

# Exercise 6

PLSCI 7201

9/14/2020

## Summary

In this exercise we will learn how to analyze an augmented design using a mixed model. An augmented design is used to evaluate many, many treatments. There are a small number of treatments that are replicated are replicated multiple times, and many more treatments that are only observed once. These designs are often used in in plant breeding to screen a large number of lines (each line is a treatment). I will use the term “check” to indicate the treatments that have multiple replicates and “entries” to indicate in replicated treatments. One way to think of this experiment is that we have a randomized complete block design for the check lines and each block is augmented with unreplicated entries. We can get an initial estimate of the experimental error from the replicated check lines, which can be further improved by using the entries (if entry is considered a random effect).

We will look at two ways to analyze this experiment. The first (M1) is where we treat all checks, as well as the overall average of entries, as a fixed effect and the entries as a random effect. The second approach (M2) we create a variable that specifies that the lines come from two groups, checks and entries. We include a fixed term in the model for this group effect. The lines in each group are specified as separate random effects (one random effect for all the check lines and one random effect for all the entry lines). Fitting these models requires us to code a few new variables in the data. A small example is provided below that shows how to recode the variables and fit each model. Burgueño et al, (2018) describes these designs in detail and discusses the advantages and disadvantages of each. This text is part of a very nice book on experimental design. A free PDF can be downloaded through the Cornell libraries.

## Simple example

The data below consists of 25 observations. The three check lines are indicated with “Ck” and are each replicated three times. The unreplicated entries start with “En”.

```
library(asreml)

M1data <- readRDS("Aug_exData.RDS")

M1data
```

### Example: Model 1

In this model we treat all checks and the average of the entry lines as a fixed effect, and only the entries as a random effect. If we ignore all other components of the model, except for the terms specified the model is

$$\hat{y}_{ij} = \mu_i + u(\mu)_{ij} \quad i = (1, 2, \dots, C + 1); \quad j = (1, 2, \dots, E)$$

Here,  $\mu_i$  is the average for each check and the the average of all entries,  $u(\mu)_{ij}$  is the random effect for the  $j$ th entry nested in the  $i$ th level of the fixed effect.  $C$  is the number of checks,  $E$  is the number of lines. Note that there will be four levels for the fixed factor. We will be getting BLUPs for only the entries.

We will need to create a column in the data that allows us to fit this fixed effect. This column will be a factor with four levels (Ck1, Ck2, Ck3, and Ent). All observations from entries will be assigned to the Ent level. We can create this column using the code below.

```
ckLines <- c("Ck1", "Ck2", "Ck3")

M1data$Chks <- as.character(M1data$Line)
M1data$Chks[!M1data$Line %in% ckLines] <- "Ent"
```

What we're doing here is creating a new column called Chks `M1data$Chks <- as.character(M1data$Line)`. Initially this column is just the names of each line. For any line that is not a check (i.e. not in the vector `ckLines`), we replace those values in the column with "Ent" (`M1data$Chks[! M1data$Line %in% ckLines] <- "Ent"`).

Now we'll create a new column that will allow us to get BLUPs for only the entries. We'll create a dummy variable (0, 1). Lines that are not an entry will be assigned 0 and all entries will be 1. If we specify an interaction between this dummy variable and the column Lines we should only get BLUPs for observations with 1 for the dummy variable. This column will be called new.

```
M1data$New <- NA
M1data$New <- ifelse(M1data$Chks == "Ent", 1, 0)
```

I created a column called `New` that is initially just NA. Then I fill this column with either 1 or 0 depending on whether the line is a check or entry. This if/else statement (`ifelse`) is telling R to find any values in the column `Chks` that have the value `Ent` and set those to 1. All other values will be 0 (`M1data$Chks == "Ent"`).

The data looks like this now.

```
M1data
```

We can fit this model in `asreml` using:

```
M1data$Line <- as.factor(M1data$Line)
M1data$Chks <- as.factor(M1data$Chks)
M1data$New <- as.numeric(M1data$New)

M1 <- asreml(fixed = Y ~ Chks, random = ~New:Line, data = M1data)
# summary(M1)$varcomp
```

Notice we estimate the variance component for the entries `New:Line`. We can check that we're getting BLUPs only for the entries.

```
M1$coefficients$random
```

We get BLUEs for each check line and the overall mean for the entries.

```
M1$coefficients$fixed
```

We can also test the hypothesis of whether the BLUEs for all checks and the overall entry BLUE are equal.

```
wald(M1)
```

## Example: Model 2

This model differs from the one above in that we are fitting two random effects: a random effect for check and a random effect for entries. This will estimate variance components each group. In addition, we will fit a fixed effect that represents the overall mean for each group.

$$\hat{y}_{ij} = \mu_i + u(\mu)_{ij} \quad i = (1, 2) \quad j = (1, 2, \dots, C) \quad \text{if } i = 1 \quad j = (1, 2, \dots, E) \quad \text{if } i = 2$$

Here,  $\mu_i$  is the average for all checks ( $i = 1$ ) and the the average of all entries ( $i = 2$ ),  $u(\mu)_{ij}$  is the random effect for the  $j$ th line nested in the  $i$ th level of the fixed effect.  $C$  is the number of checks,  $E$  is the number of lines. Note that there will be two levels for the fixed factor. We will be getting BLUPs for entries and BLUPs for checks. Each type of line (check or entry) will have its own variance component.

We will modify the data from the previous example to create some new variables for this model. We will need a variable that specifies the type of line (check or entry). We will need to create two new dummy variables, similar to those in the example above, that will allow us to estimate variance components for all checks and all entries.

First, we'll create the column for the main effect in the model (type of line i.e. check or entry).

```
M2data <- M1data
M2data$Type <- ifelse(M2data$New == 0, "Ck", "Ent")
```

Now we'll create the two dummy variables (DV) for the random effects. The first dummy variable (`ch_DV`) is 1 if the line is a check and 0 if the line is an entry. The dummy variable (`new_DV`) is 1 if the line is an entry and 0 if the line is a check. If we fit an interaction effect between the dummy variable and the variable Line, random effects will only be estimated if the dummy variable is 1.

```
M2data$ch_DV <- ifelse(M2data$Type == "Ck", 1, 0)
M2data$new_DV <- ifelse(M2data$Type == "Ent", 1, 0)
```

The data looks like this now.

```
M2data
```

We can fit this model in `asreml` using:

```
M2data$Line <- as.factor(M2data$Line)
M2data$Chks <- as.factor(M2data$Chks)
M2data$Type <- as.factor(M2data$Type)

M2data$ch_DV <- as.numeric(M2data$ch_DV)
M2data$new_DV <- as.numeric(M2data$new_DV)

M2 <- asreml(fixed = Y ~ Type, random = ~ch_DV:Line + new_DV:Line, data = M2data)
# summary(M2)$varcomp
```

Notice we get variance components for the entries (`new_DV:Line`) and checks (`ch_DV:Line`), and BLUPs for both groups.

```
M2$coefficients$random
```

The means for the two groups.

```
M2$coefficients$fixed
```

We can also test the whether the means for the two groups differ.

```
wald(M2)
```

## Exercise

To get a better sense of how to analyze these designs, I've provided a modified data set from Belamkar et al (2018). These data are from a preliminary yield trial for winter wheat. Two-hundred seventy experimental lines were evaluated in a augmented block design in Alliance, Nebraska. The field was split into ten blocks and three check lines (Camelot, Freeman, and Goodstreak) were included in each block. Each block consists of 27 experimental lines. All experimental lines are only replicated once. We will fit the models described above. I'll provide the code needed to create each of the variables, and will let you fit the models and interpret the results. The data is included in the file "Belamkar\_2018.rds".

### Model 1

We'll start with the model that fits the checks and the overall mean of the entries as fixed effects.

```
vik_1 <- readRDS("Belamkar_2018.rds")  
  
head(vik_1)
```

The first thing to do is to find which lines are checks. In this case we know the checks and know how many times they are replicated, but I will provide this code in case you want to get some more insight into how this can be done for other data sets.

```
lineCnts <- data.frame(table(vik_1$Line)) #count the number of times each line is observed  
  
ckLines <- lineCnts$Var1[lineCnts$Freq > 1]
```

Create the column for the fixed effect.

```
vik_1$Chks <- as.character(vik_1$Line)  
vik_1$Chks[!vik_1$Line %in% ckLines] <- "Ent"
```

Now we'll create a new column that will allow us to get BLUPs for only the entries.

```
vik_1$New <- NA  
vik_1$New <- ifelse(vik_1$Chks == "Ent", 1, 0)  
  
vik_1$Block <- as.factor(vik_1$Block)  
vik_1$Line <- as.factor(vik_1$Line)  
vik_1$Chks <- as.factor(vik_1$Chks)  
vik_1$New <- as.numeric(vik_1$New)  
  
head(vik_1)  
str(vik_1)
```

Fit the model M1 and include a random effect for block.

- **Q1** Do you think there is a difference between checks (and the average of all entries)? Why?
- **Q2** Report the variance component for each random term.
- **Q3** Compute the predicted yield for the line Camelot. How did you calculate this value? What is the average yield for all entries?

## Model 2

Now we'll fit the model that includes a random effect for both checks and entries.

```
rm(vik_1)

vik_2 <- readRDS("Belamkar_2018.rds")

head(vik_2)
```

Getting the check lines again.

```
lineCnts <- data.frame(table(vik_2$Line)) #count the number of times each line is observed

ckLines <- lineCnts$Var1[lineCnts$Freq > 1]
```

Create the column for the fixed effect.

```
vik_2$Type <- NA
vik_2$Type[vik_2$Line %in% ckLines] <- "Chk"
vik_2$Type[!vik_2$Line %in% ckLines] <- "Ent"

head(vik_2)
```

Create the dummy variables for the random effects.

```
vik_2$ch_DV <- ifelse(vik_2$Type == "Chk", 1, 0)
vik_2$new_DV <- ifelse(vik_2$Type == "Ent", 1, 0)

head(vik_2)

vik_2$Block <- as.factor(vik_2$Block)
vik_2$Line <- as.factor(vik_2$Line)
vik_2$Type <- as.factor(vik_2$Type)
vik_2$ch_DV <- as.numeric(vik_2$ch_DV)
vik_2$new_DV <- as.numeric(vik_2$new_DV)

head(vik_2)
str(vik_2)
```

Fit the model M2 and include a random effect for block. (You may get a warning message.)

- **Q1** Do the means of checks and entries differ? Why?
- **Q2** Report the variance component for each random term.
- **Q3** If you wanted to select the highest yielding entry, which would you choose? Why?
- **Q4** What is the predicted yield for Camelot? How did you compute this? How does it compare to the value from M1?