

NavCity Demographic and Cognitive Correlates Analysis - Part 1

Your Name

2025-10-10

Load Required Libraries

```
library(tidyverse)
library(broom)
library(knitr)
library(kableExtra)
library(reshape2)
library(car)
```

Data Import and Preparation

```
# Load all data files
demographic_data <- read_csv("/Volumes/YB_Drive/NavAging_Paper/data/demographic_data.csv")
non_nav <- read_csv("/Volumes/YB_Drive/NavAging_Paper/data/non_nav_data.csv")
ya_nav <- read_csv("/Volumes/YB_Drive/NavAging_Paper/data/ya_averaged_results.csv")
oa_nav <- read_csv("/Volumes/YB_Drive/NavAging_Paper/data/oa_averaged_results.csv")

# Combine YA and OA navigation data
nav_data <- bind_rows(
  ya_nav %>% mutate(Group = "YA"),
  oa_nav %>% mutate(Group = "OA")
)

# Average navigation metrics across all blocks for each participant
nav_averaged <- nav_data %>%
  group_by(Participant, Group) %>%
  summarise(
    Mean_Speed = mean(Speed, na.rm = TRUE),
    Mean_Distance = mean(Distance, na.rm = TRUE),
    Mean_Navigation_Time = mean(Navigation_Time, na.rm = TRUE),
    .groups = "drop"
  )

# Merge all datasets
merged_temp <- demographic_data %>%
  full_join(non_nav, by = "Participant", suffix = c("_demo", "_non_nav")) %>%
  mutate(Group = coalesce(Group_non_nav,
```

```

            if_else(Group_demo == 1, "YA", "OA")) %>%
  select(-Group_demo, -Group_non_nav)

merged_data <- merged_temp %>%
  left_join(nav_averaged, by = c("Participant", "Group"))

# Calculate change scores
merged_data <- merged_data %>%
  mutate(
    SSS_Diff = SSS_Post - SSS_Pre,
    SSQ_Diff = SSQ_Post - SSQ_Pre
  )

# Display sample of merged data
cat("Total participants:", nrow(merged_data), "\n")

## Total participants: 60

cat("YA participants:", sum(merged_data$Group == "YA", na.rm = TRUE), "\n")

## YA participants: 30

cat("OA participants:", sum(merged_data$Group == "OA", na.rm = TRUE), "\n\n")

## OA participants: 30

head(merged_data) %>%
  select(Participant, Group, Gender, Handedness, SBSOD, Mean_Speed) %>%
  kable() %>%
  kable_styling()

```

Participant	Group	Gender	Handedness	SBSOD	Mean_Speed
BNC01	YA	W	R	2.20	10.750260
BNC02	YA	W	R	3.40	16.197016
BNC03	YA	M	M	5.20	12.966692
BNC04	YA	W	R	5.27	8.770131
BNC05	YA	W	R	4.87	17.149700
BNC06	OA	M	R	6.87	14.922086

Part 1: Group Comparisons (YA vs. OA)

Test 1: Gender Distribution

```

# Create contingency table
gender_table <- table(merged_data$Gender, merged_data$Group)

cat("Contingency Table:\n")

```

```

## Contingency Table:

print(gender_table)

## 
##      OA YA
##      M 12 13
##      W 18 17

cat("\n")

# Add row and column totals for clarity
gender_table_with_totals <- addmargins(gender_table)
cat("With Totals:\n")

## With Totals:

print(gender_table_with_totals)

## 
##      OA YA Sum
##      M   12 13 25
##      W   18 17 35
##      Sum 30 30 60

cat("\n")

# Calculate proportions within each group
cat("Proportions by Group:\n")

## Proportions by Group:

prop_table <- prop.table(gender_table, margin = 2)
print(round(prop_table, 3))

## 
##      OA     YA
##      M 0.400 0.433
##      W 0.600 0.567

cat("\n")

# Perform chi-square test (without Yates' continuity correction)
chi_test <- chisq.test(gender_table, correct = FALSE)

cat("Chi-Square Test Results:\n")

## Chi-Square Test Results:

```

```

cat("² =", chi_test$statistic, "\n")

## ² = 0.06857143

cat("df =", chi_test$parameter, "\n")

## df = 1

cat("p-value =", chi_test$p.value, "\n")

## p-value = 0.7934282

# Check expected frequencies
cat("\nExpected Frequencies:\n")

## 
## Expected Frequencies:

print(round(chi_test$expected, 2))

## 
##      OA    YA
## M 12.5 12.5
## W 17.5 17.5

cat("\n")

# Calculate effect size (Cramér's V)
n <- sum(gender_table)
cramers_v <- sqrt(chi_test$statistic / n)
cat("Cramér's V =", round(cramers_v, 3), "\n")

## Cramér's V = 0.034

# Interpretation
cat("\n--- INTERPRETATION ---\n")

## 
## --- INTERPRETATION ---

if(chi_test$p.value < 0.05) {
  cat("Result: SIGNIFICANT difference in gender distribution between groups (p < 0.05)\n")
} else {
  cat("Result: NO significant difference in gender distribution between groups (p >= 0.05)\n")
}

## Result: NO significant difference in gender distribution between groups (p >= 0.05)

```

```

# Check assumption
if(min(chi_test$expected) < 5) {
  cat("  WARNING: Some expected frequencies < 5. Consider Fisher's exact test.\n")
  fisher_test <- fisher.test(gender_table)
  cat("\nFisher's Exact Test (alternative):\n")
  cat("p-value =", fisher_test$p.value, "\n")
}

```

Test 2: Handedness Distribution

```

# Create contingency table (original 3 categories)
handedness_table_original <- table(merged_data$Handedness, merged_data$Group)

cat("Original Contingency Table (R, L, M):\n")

```

```
## Original Contingency Table (R, L, M):
```

```
print(handedness_table_original)
```

```
##
##      OA YA
##      L  2  2
##      M  2  3
##      R 26 25
```

```
cat("\n")
```

```
# Check if we have enough observations in each cell
cat("Raw counts by category:\n")
```

```
## Raw counts by category:
```

```
cat("Right-handed: YA =", handedness_table_original["R", "YA"], ", OA =", handedness_table_original["R",
```

```
## Right-handed: YA = 25 , OA = 26
```

```
cat("Left-handed: YA =", handedness_table_original["L", "YA"], ", OA =", handedness_table_original["L",
```

```
## Left-handed: YA = 2 , OA = 2
```

```
cat("Mixed: YA =", handedness_table_original["M", "YA"], ", OA =", handedness_table_original["M", "OA"])
```

```
## Mixed: YA = 3 , OA = 2
```

```

# Create collapsed category: Right vs. Non-Right
merged_data <- merged_data %>%
  mutate(Handedness_Collapsed = ifelse(Handedness == "R", "Right-handed", "Non-right-handed"))

# Create new contingency table
handedness_table <- table(merged_data$Handedness_Collapsed, merged_data$Group)

cat("Collapsed Contingency Table (Right vs. Non-Right):\n")

## Collapsed Contingency Table (Right vs. Non-Right):

print(handedness_table)

##                                     OA YA
##   Non-right-handed    4  5
##   Right-handed       26 25

cat("\n")

# Add row and column totals
handedness_table_with_totals <- addmargins(handedness_table)
cat("With Totals:\n")

## With Totals:

print(handedness_table_with_totals)

##                                     OA YA Sum
##   Non-right-handed    4  5   9
##   Right-handed       26 25  51
##   Sum                 30 30  60

cat("\n")

# Calculate proportions within each group
cat("Proportions by Group:\n")

## Proportions by Group:

prop_table <- prop.table(handedness_table, margin = 2)
print(round(prop_table, 3))

##                                     OA     YA
##   Non-right-handed 0.133 0.167
##   Right-handed     0.867 0.833

```

```

cat("\n")

# Perform chi-square test (without Yates' correction)
chi_test <- chisq.test(handedness_table, correct = FALSE)

cat("Chi-Square Test Results:\n")

## Chi-Square Test Results:

cat("  ^  =", chi_test$statistic, "\n")

##  ^  = 0.130719

cat("df =", chi_test$parameter, "\n")

## df = 1

cat("p-value =", chi_test$p.value, "\n")

## p-value = 0.7176878

# Check expected frequencies
cat("\nExpected Frequencies:\n")

## 
## Expected Frequencies:

print(round(chi_test$expected, 2))

## 
##          OA    YA
## Non-right-handed 4.5  4.5
## Right-handed    25.5 25.5

cat("\n")

# Calculate effect size (Cramér's V)
n <- sum(handedness_table)
cramers_v <- sqrt(chi_test$statistic / n)
cat("Cramér's V =", round(cramers_v, 3), "\n")

## Cramér's V = 0.047

# Interpretation
cat("\n--- INTERPRETATION ---\n")

## 
## --- INTERPRETATION ---

```

```

if(chi_test$p.value < 0.05) {
  cat("Result: SIGNIFICANT difference in handedness distribution between groups (p < 0.05)\n")
} else {
  cat("Result: NO significant difference in handedness distribution between groups (p >= 0.05)\n")
}

## Result: NO significant difference in handedness distribution between groups (p >= 0.05)

# Check assumptions
if(min(chi_test$expected) < 5) {
  cat("  WARNING: Some expected frequencies < 5. Consider Fisher's exact test.\n")
  fisher_test <- fisher.test(handedness_table)
  cat("\nFisher's Exact Test (alternative):\n")
  cat("p-value =", fisher_test$p.value, "\n")
}

```

WARNING: Some expected frequencies < 5. Consider Fisher's exact test.
Fisher's Exact Test (alternative):
p-value = 1

Test 3: Prior VR Experience Distribution

```

cat("VR Experience: Prior VR Exposure\n")

## VR Experience: Prior VR Exposure

cat("Note: Ordinal scale (0 = never, 1 = 1-3 times, 2 = >3 times)\n")

## Note: Ordinal scale (0 = never, 1 = 1-3 times, 2 = >3 times)

cat("Analysis: Mann-Whitney U test (appropriate for ordinal data)\n\n")

## Analysis: Mann-Whitney U test (appropriate for ordinal data)

# Show the coding scheme
cat("VR Experience Coding:\n")

## VR Experience Coding:

cat("  0 = Never\n")

## 0 = Never

cat("  1 = 1-3 times\n")

## 1 = 1-3 times

```

```

cat(" 2 = More than 3 times\n\n")

## 2 = More than 3 times

# Summary statistics by group (emphasizing median for ordinal data)
vr_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(VR_Experience_Quantified)),
    Median = median(VR_Experience_Quantified, na.rm = TRUE),
    Q1 = quantile(VR_Experience_Quantified, 0.25, na.rm = TRUE),
    Q3 = quantile(VR_Experience_Quantified, 0.75, na.rm = TRUE),
    Mean = mean(VR_Experience_Quantified, na.rm = TRUE),
    SD = sd(VR_Experience_Quantified, na.rm = TRUE),
    Min = min(VR_Experience_Quantified, na.rm = TRUE),
    Max = max(VR_Experience_Quantified, na.rm = TRUE)
  )

cat("Summary Statistics:\n")

## Summary Statistics:

print(vr_summary)

## # A tibble: 2 x 9
##   Group     n Median     Q1     Q3   Mean     SD   Min   Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30     0     0     1 0.333 0.547     0     2
## 2 YA      30     1     0     2 0.933 0.868     0     2

cat("\n")

# Show frequency distribution
cat("Frequency Distribution:\n")

## Frequency Distribution:

freq_table <- table(merged_data$VR_Experience_Quantified, merged_data$Group)
print(freq_table)

## 
##      OA YA
##      0 21 12
##      1  8  8
##      2  1 10

cat("\n")

```

```

# Filter out NA values for testing
vr_data_clean <- merged_data %>%
  filter(!is.na(VR_Experience_Quantified), !is.na(Group))

ya_vr <- vr_data_clean %>% filter(Group == "YA") %>% pull(VR_Experience_Quantified)
oa_vr <- vr_data_clean %>% filter(Group == "OA") %>% pull(VR_Experience_Quantified)

# Mann-Whitney U test
mw_test <- wilcox.test(ya_vr, oa_vr, exact = FALSE)

cat("Mann-Whitney U Test Results:\n")

## Mann-Whitney U Test Results:

cat("W statistic =", mw_test$statistic, "\n")

## W statistic = 621

cat("p-value =", mw_test$p.value, "\n")

## p-value = 0.00506242

# Effect size (rank-biserial correlation)
n1 <- length(ya_vr)
n2 <- length(oa_vr)
r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

## Rank-biserial correlation = -0.38

# Report IQR for each group
cat("\nIQR by group:\n")

## 
## IQR by group:

cat("YA: [", quantile(ya_vr, 0.25), ", ", quantile(ya_vr, 0.75), "]\\n", sep = "")

## YA: [0, 2]

cat("OA: [", quantile(oa_vr, 0.25), ", ", quantile(oa_vr, 0.75), "]\\n", sep = "")

## OA: [0, 1]

# Interpretation
cat("\n--- INTERPRETATION ---\n")

## 
## --- INTERPRETATION ---

```

```

if(mw_test$p.value < 0.05) {
  cat("Result: SIGNIFICANT difference in VR experience between groups (p < 0.05)\n")
  if(median(ya_vr) > median(oa_vr)) {
    cat("Direction: YA group has MORE prior VR experience (Median YA =", median(ya_vr),
        ", OA =", median(oa_vr), ")\n")
  } else if(median(ya_vr) < median(oa_vr)) {
    cat("Direction: OA group has MORE prior VR experience (Median YA =", median(ya_vr),
        ", OA =", median(oa_vr), ")\n")
  } else {
    cat("Direction: Medians are equal, but distributions differ\n")
  }
} else {
  cat("Result: NO significant difference in VR experience between groups (p >= 0.05)\n")
  cat("Median YA =", median(ya_vr), ", Median OA =", median(oa_vr), "\n")
}

```

```

## Result: SIGNIFICANT difference in VR experience between groups (p < 0.05)
## Direction: YA group has MORE prior VR experience (Median YA = 1 , OA = 0 )

```

Test 4: Weekly Video Game Usage

```

cat("Video Game Usage (numeric variable)\n\n")

## Video Game Usage (numeric variable)

# Summary statistics by group
vg_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(Video_Game_Experience_Quantified)),
    Mean = mean(Video_Game_Experience_Quantified, na.rm = TRUE),
    SD = sd(Video_Game_Experience_Quantified, na.rm = TRUE),
    Median = median(Video_Game_Experience_Quantified, na.rm = TRUE),
    Q1 = quantile(Video_Game_Experience_Quantified, 0.25, na.rm = TRUE),
    Q3 = quantile(Video_Game_Experience_Quantified, 0.75, na.rm = TRUE),
    Min = min(Video_Game_Experience_Quantified, na.rm = TRUE),
    Max = max(Video_Game_Experience_Quantified, na.rm = TRUE)
  )

cat("Summary Statistics:\n")

## Summary Statistics:

print(vg_summary)

## # A tibble: 2 x 9
##   Group     n   Mean    SD Median     Q1     Q3    Min    Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30  0.567  1.92     0     0     0     0     8
## 2 YA      30  2.7    3.39    1.5     0    3.75     0    10

```

```

cat("\n")

# Filter out NA values for testing
vg_data_clean <- merged_data %>%
  filter(!is.na(Video_Game_Experience_Quantified), !is.na(Group))

ya_vg <- vg_data_clean %>% filter(Group == "YA") %>% pull(Video_Game_Experience_Quantified)
oa_vg <- vg_data_clean %>% filter(Group == "OA") %>% pull(Video_Game_Experience_Quantified)

# Check normality with Shapiro-Wilk test
cat("Normality Tests (Shapiro-Wilk):\n")

## Normality Tests (Shapiro-Wilk):

shapiro_ya <- shapiro.test(ya_vg)
shapiro_oa <- shapiro.test(oa_vg)
cat("YA: W =", round(shapiro_ya$statistic, 4), ", p =", shapiro_ya$p.value)

## YA: W = 0.7793 , p = 2.812079e-05

if(shapiro_ya$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (non-normal)

cat("\n")

cat("OA: W =", round(shapiro_oa$statistic, 4), ", p =", shapiro_oa$p.value)

## OA: W = 0.3282 , p = 1.229509e-10

if(shapiro_oa$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (non-normal)

cat("\n\n")

# Check homogeneity of variance with Levene's test
levene_test <- leveneTest(Video_Game_Experience_Quantified ~ Group, data = vg_data_clean)
cat("Levene's Test for Homogeneity of Variance:\n")

## Levene's Test for Homogeneity of Variance:

cat("F =", round(levene_test$`F value`[1], 4), ", p =", levene_test$`Pr(>F)`[1])

## F = 13.4959 , p = 0.0005232028

```

```

if(levene_test$`Pr(>F)`[1] < 0.05) cat(" (variances unequal)") else cat(" (variances equal)")

## (variances unequal)

cat("\n\n")

# Decide which test to use
use_welch <- levene_test$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_ya$p.value < 0.05 | shapiro_oa$p.value < 0.05

if(use_nonparametric) {
  cat(" Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(ya_vg, oa_vg, exact = FALSE)
  cat("Mann-Whitney U Test Results:\n")
  cat("W statistic =", mw_test$statistic, "\n")
  cat("p-value =", mw_test$p.value, "\n")

  # Effect size
  n1 <- length(ya_vg)
  n2 <- length(oa_vg)
  r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
  cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

  # Report IQR for each group
  cat("\nIQR by group:\n")
  cat("YA: [", quantile(ya_vg, 0.25), ", ", quantile(ya_vg, 0.75), "]\n", sep = "")
  cat("OA: [", quantile(oa_vg, 0.25), ", ", quantile(oa_vg, 0.75), "]\n", sep = "")

  p_value <- mw_test$p.value

} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
  t_test <- t.test(ya_vg, oa_vg, var.equal = FALSE)
  cat("Welch's t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "]\n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_vg)^2 + sd(oa_vg)^2) / 2)
  cohens_d <- (mean(ya_vg) - mean(oa_vg)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value

} else {
  cat("Assumptions met. Using standard independent samples t-test.\n\n")
  t_test <- t.test(ya_vg, oa_vg, var.equal = TRUE)
  cat("Independent Samples t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
}

```

```

cat("95% CI for difference: [", t_test$conf.int[1], ", ", ", t_test$conf.int[2], "]\\n", sep = "")

# Effect size (Cohen's d)
pooled_sd <- sqrt((sd(ya_vg)^2 + sd(oa_vg)^2) / 2)
cohens_d <- (mean(ya_vg) - mean(oa_vg)) / pooled_sd
cat("Cohen's d =", round(cohens_d, 3), "\\n")

p_value <- t_test$p.value
}

## Data violates normality assumption. Using Mann-Whitney U test.
##
## Mann-Whitney U Test Results:
## W statistic = 643.5
## p-value = 0.0005407829
## Rank-biserial correlation = -0.43
##
## IQR by group:
## YA: [0, 3.75]
## OA: [0, 0]

# Interpretation
cat("\\n--- INTERPRETATION ---\\n")

##
## --- INTERPRETATION ---

if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in video game usage between groups (p < 0.05)\\n")
  if(mean(ya_vg) > mean(oa_vg)) {
    cat("Direction: YA group plays MORE video games than OA group\\n")
  } else {
    cat("Direction: OA group plays MORE video games than YA group\\n")
  }
} else {
  cat("Result: NO significant difference in video game usage between groups (p >= 0.05)\\n")
}

## Result: SIGNIFICANT difference in video game usage between groups (p < 0.05)
## Direction: YA group plays MORE video games than OA group

```

Test 5: Weekly Exercise Frequency

```
cat("Exercise Frequency (numeric variable)\\n\\n")
```

```
## Exercise Frequency (numeric variable)
```

```

# Summary statistics by group
exercise_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(Exercise_Quantified)),
    Mean = mean(Exercise_Quantified, na.rm = TRUE),
    SD = sd(Exercise_Quantified, na.rm = TRUE),
    Median = median(Exercise_Quantified, na.rm = TRUE),
    Q1 = quantile(Exercise_Quantified, 0.25, na.rm = TRUE),
    Q3 = quantile(Exercise_Quantified, 0.75, na.rm = TRUE),
    Min = min(Exercise_Quantified, na.rm = TRUE),
    Max = max(Exercise_Quantified, na.rm = TRUE)
  )

cat("Summary Statistics:\n")

## Summary Statistics:

print(exercise_summary)

## # A tibble: 2 x 9
##   Group     n   Mean    SD Median    Q1    Q3   Min   Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30   5.04  3.19  4.75   3     7     0    14
## 2 YA      30   4.69  3.30  3.9    2.62   6     0    15

cat("\n")

# Filter out NA values for testing
exercise_data_clean <- merged_data %>%
  filter(!is.na(Exercise_Quantified), !is.na(Group))

ya_ex <- exercise_data_clean %>% filter(Group == "YA") %>% pull(Exercise_Quantified)
oa_ex <- exercise_data_clean %>% filter(Group == "OA") %>% pull(Exercise_Quantified)

# Check normality with Shapiro-Wilk test
cat("Normality Tests (Shapiro-Wilk):\n")

## Normality Tests (Shapiro-Wilk):

shapiro_ya <- shapiro.test(ya_ex)
shapiro_oa <- shapiro.test(oa_ex)
cat("YA: W =", round(shapiro_ya$statistic, 4), ", p =", shapiro_ya$p.value)

## YA: W = 0.889 , p = 0.004559315

if(shapiro_ya$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

##  (non-normal)

```

```

cat("\n")

cat("OA: W =", round(shapiro_oa$statistic, 4), ", p =", shapiro_oa$p.value)

## OA: W = 0.951 , p = 0.1800439

if(shapiro_oa$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (normal)

cat("\n\n")

# Check homogeneity of variance with Levene's test
levene_test <- leveneTest(Exercise_Quantified ~ Group, data = exercise_data_clean)
cat("Levene's Test for Homogeneity of Variance:\n")

## Levene's Test for Homogeneity of Variance:

cat("F =", round(levene_test$`F value`[1], 4), ", p =", levene_test$`Pr(>F)`[1])

## F = 0.0265 , p = 0.871163

if(levene_test$`Pr(>F)`[1] < 0.05) cat(" (variances unequal)") else cat(" (variances equal)")

## (variances equal)

cat("\n\n")

# Decide which test to use
use_welch <- levene_test$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_ya$p.value < 0.05 | shapiro_oa$p.value < 0.05

if(use_nonparametric) {
  cat(" Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(ya_ex, oa_ex, exact = FALSE)
  cat("Mann-Whitney U Test Results:\n")
  cat("W statistic =", mw_test$statistic, "\n")
  cat("p-value =", mw_test$p.value, "\n")

  # Effect size
  n1 <- length(ya_ex)
  n2 <- length(oa_ex)
  r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
  cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

  # Report IQR for each group
  cat("\nIQR by group:\n")
  cat("YA: [", quantile(ya_ex, 0.25), ", ", quantile(ya_ex, 0.75), "]\n", sep = "")
  cat("OA: [", quantile(oa_ex, 0.25), ", ", quantile(oa_ex, 0.75), "]\n", sep = "")
}

```

```

p_value <- mw_test$p.value

} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
  t_test <- t.test(ya_ex, oa_ex, var.equal = FALSE)
  cat("Welch's t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "] \n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_ex)^2 + sd(oa_ex)^2) / 2)
  cohens_d <- (mean(ya_ex) - mean(oa_ex)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value

} else {
  cat("Assumptions met. Using standard independent samples t-test.\n\n")
  t_test <- t.test(ya_ex, oa_ex, var.equal = TRUE)
  cat("Independent Samples t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "] \n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_ex)^2 + sd(oa_ex)^2) / 2)
  cohens_d <- (mean(ya_ex) - mean(oa_ex)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value
}

## Data violates normality assumption. Using Mann-Whitney U test.
## 
## Mann-Whitney U Test Results:
## W statistic = 408.5
## p-value = 0.5435665
## Rank-biserial correlation = 0.092
## 
## IQR by group:
## YA: [2.625, 6]
## OA: [3, 7]

# Interpretation
cat("\n--- INTERPRETATION ---\n")

##
## --- INTERPRETATION ---

```

```

if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in exercise frequency between groups (p < 0.05)\n")
  if(mean(ya_ex) > mean(oa_ex)) {
    cat("Direction: YA group exercises MORE than OA group\n")
  } else {
    cat("Direction: OA group exercises MORE than YA group\n")
  }
} else {
  cat("Result: NO significant difference in exercise frequency between groups (p >= 0.05)\n")
}

```

```
## Result: NO significant difference in exercise frequency between groups (p >= 0.05)
```

Test 6: SBSOD Scores

```
cat("SBSOD: Santa Barbara Sense of Direction Scale\n\n")
```

```
## SBSOD: Santa Barbara Sense of Direction Scale
```

```
# Summary statistics by group
sbsod_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(SBSOD)),
    Mean = mean(SBSOD, na.rm = TRUE),
    SD = sd(SBSOD, na.rm = TRUE),
    Median = median(SBSOD, na.rm = TRUE),
    Q1 = quantile(SBSOD, 0.25, na.rm = TRUE),
    Q3 = quantile(SBSOD, 0.75, na.rm = TRUE),
    Min = min(SBSOD, na.rm = TRUE),
    Max = max(SBSOD, na.rm = TRUE)
  )
```

```
cat("Summary Statistics:\n")
```

```
## Summary Statistics:
```

```
print(sbsod_summary)
```

```
## # A tibble: 2 x 9
##   Group     n   Mean    SD Median    Q1    Q3   Min   Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30  5.08  0.925  5.16  4.34  5.53  3.27  6.87
## 2 YA      30  4.35  1.10   4.5   3.55  5.32  2.13  6
```

```
cat("\n")
```

```

# Filter out NA values for testing
sbsod_data_clean <- merged_data %>%
  filter(!is.na(SBSOD), !is.na(Group))

ya_sbsod <- sbsod_data_clean %>% filter(Group == "YA") %>% pull(SBSOD)
oa_sbsod <- sbsod_data_clean %>% filter(Group == "OA") %>% pull(SBSOD)

# Check normality with Shapiro-Wilk test
cat("Normality Tests (Shapiro-Wilk):\n")

## Normality Tests (Shapiro-Wilk):

shapiro_ya <- shapiro.test(ya_sbsod)
shapiro_oa <- shapiro.test(oa_sbsod)
cat("YA: W =", round(shapiro_ya$statistic, 4), ", p =", shapiro_ya$p.value)

## YA: W = 0.9522 , p = 0.1939842

if(shapiro_ya$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (normal)

cat("\n")

cat("OA: W =", round(shapiro_oa$statistic, 4), ", p =", shapiro_oa$p.value)

## OA: W = 0.9704 , p = 0.5493297

if(shapiro_oa$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (normal)

cat("\n\n")

# Check homogeneity of variance with Levene's test
levene_test <- leveneTest(SBSOD ~ Group, data = sbsod_data_clean)
cat("Levene's Test for Homogeneity of Variance:\n")

## Levene's Test for Homogeneity of Variance:

cat("F =", round(levene_test$`F value`[1], 4), ", p =", levene_test$`Pr(>F)`[1])

## F = 2.0401 , p = 0.1585664

if(levene_test$`Pr(>F)`[1] < 0.05) cat(" (variances unequal)") else cat(" (variances equal)")

## (variances equal)

```

```

cat("\n\n")

# Decide which test to use
use_welch <- levene_test$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_ya$p.value < 0.05 | shapiro_oa$p.value < 0.05

if(use_nonparametric) {
  cat("  Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(ya_sbsod, oa_sbsod, exact = FALSE)
  cat("Mann-Whitney U Test Results:\n")
  cat("W statistic =", mw_test$statistic, "\n")
  cat("p-value =", mw_test$p.value, "\n")

  # Effect size
  n1 <- length(ya_sbsod)
  n2 <- length(oa_sbsod)
  r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
  cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

  # Report IQR for each group
  cat("\nIQR by group:\n")
  cat("YA: [", quantile(ya_sbsod, 0.25), ", ", quantile(ya_sbsod, 0.75), "]\n", sep = "")
  cat("OA: [", quantile(oa_sbsod, 0.25), ", ", quantile(oa_sbsod, 0.75), "]\n", sep = "")

  p_value <- mw_test$p.value
}

} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
  t_test <- t.test(ya_sbsod, oa_sbsod, var.equal = FALSE)
  cat("Welch's t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "]\n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_sbsod)^2 + sd(oa_sbsod)^2) / 2)
  cohens_d <- (mean(ya_sbsod) - mean(oa_sbsod)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value
}

} else {
  cat("Assumptions met. Using standard independent samples t-test.\n\n")
  t_test <- t.test(ya_sbsod, oa_sbsod, var.equal = TRUE)
  cat("Independent Samples t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "]\n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_sbsod)^2 + sd(oa_sbsod)^2) / 2)
}

```

```

cohens_d <- (mean(ya_sbsod) - mean(oa_sbsod)) / pooled_sd
cat("Cohen's d =", round(cohens_d, 3), "\n")

p_value <- t_test$p.value
}

## Assumptions met. Using standard independent samples t-test.
##
## Independent Samples t-test Results:
## t = -2.791476
## df = 58
## p-value = 0.007092444
## 95% CI for difference: [-1.259194, -0.2074731]
## Cohen's d = -0.721

# Interpretation
cat("\n--- INTERPRETATION ---\n")

##
## --- INTERPRETATION ---

if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in SBSOD scores between groups (p < 0.05)\n")
  if(mean(ya_sbsod) > mean(oa_sbsod)) {
    cat("Direction: YA group has BETTER sense of direction than OA group\n")
  } else {
    cat("Direction: OA group has BETTER sense of direction than YA group\n")
  }
} else {
  cat("Result: NO significant difference in SBSOD scores between groups (p >= 0.05)\n")
}

## Result: SIGNIFICANT difference in SBSOD scores between groups (p < 0.05)
## Direction: OA group has BETTER sense of direction than YA group

```

Test 7: PSQI Scores

```

cat("PSQI: Pittsburgh Sleep Quality Index\n")

## PSQI: Pittsburgh Sleep Quality Index

cat("Note: Higher scores indicate WORSE sleep quality (range 0-21)\n\n")

## Note: Higher scores indicate WORSE sleep quality (range 0-21)

```

```

# Summary statistics by group
psqi_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(PSQI)),
    Mean = mean(PSQI, na.rm = TRUE),
    SD = sd(PSQI, na.rm = TRUE),
    Median = median(PSQI, na.rm = TRUE),
    Q1 = quantile(PSQI, 0.25, na.rm = TRUE),
    Q3 = quantile(PSQI, 0.75, na.rm = TRUE),
    Min = min(PSQI, na.rm = TRUE),
    Max = max(PSQI, na.rm = TRUE)
  )

cat("Summary Statistics:\n")

## Summary Statistics:

print(psqi_summary)

## # A tibble: 2 x 9
##   Group     n   Mean    SD Median    Q1    Q3   Min   Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30   4.83  2.90   4.5    2     7     0    11
## 2 YA      30   5.1    1.95   5      4     6     2    12

cat("\n")

# Filter out NA values for testing
psqi_data_clean <- merged_data %>%
  filter(!is.na(PSQI), !is.na(Group))

ya_psqi <- psql_data_clean %>% filter(Group == "YA") %>% pull(PSQI)
oa_psqi <- psql_data_clean %>% filter(Group == "OA") %>% pull(PSQI)

# Check normality with Shapiro-Wilk test
cat("Normality Tests (Shapiro-Wilk):\n")

## Normality Tests (Shapiro-Wilk):

shapiro_ya <- shapiro.test(ya_psqi)
shapiro_oa <- shapiro.test(oa_psqi)
cat("YA: W =", round(shapiro_ya$statistic, 4), ", p =", shapiro_ya$p.value)

## YA: W = 0.8731 , p = 0.001963064

if(shapiro_ya$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

##  (non-normal)

```

```

cat("\n")

cat("OA: W =", round(shapiro_oa$statistic, 4), ", p =", shapiro_oa$p.value)

## OA: W = 0.9554 , p = 0.2347819

if(shapiro_oa$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (normal)

cat("\n\n")

# Check homogeneity of variance with Levene's test
levene_test <- leveneTest(PSQI ~ Group, data = psqi_data_clean)
cat("Levene's Test for Homogeneity of Variance:\n")

## Levene's Test for Homogeneity of Variance:

cat("F =", round(levene_test$`F value`[1], 4), ", p =", levene_test$`Pr(>F)`[1])

## F = 10.414 , p = 0.002058421

if(levene_test$`Pr(>F)`[1] < 0.05) cat(" (variances unequal)") else cat(" (variances equal)")

## (variances unequal)

cat("\n\n")

# Decide which test to use
use_welch <- levene_test$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_ya$p.value < 0.05 | shapiro_oa$p.value < 0.05

if(use_nonparametric) {
  cat(" Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(ya_psqi, oa_psqi, exact = FALSE)
  cat("Mann-Whitney U Test Results:\n")
  cat("W statistic =", mw_test$statistic, "\n")
  cat("p-value =", mw_test$p.value, "\n")

  # Effect size
  n1 <- length(ya_psqi)
  n2 <- length(oa_psqi)
  r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
  cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

  # Report IQR for each group
  cat("\nIQR by group:\n")
  cat("YA: [", quantile(ya_psqi, 0.25), ", ", quantile(ya_psqi, 0.75), "]\n", sep = "")
  cat("OA: [", quantile(oa_psqi, 0.25), ", ", quantile(oa_psqi, 0.75), "]\n", sep = "")
}

```

```

p_value <- mw_test$p.value

} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
  t_test <- t.test(ya_psqi, oa_psqi, var.equal = FALSE)
  cat("Welch's t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "] \n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_psqi)^2 + sd(oa_psqi)^2) / 2)
  cohens_d <- (mean(ya_psqi) - mean(oa_psqi)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value

} else {
  cat("Assumptions met. Using standard independent samples t-test.\n\n")
  t_test <- t.test(ya_psqi, oa_psqi, var.equal = TRUE)
  cat("Independent Samples t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "] \n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_psqi)^2 + sd(oa_psqi)^2) / 2)
  cohens_d <- (mean(ya_psqi) - mean(oa_psqi)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value
}

## Data violates normality assumption. Using Mann-Whitney U test.
## 
## Mann-Whitney U Test Results:
## W statistic = 472
## p-value = 0.7486007
## Rank-biserial correlation = -0.049
##
## IQR by group:
## YA: [4, 6]
## OA: [2, 7]

# Interpretation
cat("\n--- INTERPRETATION ---\n")

##
## --- INTERPRETATION ---

```

```

if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in sleep quality between groups (p < 0.05)\n")
  if(mean(ya_psqi) > mean(oa_psqi)) {
    cat("Direction: YA group has WORSE sleep quality than OA group\n")
  } else {
    cat("Direction: OA group has WORSE sleep quality than YA group\n")
  }
} else {
  cat("Result: NO significant difference in sleep quality between groups (p >= 0.05)\n")
}

```

```
## Result: NO significant difference in sleep quality between groups (p >= 0.05)
```

Test 8: Trails Making Test A Performance

```

cat("Trails Making Test A: Completion Time\n")

## Trails Making Test A: Completion Time

cat("Note: Lower completion time = BETTER performance\n\n")

```

```
## Note: Lower completion time = BETTER performance
```

```

# Summary statistics by group
trails_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(Trails_A_CT)),
    Mean = mean(Trails_A_CT, na.rm = TRUE),
    SD = sd(Trails_A_CT, na.rm = TRUE),
    Median = median(Trails_A_CT, na.rm = TRUE),
    Q1 = quantile(Trails_A_CT, 0.25, na.rm = TRUE),
    Q3 = quantile(Trails_A_CT, 0.75, na.rm = TRUE),
    Min = min(Trails_A_CT, na.rm = TRUE),
    Max = max(Trails_A_CT, na.rm = TRUE)
  )

```

```
cat("Summary Statistics:\n")
```

```
## Summary Statistics:
```

```
print(trails_summary)
```

```

## # A tibble: 2 x 9
##   Group     n   Mean    SD Median     Q1     Q3   Min   Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30  34.2   6.84  34.7  29.5  36.4  22.4  48.6
## 2 YA      30  25.9   4.56  24.6  22.9  30.0  17.1  35.2

```

```

cat("\n")

# Filter out NA values for testing
trails_data_clean <- merged_data %>%
  filter(!is.na(Trails_A_CT), !is.na(Group))

ya_trails <- trails_data_clean %>% filter(Group == "YA") %>% pull(Trails_A_CT)
oa_trails <- trails_data_clean %>% filter(Group == "OA") %>% pull(Trails_A_CT)

# Check normality with Shapiro-Wilk test
cat("Normality Tests (Shapiro-Wilk):\n")

## Normality Tests (Shapiro-Wilk):

shapiro_ya <- shapiro.test(ya_trails)
shapiro_oa <- shapiro.test(oa_trails)
cat("YA: W =", round(shapiro_ya$statistic, 4), ", p =", shapiro_ya$p.value)

## YA: W = 0.9444 , p = 0.119447

if(shapiro_ya$p.value < 0.05) cat("(non-normal)") else cat("(normal)")

## (normal)

cat("\n")

cat("OA: W =", round(shapiro_oa$statistic, 4), ", p =", shapiro_oa$p.value)

## OA: W = 0.9406 , p = 0.0946305

if(shapiro_oa$p.value < 0.05) cat("(non-normal)") else cat("(normal)")

## (normal)

cat("\n\n")

# Check homogeneity of variance with Levene's test
levene_test <- leveneTest(Trails_A_CT ~ Group, data = trails_data_clean)
cat("Levene's Test for Homogeneity of Variance:\n")

## Levene's Test for Homogeneity of Variance:

cat("F =", round(levene_test$`F value`[1], 4), ", p =", levene_test$`Pr(>F)`[1])

## F = 1.9659 , p = 0.1662154

```

```

if(levene_test$`Pr(>F)`[1] < 0.05) cat(" (variances unequal)") else cat(" (variances equal)")

## (variances equal)

cat("\n\n")

# Decide which test to use
use_welch <- levene_test$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_ya$p.value < 0.05 | shapiro_oa$p.value < 0.05

if(use_nonparametric) {
  cat(" Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(ya_trails, oa_trails, exact = FALSE)
  cat("Mann-Whitney U Test Results:\n")
  cat("W statistic =", mw_test$statistic, "\n")
  cat("p-value =", mw_test$p.value, "\n")

  # Effect size
  n1 <- length(ya_trails)
  n2 <- length(oa_trails)
  r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
  cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

  # Report IQR for each group
  cat("\nIQR by group:\n")
  cat("YA: [", quantile(ya_trails, 0.25), ", ", quantile(ya_trails, 0.75), "]\n", sep = "")
  cat("OA: [", quantile(oa_trails, 0.25), ", ", quantile(oa_trails, 0.75), "]\n", sep = "")

  p_value <- mw_test$p.value

} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
  t_test <- t.test(ya_trails, oa_trails, var.equal = FALSE)
  cat("Welch's t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "]\n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_trails)^2 + sd(oa_trails)^2) / 2)
  cohens_d <- (mean(ya_trails) - mean(oa_trails)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value

} else {
  cat("Assumptions met. Using standard independent samples t-test.\n\n")
  t_test <- t.test(ya_trails, oa_trails, var.equal = TRUE)
  cat("Independent Samples t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
}

```

```

cat("95% CI for difference: [", t_test$conf.int[1], ", ", ", t_test$conf.int[2], "]\\n", sep = "")

# Effect size (Cohen's d)
pooled_sd <- sqrt((sd(ya_trails)^2 + sd(oa_trails)^2) / 2)
cohens_d <- (mean(ya_trails) - mean(oa_trails)) / pooled_sd
cat("Cohen's d =", round(cohens_d, 3), "\\n")

p_value <- t_test$p.value
}

## Assumptions met. Using standard independent samples t-test.
##
## Independent Samples t-test Results:
## t = -5.534026
## df = 58
## p-value = 7.879815e-07
## 95% CI for difference: [-11.31128, -5.302055]
## Cohen's d = -1.429

# Interpretation
cat("\\n--- INTERPRETATION ---\\n")

##
## --- INTERPRETATION ---

if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in Trails A performance between groups (p < 0.05)\\n")
  if(mean(ya_trails) < mean(oa_trails)) {
    cat("Direction: YA group is FASTER (better) than OA group\\n")
  } else {
    cat("Direction: OA group is FASTER (better) than YA group\\n")
  }
} else {
  cat("Result: NO significant difference in Trails A performance between groups (p >= 0.05)\\n")
}

## Result: SIGNIFICANT difference in Trails A performance between groups (p < 0.05)
## Direction: YA group is FASTER (better) than OA group

```

Test 9: Trails Making Test B Performance

```

cat("Trails Making Test B: Completion Time\\n")

## Trails Making Test B: Completion Time

cat("Note: Lower completion time = BETTER performance\\n")

## Note: Lower completion time = BETTER performance

```

```
cat("Trails B measures executive function and set-shifting\n\n")
```

```
## Trails B measures executive function and set-shifting
```

```
# Summary statistics by group
trails_b_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(Trails_B_CT)),
    Mean = mean(Trails_B_CT, na.rm = TRUE),
    SD = sd(Trails_B_CT, na.rm = TRUE),
    Median = median(Trails_B_CT, na.rm = TRUE),
    Q1 = quantile(Trails_B_CT, 0.25, na.rm = TRUE),
    Q3 = quantile(Trails_B_CT, 0.75, na.rm = TRUE),
    Min = min(Trails_B_CT, na.rm = TRUE),
    Max = max(Trails_B_CT, na.rm = TRUE)
  )
```

```
cat("Summary Statistics:\n")
```

```
## Summary Statistics:
```

```
print(trails_b_summary)
```

```
## # A tibble: 2 x 9
##   Group     n   Mean    SD Median    Q1    Q3   Min   Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30  59.5  28.9  48.6  44.6  65.9  26.4 160.
## 2 YA      30  36.9  11.1  35.5  29.7  38     21.5  75.1
```

```
cat("\n")
```

```
# Filter out NA values for testing
trails_b_data_clean <- merged_data %>%
  filter(!is.na(Trails_B_CT), !is.na(Group))
```

```
ya_trails_b <- trails_b_data_clean %>% filter(Group == "YA") %>% pull(Trails_B_CT)
oa_trails_b <- trails_b_data_clean %>% filter(Group == "OA") %>% pull(Trails_B_CT)
```

```
# Check normality with Shapiro-Wilk test
cat("Normality Tests (Shapiro-Wilk):\n")
```

```
## Normality Tests (Shapiro-Wilk):
```

```
shapiro_ya <- shapiro.test(ya_trails_b)
shapiro_oa <- shapiro.test(oa_trails_b)
cat("YA: W =", round(shapiro_ya$statistic, 4), ", p =", shapiro_ya$p.value)
```

```
## YA: W = 0.8805 , p = 0.002891261
```

```

if(shapiro_ya$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (non-normal)

cat("\n")

cat("OA: W =", round(shapiro_oa$statistic, 4), ", p =", shapiro_oa$p.value)

## OA: W = 0.7613 , p = 1.393277e-05

if(shapiro_oa$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (non-normal)

cat("\n\n")

# Check homogeneity of variance with Levene's test
levene_test <- leveneTest(Trails_B_CT ~ Group, data = trails_b_data_clean)
cat("Levene's Test for Homogeneity of Variance:\n")

## Levene's Test for Homogeneity of Variance:

cat("F =", round(levene_test$`F value`[1], 4), ", p =", levene_test$`Pr(>F)`[1])

## F = 3.8705 , p = 0.05393151

if(levene_test$`Pr(>F)`[1] < 0.05) cat(" (variances unequal)") else cat(" (variances equal)")

## (variances equal)

cat("\n\n")

# Decide which test to use
use_welch <- levene_test$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_ya$p.value < 0.05 | shapiro_oa$p.value < 0.05

if(use_nonparametric) {
  cat(" Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(ya_trails_b, oa_trails_b, exact = FALSE)
  cat("Mann-Whitney U Test Results:\n")
  cat("W statistic =", mw_test$statistic, "\n")
  cat("p-value =", mw_test$p.value, "\n")

  # Effect size
  n1 <- length(ya_trails_b)
  n2 <- length(oa_trails_b)
  r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
  cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")
}

```

```

# Report IQR for each group
cat("\nIQR by group:\n")
cat("YA: [", quantile(ya_trails_b, 0.25), ", ", quantile(ya_trails_b, 0.75), "]\\n", sep = "")
cat("OA: [", quantile(oa_trails_b, 0.25), ", ", quantile(oa_trails_b, 0.75), "]\\n", sep = "")

p_value <- mw_test$p.value

} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
  t_test <- t.test(ya_trails_b, oa_trails_b, var.equal = FALSE)
  cat("Welch's t-test Results:\n")
  cat("t =", t_test$statistic, "\\n")
  cat("df =", t_test$parameter, "\\n")
  cat("p-value =", t_test$p.value, "\\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "]\\n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_trails_b)^2 + sd(oa_trails_b)^2) / 2)
  cohens_d <- (mean(ya_trails_b) - mean(oa_trails_b)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\\n")

  p_value <- t_test$p.value

} else {
  cat("Assumptions met. Using standard independent samples t-test.\n\n")
  t_test <- t.test(ya_trails_b, oa_trails_b, var.equal = TRUE)
  cat("Independent Samples t-test Results:\n")
  cat("t =", t_test$statistic, "\\n")
  cat("df =", t_test$parameter, "\\n")
  cat("p-value =", t_test$p.value, "\\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "]\\n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_trails_b)^2 + sd(oa_trails_b)^2) / 2)
  cohens_d <- (mean(ya_trails_b) - mean(oa_trails_b)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\\n")

  p_value <- t_test$p.value
}

## Data violates normality assumption. Using Mann-Whitney U test.
## 
## Mann-Whitney U Test Results:
## W statistic = 151.5
## p-value = 1.052357e-05
## Rank-biserial correlation = 0.663
##
## IQR by group:
## YA: [29.675, 38]
## OA: [44.55, 65.9]

```

```

# Interpretation
cat("\n--- INTERPRETATION ---\n")

## 
## --- INTERPRETATION ---

if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in Trails B performance between groups (p < 0.05)\n")
  if(mean(ya_trails_b) < mean(oa_trails_b)) {
    cat("Direction: YA group is FASTER (better) than OA group\n")
  } else {
    cat("Direction: OA group is FASTER (better) than YA group\n")
  }
} else {
  cat("Result: NO significant difference in Trails B performance between groups (p >= 0.05)\n")
}

## Result: SIGNIFICANT difference in Trails B performance between groups (p < 0.05)
## Direction: YA group is FASTER (better) than OA group

```

Test 10: Trails B-A Difference (Cognitive Switching Cost)

```

cat("Trails B-A Difference: Cognitive Switching Cost\n")

## Trails B-A Difference: Cognitive Switching Cost

cat("Note: Higher values = GREATER switching cost (worse executive function)\n")

## Note: Higher values = GREATER switching cost (worse executive function)

cat("This measure isolates set-shifting ability from processing speed\n\n")

## This measure isolates set-shifting ability from processing speed

# Calculate B-A difference
merged_data <- merged_data %>%
  mutate(Trails_BA_Diff = Trails_B_CT - Trails_A_CT)

# Summary statistics by group
trails_ba_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(Trails_BA_Diff)),
    Mean = mean(Trails_BA_Diff, na.rm = TRUE),
    SD = sd(Trails_BA_Diff, na.rm = TRUE),
    Median = median(Trails_BA_Diff, na.rm = TRUE),
    Q1 = quantile(Trails_BA_Diff, 0.25, na.rm = TRUE),
    Q3 = quantile(Trails_BA_Diff, 0.75, na.rm = TRUE),

```

```

    Min = min(Trails_BA_Diff, na.rm = TRUE),
    Max = max(Trails_BA_Diff, na.rm = TRUE)
)

cat("Summary Statistics:\n")

## Summary Statistics:

print(trails_ba_summary)

## # A tibble: 2 x 9
##   Group     n   Mean    SD Median    Q1    Q3   Min   Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30   25.3  26.3  17.3  11.6  23.5 -3    111.
## 2 YA      30   11.0  9.81  8.45  5.13  14.8 -2.5  39.9

cat("\n")

# Filter out NA values for testing
trails_ba_data_clean <- merged_data %>%
  filter(!is.na(Trails_BA_Diff), !is.na(Group))

ya_trails_ba <- trails_ba_data_clean %>% filter(Group == "YA") %>% pull(Trails_BA_Diff)
oa_trails_ba <- trails_ba_data_clean %>% filter(Group == "OA") %>% pull(Trails_BA_Diff)

# Check normality with Shapiro-Wilk test
cat("Normality Tests (Shapiro-Wilk):\n")

## Normality Tests (Shapiro-Wilk):

shapiro_ya <- shapiro.test(ya_trails_ba)
shapiro_oa <- shapiro.test(oa_trails_ba)
cat("YA: W =", round(shapiro_ya$statistic, 4), ", p =", shapiro_ya$p.value)

## YA: W = 0.8963 , p = 0.006828979

if(shapiro_ya$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (non-normal)

cat("\n")

cat("OA: W =", round(shapiro_oa$statistic, 4), ", p =", shapiro_oa$p.value)

## OA: W = 0.7318 , p = 4.690528e-06

```

```

if(shapiro_oa$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (non-normal)

cat("\n\n")

# Check homogeneity of variance with Levene's test
levene_test <- leveneTest(Trails_BA_Diff ~ Group, data = trails_ba_data_clean)
cat("Levene's Test for Homogeneity of Variance:\n")

## Levene's Test for Homogeneity of Variance:

cat("F =", round(levene_test$`F value`[1], 4), ", p =", levene_test$`Pr(>F)`[1])

## F = 3.536 , p = 0.06507473

if(levene_test$`Pr(>F)`[1] < 0.05) cat(" (variances unequal)") else cat(" (variances equal)")

## (variances equal)

cat("\n\n")

# Decide which test to use
use_welch <- levene_test$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_ya$p.value < 0.05 | shapiro_oa$p.value < 0.05

if(use_nonparametric) {
  cat(" Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(ya_trails_ba, oa_trails_ba, exact = FALSE)
  cat("Mann-Whitney U Test Results:\n")
  cat("W statistic =", mw_test$statistic, "\n")
  cat("p-value =", mw_test$p.value, "\n")

  # Effect size
  n1 <- length(ya_trails_ba)
  n2 <- length(oa_trails_ba)
  r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
  cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

  # Report IQR for each group
  cat("\nIQR by group:\n")
  cat("YA: [", quantile(ya_trails_ba, 0.25), ", ", quantile(ya_trails_ba, 0.75), "]\n", sep = "")
  cat("OA: [", quantile(oa_trails_ba, 0.25), ", ", quantile(oa_trails_ba, 0.75), "]\n", sep = "")

  p_value <- mw_test$p.value

} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
  t_test <- t.test(ya_trails_ba, oa_trails_ba, var.equal = FALSE)
  cat("Welch's t-test Results:\n")
}

```

```

cat("t =", t_test$statistic, "\n")
cat("df =", t_test$parameter, "\n")
cat("p-value =", t_test$p.value, "\n")
cat("95% CI for difference: [", t_test$conf.int[1], " , ", t_test$conf.int[2], "] \n", sep = "")

# Effect size (Cohen's d)
pooled_sd <- sqrt((sd(ya_trails_ba)^2 + sd(oa_trails_ba)^2) / 2)
cohens_d <- (mean(ya_trails_ba) - mean(oa_trails_ba)) / pooled_sd
cat("Cohen's d =", round(cohens_d, 3), "\n")

p_value <- t_test$p.value

} else {
  cat("Assumptions met. Using standard independent samples t-test.\n\n")
  t_test <- t.test(ya_trails_ba, oa_trails_ba, var.equal = TRUE)
  cat("Independent Samples t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], " , ", t_test$conf.int[2], "] \n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_trails_ba)^2 + sd(oa_trails_ba)^2) / 2)
  cohens_d <- (mean(ya_trails_ba) - mean(oa_trails_ba)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value
}

## Data violates normality assumption. Using Mann-Whitney U test.
##
## Mann-Whitney U Test Results:
## W statistic = 237
## p-value = 0.001679756
## Rank-biserial correlation = 0.473
##
## IQR by group:
## YA: [5.125, 14.775]
## OA: [11.6, 23.475]

# Interpretation
cat("\n--- INTERPRETATION ---\n")

##
## --- INTERPRETATION ---

if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in cognitive switching cost between groups (p < 0.05)\n")
  if(mean(ya_trails_ba) < mean(oa_trails_ba)) {
    cat("Direction: YA group has LOWER switching cost (better executive function) than OA group\n")
  } else {
    cat("Direction: OA group has LOWER switching cost (better executive function) than YA group\n")
}

```

```

    }
} else {
  cat("Result: NO significant difference in cognitive switching cost between groups (p >= 0.05)\n")
}

## Result: SIGNIFICANT difference in cognitive switching cost between groups (p < 0.05)
## Direction: YA group has LOWER switching cost (better executive function) than OA group

```

Test 11: Corsi Block Scores

```

cat("Corsi Block Test: Visuospatial Working Memory\n")

## Corsi Block Test: Visuospatial Working Memory

cat("Using: Corsi_Score_Total\n")

## Using: Corsi_Score_Total

cat("Note: Higher scores = BETTER visuospatial working memory\n\n")

## Note: Higher scores = BETTER visuospatial working memory

# Summary statistics by group
corsi_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(Corsi_Score_Total)),
    Mean = mean(Corsi_Score_Total, na.rm = TRUE),
    SD = sd(Corsi_Score_Total, na.rm = TRUE),
    Median = median(Corsi_Score_Total, na.rm = TRUE),
    Q1 = quantile(Corsi_Score_Total, 0.25, na.rm = TRUE),
    Q3 = quantile(Corsi_Score_Total, 0.75, na.rm = TRUE),
    Min = min(Corsi_Score_Total, na.rm = TRUE),
    Max = max(Corsi_Score_Total, na.rm = TRUE)
  )

cat("Summary Statistics:\n")

## Summary Statistics:

print(corsi_summary)

## # A tibble: 2 x 9
##   Group     n   Mean    SD Median    Q1    Q3   Min   Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30  70.9  9.83    73   63.2  76.8    54    89
## 2 YA      30  89.2 10.4     87.5  83.2   95     66   115

```

```

cat("\n")

# Filter out NA values for testing
corsi_data_clean <- merged_data %>%
  filter(!is.na(Corsi_Score_Total), !is.na(Group))

ya_corsi <- corsi_data_clean %>% filter(Group == "YA") %>% pull(Corsi_Score_Total)
oa_corsi <- corsi_data_clean %>% filter(Group == "OA") %>% pull(Corsi_Score_Total)

# Check normality with Shapiro-Wilk test
cat("Normality Tests (Shapiro-Wilk):\n")

## Normality Tests (Shapiro-Wilk):

shapiro_ya <- shapiro.test(ya_corsi)
shapiro_oa <- shapiro.test(oa_corsi)
cat("YA: W =", round(shapiro_ya$statistic, 4), ", p =", shapiro_ya$p.value)

## YA: W = 0.9669 , p = 0.4592182

if(shapiro_ya$p.value < 0.05) cat("(non-normal)") else cat("(normal)")

## (normal)

cat("\n")

cat("OA: W =", round(shapiro_oa$statistic, 4), ", p =", shapiro_oa$p.value)

## OA: W = 0.9648 , p = 0.407813

if(shapiro_oa$p.value < 0.05) cat("(non-normal)") else cat("(normal)")

## (normal)

cat("\n\n")

# Check homogeneity of variance with Levene's test
levene_test <- leveneTest(Corsi_Score_Total ~ Group, data = corsi_data_clean)
cat("Levene's Test for Homogeneity of Variance:\n")

## Levene's Test for Homogeneity of Variance:

cat("F =", round(levene_test$`F value`[1], 4), ", p =", levene_test$`Pr(>F)`[1])

## F = 0.0272 , p = 0.8695673

```

```

if(levene_test$`Pr(>F)`[1] < 0.05) cat(" (variances unequal)") else cat(" (variances equal)")

## (variances equal)

cat("\n\n")

# Decide which test to use
use_welch <- levene_test$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_ya$p.value < 0.05 | shapiro_oa$p.value < 0.05

if(use_nonparametric) {
  cat(" Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(ya_corsi, oa_corsi, exact = FALSE)
  cat("Mann-Whitney U Test Results:\n")
  cat("W statistic =", mw_test$statistic, "\n")
  cat("p-value =", mw_test$p.value, "\n")

  # Effect size
  n1 <- length(ya_corsi)
  n2 <- length(oa_corsi)
  r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
  cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

  # Report IQR for each group
  cat("\nIQR by group:\n")
  cat("YA: [", quantile(ya_corsi, 0.25), ", ", quantile(ya_corsi, 0.75), "]\n", sep = "")
  cat("OA: [", quantile(oa_corsi, 0.25), ", ", quantile(oa_corsi, 0.75), "]\n", sep = "")

  p_value <- mw_test$p.value

} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
  t_test <- t.test(ya_corsi, oa_corsi, var.equal = FALSE)
  cat("Welch's t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "]\n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_corsi)^2 + sd(oa_corsi)^2) / 2)
  cohens_d <- (mean(ya_corsi) - mean(oa_corsi)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value

} else {
  cat("Assumptions met. Using standard independent samples t-test.\n\n")
  t_test <- t.test(ya_corsi, oa_corsi, var.equal = TRUE)
  cat("Independent Samples t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")

```

```

cat("95% CI for difference: [", t_test$conf.int[1], ", ", ", t_test$conf.int[2], "]\\n", sep = "")

# Effect size (Cohen's d)
pooled_sd <- sqrt((sd(ya_corsi)^2 + sd(oa_corsi)^2) / 2)
cohens_d <- (mean(ya_corsi) - mean(oa_corsi)) / pooled_sd
cat("Cohen's d =", round(cohens_d, 3), "\\n")

p_value <- t_test$p.value
}

## Assumptions met. Using standard independent samples t-test.
##
## Independent Samples t-test Results:
## t = 7.006708
## df = 58
## p-value = 2.85609e-09
## 95% CI for difference: [13.09576, 23.57091]
## Cohen's d = 1.809

# Interpretation
cat("\\n--- INTERPRETATION ---\\n")

##
## --- INTERPRETATION ---

if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in Corsi Block performance between groups (p < 0.05)\\n")
  if(mean(ya_corsi) > mean(oa_corsi)) {
    cat("Direction: YA group has BETTER visuospatial working memory than OA group\\n")
  } else {
    cat("Direction: OA group has BETTER visuospatial working memory than YA group\\n")
  }
} else {
  cat("Result: NO significant difference in Corsi Block performance between groups (p >= 0.05)\\n")
}

## Result: SIGNIFICANT difference in Corsi Block performance between groups (p < 0.05)
## Direction: YA group has BETTER visuospatial working memory than OA group

```

Test 12: Pre-VR SSQ Scores

```

cat("SSQ Pre: Simulator Sickness Questionnaire - Baseline\\n")

## SSQ Pre: Simulator Sickness Questionnaire - Baseline

cat("Note: Higher scores = MORE simulator sickness symptoms\\n\\n")

## Note: Higher scores = MORE simulator sickness symptoms

```

```

# Summary statistics by group
ssq_pre_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(SSQ_Pre)),
    Mean = mean(SSQ_Pre, na.rm = TRUE),
    SD = sd(SSQ_Pre, na.rm = TRUE),
    Median = median(SSQ_Pre, na.rm = TRUE),
    Q1 = quantile(SSQ_Pre, 0.25, na.rm = TRUE),
    Q3 = quantile(SSQ_Pre, 0.75, na.rm = TRUE),
    Min = min(SSQ_Pre, na.rm = TRUE),
    Max = max(SSQ_Pre, na.rm = TRUE)
  )

cat("Summary Statistics:\n")

## Summary Statistics:

print(ssq_pre_summary)

## # A tibble: 2 x 9
##   Group     n   Mean    SD Median    Q1    Q3   Min   Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30  2.77  3.63     1     0     4     0    13
## 2 YA      30  2.53  2.27     2     0     4     0     7

cat("\n")

# Filter out NA values for testing
ssq_pre_data_clean <- merged_data %>%
  filter(!is.na(SSQ_Pre), !is.na(Group))

ya_ssq_pre <- ssq_pre_data_clean %>% filter(Group == "YA") %>% pull(SSQ_Pre)
oa_ssq_pre <- ssq_pre_data_clean %>% filter(Group == "OA") %>% pull(SSQ_Pre)

# Check normality with Shapiro-Wilk test
cat("Normality Tests (Shapiro-Wilk):\n")

## Normality Tests (Shapiro-Wilk):

shapiro_ya <- shapiro.test(ya_ssq_pre)
shapiro_oa <- shapiro.test(oa_ssq_pre)
cat("YA: W =", round(shapiro_ya$statistic, 4), ", p =", shapiro_ya$p.value)

## YA: W = 0.8979 , p = 0.007457235

if(shapiro_ya$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (non-normal)

```

```

cat("\n")

cat("OA: W =", round(shapiro_oa$statistic, 4), ", p =", shapiro_oa$p.value)

## OA: W = 0.7802 , p = 2.91558e-05

if(shapiro_oa$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (non-normal)

cat("\n\n")

# Check homogeneity of variance with Levene's test
levene_test <- leveneTest(SSQ_Pre ~ Group, data = ssq_pre_data_clean)
cat("Levene's Test for Homogeneity of Variance:\n")

## Levene's Test for Homogeneity of Variance:

cat("F =", round(levene_test$`F value`[1], 4), ", p =", levene_test$`Pr(>F)`[1])

## F = 1.2886 , p = 0.2609821

if(levene_test$`Pr(>F)`[1] < 0.05) cat(" (variances unequal)") else cat(" (variances equal)")

## (variances equal)

cat("\n\n")

# Decide which test to use
use_welch <- levene_test$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_ya$p.value < 0.05 | shapiro_oa$p.value < 0.05

if(use_nonparametric) {
  cat(" Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(ya_ssq_pre, oa_ssq_pre, exact = FALSE)
  cat("Mann-Whitney U Test Results:\n")
  cat("W statistic =", mw_test$statistic, "\n")
  cat("p-value =", mw_test$p.value, "\n")

  # Effect size
  n1 <- length(ya_ssq_pre)
  n2 <- length(oa_ssq_pre)
  r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
  cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

  # Report IQR for each group
  cat("\nIQR by group:\n")
  cat("YA: [", quantile(ya_ssq_pre, 0.25), ", ", quantile(ya_ssq_pre, 0.75), "]\n", sep = "")
  cat("OA: [", quantile(oa_ssq_pre, 0.25), ", ", quantile(oa_ssq_pre, 0.75), "]\n", sep = "")
}

```

```

p_value <- mw_test$p.value

} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
  t_test <- t.test(ya_ssq_pre, oa_ssq_pre, var.equal = FALSE)
  cat("Welch's t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "] \n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_ssq_pre)^2 + sd(oa_ssq_pre)^2) / 2)
  cohens_d <- (mean(ya_ssq_pre) - mean(oa_ssq_pre)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value

} else {
  cat("Assumptions met. Using standard independent samples t-test.\n\n")
  t_test <- t.test(ya_ssq_pre, oa_ssq_pre, var.equal = TRUE)
  cat("Independent Samples t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "] \n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_ssq_pre)^2 + sd(oa_ssq_pre)^2) / 2)
  cohens_d <- (mean(ya_ssq_pre) - mean(oa_ssq_pre)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value
}

## Data violates normality assumption. Using Mann-Whitney U test.
##
## Mann-Whitney U Test Results:
## W statistic = 482
## p-value = 0.6329784
## Rank-biserial correlation = -0.071
##
## IQR by group:
## YA: [0, 4]
## OA: [0, 4]

# Interpretation
cat("\n--- INTERPRETATION ---\n")

##
## --- INTERPRETATION ---

```

```

if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in baseline SSQ between groups (p < 0.05)\n")
  if(mean(ya_ssq_pre) > mean(oa_ssq_pre)) {
    cat("Direction: YA group has MORE baseline symptoms than OA group\n")
  } else {
    cat("Direction: OA group has MORE baseline symptoms than YA group\n")
  }
} else {
  cat("Result: NO significant difference in baseline SSQ between groups (p >= 0.05)\n")
  cat("(This is expected - both groups should be similar before VR exposure)\n")
}

```

```

## Result: NO significant difference in baseline SSQ between groups (p >= 0.05)
## (This is expected - both groups should be similar before VR exposure)

```

Test 13: Post-VR SSQ Scores

```
cat("SSQ Post: Simulator Sickness Questionnaire - After VR\n")
```

```
## SSQ Post: Simulator Sickness Questionnaire - After VR
```

```
cat("Note: Higher scores = MORE simulator sickness symptoms\n\n")
```

```
## Note: Higher scores = MORE simulator sickness symptoms
```

```

# Summary statistics by group
ssq_post_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(SSQ_Post)),
    Mean = mean(SSQ_Post, na.rm = TRUE),
    SD = sd(SSQ_Post, na.rm = TRUE),
    Median = median(SSQ_Post, na.rm = TRUE),
    Q1 = quantile(SSQ_Post, 0.25, na.rm = TRUE),
    Q3 = quantile(SSQ_Post, 0.75, na.rm = TRUE),
    Min = min(SSQ_Post, na.rm = TRUE),
    Max = max(SSQ_Post, na.rm = TRUE)
  )

```

```
cat("Summary Statistics:\n")
```

```
## Summary Statistics:
```

```
print(ssq_post_summary)
```

```

## # A tibble: 2 x 9
##   Group     n   Mean    SD Median     Q1     Q3    Min    Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30  3.7   4.35    2     0  5.75    0    17
## 2 YA      30  3.53  4.13    1.5    0  6.75    0    14

```

```

cat("\n")

# Filter out NA values for testing
ssq_post_data_clean <- merged_data %>%
  filter(!is.na(SSQ_Post), !is.na(Group))

ya_ssq_post <- ssq_post_data_clean %>% filter(Group == "YA") %>% pull(SSQ_Post)
oa_ssq_post <- ssq_post_data_clean %>% filter(Group == "OA") %>% pull(SSQ_Post)

# Check normality with Shapiro-Wilk test
cat("Normality Tests (Shapiro-Wilk):\n")

## Normality Tests (Shapiro-Wilk):

shapiro_ya <- shapiro.test(ya_ssq_post)
shapiro_oa <- shapiro.test(oa_ssq_post)
cat("YA: W =", round(shapiro_ya$statistic, 4), ", p =", shapiro_ya$p.value)

## YA: W = 0.8275 , p = 0.0002166613

if(shapiro_ya$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (non-normal)

cat("\n")

cat("OA: W =", round(shapiro_oa$statistic, 4), ", p =", shapiro_oa$p.value)

## OA: W = 0.8255 , p = 0.0001978725

if(shapiro_oa$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (non-normal)

cat("\n\n")

# Check homogeneity of variance with Levene's test
levene_test <- leveneTest(SSQ_Post ~ Group, data = ssq_post_data_clean)
cat("Levene's Test for Homogeneity of Variance:\n")

## Levene's Test for Homogeneity of Variance:

cat("F =", round(levene_test$`F value`[1], 4), ", p =", levene_test$`Pr(>F)`[1])

## F = 0.0144 , p = 0.9050047

```

```

if(levene_test$`Pr(>F)`[1] < 0.05) cat(" (variances unequal)") else cat(" (variances equal)")

## (variances equal)

cat("\n\n")

# Decide which test to use
use_welch <- levene_test$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_ya$p.value < 0.05 | shapiro_oa$p.value < 0.05

if(use_nonparametric) {
  cat(" Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(ya_ssq_post, oa_ssq_post, exact = FALSE)
  cat("Mann-Whitney U Test Results:\n")
  cat("W statistic =", mw_test$statistic, "\n")
  cat("p-value =", mw_test$p.value, "\n")

  # Effect size
  n1 <- length(ya_ssq_post)
  n2 <- length(oa_ssq_post)
  r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
  cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

  # Report IQR for each group
  cat("\nIQR by group:\n")
  cat("YA: [", quantile(ya_ssq_post, 0.25), ", ", quantile(ya_ssq_post, 0.75), "]\n", sep = "")
  cat("OA: [", quantile(oa_ssq_post, 0.25), ", ", quantile(oa_ssq_post, 0.75), "]\n", sep = "")

  p_value <- mw_test$p.value

} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
  t_test <- t.test(ya_ssq_post, oa_ssq_post, var.equal = FALSE)
  cat("Welch's t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "]\n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_ssq_post)^2 + sd(oa_ssq_post)^2) / 2)
  cohens_d <- (mean(ya_ssq_post) - mean(oa_ssq_post)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value

} else {
  cat("Assumptions met. Using standard independent samples t-test.\n\n")
  t_test <- t.test(ya_ssq_post, oa_ssq_post, var.equal = TRUE)
  cat("Independent Samples t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
}

```

```

cat("95% CI for difference: [", t_test$conf.int[1], ", ", ", t_test$conf.int[2], "]\\n", sep = "")

# Effect size (Cohen's d)
pooled_sd <- sqrt((sd(ya_ssq_post)^2 + sd(oa_ssq_post)^2) / 2)
cohens_d <- (mean(ya_ssq_post) - mean(oa_ssq_post)) / pooled_sd
cat("Cohen's d =", round(cohens_d, 3), "\\n")

p_value <- t_test$p.value
}

## Data violates normality assumption. Using Mann-Whitney U test.
##
## Mann-Whitney U Test Results:
## W statistic = 428.5
## p-value = 0.7505011
## Rank-biserial correlation = 0.048
##
## IQR by group:
## YA: [0, 6.75]
## OA: [0, 5.75]

# Interpretation
cat("\\n--- INTERPRETATION ---\\n")

##
## --- INTERPRETATION ---

if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in post-VR SSQ between groups (p < 0.05)\\n")
  if(mean(ya_ssq_post) > mean(oa_ssq_post)) {
    cat("Direction: YA group has MORE post-VR symptoms than OA group\\n")
  } else {
    cat("Direction: OA group has MORE post-VR symptoms than YA group\\n")
  }
} else {
  cat("Result: NO significant difference in post-VR SSQ between groups (p >= 0.05)\\n")
}

## Result: NO significant difference in post-VR SSQ between groups (p >= 0.05)

```

Test 14: SSQ Change Scores

```

cat("SSQ Change: Simulator Sickness Change (Post - Pre)\\n")

## SSQ Change: Simulator Sickness Change (Post - Pre)

cat("Note: Positive values = INCREASE in symptoms, Negative = DECREASE\\n\\n")

## Note: Positive values = INCREASE in symptoms, Negative = DECREASE

```

```

# Summary statistics by group
ssq_diff_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(SSQ_Diff)),
    Mean = mean(SSQ_Diff, na.rm = TRUE),
    SD = sd(SSQ_Diff, na.rm = TRUE),
    Median = median(SSQ_Diff, na.rm = TRUE),
    Q1 = quantile(SSQ_Diff, 0.25, na.rm = TRUE),
    Q3 = quantile(SSQ_Diff, 0.75, na.rm = TRUE),
    Min = min(SSQ_Diff, na.rm = TRUE),
    Max = max(SSQ_Diff, na.rm = TRUE)
  )

cat("Summary Statistics:\n")

## Summary Statistics:

print(ssq_diff_summary)

## # A tibble: 2 x 9
##   Group     n   Mean    SD Median    Q1    Q3   Min   Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30  0.933  4.35     0   -1     3   -6    12
## 2 YA      30  1       3.31     0     0     3   -5    10

cat("\n")

# Filter out NA values for testing
ssq_diff_data_clean <- merged_data %>%
  filter(!is.na(SSQ_Diff), !is.na(Group))

ya_ssq_diff <- ssq_diff_data_clean %>% filter(Group == "YA") %>% pull(SSQ_Diff)
oa_ssq_diff <- ssq_diff_data_clean %>% filter(Group == "OA") %>% pull(SSQ_Diff)

# Check normality with Shapiro-Wilk test
cat("Normality Tests (Shapiro-Wilk):\n")

## Normality Tests (Shapiro-Wilk):

shapiro_ya <- shapiro.test(ya_ssq_diff)
shapiro_oa <- shapiro.test(oa_ssq_diff)
cat("YA: W =", round(shapiro_ya$statistic, 4), ", p =", shapiro_ya$p.value)

## YA: W = 0.942 , p = 0.1026877

if(shapiro_ya$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

##  (normal)

```

```

cat("\n")

cat("OA: W =", round(shapiro_oa$statistic, 4), ", p =", shapiro_oa$p.value)

## OA: W = 0.9268 , p = 0.04048631

if(shapiro_oa$p.value < 0.05) cat(" (non-normal)") else cat(" (normal)")

## (non-normal)

cat("\n\n")

# Check homogeneity of variance with Levene's test
levene_test <- leveneTest(SSQ_Diff ~ Group, data = ssq_diff_data_clean)
cat("Levene's Test for Homogeneity of Variance:\n")

## Levene's Test for Homogeneity of Variance:

cat("F =", round(levene_test$`F value`[1], 4), ", p =", levene_test$`Pr(>F)`[1])

## F = 0.788 , p = 0.378359

if(levene_test$`Pr(>F)`[1] < 0.05) cat(" (variances unequal)") else cat(" (variances equal)")

## (variances equal)

cat("\n\n")

# Decide which test to use
use_welch <- levene_test$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_ya$p.value < 0.05 | shapiro_oa$p.value < 0.05

if(use_nonparametric) {
  cat(" Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(ya_ssq_diff, oa_ssq_diff, exact = FALSE)
  cat("Mann-Whitney U Test Results:\n")
  cat("W statistic =", mw_test$statistic, "\n")
  cat("p-value =", mw_test$p.value, "\n")

  # Effect size
  n1 <- length(ya_ssq_diff)
  n2 <- length(oa_ssq_diff)
  r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
  cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

  # Report IQR for each group
  cat("\nIQR by group:\n")
  cat("YA: [", quantile(ya_ssq_diff, 0.25), ", ", quantile(ya_ssq_diff, 0.75), "]\n", sep = "")
  cat("OA: [", quantile(oa_ssq_diff, 0.25), ", ", quantile(oa_ssq_diff, 0.75), "]\n", sep = "")
}

```

```

p_value <- mw_test$p.value

} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
  t_test <- t.test(ya_ssq_diff, oa_ssq_diff, var.equal = FALSE)
  cat("Welch's t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "] \n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_ssq_diff)^2 + sd(oa_ssq_diff)^2) / 2)
  cohens_d <- (mean(ya_ssq_diff) - mean(oa_ssq_diff)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value

} else {
  cat("Assumptions met. Using standard independent samples t-test.\n\n")
  t_test <- t.test(ya_ssq_diff, oa_ssq_diff, var.equal = TRUE)
  cat("Independent Samples t-test Results:\n")
  cat("t =", t_test$statistic, "\n")
  cat("df =", t_test$parameter, "\n")
  cat("p-value =", t_test$p.value, "\n")
  cat("95% CI for difference: [", t_test$conf.int[1], ", ", t_test$conf.int[2], "] \n", sep = "")

  # Effect size (Cohen's d)
  pooled_sd <- sqrt((sd(ya_ssq_diff)^2 + sd(oa_ssq_diff)^2) / 2)
  cohens_d <- (mean(ya_ssq_diff) - mean(oa_ssq_diff)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")

  p_value <- t_test$p.value
}

## Data violates normality assumption. Using Mann-Whitney U test.
##
## Mann-Whitney U Test Results:
## W statistic = 473
## p-value = 0.7348223
## Rank-biserial correlation = -0.051
##
## IQR by group:
## YA: [0, 3]
## OA: [-1, 3]

# Interpretation
cat("\n--- INTERPRETATION ---\n")

##
## --- INTERPRETATION ---

```

```

if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in SSQ change between groups (p < 0.05)\n")
  if(mean(ya_ssq_diff) > mean(oa_ssq_diff)) {
    cat("Direction: YA group had GREATER increase in symptoms than OA group\n")
  } else {
    cat("Direction: OA group had GREATER increase in symptoms than YA group\n")
  }
} else {
  cat("Result: NO significant difference in SSQ change between groups (p >= 0.05)\n")
}

```

```
## Result: NO significant difference in SSQ change between groups (p >= 0.05)
```

Test 15: Pre-Study SSS Scores

```

cat("SSS Pre: Stanford Sleepiness Scale - Baseline\n")

## SSS Pre: Stanford Sleepiness Scale - Baseline

cat("Note: Single-item ordinal scale (1-7), Higher = MORE sleepy\n")

## Note: Single-item ordinal scale (1-7), Higher = MORE sleepy

cat("Analysis: Mann-Whitney U test (appropriate for ordinal data)\n\n")

## Analysis: Mann-Whitney U test (appropriate for ordinal data)

# Summary statistics by group (emphasizing median for ordinal data)
sss_pre_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(SSS_Pre)),
    Median = median(SSS_Pre, na.rm = TRUE),
    Q1 = quantile(SSS_Pre, 0.25, na.rm = TRUE),
    Q3 = quantile(SSS_Pre, 0.75, na.rm = TRUE),
    Mean = mean(SSS_Pre, na.rm = TRUE),
    SD = sd(SSS_Pre, na.rm = TRUE),
    Min = min(SSS_Pre, na.rm = TRUE),
    Max = max(SSS_Pre, na.rm = TRUE)
  )

cat("Summary Statistics:\n")

## Summary Statistics:

print(sss_pre_summary)

```

```
## # A tibble: 2 x 9
##   Group     n Median    Q1    Q3  Mean    SD  Min  Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30     1     1  1.75  1.33  0.606     1     3
## 2 YA      30     2     1  2     2     1.05     1     6
```

```
cat("\n")
```

```
# Show frequency distribution
cat("Frequency Distribution:\n")
```

```
## Frequency Distribution:
```

```
freq_table <- table(merged_data$SSS_Pre, merged_data$Group)
print(freq_table)
```

```
##
##          OA  YA
## 1      22 10
## 2      6 13
## 3      2  6
## 6      0  1
```

```
cat("\n")
```

```
# Filter out NA values for testing
sss_pre_data_clean <- merged_data %>%
  filter(!is.na(SSS_Pre), !is.na(Group))

ya_sss_pre <- sss_pre_data_clean %>% filter(Group == "YA") %>% pull(SSS_Pre)
oa_sss_pre <- sss_pre_data_clean %>% filter(Group == "OA") %>% pull(SSS_Pre)

# Mann-Whitney U test
mw_test <- wilcox.test(ya_sss_pre, oa_sss_pre, exact = FALSE)

cat("Mann-Whitney U Test Results:\n")
```

```
## Mann-Whitney U Test Results:
```

```
cat("W statistic =", mw_test$statistic, "\n")
```

```
## W statistic = 639
```

```
cat("p-value =", mw_test$p.value, "\n")
```

```
## p-value = 0.002014047
```

```

# Effect size (rank-biserial correlation)
n1 <- length(ya_sss_pre)
n2 <- length(oa_sss_pre)
r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

## Rank-biserial correlation = -0.42

# Report IQR for each group
cat("\nIQR by group:\n")

## IQR by group:

cat("YA: [", quantile(ya_sss_pre, 0.25), ", ", quantile(ya_sss_pre, 0.75), "]\n", sep = "")

## YA: [1, 2]

cat("OA: [", quantile(oa_sss_pre, 0.25), ", ", quantile(oa_sss_pre, 0.75), "]\n", sep = "")

## OA: [1, 1.75]

# Interpretation
cat("\n--- INTERPRETATION ---\n")

## --- INTERPRETATION ---

if(mw_test$p.value < 0.05) {
  cat("Result: SIGNIFICANT difference in baseline sleepiness between groups (p < 0.05)\n")
  if(median(ya_sss_pre) > median(oa_sss_pre)) {
    cat("Direction: YA group was MORE sleepy at baseline (Median YA =", median(ya_sss_pre),
        ", OA =", median(oa_sss_pre), ")\n")
  } else if(median(ya_sss_pre) < median(oa_sss_pre)) {
    cat("Direction: OA group was MORE sleepy at baseline (Median YA =", median(ya_sss_pre),
        ", OA =", median(oa_sss_pre), ")\n")
  } else {
    cat("Direction: Medians are equal, but distributions differ\n")
  }
} else {
  cat("Result: NO significant difference in baseline sleepiness between groups (p >= 0.05)\n")
  cat("Median YA =", median(ya_sss_pre), ", Median OA =", median(oa_sss_pre), "\n")
  cat("(This is expected - both groups should be similar before the study)\n")
}

## Result: SIGNIFICANT difference in baseline sleepiness between groups (p < 0.05)
## Direction: YA group was MORE sleepy at baseline (Median YA = 2 , OA = 1 )

```

Test 16: Post-Study SSS Scores

```

cat("SSS Post: Stanford Sleepiness Scale - After Study\n")

## SSS Post: Stanford Sleepiness Scale - After Study

cat("Note: Single-item ordinal scale (1-7), Higher = MORE sleepy\n")

## Note: Single-item ordinal scale (1-7), Higher = MORE sleepy

cat("Analysis: Mann-Whitney U test (appropriate for ordinal data)\n\n")

## Analysis: Mann-Whitney U test (appropriate for ordinal data)

# Summary statistics by group (emphasizing median for ordinal data)
sss_post_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(
    n = sum(!is.na(SSS_Post)),
    Median = median(SSS_Post, na.rm = TRUE),
    Q1 = quantile(SSS_Post, 0.25, na.rm = TRUE),
    Q3 = quantile(SSS_Post, 0.75, na.rm = TRUE),
    Mean = mean(SSS_Post, na.rm = TRUE),
    SD = sd(SSS_Post, na.rm = TRUE),
    Min = min(SSS_Post, na.rm = TRUE),
    Max = max(SSS_Post, na.rm = TRUE)
  )

cat("Summary Statistics:\n")

## Summary Statistics:

print(sss_post_summary)

## # A tibble: 2 x 9
##   Group     n Median     Q1     Q3   Mean     SD   Min   Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30     1     1     2  1.43  0.774     1     4
## 2 YA      30     1     1     3  1.83  1.12      0     4

cat("\n")

# Show frequency distribution
cat("Frequency Distribution:\n")

## Frequency Distribution:

freq_table <- table(merged_data$SSS_Post, merged_data$Group)
print(freq_table)

```

```

##          OA YA
##  0   0  1
##  1  21 15
##  2   6  5
##  3   2  6
##  4   1  3

cat("\n")

# Filter out NA values for testing
sss_post_data_clean <- merged_data %>%
  filter(!is.na(SSS_Post), !is.na(Group))

ya_sss_post <- sss_post_data_clean %>% filter(Group == "YA") %>% pull(SSS_Post)
oa_sss_post <- sss_post_data_clean %>% filter(Group == "OA") %>% pull(SSS_Post)

# Mann-Whitney U test
mw_test <- wilcox.test(ya_sss_post, oa_sss_post, exact = FALSE)

cat("Mann-Whitney U Test Results:\n")

## Mann-Whitney U Test Results:

cat("W statistic =", mw_test$statistic, "\n")

## W statistic = 534

cat("p-value =", mw_test$p.value, "\n")

## p-value = 0.1609297

# Effect size (rank-biserial correlation)
n1 <- length(ya_sss_post)
n2 <- length(oa_sss_post)
r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

## Rank-biserial correlation = -0.187

# Report IQR for each group
cat("\nIQR by group:\n")

## IQR by group:

cat("YA: [", quantile(ya_sss_post, 0.25), ", ", quantile(ya_sss_post, 0.75), "]\\n", sep = "")

## YA: [1, 3]

```

```

cat("OA: [", quantile(oa_sss_post, 0.25), ", ", quantile(oa_sss_post, 0.75), "]\\n", sep = "")

## OA: [1, 2]

# Interpretation
cat("\n--- INTERPRETATION ---\\n")

## 
## --- INTERPRETATION ---

if(mw_test$p.value < 0.05) {
  cat("Result: SIGNIFICANT difference in post-study sleepiness between groups (p < 0.05)\\n")
  if(median(ya_sss_post) > median(oa_sss_post)) {
    cat("Direction: YA group was MORE sleepy after study (Median YA =", median(ya_sss_post),
        ", OA =", median(oa_sss_post), ")\\n")
  } else if(median(ya_sss_post) < median(oa_sss_post)) {
    cat("Direction: OA group was MORE sleepy after study (Median YA =", median(ya_sss_post),
        ", OA =", median(oa_sss_post), ")\\n")
  } else {
    cat("Direction: Medians are equal, but distributions differ\\n")
  }
} else {
  cat("Result: NO significant difference in post-study sleepiness between groups (p >= 0.05)\\n")
  cat("Median YA =", median(ya_sss_post), ", Median OA =", median(oa_sss_post), "\\n")
}
}

## Result: NO significant difference in post-study sleepiness between groups (p >= 0.05)
## Median YA = 1 , Median OA = 1

```

Test 17: SSS Change Scores

```

cat("SSS Change: Sleepiness Change (Post - Pre)\\n")

## SSS Change: Sleepiness Change (Post - Pre)

cat("Note: Positive values = INCREASE in sleepiness, Negative = DECREASE\\n")

## Note: Positive values = INCREASE in sleepiness, Negative = DECREASE

cat("Analysis: Mann-Whitney U test (appropriate for ordinal data)\\n\\n")

## Analysis: Mann-Whitney U test (appropriate for ordinal data)

# Summary statistics by group (emphasizing median for ordinal data)
sss_diff_summary <- merged_data %>%
  group_by(Group) %>%
  summarise(

```

```

n = sum(!is.na(SSS_Diff)),
Median = median(SSS_Diff, na.rm = TRUE),
Q1 = quantile(SSS_Diff, 0.25, na.rm = TRUE),
Q3 = quantile(SSS_Diff, 0.75, na.rm = TRUE),
Mean = mean(SSS_Diff, na.rm = TRUE),
SD = sd(SSS_Diff, na.rm = TRUE),
Min = min(SSS_Diff, na.rm = TRUE),
Max = max(SSS_Diff, na.rm = TRUE)
)

cat("Summary Statistics:\n")

## Summary Statistics:

print(sss_diff_summary)

## # A tibble: 2 x 9
##   Group     n Median    Q1    Q3    Mean     SD    Min    Max
##   <chr> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA      30     0     0     0    0.1    0.662   -2     1
## 2 YA      30     0   -0.75    0.75 -0.167  1.46    -5     2

cat("\n")

# Show frequency distribution
cat("Frequency Distribution:\n")

## Frequency Distribution:

freq_table <- table(merged_data$SSS_Diff, merged_data$Group)
print(freq_table)

## 
##      OA  YA
##      -5  0  1
##      -3  0  1
##      -2  1  2
##      -1  2  4
##      0  20 14
##      1   7  5
##      2   0  3

cat("\n")

# Filter out NA values for testing
sss_diff_data_clean <- merged_data %>%
  filter(!is.na(SSS_Diff), !is.na(Group))

ya_sss_diff <- sss_diff_data_clean %>% filter(Group == "YA") %>% pull(SSS_Diff)
oa_sss_diff <- sss_diff_data_clean %>% filter(Group == "OA") %>% pull(SSS_Diff)

```

```

# Mann-Whitney U test
mw_test <- wilcox.test(ya_sss_diff, oa_sss_diff, exact = FALSE)

cat("Mann-Whitney U Test Results:\n")

## Mann-Whitney U Test Results:

cat("W statistic =", mw_test$statistic, "\n")

## W statistic = 413.5

cat("p-value =", mw_test$p.value, "\n")

## p-value = 0.5540237

# Effect size (rank-biserial correlation)
n1 <- length(ya_sss_diff)
n2 <- length(oa_sss_diff)
r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")

## Rank-biserial correlation = 0.081

# Report IQR for each group
cat("\nIQR by group:\n")

## 
## IQR by group:

cat("YA: [", quantile(ya_sss_diff, 0.25), ", ", quantile(ya_sss_diff, 0.75), "] \n", sep = "")

## YA: [-0.75, 0.75]

cat("OA: [", quantile(oa_sss_diff, 0.25), ", ", quantile(oa_sss_diff, 0.75), "] \n", sep = "")

## OA: [0, 0]

# Interpretation
cat("\n--- INTERPRETATION ---\n")

## 
## --- INTERPRETATION ---

```

```

if(mw_test$p.value < 0.05) {
  cat("Result: SIGNIFICANT difference in sleepiness change between groups (p < 0.05)\n")
  if(median(ya_sss_diff) > median(oa_sss_diff)) {
    cat("Direction: YA group had GREATER increase in sleepiness (Median change: YA =", median(ya_sss_diff),
        ", OA =", median(oa_sss_diff), ")\n")
  } else if(median(ya_sss_diff) < median(oa_sss_diff)) {
    cat("Direction: OA group had GREATER increase in sleepiness (Median change: YA =", median(ya_sss_diff),
        ", OA =", median(oa_sss_diff), ")\n")
  } else {
    cat("Direction: Median changes are equal, but distributions differ\n")
  }
} else {
  cat("Result: NO significant difference in sleepiness change between groups (p >= 0.05)\n")
  cat("Median change: YA =", median(ya_sss_diff), ", OA =", median(oa_sss_diff), "\n")
}
}

## Result: NO significant difference in sleepiness change between groups (p >= 0.05)
## Median change: YA = 0 , OA = 0

```

Session Information

```
sessionInfo()
```

```

## R version 4.3.1 (2023-06-16)
## Platform: x86_64-apple-darwin20 (64-bit)
## Running under: macOS Ventura 13.2.1
##
## Matrix products: default
## BLAS:      /Library/Frameworks/R.framework/Versions/4.3-x86_64/Resources/lib/libRblas.0.dylib
## LAPACK:   /Library/Frameworks/R.framework/Versions/4.3-x86_64/Resources/lib/libRlapack.dylib; LAPACK
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## time zone: America/New_York
## tzcode source: internal
##
## attached base packages:
## [1] stats      graphics   grDevices utils      datasets   methods    base
##
## other attached packages:
## [1] car_3.1-2       carData_3.0-5     reshape2_1.4.4   kableExtra_1.4.0
## [5] knitr_1.43      broom_1.0.5      lubridate_1.9.2 forcats_1.0.0
## [9] stringr_1.5.0   dplyr_1.1.2      purrr_1.0.1     readr_2.1.4
## [13] tidyverse_2.0.0  tibble_3.2.1     ggplot2_3.5.1   tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
## [1] utf8_1.2.3       generics_0.1.3    xml2_1.3.4      stringi_1.7.12
## [5] hms_1.1.3        digest_0.6.32    magrittr_2.0.3   evaluate_0.21
## [9] grid_4.3.1       timechange_0.2.0  fastmap_1.1.1   plyr_1.8.8

```

```
## [13] backports_1.4.1    fansi_1.0.4      viridisLite_0.4.2 scales_1.3.0
## [17] abind_1.4-5        cli_3.6.1       crayon_1.5.2     rlang_1.1.6
## [21] bit64_4.0.5       munsell_0.5.0   withr_2.5.0      yaml_2.3.7
## [25] parallel_4.3.1    tools_4.3.1     tzdb_0.4.0       colorspace_2.1-0
## [29] vctrs_0.6.3       R6_2.5.1       lifecycle_1.0.3  bit_4.0.5
## [33] vroom_1.6.3       pkgconfig_2.0.3 pillar_1.9.0      gtable_0.3.3
## [37] glue_1.6.2        Rcpp_1.0.10     systemfonts_1.0.4 xfun_0.39
## [41] tidyselect_1.2.0   rstudioapi_0.14 htmltools_0.5.5   rmarkdown_2.23
## [45] svglite_2.1.2     compiler_4.3.1
```