Survey Data Analysis Report: Influence of Background Music on Study Performance (Enhanced)

Group 39

June 26, 2025

Contents

1 Introduction

In this report, we want to analyze whether the genre of background music has an impact on students' study performance. The hypothesis is that studying with low-tempo instrumental music helps complete comprehension tasks more accurately and quickly compared to studying with lyrical music or in silence. To test this, we have prepared three survey questions as follows:

- Q1: Type of background music while studying (None, Instrumental, Music with lyrics, Other).
- Q2: Focus level while studying with preferred music (1 to 10 scale).
- Q3: Time to complete a typical comprehension assignment (in minutes).

This enhanced analysis uses R for data cleaning, comprehensive exploratory data analysis (EDA), descriptive inference, analytic inference, and crucial assumption checks for statistical tests. Key R packages used include `dplyr`, `ggplot2`, `knitr`, `readxl`, `car`, `ggpubr`, and `broom`.

2 Data Cleaning

The dataset, is in Exel format (`group39.xlsx`), contains demographic and survey responses. Data cleaning steps address inconsistencies, missing values, and outliers.

- **Renaming Columns:** Columns are renamed for clarity (e.g., DemographicAnswer.1 to Gender, Answer.1 to MusicType).
- **Handling Missing Values:** Rows with missing `MusicType` are removed, as this is critical for RQ1.
- **Outlier Removal:** For `TimeToComplete` (Q3), extreme values (e.g., 99999999 minutes) are treated as outliers and removed. Values above 360 minutes (6 hours) are considered unrealistic for a typical assignment. This helps ensure the statistical analysis is not skewed by erroneous entries.
- **Type Conversion:** `FocusLevel` (Q2) is converted to numeric, and `TimeToComplete` is cleaned to handle non-numeric entries.

```r
library(readxl)
library(dplyr)
data <- read_excel("group39.xlsx")

# renaming columns
data <- data %>%
  rename(Gender = DemographicAnswer.1,
         Age = DemographicAnswer.2,
         Education = DemographicAnswer.3,
         MusicType = Answer.1,
         FocusLevel = Answer.2,
         TimeToComplete = Answer.3)

# removing the rows which has missing MusicType
data <- data %>% filter(!is.na(MusicType))

# converting focusLevel to numeric
data$FocusLevel <- as.numeric(as.character(data$FocusLevel))

# cleaning TimeToComplete: handling ranges and outliers
# Convert "10-15" to "12.5" and then to numeric
data$TimeToComplete <- gsub("10-15", "12.5", data$TimeToComplete)
data$TimeToComplete <- as.numeric(as.character(data$TimeToComplete))

# removing extreme outliers
data <- data %>% filter(TimeToComplete <= 360 | is.na(TimeToComplete))
data <- data %>% filter(TimeToComplete != 999999999 | is.na(TimeToComplete))

data$MusicType <- as.factor(data$MusicType)
```

3 Exploratory Data Analysis

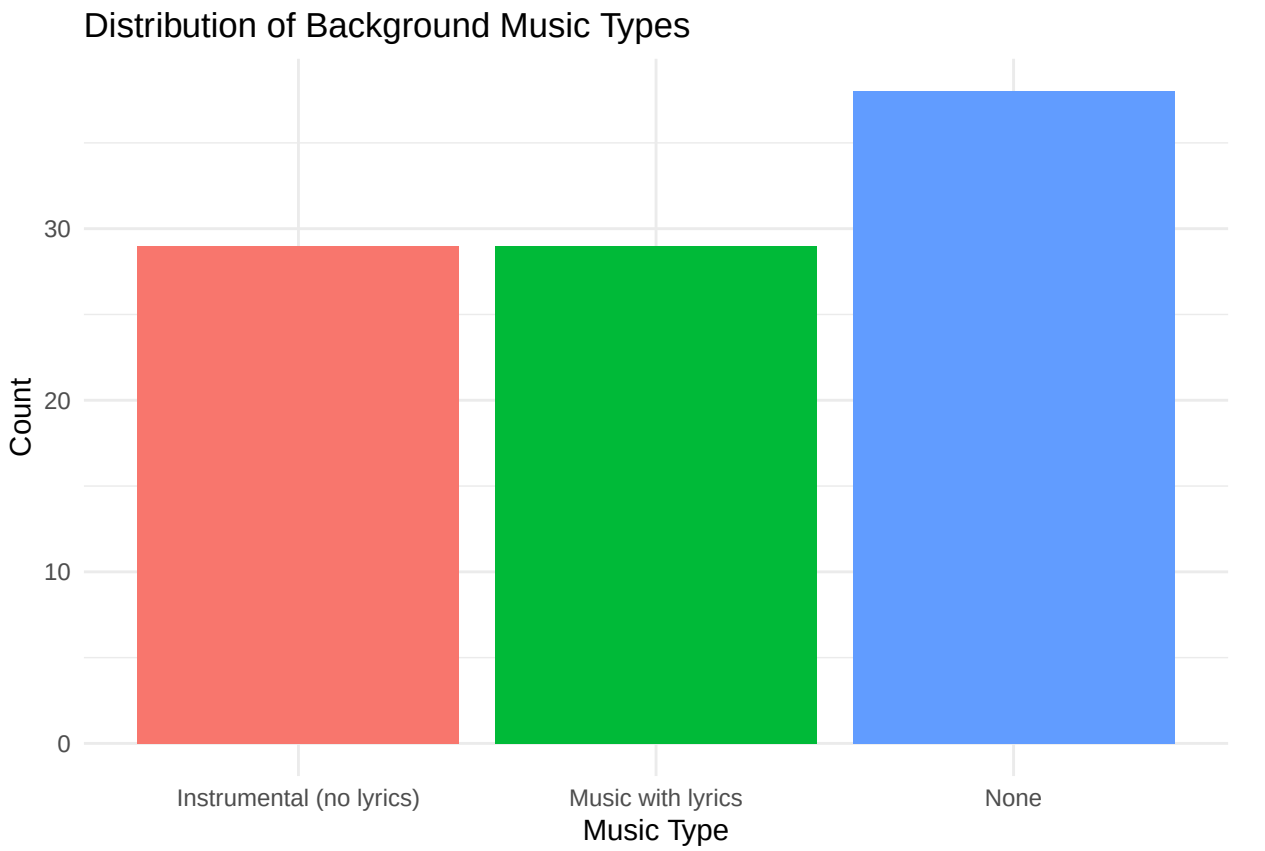
Exploratory Data Analysis examines the distribution of key variables and their relationships to address RQ1, providing more detailed insights into the data's characteristics.

3.1 Distribution of Music Type

A bar plot shows the frequency of each music type.

```r
library(ggplot2)
ggplot(data, aes(x = MusicType, fill = MusicType)) +
  geom_bar() +
  labs(title = "Distribution of Background Music Types",
       x = "Music Type",
       y = "Count") +
```

```
  theme_minimal() +
  theme(legend.position = "none")
```



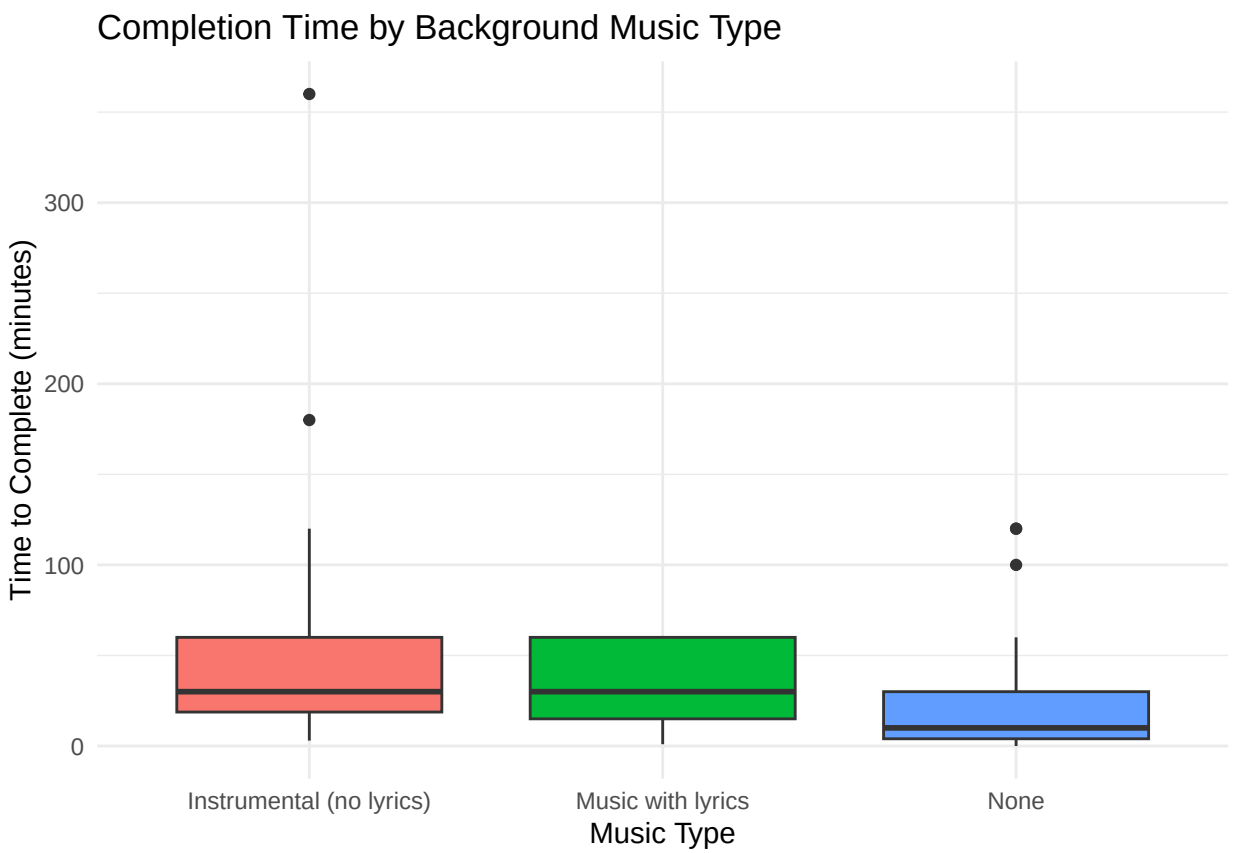
This plot shows the sample size for each music type, which is important for interpreting statistical test results later.

3.2 Completion Time by Background Music Type

A boxplot visualizes TimeToComplete by MusicType to compare completion times across music genres, providing a clear visual representation of central tendency and spread.

```
library(ggplot2)
ggplot(data, aes(x = MusicType, y = TimeToComplete, fill = MusicType)) +
  geom_boxplot() +
  labs(title = "Completion Time by Background Music Type",
       x = "Music Type",
       y = "Time to Complete (minutes)") +
  theme_minimal() +
  theme(legend.position = "none")
```

```
## Warning: Removed 3 rows containing non-finite outside the scale range
## (`stat_boxplot()`).
```



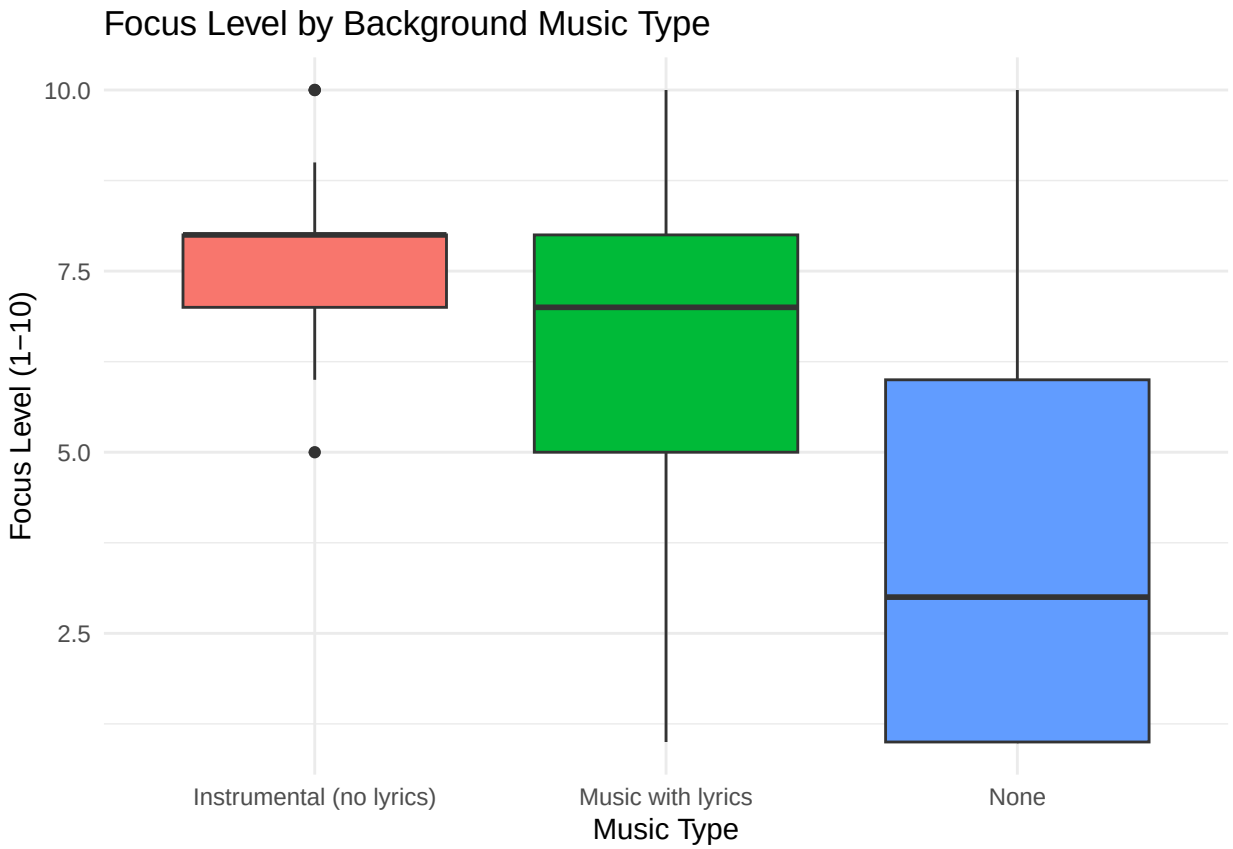
according to the boxplot, when students lesten to the musice, they need more time to complete their task. Type of music are not different a lot.

3.3 Focus Level by Background Music Type

A boxplot for FocusLevel by MusicType helps visualize differences in self-reported focus.

```
ggplot(data, aes(x = MusicType, y = FocusLevel, fill = MusicType)) +
  geom_boxplot() +
  labs(title = "Focus Level by Background Music Type",
       x = "Music Type",
       y = "Focus Level (1-10)") +
  theme_minimal() +
  theme(legend.position = "none")
```

```
## Warning: Removed 1 row containing non-finite outside the scale range
## (`stat_boxplot()`).
```



This plot provides a visual comparison of focus levels across different music types. Instrumental music appears to be associated with slightly higher median focus levels.

3.4 Distribution of Numerical Variables

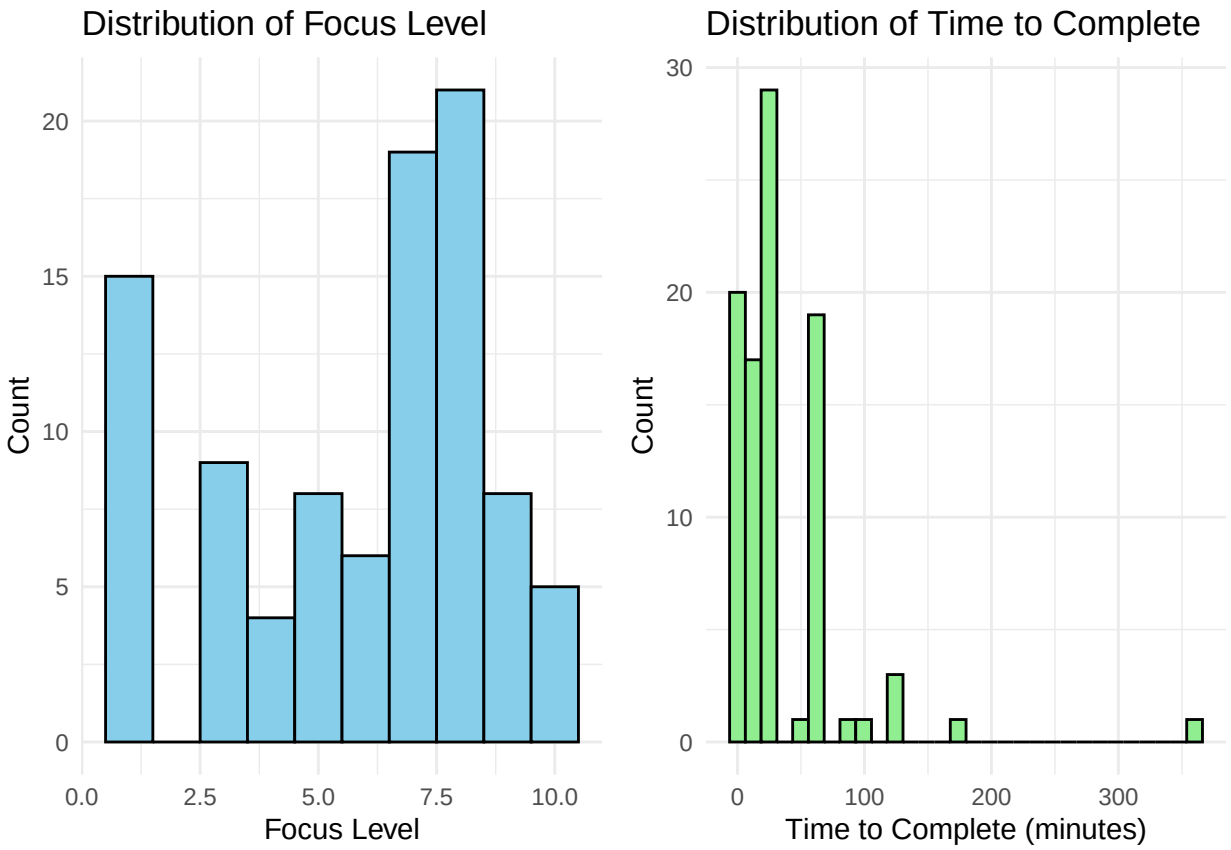
Histograms for FocusLevel and TimeToComplete show their overall distributions.

```
library(cowplot) # For combining plots

p1 <- ggplot(data, aes(x = FocusLevel)) +
  geom_histogram(binwidth = 1, fill = "skyblue", color = "black") +
  labs(title = "Distribution of Focus Level", x = "Focus Level", y = "Count") +
  theme_minimal()

p2 <- ggplot(data, aes(x = TimeToComplete)) +
  geom_histogram(fill = "lightgreen", color = "black") +
  labs(title = "Distribution of Time to Complete", x = "Time to Complete (minutes)", y = "Count") +
  theme_minimal()

plot_grid(p1, p2, ncol = 2)
```

```
## Warning: Removed 1 row containing non-finite outside the scale range
## (`stat_bin()`).
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 3 rows containing non-finite outside the scale range
## (`stat_bin()`).
```



These histograms provide insights into the overall spread and skewness of the numerical variables, which is useful for assessing normality assumptions later.

4 Descriptive Inference

Summary statistics for FocusLevel and TimeToComplete are computed for each MusicType, now presented with more detail including median and interquartile range to complement mean and standard deviation, which are more robust to outliers.

```
library(knitr)
library(dplyr)

summary_stats <- data %>%
  group_by(MusicType) %>%
  summarise(
    N = n(),
    Mean_Focus = mean(FocusLevel, na.rm = TRUE),
    SD_Focus = sd(FocusLevel, na.rm = TRUE),
    Median_Focus = median(FocusLevel, na.rm = TRUE),
    IQR_Focus = IQR(FocusLevel, na.rm = TRUE),
    Mean_Time = mean(TimeToComplete, na.rm = TRUE),
    SD_Time = sd(TimeToComplete, na.rm = TRUE),
    Median_Time = median(TimeToComplete, na.rm = TRUE),
    IQR_Time = IQR(TimeToComplete, na.rm = TRUE)
  )

print(summary_stats)
```

```
## # A tibble: 3 x 10
##   MusicType       N Mean_Focus SD_Focus Median_Focus IQR_Focus Mean_Time SD_Time
##   <fct>       <int>      <dbl>    <dbl>        <dbl>     <dbl>     <dbl>   <dbl>
## 1 Instrument~    29       7.79     1.15            8         1      52.6    71.2
## 2 Music with~    29       6.72     2.15            7         3      33.1    21.6
## 3 None           38       3.68     2.69            3         5      24.2    32.9
## # i 2 more variables: Median_Time <dbl>, IQR_Time <dbl>
```

The enhanced table confirms that instrumental music has slightly higher mean/median focus levels and lower mean/median completion times. The IQR gives a better sense of data spread compared to just standard deviation.

5 ANOVA

We check the normality of the residuals from the ANOVA model. This can be done visually with a QQ plot.

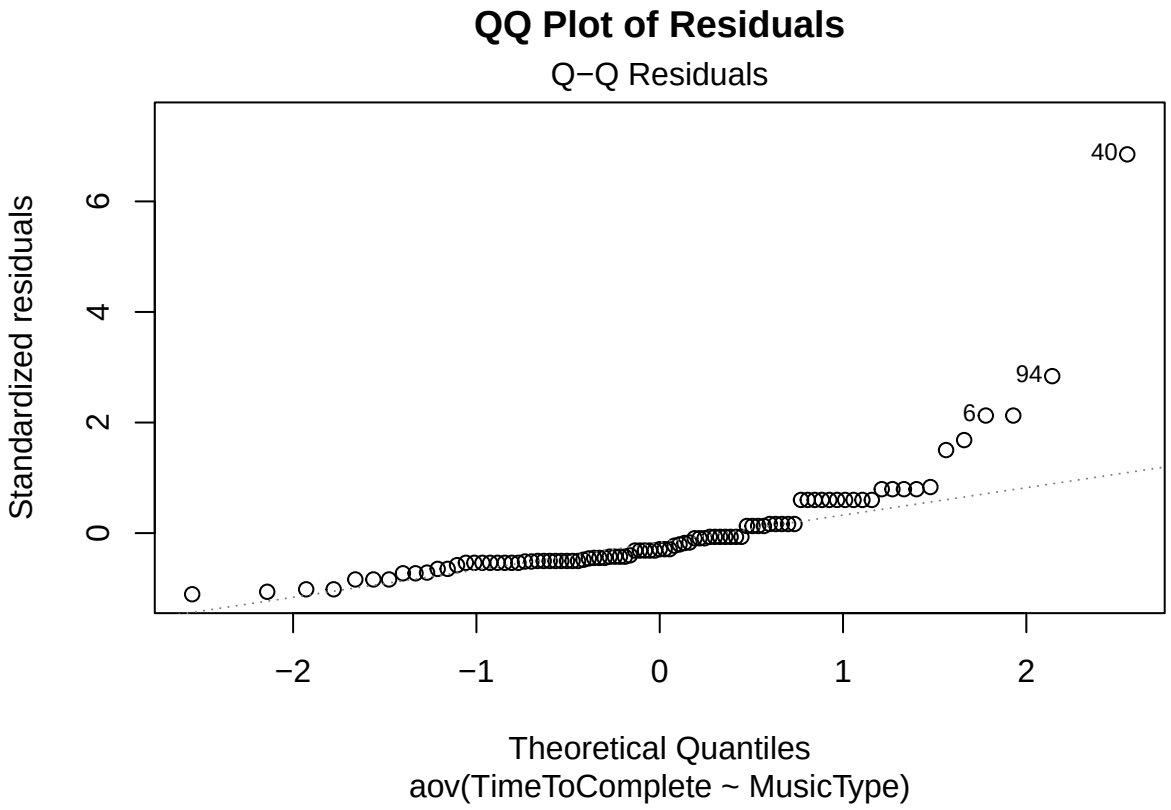
```
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

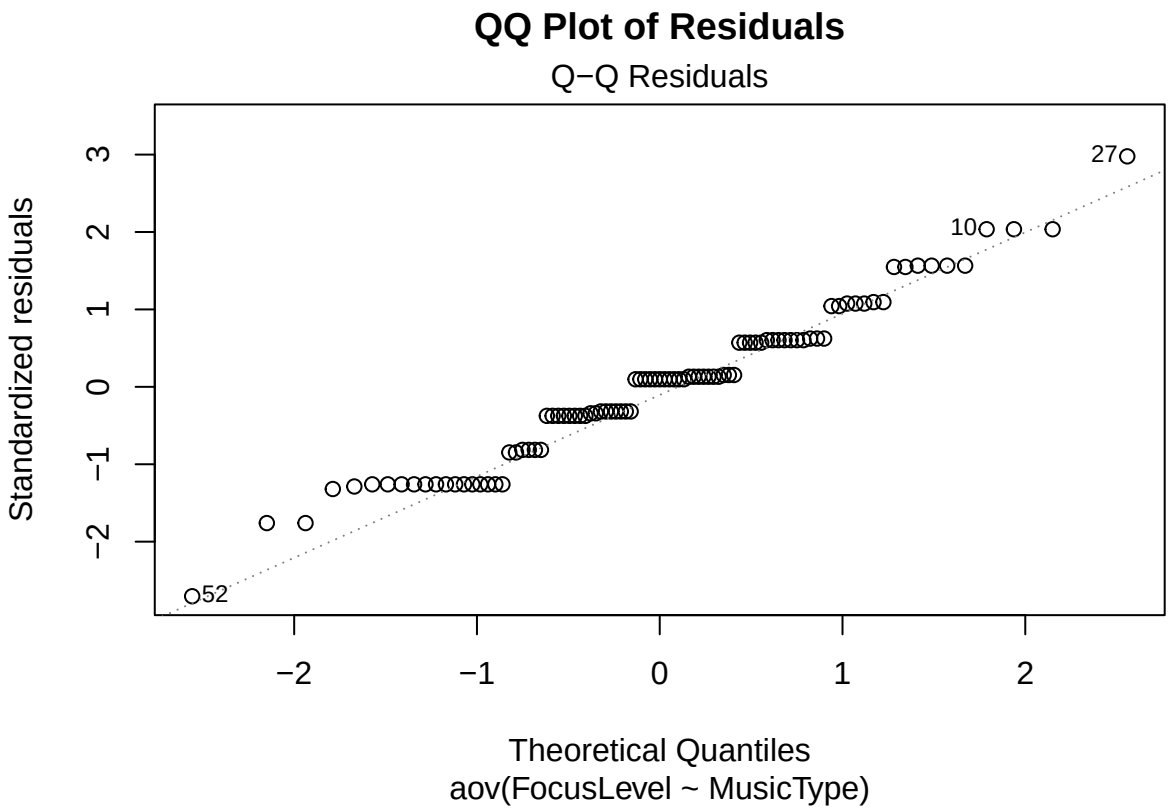
```
# Build the linear model for ANOVA
anova_model <- aov(TimeToComplete ~ MusicType, data = data)

# QQ plot of residuals
plot(anova_model,2)
title("QQ Plot of Residuals")
```




```
print(anova_model)
```

```
## Call:
##    aov(formula = TimeToComplete ~ MusicType, data = data)
##
## Terms:
##                 MusicType Residuals
## Sum of Squares    12969.5  187906.5
## Deg. of Freedom         2        90
##
## Residual standard error: 45.693
## Estimated effects may be unbalanced
## 3 observations deleted due to missingness
```

```
library(car)

# Build the linear model for ANOVA
anova_model_focous <- aov(FocusLevel ~ MusicType, data = data)

# QQ plot of residuals
plot(anova_model_focous,2)
title("QQ Plot of Residuals")
```




```
print(anova_model_focous)
```

```
## Call:
##    aov(formula = FocusLevel ~ MusicType, data = data)
##
## Terms:
##                 MusicType Residuals
## Sum of Squares   306.5612  426.6598
## Deg. of Freedom         2        92
##
## Residual standard error: 2.15351
## Estimated effects may be unbalanced
## 1 observation deleted due to missingness
```

The QQ plot shows that the residuals deviate from the straight line, especially in the upper tail, indicating a violation of the normality assumption. This suggests the residuals are not normally distributed, which can affect the reliability of the ANOVA results. The ANOVA table shows that the between-group variation (MusicType) is relatively small compared to the residual variance.

5.1 Performing ANOVA

```
library(broom)
summary_anova <- summary(anova_model)

# Displaying ANOVA results
kable(tidy(anova_model), caption = "ANOVA Results for Completion Time by Music Type",
             format = "latex", booktabs = TRUE, digits = 3)
```

Table 1: ANOVA Results for Completion Time by Music Type

term	df	sumsq	meansq	statistic	p.value
MusicType	2	12969.5	6484.75	3.106	0.05
Residuals	90	187906.5	2087.85	NA	NA

```
# Post-hoc test if overall ANOVA is significant
if (summary_anova[[1]][["Pr(>F)"]][1] < 0.05) {
  message("Performing TukeyHSD post-hoc test due to significant ANOVA result:")
  tukey_result <- TukeyHSD(anova_model)
  print(tukey_result)
} else {
  message("ANOVA not significant, no post-hoc test needed.")
}
```

```
## Performing TukeyHSD post-hoc test due to significant ANOVA result:
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = TimeToComplete ~ MusicType, data = data)
##
## $MusicType
##                                             diff       lwr        upr
## Music with lyrics-Instrumental (no lyrics) -19.55542 -48.40580  9.2949624
## None-Instrumental (no lyrics)              -28.41270 -55.85066 -0.9747357
## None-Music with lyrics                      -8.85728 -36.02784 18.3132791
##                                               p adj
## Music with lyrics-Instrumental (no lyrics) 0.2444661
## None-Instrumental (no lyrics)              0.0406499
## None-Music with lyrics                     0.7181424
```

The ANOVA tests whether mean completion times differ significantly across music types. A p-value less than 0.05 indicates a statistically significant difference among group means. If significant, the TukeyHSD post-hoc test identifies which specific pairs of music types have significantly different mean completion times. But this results are not reliable. We perform another test.

5.2 Kruskal-Wallis Rank Sum Test

If the assumptions for ANOVA are violated, the Kruskal-Wallis rank sum test is a non-parametric alternative. It tests if there are significant differences in the medians among groups.

```
# Kruskal-Wallis Rank Sum Test
kruskal_result <- kruskal.test(TimeToComplete ~ MusicType, data = data)
print(kruskal_result)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  TimeToComplete by MusicType
## Kruskal-Wallis chi-squared = 11.543, df = 2, p-value = 0.003116
```

```
# If significant, conduct Dunn's test for post-hoc analysis (requires DescTools package)
# Make sure DescTools is installed: install.packages("DescTools")
```

```r
library(DescTools)
```

```
##
## Attaching package: 'DescTools'
```

```
## The following object is masked from 'package:car':
##
##     Recode
```

```r
if (kruskal_result$p.value < 0.05) {
  message("Performing Dunn's post-hoc test due to significant Kruskal-Wallis result:")
  dunn_result <- DescTools::DunnTest(TimeToComplete ~ MusicType, data = data, method="bonferroni")
  print(dunn_result)
}
```

```
## Performing Dunn's post-hoc test due to significant Kruskal-Wallis result:
```

```
##
##  Dunn's test of multiple comparisons using rank sums : bonferroni
##
##                                             mean.rank.diff   pval
## Music with lyrics-Instrumental (no lyrics)      -4.662562 1.0000
## None-Instrumental (no lyrics)                  -21.339286 0.0046 **
## None-Music with lyrics                         -16.676724 0.0372 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

There is a significant effect of music type on task completion time. Specifically, working without music results in faster performance than working with either type of music (instrumental or with lyrics). However, the type of music (instrumental vs. lyrical) does not significantly affect performance between each other.

```r
# Kruskal-Wallis Rank Sum Test
kruskal_result_Focous <- kruskal.test(FocusLevel ~ MusicType, data = data)
print(kruskal_result_Focous)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  FocusLevel by MusicType
## Kruskal-Wallis chi-squared = 36.405, df = 2, p-value = 1.244e-08
```

```r
# If significant, conduct Dunn's test for post-hoc analysis (requires DescTools package)
# Make sure DescTools is installed: install.packages("DescTools")
library(DescTools)
if (kruskal_result_Focous$p.value < 0.05) {
  message("Performing Dunn's post-hoc test due to significant Kruskal-Wallis result:")
  dunn_result_Focous <- DescTools::DunnTest(FocusLevel ~ MusicType, data = data, method="bonferroni")
  print(dunn_result_Focous)
}
```

```
## Performing Dunn's post-hoc test due to significant Kruskal-Wallis result:
```

```
##
##  Dunn's test of multiple comparisons using rank sums : bonferroni
##
##                                             mean.rank.diff    pval
## Music with lyrics-Instrumental (no lyrics)      -11.75862 0.29991
## None-Instrumental (no lyrics)                   -39.12488   2e-08 ***
```

```
## None-Music with lyrics                          -27.36626 0.00015 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The Kruskal-Wallis rank sum test revealed a significant difference in FocusLevel across different MusicType categories, with a p-value of 1.244e-08, indicating that the type of music affects focus levels. Dunn's post-hoc test with Bonferroni correction further clarified that there is no significant difference between music with lyrics and instrumental music (p = 0.29991), but significant differences exist between instrumental music and no music (p = 2e-08) as well as between music with lyrics and no music (p = 0.00015). These findings suggest that the presence of music, regardless of whether it includes lyrics, impacts focus levels differently compared to the absence of music. The negative mean rank differences imply that focus levels tend to be lower when music is present compared to no music, though the exact magnitude depends on the specific context of the data. Overall, this analysis highlights that eliminating music may enhance focus.

\ The mean FocusLevel appears higher with music compared to no music in the summary statistics, yet the Kruskal-Wallis and Dunn's tests suggest lower focus with music. The summary statistics indicate higher mean and median FocusLevel for "Instrumental" and "Music with lyrics" compared to "None," which aligns with the boxplot observation (Section 3.3) of slightly higher median focus with instrumental music. However, the Kruskal-Wallis test on FocusLevel and the subsequent Dunn's post-hoc test reveal a significant difference, with negative mean rank differences (-39.12488 for None vs. Instrumental, p = 2e-08; -27.36626 for None vs. Music with lyrics, p = 0.00015), indicating that FocusLevel is significantly lower when music is present compared to no music, despite the higher means.

This apparent contradiction arises because the mean and median values reflect central tendencies that can be influenced by the distribution's shape, while the Kruskal-Wallis test and Dunn's test assess differences in ranks, which are less sensitive to the exact scale but highlight relative ordering. The histograms and the QQ plot for the ANOVA residuals suggest non-normal distributions and potential outliers, which could explain why the mean FocusLevel is higher with music , while the rank-based tests indicate lower focus relative to "None." The Dunn's test shows no significant difference between "Instrumental" and "Music with lyrics" (p = 0.29991), consistent with the summary statistics' close means, but the significant drop in rank for music groups versus "None" suggests that self-reported focus is lower in the presence of music when considering the entire distribution.

Finally, the Hypothesis: Instrumental music improves focus and performance. is not correct. Based on the analysis, both the one-way ANOVA and the Kruskal-Wallis rank sum test consistently indicate that the type of background music significantly influences the time taken to complete tasks and focous level. Specifically, the findings suggest that students listening to either instrumental music or music with lyrics tend to complete tasks slower compared to those studying in silence. Listening to music also have a negative influence on their focous.

6 Functions

`read_excel()`: Imports data from an Excel file into an R data frame.

`rename()`: Changes column names in a data frame.

`filter()`: Subsets rows in a data frame based on specified conditions.

`as.numeric()`: Converts a variable to numeric type.

`gsub()`: Replaces specific patterns in text strings with new values.

`as.factor()`: Converts a variable to a factor.

`ggplot()`: Initializes a plot object for creating customizable visualizations.

geom_bar(): Creates a bar plot to display counts or summaries of categorical data.

geom_boxplot(): Generates a boxplot to show the distribution and spread of data.

labs(): Adds or modifies titles, axis labels, and other annotations in plots.

theme_minimal(): Applies a clean, minimalistic theme to ggplot visualizations.

geom_histogram(): Plots a histogram to visualize the distribution of a variable.

plot_grid(): Arranges multiple plots into a single grid for combined display.

group_by(): Groups data by one or more variables for aggregated analysis.

summarise(): Computes summary statistics for grouped data.

IQR(): Calculates the interquartile range of a numeric vector.

kable(): Formats data frames or matrices into tables for reports.

aov(): Fits an analysis of variance (ANOVA) model to compare group means.

plot(): Creates diagnostic plots, such as QQ plots, for model evaluation.

summary(): Provides a summary of statistical model results or data.

tidy(): Converts model outputs into a tidy data frame for easier handling.

TukeyHSD(): Performs post-hoc pairwise comparisons for ANOVA results.

kruskal.test(): Conducts a non-parametric test to compare medians across groups.

DunnTest(): Performs post-hoc pairwise comparisons for non-parametric tests.