Multivariate data analysis and data reduction techniques

V. Geethalakshmi, Principal Scientist
ICAR-CIFT, Cochin
geethasankar@gmail.com

SAS stands for Statistical Analysis System. It is a software suite developed by SAS Institute. The applications of SAS are data management, advanced analytics, predictive analytics, multivariate analysis and business intelligence and reporting with perfect graphics. More than 200 components are available in SAS. The important SAS components are:

- Base SAS: It is the most widely used component. It has data management facility. You can do data analysis using Base SAS.
- SAS/GRAPH: With the use SAS/Graph you can represent data as graphs. This makes data visualization easy.
- SAS/STAT: It lets you perform Statistical analysis, such as Variance, Regression, Multivariate, Survival and Psychometric analysis.
- SAS/ETS: It is suited for Time Series Analysis.

DATA step and PROC step form the basic building blocks of a SAS program. We start a program with a DATA step to create a SAS data set and then pass the data onto a PROC step. The PROC step processes the data.

Creating dataset

The dataset can be created and variables named by typing directly in the code. Another method is to import files from the computer into SAS workspace. To create a dataset say 'Game' use the following code :

```
DATA Game;              #Name the data set.
INPUT x,y,z;              #Define the variables in this data set
DATALINES;                #In the following lines type the data
58 65 70
23 45 89
11 25 32
;run;
```

The data file can also be read from Excel using 'import' command.

Importing Excel file

```
proc import out=work.myfile datafile="<pathname>" dbms=xlsx replace;
run;
```

Procedures in SAS

The PROC means command is used to extract the descriptive statistics from the dataset. Usage of PROC means and the various options are given below.

1)Descriptive statistics

```
proc means;
variables tpc tc fs;
run;
```

2)Descriptive statistics specified output

```
proc means data=work.myfile n mean max min range std fw=8;
run;
```

3)Descriptive statistics crosstabulation

```
proc means data=work.geetha maxdec=3;
variables TPC TC FS;
class Lake sp;
types () Lake*sp;
title Average microbial load;
run;
```

4) The By statement

```
proc means data=work.geetha;
by season;
variables TPC TC FS;
class Lake sp;
run;
```

5)Confidence limits for mean

```
proc means data=work.geetha fw=8 alpha=0.1 clm mean std;
variables TPC TC FS;
run;
```

Correlation and regression

PROC corr and PROC reg are used to compute correlations from the x,y data. The various options are described below :

Finding correlation

```
proc corr data=work.myfile;
variables salinity ph rainfall;
run;
```

Finding regression

```
proc reg data=work.geetha;
model TPC = salinity pH Temp;
run;
```

The selection option of PROC reg will specify the method of variable to be included in the model. The following example gives the data on fitness and code to fitting regression model using various selection methods.

```
data Fitness;
    input Age Weight Oxygen RunTime RestPulse RunPulse MaxPulse @@;
  datalines;
  44 89.47 44.609 11.37 62 178 182   40 75.07 45.313 10.07 62 185 185
  42 85.84 54.297  8.65 45 156 168   42 68.15 59.571  8.17 40 166 172
  38 89.02 49.874  9.22 55 178 180   47 77.45 44.811 11.63 58 176 176
  40 75.98 45.681 11.95 70 176 180   43 81.19 49.091 10.85 64 162 170
  44 81.42 39.442 13.08 63 174 176   38 81.87 60.055  8.63 48 170 186
  44 73.03 50.541 10.13 45 168 168   45 87.66 37.388 14.03 56 186 192
  45 66.45 44.754 11.12 51 176 176   47 79.15 47.273 10.60 47 162 164
  34 83.12 51.855 10.33 50 166 170   49 81.42 49.156  8.95 44 180 185
  49 69.63 40.836 10.95 57 168 172   51 77.91 46.672 10.00 48 162 168
  48 91.63 46.774 10.25 48 162 164   49 73.37 50.388 10.08 67 168 168
  47 73.37 39.407 12.63 58 174 176   54 79.38 46.080 11.17 62 156 165
  42 76.32 45.441  9.63 48 164 166   50 70.87 54.625  8.92 48 146 155
  41 67.25 45.118 11.08 48 172 172   54 91.63 39.203 12.88 44 168 172
  45 73.71 45.790 10.47 59 186 188   57 59.08 50.545  9.93 49 148 155
  49 76.32 48.673  9.40 56 186 188   48 61.24 47.920 11.50 52 170 176
  42 82.78 47.467 10.50 53 170 172
  ;
proc reg data=fitness;
    model Oxygen=Age Weight RunTime RunPulse RestPulse MaxPulse
        / selection=forward;
    model Oxygen=Age Weight RunTime RunPulse RestPulse MaxPulse
        / selection=backward;
    model Oxygen=Age Weight RunTime RunPulse RestPulse MaxPulse
        / selection=maxr;
  run;
```

Multivariate data analysis

Multivariate Data Analysis refers to any statistical technique used to analyze data that arises from more than one variable. The most popular multivariate data analysis technique is the Principal Component Analysis. Principal component analysis is a variable reduction procedure. It is useful when you have obtained data on a number of variables (possibly a large number of variables), and believe that there is some redundancy in those variables. In this case, redundancy means that some of the variables are correlated with one another, possibly because they are measuring the same construct. Because of this redundancy, it is always possible to reduce the observed variables into a smaller number of principal components (artificial variables) that will account for most of the variance in the observed variables. Technically, a principal component can be defined as a linear combination of optimally-weighted observed variables.

Principal Component Analysis (PCA)

PCA helps identifying patterns in data, and expressing the data in such a way as to highlight their similarities and differences. Since patterns in data can be hard to find in data of high dimension, where the luxury of graphical representation is not available, PCA is a powerful tool for analysing data. Principal Component Analysis computes the covariance matrix and the eigen vectors. Let X_1, X_2,....,X_p be the p variables, say, spoilage indicators of fish based on lab analysis for shelf life study. The values recorded by each sample on the indicators of spoilage. In PCA, its possible to compute a score for each sample on a given principal component. The sample's actual scores on the p variables would be optimally weighted and then summed to compute their scores on a given component.

The general form for the formula to compute scores on the first component extracted (created) in a principal component analysis:

$C_1 = b_{11}(X_1) + b_{12}(X_2) + ... b_{1p}(X_p)$

where

C_1= the subject's score on principal component 1 (the first component extracted)

b_{1p} = the regression coefficient (or weight) for observed variable p, as used in creating principal component 1

X_p= the subject's score on observed variable p.

In PCA, the observed variables are weighted in such a way that the resulting components account for a maximal amount of variance in the data set. Similarly C_2 gives the sample score on another Principal Component which is a linear combination of variables with another set of b's and so on. The number of components extracted in a principal component analysis is equal to the number of observed variables being analyzed. In most analyses, only the first few components account for meaningful amounts of variance, so only these first few components are retained, interpreted, and used in subsequent analyses

The first component extracted in a Principal Component Analysis accounts for a maximal amount of total variance in the observed variables. The second component extracted will account for a maximal amount of variance in the data set that was not accounted for by the first component. The second PC will be uncorrelated with the first one. A principal component analysis proceeds in this fashion, with each new component accounting for progressively smaller and smaller amounts of variance (this is why only the first few components are usually retained and interpreted). When the analysis is complete, the resulting components will display varying degrees of correlation with the observed variables, but are completely uncorrelated with one another.

SAS procedure for PCA

Principal component analysis can be performed using either the PRINCOMP or FACTOR procedures.

PROC FACTOR statement

The general form for the SAS program to perform a principal component analysis is presented here:

```
PROC FACTOR
DATA=data-set-name
SIMPLE
METHOD=PRIN
PRIORS=ONE
MINEIGEN=p
SCREE
ROTATE=VARIMAX
ROUND
FLAG=desired-size-of-"significant"-factor-loadings ;
VAR variables-to-be-analyzed ; RUN;
```

Options used with PROC FACTOR

The PROC FACTOR statement begins the FACTOR procedure, and a number of options may be requested in this statement before it ends with a semicolon. Some options that may be especially useful in social science research are:

FLAG=desired-size-of-”significant”-factor-loadings

causes the printer to flag (with an asterisk) any factor loading whose absolute value is greater than some specified size. For example, if you specify

FLAG=.35

an asterisk will appear next to any loading whose absolute value exceeds .35. This option can make it much easier to interpret a factor pattern. Negative values are not allowed in the FLAG option, and the FLAG option should be used in conjunction with the ROUND option.

METHOD=factor-extraction-method

specifies the method to be used in extracting the factors or components. The current program specifies METHOD=PRIN to request that the principal axis (principal factors) method be used for the initial extraction. This is the appropriate method for a principal component analysis.

MINEIGEN=p

specifies the critical eigenvalue a component must display if that component is to be retained (here, p = the critical eigenvalue). For example, the current program specifies

MINEIGEN=1

This statement will cause PROC FACTOR to retain and rotate any component whose eigenvalue is 1.00 or larger. Negative values are not allowed.

NFACT=n

allows you to specify the number of components to be retained and rotated, where n = the number of components.

OUT=name-of-new-data-set

creates a new data set that includes all of the variables of the existing data set, along with factor scores for the components retained in the present analysis. Component 1 is given the varible name FACTOR1, component 2 is given the name FACTOR2, and so forth. It must be used in conjunction with the NFACT option, and the analysis must be based on raw data.

PRIORS=prior-communality-estimates

specifies prior communality estimates. Users should always specify PRIORS=ONE to perform a principal component analysis.

ROTATE=rotation-method

specifies the rotation method to be used. The preceding program requests a varimax rotation, which results in orthogonal (uncorrelated) components. Oblique rotations may also be requested; oblique rotations are discussed in Chapter 2.

ROUND

causes all coefficients to be limited to two decimal places, rounded to the nearest integer, and multiplied by 100 (thus eliminating the decimal point). This generally makes it easier to read the coefficients because factor loadings and correlation coefficients in the matrices printed by PROC FACTOR are normally carried out to several decimal places.

SCREE

creates a plot that graphically displays the size of the eigenvalue associated with each component. This can be used to perform a scree test to determine how many components should be retained. Specifying the SCREE option in the PROC FACTOR statement causes the SAS System to print an eigenvalue plot as part of the output.

SIMPLE

requests simple descriptive statistics: the number of usable cases on which the analysis was performed, and the means and standard deviations of the observed variables.

The VAR statement

The variables to be analyzed are listed in the VAR statement, with each variable separated by at least one space. Remember that the VAR statement is a separate statement, not an option within the FACTOR statement, so don't forget to end the FACTOR statement with a semicolon before beginning the VAR statement.

PROC princomp

The PRINCOMP procedure performs principal component analysis. As input you can use raw data, a correlation matrix, a covariance matrix, or a sums of squares and crossproducts (SSCP) matrix. You can create output data sets containing eigenvalues, eigenvectors, and standardized or unstandardized principal component scores.

The PROC PRINCOMP statement starts the PRINCOMP procedure and optionally identifies input and output data sets, specifies the analyses performed, and controls displayed output.

```
proc princomp cov out=a;
var <variable_list>;
run;
```

Summary of PROC PRINCOMP Statement Options

Option	Description
Specify datasets	
DATA=	specifies input data set name
OUT=	specifies output data set name
OUTSTAT=	specifies output data set name containing various statistics
Specify details of analysis	
COV	computes the principal components from the covariance matrix
N=	specifies the number of principal components to be computed
NOINT	Omits the intercept from the model
PREFIX=	specifies a prefix for naming the principal components
RPREFIX=	specifies a prefix for naming the residual variables
SINGULAR=	Specifies the singularity criterion
STD	standardizes the principal component scores
VARDEF=	specifies the divisor used in calculating variances and standard deviations
Supress the display of output	
NOPRINT	Supresses display of all output
Specify ODS graphics details	
PLOTS=	specifies options that control the details of the plots

The following list provides details about these options.

COVARIANCE

COV

computes the principal components from the covariance matrix. If you omit the COV option, the correlation matrix is analyzed. Use of the COV option causes variables with large variances to be more strongly associated with components with large eigenvalues and causes variables with small variances to be more strongly associated with components with small eigenvalues. You should not specify the COV option unless the units in which the variables are measured are comparable or the variables are standardized in some way.

DATA=SAS-data-set

specifies the SAS data set to be analyzed. The data set can be an ordinary SAS data set or a TYPE=ACE, TYPE=CORR, TYPE=COV, TYPE=FACTOR, TYPE=SSCP, TYPE=UCORR, or TYPE=UCOV data set (see Appendix A, Special SAS Data Sets). Also, the PRINCOMP procedure can read the _TYPE_='COVB' matrix from a TYPE=EST data set. If you omit the DATA= option, the procedure uses the most recently created SAS data set.

N=number

specifies the number of principal components to be computed. The default is the number of variables. The value of the N= option must be an integer greater than or equal to zero.

NOINT

omits the intercept from the model. In other words, the NOINT option requests that the covariance or correlation matrix not be corrected for the mean. When you use the PRINCOMP procedure with the NOINT option, the covariance matrix and, hence, the standard deviations are not corrected for the mean. If you are interested in the standard deviations corrected for the mean, you can get them by using a procedure such as the MEANS procedure.

If you use a TYPE=SSCP data set as input to the PRINCOMP procedure and list the variable Intercept in the VAR statement, the procedure acts as if you had also specified the NOINT option. If you use NOINT and also create an OUTSTAT= data set, the data set is TYPE=UCORR or TYPE=UCOV rather than TYPE=CORR or TYPE=COV.

NOPRINT

suppresses the display of all output. Note that this option temporarily disables the Output Delivery System (ODS). For more information, see Chapter 20, Using the Output Delivery System.

OUT=SAS-data-set

creates an output SAS data set that contains all the original data as well as the principal component scores.

If you want to create a permanent SAS data set, you must specify a two-level name. For details about OUT= data sets, see the section Output Data Sets. See SAS Language Reference: Concepts for more information about permanent SAS data sets.

OUTSTAT=SAS-data-set

creates an output SAS data set that contains means, standard deviations, number of observations, correlations or covariances, eigenvalues, and eigenvectors. If you specify the COV option, the data set is TYPE=COV or TYPE=UCOV, depending on the NOINT option, and it contains covariances; otherwise, the data set is TYPE=CORR or TYPE=UCORR, depending on the NOINT option, and it contains correlations. If you specify the PARTIAL statement, the OUTSTAT= data set contains R squares as well.

If you want to create a permanent SAS data set, you must specify a two-level name. For details about OUTSTAT= data sets, see the section Output Data Sets. See SAS Language Reference: Concepts for more information about permanent SAS data sets.