NavCity OA Subgroup Analysis - High vs Low NARA

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,
```

```r
                          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
  )

# FILTER TO ONLY OA PARTICIPANTS
merged_data <- merged_data %>%
  filter(Group == "OA")

# CREATE OA SUBGROUPS BASED ON NARA SCORE
merged_data <- merged_data %>%
  mutate(OA_Subgroup = ifelse(NARA >= 3.5, "OA_High", "OA_Low"))

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

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

```r
cat("OA_High participants (NARA   3.5):", sum(merged_data$OA_Subgroup == "OA_High", na.rm = TRUE), "\n")
```

```
## OA_High participants (NARA   3.5): 16
```

```r
cat("OA_Low participants (NARA < 3.5):", sum(merged_data$OA_Subgroup == "OA_Low", na.rm = TRUE), "\n\n")
```

```
## OA_Low participants (NARA < 3.5): 14
```

```r
# Show NARA distribution
cat("NARA Score Distribution:\n")
```

```
## NARA Score Distribution:
```

```r
cat("Range:", range(merged_data$NARA, na.rm = TRUE), "\n")
```

```
## Range: 1 7
```

```r
cat("Mean:", mean(merged_data$NARA, na.rm = TRUE), "\n")
```

```
## Mean: 3.9
```

```r
cat("Median:", median(merged_data$NARA, na.rm = TRUE), "\n\n")
```

```
## Median: 4.25
```

```r
head(merged_data) %>%
  select(Participant, OA_Subgroup, NARA, SBSOD, Mean_Speed) %>%
  kable() %>%
  kable_styling()
```

Participant	OA_Subgroup	NARA	SBSOD	Mean_Speed
BNC06	OA_High	4.5	6.87	14.922086
BNC10	OA_Low	2.0	6.13	7.488175
BNC13	OA_High	4.5	4.00	9.221490
BNC14	OA_High	5.5	4.53	10.538845
BNC15	OA_High	5.0	5.00	9.300569
BNC16	OA_Low	1.0	3.53	6.656084

OA Subgroup Comparisons (High vs. Low NARA)

Test 1: Gender Distribution

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

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

```
## Contingency Table:
```

```r
print(gender_table)
```

```
##
##     OA_High OA_Low
##   M       7      5
##   W       9      9
```

```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_High OA_Low Sum
##   M         7      5  12
##   W         9      9  18
##   Sum      16     14  30
```

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

```r
# Calculate proportions within each subgroup
cat("Proportions by OA Subgroup:\n")
```

```
## Proportions by OA Subgroup:
```

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

```
##
##     OA_High OA_Low
##   M   0.438  0.357
##   W   0.562  0.643
```

```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 = 0.2008929
```

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

```
## df = 1
```

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

```
## p-value = 0.6540011
```

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

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

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

```
##
##     OA_High OA_Low
##   M     6.4    5.6
##   W     9.6    8.4
```

```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 = 0.082
```

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

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

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

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

```r
# 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

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

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

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

```r
print(handedness_table_original)
```

```
##
##     OA_High OA_Low
##   L       1      1
##   M       1      1
##   R      14     12
```

```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: OA_High =", handedness_table_original["R", "OA_High"], ", OA_Low =", handedness_table
```

```
## Right-handed: OA_High = 14 , OA_Low = 12
```

```r
cat("Left-handed: OA_High =", handedness_table_original["L", "OA_High"], ", OA_Low =", handedness_table_
```

```
## Left-handed: OA_High = 1 , OA_Low = 1
```

```r
cat("Mixed: OA_High =", handedness_table_original["M", "OA_High"], ", OA_Low =", handedness_table_origin
```

```
## Mixed: OA_High = 1 , OA_Low = 1
```

```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$Handedness_Collapsed, merged_data$OA_Subgroup)

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

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

```r
print(handedness_table)
```

```
##
##                    OA_High OA_Low
##   Non-right-handed       2      2
##   Right-handed          14     12
```

```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_High OA_Low Sum
##   Non-right-handed         2      2   4
##   Right-handed            14     12  26
##   Sum                     16     14  30
```

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

```r
# Calculate proportions within each subgroup
cat("Proportions by OA Subgroup:\n")
```

```
## Proportions by OA Subgroup:
```

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

```
##
##                      OA_High OA_Low
##   Non-right-handed     0.125  0.143
##   Right-handed         0.875  0.857
```

```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 = 0.0206044
```

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

```
## df = 1
```

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

```
## p-value = 0.8858619
```

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

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

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

```
##
##                     OA_High OA_Low
##   Non-right-handed    2.13   1.87
##   Right-handed       13.87  12.13
```

```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 = 0.026
```

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

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

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

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

```r
# 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

```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 subgroup (emphasizing median for ordinal data)
vr_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  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
##   OA_Subgroup     n Median    Q1    Q3  Mean    SD   Min   Max
##   <chr>       <int>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16      0     0     0 0.188 0.403     0     1
## 2 OA_Low         14      0     0     1 0.5   0.650     0     2
```

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

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

```
## Frequency Distribution:
```

```r
freq_table <- table(merged_data$VR_Experience_Quantified, merged_data$OA_Subgroup)
print(freq_table)
```

```
##
##     OA_High OA_Low
##   0      13      8
##   1       3      5
##   2       0      1
```

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

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

oa_high_vr <- vr_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(VR_Experience_Quantified)
oa_low_vr <- vr_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(VR_Experience_Quantified)

# Mann-Whitney U test
mw_test <- wilcox.test(oa_high_vr, oa_low_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 = 83.5
```

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

```
## p-value = 0.1452715
```

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

```
## Rank-biserial correlation = 0.254
```

```r
# Report IQR for each subgroup
cat("\nIQR by OA subgroup:\n")
```

```
##
## IQR by OA subgroup:
```

```r
cat("OA_High: [", quantile(oa_high_vr, 0.25), ", ", quantile(oa_high_vr, 0.75), "]\n", sep = "")
```

```
## OA_High: [0, 0]
```

```r
cat("OA_Low: [", quantile(oa_low_vr, 0.25), ", ", quantile(oa_low_vr, 0.75), "]\n", sep = "")
```

```
## OA_Low: [0, 1]
```

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

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

```r
if(mw_test$p.value < 0.05) {
  cat("Result: SIGNIFICANT difference in VR experience between OA subgroups (p < 0.05)\n")
  if(median(oa_high_vr) > median(oa_low_vr)) {
    cat("Direction: OA_High has MORE prior VR experience (Median High =", median(oa_high_vr),
        ", Low =", median(oa_low_vr), ")\n")
  } else if(median(oa_high_vr) < median(oa_low_vr)) {
    cat("Direction: OA_Low has MORE prior VR experience (Median High =", median(oa_high_vr),
        ", Low =", median(oa_low_vr), ")\n")
  } else {
    cat("Direction: Medians are equal, but distributions differ\n")
  }
} else {
  cat("Result: NO significant difference in VR experience between OA subgroups (p >= 0.05)\n")
  cat("Median OA_High =", median(oa_high_vr), ", Median OA_Low =", median(oa_low_vr), "\n")
}
```

```
## Result: NO significant difference in VR experience between OA subgroups (p >= 0.05)
## Median OA_High = 0 , Median OA_Low = 0
```

Test 4: Weekly Video Game Usage

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

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

```r
# Summary statistics by subgroup
vg_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  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:
```

```r
print(vg_summary)
```

```
## # A tibble: 2 x 9
##   OA_Subgroup     n  Mean    SD Median    Q1    Q3   Min   Max
##   <chr>       <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16 0.125  0.5       0     0     0     0     2
## 2 OA_Low         14 1.07   2.73      0     0     0     0     8
```

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

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

oa_high_vg <- vg_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(Video_Game_Experience_Quantifi
oa_low_vg <- vg_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(Video_Game_Experience_Quantifie

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

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

```r
shapiro_high <- shapiro.test(oa_high_vg)
shapiro_low <- shapiro.test(oa_low_vg)
cat("OA_High: W =", round(shapiro_high$statistic, 4), ", p =", shapiro_high$p.value)
```

```
## OA_High: W = 0.2727 , p = 4.553108e-08
```

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

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

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

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

```
## OA_Low: W = 0.4392 , p = 2.048179e-06
```

```r
if(shapiro_low$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(Video_Game_Experience_Quantified ~ OA_Subgroup, 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 = 1.86 , p = 0.1834841
```

```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$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_high$p.value < 0.05 | shapiro_low$p.value < 0.05

if(use_nonparametric) {
  cat("  Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(oa_high_vg, oa_low_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(oa_high_vg)
  n2 <- length(oa_low_vg)
  r_rank_biserial <- 1 - (2*mw_test$statistic) / (n1 * n2)
  cat("Rank-biserial correlation =", round(r_rank_biserial, 3), "\n")
```

```r
  # Report IQR for each subgroup
  cat("\nIQR by OA subgroup:\n")
  cat("OA_High: [", quantile(oa_high_vg, 0.25), ", ", quantile(oa_high_vg, 0.75), "]\n", sep = "")
  cat("OA_Low: [", quantile(oa_low_vg, 0.25), ", ", quantile(oa_low_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(oa_high_vg, oa_low_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(oa_high_vg)^2 + sd(oa_low_vg)^2) / 2)
  cohens_d <- (mean(oa_high_vg) - mean(oa_low_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(oa_high_vg, oa_low_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(oa_high_vg)^2 + sd(oa_low_vg)^2) / 2)
  cohens_d <- (mean(oa_high_vg) - mean(oa_low_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 = 102
## p-value = 0.4482387
## Rank-biserial correlation = 0.089
##
## IQR by OA subgroup:
## OA_High: [0, 0]
## OA_Low: [0, 0]
```

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

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

```r
if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in video game usage between OA subgroups (p < 0.05)\n")
  if(mean(oa_high_vg) > mean(oa_low_vg)) {
    cat("Direction: OA_High plays MORE video games than OA_Low\n")
  } else {
    cat("Direction: OA_Low plays MORE video games than OA_High\n")
  }
} else {
  cat("Result: NO significant difference in video game usage between OA subgroups (p >= 0.05)\n")
}
```

```
## Result: NO significant difference in video game usage between OA subgroups (p >= 0.05)
```

Test 5: Weekly Exercise Frequency

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

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

```r
# Summary statistics by subgroup
exercise_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  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:
```

```r
print(exercise_summary)
```

```
## # A tibble: 2 x 9
##   OA_Subgroup     n  Mean    SD Median    Q1    Q3   Min   Max
##   <chr>       <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16  5.19  3.89   4.25  2.38  7.5      0    14
## 2 OA_Low         14  4.86  2.27   5     3.12  6.75     0     8
```

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

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

oa_high_ex <- exercise_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(Exercise_Quantified)
oa_low_ex <- exercise_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(Exercise_Quantified)

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

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

```r
shapiro_high <- shapiro.test(oa_high_ex)
shapiro_low <- shapiro.test(oa_low_ex)
cat("OA_High: W =", round(shapiro_high$statistic, 4), ", p =", shapiro_high$p.value)
```

```
## OA_High: W = 0.915 , p = 0.1398632
```

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

```
##  (normal)
```

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

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

```
## OA_Low: W = 0.9522 , p = 0.5958984
```

```r
if(shapiro_low$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 ~ OA_Subgroup, 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 = 2.3328 , p = 0.137892
```

```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$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_high$p.value < 0.05 | shapiro_low$p.value < 0.05

if(use_nonparametric) {
  cat("  Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(oa_high_ex, oa_low_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(oa_high_ex)
  n2 <- length(oa_low_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 subgroup
  cat("\nIQR by OA subgroup:\n")
  cat("OA_High: [", quantile(oa_high_ex, 0.25), ", ", quantile(oa_high_ex, 0.75), "]\n", sep = "")
  cat("OA_Low: [", quantile(oa_low_ex, 0.25), ", ", quantile(oa_low_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(oa_high_ex, oa_low_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(oa_high_ex)^2 + sd(oa_low_ex)^2) / 2)
  cohens_d <- (mean(oa_high_ex) - mean(oa_low_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(oa_high_ex, oa_low_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")
```

```r
  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(oa_high_ex)^2 + sd(oa_low_ex)^2) / 2)
  cohens_d <- (mean(oa_high_ex) - mean(oa_low_ex)) / 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 = 0.2757724
## df = 28
## p-value = 0.784748
## 95% CI for difference: [-2.100543, 2.754114]
## Cohen's d = 0.103
```

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

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

```r
if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in exercise frequency between OA subgroups (p < 0.05)\n")
  if(mean(oa_high_ex) > mean(oa_low_ex)) {
    cat("Direction: OA_High exercises MORE than OA_Low\n")
  } else {
    cat("Direction: OA_Low exercises MORE than OA_High\n")
  }
} else {
  cat("Result: NO significant difference in exercise frequency between OA subgroups (p >= 0.05)\n")
}
```

```
## Result: NO significant difference in exercise frequency between OA subgroups (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 subgroup
sbsod_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  summarise(
    n = sum(!is.na(SBSOD)),
```

```r
    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:
```

```r
print(sbsod_summary)
```

```
## # A tibble: 2 x 9
##   OA_Subgroup     n  Mean    SD Median    Q1    Q3   Min   Max
##   <chr>       <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16  5.28 0.870   5.17  4.98  5.62  3.6   6.87
## 2 OA_Low         14  4.86 0.968   5.16  4.16  5.46  3.27  6.13
```

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

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

oa_high_sbsod <- sbsod_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(SBSOD)
oa_low_sbsod <- sbsod_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(SBSOD)

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

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

```r
shapiro_high <- shapiro.test(oa_high_sbsod)
shapiro_low <- shapiro.test(oa_low_sbsod)
cat("OA_High: W =", round(shapiro_high$statistic, 4), ", p =", shapiro_high$p.value)
```

```
## OA_High: W = 0.9512 , p = 0.50858
```

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

```
##  (normal)
```

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

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

```
## OA_Low: W = 0.9256 , p = 0.2643187
```

```r
if(shapiro_low$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 ~ OA_Subgroup, 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 = 0.5842 , p = 0.4510457
```

```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$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_high$p.value < 0.05 | shapiro_low$p.value < 0.05

if(use_nonparametric) {
  cat("  Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(oa_high_sbsod, oa_low_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(oa_high_sbsod)
  n2 <- length(oa_low_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 subgroup
  cat("\nIQR by OA subgroup:\n")
  cat("OA_High: [", quantile(oa_high_sbsod, 0.25), ", ", quantile(oa_high_sbsod, 0.75), "]\n", sep = "")
  cat("OA_Low: [", quantile(oa_low_sbsod, 0.25), ", ", quantile(oa_low_sbsod, 0.75), "]\n", sep = "")

  p_value <- mw_test$p.value
```

```r
} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
  t_test <- t.test(oa_high_sbsod, oa_low_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(oa_high_sbsod)^2 + sd(oa_low_sbsod)^2) / 2)
  cohens_d <- (mean(oa_high_sbsod) - mean(oa_low_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(oa_high_sbsod, oa_low_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(oa_high_sbsod)^2 + sd(oa_low_sbsod)^2) / 2)
  cohens_d <- (mean(oa_high_sbsod) - mean(oa_low_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 = 1.243768
## df = 28
## p-value = 0.2238977
## 95% CI for difference: [-0.2699228, 1.104387]
## Cohen's d = 0.453
```

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

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

```r
if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in SBSOD scores between OA subgroups (p < 0.05)\n")
  if(mean(oa_high_sbsod) > mean(oa_low_sbsod)) {
    cat("Direction: OA_High has BETTER sense of direction than OA_Low\n")
```

```r
  } else {
    cat("Direction: OA_Low has BETTER sense of direction than OA_High\n")
  }
} else {
  cat("Result: NO significant difference in SBSOD scores between OA subgroups (p >= 0.05)\n")
}
```

```
## Result: NO significant difference in SBSOD scores between OA subgroups (p >= 0.05)
```

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 subgroup
psqi_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  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:
```

```r
print(psqi_summary)
```

```
## # A tibble: 2 x 9
##   OA_Subgroup     n  Mean    SD Median    Q1    Q3   Min   Max
##   <chr>       <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16  4.94  2.93    5.5  2.75  7.25     0     9
## 2 OA_Low         14  4.71  2.97    4    2     7        1    11
```

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

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

oa_high_psqi <- psqi_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(PSQI)
oa_low_psqi <- psqi_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(PSQI)

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

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

```r
shapiro_high <- shapiro.test(oa_high_psqi)
shapiro_low <- shapiro.test(oa_low_psqi)
cat("OA_High: W =", round(shapiro_high$statistic, 4), ", p =", shapiro_high$p.value)
```

```
## OA_High: W = 0.9433 , p = 0.3914036
```

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

```
##  (normal)
```

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

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

```
## OA_Low: W = 0.9078 , p = 0.146485
```

```r
if(shapiro_low$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 ~ OA_Subgroup, 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 = 2e-04 , p = 0.9884679
```

```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$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_high$p.value < 0.05 | shapiro_low$p.value < 0.05

if(use_nonparametric) {
  cat("  Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(oa_high_psqi, oa_low_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(oa_high_psqi)
  n2 <- length(oa_low_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 subgroup
  cat("\nIQR by OA subgroup:\n")
  cat("OA_High: [", quantile(oa_high_psqi, 0.25), ", ", quantile(oa_high_psqi, 0.75), "]\n", sep = "")
  cat("OA_Low: [", quantile(oa_low_psqi, 0.25), ", ", quantile(oa_low_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(oa_high_psqi, oa_low_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(oa_high_psqi)^2 + sd(oa_low_psqi)^2) / 2)
  cohens_d <- (mean(oa_high_psqi) - mean(oa_low_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(oa_high_psqi, oa_low_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(oa_high_psqi)^2 + sd(oa_low_psqi)^2) / 2)
```

```r
  cohens_d <- (mean(oa_high_psqi) - mean(oa_low_psqi)) / 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 = 0.2067056
## df = 28
## p-value = 0.8377356
## 95% CI for difference: [-1.98879, 2.435219]
## Cohen's d = 0.076
```

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

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

```r
if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in sleep quality between OA subgroups (p < 0.05)\n")
  if(mean(oa_high_psqi) > mean(oa_low_psqi)) {
    cat("Direction: OA_High has WORSE sleep quality than OA_Low\n")
  } else {
    cat("Direction: OA_Low has WORSE sleep quality than OA_High\n")
  }
} else {
  cat("Result: NO significant difference in sleep quality between OA subgroups (p >= 0.05)\n")
}
```

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

Test 8: Trails Making Test 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 subgroup
trails_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  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
##   OA_Subgroup     n  Mean    SD Median    Q1    Q3   Min   Max
##   <chr>       <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16  33.5  6.08   35.3  29.3  36.3  22.4  45
## 2 OA_Low         14  35.0  7.78   33.4  30.0  36.4  24.3  48.6
```

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

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

oa_high_trails <- trails_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(Trails_A_CT)
oa_low_trails <- trails_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(Trails_A_CT)

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

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

```
shapiro_high <- shapiro.test(oa_high_trails)
shapiro_low <- shapiro.test(oa_low_trails)
cat("OA_High: W =", round(shapiro_high$statistic, 4), ", p =", shapiro_high$p.value)
```

```
## OA_High: W = 0.9242 , p = 0.1972416
```

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

```
##  (normal)
```

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

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

```
## OA_Low: W = 0.8746 , p = 0.04876468
```

```r
if(shapiro_low$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_A_CT ~ OA_Subgroup, 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 = 0.5722 , p = 0.4556928
```

```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$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_high$p.value < 0.05 | shapiro_low$p.value < 0.05

if(use_nonparametric) {
  cat("  Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(oa_high_trails, oa_low_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(oa_high_trails)
  n2 <- length(oa_low_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 subgroup
  cat("\nIQR by OA subgroup:\n")
  cat("OA_High: [", quantile(oa_high_trails, 0.25), ", ", quantile(oa_high_trails, 0.75), "]\n", sep =
  cat("OA_Low: [", quantile(oa_low_trails, 0.25), ", ", quantile(oa_low_trails, 0.75), "]\n", sep = "")

  p_value <- mw_test$p.value
```

```r
} else if(use_welch) {
  cat("Variances are unequal. Using Welch's t-test.\n\n")
  t_test <- t.test(oa_high_trails, oa_low_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(oa_high_trails)^2 + sd(oa_low_trails)^2) / 2)
  cohens_d <- (mean(oa_high_trails) - mean(oa_low_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(oa_high_trails, oa_low_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(oa_high_trails)^2 + sd(oa_low_trails)^2) / 2)
  cohens_d <- (mean(oa_high_trails) - mean(oa_low_trails)) / 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 = 114
## p-value = 0.9502739
## Rank-biserial correlation = -0.018
##
## IQR by OA subgroup:
## OA_High: [29.25, 36.275]
## OA_Low: [29.975, 36.4]
```

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

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

```r
if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in Trails A performance between OA subgroups (p < 0.05)\n")
  if(mean(oa_high_trails) < mean(oa_low_trails)) {
    cat("Direction: OA_High is FASTER (better) than OA_Low\n")
  } else {
    cat("Direction: OA_Low is FASTER (better) than OA_High\n")
  }
} else {
  cat("Result: NO significant difference in Trails A performance between OA subgroups (p >= 0.05)\n")
}
```

```
## Result: NO significant difference in Trails A performance between OA subgroups (p >= 0.05)
```

Test 9: Trails Making Test B Performance

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

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

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

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

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

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

```r
# Summary statistics by subgroup
trails_b_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  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:
```

```r
print(trails_b_summary)
```

```
## # A tibble: 2 x 9
##   OA_Subgroup     n  Mean    SD Median    Q1    Q3   Min   Max
##   <chr>       <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16  56.2  22.9   49.3  46.2  57.4  33.5  132.
## 2 OA_Low         14  63.3  35.0   48.4  43.6  70.6  26.4  160.
```

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

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

oa_high_trails_b <- trails_b_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(Trails_B_CT)
oa_low_trails_b <- trails_b_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(Trails_B_CT)

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

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

```r
shapiro_high <- shapiro.test(oa_high_trails_b)
shapiro_low <- shapiro.test(oa_low_trails_b)
cat("OA_High: W =", round(shapiro_high$statistic, 4), ", p =", shapiro_high$p.value)
```

```
## OA_High: W = 0.7044 , p = 0.0001885612
```

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

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

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

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

```
## OA_Low: W = 0.815 , p = 0.007696439
```

```r
if(shapiro_low$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 ~ OA_Subgroup, 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 = 1.1606 , p = 0.2905458
```

```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$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_high$p.value < 0.05 | shapiro_low$p.value < 0.05

if(use_nonparametric) {
  cat("  Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(oa_high_trails_b, oa_low_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(oa_high_trails_b)
  n2 <- length(oa_low_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 subgroup
  cat("\nIQR by OA subgroup:\n")
  cat("OA_High: [", quantile(oa_high_trails_b, 0.25), ", ", quantile(oa_high_trails_b, 0.75), "]\n", sep
  cat("OA_Low: [", quantile(oa_low_trails_b, 0.25), ", ", quantile(oa_low_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(oa_high_trails_b, oa_low_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(oa_high_trails_b)^2 + sd(oa_low_trails_b)^2) / 2)
  cohens_d <- (mean(oa_high_trails_b) - mean(oa_low_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")
```

```r
  t_test <- t.test(oa_high_trails_b, oa_low_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(oa_high_trails_b)^2 + sd(oa_low_trails_b)^2) / 2)
  cohens_d <- (mean(oa_high_trails_b) - mean(oa_low_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 = 109.5
## p-value = 0.933732
## Rank-biserial correlation = 0.022
##
## IQR by OA subgroup:
## OA_High: [46.2, 57.425]
## OA_Low: [43.6, 70.575]
```

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

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

```r
if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in Trails B performance between OA subgroups (p < 0.05)\n")
  if(mean(oa_high_trails_b) < mean(oa_low_trails_b)) {
    cat("Direction: OA_High is FASTER (better) than OA_Low\n")
  } else {
    cat("Direction: OA_Low is FASTER (better) than OA_High\n")
  }
} else {
  cat("Result: NO significant difference in Trails B performance between OA subgroups (p >= 0.05)\n")
}
```

```
## Result: NO significant difference in Trails B performance between OA subgroups (p >= 0.05)
```

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

```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 subgroup
trails_ba_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  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:
```

```r
print(trails_ba_summary)
```

```
## # A tibble: 2 x 9
##   OA_Subgroup     n  Mean    SD Median    Q1    Q3   Min   Max
##   <chr>       <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16  22.7  24.0   17.6  11.4  22.5 -3     104.
## 2 OA_Low         14  28.3  29.4   16.6  12.0  36.7 -2.20  111.
```

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

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

oa_high_trails_ba <- trails_ba_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(Trails_BA_Diff)
oa_low_trails_ba <- trails_ba_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(Trails_BA_Diff)

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

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

```r
shapiro_high <- shapiro.test(oa_high_trails_ba)
shapiro_low <- shapiro.test(oa_low_trails_ba)
cat("OA_High: W =", round(shapiro_high$statistic, 4), ", p =", shapiro_high$p.value)
```

```
## OA_High: W = 0.6778 , p = 9.702657e-05
```

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

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

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

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

```
## OA_Low: W = 0.7888 , p = 0.003641066
```

```r
if(shapiro_low$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 ~ OA_Subgroup, 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 = 0.4567 , p = 0.5047014
```

```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$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_high$p.value < 0.05 | shapiro_low$p.value < 0.05

if(use_nonparametric) {
  cat("  Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(oa_high_trails_ba, oa_low_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(oa_high_trails_ba)
  n2 <- length(oa_low_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 subgroup
  cat("\nIQR by OA subgroup:\n")
  cat("OA_High: [", quantile(oa_high_trails_ba, 0.25), ", ", quantile(oa_high_trails_ba, 0.75), "]\n", s
  cat("OA_Low: [", quantile(oa_low_trails_ba, 0.25), ", ", quantile(oa_low_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(oa_high_trails_ba, oa_low_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(oa_high_trails_ba)^2 + sd(oa_low_trails_ba)^2) / 2)
  cohens_d <- (mean(oa_high_trails_ba) - mean(oa_low_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(oa_high_trails_ba, oa_low_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(oa_high_trails_ba)^2 + sd(oa_low_trails_ba)^2) / 2)
  cohens_d <- (mean(oa_high_trails_ba) - mean(oa_low_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 = 102
## p-value = 0.6929018
## Rank-biserial correlation = 0.089
##
## IQR by OA subgroup:
## OA_High: [11.4, 22.525]
## OA_Low: [12.025, 36.725]
```

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

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

```r
if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in cognitive switching cost between OA subgroups (p < 0.05)\n")
  if(mean(oa_high_trails_ba) < mean(oa_low_trails_ba)) {
    cat("Direction: OA_High has LOWER switching cost (better executive function) than OA_Low\n")
  } else {
    cat("Direction: OA_Low has LOWER switching cost (better executive function) than OA_High\n")
  }
} else {
  cat("Result: NO significant difference in cognitive switching cost between OA subgroups (p >= 0.05)\n
}
```

```
## Result: NO significant difference in cognitive switching cost between OA subgroups (p >= 0.05)
```

Test 11: 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 subgroup
corsi_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  summarise(
    n = sum(!is.na(Corsi_Score_Total)),
    Mean = mean(Corsi_Score_Total, na.rm = TRUE),
    SD = sd(Corsi_Score_Total, na.rm = TRUE),
```

```r
    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:
```

```r
print(corsi_summary)
```

```
## # A tibble: 2 x 9
##   OA_Subgroup     n  Mean    SD Median    Q1    Q3   Min   Max
##   <chr>       <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16  71.6  9.24     75  64.8  77.2    54    87
## 2 OA_Low         14  70   10.7      69  60.8  75.8    54    89
```

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

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

oa_high_corsi <- corsi_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(Corsi_Score_Total)
oa_low_corsi <- corsi_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(Corsi_Score_Total)

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

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

```r
shapiro_high <- shapiro.test(oa_high_corsi)
shapiro_low <- shapiro.test(oa_low_corsi)
cat("OA_High: W =", round(shapiro_high$statistic, 4), ", p =", shapiro_high$p.value)
```

```
## OA_High: W = 0.9528 , p = 0.5355655
```

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

```
##  (normal)
```

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

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

```
## OA_Low: W = 0.9611 , p = 0.7413846
```

```r
if(shapiro_low$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 ~ OA_Subgroup, 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 = 0.5876 , p = 0.4497549
```

```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$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_high$p.value < 0.05 | shapiro_low$p.value < 0.05

if(use_nonparametric) {
  cat("  Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(oa_high_corsi, oa_low_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(oa_high_corsi)
  n2 <- length(oa_low_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 subgroup
  cat("\nIQR by OA subgroup:\n")
  cat("OA_High: [", quantile(oa_high_corsi, 0.25), ", ", quantile(oa_high_corsi, 0.75), "]\n", sep = ""
  cat("OA_Low: [", quantile(oa_low_corsi, 0.25), ", ", quantile(oa_low_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(oa_high_corsi, oa_low_corsi, var.equal = FALSE)
  cat("Welch's t-test Results:\n")
```

```r
  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(oa_high_corsi)^2 + sd(oa_low_corsi)^2) / 2)
  cohens_d <- (mean(oa_high_corsi) - mean(oa_low_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(oa_high_corsi, oa_low_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(oa_high_corsi)^2 + sd(oa_low_corsi)^2) / 2)
  cohens_d <- (mean(oa_high_corsi) - mean(oa_low_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 = 0.4454887
## df = 28
## p-value = 0.6593921
## 95% CI for difference: [-5.846932, 9.096932]
## Cohen's d = 0.162
```

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

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

```r
if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in Corsi Block performance between OA subgroups (p < 0.05)\n")
  if(mean(oa_high_corsi) > mean(oa_low_corsi)) {
    cat("Direction: OA_High has BETTER visuospatial working memory than OA_Low\n")
  } else {
    cat("Direction: OA_Low has BETTER visuospatial working memory than OA_High\n")
  }
} else {
```

```r
  cat("Result: NO significant difference in Corsi Block performance between OA subgroups (p >= 0.05)\n")
}
```

```
## Result: NO significant difference in Corsi Block performance between OA subgroups (p >= 0.05)
```

Test 12: Pre-VR SSQ Scores

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

```
## SSQ Pre: Simulator Sickness Questionnaire - Baseline
```

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

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

```r
# Summary statistics by subgroup
ssq_pre_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  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:
```

```r
print(ssq_pre_summary)
```

```
## # A tibble: 2 x 9
##   OA_Subgroup     n  Mean    SD Median    Q1    Q3   Min   Max
##   <chr>       <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16  2.5   4.29      0     0  2.5      0    13
## 2 OA_Low         14  3.07  2.81      3     1  4.75     0     9
```

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

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

oa_high_ssq_pre <- ssq_pre_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(SSQ_Pre)
```

```r
oa_low_ssq_pre <- ssq_pre_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(SSQ_Pre)

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

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

```r
shapiro_high <- shapiro.test(oa_high_ssq_pre)
shapiro_low <- shapiro.test(oa_low_ssq_pre)
cat("OA_High: W =", round(shapiro_high$statistic, 4), ", p =", shapiro_high$p.value)
```

```
## OA_High: W = 0.6538 , p = 5.464436e-05
```

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

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

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

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

```
## OA_Low: W = 0.9024 , p = 0.1222297
```

```r
if(shapiro_low$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(SSQ_Pre ~ OA_Subgroup, data = ssq_pre_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 = 0.0142 , p = 0.9061122
```

```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$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_high$p.value < 0.05 | shapiro_low$p.value < 0.05

if(use_nonparametric) {
  cat("  Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(oa_high_ssq_pre, oa_low_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(oa_high_ssq_pre)
  n2 <- length(oa_low_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 subgroup
  cat("\nIQR by OA subgroup:\n")
  cat("OA_High: [", quantile(oa_high_ssq_pre, 0.25), ", ", quantile(oa_high_ssq_pre, 0.75), "]\n", sep =
  cat("OA_Low: [", quantile(oa_low_ssq_pre, 0.25), ", ", quantile(oa_low_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(oa_high_ssq_pre, oa_low_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(oa_high_ssq_pre)^2 + sd(oa_low_ssq_pre)^2) / 2)
  cohens_d <- (mean(oa_high_ssq_pre) - mean(oa_low_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(oa_high_ssq_pre, oa_low_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(oa_high_ssq_pre)^2 + sd(oa_low_ssq_pre)^2) / 2)
  cohens_d <- (mean(oa_high_ssq_pre) - mean(oa_low_ssq_pre)) / pooled_sd
  cat("Cohen's d =", round(cohens_d, 3), "\n")
```

```r
  p_value <- t_test$p.value
}
```

```
##   Data violates normality assumption. Using Mann-Whitney U test.
##
## Mann-Whitney U Test Results:
## W statistic = 78.5
## p-value = 0.1547101
## Rank-biserial correlation = 0.299
##
## IQR by OA subgroup:
## OA_High: [0, 2.5]
## OA_Low: [1, 4.75]
```

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

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

```r
if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in baseline SSQ between OA subgroups (p < 0.05)\n")
  if(mean(oa_high_ssq_pre) > mean(oa_low_ssq_pre)) {
    cat("Direction: OA_High has MORE baseline symptoms than OA_Low\n")
  } else {
    cat("Direction: OA_Low has MORE baseline symptoms than OA_High\n")
  }
} else {
  cat("Result: NO significant difference in baseline SSQ between OA subgroups (p >= 0.05)\n")
  cat("(This is expected - both subgroups should be similar before VR exposure)\n")
}
```

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

Test 13: Post-VR SSQ Scores

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

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

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

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

```r
# Summary statistics by subgroup
ssq_post_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  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:
```

```r
print(ssq_post_summary)
```

```
## # A tibble: 2 x 9
##   OA_Subgroup     n  Mean    SD Median    Q1    Q3   Min   Max
##   <chr>       <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16  3.19  3.82    1.5  0     5        0    12
## 2 OA_Low         14  4.29  4.97    2.5  0.25  6.75     0    17
```

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

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

oa_high_ssq_post <- ssq_post_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(SSQ_Post)
oa_low_ssq_post <- ssq_post_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(SSQ_Post)

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

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

```r
shapiro_high <- shapiro.test(oa_high_ssq_post)
shapiro_low <- shapiro.test(oa_low_ssq_post)
cat("OA_High: W =", round(shapiro_high$statistic, 4), ", p =", shapiro_high$p.value)
```

```
## OA_High: W = 0.8133 , p = 0.004109605
```

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

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

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

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

```
## OA_Low: W = 0.8391 , p = 0.01586586
```

```r
if(shapiro_low$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(SSQ_Post ~ OA_Subgroup, data = ssq_post_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 = 0.5488 , p = 0.4649744
```

```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$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_high$p.value < 0.05 | shapiro_low$p.value < 0.05

if(use_nonparametric) {
  cat("  Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(oa_high_ssq_post, oa_low_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(oa_high_ssq_post)
  n2 <- length(oa_low_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 subgroup
  cat("\nIQR by OA subgroup:\n")
  cat("OA_High: [", quantile(oa_high_ssq_post, 0.25), ", ", quantile(oa_high_ssq_post, 0.75), "]\n", se
  cat("OA_Low: [", quantile(oa_low_ssq_post, 0.25), ", ", quantile(oa_low_ssq_post, 0.75), "]\n", sep =
```

```r
  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(oa_high_ssq_post, oa_low_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(oa_high_ssq_post)^2 + sd(oa_low_ssq_post)^2) / 2)
  cohens_d <- (mean(oa_high_ssq_post) - mean(oa_low_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(oa_high_ssq_post, oa_low_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(oa_high_ssq_post)^2 + sd(oa_low_ssq_post)^2) / 2)
  cohens_d <- (mean(oa_high_ssq_post) - mean(oa_low_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 = 100.5
## p-value = 0.6418583
## Rank-biserial correlation = 0.103
##
## IQR by OA subgroup:
## OA_High: [0, 5]
## OA_Low: [0.25, 6.75]
```

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

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

```r
if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in post-VR SSQ between OA subgroups (p < 0.05)\n")
  if(mean(oa_high_ssq_post) > mean(oa_low_ssq_post)) {
    cat("Direction: OA_High has MORE post-VR symptoms than OA_Low\n")
  } else {
    cat("Direction: OA_Low has MORE post-VR symptoms than OA_High\n")
  }
} else {
  cat("Result: NO significant difference in post-VR SSQ between OA subgroups (p >= 0.05)\n")
}
```

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

Test 14: SSQ Change Scores

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

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

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

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

```r
# Summary statistics by subgroup
ssq_diff_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  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:
```

```r
print(ssq_diff_summary)
```

```
## # A tibble: 2 x 9
##   OA_Subgroup     n  Mean    SD Median    Q1    Q3   Min   Max
##   <chr>       <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16 0.688  4.42    0   -0.25   3      -6    12
## 2 OA_Low         14 1.21   4.42    0.5 -1.75   2.5    -4    10
```

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

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

oa_high_ssq_diff <- ssq_diff_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(SSQ_Diff)
oa_low_ssq_diff <- ssq_diff_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(SSQ_Diff)

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

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

```r
shapiro_high <- shapiro.test(oa_high_ssq_diff)
shapiro_low <- shapiro.test(oa_low_ssq_diff)
cat("OA_High: W =", round(shapiro_high$statistic, 4), ", p =", shapiro_high$p.value)
```

```
## OA_High: W = 0.8941 , p = 0.06468658
```

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

```
##  (normal)
```

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

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

```
## OA_Low: W = 0.8992 , p = 0.1097522
```

```r
if(shapiro_low$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(SSQ_Diff ~ OA_Subgroup, data = ssq_diff_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 = 0.1159 , p = 0.7361071
```

```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$`Pr(>F)`[1] < 0.05
use_nonparametric <- shapiro_high$p.value < 0.05 | shapiro_low$p.value < 0.05

if(use_nonparametric) {
  cat("  Data violates normality assumption. Using Mann-Whitney U test.\n\n")
  mw_test <- wilcox.test(oa_high_ssq_diff, oa_low_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(oa_high_ssq_diff)
  n2 <- length(oa_low_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 subgroup
  cat("\nIQR by OA subgroup:\n")
  cat("OA_High: [", quantile(oa_high_ssq_diff, 0.25), ", ", quantile(oa_high_ssq_diff, 0.75), "]\n", sep
  cat("OA_Low: [", quantile(oa_low_ssq_diff, 0.25), ", ", quantile(oa_low_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(oa_high_ssq_diff, oa_low_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(oa_high_ssq_diff)^2 + sd(oa_low_ssq_diff)^2) / 2)
  cohens_d <- (mean(oa_high_ssq_diff) - mean(oa_low_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(oa_high_ssq_diff, oa_low_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")
```

```r
  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(oa_high_ssq_diff)^2 + sd(oa_low_ssq_diff)^2) / 2)
  cohens_d <- (mean(oa_high_ssq_diff) - mean(oa_low_ssq_diff)) / 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 = -0.3254374
## df = 28
## p-value = 0.7472695
## 95% CI for difference: [-3.842543, 2.788972]
## Cohen's d = -0.119
```

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

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

```r
if(p_value < 0.05) {
  cat("Result: SIGNIFICANT difference in SSQ change between OA subgroups (p < 0.05)\n")
  if(mean(oa_high_ssq_diff) > mean(oa_low_ssq_diff)) {
    cat("Direction: OA_High had GREATER increase in symptoms than OA_Low\n")
  } else {
    cat("Direction: OA_Low had GREATER increase in symptoms than OA_High\n")
  }
} else {
  cat("Result: NO significant difference in SSQ change between OA subgroups (p >= 0.05)\n")
}
```

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

Test 15: 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 subgroup (emphasizing median for ordinal data)
sss_pre_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  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:
```

```r
print(sss_pre_summary)
```

```
## # A tibble: 2 x 9
##   OA_Subgroup     n Median    Q1    Q3  Mean    SD   Min   Max
##   <chr>       <int>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16      1     1  1.25  1.31 0.602     1     3
## 2 OA_Low         14      1     1  1.75  1.36 0.633     1     3
```

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

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

```
## Frequency Distribution:
```

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

```
##
##     OA_High OA_Low
##   1      12     10
##   2       3      3
##   3       1      1
```

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

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

oa_high_sss_pre <- sss_pre_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(SSS_Pre)
oa_low_sss_pre <- sss_pre_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(SSS_Pre)

# Mann-Whitney U test
mw_test <- wilcox.test(oa_high_sss_pre, oa_low_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 = 108
```

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

```
## p-value = 0.8507591
```

```r
# Effect size (rank-biserial correlation)
n1 <- length(oa_high_sss_pre)
n2 <- length(oa_low_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 = 0.036
```

```r
# Report IQR for each subgroup
cat("\nIQR by OA subgroup:\n")
```

```
##
## IQR by OA subgroup:
```

```r
cat("OA_High: [", quantile(oa_high_sss_pre, 0.25), ", ", quantile(oa_high_sss_pre, 0.75), "]\n", sep =
```

```
## OA_High: [1, 1.25]
```

```r
cat("OA_Low: [", quantile(oa_low_sss_pre, 0.25), ", ", quantile(oa_low_sss_pre, 0.75), "]\n", sep = "")
```

```
## OA_Low: [1, 1.75]
```

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

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

```r
if(mw_test$p.value < 0.05) {
  cat("Result: SIGNIFICANT difference in baseline sleepiness between OA subgroups (p < 0.05)\n")
  if(median(oa_high_sss_pre) > median(oa_low_sss_pre)) {
    cat("Direction: OA_High was MORE sleepy at baseline (Median High =", median(oa_high_sss_pre),
        ", Low =", median(oa_low_sss_pre), ")\n")
  } else if(median(oa_high_sss_pre) < median(oa_low_sss_pre)) {
    cat("Direction: OA_Low was MORE sleepy at baseline (Median High =", median(oa_high_sss_pre),
        ", Low =", median(oa_low_sss_pre), ")\n")
  } else {
    cat("Direction: Medians are equal, but distributions differ\n")
  }
} else {
  cat("Result: NO significant difference in baseline sleepiness between OA subgroups (p >= 0.05)\n")
  cat("Median OA_High =", median(oa_high_sss_pre), ", Median OA_Low =", median(oa_low_sss_pre), "\n")
  cat("(This is expected - both subgroups should be similar before the study)\n")
}
```

```
## Result: NO significant difference in baseline sleepiness between OA subgroups (p >= 0.05)
## Median OA_High = 1 , Median OA_Low = 1
## (This is expected - both subgroups should be similar before the study)
```

Test 16: 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 subgroup (emphasizing median for ordinal data)
sss_post_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  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
##   OA_Subgroup     n Median    Q1    Q3  Mean    SD   Min   Max
##   <chr>       <int>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16      1     1  2      1.5 0.894     1     4
## 2 OA_Low         14      1     1  1.75   1.36 0.633     1     3
```

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

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

```
## Frequency Distribution:
```

```r
freq_table <- table(merged_data$SSS_Post, merged_data$OA_Subgroup)
print(freq_table)
```

```
##
##     OA_High OA_Low
##   1      11     10
##   2       3      3
##   3       1      1
##   4       1      0
```

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

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

oa_high_sss_post <- sss_post_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(SSS_Post)
oa_low_sss_post <- sss_post_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(SSS_Post)

# Mann-Whitney U test
mw_test <- wilcox.test(oa_high_sss_post, oa_low_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 = 117
```

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

```
## p-value = 0.8164309
```

```r
# Effect size (rank-biserial correlation)
n1 <- length(oa_high_sss_post)
n2 <- length(oa_low_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 = -0.045
```

```r
# Report IQR for each subgroup
cat("\nIQR by OA subgroup:\n")
```

```
##
## IQR by OA subgroup:
```

```r
cat("OA_High: [", quantile(oa_high_sss_post, 0.25), ", ", quantile(oa_high_sss_post, 0.75), "]\n", sep =
```

```
## OA_High: [1, 2]
```

```r
cat("OA_Low: [", quantile(oa_low_sss_post, 0.25), ", ", quantile(oa_low_sss_post, 0.75), "]\n", sep = "
```

```
## OA_Low: [1, 1.75]
```

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

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

```r
if(mw_test$p.value < 0.05) {
  cat("Result: SIGNIFICANT difference in post-study sleepiness between OA subgroups (p < 0.05)\n")
  if(median(oa_high_sss_post) > median(oa_low_sss_post)) {
    cat("Direction: OA_High was MORE sleepy after study (Median High =", median(oa_high_sss_post),
        ", Low =", median(oa_low_sss_post), ")\n")
  } else if(median(oa_high_sss_post) < median(oa_low_sss_post)) {
    cat("Direction: OA_Low was MORE sleepy after study (Median High =", median(oa_high_sss_post),
        ", Low =", median(oa_low_sss_post), ")\n")
  } else {
    cat("Direction: Medians are equal, but distributions differ\n")
  }
} else {
  cat("Result: NO significant difference in post-study sleepiness between OA subgroups (p >= 0.05)\n")
  cat("Median OA_High =", median(oa_high_sss_post), ", Median OA_Low =", median(oa_low_sss_post), "\n")
}
```

```
## Result: NO significant difference in post-study sleepiness between OA subgroups (p >= 0.05)
## Median OA_High = 1 , Median OA_Low = 1
```

Test 17: 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 subgroup (emphasizing median for ordinal data)
sss_diff_summary <- merged_data %>%
  group_by(OA_Subgroup) %>%
  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
##   OA_Subgroup     n Median    Q1    Q3  Mean    SD   Min   Max
##   <chr>       <int>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 OA_High        16      0     0  0     0.188 0.403     0     1
## 2 OA_Low         14      0     0  0.75 0      0.877    -2     1
```

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

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

```
## Frequency Distribution:
```

```r
freq_table <- table(merged_data$SSS_Diff, merged_data$OA_Subgroup)
print(freq_table)
```

```
##
##      OA_High OA_Low
##   -2       0      1
##   -1       0      2
##   0       13      7
##   1        3      4
```

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

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

oa_high_sss_diff <- sss_diff_data_clean %>% filter(OA_Subgroup == "OA_High") %>% pull(SSS_Diff)
oa_low_sss_diff <- sss_diff_data_clean %>% filter(OA_Subgroup == "OA_Low") %>% pull(SSS_Diff)

# Mann-Whitney U test
mw_test <- wilcox.test(oa_high_sss_diff, oa_low_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 = 120.5
```

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

```
## p-value = 0.6891971
```

```r
# Effect size (rank-biserial correlation)
n1 <- length(oa_high_sss_diff)
n2 <- length(oa_low_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 = -0.076
```

```r
# Report IQR for each subgroup
cat("\nIQR by OA subgroup:\n")
```

```
##
## IQR by OA subgroup:
```

```r
cat("OA_High: [", quantile(oa_high_sss_diff, 0.25), ", ", quantile(oa_high_sss_diff, 0.75), "]\n", sep =
```

```
## OA_High: [0, 0]
```

```r
cat("OA_Low: [", quantile(oa_low_sss_diff, 0.25), ", ", quantile(oa_low_sss_diff, 0.75), "]\n", sep = "
```

```
## OA_Low: [0, 0.75]
```

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

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

```r
if(mw_test$p.value < 0.05) {
  cat("Result: SIGNIFICANT difference in sleepiness change between OA subgroups (p < 0.05)\n")
  if(median(oa_high_sss_diff) > median(oa_low_sss_diff)) {
    cat("Direction: OA_High had GREATER increase in sleepiness (Median change: High =", median(oa_high_s
        ", Low =", median(oa_low_sss_diff), ")\n")
  } else if(median(oa_high_sss_diff) < median(oa_low_sss_diff)) {
    cat("Direction: OA_Low had GREATER increase in sleepiness (Median change: High =", median(oa_high_ss
        ", Low =", median(oa_low_sss_diff), ")\n")
  } else {
    cat("Direction: Median changes are equal, but distributions differ\n")
  }
} else {
  cat("Result: NO significant difference in sleepiness change between OA subgroups (p >= 0.05)\n")
  cat("Median change: OA_High =", median(oa_high_sss_diff), ", OA_Low =", median(oa_low_sss_diff), "\n")
}
```

```
## Result: NO significant difference in sleepiness change between OA subgroups (p >= 0.05)
## Median change: OA_High = 0 , OA_Low = 0
```

Session Information

```r
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 v
##
## 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] tidyr_1.3.0      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
```