NavCity Correlation Analyses with FDR Correction

Your Name

2025-10-10

Contents

Load Required Libraries

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

Data Import and Preparation

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

```r
ya_nav <- read_csv("/Volumes/YB_Drive/NavAging_Paper/data/ya_averaged_results.csv")
oa_nav <- read_csv("/Volumes/YB_Drive/NavAging_Paper/data/oa_averaged_results.csv")

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

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

# Merge all datasets
merged_temp <- demographic_data %>%
  full_join(non_nav, by = "Participant", suffix = c("_demo", "_non_nav")) %>%
  mutate(Group = coalesce(Group_non_nav,
                          if_else(Group_demo == 1, "YA", "OA"))) %>%
  select(-Group_demo, -Group_non_nav)

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

cat("Total participants:", nrow(merged_data), "\n")
```

```
## Total participants: 60
```

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

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

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

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

Create Numeric Variables for Correlation

```r
# Create numeric versions of variables for correlation
merged_data <- merged_data %>%
  mutate(
    # VR Experience already numeric (0, 1, 2)
    VR_Experience_numeric = VR_Experience_Quantified,
    # Calculate Trails B-A difference (switching cost)
    Trails_BA_Diff = Trails_B_CT - Trails_A_CT
  )

# Show coding scheme
cat("Variable Information:\n")
```

```
## Variable Information:
```

```r
cat("- VR Experience: 0 (never), 1 (1-3 times), 2 (>3 times) [Non-parametric]\n")
```

```
## - VR Experience: 0 (never), 1 (1-3 times), 2 (>3 times) [Non-parametric]
```

```r
cat("- Video Game Usage: Continuous (hours per week) [Non-parametric]\n")
```

```
## - Video Game Usage: Continuous (hours per week) [Non-parametric]
```

```r
cat("- SSS Pre: Stanford Sleepiness Scale baseline (1-7) [Non-parametric]\n")
```

```
## - SSS Pre: Stanford Sleepiness Scale baseline (1-7) [Non-parametric]
```

```r
cat("- SBSOD: Santa Barbara Sense of Direction Scale (higher = better) [Parametric]\n")
```

```
## - SBSOD: Santa Barbara Sense of Direction Scale (higher = better) [Parametric]
```

```r
cat("- Trails A: Completion time (lower = better) [Parametric]\n")
```

```
## - Trails A: Completion time (lower = better) [Parametric]
```

```r
cat("- Trails B: Completion time (lower = better) [Non-parametric]\n")
```

```
## - Trails B: Completion time (lower = better) [Non-parametric]
```

```r
cat("- Trails B-A: Cognitive switching cost (higher = worse) [Non-parametric]\n")
```

```
## - Trails B-A: Cognitive switching cost (higher = worse) [Non-parametric]
```

```r
cat("- Corsi Blocks: Total score (higher = better) [Parametric]\n\n")
```

```
## - Corsi Blocks: Total score (higher = better) [Parametric]
```

```r
# Define predictors and outcomes
predictors <- c(
  "VR_Experience_numeric",
  "Video_Game_Experience_Quantified",
  "SSS_Pre",
  "SBSOD",
  "Trails_A_CT",
  "Trails_B_CT",
  "Trails_BA_Diff",
  "Corsi_Score_Total"
)

predictor_labels <- c(
  "VR Experience",
  "Video Game Usage",
  "SSS Pre",
  "SBSOD",
  "Trails A",
  "Trails B",
  "Trails B-A (Switch Cost)",
  "Corsi Total"
)

# Define which test to use for each predictor
# Parametric: SBSOD, Trails A, Corsi (use Pearson)
# Non-parametric: VR Experience, Video Games, SSS, Trails B, Trails B-A (use Spearman)
predictor_methods <- c(
  "spearman",  # VR Experience
  "spearman",  # Video Game Usage
  "spearman",  # SSS Pre
  "pearson",   # SBSOD
  "pearson",   # Trails A
  "spearman",  # Trails B
  "spearman",  # Trails B-A
  "pearson"    # Corsi Total
)

outcomes <- c(
  "Mean_Speed",
  "Mean_Distance",
  "Mean_Navigation_Time"
)

outcome_labels <- c(
  "Mean Speed",
  "Mean Distance",
  "Mean Nav Time"
)

cat("8 Predictors × 3 Outcomes = 24 correlations per group\n")
```

```
## 8 Predictors × 3 Outcomes = 24 correlations per group
```

```r
cat("Total correlations: 48 (24 YA + 24 OA)\n")
```

```
## Total correlations: 48 (24 YA + 24 OA)
```

```r
cat("Parametric tests (Pearson): SBSOD, Trails A, Corsi Total\n")
```

```
## Parametric tests (Pearson): SBSOD, Trails A, Corsi Total
```

```r
cat("Non-parametric tests (Spearman): VR Experience, Video Games, SSS, Trails B, Trails B-A\n")
```

```
## Non-parametric tests (Spearman): VR Experience, Video Games, SSS, Trails B, Trails B-A
```

```r
cat("FDR correction applied within each age group (24 tests each)\n")
```

```
## FDR correction applied within each age group (24 tests each)
```

YA Group Correlations with FDR Correction

```r
cat("=== YOUNGER ADULTS (YA) CORRELATIONS ===\n\n")
```

```
## === YOUNGER ADULTS (YA) CORRELATIONS ===
```

```r
# Filter YA data
ya_data <- merged_data %>% filter(Group == "YA")

cat("YA sample size:", nrow(ya_data), "\n\n")
```

```
## YA sample size: 30
```

```r
# Function to compute correlations with specified methods
compute_correlations <- function(data, predictors, outcomes, methods) {
  results <- tibble(
    predictor = rep(predictors, each = length(outcomes)),
    outcome = rep(outcomes, times = length(predictors)),
    test_method = rep(methods, each = length(outcomes))
  ) %>%
    mutate(
      correlation = pmap(list(predictor, outcome, test_method), function(pred_var, out_var, test_method
        # Filter to complete cases for this pair
        valid_indices <- !is.na(data[[pred_var]]) & !is.na(data[[out_var]])

        if (sum(valid_indices) < 3) {
          return(tibble(r = NA, p_value = NA, n = 0, method = test_method))
        }

        test <- cor.test(data[[pred_var]][valid_indices],
                         data[[out_var]][valid_indices],
                         method = test_method)
        tibble(
          r = test$estimate,
          p_value = test$p.value,
          n = sum(valid_indices),
          method = test_method
        )
      })
    ) %>%
    unnest(correlation) %>%
    select(-test_method)

  return(results)
}

# Compute all YA correlations
ya_correlations <- compute_correlations(ya_data, predictors, outcomes, predictor_methods)

# Add FDR correction within YA group
ya_correlations <- ya_correlations %>%
  mutate(
    p_fdr = p.adjust(p_value, method = "fdr")
  )

# Add labels for better readability
ya_correlations <- ya_correlations %>%
  mutate(
    predictor_label = predictor_labels[match(predictor, predictors)],
    outcome_label = outcome_labels[match(outcome, outcomes)],
    test_type = ifelse(method == "pearson", "Pearson", "Spearman")
  )

# Display correlation table with both raw and FDR-corrected p-values
cat("YA Correlation Results (with FDR correction):\n\n")
```

```
## YA Correlation Results (with FDR correction):
```

```
ya_correlations %>%
  select(Predictor = predictor_label,
         Outcome = outcome_label,
         Test = test_type,
         r, p_raw = p_value, p_fdr, n) %>%
  mutate(
    r = round(r, 3),
    p_raw = format(p_raw, scientific = FALSE, digits = 4),
    p_fdr = format(p_fdr, scientific = FALSE, digits = 4),
    sig_raw = case_when(
      as.numeric(p_raw) < 0.001 ~ "***",
      as.numeric(p_raw) < 0.01 ~ "**",
      as.numeric(p_raw) < 0.05 ~ "*",
      TRUE ~ ""
    ),
    sig_fdr = case_when(
      as.numeric(p_fdr) < 0.001 ~ "***",
      as.numeric(p_fdr) < 0.01 ~ "**",
      as.numeric(p_fdr) < 0.05 ~ "*",
      TRUE ~ ""
    )
  ) %>%
  kable(caption = "YA Group: Correlations with Raw and FDR-Corrected P-values") %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed")) %>%
  scroll_box(height = "600px")
```

Table 1: YA Group: Correlations with Raw and FDR-Corrected P-values

Predictor	Outcome	Test	r	p_raw	p_fdr	n	sig_raw	sig_fdr
VR Experience	Mean Speed	Spearman	0.445	0.01382	0.3317	30	*	
VR Experience	Mean Distance	Spearman	0.017	0.93066	0.9944	30		
VR Experience	Mean Nav Time	Spearman	-0.074	0.69590	0.9374	30		
Video Game Usage	Mean Speed	Spearman	0.146	0.44083	0.8037	30		
Video Game Usage	Mean Distance	Spearman	0.137	0.46882	0.8037	30		
Video Game Usage	Mean Nav Time	Spearman	0.029	0.87928	0.9944	30		
SSS Pre	Mean Speed	Spearman	0.124	0.51405	0.8225	30		
SSS Pre	Mean Distance	Spearman	0.284	0.12854	0.8037	30		
SSS Pre	Mean Nav Time	Spearman	0.209	0.26688	0.8037	30		
SBSOD	Mean Speed	Pearson	0.161	0.39468	0.8037	30		
SBSOD	Mean Distance	Pearson	-0.053	0.78117	0.9374	30		
SBSOD	Mean Nav Time	Pearson	-0.153	0.41852	0.8037	30		
Trails A	Mean Speed	Pearson	-0.293	0.11662	0.8037	30		
Trails A	Mean Distance	Pearson	0.056	0.77071	0.9374	30		
Trails A	Mean Nav Time	Pearson	0.160	0.39931	0.8037	30		
Trails B	Mean Speed	Spearman	0.005	0.97952	0.9944	30		
Trails B	Mean Distance	Spearman	0.080	0.67394	0.9374	30		
Trails B	Mean Nav Time	Spearman	0.001	0.99441	0.9944	30		
Trails B-A (Switch Cost)	Mean Speed	Spearman	0.168	0.37467	0.8037	30		
Trails B-A (Switch Cost)	Mean Distance	Spearman	0.064	0.73677	0.9374	30		
Trails B-A (Switch Cost)	Mean Nav Time	Spearman	-0.181	0.33613	0.8037	30		
Corsi Total	Mean Speed	Pearson	0.216	0.25109	0.8037	30		
Corsi Total	Mean Distance	Pearson	-0.196	0.29998	0.8037	30		
Corsi Total	Mean Nav Time	Pearson	-0.213	0.25853	0.8037	30		

```
# Count significant correlations (using FDR-corrected p-values)
ya_sig_summary <- ya_correlations %>%
  summarise(
    total = n(),
    raw_p_05 = sum(p_value < 0.05, na.rm = TRUE),
    raw_p_01 = sum(p_value < 0.01, na.rm = TRUE),
    raw_p_001 = sum(p_value < 0.001, na.rm = TRUE),
    fdr_p_05 = sum(p_fdr < 0.05, na.rm = TRUE),
    fdr_p_01 = sum(p_fdr < 0.01, na.rm = TRUE),
    fdr_p_001 = sum(p_fdr < 0.001, na.rm = TRUE)
  )

cat("\n\nYA Significant Correlations (Raw p-values):\n")
```

```
##
```

```
##
## YA Significant Correlations (Raw p-values):
```

```
cat("p < 0.05:", ya_sig_summary$raw_p_05, "out of", ya_sig_summary$total, "\n")
```

```
## p < 0.05: 1 out of 24
```

```
cat("p < 0.01:", ya_sig_summary$raw_p_01, "out of", ya_sig_summary$total, "\n")
```

```
## p < 0.01: 0 out of 24
```

```
cat("p < 0.001:", ya_sig_summary$raw_p_001, "out of", ya_sig_summary$total, "\n")
```

```
## p < 0.001: 0 out of 24
```

```
cat("\n\nYA Significant Correlations (FDR-corrected):\n")
```

```
##
##
## YA Significant Correlations (FDR-corrected):
```

```
cat("p < 0.05:", ya_sig_summary$fdr_p_05, "out of", ya_sig_summary$total, "\n")
```

```
## p < 0.05: 0 out of 24
```

```
cat("p < 0.01:", ya_sig_summary$fdr_p_01, "out of", ya_sig_summary$total, "\n")
```

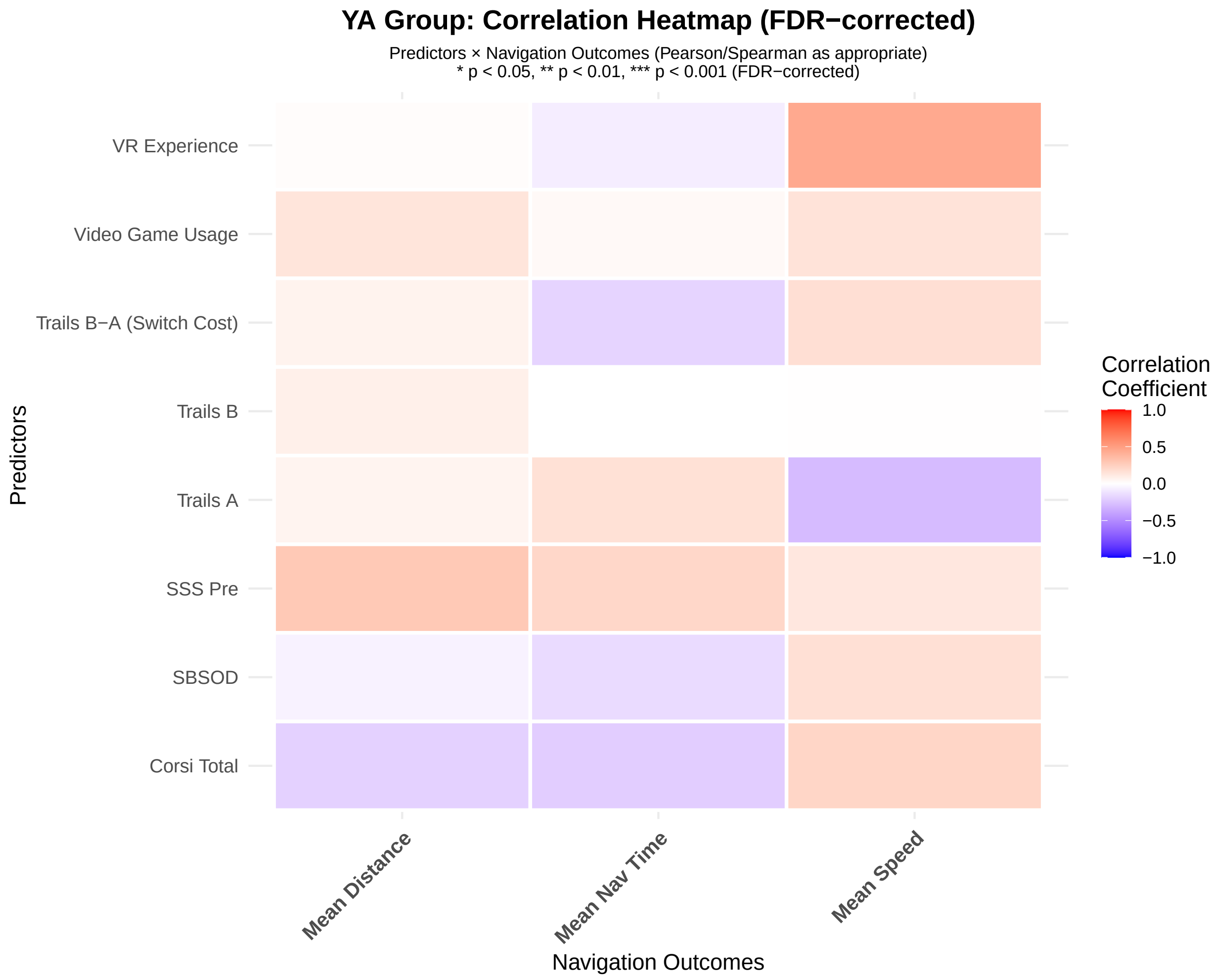
```
## p < 0.01: 0 out of 24
```

```
cat("p < 0.001:", ya_sig_summary$fdr_p_001, "out of", ya_sig_summary$total, "\n")
```

```
## p < 0.001: 0 out of 24
```

YA Correlation Heatmap (FDR-corrected)

```
# Create correlation matrix for heatmap
ya_corr_matrix <- ya_correlations %>%
  select(predictor_label, outcome_label, r) %>%
  pivot_wider(names_from = outcome_label, values_from = r) %>%
  column_to_rownames("predictor_label") %>%
  as.matrix()

ya_pval_matrix <- ya_correlations %>%
  select(predictor_label, outcome_label, p_fdr) %>%
  pivot_wider(names_from = outcome_label, values_from = p_fdr) %>%
  column_to_rownames("predictor_label") %>%
  as.matrix()

# Create significance annotations (using FDR-corrected p-values)
ya_sig_text <- matrix("", nrow = nrow(ya_pval_matrix), ncol = ncol(ya_pval_matrix))
rownames(ya_sig_text) <- rownames(ya_pval_matrix)
colnames(ya_sig_text) <- colnames(ya_pval_matrix)
ya_sig_text[ya_pval_matrix < 0.001] <- "***"
ya_sig_text[ya_pval_matrix >= 0.001 & ya_pval_matrix < 0.01] <- "**"
ya_sig_text[ya_pval_matrix >= 0.01 & ya_pval_matrix < 0.05] <- "*"

# Prepare data for ggplot
ya_corr_df <- as.data.frame(ya_corr_matrix) %>%
  rownames_to_column("Predictor") %>%
  pivot_longer(cols = -Predictor, names_to = "Outcome", values_to = "correlation")

ya_sig_df <- as.data.frame(ya_sig_text, stringsAsFactors = FALSE) %>%
  rownames_to_column("Predictor") %>%
  pivot_longer(cols = -Predictor, names_to = "Outcome", values_to = "sig")

plot_data <- ya_corr_df %>%
  left_join(ya_sig_df, by = c("Predictor", "Outcome"))

# Create heatmap
ggplot(plot_data, aes(x = Outcome, y = Predictor, fill = correlation)) +
  geom_tile(color = "white", size = 1) +
  geom_text(aes(label = sig), size = 6, vjust = 0.7, fontface = "bold") +
  scale_fill_gradient2(
    low = "blue", mid = "white", high = "red",
    midpoint = 0, limit = c(-1, 1), na.value = "grey90",
    name = "Correlation\nCoefficient"
  ) +
  theme_minimal(base_size = 13) +
  labs(
    title = "YA Group: Correlation Heatmap (FDR-corrected)",
    subtitle = "Predictors × Navigation Outcomes (Pearson/Spearman as appropriate)\n* p < 0.05, ** p < 0
    x = "Navigation Outcomes",
    y = "Predictors"
  ) +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1, size = 12, face = "bold"),
    axis.text.y = element_text(size = 11),
    plot.title = element_text(hjust = 0.5, size = 16, face = "bold"),
    plot.subtitle = element_text(hjust = 0.5, size = 10),
    legend.position = "right"
  )
```



OA Group Correlations with FDR Correction

```
cat("=== OLDER ADULTS (OA) CORRELATIONS ===\n\n")
```

```
## === OLDER ADULTS (OA) CORRELATIONS ===
```

```
# Filter OA data
oa_data <- merged_data %>% filter(Group == "OA")

cat("OA sample size:", nrow(oa_data), "\n\n")
```

```
## OA sample size: 30
```

```
# Compute all OA correlations
oa_correlations <- compute_correlations(oa_data, predictors, outcomes, predictor_methods)

# Add FDR correction within OA group
```

```r
oa_correlations <- oa_correlations %>%
  mutate(
    p_fdr = p.adjust(p_value, method = "fdr")
  )

# Add labels
oa_correlations <- oa_correlations %>%
  mutate(
    predictor_label = predictor_labels[match(predictor, predictors)],
    outcome_label = outcome_labels[match(outcome, outcomes)],
    test_type = ifelse(method == "pearson", "Pearson", "Spearman")
  )

# Display correlation table
cat("OA Correlation Results (with FDR correction):\n\n")
```

```
## OA Correlation Results (with FDR correction):
```

```r
oa_correlations %>%
  select(Predictor = predictor_label,
         Outcome = outcome_label,
         Test = test_type,
         r, p_raw = p_value, p_fdr, n) %>%
  mutate(
    r = round(r, 3),
    p_raw = format(p_raw, scientific = FALSE, digits = 4),
    p_fdr = format(p_fdr, scientific = FALSE, digits = 4),
    sig_raw = case_when(
      as.numeric(p_raw) < 0.001 ~ "***",
      as.numeric(p_raw) < 0.01 ~ "**",
      as.numeric(p_raw) < 0.05 ~ "*",
      TRUE ~ ""
    ),
    sig_fdr = case_when(
      as.numeric(p_fdr) < 0.001 ~ "***",
      as.numeric(p_fdr) < 0.01 ~ "**",
      as.numeric(p_fdr) < 0.05 ~ "*",
      TRUE ~ ""
    )
  ) %>%
  kable(caption = "OA Group: Correlations with Raw and FDR-Corrected P-values") %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed")) %>%
  scroll_box(height = "600px")
```

Table 2: OA Group: Correlations with Raw and FDR-Corrected P-values

Predictor	Outcome	Test	r	p_raw	p_fdr	n	sig_raw	sig_fdr
VR Experience	Mean Speed	Spearman	0.219	0.24539	0.7737	30		
VR Experience	Mean Distance	Spearman	-0.065	0.73126	0.9022	30		
VR Experience	Mean Nav Time	Spearman	-0.213	0.25791	0.7737	30		
Video Game Usage	Mean Speed	Spearman	0.012	0.95176	0.9518	30		

Video Game Usage	Mean Distance	Spearman	0.060	0.75186	0.9022	30	
Video Game Usage	Mean Nav Time	Spearman	0.034	0.85947	0.9518	30	
SSS Pre	Mean Speed	Spearman	-0.087	0.64584	0.9022	30	
SSS Pre	Mean Distance	Spearman	0.030	0.87529	0.9518	30	
SSS Pre	Mean Nav Time	Spearman	0.016	0.93266	0.9518	30	
SBSOD	Mean Speed	Pearson	0.215	0.25448	0.7737	30	
SBSOD	Mean Distance	Pearson	-0.311	0.09453	0.5958	30	
SBSOD	Mean Nav Time	Pearson	-0.318	0.08690	0.5958	30	
Trails A	Mean Speed	Pearson	-0.307	0.09931	0.5958	30	
Trails A	Mean Distance	Pearson	0.080	0.67399	0.9022	30	
Trails A	Mean Nav Time	Pearson	0.144	0.44645	0.8584	30	
Trails B	Mean Speed	Spearman	-0.139	0.46364	0.8584	30	
Trails B	Mean Distance	Spearman	0.112	0.55442	0.9022	30	
Trails B	Mean Nav Time	Spearman	0.176	0.35099	0.8584	30	
Trails B-A (Switch Cost)	Mean Speed	Spearman	-0.079	0.67725	0.9022	30	
Trails B-A (Switch Cost)	Mean Distance	Spearman	0.138	0.46496	0.8584	30	
Trails B-A (Switch Cost)	Mean Nav Time	Spearman	0.172	0.36196	0.8584	30	
Corsi Total	Mean Speed	Pearson	0.387	0.03463	0.5958	30	*
Corsi Total	Mean Distance	Pearson	-0.104	0.58606	0.9022	30	
Corsi Total	Mean Nav Time	Pearson	-0.233	0.21487	0.7737	30	

```r
# Count significant correlations
oa_sig_summary <- oa_correlations %>%
  summarise(
    total = n(),
    raw_p_05 = sum(p_value < 0.05, na.rm = TRUE),
    raw_p_01 = sum(p_value < 0.01, na.rm = TRUE),
    raw_p_001 = sum(p_value < 0.001, na.rm = TRUE),
    fdr_p_05 = sum(p_fdr < 0.05, na.rm = TRUE),
    fdr_p_01 = sum(p_fdr < 0.01, na.rm = TRUE),
    fdr_p_001 = sum(p_fdr < 0.001, na.rm = TRUE)
  )

cat("\n\nOA Significant Correlations (Raw p-values):\n")
```

```
##
##
## OA Significant Correlations (Raw p-values):
```

```r
cat("p < 0.05:", oa_sig_summary$raw_p_05, "out of", oa_sig_summary$total, "\n")
```

```
## p < 0.05: 1 out of 24
```

```r
cat("p < 0.01:", oa_sig_summary$raw_p_01, "out of", oa_sig_summary$total, "\n")
```

```
## p < 0.01: 0 out of 24
```

```r
cat("p < 0.001:", oa_sig_summary$raw_p_001, "out of", oa_sig_summary$total, "\n")
```

```
## p < 0.001: 0 out of 24
```

```r
cat("\n\nOA Significant Correlations (FDR-corrected):\n")
```

```
##
##
## OA Significant Correlations (FDR-corrected):
```

```r
cat("p < 0.05:", oa_sig_summary$fdr_p_05, "out of", oa_sig_summary$total, "\n")
```

```
## p < 0.05: 0 out of 24
```

```r
cat("p < 0.01:", oa_sig_summary$fdr_p_01, "out of", oa_sig_summary$total, "\n")
```

```
## p < 0.01: 0 out of 24
```

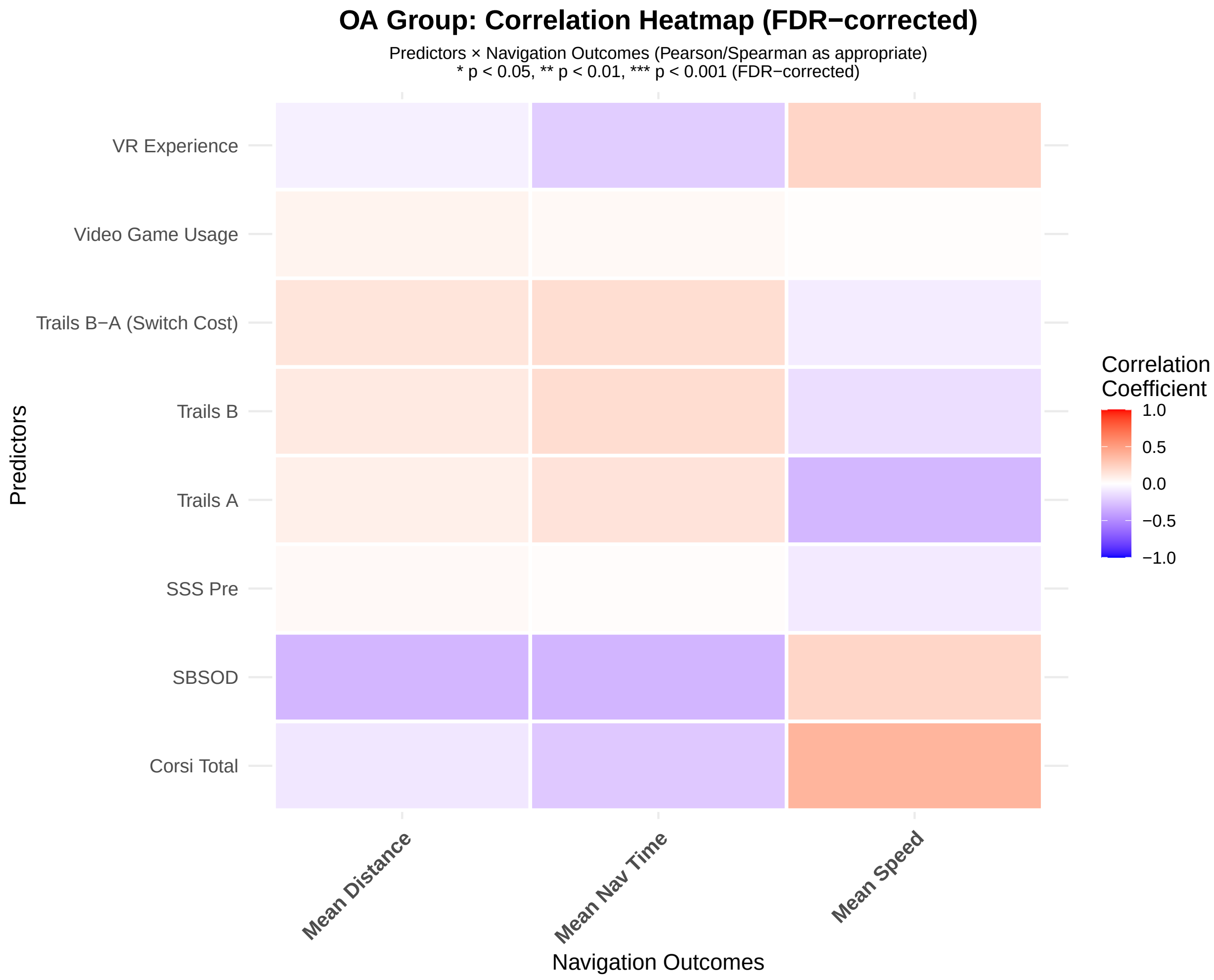
```r
cat("p < 0.001:", oa_sig_summary$fdr_p_001, "out of", oa_sig_summary$total, "\n")
```

```
## p < 0.001: 0 out of 24
```

OA Correlation Heatmap (FDR-corrected)

```r
# Create correlation matrix for heatmap
oa_corr_matrix <- oa_correlations %>%
  select(predictor_label, outcome_label, r) %>%
  pivot_wider(names_from = outcome_label, values_from = r) %>%
  column_to_rownames("predictor_label") %>%
  as.matrix()

oa_pval_matrix <- oa_correlations %>%
  select(predictor_label, outcome_label, p_fdr) %>%
  pivot_wider(names_from = outcome_label, values_from = p_fdr) %>%
  column_to_rownames("predictor_label") %>%
  as.matrix()

# Create significance annotations (using FDR-corrected p-values)
oa_sig_text <- matrix("", nrow = nrow(oa_pval_matrix), ncol = ncol(oa_pval_matrix))
oa_sig_text[oa_pval_matrix < 0.001] <- "***"
oa_sig_text[oa_pval_matrix >= 0.001 & oa_pval_matrix < 0.01] <- "**"
oa_sig_text[oa_pval_matrix >= 0.01 & oa_pval_matrix < 0.05] <- "*"

# Prepare data for ggplot
oa_corr_long <- melt(oa_corr_matrix, value.name = "correlation")
oa_sig_long <- melt(oa_sig_text, value.name = "sig")

# Convert factors to character
```

```r
oa_corr_long <- oa_corr_long %>%
  mutate(Var1 = as.character(Var1), Var2 = as.character(Var2))
oa_sig_long <- oa_sig_long %>%
  mutate(Var1 = as.character(Var1), Var2 = as.character(Var2))

plot_data <- oa_corr_long %>%
  left_join(oa_sig_long, by = c("Var1", "Var2"))

# Create heatmap
ggplot(plot_data, aes(x = Var2, y = Var1, fill = correlation)) +
  geom_tile(color = "white", size = 1) +
  geom_text(aes(label = sig), size = 6, vjust = 0.7, fontface = "bold") +
  scale_fill_gradient2(
    low = "blue", mid = "white", high = "red",
    midpoint = 0, limit = c(-1, 1), na.value = "grey90",
    name = "Correlation\nCoefficient"
  ) +
  theme_minimal(base_size = 13) +
  labs(
    title = "OA Group: Correlation Heatmap (FDR-corrected)",
    subtitle = "Predictors × Navigation Outcomes (Pearson/Spearman as appropriate)\n* p < 0.05, ** p < 0
    x = "Navigation Outcomes",
    y = "Predictors"
  ) +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1, size = 12, face = "bold"),
    axis.text.y = element_text(size = 11),
    plot.title = element_text(hjust = 0.5, size = 16, face = "bold"),
    plot.subtitle = element_text(hjust = 0.5, size = 10),
    legend.position = "right"
  )
```



Summary Comparison

```
cat("=== SUMMARY: YA vs OA CORRELATIONS (FDR-corrected) ===\n\n")
```

```
## === SUMMARY: YA vs OA CORRELATIONS (FDR-corrected) ===
```

```
# Combine results for comparison
combined_summary <- bind_rows(
  ya_sig_summary %>% mutate(Group = "YA"),
  oa_sig_summary %>% mutate(Group = "OA")
) %>%
  select(Group, total,
         raw_p_05, raw_p_01, raw_p_001,
         fdr_p_05, fdr_p_01, fdr_p_001)

cat("Significant Correlations by Group:\n")
```

```
## Significant Correlations by Group:
```

```
print(combined_summary)
```

```
## # A tibble: 2 x 8
##   Group total raw_p_05 raw_p_01 raw_p_001 fdr_p_05 fdr_p_01 fdr_p_001
##   <chr> <int>    <int>    <int>     <int>    <int>    <int>     <int>
## 1 YA       24        1        0         0        0        0         0
## 2 OA       24        1        0         0        0        0         0
```

```
# Identify strongest correlations that survive FDR correction
cat("\n\nStrongest FDR-Significant Positive Correlations:\n\n")
```

```
##
##
## Strongest FDR-Significant Positive Correlations:
```

```
cat("YA Group:\n")
```

```
## YA Group:
```

```
ya_correlations %>%
  filter(p_fdr < 0.05) %>%
  arrange(desc(r)) %>%
  head(5) %>%
  select(Predictor = predictor_label, Outcome = outcome_label, Test = test_type, r, p_fdr, n) %>%
  mutate(r = round(r, 3), p_fdr = round(p_fdr, 4)) %>%
  kable() %>%
  kable_styling()
```

Predictor	Outcome	Test	r	p_fdr	n
NA	NA	NA	NA	NA	NA
:———	:——-	:—-	–:	——:	–:

```
cat("\n\nOA Group:\n")
```

```
##
##
## OA Group:
```

```
oa_correlations %>%
  filter(p_fdr < 0.05) %>%
  arrange(desc(r)) %>%
  head(5) %>%
  select(Predictor = predictor_label, Outcome = outcome_label, Test = test_type, r, p_fdr, n) %>%
  mutate(r = round(r, 3), p_fdr = round(p_fdr, 4)) %>%
  kable() %>%
  kable_styling()
```

Predictor	Outcome	Test	r	p_fdr	n
NA	NA	NA	NA	NA	NA
:———	:———-	:—-	–:	——:	–:

```r
cat("\n\nStrongest FDR-Significant Negative Correlations:\n\n")
```

```
##
##
## Strongest FDR-Significant Negative Correlations:
```

```r
cat("YA Group:\n")
```

```
## YA Group:
```

```r
ya_correlations %>%
  filter(p_fdr < 0.05) %>%
  arrange(r) %>%
  head(5) %>%
  select(Predictor = predictor_label, Outcome = outcome_label, Test = test_type, r, p_fdr, n) %>%
  mutate(r = round(r, 3), p_fdr = round(p_fdr, 4)) %>%
  kable() %>%
  kable_styling()
```

Predictor	Outcome	Test	r	p_fdr	n
NA	NA	NA	NA	NA	NA
:———	:———-	:—-	–:	——:	–:

```r
cat("\n\nOA Group:\n")
```

```
##
##
## OA Group:
```

```r
oa_correlations %>%
  filter(p_fdr < 0.05) %>%
  arrange(r) %>%
  head(5) %>%
  select(Predictor = predictor_label, Outcome = outcome_label, Test = test_type, r, p_fdr, n) %>%
  mutate(r = round(r, 3), p_fdr = round(p_fdr, 4)) %>%
  kable() %>%
  kable_styling()
```

Predictor	Outcome	Test	r	p_fdr	n
NA	NA	NA	NA	NA	NA
:———	:———-	:—-	–:	——:	–:

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
## 
## 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] corrplot_0.92    reshape2_1.4.4   kableExtra_1.4.0 knitr_1.43      
##  [5] broom_1.0.5      lubridate_1.9.2  forcats_1.0.0    stringr_1.5.0   
##  [9] dplyr_1.1.2      purrr_1.0.1      readr_2.1.4      tidyr_1.3.0     
## [13] 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] cli_3.6.1         crayon_1.5.2      rlang_1.1.6       bit64_4.0.5      
## [21] munsell_0.5.0     withr_2.5.0       yaml_2.3.7        parallel_4.3.1   
## [25] tools_4.3.1       tzdb_0.4.0        colorspace_2.1-0  vctrs_0.6.3      
## [29] R6_2.5.1          lifecycle_1.0.3   bit_4.0.5         vroom_1.6.3      
## [33] pkgconfig_2.0.3   pillar_1.9.0      gtable_0.3.3      glue_1.6.2       
## [37] Rcpp_1.0.10       systemfonts_1.0.4 highr_0.10        xfun_0.39        
## [41] tidyselect_1.2.0  rstudioapi_0.14   farver_2.1.1      htmltools_0.5.5  
## [45] labeling_0.4.2    rmarkdown_2.23    svglite_2.1.2     compiler_4.3.1
```