Multilevel Modeling Summative

Anonymous Marking Code: Z0195806

2024-03-30

Contents

Part 1 Introduction

1.1 Background of Multisite Trials

1.1.1 Definition of Multisite Trials

Multisite trials are a type of clinical research study where the intervention being tested is administered across multiple sites or locations. These trials are particularly valuable in assessing the effectiveness of an intervention in a broader, more diverse population. By including a variety of settings, such as different hospitals, clinics, or communities, multisite trials can provide more generalizable results, ensuring that the findings are not specific to a single location or population (Youth Endowment Fund 2024).

1.1.2 Relevance for assessing the effectiveness of an intervention

1. **Generalizability**: Multisite trials enhance the external validity of the study findings. By testing the intervention across various demographic and geographic settings, the results are more likely to be applicable to a wider population.
2. **Variability and Robustness**: These trials capture the variability across different sites, which can include differences in implementation, participant characteristics, and contextual factors. This variability helps in assessing the robustness of the intervention's effectiveness.
3. **Standardization vs. Adaptation**: Multisite trials can explore the balance between the standardization of the intervention (to ensure fidelity) and its adaptation to different settings (to ensure relevance). This balance is crucial for interventions that aim to be scaled up or replicated in diverse contexts.
4. **Statistical Power**: Conducting a trial across multiple sites often allows for a larger sample size, which increases the statistical power of the study. This is particularly important for detecting small to moderate effects of interventions.
5. **Complex Interventions**: Many interventions, especially in healthcare, are complex and multifaceted. Multisite trials can provide insights into how different components of the intervention perform across various settings.
6. **Healthcare System Insights**: For interventions implemented in healthcare settings, multisite trials can offer valuable insights into how different healthcare systems or practices impact the effectiveness of the intervention.

1.1.3 Bias of Multisite Trials

Multisite trials enhance the relevance of findings but face challenges like selection bias, variability in implementation, and contextual influences, which can skew results. Mitigating these requires careful site selection, standardization across sites, and statistical techniques like multilevel modeling to ensure the trials' findings are both robust and widely applicable.

1.1.4 Pros and Cons of Multisite Trials

Multisite trials offer enhanced generalizability and statistical power due to their diverse and large participant pools, and can be more resource-efficient through shared infrastructure. However, they also face challenges such as logistical complexities, variability in intervention implementation, regulatory hurdles, potential site-specific biases, and data integration issues. Balancing these pros and cons requires careful planning, standardization of protocols, and sophisticated statistical methods to ensure the reliability and applicability of the findings across varied settings (Mudaranthakam et al. 2021).

1.2 Intro to the MST Dataset

1.2.1 Read the Dataset

```
# ------------------------------------------------------
## clear the environment var area
# rm(list = ls())
## clear all plots
# graphics.off()
## clear the console area
```

```
# cat("\014")
# ------------------------------------------------------------
# install.packages("gridExtra")
# ------------------------------------------------------------
require(lme4)
require(lmerTest)
require(ggplot2)
require(sjPlot)
```

Download the dataset "MST" only once from GitHub and save it to csv files.

```
# ----Download the Data -------------------------------
# MST <-
#   read.csv(
#     "https://andygolightly.github.io/teaching/MATH43515/summative/andy.csv",
#     header = TRUE
#   )
# write.csv(MST, "./Data/MST.csv", row.names = FALSE)
# ------------------------------------------------------------
MST = read.csv("./Data/MST.csv")
dim(MST)
```

```
## [1] 200   9
```

```
head(MST)
```

```
##   ID Hospital Responset1 Responset2 Responset3 Trt Experience Gender Size
## 1  1        1         36         38         38   1        6.8      1    0
## 2  2        1         35         39         39   1        9.1      1    0
## 3  3        1         46         41         41   0        6.0      1    0
## 4  4        1         31         31         40   1        3.7      0    0
## 5  5        1         36         36         39   1       12.1      1    0
## 6  6        1         29         33         36   1       15.8      0    0
```

```
## Show three line table MST with sjPlot::tab_df
# tab_df(MST[1:5, ])
```

The MST dataset encompasses data from a longitudinal multisite trial assessing a stress-coping training program's effectiveness for nurses in 20 hospitals' Accident and Emergency (A&E) departments. It comprises 200 entries across 9 columns, detailing anonymized nurse identifiers (ID), hospital IDs, treatment assignment (Trt) with 0 indicating control and 1 indicating receipt of the training program, nurse experience in years, gender (0 for male, 1 for female), and A&E department size (0 for small, 1 for large). The dataset tracks the program's impact on job-related stress through post-test stress scores (Responset1, Responset2, Responset3) measured at 1, 2, and 3 months post-training, respectively, on a scale from 0 (no stress) to 100 (maximum stress), providing a multifaceted view of the intervention's short-term effects on nurse stress levels across diverse hospital settings.

1.2.2 Identify the variables at each level

Considerations for "Size" Placement: It is no longer true that other variables are identified as different hierarchies. However, it is worth discussing whether the Size is placed on the Nurse level or the Hospital

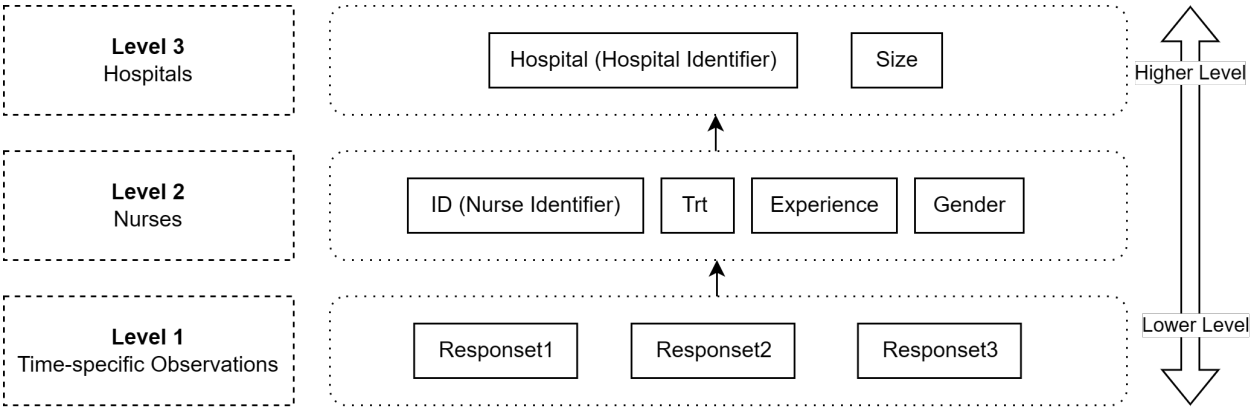


Figure 1: Variables at each level

level. Based on my observations of the Hospital and Size columns in the dataset, there is only one possible Size value for each hospital. For the study's focus on the intervention's impact on stress responses, placing Size at Level 3 is generally more suitable. This approach recognizes department size as a contextual factor at the hospital level that may influence the stress-coping intervention's effectiveness across different settings. It allows the study to explore how the broader environmental and organizational context, such as the scale of A&E departments, moderates the treatment outcomes, providing insights into adapting the intervention for varied hospital environments to enhance its efficacy.

1.2.3 Aims of Multilevel Modeling

The study aims to determine whether the experimental intervention, a stress-coping training program, significantly reduces job-related stress among nurses in A&E departments post-test. Additionally, it seeks to explore how the intervention's impact on stress levels changes over time, providing insights into the sustainability and temporal dynamics of the program's effectiveness.

1.3 Exploratory Data Analysis

1.3.1 Check missing values and imputation

```
# Check if there are any missing values in the entire dataset
any_na <- any(is.na(MST))
print(paste("Are there any missing values in the dataset? ", any_na))
```

```
## [1] "Are there any missing values in the dataset?  FALSE"
```

```
remove(any_na)
```

1.3.2 Summary Statistics

```
# Summary statistics for continuous variables
summary(MST$Experience)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.700   5.675   7.250   8.387  10.700  27.800
```

```
summary(MST$Responset1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   29.00   36.00   39.00   38.94   42.00   48.00
```

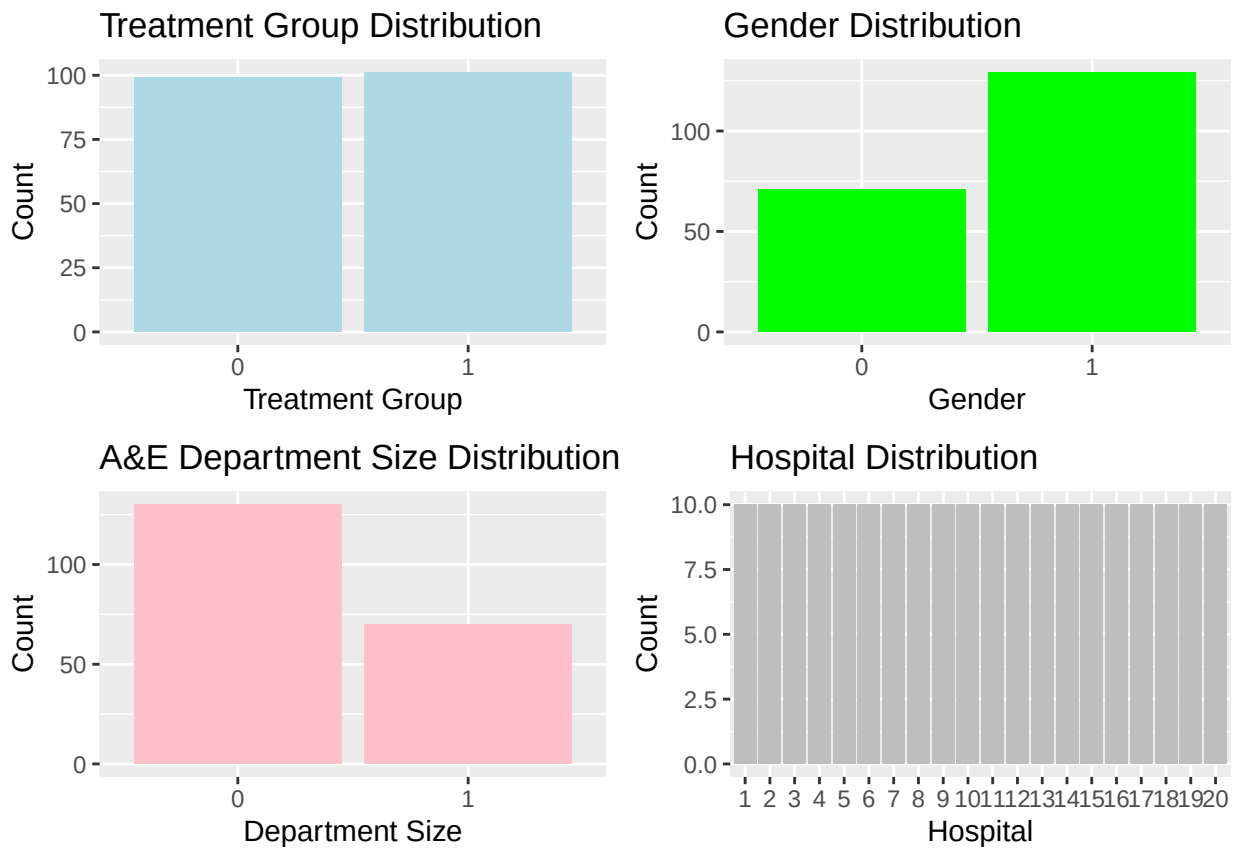
```
summary(MST$Responset2)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    31.0    37.0    40.0    39.7    42.0    47.0
```

```
summary(MST$Responset3)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   33.00   38.00   40.00   40.23   43.00   48.00
```

```
# Load the necessary libraries
library(gridExtra)

# Frequency counts for categorical variables
# Bar chart for Treatment Groups
bar.Trt <- ggplot(data = MST, aes(factor(Trt))) +
  geom_bar(fill="lightblue") +
  labs(x="Treatment Group", y="Count", title="Treatment Group Distribution")

# Bar chart for Gender
bar.Gender <- ggplot(data = MST, aes(factor(Gender))) +
  geom_bar(fill="green") +
  labs(x="Gender", y="Count", title="Gender Distribution")

# Bar chart for Department Size
bar.Size <- ggplot(data = MST, aes(factor(Size))) +
  geom_bar(fill="pink") +
  labs(x="Department Size", y="Count", title="A&E Department Size Distribution")

# Bar chart for Hospital
bar.Hospital <- ggplot(data = MST, aes(factor(Hospital))) +
  geom_bar(fill="gray") +
  labs(x="Hospital", y="Count", title="Hospital Distribution")


grid.arrange(bar.Trt, bar.Gender, bar.Size, bar.Hospital, ncol = 2)
```



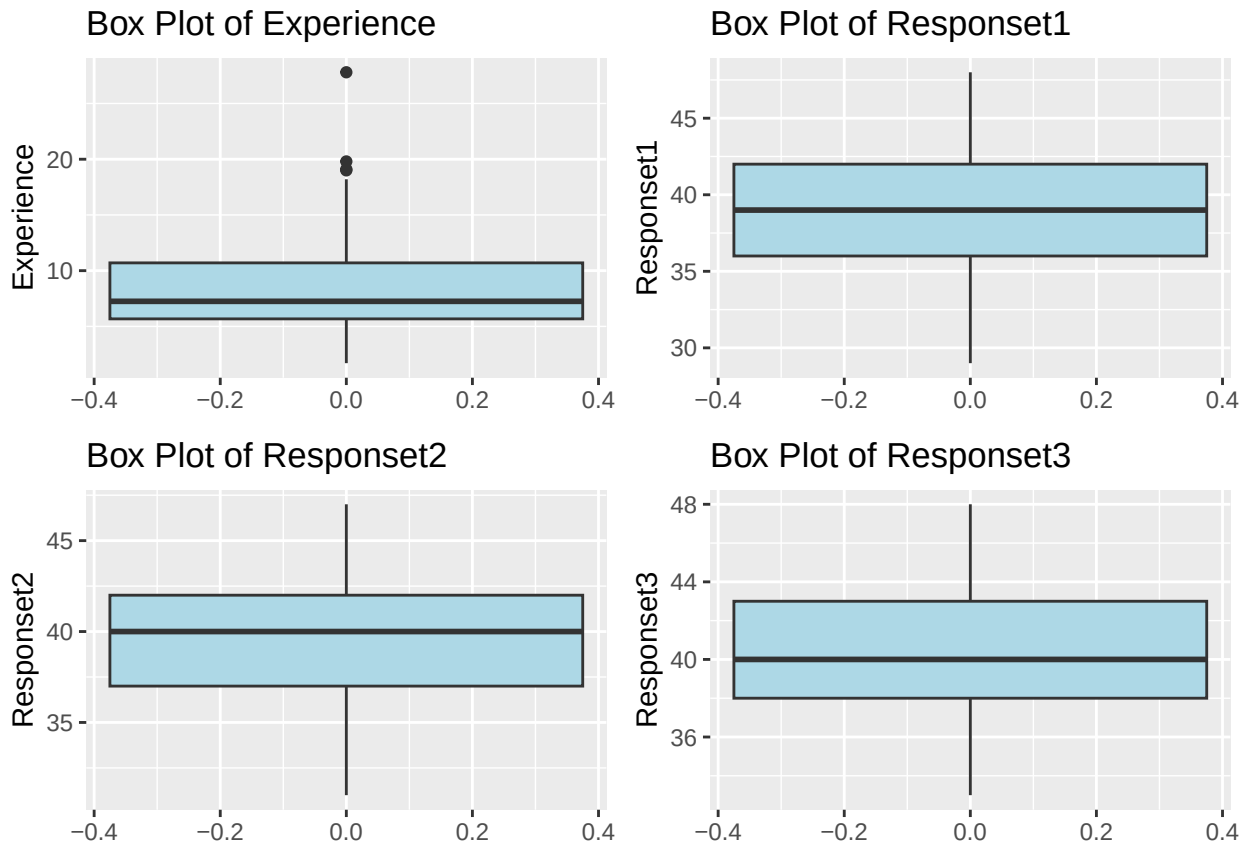

```
remove(bar.Trt, bar.Gender, bar.Size, bar.Hospital)
```

1.3.3 Distribution Analysis

```
# List of variable names to plot
variables_to_plot <-
  c("Experience", "Responset1", "Responset2", "Responset3")

# Initialize lists to store the plots
hist_plots <- list()
box_plots <- list()

# Loop through each variable and create histograms and box plots
for (i in 1:length(variables_to_plot)) {
  var <- variables_to_plot[i]

  # Histogram with Density Plot
  hist_plots[[i]] <- ggplot(MST, aes_string(x = var)) +
    geom_histogram(
      aes(y = ..density..),
      binwidth = 1,
      color = "black",
      fill = "skyblue"
    ) +
```

```
    geom_density(alpha = .2, fill = "#FF6666") +
    ggtitle(paste("Distribution of", var)) +
    xlab(var) +
    ylab("Density")

  # Box Plot
  box_plots[[i]] <- ggplot(MST, aes_string(y = var)) +
    geom_boxplot(fill = "lightblue") +
    ggtitle(paste("Box Plot of", var)) +
    ylab(var)
}

# Arrange the histograms in a 4-graph plot
grid.arrange(grobs = hist_plots, ncol = 2)
```


```
# Arrange the box plots in a 4-graph plot
grid.arrange(grobs = box_plots, ncol = 2)
```



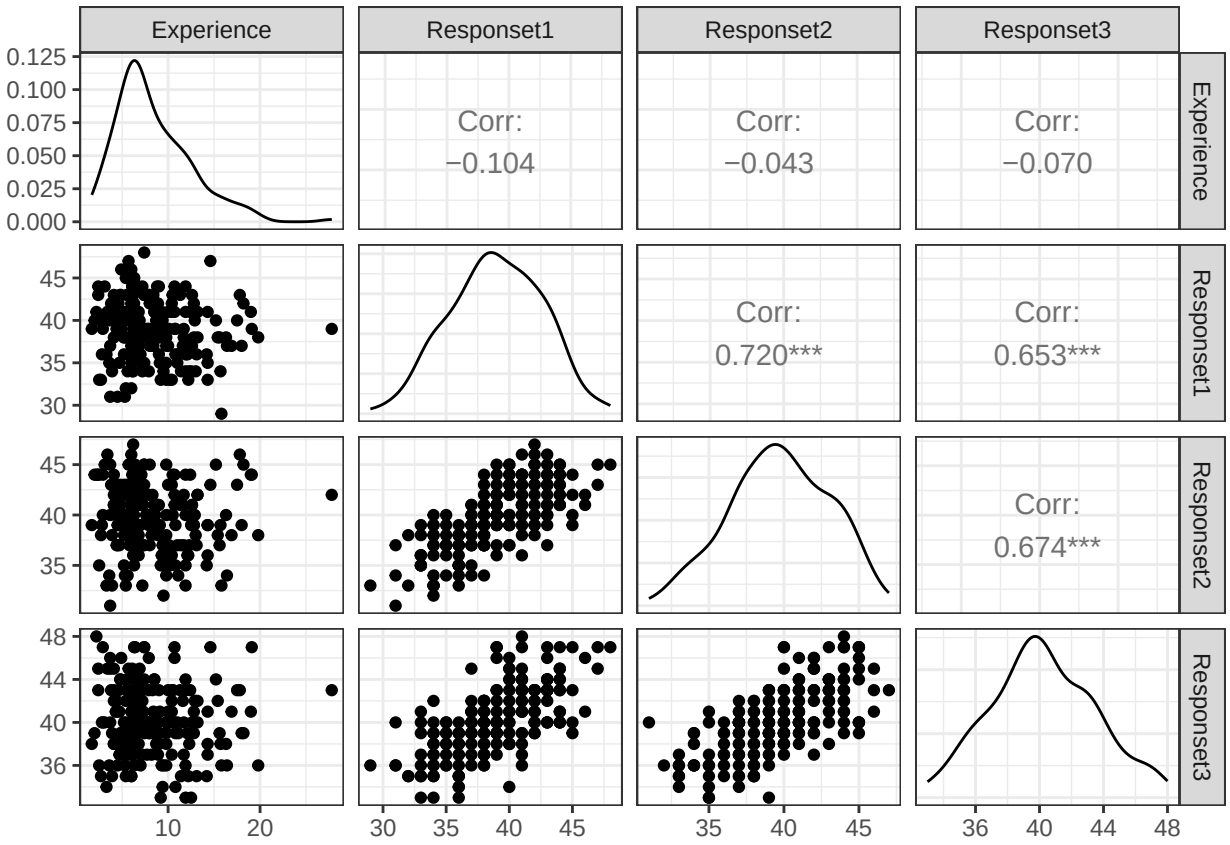

```
remove(hist_plots, box_plots, variables_to_plot, i, var)
```

1.3.4 Correlations

Calculate the correlations between every two variables (X and Y) with the Pearson correlation coefficient:

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

```
# Correlation between two varibales with GGpairs
library("GGally")
ggpairs(MST[, c("Experience", "Responset1",
                "Responset2", "Responset3")]) +
  theme_bw()
```



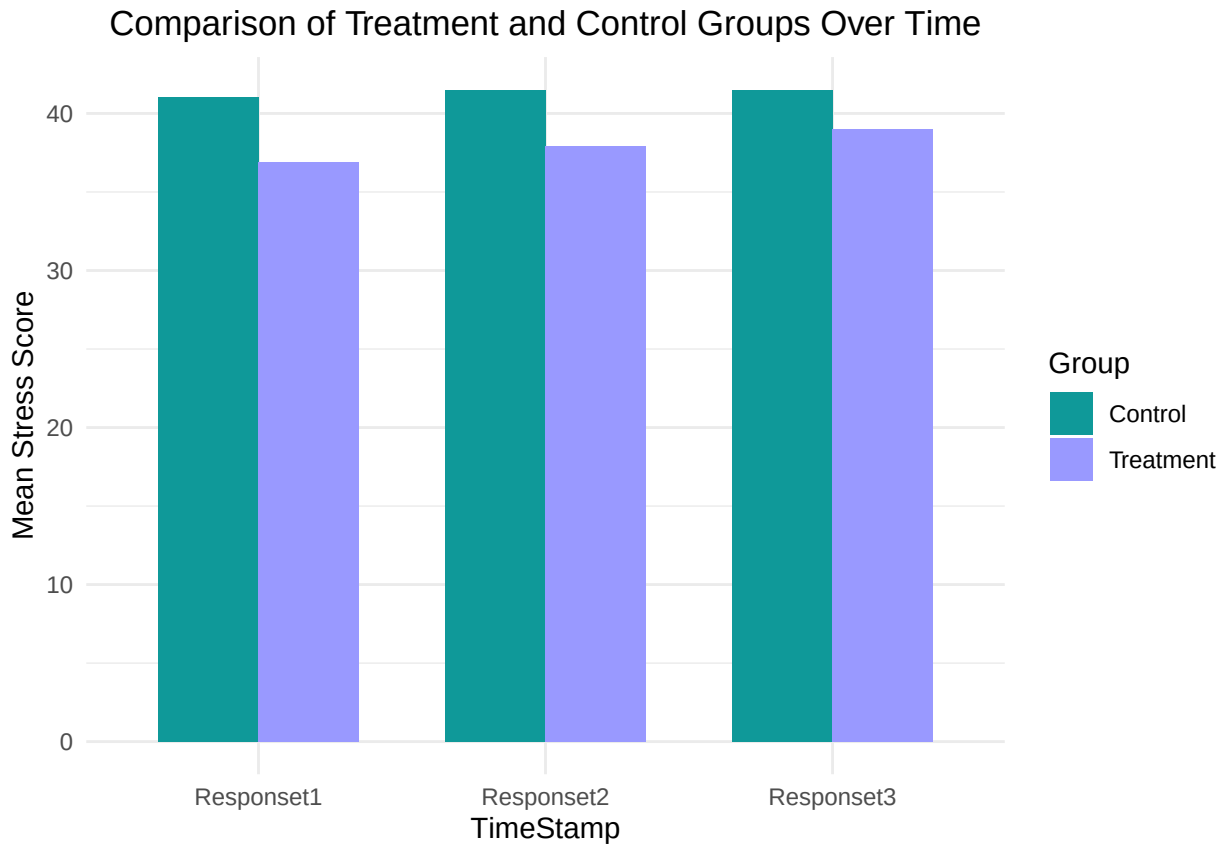
1.3.5 Treatment and Control Group Comparison

In Section 1.3.5-13.7, I focus on the Intervention. For easier making the graphs, I change the dataset into **long format**.

```
# Reshape data to "long format" for easier plotting
library(tidyr)
MST_long <-
  pivot_longer(
    MST,
    cols = starts_with("Responset"),
    names_to = "TimeStamp",
    values_to = "StressScore"
  )

# Convert TimeStamp to a factor for ordered plotting
MST_long$TimeStamp <-
  factor(MST_long$TimeStamp,
         levels = c("Responset1", "Responset2", "Responset3"))
```

```
# Create a bar plot comparing treatment and control groups over time
ggplot(MST_long, aes(x = TimeStamp, y = StressScore,
                     fill = factor(Trt))) +
  geom_bar(
    stat = "summary",
```

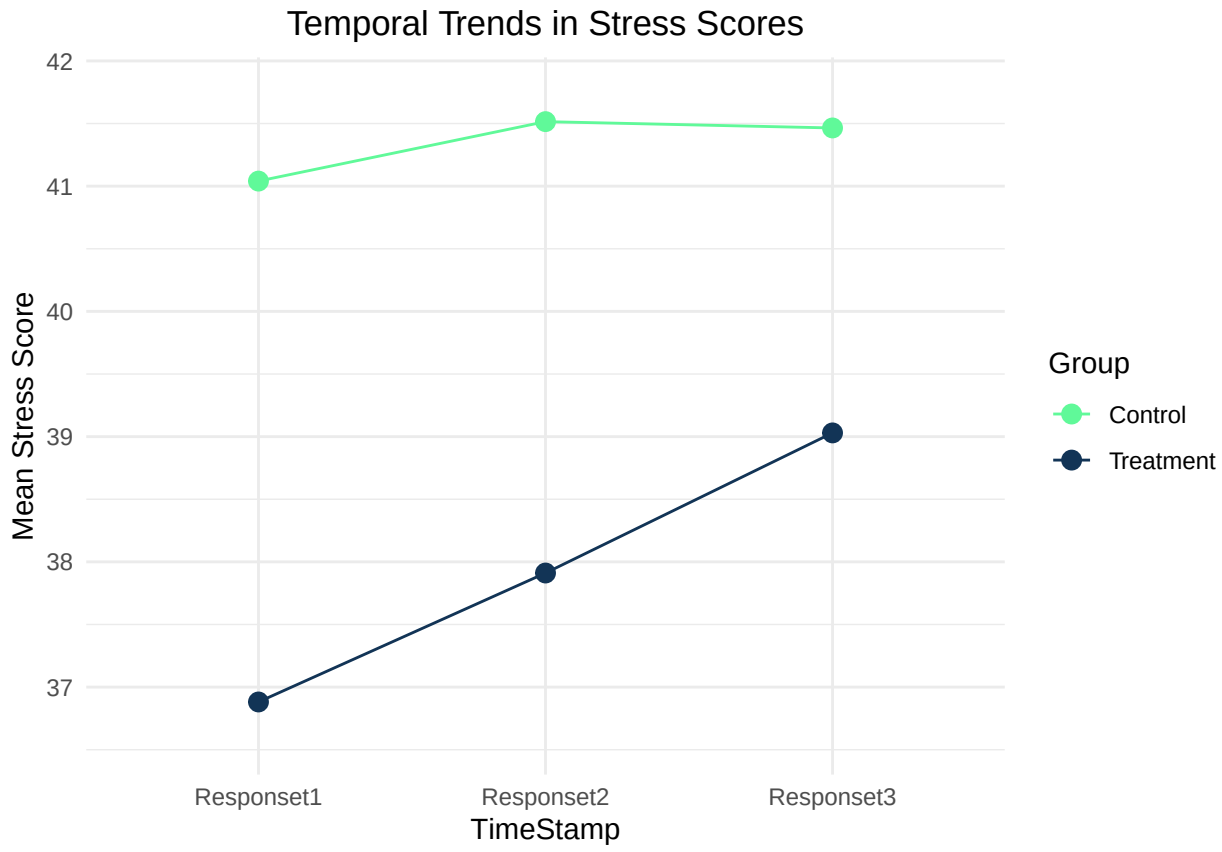
```
    fun = "mean",
    position = "dodge",
    width = 0.7
  ) +
  scale_fill_manual(
    values = c("#0F9999", "#9999FF"),
    labels = c("Control", "Treatment")
  ) +
  labs(title = "Comparison of Treatment and Control Groups Over Time",
       x = "TimeStamp",
       y = "Mean Stress Score",
       fill = "Group") +
  theme_minimal() +
  # Center the plot title
  theme(plot.title = element_text(hjust = 0.5))
```



1.3.6 Temporal Trends

```
# Plotting temporal trends
ggplot(MST_long,
       aes(
         x = TimeStamp,
         y = StressScore,
```

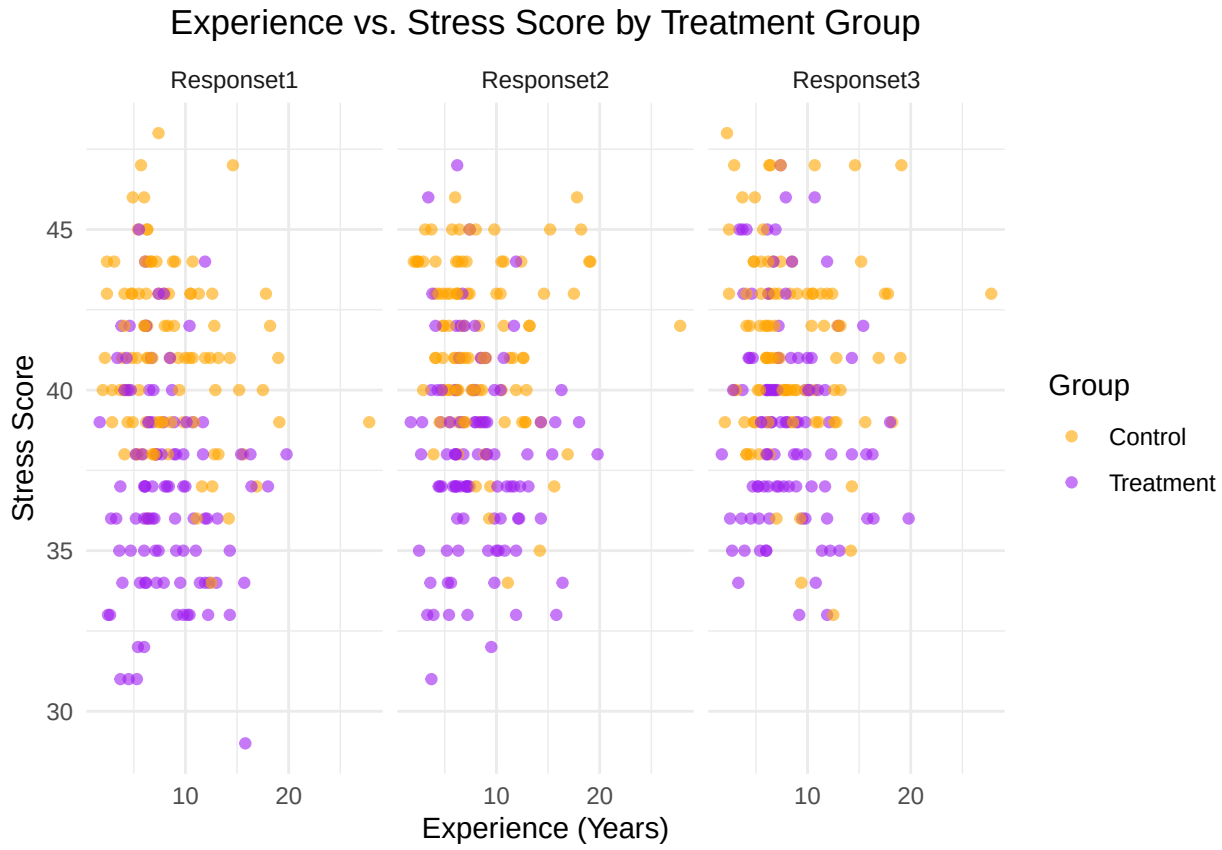
```
        group = Trt,
        color = factor(Trt)
      )) +
  geom_line(stat = "summary", fun.y = "mean") +
  geom_point(stat = "summary", fun.y = "mean", size = 3) +
  scale_color_manual(
    values = c("#60F999", "#123456"),
    labels = c("Control", "Treatment")
  ) +
  labs(
    title = "Temporal Trends in Stress Scores",
    x = "TimeStamp",
    y = "Mean Stress Score",
    color = "Group"
  ) +
  theme_minimal() +
  # Center the plot title
  theme(plot.title = element_text(hjust = 0.5))
```



1.3.7 Categorical Variable Analysis

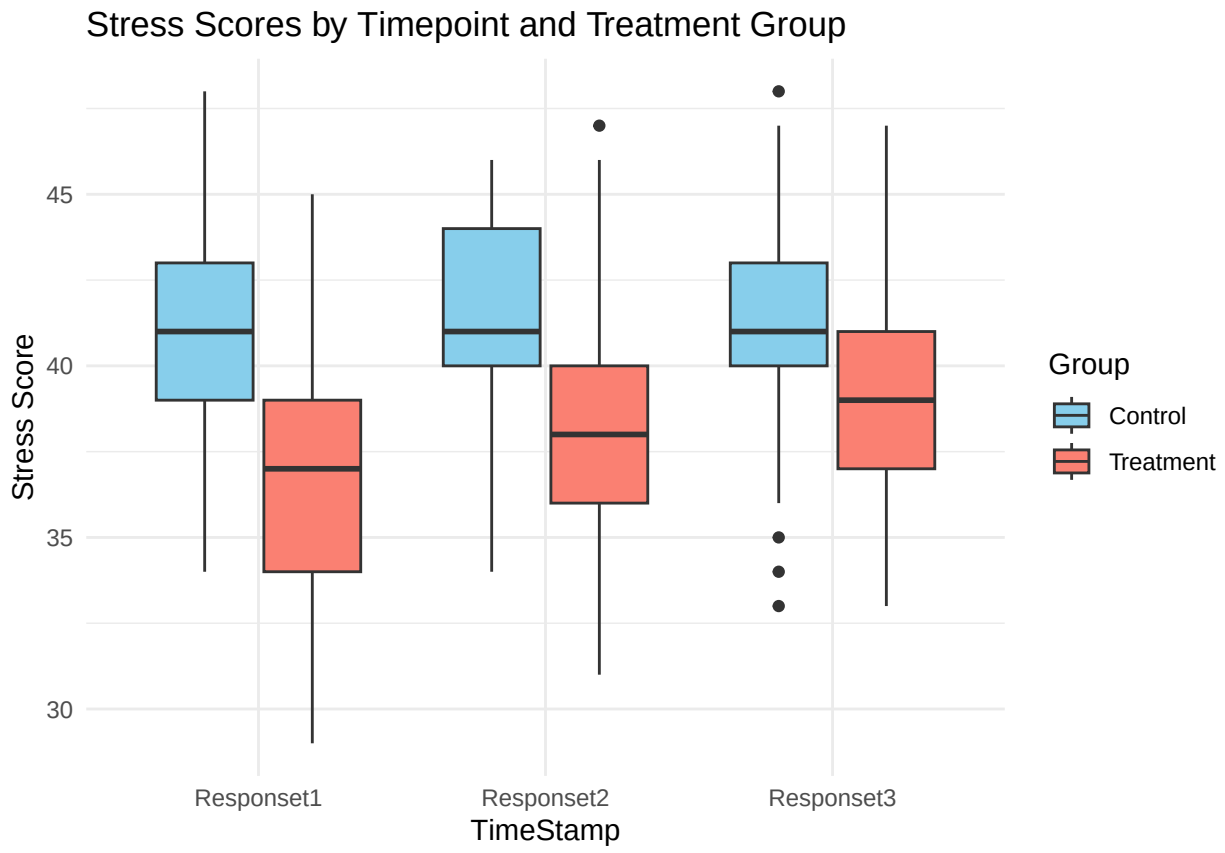
```
# Scatterplot of Experience vs. StressScore, colored by Treatment Group
ggplot(MST_long, aes(x = Experience, y = StressScore, color = factor(Trt))) +
```

```
  geom_point(alpha = 0.6) +
  scale_color_manual(
    values = c("orange", "purple"),
    labels = c("Control", "Treatment")
  ) +
  # Optional: To separate plots by time points
  facet_wrap( ~ TimeStamp) +
  labs(
    title = "Experience vs. Stress Score by Treatment Group",
    x = "Experience (Years)",
    y = "Stress Score",
    color = "Group"
  ) +
  theme_minimal() +
  # Center the plot title
  theme(plot.title = element_text(hjust = 0.5))
```




```
# Create box plots
boxplot_stress <-
  ggplot(MST_long, aes(x = TimeStamp, y = StressScore, fill = factor(Trt))) +
  geom_boxplot() +
  scale_fill_manual(values = c("skyblue", "salmon"),
                    labels = c("Control", "Treatment")) +
  labs(title = "Stress Scores by Timepoint and Treatment Group",
       x = "TimeStamp",
```

```
       y = "Stress Score",
       fill = "Group") +
  theme_minimal()

# Display the plot
print(boxplot_stress)
```




```
remove(boxplot_stress)
```

1.4 Convert the Numeric Data into Factors

Factors explicitly inform the statistical model that the data is categorical, not continuous, such as Hospital, ID, Trt, Gender (male or female), Size (small or large). Their original data structure is integer, I convert them into factor with tidyverse below (in addition, TimeStamp is a string, but our study focus on the treatment impact evolving over time, so I change them into integer as 1, 2 and 3):

```
library(dplyr)
```

```
##
##     'dplyr'

## The following object is masked from 'package:gridExtra':
##
##     combine
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
# Assuming 'data' is your dataset
MST_long = MST_long %>%
  mutate(
    Trt.f = as.factor(Trt),
    Gender.f = as.factor(Gender),
    Hospital.f = as.factor(Hospital),
    ID.f = as.factor(ID),
    Time = as.numeric(TimeStamp),
    Size.f = as.factor(Size)
  )

head(MST_long)
```

```
## # A tibble: 6 x 14
##      ID Hospital   Trt Experience Gender  Size TimeStamp  StressScore Trt.f
##   <int>    <int> <int>      <dbl>  <int> <int> <fct>            <int> <fct>
## 1     1        1     1        6.8      1     0 Responset1          36 1
## 2     1        1     1        6.8      1     0 Responset2          38 1
## 3     1        1     1        6.8      1     0 Responset3          38 1
## 4     2        1     1        9.1      1     0 Responset1          35 1
## 5     2        1     1        9.1      1     0 Responset2          39 1
## 6     2        1     1        9.1      1     0 Responset3          39 1
## # i 5 more variables: Gender.f <fct>, Hospital.f <fct>, ID.f <fct>, Time <dbl>,
## #   Size.f <fct>
```

Part 2 Methods

2.1 Multilevel models for MST Dataset

2.1.1 Multilevel Models Overview

Multilevel models, also known as hierarchical linear models or mixed-effects models, are a class of statistical models designed to analyze data with a nested or hierarchical structure. These models are particularly relevant for datasets where observations are grouped at more than one level, such as our case, repeated stress measurements within nurses who are, in turn, nested within hospitals.

2.1.2 Relevant for the modelling of this data set

Multilevel models (MLMs) are crucial for analyzing the MST dataset's hierarchical structure, effectively addressing the correlations within grouped data. By acknowledging the nesting of measurements within individuals and hospitals, MLMs prevent the underestimation of standard errors that could lead to incorrect significance claims. They meticulously partition variance to reveal the sources influencing outcomes, integrating both fixed effects, such as the overall intervention impact, and random effects that capture variability

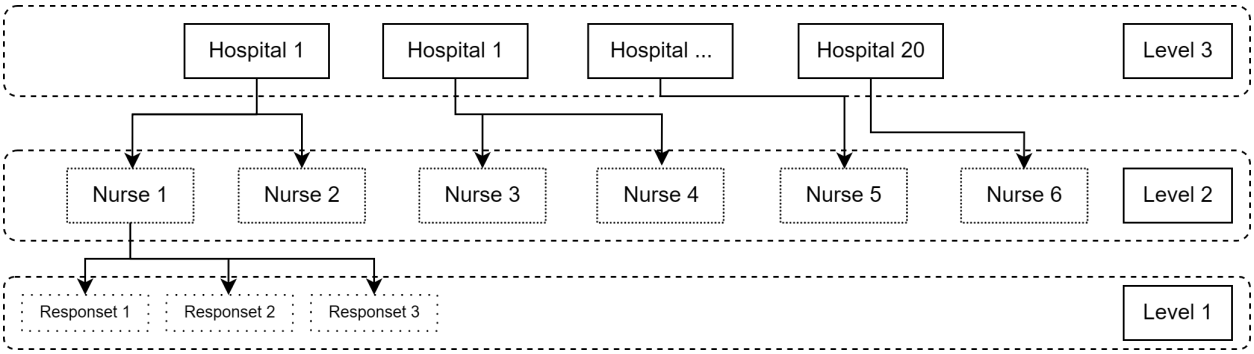


Figure 2: Hierarchical Data Structure

at individual and group levels. This approach enables a comprehensive assessment of interventions in varied contexts. Overlooking the dataset's hierarchical nature risks missing key insights and making inaccurate conclusions, particularly regarding how group characteristics like hospital policies may affect results. Therefore, MLMs are indispensable for ensuring the robustness and relevance of findings in nested data analyses.

2.2 Generalized Linear Mixed Model for MLM

According to fit the model, I choose Generalized Linear Mixed Model (GLMM), the reasons are shown below:

- Limitation of Traditional Linear Model: Unlike traditional linear models that assume normally distributed residuals, GLMM can handle various types of distributions for the response variable (Poisson for count data, etc.), which is essential for modeling the positive integer nature of stress scores. In Section 1.3.2, we found that the Stress Scores of nurses we want to fit are positive integers (count data, not continuous data).
- Hierarchical Structure and Longitudinal Data: A nested structure with multiple levels (Repeated measures within nurses, nurses within hospitals)
- Allows for Fixed and Random Effects: Fixed effects (e.g., treatment or intervention, time points) capture population-level effects that are consistent across all observations, while random effects account for variations at the group level (e.g., differences among nurses or hospitals) that might affect the response variable.

2.3 Decompose variance

Decomposing variance in a multilevel model involves quantifying how much of the total variability in the outcome is attributable to differences at each level of the hierarchy. One key measure used in this context is the Intra-class Correlation Coefficient (ICC), which provides an estimate of the proportion of the total variance that is due to the grouping structure.

Steps for Decompose variance

1. Using the MST_long (long format) to fit an empty GLMM

```
MST_long$ID <- as.factor(MST_long$ID)

## Empty model with longitudinal data
Model.0 <- glmer(
```

```
  StressScore ~ Trt.f * Time + (1 | Hospital) + (1 | Hospital:ID),
  data = MST_long,
  family = gaussian(link = "identity")
)
```

```
## Warning in glmer(StressScore ~ Trt.f * Time + (1 | Hospital) + (1 |
## Hospital:ID), : calling glmer() with family=gaussian (identity link) as a
## shortcut to lmer() is deprecated; please call lmer() directly
```

The warning above is said when using glmer(family=gaussian) is same to lmer(). So the code bellowing I will use lmer() instead.

```
MST_long_test = MST_long
MST_long_test$pred = predict(Model.0)
```

```
MST_long_new = MST_long
MST_long_new$TimeStamp_new = as.numeric(MST_long_new$TimeStamp)
```

```
MST_long_new
```

```
## # A tibble: 600 x 15
##    ID    Hospital   Trt Experience Gender  Size TimeStamp  StressScore Trt.f
##    <fct>    <int> <int>      <dbl>  <int> <int> <fct>            <int> <fct>
##  1 1            1     1        6.8      1     0 Responset1          36 1
##  2 1            1     1        6.8      1     0 Responset2          38 1
##  3 1            1     1        6.8      1     0 Responset3          38 1
##  4 2            1     1        9.1      1     0 Responset1          35 1
##  5 2            1     1        9.1      1     0 Responset2          39 1
##  6 2            1     1        9.1      1     0 Responset3          39 1
##  7 3            1     0        6        1     0 Responset1          46 0
##  8 3            1     0        6        1     0 Responset2          41 0
##  9 3            1     0        6        1     0 Responset3          41 0
## 10 4            1     1        3.7      0     0 Responset1          31 1
## # i 590 more rows
## # i 6 more variables: Gender.f <fct>, Hospital.f <fct>, ID.f <fct>, Time <dbl>,
## #   Size.f <fct>, TimeStamp_new <dbl>
```

Part 3 Analysis

ICC -> 3-level

Init model

Part 4 Discussion of results

Word Count

```r
# install.packages("devtools")
# devtools::install_github("benmarwick/wordcountaddin",
#                          type = "source", dependencies = TRUE)
require(wordcountaddin)
word_count()
```

```
## [1] 1515
```

```r
text_stats()
```

Method	koRpus	stringi
Word count	1515	1398
Character count	10109	10126
Sentence count	117	Not available
Reading time	7.6 minutes	7 minutes

References

Mudaranthakam, Dinesh Pal, Alexandra Brown, Elizabeth Kerling, Susan E. Carlson, Christina J. Valentine, and Byron Gajewski. 2021. "The Successful Synchronized Orchestration of an Investigator-Initiated Multicenter Trial Using a Clinical Trial Management System and Team Approach: Design and Utility Study." *JMIR Formative Research* 5 (12): e30368. https://doi.org/10.2196/30368.

Youth Endowment Fund. 2024. "Multi-Site Trials." https://youthendowmentfund.org.uk/multi-site-trials/.

Appendix

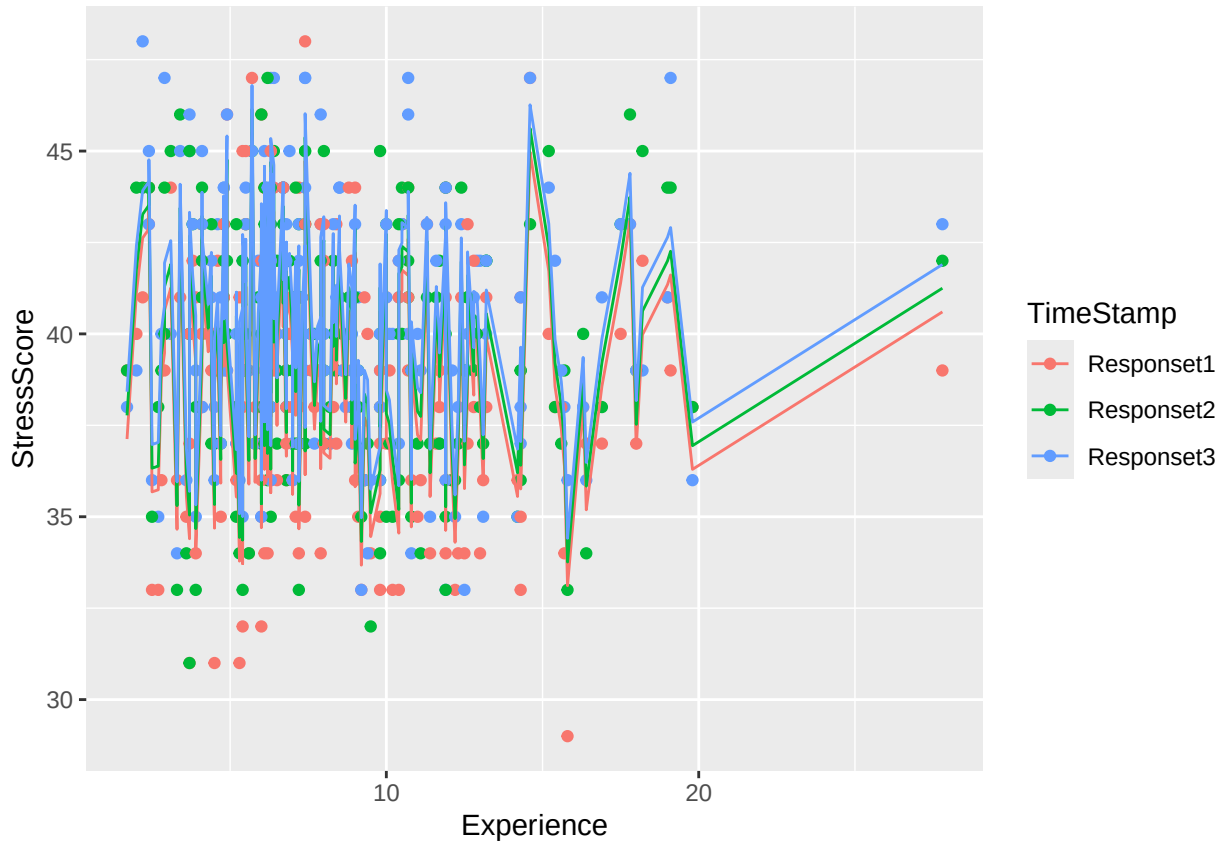
```r
MST_long_new$ID <- as.factor(MST_long$ID)
MST_long_new$Hospital <- as.factor(MST_long$Hospital)
MST_long_new$Time = as.numeric(MST_long$TimeStamp)

model <- glmer(
  StressScore ~ Trt + Time + Experience + Size + (Gender | Hospital:ID) + (1 | Hospital),
  data = MST_long_new,
  family = gaussian(link = "identity")
)

summary(model)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula:
## StressScore ~ Trt + Time + Experience + Size + (Gender | Hospital:ID) +
##     (1 | Hospital)
##    Data: MST_long_new
##
## REML criterion at convergence: 2682.2
##
```

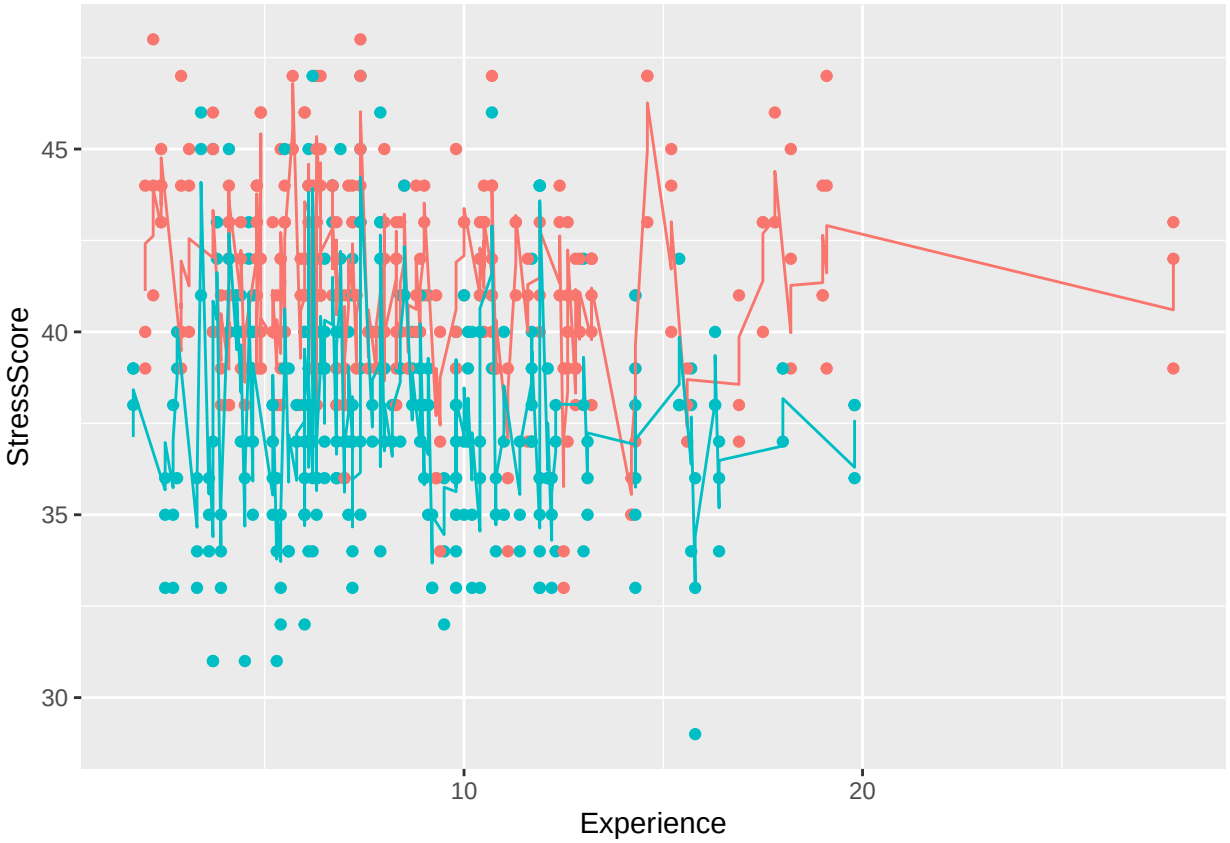
```
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.47188 -0.64256 -0.01712  0.63552  2.95385
##
## Random effects:
##  Groups      Name        Variance Std.Dev. Corr
##  Hospital:ID (Intercept) 2.977    1.725
##              Gender      3.737    1.933    -0.89
##  Hospital    (Intercept) 2.054    1.433
##  Residual                3.683    1.919
## Number of obs: 600, groups:  Hospital:ID, 200; Hospital, 20
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept) 40.23063    0.53460  75.253
## Trt         -3.55364    0.23605 -15.054
## Time         0.64750    0.09596   6.748
## Experience  -0.09587    0.02739  -3.501
## Size         2.61597    0.71016   3.684
##
## Correlation of Fixed Effects:
##            (Intr) Trt    Time   Exprnc
## Trt        -0.254
## Time       -0.359  0.000
## Experience -0.445  0.057  0.000
## Size       -0.472  0.018  0.000  0.003
```

```
MST_long_new$Pred = predict(model)
```

```
ggplot(MST_long_new, aes(Experience, StressScore)) +
  geom_point(aes(Experience, StressScore, col=TimeStamp)) +
  geom_line(aes(x=Experience, y=Pred, group=TimeStamp, col=TimeStamp))
```




```
MST_long_new$Trt.f = as.factor(MST_long_new$Trt)
ggplot(MST_long_new, aes(Experience, StressScore)) +
  geom_point(aes(Experience, StressScore, col=Trt.f)) +
  geom_line(aes(y=Pred, group=Trt.f, col=Trt.f)) +
  theme(legend.position = "none")
```


StressScore
45
40
35
30
10
20
Experience