

7.4: Blocking in 2 Dimensions - Latin Square

The fundamental idea of blocking can be extended to more dimensions. However, the full use of multiple blocking variables in a complete block design usually requires many experimental units. Latin Square design can be useful when we want to achieve blocking simultaneously in two directions with a limited number of experimental units.

The limitation is that the Latin Square experimental layout will only be possible if:

$$\text{number of Row blocks} = \text{number of Column blocks} = \text{number of treatment levels} \quad (7.4.1)$$

The experimental design process begins with a *Standard Latin Square*. These have the treatment levels ordered across the first row and first column. For example, a single factor with three levels (A, B, C) to be blocked in two directions could begin with this standard 3×3 square:

A	B	C
B	C	A
C	A	B

To randomize, first randomly permute the order of the rows and produce a new square.

B	C	A
C	A	B
A	B	C

Then randomly permute the order of the columns to yield the final square for the experimental layout.

C	A	B
A	B	C
B	C	A

This process assures that any row or column will have all treatment levels. To obtain the design in SAS we can use:

```
proc plan;
  factors Row=4 ordered Col=4 ordered / noprint;
  treatments Treatment=4 cyclic;
  output out=LatinSquare
    Row cvals=('RowBlock 1' 'RowBlock 2' 'RowBlock 3' 'RowBlock 4') random
    Col cvals=('ColBlock 1' 'ColBlock 2' 'ColBlock 3' 'ColBlock 4') random
    Treatment nvals=(1 2 3 4) random;
run;
```

The ANOVA for the Latin Square is a direct extension of the RCBD with random blocking effects. The SAS `random` statement has to be modified accordingly to incorporate both blocking factors and with the assumption of no interaction between them (because of only one observation for each cell). For example, we could use the following SAS code to estimate the model:

```
proc mixed data=LatinSquare method=type3;
  class Row Col Treatment;
  model Response = Treatment;
```

```
random Row Col;  
run;
```

Using R

To obtain a Latin Square Design for four treatments we can use the following commands:

```
library(magic)  
latin_square_design<-rlatin(4)  
# latin_square_design  
#      [,1] [,2] [,3] [,4]  
# [1,]    3    1    2    4  
# [2,]    4    2    1    3  
# [3,]    2    4    3    1  
# [4,]    1    3    4    2
```

This page titled [7.4: Blocking in 2 Dimensions - Latin Square](#) is shared under a [CC BY-NC 4.0](#) license and was authored, remixed, and/or curated by [Penn State's Department of Statistics](#).