Final Project
PSTAT122: Design and Analysis of Experiments

Fall 2025

STUDENT NAME

- Alex Dieter (A070X06)
- STUDENT 2 (NetID 2)
- STUDENT 3 (NetID 3)
- STUDENT 4 (NetID 4)

🔥 Due Date

Due Date: Monday, December 8, 2025, 11:59 PM

1 Introduction

Reaction time - Clear statement of the objective or research question. - Brief context or motivation.

2 Experimental Design

- Description of factors and treatment structure.
- Clearly state what you are measuring and the units. Examples: Number of words recalled (count), reaction time (seconds), taste rating (1–5 scale).
- Identify which factors are fixed vs. random.
- Description of design type (CRD, RCBD, factorial, etc.).
- Explain how randomization, replication, and (if used) blocking were implemented.
- Sample size: Provide number of observations per condition. Guideline: 5–10 per treatment for CRD, 3–5 blocks for RCBD, total feasible within 1 hour.

(You are encouraged to explore more resources for determining the sample size)

3 Data Collection

- **Procedure**: Describe how and when the experiment was conducted (e.g., location, date, steps taken).
- **Challenges/Adjustments**: Mention any difficulties or changes made during data collection (e.g., technical issues, time adjustments).
- **Data Presentation**: Display the collected data in tables or graphs, summarizing key measures like mean and standard deviation..

4 Analysis

- **Exploratory Data**: Start with basic statistics (mean, SD) and visualizations (e.g., boxplots) to understand the data.

- **Hypothesis Testing**: Test your hypothesis with an appropriate statistical test (e.g., ANOVA).

- **Tables, Figures, & Code**: Include key results (ANOVA table, post-test) and relevant `R` code excerpts where needed.

- Use `R` to analyze the data.

5 Conclusions

- Summarize key findings.
- Comment on limitations and possible improvements.

6 References

(If needed.)

7 Appendices

- `R` code.

Setting up Randomization part

```r
library(dplyr)

set.seed(122)

subjects <- 1:4
lights_levels <- c("off", "on")
music_levels  <- c("no", "yes")

# full factorial 2x2
treatments <- expand.grid(
  lights = lights_levels,
  music  = music_levels
)

# function: randomize the 4 treatments for one subject
assign_order <- function(subject_id) {
  treatments %>%
    mutate(
      subject = subject_id,
      order = sample(1:4)  # randomly permuted order for this subject
    )
}

# apply to all 4 subjects
random_schedule <- bind_rows(
  assign_order(1),
  assign_order(2),
```

```
    assign_order(3),
    assign_order(4)
) %>%
    arrange(subject, order)

random_schedule
```

```
   lights music subject order
1      on    no       1     1
2     off   yes       1     2
3      on   yes       1     3
4     off    no       1     4
5     off   yes       2     1
6      on   yes       2     2
7     off    no       2     3
8      on    no       2     4
9     off   yes       3     1
10    off    no       3     2
11     on    no       3     3
12     on   yes       3     4
13     on    no       4     1
14    off   yes       4     2
15    off    no       4     3
16     on   yes       4     4
```