SAS MACRO FOR GENERATION OF TREND FREE DESIGNS UNDER TWO SOURCES OF HETEROGENEITY BALANCED FOR NON-DIRECTIONAL NEIGHBOUR EFFECTS

Arpan Bhowmik, Seema Jaggi, Eldho Varghese and Sunil Kumar Yadav

ICAR-Indian Agricultural Statistics Research Institute, New Delhi
[1]ICAR-Central Marine Fisheries Research Institute, Kochi

Case 1: SAS MACRO FOR GENERATION OF COMPLETE TREND FREE DESIGNS UNDER TWO SOURCES OF HETEROGENEITY BALANCED FOR NON-DIRECTIONAL NEIGHBOUR EFFECTS

In agricultural experiments, many situations arise in which apart from the known source of variations, the response may also depend on the spatial or temporal position of the experimental unit, i.e. **systematic trend** effects may affect the experimental units. Beside, agricultural experiments may witness the evidences of neighbour effects. Although these effects are remote, but can affect the precision of the experiments if not taken in to consideration. Thus, both trend effects and neighbour effects need to be incorporated in to the model for proper model specification. Here, the following SAS macro using SAS 9.3 has been developed to generate a class of complete trend free designs under two sources of heterogeneity balanced for non-directional neighbour effects where the neighbour effects have been considered row-wise. Here, user need to enter the **number of treatments** as **v ($\geq$2)** where v should be a prime number. If user run the macro after entering any prime number ($\geq$2) as the value of v, then the SAS Macro will generate a particular class of complete trend free designs under two sources of heterogeneity balanced for non-directional neighbour effects [where the neighbour effects have been considered row-wise] based on the method developed by Bhowmik *et al*. (2018) under the heading **Trend Free Design under Two Sources of Heterogeneity Setup Balanced for Non-directional Neighbour Effects**. Along with the design, the macro will also generate a polynomial coefficient (linear) which will be used to measure the effect of trend component. Once user run the macro, every time the SAS macro would also generate a word file containing the output. User can then save the word file. If user does not enter a prime number and execute the programme, the macro will display the message **Entered number is not a prime number**. The code and output are as follows:

CODE

```
/*This macro will generate designs with maximum v=26*26*/
%let v=7;/*Enter a prime number*/
ods rtf  file= 'output.rtf' startpage=no;
proc iml;
TRT_Labels1={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};
```

```
TRT_Labels2=j(26*26,1,'ab');
k=1;
do i=1 to 26;
do j=1 to 26;
TRT_Labels2[k,1]=TRT_Labels1[i,1]+TRT_Labels1[j,1];
k=k+1;
end;
end;
TRT_Labels2=TRT_Labels1//TRT_Labels2;
*print TRT_Labels;
TRT_Labels=j(&v,1,'ab');
do i=1 to &v;
TRT_Labels[i, ]=TRT_Labels2[i, ];
end;
*print TRT_Labels;
pp1=1;
do i=2 to &v-1;
pp=mod(&v,i);
if pp=0 then pp1=0;
end;
if pp1=0 then do;
print 'Entered number is not a prime number';
end;
if pp1^=0 then do;
x=j(&v,&v,0);
m=mod(&v,2);
a=j(&v,1,0);
do i=1 to &v;
do j=i to &v;
if m=1 then
                    do;
```

```
a[j,1]=-((&v-1)/2)+(i-1);
end;
else
do;
if -(&v/2)+(i-1)<0 then do;
a[j,1]=-(&v/2)+(i-1);
end;
else do;
a[j,1]=-(&v/2)+i;
end;
end;
end;
end;
*print a;
col=a`;
row=j(&v,1,0);
do i=1 to &v;
row[i, ]=i;
end;
*print row col;
row1=char(row,5,0);
col1=char(col,5,0);
/*contructing TWO LS*/
x=j(&v,&v,0);
k=0;
do j=1 to (&v+1)/2;
x[j+k, 1]=j;
k=k+1;
end;
k=1;
do j=1 to (&v-1)/2;
```

```
x[j+k, 1]=(nrow(x)+1)-j;
k=k+1;
end;
do i=1 to nrow(x);
k=2;
do j=1 to (&v-1)/2;
x[i,j+k]=mod(x[i,1]+j,&v);
if x[i,j+k]=0 then x[i,j+k]=&v;
k=k+1;
end;
end;
do i=1 to nrow(x);
k=1;
do j=1 to (&v-1)/2;
x[i,j+k]=mod(x[i,1]+&v-j,&v);
if x[i,j+k]=0 then x[i,j+k]=&v;
k=k+1;
end;
end;
x=x-1;
*print x;
*print a1;
b=j(&v,1,0);
do i=1 to nrow(x);
b[i,1]=(mod((i-1)*2,&v))+1;
end;
*print b;
x1=j(&v,&v,0);
do i=1 to nrow(x1);
do j=1 to ncol(x1);
x1[i,j]=x[b[i, ],j];
```

```
end;
end;
x=x+1;
x1=x1+1;
*print x;
*print x1;
/****/
x2=j(&v,&v,"ab");
do i=1 to &v;
do j=1 to &v;
x2[i,j]=TRT_Labels[x1[i,j],];
end;
end;
*print x2;
x3=char(x,5,0);
*print x3;
Trend_Free_Design=x2+x3;
*print Trend_Free_Design;
print 'Trend Free Design under Two Sources of Heterogeneity Setup Balanced for Non-
directional Neighbour Effects';
print Trend_Free_Design[rowname=row1 colname=col1];
print 'Column represents trend component, Row represent Source 1,
Alphabets with in Row-column intersection is Source 2 and Number represents the treatments';
end;
run;
ods rtf close;
quit;
```

Trend Free Design under Two Sources of Heterogeneity Setup Balanced for Non-directional Neighbour Effects

Trend_Free_Design							
	-3	**-2**	**-1**	**0**	**1**	**2**	**3**
1	A 1	G 7	B 2	F 6	C 3	E 5	D 4
2	B 7	A 6	C 1	G 5	D 2	F 4	E 3
3	C 2	B 1	D 3	A 7	E 4	G 6	F 5
4	D 6	C 5	E 7	B 4	F 1	A 3	G 2
5	G 3	F 2	A 4	E 1	B 5	D 7	C 6
6	F 5	E 4	G 6	D 3	A 7	C 2	B 1
7	E 4	D 3	F 5	C 2	G 6	B 1	A 7

Column represents trend component, Row represent Source 1,Alphabets with in Row-column intersection is Source 2 and Number represents the treatments

Case 2: SAS MACRO FOR GENERATION OF INCOMPLETE (ROW-WISE) TREND FREE DESIGNS UNDER TWO SOURCES OF HETEROGENEITY PARTIALLY BALANCED FOR NON-DIRECTIONAL NEIGHBOUR EFFECTS

In agricultural experiments, many situations arise in which apart from the known source of variations, the response may also depend on the spatial or temporal position of the experimental unit, i.e. **systematic trend** effects may affect the experimental units. Beside, agricultural experiments may witness the evidences of neighbour effects. Although these effects are remote, but can affect the precision of the experiments if not taken in to consideration. Thus, both trend effects and neighbour effects need to be incorporated in to the model for proper model specification. Here, the following SAS macro using SAS 9.3 has been developed to generate a class of incomplete (row-wise) trend free designs under two sources of heterogeneity partially balanced for non-directional neighbour effects when neighbour effects have been considered row-wise. Here, user need to enter the **number of treatments** as **v ($\geq$2)** where v should be a prime number. If user run the macro after entering any prime number ($\geq$2) as the value of v, then the SAS Macro will generate a particular class class of incomplete (row-wise) trend free designs under two sources of heterogeneity partially balanced for non-directional neighbour effects [when neighbour effects have been considered row-wise] based on the method developed by Bhowmik *et al*. (2018) under the heading **Incomplete (row-wise) Trend Free Design under Two Sources of Heterogeneity Setup Partially Balanced for Non-directional Neighbour**

Effects'. Along with the design, the macro will also generate a polynomial coefficient (linear) which will be used to measure the effect of trend component. Once user run the macro, every time the SAS macro would also generate a word file containing the output. User can then save the word file. If user does not enter a prime number and execute the programme, the macro will display the message **Entered number is not a prime number**. The code and output are as follows:

CODE

```
/*This macro will generate designs with maximum v=26*26*/
%let v=7;/*Enter a prime number*/
ods rtf  file= 'output.rtf' startpage=no;
proc iml;
TRT_Labels1={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};
TRT_Labels2=j(26*26,1,'ab');
k=1;
do i=1 to 26;
do j=1 to 26;
TRT_Labels2[k,1]=TRT_Labels1[i,1]+TRT_Labels1[j,1];
k=k+1;
end;
end;
TRT_Labels2=TRT_Labels1//TRT_Labels2;
*print TRT_Labels;
TRT_Labels=j(&v,1,'ab');
do i=1 to &v;
TRT_Labels[i, ]=TRT_Labels2[i, ];
end;
*print TRT_Labels;
pp1=1;
do i=2 to &v-1;
pp=mod(&v,i);
if pp=0 then pp1=0;
end;
```

```
if pp1=0 then do;
print 'Entered number is not a prime number';
end;
if pp1^=0 then do;
x=j(&v,&v,0);
m=mod(&v,2);
a=j(&v-1,1,0);
kk=2;
do i=1 to &v-1;
a[i,1]=-(&v)+kk;
kk=kk+2;
end;
*print a;
col=a`;
row=j(&v,1,0);
do i=1 to &v;
row[i, ]=i;
end;
*print row col;
row1=char(row,5,0);
col1=char(col,5,0);
/*contructing TWO LS*/
x=j(&v,&v,0);
k=0;
do j=1 to (&v+1)/2;
x[j+k, 1]=j;
k=k+1;
end;
k=1;
do j=1 to (&v-1)/2;
x[j+k, 1]=(nrow(x)+1)-j;
```

```
k=k+1;
end;
do i=1 to nrow(x);
k=2;
do j=1 to (&v-1)/2;
x[i,j+k]=mod(x[i,1]+j,&v);
if x[i,j+k]=0 then x[i,j+k]=&v;
k=k+1;
end;
end;
do i=1 to nrow(x);
k=1;
do j=1 to (&v-1)/2;
x[i,j+k]=mod(x[i,1]+&v-j,&v);
if x[i,j+k]=0 then x[i,j+k]=&v;
k=k+1;
end;
end;
x=x-1;
*print x;
*print a1;
b=j(&v,1,0);
do i=1 to nrow(x);
b[i,1]=(mod((i-1)*2,&v))+1;
end;
*print b;
x1=j(&v,&v,0);
do i=1 to nrow(x1);
do j=1 to ncol(x1);
x1[i,j]=x[b[i, ],j];
end;
```

```
end;
x=x+1;
x1=x1+1;
*print x;
*print x1;
/****/
x2=j(&v,&v-1,"ab");
do i=1 to &v;
do j=1 to &v-1;
x2[i,j]=TRT_Labels[x1[i,j],];
end;
end;
*print x2;
xx=j(&v,&v-1,0);
do i=1 to &v;
do j=1 to &v-1;
xx[i,j]=x[i,j];
end;
end;
*print xx;
x3=char(xx,5,0);
*print x3;
Trend_Free_Design=x2+x3;
*print Trend_Free_Design;
print 'Incomplete (row-wise) Trend Free Design under Two Sources of Heterogeneity Setup
Partially Balanced for Non-directional Neighbour Effects';
print Trend_Free_Design[rowname=row1 colname=col1];
print 'Column represents trend component, Row represent Source 1,
Alphabets with in Row-column intersection is Source 2 and Number represents the treatments';
end;
run;
```

```
ods rtf close;
quit;
```

Incomplete (row-wise) Trend Free Design under Two Sources of Heterogeneity Setup Partially Balanced for Non-directional Neighbour Effects

Trend_Free_Design						
	-5	**-3**	**-1**	**1**	**3**	**5**
1	A 1	G 7	B 2	F 6	C 3	E 5
2	B 7	A 6	C 1	G 5	D 2	F 4
3	C 2	B 1	D 3	A 7	E 4	G 6
4	D 6	C 5	E 7	B 4	F 1	A 3
5	G 3	F 2	A 4	E 1	B 5	D 7
6	F 5	E 4	G 6	D 3	A 7	C 2
7	E 4	D 3	F 5	C 2	G 6	B 1

Column represents trend component, Row represent Source 1,Alphabets with in Row-column intersection is Source 2 and Number represents the treatments

Reference

Bhowmik, Arpan, Jaggi, Seema, Varghese, Eldho and Yadav, S. K. (2018). Some investigations on trend resistant Row-column designs. Project Report, I.A.S.R.I/PR-04/2018. Available at `http://krishi.icar.gov.in/jspui/handle/123456789/47488`