Topic 12. The Split-plot Design and its Relatives (Part I) [ST&D Ch. 16]

12.1. Definition

The split-plot design results from a specialized randomization scheme for a factorial experiment.

> The basic split-plot design involves assigning the levels of one factor to **main plots** arranged in a CRD, RCBD, or a Latin-Square and then assigning the levels of a second factor to **subplots** within each main plot.

Note that randomization is a **two-stage** process. First, levels of factor A are randomized over the main plots and then levels of factor B are randomized over the subplots within each main plot. Each main plot may be considered as a block as far as factor B is concerned but only as an **incomplete block** as far as the full set of treatments is concerned because not every subplot has the same chance of receiving every treatment combination.

This restriction in randomization results in the presence of **two distinct error terms**, one appropriate for the main plots (i.e. for testing the effect of factor A) and one appropriate for the subplots (i.e. for testing the effect of factor B). Ordinarily, the error term for the main plots is larger than it would be in a complete design since the main plots are larger, further apart, and encompass greater heterogeneity, while the subplot error is smaller than it would be in a complete design. Since the interactions are compared using the smaller subplot error, the precision in estimating interactions is usually increased in a split-plot design relative to a simple factorial.

A typical example of a split-plot design is an irrigation experiment where irrigation levels are applied to large areas, and factors like varieties and fertilizers are assigned to smaller areas within particular irrigation treatments. The proper analysis of a split-plot design recognizes that treatments applied to main plots are subject to larger experimental errors than those applied to subplots; hence, different mean squares are used as denominators for the corresponding *F* ratios. This concept is explored here in terms of expected mean squares.

> In summary, in a split-plot design, the factor assigned to the subplots is the factor that requires smaller amounts of experimental material, that is of primary importance, that is expected to exhibit smaller differences, or for which greater precision is desired.

12.2. Uses of Split-plot designs

1. Split-plot designs (and a common variation, the split-block) are frequently used for factorial experiments in which the nature of the experimental material or the operations involved make it difficult to handle all factor combinations in the same manner. It may be used when the treatment levels associated with one of the factors require larger amounts of experimental material in than do treatment levels for other factors.

2. These designs are also used when the investigator wishes to increase precision in estimating certain effects and is willing to sacrifice precision in estimating other effects. The design usually sacrifices precision in estimating the average effects of the treatments assigned to main plots. It often improves the precision for comparing the average effects of treatments assigned to subplots and, when interactions exist, for detecting those interactions. This arises from the fact that the experimental error for main plots is usually larger than the experimental error used to compare subplot treatments. Usually, the error term for subplot treatments is smaller than would be obtained if treatments were randomly assigned to experimental units as *combinations* of factor levels (a one-stage randomization process).

3. The design may also be useful when an additional factor is to be incorporated into an experiment to increase its scope. For example, suppose that the primary purpose of an experiment is to compare the effects of several seed protectants. To increase the scope of the experiment across a range of varieties, several varieties could be used as main plots and the seed protectants used as subplots.

12.3. The split-plot design

Suppose factor A is the main plot factor, with 3 levels, while factor B is the subplot factor, with 2 levels. There are 4 reps per main plot. We will see how such an experiment could be arranged according to 3 different designs: 1. Factorial (no split), arranged as a CRD; 2. Split-plot, with main plots arranged as a CRD; and 3. Split-plot, with main plots arranged as an RCBD.

12.3.1. Factorial (no split), arranged as a CRD

This is a simple 3x2 factorial arranged as a CRD, like you've seen before. With 6 possible treatment combinations and 4 replications, 24 experimental units (e.g. plots in a field) are required. The six treatment combinations are randomly assigned to the plots in a single randomization process. The resulting field could look like this:

a1b1	a2b2	a2b1	a1b2	a3b2	a1b1	a2b2	a2b1	a1b2	a3b2	a1b1	a3b2
a2b2	a3b1	a1b2	a3b1	a1b2	a3b2	a2b1	a1b1	a2b2	a3b1	a2b1	a3b1

12.3.2. Split-plot, with main plots arranged as a CRD

In this scenario, the randomization process is divided into 2 stages.

Stage 1: Randomize the levels of factor A over the main plots:

a2	a3	a2	a1	a2	a3	a2	a3	a1	a3	a1	a1
a2	a3	a2	a1	a2	a3	a2	a3	a1	a3	a1	a1

Stage 2: Randomize the levels of factor B over the subplots:

a2b2	a3b2	a2b1	a1b1	a2b1	a3b2	a2b1	a3b2	a1b1	a3b1	a1b1	a1b2
a2b1	a3b1	a2b2	a1b2	a2b2	a3b1	a2b2	a3b1	a1b2	a3b2	a1b2	a1b1

12.3.3. Split-plot, with main plots arranged as an RCBD

In this scenario, the randomization process is divided into 2 stages *per block*.

Stage 1: Randomize the levels of factor A within each block.

a2	a1	a3
a2	a1	a3

a1	a2	a3
a1	a2	a3

a1	a3	a2
a1	a3	a2

a3	a2	a1
a3	a2	a1

Stage 2: Randomize the levels of factor B over the subplots.

a2b1	a1b1	a3b2
a2b2	a1b2	a3b1

a1b2	a2b1	a3b2
a1b1	a2b2	a3b1

a1b2	a3b1	a2b2
a1b1	a3b2	a2b1

a3b1	a2b1	a1b1
a3b2	a2b2	a1b2

In split-plot designs, the effect of Factor B (i.e. the difference between b1 and b2 values) is generally more consistent across the experiment due to their proximity to one another within each main plot. Another way of saying this is that there is usually a positive correlation between b1 and b2 values within each main plot. This results in a smaller variance among levels of Factor B than in a normal factorial experiment, thereby increasing the precision with which differences among levels of Factor B are detected.

12.4. Linear models for the split-plot

The linear model for the split-plot, with main plots arranged as a CRD, is:

$$Y_{ijk} = \mu + \tau_{Ai} + (\tau_A{:}\rho)_{ij} + \tau_{Bk} + (\tau_A{:}\tau_B)_{ik} + \varepsilon_{ijk}$$

where

i = 1,...,a	indexes the main plot levels
j = 1,...,r	indexes the replications (ρ = "rho" = rep)
k = 1,...,b	indexes the subplot levels

The variance associated with $(\tau_A{:}\rho)_{ij}$ (i.e. $\sigma^2_{A:Rep}$) is the correct error term for testing the main plot effects. The variance associated with ε_{ijk} (i.e. σ^2_ε) is the correct error term for testing the subplot effects. Of the two, $\sigma^2_{A:Rep}$ is usually larger.

The linear model for the split-plot, with main plots arranged as an RCBD, is:

$$Y_{ijk} = \mu + \tau_{Ai} + \beta_j + (\tau_A{:}\beta)_{ij} + \tau_{Bk} + (\tau_A{:}\tau_B)_{ik} + \varepsilon_{ijk}$$

In this case,

$j = 1,...,r$ indexes the blocks

and the extra term β_j represents the effect of the j^{th} block.

12.5. Split-plot ANOVA

The total degrees of freedom in a split-plot experiment are one less than the total number of subplots. In other words, $df_{Total} = rab - 1$, where r = number of replications (in a CRD) or the number of blocks (RCBD), a = number of main plots, and b = number of subplots per main plot. The main plot (factor A) SS has $df_{MP} = a - 1$ and the subplot (factor B) SS has $df_{SP} = b - 1$.

The main plot error

The appropriate mean square error to test effects of the main plot factor is often called "error A" or MS(MPE) (i.e. mean square of the main plot error). This error is computationally equivalent to the *Main plot x Replication* interaction term in a CRD and to the *Main plot x Block* interaction in a RCBD. This error term is the appropriate error term for testing differences among levels of the main plot factor.

CRD Main plot error = *Main plot x Replication*
RCBD Main plot error = *Main plot x Block*

Why is this the correct error term? From the perspective of the main plot (i.e. Factor A), the subplots are simply subsamples; so it is reasonable to average them when testing the main plot effects.

Consider the case of the CRD. If the values of the subplots within each main plot are averaged, the resulting design is a simple CRD. We've never pointed it out, but what exactly IS the error term of a simple CRD? It's the Treatment:Replication interaction. So, like all CRD's, the appropriate error term is the Treatment:Replication interaction; but in order to use it, we must explicitly extract it from the error term and put it in the model.

Now consider the case of a RCBD with one observation per block-treatment combination. If the values of the subplots within each main plot are averaged, the resulting design is a simple RCBD. Remember that in this case, the appropriate error term is the *Block:Treatment* interaction. Therefore, it makes sense to use this error term in the split-plot to compare the main plot effects.

The subplot error

The appropriate mean square error to test effects of the subplot factor is often called "error B" or MS(SPE) (i.e. mean square of the subplot error). This error is computationally equivalent to the [*Subplot : Replication + Main plot : Subplot : Replication*] in a CRD and to [*Subplot : Block + Main plot : Subplot : Block*] in a RCBD. In either model, this is the *residual error*; in other words, it is the variation that is left after all other factors have been accounted for. This error term is the appropriate error term for testing significance of the subplot effect and the subplot x main plot interaction effect.

CRD Subplot error = *Subplot : Replication + Main plot : Subplot : Replication*
RCBD Subplot error = *Subplot : Block + Main plot : Subplot : Block*

The general ANOVA table for the split-plot CRD:

Source	df	SS	MS	F
Total (subplots)	rab - 1	SS		
Factor A	a - 1	SSA	MSA	MSA/MS(MPE)
Main plot error	a(r - 1)	SS(MPE)	MS(MPE)	
Factor B	b - 1	SSB	MSB	MSB/MS(SPE)
A : B	(a - 1)(b - 1)	SS(A:B)	MS(A:B)	MS(A:B)/MS(SPE)
Subplot error	a(r - 1)(b - 1)	SS(SPE)	MS(SPE)	

The general ANOVA tables for the split-plot RCBD and the split-plot LS are similar to the CRD case and are given in Table 16.1 of ST&D (page 402). These different designs have no effect on the last four rows of the previous table. But the upper lines, corresponding to the main plot effects, do change:

CRD		RCBD		Latin Square	
Total	ra-1	Total	ra-1	Total	ra-1
				Row	**a-1**
		Block	**r-1**	**Column**	**a-1**
A	**a-1**	**A**	**a-1**	**A**	**a-1**
Error A	**a(r-1)**	**Error A**	**(r-1)(a-1)**	**Error A**	**(a-1)(a-2)**
B	b-1	B	b-1	B	b-1
A : B	(a-1)(b-1)	A : B	(a-1)(b-1)	A : B	(a-1)(b-1)
Error B	a(r-1)(b-1)	Error B	a(r-1)(b-1)	Error B	a(r-1)(b-1)
Total	rab-1	Total	rab-1	Total	rab-1

RCBD sample calculation:

$$\begin{aligned}\text{Subplot error df} = \text{Error B df} &= \text{(B:Block + A:B:Block) df} \\ &= (b-1)(r-1) + (b-1)(a-1)(r-1) \\ &= (b-1)(r-1)*[1+(a-1)] = \mathbf{a(b-1)(r-1)}\end{aligned}$$

And the corresponding lm() syntax for these models:

CRD

```
lm( Y ~ A + Rep:A + B + A:B )
```

RCBD

```
lm( Y ~ Block + A + Block:A + B + A:B )
```

Latin Square

```
lm( Y ~ Row + Col + A + Row:Col:A + B + A:B )
```

Replicated Latin Square (shared rows and columns)

```
lm( Y ~ Square + Row + Col + A + Square:Row:Col:A + B + A:B )
```

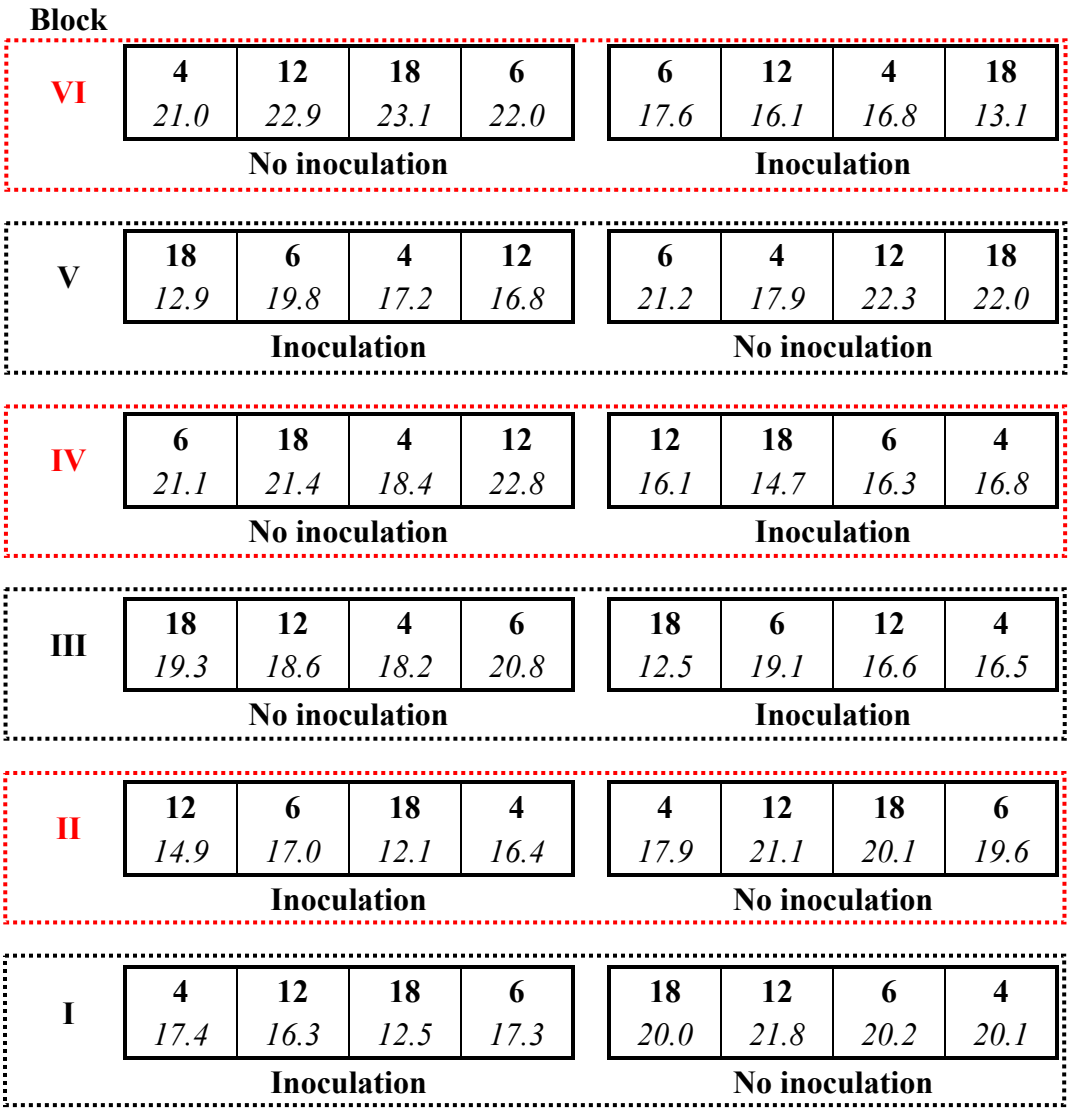
12.6 Example of a split-plot with main plots arranged as an RCBD

To illustrate this design, we will consider an experiment from Thomson *et al.* (Phytopathology **71**: 605-608) carried out to determine the effect of bacterial vascular necrosis on the root yield of sugar beets planted at different in-row spacings. The two factors in the experiment were inoculation (inoculated versus not inoculated with *Erwinia carotovora*) and in-row spacing between plants (4, 6, 12, and 18 inches). The layout of this field experiment is shown on the next page.

Note that in this experiment, the bacterial inoculation levels were applied to large plots (main plot or whole plot) and the spacing levels were assigned to small plots (subplots) within the main plots. There were two reasons for assigning inoculation levels to main plots: 1) To confine the inoculum as well as possible to its assigned plots (i.e. to avoid contaminating non-inoculated plants); and 2) To allocate precision in the experiment to where it is needed most (i.e. while large differences in yield are expected between healthy and diseased plants, relatively smaller differences in yield are expected due to in-row spacing effects).

The two inoculation levels were randomly assigned to the main plots within each of the six blocks. As far as the main plot treatments are concerned, then, this is a simple RCBD. The subplot treatment levels (spacings) were then randomly assigned within each main plot. A separate randomization of subplot levels occurred within *each* main plot.

Figure: Split-plot field layout of the sugar beet root rot study. Each block contains 2 main plots, to which the inoculation treatment levels were assigned (Inoculation, No Inoculation). Each main plot is split into 4 subplots, to which the in-row spacing levels were assigned (4, 6, 12, and 18 inches). The yields of the subplots are shown in italics.



12.6.1 Analysis for this experiment

```
sugar_bad_mod<-lm(yield ~ A_inoc + block + A_inoc:block + B_space +
      A_inoc:B_space, sugar_dat)
anova(sugar_bad_mod)
```

The above command follows the linear model specified on the previous page and therefore seems to be a reasonable approach to analyzing this particular dataset. If you run this code, you obtain the following output:

```
Analysis of Variance Table

Response: yield
               Df  Sum Sq Mean Sq  F value    Pr(>F)
A_inoc          1 256.687 256.687 327.6165 < 2.2e-16 *** <--WRONG!
block           5  16.250   3.250   4.1481  0.005541 **  <--WRONG!

A_inoc:block    5  11.535   2.307   2.9445  0.028023 *   <--MP error

B_space         3  39.638  13.213  16.8634 1.320e-06 ***
A_inoc:B_space  3  64.438  21.479  27.4144 9.838e-09 ***
Residuals      30  23.505   0.783
```

Programmed in this way, all the effects in the model are being tested with the residual error (0.783, df = 30). Unfortunately, this is the incorrect error term for testing the effect of the main plot (A_inoc) and the Block factors, as can be seen in the following table of expected mean squares:

Source	Expected Mean Square
Block	**Var(Error) + 4 Var(Block:A_Inoc)** + 8 Var(Block)
A_Inoc	**Var(Error) + 4 Var(Block:A_Inoc)** + Q(A_Inoc,A_Inoc:B_Space)
Block:A_Inoc	Var(Error) + 4 Var(Block:A_Inoc)
B_Space	**Var(Error)** + Q(B_Space,A_Inoc:B_Space)
A_Inoc:B_Space	**Var(Error)** + Q(A_Inoc:B_Space)

The appropriate error term for A is Block:A
The appropriate error term for B and A:B is the residual error

To obtain the correct F and p-values for the main plot factor and the blocks, you need to tell R to test those factors with the appropriate error term, namely A:Block. An easy way to do this is to use the aov() function:

```
sugar_good_mod<-aov(yield ~ A_inoc + block + Error(A_inoc:block) + B_space +
      A_inoc:B_space, sugar_dat)
summary(sugar_good_mod)
```

The result of this line of code is:

```
Error: A_inoc:block
          Df Sum Sq Mean Sq F value   Pr(>F)
A_inoc     1 256.69  256.69 111.265 0.000132 ***
block      5  16.25    3.25   1.409 0.358023
Residuals  5  11.54    2.31

Error: Within
               Df Sum Sq Mean Sq F value   Pr(>F)
B_space         3  39.64  13.213   16.86 1.32e-06 ***
A_inoc:B_space  3  64.44  21.479   27.41 9.84e-09 ***
Residuals      30  23.50   0.783
```

Using the wrong error term before, we concluded that there were significant yield differences among blocks and highly a significant effect of inoculation (main plot $p < 2.2e\text{-}16$). Now, using the correct error term for these effects, we see that the differences among blocks are not significant and the effect of inoculation is twelve orders of magnitude smaller than we thought ($p = 1.32e\text{-}4$)!

Putting everything together, we arrive at this final ANOVA table for the sugar beet root rot study:

Source	df	SS	MS	F
Total (subplots)	47	412.06		
Block	5	16.26	3.25	1.41 NS
Inoculation (A)	1	256.69	256.69	111.26 ***
Error A (Block:A)	5	11.54	2.31	
Spacing (B)	3	39.64	13.21	16.86 ***
Interaction (A:B)	3	64.44	21.48	27.41 ***
Error B	30	23.50	0.78	

Notice: MSE_A (2.31) > MSE_B (0.78)

Interpretation: Note that the mean square for Error A (2.31) is greater than the mean square for Error B (0.78). The coefficient of variation (CV) for the main plots is 8.3% [(√2.31/18.26) x 100]; for the subplots, it is 4.8% [(√0.78/18.26) x 100]. This is usual for split-plot experiments. In a factorial experiment, the interaction of the treatment factors is usually of primary importance, and this reduced Error-B increases the chance of detecting such interactions. In this particular case, the interaction between inoculation and spacing is highly significant. This indicates that the magnitude of the difference between inoculation treatments depends on in-row spacing, and vice-versa. For example, the difference for the 4 inch spacing between inoculated and non-inoculated plots was 2.1 tons/acre, while an 8 tons/acre difference was observed at the 18-inch spacing. The significant interaction effect implies that the difference between levels of one factor depends on the level of the other factor. For this fixed-effects study, such a result dictates that subsequent analysis be performed on the *simple effects* of each factor.

12.6.4 Mean comparisons

Comparing group means in a split-plot design is more complicated than in those designs which involve only a single error term for all factors. When performing a means separation analysis, there are **four** distinct minimum significant differences that are possible, depending on the nature of the desired comparisons. The possible comparisons:

If the **interaction** between main plot and subplot is **not significant**

12.6.4.1 Comparisons among main plot levels

12.6.4.2 Comparisons among subplot levels

If **interaction** between main plot and subplot is **significant**

12.6.4.3 Comparisons among subplot levels *within* a main plot level

12.6.4.4 Comparisons among main plot levels *within* a subplot level

12.6.4.5 Comparisons among subplot levels across different main plot levels

12.6.4.1 Main plot comparisons in the absence of an interaction

If no significant interaction is detected between main plot and subplot effects, it is valid to compare each factor disregarding the levels of the other factor (i.e. to analyze the main effects). For didactic purposes, we will use the previous example even though a significant interaction was found.

A valid comparison among the means of the main plot levels requires the appropriate error variance ($MS_{Block:A} = 2.307$).

```
MP_comparison<-LSD.test(sugar_dat$yield, sugar_dat$A_inoc, DFerror = 5,
      MSerror = 2.307)
MP_comparison
```

The output:

```
      Mean       CV MSerror      LSD
  18.2625 8.31694    2.307 1.127106

  trt  means M
1   0 20.575 a
2   1 15.950 b
```

Note that the MSE used (2.307) is the **Block:A_Inoc** mean square, as specified. In this particular case, this test is uninformative because there are only two main plots. It is included here only as an example.

Incidentally, just like the means comparison tests, any orthogonal contrasts must also specify the correct error term! Example:

```
# Contrast 'No_Inoc vs. Inoc'       1,-1
contrastmatrix<-cbind(c(1,-1))
contrasts(sugar_dat$A_inoc)<-contrastmatrix

sugar_contrast_mod<-aov(yield ~ A_inoc + block + Error(A_inoc:block) +
      B_space + A_inoc:B_space, sugar_dat)
summary(sugar_contrast_mod, split = list(A_inoc = list("No_inoc vs. Inoc" =
      1)))
```

12.6.4.2 Subplot comparisons in the absence of an interaction

To compare subplots, it is not necessary to specify ERROR B because it is the residual error (the default MSE for all F tests):

```
SP_comparison<-LSD.test(sugar_dat$yield, sugar_dat$B_space, DFerror = 30,
      MSerror = 0.7835)
SP_comparison
```

The output:

```
      Mean        CV MSerror       LSD
  18.2625 4.846847   0.7835 0.738002

  trt    means M
1   6 19.33333 a
2  12 18.85833 a
3   4 17.88333 b
4  18 16.97500 c
```

Note the different MSE used.

12.6.4.3 Comparisons among subplot levels *within* a common main plot level

If the main plot:subplot interaction **is** significant, we are not justified in carrying out the above analyses of main effects. Instead, we are interested in the simple effects of each factor.

When the A:B interaction is significant, the most usual subsequent analysis is that of subplot effects within the different levels of the main plot factor. In the previous example, there is a significant interaction between main plot and subplot effects; so it is appropriate to analyze the simple effects. To analyze the differences among the four spacing treatments *within each inoculation level*, the following code can be used:

```
no_inoc_dat<-subset(sugar_dat, A_inoc == 0)
no_inoc_mod<-lm(yield ~ block + B_space, no_inoc_dat)
MP_comp1<-LSD.test(no_inoc_mod, "B_space")
MP_comp1

inoc_dat<-subset(sugar_dat, A_inoc == 1)
inoc_mod<-lm(yield ~ block + B_space, inoc_dat)
MP_comp2<-LSD.test(inoc_mod, "B_space")
MP_comp2
```

The output:

The ANOVAS for Inoculation = 0 and Inoculation = 1 both showed a significant effect of in-row spacing on yield. The LSD means separations obtained in each case are:

```
A_inoc = 0
alpha: 0.05 ; Df Error: 15
Mean Square Error: 0.8465556
Critical Value of t: 2.13145
Least Significant Difference 1.13225

Groups, Treatments and means
a       12      21.58
a       18      20.98
a       6       20.82
b       4       18.92


A_inoc = 1
alpha: 0.05 ; Df Error: 15
Mean Square Error: 0.7204444
Critical Value of t: 2.13145
Least Significant Difference 1.044515

Groups, Treatments and means
a       6       17.85
ab      4       16.85
b       12      16.13
c       18      12.97
```

Notice that the subplot error used in the original analysis (MSE = 0.7835) is just the average of the MSE's of these two simple effects ANOVAs.

12.6.4.4 Comparisons among main plot levels *within* a common subplot level

Another possible set of comparisons to make is among main plots levels within a common subplot level. The following statements can be added to the previous program to test the differences between the inoculation levels within each spacing level:

```
sp_4_dat<-subset(sugar_dat, B_space == 4)
sp_4_mod<-lm(yield ~ block + A_inoc, sp_4_dat)
SP_comp1<-LSD.test(sp_4_mod, "A_inoc")
SP_comp1

sp_6_dat<-subset(sugar_dat, B_space == 6)
sp_6_mod<-lm(yield ~ block + A_inoc, sp_6_dat)
SP_comp2<-LSD.test(sp_6_mod, "A_inoc")
SP_comp2

sp_12_dat<-subset(sugar_dat, B_space == 12)
sp_12_mod<-lm(yield ~ block + A_inoc, sp_12_dat)
SP_comp3<-LSD.test(sp_12_mod, "A_inoc")
SP_comp3

sp_18_dat<-subset(sugar_dat, B_space == 18)
sp_18_mod<-lm(yield ~ block + A_inoc, sp_18_dat)
SP_comp4<-LSD.test(sp_18_mod, "A_inoc")
SP_comp4
```

Note that the residual error here is automatically the mean square of the Block:A interaction, which is the correct error for main plot comparisons.

12.6.4.5 Mixed Comparisons: Comparisons between subplot levels across different main plot levels

The comparison of subplot means across different main plot levels is more difficult because the comparisons are across two separate levels of the experiment (across subplots and across main plots), each of which has its own appropriate error term. In the case of such mixed comparisons, the accepted protocol is to create an error term (MSE_{Mix}) that is a weighted average of MSE_A and MSE_B, with emphasis on MSE_B. Such comparisons require hand computations.

The appropriate weighted error is:

$$MSE_{Mix} = \frac{(b-1)*MSE_B + MSE_A}{b} = \frac{(4-1)*0.7835 + 2.307}{4} = 1.164375$$

Each of the two error terms in the original analysis (MSEA and MSEB) also each have their critical t values, based on their different degrees of freedom. The accepted protocol for such mixed comparisons is to generate an intermediate *t* value between the *t* value for the main plot

($t_{A,\ 5\ df} = 2.571$) and that for the subplot ($t_{B,\ 30\ df} = 2.042$). The formula to calculate this intermediate *t* value is (ST&D page 404):

$$t_{Mix} = \frac{(b-1) * t_B * MSE_B + t_A * MSE_A}{(b-1)MSE_B + MSE_A} = \frac{3 * 2.042 * 0.78 + 2.571 * 2.31}{3 * 0.78 + 2.31} = 2.305$$

Note that this t_{Mix} value is between t_A and t_B.

With this weighted error term and its associated weighted critical t value, the LSD minimum significant difference can be calculated:

$$LSD_{\alpha=0.05} = t_{Mix}\sqrt{\frac{2MSE_{Mix}}{r}} = 2.305 * \sqrt{\frac{2(1.164375)}{6}} = 1.436$$

If the absolute value of the difference between the means being compared is larger than this critical value, H_0 is rejected (i.e. one concludes that there are significant differences between the subplot means in the different main plot levels).

For example, if we want to compare the mean of inoculated / spacing 4 = 16.85 with the mean of not inoculated / spacing 6 = 20.82:

$$|20.82 - 16.85| = 4.32$$

Since 4.32 > **1.436** $\Rightarrow$ This difference is significant

12.7 Split-split plot design

The concept of the split-plot design is easily extended to three factors. Here, more options present themselves, based on the manner in which these three factors are assigned to the hierarchy of plots:

1. **Split-plot with factorial main plot:** Combinations of levels of Factors A and B are assigned to main plots, levels of Factor C to subplots within each mainplot.
2. **Split-plot with factorial subplot:** Levels of Factor A are assigned to main plots, combinations of levels of Factors B and C are assigned to subplots.
3. **Split-split plot:** Levels of Factor A are assigned to main plots, levels of Factor B to subplots within each mainplot, and levels of Factor C to sub-subplots within each subplot.

The first two designs are the same as the split-plots discussed before, except now the levels of the mainplots or subplots are themselves combinations of two factors. The addition of a third factor by splitting subplots of a split-plot design results in a **split-split plot design** (#3 above). This technique is often quite useful for a three-factor experiment to facilitate field operations or when it is desirable to keep certain treatment combinations together. However, the additional restriction on randomization makes it necessary to compute a **third unique error term** that is

used to test for main effects of the factor applied to the second split and for all interactions involving this factor. So, while the design may have certain advantages in terms of physical operations with the experimental units, the necessity of a third error term can make means separations complicated.

The randomization procedure follows the procedure for the split-plot design. Then, the subplots are split into sub-subplots, equal in number to the levels of the third factor, to which the levels of the third factor are randomly assigned. This operation requires an independent randomization within each subplot.

The following figure from Little & Hills illustrates the layout of a split-split plot to evaluate the effects of dates of planting (A), aphid control (B), and date of harvest (C) on the control of an aphid-borne sugar beet virus. The diagram is presented such that each block shows the results of each stage of the randomization process.

<table>
<tr><th colspan="3">Block I</th><th colspan="9">Block II</th></tr>
<tr><td colspan="3"></td><td colspan="3">A_1</td><td colspan="3">A_3</td><td colspan="3">A_2</td></tr>
<tr><th colspan="3">Block III</th><th colspan="9">Block IV</th></tr>
<tr><td>A_3
B_1</td><td>A_1
B_1</td><td>A_2
B_2</td><td>A_2
B_1
C_1</td><td>A_2
B_1
C_3</td><td>A_2
B_1
C_2</td><td>A_1
B_2
C_3</td><td>A_1
B_2
C_1</td><td>A_1
B_2
C_2</td><td>A_3
B_2
C_1</td><td>A_3
B_2
C_3</td><td>A_3
B_2
C_2</td></tr>
<tr><td>A_3
B_2</td><td>A_1
B_2</td><td>A_2
B_1</td><td>A_2
B_2
C_3</td><td>A_2
B_2
C_2</td><td>A_2
B_2
C_1</td><td>A_1
B_1
C_1</td><td>A_1
B_1
C_3</td><td>A_1
B_1
C_2</td><td>A_3
B_1
C_3</td><td>A_3
B_1
C_1</td><td>A_3
B_1
C_2</td></tr>
</table>

The analysis of variance for the split-split plot design is an extension of the split-plot case. The various error terms are constructed by pooling together different sources of variation.

Level One:
Block
A
Tested using (Block:A) = **Error A**

Level Two:
B
A*B
Tested using (Block:B + Block:A:B) = Block*B(A) = **Error B**

Level Three:
C
A*C
B*C
A*B*C
Tested using (Block:C + Block:A:C + Block:B:C + Block:A:B:C) = residual error
= **Error C**

What this means is that a full analysis requires specifying two special error terms for custom F tests, in addition to the model's residual error. Unfortunately, the aov() function only permits the specification of ONE custom error term. There are several options as to how to proceed. One strategy is to run a couple of different aov() models and combine the results into a final ANOVA table:

Model 1: Testing the main plot effects

```
split_split_MP_mod<-aov(Y ~ A + Block +
                            Error(A:Block) +
                            B + A:B +
                            C + A:C + B:C + A:B:C,
                            split_split_dat)
summary(split_split_MP_mod)
```

The resulting ANOVA table:

```
Error: A:Block
          Df Sum Sq Mean Sq F value  Pr(>F)
A          3 251.49   83.83   12.50 0.00543 **
Block      2 275.43  137.72   20.54 0.00207 **
Residuals  6  40.23    6.70

Error: Within
          Df Sum Sq Mean Sq F value   Pr(>F)
B          1  200.3  200.28  52.369 8.80e-09 ***   ← WRONG!!
C          2  388.1  194.05  50.742 1.06e-11 ***   ← WRONG!!
```

```
A:B          3    2.5    0.83   0.216   0.885   ← WRONG!!
A:C          6   29.4    4.89   1.280   0.288   ← WRONG!!
B:C          2    2.1    1.05   0.275   0.761   ← WRONG!!
A:B:C        6   16.6    2.76   0.722   0.634   ← WRONG!!
Residuals   40  153.0    3.82
```

Model 2: Testing the subplot effects

```
split_split_SP_mod<-aov(Y ~  A + Block +
                             A:Block +
                             B + A:B +
                             Error(A:B:Block) +
                             C + A:C + B:C + A:B:C,
                             split_split_dat)
summary(split_split_SP_mod)
```

The resulting ANOVA table:

```
Error: A:B:Block
          Df Sum Sq Mean Sq F value   Pr(>F)
A          3 251.49   83.83  51.133 1.46e-05 ***  ← WRONG!!
Block      2 275.43  137.72  84.002 4.27e-06 ***  ← WRONG!!
B          1 200.28  200.28 122.162 4.00e-06 ***
A:Block    6  40.23    6.70   4.090   0.0355 *    ← WRONG!!
A:B        3   2.48    0.83   0.503   0.6906
Residuals  8  13.12    1.64

Error: Within
          Df Sum Sq Mean Sq F value   Pr(>F)
C          2  388.1  194.05  44.400 5.88e-10 ***
A:C        6   29.4    4.89   1.120    0.373
B:C        2    2.1    1.05   0.241    0.787
A:B:C      6   16.6    2.76   0.632    0.704
Residuals 32  139.9    4.37
```

Combining the results into a complete table:

```
          Df Sum Sq Mean Sq F value   Pr(>F)
A          3 251.49   83.83   12.50  0.00543 **
Block      2 275.43  137.72   20.54  0.00207 **
 A:Block   6  40.23    6.70                   .
B          1 200.28  200.28 122.162 4.00e-06 ***
A:B        3   2.48    0.83   0.503   0.6906
 A:B:Block 8  13.12    1.64                   .
C          2  388.1  194.05  44.400 5.88e-10 ***
A:C        6   29.4    4.89   1.120    0.373
B:C        2    2.1    1.05   0.241    0.787
A:B:C      6   16.6    2.76   0.632    0.704
 Error    32  139.9    4.37
```

Another way to do it is to obtain the SS for all the factors in the linear model and then carry out the custom F-tests manually. For this, one would specify the full model:

```
split_split_mod<-aov(Y ~  A + Block +
                          A:Block +
                          B + A:B +
                          B:Block + A:Block:B +
                          C + A:C + B:C + A:B:C,
                          split_split_dat)
summary(split_split_mod)
```

The resulting ANOVA table:

	Df	Sum Sq	Mean Sq	F value	Pr(>F)	
A	3	251.5	83.83	19.180	2.66e-07	***
Block	2	275.4	137.72	31.510	2.74e-08	***
B	1	200.3	200.28	45.824	1.19e-07	***
C	2	388.1	194.05	44.400	5.88e-10	***
A:Block	6	40.2	6.70	1.534	0.199	
A:B	3	2.5	0.83	0.189	0.903	
Block:B	2	0.7	0.35	0.079	0.924	
A:C	6	29.4	4.89	1.120	0.373	
B:C	2	2.1	1.05	0.241	0.787	
A:Block:B	6	12.4	2.07	0.474	0.823	
A:B:C	6	16.6	2.76	0.632	0.704	
Residuals	32	139.9	4.37			

To test the main plot effect (A):
F = 83.83/6.70 = 12.50 p(12.50,3,6) = 0.00543

To test the subplot effect (B):
F = 200.28/[(0.7+12.4)/(2+6)] = 122.162 p(122.162,1,8) = 4.00e-06

To test the interaction (A:B):
F = 0.83/[(0.7+12.4)/(2+6)] = 0.507 p(0.507,3,8) = 0.69

A complete analysis of a split-split plot example is discussed in Little and Hills.

12.8 Strip-plot (or split-block) design

> In the strip-plot or split-block design, the subunit treatments are applied in strips across a complete set (replication) of main plot levels.

Below is a comparison of the layouts for a 5x4 split-plot design and a 5x4 strip-plot design (only one replication, or block, is shown). Although the terms main plot and subplot are still used, from a theoretical perspective there is no longer any difference between the two (i.e. they are symmetric; there is no logical hierarchy to them).

Split-plot

A3	**A2**	**A1**	**A5**	**A4**
B2	B1	B2	B3	**B4**
B1	B3	B1	B2	B3
B3	B2	**B4**	**B4**	B1
B4	**B4**	B3	B1	B2

Strip-plot (or split-block)

A3	**A2**	**A1**	**A5**	**A4**
B2	B2	B2	B2	B2
B4	**B4**	**B4**	**B4**	**B4**
B1	B1	B1	B1	B1
B3	B3	B3	B3	B3

Note that the subunit treatments are contiguous across the entire block or main plot, and thus each subunit treatments "split" the block. This design is also called a **strip-plot**, as both A and B treatments are in strips. The A and B treatments are independently randomized within each replication.

12.8.1 Reasons for arranging an experiment as a strip-plot design

1. Physical operations (e.g. tractor manipulation, irrigation, harvesting) may be easier.

2. The design tends to reduce precision in testing the main effects but **improves precision in detecting interaction effects**, which may be the most important objective of the experiment.

12.8.2 Linear model for the strip-plot design

$$Y_{ijk} = \mu + \tau_{Ai} + \tau_{Bj} + \beta_k + (\tau_A{:}\beta)_{ik} + (\tau_B{:}\beta)_{jk} + (\tau_A{:}\tau_B)_{ij} + \varepsilon_{ijk}$$

where

i = 1,...,a indexes the main plot levels
j = 1,...,b indexes the subplot levels
k = 1,...,r indexes the blocks

The extra term $(\tau_B{:}\beta)_{jk}$ represents the interaction effect of subplot levels with blocks. In the split-plot model, this $(\tau_B{:}\beta)_{jk}$ term was not specified; so the variation ascribed to this term was

included in the Subplot error [MS(SPE) = *Subplot : Replication + Main plot : Subplot : Replication*] (see topic 12.5).

12.8.3 ANOVA for the split-block design

In the strip-plot design, the subplot error for testing the main effect of Factor B is MS(StPE) = *Subplot x Block*. This test is symmetric to the test for Factor A, where SS(MPE) = *Main plot x Block* is the denominator of the *F* test. This is a reasonable result, considering that in the split-block design the randomization procedures for both factors are symmetric. Another way to think about this error term is to consider the average of all main plots within each subplot. Averaging in this way results in an RCBD for Factor B with one replication per cell. As we have seen before, the appropriate error term in this case is the MS of *Factor B x Block*.

The general ANOVA table for the RCBD strip-plot design:

Source	df	SS	MS	F
Total	rab - 1	TSS		
Block	r - 1	SS(Block)		
Factor A	a - 1	SSA	MSA	MSA / MS(MPE)
MSE_A= A:Block	(a - 1)(r - 1)	SS(MPE)	MS(MPE)	
Factor B	b - 1	SSB	MSB	MSB / MS(StPE)
MSE_B= B:Block	(b - 1)(r - 1)	SS(StPE)	MS(StPE)	
A x B	(a - 1)(b - 1)	SS(A:B)	MS(A:B)	MS(A:B) / MS(SPE)
MSE_{AB} =A:B:Block	(a-1)(r-1)(b-1)	SS(SPE)	MS(SPE)	

This new error term, the mean square of the strip-plot error MS(StPE), is subtracted from the subplot error (MSE_{AB}), taking (r-1)(b-1) degrees of freedom from that error. The result of this subtraction is a smaller MSE_{AB}, which is the error term used to test the interaction AxB. This results in an **improved precision in the tests for interaction effects.**

12.8.4 Example of a split-block (modified from Little and Hills, Chapter 10)

The following figure gives the layout of an experiment designed to examine the effect of nitrogen fertilizer rate on sugar beet root yield at various harvest times. The main plots are four nitrogen fertilizer rates, arranged as an RCBD with two blocks. The subplot treatment levels are five dates of harvest. The subplots to be harvested at each date span continuous strips across a full set of main plot levels. The harvest date strips, orthogonal to the N fertilizer strips, are also randomized within each of the two blocks. The strip-plot design is helpful here because harvest operations are easier to conduct when the plots to be harvested on a certain date lie along one continuous pass.

The root yield in tons per acre for each subplot are given in the diagram on the next page. Internal dashed lines emphasizing the main plot randomization (Nitrogen levels N0, 80, 160, and 320) are shown in Block I; internal dashed lines emphasizing the split-plot randomization (Harvest levels 1 – 4) are shown in Block II.

	Block I				
	H4	**H5**	**H1**	**H3**	**H2**
N80	26.4	29.3	10.1	23.1	18.2
N320	31.2	34.2	10.3	25.9	19.2
N160	28.0	31.2	10.2	22.3	16.9
N0	10.1	11.4	2.3	9.8	8.8

	Block II				
	H4	**H2**	**H3**	**H5**	**H1**
N160	34.2	18.5	22.4	30.3	10.8
N0	21.3	12.5	16.7	19.1	5.2
N80	29.5	16.9	20.4	26.6	9.5
N320	31.9	17.8	22.8	29.2	7.4

12.8.4.1 R code for a strip-plot (split-block) design

Again, we are in situation where the results from two separate models must be combined to produce a complete and correct ANOVA table.

```
#The ANOVA to test A
split_blockA_mod<-aov(yield ~  A_nitrogen + block +
                               Error(A_nitrogen:block) +
                               B_harvest +
                               B_harvest:block +
                               A_nitrogen:B_harvest,
                               split_block_dat)

#The ANOVA to test B
split_blockB_mod<-aov(yield ~  A_nitrogen + block +
                               A_nitrogen:block +
                               B_harvest +
                               Error(B_harvest:block) +
                               A_nitrogen:B_harvest,
                               split_block_dat)
```

The resulting ANOVA tables:

```
Error: A_nitrogen:block
            Df Sum Sq Mean Sq F value Pr(>F)
A_nitrogen   3  838.3  279.43   7.506  0.066 .
block        1   14.5   14.52   0.390  0.577
Residuals    3  111.7   37.23

Error: Within
                     Df Sum Sq Mean Sq F value   Pr(>F)
B_harvest             4 1898.9   474.7 375.434 1.73e-12 ***
block:B_harvest       4   42.8    10.7   8.459 0.001748 **
A_nitrogen:B_harvest 12  121.0    10.1   7.976 0.000536 ***
Residuals            12   15.2     1.3
```

```
Error: B_harvest:block
          Df Sum Sq Mean Sq F value  Pr(>F)
block      1   14.5    14.5   1.357 0.30872
B_harvest  4 1898.9   474.7  44.382 0.00144 **
Residuals  4   42.8    10.7

Error: Within
                     Df Sum Sq Mean Sq F value   Pr(>F)
A_nitrogen            3  838.3  279.43 220.983 9.19e-11 ***
A_nitrogen:block      3  111.7   37.23  29.441 8.14e-06 ***
A_nitrogen:B_harvest 12  121.0   10.09   7.976 0.000536 ***
Residuals            12   15.2    1.26
```

Combining results into a final ANOVA table:

	Df	Sum Sq	Mean Sq	F value	Pr(>F)	
A_nitrogen	3	838.3	279.43	7.506	0.066	.
A:Block	*3*	*111.7*	*37.23*			
B_harvest	4	1898.9	474.7	44.382	0.00144	**
B:Block	*4*	*42.8*	*10.7*			
A:B	12	121.0	10.1	7.976	0.000536	***
Error	*12*	*15.2*	*1.3*			

Our conclusions? The interaction Nitrogen: Harvest date is highly significant. Therefore, even though no significant differences were detected among Nitrogen levels, we do not accept this result; it is necessary to examine the simple effects. Similarly, even though significant differences were found among harvest dates, we do not accept this result; it is necessary to examine the simple effects.

For the sake of covering one more concept here, let's assume the interaction was found to be non-significant, thereby justifying an analysis of the main effects. In this scenario, notice that the *F* test for the nitrogen levels is *almost* significant. Because four levels of nitrogen were tested, the resulting SS can be partitioned into its linear, quadratic, and cubic components. It is not easy to write a contrast for these effects because the selected levels of nitrogen are not equally spaced. However, as shown in the lab handout for Topics 4&5, this problem can be overcome using a multiple regression approach. In this example, the following simplified program can be used to partition the Nitrogen sum of squares into its three components:

```
#To perform a trend analysis of A_nitrogen using a multiple regression
#approach,first re-load the dataset and maintain A_nitrogen as an
#integer (not a factor); then:
A_nit<-split_block_dat$A_nitrogen
A_nit2<-A_nit^2
```

```
A_nit3<-A_nit^3
A_nit4<-A_nit^4
anova(lm(yield ~ A_nit + A_nit2 + A_nit3 + A_nit4, split_block_dat))
```

The output:

```
Response: yield
          Df  Sum Sq Mean Sq F value  Pr(>F)
A_nit      1  508.21  508.21  8.3005 0.00664 **
A_nit2     1  290.19  290.19  4.7396 0.03611 *
A_nit3     1   39.90   39.90  0.6517 0.42481
Residuals 36 2204.14   61.23
```

Things look good. The addition of the linear component (508.21), the quadratic component (290.19), and the cubic component (39.90) equals the total sum of squares for Nitrogen (838.30) from the previous model. So why is everything crossed out? Because, like before, the appropriate error term for these tests is the Block:A interaction. A_Nitrogen is not a class variable (it is a regression variable), so we have no way of telling R to use the Block:A interaction as an arror term. The result? It must be done by hand.

The following function makes custom F tests relatively easy:

```
customF <- function(x) {
  SS_num=x[1]
  df_num=x[2]
  SS_den=x[3]
  df_den=x[4]
  Fvalue<-(SS_num/df_num)/(SS_den/df_den)
  pFvalue<-pf(Fvalue,df_num,df_den,lower.tail=FALSE)
  print(pFvalue)
}

#To use:
#customF(c(SS_num, df_num, SS_err, df_err))
```

The manually-adjusted ANOVA table, featuring the appropriate F tests, shows a significant ($p < 0.05$) linear effect:

```
Response: yield
          Df  Sum Sq Mean Sq  Pr(>F)
A_nit      1  508.21  508.21   0.034 *
A_nit2     1  290.19  290.19   0.068 .
A_nit3     1   39.90   39.90   0.377 NS
Block:A    3   111.7   37.23
```

As mentioned above, the study of the simple effects of Nitrogen at each Harvest date and the simple effects of Harvest date at each Nitrogen level would be the appropriate continuation of this study. This same partitioning of the Nitrogen SS could be conducted within that simple effects analysis.

What about the correct error term for Block?

Consider the following table of expected mean squares for this split-block RCBD:

Source	Expected Mean Square
Block	Var(Err) + **4 Var(Block*B)** + **5 Var(Block*A)** + 20 Var(Block)
Nitrogen_A	**Var(Err) + 5 Var(Block*A)** + Fixed_effect(A)
Block*Nitrogen_A	**Var(Err) + 5 Var(Block*A)**
Harvest_B	**Var(Err) + 4 Var(Block*B)** + Fixed_effect(B)
Block*Harvest_B	**Var(Err) + 4 Var(Block*B)**
Nitrogen_A*Harvest_B	Var(Err) + Fixed_effect(A*B)

Block:A is the correct error for A
Block:B is the correct error for B
A synthetic F test is required for Block