

# Latin Squares in Experimental Design

Lei Gao

Michigan State University

December 10, 2005

**Abstract:** For the past three decades, Latin Squares techniques have been widely used in many statistical applications. Much effort has been devoted to Latin Square Design. In this paper, I introduce the mathematical properties of Latin squares and the application of Latin squares in experimental design. Some examples and SAS codes are provided that illustrates these methods.

Work done in partial fulfillment of the requirements of Michigan State University MTH 880 advised by Professor J. Hall.

# Index

Index .....	2
1. Introduction.....	3
1.1 Latin square.....	3
1.2 Orthogonal array representation .....	3
1.3 Equivalence classes of Latin squares.....	3
2. Latin Square Design.....	4
2.1 Latin square design .....	4
2.2 Pros and cons of Latin square design.....	4
2.3 An example of Latin square design .....	5
2.4 Procedure to create a Latin square design.....	6
3. Analysis of variance.....	8
3.1 Model for the classical Latin square design.....	8
3.2 Analysis of a Latin square.....	9
3.3 SAS program and output.....	10
4. Latin rectangles and multiple Latin squares .....	14
4.1 An example of Latin rectangle design .....	14
4.2 SAS program and output.....	15
5. Crossover designs: .....	17
5.1 An example of crossover design.....	17
5.2 Model of crossover design .....	19
References:.....	19

# 1. Introduction

## 1.1 Latin square

A Latin square is an  $n \times n$  table filled with  $n$  different symbols in such a way that each symbol occurs exactly once in each row and exactly once in each column. Here are two examples.

$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{bmatrix} \quad \begin{bmatrix} a & b & d & c \\ b & c & a & d \\ c & d & b & a \\ d & a & c & b \end{bmatrix}$$

## 1.2 Orthogonal array representation

If each entry of an  $n \times n$  Latin square is written as a triple  $(r,c,s)$ , where  $r$  is the row,  $c$  is the column, and  $s$  is the symbol, we obtain a set of  $n^2$  triples called the orthogonal array representation of the square. For example, the orthogonal array representation of the first Latin square displayed above is

$$\{ (1,1,1), (1,2,2), (1,3,3), (2,1,2), (2,2,3), (2,3,1), (3,1,3), (3,2,1), (3,3,2) \},$$

where for example the triple  $(2,3,1)$  means that in row 2, column 3 there is the symbol 1.

The definition of a Latin square can be written in terms of orthogonal arrays as follows:

- There are  $n^2$  triples of the form  $(r,c,s)$ , where  $1 \leq r, c, s \leq n$ .
- All of the pairs  $(r,c)$  are different, all the pairs  $(r,s)$  are different, and all the pairs  $(c,s)$  are different.

The orthogonal array representation shows that rows, columns and symbols play rather similar roles, as will be made clear below.

## 1.3 Equivalence classes of Latin squares

Many operations on a Latin square produce another Latin square (for example, turning it upside down).

If we permute the rows, permute the columns, and permute the names of the symbols of a Latin square, we obtain a new Latin square said to be isotopic to the first. Isotopism is an equivalence relation, so the set of all Latin squares is divided into subsets, called

isotopy classes, such that two squares in the same class are isotopic and two squares in different classes are not isotopic.

Another type of operation is easiest to explain using the orthogonal array representation of the Latin square. If we systematically and consistently reorder the three items in each triple, another orthogonal array (and, thus, another Latin square) is obtained. For example, we can replace each triple  $(r,c,s)$  by  $(c,r,s)$  which corresponds to transposing the square (reflecting about its main diagonal), or we could replace each triple  $(r,c,s)$  by  $(c,s,r)$ , which is a more complicated operation. Altogether there are 6 possibilities including “do nothing”, giving us 6 Latin squares called the conjugates (also parastrophes) of the original square.

Although a Latin square is a simple object to a mathematician, it is multifaceted to an experimental designer.

## 2. Latin Square Design

### 2.1 Latin square design

A Latin square design is a method of placing treatments so that they appear in a balanced fashion within a square block or field. Treatments appear once in each row and column. Replicates are also included in this design.

- Treatments are assigned at random within rows and columns, with each treatment once per row and once per column.
- There are equal numbers of rows, columns, and treatments.
- Useful where the experimenter desires to control variation in two different directions

The Latin square design, perhaps, represents the most popular alternative design when two (or more) blocking factors need to be controlled for. A Latin square design is actually an extreme example of an incomplete block design, with any combination of levels involving the two blocking factors assigned to *one* treatment only, rather than to all!

### 2.2 Pros and cons of Latin square design

The advantages of Latin square designs are:

- They handle the case when we have several nuisance factors and we either cannot combine them into a single factor or we wish to keep them separate.
- They allow experiments with a relatively small number of runs.

The disadvantages are:

- The number of levels of each blocking variable must equal the number of levels of the treatment factor.
- The Latin square model assumes that there are no interactions between the blocking variables or between the treatment variable and the blocking variable.

### 2.3 An example of Latin square design

Actually, in many cases, Latin squares are necessary because one such combination of levels from two blocking factors can be combined with one treatment, and not all. The following example taken from Mead *et al.* (2003) illustrates this:

**Example 1:** An experiment to investigate the effects of various dietary starch levels on milk production was conducted on four cows. The four diets, T1, T2, T3, and T4, (in order of increasing starch equivalent), were fed for three weeks to each cow and the total yield of milk in the third week of each period was recorded (i.e. third week to minimize carry-over effects due to the use of treatments administered in a previous period). That is, the trial lasted 12 weeks since each cow received each treatment, and each treatment required three weeks. The investigator felt strongly that time period effects might be important (i.e. earlier periods in the experiment might influence milk yields differently compared to later periods). Hence, the investigator wanted to block on both cow and period. However, each cow cannot possibly receive more than one treatment during the same time period; that is, all possible cow-period blocking combinations could not logically be considered.

To start the randomization for a Latin square that accommodates these types of concerns, let's choose at random from one of the 4 standard Latin squares when  $a = 4$  treatments

**Table 2.1 A standard 4×4 Latin square**

	Column 1	Column 2	Column 3	Column 4
Row 1	A	B	C	D
Row 2	B	A	D	C
Row 3	C	D	B	A
Row 4	D	C	A	B

The two blocking variables in a Latin square design are often generically labeled as row and column blocking variables. In this example, cow is identified as the column variable and period as the row variable. Standard Latin squares are Latin squares in which elements of the first row and first column are arranged alphabetically by treatment category (i.e. the letters in the square above denote different treatments). There are a number of standard Latin squares that might exist for different values of  $a$  (i.e. total number of treatment effects).

For each value of  $a$ , (the size of the square), there are a large number of different  $a$  by  $a$  squares that have the Latin square property that each letter (treatment group label) appears once in each row and once in each column. As with randomized block designs, in order to make the analysis of data from a design statistically valid, we must choose one design randomly from a larger set of possible Latin squares.

## **2.4 Procedure to create a Latin square design**

An appropriate randomization strategy is as follows:

- Write down any Latin square of the required size (it could be a standard Latin square).
- Randomize the order of the rows.
- Randomize the order of the columns.
- Randomize the allocation of treatments to the letters of the square.

Consider Example 1, to randomize the order of the rows, choose in random order, the numbers (e.g., 1, 2, 3, 4) of the rows (you could use a computer to do this). Suppose in random order, one chooses 2, 4, 1, 3. Then one should rearrange the order of the rows as follows:

**Table 2.2 A 4×4 Latin square (rearrange the order of rows)**

	Column 1	Column 2	Column 3	Column 4
Row 2	B	A	D	C
Row 4	D	C	A	B
Row 1	A	B	C	D
Row 3	C	D	B	A

Now let's randomize the order of the columns. Suppose the following order: 1, 4, 3, 2 is chosen at random. Then one should rearrange the order of the columns as follows:

**Table 2.3 A 4×4 Latin square  
(rearrange the order of rows and columns)**

	Column 1	Column 4	Column 3	Column 2
Row 2	B	C	D	A
Row 4	D	B	A	C
Row 1	A	D	C	B
Row 3	C	A	B	D

Finally randomize treatments to the letters. Choosing at random, say T4, T1, T3, T2, in that order, one should assign, say A as Treatment T4, B as T1, C as T3 and D as T2. i.e.

**Table 2.4 A 4×4 Latin square (rearrange the order of rows and columns)  
(also randomize treatments to the letters)**

	Column 1	Column 4	Column 3	Column 2
Row 2	T1	T3	T2	T4
Row 4	T2	T1	T4	T3
Row 1	T4	T2	T3	T1
Row 3	T3	T4	T1	T2

If write this in the context of the experiment in our example:

**Table 2.5 The 4 × 4 Latin square used in Example 1**

	Cow 1	Cow 4	Cow 3	Cow 2
Period 2	T1	T3	T2	T4
Period 4	T2	T1	T4	T3
Period 1	T4	T2	T3	T1
Period 3	T3	T4	T1	T2

According to our randomization scheme, we should assign Treatment T4 to Cow 1 during Period 1, followed by Treatment T1 during Period 2, etc.

### 3. Analysis of variance

#### 3.1 Model for the classical Latin square design

We write the model for the classical Latin square design as

$$Y_{ijk} = \mu + \rho_i + \delta_j + \alpha_k + e_{ijk}$$

Where  $y_{ijk}$ ,  $i = 1, \dots, a$ ,  $j = 1, \dots, a$ ,  $k = 1, \dots, a$ , is the observation for the experimental unit in the  $i$ th row block level,  $j$ th column block level and the  $k$ th treatment effect.

Upon choosing one Latin square arrangement at random and running his experiment accordingly, the investigator came up with the following data upon completion of the experiment. The data is given in parenthesis:

**Table 3.1 The 4 × 4 Latin square used in Example 1 together with data**

	Cow 1	Cow 2	Cow 3	Cow 4
Period 1	T4 (192)	T1 (195)	T3 (292)	T2 (249)
Period 2	T1 (190)	T4 (203)	T2 (218)	T3 (210)
Period 3	T3 (214)	T2 (139)	T1 (245)	T4 (163)
Period 3	T2 (221)	T3 (152)	T4 (204)	T1 (134)

### 3.2 Analysis of a Latin square

The analysis of a Latin square is very easy, if there are no empty cells. The least squares means for the treatment effects can be shown to be simple averages as

$$\mu_{..k} = \bar{y}_{..k} = \frac{\sum_{j=1}^a y_{ijk}}{a} = \frac{\sum_{i=1}^a y_{ijk}}{a}$$

Note that either summation on the right side of the equation above is the same. For example, for Treatment T1, summing over levels of Periods:

$$\bar{y}_{..1} = \frac{195 + 190 + 245 + 134}{4} = 191$$

or summing over levels of Cows:

$$\bar{y}_{..1} = \frac{190 + 195 + 245 + 134}{4} = 191$$

Likewise, one can find the following treatment means for the other diets:

$$\bar{y}_{..2} = 206.75 \quad \bar{y}_{..3} = 190.5 \quad \bar{y}_{..4} = 217$$

The factor *SS* are also easy to determine for a Latin square design (with no missing data):

$$SSROW = a \sum_{i=1}^a (\bar{y}_{i..} - \bar{y}_{...})^2 \quad SSCOL = a \sum_{j=1}^a (\bar{y}_{.j.} - \bar{y}_{...})^2$$

$$SSTR = a \sum_{k=1}^a (\bar{y}_{..k} - \bar{y}_{...})^2$$

The generic ANOVA table is as follows:

**Table 3.2 The ANOVA table**

Source	SS	df	MS
Row	SSROW	a-1	MSROW
Column	SSCOL	a-1	MSCOL
Treatments	SSTR	a-1	MSTR
Error	SSE	(a-1)(a-2)	MSE

The ANOVA table for our example is:

**Table 3.3 The ANOVA table used in Example 1**

Source	SS	df	MS	F	Pr > F
Periods	6539.19	3	2179.7	1.76	0.2540
Cows	9929.19	3	3309.7	2.68	0.1409
Diets	1995.69	3	665.23	0.54	0.6736
Error	SSE	6	135.22		

Doing a Tukey's test on pairwise comparisons of the diets, one would come up with the following:

T3	T2	T1	T4
<u>217.00</u>	<u>206.75</u>	<u>191.00</u>	<u>190.50</u>

### 3.3 SAS program and output

The SAS program and output might look as follows:

```

data Latin;
input per $ trt1 $ y1 trt2 $ y2 trt3 $ y3 trt4 $ y4;
cards;
Per1 T4 192 T1 195 T3 292 T2 249
Per2 T1 190 T4 203 T2 218 T3 210
Per3 T3 214 T2 139 T1 245 T4 163
Per4 T2 221 T3 152 T4 204 T1 134
;
data setup(drop=trt1-trt4 y1-y4);
set Latin;
trt = trt1; y = y1; cow=1; output;
trt = trt2; y = y2; cow=2; output;
trt = trt3; y = y3; cow=3; output;
trt = trt4; y = y4; cow=4; output;
run;
proc glm;
class per trt cow;
model y = per trt cow;
means trt /tukey;
run;

```

The GLM Procedure					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	9	18464.06250	2051.56250	1.66	0.2770
Error	6	7423.37500	1237.22917		
Corrected Total	15	25887.43750			
Source	DF	Type I SS	Mean Square	F Value	Pr > F
per	3	6539.187500	2179.729167	1.76	0.2540
trt	3	1995.687500	665.229167	0.54	0.6736

cow	3	9929.187500	3309.729167	2.68	0.1409
Source	DF	Type III SS	Mean Square	F Value	Pr > F
per	3	6539.187500	2179.729167	1.76	0.2540
trt	3	1995.687500	665.229167	0.54	0.6736
cow	3	9929.187500	3309.729167	2.68	0.1409

One could have accounted for alternative variance-covariance structures for the residuals over time within a subject, using the REPEATED option of PROC MIXED. Consider the following two options:

```
proc mixed covtest;
title 'Compound symmetry error structure';
class per trt cow;
model y = per trt /ddfm=kr;
repeated per / subject = cow type = cs;
lsmeans trt /pdiff adjust=tukey;
run;
proc mixed covtest;
title 'Autoregressive error structure';
class per trt cow;
model y = per trt /ddfm=kr;
repeated per / subject = cow type = ar(1);
lsmeans trt /pdiff adjust=tukey;
run;
```

The output is as follows

The Mixed Procedure					
Model Information					
Data Set	WORK.SETUP				
Dependent Variable	y				
Covariance Structure	Compound Symmetry				
Subject Effect	cow				
Estimation Method	REML				
Residual Variance Method	Profile				
Fixed Effects SE Method	Prasad-Rao-Jeske-Kackar-Harville				
Degrees of Freedom Method	Kenward-Roger				
Dimensions					
Covariance Parameters	2				
Columns in X	9				
Columns in Z	0				
Subjects	4				
Max Obs Per Subject	4				
Covariance Parameter Estimates					
Cov Parm	Subject	Estimate	Standard Error	Z Value	Pr > Z
CS	cow	518.13	698.80	0.74	0.4584
Residual		1237.23	714.31	1.73	0.0416
Fit Statistics					
-2 Res Log Likelihood			100.9		
AIC (smaller is better)			104.9		

AICC (smaller is better) 106.9  
 BIC (smaller is better) 103.7

Type 3 Tests of Fixed Effects

Effect	Num DF	Den DF	F Value	Pr > F
per	3	6	1.76	0.2540
trt	3	6	0.54	0.6736

Least Squares Means

Effect	trt	Estimate	Standard Error	DF	t Value	Pr >  t
trt	T1	191.00	20.9485	8.29	9.12	<.0001
trt	T2	206.75	20.9485	8.29	9.87	<.0001
trt	T3	217.00	20.9485	8.29	10.36	<.0001
trt	T4	190.50	20.9485	8.29	9.09	<.0001

Differences of Least Squares Means

Effect	trt	_trt	Estimate	Standard Error	DF	t Value	Pr >  t	Adjustment	Adj P
trt	T1	T2	-15.7500	24.8720	6	-0.63	0.5499	Tukey-Kramer	0.9176
trt	T1	T3	-26.0000	24.8720	6	-1.05	0.3361	Tukey-Kramer	0.7317

Differences of Least Squares Means

Effect	trt	_trt	Estimate	Standard Error	DF	t Value	Pr >  t	Adjustment	Adj P
trt	T1	T4	0.5000	24.8720	6	0.02	0.9846	Tukey-Kramer	1.0000
trt	T2	T3	-10.2500	24.8720	6	-0.41	0.6946	Tukey-Kramer	0.9744
trt	T2	T4	16.2500	24.8720	6	0.65	0.5377	Tukey-Kramer	0.9107
trt	T3	T4	26.5000	24.8720	6	1.07	0.3277	Tukey-Kramer	0.7210

The Mixed Procedure

Model Information

Data Set	WORK. SETUP
Dependent Variable	y
Covariance Structure	Autoregressive
Subject Effect	cow
Estimation Method	REML
Residual Variance Method	Profile
Fixed Effects SE Method	Prasad-Rao-Jeske-Kacker-Harville
Degrees of Freedom Method	Kenward-Roger

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error	Z Value	Pr > Z
AR(1)	cow	0.4418	0.3795	1.16	0.2444
Residual		1819.02	932.60	1.95	0.0256

Fit Statistics

-2 Res Log Likelihood	100.8
AIC (smaller is better)	104.8
AICC (smaller is better)	106.8
BIC (smaller is better)	103.6

Type 3 Tests of Fixed Effects

Effect	Num DF	Den DF	F Value	Pr > F
per	3	4.68	0.95	0.4859
trt	3	4.52	0.29	0.8316

Least Squares Means

Effect	trt	Estimate	Standard Error	DF	t Value	Pr >  t
trt	T1	193.85	21.2314	8.04	9.13	<.0001
trt	T2	197.86	21.2314	8.04	9.32	<.0001
trt	T3	216.40	21.2314	8.04	10.19	<.0001
trt	T4	197.14	21.2314	8.04	9.29	<.0001

Differences of Least Squares Means

Effect	trt	_trt	Estimate	Standard Error	DF	t Value	Pr >  t	Adjustment	Adj P
trt	T1	T2	-4.0149	29.5204	6.06	-0.14	0.8962	Tukey-Kramer	0.9990
trt	T1	T3	-22.5546	29.5204	6.06	-0.76	0.4735	Tukey-Kramer	0.8672

## 4. Latin rectangles and multiple Latin squares

Sometimes, one Latin square is not quite enough. In fact, it is often held that if  $a \leq 4$ , and then more than one square should be considered in a design. On the other hand, when  $a \geq 8$ , a Latin square may be considered to be too large of an experiment. The unwritten rule seemingly implied here is that in the absence of any other knowledge, the error degrees of freedom should be anywhere between 10 and 40 in most designs. Let's consider the following example:

### 4.1 An example of Latin rectangle design

An experiment was designed to determine the effects of three diets on liver cholesterol in rats (A=control, B=control + vegetable fat, C=control + animal fat). Body weight classifications (H, M or L) of the rats and the litters from which they came were used to form a balanced set of Latin squares. The litters were nested in squares (i.e. different litters were used in each square), whereas rows (weight classification) were not nested. The data is reported in the table below.

<i>Square 1</i>			
Weight Class	Litter 1	Litter 2	Litter 3
H	1.60 (B)	1.97 (A)	2.07 (C)
M	1.83 (C)	1.71 (B)	1.56 (A)
L	1.44 (A)	1.84 (C)	1.72 (B)
<i>Square 2</i>			
Weight class	Litter 4	Litter 5	Litter 6
H	1.71 (A)	2.02 (C)	1.85 (B)
M	1.63 (B)	1.75 (A)	2.06 (C)
L	1.70 (C)	1.59 (B)	1.68 (A)
<i>Square 3</i>			
Weight class	Litter 7	Litter 8	Litter 9
H	2.09 (C)	1.83 (B)	1.98 (A)
M	1.63 (A)	1.91 (C)	1.83 (B)
L	1.67 (B)	1.63 (A)	2.00 (C)

This design is sometimes labeled a Latin rectangle as the row identifications (e.g. weight class) are common to all squares but the column identification (e.g. litters) isn't (i.e. different litters in each square). One linear model for the Latin rectangle that has common row blocking criteria for  $s$  complete Latin squares and unique column blocking is:

$$y_{ijk} = \mu + \rho_i + \gamma_j + \tau_k + e_{ijk}$$

where  $\rho_i$  is the fixed effect of the  $i$ th weight class ( $i = 1,2,3$ ),  $\gamma_j$  is the random effect of the  $j$ th litter ( $j = 1,2,\dots,9$ ) with  $\gamma_j \sim NIID(0, \sigma_\gamma^2)$  over all  $j$ , and  $\tau_k$  is the fixed effect of the  $k$ th treatment ( $k = 1,2,3$ ).

One can see that in the data design above, there exist at least two observations for each weight class by treatment (i.e.  $ik$ ) combination. So the above model can be further extended to infer upon weight class by treatment interaction effects.

$$y_{ijk} = \mu + \rho_i + \gamma_j + \tau_k + \rho\tau_{ik} + e_{ijk}$$

where  $\rho\tau_{ik}$  is the interaction between the  $i$ th weight class and the  $k$ th treatment group.

## 4.2 SAS program and output

The SAS program and output might look as follows:

```
proc glm;
class treat litter weight ;
model chol = treat weight treat*weight litter;
lsmeans treat /pdiff adjust=tukey;
random litter;
run;
proc mixed;
class treat litter weight ;
model chol = treat weight treat*weight;
lsmeans treat /pdiff adjust=tukey;
random litter;
run;
```

The resulting output is as follows:

General Linear Models Procedure

Dependent Variable: CHOL

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	14	0.76633333	0.05473810	13.70	0.0001
Error	12	0.04793333	0.00399444		
Corrected Total	26	0.81426667			

  

	R-Square	C.V.	Root MSE	CHOL Mean
	0.941133	3.533010	0.063202	1.788889

  

Source	DF	Type I SS	Mean Square	F Value	Pr > F
TREAT	2	0.33642222	0.16821111	42.11	0.0001
WEIGHT	2	0.19615556	0.09807778	24.55	0.0001
TREAT*WEIGHT	4	0.04188889	0.01047222	2.62	0.0877
LITTER	6	0.19186667	0.03197778	8.01	0.0012

  

Source	DF	Type III SS	Mean Square	F Value	Pr > F
TREAT	2	0.33642222	0.16821111	42.11	0.0001
WEIGHT	2	0.19615556	0.09807778	24.55	0.0001
TREAT*WEIGHT	2	0.02568889	0.01284444	3.22	0.0762
LITTER	6	0.19186667	0.03197778	8.01	0.0012

The MIXED Procedure

Class Level Information

Class	Levels	Values
TREAT	3	(A) (B) (C)
LITTER	9	1 2 3 4 5 6 7 8 9
WEIGHT	3	H L M

  

REML Estimation Iteration History

Iteration	Evaluations	Objective	Criterion
0	1	-49.84228170	
1	1	-59.04298304	0.00000000

Convergence criteria met.

  

Covariance Parameter Estimates (REML)

Cov Parm	Estimate	
LITTER	0.00932778	$\hat{\sigma}_u^2$
Residual	0.00399444	$\hat{\sigma}_e^2$

  

Tests of Fixed Effects

Source	NDF	DDF	Type III F	Pr > F
TREAT	2	12	42.11	0.0001
WEIGHT	2	12	24.55	0.0001
TREAT*WEIGHT	4	12	1.73	0.2070

## 5. Crossover designs:

As with typical Latin square and some repeated measures studies, crossover studies describes those experiments with treatments administered in sequence to each experimental unit. A treatment is administered to a subject for a certain period of time, after which another treatment is administered to the same subject. The treatments are successively administered to the subject until it has received all treatments. The only difference between a crossover design and regular Latin square is that treatment sequence is actually considered to be one of the blocking factors in the study. Consider the following example:

### 5.1 An example of crossover design

**Example2 (from Kuehl, 2000):** Associative or synergistic effects occur in animal diets when feedstuffs are combined and diet utilization or animal performance is different from that predicted from a sum of the individual ingredients. The addition of roughage to the diets of ruminant animals has been shown to influence various diet utilization factors such as ruminal retention time. However, information about the relative associative effects of different roughage was scarce, especially in mixed feedlot diets.

The research hypothesis was that roughage source could influence utilization of mixed diets of beef steers by altering ruminal digestion of other diet ingredients. There were three treatments. All diets were based on 65% concentrate (steamflaked milo base) and 35% roughage. The three roughage treatments were: (A) 35% alfalfa hay (control) (B) 17.5% wheat straw and 17.5% alfalfa (C) 17.5% cottonseed hulls and 17.5 % alfalfa. The experimental design was as follows: 12 beef steers were used in the study; each of three roughage diets were fed to the steers in one of 6 possible sequences of the three diets. Each diet in each sequence was fed to two steers for 30 days. The steers were allowed a period of 21 days (washout period) to adapt to a diet change before any data were collected. The Neutral Detergent Fiber (NDF) digestion coefficient calculated for each steer on each diet is shown below. The NDF digestion coefficient indicates the percent of dietary fiber digested by the steer:

**Table 5.1 Data and crossover design of Example 2**

	Sequence																	
	1			2			3			4			5			6		
Steers		1	2		3	4		5	6		7	8		9	10		11	12
Period I	A	50	55	B	44	51	C	35	41	A	54	58	B	50	55	C	41	46
Period II	B	61	63	C	42	46	A	55	56	C	48	51	A	57	59	B	56	58
Period III	C	53	57	A	57	59	B	47	50	B	51	54	C	51	55	A	58	61

Please note that blocking on steers in this way potentially increases the precision of treatment mean differences so the design is efficient in that sense. Now compare this design to the Latin rectangle that was previously considered. In fact, note above that this design is a clear example of a Latin rectangle, except that the randomization is a little different. One should see that the above design is a balanced row-column design as far as treatments are considered; each treatment is considered 4 times in each period and twice within each sequence. However, this is not just any Latin rectangle. This is a design set up such that each treatment follows every other treatment an equal number of times...i.e. the design is balanced for assessment of carryover effects.

To allow this, one can't have two different squares above that are randomized independently from each other. The data design above is a special case of a crossover study in which we have two steers considered within each sequence (a normal crossover study might consider just one subject within each sequence); i.e. as if we replicated the Latin rectangle twice.

Generally, we might hope that carryover effects are not important but if they potentially are, we need to protect our inferences on treatment effects against them. That is, you should allow for the possibility that the physiological state of a subject may have been altered by one treatment sufficiently to have some effect on the responses in the succeeding treatment period. Sometimes a "washout period" between subsequent treatment administrations on the same experimental unit is helpful to mute carryover effects. Now the balance, as indicated above, only applies to first-order carryover effects (i.e. effect of the treatment applied in the period immediately preceding the current

treatment), hopefully, there are no higher order carryover effects (although there are some more complicated designs that could assess this as well).

## 5.2 Model of crossover design

Let's write the linear model for the crossover design above:

$$y_{ijkl} = \mu + \alpha_i + \rho_{j(i)} + \gamma_k + \tau_l + \lambda_m + \gamma\tau_{kl} + e_{ijkl},$$

where  $\mu$  is the overall mean,  $\alpha_i$  is the effect of the  $i$ th sequence,  $\rho_{j(i)}$  is the random effect of the  $j$ th steer within the  $i$ th sequence (i.e.  $\rho_{j(i)} \sim N(0, \sigma_\rho^2)$ ),  $\gamma_k$  is the effect of the  $k$ th period,  $\tau_l$  is the effect of the  $l$ th treatment effect and  $\lambda_m$  is the effect of the  $m$ th carryover effect with  $e_{ijk}$  being the NIID residual term (i.e.  $e_{ijk} \sim N(0, \sigma_e^2)$ ).

## References:

- Jones, B. and M.G. Kenward. 1989. *Design and analysis of crossover trials*. Chapman and Hall, London.
- Kuehl, R.O. 2000. *Design of experiments: statistical principles in research design and analysis*. Duxbury Press, Pacific Grove, CA.
- Mead, R., R.N. Curnow, and A.M. Hasted. 2003. *Statistical methods in agriculture and experimental biology*. Chapman and Hall/CRC, Boca Raton, FL.
- Williams, E.J. 1949. Experimental designs balanced for the estimation of residual effects of treatments. *Australian Journal of Scientific Research* 2: 149-168.