Beyond MLR Lab 2: Randomized Block Designs

Welcome to lab 2 of the Beyond MLR course. In this lab, we will use the R packages `ggplot2`, `dplyr`, `emmeans`, and `lmerTest`. You can use the script below to automatically install and load them using the `pacman` package.

```r
# Install pacman if not already available
if (!require("pacman")) install.packages("pacman")

# Use pacman to install (if needed) and load the required packages
pacman::p_load(dplyr, ggplot2, emmeans, lmerTest)
```

Example: Testing Wound Healing Treatments Across Different Laboratory Facilities

Suppose we are conducting an experiment to compare the effectiveness of three different wound healing treatments (Treatment A, Treatment B, and Treatment C) on cell cultures. The experimental units are the cell culture plates. To account for potential variability in laboratory conditions (such as humidity and temperature), we use a **randomized block design**, with laboratory facility as the blocking factor. Within each laboratory (block), the three treatments are randomly assigned to one cell culture plate each, ensuring that every treatment is represented once within each block. This design helps control for variability between laboratory facilities, allowing us to more effectively isolate the treatment effects.

```r
# Set seed for reproducibility
set.seed(456)

# Define blocks and treatments
blocks <- factor(paste0("Laboratory ", rep(1:5, each = 3)))  # Labs 1 to 5
treatments <- factor(rep(c("A", "B", "C"), times = 5))

# Simulate data
```

```
block_effect <- rnorm(5, mean = 0, sd = 2)   # Random effect for each block
treatment_effect <- c(A = 5, B = 7, C = 6)   # Fixed effects for treatments

# Create data frame
data_block <- data.frame(
  Laboratory = blocks,
  Treatment = treatments,
  WoundHealing = NA
)

# Assign responses
for (i in 1:nrow(data_block)) {
  b <- as.numeric(data_block$Laboratory[i])
  t <- data_block$Treatment[i]
  data_block$WoundHealing[i] <-
    50 + block_effect[b] + treatment_effect[t] + rnorm(1, mean = 0, sd = 1)
}
```

The R code chunk above simulates wound healing observations (measured on a continuous scale, where higher values indicate better healing) for 15 cell culture plates, divided across five laboratory facilities. The results are stored in the data frame `data_block`, which consists of the following three variables:

- `Laboratory`: A factor variable indicating the laboratory facility.
- `Treatmment`: A factor variable indicating the applied treatment.
- `WoundHealing`: A numeric variable representing the degree of wound healing on a continuous scale.

Exploratory Data Analysis

To explore the data and investigate the possible presence of a laboratory effect, we start by creating a scatter plot with the wound healing variable on the x-axis and the laboratory facility variable on the y-axis:

```
ggplot(data_block, aes(x = WoundHealing, y = Laboratory, color = Treatment)) +
  geom_point(size = 4) +
  labs(title = "Wound Healing by Treatment and Laboratory",
       x = "Wound Healing Measure",
       y = "Laboratory") +
  theme_minimal()
```

In addition to the scatter plot, we compute descriptive statistics to summarize the wound healing measures within treatments and within laboratories. This will help us assess variability both across treatments and across laboratory facilities.

```r
# Summarizing data by Treatment
summary_stats_treatment <- data_block %>%
  group_by(Treatment) %>%
  summarise(
    Mean_WoundHealing = mean(WoundHealing),
    SD_WoundHealing = sd(WoundHealing)
  )

# Summarizing data by Laboratory
summary_stats_laboratory <- data_block %>%
  group_by(Laboratory) %>%
  summarise(
    Mean_WoundHealing = mean(WoundHealing),
    SD_WoundHealing = sd(WoundHealing)
  )

# Displaying the produced summary tables
summary_stats_treatment
summary_stats_laboratory
```

> **Question**
>
> What patterns do you observe regarding the effects of treatment and laboratory based on the plot and summary statistics?

A Mixed Effects Model for the Randomized Block Design

In a mixed-effects model, both fixed effects and random effects are used to account for different sources of variation.

In the wound healing example, it is essential to account for the **laboratory facility** because differences between laboratories (such as variations in environmental conditions) could influence the outcome. By including laboratory as a blocking variable, we can more efficiently estimate the treatment effects by controlling for this source of variability. This can be statistically achieved by modeling laboratory as either a fixed effect or a random effect.

While we could model laboratory as a fixed effect, treating it as a **random effect** is preferred in this case for two reasons: first, it better reflects reality, as the laboratories are considered a

random sample of possible laboratory conditions; second, it reduces the number of parameters we need to estimate, making the model more parsimonious. Modeling the lab as random allows us to account for variability between laboratories without estimating a separate effect for each one, which ultimately helps to isolate the treatment effects more effectively.

On the other hand, **treatment** is modeled as a **fixed effect** because we are specifically interested in estimating and comparing the effectiveness of the three particular wound healing treatments (Treatment A, Treatment B, and Treatment C). These treatments are not considered random samples from a larger population of treatments; rather, they are the specific interventions under study. By modeling treatment as a fixed effect, we aim to draw conclusions about the differences between these particular treatments, and their impact on wound healing.

Model Specification

The mixed-effects model for our randomized block design can be specified as:

$$Y_{ij} = \mu + \tau_i + b_j + \epsilon_{ij}$$

where:

- Y_{ij}: The observed wound healing response for treatment i in laboratory j.
- μ: The overall mean response across all treatments and laboratories.
- τ_i: The fixed effect of treatment i, representing the deviation of treatment i from the overall mean μ.
- b_j: The random effect of laboratory j, assumed to be normally distributed with mean zero and variance σ_b^2).
- ϵ_{ij}: The error term, assumed to be normally distributed with mean zero and variance σ^2.

> **i** Notation convention
>
> In mixed-effects models, fixed effects are typically represented by **Greek letters**, while random effects are represented by **Roman letters**. This helps differentiate between the two types of effects in model specification.

Model Estimation

To fit the previously specified mixed effects model, we use the `lmer()` function from the `lmeTest` package:

```
options(contrasts = c("contr.sum", "contr.poly"))  # Effects coding

model_block <- lmer(WoundHealing ~ Treatment + (1 | Laboratory), data = data_block)
summary(model_block)
```

Syntax overview

Let's break down the model formula `WoundHealing ~ Treatment + (1 | Laboratory)`

- `WoundHealing ~ Treatment`: This specifies that `WoundHealing` is the outcome variable and that `Treatment` is included as a fixed effect to estimate differences between the three treatments. Note that the intercept is included by default in the model, so `WoundHealing ~ Treatment` is essentially a shortcut for `WoundHealing ~ 1 + Treatment`. Also remember that with effects coding, the intercept represents the overall mean across all levels of the treatment variable, and the treatment coefficients represent deviations from this mean.

- `(1 | Laboratory)`: This specifies the random effect for `Laboratory`, representing laboratory-specific deviations from the overall mean.

Summary overview

When you run `summary(model_block)`, the output will display:

- **Fixed Effects:** Estimates of the fixed effects
  - **Intercept:** The overall mean wound healing score averaged across all treatments and laboratories.
  - **Treatment:** The differences in wound healing associated with each treatment relative to the overall mean.

- **Random Effects:** Estimates of the variance components
  - **Laboratory - Intercept:** Between-laboratory variability (σ_b^2).
  - **Residual:** Residual variance (σ^2).

> ⚠ Caution: Interpreting p-values in Mixed Effects Models
>
> While the `lmerTest` package provides p-values for fixed effects, it is important to use them with caution, primarily because methods for calculating degrees of freedom, such as Satterthwaite, are approximations that may not always be reliable, particularly in complex models fitted to imbalanced data. Generally, this sensitivity is not a concern for the randomized block design considered in this lab.

Assessing the variance components

Assessing the fixed effects

In addition to providing p-values for the fixed effects estimates, the `lmerTest` package also implements the `anova()` function to test the overall significance of fixed effects in the model. Additionally, the `ls_means()` function can be used to calculate the estimated marginal means for the factors (i.e., variables that are defined as factors in the data frame or the model formula) included in the fixed effects structure of the model. Alternatively, these estimated marginal means can be calculated using the `emmeans()` function from the `emmeans` package, which provides more options and greater flexibility for advanced users.

```
# Obtain ANOVA table for the fixed effects
anova(model_block)

# Calculate estimated marginal means for the Treatment variable
ls_means(model_block)
```

Pairwise comparisons

Since we found a significant treatment effect, we proceed with computing pairwise differences of the estimated marginal means. This can be achieved by adding the argument `pairwise=TRUE` to the call to the `ls_means()` function. Unlike the `emmmeans` package, there is no multiplicity correction for pairwise comparisons in the `lmerTest` package. We therefore use the `p.adjust()` function from the base R `stats` package to compute the Bonferroni corrected p-values (i.e., the original p-values multiplied by the number of comparisons).

```
# Compute pairwise differences of the estimated marginal means
emms <- ls_means(model_block, pairwise=TRUE)
emms

# Retrieve the p-values from the emms object and adjust them
# for multiple testing using the Bonferroni correction
p.adjust(emms$`Pr(>|t|)`, method="bonferroni")
```

> **Question**
>
> Based on the results of the pairwise comparisons, which treatment groups differ significantly?

Model Diagnostics

In mixed-effects models, model diagnostics involve analyzing different types of residuals to evaluate how well the model fits the data and to assess whether key assumptions are satisfied.

The default type of residuals used in these models are the **conditional residuals**, which are calculated based on predicted values that incorporate both fixed effects and estimated random effects. Conditional residuals represent the deviation of the observed data from the model's predictions, accounting for variability from both sources: fixed effects and random effects. They can be considered estimates of the errors ϵ_{ij}, which are assumed to be normally distributed with a mean of zero and constant variance.

To check these assumptions, we can generate diagnostic plots similar to those used in linear regression models:

- Residuals vs. Fitted Plot: This plot helps check for homoscedasticity (constant variance).
- Normal Q-Q Plot: This plot assesses whether the residuals follow a normal distribution.

```
# Extract conditional residuals
residuals_cond <- resid(model_block)

# Residuals vs Fitted
```

```
plot(fitted(model_block), residuals_cond,
     main = "Residuals vs Fitted",
     xlab = "Fitted values",
     ylab = "Residuals")
abline(h = 0, col = "red")

# Normal Q-Q Plot
qqnorm(residuals_cond)
qqline(residuals_cond, col = "red")
```

> **Question**
>
> Do the diagnostic plots suggest any violations of the model assumptions regarding the error term?