NavCity Demographic and Cognitive Correlates Analysis

Your Name

2025-10-10

Load Required Libraries

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

Data Import and Preparation

```r
# 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
# First merge demographic and non_nav data
merged_temp <- demographic_data %>%
  full_join(non_nav, by = "Participant", suffix = c("_demo", "_non_nav")) %>%
  # Keep Group from non_nav data (it's a string like "YA" or "OA")
```

```r
  mutate(Group = coalesce(Group_non_nav,
                          if_else(Group_demo == 1, "YA", "OA"))) %>%
  select(-Group_demo, -Group_non_nav)

# Then merge with navigation data
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
```

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

```
## YA participants: 30
```

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

```
## OA participants: 30
```

```r
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

Group Comparisons (YA vs. OA)

Test 1: Gender Distribution

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

cat("Contingency Table:\n")
```

```
## Contingency Table:
```

```r
print(gender_table)
```

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

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

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

```
## With Totals:
```

```r
print(gender_table_with_totals)
```

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

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

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

```
## Proportions by Group:
```

```r
prop_table <- prop.table(gender_table, margin = 2) # margin=2 means by column (Group)
print(round(prop_table, 3))
```

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

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

```r
# 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:
```

```r
cat(" ² =", chi_test$statistic, "\n")
```

```
##  ² = 0.06857143
```

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

```
## df = 1
```

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

```
## p-value = 0.7934282
```

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

```
##
## Expected Frequencies:
```

```r
print(round(chi_test$expected, 2))
```

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

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

```r
# 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
```

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

```
##
## --- INTERPRETATION ---
```

```r
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)
```

```r
# Check assumption: all expected frequencies >= 5
if(min(chi_test$expected) < 5) {
  cat("  WARNING: Some expected frequencies < 5. Consider Fisher's exact test.\n")

  # Run Fisher's exact test as alternative
  fisher_test <- fisher.test(gender_table)
  cat("\nFisher's Exact Test (alternative):\n")
  cat("p-value =", round(fisher_test$p.value, 4), "\n")
}
```

Test 2: Handedness Distribution

```r
# 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):
```

```r
print(handedness_table_original)
```

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

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

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

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

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

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

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

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

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

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

```r
# 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):
```

```r
print(handedness_table)
```

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

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

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

```
## With Totals:
```

```r
print(handedness_table_with_totals)
```

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

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

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

```
## Proportions by Group:
```

```r
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
```

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

```r
# 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:
```

```r
cat(" ² =", chi_test$statistic, "\n")
```

```
##  ² = 0.130719
```

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

```
## df = 1
```

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

```
## p-value = 0.7176878
```

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

```
##
## Expected Frequencies:
```

```r
print(round(chi_test$expected, 2))
```

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

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

```r
# 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
```

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

```
##
## --- INTERPRETATION ---
```

```r
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)
```

```r
# Check assumptions
if(min(chi_test$expected) < 5) {
  cat("   WARNING: Some expected frequencies < 5. Consider Fisher's exact test.\n")

  # Run Fisher's exact test as alternative
  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

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

```
## VR Experience: Prior VR Exposure
```

```r
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)
```

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

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

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

```
## VR Experience Coding:
```

```r
cat("  0 = Never\n")
```

```
##   0 = Never
```

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

```
##   1 = 1-3 times
```

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

```
##   2 = More than 3 times
```

```r
# 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:
```

```r
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
```

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

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

```
## Frequency Distribution:
```

```r
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
```

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

```r
# 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:
```

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

```
## W statistic = 621
```

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

```
## p-value = 0.00506242
```

```r
# 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
```

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

```
##
## --- INTERPRETATION ---
```

```r
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)) {
```

```r
    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

```r
# Show the coding scheme if known
cat("Video Game Usage (numeric variable)\n\n")
```

```
## Video Game Usage (numeric variable)
```

```r
# 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),
    Min = min(Video_Game_Experience_Quantified, na.rm = TRUE),
    Max = max(Video_Game_Experience_Quantified, na.rm = TRUE)
  )

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

```
## Summary Statistics:
```

```r
print(vg_summary)
```

```
## # A tibble: 2 x 7
##   Group     n  Mean    SD Median   Min   Max
##   <chr> <int> <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1 OA       30 0.567  1.92    0       0     8
## 2 YA       30 2.7    3.39    1.5     0    10
```

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

```r
# 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):
```

```r
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
```

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

```
##  (non-normal)
```

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

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

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

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

```
##  (non-normal)
```

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

```r
# Check homogeneity of variance with Levene's test
library(car)
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:
```

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

```
## F = 13.4959 , p = 0.0005232028
```

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

```
##  (variances unequal)
```

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

```r
# 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")

  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")

  # 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")

  # 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
```

```
# 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),
    Min = min(Exercise_Quantified, na.rm = TRUE),
    Max = max(Exercise_Quantified, na.rm = TRUE)
  )

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

```
## Summary Statistics:
```

```r
print(exercise_summary)
```

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

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

```r
# 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):
```

```r
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
```

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

```
##  (non-normal)
```

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

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

```
## OA: W = 0.951 , p = 0.1800439
```

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

```
##  (normal)
```

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

```r
# 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:
```

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

```
## F = 0.0265 , p = 0.871163
```

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

```
##  (variances equal)
```

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

```r
# 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")

  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")

  # 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 {
```

```r
  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")

  # 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
```

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

```
##
## --- INTERPRETATION ---
```

```r
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

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

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

```r
# 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),
    Min = min(SBSOD, na.rm = TRUE),
    Max = max(SBSOD, na.rm = TRUE)
  )

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

```
## Summary Statistics:
```

```r
print(sbsod_summary)
```

```
## # A tibble: 2 x 7
##   Group     n  Mean    SD Median   Min   Max
##   <chr> <int> <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1 OA       30  4.08 0.925   5.16  3.27  6.87
## 2 YA       30  4.35 1.10    4.5   2.13  6
```

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

```r
# 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):
```

```r
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
```

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

```
##  (normal)
```

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

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

```
## OA: W = 0.9704 , p = 0.5493297
```

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

```
##  (normal)
```

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

```r
# 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:
```

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

```
## F = 2.0401 , p = 0.1585664
```

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

```
##  (variances equal)
```

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

```r
# 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")

  p_value <- mw_test$p.value

} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
```

```r
  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")

  # 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")

  # 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
## Cohen's d = -0.721
```

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

```
##
## --- INTERPRETATION ---
```

```r
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

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

```
## PSQI: Pittsburgh Sleep Quality Index
```

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

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

```r
# 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),
    Min = min(PSQI, na.rm = TRUE),
    Max = max(PSQI, na.rm = TRUE)
  )

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

```
## Summary Statistics:
```

```r
print(psqi_summary)
```

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

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

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

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

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

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

```r
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
```

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

```
##  (non-normal)
```

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

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

```
## OA: W = 0.9554 , p = 0.2347819
```

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

```
##  (normal)
```

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

```r
# 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:
```

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

```
## F = 10.414 , p = 0.002058421
```

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

```
##  (variances unequal)
```

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

```r
# 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)
```

```r
  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")

  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")

  # 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")

  # 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
```

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

```
##
```

```
## --- INTERPRETATION ---
```

```r
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 A Performance

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

```
## Trails Making Test A: Completion Time
```

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

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

```r
# 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),
    Min = min(Trails_A_CT, na.rm = TRUE),
    Max = max(Trails_A_CT, na.rm = TRUE)
  )

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

```
## Summary Statistics:
```

```r
print(trails_summary)
```

```
## # A tibble: 2 x 7
##   Group     n  Mean    SD Median   Min   Max
##   <chr> <int> <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1 OA       30  34.2  6.84   34.7  22.4  48.6
## 2 YA       30  25.9  4.56   24.6  17.1  35.2
```

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

```r
# 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):
```

```r
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
```

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

```
##  (normal)
```

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

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

```
## OA: W = 0.9406 , p = 0.0946305
```

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

```
##  (normal)
```

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

```r
# 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:
```

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

```
## F = 1.9659 , p = 0.1662154
```

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

```
##  (variances equal)
```

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

```r
# 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")

  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")

  # 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")

  # 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
## 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 11: Trails Making 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
```

```r
# 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),
    Min = min(Trails_B_CT, na.rm = TRUE),
    Max = max(Trails_B_CT, na.rm = TRUE)
  )

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

```
## Summary Statistics:
```

```r
print(trails_b_summary)
```

```
## # A tibble: 2 x 7
##   Group     n  Mean    SD Median   Min   Max
##   <chr> <int> <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1 OA       30  59.5  28.9   48.6  26.4 160.
## 2 YA       30  36.9  11.1   35.5  21.5  75.1
```

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

```r
# 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):
```

```r
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
```

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

```
##  (non-normal)
```

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

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

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

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

```
##  (non-normal)
```

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

```r
# 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:
```

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

```
## F = 3.8705 , p = 0.05393151
```

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

```
##  (variances equal)
```

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

```r
# 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")

  p_value <- mw_test$p.value

} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
```

```r
  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")

  # 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")

  # 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
```

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

```
##
## --- INTERPRETATION ---
```

```r
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 11: Trails B-A Difference

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

```
## Trails B-A Difference: Cognitive Switching Cost
```

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

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

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

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

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

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

```
## Summary Statistics:
```

```r
print(trails_ba_summary)
```

```
## # A tibble: 2 x 7
##   Group     n  Mean    SD Median   Min   Max
##   <chr> <int> <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1 OA       30  25.3 26.3   17.3    -3   111.
## 2 YA       30  11.0  9.81   8.45  -2.5  39.9
```

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

```r
# 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):
```

```r
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
```

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

```
##  (non-normal)
```

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

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

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

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

```
##  (non-normal)
```

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

```r
# 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:
```

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

```
## F = 3.536 , p = 0.06507473
```

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

```
##  (variances equal)
```

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

```r
# 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")

  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")

  # 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")

  # 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
```

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

```
##
## --- INTERPRETATION ---
```

```r
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 12: Corsi Block Scores

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

```
## Corsi Block Test: Visuospatial Working Memory
```

```r
cat("Using: Corsi_Score_Total\n")
```

```
## Using: Corsi_Score_Total
```

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

```
## Note: Higher scores = BETTER visuospatial working memory
```

```r
# 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),
```

```r
    Min = min(Corsi_Score_Total, na.rm = TRUE),
    Max = max(Corsi_Score_Total, na.rm = TRUE)
  )

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

```
## Summary Statistics:
```

```r
print(corsi_summary)
```

```
## # A tibble: 2 x 7
##   Group     n  Mean    SD Median   Min   Max
##   <chr> <int> <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1 OA       30  70.9  9.83   73      54    89
## 2 YA       30  89.2 10.4    87.5    66   115
```

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

```r
# 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):
```

```r
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
```

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

```
##  (normal)
```

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

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

```
## OA: W = 0.9648 , p = 0.407813
```

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

```
##  (normal)
```

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

```r
# 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:
```

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

```
## F = 0.0272 , p = 0.8695673
```

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

```
##  (variances equal)
```

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

```r
# 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")

  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")

  # Effect size (Cohen's d)
```

```r
  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")

  # 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
## Cohen's d = 1.809
```

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

```
##
## --- INTERPRETATION ---
```

```r
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 13: 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> <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:
```

```r
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
```

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

```r
# 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:
```

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

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

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

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

```r
# 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
```

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

```
##
## --- INTERPRETATION ---
```

```r
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 14: Post-Study SSS Scores

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

```
## SSS Post: Stanford Sleepiness Scale - After Study
```

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

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

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

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

```r
# 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:
```

```r
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> <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
```

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

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

```
## Frequency Distribution:
```

```r
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
```

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

```r
# 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:
```

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

```
## W statistic = 534
```

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

```
## p-value = 0.1609297
```

```r
# 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
```

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

```
##
## --- INTERPRETATION ---
```

```r
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 15: SSS Change Scores

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

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

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

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

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

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

```r
# 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:
```

```r
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
```

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

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

```
## Frequency Distribution:
```

```r
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
```

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

```r
# 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:
```

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

```
## W statistic = 413.5
```

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

```
## p-value = 0.5540237
```

```r
# 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
```

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

```
##
## --- INTERPRETATION ---
```

```r
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_di
        ", 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_di
        ", 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
```

Test 16: Pre-Study SSS Scores

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

```
## SSS Pre: Stanford Sleepiness Scale - Baseline
```

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

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

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

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

```r
# 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),
    IQR = IQR(SSS_Pre, 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:
```

```r
print(sss_pre_summary)
```

```
## # A tibble: 2 x 8
##   Group     n Median   IQR  Mean    SD   Min   Max
##   <chr> <int>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA       30      1  0.75  1.33 0.606     1     3
## 2 YA       30      2  1     2    1.05      1     6
```

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

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

```
## Frequency Distribution:
```

```r
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
```

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

```r
# 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:
```

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

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

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

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

```r
# 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
```

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

```
##
## --- INTERPRETATION ---
```

```r
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 17: Post-Study SSS Scores

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

```
## SSS Post: Stanford Sleepiness Scale - After Study
```

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

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

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

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

```r
# 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),
    IQR = IQR(SSS_Post, 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:
```

```r
print(sss_post_summary)
```

```
## # A tibble: 2 x 8
##   Group     n Median   IQR  Mean    SD   Min   Max
##   <chr> <int>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA       30      1     1  1.43 0.774     1     4
## 2 YA       30      1     2  1.83 1.12      0     4
```

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

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

```
## Frequency Distribution:
```

```r
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
```

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

```r
# 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:
```

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

```
## W statistic = 534
```

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

```
## p-value = 0.1609297
```

```r
# 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
```

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

```
##
## --- INTERPRETATION ---
```

```r
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 18: SSS Change Scores

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

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

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

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

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

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

```r
# 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),
    IQR = IQR(SSS_Diff, 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:
```

```r
print(sss_diff_summary)
```

```
## # A tibble: 2 x 8
##   Group     n Median   IQR   Mean    SD   Min   Max
##   <chr> <int>  <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 OA       30      0   0    0.1   0.662    -2     1
## 2 YA       30      0   1.5 -0.167 1.46     -5     2
```

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

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

```
## Frequency Distribution:
```

```r
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
```

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

```r
# 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:
```

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

```
## W statistic = 413.5
```

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

```
## p-value = 0.5540237
```

```r
# 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
```

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

```
##
## --- INTERPRETATION ---
```

```r
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_di
        ", 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_di
        ", 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
```