of distance or dissimilarity are given by the Canberra metric and the Czekanowski coefficient. Both of these measures are defined for non negative variables only. We have

Canberra Metric: $d(X, Y) = \sum_{i=1}^{p} \frac{|X_i - Y_i|}{(X_i + Y_i)}$

Czekanowski Coefficient = $1 - \frac{2\sum_{i=1}^{p} \min(X_i, Y_i)}{\sum_{i=1}^{p} (X_i - Y_i)}$

5. Illustration (Chatfield and Collins, 1990)

Given below is food nutrient data on calories, protein, fat, calcium and iron. The objective of the study is to identify suitable clusters of food nutrient data based on the five variables**.**

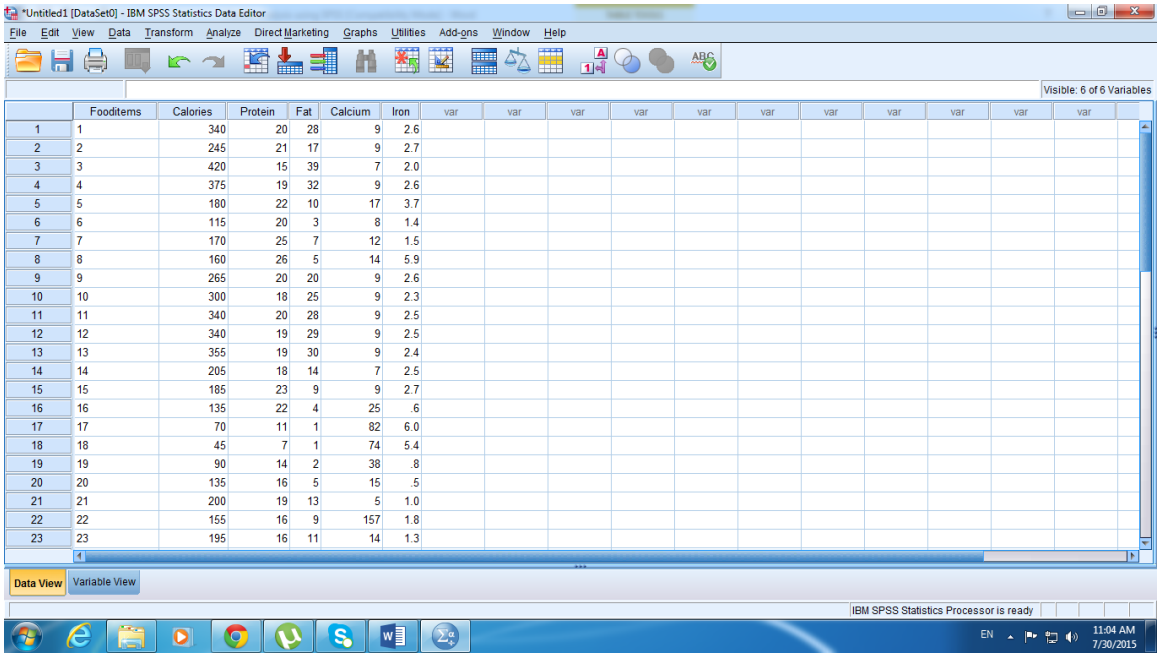
Food Items	Calories	Protein	Fat	Calcium	Iron
1	340	20	28	9	2.6
2	245	21	17	9	2.7
3	420	15	39	7	2
4	375	19	32	9	2.6
5	180	22	10	17	3.7
6	115	20	3	8	1.4
7	170	25	7	12	1.5
8	160	26	5	14	5.9
9	265	20	20	9	2.6
10	300	18	25	9	2.3
11	340	20	28	9	2.5
12	340	19	29	9	2.5

13	355	19	30	9	2.4
14	205	18	14	7	2.5
15	185	23	9	9	2.7
16	135	22	4	25	0.6
17	70	11	1	82	6
18	45	7	1	74	5.4
19	90	14	2	38	0.8
20	135	16	5	15	0.5
21	200	19	13	5	1
22	155	16	9	157	1.8
23	195	16	11	14	1.3
24	120	17	5	159	0.7
25	180	22	9	367	2.5
26	170	25	7	7	1.2
27	170	23	1	98	2.6

4.1 Analysis using SPSS:

Start by entering the datasheet into SPSS using the steps below.
Step: Go to file→ open→ browse the datasheet→ click open or

Enter all the data in the data editor as shown in Figure1.



	Fooditems	Calories	Protein	Fat	Calcium	Iron
1	1	340	20	28	9	2.6
2	2	245	21	17	9	2.7
3	3	420	15	39	7	2.0
4	4	375	19	32	9	2.6
5	5	180	22	10	17	3.7
6	6	115	20	3	8	1.4
7	7	170	25	7	12	1.5
8	8	160	26	5	14	5.9
9	9	265	20	20	9	2.6
10	10	300	18	25	9	2.3
11	11	340	20	28	9	2.5
12	12	340	19	29	9	2.5
13	13	355	19	30	9	2.4
14	14	205	18	14	7	2.5
15	15	185	23	9	9	2.7
16	16	135	22	4	25	.6
17	17	70	11	1	82	6.0
18	18	45	7	1	74	5.4
19	19	90	14	2	38	.8
20	20	135	16	5	15	.5
21	21	200	19	13	5	1.0
22	22	155	16	9	157	1.8
23	23	195	16	11	14	1.3

Fig.1: Screen shot after entering the data in data editor

Now click Analyze→ Classify→ Hierarchical Cluster as shown in Figure2.

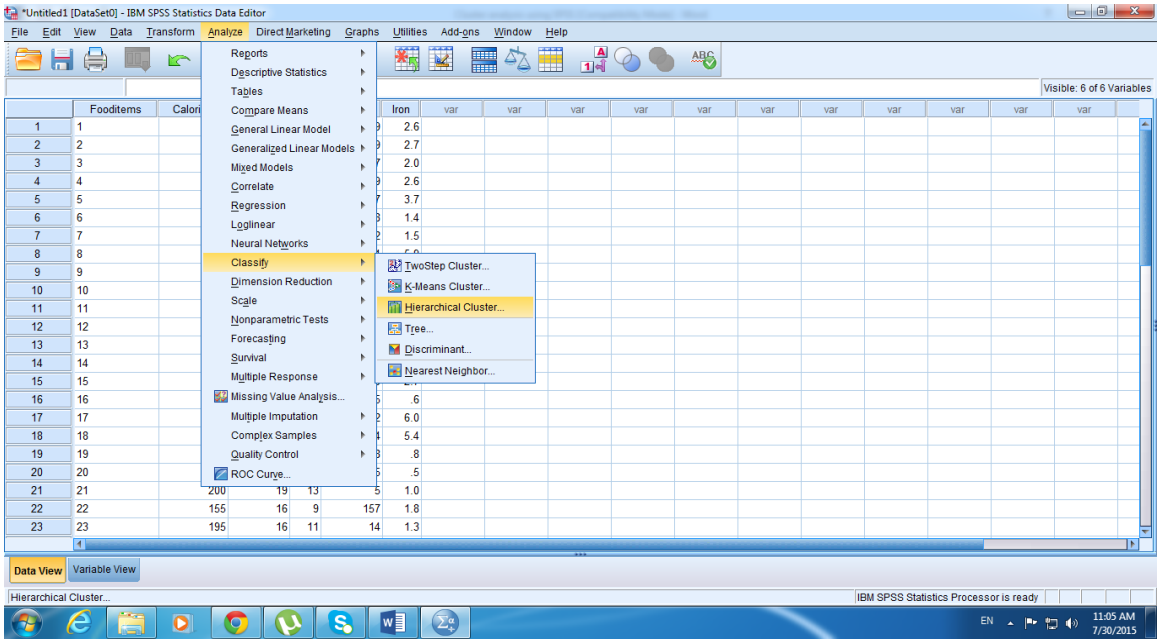


Fig.2: Screen shot of selecting the analysis procedure

Then Identify Name as the variable by which to label cases and Calories, Protein, Fat, Calcium, and Iron as the variables. Indicate that you want to cluster cases rather than variables and want to display both statistics and plots as shown in Figure3.

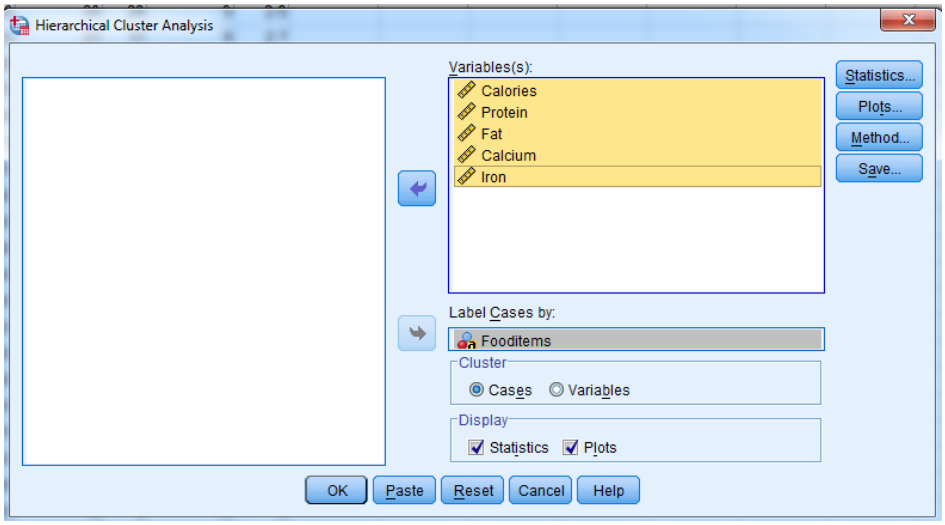


Fig.3: Cluster cases rather than variables and want to display both statistics and plots.

Click Statistics and indicate that you want to see an Agglomeration schedule with 2, 3, 4, and 5 cluster solutions. Click Continue as shown in Figure4

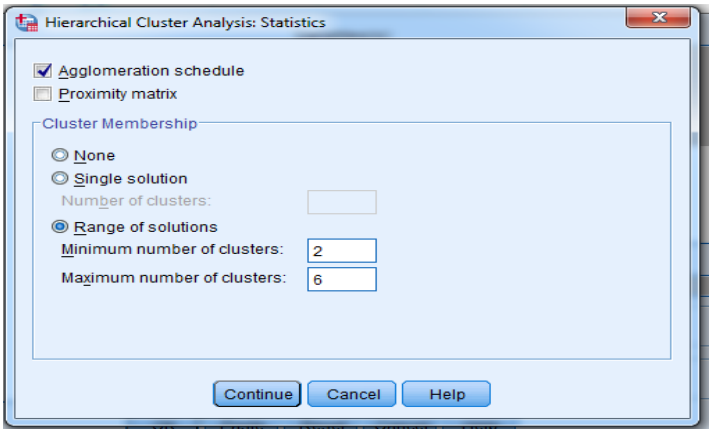


Fig.4: Hierarchical cluster analysis statistics

Click plots and indicate that you want a Dendogram and a verticle Icicle plot with 2, 3, and 4 cluster solutions. Click Continue as shown in Figure5

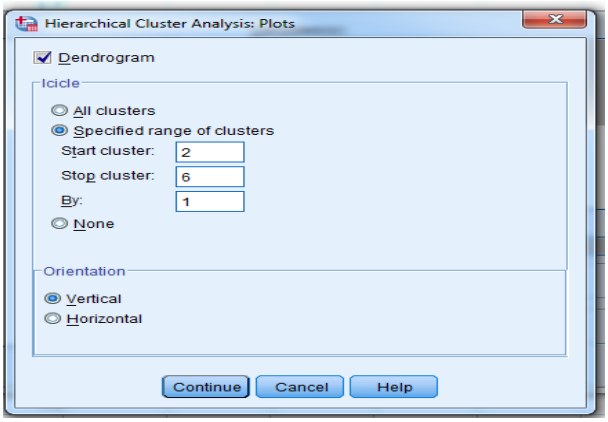


Fig.5: Hierarchical cluster analysis plot

Click Method and indicate that you want to use the Between-groups linkage method of clustering, squared Euclidian distances, and variables standardized to *z* scores (so each variable contributes equally). Click Continue as shown in Figure6.

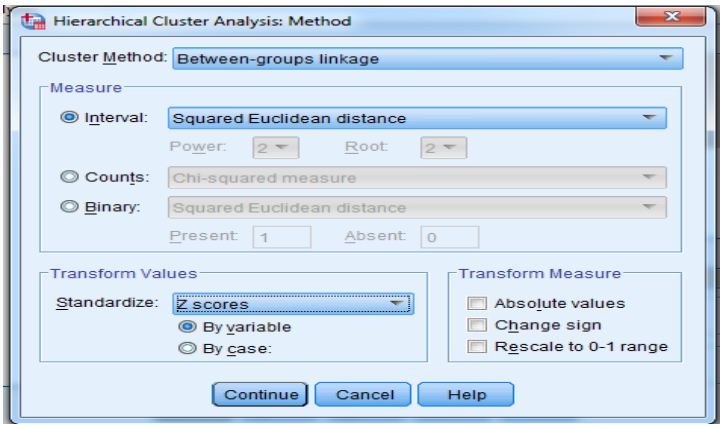
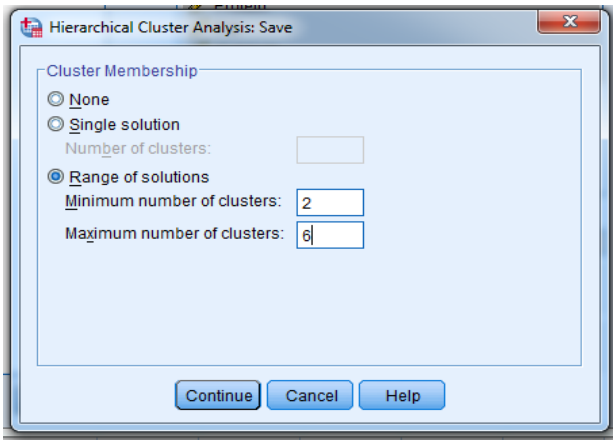


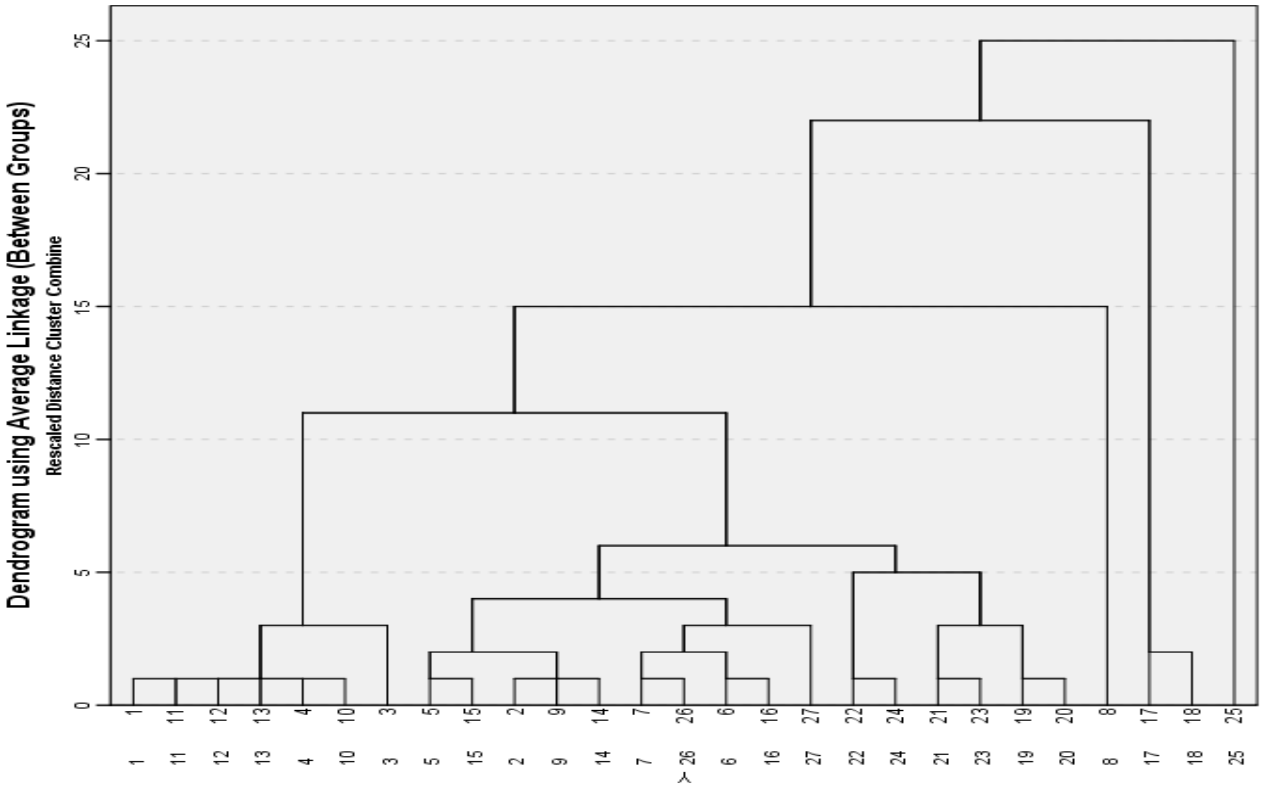
Fig.6: Hierarchical cluster analysis method

Click Save and indicate that you want to save, for each case, the cluster to which the case is assigned for 2, 3, 4, 5 and 6 cluster solutions. Click Continue, OK as shown in figure7



SPSS starts by standardizing all of the variables to mean 0, variance 1. This results in all the variables being on the same scale and being equally weighted.

Dendrogram:



4.2 Interpretation

The main objective of our analysis is to grouping the food items on the basis of their nutrient content based on the five variables such that food items with in the groups are homogeneous and between the groups are heterogeneous.

Number of groups	Food items
Two groups	Group-1 (1,11,12,…,18) Group-2 (25)
Three groups	Group-1 (1,11,…,8) Group-2 (17,18) Group-3 (25)
Four groups	Group-1 (1,11,…,20) Group-2 (8) Group-3 (17,18) Group-4 (25)
Five groups	Group-1 (1,11,…,3) Group-2 (5,15,…,20) Group-3 (8) Group-4 (17,18) Group-5 (25)
Six groups	Group-1 (1,11,…,3) Group-2 (5,15,…,27) Group-3 (22,24,…20) Group-4 (8) Group-5 (17,18) Group-6 (25)

4.3 Analysis using SAS

SAS codes used for performing the cluster analysis using unweighted pair-group method using arithmetic averages (UPGMA) method are as follows:

```
/* The following statement prepare data file for data entry.*/

DATA cluster;
input items $ Calories Protein Fat Calcium Iron;
cards;

/* This statement uses the DISTANCE procedure to obtain a distance
matrix that will be used as input to a subsequent clustering
procedure.An output SAS data set called Distmat that contains the
distance  matrix is created through the OUT= option.METHOD= Euclid
requests that Euclidean (which also is the default) distances should be
computed. */

PROC distance data=cluster method=EUCLID  out=distmat;
      var interval(Calories Protein Fat Calcium Iron);
      id items;run;

  /*The ID statement specifies that the variable Strain should be
copied to the OUT= data set and used to  generate names for the
distance variables. The distance variables in the output data set are
```

```
named by the values in the ID variable. This statement prints the
Euclidean distances between pairs of treatments*/

   PROC print data=distmat;
      id items;run;

 /*The PROC CLUSTER statement starts the CLUSTER procedure, identifies
a clustering method. The METHOD= specification determines the
clustering method used by the procedure.Here we have used method =
AVERAGE (unweighted pair-group method using arithmetic averages,
UPGMA). The output data set Tree is created (through outtree= Tree)
and used as input to the TREE procedure that produces the dendrogram
.*/

   PROC cluster data=distmat method=average outtree=tree;
     id items;
   run;

 /* The following statements use the TREE procedure to produce a
dendrogram of the clusters.The preceding statements use the SAS data
set Tree as input. The OUT= option creates an output  SAS data set
named out to contain information on cluster membership. The NCLUSTERS=
option specifies the number of clusters desired in the data set
out(here we have taken 3 clusters).*/

goptions   htext=2pct ;
PROC tree data=Tree nclusters=6 horizontal hordisplay=left
lines=(color=blue) out=out;
id items;
run;

/* The following statement use the SORT procedure to sort the data set
out*/
   PROC sort data=out;
      by items; run;

 /* The following statement use the SORT procedure to sort the data set
cluster*/

   PROC sort data = cluster;
   by items; run;

    /* The following statement merges the two data sets cluster and
out*/

   DATA clus;
      merge cluster out;
      by items;run;

    /* The following statement use the SORT procedure to sort the data
set clus*/

   PROC sort data=clus;
      by cluster;run;

    /* The following statement use the PRINT procedure to print the
clusters*/
```

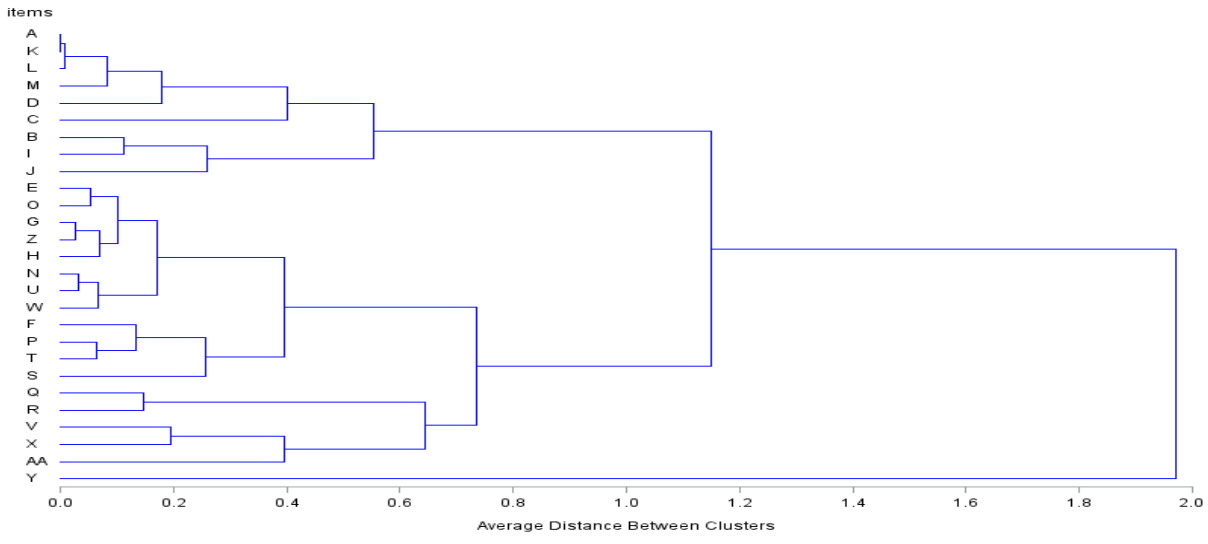
```
PROC print;
   id items;
   by cluster; run;
```

Output

The SAS System

Items	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
A	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
B	95.640	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
C	80.934	176.492	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
D	35.242	130.878	45.764	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
E	161.224	65.879	242.060	196.426	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
F	226.390	130.766	307.160	261.619	66.063	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
G	171.395	75.838	252.288	206.631	12.159	55.516	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
H	181.661	86.193	262.567	216.885	21.327	46.057	11.330	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
I	75.425	20.249	156.254	110.657	85.990	150.968	96.069	106.404	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
J	40.163	55.661	120.868	75.333	121.272	186.319	131.463	141.781	35.413	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
K	0.100	95.640	80.934	35.242	161.225	226.390	171.394	181.663	75.426	40.163	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
L	1.418	95.776	80.748	35.128	161.355	226.504	171.552	181.828	75.545	40.212	1.414	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
M	15.167	110.784	65.774	20.101	176.351	241.520	186.547	196.815	90.560	55.236	15.166	15.034	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.	.
N	135.753	40.275	216.470	170.965	27.540	90.704	36.729	47.229	60.366	95.656	135.753	135.849	150.868	0.000	.	.	.	.	.	.	.	.	.	.	.	.	.
O	156.189	60.564	237.052	191.429	9.592	70.340	15.603	26.177	80.808	116.216	156.189	156.336	171.339	21.308	0.000	.	.	.	.	.	.	.	.	.	.	.	.
P	207.039	111.939	287.793	242.184	46.202	26.356	37.587	28.126	131.985	167.155	207.038	207.168	222.136	73.100	52.786	0.000	.	.	.	.	.	.	.	.	.	.	.
Q	281.159	190.580	360.001	315.263	128.578	87.219	123.094	113.864	209.303	242.627	281.161	281.228	295.757	155.178	137.014	87.368	0.000	.	.	.	.	.	.	.	.	.	.
R	303.572	211.387	382.928	337.991	147.590	97.185	140.869	131.157	230.571	264.493	303.573	303.622	318.306	174.320	155.410	103.721	26.559	0.000	.	.	.	.	.	.	.	.	.
S	253.093	158.567	333.515	288.087	93.153	39.527	84.985	75.200	178.408	213.280	253.093	253.176	268.100	119.787	99.998	47.561	48.714	58.242	0.000	.	.	.	.	.	.	.	.
T	206.423	110.950	287.138	241.616	45.828	21.675	36.332	27.480	131.078	166.338	206.422	206.519	221.526	71.084	51.047	11.705	93.730	108.176	50.666	0.000	.	.	.	.	.	.	.
U	140.871	45.430	221.578	176.081	23.860	85.646	31.957	42.638	65.525	100.810	140.870	140.976	155.987	5.766	16.610	68.667	151.862	170.568	115.477	66.319	0.000	.	.	.	.	.	.
V	237.711	173.476	305.984	266.163	142.357	154.445	146.065	143.551	184.775	207.820	237.711	237.778	249.709	158.207	151.175	133.740	113.827	138.373	135.794	143.463	158.602	0.000	.	.	.	.	.
W	146.139	50.872	226.847	181.319	16.636	80.722	26.945	37.177	70.878	106.071	146.138	146.234	161.233	12.784	13.415	61.697	142.815	162.166	108.103	60.313	10.913	148.503	0.000	.	.	.	.
X	267.286	195.676	338.032	297.089	154.347	151.127	155.492	150.775	209.195	235.167	267.285	267.363	279.923	174.397	163.649	134.933	92.245	113.965	124.736	144.783	173.736	35.316	163.363	0.000	.	.	.
Y	392.593	363.942	433.762	408.322	350.003	364.893	355.161	353.628	368.122	377.937	392.593	392.649	399.048	360.924	358.036	344.989	305.814	323.066	341.258	354.944	362.590	211.569	353.377	216.583	0.000	.	.
Z	171.382	75.811	252.239	206.620	14.975	55.381	5.009	13.270	96.047	131.447	171.382	171.539	186.535	36.396	15.468	39.590	126.016	143.149	86.644	37.074	31.242	151.031	27.767	160.226	360.159	0.0000	.
AA	193.801	117.499	268.867	225.661	82.123	105.544	86.239	84.805	131.590	159.443	193.801	193.961	207.372	98.489	90.609	81.043	102.037	128.315	100.425	90.462	98.547	61.803	88.497	79.225	269.307	91.2303	0

The TREE Procedure
Average Linkage Cluster Analysis



CLUSTER=1

items	Calories	Protein	Fat	Calcium	Iron	CLUSNAME
A	340	20	28	9	2.6	CL6
C	420	15	39	7	2.0	CL6
D	375	19	32	9	2.6	CL6
K	340	20	28	9	2.5	CL6
L	340	19	29	9	2.5	CL6
M	355	19	30	9	2.4	CL6

CLUSTER=2

items	Calories	Protein	Fat	Calcium	Iron	CLUSNAME
E	180	22	10	17	3.7	CL7
F	115	20	3	8	1.4	CL7
G	170	25	7	12	1.5	CL7
H	160	26	5	14	5.9	CL7
N	205	18	14	7	2.5	CL7
O	185	23	9	9	2.7	CL7
P	135	22	4	25	0.6	CL7
S	90	14	2	38	0.8	CL7
T	135	16	5	15	0.5	CL7
U	200	19	13	5	1.0	CL7
W	195	16	11	14	1.3	CL7
Z	170	25	7	7	1.2	CL7

CLUSTER=3

items	Calories	Protein	Fat	Calcium	Iron	CLUSNAME
B	245	21	17	9	2.7	CL9
I	265	20	20	9	2.6	CL9
J	300	18	25	9	2.3	CL9

CLUSTER=4

items	Calories	Protein	Fat	Calcium	Iron	CLUSNAME
Q	70	11	1	82	6.0	CL14
R	45	7	1	74	5.4	CL14

CLUSTER=5

items	Calories	Protein	Fat	Calcium	Iron	CLUSNAME
AA	170	23	1	98	2.6	CL8
V	155	16	9	157	1.8	CL8
X	120	17	5	159	0.7	CL8

CLUSTER=6

items	Calories	Protein	Fat	Calcium	Iron	CLUSNAME
Y	180	22	9	367	2.5	Y

References:

Afifi, A., Clark, V. A. and Marg, S. (2004). *Computer Aided Multivariate Analysis*. USA, Chapman & Hall.

Chatfield, C. and Collins, A. J. (1990). *Introduction to Multivariate Analysis*. Chapman and Hall Publications.

Hair, J. F., Anderson R. E., Tatham, R. L. and Black, W. C. (2006). *Multivariate Data Analysis.* 5th Edn., Pearson Education Inc.

Johnson, R. A. and Wichern, D. W. (2006). *Applied Multivariate Statistical Analysis*. 5th Edn., London, Inc. Pearson Prentice Hall.

Timm, N. H. (2002). *Applied Multivariate Analysis*. 2nd Edn. , New York, Springer-Verlag.

STABILITY ANALYSIS: PART I

A.R. RAO
ICAR-IASRI, Library Avenue, New Delhi – 110 012
ar.rao@icar.gov.in

1. Introduction

In various research programmes it is quite common to carry out the same experiment at a number of different places, on a number of seasons or years. Some times the purpose of the research is to produce recommendations which are to apply to a population that is extensive either in space or time or both. In agricultural and animal field experimentation, many projects are undertaken in the hope that their results can be applied in practical farming. It has been found that the effectiveness of different varieties of a crop usually varies from location to location and, even more markedly, from season to season. A single experiment will precisely furnish information about only one place or one season in which the experiment is conducted. It has, thus, become a common practice to repeat an experiment at different places or over a number of years to obtain valid recommendations taking into account place to place variation or variation over time or both. In such cases of repeated experiments appropriate statistical procedures for a combined analysis of the data would have to be followed.

In a pooled analysis, the main points of interest would be (i) to estimate the average response of varieties and (ii) to test the consistency of the responses from place to place or occasion to occasion *i.e.* the interaction of the varietal effects with the places or years. The average response of varieties, thus, depend largely upon the absence or presence of Variety × Place or Genotype × Environment (GE) interaction (interplay of genetic and non-genetic effects on the phenotypic expression), coupled with high yield indicate that the genotypes are suitable for general adaptation in the range of environments considered. But this ideal situation is rarely found because the phenotypic stability of a genotype is inversely proportional to the mean yield. The varieties with high stability are generally low yielders and *vice versa.* A balance between these extremes, is therefore, set as goal of any breeding programmes for crop or animal improvement. GE interaction causes difficulty in demonstrating the significant superiority of any variety / breed when varieties / breeds are compared over a series of environments. Hence, methods are in need to be investigated for reducing GE interaction. One such method would be to select stable genotypes that interact less with the environment in which they are grown. Then our preliminary evaluation is to identify the stable genotypes. However, selection for stability is not possible until a biometrical model with suitable parameters is available to provide criteria necessary to rank varieties / breeds for stability.

The exact nature of stability in terms of parameters should immediately follow the confirmatory analysis of GE interaction. It has been observed that an approximately linear relationship exists between phenotype and environment when the latter is measured by its effect on the character under study. This regression relationship provides two measures of stability *viz.* the regression coefficient known as linear sensitivity and the deviation from mean squares due to linear regression known as non-linear sensitivity, which serve as the basis for studying the nature of adaptation of genotypes under investigation.

2. Linear Regression Model for the Analysis of GE Interaction

Let the average phenotypic value Y_{ij}, of i th genotype ($i = 1,2,...,t$) at the j th environment ($j = 1,2,\ldots,s$) be

$$Y_{ij} = \mu + d_i + (1+\beta_i)e_j + \delta_{ij} + \bar{\varepsilon}_{ij} \qquad (2.1)$$

where μ is the general mean, d_i is the effect of i th genotype, e_j is the effect of j th environment, $(1+\beta_i)$ is the regression of Y_{ij} on e_j, δ_{ij} is the deviation from regression for the i th genotype in the j th environment and $\bar{\varepsilon}_{ij}$ is the random error. For the genotype-environment interaction effect g_{ij} the following relationship, holds:

$$g_{ij} = \beta_i e_j + \delta_{ij}$$

In model (2.1), the effects d_i, e_j, δ_{ij} are such that

$$\sum_i d_i = \sum_j e_j = \sum_i \delta_{ij} = \sum_j \delta_{ij} = \sum_{i,j} \delta_{ij} = 0$$

Further, the least square estimates of μ, d_i, e_j and β_i are:

$$\hat{\mu} = \bar{Y}_{..} = (\sum_{i,j} Y_{ij}) / st$$

$$\hat{d}_i = \bar{Y}_{i.} - \bar{Y}_{..}; \hat{e}_j = \bar{Y}_{.j} - \bar{Y}_{..}$$

$$b_i = 1 + \hat{\beta}_i = \sum_j Y_{ij}\hat{e}_j / \sum_j \hat{e}_j^2 \text{ and } \hat{g}_{ij} = (Y_{ij} - \bar{Y}_{i.} - \bar{Y}_{.j} + \bar{Y}_{..})$$

where, $\bar{Y}_{i.} = (\sum_j Y_{ij} / s)$, $\bar{Y}_{.j} = (\sum_i Y_{ij} / t)$

The deviation mean squares for the i th genotype ($s^2_{d_i}$) can be defined as,

$$s^2_{d_i} = \sum_j \hat{\delta}^2_{ij} / (s-2), \text{ where } \sum_i \hat{\delta}^2_{ij} = \sum_{i,j} Y_{ij} - s\bar{Y}^2_{.i} - b_i^2 \sum_j e_j^2$$

The form of analysis based on the model (2.1) is as presented in Table 1. It is possible to determine whether GE interaction is significant when compared to $\bar{S}^2_e$ and also testing whether the GE interaction is a linear function of the additive environmental component. For this purpose the GE interaction with (s-1)(t-1)

Table 1. Analysis of Variance for GxE interaction

Source	d.f.	S.S	M.S
Genotypes(G)	(t-1)	$s\sum_i g_i^2$	MS1
Environments(E)	(s-1)	$t\sum_j e_j^2$	
G×E interaction	(t-1)(s-1)		
Heterogeneity between regression	(t-1)	$\sum_i \beta_i^2 (\sum_j e_j^2)$	MS2
Residual	(t-1)(s-2)	$\sum_{i,j} \delta_{ij}^2$	MS3
Average error	s(t-1)(r-1)		$\overline{S}_e^2$

Degrees of freedom is partitioned into two components:
(i) heterogeneity between regression sum of squares with (t-1) degrees of freedom and
(ii) the remainder sum of squares with (t-1)(s-2) degrees of freedom.

For testing the significance of regression for individual lines the following analysis can be undertaken.

Source	d.f.	MS
Regression	1	$(1+\beta_i^2)(\sum_j e_j^2)$
Residual	(s-2)	$\sum_j \delta_{ij}^2 /(s-2)$
Total	(s-1)	$\sum_j (y_{ij} - y_{i.}/s)^2 /(s-1)$

Here we are testing the hypothesis that $(1+\beta_i)$ has non-zero expectation. The component $\beta_i^2 \sum_j e_j^2$ for each genotype can also be extracted and compared with the same residual.

3. Biological and Agronomic Concepts of Stability

Depending upon the final goal of the breeder and the character under consideration two concepts of stability can be introduced the biological and agronomic concepts (Becker, 1981). These are also known as static and dynamic concepts respectively (Leon, 1985). Under the biological concept a stable genotype is one whose phenotype Y_{ij} shows little deviation from the expected character level $\overline{Y}_{i.}$ when the genotype is performance-tested over a number of environments. This type of yield stability is analogous to the concept of genetic homeostasis, first introduced by Lerner (1954). In view of the restriction regarding constant performance level at the different environments considered, this concept is also termed the static concept of stability.

The biological concept as applied to a character like grain yield would mean a stable genotype performs well under adverse environments but not so well under favourable environments. But with increased inputs, improved technology, etc. the breeder would prefer a genotype whose performance in a particular environment is at a level expected depending on the level of productivity of the location as measured by the average productivity of all the genotypes grown in that environment. In other words he is interested in a variety which does not show any genotype-environment interaction. *i.e.* $(Y_{ij} - \bar{Y}_{i.} - \bar{Y}_{.j} + \bar{Y}_{..}) \cong 0$, for all i. This concept which permits a predictable response in each environment and no deviation from the amount predicted is known as the agronomic or dynamic concept of stability.

An overview of the widely used parametric stability measures and their underlying stability concept is provided in Table 2.

Table2: Common stability measures and their underlying stability concepts

Stability measures	**Symbol**	**Stability concept involved**
Environmental variance	$S^2_{Y_i}$	biological
Ecovalence	W_i	agronomic
Stability variance	$\hat{\sigma}^2_i$	agronomic
Regression coefficient	b_i	biological/agronomic
Deviation mean square	$s^2_{d_i}$	agronomic
Coefficient of determination	r^2_i	agronomic
Hanson's stability measure	$\hat{D}^2_{(i.)}$	agronomic

For a better appreciation of the idea conveyed through this table, we now turn to a formal definition of these measures.

Most of the yield stability statistics in vogue are measures according to the agronomic concept. For the biological concept only two measures are available and they are the environmental variance $S^2_{Y_i}$ and the environmental coefficient of variation (CV_i)

$$S^2_{Y_i} = \sum_j (Y_{ij} - \bar{Y}_{i.})^2 / (s-1) \qquad (3.1)$$

$$CV_i = (S_{Y_i} / \bar{Y}_{i.}) \times 100 \qquad (3.2)$$

lesser values of which are always sought after for higher stability. Although the measures are quite sound theoretically, they are not of much practical utility in the assessment of stability owing to the fact that (i) stability under biological concept is usually associated with relatively poor yield and (ii) high level of performance over a wide range of environments is difficult to materialize. Accordingly, these measures are rarely useful to the breeder who is always

looking for high yield stability. They are however useful in disease and quality traits, the levels of which are to be maintained at all costs. We now turn to stability measures under agronomic concept.

Wricke's ecovalence measure

The contribution of a genotype to the interaction sum of squares provides a simple and easy to compute measure known as ecovalence measure (W_i):

$$W_i = \sum_j (Y_{ij} - \bar{Y}_{i.} - \bar{Y}_{.j} + \bar{Y}_{..})^2 = \sum_j \hat{g}_{ij}^2 \qquad (3.3)$$

The lower the value of W_i the smaller will be the fluctuations from the predictable response in different environments so much so that the genotype with the least ecovalence is considered to be the ideal from the point of view of yield stability.

Shukla's stability variance measure

An estimate of the variance $\hat{\sigma}_i^2$ of $g_{ij} + \bar{\varepsilon}_{ij}$ in terms of the residuals in a two-way classification is a useful indicator of stability of the i th genotype. This statistic $\hat{\sigma}_i^2$, termed stability variance is defined as follows:

$$\hat{\sigma}_i^2 = \frac{t}{(s-1)(t-2)} W_i - \frac{MS(GE)}{(t-2)} \qquad (3.4)$$

where W_i is as defined in (3.3) and MS(GE) is the GE interaction mean square [MS(GE) = $\sum_{i,j} \hat{g}_{ij}^2 /(s-1)(t-1)$]. The statistic being a linear combination of W_i, both W_i and $\hat{\sigma}_i^2$ are equivalent for the purpose of ranking the genotypes.

Hanson's stability measure

Hanson's genotypic stability measure $\hat{D}_{(i.)}^2$ is defined as

$$\hat{D}_{(i.)}^2 = \sum [Y_{ij} - \bar{Y}_{i.} - b_{\min}(\bar{Y}_{.j} - \bar{Y}_{..})]^2$$

$$= \sum [Y_{ij} - \bar{Y}_{i.} - b_{\min} e_j]^2 \qquad (3.5)$$

where $b_{\min}$ is the minimum of b_i ($i = 1,2,...,t$) values in Eberhart and Russell sense. This shows that the stable genotype is one which does not deviate from the straight line

$$Y_{ij} = \bar{Y}_{i.} + b_{\min}(\bar{Y}_{.j} - \bar{Y}_{..})$$

Coefficient of determination measure

This measure of stability proposed by Pinthus (1973) and symbolized as r_i^2 is defined as follows:

$$r_i^2 = \frac{b_i^2 \sum_j \hat{e}_j^2}{b_i^2 \sum_j \hat{e}_j^2 + \sum_j \hat{\delta}_{ij}^2} \qquad (3.6)$$

where $\sum_j \hat{e}_j^2 = \sum (\bar{Y}_{.j.} - \bar{Y}_{...})^2$ and

$$\sum_j \hat{\delta}_{ij}^2 = \sum_j (Y_{ij} - \bar{Y}_{i.} - \bar{Y}_{.j} + \bar{Y}_{..})^2 - (b_i - 1)^2 \sum_j (\bar{Y}_{.j} - \bar{Y}_{..})^2$$

$$= W_i - (b_i - 1)^2 \sum_j \hat{e}_j^2$$

the latter representing the sum of squared deviations from regression which is also symbolized as $(s-1)s_{d_i}^2$. Unlike b_i, r_i^2 is independent of the scale of measurement. For ranking of genotypes high values of r_i^2 are regarded as being desired.

Eberhart and Russell's two-parameter measure

Eberhart and Russell (1966) considered the regression coefficient, b_i as his first measure of stability:

$$b_i = \sum_j (Y_{ij} - \bar{Y}_{i.})(\bar{Y}_{.j} - \bar{Y}_{..}) / \sum_j (\bar{Y}_{.j} - \bar{Y}_{..})^2$$

$$= 1 + [\sum_j (Y_{ij} - \bar{Y}_{i.} - \bar{Y}_{.j} + \bar{Y}_{..})(\bar{Y}_{.j} - \bar{Y}_{..}) / \sum_j (\bar{Y}_{.j} - \bar{Y}_{..})^2] \qquad (3.7)$$

As a second measure, he considered

$$s'^2_{d_i} = [\sum_j \hat{\delta}_{ij}^2 / (s-2)] - \bar{S}_e^2 \qquad (3.8)$$

where $\sum_j \hat{\delta}_{ij}^2$ is as defined earlier and $\bar{S}_e^2$, the average error is $\bar{S}_e^2 = (\sum_j s_j^2 / sr)$, s_j^2's being the error mean squares for different experiments, each conducted with the same number of replications r.

Perkins and Jinks two-parameter measure

By a slight modification of the regression technique based on Eberhart and Russell model Perkins and Jinks (1968) obtained the following measures:

$$\hat{\beta}_i = \sum_j (Y_{ij} - \bar{Y}_{i.} - \bar{Y}_{.j} + \bar{Y}_{..})(\bar{Y}_{.j} - \bar{Y}_{..}) / \sum_j (\bar{Y}_{.j} - \bar{Y}_{..})^2 \qquad (3.9)$$

$$s_{d_i}^2 = \sum_j \hat{\delta}_{ij}^2 / (s-2) \qquad (3.10)$$

where β_i is related to b_i such that $\beta_i = b_i - 1$ holds.

EXAMPLE 1:

LOCATION 1:

	R1	R2	R3	R4
V1	1250	1150	1175	1000
V2	1350	1450	1150	1050
V3	1250	750	925	1100
V4	1300	1500	1425	1575
V5	1550	1375	1475	1300
V6	1300	1325	1525	1675
V7	1350	900	1475	1675
V8	1175	1500	1250	1225
V9	1150	1225	1525	1275
V10	1200	1250	900	1000
V11	1725	1725	2100	1650
V12	1500	1450	1125	1625
V13	1300	1000	1250	950
V14	1175	1200	1275	1400

LOCATION 2:

	R1	R2	R3	R4
V1	1850	1650	1700	1150
V2	1800	1600	1800	1200
V3	2000	1950	1900	1700
V4	2200	2000	1800	2100
V5	1350	1650	1300	1250
V6	1900	1900	1550	1700
V7	1500	1500	1800	1400
V8	1000	1200	1750	1150
V9	1600	1900	1600	1700
V10	2100	1550	1650	2100
V11	1550	1400	1350	1550
V12	1500	1800	2050	2150
V13	900	1150	1050	900
V14	1900	1900	1700	1800

LOCATION 3:

	R1	R2	R3	R4
V1	1150	1210	1170	1200
V2	910	1200	1130	1100
V3	1030	1200	1100	850
V4	750	900	770	660
V5	690	620	800	675
V6	540	860	630	550
V7	1050	1070	1000	1260
V8	560	650	470	640
V9	830	1160	1170	1100
V10	1270	900	900	830
V11	400	950	780	1010
V12	1130	1450	1150	1250
V13	1450	1500	1200	1350
V14	600	850	660	760

LOCATION 4:

	R1	R2	R3	R4
V1	880	860	895	910
V2	870	880	785	870
V3	755	750	795	760
V4	770	810	810	935
V5	890	945	900	915
V6	800	800	710	870
V7	1095	995	1050	1045
V8	1000	980	890	980
V9	920	895	955	990
V10	885	930	805	855
V11	1045	900	1045	1065
V12	860	805	890	955
V13	860	975	940	970
V14	980	890	805	865

Ranking of genotypes/varieties based on stability measures (for EXAMPLE 1)

Varieties	Mean	b_i	$\sum_j \hat{\delta}_{ij}^2$	CV_I	ECOVALENCE	r_i^2	$\hat{\sigma}_i^2$
V1	6	3	8	3	5	10	5
V2	7	5	4	4	2	6	2
V3	10	11	13	12	11	9	11
V4	2	14	3	14	12	2	12
V5	13	6	10	7	7	11	7
V6	11	13	6	13	9	4	9
V7	4	2	1	2	3	1	3
V8	14	4	12	8	10	12	10
V9	5	7	2	5	1	3	1
V10	8	4	9	10	8	8	8
V11	3	10	14	9	13	13	13
V12	1	9	7	6	4	7	4
V13	12	1	11	1	14	14	14
V14	9	12	5	11	6	5	6

4. Non-linear GE Interaction Model

Most of the regression techniques currently employed in the analysis of GE interaction can adequately describe the behaviour of genotypes over different environments only when the genotypic response is fairly linear, a situation characterized by the overwhelming contribution of linear regression component to the total GE interaction variation. In the event of the remainder mean squares (non-linear component), relative to heterogeneity of regression mean square, accounting for a large part of the interaction variation, a characterization of the genotypes based on the linear regression coefficient will be largely misleading. Presence of significant non-linear interaction have been reported in different crops such as Carrots and *Nicotiana rustica.* The non-linearity can largely be attributed to the presence of yield thresholds, after attaining which some of the genotypes cease to respond to further environmental changes, environmental specificity in gene expression in crop plants, *etc.* A different approach is altogether is needed to deal with such non-linear GE interactions.

EXAMPLE 2: Given below is the data of RBD experiments conducted over 6 locations for 8 varieties (No. of replications=4).

	L1	L2	L3	L4	L5	L6
V1	360.5	362.77	363.12	172	305	660.3
V2	325	463.4	457.75	193.75	294	611
V3	314	300.9	457.18	175.09	315.7	492
V4	432	365.66	430.5	217.75	377.9	488.8
V5	421	463.6	460	314.41	336.2	625.9
V6	334	303	439.68	277.3	408.4	758
V7	470	270.33	433.43	241	488.7	536.6
V8	383	168.83	382.1	254.86	327.3	424.1
S.E.	50.73	46.22	68.82	40.81	51.04	73.42

ANOVA TABLE FOR STABILITY

SOURCE	D.F.	SS.	MSS.	F-VALUE
LINES(GENOTYPES)	7	0.61404076E+05	0.87720109E+04	
ENVIRONMENT	5	0.51743219E+06	0.10348644E+06	
LINEXENVIRONMENT	35	0.17016124E+06	0.48617496E+04	6.106^{**}
HETERO BET REG	7	0.38791072E+05	0.55415817E+04	1.181^{NS}
REMAINDER	28	0.13137014E+06	0.46917907E+04	5.893^{**}
AVERAGE ERROR	126	0.10031200E+06	0.79612700E+03	

By making interaction analysis of yield for individual varieties in six environments, it can be observed that large amount of non-linear interaction is due to v7 & v8.

4.1 Regression Models for Analysis of Non-linear GE Interactions

A common approach in non-linear situations is to reduce the complex interaction responses to a series of orderly linear responses. Assuming that the threshold limits differ among genotypes and are subjected to genotypic control the response of each genotype can be represented by a pair of intersecting straight lines; from the poorest environment to the environment in which further response ceases the straight line will have a positive slope and then it will have a zero slope.

Two-phase regression technique

Verma *et al.* (1978) proposed this technique which is consistent with the assumption that distinct sets of genes control linear sensitivity in contrasting environments. In the proposed analysis, the environments are first truncated at zero environmental index providing two subsets of environments, first comprising all the negative environmental indices (below average environment) and the second with all positive environmental indices (above average environment) as well as the negative index with the smallest absolute value. Two response curves for the trial genotypes are fitted separately for the two sets of graded environments.

The linear regression coefficients for the two sub sets of environments can be classified into nine categories as indicated in Table 3.

Table 3. Classification of genotypes on the basis of regression coefficients in the two sets of environments

Category	Regression coefficient	
	Below-average environments	Above -average environments
1.	<1.0	<1.0
2.	<1.0	=1.0
3.	<1.0	>1.0
4.	=1.0	<1.0
5.	=1.0	=1.0
6.	=1.0	>1.0
7.	>1.0	<1.0
8.	>1.0	=1.0
9.	>1.0	>1.0

The categories (2) and (3) approach the ideal genotype depending on their mean performance. The categories (1) and (7) are suited to adverse environmental conditions while (9) to favourable conditions.

The environmental indices in example 2 are calculated as

e_j = -2.515 -57.649 45.516 -151.683 -25.803 192.134

By applying the two-phase regression technique to v7 & v8, we can get results as

Variety	Sensitivity in	
	poor environment	favourable environment
V7	1.66±0.76	0.43±0.27
V8	0.74±0.85	0.23±0.06

The variety V7 shows unit sensitivities in poor as well as favourable environments, while V8 shows unit sensitivity under poor conditions and <1 in favourable conditions. Hence both varieties are not dependable from the point of general or specific adaptation.

Refined two-phase regression model

Pooni and Jinks (1980) proposed the procedure which is more general in the sense that a pair of regression lines are fitted without the tacit assumption that the switch over between regressions takes place around the same environmental value. This is achieved by refining the two-phase regression technique on the principle that significant part of the non-linearity results from thresholds in response of genotypes to changes in environmental conditions. Here the environmental value at which a particular genotype changes its slope is determined as the point of intersection of a pair of best fitting straight lines, which allow for a

considerable reduction in the residual mean square. The model for fitting a pair of intersecting straight lines in a matrix form as

$$\mathbf{Y_i} = a_1\mathbf{D_i} + a2(\mathbf{1} - \mathbf{D_i}) + [b1\mathbf{D_i} + b2(\mathbf{1} - \mathbf{D_i})]\mathbf{X_i} + \mathbf{U_{1i}}\mathbf{D_i} + \mathbf{U_{2i}}(\mathbf{1} - \mathbf{D_i}) \qquad (4.1)$$

where $a_1, a_2, b1, and\ b_2$ are regression parameters; $\mathbf{Y_i}$ *and* $\mathbf{X_i}$ the dependent and independent variable vectors; $\mathbf{U_{1i}}$ *and* $\mathbf{U_{2i}}$ are the remainder vectors for the two regressions and $\mathbf{D_i}$ a discrete vector of coefficients with values of 0 and 1 respectively below and above the change-over point between the two sets of environments. Model (4.1) can be expressed in an alternative form, by considering the expectation of $\mathbf{Y}$ as

$$\mathbf{Y} = \mathbf{X}\beta$$

$$\mathbf{Y}' = (\mathbf{Y_{1j}}, \mathbf{Y_{2j}}, ...\mathbf{Y_{nj}}), \textit{the vector of dependent variables}$$

$$\beta = (a_1, b_1, a_2, b_2), \textit{the vector of regression parameters}$$

$$\mathbf{X} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}, \textit{a partitioned matrix of in which } \mathbf{B} \textit{ and } \mathbf{C} \textit{ are null}$$

matrices of order k x 2 and (s-k) x 2 respectively ($3 \le k \le s-3$). The first column of $\mathbf{A}$ *and* $\mathbf{D}$ are vectors with k, (s-k) unit elements respectively. The second column of $\mathbf{A}$ contains the first k environmental values and that of $\mathbf{D}$ the remaining (s-k) environmental values. It is easy to see that $(\mathbf{X}'\mathbf{X})$ reduces to

$$(\mathbf{X}'\mathbf{X}) = \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{N} \end{bmatrix}$$

where $\mathbf{M}$ *and* $\mathbf{N}$ *are* non-zero 2 x 2 matrices and $\mathbf{0}$ is the null matrix of the same order clearly,

$$(\mathbf{X}'\mathbf{X})^{-1} = \begin{bmatrix} \mathbf{M}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{N}^{-1} \end{bmatrix}$$

and the least squares solution of Eq. ( ) is

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-\mathbf{1}}\mathbf{X}'\mathbf{Y}$$

with the corresponding sum of squares due to regression as

$$\hat{\beta}'\mathbf{X}'\mathbf{Y} = \mathbf{Y}'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-\mathbf{1}}\mathbf{X}'\mathbf{Y}$$

The variance of $\hat{\beta}$ can be obtained as

$$\mathbf{V}(\hat{\beta}) = \hat{\sigma}^{\mathbf{2}}(\mathbf{X}'\mathbf{X})^{-\mathbf{1}}$$

where

$$\hat{\sigma}^2 = \frac{Residual\ sum\ of\ squares}{(s-4)}$$

$$= \frac{\mathbf{Y}'\mathbf{Y} - \hat{\beta}'\mathbf{X}'\mathbf{Y}}{(s\text{-}4)}$$

In the case of EXAMPLE 2 we have $\mathbf{X} = \begin{bmatrix} 1 & -151.683 & 0 & 0 \\ 1 & -57.649 & 0 & 0 \\ 1 & -25.803 & 0 & 0 \\ 0 & 0 & 1 & -2.515 \\ 0 & 0 & 1 & 45.516 \\ 0 & 0 & 1 & 192.134 \end{bmatrix}$

By applying two-phase regression technique, we can observe

for V7: $b_1=1.59\pm0.98$ and $b_2=0.42\pm0.63$ with $R^2=0.77$
and for V8: $b_1=0.23\pm0.84$ and $b_2=0.23\pm0.54$ with $R^2=0.73$

4.2 Principle Component Approach (PCA)

William (1952) considered an equivalent model to the linear regression approach which was regarded as first principle component technique of GE interaction. He showed that the least square estimation of the regression coefficient was equivalent to extracting the first principle component of the genotypic performance. William's method was rediscovered by Gallob (1968) and Mendel (1969). The PCA model is

$$Y_{ij} = \mu + \sum_{n=1}^{N} \lambda_n \alpha_{in} \gamma_{jn} + \varepsilon_{ij} \qquad (4.2)$$

where,

Y_{ij}, μ and ε_{ij} are same as defined earlier.

λ_n is the eigen value of the principle component analysis (PCA) axes, n.
α_{in} is the ith genotype PCA score for n the PCA axis,
γ_{jn} is the jth environment PCA score for nth PCA axes,
N is the number of PCA axes retained on the model.

This approach of studying GE interaction is based on the multiplicative model, which describes the interaction variation efficiently but not the main effect, unlike the variance component approach.

The regression approach and PCA explained for studying GE interaction requires the data of the estimates of yield or performance of genotypes in particular environments for analysis. In India multilocation yield trials of genotypes are conducted by the All India Crop Improvement projects of different crops, and till date the estimate of performance of the genotypes in an environment is taken as cell means (averaged over replications). Averaging over few replications may not be good enough to get an accurate estimate of performance, hence, computer intensive statistical method like Additive main effects and Multiplicative Interaction (AMMI) analysis and Best Linear Unbiased Prediction (BLUP) can provide better estimates than the traditional one with same number of replications and the cost incurred.

References

Becker, H.C. (1981). Biometrical and empirical relations between different concepts of phenotypic stability in: Gallais, A.(Ed) *Quantitative Genetics and Breeding Methods*. 307-314. Versailles: I.N.R.A.

Eberhart, S.A., and Russell, W.A. (1966). Stability parameters for comparing varieties. *Crop. Sci*., **6,**36-40.

Gauch, H.G. Jr. (1988). Model selection and validation of yield trials with interaction. *Biometrics*, **44,** 705-715.

Gauch, H.G. Jr. (1992). *Statistical analysis of regional yield trial*. Elsevier, Amsterdam.

Gollob, H.F. (1968). A statistical model which combines features of factor analysis and analysis of variance techniques. *Psychometrika*, **33,** 73-115.

Leon, J. (1985). Beitrage zur Erfassung der Phanotypischen Stabilitat unter besonderer Beruckischtigug unterschidilicher Heterogenitats- und Heterozygotiegrade sowie einer zusammenfassenden Beurteilung von Ertragshohe and Ertragssicherheit. Dissertation, Christian-Albrechts-Universitat Kiel.

Lerner, I.M. (1954). Genetic Homeostasis. *Oliver & Boyd*, London.

Mendel, J. (1969). The partitioning of interaction in analysis of variance. *J. Res. natn. Bur. stand*, **73B,** 309-328.

Perkins, J.M. and Jinks, J.L. (1968). Environmental and genotype-environmental components of variability.III Multiple lines and crosses. *Heredity*, **23,**339-356.

Pinthus, M.J. (1973). Estimate of genotypic value:A proposed method. *Euphytica*, **22,**121-123.

Pooni, H.S. and Jinks, J.L. (1980). Non-linear genotype X environment interaction. II Statistical models and genetical control. *Heredity*, **45,** 389-400.

Rao, A.R. (1997). Some Contribution to Improved estimation of Genetic Parameters. *Unpublished Ph.D. Thesis, P.G.School, I.A.R.I., New Delhi.*

Singh, M., Leccarelli, S. and Hamblin, J. (1993). Estimation of heritability from varietal trial data. *Theor. Appl. Genet.*, **86,** 437-441.

Verma, M.M., Chahal, G.S. and Murty, B.R. (1978). Limitations of conventional regression analysis - A proposed modification. *Theor. Appl. Genet*., **53,** 89-91.

Zobel, R.W., M. J. Wright and H.G. Gauch, Jr. (1988). Statistical analysis of a yield trial. *Agron. J.* **80,** 388-393.

APPENDIX

INTERACTION ANALYSIS OF V8 & V7

```
data cas10;
input y1 y2 x;
cards;
254.86  241          -151.683
168.83  270.33       -57.649
327.3    488.7       -25.803
383      470         -2.515 ;
382.1   433.43       45.516
424.1   536.6        192.134 ;
proc reg;
model y1 y2=x;
run;
```

V8 & V7 (for finding values of regression coefficients in poor and favourable environments- ***Two phase regression technique***)

```
data cas1;
input y1 y2 x;
cards;
254.86  241          -151.683
168.83  270.33       -57.649
327.3    488.7       -25.803
383      470         -2.515 ;

proc reg;
model y1 y2=x;
run;

data cas2;
inputs y1 y2 x;
cards;
383     470     -2.515
382.1   433.43       45.516
424.1   536.6        192.134 ;
proc reg;
model y1 y2=x;
run;
```

Two - intersecting lines model

```
proc iml;
x={1 -151.683 0 0,1 -57.649 0 0,1 -25.803 0 0,0 0 1 -2.515 ,0 0 1 45.516,0 0 1 192.134};
a=x`*x;b=inv(a);
y={254.86, 168.83, 327.30, 383 ,382.10, 424.1};
c1=x`*y;
c=b*c1;
d=(y`*y-c`*c1)/2;
e1=d*b;
print x; print y;
print c; print d;
print e1;
run;
proc iml;
x={1 -151.683 0 0,1 -57.649 0 0,1 -25.803 0 0,0 0 1 -2.515 ,0 0 1 45.516,0 0 1 192.134};
a=x`*x;b=inv(a);
y={241, 270.33, 488.7, 470, 433.43, 536.6};
c1=x`*y;
c=b*c1;
d=(y`*y-c`*c1)/2;
e1=d*b;
print x; print y;
print c; print d;
print e1;
run;
```

EXERCISE 1: Given below the data obtained from RBD experiments conducted at ten locations for eighteen varieties over three replications. The standard errors for all the environments are also given. Analyze the data for stability and rank the varieties over all the locations.

	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10
V1	1261	1082	1880	844	570	150	1019	400	322	567
V2	1228	1028	1370	602	627	133	833	461	478	808
V3	1267	861	1681	412	497	122	833	583	456	358
V4	1572	1028	1269	538	640	78	945	256	439	508
V5	1072	722	1024	551	430	72	796	1039	617	325
V6	1194	1194	1708	900	618	217	778	411	600	933
V7	817	750	1431	800	530	128	963	436	494	900
V8	917	1306	1153	742	655	177	759	689	567	492
V9	1061	861	736	533	277	144	806	483	589	450
V10	561	611	972	471	350	222	741	544	389	375
V11	533	1250	981	676	458	89	787	350	483	433
V12	794	1028	1292	884	253	72	880	344	517	317
V13	700	806	1421	884	610	78	741	339	422	600
V14	1094	1333	1333	751	728	89	602	406	300	675
V15	1167	583	1745	827	550	156	1083	556	528	375

V16	1328	694	1389	827	707	94	676	389	633	375
V17	1272	750	1565	751	597	100	1028	472	661	600
V18	1317	861	1755	876	477	72	741	550	594	433
S.E.	67.9	38.4	100.8	72.9	23.4	11.8	100	44.9	36.8	15.6

EXERCISE 2: Given below the data obtained from RBD experiments conducted at ten locations for eighteen varieties over three replications. The standard error for all the environments are also given. Analysis the data for stability and identify whether non-linear interactions are present or not? If so, how will you deal with the situation.

	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10
V1	1657	1150	1275	1891	1000	805	1200	2229	1356	1472
V2	1454	994	1250	1300	661	819	1289	1700	1211	1472
V3	1287	928	570	2783	1331	541	1178	1937	1117	1528
V4	1278	1183	811	1466	872	577	970	2104	1294	1528
V5	1435	1544	694	1678	1389	667	1019	1654	950	1972
V6	1481	1272	1369	1608	872	534	1044	1875	1222	1639
V7	1231	1289	1067	1835	950	626	778	1812	1289	1286
V8	1769	1467	867	2057	789	671	1007	1817	1172	1083
V9	1667	1528	475	1986	917	643	496	1417	672	1417
V10	1287	1389	439	1915	1244	691	889	1654	256	2194
V11	1102	1456	461	1830	950	585	519	1610	300	1111
V12	1472	1667	614	1641	1278	655	519	1733	667	1222
V13	1657	1211	847	1371	806	464	881	1667	1017	1333
V14	1472	1333	864	1229	1289	529	822	1700	1317	1639
V15	1769	1594	1094	1820	1489	370	1030	1450	1256	1139
V16	1796	1806	647	1584	1511	435	1215	1844	1056	1139
V17	1854	1417	967	1546	1053	742	1230	1696	1600	1528
V18	1454	1644	792	1560	556	886	830	1946	878	1222
S.E.	73.8	89.4	24.4	86.7	34.3	32.5	51.3	134.1	53.4	87.6

STABILITY ANALYSIS: PART II
ADDITIVE MAIN EFFECTS AND MULTIPLICATIVE INTERACTION (AMMI) MODEL IN CROP IMPROVEMENT

A.R. RAO
ICAR-IASRI, Library Avenue, New Delhi – 110 012
ar.rao@icar.gov.in

1. Introduction

United Nations projections estimate that the world population will continue to grow from the current 6 billion to about 10 billion by 2050 [FAO, 1996]. The increase in population and the subsequent rise in the demand for agricultural produce are expected to be greater in regions where production is already insufficient, in particular in Sub-Saharan Africa and South Asia. The necessary increase in agricultural production represents a huge challenge to local farming systems and must come mainly from increased yield per unit area, given the limited scope for extension of cultivated land worldwide. To meet this requirement various crop improvement programmes all over the world have been initiated.

Under any crop improvement programme a sample of promising genotypes are performance tested each year at a number of sites, representing the major growing area of the crop with a view to identify genotypes which possess the dual qualities of high-yield sustainability and low sensitivity to adverse changes in environmental condition. One of the important steps here is to assess the performance of improved genotypes in multi-environment (multi-location, multi-year or both) trials. Quite often it is observed that varieties perform differently in different environments. A specified difference in environment may produce differential (different) effect on genotype. This interplay of genetic and non-genetic effects causing differential relative performances of genotypes in different environments is called Genotype x Environment Interaction (GEI). Presence of GEI causes difficulty in identifying superior genotypes. Not withstanding its importance GEI is often a distraction in genetical analysis for which effort is usually made to overcome such interactions. One way of reducing GEI is through resistant breeding, usually adopted by plant breeders. Since only a minor part of the GEI can be attributed to controllable environmental determinants, much reduction in interaction can not be achieved. The most practical alternative is to produce progressively better adapted populations to the existing environments.

A detailed description and discussion of various aspects of GE interaction analysis is available in numerous review articles (Freeman, 1973; Hill, 1975; Denis and Vincourt, 1982; Westcott, 1986; Lin *et al*., 1986; Becker and Léon, 1988; Crossa, 1990; Romagosa and Fox, 1993; Cooper and DeLacy, 1994; van Eeuwijk, 1995; Brancourt-Hulmel *et al*., 1997; Kang, 1998), in papers included in the books edited by Williams (1976), Kang (1990), Kang and Gauch (1996), Cooper and Hammer (1996) and Kang (2002), and in the monographs by Gauch (1992), Prabhakaran and Jain (1994) and Basford and Tukey (2000). In this lecture, analysis of GE data through AMMI approach will be mainly discussed along with the examples.

2. AMMI Model

Gauch (1988, 1992) has advocated the use of AMMI analysis for yield trials. Gauch and Zobel (1988) compared the performance of AMMI analysis with the ANOVA approach and regression approach and found that ANOVA fails to detect a significant interaction component and the regression approach accounts only a small portion of the interaction sum of squares only when the pattern fits a specific regression model. The AMMI model for T genotypes and S environments is given as

$$Y_{ij} = \mu + g_i + e_j + \sum_{n=1}^{n'} \lambda_n \alpha_{in} \gamma_{jn} + \theta_{ij} \qquad (1)$$

$$\theta_{ij} \sim N(0, \sigma^2); \text{ i = 1,2,..., T; j = 1,2,..., S.}$$

where, Y_{ij} is the mean yield of i*th* genotype in the j*th* environment; μ is the general mean; g_i is the i*th* genotypic effect; e_j is the j*th* location effect; λ_n is the eigen value of the PCA axis *n*; α_{in} and γ_{jn} are the i*th* genotype j*th* environment PCA scores for the PCA axis *n*; θ_{ij} is the residual; n' is the number of PCA axes retained in the model. Ordinarily the number n' is judged on the basis of empirical consideration of F-test of significance [Gauch (1988, 1992)]). The residual combines the PCA scores from the *N - n'* discarded axes, where *N* = min (t-1, s-1). The other constraints in the model (1) are $\sum_i \alpha_{in}^2 = \sum_j \gamma_{jn}^2 = 1 \quad \forall$ n; $\sum_i \alpha_{in}\alpha_{in*} = \sum_j \gamma_{jn}\gamma_{jn*} = 0$, $n \neq n^*$; and $\lambda_1 > \lambda_2 > \cdots > \lambda_{n'} > 0$. The model in (1) can be reparameterized as

$$Y_{ij} = \mu + g_i + e_j + Z_{ij} \qquad (2)$$

where $Z_{ij} = \sum_{n=1}^{n'} \lambda_n \alpha_{in} \gamma_{jn} + \theta_{ij}$.

Let the estimate of interaction in the (i, j) th cell Z_{ij} be $\hat{Z}_{ij} = Y_{ij} - \hat{\mu} - \hat{g}_i - \hat{e}_j$. Using matrix notation, denote **Z** = ($\hat{Z}_{ij}$) a matrix of order T x S. Now, the estimates of the parameters of the model are:

$\hat{\lambda}_n$ = the non-zero eigen values of **Z′Z** (in descending order), and

$\hat{\alpha}_{in}$ = the principal components of the row sum of squares and cross product matrix **ZZ′**

$\hat{\gamma}_{jn}$ = the principal components of the column sum of squares and cross product matrix **Z′Z**

Using these we can write

$$\hat{z}_{ij} = \sum_{n=1}^{N} \hat{\lambda}_n \hat{\alpha}_{in} \hat{\gamma}_{jn} \qquad (3)$$

It follows that, $\alpha^*_{in} = \lambda_n^c \hat{\alpha}_{in}$ is the ith genotype PCA score for the *n*th axis, and $\gamma^*_{jn} = \lambda_n^{1-c} \hat{\gamma}_{jn}$ is the nth PCA score of the jth environment; where *c* is a scaling constant varying between 0 to 1. Also, Using factor analytic decomposition, **Z** may be written as

$$\mathbf{Z} = \mathbf{ADB'} \quad (4)$$

where **A** is T×N orthonormal matrix, **D** is NxN diagonal matrix with elements $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_{n'} \geq ...\lambda_N$, **B** is NxS orthonormal matrix, N is the rank of **Z**. The matrices **A**, **D** and **B** of equation (4) can be obtained from the eigen vectors and eigen values of **ZZ'** of the order TxT. The matrix **A** consists of the eigen vectors (principal components α_{in}) of **ZZ'** and the diagonal matrix **D** with square root of eigen values as diagonal elements of **ZZ'**. The matrix **B** consists of the eigen vectors (principal components γ_{jn}) can be obtained by solving $\mathbf{B} = \mathbf{Z'\,AD^{-1}}$. For many practical situations, the number of PCA axes to be retained is determined by testing the mean square of each axis with the estimate of residual through F-statistic [Gollob (1968), Gauch (1988)]. The mean sum of squares of each PCA axis is equal to the ratio of square of the corresponding eigen value and the degree of freedom of each axis obtained as T + S $-1-2n$.

Biplots

The model formulation for AMMI shows its interaction part consists of summed orthogonal products. Because of this form the interaction lends itself to graphical display in the form of so-called biplots (Gabriel, 1971). Let us start with AMMI and assume that either two terms suffice for an adequate description of the interaction. For AMMI the interaction consists then of the sum of two products: $\alpha^*_{i1}\gamma^*_{j1} + \alpha^*_{i2}\gamma^*_{j2}$. The choice of the scaling constant *c* depends on the purposes of the analysis. Usually one is more interested in the genotypes and *c* is chosen equal to one (Kempton [12]). The features of the biplots, however, are not too critically dependent on *c*, and *c* = 0.5 may suit well for most problems. The genotypic scores, α^*_{i1} and α^*_{i2}, are now interpreted as coordinates for a planar depiction of the genotypes, and the environmental scores, γ^*_{j1} and γ^*_{j2}, for a similar depiction of the environments. The scores determine the end points of genotypic and environmental vectors, which depart from the origin. Simple geometry reveals that the interaction between a genotype *i* and an environment *j* can be obtained from a projection of either vector onto the other. The reason is that the interaction according to an AMMI model with two product terms of interaction, $\alpha^*_{i1}\gamma^*_{j1} + \alpha^*_{i2}\gamma^*_{j2}$, is equal to the inner product between vectors $(\alpha^*_{i1}, \alpha^*_{i2})$ and $(\gamma^*_{j1}, \gamma^*_{j2})$, or the projection of either vector onto the other, times the length of the vector on which projection takes place. It is easy to read from a biplot the relative interactions that genotypes exhibit in a particular environment.

Example: Shown below is the pod yield of 15 varieties of ground nut crop raised at 20 locations. The experimental design used is RCBD at each location with 3 replications.

	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10
G1	1773	880	2841	2020	856	1382	1458	282	1190	1001
G2	1715	861	2497	2020	505	1104	1153	275	1394	882
G3	1241	424	3266	1717	1148	1225	1130	113	701	705
G4	1472	917	3172	2222	1505	1475	1222	632	1308	334
G5	1208	1435	3625	1919	903	1432	921	862	1081	539
G6	1893	1310	2716	2374	1320	1476	1482	680	1468	591
G7	1852	1169	2527	2222	903	1220	1407	455	1637	521
G8	1266	993	2245	1869	292	972	1171	275	1419	767
G9	1736	792	2376	2172	981	1113	1051	364	1579	364
G10	1442	695	2800	2071	1051	1890	1051	605	1684	67
G11	1530	1055	2643	2172	1412	1049	1051	567	1211	174
G12	1697	1222	2770	2273	1759	1343	1153	572	1169	353
G13	1637	1097	2715	2071	1806	1158	1199	636	1269	437
G14	1641	1403	2712	2071	792	1037	1199	757	1296	643
G15	1727	1139	2452	2071	481	883	1519	299	1330	366

	L11	L12	L13	L14	L15	L16	L17	L18	L19	L20
G1	2708	1832	1188	2252	1583	2014	2199	810	1033	992
G2	1956	1907	729	1658	1285	1986	2014	865	600	842
G3	1688	1568	1153	2073	1303	2361	2893	1028	1000	997
G4	2833	1157	792	956	1374	2570	611	486	333	1049
G5	2303	1778	577	1132	1368	2691	495	639	300	877
G6	2877	2333	1005	2636	1438	2812	1968	963	1100	1413
G7	2042	1732	1285	2046	1333	2500	2060	949	633	877
G8	2184	2037	799	1749	1368	2083	1537	732	667	965
G9	2940	1500	819	1668	1041	1944	2431	1000	633	967
G10	2083	1419	1146	1295	1750	2726	1713	50	600	1166
G11	1977	1963	1083	2063	1319	1789	1435	944	633	1309
G12	2014	2222	792	1634	1319	2271	2014	1176	1200	1026
G13	1574	1843	958	1719	1299	2014	2431	1014	1033	1379
G14	2347	1889	1035	1551	1375	1993	2222	875	933	1092
G15	1535	1574	1070	1940	1146	1514	2208	745	567	904

Using the SAS programme, SISGYS2 (Rao *et al.*, 2004), the ANOVA for the ground nut data is obtained and presented in Table 1. It is evident from Table 1 that the use of biplots to explain efficiently the interaction is very much limited, since the first two PCA axes explain only 55% of the total interaction variation. Hence at least six axes must be retained to explain GEI. All other calculations can be seen from the out put.

Table 1: AMMI analysis of variance for groundnut data

Source	Df	Sum of squares	Mean square	Variance ratio
Genotypes	14	3565604.00	254686.00	12.63**
Environments	19	107622796.00	5664357.70	280.86**
G × E interaction	266	25408293.00	95519.90	4.74**
PCA1	32	10240492.00	320015.38	12.05**
PCA2	30	3899700.40	129990.01	4.90**
PCA3	28	2795398.80	99835.67	3.76**
PCA4	26	2377000.80	91423.11	3.44**
PCA5	24	1961588.90	81732.87	3.08**
PCA6	22	1372729.60	62396.80	2.35**
Remainder	104	2761382.50	26551.75	
Average error	560	11294080.00	20168.00	

3. Simultaneous selection of varieties for yield and stability

Genotype x environment interaction continues to be a challenging issue among plant breeders, geneticists, and production agronomists who conduct crop performance trials across diverse environments. GEI can reduce progress from selection. The methods of partitioning GEI into components attributable to each genotype measure the contribution of each genotype to GEI. A universally acceptable selection criterion that takes GEI into consideration does not exist. Whenever an interaction is significant, the use of main effects, e.g. overall genotypes means across environments, is questionable. Stability of performance should be considered an important aspect of yield trials. Researchers need a statistic that provides a reliable measure of stability or consistency of performance across a range of environments, particularly one that reflects the contribution of each genotype to the total GEI. However, the stability measure alone is of limited use. To be of practical utility in a breeding or cultivar testing programme, both stability and yield (or any other trait) must be considered simultaneously so as to make selection of genotypes more precise and reliable. Also integration of stability of performance with yield through suitable measures will reduce the effect of GEI and will help in selecting cultivars in a more refined manner.

Several methods of simultaneous selection for yield and stability and relationships among them were discussed by Kang and Pham (1991). Kang (1993) discussed the reasons for emphasizing stability in the selection process. The development and use of Yield-Stability statistic (YSi) has enabled incorporation of stability in the selection process (Kang, 1993). A computer program (STABLE) for calculating this statistic is available free of charge (Kang and Magari, 1995). Kang's Yield-Stability statistic (Kang, 1993) has been evaluated and found to be useful for recommending varieties (Pazdernik et al., 1997; Hussein et al., 2000). However, Bajpai and Prabhakaran (2000) observed that Kang's rank-sum method has an inherent weakness that it is weighing heavily towards yield performance, apart from the arbitrariness in the scoring procedure. Accordingly they proposed three new indices (I1, I2, I3), which were found to be superior to Kang (1993) indices.

AMMI based Selection indices for cultivar × environment data

Rao *et al.* (2004) proposed a new stability measure and incorporating it as a stability component, a new family of selection indices is constructed. As evident from literature on

AMMI the scope of biplots is very much limited. The inferences drawn from biplots will be valid only when the first principal component axis (PCA) or the first two PCAs explain maximum interaction variation. Whenever more than two axes are retained in the AMMI model, the biplot formulation of interaction will fail. Consequently the conclusions drawn on stability of varieties will be imprecise. However, the plant breeders would like to identify varieties which are stable and high yielding when the PCA axes retained in the AMMI model will be more than two, if the axes together accumulate considerable portion of interaction variation. Suppose that n' of the N axes are retained in the AMMI model to explain GEI, then the stability measure of i-th variety can now be determined as the end point of its vector α^*_{1i}, α^*_{2i}, . . . , $\alpha^*_{n'i}$ from the origin $0'_{n'x1}$. This can also be taken as the squared Euclidean distance between the vector $\gamma = (\alpha^*_{1i}, \alpha^*_{2i}, \ldots, \alpha^*_{n'i})'$ from the origin, in the n'- dimensional Euclidean space.

$$ASTAB_i = d_i(\gamma, 0) = \alpha_{1i}^{2*} + \alpha_{2i}^{2*} + \ldots + \alpha_{n'i}^{2*} = \sum_{n=1}^{n'} \alpha_{ni}^{2*} = \sum_{n=1}^{n'} \lambda_n \alpha_{ni}^{2} \quad (5)$$

The algebraic expression of the above said stability measure can also be derived from the spectral decomposition of the **ZZ**' matrix. As we know that

$$\mathbf{ZZ}' = \lambda_1 \alpha_1 \alpha_1' + \lambda_2 \alpha_2 \alpha_2' + \cdots + \lambda_n \alpha_n \alpha_n' + \cdots + \lambda_N \alpha_N \alpha_N' ,$$

the ith diagonal element of **ZZ**', i.e., $\sum_{j=1}^{s} Z_{ij}^2$, is nothing but the interaction effects of i-th genotype over s environments. Therefore

$$\sum_{j=1}^{s} Z_{ij}^2 = \lambda_1 \alpha_{1i}^2 + \lambda_2 \alpha_{2i}^2 + \cdots + \lambda_N \alpha_{Ni}^2 = \sum_{n=1}^{N} \lambda_n \alpha_{ni}^2 \quad (6)$$

The proposed stability measure of the i-th genotype in (5), mentioned earlier as a squared Euclidean distance, will be equal to the expression given in (6) when N = n', n' being the number of PCA axes retained in the AMMI model to explain the larger part of the GEI variation. A variety is considered as highly stable when the value of $ASTAB_i$ is small or closer to zero. The stability measure given in (5) will now be used as a stability component in the simultaneous selection index. A new family of simultaneous selection indices can thus be evolved, which consists of a yield component, measured as the ratio of the average performance of the ith genotype to the overall mean performance of the genotypes under test and a stability component, measured as the ratio of stability information ($1/ASTAB_i$) of the *i* th genotype to the mean stability information of the genotypes under test. The expression of the index is given as

$$I_2 = \frac{\bar{Y}_{i.}}{\bar{Y}_{..}} + \psi \frac{(1/ASTAB_i)}{(\frac{1}{t}\sum_{i}^{t} \frac{1}{ASTAB_i})} \quad (7)$$

where ψ is the ratio of the weights given to the stability component (w_2) and yield component (w_1) with a restriction that $w_1 + w_2 = 1$. The family of indices will consists of four indices I_{21}, I_{22}, I_{23} and I_{24} by considering the value of ψ as 1.0, 0.66, 0.43 and 0.25 respectively. The performance of the new family is assessed by standard techniques like, the percentage of high yielders and highly stable varieties present in the top 50% of the varieties selected based on the indices. The rank correlations are worked out between yield based

ranks and index based ranks, stability based ranks and index based ranks. It is evident from Table 1that at least six axes must be retained for using the proposed simultaneous selection indices. The rank orders based on yield, stability ($ASTAB_i$), proposed index and Bajpai index for each genotype and for different ψ values are presented in Table 2. Table 3 shows the rank correlations between yield, stability with the proposed indices and Bajpai indices. Significant correlations of order 0.59, 0.61, 0.64 and 0.78 are observed between yield and proposed index when value of α is taken as 1.0, 0.67, 0.43 and 0.25 respectively, whereas with the Bajpai's index the correlations are to the extent 0.49, 0.51, 0.55 and 0.64. Further, the correlations indicate superiority of the proposed index over the Bajpai's index. Also Table 3 indicates the extent of high linear relationship between the rank orders of proposed index with the stability. Besides, these correlations are at par with the correlations observed between stability and Bajpai's index. The proportion of high yielders and stable performers present in the 50% top selected genotypes based on simultaneous selection index values are presented in Table 4. From this table, it is evident that among the top 50% varieties selected based on the proposed indices, around 70% are the high yielders and 85% are high stable performers. Since the proposed indices show significant correlations with both high yield and stability as well as selects large proportion of high yielders and stable performers, they can be safely recommended to the breeders and production agronomists.

A computer programme

A SAS programme named SISGYS2 is developed for selecting genotypes simultaneously for yield and stability. This programme requires genotype means over replications from individual locations. The input file should be in Excel and should contain a single field with *yld* as first row and the subsequent rows should be the mean yield over replications for each genotype nested within locations. The input file should be named 'data.xls'. The number of genotypes and the number of locations should be provided inside the programme codes. The programme calculates the following steps: (i) genotype's mean performance (ii) genotype's stability measure ($ASTAB_i$) or d_i (iii) genotype's index value I. Based on the index values genotypes are ranked. The genotype with highest index value will be ranked 1. The SAS code developed for the purpose is given in ANNEXURE-I. To demonstrate the programme, the groundnut yields of 15 varieties in 20 locations, under cultivar X location set up, are taken. The input data is arranged in a nested fashion as genotypes within locations and output (result) is as below:

INDEX VALUE	RANK	YIELD (t/ha)	RANK1	STABILITY ($x 10^6$)	RANK2
1.33	5	1.51	2	1.98	8
1.22	8	1.31	12	1.76	6
1.11	12	1.39	7	4.04	13
1.03	14	1.32	11	5.95	14
1.01	15	1.30	13	6.03	15
1.59	1	1.69	1	1.31	3
1.46	3	1.47	4	1.23	2
1.19	10	1.27	15	1.79	7
1.21	9	1.37	8	2.18	9
1.13	11	1.37	10	3.24	12
1.33	6	1.37	9	1.41	4
1.36	4	1.50	3	1.71	5
1.23	7	1.46	5	2.67	10
1.47	2	1.44	6	1.15	1
1.07	13	1.27	14	3.16	11

Table 2: Effect of variation of weights on the rank orders of groundnut varieties in the simultaneous selection indices

Variety	Yield (t/ha)	Yield based Rank	Stability based Rank	ψ = 1.0		ψ = 0.67		ψ = 0.43		ψ = 0.25	
				a	b	a	b	a	b	a	b
G1	1.51	2	8	6	5	6	5	6	5	5	4
G2	1.34	12	6	7	3	7	3	7	4	8	5
G3	1.38	7	13	13	13	13	13	12	13	12	13
G4	1.32	11	14	14	14	14	14	14	14	14	14
G5	1.30	13	15	15	15	15	15	15	15	15	15
G6	1.69	1	3	2	4	1	4	1	3	1	3
G7	1.46	4	2	3	2	3	2	3	2	3	2
G8	1.27	15	7	8	8	8	8	8	8	10	9
G9	1.37	8	9	9	10	9	10	9	10	9	10
G10	1.36	10	12	11	12	11	12	11	12	11	11
G11	1.37	9	4	4	6	4	7	4	7	6	7
G12	1.50	3	5	5	7	5	6	5	6	4	6
G13	1.46	5	10	10	9	10	9	10	9	7	8
G14	1.44	6	1	1	1	2	1	2	1	2	1
G15	1.27	14	11	12	11	12	11	13	11	13	12

a: Proposed index based rank; b: Bajpai's index based rank

Table 3: Rank correlations between simultaneous selection indices and yield, stability for groundnut data

Index Type	Weightage on components of index							
	ψ = 1.00		ψ = 0.67		ψ = 0.43		ψ = 0.25	
	Yield	Stability	Yield	Stability	Yield	Stability	Yield	Stability
Proposed index	0.596*	0.982**	0.614**	0.975**	0.639**	0.968**	0.782**	0.914**
Bajpai's index	0.493^{NS}	0.946**	0.514*	0.943**	0.553*	0.953**	0.639**	0.932**

Table 4: Proportion of high yielders (HY) and highly stable performers (HSP) present out of top 50% genotypes selected on the basis of simultaneous selection indices

Index	Weightage (α)							
	ψ = 1.00		ψ = 0.67		ψ = 0.43		ψ = 0.25	
Type	HY	HSP	HY	HSP	HY	HSP	HY	HSP
Proposed index	0.71 (6,14,7,12,1)	0.86 (6,14,7,11,12,2)	0.71 (6,14,7,12,1)	0.86 (6,14,7,11,12,2)	0.71 (6,14,7,12,1)	0.86 (6,14,7,11,12,2)	0.86 (6,14,7,12,1,13)	0.86 (6,14,7,11,12,2)
Bajpai's index	0.71 (14,7,6,1,12)	0.86 (14,7,6,2,12,11)	0.71 (14,7,6,1,12)	0.86 (14,7,6,2,12,11)	0.71 (14,7,6,1,12)	0.86 (14,7,6,2,12,11)	0.71 (14,7,6,1,12)	0.86 (14,7,6,2,12,11)

References

Bajpai, P.K. and Prabhakaran, V.T. 2000. A new procedure of simultaneous selection for high yielding and stable crop genotypes. *Ind. J. Genet.,* 60(2), 141-146.

Basford, K.E. and Tukey, J.W. 2000. *Graphical analysis of multiresponse data: illustrated with a plant breeding trial*. Boca Raton, FL, Chapman & Hall/CRC Press.

Becker, H.C. and Léon, J. 1988. Stability analysis in plant breeding. *Plant Breed*., 101: 1-23.

Brancourt-Hulmel, M., Biarnès-Dumoulin, V. and Denis, J.B. 1997. Points de repère dans l'analyse de la stabilité et de l'interaction génotype-milieu en amélioration des plants. *Agronomie*, 17: 219-246.

Cooper, M. & DeLacy, I.H. 1994. Relationships among analytical methods used to study genotypic variation and genotype-by-environment interaction in plant breeding multi-environment experiments. *Theor. Appl. Genet*., 88: 561-572.

Cooper, M. and Hammer, G.L. (eds). 1996. *Plant adaptation and crop improvement*. Wallingford, UK, CABI.

Crossa, J., Gauch, H.G. and Zobel, R.W. 1990. Additive main effects and multiplicative interaction analysis of two international maize cultivar trials. *Crop Sci*., 30: 493-500.

Denis, J.B. and Vincourt, P. 1982. Panorama des méthodes statistiques d'analyse des interactions génotype x milieu. *Agronomie*, 2: 219-230.

FAO**.** 1996. *Food requirements and population growth*. Technical Background Document No. 4. Rome.

Freeman, G.H. 1973. Statistical methods for the analysis of genotype-environment interaction. *Heredity*, 31: 339-354.

Gabriel, K.R. 1971. The biplot-graphical display of matrices with applications to principal component analysis. *Biometrika*, 58, 453-467.

Gauch, H.G. 1988. Model selection and validation for yield trials with interaction. *Biometrics*, 44: 705-715.

Gauch, H.G. 1992. *Statistical analysis of regional yield trials: AMMI analysis of factorial designs*. Amsterdam, Elsevier.

Gauch, H.G.and Zobel, R.W. 1988. Predictive and postdictive success of statistical analysis of yield trial. *Theor. Appl. Genet. ,* 76, 1-10.

Gollob, H.F. (1968). A statistical model which combines features of factor analytic and analysis of variance techniques. *Psychometrika*, **33**, 73-115.

Hill, J. 1975. Genotype-environment interactions - A challenge for plant breeding. *J. Agric. Sci., Camb*., 85: 477-493.

Hussein, M.A., A. Bjornstad, and A.H. Aastveit. 2000. SASG x ESTAB: A SAS program for computing genotype x environment stability statistics. *Agron. J*. 92:454-459.

Kang, M. S. 1993. Simultaneous selection for yield and stability in crop performance trials: Consequences for growers. *Agron. J*., 85, 754-757.

Kang, M.S. and Gauch, H.G. (eds). 1996. *Genotype-by-environment interaction*. Boca Raton, FL, CRC Press.

Kang, M.S. (ed.) 1990. *Genotype-by-environment interaction and plant breeding*. Baton Rouge, LA, Louisiana State Univ.

Kang, M.S. (ed.) 2002. *Quantitative genetics, genomics, and plant breeding*. Wallingford, UK, CABI.

Kang, M.S. 1998. Using genotype-by-environment interaction for crop cultivar development. *Adv. Agron*., 62: 199-252.

Kang, M.S., and Magari, R. 1995. STABLE: Basic program for calculating yield-stability statistic. *Agron. J*., 87, 276-277.

Kang, M.S., and Pham, H.N. 1991. Simultaneous selection for high yielding and stable crop genotypes. *Agron. J*., 83,161-165.

Pazdernik, D.L., Hardman, L.L. and Orf, J.H. 1997. Agronomic performance of soybean varieties grown in three maturity zones of Minnesota. *J. Prod. Agric*., 10, 425-430.

Prabhakaran, V.T. and Jain, J.P. 1994. *Statistical techniques for studying genotype-environment interactions*. New Delhi, South Asian Publ.

Rao, A.R., Prabhkaran, V.T. and Singh, A.K. 2004. Development of statistical procedures for selecting genotypes simultaneously for yield and stability. *IASRI Publication*.

Romagosa, I., Fox, P.N., Garcìa del Moral, L.F., Ramos, J.M., Garcìa del Moral, B., Roca de Togores, F. & Molina-Cano, J.L. 1993. Integration of statistical and physiological analyses of adaptation of near-isogenic barley lines. *Theor. Appl. Genet*., 86: 822-826.

van Eeuwijk, F.A. 1995. Linear and bilinear models for the analysis of multi-environment trials. I. An inventory of models. *Euphytica*, 84: 1-7.

Westcott, B. 1986. Some methods of analysing genotype-environment interaction. *Heredity*, 56: 243-253.

Williams, W.T. (ed.) 1976. *Pattern analysis in agricultural science*. Amsterdam, Elsevier.

Data Mining: An Overview

Shashi Dahiya
ICAR-IASRI, Library Avenue, New Delhi – 110 012
shashi.dahiya@icar.gov.in

Introduction

Rapid advances in data collection and storage technology have enables organizations to accumulate vast amounts of data. However, extracting useful information has proven extremely challenging. Often, traditional data analysis tools and techniques cannot be used because of the massive size of a data set. Sometimes, the non-traditional nature of the data means that traditional approaches cannot be applied even if the data set is relatively small. In other situations, the questions that need to be answered cannot be addressed using existing data analysis techniques, and thus, new methods need to be developed.

Data mining is a technology that blends traditional data analysis methods with sophisticated algorithms for processing large volumes of data. It has also opened up exiting opportunities for exploring and analyzing new types of data and for analyzing old types of data in new ways. Data Mining is the process of automatically discovering useful information in large data repositories. Data mining techniques are deployed to scour large databases in order to find novel and useful patterns that might otherwise remain unknown. They also provide capabilities to predict the outcome of a future observation, such as predicting whether a newly arrived customer will spend more than Rs.1000 at a department store.

Data mining, or knowledge discovery, has become an indispensable technology for businesses and researchers in many fields. Drawing on work in such areas as statistics, machine learning, pattern recognition, databases, and high performance computing, data mining extracts useful information from the large data sets now available to industry and science.

Knowledge Discovery in Database

The transformation of data into knowledge has been using mostly manual methods for data analysis and interpretation, which makes the process of pattern extraction of databases too expensive, slow and highly subjective, as well as unthinkable if the volume of data is huge. The interest in automating the analysis process of great volumes of data has been fomenting several research projects in an emergent field called *Knowledge Discovery in Databases* (KDD). KDD is the process of knowledge extraction from great masses of data with the goal of obtaining meaning and consequently understanding of the data, as well as to acquire new knowledge. This process is very complex because it consists of a technology composed of a group of mathematical and technical models of software that are used to find patterns and regularities in the data.

Knowledge discovery in databases (KDD) has been defined as the *process of discovering valid, novel, and potentially useful patterns from data*. Let us examine these terms in more details:

- Data is a set of facts *F* (e.g. cases in databases).
- Pattern is an expression *E* in a language L describing facts in a subset F_E of *F*. *E* is called a pattern if it simpler than the enumeration of all facts in F_E.
- Process: Usually in KDD is a multi step process, which involves data preparation, search for patterns, knowledge evaluation, and refinement involving iteration after modification. The process is assumed to be non-trivial-that is, to have some degree of search autonomy.
- Validity: The discovered patterns should be valid on new data with some degree of certainty.
- Novel: The patterns are novel (at least to the system). Novelty can be measured with respect to changes in data (by comparing current values to previous or expected values) or knowledge (how a new finding is related to old ones). In general, it can be measured by a function *N (E, F),* which can be a Boolean function or a measure of degree of novelty or unexpectedness.
- Potentially useful: The patterns should potentially lead to some useful actions, as measured by some utility function. Such a function U maps expressions in L to a partially or totally ordered measure space M_U: hence u=*U (E,F).*
- Ultimately Understandable: A goal of KDD is to make patterns understandable to humans in order to facilitate a better understanding of the underlying data. While this is difficult to measure precisely, one frequent substitute is the simplicity measure. Several measure of simplicity exist, and they range form the purely syntactic to the semantic. It is assumed that this is measured, if possible, by a function S mapping expressions E in L to a partially or totally ordered space M_S: hence, *s= S (E, F).*

An important notion, called interestingness, is usually taken as an overall measure of pattern value, combining validity, novelty, usefulness, and simplicity. Some KDD systems have an explicit interestingness function *i = I (E, F, C, N, U, S)* which maps expressions in L to a measure space M_I. Other systems define interestingness indirectly via an ordering of the discovered patterns.

Based on the notions given above, we can now make an attempt to define knowledge.

Knowledge: A pattern E$\in$ is called knowledge if for some user-specified threshold i $\in$ M_I, *I (E, F, C, N, U, S) > i.*

This definition of knowledge is purely user-oriented and determined by whatever functions and thresholds the user chooses.

To extract knowledge from databases, it is essential that the *Expert* follows some steps or basic stages in order to find a path from the raw data to the desired knowledge. The KDD process organizes these stages in a sequential and iterative form. In this way, it would be interesting if the obtained results of these steps were analyzed in a more interactive and friendly way, seeking a better evaluation of these results. The process of knowledge extraction from databases combines methods and statistical tools, machine learning and databases to find a mathematical and/or logical description, which can be eventually complex, of patterns and regularities in data. The knowledge extraction from a large amount of data should be seen as an interactive and iterative process, and not as a system of automatic analysis.

The interactivity of the KDD process refers to the greater understanding, on the part of the users of the process, of the application domain. This understanding involves the selection of a representative data subset, appropriate pattern classes and good approaches to evaluating the knowledge. For a better understanding the functions of the users that use the KDD process can be divided in three classes:

(a) *Domain Expert*, who should possess a large understanding of the application domain;

(b) *Analyst*, who executes the KDD process and, therefore, he should have a lot of knowledge of the stages that compose this process and

(c) *Final User*, who does not need to have much knowledge of the domain, the *Final User* uses knowledge extracted from the KDD process to aid him in a decision-making process.

KDD Process: Knowledge discovery from data can be understood as a process that contains, at least, the steps of application domain understanding, selection and preprocessing of data, Data Mining, knowledge evaluation and consolidation and use of the knowledge. The KDD process is interactive and iterative, involving numerous steps with many decisions being made by the user. Practical view of the KDD process emphasizing the interactive nature of the process outlines the following basic steps:

- **Data Selection**: Where data relevant to the analysis task are retrieved from the database.
- **Data Preprocessing**: To remove noise and inconsistent data which is called cleaning and integration of data that is combining multiple data sources.
- **Data Transformation**: Where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations.
- **Data Mining**: An essential process where intelligent methods are applied in order to extract data patterns.
- **Pattern Evaluation**: To identify the truly interesting patterns representing knowledge based on some interestingness measures.
- **Knowledge Presentation**: Where visualization and knowledge representation techniques are used to present the mined knowledge to the user.

The several steps of KDD have been shown in the following figure.

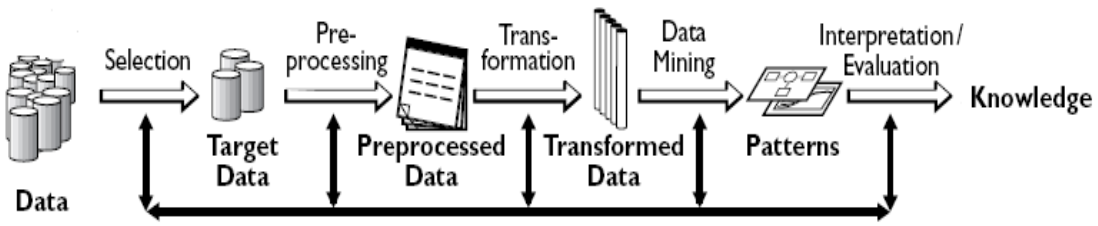


Figure: Various Steps of KDD process

The KDD process begins with the understanding of the application domain, considering aspects such as the objectives of the application and the data sources. Next, a representative sample (e.g. using statistical techniques) is removed from database, preprocessed and submitted to the methods and tools of the Data Mining stage with the objective of finding patterns/models (knowledge) in the data. This knowledge is then

evaluated as to its quality and/or usefulness, so that it can be used to support a decision-making process.

The data mining component of the KDD process is mainly concerned with means by which patterns are extracted and enumerated from the data. Knowledge discovery involves the evaluation and possibly interpretation of the patterns to make the decision of what constitutes knowledge and what does not. It also includes of encoding schemes, preprocessing, sampling and projections of the data prior to the data mining step.

Data Mining

Generally, Data Mining is the process of analyzing data from different perspectives and summarizing it into useful information. Data Mining can be defined as "the nontrivial extraction of implicit, previously unknown, and potentially useful information from data" and "the science of extracting useful information from large data sets or databases". Although it is usually used in relation to analysis of data, data mining, like artificial intelligence, is an umbrella term and is used with varied meaning in a range of wide contexts. It is usually associated with a business or other organization's need to identify trends.

Data Mining involves the process of analyzing data to show patterns or relationships; sorting through large amounts of data; and picking out pieces of relative information or patterns that occur e.g., picking out statistical information from some data.

The Data-Mining Communities: As data-mining has become recognized as a powerful tool, several different communities have laid claim to the subject:

1. Statistics.
2. AI, where it is called \machine learning."
3. Researchers in clustering algorithms.
4. Visualization researchers.
5. Databases.

In a sense, data mining can be thought of as algorithms for executing very complex queries on non-main-memory data.

Motivating Challenges

Traditional data analysis techniques have often encountered practical difficulties in meeting the challenges posed by new data sets. The following are some of the specific challenges that motivated the development of data mining:

- **Scalability:** Because of advances in data generation and collection datasets with sizes of gigabytes, terabytes, or even petabytes are becoming common. If data mining algorithms are to handle these massive datasets, then they must be scalable. Many data mining algorithms employ special search strategies to handle exponential search problems. Scalability may also require the implementation of novel data structures to access individual records in an efficient manner. For instance, out-of-core algorithms may be necessary when processing data sets that cannot fit into main memory. Scalability can also be improved by using sampling or developing parallel and distributed algorithms.

- **High Dimensionality:** It is now common to encounter data sets with hundreds or thousands of attributes instead of the handful common a few decades ago. In bioinformatics, progress in microarray technology has produced gene expression data involving thousands of features. Data sets with temporal or spatial components also tend to have high dimensionality. For example, consider a data set that contains measurements of temperature at various locations. It the temperature measurements are taken repeatedly for an extended period, the number of dimensions (features) increases in proportion to the number of measurements taken. Traditional data analysis techniques that were developed for low-dimensional data often do not work well for such high-dimensional data. Also, for some data analysis algorithms, the computational complexity increase rapidly as the dimensionality (the number of features) increases.
- **Heterogeneous and Complex Data:** Traditional data analysis methods often deal with data sets containing attributes of the same type, either continuous or categorical. As the role of data mining in business, science, medicine, and other fields has grown, so has the need for techniques that can handle heterogeneous attributes. Recent years have also seen the emergence of more complex data objects. Examples of such non-traditional types of data include collections of Web pages containing semi-structured text and hyper lines; DNA data with sequential and three-dimensional structure; and climate data that consists of time series measurements (temperature, pressure, etc.) at various locations on the Earth's surface. Techniques developed for mining such complex objects should take into consideration relationships in the data, such as temporal and spatial autocorrelation, graph connectivity, and parent-child relationships between the elements in semi-structures text and XML documents.
- **Data Ownership and Distribution:** Sometimes, the data needed for an analysis is not stored in one location or owned by one organization. Instead, the data is geographically distributed among resources belonging to multiple entities. This requires the development of distributed data mining techniques. Among the key challenges faced distributed data mining algorithms include (1) how to reduce the amount of communication needed to perform the distributed computation, (2) how to effectively consolidate the data mining results obtained from multiple sources, and (3) how to address data security issues.
- **Non-Traditional Analysis:** The traditional statistical approach is based on a hypothesize-the test paradigm. In other words, a hypothesis is proposed, an experiment is designed to gather the data, and then the data is analysed with respect to the hypothesis. Unfortunately, this process is extremely labor-intensive. Current data analysis tasks often require the generation and evaluated of thousands of hypotheses, and consequently, the development of some data mining techniques has been motivated by the desire to automate the process of hypothesis generation and evaluation. Furthermore, the data sets analyzed in data mining are typically nor the result of a carefully designed experiments and often represent opportunistic samples of the data, rather than random samples. Also, the data sets frequently involve non-traditional types of data and data distributions.

Data Preprocessing

Data preprocessing is a broad area and consists of a number of different strategies and techniques that are interrelated in complex ways. We will present some of the most

important ideas and approaches, and try to point the interrelationships among them. The preprocessing techniques fall into two categories: selecting data objects and attributes for the analysis or creating/ changing the attributes. In both cases the goal is to improve the data mining analysis with respect to time, cost, and quality. Specifically, following are the important preprocessing techniques:

- **Aggregation:** Sometimes "less is more" and this is the case with aggregation, the combining of two or more objects into a single object.. Consider a dataset consisting of transactions (data objects) recording the daily sales of products in various store locations for different days over the course of a year. One way of aggregate the transactions of this data set is to replace all the transactions of a single store with a single storewide transaction. This reduces the hundreds or thousands of transactions that occur daily at a specific store to a single daily transaction, and the number of data objects is reduced to the number of stores.

  An obvious issue is how an aggregate transaction is created; i.e. how the values of each attribute are combined across all the records corresponding to a particular location to create the aggregate transaction that represents the sales of a single store or date. Quantitative attributes, such as price, are typically aggregated by taking a sum or an average. A qualitative attribute, such as item, can either be omitted or summarized as the set of all the items that were sold at that location.
- **Sampling:** Sampling is a commonly used approach for selecting a subset of the data objects to be analyzed. In statistics, it has long been used for both the preliminary investigation of the data and the final data analysis. Sampling can also be very useful in data mining. However, the motivations for sampling in statistics and data mining are often different. Statisticians use sampling because obtaining the entire set of data of interest is too expensive or time consuming to process all the data. In some cases, using a sampling algorithm can reduce the data size to the point where a better, but more expensive algorithm can be used.
- **Dimensionality reduction:** Datasets can have a large number of feature. Consider set documents, where each documents is represented by a vector whose components are the frequencies with which each word occurs in the document. In such cases, there are typically thousands or tens of thousands of attributes (components), one for each word in the vocabulary. As another example, consider a set of time series consisting of the daily closing price of various stocks over a period of 30 days. In this case, the attributes, which are the prices on specific days again number in the thousands.

  There is variety of benefits to dimensionality reduction. A key benefit is that many data mining algorithms work better if the dimensionality—the number of attributes in the data—is lower. This is partly because the dimensionality reduction can eliminate irrelevant features and reduce noise and partly because of the curse of dimensionality. Another benefit of dimensionality reduction is that a reduction of dimensionality can lead to a more understandable model because the model may involve fewer attributes. Also, dimensionality reduction may allow the data to be more easily visualized. Even if dimensionality reduction doesn't reduce the data to two or three dimensions, data is often visualized by looking at pairs or triplets of attributes, and the number of such combinations is greatly reduced. Finally, the amount of time and memory required by the data mining algorithms is reduced with a reduction in dimensionality.

- **Feature subset selection:** The term dimensionality reduction is often those techniques that reduce the dimensionality of data set by creating new attributes that are a combination of the old attributes. The reduction of dimensionality by selecting new attributes that are a subset of the old is known as feature subset selection or feature selection. While it might seem that such as approach would lose information, this is not the case if redundant and irrelevant features are present. Redundant features duplicate much or all the information contained in one or more other attributes. For example, the purchase price of a product and ge amount of sales tax paid contain much of the same information. Irrelevant features contain almost no useful information for the data mining task at hand. For instance, student's ID numbers are irrelevant to the task of predicting student's grade point averages. Redundant and irrelevant features can reduce classification accuracy and the quality of the clusters that are found.
- **Feature creation:** It is frequently possible to create, from the original attributes, a new set of attributes that captures the important information in a data set much more effectively. Furthermore, the number of new attributes can be smaller than the number of original attributes, allowing us to reap all the benefits of dimensionality reduction. Three related methodologies for creating new attributes are: feature extraction, mapping the data to a new space, and feature construction.
- **Discretization and Binarization:** Some data mining algorithms, especially certain classification algorithms, require that the data be in the form of categorical attributes. Algorithms that fine association patterns require that the data be in the form of binary attributes. Thus, it is often necessary to transform a continuous attribute into a categorical attribute (discretization), and both continuous and discrete attributes may need to be transformed into one or more binary attributes (binarization). Additionally, if a categorical attribute has a large number of values (categories), or some values occur infrequently, then it may be beneficial for certain data mining tasks to reduce the number of categories by combining some of the values.
- **Variable transformation:** A variable transformation refers to a transformation that is applied to all the values of a variable. In other words, for each subject, the transformation is applied to the value of the variable for that object. For example, if only the magnitude of a variable is important, then the values of the variable can be transformed by taking the absolute value.

Tasks in Classical Data Mining

The two "high-level" primary goals of data mining in practice tend to be prediction and description. Data Mining tasks are generally divided into two major categories:

Predictive Tasks: the objective of these tasks is to predict the value of a particular attribute based on the values of other attributes. The attribute to be predicted is commonly known as the *target* or *dependent variable*, while the attributes used for making the prediction are known as the *explanatory* or *independent variables*.

Descriptive Tasks: Here, the objective is to derive patterns (correlations, trends, clusters, trajectories and anomalies) that summarize the underlying relationships in data. Descriptive data mining tasks are often explanatory in nature and frequently require post processing techniques to validate and explain and results.

The relative importance of prediction and description for particular data mining applications can vary considerably. However, in context of KDD, description tends to be more important than prediction.

Discovering patterns and rules: Other data mining applications are concerned with pattern detection. One example is spotting fraudulent behavior by detecting regions of the space defining the different types of transactions where the data points significantly different from the rest. Another use is in astronomy, where detection of unusual stars or galaxies may lead to the discovery of previously unknown phenomenon. Yet another is the task of finding combinations of items that occur frequently in transaction databases (e.g., grocery products that are often purchased together). This problem has been the focus of much attention in data mining and has been addressed using algorithmic techniques based on association rules.

A significant challenge here, one that statisticians have traditionally dealt with in the context of outlier detection, is deciding what constitutes truly unusual behavior in the context of normal variability. In high dimensions, this can be particularly difficult. Background knowledge and human interpretation can be invaluable.

To achieve the goals of prediction and description, following data mining tasks are carried out.

- Classification
- Association Rule Mining
- Clustering
- Evolution Analysis
- Outlier Detection
- Dependency Modeling
- Change and Deviation Detection

1. Classification: Classification, which is the task of assigning objects to one of several predefined categories, is a pervasive problem that encompasses many diverse applications. Examples include, detecting spam email messages based upon the message header and content, categorizing cells as malignant or benign based upon the results of MRI scans, and classifying galaxies based upon their shapes.

The input data for a classification task is a collection of records. Each record, also known as an instance or example, is categorized by a tuple (x, y), where x is the attribute set and y is a special attribute, designated as the class label (also known as category or the target attribute). The attributes set in a dataset for classification can be either discrete or continuous but the class label must be a discrete attribute. This is the key characteristic that distinguishes classification from regression, a predictive modeling task in which y is a continuous attribute.

Definition (classification): Classification is the task of learning a target function *f* that maps each attribute set x to one of the predefined class labels y.

The target function is also known informally as a classification model. A classification model is useful for the following purposes.

Descriptive Modeling: A classification model can serve as an explanatory tool to distinguish between objects of different classes. For example, it would be useful-for both

biologists and others-to have a descriptive model that summarizes that data shown… and explains what features define a vertebrate as a mammal, reptile, bird, fish, and amphibian.

Predictive Modeling: A classification model can also be used to predict the class label of unknown records. A classification model can be treated as a black box that automatically assigns a class label when presented with the attribute set of an unknown record.

Classification techniques are most suited for predicting or describing data sets with binary or nominal categories. They are less effective for ordinal categories (e.g., to classify a person as a member of high, medium or low income group) because they do not consider the implicit order among the categories. Other forms of relationships, such as subclass-superclass relationships among categories (e.g., humans and apes are primates, which in turn is a subclass of mammals) are also ignored.

2. Association Rule Mining: Association rule mining, one of the most important and well researched techniques of data mining, was first introduced in 1993.It aims to extract interesting correlations, frequent patterns, associations or casual structures among sets of items in the transaction databases or other data repositories. Association rules are widely used in various areas such as telecommunication networks, market and risk management, inventory control etc. Various association mining techniques and algorithms will be briefly introduced and compared later. Association rule mining is to find out association rules that satisfy the predefined minimum support and confidence from a given database. The problem is usually decomposed into two sub-problems. One is to find those itemsets whose occurrences exceed a predefined threshold in the database; those itemsets are called frequent or large itemsets. The second problem is to generate association rules from those large itemsets with the constraints of minimal confidence. Suppose one of the large itemsets is L_k, $L_k = \{I_1, I_2, \dots, I_k\}$, association rules with this itemsets are generated in the following way: the first rule is $\{I_1, I_2, \dots, I_{k-1}\} \Rightarrow \{I_k\}$, by checking the confidence this rule can be determined as interesting or not. Then other rule are generated by deleting the last items in the antecedent and inserting it to the consequent, further the confidences of the new rules are checked to determine the interestingness of them. Those processes iterated until the antecedent becomes empty. Since the second sub problem is quite straight forward, most of the researches focus on the first sub problem. The first sub-problem can be further divided into two sub-problems: candidate large itemsets generation process and frequent itemsets generation process. We call those itemsets whose support exceed the support threshold as large or frequent item- sets, those itemsets that are expected or have the hope to be large or frequent are called candidate itemsets. In many cases, the algorithms generate an extremely large number of association rules, often in thousands or even millions. Further, the association rules are sometimes very large. It is nearly impossible for the end users to comprehend or validate such large number of complex association rules, thereby limiting the usefulness of the data mining results. Several strategies have been proposed to reduce the number of association rules, such as generating only “interesting” rules, generating only “nonredundant” rules, or generating only those rules satisfying certain other criteria such as coverage, leverage, lift or strength.

3. Clustering: Clustering or cluster analysis divides the data into groups (clusters) that are meaningful, useful or both. If meaningful groups are the goal, then the clusters should capture the natural structure of the data. In some cases, however, cluster analysis is only a useful starting point for other purposes, such as data summarization. Cluster analysis groups data objects based only on information found in the data that describes the objects and their relationships. The goal is that the objects within a group be similar (or related) to

one another and different from (or unrelated to) the objects in other groups. The greater the similarity (or homogeneity) within a group and the greater the difference between groups, the better or more distinct the clustering. There are various types of clustering:

- Hierarchical versus Partitional
- Exclusive versus Overlapping versus Fuzzy
- Complete versus Partial

4. Evolution Analysis: Data evolution analysis describes and models regularities or trends for objects whose behaviors changes over time. Although this may include characterization, discrimination, association, classification, or clustering of time-related data, distinct feature of such an analysis include time-series data analysis, sequence or periodicity pattern matching, and similarity-based data analysis.

5. Outlier Detection: A database may contain data objects that do not comply with the general behaviour or model of the data. Theses data objects are outliers. Most data mining methods discard outliers as noise as exceptions. However, in some applications such as fraud detection, the rare events can be more interesting than the more regularly occurring ones. The analysis of outlier data is referred to as outlier mining.

6. Dependency modeling: Dependency modeling consists of finding a model that describes significant dependencies between variables. Dependency models exist at two levels: (1) the *structural level* of the model specifies (often in graphic form) which variables are locally dependent on each other and (2) the *quantitative level* of the model specifies the strengths of the dependencies using some numeric scale. For example, probabilistic dependency networks use conditional independence to specify the structural aspect of the model and probabilities or correlations to specify the strengths of the dependencies. Probabilistic dependency networks are increasingly finding applications in areas as diverse as the development of probabilistic medical expert systems from databases, information retrieval, and modeling of the human genome.

7. Change and deviation detection: Change and deviation detection focuses on discovering the most significant changes in the data from previously measured or normative values.

Components of Data Mining Algorithms

The data mining algorithms that address various data mining tasks have four basic components:

1. **Model or Pattern Structure:** Determining the underlying structure of functional forms that we seek from the data.
2. **Score Function:** Score functions are for judging the quality of a fitted model. Score Functions quantify how well a model or parameter structure fits a given data set. In an ideal world the choice of score function would precisely reflect the utility (i.e., the true expected benefit) of a particular predictive model. In practice, however, it is often difficult to specify precisely the true utility of a model's predictions. Hence, simple, "generic" score functions, such as least squares and classification accuracy are commonly used.
3. **Optimization and Search Method:** Optimizing the score function and searching over different model and pattern structures. The score function is a measure of how ell aspects of the data match proposed models or patterns. Usually, these models or

patters are described in terms of a structure, sometimes with unknown parameter values. The goal of optimization and search is to determine the structure and the parameter values that achieve a minimum (or maximum, depending on the context) value of the score function. The task of finding the " best" values of parameters in models is typically cast as an optimization (for estimation) problem. The task of finding interesting patterns (such as rules) from a large family of potential patterns is typically cast as a combinatorial search problem, and is, often accomplished using heuristic search techniques. In linear regression, a prediction rule is usually found by minimizing a least squares score function (the sum of squared errors between the prediction from a model and the observed values of the predicted variable). Such a score function is amenable to mathematical manipulation, and the model that minimizes it can be found algebraically. In contrast, a score function such as misclassification rate in supervised classification is difficult to minimize analytically.

4. **Data Management Strategy:** Handling the data access efficiently during the search/optimization. The final component in any data mining algorithm is the data management strategy: the ways in which the data stored, indexed, and accessed. Most well-known data analysis algorithms in statistics and machine learning have been developed under the assumption that all individual data points can be accessed quickly and efficiently in random-access memory(RAM), while main memory technology has improved rapidly, there have been equally rapid improvements in secondary (disk) and tertiary tape) storage technologies, to the extent that many massive data sets still reside largely on disk or tape and will not fit in available RAM. Thus, there will probably be a price to pay for accessing massive data sets, since not all data points can be simultaneously close to the main processor.

Problems in Data Mining

Data mining systems face a lot of problems and pitfalls. A system which is quick and correct on some small training sets, could behave completely different when applied to a larger database. A data mining system may work perfect for consistent data and perform significant worse when a little noise is added to the training set. In this section we take a look at what we mean are the most prominent problems and challenges of data mining systems today.

- Noisy Data
- Difficult Training Set
- Databases are Dynamic
- Databases may be Huge

Noisy Data: In a large database, many of the attribute values will be inexact or incorrect. This may be due to erroneous instruments measuring some property, or human error when registering it. We will distinguish between two forms of noise in the data, both described below:

Corrupted Values: Sometimes some of the values in the training set are altered from what they should have been. This may result in one or more tuples in the database conflict with the rules already established. The system may then regard these extreme values as noise, and ignore them. Alternatively, one may take the values into account possibly changing correct patterns recognized. The problem is that one never knows if the extreme values are correct or not, and the challenge is how to handle ``weird'' values in the best manner.

Missing Attribute Values: One or more of the attribute values may be missing both for examples in the training set and for object which are to be classified. If attributes are missing in the training set, the system may either ignore this object totally, try to take it into account by for instance finding what is the missing attribute's most probable value, or use the value ``unknown'' as a separate value for the attribute. When an attribute value is missing for an object during classification, the system may check all matching rules and calculate the most probable classification.

Difficult Training Set: Sometimes the training set is not the ultimate training set due to several reasons. These are the following:

Not Representative Data: If the data in the training set is not representative for the objects in the domain, we have a problem. If rules for diagnosing patients are being created and only elderly people are registered in the training set, the result for diagnosing a kid based on these data probably will not be good. Even though this may have serious consequences, we would say that not representative data is mainly a problem of machine learning when the learning is based on few examples. When using large data sets, the rules created probably are representative, as long as the data being classified belongs to the same domain as those in the training set.

No Boundary Cases: To find the real differences between two classes, some boundary cases should be present. If a data mining system for instance is to classify animals, the property counting for a bird might be that it has wings and not that it can fly. This kind of detailed distinction will only be possible if e.g. penguins are registered. Limited Information: In order to classify an object to a specific class, some condition attributes are investigated. Sometimes, two objects with the same values for condition attributes have a different classification. Then, the objects have some properties which are not among the attributes in the training set, but still make a difference. This is a problem for the system, which does not have any way of distinguish these two types of objects.

Databases are Dynamic:Databases usually change continually. We would like rules which reflect the content of the database at all times, in order to make the best possible classification. Many existing data mining systems require that all the training examples are given at once. If something is changed at a later time, the whole learning process may have to be conducted again. An important challenge for data mining systems is to avoid this, and instead change its current rules according to updates performed.

Databases may be Huge: The size of databases seem to be ever increasing. Most machine learning algorithms have been created for handling only a small training set, for instance a few hundred examples. In order to use similar techniques in databases thousands of times bigger, much care must be taken. Having very much data is advantageous since they probably will show relations really existing, but the number of possible descriptions of such a dataset is enormous. Some possible ways of coping with this problem, are to design algorithms with lower complexity and to use heuristics to find the best classification rules. Simply using a faster computer is seldom a good solution.

Conclusion

The subject of data mining, while relatively young. This overview introduces the major definitions, issues and tasks used in data mining.

The state of the art in automated methods in data mining is still in a fairly early stage of development. There are no established criteria for deciding which methods to use in which circumstances, and many of the approaches are based on crude heuristic approximations to

avoid the expensive search required to find optimal, or even good, solutions. Hence, one should be careful when confronted with overstated claims about the great ability of a system to mine useful information from large (or even small) databases.

Knowledge Representation using Ontologies

Sudeep
ICAR-IASRI, Library Avenue, New Delhi – 110 012
sudeep@icar.gov.in

Introduction

Traditional knowledge representation includes a broad range of techniques from term lists to taxonomies to complex thesauri in a print or database format, first order logics to frames. These systems are intended to use by human for different purposes like searching for the documents on keywords, searching for the books and journals in library, checking grammatical and spelling errors in documents, building expert systems and decision support systems etc. These knowledge representation techniques are not designed to take into account their use for agents and Semantic Web. For making real life Semantic Web applications in different knowledge domains, there is a need to convert these existing knowledge resources to new systems best suited for Semantic Web and agents. Ontologies are the new form of knowledge representation that acts in synergy with agents and Semantic Web architecture. The Semantic Web architecture has different layers of technologies piled over one another to form a single stack of technologies. Resource Description Framework (RDF), a layer in the architecture, is the Web metadata language to describe information about Web resources. The Web Ontology Language (OWL) is widely accepted as a standard language for sharing Semantic Web contents, resides over RDF. OWL facilitates greater machine interpretability of Web content than supported by XML, RDF, and RDF Schema (RDF-S) by providing additional vocabulary along with a formal semantics. OWL has three increasingly-expressive sublanguages: OWL Lite, OWL DL and OWL Full designed to use by specific communities of developers and users. Over the years, lots of advancement have taken place in all the layers of Semantic Web architecture, and demerits of existing architecture are pointed out. Multi Stack Architecture was released to overcome the demerits of earlier Single Stack Architecture.

Ontologies are extendible and highly reusable and deliver the user better access to his relevant content but are difficult to build from scratch. Ontologies provide formal models of domain knowledge that can be exploited by intelligent agents. Taxonomies describe real world concepts in well-defined hierarchy and exist in standardized form for numerous domains of knowledge. The work presented in the thesis starts with devising an approach to build ontologies from taxonomies, in easy and quick manner. The approach helps the knowledge engineer to build ontologies in a given domain without going into deep understanding of concepts prevalent in the domain. On the other hand, the approach act as a ready reference for the domain experts and helps them to transform the domain concepts into ontology resources by using ontology editor. The presented approach saves lot of time and gives a kick-start to the ontology development. Protégé with its OWL plug-in, a widely used tool for creation and maintenance of OWL ontologies is used for developing Soil Ontology from Soil Taxonomy to validate the approach.

Semantic Web

The W3C Consortium's broadly stated mission is to use the Web to its "full potential". According to W3C “The Web can be viewed as a universe of network-accessible

information, acting as a means of human-to-human communication, and as a space in which software agents can become tools to work with us having access to a vast amount of everything which is society, science and its problems".

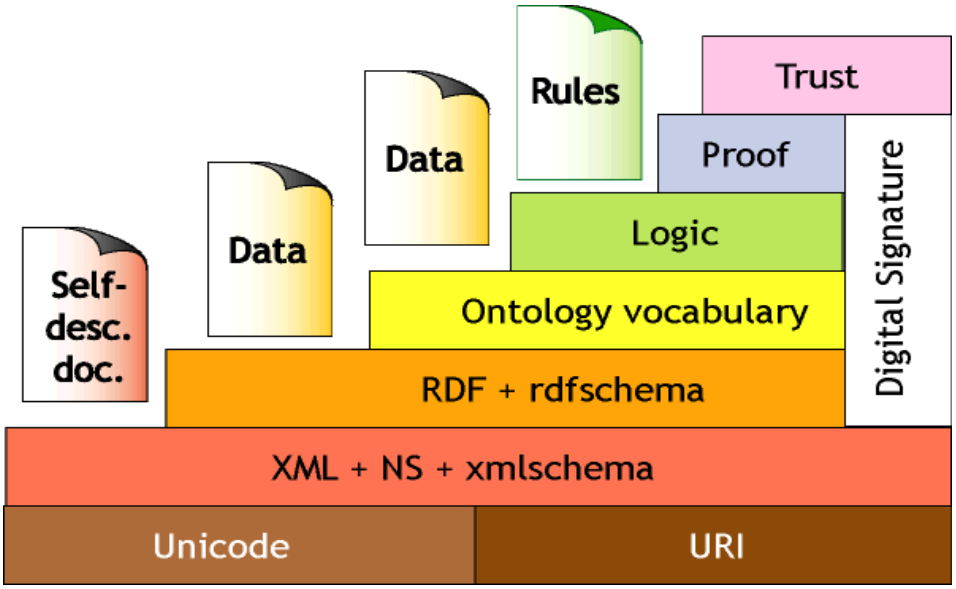


Figure 1. Single Stack Architecture of Semantic Web

Web as a universe of network-accessible information means that the principles of universality of access irrespective of hardware or software platform, network infrastructure, language, culture, geographical location, or physical or mental impairment holds. The Web in its current state is an incredible success with millions of users and information sources. Accessing, extracting, interpreting, and maintaining available information is a huge and complex task left to the human user. Tim Berners-Lee, the inventor of the WWW, coined the vision of a Semantic Web in which knowledge about Web resources is represented as conceptual and machine processable metadata. There are many applications based on software agents that may exploit and benefit from a semantics-driven approach, e.g., document and content management, information integration, or knowledge management, to name just a few. The aim of Semantic Web is to enrich the existing Web with a layer of machine-interpretable metadata so that a computer program can draw conclusions predictably. Drawing conclusions predictably means all programs that receive this information draw the same new information from it. In a Semantic Web environment, independent systems can exchange knowledge and action specifications, resulting in the execution of an activity acceptable to all systems involved.

Semantic Web Architecture

The Semantic Web architecture as shown in the Figure 2.1, forms single-stack architecture (SSA) and every new development in that area should build on top of the previous layers. The older layers at the lower part of the stack are supposed to be upward compatible with the new developments, and in this way any investment made in the old technology is preserved as the Semantic Web technology matures and expands.

The Semantic Web architecture is based on the most fundamental specification of Web architecture, the Universal Resource Identifier (URI) . The principle that anything, absolutely anything, "on the Web" should be identified distinctly by a string of characters is core to the universality. Great multiplicative power of reuse is derived from the fact that all languages use URIs as identifiers. This allows concepts written in one language to refer to concepts defined in another language. The use of URIs allows a language to leverage many forms of persistence, identity and various forms of equivalence. There are many design decisions about the properties of URIs which are fundamental in that they determine the properties of the Web. These include the rules for the parsing and use of relative URI syntax, the relationship of fragment ids to URIs. The HTTP URI space is most well-known and is characterized by a flexible notion of identity and a richness of information about and relating resources, and a dereferencing algorithm which is defined for reference by the HTTP 1.1 wire protocol. The HTTP space consists of two parts, one hierarchically delegated, for which the Domain Name System (DNS) is used, and the second an opaque string whose significance is locally defined by the authority owning the domain name.

The second part of the base layer in the Semantic Web Architecture is Unicode. Unicode provides a unique number for every character, independent of platform, programs and languages. The Unicode Standard has already become an industry standard and is used by other modern standards such as XML, Java, JavaScript, LDAP, CORBA 3.0, WML. It is supported in major operating systems, all modern browsers, and many other products. The emergence of the Unicode Standard, and the availability of tools supporting it, is among the most significant recent global software technology trends. Incorporating Unicode into client-server or multi-tiered applications and websites offers significant cost savings over the use of legacy character sets. Unicode enables a single software product or a single website to be targeted across multiple platforms, languages and countries without re-engineering. It allows data to be transported through many different systems without being corrupted. Semantic Web architecture takes all these advantages of Unicode for its easy adoption across the globe.

XML together with namespace and XML Schema form the second layer. The Extensible Markup Language (XML) is a general-purpose markup language. It is classified as an extensible language because it allows its users to define their own tags. The primary purpose of XML is to facilitate the sharing of structured data across different applications. It is used to encode documents and serialize data as well. XML provides a basic format for structured documents, with no particular semantics. As XML is already a W3C standard for sharing data, its inclusion in the Semantic Web architecture ensures smooth transition between the current Web and Semantic Web. All the technologies present in higher layers of the Semantic Web architecture use XML syntax, and takes leverage from namespaces and XML Schema. An XML namespace is also a W3C recommendation for providing uniquely named elements and attributes in an XML instance. An XML instance may contain element or attribute names from more than one XML vocabulary. If each vocabulary is given a namespace then an ambiguity between identically named elements or attributes can be resolved. All element names within a namespace must be unique. An XML namespace does not require its vocabulary to be defined, though it is fairly common practice to place either a Document Type Definition (DTD) or an XML Schema defining the precise data structure at the location of the namespace's URI. XML Schema can be used to express a set of rules to which an XML document must conform in order to be considered “valid” according to that schema. XML Schema validates XML documents

resulting in a collection of information adhering to specific datatypes, which can be useful in the development of XML document processing software.

The third layer of Semantic Web architecture consists of Resource Description Framework (RDF) and RDF Schema. RDF is a foundation for processing metadata and provides interoperability between applications by exchanging machine-understandable information on the Web. RDF emphasizes facilities to enable automated processing of Web resources. The RDF provides the concepts of assertion (property). This allows an entity-relationship-like model to be made for the data, giving it the semantics of assertions propositional logic.

The Ontology layer provides more powerful schema concepts, such as functional, inverse, and transitivity. Known uniqueness and/or unambiguousness of properties, allow a system to spot different identifiers which in fact are talking about the same thing. The Web Ontology Language (OWL) is the W3C recommendation for describing an Ontology.

The logical layer turns a limited declarative language into a Turing-complete logical language, with inference and functions. This is powerful enough to be able to define all the rest, and allows any two RDF applications to be connected together. One can see this language as being a universal language to unify all data systems just as HTML was a language to unify all human documentation systems. There is no W3C recommendation for this layer and it is a current research area to work.

A conversion language allows the expression of inference rules allowing information in one schema to be inferred from a document in another. This is part of Rules Layer. An evolution rules language allows inference rules to be given which allow a machine with a certain algorithm to do convert documents from one RDF schema into another. This is a fundamental key to evolution of the technology. There may be more than one rule standard, as different classes of rule-based system have different capabilities. Rule Interchange Format (RIF) Working Group of W3C has released the core design of Rule Interchange Format (RIF). Query languages assume different forms of query engine, but they work in basically the same problem space as rule systems. One can imagine standardizing both certain query engines and a language for defining query engines.

In order for agents or Semantic Web services to explain their results, they need to be able to generate justifications of their results in an exchangeable, combinable format. A sequential trace of information manipulations used to generate an answer can be referred as a proof of the answer. A proof language is a form of RDF or OWL that allows them to send to another an assertion, together with the inference path to that assertion from assumptions acceptable to the receiver. Proof Markup Language (PML) provides an interlingua for capturing the information agent's need to understand results and to justify why they should believe in the results. Proof Markup Language can be used to represent proof traces produced by highly optimized theorem prover, proofs such as formal natural deduction derivations, proofs resulting from services aimed at information retrieval or entity extraction, proofs representing different kinds of explanations varying from summaries of assertions or knowledge sources, and for supporting the exchange of proof information among automated reasoners in hybrid reasoning systems.

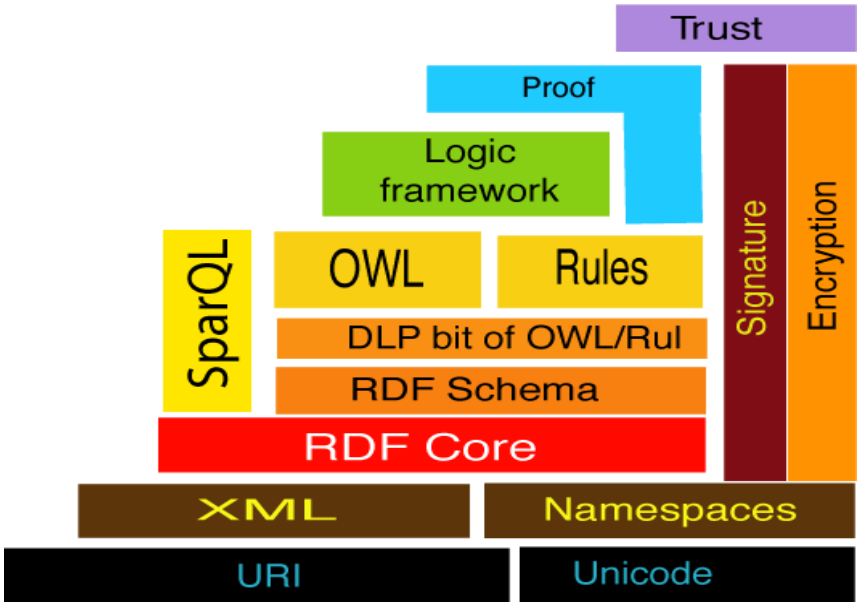


Figure 2 Multi Stack Architecture of Semantic Web

The introduction of digital signature turns a Web of reason into a Web of trust. A single digital signature format for XML documents is important. The power of the RDF logical layers will allow existing certificate schemes to be converted into RDF, and a trust path to be verified by a generic RDF engine.

An alternative architecture as shown in Figure 2 for the Semantic Web was proposed. While single-stack architecture would hold aesthetic appeal and simplify interoperability, many workers in the field believe that such architecture is unrealistic and unsustainable. The Semantic Web within a short history has encountered lot of technological difficulties. RDF(S), the first layer in the Web architecture that was dubbed “semantic" was proposed and standardized without a semantics. The difficulties that this created for OWL the subsequent and more expressive layer of the Semantic Web are well-documented in gave a formal semantics of RDF(S). It is well-known, but rarely mentioned, that OWL-DL is not properly layered on top of RDF(S), and that the single-stack architecture for the budding Semantic Web technology already has a small crack. The alternative architecture also called as multi stack architecture (MSA) proposed at the workshop recognizes the difficulties both at the philosophical level as well as implementation level with the SSA. The MSA allows multiple technological stacks to exist side-by-side, making them interoperable to a high degree, but when this is not possible, a loosely coupled integration will be acceptable in practice.

The differences between SSA and MSA are evident as some new layers are introduced and some are split. The RDF and RDF(S) layer is broken into two layers viz. RDF Core and RDF Schema. This gives more emphasis to RDF Core on which whole of the semantics are placed. Treating RDF Schema as a different layer allows research avenues in this as researchers are making new schemas for meeting their custom requirements. Also, the SSA was not having any query language to access RDF(S) or OWL and with the release of W3C recommendations on SPARQL, it has to find its place in the architecture. The MSA has a Description Logic Program (DLP) layer sandwiched between RDF Schema and OWL/Rule Layer. The details of Description Logic are presented in chapter 4. This layer

strengthen the semantic bond between the RDF(S) and OWL/Rule layer. Also, the relationship between the logical framework, rules and proof are modified to make them more realistic from implementation point of view. The security is always a concern for today's Web applications and so it will remain an issue for the future Semantic Web. The digital signatures bar in the SSA is strengthened with the encryption bar in MSA to make the Semantic Web a "Web of Trust".

Knowledge Representation using Ontologies

Traditionally knowledge is stored as unstructured text compiled into books. With the outburst of knowledge it becomes necessary to convert this unstructured knowledge into structured one. People compiled the unstructured text with different knowledge representation techniques, each one specific for a particular use like dictionaries to act as a handy tool for learning and retrieving generic or domain specific words and terms, thesauri for brief definition, synonyms, antonyms and concepts, general idea underlying the terms, and taxonomies for representing concepts or objects in the hierarchical tree structure. All these knowledge representation techniques have many limitations like lack of conceptual abstraction, limited semantic coverage, lack of consistency and limited automated processing. To overcome these limitations traditional knowledge representation techniques must be reengineered into ones that contain domain concepts linked through a rich network of well-defined relationships and a rich set of terms identifying these concepts. Knowledge representations techniques such as rule-base, semantic networks and frames provide these features.

In a rule-base system, facts consisting of widely shared domain knowledge that is accepted by the professional and other accepted sources of data form the working memory. The second type of knowledge stored in the rule-base system is called heuristic, which is the knowledge of good judgment and common good practice or "rule of thumb' in a field. The heuristic knowledge is encoded into If-Then rules. The system examines all the rule conditions (IF) and determines a subset, the conflict set, of the rules whose conditions are satisfied based on the working memory. Based on a conflict resolution strategy, one of the rules from the conflict set is triggered and any actions specified in its THEN clause are carried out. These actions can modify the working memory, the rule-base itself, or take an appropriate action as per the system logic. This loop of firing rules and performing actions continues until there are no more rules whose conditions are satisfied or a rule is fired whose action specifies the program should terminate.

A semantic network is a graphic notation for representing knowledge in patterns of interconnected nodes and arcs. Semantic networks are extensively used in machine translation, and find its application in philosophy, psychology, and linguistics. A frame is basically a structure for holding various types of knowledge. Frames are given names, with the presumption that the knowledge contained within a particular frame is in some way interrelated. Many authors and speakers may refer to frame-like structures as units, objects, concepts, schemas or entities. Frames are structured ways of representing descriptive information & they provide a natural mapping for the kind of knowledge that is centred around one concept or object. The organization allows efficient searching because there is immediate access to relevant information. Frame structure provides a natural method for representing hierarchies of information, thus allowing the inheritance of values. Although frames were conceived independently of the object-oriented paradigm they are in fact consistent with it, and provide an excellent demonstration of their power. Indeed frames are capable of representing both specific and general

knowledge, and are capable of accommodating both descriptive and prescriptive computations.

Frame Based Systems have many advantages including immediate access to relevant information, easy to include default information and detect missing values etc. Frames can easily be used in conjunction with Production rules, thereby facilitating partitioning, indexing and organizing production rules of a system. The implicit hierarchy available in frame taxonomies also permits hierarchical segmentation of rules. All these merits of the frame allow it to be the basis for knowledge representation technique called Ontology for the Web to facilitate more powerful searching and intelligent information processing.

Ontologies define domain concepts and the relationships between them, and thus provide a domain language that is meaningful to both human and machines. Ontology is a term borrowed from philosophy that refers to the science of describing the kinds of entities in the world and how they are related. In contrast to traditional knowledge representation techniques, ontologies provide conceptual abstraction and differentiated relationships. In ontologies, semantics are developed through ensuring that each concept within the domain is uniquely and precisely defined and by specifying elaborated relationships among the concepts. The relationships in ontology are explicitly named and developed with specification of rules and constraints so that they reflect the context of the domain for which the knowledge is modeled.

Given their more precise and unambiguous semantics, ontologies allow further knowledge to be inferred from the explicitly represented knowledge by applying generalization or transitivity rules, the level of applicability of which is limited in a traditional knowledge representation techniques. This added knowledge in the ontology makes it powerful when employed for intelligent information processing. Ontology is a critical component of the Semantic Web. Ontology is a controlled and common vocabulary to support the sharing and reuse of knowledge. It describes objects and the relations between them in a formal way, and has a grammar for using the vocabulary terms to express something meaningful within a specified domain of interest. The vocabulary is used to make queries and assertions. When two parties agree to use the same ontology, they agree to the meanings for all terms from that ontology and their information can be combined easily. There are many definitions of ontology given by researchers. The most agreed upon general definition of Ontology is

“Ontology is an explicit specification of a shared conceptualization”. He also laid out following design criteria:

Clarity: An ontology should effectively communicate the intended meaning of defined terms and should be documented with natural language. Definitions should be objective and stated formally so that they are independent of social or computational context. Where possible, a complete definition (a predicate defined by necessary and sufficient conditions) is preferred over a partial definition (defined by only necessary or sufficient conditions).

Coherence: An ontology should be coherent: that is, it should sanction inferences that are consistent with the terms defined formally or informally, such as those described in natural language documentation and examples. If a sentence that can be inferred from the axioms contradicts a definition or example given informally, then the ontology is incoherent.

Extendibility: An ontology should be designed to anticipate the uses of the shared vocabulary. It should be able to define new terms for special uses based on the existing vocabulary, in a way that does not require the revision of the existing definitions.

Minimal encoding bias: An encoding bias results when representation choices are made purely for the convenience of notation or implementation and should be minimized. The conceptualization should be specified at the knowledge level without depending on a particular symbol-level encoding.

Minimal ontological commitment: An ontology should require the minimal ontological commitment sufficient to support the intended knowledge sharing activities. An ontology should make as few claims as possible about the world being modeled, allowing the parties committed to the ontology freedom to specialize and instantiate the ontology as needed.

An ontology defines the terms used to describe and represent an area of knowledge. Ontologies are used by people, databases, and applications that need to share domain information (a domain is just a specific subject area or area of knowledge, like medicine, tool manufacturing, real estate, automobile repair, financial management, etc.). Ontologies include computer-usable definitions of basic concepts in the domain and the relationships among them. They encoded knowledge in a domain and also knowledge that spans domains. In this way, they made that knowledge reusable.

Resource Description Framework (Schema)

Resource Description Framework (RDF) was originally created in 1999 as a standard on top of XML for encoding metadata information. The RDF is used for encoding information about Web resources as well as information about and relations between things in the real world: people, places, concepts, etc. The semantics associated with this information enables Web applications interoperability. An RDF model that describes some Web resources is also called an RDF instance. An RDF Schema can be used to define application specific vocabularies. This schema can be associated with an RDF instance in order to validate the instance. Both RDF instance and RDF Schema are RDF models. RDF is designed to represent information in a minimally constraining, flexible way. It can be used in isolated applications, where individually designed formats might be more direct and easily understood, but RDF's generality offers greater value from sharing. The value of information thus increases as it becomes accessible to more applications across the entire Internet. The design of RDF is intended to meet the following goals:

- having a simple data model
- having formal semantics and provable inference
- using an extensible URI-based vocabulary
- using an XML-based syntax
- supporting use of XML schema datatypes
- allowing anyone to make statements about any resource

RDF uses the following key concepts:

- Graph data model
- URI-based vocabulary

- Datatypes
- Literals
- XML serialization syntax
- Expression of simple facts
- Entailment

RDF provides a general, flexible method to decompose any knowledge into small pieces, called triples, with some rules about the semantics (meaning) of those pieces. RDF builds on existing XML and URI (Uniform Resource Identifier) technologies, using a URI to identify every resource, and using URIs to make statements about resources. RDF statements describe a resource (identified by a URI), the resource's properties, and the values of those properties.

The foundation is breaking knowledge down into a labeled, directed graph. Each edge in the graph represents a fact, or a relation between two things. The edge in the example from

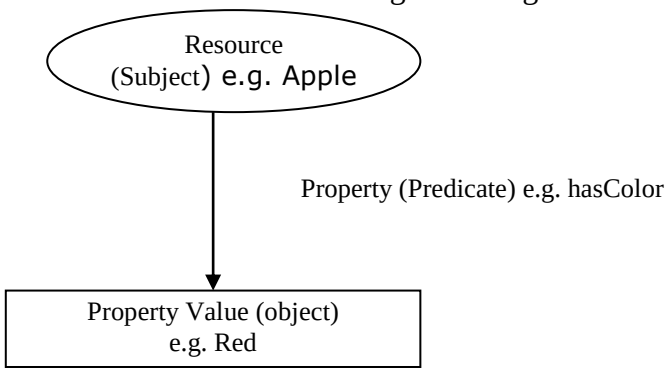


Figure 3 RDF Triple

the node "Apple" labeled "hasColor" to the node "Red" represents the fact that Apple has red color. A fact represented this way has three parts: a subject, a predicate (i.e., verb), and an object. The subject is what's at the start of the edge, the predicate is the type of edge (its label), and the object is what's at the end of the edge (Figure 3).

The abstract model of RDF comes down to four simple rules:

- A fact is expressed as a Subject-Predicate-Object triple, also known as a statement.
- Subjects, predicates and objects are given as names for entities, also called resources or nodes (from graph terminology). Entities represent something, a person, website, or something more abstract like states and relations.
- Names are *URIs*, which are global in scope, always referring to the same entity in any RDF document in which they appear.
- Objects can also be given as text values, called literal values, which may or may not be typed using XML Schema datatypes.

Entities are named by Uniform Resource Identifiers (URIs), and this provides the globally unique, distributed naming system. Since, URIs can be quite long, in RDF notations they are usually abbreviated using the concept of namespaces from XML.

Literal values, like "computer science," allow text to be included in RDF. This is used heavily when RDF is used for metadata--its original purpose. In fact, literal values are primarily what tie RDF to the real world, since URIs are just arbitrary strings.

Once triples are defined graphically, they can be coded in either RDF/XML to be accessed programmatically.

RDF/XML syntax to represent RDF graphs is as follows:

- rdf:Description - used to define a Triple, multiple triples having same subject can be defined under one rdf:Description.
- rdf:about - The subject of triple is defined by this attribute.
- Properties are defined by their URI as tag using xml namespace.
- Value of property tag can be plain literal or a resource.
- If the value of a property is a resource it is denoted by rdf:resource attribute.
- Data type of literals can be assigned by using rdf:datatype attribute.
- rdf:ID – can be used in place of rdf:about attribute if the resource URI is assigned in terms of RDF document's base URI.

By creating triples with subjects, predicates, and objects, RDF allows machines to make logical assertions based on the associations between subjects and objects. And since RDF uses URIs to identify resources, each resource is tied to a unique definition available on the Web. However, while RDF provides a model and syntax (the rules that specify the elements of a sentence) for describing resources, it does not specify the semantics (the meaning) of the resources. To truly define semantics, RDFS and OWL are needed.

RDFS is used to create vocabularies that describe groups of related RDF resources and the relationships between those resources. An RDFS vocabulary defines the allowable properties that can be assigned to RDF resources within a given domain. RDFS also allows creating classes of resources that share common properties.

Using the same triples paradigm defined by RDF, RDFS triples consist of classes, class properties, and values that define the classes and relationships between the resources within a particular domain.

In an RDFS vocabulary, resources are defined as instances of classes. A class is a resource too, and any class can be a subclass of another. This hierarchical semantic information is what allows machines to determine the meanings of resources based on their properties and classes. RDFS tags are:

- rdfs:Class : used to define a class in RDFS.
- rdfs:subClassOf : used to assign a class its parent class.
- rdf:Property :used to define a property .
- rdfs:subPropertyOf : used to assign a property its parent property.
- rdfs:domain and rdfs:rannge : schema properties to describe application specific properties.

- rdfs:Resource : RDF Schema defines all the classes as subclass of this class.

Figure 2.4 shows a sample vehicle schema in RDFS:

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [<!ENTITY xsd
"http://www.w3.org/2001/XMLSchema#">]>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xml:base="http://example.org/schemas/vehicles">

<rdfs:Class rdf:ID="MotorVehicle"/>

<rdfs:Class rdf:ID="PassengerVehicle">
  <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
</rdfs:Class>

<rdfs:Class rdf:ID="Van">
  <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
</rdfs:Class>

<rdfs:Class rdf:ID="MiniVan">
  <rdfs:subClassOf rdf:resource="#Van"/>
  <rdfs:subClassOf rdf:resource="#PassengerVehicle"/>
</rdfs:Class>

<rdfs:Class rdf:ID="Person"/>

<rdfs:Datatype rdf:about="&xsd;integer"/>

<rdf:Property rdf:ID="registeredTo">
  <rdfs:domain rdf:resource="#MotorVehicle"/>
  <rdfs:range rdf:resource="#Person"/>
</rdf:Property>


<rdf:Property rdf:ID="rearSeatLegRoom">
  <rdfs:domain rdf:resource="#PassengerVehicle"/>
  <rdfs:range rdf:resource="&xsd;integer"/>
</rdf:Property>

<rdf:Property rdf:ID="driver">
  <rdfs:domain rdf:resource="#MotorVehicle"/>
</rdf:Property>

<rdf:Property rdf:ID="primaryDriver">
  <rdfs:subPropertyOf rdf:resource="#driver"/>
</rdf:Property>

</rdf:RDF>
```

Figure 4 RDFS for vehicle schema

Web Ontology Language (OWL)

The OWL Web Ontology Language is designed to use by applications that need to process the content of information instead of just presenting information to humans. OWL facilitates greater machine interpretability of Web content than supported by XML, RDF, and RDF Schema (RDF-S) by providing additional vocabulary along with a formal semantics. OWL has three increasingly-expressive sublanguages: OWL Lite, OWL DL, and OWL Full designed to use by specific communities of implementers and users.

OWL Lite supports those users, who primarily need a classification hierarchy and simple constraints. For example, while it supports cardinality constraints, it only permits cardinality values of 0 or 1. It should be simpler to provide tool support for OWL Lite than its more expressive relatives. OWL Lite provides a quick migration path for thesauri and other taxonomies. It also has a lower formal complexity than OWL DL.

OWL DL supports those users who want the maximum expressiveness while retaining computational completeness (all conclusions are guaranteed to be computable) and decidability (all computations will finish in finite time). OWL DL includes all OWL language constructs, but they can be used only under certain restrictions (for example, while a class may be a subclass of many classes, a class cannot be an instance of another class). OWL DL is so named due to its correspondence with Description Logics a field of research that has studied the logics that form the formal foundation of OWL.

OWL Full permits maximum expressiveness and syntactic freedom of RDF with no computational guarantees. For example, in OWL Full a class can be treated simultaneously as a collection of individuals and as an individual in its own right. OWL Full allows an ontology to augment the meaning of the pre-defined (RDF or OWL) vocabulary. It is unlikely that any reasoning software will be able to support complete reasoning for every feature of OWL.

Building upon RDF and RDFS, OWL defines the types of relationships that can be expressed in RDF using an XML vocabulary to indicate the hierarchies and relationships between different resources. In fact, this is the very definition of "ontology" in the context of the Semantic Web: a schema that formally defines the hierarchies and relationships between different resources. Semantic Web ontologies consist of taxonomy and a set of inference rules from which machines can make logical conclusions. Taxonomy in this context is a system of classification for classifying resources into classes and sub-classes based on their relationships and shared properties. Since taxonomies (systems of classification) express the hierarchical relationships that exist between resources, OWL can be used to assign properties to classes of resources and allow their subclasses to inherit the same properties. OWL adds several features which enhances semantic expressibility of RDFS. List of additional features that can be defined in OWL:

- Classes can be defined as Boolean combinations of other classes using the set operators union, intersection, and complement.

  OWL Set Operators are intersectionOf, unionOf, complementOf. Classes constructed using the set operations are more like definitions than anything else. The members of the class are completely specified by the set operation. For example: making class Z as union of class X and class Y defines class Z as union of all the instances of class X and class Y.

- Classes can be stated as disjoint.

  The disjointness of a set of classes can be expressed using the owl:disjointWith constructor. It guarantees that an individual that is a member of one class cannot simultaneously be an instance of a specified other class. For example: class A1, a subclass of A is disjoint with other subclasses of A i.e. A2, A3 and A4.

- It can be stated that the two classes (with different URI) are same, and that two different instances actually represent the same individual.

  OWL constructs equivalentClass, equivalentProperty are used to tie together a set of component ontologies as part of a third. It is frequently useful to be able to indicate that a particular class or property in one ontology is equivalent to a class or property in a second ontology.

- Individuals can be declared sameAs, differentFrom, AllDifferent

  sameAs: This mechanism is similar to that for classes, but declares two individuals to be identical.

  differentFrom: This mechanism provides the opposite effect from sameAs. This is one way to assert that the stated individuals are mutually distinct.

  AllDifferent: This is a convenient mechanism to define a set of mutually distinct individuals. The following asserts that Red, White, and Rose are pairwise distinct.

- Cardinality restrictions can be specified for properties.

  OWL provides three cardinality constructs :

  owl:cardinality, which permits the specification of *exactly* the number of elements in a relation.

  owl:mincardinality, which permits the specification of the minimum number of elements in a relation.

  owl:maxcardinality, which permits the specification of the maximum number of elements in a relation.

- It can be specified that a property is transitive, symmetric, Functional, inverseOf, Inverse Functional Property.

  Transitive property: If a property P, is specified as transitive then for any x, y, and z:

  $$P(x,y) \text{ and } P(y,z) \text{ implies } P(x,z)$$

  Symmetric property: If a property P, is tagged as symmetric then for any x and y:

  $$P(x,y) \text{ iff } P(y,x)$$

  Functional property: If a property P, is tagged as functional then for all x, y, and z:

  $$P(x,y) \text{ and } P(x,z) \text{ implies } y = z$$

  inverseOf : If a property P1, is tagged as the owl:inverseOf P2, then for all x and y:

  $$P1(x,y) \text{ iff } P2(y,x)$$

  InverseFunctionalProperty: If a property P, is tagged as InverseFunctional then for all x, y and z:

$$P(y,x) \text{ and } P(z,x) \text{ implies } y = z$$

Tools for Building Ontologies

Ontology editors are applications designed to assist in the creation or manipulation of ontologies. They often express ontologies in one of many ontology languages. Among the most relevant criteria for choosing ontology editor are the degree to which the editor abstracts from the actual ontology representation language used for persistence and the visual navigation possibilities within the knowledge model. Denny, presents a detailed survey of over 50 popular software tools that have ontology editing capabilities and are in use today. The tools are useful for building ontology schemas (terminological component) alone or together with instance data. These tools are compared on number of parameters like ontology language support, Web support, graph view, multi-user support, modeling features and limitations, consistency checks, import/export formats, information extraction, merging of different ontologies.

OntoEdit, is a sophisticated ontology editor that supports methodology-based ontology construction and that takes comprehensive advantage of its inference capabilities. OntoEdit also has features such as an extremely capable plug-in structure, a lexicon component and support for collaborative engineering of ontologies. OntoEdit is based on F-logic and its core functionalities are extended by two plug-ins, viz., OntoKick and Mind2Onto to support first stages of the ontology development. OntoKick targets at creation of the requirement specification document and extraction of relevant structures for the building of the semi-formal ontology description. Mind2Onto targets at integration of brainstorming processes to build relevant structures of the semi-formal ontology description.

Protégé is an integrated software tool used by system developers and domain experts to develop knowledge-based systems. Applications developed with Protégé-2000 are used in problem-solving and decision-making in a particular domain. The Protégé-2000 GUI (graphical user interface) top-level consists of overlapping tabs for compact presentation of the parts and for convenient co-editing between them. Protégé with OWL plugin unlike all the above tools is based on W3C OWL, it is a product of KSL, Stanford University, a well known center of excellence in knowledge engineering, it is a open source, and is supported by huge research community. In the presented work, Protégé is used to develop ontologies and is discussed in brief in next section.

OBO-Edit previously known as DAG-Edit is an ontology editor developed by the Gene Ontology consortium for developing Gene Ontology. OBO-Edit is an open source, platform-independent application for viewing and editing any OBO format ontologies. OBO-Edit is a graph-based tool and provides a friendly interface for biologists for the rapid generation of large ontologies focusing on relationships between relatively simple classes.

The Integrated Ontology Development Environment (IODE) is an integrated tool set for creating and managing ontology models of an application domain. This is the commercially available tool designed to facilitate the production of ontologies specifically for managing large data volumes. The IODE has an inbuilt upper level ontology containing various models that provides both ontological structure and data integrity constraints, enabling the user to quickly and efficiently produce rigorous and robust ontologies. It is based on the OWL, based on KIF and is different from the W3C OWL.

Protégé

Protégé can be characterized as an ontology development environment. It provides functionality for editing classes, slots (properties), and instances. One of its strengths is that it can automatically generate a user interface from class definitions, and thus can support rapid knowledge acquisition. The current version of Protégé 3.1 is highly extensible and customizable. At its core is a frame-based knowledge model with support for metaclasses. These metaclasses can be extended to define other languages on top of the core frame model. Protégé can be extended with backends for alternative file formats. Currently, back-ends for Clips, UML, XML, RDF, DAML+OIL, and OWL are available. Protégé not only allows developers to extend the internal model representation, but also to customize the user interface freely. Protégé's user interface consists of several screens, called tabs, which display different aspects of the ontology in different views. Each of the tabs can be filled with arbitrary components. Most of the existing tabs provide a tree-browser view of the model, with a tree on the left and details of the selected node on the right hand side. The details of the selected object are typically displayed by means of forms. The forms consist of configurable components, called widgets. Typically, each widget displays one property of the selected object. There are standard widgets for the most common property types, but ontology developers are free to replace the default widgets with specialized components. Widgets, tabs, and back-ends are called plug-ins. Protégé's architecture allows developers to add and activate plug-ins arbitrarily, so that the default system's appearance and behavior can be completely adapted to a project's needs.

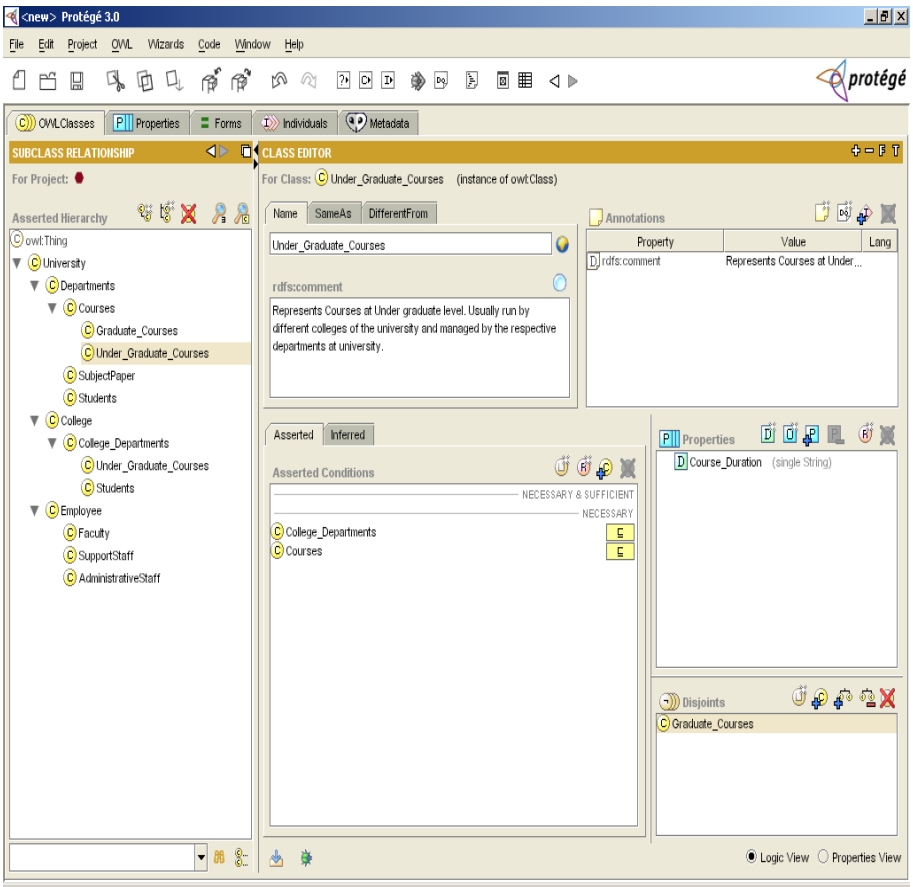


Figure 5 Protégé with OWL Plugin

```
<?xml version="1.0"?>
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   xmlns:owl="http://www.w3.org/2002/07/owl#"
   xmlns="http://www.owl-ontologies.com/unnamed.owl#"
  xml:base="http://www.owl-ontologies.com/unnamed.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Departments">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="University"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Graduate_Courses">
    <owl:disjointWith>
      <owl:Class rdf:ID="Under_Graduate_Courses"/>
    </owl:disjointWith>
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Courses"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:about="#Under_Graduate_Courses">
    <rdfs:subClassOf>
  <owl:Class rdf:about="#Courses"/>
    </rdfs:subClassOf>
    <owl:disjointWith rdf:resource="#Graduate_Courses"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Represents Courses at Under graduate level. Usually run by different colleges
of the university and managed by the respective departments at
university.</rdfs:comment>
    <rdfs:subClassOf>
      <owl:Class rdf:ID="College_Departments"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:about="#Courses">
    <rdfs:subClassOf rdf:resource="#Departments"/>
  </owl:Class>
  <owl:Class rdf:ID="Students">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#College_Departments"/>
    </rdfs:subClassOf>
    <rdfs:subClassOf rdf:resource="#Departments"/>
  </owl:Class>
```

Figure 6 Sample OWL code of University Ontology

The OWL Plug-in is an extension of Protégé. The OWL Plug-in as shown in Figure 2.5 can be used to edit OWL ontologies, to access description logic (DL) reasoners, and to acquire instances for semantic markup. As an extension of Protégé, the OWL Plug-in profits from the benefits of a large user community, a library of reusable components, and a flexible architecture. The OWL Plug-in therefore has the potential to become a standard infrastructure for building ontology-based Semantic Web applications. The OWL Plug-in is a complex Protégé extension which can be used to edit OWL files and databases. The OWL Plug-in includes a collection of custom-tailored tabs and widgets for OWL, and provides access to OWL-related services such as classification, consistency checking, and ontology testing. Figure 2.6 shows the sample code of a University Ontology.

Designing Ontology from Taxonomy

Ontology creation from the scratch is a complex process and is expensive, as many iterations are required to relate each and every concept with one another. Migration from taxonomy is relatively much easier and cost efficient proposition as taxonomies are already standardized. Ontology engineering requires in depth knowledge of the domain as well as expertise in building knowledge representation and organization techniques, a rare combination that can be found for any domain. However, due to the standardization of Ontology Web Language and tools such as Protégé 2.1, building ontologies have become significantly easier. But for domain experts, especially in the fields that are not much concerned with the computers and information technology, it is still harder to use these tools and standards. On the other hand, for the knowledge engineers, the problem of getting knowledge from the domain expert remains unsolved. In this section, efforts for devising an approach to create ontology from taxonomy for any knowledge domain are presented. This requires some knowledge source in the requisite domain and the existing standardized taxonomies are the right choice. Taxonomies are not just built by a single expert but are the result of sharing knowledge in the domain by many experts after investigating the field for years. Also, they provide hierarchical view of concepts of the domain that is required to build ontology. Protégé OWL Plug-in tool is used to demonstrate the implementation of generating Soil Ontology from Soil Taxonomy. Soil ontology in OWL, can be easily ported to any system or to any other ontology editor having OWL support. This makes it easy to integrate with other ontologies or agent based systems that make Semantic Web a realization. List of steps for designing the ontology from domain taxonomy are as follows:

1. Study the taxonomy and list the major entities in which the concepts are classified. Put them in the hierarchical format.
2. These entities will be the classes in the ontology in such a fashion that the first classification term will be the top class under owl:thing and the second term will be its first sub class and next heading will be the next sub class of the first sub class and so on.
3. Create top level classes of the concepts that appear under the top level classification term as a separate hierarchy under owl:thing class. Similarly, create other classes in a hierarchy corresponding to the concepts as they appear in the taxonomy hierarchy.
4. List different properties that are applicable to the instances or individuals of the classes defined in step 2. These properties are assigned appropriate values at the

time of creation of individuals. By default, these properties are inherited to the individuals of the classes that are derived from them.

5. Decide the range and data type of values for each property of the class. In OWL, one can have two types of properties Data type Property and Object property. The Data type property can have data types defined by XML Schema such as Boolean, Float, Integer, String, and Symbol values while Object property can point to other class or individual of the class.
6. Also, establish the rdfs:subClaassOf relationship as a necessary condition between the concept classes defined in step 3 and corresponding classification term classes defined in step 2.
7. Repeat Step 4 and Step 5 for all the concept classes.
8. Create individuals for all the sub classes of the ontology and assign values to the different properties for the individuals of different classes of the ontology.
9. Refine and enhance the ontology by adding appropriate restrictions and other annotation properties that are out of the scope of taxonomy.

Validation of Approach - Soil Ontology

By using the above approach, the Soil Ontology in Protégé with OWL plug-in is implemented. However, the approach is generic and can be implemented with any tool that provides support to implement ontology in OWL.

1) First step is to identify the hierarchy in which the soil taxonomy is classified. In Soil Ontology following is the hierarchy

   Order → Sub order → Great group → Sub group.

Table 1 Table shows few rows of Soil Taxonomy

Order	Sub order	Great group	Sub group
Alfisols	Aqualfs	Linthaqualfs	Typic Plinthaqualfs
		Natraqualfs	Vertic Natraqualfs
			AlbGlossicNatraqualfs
			Glossic Natraqualfs
			Mollic Natraqualfs
			Typic Natraqualfs
		Duraqualfs	Typic Duraqualfs
		Fragiaqualfs	Aeric Fragiaqualfs
			Plinthic Fragiaqualfs
			Umbric Fragiaqualfs
			Typic Fragiaqualfs
		Kandiaqualfs	Arenic Kandiaqualfs
			Grossarenic Kandiaqualfs
			Plinthic Kandiaqualfs
			Aeric Umbric Kandiaqualfs

2) Being Order is at the top of the hierarchy, the Order class is created as the base class from which other classes are derived. Sub order class is the first sub class, Great group class is the next sub class and finally sub group is last class of the hierarchy. As shown in Asserted Hierarchy Box of Figure 2.7.

3) Class Alfisols is created as a top level class under owl:thing class as it is one of the concepts that appear under the top level heading Order. Similarly, other classes are created in a hierarchy corresponding to the concepts as they appear in the taxonomy hierarchy.

4) Also, the rdfs:subClassOf relationship is established as a necessary condition between the concept classes defined in step 3 and corresponding classification term classes defined in step 2. For example : class Alfisols is a subclass of Order and class Aqualfs is a subclass of both Suborder class and Alfisols.

5) Then the different properties that are applicable for all the individuals/instances of the taxonomy are identified. Soil color, soil texture, soil surface form are such few properties that are applicable for all the individuals of the soil taxonomy.

6) The identified properties of the Order class are assigned proper data types as shown in Properties at Class box in Figure 2.7. In OWL there are two types of properties Data type Property and Object property. The Data type property can have all the values specified by datatypes in XML Schema like Boolean, Float, Integer, String, and Symbol values while Object property can point to class and instance or individual of the classes. These properties are created for each class through Properties tab of protégé as shown in Figure 2.7 on the right hand side section of the screen.

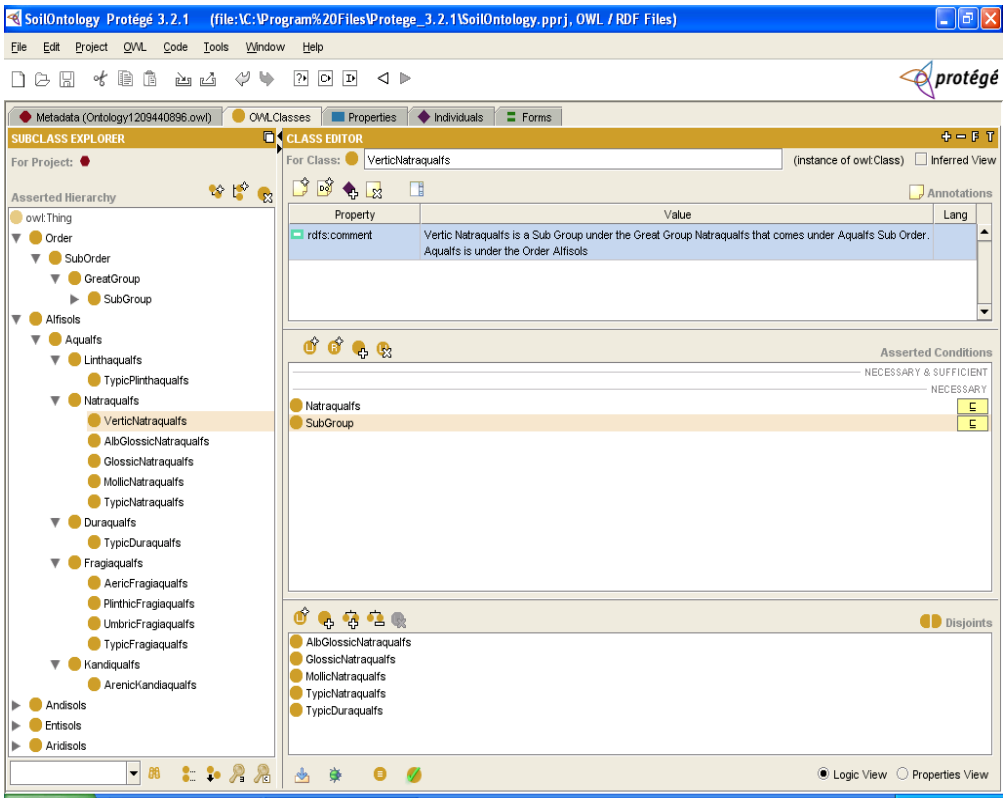


Figure 7 Class hierarchy of soil ontology in Protégé

7) Similarly, the relationship between all other classes corresponding to concepts and hierarchy classes are established.
8) After creating all the classes in the ontology, the individuals of the concept classes can be created. Also, the restrictions are put on the hierarchy classes so that there individuals can not be created. In the case of Soil Ontology, no individuals are created for the concept classes but one can create individuals of a class that is derived from some other domain classes and a concept class of Soil Ontology like a location class, which is derived from Area class and Alfisols Class. The individual in this case shows that a particular area has an Alfisols type of soils.
9) Protégé OWL plug-in stores the ontology in the OWL and code can be seen by clicking on the show OWL/RDF source code button on the main tool bar. The code can be edited manually to incorporate finer details if required.

Association Rules Mining using R

Anshu Bharadwaj
ICAR-IASRI, Library Avenue, New Delhi – 110 012
anshu.bharadwaj@icar.gov.in

1. Introduction

Mining association rules is one of the most useful data mining applications. Association rules, were first introduced in 1993 [Agrawal1993], and are used to identify relationships among a set of items in a database. These relationships are not based on inherent properties of the data themselves (as in the case of functional dependencies), but are rather based on co-occurrence of the data items. Association rules are mainly used to analyze transactional data. The association rules are useful in management, to increase the effectiveness and/or reduce the cost associated with advertising, marketing, inventory, stock location on the floor etc. Association rules also provide assistance in other applications such as prediction by identifying what events occur before a set of particular events. An association rule may be one of the following types: Boolean, Spatial, Temporal, Generalized, Quantitative, Interval, and Multiple Min-Support Association etc or a mix of them.

Formally the association rule as stated in [Agrawal1993] and [Cheung1996] is,

Let D be a transaction database and $I = \{I_1, I_2, \ldots, I_m\}$ be a set of m distinct items (attributes) of D, where each transaction (record) T is a set of items such that $T \subseteq I$ and has unique identifier. A transaction T is said to contain a set of item A if and only if $A \subseteq T$. An *association rule* is of the form of an implication expression $A \Rightarrow B$, where $A, B \subset I$, are sets of items called *itemsets*, and $A \cap B = \phi$. The rule $A \Rightarrow B$ holds in the transaction data D with *support (s*) where *s* is the ratio (in percent) of the records that contain $A \cup B$ (i.e. both A and B) to the total number of records in the database, i.e. the probability $P(A \cup B)$. The rule $A \Rightarrow B$ has *confidence* (*c*) in the D, the ratio (in percent) of the number of records that contain $A \cup B$ to the number of records that contain A. This is taken to be the conditional probability $P(B \mid A)$. Mining of association rules from a database consists of finding all rules that meet the user-specified thresholds of support and confidence termed as minimum support and minimum confidence. The problem of mining association rules has been decomposed into the following two subproblems [Agrawal1994]:

1) To find all sets of items which occur with a frequency that is greater than or equal to the user-specified threshold support, say *s*.
2) To generate the rules using the frequent itemsets, which have confidence greater than or equal to the user-specified threshold confidence, say c.

The Association relationships are not based on inherent properties of the data themselves but rather based on co-occurrence of the data items. Application of association rules spans across a wide range of domains such as, business, finance, health, geographical information system, weather forecast and many such areas of real life application. The association rules in management may be handy to increase the effectiveness and/or reduce the cost associated with advertising, marketing, inventory, stock location on the floor etc.

Association rules could assist in prediction of an event co-occurrence of a set of events. Association rules are generally categorized in following types: Boolean, Spatial, Temporal, Generalized, Quantitative, and Interval or may be mixed of them. The above definition of association rule is also known as Boolean Association Rule.

Association rule mining is:

- Unsupervised learning
- Used for pattern discovery
- Each rule has form: A -> B, or Left -> Right

For example: "70% of customers who purchase 2% milk will also purchase whole wheat bread."

Data mining using association rules is the process of looking for strong rules:

1. Find the large itemsets (i.e. most frequent combinations of items)
2. Generate association rules for the above itemsets.

2. Performance Evaluation Measure of Association Rules

How to measure the strength of an association rule? Using support/confidence

Support: Support shows the frequency of the patterns in the rule; it is the percentage of transactions that contain both A and B, i.e.

Support = Probability(A and B)

Support = (# of transactions involving A and B) / (total number of transactions).

Confidence: Confidence is the strength of implication of a rule; it is the percentage of transactions that contain B if they contain A, ie.

Confidence = Probability (B if A) = P(B/A)

Confidence =

(# of transactions involving A and B) / (total number of transactions that have A).

3. The Apriori Algorithm

The Apriori Algorithm is an influential algorithm for miningfrequent itemsets for boolean association rules. Some keyconcepts for Apriori algorithm are:

- Frequent Itemsets: The sets of item which hasminimum support (denoted by Li for ith-Itemset).
- Apriori Property: Any subset of frequent itemset mustbe frequent.
- Join Operation: To find Lk , a set of candidate kitemsets is generated by joining Lk-1 with itself.

Very first algorithm proposed for association rules miningwas the Apriori for frequent itemset mining. The mostpopular algorithm for pattern mining is without a doubt Apriori.It is designed to be applied on a transaction database todiscover patterns in transactions made by customers in stores.But it can also be applied in several other applications. Atransaction is defined a set of distinct items (symbols).
Aprioritakes as input

(1) a minsup threshold set by the user and
(2) atransaction database containing a set of transactions.

Aprioritoutputs all frequent itemsets, i.e. groups of items shared by noless than minsup transactions in the input database. Forexample, consider the following transaction databasecontaining four transactions. Given a minsup of twotransactions, frequent itemsets are“bread, butter”, “breadmilk”, “bread”, “milk” and “butter”.

T1: bread, butter, spinach
T2: butter, salmon
T3: bread, milk, butter
T4: cereal, bread, milk

The Apriori algorithm employs the downward closureproperty if an item set is not frequent, any superset of it cannotbe frequent either. The Apriori algorithm performs a breadthfirstsearch in the search space by generating candidate k+1-itemsets from frequent k itemsets.

The frequency of an item set is computed by counting itsoccurrence in each transaction. Apriori is an significantalgorithm for mining frequent itemsets for Boolean associationrules. Since the Algorithm uses prior knowledge of frequentitem set it has been given the name Apriori. Apriori is aniterative level wise search Algorithm, where k-itemsets areused to explore (k+1)-itemsets. First, the set of frequents 1- itemsets is found.

This set is denoted by L1. L1 is used to find L2, the set offrequent 2-itemsets , which is used to find L3 and so on , untilno more frequent k-itemsets can be found. The finding of eachLk requires one full scan of database.

There are twosteps for understanding that how Lk-1 is usedto find Lk:-

1) The join step: To find Lk , a set of candidate k-itemsets isgenerated by joining Lk-1 with itself. This set ofcandidates is denoted Ck.
2) The prune step: Ck is a superset of Lk , that is , itsmembers may or may not be frequent , but all of thefrequent k-itemsets are included in Ck .

A scan of the database to determine the count of eachcandidate in Ck would result in the determination of Lk. Ck,however, can be huge, and so this could involve heavycomputation.

To reduce the size of Ck , the Apriori property is used as follows:

i. Any (k-1)-item set that is not frequent cannot be asubset of frequent k-item set.

ii. Hence, if (k-1) subset of a candidate k item set is notin Lk-1 then the candidate cannot be frequent eitherand so can be removed from C.

Based on the Apriori property that all subsets of a frequentitemset must also be frequent, we can determine that four lattercandidates cannot possibly be frequent. How?

For example, let's take {I1, I2, I3}. The 2-item subsets of itare {I1, I2}, {I1, I3} & {I2, I3}. Since all 2-item subsets of {I1,I2, I3} are members of L2, We will keep {I1, I2, I3} in C3.

Let's take another example of {I2, I3, I5} which shows howthe pruning is performed. The 2-item subsets are {I2, I3}, {I2,I5} & {I3,I5}.

BUT, {I3, I5} is not a member of L2 and hence it is notfrequent violating Apriori Property. Thus, we will have toremove {I2, I3, I5} from C3.

Therefore, C3 = {{I1, I2, I3}, {I1, I2, I5}} after checking forall members of result of Join operation for Pruning.

Example : The Titanic Dataset

The Titanic dataset in the datasets package is a 4-dimensional table with summarized information on the fate of passengers on the Titanic according to social class, sex, age and survival. I To make it suitable for association rule mining, we reconstruct the raw data as titanic.raw, where each row represents a person. The reconstructed raw data can also be downloaded at http://www.rdatamining.com/data/titanic.raw.rdata.

```
> str(titanic.raw)
'data.frame': 2201 obs. of 4 variables:
$ Class : Factor w/ 4 levels "1st","2nd","3rd",..: 3 3 3 3 3 3 3 3 3 3 ...
$ Sex : Factor w/ 2 levels "Female","Male": 2 2 2 2 2 2 2 2 2 2 ...
$ Age : Factor w/ 2 levels "Adult","Child": 2 2 2 2 2 2 2 2 2 2 ...
$ Survived: Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...
```

Association Rule Mining

```
> library(arules)
> # find association rules with default settings
> rules <- apriori(titanic.raw)
> inspect(rules)
  lhs             rhs        support   confidence lift
1 {}           => {Age=Adult} 0.9504771 0.9504771  1.0000000
2 {Class=2nd}    => {Age=Adult} 0.1185825 0.9157895  0.9635051
3 {Class=1st}    => {Age=Adult} 0.1449341 0.9815385  1.0326798
4 {Sex=Female}   => {Age=Adult} 0.1930940 0.9042553  0.9513700
5 {Class=3rd}    => {Age=Adult} 0.2848705 0.8881020  0.9343750
6 {Survived=Yes} => {Age=Adult} 0.2971377 0.9198312  0.9677574
7 {Class=Crew}   => {Sex=Male}  0.3916402 0.9740113  1.2384742
```

...

We then set rhs=c("Survived=No", "Survived=Yes") in appearance to make sure that only "Survived=No" and "Survived=Yes" will appear in the rhs of rules.

```
> # rules with rhs containing "Survived" only
> rules <- apriori(titanic.raw,
  + parameter = list(minlen=2, supp=0.005, conf=0.8),
  + appearance = list(rhs=c("Survived=No", "Survived=Yes"),
  + default="lhs"),
  + control = list(verbose=F))
> rules.sorted <- sort(rules, by="lift")
> inspect(rules.sorted)
```

```
    lhs              rhs                 support confidence     lift
 1  {Class=2nd,
     Age=Child}  => {Survived=Yes} 0.010904134  1.0000000 3.095640
 2  {Class=2nd,
     Sex=Female,
     Age=Child}  => {Survived=Yes} 0.005906406  1.0000000 3.095640
 3  {Class=1st,
     Sex=Female} => {Survived=Yes} 0.064061790  0.9724138 3.010243
 4  {Class=1st,
     Sex=Female,
     Age=Adult}  => {Survived=Yes} 0.063607451  0.9722222 3.009650
 5  {Class=2nd,
     Sex=Female} => {Survived=Yes} 0.042253521  0.8773585 2.715986
 6  {Class=Crew,
     Sex=Female} => {Survived=Yes} 0.009086779  0.8695652 2.691861
 7  {Class=Crew,
     Sex=Female,
     Age=Adult}  => {Survived=Yes} 0.009086779  0.8695652 2.691861
 8  {Class=2nd,
     Sex=Female,
     Age=Adult}  => {Survived=Yes} 0.036347115  0.8602151 2.662916
 9  {Class=2nd,
     Sex=Male,
     Age=Adult}  => {Survived=No}  0.069968196  0.9166667 1.354083
 10 {Class=2nd,
     Sex=Male}   => {Survived=No}  0.069968196  0.8603352 1.270871
 11 {Class=3rd,
     Sex=Male,
     Age=Adult}  => {Survived=No}  0.175829169  0.8376623 1.237379
 12 {Class=3rd,
     Sex=Male}   => {Survived=No}  0.191731031  0.8274510 1.222295
```

Pruning Redundant Rules

In the above result, rule 2 provides no extra knowledge in addition to rule 1, since rules 1 tells us that all 2nd-class children survived. Generally speaking, when a rule (such as rule 2) is a super rule of another rule

(such as rule 1) and the former has the same or a lower lift, the former rule (rule 2) is considered to be redundant. Below we prune redundant rules.

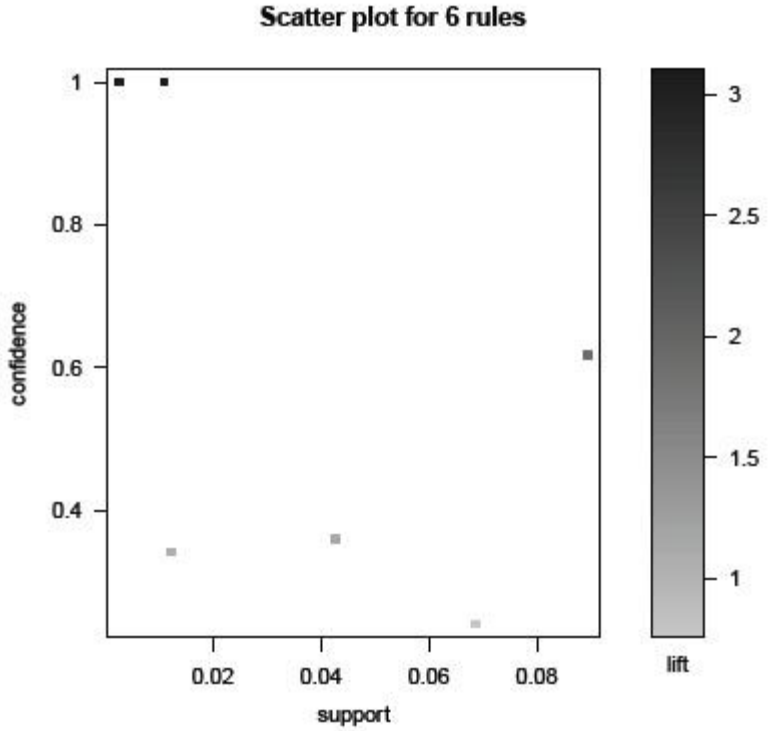
```
> # find redundant rules
> subset.matrix <- is.subset(rules.sorted, rules.sorted)
> subset.matrix[lower.tri(subset.matrix, diag=T)] <- NA
> redundant <- colSums(subset.matrix, na.rm=T) >= 1
> which(redundant)
[1] 2 4 7 8
> # remove redundant rules
> rules.pruned <- rules.sorted[!redundant]
> inspect(rules.pruned)
```

```
  lhs                     rhs                support confidence     lift
1 {Class=2nd,
   Age=Child}          => {Survived=Yes} 0.010904134  1.0000000 3.095640
2 {Class=1st,
   Sex=Female}         => {Survived=Yes} 0.064061790  0.9724138 3.010243
3 {Class=2nd,
   Sex=Female}         => {Survived=Yes} 0.042253521  0.8773585 2.715986
4 {Class=Crew,
   Sex=Female}         => {Survived=Yes} 0.009086779  0.8695652 2.691861
5 {Class=2nd,
   Sex=Male,
   Age=Adult}          => {Survived=No}  0.069968196  0.9166667 1.354083
6 {Class=2nd,
   Sex=Male}           => {Survived=No}  0.069968196  0.8603352 1.270871
7 {Class=3rd,
   Sex=Male,
   Age=Adult}          => {Survived=No}  0.175829169  0.8376623 1.237379
8 {Class=3rd,
   Sex=Male}           => {Survived=No}  0.191731031  0.8274510 1.222295
```

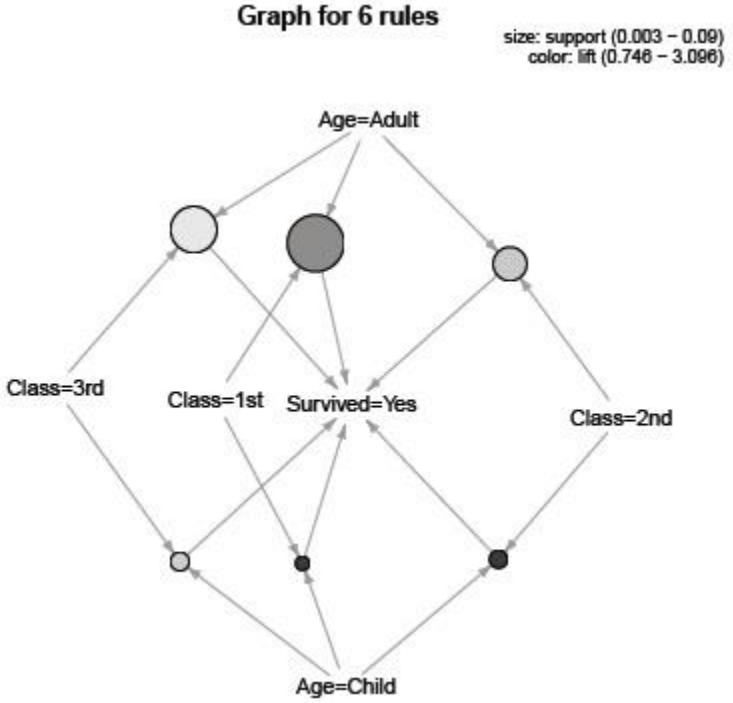
Visualizing Association Rules

Package arules Viz supports visualization of association rules with scatter plot, balloon plot, graph, parallel coordinates plot, etc.

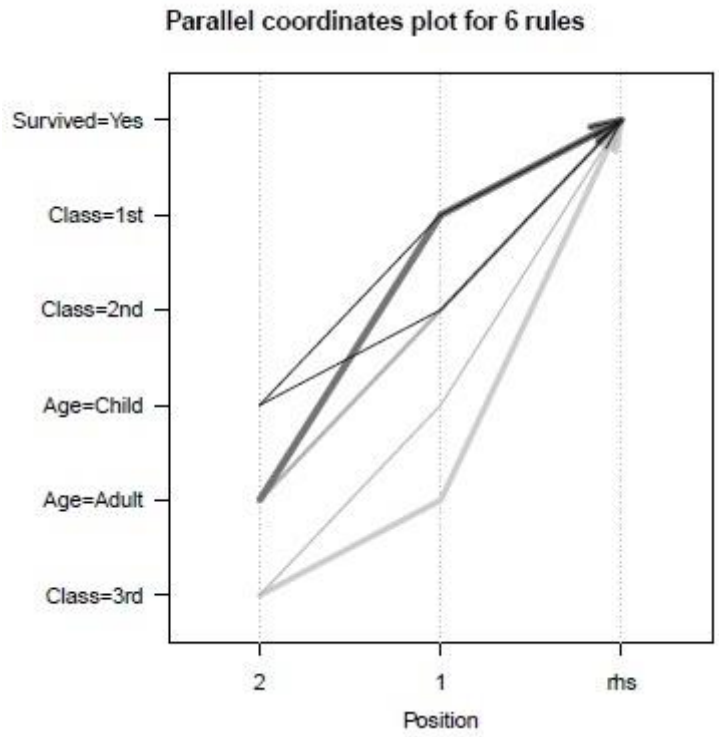
```
> library(arulesViz)
> plot(rules)
```




```
> plot(rules, method="graph", control=list(type="items"))
```




```
> plot(rules, method="paracoord", control=list(reorder=TRUE))
```



4. Frequent Pattern (FP) Growth Method

The FP-Growth Algorithm is an alternative way to find frequent itemsets without using candidate generations, thus improving performance. For so much it uses a divide-and-conquer strategy. The core of this method is the usage of a special data structure named frequent-pattern tree (FP-tree), which retains the itemset association information.

In simple words, this algorithm works as follows: first it compresses the input database creating an FP-tree instance to represent frequent items. After this first step it divides the compressed database into a set of conditional databases, each one associated with one frequent pattern. Finally, each such database is mined separately. Using this strategy, the FP-Growth reduces the search costs looking for short patterns recursively and then concatenating them in the long frequent patterns, offering good selectivity.

In large databases, it's not possible to hold the FP-tree in the main memory. A strategy to cope with this problem is to firstly partition the database into a set of smaller databases (called projected databases), and then construct an FP-tree from each of these smaller databases.

4.1 FP-Tree structure

The frequent-pattern tree (FP-tree) is a compact structure that stores quantitative information about frequent patterns in a database [4].

Han defines the FP-tree as the tree structure io below [1]:

1. One root labeled as "null" with a set of item-prefix subtrees as children, and a frequent-item-header table (presented in the left side of Figure 1);
2. Each node in the item-prefix subtree consists of three fields:

    1. Item-name: registers which item is represented by the node;

2. Count: the number of transactions represented by the portion of the path reaching the node;
3. Node-link: links to the next node in the FP-tree carrying the same item-name, or null if there is none.

1. Each entry in the frequent-item-header table consists of two fields:

    1. Item-name: as the same to the node;
    2. Head of node-link: a pointer to the first node in the FP-tree carrying the item-name.

Additionally the frequent-item-header table can have the count support for an item. The Figure below show an example of a FP-tree.

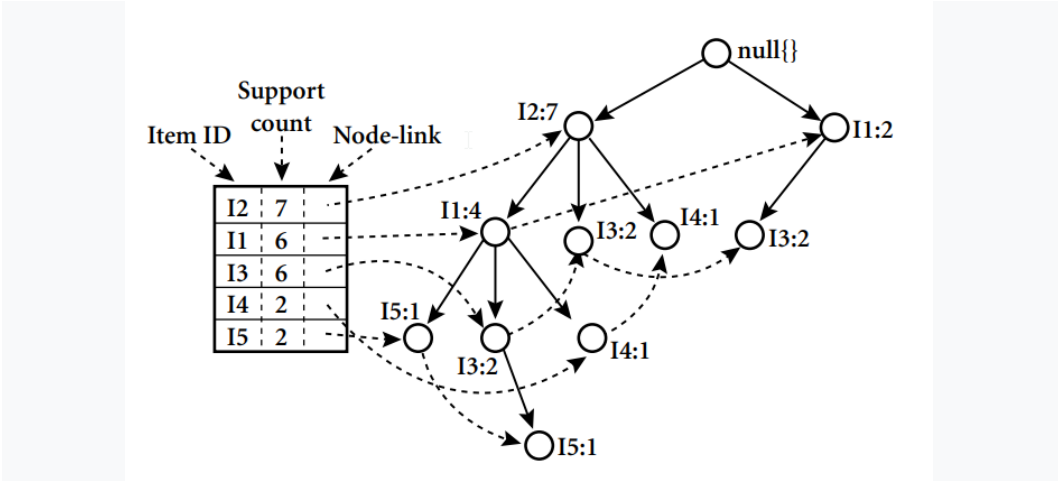


Figure 1: An FP-tree registers compressed, frequent pattern information

Table 1: Transactional data for an AllElectronics branch.

TID	*List of item_IDs*
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

The first scan of the database is the same as Apriori, which derives the set of frequent items (1-itemsets) and their support counts (frequencies). Let the minimum support count be 2. The set of frequent items is sorted in the order of descending support count. This resulting set or list is denoted by L. Thus, we have L ={{I2: 7}, {I1: 6}, {I3: 6}, {I4: 2}, {I5: 2}}. An FP-tree is then constructed as follows. First, create the root of the tree, labeled with “null.” Scan database D a second time. The items in each transaction are processed inL order (i.e., sorted according to descending support count), and a branchis

created for each transaction. For example, the scan of thefirst transaction, “T100: I1, I2, I5,” which contains three items (I2, I1, I5 in L order), leads to the construction of the first branch of the tree with three nodes,hI2: 1i,hI1:1i, and hI5: 1i, where I2islinked as a child to the root, I1islinked to I2, and I5islinked to I1. The second transaction, T200, contains theitems I2 and I4inLorder, whichwould result in a branch where I2 is linked to the root and I4 is linked to I2. However, this branch would share a common prefix, I2, with the existing path for T100. Therefore, we insteadincrement the count of the I2 node by 1, and create a new node,hI4: 1i, which is linked as a child to hI2: 2i. In general, when considering the branch to be addedfor a transaction, the count of each node along a common prefix is incremented by 1, and nodes for the items following the prefix are created and linked accordingly. To facilitate tree traversal, an item header table is built so that each item points to its occurrences in the tree via a chain of node-links. The tree obtained after scanning all of the transactions is shown in Figure 6.7 with the associated node-links. In this way, the problem of mining frequent patterns in databases is transformed to that of mining the FP-tree. The FP-tree is mined as follows. Start from each frequent length-1 pattern (as an initial suffix pattern), construct its conditional pattern base (a “sub-database,” which consists of the set of prefix paths in the FP-tree co-occurring with the suffix pattern), then construct its (conditional) FP-tree, and perform mining recursively on such a tree. The pattern growth is achieved by the concatenation of the suffix pattern with the frequent patterns generated from a conditional FP-tree.

Mining of the FP-tree is summarized in Table 2 and detailed as follows. We first consider I5, which is the last item in L, rather than the first. The reason for starting at the end of the list will become apparent as we explain the FP-tree mining process. I5 occurs in two branches of the FP-tree of Figure 2. (The occurrences of I5 can easily be found by following its chain of node-links.) The paths formed by these branches are hI2, I1, I5: 1i and hI2, I1, I3, I5: 1i. Therefore, considering I5 as a suffix, its corresponding two prefix paths are hI2, I1: 1i and hI2, I1, I3: 1i, which form its conditional pattern base. Using this conditional pattern base as a transaction database, we build an I5-conditional FP-tree, which contains only a single path, hI2: 2, I1: 2i; I3 is not included because its support count of 1 is less than the minimum support count. The single path generates all the combinations of frequent patterns: {I2, I5: 2}, {I1, I5: 2}, {I2, I1, I5: 2}. For I4, its two prefix paths form the conditional pattern base, {{I2 I1: 1}, {I2: 1}}, which generates a single-node conditional FP-tree, hI2: 2i, and derives one frequent pattern, {I2, I4: 2}.

Table 2: Mining the FP-tree by creating conditional (sub-)pattern bases

Item	*Conditional Pattern Base*	*Conditional FP-tree*	*Frequent Patterns Generated*
I5	{{I2, I1: 1}, {I2, I1, I3: 1}}	⟨I2: 2, I1: 2⟩	{I2, I5: 2}, {I1, I5: 2}, {I2, I1, I5: 2}
I4	{{I2, I1: 1}, {I2: 1}}	⟨I2: 2⟩	{I2, I4: 2}
I3	{{I2, I1: 2}, {I2: 2}, {I1: 2}}	⟨I2: 4, I1: 2⟩, ⟨I1: 2⟩	{I2, I3: 4}, {I1, I3: 4}, {I2, I1, I3: 2}
I1	{{I2: 4}}	⟨I2: 4⟩	{I2, I1: 4}

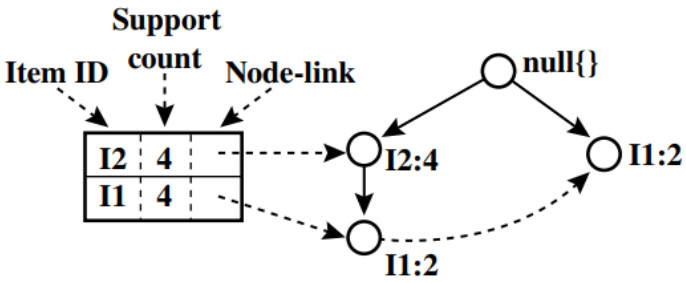


Figure 2: The conditional FP-tree associated with the conditional node I3

Similar to the above analysis, I3's conditional pattern base is {{I2, I1: 2}, {I2: 2}, {I1: 2}}. Its conditional FP-tree has two branches, hI2: 4, I1: 2i and hI1: 2i, as shown in Figure 6.8, which generates the set of patterns {{I2, I3: 4}, {I1, I3: 4}, {I2, I1, I3: 2}}. Finally, I1's conditional pattern base is {{I2: 4}}, whose FP-tree contains only one node, hI2: 4i, which generates one frequent pattern, {I2, I1: 4}. This mining process is summarized in Figure 6.9. The FP-growth method transforms the problem of finding long frequent patterns to searching for shorter ones in much smaller conditional databases recursively and then concatenating the suffix. It uses the least frequent items as a suffix, offering good selectivity. The method substantially reduces the search costs.

5. Basic Association Rules: Problems, Solutions and New Applications

Most of the research efforts in the scope of association rules have been oriented to simplify the rule set and to improve performance of algorithm. But these are not the only problems that can be found and when rules are generated and applied in different domains. Troubleshooting for them should also take into consideration the purpose of association model and data they come from. Some of the major drawbacks of association rule algorithms are as follows:

- Obtaining huge number of rules
- Obtaining non interesting rules
- Low algorithm performance
- Cannot incorporate domain/ user defined knowledge
- Not suitable for supervised learning

Some of the recent studies have focused on overcoming these limitations. Many algorithms for obtaining a reduced number of rules with high **support** and **confidence** have been produced. However these measures are insufficient to determine if discovered associations are really useful. An important property of discovered association rules is that they should be **interesting** and **useful**. Though interestingness of rule is a subjective aspect, many researchers have tried to come up with some ways of **measuring of interest**. It has been suggested that the rules are interesting if they are **unexpected** (unknown to user) and **actionable** (users can do something with them to their advantage). Further some other measures namely: **any-confidence, all confidence and bond** has been suggested as alternative measures of interestingness. Some authors have considered alternative measures of interest as : **gini index, entropy gain or chisquared for database or a measure of implication called conviction.** Most of the approaches for finding interesting rules require user participation to articulate his knowledge or to express what rules are interesting for him. Systems have been developed to analyze the discovered rules against

user's knowledge. Discovered rules can be pruned to remove redundant and insignificant rules and further user's evaluation can be used to rank the rules. Unexpected patterns discovered may represent "holes" in domain knowledge which needs to be resolved. These patterns can thus be used to refine already existing beliefs.

Traditionally, association analysis has been considered as an unsupervised technique, so it has been applied for knowledge discovery tasks. Recent studies have shown that knowledge discovery algorithms such as association rule mining can be successfully applied for prediction in classification problems. In such cases the algorithms used for generating association rules must be tailored to peculiarities of predictions in order to build effective classifiers. Some work has been done, where association mining algorithms have been extended so that they can be used for classification/ prediction. A proposal of this category is Classification Based on Association (CBA) algorithm. The algorithm consists of two parts, a rule generator for finding association rules and a classifier builder based on these rules. Main contribution of this algorithm is possibility of making prediction on any attribute in database. Moreover, new incomplete observations can be classified.

In conclusion we can say that association rule mining is an important area of data mining research and a comparatively a younger member of data mining community. In addition to finding co-occurrence relation between items, which is basic objective, the algorithm has been applied for diverse applications. Many extensions of standard methods have been proposed. A major research area on association rules is interestingness of discovered rules. In fact its potential has still to be tapped, so that it can be tailored to solve different types of data mining problems.

Text Mining and Natural Language Processing (NLP)

Sanchita Naha
ICAR-IASRI, Library Avenue, New Delhi – 110 012
sanchita.naha@icar.gov.in

1. Text Mining

Mining refers to extraction of meaningful information, discovery of hidden patterns from data. When this technique is used to extract hidden patterns in natural language text data, it is called Text mining [Hearst, 2003]. Primary goal of text mining is to discover interesting patterns from unstructured text data in a dynamic and scalable way. It is a process of transforming unstructured data in a structured format to be analysed in traditional ways. Increasing amounts of text data generating from different applications such as emails, WhatsApp chats, social media applications, search engine queries etc. has created the need for analysing these data.

2. Text Mining Process:

- **Data Acquisition:** Text data collection is the process of gathering textual data from a variety of sources. In order to build intelligent applications capable of understanding human language, machine learning models need to train with large amounts of structured text data. Gathering sufficient text data is the first step in solving any language-based machine learning problem.
- **Pre-processing:** Data pre-processing is a most important task in Text mining. With irrelevant and redundant information or noise present in the data, knowledge discovery during the training phase will be more difficult. Data pre-processing specific to text mining includes lower casing, removal of HTML tags, emoticons, stop words, punctuations, lemmatization, tokenization, normalization etc.
- **Text Transformation:** Algorithmic techniques cannot be directly applied to text data that's why this needs to be represented in a structured format such as numeric vectors with frequency of words or other patterns of their occurrence in a document. These kind of transformations are also called text vectorizations e.g. Bag of Words, TF IDF, word2vec etc.
- **Feature Selection:** Feature selection is used to characterize each given document according to the task at hand. It is done by selecting the sub set of relevant variables to be used in the creation of the model.

- **Pattern Mining:** Pattern mining techniques are used to find patterns in the text data, such as frequent item sets, closed frequent item sets, co-occurring terms. In case of document classification tasks, it finds out the similar pattern in all the example data values coming under one class label.
- **Evaluate:** After pattern discovery, the learned models are evaluated with unknown example data which were not present at the time of model development.

3. Practice Areas of Text Mining:

Different text mining areas are in different maturity stages [Miner *et al.*, 2012]. While some areas are emerging some of them are easily accessible by practitioners via commercial softwares. The following seven practicing areas of text mining are most easily relatable to available technologies.

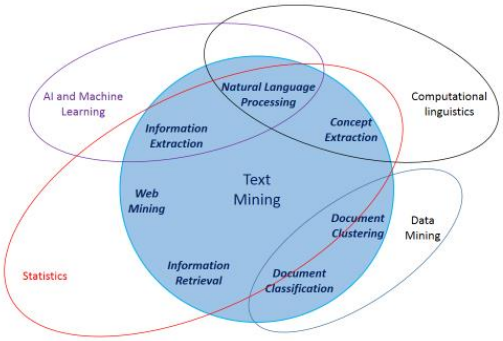


Figure 1: Venn diagram depicting seven major practicing areas of Text Mining and their intersection with other related fields such as AI, computational linguistics, statistics, machine learning and data mining.

3.1. Search and Information Retrieval (IR): It is regarded as the process of obtaining relevant information resources such as documents or web pages from a collection of similar resources which are against a search query posed by a user e.g. web search engines.

3.2. Document Clustering: Grouping and categorizing terms, paragraphs or documents using data mining clustering methods. Clustering algorithms in computational text analysis groups documents into grouping a set of text, called subsets or clusters where the algorithm's goal is to create internally coherent clusters that are distinct from one another [Carpineto et al., 2009] e.g. Topic extraction.

3.3. Document Classification: Document classification is the task of assigning a document one or more classes or categories, based on models trained on labelled examples e.g. Language Detection, Sentiment Analysis.

3.4. Web Mining: Data and text mining on the Internet, with a specific focus on the scale and interconnectedness of the web. The main purpose of web mining is discovering useful information from the World-Wide Web and its usage patterns.

3.5. Information Extraction: Identification and extraction of relevant facts and relationships from unstructured text; the process of making structured data from unstructured and semi structured text e.g. keyword extraction, Named entity recognition etc.

3.6. Natural Language Processing (NLP): It is a branch of AI that deals with the interaction between computers and humans using the natural language. The ultimate objective of NLP is to read, decipher, understand, and make sense of the human languages in a manner that is valuable. Broadly NLP is divided into Natural Language Understanding (NLU) and Natural Language Generation (NLG).

3.7. Concept Extraction: Grouping of words and phrases into semantically similar groups.

4. Natural Language Processing (NLP): NLP is a branch of Artificial Intelligence that deals with the interaction between computers and humans using the natural language [Garbade, 2018]. The ultimate objective of NLP is to read, decipher, understand, and make sense of the human languages in a manner that is valuable.

4.1. Everyday use of NLP:

- Faster typing with auto complete and predictive typing. Auto complete helps you to complete the rest of the word while predictive typing suggest the next word in the sentence.
- While composing an email or editing any word document, spell checkers and grammar checkers ensure that we use punctuations correctly or gives alert if any wrong words are used.
- Using NLP, features like auto correct finds the right search keyword if any word is misspelled or any word is used mistakenly for making search queries. This eventually provides better search results.
- Emails those pass through traditional spam filter, NLP differentiates between different types of email spams.

5. Text Mining vs. NLP: Text mining is a process where to attain certain specific goals, sometimes NLP is used, although NLP has the ability to achieve more. So first of all text mining and NLP has different goals. At the same time, there is indeed some overlaps between the two

techniques. Text mining does not consider the structure and context of the data while NLP does so, additionally NLP considers the latent metadata.

Text mining answers questions like frequency count of words, length of the sentence, and presence/absence of certain words while NLP is more concerned with the grammatical part of speeches and the lexical relations among them.

6. Text Mining Applications:

- **Customer service:** Customer surveys, online reviews, and social media profiles combined with text analytics tools enable companies to improve customer experience with speed. Text mining and sentiment analysis can provide a mechanism for companies to prioritize key points for their customers, allowing businesses to respond to urgent issues in real-time and increase customer satisfaction.
- **Risk management:** Text mining has applications in risk management, where it can provide insights around industry trends and financial markets by monitoring shifts in sentiment and by extracting information from analyst reports. This is particularly valuable to banking institutions as this data provides more confidence when considering business investments across various sectors.
- **Healthcare:** Text mining techniques have been increasingly valuable to researchers in the biomedical field, particularly for clustering information. Manual investigation of medical research can be costly and time-consuming; text mining provides an automation method for extracting valuable information from medical literature.
- **Spam filtering:** Spam frequently serves as an entry point for hackers to infect computer systems with malware. Text mining can provide a method to filter and exclude these e-mails from inboxes, improving the overall user experience and minimizing the risk of cyber-attacks to end users.

7. References:

Carpineto, C., Osiński, S., Romano, G., & Weiss, D. (2009). A survey of web clustering engines. ACM Computing Surveys (CSUR), 41(3), 1-38.

Garbade, J, M. (2018). 'A simple introduction to natural language processing', *becominghuman.ai,* 15th October. Available at: https://becominghuman.ai/a-simple-introduction-to-natural-language-processing-ea66a1747b32

Hearst, M. (2003). What is text mining. Summer Institute for the Mathematical Sciences, UC Berkeley, 5.

Miner, G., Delen, D., Elder, J., Fast, A., Hill, T., & Nisbet, R. A. (2012). The Seven Practice Areas of Text Mining. Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications, 29-41.

8. Books:

Aggarwal, C. C., & Zhai, C. (2012). *Mining text data*. Springer Science & Business Media.

Jurafsky, D. (2000). *Speech & language processing*. Pearson Education India.

Machine Learning Techniques – ANN and SVM

Chandan Kumar Deb
ICAR-IASRI, Library Avenue, New Delhi – 110 012
chandan.deb@icar.gov.in

1 Introduction

Artificial Intelligence (AI) have touched almost every aspect of today's lives. Scientific community is majorly shifting towards the use of AI in respective fields. A deep insight of data is the key tool of AI in terms of imposition of AI techniques in a particular field. The field of insight extraction from data called "data mining". The term "machine leaning" broadly comes under data mining where machines learn from data. Now learning process is not confined only to intelligent living organism.

Solving computational problem with analytical speed was always easier for machine as compared to humans, but computer was pretty much incompetent to day- to -day job. A child can take only fraction of seconds to identify his mother. An apple can be identified by a human in any form i.e. red colour, green colour, or in fruit salad. A human child takes only few years to completely learn a language. By the conventional computing i.e. hard computing, it is not possible to capture all the situation only by using if and else condition. Algorithms tells machine how to learn. Machine learns by its own, that leads us to avoid explicit or hard computing. So we have started usage of soft computing i.e. no need to hard code all the situations (which is practically impossible). This learning process of machine is referred to as "machine learning".

Machine Learning is the science (and art) of programming computers so it can learn from data. Here is a slightly more general definition: "It is a field of study that gives computers the ability to learn without being explicitly programmed".—Arthur Samuel, 1959

Engineering definitions

"A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E."—Tom Mitchell, 1997

Machine can also improve the performance when it will encounter new data.

2 Diurnal activities influenced by machine learning

Machine learning is no more a futuristic science fiction, it is already in our daily life and influence many aspects of our life. If one uploads a photo in Facebook, it can identify us and our friends on the basis of previous data and suggests us to tag them. Some of the friend of friends is also being suggested by Facebook on the basis of our account category and behavioural patterns of our account. Pinterest, Snapchat, Twitter, Instagram has the capability to segregate the image as well as the emotions by machine learning techniques. Spam filter, Email classification, Smart replies to the mail has become automated due to machine learning. The money is also safe due to fraud detection and prevention, credit decisions and medical diagnosis. The core of machine learning is the "data". From the data the machine attempts to identify the pattern. Data is the key point that make it possible get the learning process feasible.

3 Types of learning

Machine learning techniques includes a vast range of algorithms that broadly solve four categories of problem i.e. supervised, semi-supervised, unsupervised and reinforcement learning.

- **Supervised learning**: The supervised learning is a learning process where every learning steps done through a supervision. The process of classification comes under this category. Some of the machine learning algorithms that follows the supervised learning principal are *k*-nearest neighbors (KNN), linear regression, logistic regression, support vector machines (SVMs), decision trees and random forests.

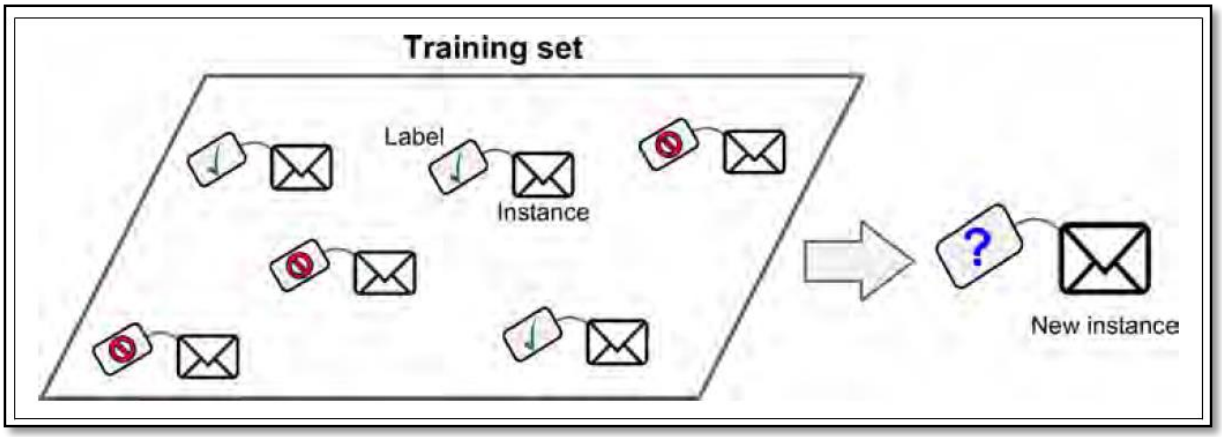


Figure 1: Example of supervised learning process where the spam and non-spam mail tagged with label of spam or non-spam, assists machine to learn from the supervised data and to classify a new mail.

Analogy: It can be explained by the following example. Suppose a teacher is taking practical class of forest tree identification by taking all the student in the field and showing a particular tree with its name. Then he asks the students to identify a tree and then he evaluates individual students.

Examples: Facebook can automatically find the person in the photo uploaded and suggests to tag them.

- **Semi-supervised learning:** These types of algorithms can deal with unlabelled data, partially labelled training data, and a little bit of labelled data. Most of the semi supervised learning is a combination of supervised and unsupervised learning.

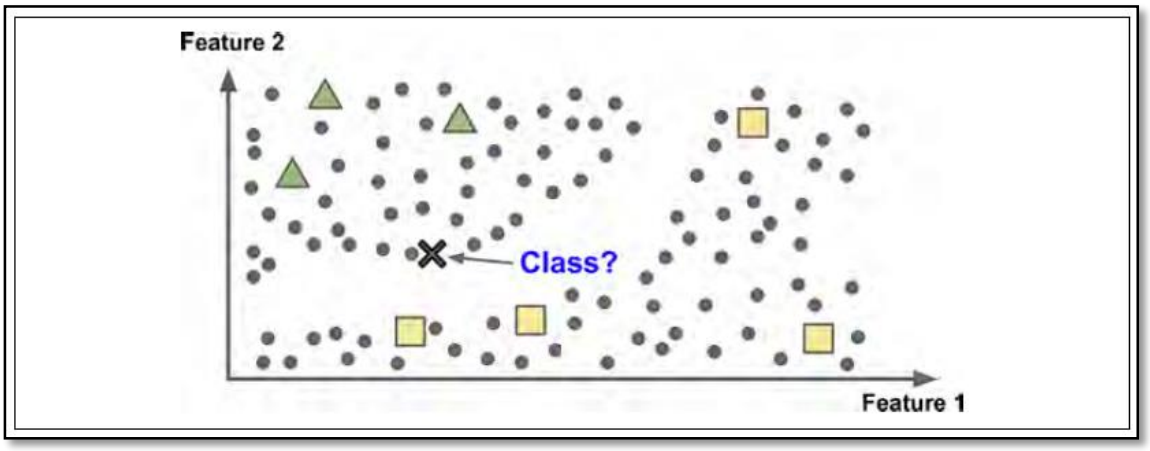


Figure 2: Picture depicting semi supervised learning.

Examples: Google photos asks only a few person to identify by user in a family and remaining photos can automatically be analysed and the person be identified in a particular account.

- **Unsupervised learning**: In unsupervised learning process no supervision is done to make the learner or machine to learn. Unsupervised learning does not require any supervision. Training data is also unlabelled.

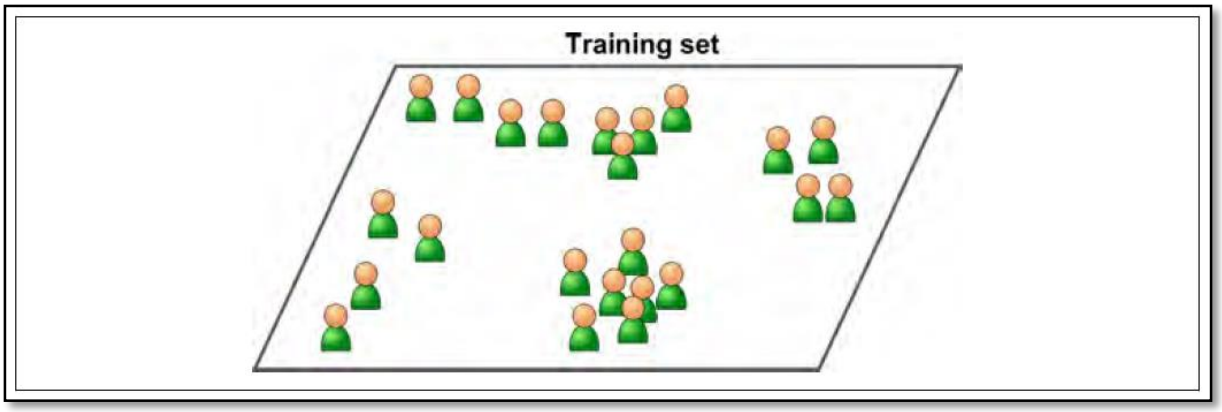


Figure 3: Picture depicting training data with unlabelled data points

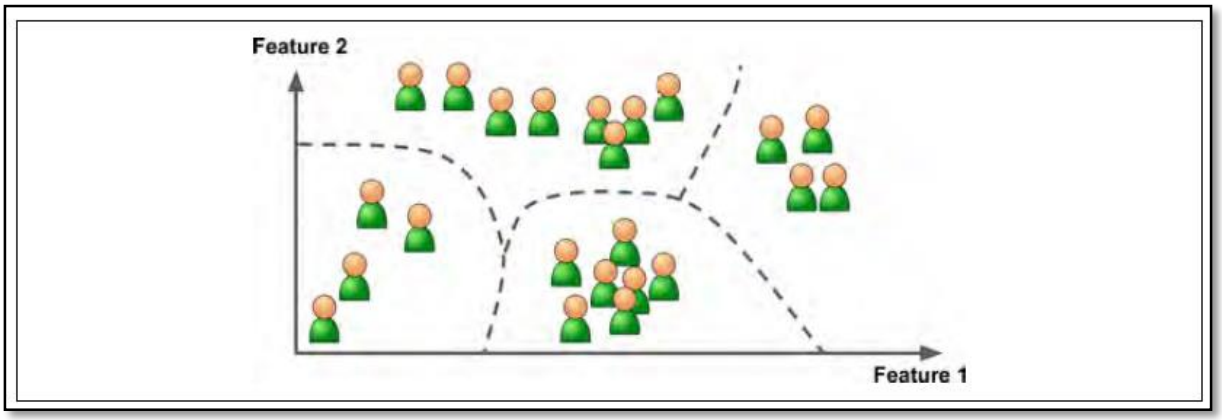


Figure 4: Picture depicting cluster on the basis of features available for training data

Clustering (k-Means, Hierarchical Cluster Analysis (HCA), Expectation Maximization), **Visualization and dimensionality reduction** (Principal Component Analysis (PCA), Kernel PCA, Locally-Linear Embedding (LLE), t-distributed Stochastic Neighbor Embedding (t-SNE)), **Association rule learning** (Apriori, Eclat).

Reinforcement learning: The reinforcement learning process depends on the rewards (Positive/Negative) to the learner. In this learning process, the machine learn the things over the period of time by reviewing its own action.

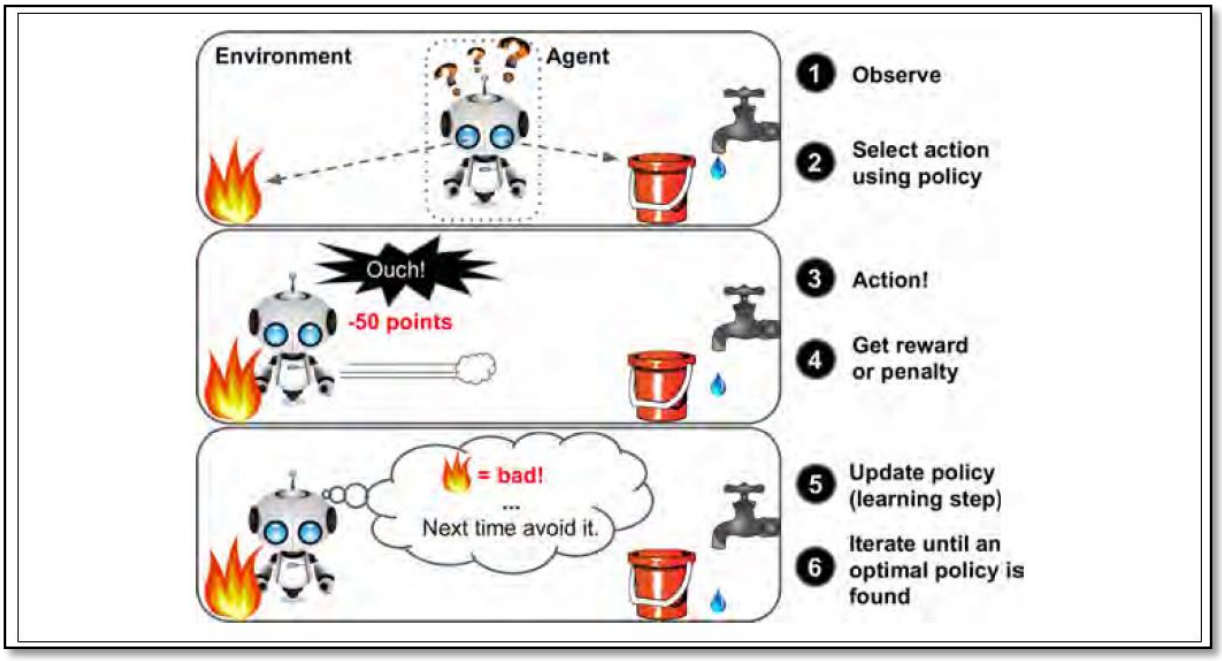


Figure 5: Picture depicting agent based reinforcement learning process

Examples: Computer games have the capacity to learn things from opponent actions as well from own action.

4 Artificial Neural Network

Efficient machine learning technique involves solving of mainly supervised classification problem and also capable of semi supervised, unsupervised and reinforcement machine leaning problem. Artificial neural networks (ANNs) provides a general and practical method for learning real-valued, discrete-valued, and vector-valued functions from examples suitable for predicting the model input-output pairs. ANN learning is robust in detecting errors in the training data and has been successfully applied to problems such as interpreting visual scenes, speech recognition, and learning robot control strategies.

4.1 Biological motivation of Neural Network

One type of ANN is inspired by the human nervous system that mimic basic computational unit of the brain, a neuron. Approximately 86 billion neurons can be found in the human nervous system and they are connected with approximately 10^{14} — 10^{15} synapses. Neuron activity is typically excited or inhibited through connections to other neurons each connected, on average, to 10^4 others neurons. The fastest neuron switching times are known to be on the order of 10^3 seconds--quite slow compared to computer switching speeds of 10^{-10} seconds. Yet

humans are able to make surprisingly complex decisions. The diagram below shows a cartoon drawing of a biological neuron (Figure 6) and a common mathematical model (Figure 7) of ANN.

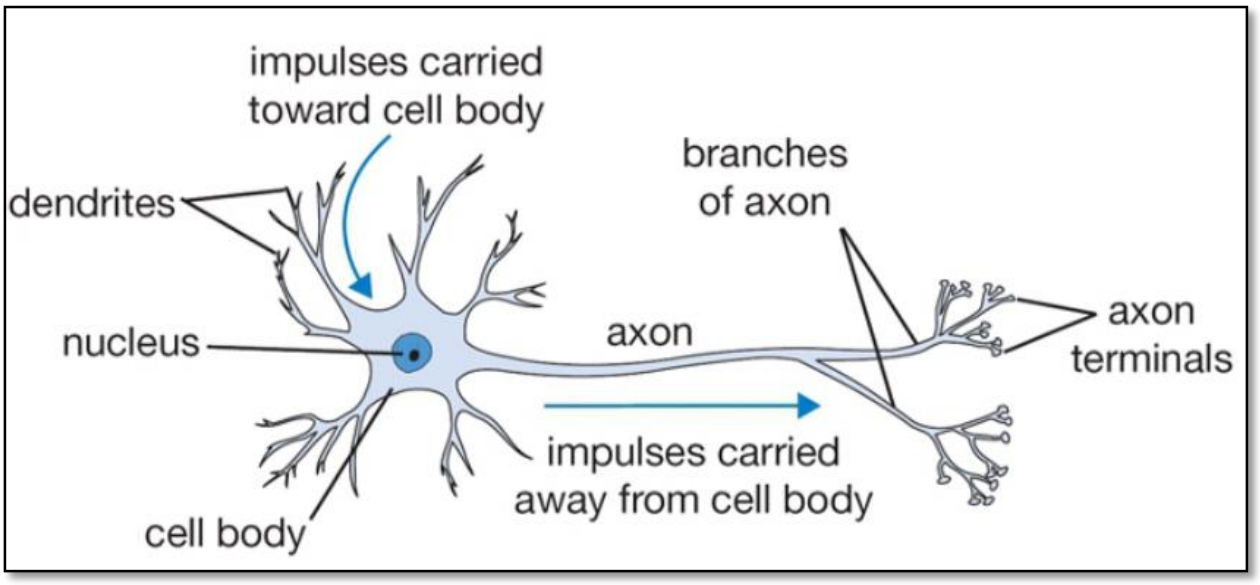


Figure 6: Schematic diagram of single neuron cell in human nervous system.

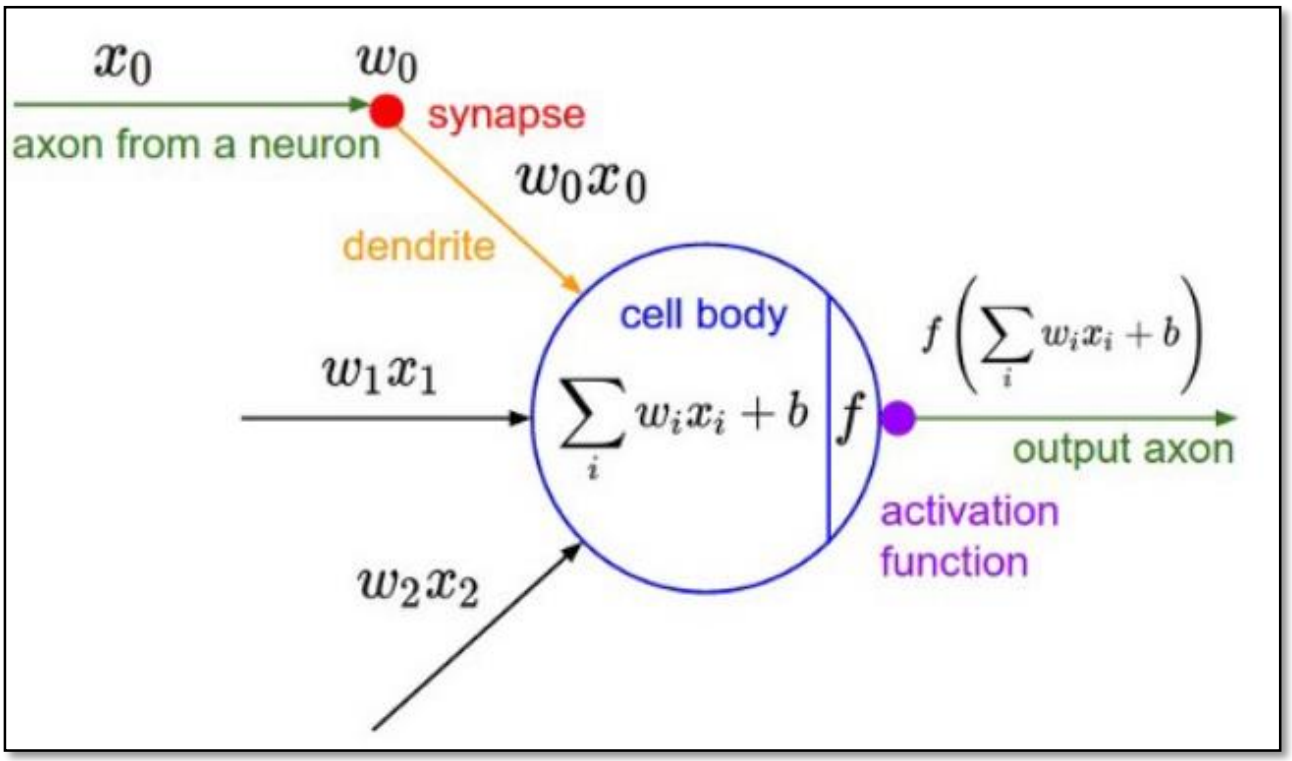


Figure 7: Picture depicting the basic unit of ANN inspired from the human nervous system.

4.2 Perceptron

One type of ANN system is based on a unit called perceptron, takes real valued inputs and calculate the linear combination of the inputs. A perceptron takes a vector of real-valued inputs, calculates a linear combination of these inputs, then outputs a 1 if the result is greater than some threshold and -1 otherwise.

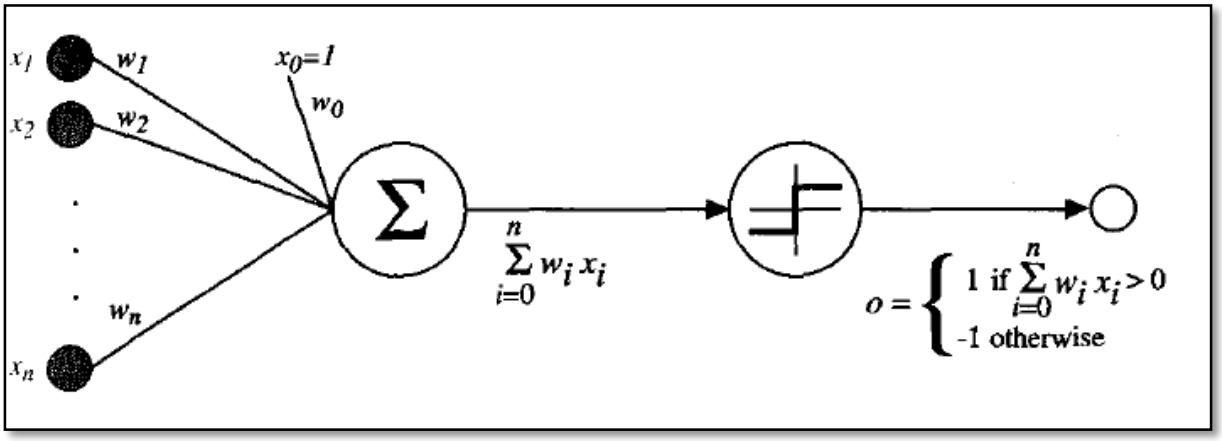


Figure 8: Picture depicting a simplified architecture of perceptron

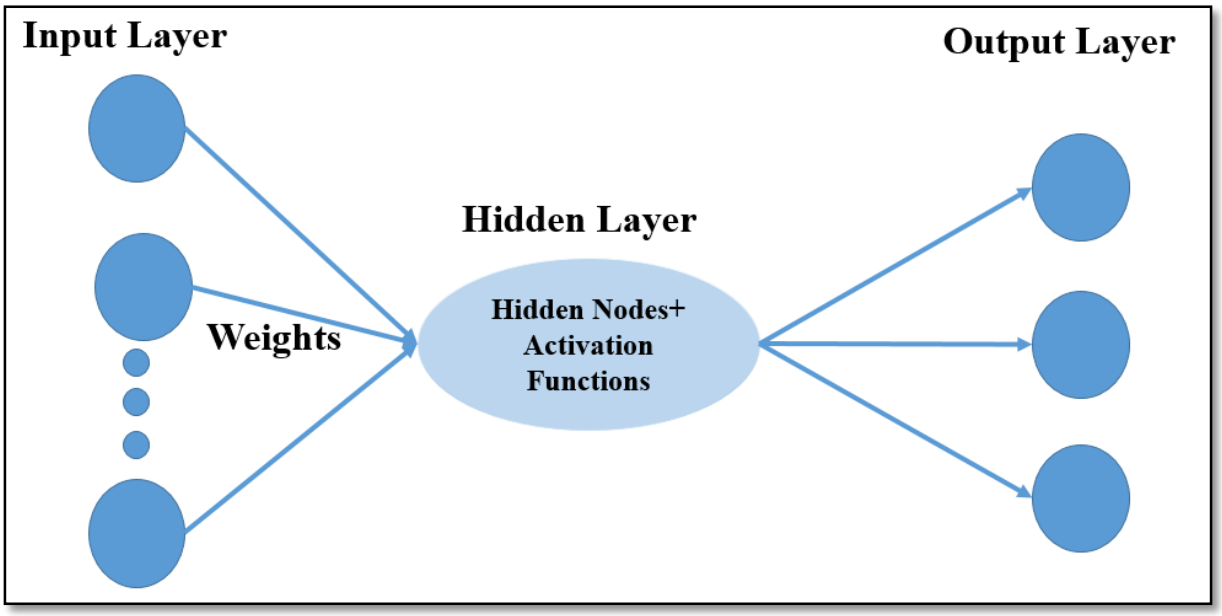


Figure 9: Picture depicting a basic structure of neural network

Input layer: No computation is done here within this layer, they just pass the information to the next layer. A block of nodes is also called layer which take the features.

Hidden layer: In Hidden layers is where intermediate processing or computation is done. It perform computations and then transfer the weights from the input layer to the following layer. It is possible to have a neural network without a hidden layer.

Output Nodes (output layer): Here we finally use an activation function that maps to the desired output format (e.g. softmax for classification).

Connections and weights: The network consists of connections.

Activation function: The activation function of a node defines the output of that node given an input or set of inputs. However, it is the nonlinear activation function that allows such networks to compute nontrivial problems using only a small number of nodes. In artificial neural networks, this function is also called the transfer function.

Learning rule: The learning rule is a rule or an algorithm which modifies the parameters of the neural network, in order for a given input to the network to produce a favoured output. This learning process typically amounts to modifying the weights and thresholds.

4.3 Types of Neural Networks

The neural networks have several architecture with different capabilities. Some of the types of the architecture are described in the following sections.

4.3.1 Feedforward Neural Network

A feedforward neural network is one of the basic ANN architecture where connections between the units do not form a cycle. The information flow happens in only one direction, forward, from the input nodes through the hidden nodes (if any) and finally to the output nodes.

4.3.1.1 Single-layer Perceptron

This is the simplest feedforward neural network and does not contain any hidden layer, which means it only consists of a single layer of output nodes. This is said to be single because when we count the layers we do not include the input layer, the reason for that is because at the input layer no computations is done and the inputs are fed directly to the outputs via a series of weights.

4.3.1.2 Multi-layer perceptron (MLP)

These class of networks consist of multiple layers of computational units, usually interconnected in a feed-forward way. Each neuron in one layer has directed connections to the neurons of the subsequent layer. In many applications, the units of these networks apply a sigmoid function as an activation function.

4.3.2 Convolutional Neural Network (CNN)

Convolutional Neural Networks are very similar to ordinary Neural Networks. They are made up of neurons that have learnable weights and biases. In CNN, the unit connectivity pattern is inspired by the organization of the visual cortex. Units respond to stimuli in a restricted region of space known as the receptive field. Receptive fields partially overlap, over-covering the entire visual field. Unit response can be approximated mathematically by a convolution operation. There are variations of multilayer perceptrons that use minimal preprocessing. Their wide applications is in image and video recognition, recommender systems and natural language processing.

Recurrent neural networks

In recurrent neural network (RNN), connections between units form a directed cycle (they propagate data forward, but also backwards, from later processing stages to earlier stages). This allows it to exhibit dynamic temporal behavior. Unlike feedforward neural networks, RNNs can use their internal memory to process arbitrary sequences of inputs. This makes them applicable to tasks such as unsegmented, connected handwriting recognition, speech recognition and other general sequence processors.

5 Support Vector Machine

Support Vector Machine (SVM) is a very powerful machine learning tool. The publication of the first papers of SVM by Vapnik, Chervonenkis and co-workers in 1964/65 went largely unnoticed till 1992. They were taken seriously only when excellent results on practical learning benchmarks were achieved in digit recognition, computer vision and text categorization. Support vector machine is a discriminant classifier through an optimal hyper plane that can categorizes the new examples. SVM is a two class classifier. It is well suited for classification of complex data of small and medium size.

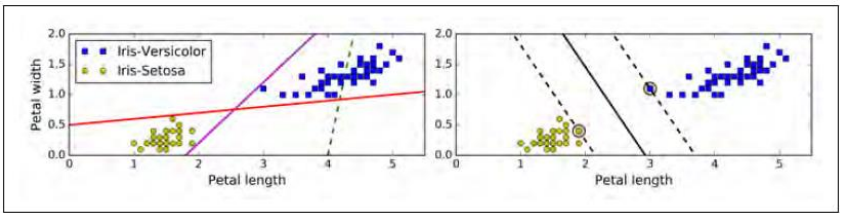


Figure 10: Picture depicting large margin classification

The Figure 10 shows some of the examples of the classifier that segregate the Versicolor and Setosa. The left side of the figure shows some separator line with a very poor decision boundary that has the capability to hardly separate the new instances. On the other hand, the right side separator decision boundary is good enough to separate the new data.

5.1 Linear SVM Classification

If the purpose is to separate the two classes with a straight line, one can use the linear SVM classification. The SVM separator uses the soft margin classification concept when the data is not easily linearly separable.

5.1.1 Soft Margin Classification

It is a more flexible model that can find a good balance between keeping the street as large as possible and limiting the margin violations.

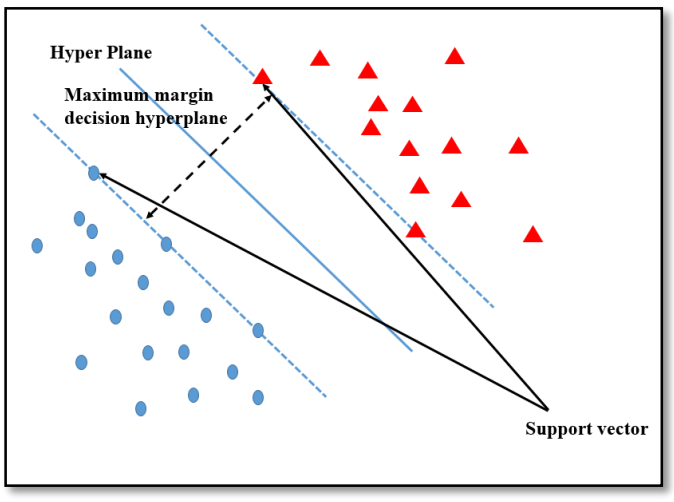


Figure 11: Conceptual view of SVM classifier with decision hyperplane and decision boundary and support vector.

5.2 Nonlinear SVM Classification

Although linear SVM classifiers are efficient and work surprisingly well in many cases, many datasets are not even close to being linearly separable. One approach to handling nonlinear datasets is to add more features, such as polynomial features. This dataset is not linearly separable, as you can see. But if you add a second feature x2 = (x1)2, the resulting 2D dataset is perfectly linearly separable.

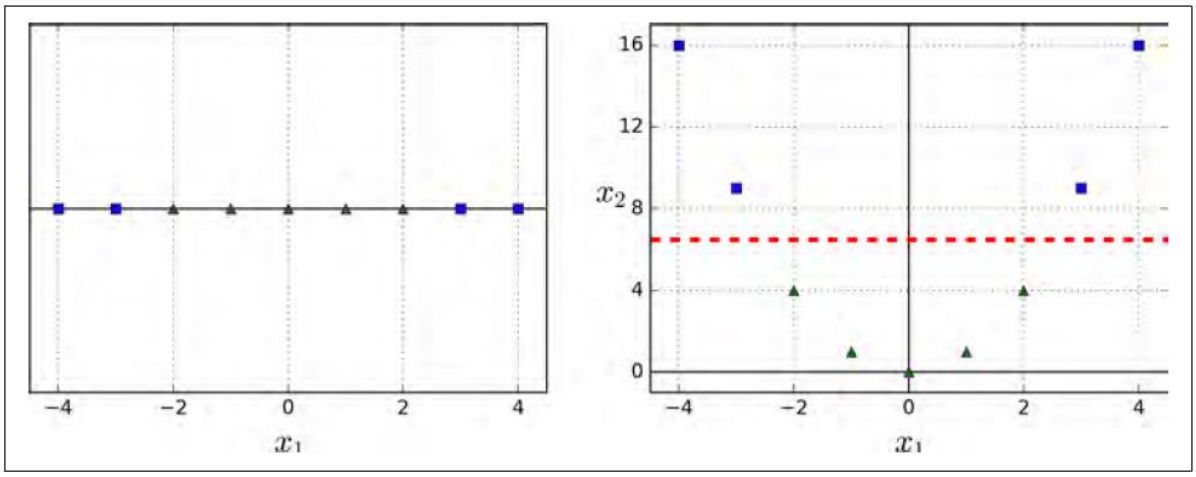


Figure 12: Picture depicting use of polynomial transformation to make the dataset linearly separable

The above solution is not suitable for highly complex data, because it make the computation very expensive. The solution is the kernel trick. The kernel trick is a very important concept in SVM classifier. When data is not linearly separable we try to map it in the higher dimension that automatically make some hyper plane which can easily segregate the data.

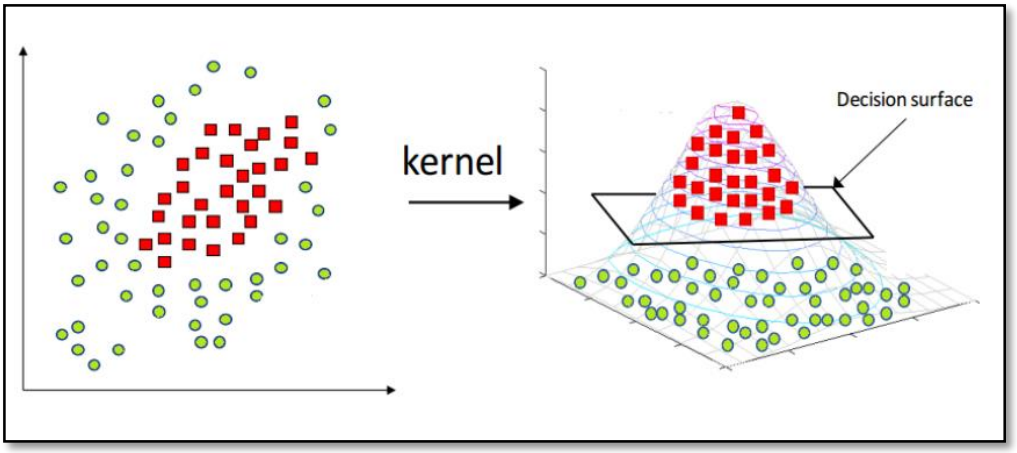


Figure 13: Picture depicting kernel trick to classify the data linearly non-separable data

References

Aurélien, G. (2017). Hands-on machine learning with scikit-learn & tensorflow. Geron Aurelien.

Coolen, A. C. (1998). A beginner's guide to the mathematics of neural networks. In *Concepts for Neural Networks* (pp. 13-70). Springer, London.

https://medium.com/@zxr.nju/what-is-the-kernel-trick-why-is-it-important-98a98db0961d

https://towardsdatascience.com/a-gentle-introduction-to-neural-networks-series-part-1-2b90b87795bc

Mitchell, T. M. (1997). Machine Learning, McGraw-Hill Higher Education. *New York*.

Overview of Python

Ashraful Haque
ICAR-IASRI, Library Avenue, New Delhi – 110 012
ashraful.haque@icar.gov.in

Python is the one of the most popular programming languages now-a-days. It is a high-level, interpreted, interactive, object-oriented programming language. Python language was created by Guido van Rossum in 1991 at the National Research Institute for Mathematics and Computer Science in the Netherlands. Python programming language is mainly used for-

- Data handling and visualization
- Analysis of variety of data such as numerical, textual, image, videos, audio etc.
- Performing complex mathematical computations
- Server-side scripting for developing web applications
- Standalone software development etc.

Why Python?

Python is very easy learn language. It can work in any system irrespective of the operating system. Syntax of python language is very simple and allows programmers to write programs in very few lines. Python runs on an interpreter system, which means that the code is being executed as soon as it is written. And last but not the least that python has a very large and mature community for the developers. There are lots of blogs, tutorials, documents, guide videos available online for the python developers.

Python Installation:

Most of the latest computer systems have python already installed. To check if you have python installed on a Windows PC, search in the start bar for Python or run the following on the Command Line (cmd.exe):

```
C:\your\python\installation\folder>python --version
```

If not, then one can download the latest version of python (latest version is 3.9.2) from https://www.python.org/downloads/ for the particular operating system and follow the guidelines while installation.

Getting Started with Python:

Any python script or file is saved with .py file extension. Let's us write the first python program that prints 'Hello, Everyone!!!'. So, first open a text editor and write the following code in it:

e.g.

```
print("Hello, Everyone!!!")
```

Now save it as 'first.py'. Now open command prompt, go to the python installation folder and type the following command:

```
C:\your\python\installation\path>python /your/program/path/first.py
```

The output should read:

```
Hello, Everyone!!!
```

Python from Command Line:

In case of python it is possible to run the code as a command line itself using the command prompt.

Type the following on the Windows, Mac or Linux command line:

```
C:\your\python\installation\path>python
```

From there one can write any python code, including our first example from earlier in the :

```
C:\your\python\installation\path>python
Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:04:45) [MSC v.1900 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Which will write "Hello, Everyone!!!" in the command line:

```
C:\your\python\installation\path>python
Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:04:45) [MSC v.1900 32 bit
(Intel)]                              on                              win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello, Everyone!!!")
Hello, Everyone!!!
```

Whenever you are done in the python command line, you can simply type the following to quit the python command line interface:

```
exit()
```

Python Syntax:

The major syntactical rules of python programs has been provided below-

Execution of code

a. python can be executed directly from command line.

```
>>> print("Hello, Everyone!!!")
Hello, Everyone!!!
```

b. Python can also be executed using a file with ‘.py’ extension

```
C:\your\python\installation\path>python /your/program/path/first.py
```

Indentation

The indentation refers to the spaces at the beginning of a program line. Indentation is very important and stricter in python. Python uses indentation as a block of code.

e.g.

```
if 5 > 2:
  print("Five is greater than two!")
```

Comments

In python, comments can be included in the code by using '#' symbol. Comments can be used in the beginning, middle, or in the end pf the code. Comments can be multiline. For multiline comments one can use triple quotes (""").

Variables in Python:

In python, the variables are simple storage structures for storing data values. There is no requirement of *type* declaration for the variables in python. The *type* of any variable can be acquired by *type()* function.

e.g.

```
x = 5
y = "python"
print(type(x))
print(type(y))
```

In python variables names -

- are case sensitive
- Must starts with a letter or underscore
- Can be alphanumeric

Python variables can store different types of data.

Text Type:	str
Numeric Types:	int, float, complex
Sequence Types:	list, tuple, range
Mapping Type:	dict
Set Types:	set, frozenset
Boolean Type:	bool
Binary Types:	bytes, bytearray, memoryview

Operators in Python:

Python divides the operators in the following groups:

Arithmetic operators	+, -, /, *, %, **, //
Assignment operators	=, +=, -=, *=, /=
Comparison operators	==, !=, >, <, >=, <=
Logical operators	And, or, not
Identity operators	is, is not,
Membership operators	in, in not
Bitwise operators	&, |, ^, ~, >>, <<

Data structures in python:

There are mainly four types of built-in data structures in python which are used for storing collection of data. These data structures are List, Tuple, Set and Dictionary.

a. Lists- List are used to store more than one data in single variable. Items in the list are indexed (starting from 0), ordered and changeable. Lists allow duplicate values. Lists are created by using the square brackets and items can be access by mentioning the index number inside the square brackets.

e.g.

```
list1 = ["apple", "banana", "cherry"]
list2 = [1, 5, 7, 9, 3]
list3 = [True, False, False]
```

N.B. Python does not have built-in support of Arrays, so Lists can be used as Arrays in python.

b. Tuple- Tuples are used to store more than one data in single variable. Items in the tuples are indexed (starting from 0), ordered and non-changeable. Tuples allow duplicate values. Tuples are created by using the round brackets and items can be access by mentioning the index number inside the square brackets.

e.g.

```
tuple1 = ("apple", "banana", "cherry")
tuple2 = (1, 5, 7, 9, 3)
tuple3 = (True, False, False)
```

c. Set- Sets are used to store more than one data in single variable. Items in the sets are unindexed, unordered and non-changeable. Sets doesn't allow duplicate values. Sets are created by using curly brackets. Items can't be access by mentioning the index number inside the square brackets. For accessing the items in the Sets one can use any loop structure.

e.g.

```
set1 = {"apple", "banana", "cherry"}
set2 = {1, 5, 7, 9, 3}
set3 = {True, False, False}
```

d. Dictionary- Dictionaries are used to store data in key:value pair. Items in the sets are ordered and changeable. But dictionary doesn't allow duplicate values. Dictionaries are created by using curly brackets having key:value pair. Items ca be access by mentioning the key name inside the square bracket.

e.g.

```
dictionary1 = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}
x = dictionary1 ["model"]
```

Control and Loops Structures in Python:

There is mainly one control structure that is *'if...else'* and two loop structure such as *'while'* and '*for'* loop.

a. *'if.. else'* structure- *'if...else'* structures are used to implement the logical conditions of the program. Syntax of '*if...else*' structure is given below-

e.g.

```
a = 33
b = 200
if b > a:
  print("b is greater than a")
else:
  print("error")
```

N.B. Indentation in the control and loop structures are very crucial in case of python programming language.

b. ***'while'* loop-** With the *'while*' loop, one can execute a set of statements as long as a condition is true.

e.g.

```
i = 1
while i < 6:
  print(i)
  i += 1
```

c. ***'for'* loop-** A 'for' loop is used for iterating over a sequence (that is either a list, a tuple, a dictionary, a set, or a string).

e.g. 1-

```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
  print(x)
```

e.g:2- Looping through the letters of a strings

```
for x in "banana":
  print(x)
```

Python Functions:

A function is a block of code that contains a set of statements and runs only when it is called explicitly. One can pass data, known as parameters, into a function. A function can return data as a result.

e.g.

```
def my_function(str):
  print(str + "! Welcome to the class.")
my_function("Bob")
```

Packages and PIP:

Package or module is a python object with arbitrarily named attributes that one can bind and reference. Packages allows us to logically locate the python code. Simply a package or module is file containing a set of python codes. Packages are also referred as library

Packages or modules or libraries can be imported by using the *'import'* keyword.

e.g.

```
import os
import sys
```

PIP is a package manager available in python. PIP is used to install, upgrade, or uninstall a packages in python environment.

```
C:\your\python\installation\path>pip install numpy
```

Some important packages or modules in Python:

NumPy:

NumPy is python library or packages used for working with arrays. NumPy was created by Travis Oliphant in 2005 and it is open source.

In python, the concepts of arrays is served by the List data structure but it is too slow in processing. NumPy provides a 50x faster access speed for the array objects in python than the List. NumPy has a lots of applications in the domain of -

- Arrays
- Matrices
- Linear Algebra

- Fourier Transformation

Creating Arrays

The object of NumPy that deals with the arrays is known as '*ndarray'*. One can create a *'ndarray'* object by using *array()* function. One can pass any type of array-like object in the *array()* function.

e.g.

```
import numpy as np
array_var = np.array([1, 2, 3, 4, 5])
```

Array can be of 0, 1, 2 or 3 dimensions.

e.g.

```
import numpy as np
array0 = np.array(42) #0 dimension
array1 = np.array([1, 2, 3, 4, 5, 6, 7, 8]) # 1 dimension
array2 = np.array([[1, 2, 3], [4, 5, 6]]) # 2 dimension
array3 = np.array([[[1, 2, 3], [4, 5, 6]], [[1, 2, 3], [4, 5, 6]]]) #3
dimension
```

Accessing Array elements

Array elements can be accessed by its index number

```
print(array1 [2]) #accessing the 3rd item from the array 'array1'
```

Slicing an Array

Slicing in python means taking elements from one given index to another given index.

```
print(array1 [1:3]) #slicing from 2nd item to the 4th element
print(array1 [2:]) #slicing from 3rd item to the last element
print(array1 [:6]) #slicing from beginning to the 5th element
```

Properties and functions:

`dtype-` returns the type of values stored in the array object

`shape-` gives the number of elements in each dimension of the array object

`reshape-` allows to change the shape of the array either by adding adding/removing dimensions or changing the number of elements in each dimension

`concatenate()-` joins two or more arrays axis wise.

`array_split()-` splitting an array into two or more parts

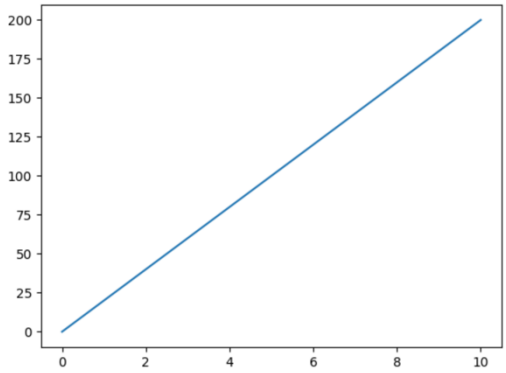
Matplotlib:

Matplotlib is a low level graph plotting library in python that serves as a visualization utility. Matplotlib was created by John D. Hunter. Matplotlib is open source and we can use it freely.

Most of the Matplotlib utilities lies under the pyplot submodule, and are usually imported under the plt alias.

e.g. Draw a line in a diagram from position (0,0) to position (10, 200):

```
import matplotlib.pyplot as plt
import numpy as np
xpoints = np.array([0, 10])
ypoints = np.array([0, 200])
plt.plot(xpoints, ypoints)
plt.show()
```



Properties and functions:

`marker-` keyword argument to emphasize each point in the plot

`linestyle/ls-` keyword argument to change the style of the plotted line

`xlabel()-` functions for setting a label for x-axis

`ylabel()-` function for setting a label for y-axis

`title() -` function for giving the title for the plot

`grid()` -function to add grid lines to the plot

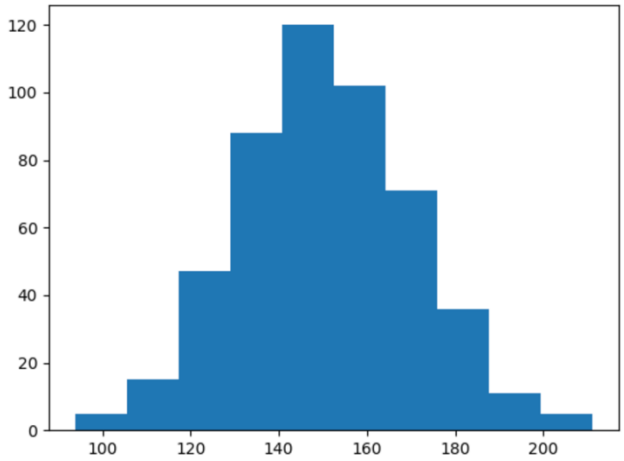
`scatter()`-function to draw a scatter plot

`bar()-` function to draw bar graphs

`hist()-` function to create histograms

e.g.

```
import matplotlib.pyplot as plt
import numpy as np
x = np.random.normal(150, 20 , 250)
plt.hist(x)
plt.show()
```



Pandas:

Pandas is a one of the most popular python package providing high-performance data manipulation and analysis tool using its powerful data structures. The name Pandas is derived from the word 'Panel Data' – an Econometrics from Multidimensional data. Pandas is well suited for many different kinds of data:

- Fast and efficient DataFrame object with default and customized indexing.
- Tools for loading data into in-memory data objects from different file formats.
- Data alignment and integrated handling of missing data.
- Reshaping and pivoting of date sets.
- Label-based slicing, indexing and subsetting of large data sets.
- Columns from a data structure can be deleted or inserted.
- Group by data for aggregation and transformations

There are mainly two data structures of pandas which handle the majority of typical use cases in finance, statistics, social science and Engineering are Series (1-dimensional) and DataFrame (2-dimensional).

DataFrame

A Data frame is a two-dimensional data structure, i.e., data is aligned in a tabular fashion in rows and columns.

Features of DataFrame:

- Potentially columns are of different types
- Size – Mutable
- Labelled axes (rows and columns)
- Can Perform Arithmetic operations on rows and columns

e.g.1

```
import pandas as pd
data = [1,2,3,4,5]
df = pd.DataFrame(data)
print df
```

```
   0
0  1
1  2
2  3
3  4
4  5
```

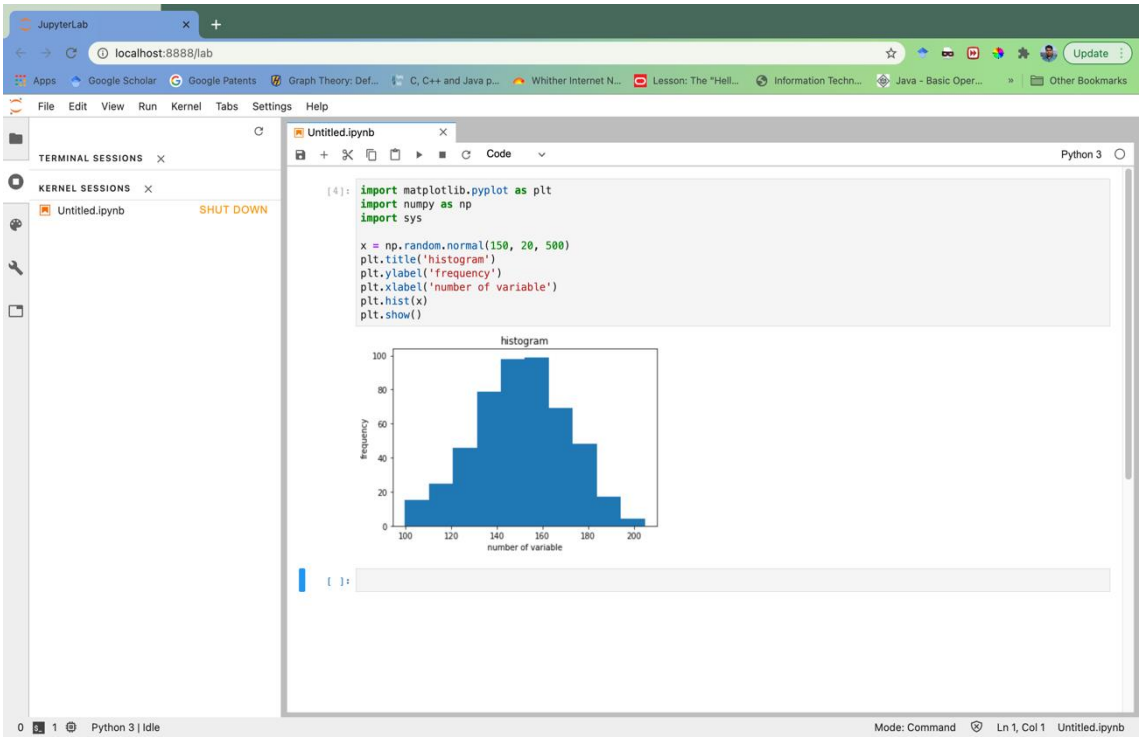
e.g. 2

```
import pandas as pd
data = [['Alex',10],['Bob',12],['Clarke',13]]
df = pd.DataFrame(data,columns=['Name','Age'],dtype=float)
print df
```

```
    Name      Age
0    Alex     10.0
1    Bob      12.0
2    Clarke   13.0
```

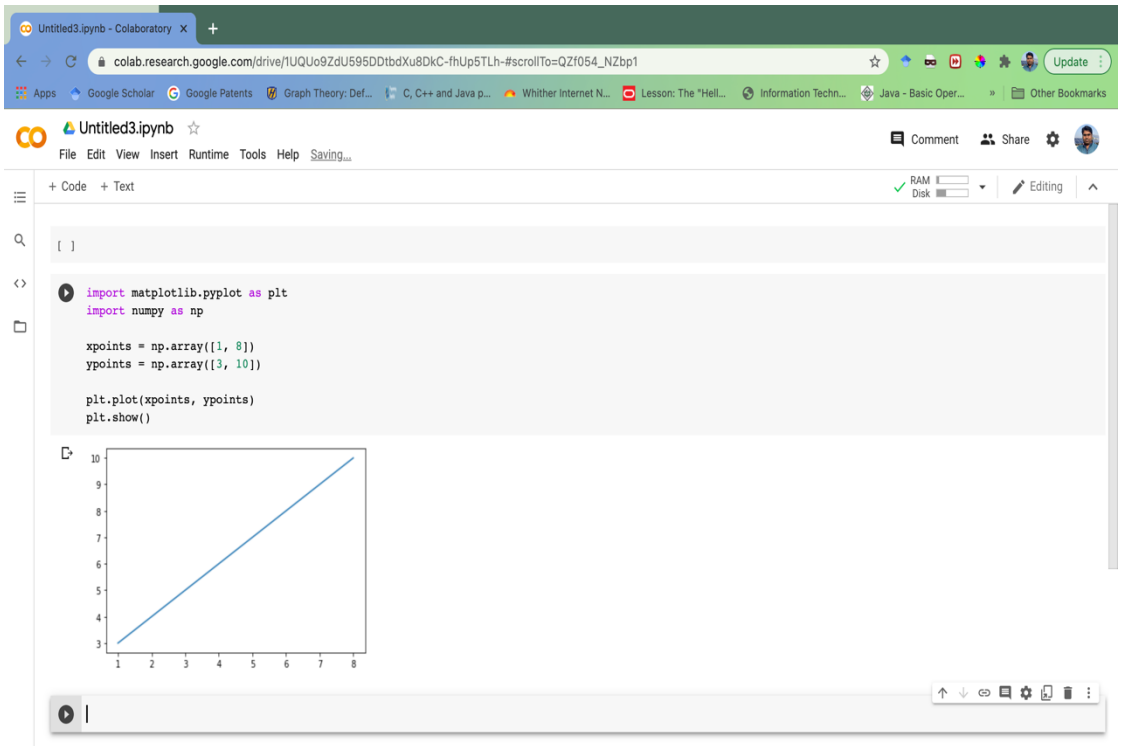
Jupyter Notebook

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.



Google Colab:

Colab is a Python development environment that runs in the browser. This facility is provided by the google in free of cost. Jupyter notebook is used for the programming purpose in colab. It allows the user to install any python library at any time. It also provide high computing programming environment that is Graphics Processing Unit (GPU) on free of cost to the users. One can upload his/her data to the google drive and analyse the data using the colab environment.



References:

1. https://colab.research.google.com/github/tensorflow/examples/blob/master/courses/udacity_intro_to_tensorflow_for_deep_learning/l01c01_introduction_to_colab_and_python.ipynb#scrollTo=F8YVA_634OFk
2. https://docs.python.org/3/tutorial/
3. https://numpy.org/
4. https://pandas.pydata.org/
5. https://www.guru99.com/python-tutorials.html
6. https://www.programiz.com/python-programming
7. https://www.tutorialspoint.com/python/index.htm
8. https://www.w3schools.com/python/default.asp

Database Management System Using MS Access

Soumen Pal
ICAR-IASRI, Library Avenue, New Delhi – 110 012
soumen.pal@icar.gov.in

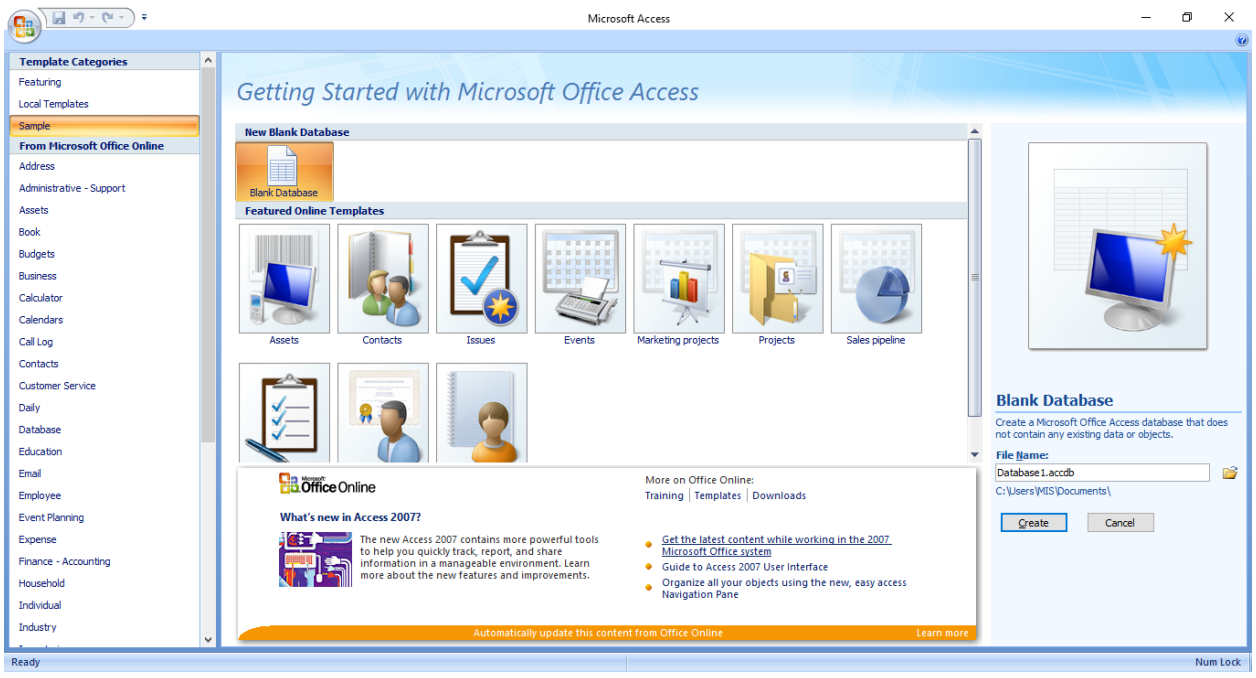
A database is an organized collection of interrelated data. A database management system (DBMS) is computer software that facilitates the process of defining, constructing and manipulating databases for various applications. Examples of Information Systems include Bank, Library and Railway Reservation which use DBMS. MS Access is the database software in the Microsoft Office suite that allows to order, manage, search, and report large amounts of information.

Create Access Database

The first step in creating an Access database is to create a blank database file. This is done from the Getting Started Screen when Access is launched. The file is saved into one of the specified folders in computer. The procedure for doing this is outlined below.

1. Launch Access

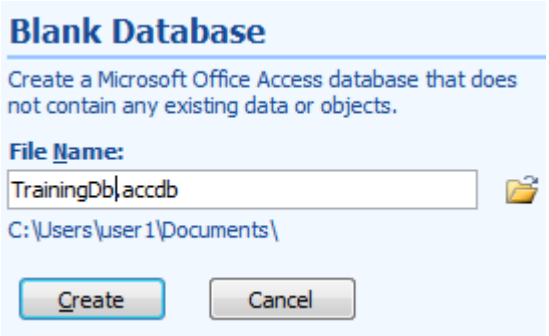
To begin, launch Access by clicking on the desktop icon, or choose Access from the start menu. This brings up the Getting Started with Microsoft Office Access screen.



2. Select Blank Database Template

Towards the top left of the screen, there is a "Blank Database" icon. Click this icon to bring up the Blank Database side bar on the right hand side of the screen. This is where one has to enter details about the database file to be created.

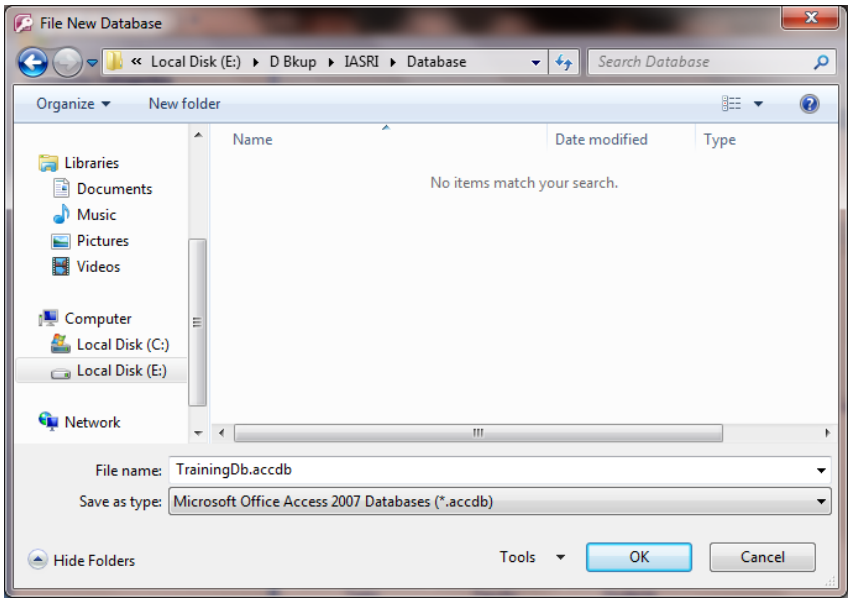
3. Enter filename for Access database.



Type TrainingDb in the **File Name** textbox.

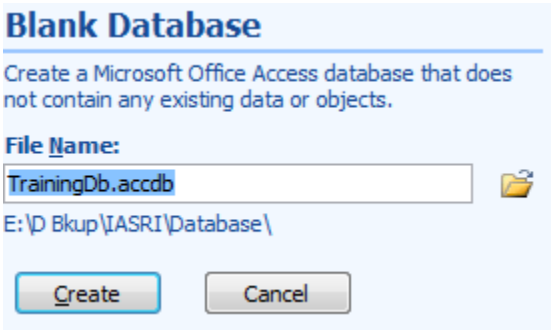
4. Browse and select folder

Next click the folder icon adjacent to **File Name** textbox and browse for a folder to put the database. Click the **OK** button once the file path has been selected.

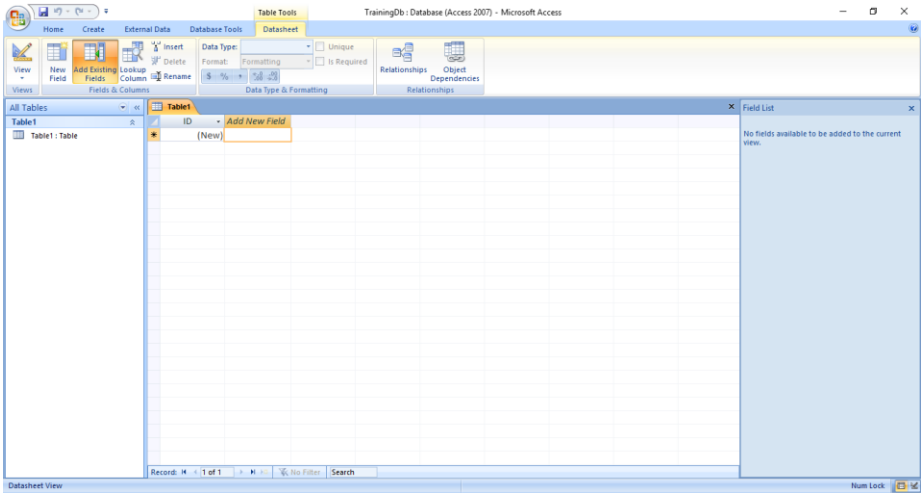


5. Click Create

Now the selected file path can be seen below the **File Name** textbox. Once the **Create** button is clicked, the database file is saved to the specified location and opened to work on.



6. The window for the TrainingDb database will open.

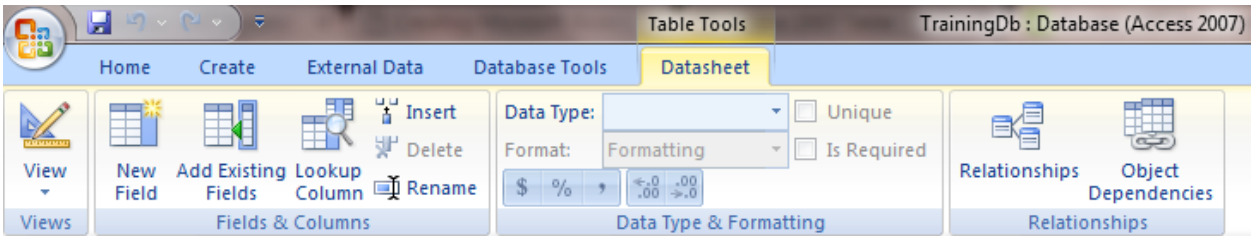


The newly created database file is ready to be worked upon.

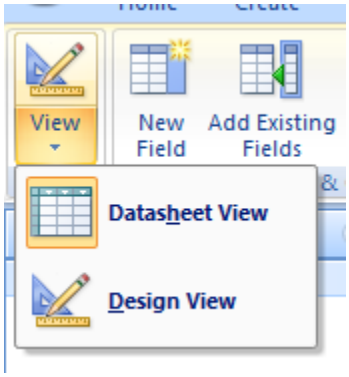
Create Access Table

Tables are the foundation of an Access database. Access stores data in tables. They look like the cells of a spreadsheet with columns and rows. Each horizontal column represents a table record, and each vertical column represents a table field.

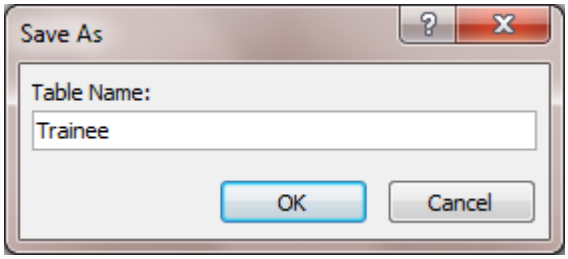
1. Click the **Datasheet** tab. In the Ribbon, click **View**.



2. When the menu appears, click **Design View**.



3. A **Save As** window will appear, type **Trainee** in the **Table Name** box. This is the first database table to be created.



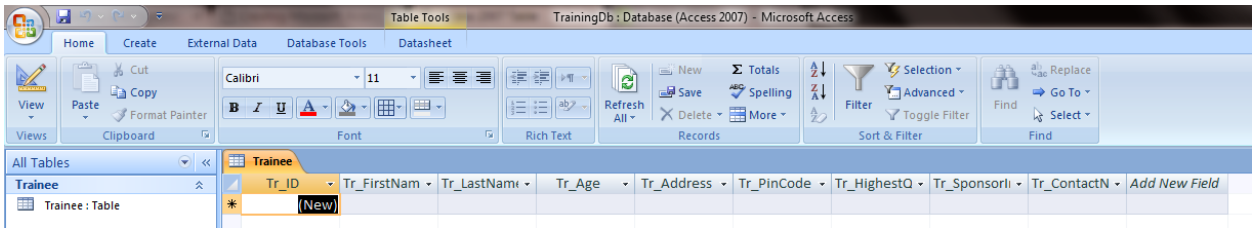
Then click on the OK button.

Create fields in Design View

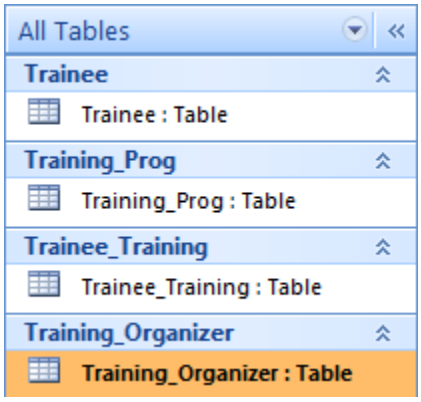
This brings up the Table Design Grid where each field name and its data type can be entered. The first field created is the Tr_ID field which is going to contain a unique reference number for each trainee record. This column is by default the primary key field. A primary key is a field or combination of fields that uniquely identify each record in a table. Enter the name "Tr_ID" into the first column of the first row in the grid. To automatically generate a unique reference number, **AutoNumber** is to be selected from the drop down list in the data type column. One can also enter a description for each field, but this is not essential.

On the next row the field is going to be called Tr_Firstname and the data type is going to be Text. On the third row the field name is Tr_LastName with the data type again being Text. Likewise, one can add as many fields as required. And finally, the last field name is Tr_ContactNo and the data type here is going to be Number.

Now the table can be saved by clicking the save icon on the top left of the screen above the Access Ribbon. To view the table, select **Datasheet View** from the **Views** group. This brings up the datasheet view of the table that is just created. One can see the field headings running across the top of the table.



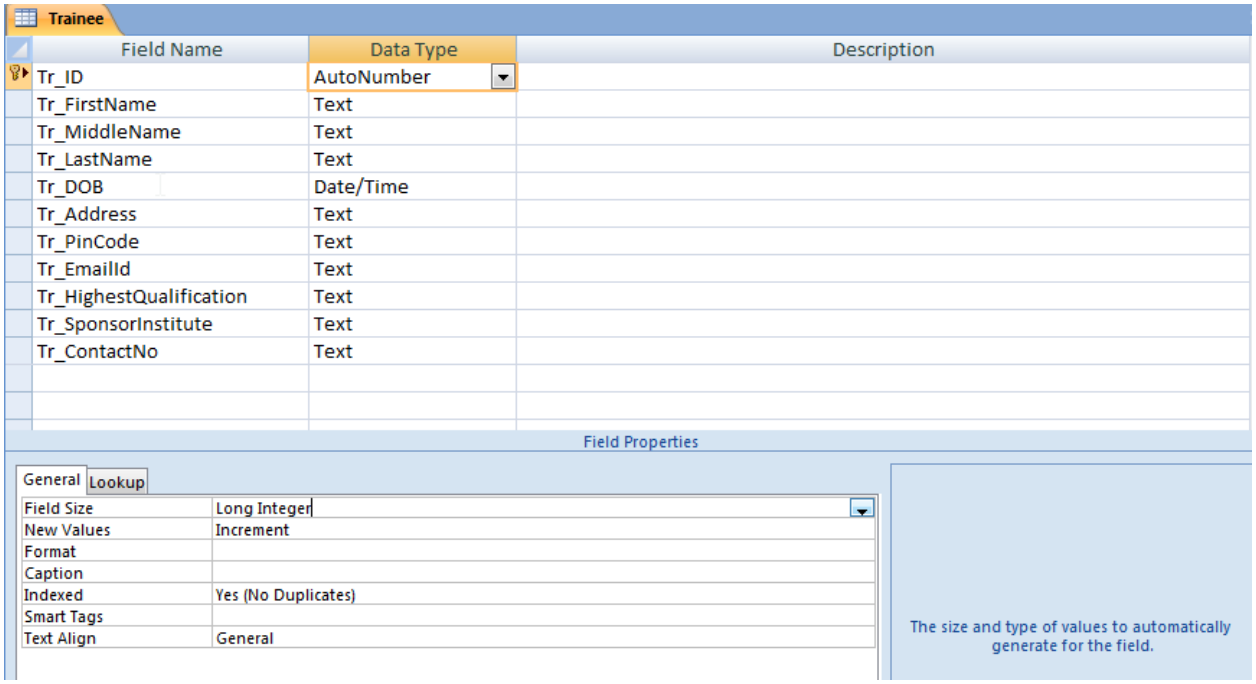
The following tables have been created: Trainee, Training_Prog, Trainee_Training, Training_Organizer.



The Trainee Table contains all the details of the trainees attending different training programmes. This table has the following fields:

Field Name	Data Type
Tr_ID	AutoNumber
Tr_FirstName	Text
Tr_MiddleName	Text
Tr_LastName	Text
Tr_DOB	Date/Time
Tr_Address	Text
Tr_PinCode	Text
Tr_EmailId	Text
Tr_HighestQualification	Text
Tr_SponsorInstitute	Text
Tr_ContactNo	Text

The properties of the data types can be viewed and modified in the **General** Tab under **Field Properties**.



Training_Prog Table contains the details of training programmes and it has the following fields: Training_ID (AutoNumber), Training_Title (Text), Training_Start_Date (Date/Time), Training_End_Date (Date/Time), Training_Host_Institute (Text), Course_Coordinator (Number).

Training_Organizer Table contains the details of the training programme organizers and it has the following fields: Organizer_ID (AutoNumber), Organizer_FirstName (Text), Organizer_MiddleName (Text), Organizer_LastName (Text), Organizer_Address (Text), Organizer_EmailId (Text).

Trainee_Training Table contains the information of the trainees participating in the training programmes and it has the following fields: TraineeTraining_ID (AutoNumber), Trainee_ID (Number), Training_ID (Number).

Building table relationships

In Access, data are stored in multiple tables. Relationships are used to join the tables. After creating relationships, data can be used from all of the related tables in a query, form, or report.

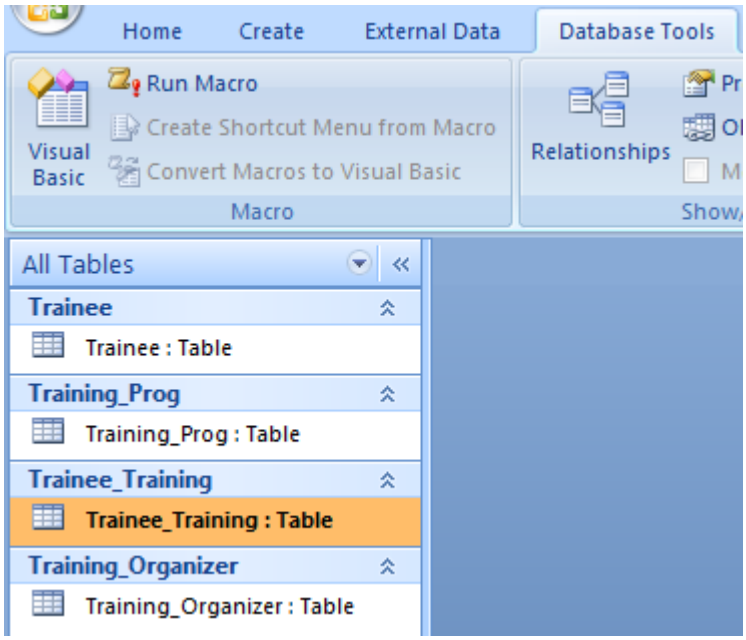
Along with primary key, the foreign key concept is required in building table relationship. A foreign key is a value in one table that must match the primary key in another table. Primary keys and foreign keys are used to join tables together. In other words, primary keys and foreign keys are used to create relationships.

There are three types of relationships: one-to-one, one-to-many (or many-to-one) and many-to-many. In a one-to-one relationship, for every occurrence of a value in table A, there can only be one matching occurrence of that value in table B, and for every occurrence of a value in table B, there can only be one matching occurrence of that value in table A. One-to-one relationships are rare because if there is a one-to-one relationship, the data is usually stored in a single table. However, a one-to-one relationship can occur when one wants to store the information in a separate table for security reasons, when tables have a large number of fields, or for other reasons. In a one-to-many relationship, for every occurrence of a value in table A, there can be zero or more matching occurrences in table B, and for every one occurrence in table B, there can only be one matching occurrence in table A. In a many-to-many relationship, for every occurrence of a value in table A, there can be zero or more matching occurrences in table B, and for every one occurrence in table B, there can be zero or more matching occurrences in table A.

In the present scenario, one Training Organizer can be Course_Coordinator in one or more training programmes, however, one particular training programme can only have one Course_Coordinator. This is a one-to-many relationship. Now, one trainee can participate in one or more training programmes and one training programme has more than one participants. So, this is an example of many-to-many relationship. In such scenario, another Table viz. Trainee_Training is introduced to break the many-to-many relationship into two one-to-many relationships.

To establish a relationship between tables:

1. Click the **Relationships** button in the **Show/Hide** group on the **Database Tools** tab. It is important that tables must be closed in order to establish relationships.

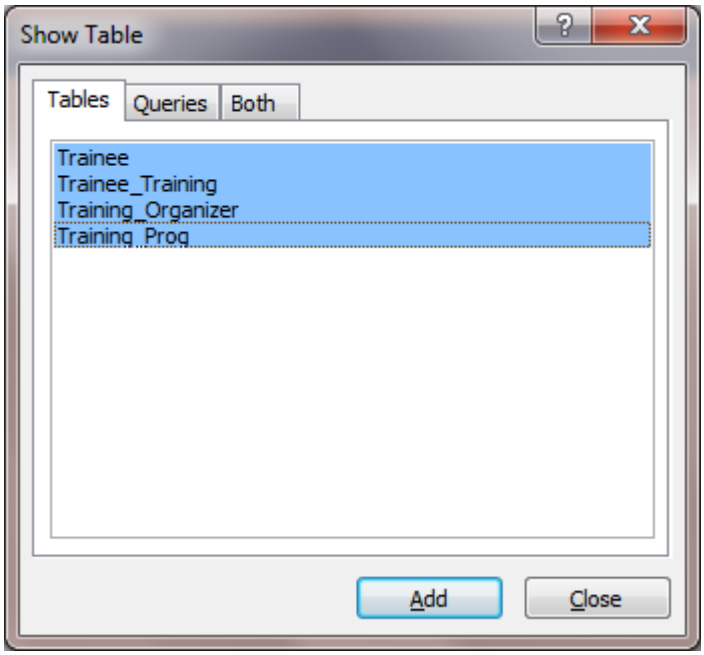


2. Click the **Show Table** button in the Relationships group. The Show Table dialog box appears.

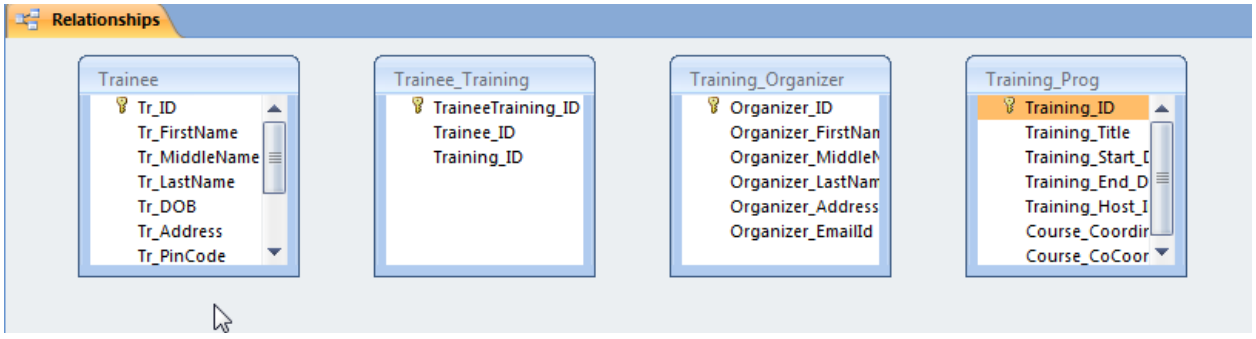
3. Activate the **Tables** tab if the relationships will be based on tables, activate the **Queries** tab if the relationships will be based on queries, or activate the **Both** tab if the relationships will be based on both.

4. Select each table name and then click **Add** for the tables to be related. One can also select multiple tables at a time by pressing the Ctrl Key and then click **Add**.

5. Click the **Close** button to close the **Show Table** dialog box.



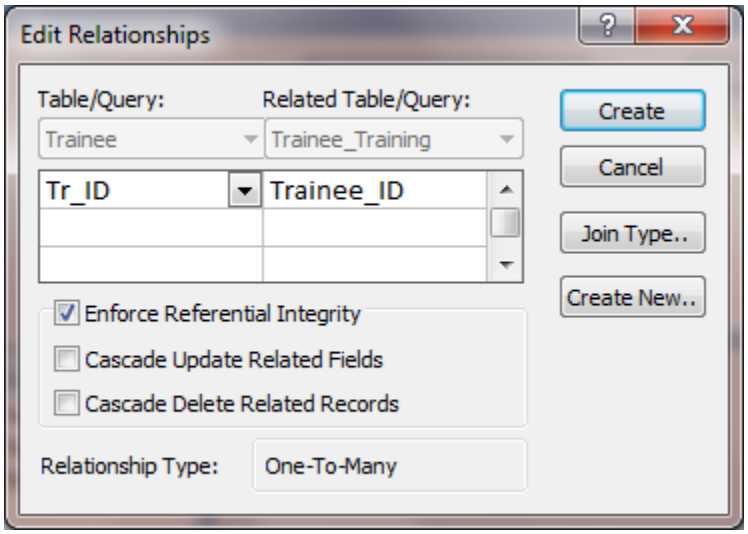
Now, one can see a relationship map that contains all of the tables those were selected.



To move a table that appears in the relationship map:

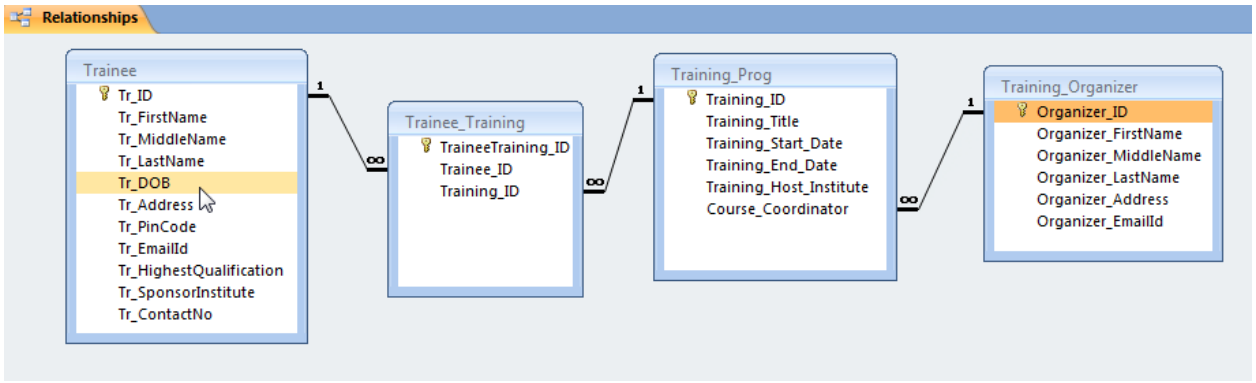
- Place the mouse over the table to be moved.
- Hold down the left mouse button, then drag the table to a new location.
- Release the mouse button to drop the table in its new place.

6. Drag the Primary table's primary key over the related table's foreign key. After dragging the primary key to the related table's box, the cursor changes to an arrow. Make sure that the arrow points to the foreign key. The **Edit Relationships** Dialog box will appear.



7. Click the Enforce Referential Integrity checkbox.

8. Click Create. Access creates a one-to-many relationship between the tables.



Note: After a relationship has been created between two tables, one must delete the relationship before making modifications to the fields on which the relationship is based. To delete a relationship:

1. Click the line that connects the tables.

2. Press the Delete key.

The other facilities available in Access include Queries, Forms and Reports. Query is used to view a subset of the data or to answer questions about the data. Access Forms are used to enter, edit or display data and they are based on tables. Reports organize and summarize data for

viewing online or for printing. A detail report displays all of the selected records. One can include summary data such as totals, counts, and percentages in a detail report.

References

1. Date, C. J. (2006). *An Introduction to Database Systems*. Pearson Education.
2. http://www.baycongroup.com/access2007/
3. http://www.dealing-with-data.net/
4. http://www.gcflearnfree.org/access2007/

SEQUENCE ALIGNMENT

SB Lal
ICAR-IASRI, Library Avenue, New Delhi – 110 012

1. Introduction

Since the development of high-throughput methods for production of gene and protein sequences during 90s, the rate of addition of new sequences to the databases increases very rapidly. However, comparing sequences with known functions with these new sequences is one way of understanding the biology of that organism from which the new sequence comes. Thus, sequence analysis can be used to study of the similarities between the compared sequences. Now a days, there are many tools and techniques that provide the sequence comparisons (sequence alignment) and analyze the alignment to understand the biology.

Sequence analysis in molecular biology and bioinformatics is an automated, computer-based examination of characteristic fragments, e.g. of a DNA strand. It basically includes relevant topics:

1. The comparison of sequences in order to find similarity and dissimilarity in compared sequences (sequence alignment)
2. Identification of gene-structures, reading frames, distributions of introns, exons and regulatory elements
3. Finding and comparing point mutations or the single nucleotide polymorphism (SNP) in organism in order to get the genetic marker.
4. Revealing the evolution and genetic diversity of organisms.
5. Functional annotation of genes.

Sequence alignment is a way to identify regions of similarity in DNA, RNA, or protein sequences that may be a consequence of functional, structural, or evolutionary relationships between the sequences. Aligned sequences of nucleotide or amino acid residues are typically represented as rows within a matrix. If two sequences share a common ancestor for the alignment, mismatches can be interpreted as point mutations and gaps as indels (that is, insertion or deletion mutations). Thus, a letter or a stretch of letters may be paired up with dashes in the other sequence to signify such an insertion or deletion. Homologous sequences may have different length, which is generally explained through insertions or deletions in sequences. Since an insertion in one sequence can always be seen as a deletion in the other one frequently uses the term "indel". In sequence alignments of proteins, the degree of similarity between amino acids sequence can be interpreted as a rough measure of how conserved a particular region or sequence motif is among lineages. The absence of substitutions, or the presence of only very conservative substitutions (that is, the substitution of amino acids whose side chains have similar biochemical properties) in a particular region of the sequence, suggest that this region has structural or functional importance. Although DNA and RNA nucleotide bases are more similar to each other than are amino acids, the conservation of base pairs can indicate a similar functional or structural role.

Very short or very similar sequences can be aligned by hand. However, most interesting problems require the alignment of lengthy, highly variable or extremely numerous sequences that cannot be aligned solely by human effort. Computational methods need to be developed for the alignment of a large pair of sequences. Computational approaches are of two categories: *global alignments* and *local alignments*. Global alignment is a form of global optimization that "forces" the alignment to span the entire length of all query sequences. Global alignment will be applied when the sequences are of similar lengths. Local alignments identify regions of similarity within long sequences. Local alignments are often preferable, but it consumes more time to calculate because of the additional challenge of identifying the regions of similarity in the local regions. Number of algorithms is being applied for the sequence alignment, including optimizing methods like dynamic programming, and heuristic algorithms or probabilistic methods designed for large-scale database search.

```
AAB24882        TYHMCQFHCRYVNNHSGEKLYECNERSKAFSCPSHLQCHKRRQIGEKTHEHNQCGKAFPT 60
AAB24881        --------------------YECNQCGKAFAQHSSLKCHYRTHIGEKPYECNQCGKAFSK 40
                                    ****: .***:  * *:** * :****.:* *******..

AAB24882        PSHLQYHERTHTGEKPYECHQCGQAFKKCSLLQRHKRTHTGEKPYE-CNQCGKAFAQ- 116
AAB24881        HSHLQCHKRTHTGEKPYECNQCGKAFSQHGLLQRHKRTHTGEKPYMNVINMVKPLHNS 98
                 **** *:***********:***:**.: .***************    :  *.: :
```

Fig. 1 Sample of sequence Alignment text based representations

In sequence alignment of graphical representations, sequences are written in rows so that aligned residues appear in successive columns. While in text formats, aligned columns containing identical or similar characters are indicated with a system of conservation symbols. An asterisk or pipe symbol is used to represent the similarity of these two columns, a colon for conservative substitutions and a period for semi-conservative substitutions.

Many sequence visualization techniques use a color coding scheme to display information about the properties of the individual sequence elements. In DNA and RNA sequences, each nucleotide is represented by a specific color. In protein alignments, color is used to indicate amino acid properties in determining the conservation of a given amino acid substitution.

2. Pair-wise Alignment

Pair-wise sequence alignment methods are used to find the best-matching pairs of two sequences. The three primary methods of pair-wise alignments are dot-matrix, dynamic programming and word methods. One way of quantifying the utility of a pair-wise alignment is the 'maximum unique match', or the longest subsequence that occurs in both query sequence.

a) Dot-Matrix Method: The two sequences are written along the top row and leftmost column of a two-dimensional matrix and a dot is placed at any point where the characters in the appropriate columns match. We try to draw lines diagonally. The dot plots of very closely related sequences will appear as a single line along the matrix's main diagonal (Fig. 2). The dot-matrix approach produces a simple way of alignments for small sequences with the similar regions but time-consuming to analyze large sequences.

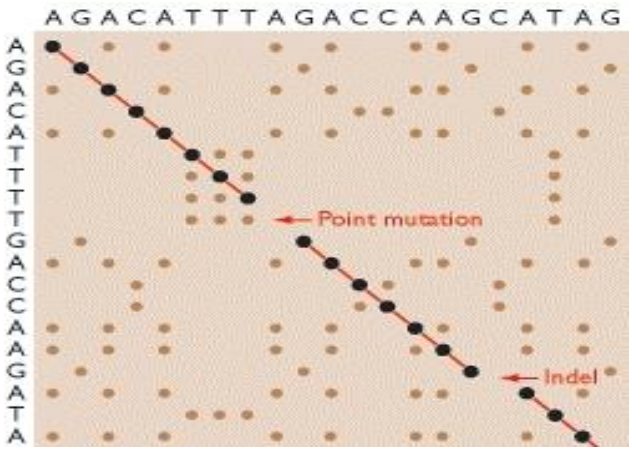


Fig. 2: The dot matrix technique for sequence alignment

There are many problems with dot plots such as noise, lack of clarity, difficulty extracting match summary statistics. Dot-plots are limited to two sequences only.

b) Dynamic Programming: Dynamic programming can be applied to produce global and local alignments. This can be done by applying Needleman-Wunsch algorithm for global alignment and Smith-Waterman algorithm for the local alignments. In general, alignments use a substitution matrix to assign scores for matches or mismatches, and a gap penalty for matching an in one sequence with a gap in the other.

DNA and RNA alignments may use a different scoring matrix, but in practice often simply assign a positive match score, a negative mismatch score, and a negative gap penalty. Dynamic programming can be useful in aligning nucleotide to protein sequences. The framesearch method produces a series of global or local pair-wise alignments between a query nucleotide sequence and a search set of protein sequences, or vice versa. The BLAST and EMBOSS provide basic tools for creating alignments of the sequences.

c) Word Method: Word or *k*-tuple methods are heuristic methods but are not guaranteed to find an optimal alignment solution. These methods are especially useful in large-scale database searches Word methods are best known for their implementation in the database search tools FASTA and BLAST family. Word methods identify a series of short, non-overlapping subsequences ("words") that are matched to candidate database sequences. The relative positions of the word in the two sequences being compared are subtracted to obtain an offset; this will indicate a region of alignment if multiple distinct words produce the same offset.

In the FASTA method, the user defines a value *k* to use as the word length with which to search the database. The method is slower but more sensitive for lower values of *k*, which are preferred for searching a very short query sequence. The BLAST family of search methods provides a number of algorithms optimized for particular types of queries. BLAST was developed to provide a faster alternative to FASTA without sacrificing accuracy. BLAST uses a word search of length *k*, but evaluates only the most significant word matches. Most BLAST implementations use a fixed default word length that is optimized for the query and database. Web based implementations are available such as EMBL FASTA and NCBI BLAST.

3. Global and Local Alignment

Global Alignment

Global alignments, which attempt to align every residue of each sequence, when the size of the sequences are similar or of equal size. A general global alignment technique is based on dynamic programming i.e., Needleman-Wunsch algorithm. This can be easily understood with the following two sequences aligned globally as follows

G A A T T C A G T T A (sequence #1)
G G A T C G A (sequence #2)

In simple dynamic programming principle, we construct a matrix. The matrix will be filled by inserting 0 or 1 where ever there is a mismatch or match. We also penalize the gaps with 0 as a simple case. Following steps are needed for construction of the matrix

i. Initialization
ii. Matrix fill (scoring)
iii. Traceback (alignment)

i. Initialization

The first step is to create a matrix with M + 1 columns and N + 1 rows where M and N are the sizes of the sequences to be aligned.

With the given sequences, length of sequence #1 = 11 and length of sequence #2 is 7. The size of the matrix will be 12*8 (11+1 * 7+1). The first row and first column of the matrix can be initially filled with 0 because we assume assumes there is no gap opening or gap extension penalty as shown in fig. 3.

		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0											
G	0											
A	0											
T	0											
C	0											
G	0											
A	0											

Fig. 3. Initial matrix with two sequences

ii. Matrix Fill

One possible way of filling the matrix is to find the maximum global alignment score by starting from the upper left hand corner of the matrix and find the maximal score $M_{i,j}$ for each position in the matrix.

For each position, $M_{i,j}$ is defined to be the maximum score at position i,j i.e.,

$M_{i,j}$ = MAXIMUM[

$M_{i-1, j-1}$ + $S_{i,j}$ (match/mismatch in the diagonal),

$M_{i,j-1}$ + w (gap in sequence #1),

$M_{i-1,j}$ + w (gap in sequence #2)**]**

In fig. 4, $M_{i-1,j-1}$ will be red, $M_{i,j-1}$ will be blue and $M_{i-1,j}$ will be green. The score at position 1,1 in the matrix can be calculated. Since the first residue in both sequences is a G i.e., a match, so score $S_{1,1} = 1$. We assumed the gap penalty as 0.

Thus, $M_{1,1} = MAX[M_{0,0} + 1, M_{1,0} + 0, M_{0,1} + 0] = MAX [1, 0, 0] = 1$.

A value of 1 is then placed in position 1,1 of the scoring matrix.

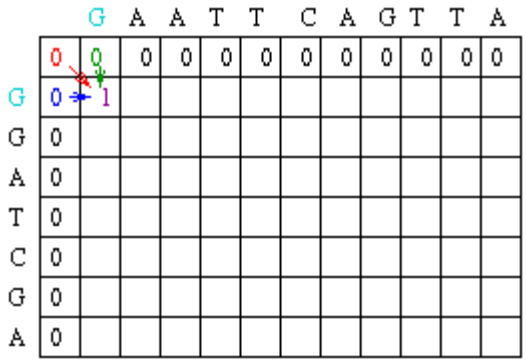


Fig. 4. Sample fill of the entry $M_{1,1}$

Now the element $M_{1,2}$, the value is the max of 0 (for a mismatch), 0 (for a vertical gap) or 1 (horizontal gap). The rest of element of first row can be filled up similarly. At this point, there is a G in both sequences (light blue). Thus, the value for the cell at row 1 column 8 is the maximum of 1 (for a match), 0 (for a vertical gap) or 1 (horizontal gap). The value will again be 1 as in fig. 5

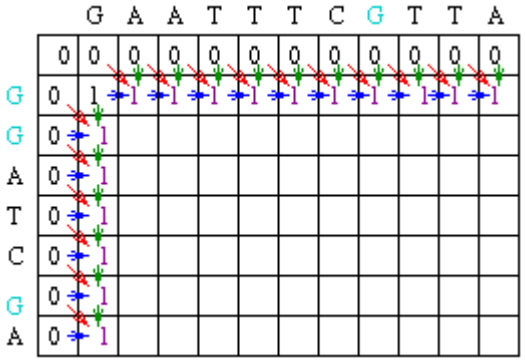


Fig. 5. Sample fill of the entry whene there is a collosion of two cells for $M_{1,8}$

Now similarly at column 2. The location at row 2 will be assigned the value of the maximum of 1(mismatch), 1(horizontal gap) or 1 (vertical gap). So its value is 1.

After filling in all of the values the score matrix is shown in fig. 6:

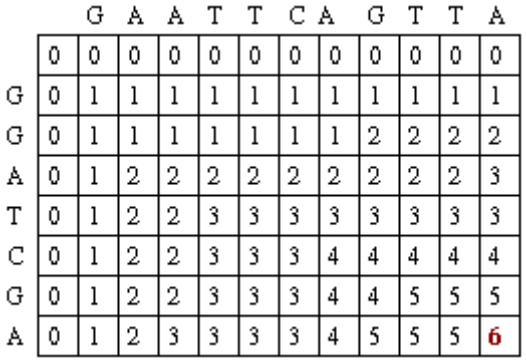


Fig. 6. Final filled matrix

iii. Traceback Step

After the matrix fill step, find the the maximum alignment score for the two test sequences. The traceback step determines the actual alignment(s) that result in the maximum score. Note that with a simple scoring algorithm such as one that is used here, there are likely to be multiple maximal alignments.

The traceback step begins in the matrix that leads to the maximal score. In this case, there is a 6 in that location. Traceback takes the current cell and looks to the neighbor cells that could be direct predecessors. This means that it looks to the neighbor to the left (gap in sequence #2), the diagonal neighbor (match/mismatch), and the neighbor above it (gap in sequence #1). The algorithm for traceback chooses as the next cell in the sequence one of the possible predacessors. In this case, the neighbors are marked in red. They are all also equal to 5 as in fig 7.

		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	1	2	2	2	2
A	0	1	2	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	5	5	5
A	0	1	2	3	3	3	4	5	5	5	5	6

Fig. 7. Traceback process start where the score is maximum

Since the current cell has a value of 6 and the scores are 1 for a match and 0 for anything else, the only possible predecessor is the diagonal match/mismatch neighbor. If more than one possible predecessor exists, any can be chosen. The corresponding row and column can be crossed out as in fig. 8. This gives us a current alignment of

(Seq #1) A

|

(Seq #2) A

		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	
G	0	1	1	1	1	1	1	1	1	1	1	
G	0	1	1	1	1	1	1	1	2	2	2	
A	0	1	2	2	2	2	2	2	2	2	2	
T	0	1	2	2	3	3	3	3	3	3	3	
C	0	1	2	2	3	3	4	4	4	4	4	
G	0	1	2	2	3	3	4	4	5	5	5	
A												6

Fig. 8. Traceback steps and crossing of the row and column

Now, look at the current cell and determine which cell is its direct predecessor. In this case, it is the cell with the red 5 as in fig. 9. The alignment as described in the above step adds a gap to sequence #2 , so the current alignment is

(Seq #1) T A

|

(Seq #2) _ A

Once again, the direct predecessor produces a gap in sequence #2.

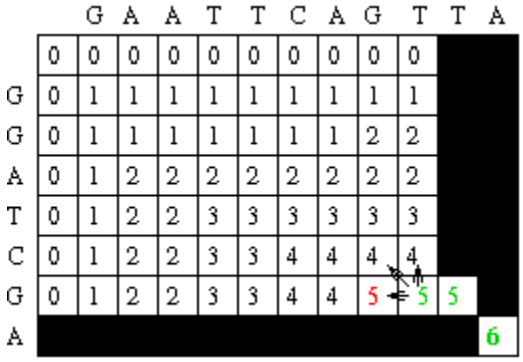


Fig. 9. Traceback steps and crossing of the row and column

After this step, the current alignment is

(Seq #1) T T A

|

_ _ A

Continuing on with the traceback step, we eventually get to a position in row 0 and column 0, which tells us that traceback is completed as in fig. 10.

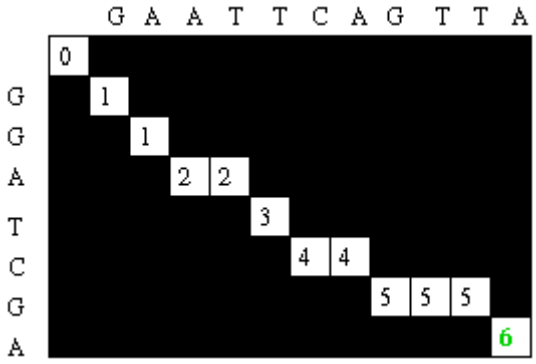


Fig. 10. Final matrix with the traceback steps

One possible maximum alignment is

G A A T T C A G T T A

| | | | | | |

G G A _ T C _ G _ _ A

Local Alignment

Local alignments are more useful for dissimilar sequences that may contains regions of similarity or similar sequence motifs within their larger sequence context. The Smith-Waterman algorithm is a general local alignment method based on dynamic programming. A local alignment searches for regions of local similarity between two sequences and need not include the entire length of the sequences. This can be done by reading a scoring matrix that contains values for every possible residue or nucleotide match or mismatch. The Smith-Waterman algorithm is a member of the class of algorithms that can calculate the best score and local alignment in the order of m*n steps, where 'm' and 'n' are the lengths of the two sequences. Local alignment methods only report the best matching areas between two sequences while there may be a large number of alternative local alignments which do not score as highly as the best alignment done by this algorithm.

Consider the two DNA sequences to be globally aligned are:

ACACACT (x=7, length of sequence 1)

AGCACAC (y=7, length of sequence 2)

It also follows three steps

i. Initialization
ii. Matrix fill (scoring)
iii. Traceback (alignment)

Let us assume the simple scoring scheme as

- $S_{i,j}$ = 2 if there is a match
- $S_{i,j}$ = -1 if there is a mismatch
- w = -1 as gap penalty

i. Initialization

The first step in the global alignment dynamic programming approach is to create a matrix with M + 1 columns and N + 1 rows where M and N correspond to the size of the sequences to be aligned. In this example, we assume that there is no gap opening or gap extension penalty. The first row and first column of the matrix can be initially filled with 0 as in fig. 11.

		A	C	A	C	A	C	T
	0	0	0	0	0	0	0	0
A	0							
G	0							
C	0							
A	0							
C	0							
A	0							
C	0							

Fig. 11. Initial matrix with first row and first column element as 0

ii. Matrix Fill

One way to fill the matrix is to find the maximum global alignment score by starting from the upper left hand corner in the matrix and get the maximal score $M_{i,j}$ for each position in the matrix. In order to find $M_{i,j}$ for any i,j it is minimal to know the score for the matrix positions to the left, above and diagonal to i, j. In terms of matrix positions, it is necessary to know $M_{i-1,j}$, $M_{i,j-1}$ and $M_{i-1,\,j-1}$.

For each position, $M_{i,j}$ is defined to be the maximum score at position i,j; i.e.

$M_{i,j}$ = MAXIMUM[

$M_{i-1,\,j-1}$ + $S_{i,j}$ (match/mismatch in the diagonal),

$M_{i,j-1}$ + w (gap in sequence #1),

$M_{i-1,j}$ + w (gap in sequence #2)**]**

Using this information, the score at position 1,1 in the matrix can be calculated. Since the first residue in both sequences is A, $S_{1,1}$ = 2, and by the assumptions stated at the beginning, w = 0. Thus, $M_{1,1}$ = MAX[$M_{0,0}$ + 2, $M_{1,\,0}$ -1, $M_{0,1}$ -1] = MAX [2, -1, -1] = 2.

A value of 2 is then placed in position 1,1 of the scoring matrix as in fig. 12. And subsequently the whole matrix is filled in the same way.

		A	C	A	C	A	C	T
	0	0	0	0	0	0	0	0
A	0	2	1	2	1	2	1	0
G	0	1	1	1	1	1	1	0
C	0	0	3	2	3	2	3	2
A	0	2	2	5	4	5	4	3
C	0	1	4	4	7	6	7	6
A	0	2	3	6	6	9	8	7
C	0	1	4	5	8	8	11	10

Fig. 12. Final filled matrix

iii. Traceback

After the matrix fill step, the maximum alignment score for these two test sequences is 11. The traceback step determines the actual alignment(s) for the maximum score. It is not mandatory that the last cell has the maximum alignment score.

The traceback step begins with the position that leads to the maximal score. In this case, there is 11 in that location.

Trace back takes the current cell and looks to the neighbor cells that could be direct predecessors. This means it looks to the neighbor to the left (gap in sequence #2), the diagonal neighbor (match/mismatch), and the neighbor above it (gap in sequence #1) as in fig. 13. The algorithm for trace back chooses as the next cell in the sequence one of the possible predecessors. This continues till cell with value 0 is reached.

		A	C	A	C	A	C	T
	0	0	0	0	0	0	0	0
A	0	2	1	2	1	2	1	0
G	0	1	1	1	1	1	1	0
C	0	0	3	2	3	2	3	2
A	0	2	2	5	4	5	4	3
C	0	1	4	4	7	6	7	6
A	0	2	3	6	6	9	8	7
C	0	1	4	5	8	8	11	10

Fig. 13. Traceback Step

The only possible predecessor is the diagonal match/mismatch neighbor. If more than one possible predecessor exists, any can be chosen. This gives us a current alignment of

(Seq #1) C
|
(Seq #2) C

So now we look at the current cell and determine which cell is its direct predecessor. In this case, it is the cell with the red 9 as in fig. 14.

(Seq #1) C A
| |
(Seq #2) C A

		A	C	A	C	A	C	T
	0	0	0	0	0	0	0	0
A	0	2	1	2	1	2	1	0
G	0	1	1	1	1	1	1	0
C	0	0	3	2	3	2	3	2
A	0	2	2	5	4	5	4	3
C	0	1	4	4	7	6	7	6
A	0	2	3	6	6	9	8	7
C	0	1	4	5	8	8	11	10

Fig. 14. Traceback step with the correct arrows

Continuing with the traceback step, we eventually get a position in column 0 or row 0 which tells us that traceback is completed as in fig. 15.

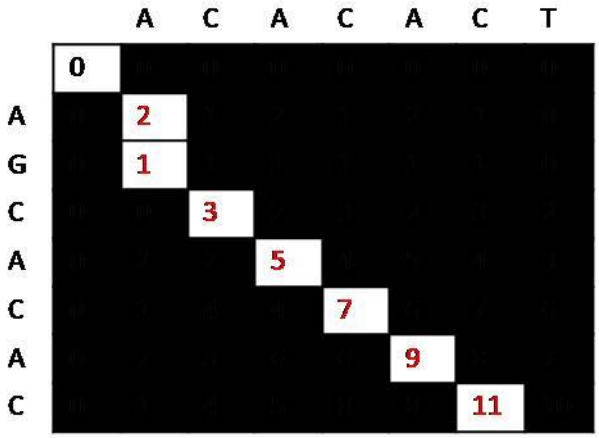


Fig. 15. Final Traceback Matrix

The possible maximum alignment is:

AG C A C A C
| | | | | |
A _ C A C A C

There is a combination of these two methods which is called hybrid methods, also known as semiglobal or "glocal" methods. This method attempts to find the best possible alignment that includes the start and end of one or the other sequence. This can be especially useful when the downstream part of one sequence overlaps with the upstream part of the other sequence. In this case, neither global nor local alignment is entirely appropriate.

4. Significance of Sequence Alignment

Sequence alignments are useful in bioinformatics for identifying sequence similarity, producing phylogenetic trees, and developing homology models of protein structures. However, the biological relevance of sequence alignments is not always clear. Alignments are often assumed to reflect a degree of evolutionary change between sequences descended from a common ancestor; however, it is formally possible that convergent evolution can occur to produce apparent similarity between proteins that are evolutionarily unrelated but perform similar functions and have similar structures.

In database searches such as BLAST, statistical methods can determine the likelihood of a particular alignment between sequences or sequence regions arising by chance with the given the size and composition of the database being searched. These values can vary significantly depending on the search space. In particular, the likelihood of finding a given alignment by chance increases, if the database consists only of sequences from the same organism as the query sequence. Repetitive sequences in the database or query can also distort both the search results and the assessment of statistical significance. BLAST automatically filters such repetitive sequences in the query to avoid apparent hits that are statistical artifacts.

The choice of a **scoring function** that reflects biological or statistical observations about known sequences is important to producing good alignments. Protein sequences are frequently aligned using substitution matrices that reflect the probabilities of given character-to-character substitutions. A series of matrices called PAM matrices (Point

Accepted Mutation matrices, originally defined by Margaret Dayhoff and sometimes referred to as "Dayhoff matrices") explicitly encode evolutionary approximations regarding the rates and probabilities of particular amino acid mutations. Another common series of scoring matrices, known as BLOSUM (Blocks Substitution Matrix), encodes empirically derived substitution probabilities. Variants of both types of matrices are used to detect sequences with differing levels of divergence, thus allowing users of BLAST or FASTA to restrict searches to more closely related matches or expand to detect more divergent sequences. Gap penalties account for the introduction of a gap - on the evolutionary model, an insertion or deletion mutation - in both nucleotide and protein sequences, and therefore the penalty values should be proportional to the expected rate of such mutations. The quality of the alignments produced therefore depends on the quality of the scoring function.

5. Sequence Databases

The repositories for the genomic sequences are

National Center for Biotechnology Information (**NCBI**) is part of the United States National Library of Medicine (NLM), a branch of the National Institutes of Health. The NCBI is located in Bethesda, Maryland and was founded in 1988 through legislation sponsored by Senator Claude Pepper. The NCBI houses genome sequencing data in GenBank and an index of biomedical research articles in PubMed Central and PubMed, as well as other information relevant to biotechnology. All these databases are available online through the Entrez search engine. The NCBI is directed by David Lipman, one of the original authors of the BLAST sequence alignment program and a widely respected figure in Bioinformatics. The NCBI has had responsibility for making available the GenBank DNA sequence database since 1992 as shown in fig. 16. GenBank coordinates with individual laboratories and other sequence databases such as those of the European Molecular Biology Laboratory (EMBL) and the DNA Data Bank of Japan (DDBJ). Since 1992, NCBI has grown to provide other databases in addition to GenBank. NCBI provides Online Mendelian Inheritance in Man, the Molecular Modeling Database (3D protein structures), dbSNP a database of single-nucleotide polymorphisms, the Unique Human Gene Sequence Collection, a Gene Map of the human genome, a Taxonomy Browser, and coordinates with the National Cancer Institute to provide the Cancer Genome Anatomy Project.

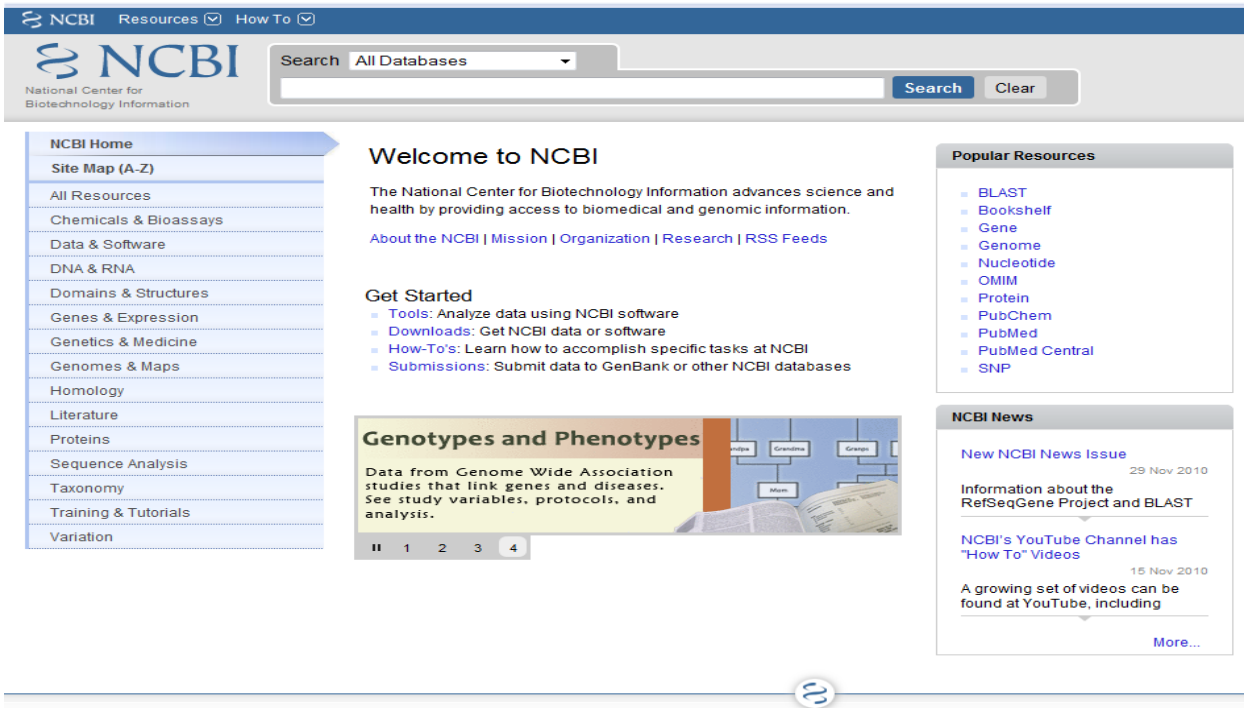


Fig. 16. NCBI portal

The NCBI assigns a unique identifier (Taxonomy ID number) to each species of organism. The NCBI has software tools that are available by WWW browsing or by FTP. For example, BLAST is a sequence similarity searching program. BLAST can do sequence comparisons against the GenBank DNA database in less than 15 seconds. The **NCBI Bookshelf** is a collection of freely available, downloadable, on-line versions of selected biomedical books. The Bookshelf has various titles covering aspects of molecular biology, biochemistry, cell biology, genetics, microbiology, a couple of disease states from a molecular and cellular point of view, research methods, and virology. Some of the books are online versions of previously published books, while others, such as Coffee Break (book), are written and edited by NCBI staff. The Bookshelf is a complement to the Entrez PubMed repository of peer-reviewed publication abstracts in that Bookshelf contents provide established perspectives on evolving areas of study and a context in which many disparate individual pieces of reported research can be organized.

European Molecular Biology Laboratory (**EMBL**) is a molecular biology research institution supported by 20 European countries and Australia as associate member state. The EMBL was created in 1974 and is a non-profit organisation funded by public research money from its member states. Research at EMBL is conducted by approximately 85 independent groups covering the spectrum of molecular biology. The Laboratory operates from five sites: the main Laboratory in Heidelberg, and Outstations in Hinxton (the European Bioinformatics Institute (**EBI**)), Grenoble, Hamburg, and Monterotondo near Rome as in fig. 17. Each of the sites has a research specific field. At EBI, the research is oriented towards computational biology and bioinformatics. At Grenoble and Hamburg the research is in the field of structural biology. At Monterotondo the research is focused mainly on mouse models for clinical research. At the headquarters in Heidelberg, there are big departments in Cell Biology and Gene Expression as well as smaller complementing the aforementioned research fields.

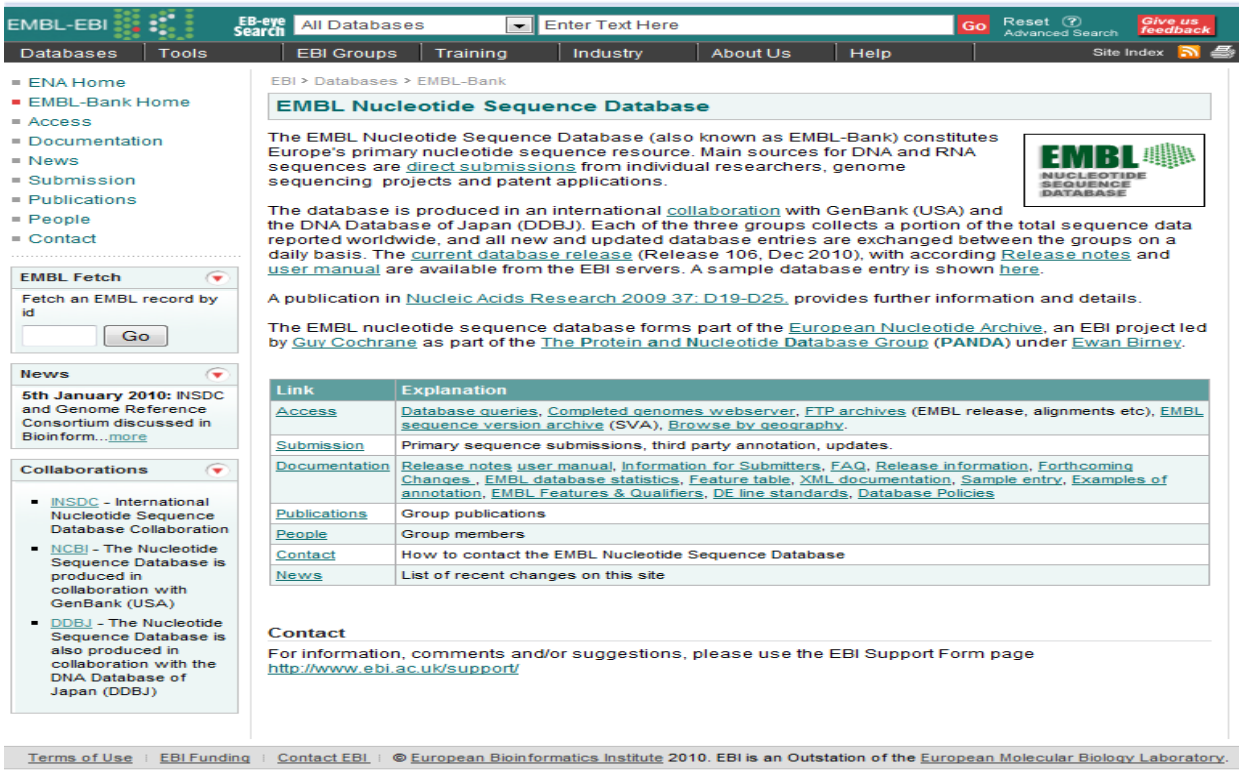


Fig. 17. EMBL portal

The cornerstones of EMBL's mission are: to perform basic research in molecular biology and molecular medicine, to train scientists, students and visitors at all levels, to offer vital services to scientists in the member states, to develop new instruments and methods in the life sciences, and to actively engage in technology transfer. EMBL's international PhD Programme has a student body of about 170. The Laboratory also sponsors an active Science and Society programme. Many scientific breakthroughs have been made at EMBL, most notably the first systematic genetic analysis of embryonic development in the fruit fly by Christiane Nüsslein-Volhard and Eric Wieschaus, for which they were awarded the Nobel Prize for Medicine in 1995.

DNA Data Bank of Japan (DDBJ) is a DNA data bank. It is located at the National Institute of Genetics (NIG) in the Shizuoka prefecture of Japan. It is also a member of the International Nucleotide Sequence Database Collaboration or INSDC. It exchanges its data with European Molecular Biology Laboratory at the European Bioinformatics Institute and with GenBank at the National Center for Biotechnology Information on a daily basis. Thus these three databanks contents the same data at any given time. DDBJ began data bank activities since 1986 at NIG and it boasts to be the only nucleotide sequence data bank in Asia. Although DDBJ mainly receives its data from Japanese researchers, however it can accept data from a contributor belonging to any other country as in fig. 18.

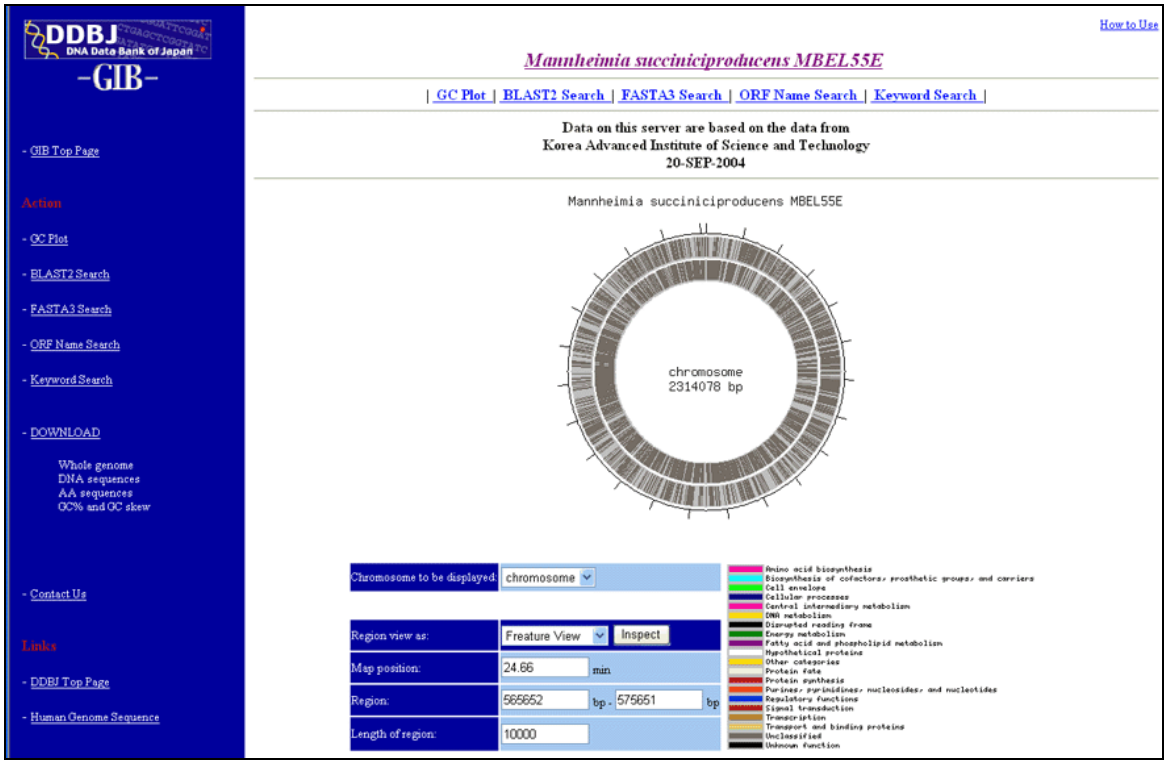


Fig. 18. DDBJ Portal

DDBJ is primarily funded by the Japanese Ministry of Education, Culture, Sports, Science and Technology (MEXT). DDBJ has an international advisory committee which consists of nine members, 3 members each from Europe, US, and Japan. This committee advice DDBJ about its maintenance, management and future plans once a year. Apart from this DDBJ also has an international collaborative committee which advises on various technical issues related to international collaboration and consists of working-level participants.

6. Softwares Used in Sequence Alignment

S. No.	Name	Function	Website Link
1	ALIGN	Sequence Analysis	http://www.ebi.ac.uk/Tools/emboss/align
2	CENSOR	Sequence Analysis	http://www.ebi.ac.uk/Tools/censor/
3	CLUSTALW2	Sequence Analysis	http://www.ebi.ac.uk/Tools/clustalw2/
4	CpG Plot/ CpGreport	Sequence Analysis	http://www.ebi.ac.uk/Tools/emboss/cpgplot/
5	Genewise	Sequence Analysis	http://www.ebi.ac.uk/Tools/Wise2/
6	Kalign	Sequence Analysis	http://www.ebi.ac.uk/Tools/kalign

7	MAFFT	Sequence Analysis	http://www.ebi.ac.uk/Tools/mafft/
8	MUSCLE	Sequence Analysis	http://www.ebi.ac.uk/Tools/muscle/
9	Pepstats/ Pepwindow/Pepinfo	Sequence Analysis	http://www.ebi.ac.uk/Tools/emboss/pepinfo/
10	PromoterWise	Sequence Analysis	http://www.ebi.ac.uk/Tools/Wise2/promoterwise.html
11	SAPS	Sequence Analysis	http://www.ebi.ac.uk/Tools/saps/
12	T-coffee	Sequence Analysis	http://www.ebi.ac.uk/Tools/t-coffee/
13	Transeq	Sequence Analysis	http://www.ebi.ac.uk/Tools/emboss/transeq/
14	COBALT	Sequence Analysis	http://www.ncbi.nlm.nih.gov/tools/cobalt/
15	Genome Workbench	Sequence Analysis	http://www.ncbi.nlm.nih.gov/projects/gbench/
16	ORF Finder	Sequence Analysis	http://www.ncbi.nlm.nih.gov/gorf/gorf/html
17	Primer - BLAST	Sequence Analysis	http://www.ncbi.nlm.nih.gov/tools/primer-blast
18	ProSplign	Sequence Analysis	http://www.ncbi.nlm.nih.gov/sutils/static/prosplin/prosplign.html
19	Splign	Sequence Analysis	http://www.ncbi.nlm.nih.gov/sutils/splign/
20	VecScreen	Sequence Analysis	http://www.ncbi.nlm.nih.gov/VecScreen/VecScreen.html
21	Sequence Analysis	Sequence analysis	http://www.informagen.com/SA/
22	SeWeR	Sequence analysis	http://www.bioinformatics.org/sewer/
23	Motif Search	Sequence analysis	http://nbc11.biologie.uni-kl.de/framed/left/menu/auto/right/motifsearch2/ index.pl

24	DNA Translator	Sequence analysis	http://nbc11.biologie.uni-kl.de/framed/left/menu/auto/right/JDT/
25	Non coding RNA Gene Finder	Sequence analysis	http://nbc11.biologie.uni-kl.de/framed/left/menu/auto/right/ncRnaGeneFinder/index.pl
26	TransTerm	Sequence analysis	http://nbc11.biologie.uni-kl.de/framed/left/menu/auto/right/transterm/
27	QRNA	Sequence analysis	http://nbc11.biologie.unikl.de/framed/left/menu/auto/right/qrna/
28	Clustalformatter 5	Sequence analysis	http://nbc11.biologie.uni-kl.de/framed/left/menu/auto/right/ClustalFormatter/
29	BioEdit	Sequence Alignment Editor	http://www.mbio.ncsu.edu/BioEdit/bioedit.html
30	FASTA	Sequence Similarity Search	http://www.ebi.ac.uk/Tools/fasta/
31	HMMER	Homology of protein	http://hmmer.janelia.org/
32	JAligner	Pairwise seq. alignment	http://jaligner.sourceforge.net/
33	JSTRING	Java Search for Tandem Repeats IN Genomes	http://bioinf.dms.med.uniroma1.it/JSTRING/
34	NCBI BLAST	Aligning Sequences	http://blast.ncbi.nlm.nih.gov/Blast.cgi
35	Gene Runner/ Motif Runner	Motif based sequence analysis	http://www.generunner.net/
36	GoCore	Protein Seq. Alignment & Analysis	http://www.helsinki.fi/project/ritvos/GoCore/
37	MAFFT	Multiple alignment	http://mafft.cbrc.jp/alignment/server/index.html
38	MAUVE	Multiple alignment	http://gel.ahabs.wisc.edu/mauve/
39	MEME Suite	Motif based sequence analysis	http://meme.nbcr.net/

40	CORAL (CDTree)	Aligning Core Conserved Regions	http://www.ncbi.nlm.nih.gov/Structure/cdtree/cdtree.shtml
41	BlastAlign	Align N Seq. with large INDELs	http://www.bioafrica.net/blast/BlastAlign.html
42	ARB software	Sequence DB Handling and Data Analysis	http://www.arb-home.de/
43	Automated Codon Usage Analysis Software - ACUA	Nucleotide Analysis	http://www.bioinsilico.com/acua
44	AnnHyb	Nucleotide Analysis	http://www.bioinformatics.org/annhyb/
45	SOAP2	Short read Alignment	http://soap.genomics.org.cn/
46	ACT (Artemis Comparison Tool)	DNA Sequence Comparison	http://www.sanger.ac.uk/resources/software/act/
47	WU-BLAST	Multiple Sequence Alignment	www.ebi.ac.uk/Tools/blast2/
48	CLUSTALW2	multiple sequence alignment	http://www.ebi.ac.uk/Tools/clustalw2/

REFERENCES

www.wikipedia.org/

cnx.org/content/m11026/latest/

www.ncbi.nlm.nih.gov/

www.ebi.ac.uk/embl/

www.ddbj.nig.ac.jp/

TRANSCRIPTOME ANALYSIS

Mohammad Samir Farooqi
ICAR-IASRI, Library Avenue, New Delhi – 110 012

Introduction

The advent of Next-Generation Sequencing (NGS) technology has transformed genomic studies. One important application of NGS technology is the study of the *transcriptome*, which is defined as the complete collection of all the RNA molecules in a cell. Various types of RNA that have been classified so far are shown in **Fig. 1**. All of these molecules are called *transcripts* since they are produced by process of transcription.

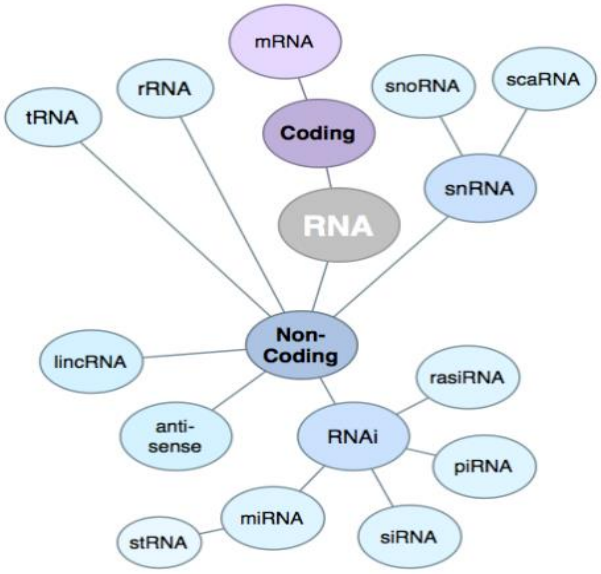


Fig. 1: Different types of RNA
(Image source: http://scienceblogs.com/digitalbio/2011/01/08/next-gene-sequencing)

Understanding the transcriptome is essential for interpreting the functional elements of the genome and revealing the molecular constituents of cells and tissues, and also for understanding development and disease [1]. The main purpose of transcriptomics are: to catalogue all species of transcript, including mRNAs, non-coding RNAs and small RNAs; to determine the transcriptional structure of genes, in terms of their start sites, 5′ and 3′ ends, splicing patterns and other post-transcriptional modifications; and to quantify the changing expression levels of each transcript during development and under different conditions.

The study of transcriptome is carried out through sequencing of RNAs. *RNA sequencing (RNA-Seq)* is a powerful method for discovering, profiling, and quantifying RNA transcripts [2]. RNA-Seq uses NGS datasets to obtain sequence reads from millions of individual RNAs. The RNA-Seq analysis is performed in several steps: First, all genes are extracted from the reference genome (using annotations of type *gene*). Other annotations on the gene sequences are preserved (e.g. CDS information about coding sequences etc). Next, all annotated

transcripts (using annotations of type *mRNA*) are extracted [3]. If there are several annotated splice variants, they are all extracted. An example is shown in below **Fig. 2(a).**

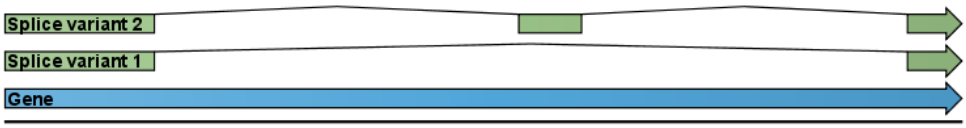


Fig. 2(a): A simple gene with three exons and two splice variants.

The given example is a simple gene with three exons and two splice variants. The transcripts are extracted as shown in **Fig. 2(b).**

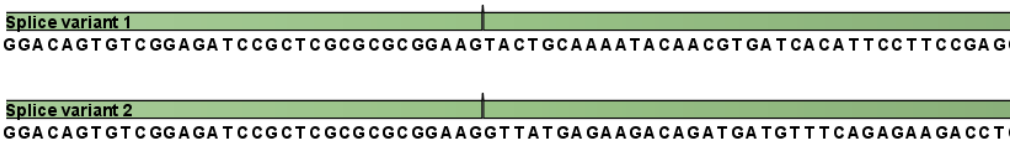


Fig. 2(b): All the exon-exon junctions are joined in the extracted transcript.

Next, the reads are mapped against all the transcripts plus the entire gene [see **Fig. 2(c)**].

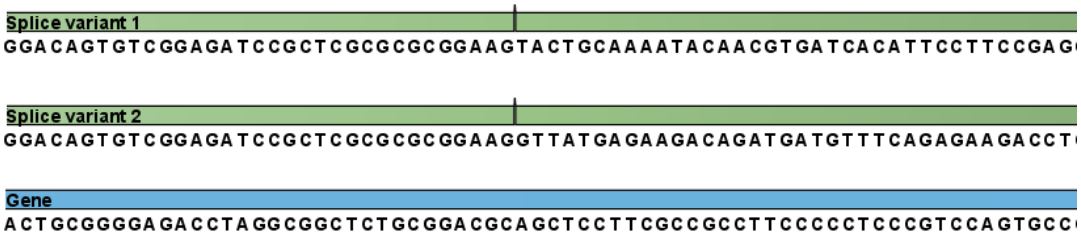


Fig. 2(c): The reference for mapping: all the exon-exon junctions and the gene.

(Image source: CLC Genomic workbench tutorials)

From this mapping, the reads are categorized and assigned to the genes and expression values for each gene and each transcript are calculated and putative exons are then identified.

RNA Sequencing Experiment

In a standard RNA-seq experiment, a sample of RNA is converted to a library of complementary DNA fragments and then sequenced on a high-throughput sequencing platform, such as Illumina's Genome Analyzer, SOLiD or Roche 454 [4]. Millions of short sequences, or reads, are obtained from this sequencing and then mapped to a reference genome (**Fig. 3**). The count of reads mapped to a given gene measures the expression level of this gene. The unmapped reads are usually discarded and mapped reads for each sample are assembled into gene-level, exon-level or transcript-level expression summaries, depending on the objectives of the experiment. The count of reads mapped to a given gene/exon/transcript measures the expression level for this region of the genome or transcriptome.

One of the primary goals for most RNA-seq experiments is to compare the gene expression levels across various treatments. A simple and common RNA-seq study involves two

treatments in a randomized complete design, for example, treated versus untreated cells, two different tissues from an organism, plants, etc. In most of the studies, researchers are particularly interested in detecting gene with differential expressions (DE). A gene is declared differentially expressed if an observed difference or change in read counts between two experimental conditions is statistically significant, i.e. if the difference is greater than what would be expected just due to random variation [5]. Detecting DE genes can also be an important pre-step for subsequent studies, such as clustering gene expression profiles or testing gene set enrichments.

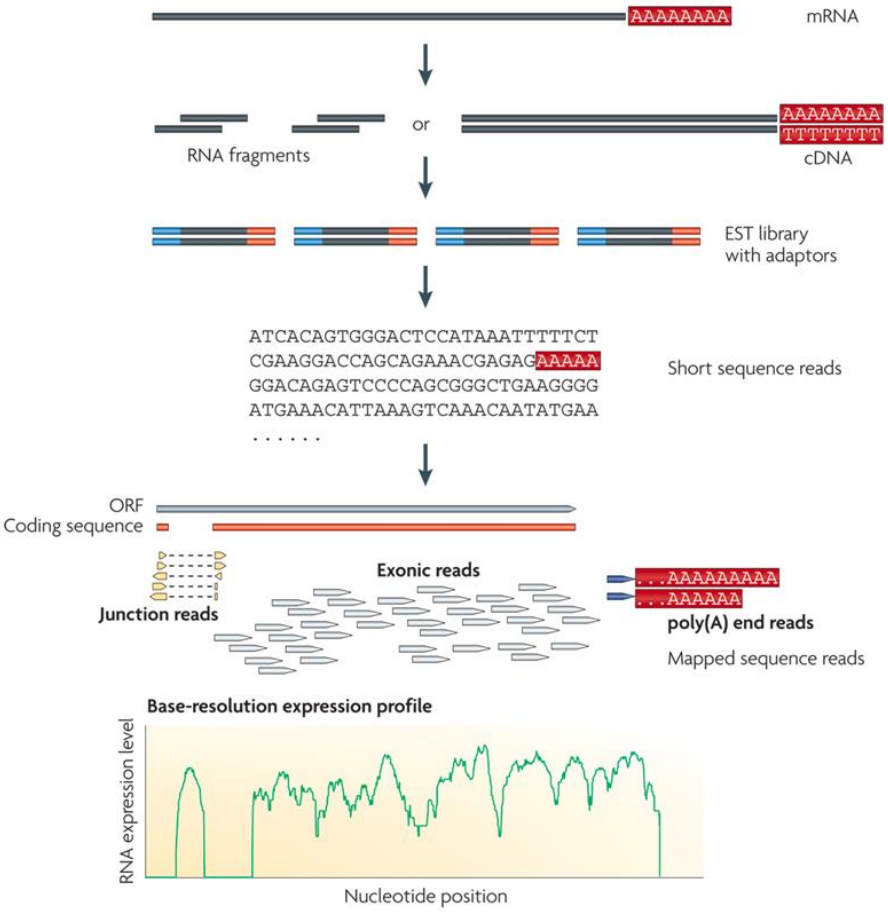


Fig. 3: General RNA-seq experiment. mRNA is converted to cDNA, and fragments from that library are used to generate short sequence reads. Those reads are assembled into contigs which may be mapped to reference sequences (Wang et al., 2009).

Analysing RNA-Seq data

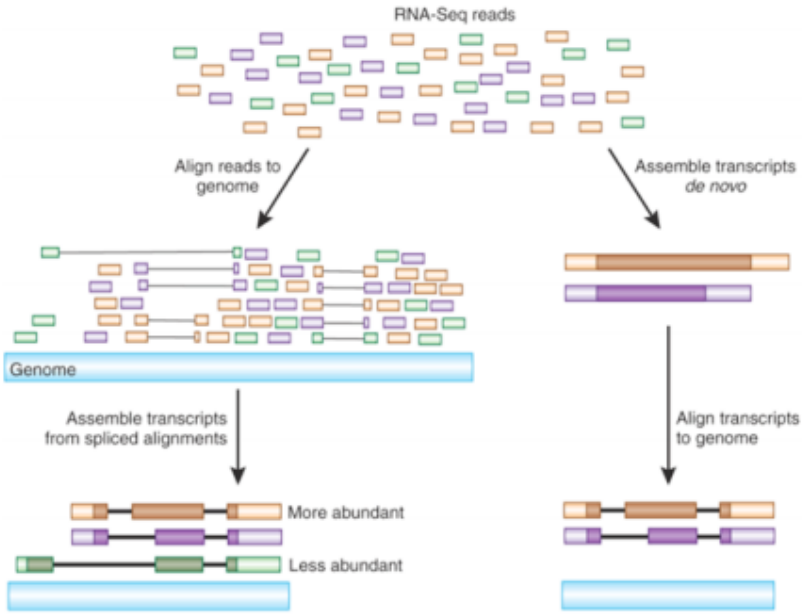
RNA-seq experiments must be analyzed with robust, efficient and statistically correct algorithms. Fortunately, the bioinformatics community has been striving hard at work for incorporating mathematics, statistics and computer science for RNA-seq and building these ideas into software tools. RNA-seq analysis tools generally fall into three categories: (i) those for read alignment; (ii) those for transcript assembly or genome annotation; and (iii) those for transcript and gene quantification. Some of the open source softwares available for RNA-seq analysis are as follows:

- **Data preprocessing**

    - Fastx toolkit
    - Samtools
- **Short reads aligners**
    - Bowtie, TOPHAT, Stampy, BWA, Novoalign, etc
- **Expression studies**
    - Cufflinks package
    - R packages (DESeq, edgeR, *more…*)
- **Visualisation**
    - CummeRbund, IGV, Bedtools, UCSC Genome Browser, etc.

Besides there are commercially data analysis pipelines like GenomeQuest, CLCBio etc available for researchers to use. The most commonly used pipeline is to identify protein coding genes by aligning RNA-Seq data to annotate data from sources like RefSeq. After generating the alignments, the number of aligning sequences is counted for each position. Since each alignment represents a transcript, the alignments allow to count the number of RNA molecules produced from every gene.

Using NGS technology, RNA-Seq enables to count the number of reads that align to one of thousands of different cDNAs, producing results similar to those of gene expression microarrays [6]. Sequences generated from an RNA-Seq experiment are usually mapped to libraries of known exons in known transcripts. RNA-Seq can be used for discovery applications such as identifying alternative splicing events, allele-specific expression, and rare and novel transcripts [7]. The sequencing output files (compressed FASTQ files) are the input for secondary analysis. Reads are aligned to an annotated reference genome, and those aligning to exons, genes and splice junctions are counted. The final steps are data visualisation and



interpretation, consisting of calculating gene- and transcript-expression and reporting differential expression. A general Bioinformatics workflow to map transcripts from RNA-seq data is shown in **Fig. 4**.

Fig. 4: RNA-seq workflow (Adapted from Advancing RNA-Seq analysis Brian J. Haas and Michael C. Zody Nature Biotechnology 28, 421-423 (2010)

RPKM (Reads per KB per million reads)

RNA-Seq provides quantitative approximations of the abundance of target transcripts in the form of counts. However, these counts must be normalized to remove technical biases inherent in the preparation steps for RNA-Seq, in particular the length of the RNA species and the sequencing depth of a sample. The most commonly used is RPKM (Reads Per Kilobase of exon model per Million mapped reads). The RPKM measure of read density reflects the molar concentration of a transcript in the starting sample by normalizing for RNA length and for the total read number in the measurement [8]. RPKM is mathematically represented as:

$$\text{RPKM} = \frac{total\ exon\ reads}{mapped\ reads\ (millions)\ \text{X}\ exon\ length\ (KB)}$$

Total exon reads

This is the number of reads that have been mapped to a region in which an exon is annotated for the gene or across the boundaries of two exons or an intron and an exon for an annotated transcript of the gene. For eukaryotes, exons and their internal relationships are defined by annotations of type mRNA.

Exon length

This is calculated as the sum of the lengths of all exons annotated for the gene. Each exon is included only once in this sum, even if it is present in more annotated transcripts for the gene. Partly overlapping exons will count with their full length, even though they share the same region.

Mapped reads

The total gene reads for a gene is the total number of reads that after mapping have been mapped to the region of the gene. A gene's region is that comprised of the flanking regions, the exons, the introns and across exon-exon boundaries of all transcripts annotated for the gene. Thus, the sum of the total gene reads numbers is the number of mapped reads for the sample.

Applications of RNA-seq

This technique can be used to:

- Measure gene expression
- Transcriptome assembly, gene discovery and annotation
- Detect differential transcript abundances between tissues, developmental stages, genetic backgrounds, and environmental conditions
- Characterize alternative splicing, alternative polyadenylation, and alternative transcription.

Future Directions

Although RNA-Seq is still in the infancy stages of use, it has clear advantages over previously developed transcriptomic methods. Compared with microarray, which has been the dominant approach of studying gene expression in the last two decades, RNA-seq technology has a wider measurable range of expression levels, less noise, higher throughput, and more information to detect allele-specific expression, novel promoters, and isoforms [9]. For these reasons, RNA-seq is gradually replacing the array-based approach as the major platform in gene expression studies. The next big challenge for RNA-Seq is to target more complex transcriptomes to identify and track the expression changes of rare RNA isoforms from all genes. Technologies that will advance achievement of this goal are pair-end sequencing, strand-specific sequencing and the use of longer reads to increase coverage and depth. As the cost of sequencing continues to fall, RNA-Seq is expected to replace microarrays for many applications that involve determining the structure and dynamics of the transcriptome.

References

1. https://www.genome.gov/13014330
2. Wang Z., Gerstein M., Synder M. (2009). Rna-seq: a revolutionary tool for transciptomics, Nat Rev Genet 10(1): 57–63.
3. http://scienceblogs.com/digitalbio/2011/01/08/next-gene-sequencing-results-a/
4. Shendure J, Ji H (2008) Next-generation RNA sequencing. Nature Biotechnology 26: 2514-2521
5. Anders S, Huber W (2010). Differential expression analysis for sequence count data. Genome Biol. 11:R106.
   Illumina, Inc,. (2011). Getting started with RNA-Seq Data Analysis. Pub. No. 470-2011-003.
6. Illumina, Inc,. (2011). RNA-Seq Data Comparison with Gene Expression Microarrays. A cross-platform comparison of differential gene expression analysis. Pub. No. 470-2011-004
7. Yaqing Si (2012). Statistical analysis of RNA-seq data from next-generation sequencing technology. PhD thesis. Iowa State University, Ames, Iowa.

8. Mortazavi, A., Williams, B. A., McCue, K., Schaeffer, L., and Wold, B. (2008). Mapping and quantifying mammalian transcriptomes by rna-seq. Nat Methods, 5(7):621-628.
9. Wang L., Si Y., Dedow L.K., Shao Y., Liu P., Brutnell T.P. (2010). A low-cost library construction protocol and data analysis pipeline for Illumina-based strand-specific multiplex RNA-seq. PLoS One 6(10):e26426.
10. Brian J. H. and Michael C. Z. (2010). Advancing RNA-Seq analysis Nature Biotechnology 28, 421-423.

ASHOKA: The Supercomputing Hub

K.K. Chaturvedi

Introduction

First HPC systems were vector-based systems (e.g. Cray)named 'supercomputers' because they were an order of magnitude more powerful than commercial systems. The 'supercomputer', a large systems are just scaled up versions of smaller systems. High performance computingcan mean high flop countper processor and totalled over many processors working on the same or related problems. This can have faster turnaround time, more powerful system, scheduled to first available system(s) and using multiple systems simultaneously. The HPC is any computational technique that solves a large problem faster than possible using single, commodity systems, Custom-designed, high-performance processors, Parallel computing, Distributed computing and Grid computing.

Parallel computing is a single system with many processors working on the common task. The Distributed computing is configured as many systems loosely coupled by a scheduler to work on related problems and Grid Computing is defined as many systems tightly coupled by software and networks to work together on single problems or on related problems.

Parallel computer is a computer that contains multiple processors where each processor works on its section of the problem and allowed to exchange information with other processors.

Two big advantages of parallel computers are performance and memory. Parallel computers enable us to solve problems that benefit from or require, fast solution, require large amounts of memory and both.

As per the Moore's Law 'predicts' that single processor performance doubles every 18 months, eventually physical limits on manufacturing technology will be reached as in figure 1.

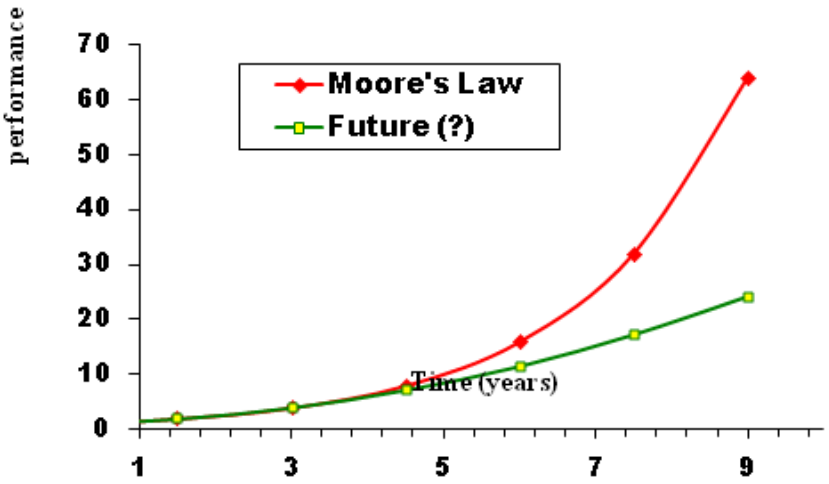


Fig. 1: Moore's Law towards performance of the system

There are two types of parallel computers by their memory model namely shared memory and distributed memory.All processors have access to a pool of shared memory (Figure 2-A) while each processor has it's own local memory indistributed memory (Figure 2-B).

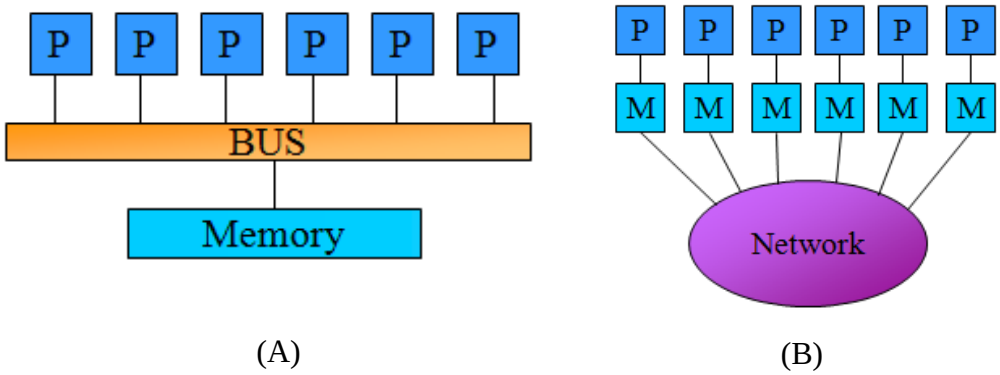


Fig. 2: Shared Memory and distributed memory system

Shared memory have two types of architecture i.e., Uniform memory access (UMA) and Non-uniform memory access (NUMA). Each processor has uniform access to memory in UMA and also called as symmetric multiprocessors, or SMPs (Figure 3-A). Time for memory access depends on location of data in NUMA as local access is faster than non-local access but it is easy to scale up than SMPs (Figure 3-B).

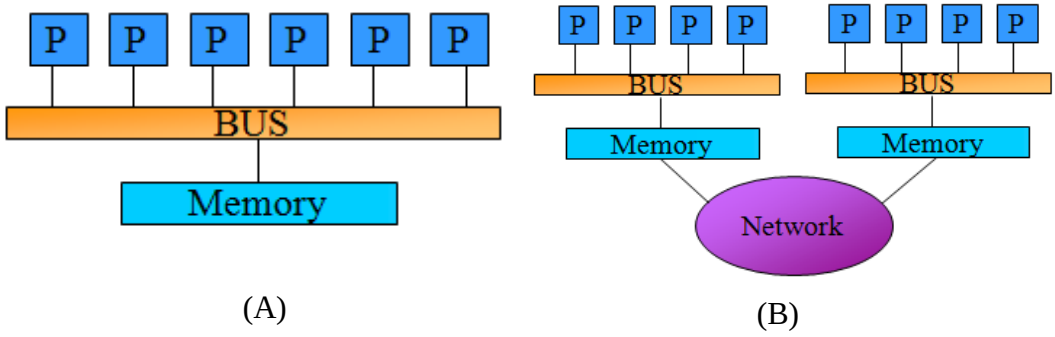


Fig. 3: Shared Memory with UMA and NUMA

The distributed memory is two types namely Massively Parallel Processor (MPP) and cluster. MPP is tightly integrated, single system image and cluster isan individual computers connected by specialized softwareand connected using interconnect network. Distributed memory is shown in figure 4.

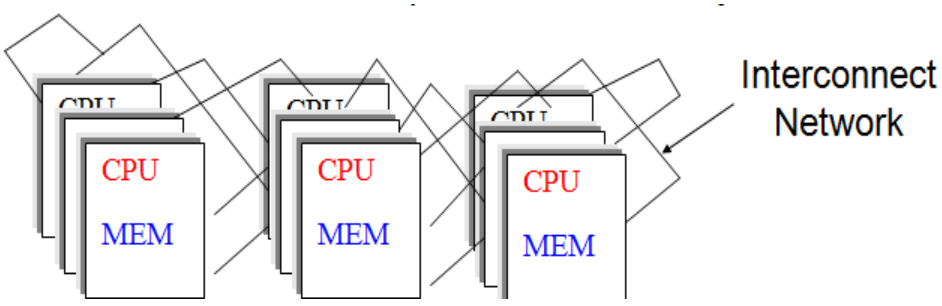


Fig. 4: Distributed Memory

Both types of memory systems haveprocessors, memory and network/interconnect.

Terminology

Clock period (cp): The minimum time interval between successive actions in the processor. It is measured in nanoseconds (~1-5 for fastest processors) which is inverse of frequency (MHz).

Instruction: An action executed by a processor, such as a mathematical operation or a memory operation.

Register: A small and extremely fast location for storing data or instructions in the processor.

Functional Unit (FU): A hardware element that performs an operation on an operand or pair of operations. Common FUs are ADD, MULT, INV, SQRT, etc.

Pipeline: A Technique enables multiple instructions to be overlapped during execution.

Superscalar: Multiple instructions are possible per clock period.

Flops: Floating point operations per second.

Cache: A Fast memoryin the processor which keep instructions and data close to functional units so processor can execute more instructions more rapidly.

SRAM: Static Random Access Memory (RAM). Very fast (~10 nanoseconds), made using the same kind of circuitry as the processors, so speed is comparable.

DRAM: Dynamic RAM. Longer access times (~100 nanoseconds), but hold more bits and are much less expensive (10x cheaper).

Memory hierarchy: The hierarchy of memory in a parallel system, from registers to cache to local memory to remote memory.

Networks Latency: How long does it take to start sending a "message"? Measured in microseconds.

Networks Processors: How long does it take to output results of some operations, such as floating point add, divide etc., which are pipelined?)

Networks Bandwidth: What data rate can be sustained once the message is started? Measured in Mbytes/sec or Gbytes/sec

Types of Clusters/Processors

Symmetric Multiprocessors (SMPs)connect processors to global shared memory using either bus or crossbar. It provides simple programming model, but has problems with buses can become saturated and crossbar size must increase with number of processors. Problem grows with number of processors, limiting maximum size of SMPs. Programming models are easier since message passing is not necessary. The techniques are auto-parallelization via compiler options, loop-level parallelism via compiler directives, OpenMP, and pthreads.

In MPP, each processor has its own memory and is not shared globally but the processors adds another layer to memory hierarchy (remote memory). The processor/memory nodes are connected by interconnect network using many possible topologies. The processors must pass data via messages so the communication overhead can be minimized. Many vendors have custom interconnects that provide high performance for their MPP system such as Gigabit Ethernet, Fast Ethernet, etc.

Clusters are similar to MPPs with processors and memory. The processor performance must be maximized and memory hierarchy needs remote memory as no shared memory for message passing to avoid the communication overhead.

Clusters are different from MPPs as commodity processors including interconnect and OS with multiple independent systems and separate I/O systems. The advantages of clusters are inexpensive, fastest processors first, potential for true parallel I/O and high availability while the disadvantages are less mature software (programming and system), more difficult to manage (changing slowly), lower performance interconnects (not as scalable to large number).

Distributed Memory Programmingprovides message passing using MPI, MPI-2 and active/one-sided messages.

There are two types of parallelism i.e., data and task. Each processor performs the same task on different sets or sub-regions of data in data parallelism. Each processor performs a different task in task parallelism. Most parallel applications fall somewhere on the continuum between these two extremes.

Example of data parallelism in a bottling plant, there are several 'processors', or bottle cappers, applying bottle caps concurrently on rows of bottles.

Example of task parallelism in a restaurant kitchen, there are several chefs, or 'processors', working simultaneously on different parts of different meals. A good restaurant kitchen also demonstrates load balancing and synchronization--more on those topics later.

A common form of parallelism used in developing applications was Master-Worker parallelism where a single processor is responsible for distributing data and collecting results (task parallelism) and all other processors perform same task on their portion of data (data parallelism).

According to Flynn's Taxonomy, the computing systems are classified into the following broad categories:

- SISD: Single Instruction and Single Data
- SIMD: Single Instruction and Multiple Data
- MISD: Multiple Instruction and Single Data
- MIMD: Multiple Instruction and Multiple Data

The purpose of High-performance computing (HPC) platform is to provide the access to the compute resources remotely. The user can login remotely and submit compute their jobs either from the command line or through the GUI based interface provided to them. The computing systems are connected together through a high bandwidth data transfer and made available to the users in a queue-based job submission system. There are many open-source and commercial software packages installed.

At ICAR-IASRI, New Delhi

The National Agricultural Bioinformatics Grid in ICAR consists of an advanced HPC infrastructure at ICAR-IASRI, New Delhi and moderate HPC facilities at the domain centres for undertaking research in the field of agricultural bioinformatics. Clusters are collections of computers that are connected together. The special sets of software are used to configure HPC environment. This set up has been named as Advanced Supercomputing Hub for Omics Knowledge in Agriculture (ASHOKA). The importance of HPC is rapidly growing because more and more scientific and technical problems are being studied on the huge data sets which require very high computational power as well. HPC offers environment for biologists, scientists, analysts, engineers and students to utilize the computing resources in making vital decisions, to speed up research and development, by reducing the execution time.

The following HPC infrastructure are set up under NAIP project NABG which are as follows in the form of clusters, network and storage.

Types of Clusters

a. 256 Nodes Linux Based Cluster with two masters
b. 16 Nodes Windows Based Cluster with one master
c. 16 Nodes GPGPU Based Linux Cluster with one master
d. 16 Nodes Linux based SMP system
e. 16 Nodes Linux Based Cluster at each of the five domains with one master

Types of Networks

a. High bandwidth network with low latency (Q-logic QDR InfiniBand switch)
b. Gigabit network for cluster administration and management
c. ILO3 Management Network

Types of Storage

a. Parallel File System (PFS) for computational purpose
b. Network Attached Storage (NAS) for user Home Directory
c. Archival Storage for back up.

The hardware configuration of the Head/Master node is as follows

Server Name	:	HP ProLiant DL380-G7 Server
Type of Processor	:	Intel Xeon X5675 3.07Ghz
Number of Processors	:	2
Core per Processor	:	6
Total memory (RAM)	:	96GB
Memory per Core	:	8GB
Hard Disk	:	6*600GB SAS
OS	:	RHEL 6.2 (Linux)

The hardware configuration of each compute node is as follows

Server Name	:	HP ProLiant SL390-G7 Server
Type of Processor	:	Intel Xeon X5675 3.07 Ghz
Number of Processors	:	2
Core per Processor	:	6
Total memory (RAM)	:	96G
Memory per Core	:	8GB
Hard Disk	:	300GB SAS
OS	:	RHEL 6.2 (Linux)

Measuring Performance

The memory is measured in terms of bytes i.e., Kilo (2^{10} or 10^3), Mega (2^{20} or 10^6) ,Giga (2^{30} or 10^9) – Tera (2^{40} or 10^{12}), Peta (2^{50} or 10^{15}) ,Exa (2^{60} or 10^{18})

The computational performance is measured in Flop/s (Flop/s = floating point operations per second) i.e., Mega Flops, Tera Flops, Peta Flops etc.

One can calculate peak performance of the cluster using standard formula i.e. Cluster Performance = (Number of nodes) * (number of CPUs per node) * (number of cores per CPU) * (CPU speed in GHz) * (CPU instruction per cycle)

The grid has been established using the following network diagram as in figure 5.

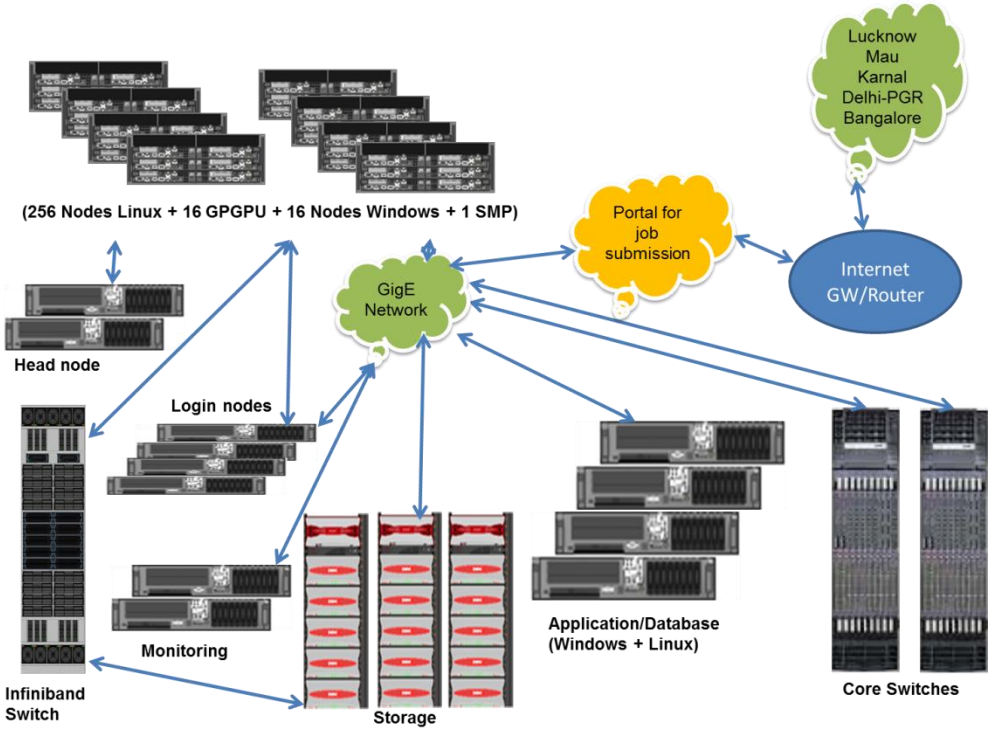


Fig. 5: Network diagram of NABG Grid

The hardware and software specifications of the SMP is as follows

Server Name	:	HP ProLiant DL 980 G7
Type of Processor	:	Intel Xeon E7- 2830 2.13GHz
Number of Processors	:	8
Core per Processor	:	8
Total memory (RAM)	:	1.5 TB
Hard Disk	:	396 GB
OS	:	RHEL 6.2

Aswitched fabric computer network communications link, is being used in HPC and enterprise data centre with InfiniBand interconnect switch. The InfiniBand architecture specification defines a connection between processor nodes and high performance I/O nodes such as storage devices as in figure 6.

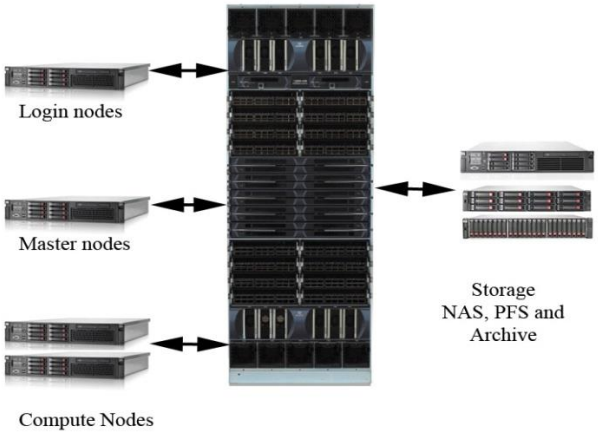


Fig. 6: InfiniBand interconnect switch

Main purpose of Ethernet network in the cluster is to provide services like cluster management, cluster monitoring, compute node deployment and many other things in figure 7.

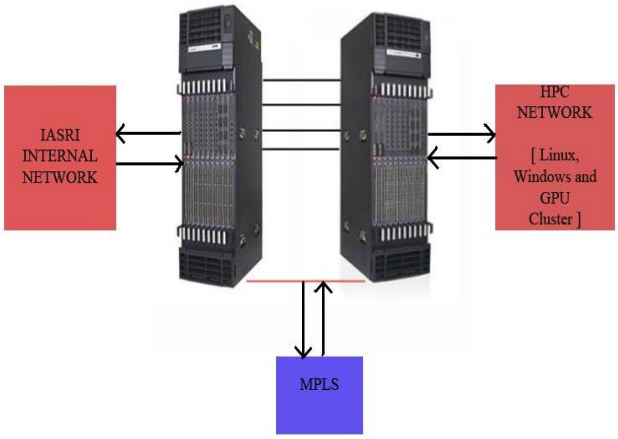


Fig.7: InfiniBand interconnect switch

Different types of file system are configured for storing user's data, running parallel jobs and archiving the important data. There are three types of storage (i) Network Attached Storage (NAS), (ii) Parallel File System (PFS) and (iii) Archival Storage.

The following challenges in bioinformatics are exists which essentially require the grid based architecture.

- Protein folding & structure prediction
- Homology search
- Multiple alignment
- Genomic sequence analysis
- Gene finding
- Gene expression data analysis
- Drug discovery
- Phylogenetic inference
- Computational genomics, proteomics
- Computational evolutionary biology

Genome Wide Association Study and Genomic Selection

Neeraj Budhlakoti, D C Mishra and Anil Rai

ICAR-IASRI, Library Avenue, New Delhi – 110 012

Genome-wide approaches i.e. Genome-Wide Association Study (GWAS) and Genomic Selection (GS) seems to offer promising strategies for genetic improvement of complex traits and to shorten the generation interval (Iwata et al., 2011; Heslot et al., 2015; Grattapaglia et al., 2018). GWAS provides potential capacity to identify the genes that are causally associated with phenotypes of underlying traits, especially traits which are governed by one or a few major genes. However, it has been not very promising in identifying such genes in species where traits are influenced by several minor quantitative trait loci (QTLs) (Kainer et al., 2015; Bhat et al., 2016). GS is fundamentally different from GWAS, as it involves use a full-genome information, regardless of its significance, in relation to a specific trait, rather than a few markers as in GWAS. In a breeding programs, GWAS and GS could substantially reduce the length of breeding cycles and increase genetic gain per unit time through early selection of superior progeny or genotypes especially in case of animals. Therefore, the entire cycles of progeny field testing can potentially be skipped in the selection process

The analysis of QTL effects for minor QTLs using linkage mapping and genome-wide association mapping is often biased. Therefore, scientific groups have been trying for years to solve the issue of how to tackle these complex traits and outcomes in terms of genomic selection (GS). Genomic selection is a breeding approach exploiting high-density DNA markers distributed across the genome to facilitate the rapid selection of the best candidates and offers opportunities to enhance genetic gains (Meuwissen et al., 2001). GS uses different prediction models by combining the genotyping and phenotyping datasets of the training population (TP), which is subsequently used to determine genomic-estimated breeding values (GEBVs) for every genotype of breeding population (BP) from their genotyping scores. These GEBVs permit breeders to envisage superior genotypes that would be suitable either as a parent in hybridization or for next-generation advancement of the breeding program. The basic principle is that the information derived from several markers widely distributed over the genome, having the potential to reveal genetic

variations in the genome, can evaluate breeding values without prior information of where the selected genes are located (Crossa et al., 2017).

Genome Wide Association Study

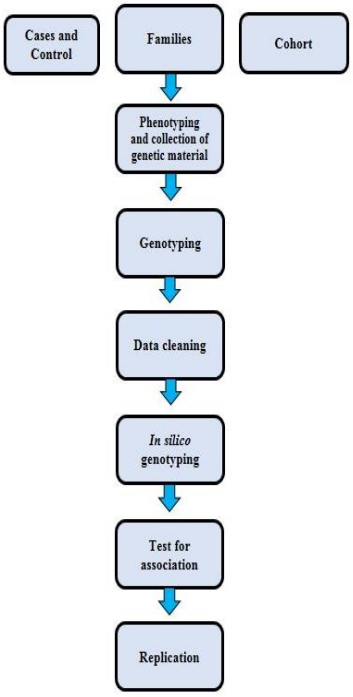
Association mapping is a method of mapping quantitative trait loci (QTLs) that exploits linkage disequilibrium to link phenotypes (observable characteristics) to genotypes (the genetic constitution of organisms), discovering genetic associations. Association mapping is based on the idea that traits that have entered in a population only recently will still be linked to the surrounding genetic sequence of the original evolutionary ancestor. It is generally performed by scanning the entire genome for looking into significant associations between a panel of SNPs and a particular phenotype i.e. linking SNP to a trait. These associations must be then independently validated to check that either they are contributing to the trait of interest directly, or are in linkage disequilibrium with a quantitative trait locus (QTL) that is related to our trait of interest.

The mapping approaches are basically of two Categories namely family mapping and population mapping. A family mapping is constructed by crossing generally two homozygous lines are used for linkage mapping of markers and genes/QTLs. That means it deals with closely related families derived from common parents using a specific mating scheme (Myles *et al*. 2009). In population mapping, also sometime referred as association mapping (AM), the mapping population consists of a diverse set of individuals/lines drawn from natural populations, e.g., random mating populations of wild species; wild relatives of crops like wheat, barley, maize, rice, etc.; as well as breeding populations. Basically there are two approaches for association mapping namely genome-wide and candidate gene approaches. In genome-wide association studies (GWAS), the markers used for genotyping are distributed, preferably evenly and densely, over the whole genome. In this approach, all the loci involved in the control of all the traits showing variation in the sample can be evaluated in one go. In later one candidate gene based approach where problem is restricted to candidate genes/QTLs for the trait(s) of interest. A candidate gene is a gene that is expected, on the basis of past knowledge and experience, to be involved in the control of a trait of interest.

In recent years, genome-wide association studies has been used as successful strategy to identify genetic contributors to complex traits. Their success has meant that genome wide association studies (GWAS) are fast becoming the default study design for discovery of new genetic variants

that influence a clinical trait or phenotype. A genome-wide association study is an approach which involves in rapidly scanning markers across the complete sets of DNA, or genomes, of many people to find genetic variations associated with a particular disease such as cancer. Once new genetic associations are identified, researchers can use the information to develop better strategies to detect, treat and prevent the disease. Such studies are particularly useful in finding genetic variations that contribute to common, complex diseases, such as asthma, cancer, diabetes, heart disease and mental illnesses

Definition: A genome-wide association study is defined as any study of genetic variation across the entire human genome that is designed to identify genetic associations with observable traits (such as blood pressure or weight), or the presence or absence of a disease or condition.



Schematic diagram of GWAS procedure

Selected Key Considerations:

Ascertainment and recruitment

- ✓ Ethical issues
- ✓ Genetic architecture of trait/s of interest
- ✓ Study design

- ✓ Is use of common control data appropriate?
- ✓ Power (sample size)

Phenotyping and collection of genetic material

- ✓ Careful and consistent phenotypic characterization
- ✓ Sample tracking and storage
- ✓ Contingency planning

Genotyping

- ✓ Choice of genotyping platform: commonly
- ✓ Illumina or Affymetrix
- ✓ Choice of genotyping product (SNP density)
- ✓ DNA quantity and preparation

Data cleaning

- ✓ Sample genotyping success rate
- ✓ SNP genotyping success rate
- ✓ Hardy Weinberg proportions
- ✓ Heterozygosity
- ✓ Minor allele frequency

***In silico* genotyping (imputation of untyped SNPs)**

- ✓ Availability of existing LD data for race/ethnic group

Tests of association

- ✓ Choice of appropriate statistical tests
- ✓ Admixture adjustment
- ✓ Multivariate approaches
- ✓ Haplotype analyses
- ✓ Adjustment for multiple tests

Replication

- ✓ Establishing new collaborations or consortia
- ✓ Comparability of study designs
- ✓ Imputation of HapMap SNPs for GWAS conducted on
- ✓ different genotyping platforms
- ✓ Meta-analyses

Populations Used for Association Mapping in Plants

The population used for AM is one of the key factors which decides success of AM. The population may be based on a natural/breeding population or it may be a family-based population. AM can also be performed in biparental and multiparent populations, but single biparental populations are generally not preferred for AM. Generally, doubled haploid, F3, etc., families derived from several biparental crosses generated by mating a group of inbreds in diallel scheme or in a random manner are favored for AM. In case of multiparent populations, two populations, namely, multiparent advanced generation intercrosses (MAGIC) and nested association mapping (NAM) populations, have become very popular because they allow both AM and linkage mapping and can also be further utilized in variety development Programme.

Need of GWAS

- ✓ Current understanding of disease etiology is limited
    - o Therefore, candidate genes or pathways are insufficient
- ✓ Current understanding of functional variants is limited
    - o Therefore, the focusing on non-synonymous changes is not sufficient
- ✓ Results from linkage studies are often inconsistent and broad
    - o Therefore, the utility of identified linkage regions is limited
- ✓ GWA studies offer an effective and objective approach
    - o Better chance to identify disease associated variants
    - o Improve understanding of disease etiology
    - o Improve ability to test gene-gene interaction and predict disease risk

Issues in GWAS

- ✓ Population stratification

- ✓ Multiple Testing: False Positives
- ✓ Gene-Environmental Interaction
- ✓ High Costs

Statistical requirements for a successful GWAS

- ✓ Sufficient sample sizes
- ✓ LD coverage
- ✓ Genotype quality
- ✓ Design of genome-wide association studies
- ✓ Handling of the multiple testing problem

Table: Statistical software packages generally used for association mapping in plants (Zhu et al., 2008; Gupta et al., 2014)

Software	Brief description
Free packages	
TASSEL	LD statistic calculation and graphic visualization; sequence analysis; association mapping using logistic regression, GLM, MLM, and some other models; structure and kinship analyses; analysis of insertion/deletion, diversity estimation, etc. (http://sourceforge.net/projects/tassel; http://www.maizegenetics.net)
EMMAX	Fast computation, for large AM studies, corrects for population structure and kinship (http://genetics.cs.ucla.edu/emmax/)
GenAMap	Implements structured association mapping, employs various algorithms, good graphical presentation (http://sailing.cs.cmu.edu/genamap/)
GenABEL	GWAS for both quantitative and qualitative traits (http://www.genabel.org/packages/GenABEL)
FaST-LMM	AM based on large samples of up to 120,000 individuals (http://fastlmm.codeplex.com/)
GAPIT	Implements CMLM, R-based, fast computation (http://www.maizegenetics.net/gapit)

STRUCTURE	Population structure analysis; generates Q matrix; computation intensive (http://pritch.bsd.uchicago.edu/structure.html)
SPAGeD	Kinship analysis; generates K matrix (http://www.ulb.ac.be/sciences/ecoevol/spagedi.html)
EINGENSTRAT	Association analysis; PCA to generate P matrix to be used in the place of Q matrix (http://genepath.med/harvard.edu/~reich/software.html)
R	Generic package; convenient for simulation work; useful for researchers with good statistics and computer programming background (http://www.r-project.org/)
GenStat	Implements GLM and MLM, corrects for population structure (http://www.vsni.co.uk/software/genstat)
JMP Genomics	Computation of population structure and kinship coefficient (marker-based) (http://www.jmp.com/software/genomics/)
SAS	Standard statistical package used for data analysis and methodology work (http://www.sas.com)

Recent successes using GWAS approaches have generated considerable enthusiasm about the utility of this approach to identify variants that contribute to variation and disease susceptibility. In a relatively short span of time, significant advances have already been made in genotyping efficiency and cost, imputation approaches, and analytical methods. In future, further understanding of the roles of epistasis (gene–gene interactions), gene–environment interactions, copy number variants, and epigenetic phenomena are anticipated to provide additional insights into our understanding of complex traits.

Genomic Selection

Traditional genetic improvement of livestock, using information on phenotypes and pedigrees to predict breeding values, has been found very successful. However, breeding values should be able to predict more accurately by using information on variation in DNA sequence between animals and plants. Research towards marker- assisted selection (MAS) has been extensive but implementation has been limited and increases in genetic gain have been small. Genomic selection has been proposed to address these deficiencies. Genomic selection is a form of marker-assisted

selection in which genetic markers covering the whole genome are used so that all quantitative trait loci (QTL) are in linkage disequilibrium (LD) with at least one marker. Genomic selection predicts the breeding values of lines in a population by analyzing their phenotypes and high-density marker scores. Various methods have been developed for predicting phenotype. Methods are based on analysis of data consist of genotype and phenotype information.

As it is known earlier selection based on phenotypic data has been successfully used in past. As abundance of DNA and marker data, trend slightly shifted to marker assisted selection. MAS is an indirect selection process where a trait of interest is selected, not based on the trait itself, but on a marker linked to it. MAS has been shown to be efficient and effective for traits that are associated with one or a few major genes with large effect but does not perform as well when it is used for selection of polygenic traits (Bernardo 2008).As most economic traits are influenced by many genes, tracking a small number of these through DNA markers will only explain a small proportion of the genetic variance. In addition, individual genes are likely to have small effects and so a large amount of data is needed to accurately estimate their effects. To overcome these difficulties, Meuwissen et al. (2001) proposed a variant of MAS that they called genomic selection. The key features of this method are that markers covering the whole genome are used so that potentially all the genetic variance is explained by the markers and the markers are assumed to be in LD with the QTL so that the number of effects per QTL to be estimated is small.

The major limitation to the implementation of genomic selection has been the large number of markers required and the cost of genotyping these markers are very high. Recently both these limitations have been overcome in most livestock and plant species following the sequencing of the livestock genomes, the subsequent availability of hundreds of thousands of single nucleotide polymorphisms (SNP), and dramatic improvements in development of SNP genotyping technology. Various regression methods have been developed for predicting phenotype. Methods are based on analysis of data consist of genotype and phenotype information. These methods are primarily based on linear models, which are easy to interpret and able to fit to the data without over fitting. However, the relationship between breeding value and genetic markers is likely to be more complex than a simple linear relationship, particularly when large numbers of SNPs are fitted simultaneously in the model. To answer these issues, model-free or so-called nonparametric methods which side-step linearity and require lesser genetic assumptions have gained more

attention (Gianola et al, 2006). Here we will only discuss the parametric genomic selection methods which are suitable for simple or additive kind genetic architecture.

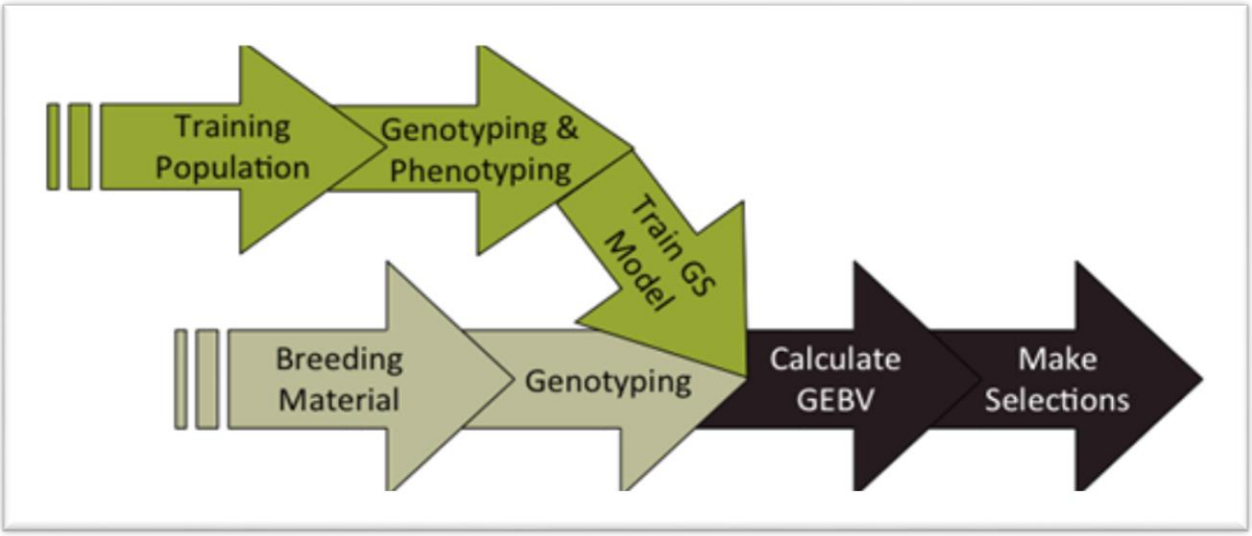


Fig. 1: Diagram Depicting the Basic Workflow of Genomic Selection.

Overview of Most Commonly Used Parametric Methods in Genomic Selection

In GS, the main goal is to predict the individual's breeding value by modeling the relationship between the individual's genotype and phenotype. Simplest form of such a model is:

$$Y_i = \mu + \sum_{j=1}^{p} X_{ij}\beta_j + e_i$$

Where μ is an intercept, X_{ij} is the genotype of the ith individual at the jth marker $j = 1,2,...,p$, and β_j is the corresponding marker effect. This simply takes the form of linear model

$$Y = X\beta + e$$

To estimate $(\hat{\mu}, \hat{\beta})$, we can use least squares to minimize the sum of squared vertical distance between the observed response and the estimated response.

$$(\hat{\mu}, \hat{\beta}) = (Y - X\beta)'(Y - X\beta)$$

$$\hat{\beta} = (\acute{X}X)^{-1}\acute{X}Y$$

Entries in design matrix X depend on the number of different alleles present. For example, individuals with marker genotypes AA, Aa, aa have elements coded as -1, 0, and 1 in X_{ij}, respectively. With advanced genotyping technologies, data generated on markers is huge, and therefore the number of parameters to be estimated, can exceed the number of observations. To confront the problems emerging in these large p with small n regressions $p > n$, estimation procedures performing variable selection, shrinkage of estimates, or a combination of both are commonly used. But it can still perform poorly if the relative ratio of the number of markers and the number of individuals or records is large or has multicollinearity problem, *i.e.*, linkage disequilibrium (LD) exists among the markers. Therefore, a second element of model choice is to utilize type of shrinkage estimation procedure. Here, we will discuss briefly the effects of shrinkage on the statistical properties of estimates and will review some of the most commonly used penalized and variable selection and shrinkage estimation procedures for genomic selection.

Effect of Shrinkage on Mean Squared Error of Estimates

Accuracy of any estimator can be calculated as squared distance between estimated, $\hat{\beta}$, and actual value of parameter β. This we can write as simply the squared deviation: $\left\|\hat{\beta}(Y)-\beta\right\|^2 = \left[\hat{\beta}(Y)-\beta\right]^2$ Then mean-squared error is defined as $MSE\left(\hat{\beta}\right) = E\left[\hat{\beta}(Y)-\beta\right]^2$

Further we can decompose this MSE in two terms i.e. $MSE\left(\hat{\beta}\right) = Var\left(\hat{\beta}\right) + Bias(\beta)^2$

As variance of the estimator decreases with increase in sample size, with standard estimation procedures, such as ordinary least squares (OLS), with sample size remains fixed, the variance of estimates increases rapidly as p increases, yielding high MSE of estimates. One way of solving this problem emerging in large p with small n regressions is by shrinking estimates toward a fixed point (e.g., 0); this may increase bias but reduces the variance of the estimator. To understand the effects of shrinkage on MSE of estimates. Let's have a simple shrinkage estimator obtained by multiplying an unbiased estimator $\hat{\beta}$, to a constant a $\alpha \in [0,1]$ so that $\hat{\beta} = \alpha\hat{\beta} + (1-\alpha)0 = \alpha\hat{\beta}$ the new estimator shrinks the original one towards 0. If $\beta \neq 0$, β is biased, but the variance of the new estimator, $Var\left(\alpha\hat{\beta}\right) = \alpha^2 Var\left(\hat{\beta}\right)$ is definitely to be lower for any $\alpha < 1$. Penalized and Bayesian methods are the two most widely used shrinkage estimation procedures, brief overview of these methods is discussed below.

Penalized methods

In penalized methods, regressions estimates are calculated as the solution to an optimization problem that control model goodness of fit to the training data and model complexity. Model complexity is commonly defined as a function of model unknown's function; $J(\beta)$ therefore, penalized estimates are generally derived as the solution to an optimization problem, generally of the form:

$$(\hat{\mu}, \hat{\beta}) = \left\{ \sum_i \left(Y_i - \mu - \sum_{j=1}^{p} X_{ij} \beta_j \right)^2 + \lambda J(\beta) \right\}$$

Where $\lambda \geq 0$ a regularization parameter that controls the trade is-offs between lack of fit and model complexity. At $\lambda = 0$ above form reduces to ordinary least square. There are several estimation procedures have been already in literature, and they differ only for choice of penalty function $J(\beta)$. In **ridge regression (RR)** (Hoerl and Kennard 1970), the penalty is directly proportional to the sum of squares of the regression coefficients i.e. L2 norm $J(\beta) = \sum_{j=1}^{p} \beta_j^2$. More generalized form known as bridge regression (Frank and Friedman 1993), and it uses penalty function $J(\beta) = \sum_{j=1}^{p} \left\| \beta_j \right\|^{\gamma}$ with $\gamma > 0$. RR is a special case at $\gamma = 2$ which yields L2 norms. Subset selection occurs as a limiting case of $\gamma \to 0$. Which penalizes the number of nonzero effects regardless of their magnitude, $J(\beta) = \sum_{j=1}^{p} 1(\beta_j \neq 0)$. At $\gamma = 1$ penalty function reduces to another special form called **least absolute angle and selection operator (LASSO)** (Tibshirani 1996), $J(\beta) = \sum_{j=1}^{p} \left\| \beta_j \right\|$. This penalty function induces a solution which zeros out some regression coefficients and shrinkage estimates of the other effects; therefore LASSO technique combines variable selection and shrinkage of estimates. These methods are quite useful for many applications. But these methods have several limitations. As in case of whole genome regression (WGR) of complex traits, there is no reason to restrict the number of markers with nonzero effect to be limited by certain number *i.e.* by n which is the number of observations. Second, when predictors are correlated, occurs when LD span over long regions, methods performing variable selection such as the

LASSO are usually outperformed by RR (Hastie et al. 2009). Therefore, in an attempt to combine the good features of RR and of LASSO in a single estimation framework, Zou and Hastie (2005) proposed to use as a penalty a weighted average of the L1 and L2 norm, that is, for $0 \le \alpha \le 1$,

$$J(\beta) = \alpha \sum_{j=1}^{p} \left\| \beta_j \right\| + (1-\alpha) \sum_{j=1}^{p} \left\| \beta_j^2 \right\|.$$

Bayesian Methods: Meuwissen et al. (2001) proposed two hierarchical models for GS based on Bayesian approach Bayes A and Bayes B. In both cases data and the variances of the marker positions need to be modelled. For individual i one can write:

$$Y_i = \mu + \sum_{j=1}^{p} X_{ij} \beta_j + e_i$$

Where $i = 1(1)n$ individual, $i = 1(1)p$ in marker position, Y_i is the phenotypic value for individual i μ is the $n \times 1$ dimensional overall mean vector, X_{ij} is an element of an incidence matrix for marker j and individual i, m_j is a random effect for marker j and e_i is a random residual.

In general the model can be written as:

$$Y = \mu + \sum_{j=1}^{p} Z_j m_j + e$$

Inferences about model parameters are based on the posterior distribution. As it is well known using Bayes Theorem, the posterior can be obtained by combining the prior distribution with likelihood function. The difference between Bayes A and Bayes B is mainly depend how we model the variances of parameters. In both methods each marker position has its own variance. The Bayes A approach applies the same prior distribution of variances for each marker positions. Here we uses scaled inverted chi-squared distribution $\varkappa^{-2}(v, S^2)$ with degrees of freedom v and scale parameter S^2 for prior distribution. This is a convenient choice because it is a conjugate prior so the posterior distribution is in the same family of distributions as the prior distribution. Here posterior distribution is also scaled inverse chi-square distribution $\varkappa^{-2}(v + n_j,\ S^2 + \acute{m}_j m_j)$ where n_j is the number of haplotype effects at marker position j.

The Bayes B approach is generally more realistic in compare to Bayes A for GS. The only difference is of the prior for the variance components. Bayes B assumes that not all markers

contributes to the genetic variation and has a prior density on the variance that is a mixture. It has a high probability mass at $\sigma_{mj} = 0$ and an inverted chi-square distribution when $\sigma_{mj} > 0$. Bayes B can be summarized as $\sigma_{mj} = 0$ with prob = π and $\sigma_{mj} \sim \varkappa^{-2}(v, S)$ with prob = $1 - \pi$.

For the Bayes B method if $\acute{m}m\acute{} > 0$, then one cannot sample $\sigma_{gj}^2 = 0$. so, we can sample m_j and σ_{gj}^2 simultaneously by $p(\sigma_{mj}^2, m_j|y^*) = p(\sigma_{mj}^2|y^*)p(m_j|\sigma_{mj}^2, y^*)$, where y^* is the data that are corrected for the mean and for all genetic effects except m_j.

To sample from the distribution $p(\sigma_{mi}^2|y^*)$ one can use the Metropolis-Hastings algorithm in the following way:

a. Sample $\sigma_{m(new)}^2$ from the prior distribution of $p(\sigma_{mj}^2)$.

b. $\sigma_{mj}^2 = \sigma_{m(new)}^2$ with probability of $Min\left[\frac{p(y^*|\sigma_{m(new)}^2)}{p(y^*|\sigma_{mj}^2)}; 1\right]$.

It can be easily shown using simulated data, that the Bayesian methods perform better than linear least squares regression, ridge regression, and BLUP methods (Meuwissen et al. 2001; Habier et al. 2009, 2010) in terms of prediction accuracy. However, the choice of the degrees of freedom and the scale parameters of the scaled inverse chi-square distribution can affect the outcome (Gianola et al., 2009). Improved Bayesian approach like Bayes C and Bayes D were developed by Habier et al. (2011) to deal with the weakness of earlier used Bayesian approaches. The method Bayes C uses a common variance for all SNPs, and for Bayes D the scale parameter of the scaled inverse chi-square distribution is estimated instead of specified by the user. Bayes Cπ and Bayes Dπ (Habier et al. 2011) are modification of Bayes C and Bayes D where the probability of having a zero effect SNP π is estimated.

Bayesian LASSO: A new method has been introduced the Bayesian LASSO for estimating the regression coefficients (Park and Casella, 2008). This idea is from Tibshirani (1996) to connect the LASSO method with the Bayesian analysis. Tibshirani (1996) noticed that the LASSO estimates of the regression coefficients can be viewed as posterior mode estimates assuming that the regression coefficients have double exponential prior distributions. The Bayesian LASSO is also used in GS (Delos Campos et al. 2009, 2010a; Long et al. 2011) using the hierarchical model with the likelihood function:

$$f(Y|\mu, X, m, \sigma^2) \sim N(\mu + Xm, \sigma^2 I)$$

Where Y is the $n \times 1$ data vector, μ is the overall mean vector, m is a vector of the marker effects, and X is the design matrix that connects m to Y. $N(\mu + Xm, \sigma^2 I)$ Denotes the normal density with mean $\mu + Xm$ and variance $\sigma^2 I$ where I is an $n \times n$ identity matrix. The prior distribution on the marker effects $m_j^{'}s\ j = 1(1)p$ can be written as $p(m_j|\tau_j^2) \sim N(0, \tau_j^2\ )$, and the prior distribution on τ_j is $p(\tau_j|\lambda) \sim Exp(\lambda)$ where $Exp(\lambda)$ denotes the exponential distribution with rate parameter λ.

Best linear unbiased prediction (BLUP): The BLUP theory and the mixed model formulation were first discussed by Henderson (1949), and they were influential for selection purposes in animal breeding (Henderson 1959). BLUP is a statistical procedure, which is useful in situations when the data is unbalanced (for example, in different locations the number of individuals varies), and it can also accommodate family information (Bernardo 2010). Model takes the form:

$$Y = X\beta + Zm + e$$

Where β is a $p \times 1$ vector of unknown fixed effects where usually the first element is the population mean, and X is the incidence matrix that relates Y to β. The above equation is generally called a mixed effect model. The vector β is estimated by the best linear unbiased estimator (BLUE) and BLUP is the predictor of the random effects.

Henderson (1953) proposed that the BLUE and BLUP of (β, m) be obtained by maximizing the joint likelihood of (y, m) given by:

$$L(y, m) = f(y|m)f(m)$$

$$= \frac{1}{2\pi^{n/2}|R|^{1/2}}\left[-\frac{1}{2}(y - X\beta - Zm)'R^{-1}(y - X\beta - Zm)\right] \times \frac{1}{2\pi^{p/2}|G|^{1/2}}\left[-\frac{1}{2}m'G^{-1}m\right]$$

By maximizing the likelihood L(y, m) with respect to b, m and equating it to zero, we obtain a set of linear equations [known as Henderson's mixed model equations (MME)]:

$$\begin{pmatrix} X'R^{-1}X & X'R^{-1}Z \\ Z'R^{-1}X & Z'R^{-1}Z + G^{-1} \end{pmatrix}\begin{pmatrix} \hat{\beta} \\ \hat{m} \end{pmatrix} = \begin{pmatrix} X'R^{-1}y \\ Z'R^{-1}y \end{pmatrix}$$

Where $R = var(e)$ and $G = var(m)$. The solution to the MME is the BLUE of β and the BLUP of m. Henderson's derivation assumes that m and e are normally distributed and maximizes the

joint likelihood of (y, m) over the unknowns β and m. Maximizing the likelihood implies an optimization criterion of $(y - X\beta - Zm)'R^{-1}(y - X\beta - Zm) + m'G^{-1}m$, and it can be viewed as the "ridge regression formulation" of the BLUP (Ruppert et al. 2003).

Widespread use of DNA markers will have a significant impact on breeding programmes. Accurate prediction of GEBVs is a central challenge in animal and plant breeding. Selection will be based on a predication equation which itself based on reference population in compare to traditional phenotypic based selection methods. This will definitely lead to improvement in selection throughput.

Reference

Bernardo, R. (2008). Molecular markers and selection for complex traits in plants: learning from the last 20 years. Crop Science, 48:1649–1664.

Bernardo, R. (2010). Breeding for Quantitative Traits in Plants, Stemma Press, Woodbury, MN.

Bhat, J.A., Ali, S., Salgotra, R.K., Mir, Z.A., Dutta, S., Jadon, V., Tyagi, A., Mushtaq, M., Jain, N., Singh, P.K. (2016). Genomic selection in the era of next generation sequencing for complex traits in plant breeding. Front. Genet, 7: 221.

Crossa, J., Pérez-Rodríguez, P., Cuevas, J., Montesinos-López, O., Jarquín, D., de los Campos, G., et al. (2017). Genomic selection in plant breeding: methods, models, and perspectives. *Trends Plant Sci.* 22: 961–975.

Frank, I. E., and Friedman, J. H. (1993). A statistical view of some chemometrics regression tools. Technometrics, 35: 109–135.

Gianola, D., Fernando, R. L. and Stella, A. (2006). Genomic-assisted prediction of genetic value with semiparametric procedures. Genetics, 173: 1761–1776.

Grattapaglia, D., Silva-Junior, O.B., Resende, R.T., Cappa, E.P., Müller, B.S., Tan, B., Isik, F., Ratcliffe, B., El-Kassaby, Y. (2018). Quantitative genetics and genomics converge to accelerate forest tree breeding. Front. Plant Sci., 9: 1693.

Gupta PK, Kulwal PL and Jaiswal V (2014) Association mapping in crop plants: opportunities and challenges. In: Friedmann T, Dunlap J, Goodwin S (eds) Advances in genetics, vol 85. Academic, Elsevier, Waltham, MA, USA, 109–148.

Hastie, T., Tibshirani, R. and Friedman, J. (2009). The elements of statistical learning, 2nd ed. Springer.

Henderson, C. R. (1949). Estimates of changes in herd environment. J. Dairy Sci. 32: 706.

Henderson, C. R. (1953). Estimation of Variance and Covariance Components. Biometrics 9: 226–252.

Henderson, C. R., Kempthorne, O., Searle, S. R. and von Krosigk, C. M. (1959). The estimation of environmental and genetic trends from records subject to culling. Biometrics, 15: 192.

Heslot, N., Jannink, J.-L., Sorrells, M.E. (2015). Perspectives for genomic selection applications and research in plants. Crop Sci., 55: 1–12.

Hoerl, A. E., and Kennard, R. W. (1970). Ridge regression: biased estimation for non- orthogonal problems. Technometrics, 12: 55–67.

Iwata, H., Hayashi, T., Tsumura, Y. (2011). Prospects for genomic selection in conifer breeding: A simulation study of Cryptomeria japonica. Tree Genet. Genomes, 7: 747–758.

Kainer, D., Lanfear, R., Foley, W.J., Külheim, C. (2015). Genomic approaches to selection in outcrossing perennials: Focus on essential oil crops. Theor. Appl. Genet., 128: 2351–2365.

Long, N., Gianola, D., Rosa, G. J. M. and Weigel, K. A. (2011). Application of support vector regression to genome-assisted prediction of quantitative traits. Theoretical and Applied Genetics, 123: 1065–1074.

Meuwissen, T. H. E., Hayes, B. J. and Goddard, M. E. (2001). Prediction of total genetic value using genome-wide dense marker maps. Genetics, 157: 1819–1829.

Myles S, Peiffer J and Brown PJ (2009) Association mapping: critical considerations shift from genotyping to experimental design. Plant Cell 21: 2194–2202.

Ruppert, D., Wand, M. P. and Carroll, R. J. (2003). Semiparametric Regression. Cambridge Univ. Press, Cambridge, UK.

Sham Pak C (1998) Statistics in Human Genetics Wiley.

Singh BD and Singh AK. Marker-Assisted Plant Breeding: Principles and Practices, DOI 10.1007/978-81-322-2316-0_8.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. Journal of Royal Statistical Society, 58: 267–288.

Zhu C, Gore M, Buckler ES et al (2008) Status and prospects of association mapping in plants. Plant Genome 1: 5–20.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society, 67: 301–320.

COMPUTATIONAL METAGENOMICS

Anu Sharma
ICAR-IASRI, Library Avenue, New Delhi – 110 012

1. Introduction

Metagenomics is the study of microbial communities sampled directly from their natural environment, without prior culturing. By enabling an analysis of populations including many (so-far) unculturable and often unknown microbes, metagenomics is revolutionizing the field of microbiology, and has excited researchers in many disciplines that could benefit from the study of environmental microbes, including those in ecology, environmental sciences, and biomedicine. Specific computational and statistical tools have been developed for metagenomic data analysis and comparison. In sequencing studies, unlike traditional microbial genomic sequencing projects, metagenomics research attempts to determine directly the whole collection of genes within an environmental sample (i.e., the metagenome), and analyze their biochemical activities and complex interactions.

Metagenomics, in principle, enables the study of any microbial organism, including the large number of microorganisms (more than 99%) that cannot be isolated or are difficult to grow in lab. More importantly, microbes by nature live in communities, where they interact with each other by exchanging nutrients, metabolites and signaling molecules. Although the conventional pure-culture paradigm remains important for complete characterization of a species, its traditional exclusive usage limits exploring the microbial world. Traditional clonal culture microbiology needs to be complemented by culture-independent microbiology that can directly characterize microbes in natural environments, and can address important biological questions related to those microbial environments, such as the diversity of microbes in different environments, microbial (and microbe-host) interactions, and the environmental and evolutionary processes.

Various computational techniques are applied to explore the relationship between microbes and their habitat to make inferences about the community:

- Marker genes
- Metagenomes
- Metatranscriptomes
- Metaproteomes
- Metametabolomes

Computational analysis has shown an even greater impact on metagenomic studies compared to traditional genomic projects, due not only to the large amount of metagenomic data, but also to the new complexity introduced by metagenomic projects (e.g., assembly of multiple genomes simultaneously is more challenging than the assembly of single genomes), and the new questions we are asking (e.g., host-microbe interaction).

2. Computational and statistical tools for metagenomic studies

2.1 Assembly Metagenome Data

Once the sequences are collected, the first step in data analysis is to reconstruct the entire microbial genomes from metagenomic sequence reads using fragment assemblers. Assembly of metagenome consists of three main steps namely (i) sequence cleaning, (ii) fragment assembly and (iii) Binning/ taxonomic assignment (Figure-1). Sequence cleaning deals with

the filtering of duplicated, short, low quality, contaminated reads and reads containing ambiguous bases. The Amplicon based sequencing methods also requires trimming of primer or tag sequences. String matching algorithms are employed to identify these sequencing errors. Fragment assembly of metagenomic sequences is performed following two approaches namely OLC and DBG. OLC based methods are found to be useful for long sequencing reads, while, DBG based methods is employed in case of short sequencing reads. Although, OLC based methods performs well with small number of reads having significant overlap but their performance degrades as genome size increases. Binning/ taxonomic assignment is needed to classify/associate these data with the organisms from which they were originated. This process of association between sequence data to their related species is called binning or classification. Existing binning methods can be classified into two categories, namely taxonomy dependent and taxonomy independent. In case of taxonomy dependent methods, the extent of 'similarity' of the reads with sequences (in reference databases) or pre-computed models (built using sequences in reference databases) helps in this assignment process. Further, based on the strategy used for comparing sequence reads with existing sequences/pre-computed models, taxonomy-dependent methods can be further classified into three categories i.e. alignment-based, composition-based and hybrid methods (Mande et al., 2012). Alignment based algorithms uses existing alignment and read mapping algorithms to align the reads to known and characterized genomes. This approach is used in MG-RAST, CAMERA, MEGAN, MetaPhyler and MARTA etc. Faster methods based on composition features uses GC content, codon usage, oligonucleotide usage patterns for comparing reads to reference databases. Under this method, specific models are built for various genomes using various statistical and data mining approaches. Some binning tools based on this method are Phylophthia, TACOA, Phymm, ClaMS and RAIphy etc. Hybrid approaches combines the alignment and composition based methods for classification of metagenomic data. In case of taxonomy independent methods, simply group/bin of sequence reads are made from given dataset based on their mutual similarity and it does not involve any database comparison. Tools following this approach are TETRA (Teeling et al., 2004), CompostBin (Chatterji et al., 2008), AbundanceBin (Wu et al., 2011) and MetaCluster (Leung et al., 2011).

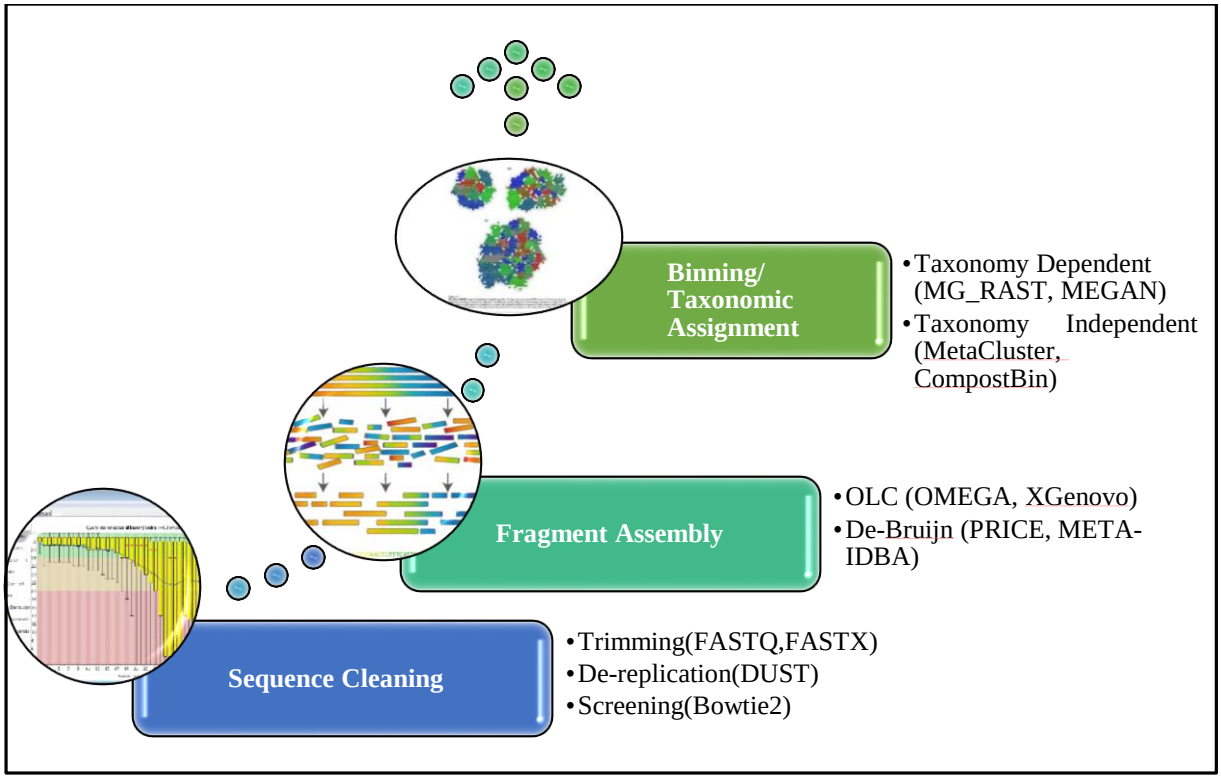


Figure-1. Metagenome Assembly Process

The various computational features of metagenomic assemblers are provided in Table 1. Also, the chronological development of metagenomic assemblers are shown in Figure-2.

Table 4: Computational Features of Metagenome Assemblers (Sharma et al. , 2020)

Name of Assembler	**Year**	**Data Structure**	**Novelty in approach**	**Parallelism**	**Limitation**	**Language Developed**	**Read Size/platform**
Genovo.	2010	OLC	Extracts more information from datasets and thus is suitable for low abundance sequences.	Single Server	Assessment with Illumima dataset	C++	454 sequencing
Meta-IDBA	2011	DBG	-Handled the polymorphism in similar species. -Change in abundance ratio does not affect the performance	Single Server	-Quality of assembly depends on the quality of DBG. -Many species leads to many false positive edges which	C++	NGS

					leads to many small components.		
MetaVelvet	2012	DBG	Decomposition of DGB graph constructed from the mixed short reads into sub-graphs	Single Server	Accurate identification of chimeric reads	C++	Short Read
PRICE	2013	DBG	Contig length extension using paired read	Multi-threading on multicore single CPU.	-	C++	Short Read Shotgun metagenomics data
XGenovo	2013	OLC	-Additional parameter for solving chimera contig case -Use of modified sampling process for the location of read in a contig.	Single Server	-Implementation of probabilistic model with DBG graph -high computational cost	C++	454 paired end reads
Omega	2014	OLC	Uses less computing time as compared to existing OLC based assemblers at higher coverage	Multi-threaded and multiprocess distributed memory	Less Efficient as compared to DBG based assemblers	C++	Illumina Long Reads
MetaVelvet-SL	2015	DBG	Better detection of chimeric nodes by supervised learning.	Single Server	Classification model for chimeric reads may be improved.	C++	Short Read
MEGAHIT	2015	Succinct DBG	-No need for normalization and partitioning -Efficient dynamic removal of edges	GPU based and 3-5 times fast as compared to its CPU counterpart.	Parallelization is restricted by small size of on-board GPU memory.	C++	NGS

MEGAHIT v1.0	2016	Refined Succinct DBG	-Long bubble merging -Faster and uses less memory than MEGAHIT	Runs on Single Sever	-	C++	NGS
metaSPAdes	2017	DBG	Detection and removal of large bulges and longer tips	Single Server	Incorporation of recent high quality jumping mate-pair libraries	C++	Single cell and TSLRs

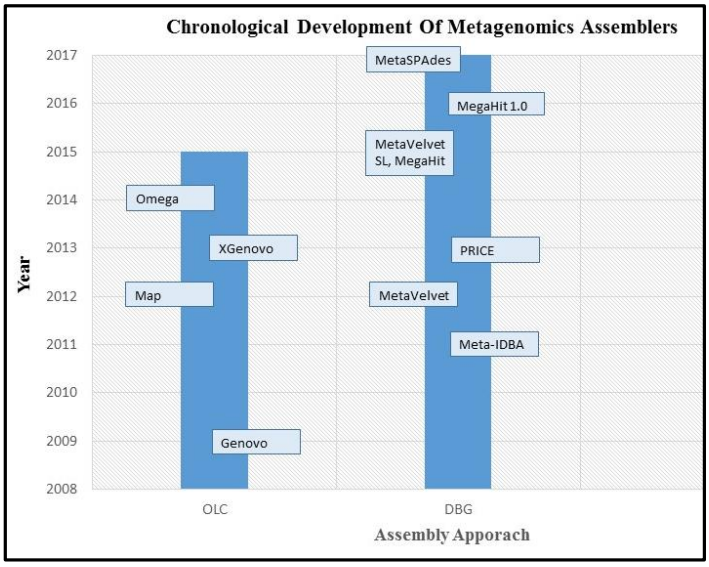


Figure-2. Chronological Development of Metagenome Assemblers

2.2. Function Prediction

Function prediction on metagenomic data is generally more difficult than that for genomic studies, because short reads are more difficult to annotate, and some information (such as the gene neighbourhood in a genome) that is very important for function prediction in sequence data from individual genomes may not be so useful in working with metagenomics data. Existing functional annotation pipelines (developed for annotating genomic sequences derived from traditional genome projects) have been extensively used for functional annotation of metagenomics sequences. An often used practice is to predict COG families, KEGG families (Kanehisa et al., 2008), FIG families (Overbeek et al., 2005), and other functional categories for metagenomic short reads based on BLAST search results. Currently, BLAST is the only readily available or established similarity tool that can handle comprehensive analyses of a metagenomic dataset (with the support of a big computer cluster with hundreds or thousands of CPUs). Very often, an E-value cutoff (e.g., 1e-5) has been adopted for annotating if a read has a certain function (Dinsdale et al., 2008). Biological pathways (or subsystems) can also be reconstructed for metagenome building from protein family predictions. (In most studies, the information obtained is on pooled biological pathways from various species in the same

microbial community). MG-RAST is an automatic server for subsystem annotation for metagenomic datasets, based on an extension of the very successful microbial genome annotation server RAST (Meyer et al., 2008). Pooled biological pathways provide key insights into the functional/metabolic ability of a microbial community (e.g., the functional profiling by KEGG pathways in (Tringe et al., 2005), although in most metagenomic studies, many details, such as the contribution of individual species to the total metabolic ability of a microbial community, are not well defined.

In addition to mapping reads onto known protein families, discovering how many protein families are encoded in a metagenome (including known and unknown families) is important for understanding the functionality and functional diversity of a community and is also important for expanding the protein (family) universe. Li et al. 2008 developed an approach for rapid analysis of the sequence diversity for very large metagenomics datasets using a clustering approach powered by CD-HIT algorithm (Li et al., 2001). Similarity search (by BLAST or by a more sensitive similarity search tool) helps predict the function of some of these protein families; yet, the functions of numerous small, rare protein families remain to be discovered.

3. Conclusion

The growth of public DNA sequence data over the last two decades has been exponential, with a doubling time of about 14 months; the doubling time appears likely to be substantially shortened due to the addition of metagenomic data, as well as the impact from NGS approaches. The massive metagenomic data poses great challenges in many areas involving data management and data mining. More developments of efficient and powerful computational tools that can deal with (or, even better, take advantage of) the huge amount of metagenomic sequences. Besides, these tools need to be able to deal with the high species complexity of the metagenomic datasets. Further, integrating metaproteomic, metatranscriptomic and metagenomics data sets to gain a more comprehensive picture of the organization and function of microbial communities is also desired.

References

Chatterji S, Yamazaki I, Bai Z et al., 2008. CompostBin: a DNA composition-based algorithm for binning environmental shotgun reads. *Res in Comp. Mol. Biol. (LNCS)* **4955**:17-28.

Kanehisa M, Goto S. KEGG: kyoto encyclopedia of genes and genomes. *Nucleic Acids Res.* 2000; 28:27–30.

Leung H C M, Yiu S M and Yang B. et al. 2011. A robust and accurate binning algorithm for metagenomic sequences with arbitrary species abundance ratio. *Bioinformatics* **27(11)**:1489-95.

Li W, Wooley JC, Godzik A. Probing metagenomics by rapid cluster analysis of very large datasets. *PLoS One.* 2008; 3:e3375.

Li W, Jaroszewski L, Godzik A. Clustering of highly homologous sequences to reduce the size of large protein databases. *Bioinformatics.* 2001; 17:282–283.

Mande S S, Mohammed M H and Ghosh T S. 2012. Classification of metagenomic sequences: methods and challenges. *Briefing in Bioinformatics* **13(6)**:669-681.

Overbeek R, Begley T, Butler RM, Choudhuri JV, Chuang HY, Cohoon M, de Crecy-Lagard V, Diaz N, Disz T, Edwards R, et al. The subsystems approach to genome annotation and its use in the project to annotate 1000 genomes. *Nucleic Acids Res.* 2005; 33:5691–5702.

Teeling H, Waldmann J, Lombardot T et al., 2004. TETRA: a web-service and a stand-alone program for the analysis and comparison of tetranucleotide usage patterns in DNA sequences. *BMC Bioinformatics* **5**:163.

Tringe S, von Mering C, Kobayashi A, Salamov A, Chen K, Chang H, Podar M, Short J, Mathur E, Detter J, et al. Comparative metagenomics of microbial communities. *Science.* 2005; 308:554–557.

Wu Y W and Ye Y. 2011. A novel abundance-based algorithm for binning metagenomic sequences using l tuples. *J Comput Biol.* **18(3)**:523-34.

Dinsdale EA, Edwards RA, Hall D, Angly F, Breitbart M, Brulc JM, Furlan M, Desnues C, Haynes M, Li L, et al. Functional metagenomic profiling of nine biomes. *Nature.* 2008;452:629–632. [PubMed] [Google Scholar]

Meyer F, Paarmann D, D'Souza M, Olson R, Glass EM, Kubal M, Paczian T, Rodriguez A, Stevens R, Wilke A, et al. The metagenomics RAST server - a public resource for the automatic phylogenetic and functional analysis of metagenomes. *BMC Bioinformatics.* 2008;9:386.

Sharma, A., Mishra, D. C., Budhlakoti, N. & Rai, A. & Lal, S. B. & Kumar, S. (2020). Algorithmic and computational comparison of metagenome assemblers. Indian Journal of Agricultural Sciences. 90. 847-54.

Introduction to Proteomics

Sudhir Srivastava, Dwijesh Chandra Mishra and K. K. Chaturvedi
ICAR-IASRI, Library Avenue, New Delhi – 110 012

Proteomics

Proteins are important large biomolecules or macromolecules performing a wide variety of functions. The word "proteome" is defined as the entire set of proteins translated and/ or modified within a living organism. The word "proteome" was coined by Marc Wilkins in 1994 in a symposium on "2D Electrophoresis: from protein maps to genomes" held in Siena in Italy while he was a Ph.D. student at Macquarie University. An organism's genome is more or less constant whereas proteome is not constant. Proteomes differs from cell to cell and from time to time. That's why proteomics is more complicated when compared to genomics.

Proteomics more generally refers to large-scale liquid chromatography (LC) coupled with mass spectrometry (MS) [LC-MS] based discovery studies designed to address both quantitative and qualitative aspects of the proteome research (Figure 1).

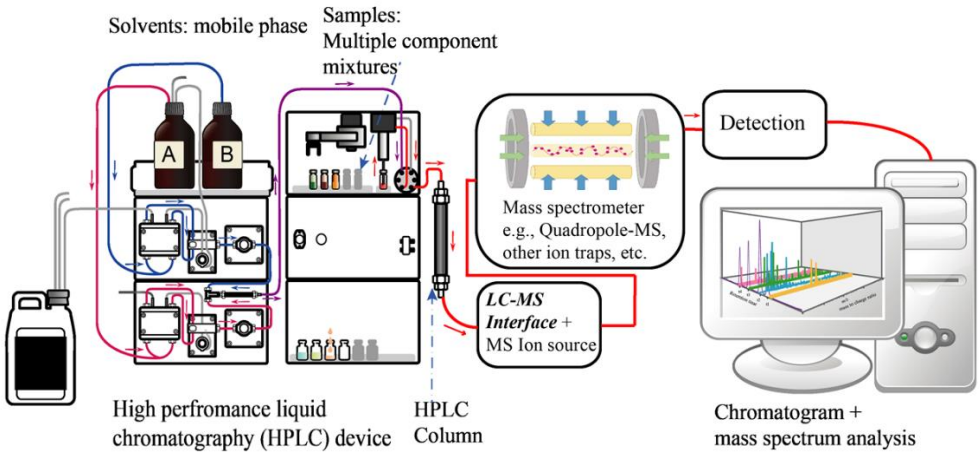


Figure 1. Liquid chromatography coupled with mass spectrometry [LC-MS]

Source: https://upload.wikimedia.org/wikipedia/en/f/f9/Liquid_chromatography_tandem_Mass_spectrometry_diagram.png

Now proteomics has emerged as a powerful tool across various fields such as biomedicine mainly applied to diseases, agriculture, and animal sciences. It is important for studying different aspects of plant functions such as identification of candidate proteins involved in the defensive response of plants to biotic and abiotic stresses, effect of global climate changes on crop production, etc. In animal sciences, proteomics studies play important role in studying physiology, immunology, reproduction and lactational biology. The practical application of proteomics includes expression proteomics, structural proteomics, biomarker discovery, interaction proteomics, protein networks, etc.

The proteomic abundance (expression) data are usually generated using high throughput technologies usually involving MS. LC-MS is used in proteomics as a method for identification and quantification of peptides and proteins in complex mixtures. There are two basic proteomics approaches, namely bottom-up and top-down. The most common proteomics approach is the bottom-up in which proteins in a sample are enzymatically digested into peptides and subjected to chromatographic separation, ionization and mass analysis. Conversely, top-down proteomics addresses the study of intact proteins and consequently is most often used to address purified or partially purified proteins, Further, the quantification of features (peptides or proteins) may be either label-free or labelled (metabolic, enzymatic, or chemical) to detect differences in feature abundances among different conditions. In label-free quantification, MS ion intensity (peak area) and spectral counting of features are the major approaches.

Problem of missing values and heterogeneity in proteomics data

Various approaches exist for proteomics data analysis in which the first step is to summarize the intensities of all features using a quantitative summary followed by logarithmic transformation to approximate it to normal distribution. In spite of availability of various tools/methods, there are various challenges in analyzing proteomics data such as missing value (MV) and data heterogeneity. There are various drawbacks of the methods which can be studied by examining the statistical properties of these methods.

The variations in the biological data or technical approaches to data collection lead to heterogeneity for the samples under study. The data set usually consists of biological replicates only or both biological and technical replicates. Biological variability arises from genetic and environmental factors and it is intrinsic to all organisms. The technical approaches include sample collection and storage, sample preparation, extraction, LC separation and MS detection.

The data set is called balanced when it contains an equal number of subjects/ samples in each group, and the features have no missing observations. However, this is not always the condition. Sometimes the data can be unbalanced having unequal number of subjects, or missing observations, or both. MVs in proteomics data can occur due to biological and/or technical issues. These are of three types of MVs: (i) missing completely at random (MCAR) in which MVs are independent of both unobserved and observed data; (ii) missing at random (MAR) if conditional on the observed data, the MVs are independent of the missing measurements; and (iii) missing not at random (MNAR) when data is neither MCAR nor MAR. The data with missing observations can be analyzed either by excluding the features having missing observations, by using statistical methods that can handle unbalanced data, or by using imputation methods. If the features having missing observations are excluded, then there is loss of information from the experiment. Therefore, the use of methods that can handle MVs, such as imputation methods, are generally

preferred. However, the use of imputation methods may lead to wrong interpretation and these methods are questionable in statistical terms.

Differential abundance/ expression data analysis

Differential abundance analysis is carried out to detect significant features in two or more conditions such as normal versus different disease conditions. However, data normalization is necessary before performing further analysis. There are various transformation and/ or normalization methods such as logarithmic transformation, quantile normalization, variance stabilizing normalization, median scaling normalization, etc. In case of missing values, the user has to impute the data using imputation techniques such as singular value decomposition, *k*-nearest neighbor, maximum likelihood estimation, etc. The statistical approaches/ tests such as t-test, moderated t-test, ANOVA, linear mixed model, etc. can be used for detecting significant features. There are various tools and packages available for proteomics abundance data analysis such as "RepExplore", "MSstats", "PANDA", "MSqRob", etc.

In this article, we are dealing with the bottom-up approach in which peak area values have been used in label-free quantification of proteins. An example of proteomics abundance data analysis using "RepExplore" has been illustrated below. A portion of an example dataset for a case-control study is shown in Figure 2. The dataset has two biological replicates each having two technical replicates in each group (case and control).

	Control				Case			
	control_1_1	control_1_2	control_2_1	control_2_2	case_1_1	case_1_2	case_2_1	case_2_2
biomolecule_1	20.84	19.93	20.78	19.24	20.03	20.87	19.65	20.07
biomolecule_2	19.18	18.79	18.88	18.43	18.97	18.88	18.82	18.64
biomolecule_3	19.5	18.84	20.14	19.06	19.58	19.29	19.1	19.31
biomolecule_4	19.23	18.52	19.67	17.73	19	18.6	16.4	18.44
biomolecule_5	19.64	19.25	19.99	18.78	19.5	19.31	19.16	19.41
biomolecule_6	19.89	19.45	19.93	18.8	19.46	18.76	18.84	18.94
biomolecule_7	22.07	21.72	23.26	21.35	22.74	21.65	20.97	22.17
biomolecule_8	21.84	21.47	22.81	21.22	22.35	21.58	21.18	22.01
biomolecule_9	17.56	17.41	17.46	17.7	16.76	18.13	18.51	17.3
biomolecule_10	20.34	19.81	21.02	19.23	20.38	19.6	19.06	19.8
biomolecule_11	19.15	18.79	17.98	19.03	17.81	19.55	19.89	18.76
biomolecule_12	24.64	24.12	23.21	24.38	23.31	24.77	25.04	24.21
biomolecule_13	26.51	26.06	26.74	25.23	26.32	25.67	25.15	25.95
biomolecule_14	25	24.42	23.27	24.79	23.48	25.16	25.45	24.58
biomolecule_15	18.05	18.3	18.51	17.98	18.52	17.4	18.36	16.85
biomolecule_16	17.82	17.34	18.24	17.07	17.8	17.66	17.45	17.65
biomolecule_17	17.98	17.31	18.28	17.27	17.77	17.37	17.47	17.31
biomolecule_18	19.32	18.13	19.33	18.04	18.81	18.64	18.66	17.88
biomolecule_19	24.89	24.43	24.82	23.74	24.4	24.24	24.1	24.35
biomolecule_20	17.94	17.25	18.39	17.19	17.19	17.08	16.88	16.82

Figure 2. A portion of test dataset for a case-control study

The user has to upload the abundance data as given in Figure 3. The user has to choose various options after uploading the data (Figure 4). Then, the user has to click on "Run Analysis!" button.

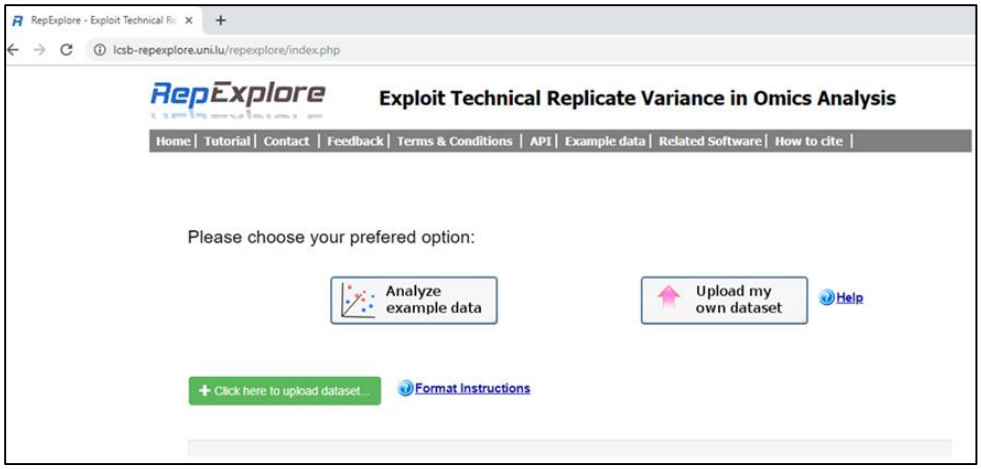


Figure 3. Upload the data

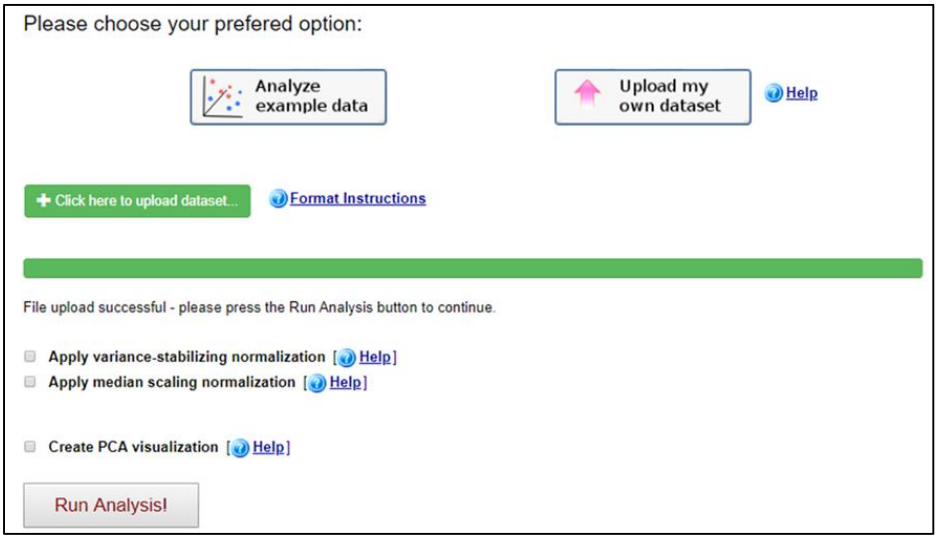


Figure 4. Selecting the options

Then, the user will get menus of various results as shown below in Figure 5.

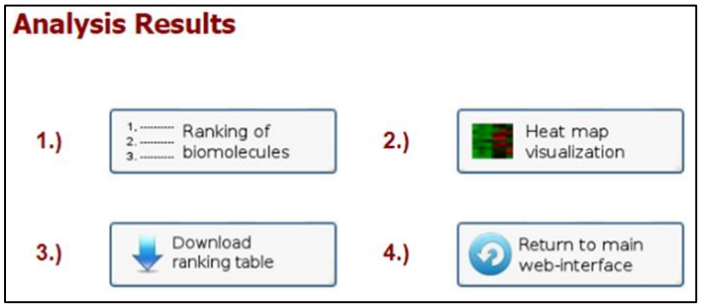


Figure 5. The menus of various results obtained

The ranking table of differential abundant/ expressed features is given in Figure 6.

Biomolecule identifier	Log. fold change	Probability of positive likelihood ratio (PPLR)	P-like significance score (min(PPLR,1-PPLR))	eBayes T-score	eBayes P-value	eBayes adj. P-value
biomolecule_90 generate bar plot	1.66	0.287	0.287	3.71	0.0116	0.542
biomolecule_36 generate bar plot	-0.8	0.633	0.367	-3.27	0.0193	0.542
biomolecule_20 generate bar plot	-0.7	0.762	0.238	-2.88	0.0309	0.542
biomolecule_93 generate bar plot	0.858	0.311	0.311	2.85	0.032	0.542
biomolecule_100 generate bar plot	0.93	0.378	0.378	2.85	0.0322	0.542
biomolecule_40 generate bar plot	0.843	0.236	0.236	2.79	0.0345	0.542
biomolecule_94 generate bar plot	-0.943	0.672	0.328	-2.72	0.0379	0.542
biomolecule_73 generate bar plot	-0.517	0.69	0.31	-2.22	0.0718	0.586

Figure 6. The ranking table of differential abundant/expressed features

The user can generate the bar plot of any feature by clicking "generate_bar_plot" button for which an example is shown below.

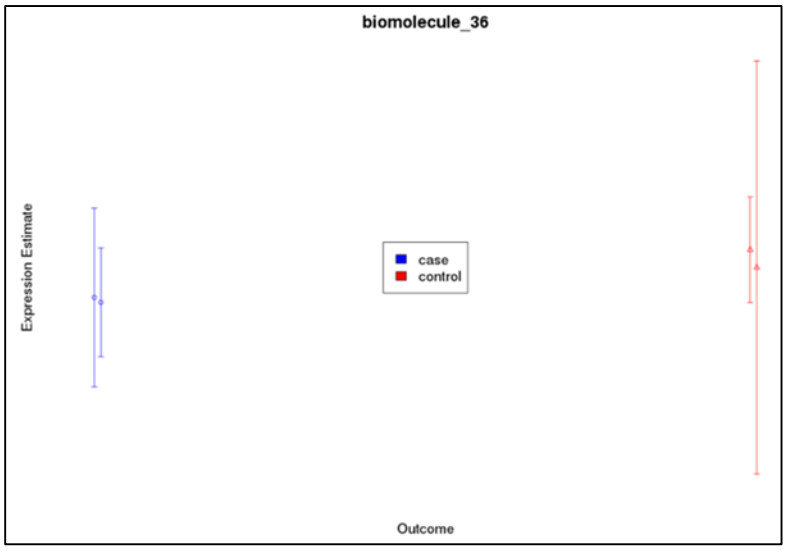


Figure 7. An example of bar plot of a feature

Conclusion

In this article, we have given the basic introduction of proteomics, problem of MVs and heterogeneity in proteomics data and different methods for analysis of proteomics data. Further, proteomics abundance data analysis has been demonstrated by using an online web tool "RepExplore". This article will be useful for the researchers working in the field of proteomics and bioinformatics. Furthermore, the methods for proteomics data analysis can further be used for analyzing the expression data obtained from similar experiments (e.g., microarray and metabolomics data).

References

Anderson NL, Anderson NG (1998). Proteome and proteomics: new technologies, new concepts, and new words. *Electrophoresis*, **19(11)**, 1853-61.

Ceciliani F, Eckersall D, Burchmore R, Lecchi C. (2014). Proteomics in veterinary medicine: applications and trends in disease pathogenesis and diagnostics. *Vet Pathol*., **51(2)**:351-62. doi: 10.1177/0300985813502819.

Chang C, et al. (2018). PANDA-view: An easy-to-use tool for statistical analysis and visualization of quantitative proteomics data. *Bioinformatics*.

Choi M, et al. (2014). MSstats: an R package for statistical analysis of quantitative mass spectrometry-based proteomic experiments. *Bioinformatics*, **30(17)**. 2524-6.

Glaab E, Schneider R (2015). RepExplore: addressing technical replicate variance in proteomics and metabolomics data analysis. *Bioinformatics*, **31(13)**, 2235-7.

Goeminne LJE, Gevaert K, and Clement L (2018). Experimental design and data-analysis in label-free quantitative LC/MS proteomics: A tutorial with MSqRob. *J Proteomics*, **171**, 23-36.

Karpievitch YV, Dabney AR, and Smith RD (2012). Normalization and missing value imputation for label-free LC-MS analysis. *BMC Bioinformatics*, **13 Suppl 16**, S5.

Rubin DB (1976). Inference and missing data. *Biometrika*, **63(3)**, 581–92.

Wasinger, VC, Cordwell, SJ, Cerpa-Poljak, A, Yan, JX, Gooley, AA, Wilkins, MR, Duncan, MW, Harris, R, Williams, KL, Humphery-Smith, I (1995). Progress with gene-product mapping of the Mollicutes: *Mycoplasma genitalium*. *Electrophoresis*, **16 (1)**, 1090-1094. doi:10.1002/elps.11501601185

https://en.wikipedia.org/wiki/Proteomics

Deep Learning: An Introduction

Kanchan Sinha
ICAR-IASRI, Library Avenue, New Delhi – 110 012
kanchan.sinha@icar.gov.in

Abstract

In recent years, a tremendous interest has emerged in the area of deep learning approach which has gained considerable prominence fuelled by numerous applications in the field of image and speech recognition, automatic translation, genomics, agriculture, time series modelling and many more areas. Deep learning can be considered as a subset of machine learning computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction. Deep learning discovers intricate structure in large data sets by following the principle of backpropagation algorithm to interpret how a machine should change its internal parameters that are used to compute the representation in each layer from the representation in the previous layer. Convolutional neural network is a deep neural network model which has focused attention in images, video recognition in recent years. Recurrent neural networks and its variants Long-Short-Term-Memory (LSTM) networks are also shown their effectiveness in modelling sequential data such as text, speech and machine translation, etc.

Keywords*: artificial, intelligence, machine, learning, deep, neural, network, recurrent, convolution, image, sequence, time, series*

1. Introduction

Deep learning has sparked a lot of interest in recent years. Deep Learning is a subset of machine learning algorithm, which has been introduced with the objective of moving machine learning closer to one of its original goals: artificial intelligence. The term artificial intelligence (AI) was first coined in the year 1956 which makes the machines to perform cognitive tasks like thinking, perceiving, learning, problem solving and decision making and is achieved by mimicking human brain, by understanding how it thinks, how it learns, decides, and work while trying to solve a problem.

Artificial intelligence is successfully applied in many fields such as self-driving car, speech recognition, understanding natural language, image recognition and in many other real-life situations. In recent years with the help of easier data collection methods and storage facility for larger datasets, specialized hardware devices like graphics processing unit (GPU), improved

software, artificial intelligence has become an indispensable part of computing for every discipline.

In the modern era of computing, machine learning (ML) techniques which can be viewed as a subset of AI, empowers the ability to learn objects, translate speech to text, environmental prediction and etc., without being explicitly programmed. Deep learning (DL) models which are subset of machine learning techniques now increasingly replacing machine learning models in several real-life application viz., image or video recognition, sequence data modelling, environmental modelling, genomics and so on. The problems of ML models due to their suffering from the curse of dimensionality for solving crucial AI problems viz., natural language processing, image recognition, handwriting recognition, etc. make them vulnerable as compared to the DL models.

The term "deep learning" was introduced by Geoffrey Hinton in the year 2006. Deep learning (DL) is machine learning algorithms based on learning multiple levels (i.e., deep) of representation inspired by the structure and function of the brain called artificial neural networks. The easiest way to think of the relationship among the AI, ML and DL can be visualized from the following figure 1 as

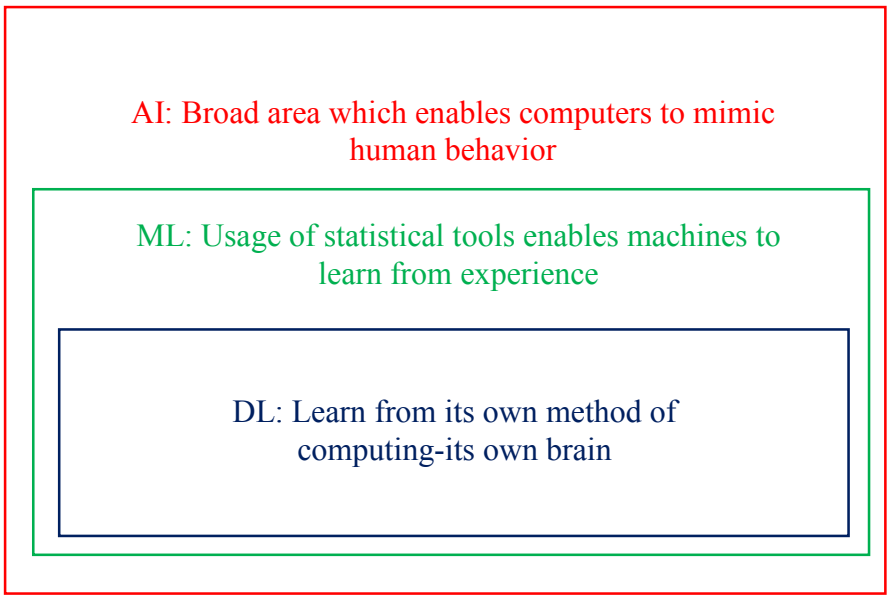


Fig. 1: DL is a subset of ML & ML is a subset of AI

2. Machine Learning Technology

ML involves a learning process with the objective to learn from the "experience" (training

data) to perform a specific task. Data in ML consists of a set of examples. Usually, an individual example is described by a set of attributes, also known as features or variables. A feature can be nominal, binary, ordinal, or numeric, etc. The performance of the ML model for a specific task is measured by a performance metric that is improved with the experience over time. The performance of ML models and algorithms, for the designed task which is achieved by learning the features can be tested by statistical models. When the learning process is over, the trained model can be used to classify, predict, or cluster new examples (testing data). ML algorithms are typically classified into three different broad categories depending on the learning type as supervised, unsupervised and reinforcement learning.

In supervised learning every input pattern that is used to train the network is associated with an output pattern, which is the target or the desired pattern i.e., with the available input variables (x) and an output variable (y), an algorithm is used to learn the mapping from the input to the output. In unsupervised Learning the target output is not presented to the network i.e., to train a model using information that is neither classified nor labelled and hence can be used to cluster the input data in classes on the basis of statistical properties whereas a reinforcement learning algorithm is achieved by interacting with space or an environment.

3. **How Deep Learning Models Work?**

The deep learning models are inspired by the working principle of a basic unit of the human brain cell or a neuron. Inspired from a neuron, an artificial neuron or a perceptron was developed. Artificial neural networks (ANNs) are computational methods that mimic the behaviour of the human brain's central nervous system (biological neural networks) that are made from layers of connected units called artificial neurons. The basic building block of a neural networks model is a single neuron which is also called as perceptron as depicted in fig.

2

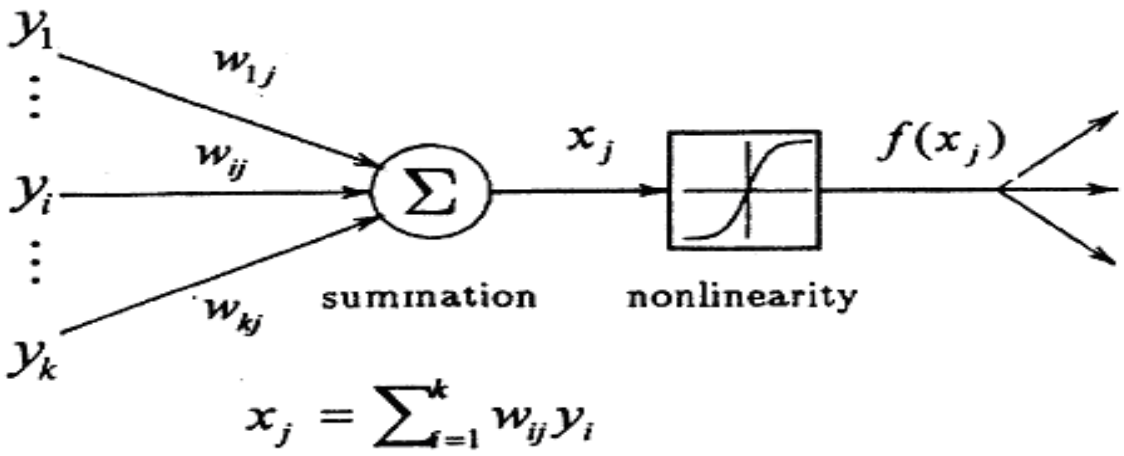


Fig. 2: A Single Neuron (Perceptron)

Neural network is a class of non-linear, nonparametric, data driven approach can be viewed as a powerful learning models that achieve state-of-the-art results in a wide range of supervised and unsupervised machine learning tasks. A general neural networks architecture consist of an input layer that accepts external information, one or more hidden or middle layer that provide non-linearity to the model and an output layer that provides the target value. Each layer contains one or more nodes. This model is characterized by a network of three layers of simple processing units, and thus termed as multilayer perceptron. All the layers in a multi-layer neural network are connected through an acyclic arc. The schematic representation of a neural networks model can be viewed in fig. 3 as

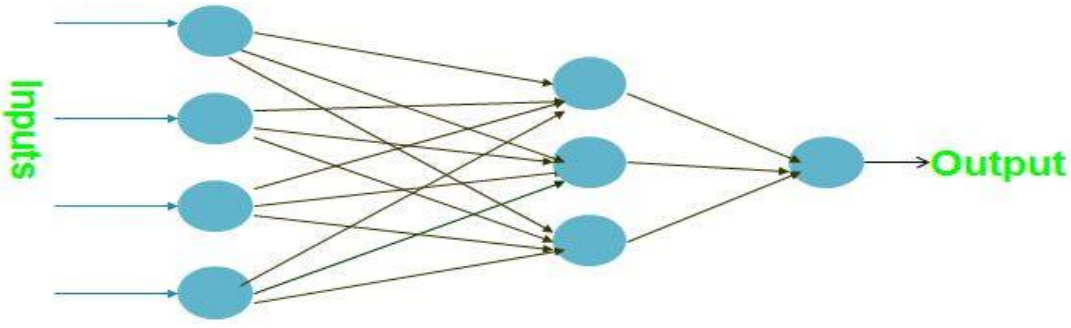


Fig. 3: Schematic representation of neural network

In this network, the information moves only in one direction i.e., from the input nodes, through the hidden nodes and to the output nodes. A neuron can be considered as a real characteristic of the input vector ($y_1, y_2, \dots, y_k$). The output is obtained as

$$f(x_j) = f(\alpha_j + \sum_{i=1}^{k} w_{ij} y_i) \quad \text{(i)}$$

where, f is an activation function, usually the sigmoid or tangent hyperbolic function. Activation function defines the relationship between inputs and outputs of a network in terms of degree of the non-linearity. To train a neural network model (update weight), the principle of gradient descent algorithm can be used by minimizing the cost function in a neural network model by following backpropagation algorithm and the steps are:

i. Input a dataset.
ii. Randomly initialize variables (weights and biases).
iii. Compute errors by calculating the difference between the actual and model predicted values.
iv. Propagate the error back through the same path to adjust the variables.
v. Steps ii-iv are repeated until it is confident to say that the variables are well defined.
vi. A prediction is made by applying these variables on a new unseen input dataset.

4. Types of neural networks architecture

There are several sorts of architectures of ANNs. However, the most broadly used ANNs are discussed here:

4.1 Feed Forward Neural Network Model

In feed forward neural network, data flows in one route alongside connecting pathways, from the input layer *via* the hidden layers to the final output layer. There is not any feedback (loops) i.e., the output of any layer does not have an effect on that same layer. A graphical illustration of feed forward network is given in figure 4.

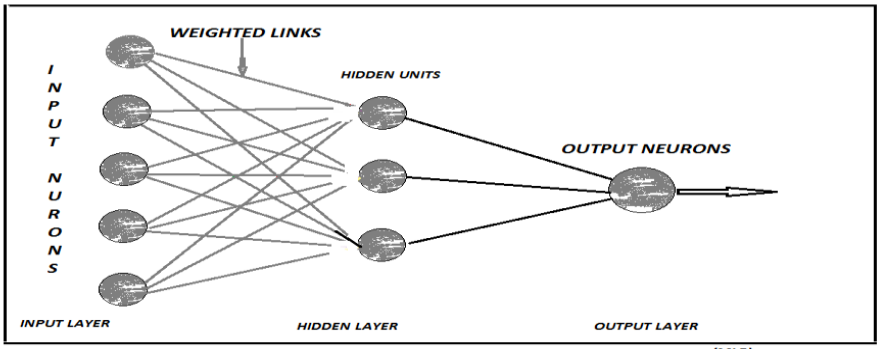


Fig. 4: Multi-layer Feed Forward Neural Network

4.2 Recurrent Neural Networks (RNNs) Model

An artificial neural network with recurrent connection is called as recurrent neural networks (RNNs) which are capable of learning features, modelling sequential data for sequence recognition and prediction. Recurrent neural networks are made up of high dimensional hidden states with non-linear dynamics. The structure of hidden states work as the memory of the network and state of the hidden layer at a time is conditioned on its previous state. The type of structure enables the RNNs to store, remember, and process past complex signals for long time periods. RNNs can map an input sequence to the output sequence at the current time step and predict the sequence in the next time step. RNNs are the heart of speech recognition, language modelling, translation, and in many more cases.

RNN is more efficient to learn the dependency between observations. The simple architecture and unrolled version of RNN is shown in figure 5. The simple RNN is a network with loops which allows persisting information to be passed from one step of the network to the next. In the diagram for the time steps $0,1,2,\ldots,t$; $x_0, x_1, x_2, \ldots, x_t$ are the inputs, A is the hidden state and $h_0, h_1, h_2, \ldots, h_t$ are the outputs. A_t hidden state is an activation function which takes its input from the hidden state of the previous step A_{t-1} and the output of the current step x_t.

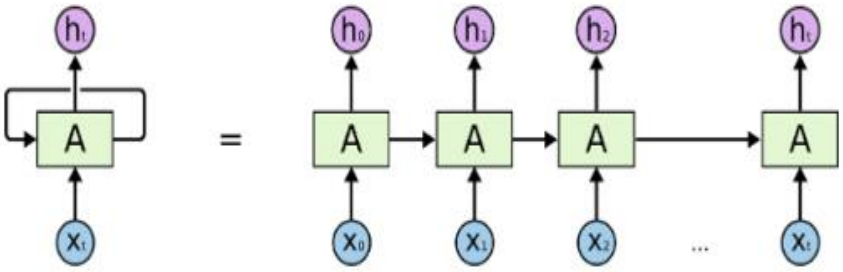


Fig. 5: An unrolled recurrent neural network

This process is described by the following equation

$$A_t = f(A_{t-1}, x_t) \quad \text{(ii)}$$

RNNs use backpropagation through time (BPTT) to optimize weights during training by using the chain rule to go back from the latest time step to the previous steps. Figure 6, presents a RNN looks like which is being unfolded into a full network. By unfolding, it means to write out the network for the complete sequence. For example, if the sequence is a sentence of five words, the network would be unrolled into five-layer neural network, one layer for each word. The formulas that govern the computation happening in a RNN are as follows:

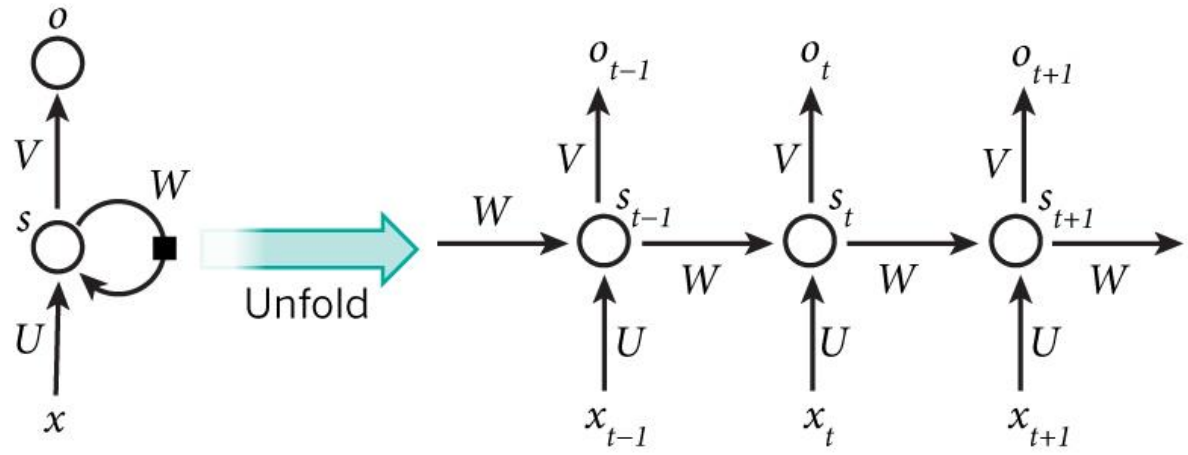


Fig. 6: A recurrent neural network and the unfolding in time of the computation involves in its forward computation

x_t is the input at time step t. s_t is the hidden state at time step t. It is the "memory" of the network. s_t is calculated based on the previous hidden state and the input at the current time step as

$$s_t = f(Ux_t + Ws_{t-1}) \quad \text{(iii)}$$

The function f is usually nonlinear such as $tanh$ or $ReLU$. s_{-1} which is required to calculate the first hidden state, is typically initialized to all zeroes. o_t is the output at time step t. For example, to predict the next word in a sentence it would be a vector of probabilities across our vocabulary.

$$o_t = softmax(Vs_t) \quad \text{(iv)}$$

5. Long-Short-Term-Memory (LSTM) Networks Model

LSTM is a special kind of RNNs that are designed to learn long term dependencies i.e., to memorize the sequence of data. The memory in LSTMs is called *cells* that are connected through layers. Each memory cell contains gates which handle information flow into and out of the cell. Internally these cells decide what to keep in (and what to erase from) memory. They then combine the previous state, the current memory and the input. Hence the gates which are based on sigmoidal neural network layer, enable the cells to optionally let data pass through or disposed. Each sigmoid layer yields numbers between 0 and 1, depicting every segment of data ought to be let through in each cell. More precisely, an estimation of 0 value implies that "let nothing pass through"; whereas; an estimation of 1 indicates that "let everything pass through". There are three types of gates in the LSTM unit with the aim of controlling the state of each cell:

i. Forget gate outputs a number between 0 and 1, where 1 shows "completely keep this"; whereas, 0 implies "completely ignore this".
ii. Memory gate chooses which new data need to be stored in the cell. First, a *sigmoid* layer, called the "input door layer" chooses which values will be modified. Next a *tanh* layer makes a vector of new candidate values that could be added to the state.
iii. Output gate decides what will be yield out of each cell. The yielded value will be passed on the cell state along with the filtered and newly added data.

The structure of the LSTM unit is shown in figure 7.

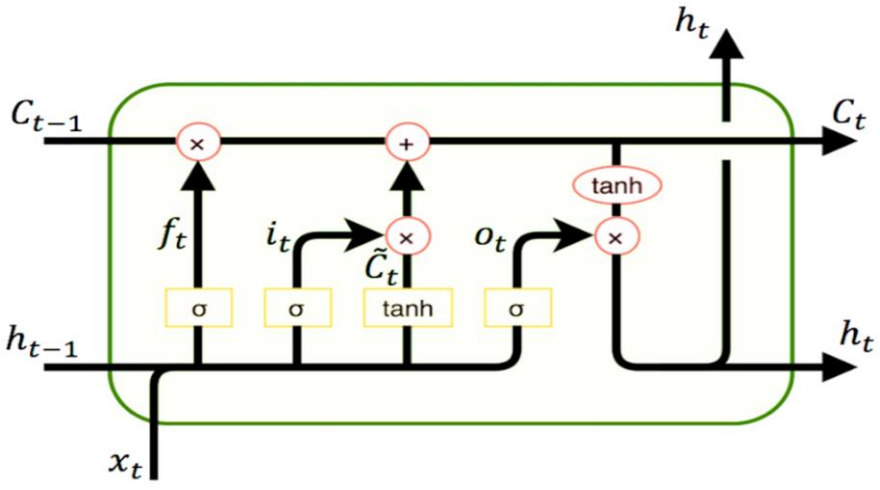


Fig. 7: LSTM unit

As seen from figure 7, Eq. v-vii, the LSTM unit gets the information from the previous state h_{t-1} and the input x_t, and uses the activation function (sigmoid) in the "input layer gate" to decide which part of the information to pass to the output and next LSTM unit.

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f) \quad \text{(v)}$$

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \quad \text{(vi)}$$

$$o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o) \quad \text{(vii)}$$

$$\widetilde{C_t} = tanh(W_C.[h_{t-1}, x_t] + b_C) \quad \text{(viii)}$$

$$C_t = f_t \otimes C_{t-1} + i_t \otimes \widetilde{C_t} \quad \text{(ix)}$$

$$h_t = o_t \otimes tanh(C_t) \quad \text{(x)}$$

Eq. v-vii describes the sigmoid function ($\sigma(x) = {}^{1}/_{1+e^{-x}}$) where *W's* and *b's* are the parameters (weights and biases) for forget, input and output gates. f_t, i_t and o_t are forget, input and output gates respectively. In Eq. viii, the *tanh* layer creates the vector of new candidate values $\widetilde{C_t}$ that could be added to the cell state. LSTM unit has two kinds of hidden state: "slow" state C_t and a "fast" state h_t. The slow state C_t is updated by summing the multiplication the forget gate f_t by the previous cell state C_{t-1} and the multiplication the input gate i_t by the new candidate value $\widetilde{C_t}$. The h_t state is updated using the hyperbolic tangent function (*tanh*) of C_t state and o_t output gate. The calculation in the step is pretty much straightforward and eventually leads to the output. However, the outputs consist of only the outputs those were decided to be carry forwarded in the previous steps and not all the outputs at once. The main feature of LSTM unit is that its cell state accumulates activities over time. As derivatives of the error are summed over time, they do not vanish quickly that's why LSTMs can implement tasks over long term dependencies.

6. Convolutional Neural Networks Model

Convolutional neural network (CNN), a deep learning model that has become dominant in various computer vision tasks, is attracting interest across a variety of domains, even in agriculture also. CNN is inspired by the organization of animal visual cortex and is designed to automatically and adaptively learn spatial hierarchies of features through backpropagation algorithm by using multiple building blocks, viz., convolution layers, pooling layers, and fully connected layers. Convolutional Neural Networks have wide applications in image and video

recognition, recommendation systems and natural language processing. For a given dataset of gray scale images with the standardized size of 32×32 pixels each a conventional feedforward neural network will need 1,024 input weights, plus one bias as each layer is full-connected with the previous input layer. But the major problem is by flattening the pixel image matrix to a long pixel value vector loses much of the image's spatial structure. The neural networks will have great difficulty with the problem as it is very easy to get over fitted and needs large volume of data and also computation time for training will increase. This spatial relation between pixels is expected and maintained by convolutional neural networks by using small squares of input data by studying internal feature representations. The characteristics are learned and used in the entire image, allowing the objects in the images to be moved throughout the scene or translated and still detectable by the network. This is why the network is very helpful for object recognition in photographs, picking out digits, faces, objects and so on with varying orientation.

The building blocks of convolutional neural networks are: Convolutional layers, pooling layers and fully-connected layers.

6.1. Convolutional Layers

Convolutional layers are comprised of filters and feature maps

6.1.1 Filters

The filters are actually the layer's neurons and are the locally connected networks, each of which learns some particular features of the images. They have weighted inputs and, like a neuron, produce an output value. The input size is a fixed square called a patch or a receptive field. If the convolutional layer is an input layer, then the input patch will be pixel values. Each filter moves around the images with the configured step size called as Stride. For example, for a filter of size 10×10 with stride 1, each time the filter window will overlay an image with an area of 10×10 pixels to calculate the inner product between the pixel values and the parameters, and will move horizontally and vertically by 1 pixel (stride) on an image until it covers the whole image

6.1.2 Feature Maps

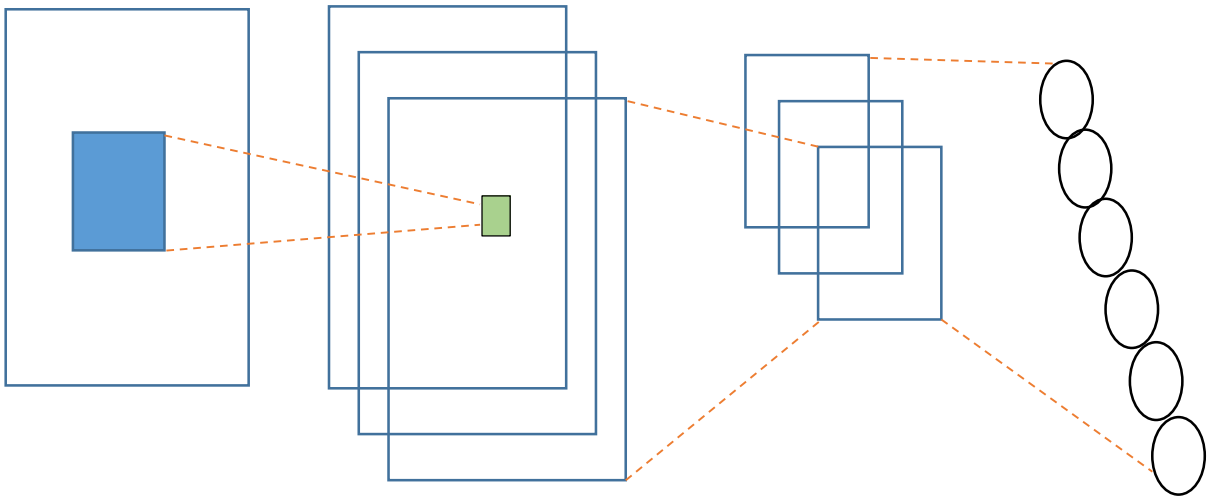
The feature map is the output of one filter applied to the previous layer. A given filter is drawn across the entire previous layer, moved one pixel at a time. Each position results in an activation of the neuron and the output is collected in the feature map.

6.2. Pooling Layers

The pooling layers down-sample the function map for the previous layers. Pooling layers follow a series of one or more convolutional layers and are structured to combine the characteristics learned and expressed in the feature map of the previous layers. There are two type of pooling layers: Max-pooling and Average pooling layers. Max pooling layer picks the maximum value from each window and average pooling layer calculate the average value from each window

6.3. Fully Connected Layers

Fully connected layers are the normal flat feedforward neural network layer. These layers may have a nonlinear activation function or a softmax activation in order to output probabilities of class predictions. Fully connected layers are used at the end of the network after feature extraction and consolidation has been performed by the convolutional and pooling layers. They are used to create final nonlinear combinations of features and for making predictions by the network. The convolutional neural network (CNN) model architecture is shown by the following figure 8 as



Input Image: For example: 6 × 6

Convolution Layer: Apply filters (For example 2 × 2 to generate feature maps

ReLU: Introduce *Non-linearity*, after every convolution layer

Maxpooling: operation on each feature map

Fully Connected Layer

Fig. 8: Convolutional Neural Network (CNN) Model Architecture

7. Application areas of deep learning

In recent years deep learning models are successfully applied in many fields from natural language processing, visual recognition, healthcare, language translations, gaming to self-driving car and in more situations. In agriculture, there are also a huge scope of deep learning techniques in respect of forecasting weather data, monitoring crop and soil health, decrease pesticide application by employing AI-enabled systems to detect pests, plant disease identification, plant recognition, land cover classification, weed classification, precision irrigation and crop yield estimation and in many agricultural situations.

8. Conclusion

Deep learning models are belonging to the field of artificial intelligence approach and work by following the principle of the network of human brain neurons to learn things automatically by extracting features and decisions making process in a human like way. The deep leaning architecture like convolutional neural network (CNN) model has achieved a tremendous success over other existing machine learning models for image and video recognition problem due to its high computational efficiency. A recurrent neural networks (RNNs) model is also a class of deep neural networks model that can use internal state (memory) to process sequences of inputs. Long-Short-Term-Memory (LSTM) networks are a modified version of RNNs with its architecture for predicting sequences containing longer term patterns of unknown length, due to their ability to maintain long term memory.

9. Suggested Readings

https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/

https://colah.github.io/posts/2015-08-Understanding-LSTMs/

Jha, G. K. and Sinha, K. (2013) Agricultural Price Forecasting Using Neural Network Model: An Innovative Information Delivery System, *Agricultural Economics Research Review*, **26 (2)**, 229-239.

Jha, G. K. and Sinha, K. (2014) Time-delay neural networks for time series prediction: an application to the monthly wholesale price of oilseeds in India, *Neural Computing and Applications*, **24 (3-4)**, 563-571.

Mandic, D. & Chambers, J. (2001) Recurrent neural networks for prediction: Learning algorithms, architectures and stability, Wiley, ISBN 978-0-471-49517-8.

Zhang, G., Patuwo, B.E. and Hu, M.Y. (1998) Forecasting with artificial neural networks: The state of the art, *International Journal of Forecasting*, **14**, 35-62.