The Pennsylvania State University
The Graduate School
Eberly College of Science

INTRABLOCK, INTERBLOCK AND COMBINED ESTIMATES IN INCOMPLETE BLOCK DESIGNS: A NUMERICAL STUDY

A Thesis in

Statistics

by

Yasin Altinisik

Submitted in Partial Fullfillment

of the requirements

for the Degree of

Master of Science

August 2013

The thesis of Yasin Altinisik was rewieved and approved* by the following:

James L. Rosenberger
Professor of Statistics
Thesis Advisor

Debashis Ghosh
Professor of Statistics

David Hunter
Professor of Statistics
Head of Department of Statistics

*Signatures are on the file in the Graduate School.

Abstract

Intrablock analysis and interblock analysis are used to estimate true treatment means in block designs. The estimates after using these analyses are called intrablock estimates and interblock estimates respectively. These estimates are unbiased and they are independent from each other. However, a linear combination of these estimates, which are called combined estimates, can also be used to estimate true treatment means. Combined estimates are also unbiased and have better precision estimating treatment contrasts. The SAS PROC MIXED procedure uses matrix notation to obtain combined estimates. First we focus on understanding the matrix notation that SAS PROC MIXED uses in complete and incomplete block designs. We show that the LMER function in R uses the same matrix notation as SAS PROC MIXED, and obtains exactly the same results that SAS PROC MIXED gives with respect to covariance parameter estimates, treatment mean estimates and treatment mean standard error estimates. Second, we focus on a multiple imputation method to deal with missing values in incomplete block designs to make the data complete. An R package called Amelia2 is used for this purpose. Treatment mean estimates are also obtained with SAS PROC MIXED after using the Amelia2 procedure. Afterwards, combined and amelia treatment mean estimates of true treatment means are compared in some balanced incomplete block designs by using a simulation study.

Contents

List of Figures

List of Tables

Acknowledgements

I am very appreciative of the contribution of my advisor, Dr. James L. Rosenberger whose guidance helped me a lot throughout the process of my research. Furthermore I would also like to thank Andrea Berger for her undying support to assemble this report. Most importantly, I appreciate the support given by my parents which plays a very important role in where I am today.

1 Introduction to the Problem

1.1 Randomized Complete Block Designs(RCBD)

The RCBD is one of the best known experimental designs and was first used for agricultural experiments. In the RCBD, there is one factor (treatments) which is our main interest. Here, the reason to use the word "complete" is to indicate that all treatments appear in each block. The strategy of creating a RCBD is to obtain homogenous blocks to eliminate the effect of nuisance factors on measured results.

At this point, a nuisance factor is a factor that affects our model fit, even if it is not our primary interest. There are three possible types of nuisance factors: unknown and uncontrollable, known but uncontrollable, and known and controllable. In the case of known and controllable nuisance factors, blocking techniques can be used to remove the variation due to these factors. When a blocking technique is used, experimental error reflects both variability between blocks and random error. Therefore, the variability between blocks can be removed from the experimental error to compare treatment means better. In other words, by using a blocking technique the effect of known and controllable nuisance factors are removed from the model and treatment means are compared with less uncertainty. The design that accomplishes this is called a randomized complete block design (RCBD) (Montgomery, 2005).

Table 1.1 shows a data example of RCBD with 4 treatments and 4 blocks.

Table 1.1: Data Example of Randomized Complete Block Design (RCBD)

	Block				
Treatment	1	2	3	4	$y_{i.}$
1	35	30	54	89	208
2	42	13	33	65	153
3	56	79	43	21	199
4	77	67	21	44	209
$y_{.j}$	210	189	151	219	$y_{..}$=769

In this study, treatments are the levels of the fixed effect factor which implies that they are measured without error. The purpose of having fixed effects in the model is to make inferences about the specific levels of the fixed effect factor used in the study. On the other hand, blocks are usually random controllable nuisance factors and their levels are a sample from a larger population of possible levels. Because blocks are chosen at random, they are considered to be representative of a larger population of blocks. The models that use both fixed and random effects are called "mixed models." Two examples of RCBDs with fixed treatment effects and random block effects are shown below.

Example 1: "In a study of the effectiveness of four different dosages of a drug, 20 litters of mice, each consisting of four mice, were utilized. The 20 litters (blocks) here may be viewed as a random sample from the population of all litters that could have been used for the study" (Kutner et. al, 2005, p.1060).

Here four different dosage levels of the drug are considered to be fixed effects because we are interested in making inferences about these specific dosage levels.

Example 2: "A researcher investigated the improvement in learning in third-grade classes by augmenting the teacher with one or two teaching assistants. Ten schools were selected at random, and three third-grade classes in each school were utilized

in the study. In each school, one class was randomly chosen to have no teaching assistant, one class was randomly chosen to have one teaching assistant, and the third class was assigned two teaching assistants" (Kutner et. al, 2005, p.1060).

Here the schools can be viewed as blocks because the schools in the study are random sample from the population of all schools suitable for the study.

In this study, the mixed model for each observation is:

$$y_{ij} = \mu_{..} + t_i + \beta_j + \epsilon_{ij} \begin{cases} i = 1, 2, ...a \\ j = 1, 2, ...b \end{cases} \tag{1.1}$$

Here, y_{ij} represents the response variable for the ith treatment and jth block, $\mu_{..}$ represents the mean of all observations, t_i represents the ith fixed treatment effect, β_j represents the jth random block effect and, as always, ϵ_{ij} values are the residuals for the model. Moreover, β_j are independent $N(0, \sigma^2_\beta)$ and t_i are constants with the restriction $\sum t_i = 0$. ϵ_{ij} values are independent $N(0, \sigma^2)$ and independent of the block effects.

Some important properties of the RCBD model (without interaction effect) with fixed treatment effects and random block effects are shown below as presented in Kutner et. al (2005) with changed letter format.

$$\text{E}(Y_{ij}) = \mu_{..} + t_i$$

$$\text{Var}(Y_{ij}) = \sigma^2_\beta + \sigma^2$$

$$\text{Cov}(Y_{ij}, Y_{ij'}) = \sigma^2_\beta \qquad j \neq j'$$

$$\text{Cov}(Y_{ij}, Y_{i'j'}) = 0 \qquad i \neq i'$$

Note, the model that is used in our study is additive. In other words, there is no interaction effect between treatments and blocks in our model (Kutner et. al, 2005).

1.2 Balanced Incomplete Block Designs(BIBD)

As shown in Table 1.1, all treatments appear in each block. However, sometimes it may not be possible to observe all of the treatments in each block which means that our design is incomplete. If the incompleteness of the observations are based on some straightforward rules, the design used for the data is called BIBD. In BIBDs, it is possible to derive formulas to find the parameter estimates, since each treatment pair is included within blocks the same number of times. These formulas are explained in subsections 2.1, 2.2 and 2.3. Table 1.2 shows a data example of a BIBD with 4 treatments and 6 blocks.

Table 1.2: Data Example of Balanced Incomplete Block Design (BIBD)

	Block						
Treatment	1	2	3	4	5	6	$y_{i.}$
1	—	—	—	48	35	38	121
2	—	50	46	—	37	—	133
3	40	—	51	58	—	—	149
4	49	58	—	—	—	54	161
$y_{.j}$	89	108	97	106	72	92	$y_{..}$=564

Here, the number of times treatment i and treatment $i^{'}$ occur together within the same blocks can be defined as $\lambda_{ii'}$ where $i \neq i^{'}$. Designs in which all possible values of $\lambda_{ii'}$s are equal to each other are known as BIBDs. As seen in Table 1.2, all $\lambda_{ii'}$ values are equal to 1 which shows that our design is balanced.

If there are a treatments, b blocks, k treatments in each block and r replicates

of each treatment, the total number of observations is $N = ar = bk$. With these assumptions a BIBD might be constructed by taking $\binom{a}{k}$ blocks and choosing a different combination of treatments to them (Montgomery, 2005).

As a result, the number of times each treatment pair is included in the same block is defined as:

$$\lambda = \frac{r(k-1)}{a-1} \tag{1.2}$$

Here, the values λ, a, b, r and k are dependent on each other and all of them are integers. Based on the data in Table 1.2, these values are: $\lambda = 1$, $a = 4$, $b = 6$, $r = 3$ and $k = 2$. Note that, having all these values integer does not mean that a BIBD is obtained. As mentioned, to have a BIBD the number of times treatment pairs included within the same blocks must be the same. Appendix A shows the code used to construct balanced incomplete block desigs with different treatment and block numbers.

1.3 Unbalanced Incomplete Block Designs(UIBD)

Sometimes in incomplete block designs it is not possible to construct a design that is balanced. In these cases, if it is not possible to construct a Balanced Incomplete Block Design, an UIB design is needed instead. An unbalanced data example are shown in Table 1.3.

Table 1.3: Data Example of Unbalanced Incomplete Block Design (UIBD)

	Block						
Treatment	1	2	3	4	5	6	$y_{i.}$
1	76	—	—	—	75	74	225
2	—	76	78	85	75	77	391
3	5	78	73	—	—	82	238
4	—	—	70	77	85	79	311
$y_{.j}$	81	154	221	162	235	312	$y_{..}$=1165

As seen in Table 1.3, each block does not contain the same number k treatments and each treatment is not replicated r times in the design. Therefore, based on the equation in (1.2), the value of λ cannot be calculated. However, the value of $\lambda_{ii'}$ which shows the number of times each treatment pair is included in the same block can be calculated. As a result, instead of using the formulas, matrix notation is used to find the parameter estimates. These matrix notations are explained in section 3. Matrix notations can also be used in BIBD cases, but since straightforward formulas exist these are often used instead.

On the other hand, for the RCBDs, only exact analyses can be used to estimate treatment means, since there are no missing observations in the data. However, for the BIBDs and UIBDs, beside using an exact analysis, a multiple imputation procedure can also be used. One of the fundamental purposes of this study is to see whether exact analysis with SAS PROC MIXED is the preferred method or if a multiple imputation procedure provides a better understanding of the data with respect to treatment means. To explore this, estimates of treatment means from the exact analysis and using multiple imputation procedures will be compared in some BIBDs.

Moreover, in the exact analysis, intrablock estimates can be used to compare

treatment means. However, it might be better to do interblock analysis to explore whether additional information can be obtained to compare treatment means in a better way. The estimates that are obtained after doing interblock analysis are called interblock estimates. After doing interblock analysis the combination of intrablock and interblock estimates can be used which are called combined estimates. In this study, these three types of estimators are used to show whether interblock analysis really provides some additional information about the differences between the treatments for the BIBD data in Table 1.2.

In addition, SAS PROC MIXED uses matrix notation to find combined estimates. The matrix notation for the RCBDs can be found in Littell et al. (1996). However, in the literature there are no clear expressions that show the matrix notation for incomplete block designs. As a result, we do not know how SAS PROC MIXED uses matrix notation to find combined estimates in BIBDs and UIBDs. Therefore, another purpose of this study is to derive matrix notation that SAS PROC MIXED uses for incomplete block designs. The matrix notation for incomplete block designs can be obtained, using the matrix notation for RCBDs in Littell et al. (1996).

Note that, missing observations in the data are estimated in all multiple imputation procedures. In this study, the "Amelia" procedure in R is used to estimate these missing observations. "Amelia" uses some multiple imputation techniques to estimate missing values. One of the assumptions of "Amelia" is that data are normally distributed. Therefore, in this study, we compare the performance of "Amelia" for some normally distributed data.

2 Recovery of Information in the BIBDs

One of the main purposes of this study is to recover as much information as possible from the data to compare treatment means in the most efficient way. There are three principle estimates used to find treatment means: intrablock estimates, interblock estimates and combined estimates. We give the straightforward formulas in the subsections 2.1, 2.2 and 2.3 to find intrablock estimates, interblock estimates and combined estimates respectively. These formulas work well when our design is BIBD. However, when we have Unbalanced Incomplete Block Design(UIBD) these formulas give poor estimates. In these cases, matrix notations are needed to find the estimates.

2.1 Recovery of Intrablock Information in the BIBDs

"The intrablock estimates are derived from treatment contrasts obtained within blocks"(Fraser, 1957, p.814). To find intrablock estimates in the BIBDs, it is always useful to create an Analysis of Variance first.

To create the Analysis of Variance, the first step is to find total variability in the data. Generally, the total variability in the data is defined as:

$$SS_T = \sum_i \sum_j y_{ij}^2 - \frac{y_{..}^2}{N} \tag{2.1}$$

For the second step, total variability is allocated into three parts. These parts are variability in blocks, variability in treatments and the variability between the data

and the regression model used. Therefore, the partition may be defined as:

$$SS_T = SS_{Treatments(adjusted)} + SS_{Blocks} + SS_E \tag{2.2}$$

In this case, treatments are adjusted for blocks so that we can distinguish the effects of treatments and blocks, where all the differences in blocks are eliminated. Because the differences in blocks are eliminated, this partition is appropriate when the blocks are either random or fixed. The adjusted treatment sum of squares and the blocks sum of squares are defined by the equations in (2.3) and in (2.4) respectively as:

$$SS_{Treatment(adjusted)} = \frac{k\sum_{i=1}^{a} Q_i^2}{\lambda a} \tag{2.3}$$

$$SS_{Blocks} = \frac{1}{k}\sum_{j=1}^{b} y_{.j}^2 - \frac{y_{..}^2}{N} \tag{2.4}$$

Note, the value of λ in equation (1.2) is used to find the variability in the treatments which are adjusted for blocks. Here, Q_i stated in (2.3) is called the adjusted total of the ith treatment. The adjusted total for the ith treatment is computed as:

$$Q_i = y_{i.} - \frac{1}{k}\sum_{j=1}^{b} n_{ij} y_{.j} \text{ where i=1,2,...a} \tag{2.5}$$

with n_{ij}= 1 if treatment i appears in block j and n_{ij}= 0 otherwise (Montgomery, 2005).

Table 2.1 shows the Analysis of Variance for Intrablock Analysis of BIBDs as presented in Toutenburg and Shalabh (2009) with changed letter format.

Table 2.1: Intrablock Analysis of Variance of BIBDs

Source	SS	df	MS	F
Between treatments(adj)	$\frac{k}{\lambda a}\sum_{i=1}^{a} Q_i^2$	a-1	$\frac{SS_{Treat(adj)}}{df_{Treat}}$	$\frac{MS_{Treat(adj)}}{MS_E}$
Between blocks(unadj)	$\frac{1}{k}\sum_{j=1}^{b} y_{.j}^2 - \frac{y_{..}^2}{N}$	b-1	$\frac{SS_{Block(unadj)}}{df_{Block}}$	
Intrablock Error	SS_{Error} by substraction	bk-a-b+1	$\frac{SS_{Error}}{df_{Er}}$	
Total	$\sum\sum y_{ij}^2 - \frac{y_{..}^2}{N}$	bk-1		

There are some computer packages that give the intrablock Analysis of Variance. Minitab is a widely used statistics package used to find the Analysis of Variance for the intrablock analysis. In the Analysis of Variance, Minitab shows the adjusted treatment sum of squares and the adjusted block sum of squares. One can also get Minitab to find the sequential treatment sum of squares and the sequential block sum of squares by clicking on "options" in the GLM command and choosing "use sequential sum of squares." However, the default option for Minitab is the adjusted sum of squares. Table 2.2 shows the output for adjusted sum of squares and sequential sum of squares for the data in Table 1.2.

Table 2.2: Minitab Output for ANOVA Table for the Data in Table 1.2

Source	DF	Seq SS	Adj SS	Adj MS	F	P
Treatments	3	309.333	253.750	84.583	22.56	0.015
Blocks	5	375.417	375.417	75.083	20.02	0.016
Error	3	11.250	11.250	3.750		
Total	11	696.000				

On the other hand, to find intrablock estimates, the first thing that we need is to find the least squares normal equations.

Rao (1997) mentions that these equations are obtained by the minimization of $\emptyset$ which is here redefined based on a model which is similar to the model in (1.1). The only difference between our new model and the model in (1.1) is that, in our new model both treatment effects and block effects are considered to be fixed effects.

$$\emptyset = \sum_{ij} n_{ij}(y_{ij} - \mu - t_i - \beta_j)^2 \tag{2.6}$$

Here, we have three sets of parameters which are μ, t_i and β_j. Therefore, we will have 3 equations corresponding to these parameter sets. After the minimization of the equation in (2.6), the least squares estimates of normal equations for intrablock estimates are obtained by solving the equations in (2.7), (2.8) and (2.9).

$$ak\hat{\mu} + r\sum_i \hat{t_i} + k\sum_j \hat{\beta}_j = y_{..} \tag{2.7}$$

$$r\hat{\mu} + r\hat{t_i} + \sum_i n_{ij}\hat{\beta}_j = y_{i.} \text{ where } i = 1, 2, ...a \tag{2.8}$$

$$k\hat{\mu} + \sum_i n_{ij}\hat{t_i} + k\hat{\beta}_j = y_{.j} \text{where } j = 1, 2, ...b \tag{2.9}$$

Treatment and block effects can be uniquely found by using the restrictions $\sum_i t_i = 0$ and $\sum_j \beta_j = 0$ (Rao,1997).

However, because our interest is only to find the estimate of treatment effects, the least square estimators of treatment effects are shown below as:

$$\hat{t_i} = \frac{kQ_i}{\lambda a} \qquad i = 1, 2, ...a \tag{2.10}$$

where Q_i is expressed with the same way in (2.5). Minitab gives the results of the intrablock information for the data in Table 1.2 as presented in Table 2.3.

Table 2.3: Intrablock Estimates and Treatment Means for the Data in Table 1.2

Parameter	Intrablock Estimate	Treatment Means
t_1	$\hat{t}_1 = -7.00$	$\hat{\mu}_1 = 40.00$
t_2	$\hat{t}_2 = -2.75$	$\hat{\mu}_2 = 44.25$
t_3	$\hat{t}_3 = 1.50$	$\hat{\mu}_3 = 48.50$
t_4	$\hat{t}_4 = 8.25$	$\hat{\mu}_4 = 55.25$

2.2 Recovery of Interblock Information in the BIBDs

In the intrablock analysis, we can treat block effects as random or fixed variables. Considering block effects as random or fixed does not make any difference in the intrablock analysis, since all the differences in blocks are already eliminated. Now let us assume that the differences in blocks are not eliminated and the block effects are all iid (independent and identically distributed) random variables."Since, the treatments in different blocks are not all the same, so the difference between block totals is expected to provide some information about the differences between the treatments"(Toutenburg and Shalabh, 2009, p.200). This analysis method is called "Interblock Analysis" and, therefore, the estimates obtained by the interblock analysis are called "Interblock Estimates."

Recovery of interblock analysis does not guarantee that additional information will be obtained. However, since the analysis does not require a lot of work and an efficiency gain is obtained before the variance analysis is implemented, it is always a good choice to use this analysis method in all cases (Yates, 1940).

In order to find the interblock estimates, again corresponding least squares normal equations are needed.

Rao (1997) uses $\emptyset$ which is here redefined based on the block totals $y_{.j}$ and the model in (1.1) as:

$$\emptyset = \sum_{j}(y_{.j} - k\mu - \sum_{i} n_{ij}t_i)^2 \tag{2.11}$$

As seen above, we have two parameter sets which are μ and t_i. So, we will have two equation sets. After the minimization of $\emptyset$, the least squares estimates of the treatment effects which are obtained by solving the normal equations for interblock estimates:

$$ak\tilde{\mu} + r\sum_{i}\tilde{t_i} = y_{..} \tag{2.12}$$

$$kr\tilde{\mu} + \sum_{j}(n_{ij}\sum_{i} n_{ij}\tilde{t_i}) = \sum_{j} n_{ij}y_{.j} \tag{2.13}$$

Because of the fact that we do not have block parameters, we only need the restriction $\sum_i t_i = 0$ (Rao,1997).

After imposing this restriction, the least square estimators of the interblock estimates are shown below in (2.14) as:

$$\tilde{t_i} = \frac{\sum_{j=1}^{b} n_{ij}y_{.j} - kr\bar{y_{..}}}{r - \lambda} \tag{2.14}$$

Interblock estimates of the treatments for the data in Table 2.1 are presented in Table 2.4.

Table 2.4: Interblock Estimates and Treatment Means for the Data in Table 1.2

Parameter	Interblock Estimate	Treatment Means
t_1	$\tilde{t_1} = -6.00$	$\tilde{\mu}_1 = 41.00$
t_2	$\tilde{t_2} = -2.50$	$\tilde{\mu}_2 = 44.50$
t_3	$\tilde{t_3} = -19.00$	$\tilde{\mu}_3 = 28.00$
t_4	$\tilde{t_4} = 3.50$	$\tilde{\mu}_4 = 50.50$

2.3 Combined Estimates in the BIBDs

Yates (1940) first showed both intrablock and interblock estimates are unbiased and they are independent from each other. He also linearly combined these two estimates to get new estimates which are also unbiased. These new estimates are called "Combined Estimates." By using combined estimates he obtained a better precison estimating the treatment contrasts compared to the precison using intrablock estimates (Kubokawa, 1988).

Montgomery (2005) states that the linear combination of intrablock and interblock estimates as:

$$t_i^* = \alpha_1 \hat{t_i} + \alpha_2 \tilde{t_i} \tag{2.15}$$

where $\alpha_1 = \frac{u_1}{u_1+u_2}$, $\alpha_2 = \frac{u_2}{u_1+u_2}$, $u_1 = \frac{1}{\text{Var}(\hat{t}_i)}$, $u_2 = \frac{1}{\text{Var}(\tilde{t}_i)}$, $V(\hat{t_i}) = \frac{k(a-1)}{\lambda a^2}\sigma^2$ and $V(\tilde{t_i}) = \frac{k(a-1)}{a(r-\lambda)}(\sigma^2 + k\sigma_\beta^2)$.

Based on the equation (2.15), the combined estimator can be defined as:

$$t_i^* = \frac{kQ_i(\sigma^2 + k\sigma_\beta^2) + (\sum_{j=1}^{b} n_{ij} y_{.j} - kr\bar{y}_{..})\sigma^2}{(r-\lambda)\sigma^2 + \lambda a(\sigma^2 + k\sigma_\beta^2)} \tag{2.16}$$

In (2.15), α_1 and α_2 correspond to the weights of the intrablock and interblock estimates respectively. Note that, the variances σ^2 and σ_β^2 cannot be used, since they

are unknown. Therefore, the estimates of these variances need to be used. The estimate of σ^2 is the MSE of intrablock analysis. On the other hand, the estimate of block variance σ_β^2 is MS(Block(adjusted)). The mean square of adjusted blocks in BIBD is defined as:

$$MS_{Blocks(adjusted)} = \frac{(\frac{k\sum_{i=1}^{a} Q_i^2}{\lambda a} + \sum_{j=1}^{b} \frac{y_{.j}^2}{k} - \sum_{i=1}^{a} \frac{y_{i.}^2}{r}}{b-1} \tag{2.17}$$

$MS_{Blocks(adjusted)}$ is redefined based on the expected value of it as:

$$\hat{\sigma_\beta^2} = \begin{cases} \frac{[MS_{Block(adjusted)} - MSE](b-1)}{a(r-1)} & , MS_{Blocks(adjusted)} > MSE \quad (2.18a) \\ 0 & , MS_{Blocks(adjusted)} \leq MSE \quad (2.18b) \end{cases}$$

As a result, the final combined estimates becomes:

$$t_i^* = \begin{cases} \frac{kQ_i(\hat{\sigma}^2 + k\hat{\sigma}_\beta^2) + (\sum_{j=1}^{b} n_{ij} y_{.j} - kr\bar{y}_{..})\hat{\sigma}^2}{(r-\lambda)\hat{\sigma}^2 + \lambda a(\hat{\sigma}^2 + k\hat{\sigma}_\beta^2)} & , \hat{\sigma}_\beta^2 > 0 \quad (2.19a) \\ \frac{y_{i.} - \frac{1}{a} y_{..}}{r} & , \hat{\sigma}_\beta^2 = 0 \quad (2.19b) \end{cases}$$

Therefore, first $\hat{\sigma}^2$ and $\hat{\sigma}_\beta^2$ are obtained. Afterwards, based on their values, the equation (2.19a) or (2.19b) is used (Montgomery, 2005).

Combined estimates of the treatments for the data in Table 1.2 are presented in Table 2.5.

Table 2.5: Combined Estimates and Treatment Means for the Data in Table 1.2

Parameter	Combined Estimate	Treatment Means
t_1	$t_1^* = -6.98$	$\mu_1^* = 40.02$
t_2	$t_2^* = -2.74$	$\mu_2^* = 44.26$
t_3	$t_3^* = 1.58$	$\mu_3^* = 48.58$
t_4	$t_4^* = 8.14$	$\mu_4^* = 55.14$

3 Complete and Incomplete Block Designs in Matrix Notations

As mentioned before, the combined estimates of the BIBD cases can be obtained by using formulas. However, these formulas are not available and are hard to apply towards UIBDs. Therefore, instead of using the formulas, matrix notation is needed in UIBDs. The matrix notation can be used for the BIBDs as well. The matrix notation for the BIBDs and UIBDs is exactly the same; however, for randomized complete block designs, matrix notation is slightly different when compared to incomplete block designs. In the subsections 3.1 and 3.2, the matrix notation is explained for the complete block designs and incomplete block designs respectively. For the complete block design case, the information in Littell et al. (1996) is used as a reference and the data in Table 1.1 are used as an example. On the other hand, for the incomplete block design case, the matrix notation is derived and the unbalanced incomplete data in Table 1.3 are used as an example. Note that, as Littell et al. (1996) explains, "Recall that the usual objectives of a randomized block design are to estimate and compare treatment means using statistical inference. Mathematical expressions are needed for the variances of means and differences between means in order to construct confidence intervals and conduct test of hypothesis"(Littell et al., 1996, p.2). Therefore, the matrix

notation is only for estimating treatment means. In order to estimate pairwise treatment contrasts additional matrix notation is needed. However, this matrix notation is out of the scope in this study. Another important point is that the matrix notation in this section is related to SAS PROC MIXED procedure and it is called generalized linear least squares matrix notation.

3.1 Complete Block Designs in Matrix Notations

To explain matrix notation in BIBDs and UIBDs in a better way, the matrix notation for the RCBDs is examined first. Now, let us rewrite the model in (1.1) with a slightly different letter format in which t corresponds to treatment number and r corresponds to block number.

$$y_{ij} = \mu + t_i + \beta_j + \epsilon_{ij} \begin{cases} i = 1, 2, ...t \\ j = 1, 2, ...r \end{cases} \tag{3.1}$$

In Littell et al. (1996) matrix notation for the complete block designs are defined in clear detail as:

$$\begin{bmatrix} Y_{11} \\ \cdot \\ \cdot \\ \cdot \\ Y_{t1} \\ \cdot \\ \cdot \\ \cdot \\ Y_{1r} \\ \cdot \\ \cdot \\ \cdot \\ Y_{tr} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & 0 & \cdot & \cdot & 1 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & 1 & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & 0 & \cdot & \cdot & 1 \end{bmatrix} \begin{bmatrix} \mu \\ t_1 \\ \cdot \\ \cdot \\ \cdot \\ t_t \end{bmatrix} + \begin{bmatrix} 1 & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & 1 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & 1 \end{bmatrix} \begin{bmatrix} b_1 \\ \cdot \\ \cdot \\ \cdot \\ b_r \end{bmatrix} + \begin{bmatrix} e_{11} \\ \cdot \\ \cdot \\ \cdot \\ e_{t1} \\ \cdot \\ \cdot \\ \cdot \\ e_{1r} \\ \cdot \\ \cdot \\ \cdot \\ e_{tr} \end{bmatrix} \tag{3.2}$$

Based on the data in Table 1.1, the matrix notation in clear detail is shown as:

$$\begin{bmatrix} Y_{11} \\ Y_{21} \\ Y_{31} \\ Y_{41} \\ Y_{12} \\ Y_{22} \\ Y_{32} \\ Y_{42} \\ Y_{13} \\ Y_{23} \\ Y_{33} \\ Y_{43} \\ Y_{14} \\ Y_{24} \\ Y_{34} \\ Y_{44} \end{bmatrix} = \begin{bmatrix} 35 \\ 42 \\ 56 \\ 77 \\ 30 \\ 13 \\ 79 \\ 67 \\ 54 \\ 33 \\ 43 \\ 21 \\ 89 \\ 65 \\ 21 \\ 44 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mu \\ t_1 \\ t_2 \\ t_3 \\ t_4 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} + \begin{bmatrix} e_{11} \\ e_{21} \\ e_{31} \\ e_{41} \\ e_{12} \\ e_{22} \\ e_{32} \\ e_{42} \\ e_{13} \\ e_{23} \\ e_{33} \\ e_{43} \\ e_{14} \\ e_{24} \\ e_{34} \\ e_{44} \end{bmatrix}$$

Matrix notation can also be shown in a more compact way as defined below:

$$Y = X\beta + Zu + e$$

where Y is the vector of observations, X is the treatment design matrix, β is the vector of treatment fixed effect parameters, Z is the block design matrix, u

is the vector of random block effects and e is the vector of experimental errors. Moreover, the variance-covariance matrix of the treatments in RCBDs are defined as:

$$V = V(Y) = \begin{bmatrix} V_b & \phi_r & \phi_r & \cdot & \cdot & \cdot & \cdot & \phi_r \\ \phi_r & V_b & \phi_r & \cdot & \cdot & \cdot & \cdot & \phi_r \\ \phi_r & \phi_r & V_b & \cdot & \cdot & \cdot & \cdot & \phi_r \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \phi_r & \phi_r & \phi_r & \cdot & \cdot & \cdot & V_b & \phi_r \\ \phi_r & \phi_r & \phi_r & \cdot & \cdot & \cdot & \phi_r & V_b \end{bmatrix} \tag{3.3}$$

where $V_b = \sigma_b^2 J_r + \sigma^2 I_r$ is the covariance matrix of all the observations in a particular block, ϕ_r is an $r \times r$ matrix of zeros, and J_r is an $r \times r$ matrix of 1's (Littell et al., 1996).

At this point, our aim is to find the estimates of the treatment means. To find the estimates of the treatment means, least squares estimates of β are required. These least squares estimates are defined in (3.4) below as:

$$\hat{\beta} = (X^\top V^{-1} X)^{-1} X^\top V^{-1} Y \tag{3.4}$$

Note again that the variance parameters σ^2 and σ_β^2 are unknown. Therefore, the estimates of σ^2 and σ_β^2 are needed to be used which are obtained in section (2.3). On the other hand, the variance matrix of least square estimates of β is defined as:

$$V(\hat{\beta}) = (X^T V^{-1} X)^{-1} \tag{3.5}$$

However, in most cases this matrix is singular. Therefore, the least square estimate of β parameters does not exist. C. R. Rao (1961) found a definition to take the inverse of a singular matrix in linear equations.This definition is called generalized inverse. If the variance of the estimates of fixed effect parameters is not singular (if least square estimates exist), least square inverse and generalized least square inverse give the same results. Therefore, in this study, generalized least square inverse is always used to find the variance matrix of the estimates of fixed effect parameters.

Minitab is a widelly used statistical program to find Intrablock estimates of treatment means. SAS PROC GLM can also be used to find Intrablock estimates of treatment means. Minitab and SAS PROC GLM give the exact same results for the estimates of treatment means. On the other hand, SAS PROC MIXED is used to find combined estimators of the treatment means. However, in the RCBDs treatments ordinary least squares matrix notation and generalized linear least squares matrix notation give the same results with respect to treatment means. In other words, the same estimates of treatment means are obtained by using SAS PROC GLM and SAS PROC MIXED in RCBDs. The intrablock estimates of treatment effects and treatment means by using SAS PROC MIXED are presented in Table 3.1 for the complete data in Table 1.1 as:

Table 3.1: Intrablock Estimates and Treatment Means for the Data in Table 1.1

Parameter	Intrablock Estimate	Treatment Means
t_1	3.94	$\hat{\mu}_1 = 52.000$
t_2	-9.81	$\hat{\mu}_2 = 38.2500$
t_3	1.69	$\hat{\mu}_3 = 49.7500$
t_4	4.19	$\hat{\mu}_4 = 52.2500$

3.2 Incomplete Block Designs in Matrix Notations

The same matrix notation in RCB designs can be used for the BIBD and UIBDs with small differences. For the incomplete designs some of the observations cannot be observed. Therefore, the missing observations in the vector of observations (Y) need to be removed. For example, there are 8 missing observations in Table 1.3. These missing observations correspond to y_{21}, y_{41}, y_{12}, y_{42}, y_{13}, y_{14}, y_{34} and y_{35} in the vector of observations (Y). Therefore, these 8 missing observations are removed from the Y vector. After removing missing values in Y, corresponding rows of missing values in the block design matrix (Z) and in the treatment design matrix (X) are also removed. For the Data in Table 1.3, the index numbers of these 8 rows corresponding to the 8 missing observations are: 2, 4, 5, 8, 9,13,15 and 19. Here, the new Y vector is obtained after removing missing observations. Therefore, this new vector is defined as the observed Y vector (Y_{Obs}). Similarly, the X and Z matrices are obtained after removing corresponding rows of missing values. Therefore, these matrices are also defined as the observed treatment design matrix (X_{Obs}) and the observed block design matrix (Z_{Obs}). On the other hand, based on the changes to the Y vector, X and Z matrices, the variance of the Y (V) is also converted to the variance of the observed Y vector which is defined as (V_{Obs}). Note that, treatments in blocks are different. Therefore, additional information can be obtained by using the difference between block totals. This means that combined estimates of treatments are needed. Therefore, SAS PROC MIXED is used to find estimates of treatment means.

Based on the data in Table 1.3, the matrix notation in clear detail is shown as:

$$\begin{bmatrix} Y_{11} \\ Y_{31} \\ Y_{22} \\ Y_{32} \\ Y_{23} \\ Y_{33} \\ Y_{43} \\ Y_{24} \\ Y_{44} \\ Y_{15} \\ Y_{25} \\ Y_{45} \\ Y_{16} \\ Y_{26} \\ Y_{36} \\ Y_{46} \end{bmatrix} = \begin{bmatrix} 76 \\ 5 \\ 76 \\ 78 \\ 78 \\ 73 \\ 70 \\ 85 \\ 77 \\ 75 \\ 75 \\ 85 \\ 74 \\ 77 \\ 82 \\ 79 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mu \\ t_1 \\ t_2 \\ t_3 \\ t_4 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \end{bmatrix} + \begin{bmatrix} e_{11} \\ e_{31} \\ e_{22} \\ e_{32} \\ e_{23} \\ e_{33} \\ e_{43} \\ e_{24} \\ e_{44} \\ e_{15} \\ e_{25} \\ e_{45} \\ e_{16} \\ e_{26} \\ e_{36} \\ e_{46} \end{bmatrix}$$

As a result, the variance matrix of generalized least square estimates of the observed values $V(\hat{\beta}_{Obs})$, generalized least square estimates of treatments of the observed values $\hat{\beta}_{Obs}$ and the variance of observed Y vector $V(Y_{Obs})$ are derived in (3.6). (3.7) and (3.8) respectively as:

$$V(\hat{\beta}_{Obs}) = (X_{Obs}^T V_{Obs}^{-1} X_{Obs})^{-1} \tag{3.6}$$

$$\hat{\beta}_{Obs} = (X_{Obs}^{\top} V_{Obs}^{-1} X_{Obs})^{-1} X_{Obs}^{\top} V_{Obs}^{-1} Y_{Obs} \tag{3.7}$$

$$V_{Obs} = V(Y_{Obs}) = \begin{bmatrix} V_{b_1} & \phi_{r_1} & \phi_{r_1} & \cdot & \cdot & \cdot & \cdot & \phi_{r_1} \\ \phi_{r_2} & V_{b_2} & \phi_{r_2} & \cdot & \cdot & \cdot & \cdot & \phi_{r_2} \\ \phi_{r_3} & \phi_{r_3} & V_{b_3} & \cdot & \cdot & \cdot & \cdot & \phi_{r_3} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \phi_{r_{r-1}} & \phi_{r_{r-1}} & \phi_{r_{r-1}} & \cdot & \cdot & \cdot & V_{b_{r-1}} & \phi_{r_{r-1}} \\ \phi_{r_r} & \phi_{r_r} & \phi_{r_r} & \cdot & \cdot & \cdot & \phi_{r_r} & V_{b_r} \end{bmatrix} \tag{3.8}$$

where $V_{b_i} = \sigma_b^2 J_{r_i} + \sigma^2 I_{r_i}$. Here, r_i is the number of non-missing observations in the ith block. ϕ_{r_i} is an $r_i \times r_i$ matrix of zeros, and J_{r_i} is an $r_i \times r_i$ matrix of 1's. Based on the matrix notations, combined estimates of the treatments and treatment means for the unbalanced data in Table 1.3 are presented in Table 3.2 as:

Table 3.2: Combined Estimates and Treatment Means for the Data in Table 1.3

Parameter	Combined Estimate	Treatment Means
t_1	2.19	$\hat{\mu}_1 = 75.00$
t_2	5.39	$\hat{\mu}_2 = 78.20$
t_3	-13.31	$\hat{\mu}_3 = 59.50$
t_4	4.94	$\hat{\mu}_4 = 77.75$

4 LMER and SAS PROC MIXED

The SAS PROC MIXED procedure and the LMER function in R are used to fit mixed linear models. These procedures help us to obtain some additional information on estimating pairwise treatment contrasts especially in the BIBDs. In this study, the LMER function is used for some simulation purposes. Therefore, one of the main aims of the study is to get the same results with the LMER function in R that SAS PROC MIXED gives with respect to treatment means and covariance parameters. In this section, three types of estimates are compared for SAS PROC MIXED and LMER. These estimates are: Covariance Parameter Estimates, Treatment Mean Estimates and Treatment Mean Standard Error Estimates. Before the comparison of the LMER function and the SAS PROC MIXED procedure, the use and output of the LMER function will be discussed briefly.

4.1 The Use of LMER Function in R

After fitting the model with the LMER function, four main sections are obtained. These sections are: fitted model information, statistics that characterize the fitted model, a summary related to random effect parameters and a summary related to fixed effect parameters. As mentioned, in this study, we are mostly interested in the inferences related to the fixed effect parameters and random effect parameters. Before explaining how to put these fixed treatment effects and random block effects into the LMER function, two important points need to be clarified.

First, fixed treatment effects and random block effects are the factors in our model. In other words, these variables can have only a restricted number of diverse values. “One of the most important uses of factors is in statistical modelling,

since categorical variables enter into statistical models differently then continuous variables, storing data as factors insures that the modeling functions will treat such data correctly " (Spector, 2008, p. 67). Therefore, fixed and random effects need to be introduced in R as factors. Otherwise, R will consider these factors as continuous variables and this will cause incorrect results. Second, to use the LMER function in R, the package LME4 needs to be installed. Now the LMER function can be used to fit the model as shown below.

$> \mathbf{lmer}(\mathbf{y} \sim \mathbf{1} + (\mathbf{treatments}) + (\mathbf{1}|\mathbf{blocks}), \mathbf{Data})$

where "$\mathbf{y}$" corresponds to the response values, "$\mathbf{treatments}$" corresponds to the fixed treatment effects, "$(\mathbf{1}|\mathbf{blocks})$" corresponds to the random block effects and "$\mathbf{Data}$" corresponds to the name of the data frame that is used. As shown above, the effects are separated by the operator (+) and the random effect factor has two terms seperated by the operator $(|)$. In the random effect factor, the term "1" shows the additivity of the model with respect to blocks.

4.2 Variance Estimates in SAS PROC MIXED and LMER

As mentioned, SAS PROC MIXED uses matrix notations to find combined treatment means. In matrix notations, the variances of blocks and residuals are substituted with the estimates of these variances. To find these variance estimates, SAS PROC MIXED uses two methods which are Maximum Likelihood Estimation (ML) and Restricted Maximum Likelihood Estimation (REML).

Rao (1997) states that explicit formulas cannot be obtained for the ML and REML estimates for the UIBDs and some iterative procedures are used to find them. Moreover, ML and REML estimators are biased. However, ML estimates

tend to be more biased when compared to REML estimates, since ML estimates do not consider the loss in degrees of freedom required for estimating μ for the fixed parameters (Rao, 1997).

As a result, for the mixed-effects models, REML estimates are used more than ML estimates, since ML estimates are more likely to be biased. In the LMER function, ML and REML estimators can be used to find the estimators of variance components. However, LMER uses REML as the default criterion. Therefore, to get the same results with respect to variance parameter estimates, the REML method is used in both the LMER function and the SAS PROC MIXED procedure.

Moreover, note that the UIBD data in Table 1.3 are used for two reasons. First reason is to show how SAS PROC MIXED uses matrix notation in incomplete block designs which is already discussed in chapter 3.2. The second one is to discuss one possible reason that leads us to obtain a zero block variance estimate. The variance parameter estimates of both LMER and SAS PROC MIXED are presented in (4.1) and (4.2) respectively for the UIBD data in Table 1.3.

Table 4.1: SAS PROC MIXED Variance Parameter Estimates

Covariance Parameter Estimates				
Cov Parm	Estimate	Standard Error	Z Value	Pr >z
Block	0	-	-	-
Residual	348.38	142.23	2.45	0.0072

Table 4.2: LMER Variance Parameter Estimates

Random effects			
Groups	Name	Variance	Std. Dev.
blocks	(Intercept)	0.000	0.0000
Residual		348.38	18.665

As seen from the tables 4.1 and 4.2, the variance component for blocks is estimated as zero. This demonstrates that there might be some outliers that increase error variance and affect the results. When error variance is increased, it may become larger than $MS_{Blocks(adj)}$. This causes method of moments (MOM) estimate of block variance to be negative. SAS PROC MIXED forces block variance to be non-negative based on the equation in (2.18b) in chapter 2.3. As a result, the block variance component is set equal to zero. Note that, blocks and treatments are the factors of our model. Although it may seem as if the blocks have no effect on the response based on the results in tables 4.1 and 4.2, the blocks should not be removed from the model. Because the placement of treatments in our design depends on our block factors.

In addition, LME4 package in R is still in the development process. Therefore, even if in most situations the LMER function in package LME4 gives the correct variance component estimates, sometimes the convergence criteria may not be met. Altman (2012) explains this by stating that "R-lme handles mixed models and uses REML for estimating variance components. However, lme avoids the boundary problem (i.e. variance ≥ 0) by parameterizing by log(variance)" (Altman, 2012, p.12).

SAS and R-lme give the same variance component estimates if the Type 3 estimates of the variance components are not negative. However, if we have a

negative Type 3 variance component estimate, LMER function in R will achieve a log(variance) estimate which is close to $-\infty$. Afterwards, this estimate is converted back to a variance component estimate. When this happens we obtain a component estimate which is almost, but not exactly zero and this causes odd and unsatisfactory results related to the Type 3 tests of fixed effects (Altman, 2012).

Note that, when this situation occurs the ratio of the block variance estimate to the residual variance estimate ($\frac{\hat{\sigma}^2_\beta}{\hat{\sigma}^2}$) is almost zero after the interblock analysis is performed. In other words, when the ratio is very close to zero, LMER function in R does not produce the same results with SAS PROC MIXED. In this study the LMER function is used for simulation purposes. Therefore, when we encounter a dataset that has a very small ratio, it is removed from the simulation.

4.3 Mean and Standard Error Estimates of the Treatments in SAS PROC MIXED and LMER

By using LSMEANS command in SAS PROC MIXED, treatment mean estimates can be obtained directly. The SAS PROC MIXED least squares means output for the data in Table 1.3 is shown in Table 4.3.

Table 4.3: PROC MIXED Treatment Mean Estimates for the Data in Table 1.3

Least Squares Means						
Effect	Treatment	Estimate	SE	DF	t Value	Pr >F
Treatment	1	75.0000	10.776	7	6.96	0.0002
Treatment	2	78.2000	8.3472	7	9.37	<.0001
Treatment	3	59.5000	9.3325	7	6.38	0.0004
Treatment	4	77.7500	9.3325	7	8.33	<.0001

On the other hand, in R, the summary related to the fixed effect parameters in

the linear mixed model is interpreted similar to in that of linear regression. There will be a constant term and slopes of each fixed effect predictor. In R, the lowest numerical value of the treatment levels is considered as the reference group. Therefore, the LMER function in R uses the first treatment group as the reference group. The summary related to the fixed effect parameters are obtained for the unbalanced data in Table 1.3 as in Table 4.4.

Table 4.4: LMER Fixed Effects Summary for the Data in Table 1.3

Fixed effects:			
	Estimate	Std. Error	t Value
(Intercept)	75.000	10.78	6.960
treatments2	3.200	13.63	0.235
treatments3	-15.50	14.26	-1.087
treatments4	2.750	14.26	0.193

The LMER fixed effect summary gives estimates μ^* and t_i^* where i= 1,2,...a. Here, the intercept term estimate μ^* in the LMER output corresponds to the first treatment mean estimate. To find the other treatment mean estimates the other rows are summed with the intercept term based on the equation $\mu_i^* = \mu^* + t_i^*$. Table 4.5 shows the result of this procedure for the data in Table 1.3.

Table 4.5: Treatment Mean Estimates in LMER for the Data in Table 1.3

Parameter	Treatment Means
t_1	μ_1^*=75.000
t_2	μ_2^*=78.200
t_3	μ_3^*=59.500
t_4	μ_4^*=77.750

Because LMER and SAS PROC MIXED use different strategies to obtain treatment mean estimates, standard error estimates in these procedures also seem dif-

ferent from each other. However, after doing some small adjustments the exact same standard error estimates can be obtained. As seen in Table 4.3, the LSMEANS command gives the standard error estimates of the treatment means directly in SAS PROC MIXED. On the other hand, the LMER summary does not give the standard error estimates of treatment means directly. To find the estimates of standard errors, the variance covariance matrix of the treatment effects is required. The variance covariance matrix structure in R can be defined as below:

$$\begin{bmatrix} \text{Var}(\mu^*) & \text{Cov}(\mu^*, t_2^*) & \text{Cov}(\mu^*, t_3^*) & \cdot & \cdot & \cdot & \cdot & \text{Cov}(\mu^*, t_a^*) \\ \text{Cov}(\mu^*, t_2^*) & \text{Var}(t_2^*) & \text{Cov}(t_2^*, t_3^*) & \cdot & \cdot & \cdot & \cdot & \text{Cov}(t_2^*, t_a^*) \\ \text{Cov}(\mu^*, t_3^*) & \text{Cov}(t_3^*, t_2^*) & \text{Var}(t_3^*) & \cdot & \cdot & \cdot & \cdot & \text{Cov}(t_3^*, t_a^*) \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ \text{Cov}(\mu^*, t_a^*) & \text{Cov}(t_a^*, t_2^*) & \text{Cov}(t_a^*, t_3^*) & \cdot & \cdot & \cdot & \cdot & \text{Var}(t_a^*) \end{bmatrix} \tag{4.1}$$

Based on the data in Table 1.3, the matrix structure is obtained as:

$$\begin{bmatrix} 116.13 & -116.13 & -116.13 & -116.13 \\ -116.13 & 185.80 & 116.13 & 116.13 \\ -116.13 & 116.13 & 203.22 & 116.13 \\ -116.13 & 116.13 & 116.13 & 203.22 \end{bmatrix}$$

On the other hand, the variance of the sum of two variables which are dependent on each other is defined as:

$$\text{Var}(X + Y) = \text{Var}(X) + \text{Var}(Y) + 2\text{Cov}(X, Y) \tag{4.2}$$

By utilizing the equation in (4.2), the equation in (4.3) can be obtained:

$$\text{Var}(\mu_i^*) = \text{Var}(\mu^* + t_i^*) = \text{Var}(\mu^*) + \text{Var}(t_i^*) + 2\text{Cov}(\mu^*, t_i^*) \tag{4.3}$$

Based on the information in (4.3), the variance and standard error estimates of the treatment means are obtained as seen in Table 4.6.

Table 4.6: Standard Error Estimates of the Treatment Means in R

Parameter	Variance Estimates	Standard Error Estimates
t_1	$\text{Var}(\mu_1^*) = 116.1265$	$\text{SE}(\mu_1^*) = 10.7762$
t_2	$\text{Var}(\mu_2^*) = 69.67583$	$\text{SE}(\mu_2^*) = 8.347205$
t_3	$\text{Var}(\mu_3^*) = 87.09479$	$\text{SE}(\mu_3^*) = 9.332459$
t_4	$\text{Var}(\mu_4^*) = 87.09479$	$\text{SE}(\mu_4^*) = 9.332459$

Appendix B shows the code used to explain obtaining combined covariance parameter estimates, combined treatment mean estimates and combined treatment mean standard error estimates in R.

5 R Package AMELIA2

Imputation methods are techniques used to deal with missing values to make data complete. After obtaining complete data, the corresponding analysis can be done based on the analysis techniques for complete cases (Yuan, 2011).

In this study, an R package, Amelia2, is used to handle missing observations. Amelia2 uses a multiple imputation method to make the data complete. Note that, the name of Amelia2 refers to a new version of the original package Amelia. At this point, our aim is to see if the multiple imputation procedure provides a better understanding of the data with respect to treatment means. Therefore, first missing observations are completed by using Amelia2. Afterwards, the same analysis is done with the complete dataset by using the SAS PROC MIXED procedure. In this section, the use of Amelia2 is explained. Note that, this section is covered based on the information in Honaker et al. (2012). Some other topics (the versions of Amelia2, imputation improving transformations etc.) are also covered clearly in the same document. However, these topics are out of the scope of this study. For more information about these topics, please refer to (Honaker et al., 2012).

5.1 The Analysis with Amelia2

Most of the statistical programs remove the missing observations based on list-wise deletion. However, these missing observations may provide some important information about the data and removing them can cause the power of the analysis to decrease dramatically. At this point, multiple imputation plays a very crucial role in recovering wasted information and increases the power of the analysis (Honaker et al., 2012).

The basic idea behind multiple imputation is predicting missing values in the data by using the information from existing values. Amelia2 uses a bootstrapped based EM algorithm (EMB) to fill in the missing parts of the data. In the Amelia2 procedure, this algorithm can be performed very easily and fast. Before explaining the use of the Amelia2 procedure in R, the assumptions and the algorithm of the procedure will be discussed briefly.

The multiple imputation model in Amelia2 has two assumptions. The first assumption is that the full dataset (D), including both observed (D^{obs}) and unobserved (D^{mis}) values, are multivariate normal with mean vector μ and covariance matrix Σ. Therefore, $D \sim$ N(μ,Σ) (Honaker et al., 2012).

Honaker et al.(2012) explains the second assumption by stating that "The essential problem of imputation is that we only observe D^{obs}, not entirely of D. In order to gain traction, we need to make the usual assumption in multiple imputation that the data are missing at random (MAR). This assumption means that the pattern of missingness only depends on the observed data D^{obs}, not the unobserved data D^{mis}" (Honaker et al., 2012, p.4).

Now let us look at the algorithm of the Amelia2 procedure. Assume that M is the missingness matrix that shows whether or not a cell is observed in the dataset. Therefore, we only use the values 0 and 1 in the M matrix. Zero indicates that the value of the cell is unobserved and 1 indicates that the value of the cell is observed. Based on the missing at random assumption:

$$p(M \setminus D) = p(M \setminus D^{obs}) \tag{5.1}$$

This indicates that the probability of obtaining a missingness matrix given observed values is the same as the probability of obtaining the same missingness matrix given the full dataset. Therefore, at this point we can focus on the observed data parameters μ and Σ. Here, the likelihood of the observed dataset can be defined as $p(D^{obs}, M \setminus \theta)$, since the M matrix already depends on D^{obs}. This can be rewritten as:

$$p(D^{obs}, M \backslash \theta) = p(M \backslash D^{obs}) p(D^{obs} \backslash \theta) \tag{5.2}$$

As mentioned, we focus on the inferences of D^{obs}. Therefore, the likelihood can be written as:

$$L(\theta \backslash D^{obs}) \propto p(D^{obs} \backslash \theta) \tag{5.3}$$

Based on the law of iterated expectations:

$$p(D^{obs} \setminus \theta) = \int p(D \setminus \theta) dD^{mis} \tag{5.4}$$

As a result, based on 5.3,

$$p(\theta \setminus D^{obs}) \propto \int p(D \setminus \theta) dD^{mis} \tag{5.5}$$

The EM algorithm is used to find the mode of this posterior. In Amelia2, the EM algorithm is combined with a bootstrap approach which is called the EMB algorithm. Afterwards, this algorithm is used to sample from this posterior (Honaker et al., 2012).

"Once we have draws of the posterior of the complete data parameters, we make imputations by drawing values of D^{mis} from its distribution conditional on D^{obs} and the draws of θ, which is a linear regression with parameters that can be calculated directly from θ "(Honaker et al., 2012, p. 5). The working scheme of the EMB algorithm in Honaker et al. (2012) is shown in figure (5.1).

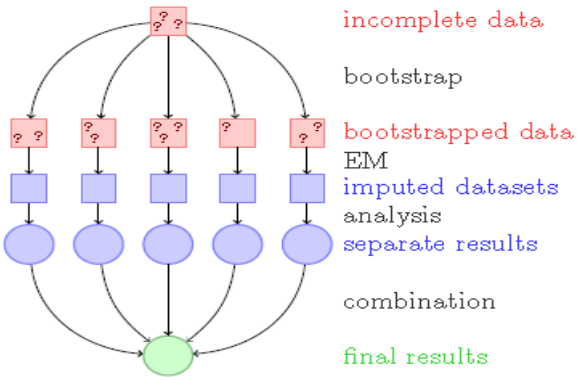

Figure 5.1: The Working Scheme of the Algorithm in Amelia

First, we have incomplete data with sample size n including both observed and missing values. The nonparametric bootstrapping method is used to generate subsamples with size n from the data m times. Afterwards, the EM algorithm is performed on each of these m subsamples to construct m completed datasets. Here, the EM algorithm is used for fitting models to the incomplete data to obtain the point and variance estimates of the parameter (quantity of population interest) θ (Takahashi and Ito, 2012).

On the other hand, we have m imputed datasets which means that there are m point and variance estimates for the parameter θ. Therefore, $\hat{\theta}_i$ and $\hat{U}_i$ values demonstrate the point and variance estimates of θ for the ith imputed dataset respectively, where $i = 1, 2, ...m$. By averaging these parameter estimates, we

obtain the point estimate of θ and within imputation variance as shown in the equations (5.6) and (5.7) respectively.

Point estimate of θ:

$$\bar{\theta} = \frac{1}{m} \sum_{i=1}^{m} \hat{\theta}_i \tag{5.6}$$

Within imputation variance:

$$\bar{U} = \frac{1}{m} \sum_{i=1}^{m} \hat{U}_i \tag{5.7}$$

After obtaining within imputation variance, we also need to calculate between-imputation variance which is the sample variance of the point estimates.

$$B = \frac{1}{m-1} \sum_{i=1}^{m} (\hat{\theta}_i - \bar{\theta})^2 \tag{5.8}$$

The total variance defined in (5.9), is the combination of within imputation variance, between imputation variance and the correction factor which represents the simulation error in $\bar{\theta}$:

$$T = \bar{U} + (1 + \frac{1}{m})B \tag{5.9}$$

The square root of total variance gives us the overall standard error related to $\bar{\theta}$. On the other hand, Rubin (1987) proves that the statistic $\frac{(\theta - \bar{\theta})}{\sqrt{T}}$ approximates to a t-distribution with v_m degrees of freedom, where v_m is defined as:

$$v_m = (m-1)[1 + \frac{\bar{U}}{(1 + m^{-1})B}]^2 \tag{5.10}$$

Note that, this degrees of freedom related to the m value and the fraction r:

$$r = \frac{(1 + m^{-1})B}{\bar{U}} \tag{5.11}$$

Here, r is called the relative increase in variance due to nonresponse (Rubin,1987). This ratio is called relative increase, because the numerator represents the sum of the between imputation variance and the correction factor for the simulation error in $\bar{\theta}$ and the denominator represents the within imputation variance. By using relative increase in variance (r) and degrees of freedom v_m, Rubin (1987) also proves that an estimate of missing information about θ is:

$$\eta = \frac{r + 2/(v_m + 3)}{r + 1} \tag{5.12}$$

Note that, η is the estimate of the fraction of missing information about the parameter θ. By using the formula in (5.12) Rubin (1987) obtains an approximate function of m and η that represents the relative efficiency of using m estimator instead of using the number infinite as:

$$RE = (1+\frac{\eta}{m})^{-1} \tag{5.13}$$

Based on the equation (5.13), in the cases the rates of missing information are not extremely high very little efficiency increases are obtained by analyzing more than a small number of imputations (Yuan, 2011).

Obtained efficiencies for some different values of m and η are shown in Table 5.1. Note that, these efficiencies calculated for a real dataset not for a simulated data.

Table 5.1: Efficiency Table

	η								
m	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
3	0.967	0.938	0.901	0.882	0.857	0.833	0.811	0.790	0.769
5	0.980	0.962	0.943	0.926	0.901	0.893	0.877	0.862	0.848
7	0.986	0.972	0.959	0.946	0.933	0.921	0.909	0.897	0.886
9	0.989	0.978	0.968	0.957	0.947	0.938	0.928	0.918	0.909
11	0.991	0.982	0.974	0.965	0.956	0.948	0.940	0.932	0.924
13	0.992	0.985	0.978	0.970	0.963	0.956	0.949	0.942	0.935
15	0.993	0.987	0.980	0.974	0.968	0.962	0.955	0.949	0.943
17	0.994	0.988	0.983	0.977	0.971	0.966	0.961	0.955	0.950
20	0.995	0.990	0.985	0.980	0.976	0.971	0.966	0.962	0.957
30	0.997	0.993	0.990	0.987	0.984	0.980	0.977	0.974	0.971
40	0.997	0.995	0.993	0.990	0.988	0.985	0.982	0.980	0.978
50	0.998	0.996	0.994	0.992	0.990	0.988	0.986	0.984	0.982
100	0.999	0.998	0.997	0.996	0.995	0.994	0.993	0.992	0.991
500	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.998	0.998

Therefore, as seen in Table 5.1, in the cases the rates of missing information is not extremely high very little efficiency increases are obtained by analyzing more than m = 5 imputed datasets. Moreover, Schafer (1999) states that standard deviation of an estimate that is obtained by using m = 5 is only at around 5 percent larger than the one with $m = \infty$ when missing information rate is 50% (Schafer, 1999). In our simulation study, we have 4 missing observations and 12 observed values. As a result, our η rate is much smaller than 50%. Therefore, in our case, standard deviation of an estimate that is obtained by using m = 5 is even less than 5 percent larger than the one with $m = \infty$. Moreover, to make sure that very little efficiency increase is obtained by using more than m = 5 imputed datasets, the analysis is also done with m = 20 based on the Table 5.1 and very similar results are obtained.

At this point, Amelia2 codes in R will be explained. First, load the data into R as:

$> \mathbf{Data} = \mathbf{read.table}(\text{“}\mathbf{Dataname.txt}\text{”}, \mathbf{header} = \mathbf{TRUE})$

Note that, here “Dataname” represents the name of the data file. “header=TRUE” option makes R notice first row of the data as header. The Amelia package can be installed on any platform by simply typing:

$> \mathbf{install.packages}(\text{“}\mathbf{Amelia}\text{”})$

After first installation of Amelia packages, instead of the code above the code below can be used to use “Amelia” package.

$> \mathbf{library}(\mathbf{Amelia})$

Afterwards, install another required package MASS and load the Amelia as shown below:

$> \mathbf{library}(\mathbf{MASS})$

$> \mathbf{require}(\mathbf{Amelia})$

Now, Amelia imputation model can be executed in R as:

$> \mathbf{a.out} = \mathbf{amelia}(\mathbf{Data}, \mathbf{m} = \mathbf{5})$

Here, “Data” represents the name of the dataset used. This dataset is of the class “data.frame” which is used for storing data tables. Moreover, Honaker et al. (2012) states that “It is crucial to include at least as much information as will be used in the analysis model. That is, any variable that will be in the analysis model should also be in the imputation model” (Honaker et al.,2012, p. 10). In our case, there are the factors containing information about the data and they are used in the analysis model. Therefore, they must be included in the data frame which name is “Data” in the code above. On the other hand, m = 5 refers that 5 imputed datasets are created by Amelia. “Unless the rate of missingness is very

high m = 5 (the program default) is probably adequate"(Honaker et al., 2012, p.4).

As mentioned, after using Amelia code with m = 5, there will be 5 imputed datasets. In these imputed datasets, the observed values will be the same with the observed values of the original data. The only change will be on the unobserved missing values. Honaker et al. (2012) states that "The mean value across all imputed values of a missing cell is the best guess from the imputation model of that missing value. The variance of the distribution across imputed datasets correctly reflects the uncertainty in that imputation" (Honaker et al., 2012, p. 26). Therefore the mean value across the 5 imputed values of each missing cell is obtained and this value is entered into the missing parts of the data. Appendix C shows the code used to explain the Amelia2 procedure in R. In our study, the output of the Amelia2 procedure produces four values which corresponds to the four missing values in the data.

6 Results

Three types of designs are used in this study: randomized complete block design (RCBD), balanced incomplete block design (BIBD) and unbalanced incomplete block design (UIBD). For each design an example with data is shown in the introduction section. RCBD data are used to explain how SAS PROC MIXED uses matrix notation in complete block designs. BIBD data are used to illustrate how to get intrablock, interblock and combined estimates with straightforward formulas. UIBD data are used to show how SAS PROC MIXED uses matrix notation in incomplete block designs.

In this study, three different statistical software packages are used: SAS, R and Minitab. Minitab and SAS PROC GLM are used to perform intrablock analysis. Minitab and SAS PROC GLM do not allow interblock analysis to be performed. Therefore, SAS PROC MIXED and LMER function in R are used to obtain combined estimates. Actually, there are two reasons to use R in this study. First, to see whether we can get exactly the same results that SAS PROC MIXED gives by using LMER with respect to treatment mean estimates, variance component estimates and standard error estiamates of the treatment means. Second, R is also used for the simulation calculations.

Two main collections of analyses are performed in this study. In the first analyses, the data in Table 1.2 are used to show how to find intrablock, interblock and combined pairwise treatment mean contrast estimates in BIBDs. These estimators are used to explore whether the interblock analysis really provides some additional information about the differences between the treatments for the BIBD data in Table 1.2. In other words, we look at whether the precision of estimating

pairwise treatment contrasts is increased.

For the second analyses, first the LMER function in R is used to get the results that SAS PROC MIXED gives with respect to variance component estimates, treatment mean estimates and standard error estimates of the treatment means. After confirming both programs get the same results, a simulation study is performed in R. In this simulation study, our main concern is to compare the performance of two methods for estimating true treatment means. These methods are the combined analysis method that SAS PROC MIXED performs and a multiple imputation method that the Amelia2 package in R performs. The simulation study will be explained in subsection 6.2 more clearly.

6.1 Precision Increase on Estimating Pairwise Treatment Mean Contrasts After Doing Interblock Analysis

In this subsection, we will explore whether or not we can get smaller variances for the pairwise treatment mean contrasts after doing interblock analysis compared to intrablock analysis for the BIBD data in Table 1.2. Treatment effects of intrablock, interblock and combined estimates are already obtained in subsections (2.1), (2.2) and (2.3) respectively. By using these estimates, intrablock, interblock and combined pairwise treatment means and their variances will be estimated. At the end, mean squared error of intrablock treatment mean contrasts and combined treatment mean contrasts will be compared based on a precision change formula to explore if interblock analysis provides a better understanding of estimating pairwise treatment mean contrasts. The precision change formula will be explained on the following pages. If the precision change is positive, it means

that additional information is obtained after doing interblock analysis. In other words, use of SAS PROC MIXED increases the precision of estimating pairwise treatment mean contrasts. If the precision change is negative or zero, it means that SAS PROC MIXED does not produce any additional information and SAS PROC GLM is a sufficient procedure.

The intrablock, interblock and combined estimates of treatment effects, obtained for the data in subsections 2.1, 2.2 and 2.3 respectively, are presented in Table 6.1 as:

Table 6.1: Treatment Effect Estimates for the Data in Table 1.2

Parameters	Intrablock Estimates	Interblock Estimates	Combined Estimates
t_1	$\hat{t}_1 = -7.00$	$\tilde{t}_1 = -6.00$	$t_1^* = -6.98$
t_2	$\hat{t}_2 = -2.75$	$\tilde{t}_2 = -2.50$	$t_2^* = -2.74$
t_3	$\hat{t}_3 = 1.50$	$\tilde{t}_3 = -19.00$	$t_3^* = 1.58$
t_4	$\hat{t}_4 = 8.25$	$\tilde{t}_4 = 3.50$	$t_4^* = 8.14$

Rao (1997) states the estimated variance of intrablock pairwise treatment mean contrasts in BIBDs as:

$$V_1 = V(\hat{t}_j - \hat{t}_l) = 2\frac{k}{\lambda a}\hat{\sigma}_1^2 \tag{6.1}$$

Here, $\hat{\sigma}_1^2$ is the MSE value in intrablock analysis. On the other hand, the estimated variance of interblock pairwise treatment contrasts in BIBDs can be defined as:

$$V_2 = V(\tilde{t}_j - \tilde{t}_l) = \frac{2k(k\hat{\sigma}_\beta^2 + \hat{\sigma}_2^2)}{(r-\lambda)} \tag{6.2}$$

Here, $\hat{\sigma}_2^2$ is the MSE value after doing interblock analysis. In other words, $\hat{\sigma}_2^2$ is the estimate of residual variance in the SAS PROC MIXED procedure. Similarly,

$\hat{\sigma}_{\beta}^{2}$ is the block variance estimate in the SAS PROC MIXED procedure. Moreover, the equation of a pairwise treatment contrasts is defined as:

$$(t_j^*-t_l^*) = w(\hat{t}_j-\hat{t}_l)+(1-w)(\tilde{t}_j-\tilde{t}_l) \tag{6.3}$$

where w is a constant. By using the equations in (6.1), (6.2) and (6.3), the variance estimate of combined pairwise treatment contrasts is defined as:

$$V(t_j^*-t_l^*) = w^2V_1+(1-w)^2V_2 \tag{6.4}$$

When $w = \frac{V_2}{(V_1+V_2)}$ the minumum of this variance is obtained. Therefore, the final optimum variance estimate of combined pairwise treatment contrasts is defined as:

$$V_{opt}(t_j^*-t_l^*) = \frac{V_1V_2}{V_1+V_2} = \frac{1}{(\frac{1}{V_1})+(\frac{1}{V_2})} \tag{6.5}$$

For this equation V_1 is the value in equation (6.1) and V_2 is the value in equation (6.2) (Rao, 1997).

In this study, the SAS PROC MIXED optimum variance estimate of combined pairwise treatment contrasts in (6.5) will be denoted by V_3. Based on the equations in (6.1), (6.2) and (6.5), the values of V_1, V_2 and V_3 for the data in Table 1.2 are defined respectively as:

$$V_1 = V(\hat{t}_j - \hat{t}_l) = \hat{\sigma}_1^2 = 3.750$$

$$V_2 = V(\tilde{t}_j - \hat{t}_l) = 4\hat{\sigma}_{\beta}^2 + 2\hat{\sigma}_2^2 = 156.454$$

$$V_3 = V(t_j^* - t_l^*) = \frac{1}{\frac{1}{\hat{\sigma}_2^2} + \frac{1}{4\hat{\sigma}_\beta^2 + 2\hat{\sigma}_2^2}} = 3.5937$$

On the other hand, Toutenburg and Shalabh (2009) states the precision change on estimating treatment mean contrasts after doing interblock analysis as:

$$\frac{\frac{1}{\text{variance of pooled estimate}}}{\frac{1}{\text{variance of intrablock estimate}}} - 1 \tag{6.6}$$

Here, variance of the pooled estimate corresponds to the variance of combined estimates. Therefore, the equation in (6.6) can be redefined as:

$$\text{Precision Change} = \frac{V_1}{V_3} - 1 \tag{6.7}$$

For the data in Table 1.2 this precision change is:

$$\text{Precision Change} = \frac{V_1}{V_3} - 1 = \frac{\hat{\sigma}_1^2}{\frac{1}{\frac{1}{\hat{\sigma}_2^2} + \frac{1}{4\hat{\sigma}_\beta^2 + 2\hat{\sigma}_2^2}}} - 1 = \frac{3.7500}{3.5937} - 1 = 0.0435 \tag{6.8}$$

As seen from the equation in (6.8), to obtain a large precision change we need to have a large residual variance estimate in intrablock analysis ($\hat{\sigma}_1^2$), a small block variance estimate after doing interblock analysis ($\hat{\sigma}_\beta^2$) and a small residual variance estimate after doing interblock analysis ($\hat{\sigma}_2^2$). For the data in Table 1.2, low increased precision estimating pairwise treatment mean contrasts is obtained after performing interblock analysis. This result is presented in Table 6.2. as:

Table 6.2: Change in Precision After Doing Interblock Analysis

Contrast Parameters	SE of Intrablock Estimates	SE of Combined Estimates	Precision Change
$\mu_2 - \mu_1$	1.9365	1.8957	0.0435
$\mu_3 - \mu_1$	1.9365	1.8957	0.0435
$\mu_4 - \mu_1$	1.9365	1.8957	0.0435
$\mu_3 - \mu_2$	1.9365	1.8957	0.0435
$\mu_4 - \mu_2$	1.9365	1.8957	0.0435
$\mu_4 - \mu_3$	1.9365	1.8957	0.0435

As seen in Table 6.2, treatment contrasts have the same standard errors for the intrablock estimates and for the combined estimates. Moreover, the same precision increase is obtained for each pairwise treatment mean contrasts. These are because our design is balanced.

6.2 A simulation Study to Explore the Performance of Amelia2 with SAS PROC MIXED

6.2.1 Data Creation

In this simulation study it is assumed that true means and true variance components are known. The true means are chosen as: μ_1= 40, μ_2= 45, μ_3= 50, μ_4= 55, and the true variance components are chosen as: σ^2_β= 25 and σ^2= 4. Based on the model in (1.1) and true means and true variance components, 1000 separate simulated datasets are created. Moreover, our design is a BIBD with 4 treatments, 4 blocks and a total of N = 12 observations. Because our design is a BIBD, there are 4 missing values in the datasets each time. In accordance with the balanced incomplete block design the pattern of the missing values are based on the positions of the observed values. The model used is shown as:

$$y_{ij} = \mu + t_i + \beta_j + \epsilon_{ij} \begin{cases} i = 1, 2, 3, 4 \\ j = 1, 2, 3, 4 \end{cases}$$

y_{ij} represents the response variable for the ith treatment and jth block, μ represents the mean of all observations, t_i represents ith treatment effect, β_j represents jth block effect and, as always, ϵ_{ij} values are the residuals for the model. Here, t_i=μ_i-μ and μ_i=(40, 45, 50, 55). Moreover, $\beta_j \sim$ N(0,25) and $\epsilon_{ij} \sim$ N(0,4). Note that all of the random variables are independent from each other. Using the information μ_i=(40, 45, 50, 55), the treatment effects vector is defined as: t_i=(-7.5, -2.5, 2.5, 7.5), since $\sum t_i$=0. Because our designs are balanced, based on the true means, the true overall mean of the treatments is μ= 47.5. One data example in our study is shown in Table 6.3 as:

Table 6.3: Data Example in the Simulation Study

	Block			
Treatment	1	2	3	4
1	37.67	—	40.36	34.83
2	42.78	54.40	—	37.96
3	—	61.89	51.39	41.58
4	55.60	67.34	54.36	—

As seen in Table 6.3, our design is a BIBD with the design values a = 4, b = 4, k = 3 and r = 3. Table 6.4 shows how we generated these data based on the positions of the missing values.

Table 6.4: Data Creation

Y_{ij}	μ	t_i	β_j	ϵ_{ij}	Y_{ij} values
Y_{11}	47.5	-7.5	-1.36	-0.97	37.67
Y_{21}	47.5	-2.5	-1.36	-0.86	42.78
Y_{41}	47.5	7.5	-1.36	1.96	55.60
Y_{22}	47.5	-2.5	9.87	-0.47	54.40
Y_{32}	47.5	2.5	9.87	2.02	61.89
Y_{42}	47.5	7.5	9.87	2.47	67.34
Y_{13}	47.5	-7.5	0.28	0.08	40.36
Y_{33}	47.5	2.5	0.28	1.11	51.39
Y_{43}	47.5	7.5	0.28	-0.92	54.36
Y_{14}	47.5	-7.5	-5.58	0.41	34.83
Y_{24}	47.5	-2.5	-5.58	-1.46	37.96
Y_{34}	47.5	2.5	-5.58	-2.84	41.58

Note that, in Table 6.4, there are 4 different values of the treatments (t_i) and 4 different values of the blocks (β_j) corresponding to 4 treatments and 4 blocks. Here, treatments are fixed effects with t_i=(-7.5, -2.5, 2.5, 7.5) and block effects are generated from a normal distribution with mean zero and variance 25. On the other hand, there are 12 different values of the residuals corresponding to 12 observations in our design. These residuals are coming from a normal distribution with mean zero and variance 4.

6.2.2 Mean Distance

In this study, treatment mean estimates obtained by the LMER function are called "combined estimates." On the other hand, the treatment mean estimates that are obtained by the LMER function after using the Amelia2 procedure are called "amelia estimates." As mentioned, we have four true treatment means which are:

40, 45, 50, 55. Therefore, for each randomly generated data sets, four treatment mean estimates are obtained as combined estimates and four treatment mean estimates are obtained as amelia estimates. They are organized in pairs.

As mentioned in chapter 4.2, even if in most situations we obtain the same variance component estimates as LMER in R and SAS PROC MIXED, sometimes convergence criteria may not be met. In this simulation study, treatment mean estimates are obtained before and after using the Amelia2 procedure. The total number of times that treatment mean estimates are obtained correctly in both cases (before and after using Amelia2) is 974 out of 1000 runs. Therefore, these 974 data examples are used. Our aim in this simulation study is to compare the performance of Amelia2 procedure in the cases where the same treatment mean estimates are obtained with SAS PROC MIXED by using the LMER function in R.

Moreover, the distances between true means versus combined treatment mean estimates and true means versus amelia treatment mean estimates are examined. The formula used for this purpose is called Mean Squared Error (MSE) and it represents the estimation error. We should note that the MSE value is not the same MSE value which is obtained in Analysis of Variance. The MSE values of the true means versus combined estimates and true means versus amelia estimates are obtained in 974 runs. The MSE_j^* denotes the jth MSE value between true means (μ_i) and combined treatment mean estimates (μ_i^*) in 974 runs and is defined as MSE_j^*=$\frac{1}{4}\sum_{i=1}^{4}(\mu_i - \mu_i^*)^2$. Similarly, the MSE_j^{**} corresponds to the jth MSE value between true means (μ_i) and amelia treatment mean estimates (μ_i^{**}) and is defined as MSE_j^{**}=$\frac{1}{4}\sum_{i=1}^{4}(\mu_i - \mu_i^{**})^2$.

After running our code, we create a data matrix with 10 columns in which the first four columns represent combined treatment mean estimates, the second four columns represent amelia treatment mean estimates and the 9th and 10th columns are for the MSE^{*} and MSE^{**} values respectively. Our data are shown in Table 6.5 as:

Table 6.5: Data After the Simulation

Index	Com1	Com2	Com3	Com4	Am1	Am2	Am3	Am4	MSE^{*}	MSE^{**}
1	38.85	43.65	47.55	52.39	40.62	41.45	48.59	52.64	3.99	5.14
2	36.77	44.56	47.95	55.15	37.10	43.94	48.42	54.81	3.71	3.02
3	40.08	44.61	47.66	55.22	39.91	44.16	49.06	55.61	1.42	0.49
4	40.58	46.62	49.89	55.63	42.10	46.44	48.69	53.76	0.84	2.43
.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.
971	37.26	46.40	48.87	55.35	37.60	47.61	48.78	54.23	2.72	3.66
972	36.58	40.37	44.92	51.78	37.14	40.39	45.93	49.97	17.33	17.82
973	39.74	48.34	53.75	58.26	40.14	47.40	53.91	59.01	8.98	9.29
974	35.88	38.60	45.82	50.75	35.74	36.75	46.84	51.04	23.37	27.97

Table 6.6 shows the mean, median and standard deviation (S.D.) of the combined and amelia treatment mean estimates for the 974 simulated data sets.

Table 6.6: The Mean, Median and S.D. of the Combined and Amelia Estimates

	Mean	Median	S.D.
Combined 1	39.98	40.02	2.829
Combined 2	44.99	44.94	2.779
Combined 3	49.96	49.92	2.787
Combined 4	54.98	54.93	2.772
Amelia 1	39.95	39.86	3.062
Amelia 2	44.99	44.93	2.880
Amelia 3	49.96	49.93	2.887
Amelia 4	55.03	55.09	2.979

As seen from Table 6.6 the mean and medians of each treatment mean estimates are very close to each other which shows that the distributions of combined and amelia mean estimates are approximately symmetric. This is because our combined and amelia mean estimates are coming from normal distributions. Therefore, the differences between them (Com1 vs Ame1, Com2 vs Ame2, Com3 vs Ame3, Com4 vs Ame4) are also coming from normal distributions. Moreover, based on the table above, the standard deviations of combined estimates are smaller than the corresponding standard deviations of amelia estimates. Now, let us first compare the means, medians and variances of the estimates respectively. Note that the data in our columns are numerical and paired. Therefore, there are two main tests to compare the locations of the distributions of combined and amelia treatment mean estimates. These are parametric paired-t test (to compare the means) and non-parametric Wilcoxon signed rank test (to compare the medians).

The paired t test is used to compare the means of two paired samples. This test assumes that the differences between pairs are normal. Based on central limit theorem, the paired t test is also valid to compare the means of two paired large

samples even if the samples are coming from non-normal distributions. This is because, for large samples, the distribution of the difference between the means of the samples is approximately normal (McClave and Dietrich, 1988).

Here, approximately normal means that the distribution of the difference between the means of the samples is close enough to consider it to be normal. Therefore, this test will also be used to compare the means of the MSE values which are not normally distributed. Comparison of the MSE values will be explained on the next pages. Now let us to compare combined and amelia treatment mean estimates separately by using four paired t tests corresponding to each treatment. Our null and alternative hypotheses are described as:

$$H_0 : \mu_{d_i} = 0$$
$$H_a : \mu_{d_i} \neq 0$$

where $i = 1, 2, 3, 4$. Here, d_i corresponds to differences between combined and amelia treatment mean estimates for the ith treatment. Table 6.7 shows obtained p-values using paired t test related to combined and amelia treatment mean estimates for each treatment separately.

Table 6.7: P-values of Paired t Tests

Sources	P-values
Combined 1 vs Amelia 1	0.4552
Combined 2 vs Amelia 2	0.8816
Combined 3 vs Amelia 3	0.8117
Combined 4 vs Amelia 4	0.1607

Based on paired t test results we cannot reject the null hypotheses for any of the mean comparisons. We cannot conclude that the mean of the differences between

combined and amelia estimates is significantly different than zero.

By using the Wilcoxon signed rank test (WSRT) we compare the distribution functions of two independent samples with respect to their locations (medians). WSRT is a nonparametric statistical test. Therefore, it does not assume that differences between pairs are normal. It only assumes that we have two independent groups from two populations (Toutenburg and Shalabh, 2009).

Wilcoxon signed rank test is generally used when the differences between pairs are not normally distributed. However, at this point we are using this test to verify the results of the paired t test. By using WSRT we expect similar results with the paired t test. This is because our combined and amelia estimates are normally distributed and therefore their means and medians are close to each other. If we denote the medians of the combined and amelia estimates for ith treatment as η_{1i} and η_{2i} respectively, our null and alternative hypotheses become:

$$H_0 : \eta_{1i} = \eta_{2i}$$
$$H_a : \eta_{1i} \neq \eta_{2i}$$

where $i = 1, 2, 3, 4$. Table 6.8 shows obtained p-values using WSRT related to combined and amelia treatment mean estimates for each treatment separately.

Table 6.8: P-values of Wilcoxon Signed Rank Tests

Sources	P-values
Combined 1 vs Amelia 1	0.7109
Combined 2 vs Amelia 2	0.8627
Combined 3 vs Amelia 3	0.8874
Combined 4 vs Amelia 4	0.1574

Based on Wilcoxon signed rank test results we cannot reject the null hypotheses

for any of the median comparisons. We cannot conclude that the median of the differences between combined and amelia treatment mean estimates are significantly different than zero.

On the other hand, we can also use a simple F test to see whether or not the variances of two populations are equal. This test is known as variance test in R. Note that, we must have normally distributed datasets to use the variance test. Our combined and amelia treatment mean estimates are already normally distributed. Therefore, this test can be performed four times to compare the variances of combined and amelia estimates on estimating true values of the four treatments separately. Let us to denote the variances of the combined and amelia estimates for ith treatment as σ_{1i}^2 and σ_{2i}^2 respectively. Our null and alternative hypotheses are:

$$H_0 : \sigma_{1i}^2 = \sigma_{2i}^2$$
$$H_a : \sigma_{1i}^2 < \sigma_{2i}^2$$

where $i = 1, 2, 3, 4$. Table 6.9 shows obtained p-values using variance tests related to combined and amelia treatment mean estimates for each treatment separately.

Table 6.9: P-values of Variance Tests

Sources	P-values
Combined 1 vs Amelia 1	0.007
Combined 2 vs Amelia 2	0.133
Combined 3 vs Amelia 3	0.138
Combined 4 vs Amelia 4	0.013

Based on variance test results for treatments 2 and 3, we cannot reject the null hypotheses that combined and amelia treatment mean estimates have the same variance. On the other hand, for treatments 1 and 4, the variances of combined

and amelia estimates are significantly different from each other and the combined estimates have smaller variance than the amelia estimates.

Combined treatment means are the results of SAS PROC MIXED in which a mixed model is used with four fixed treatment effects t_i=(-7.5,-2.5,2.5,7.5), four random block effects ($\beta_j \sim N(0, 25)$) and twelve different values of the residuals ($\epsilon_{ij} \sim N(0, 4)$) corresponding to twelve observations. Amelia estimates are also the results of SAS PROC MIXED with the same mixed model after filling missing values with a bootstrapped based multiple imputation method. Both SAS PROC MIXED and Amelia2 procedure assume normality of the data. As mentioned, one of the main purposes of this study is to compare the performances of combined and amelia estimates for estimating true treatment means. The statistics that are used for this purpose are mean squared error (MSE^* and MSE^{**}) values which are already explained at the beginning of this chapter. Based on the formulas of MSE^* and MSE^{**} values, smaller MSE demonstrates better performance for estimating true treatment means. Histograms and Q-Q normality plots of these distances are obtained as shown in figure 6.1 and figure 6.2.

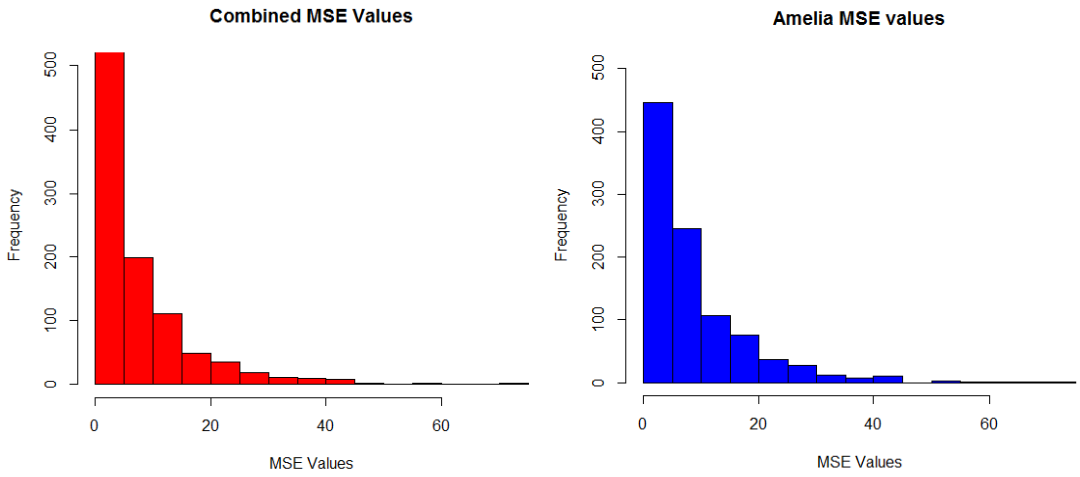

Figure 6.1: Histograms of the MSE Values

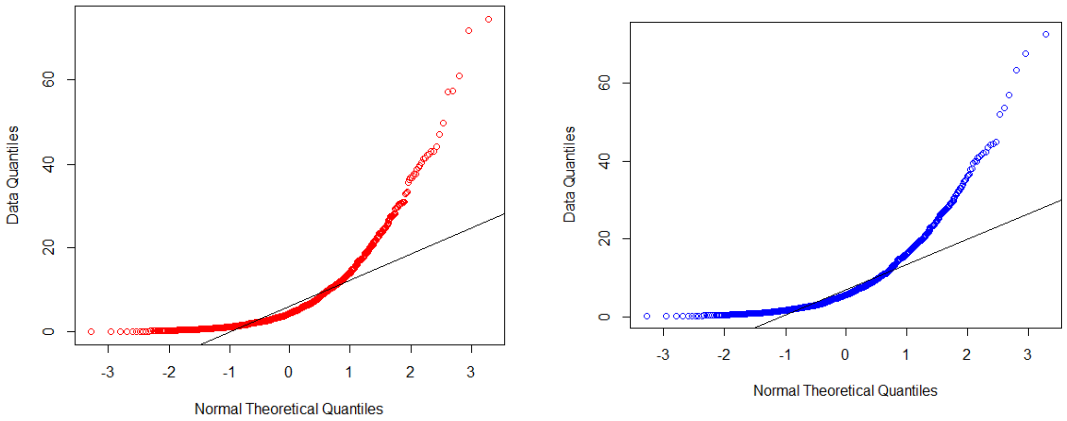

Figure 6.2: Q-Q Plots of the MSE Values

As seen in figure 6.1 and figure 6.2, both of the histograms of the MSE values are right skewed and the distributions are similar to each other. As seen from figure

6.1, combined estimates are more right skewed which demonstrates that we have smaller MSE values for combined estimates compared to amelia estimates. Table 6.10 shows the mean and standard deviation (S.D.) of the MSE values between true treatment means and combined treatment means and between true treatment means and amelia treatment means in 974 runs.

Table 6.10: Comparison of the Means and S.D. of the MSE Values

Sources	Mean	S.D.
MSE^{*} Values (Combined)	7.788	9.268
MSE^{**} Values (Amelia)	8.711	9.380

From the table above, on average, it again seems that the combined estimates have lower MSE values compared to the amelia estimates. Moreover, the standard deviation of the combined estimates is smaller than the standard deviation of the amelia estimates. In figure 6.3, the scatter plot and boxplots of the MSE^{*} and MSE^{**} values are shown.

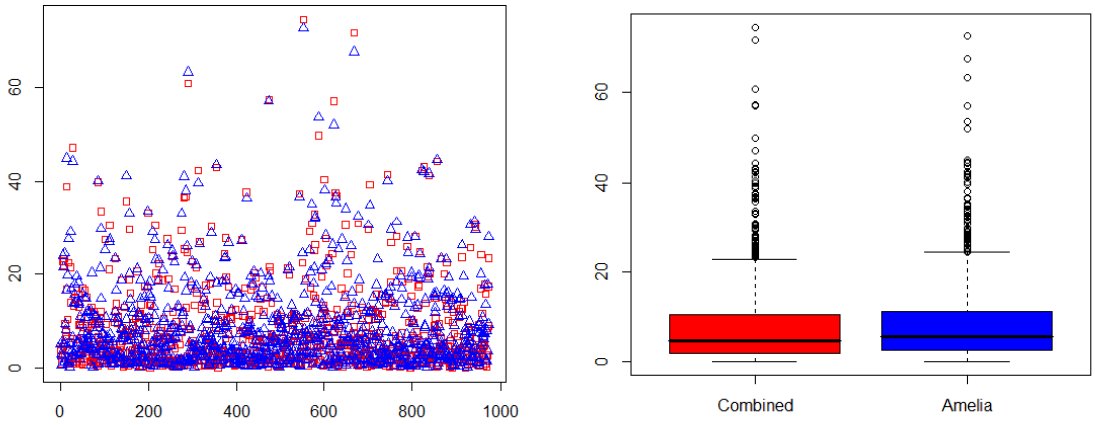

Figure 6.3: Scatter Plot and Boxplots of the MSE Values in 974 Runs

As seen from the scatter plot above, both MSE values are very close to each other. Even if the combined treatment mean estimates tend to be smaller, we cannot observe this from the scatter plot. Some extreme values (outliers) are seen in the scatter plot and box plots for both MSE values. Because we have outliers, the means of the MSE values are elevated. Therefore, as seen in the boxplots above, the medians (dark lines inside the boxes) of both MSE values are lower than their means. This again demonstrates that the distributions of both MSE values are right skewed. Moreover, kernel density plots can be used to show the probability density functions of the MSE^{*} and MSE^{**} values. By using kernel density plots we can observe the difference between the distributions of the MSE values more clearly. Figure 6.4 shows the density plots of both MSE values graphically.

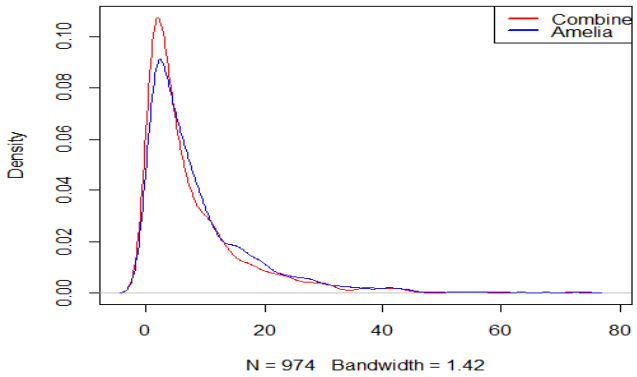

Figure 6.4: Kernel Density Plots

As mentioned, based on the central limit theorem, the paired t test is a valid test to compare the means of two paired large samples for non-normal distributions. On the other hand, the Wilcoxon signed rank test is a non-parametric test and it is also valid for testing paired two large samples without assuming that they are normally distributed. We again use a paired t test to compare the means and a Wilcoxon signed rank test to compare the medians of the MSE values. If we denote the means of the MSE^* and MSE^{**} values as μ_{MSE^*} and $\mu_{MSE^{**}}$ respectively, our null and alternative hypotheses become:

$$H_0 : \mu_{MSE^*} = \mu_{MSE^{**}}$$
$$H_a : \mu_{MSE^*} < \mu_{MSE^{**}}$$

After performing the paired t test, we obtain the p-value as < 0.001 which again demonstrates that the mean of the MSE values based on combined estimates are smaller than the mean of the MSE values based on amelia estimates. On the other hand, if we denote the medians of the MSE^* and MSE^{**} values as η_{MSE^*}

and $\eta_{MSE^{**}}$ respectively, our null and alternative hypotheses become:

$$H_0 : \eta_{MSE^*} = \eta_{MSE^{**}}$$
$$H_a : \eta_{MSE^*} < \eta_{MSE^{**}}$$

After the Wilcoxon signed rank test, we again obtain the p-value as < 0.001 which means that MSE^* values have smaller median compared to MSE^{**} values.

Until this point we showed that combined estimates have better performance on estimating true treatment mean values compared to amelia estimates. Now, by using a non-parametric sign test we can verify that the performance of combined estimates is better than the performance of amelia estimates. By using the sign test we count the number of times the MSE^* values are smaller than the MSE^{**} values. Therefore, in our case, success means obtaining a smaller MSE^* value compared to an MSE^{**} value. The number of trials is 974 and hypothesized probability of success is p = 0.5. Therefore our null and alternative hypotheses become:

$$H_0 : p = 0.5$$
$$H_a : p > 0.5$$

Based on our sign test results, the number of times that the MSE^* values are smaller than the MSE^{**} values is 632 out of 974 runs. Therefore, the probability of success (obtaining a smaller MSE^* compared to MSE^{**}) is 0.6489. The obtained p-value is < 0.001 which again shows that probability of having success is significantly bigger than 0.5.

7 Conclusions

SAS PROC MIXED uses matrix notation to obtain combined estimates which are the linear combination of intrablock and interblock estimates. For the RCBDs, this matrix notation can be found in the literature easily. However, it is difficult to find expressions that show the matrix notation for the BIBDs and UIBDs. One of the goals of this study was to show how SAS PROC MIXED uses matrix notation to find combined estimates in incomplete block designs. This matrix notation is obtained using the matrix notation for RCBDs in Littell et al. (1996). As a result, we concluded that SAS PROC MIXED removes missing observations and reconstructs the block design matrix (Z) and the treatment design matrix (X) based on the positions of observed values in the data. We also concluded that SAS PROC MIXED uses exactly the same matrix notation for the BIBDs and UIBDs.

On the other hand, intrablock analysis and interblock analysis can be used to estimate pairwise treatment mean contrasts. For the RCBDs, exactly the same treatment mean estimates are obtained by using either analysis. However, for the BIBDs and UIBDs, we can do interblock analysis to explore whether additional information can be obtained to better compare treatment means. One of the goals of this study was to determine in which cases additional information can be obtained by using interblock analysis. We concluded that to obtain large precision change after doing interblock analysis we need to have a large residual variance estimate in intrablock analysis ($\hat{\sigma}_1^2$), a small (but not zero) block variance estimate after doing interblock analysis and a small residual variance estimate after doing interblock analysis ($\hat{\sigma}_2^2$).

Moreover, as mentioned, for complete block designs only the exact intrablock

analysis can be used to estimate treatment means, since no missing observations are present in the datasets for these designs. However, for an incomplete block design SAS PROC MIXED can also be considered to be used after obtaining the complete dataset by using the EMB algorithm in the Amelia2 multiple imputation procedure. However, based on our simulation study, the performance of combined estimates estimating true treatment mean values are better than the performance of amelia estimates. Based on paired t tests and Wilcoxon signed rank tests results, the means and medians of combined and amelia treatment mean estimates are similar individually. However, based on variance tests, for the treatments 1 and 4, amelia estimates have larger variances compared to combined estimates. Therefore, on the whole, combined estimates perform a better work on estimating true treatment mean values. Based on our sign test results, for approximately 65 percent of the comparisons (974 runs), we achieved better results by using combined estimates.

References

[1] Naomi S. Altman. R-lme. *Lecture/Class, Pennsylvania State University,* Unpublished, 2012.

[2] D. A. S. Fraser. On the Combining of Interblock and Intrablock Estimates. *The Annals of Mathematical Statistics,* 28:814-816, 1957.

[3] James Honaker, Gary King and Matthew Blackwell. *AMELIA 2: A Program for Missing Data.* 2012.

[4] Tatsuya Kubokawa. *The Recovery of Interblock Information in Balanced Incomplete Block Designs.* The Indian Journal of Statistics, 50:78-79, 1988.

[5] Micheal H. Kutner, Christopher J. Nachtsheim, John Neter, William Li. *Applied Linear Statistical Models.* McGraw-Hill Irwin, NY, Fifth edition, 2005.

[6] Ramon C. Littell, George A. Miliken, Walter W. Stroup, Russell D. Wolfinger. *SAS System for Mixed Models.* SAS Institute Inc, Cary, NC, 1996.

[7] James T. McClave and Frank H. Dietrich. *Statistics.* Dellen Publishing Company,San Francisco, 4th edition, 1988.

[8] Douglas C. Montgomery. *Design and Analysis of Experiments.* John Wiley and Sons, NJ, 6th edition, 2005.

[9] C. Radhakrishna Rao. *A note on a Generalized Inverse of a Matrix with Appications to Problems in Mathematical Statistics.* Royal Statistical Society, 24:152-158, 1961.

[10] Poduri S. R. S. Rao. *Variance Components Estimation, Mixed models, methodologies and applications.* Chapman Hall, NY, London, 1997.

[11] Donald B. Rubin. *Multiple Imputation for Nonresponse in Surveys.* Newyork:John Wiley, 1987.

[12] Joseph L. Schafer. *Multiple Imputation: A Primer.* Statistical Methods in Medical Research. 8:3-15, 1999.

[13] Phil Spector. *Data Manipulation with R.* Springer Science + Business Media, LLC, NY, 2008.

[14] Masayoshi Takahashi and Takayuki Ito. *Multiple Imputation of Turnover in Edinet Data: Toward Ard the Improvement of Imputation for the Economic Census* Conference of European Statisticians, 2012.

[15] Helge Toutenburg and Shalabh. *Statistical Analysis of Designed Experiments.* Springer Science + Business Media, LLC, NY, Third edition, 2009.

[16] Frank Yates. *The Recovery of Inter-block Information in Balanced Incomplete Block Designs.* Ann. Eugenics, 10:317-325, 1940.

[17] Yang Yuan. *Multiple Imputation Using SAS Software.* SAS Institute Inc, 45(6),2011.

APPENDIX A

BALANCED INCOMPLETE BLOCK DESIGN

```
bib=function() {
# Crossdes package is required to use the function find.BIB() which is used
# to create BIBDs
require(crossdes)
# The User is asked to provide design Values: a(treatment number),
# k(treatment number in each block)
cat("a:Treatment Number is:","\ n")
a=scan(n=1)
cat("k: The Number of treatments in each block is:", "\ n")
k=scan(n=1)
# Other design values (b,r and lambda are obtained based on the
# information provided above). Moreover, some values which are used in
# the code are entered into R. These values are (mutreatments(true treatment
# means), alpha(true treatment effects). "ppp" is used in the matrix
#"number" which is created by the Null values depending on treatment
# numbers and block numbers.
mutreatments=rep(NA,a)
alpha=rep(NA,a)
b=choose(a,k)
ppp=rep(NA,(a*b))
number=matrix(c(ppp),nrow=a,ncol=b)
N=b*k
r=N/a
# All of the design values are shown on the screen
print("a:Number of Treatments")
print(a)
print("k:Number of Treatments in each block")
print(k)
if (any(a < k)) {
print( "Wrong Design ")
print( "The number of treatments is supposed to be bigger than k")
stop() }
print("b:Number of blocks")
print(b)
print("r:The number of times each treatment replicated in the design")
print(r)
lambda=(r*(k-1))/(a-1)
```

```
print("lambda value(number of treatment pairs)")
print(lambda)
# The user is asked to provide true treatment means and and true variance
# components. Moreover the overall mean and true treatment effects are
# obtained from the information that the user inputs.
print("Please enter true means of the treatments")
for (i in 1:a) {
cat(i,". True treatment mean is:","\ n")
mutreatments[i]=scan(n=1) }
print("True Treatment Means")
print(mutreatments)
mutreatments=matrix(c(mutreatments))
mumean=sum(mutreatments)/a
for (i in 1:a) {
alpha[i]=mutreatments[i]-mumean }
cat("Please enter true block variance")
sigmab=scan(n=1)
cat("Please enter true residual variance")
sigma=scan(n=1)
cat("Please press enter to continue","\n")
scan(n=1)
#We create a balanced incomplete block design based on the valaues of
# a, b and k which are already obtained. After creating a BIBD, the design
# is displayed in a more understandable format. In the new format, the
# positions of the observed values are listed (based on the treatment number
# index and block number index)
bibdesign=find.BIB(a,b,k)
observations=c(1:(b*k))
blocks=rep(NA,(b*k))
l=0
for (j in seq(1,(b*k),k)) {
l=l+1
blocks[j:(j+(k-1))]=l }
treatments=rep(NA,12)
treatments=t(bibdesign)
treatments=as.integer(treatments)
number2=cbind(observations,treatments,blocks)
print(number2)
for (m in 1:(b*k)) {
number[number2[m,2],number2[m,3]]=5 }
# Based on the information provided by the user, we create
```

```
# a normally distributed dataset. Afterwards, we output the created data on
# the screen.
options(digits=4)
index=c(1:(a*b))
mu=rep(mumean,(a*b))
t_i=rep(NA,(a*b))
l=0
for (i in seq(1,(a*b),a)) {
l=l+1
t_i[i:(i+(a-1))]=mutreatments[1:a]-mu[1:a] }
blk=rep(NA,b)
blk[1:b]=rnorm(b,mean=0,sd=sqrt(sigmab))
B_j=rep(NA,(a*b))
l=0
for (j in seq(1,(b*a),a)) {
l=l+1
B_j[j:(j+(a-1))]=blk[l] }
residual=rep(NA,(a*b))
residual[1:(a*b)]=rnorm((a*b),mean=0,sd=sqrt(sigma))
y_ij=rep(NA,(a*b))
for (i in 1:(a*b)) {
y_ij[i]=mu[i]+t_i[i]+B_j[i]+residual[i] }
createdata=cbind(index,mu, t_i,B_j,residual,y_ij)
print(createdata)
# After creating the design and the dataset, we assign the dataset into the
# design based on the positions of the observed values and show the results
# on the screen.
k=0
numberson=c(number)
ind=rep(NA,(b*k))
for (i in 1:(a*b)) {
if (!is.na(numberson[i])) {
k=k+1
ind[k]=i } }
numberson[!is.na(numberson[])]=y_ij[ind]
numberson=as.matrix(c(numberson),nrow=a,ncol=b)
number=as.matrix(c(number),nrow=1,ncol=(a*b))
number[!is.na(number)]=y_ij[ind]
number=matrix(c(number),nrow=a,ncol=b)
print(number)
}
```

APPENDIX B

FINDING COMBINED ESTIMATES FOR A DATA EXAMPLE

```
trtmeans2=function() {
#We read the data from the file program2-data.txt and we print the data on
# the R console.
Data=read.table("program2-data.txt",header=TRUE)
print(Data)
#We assign the columns of the data to the treatments, blocks and response
# values respectively. Note that we use treatments and blocks as factors by
# using factor() function.
treatments=factor(Data[,2])
blocks=factor(Data[,1])
y=Data[,3]
#Library lme4 is required to use the function lmer.
library("lme4")
#By using lmer function, we obtain the mixed model that fits the data in the
# file program2. Moreover, we obtain the variance component estimates(same
# results with SAS PROC MIXED) from the model summary.
fm2=lmer(y ~ 1 + (treatments) + (1|blocks), Data)
summ=summary(fm2)
print(summ)
slotNames(summ)
matrixsumm2=as.matrix(slot(summ,"fixef"))
matrixsumm1=as.matrix(slot(summ,"REmat"))
print("MS(block)")
sigmab=as.numeric(matrixsumm1[1,3])
print(sigmab)
print("MSE")
sigma=as.numeric(matrixsumm1[2,3])
print(sigma)
#We obtain combined treatment mean estimates(same results with SAS PROC
# MIXED).
a=nrow(matrixsumm2)
print("Treatment Means")
print(matrixsumm2[1])
for(i in 2:a) {
print(matrixsumm2[1]+matrixsumm2[i]) }
#We obtain the variance covariance matrix of the treatments.
print("Variance Covariance Matrix of Treatments")
```

```
covmatrix=as.matrix(slot(summ, "vcov"))
print(covmatrix)
#We obtain the standard errors of the treatment means (same result with
#SAS PROC MIXED).
print("Standard Errors of Treatment Means")
var=rep(NA,a)
sd=rep(NA,a)
var[1]=covmatrix[1,1]
sd[1]=sqrt(var[1])
print(sd[1])
for (i in 2:a) {
var[i]=covmatrix[1,1]+covmatrix[i,i]+2*(covmatrix[1,i])
sd[i]=sqrt(var[i])
print(sd[i]) } }
```

APPENDIX C

FINDING MISSING VALUES OF A DATA EXAMPLE BY USING AMELIA2 PROCEDURE

```
ff=function() {
# We read the data from the file "program2-data.txt"
Data=read.table("program2-data.txt",header=TRUE)
# Library MASS and Amelia package are needed.
library(MASS)
install.packages("Amelia")
require(Amelia)
#The Amelia procedure is performed based on the default value of m.
a.out = amelia(Data,m =5)
#In the program2-data.txt we have four missing values. Here each k value
# corresponds to one missing cell. We first assign them to be zero to introduce
# them to R.
k1=0
k2=0
k3=0
k4=0
# We take the sum of five estiamated values of the missing cells. In this case,
# the numbers 2,8,9 and 15 correspond to the positions of missing values in
# the dataset.
for (i in 1:5) {
k1=k1+a.out$imputations[[i]]$response[2]
k2=k2+a.out$imputations[[i]]$response[8]
k3=k3+a.out$imputations[[i]]$response[9]
k4=k4+a.out$imputations[[i]]$response[15] }
#We take the mean of five values for each missing cell and print them on
# R console.
k1=k1/5
k2=k2/5
k3=k3/5
k4=k4/5
print(k1)
print(k2)
print(k3)
print(k4) }
```

APPENDIX D

THE BIBD DATA EXAMPLE IN APPENDIX B AND APPENDIX C

blck	trt	response
1	1	73
1	2	NA
1	3	73
1	4	75
2	1	74
2	2	75
2	3	75
2	4	NA
3	1	NA
3	2	67
3	3	68
3	4	72
4	1	71
4	2	72
4	3	NA
4	4	75

Note: The balanced incomplete block design data example are obtained from (Montgomery,2005,p.146)