

CRISSP Research Analysis - Cognitive Outcomes Associated with Rare Genetic Variants in a Mexican Youth Cohort

Azra Narvel

2025-08-05

Introduction

- Investigating the relationship between rare genetic variants and cognitive performance.
- Focus on SCHEMA-implicated genes and domain-specific cognitive profiles.

Data Overview

Cohort Description

- Total participants: 548
- Diagnosed with psychosis: 276
- No psychosis diagnosis: 272
- Carrier of rare variant: 112
- Non-carrier of rare variant: 436
- Sex (Male): 251
- Sex (Female): 297
- Age Range: 10-21 years
- Mean age: 15 years

Genetic Data

- Rare CNVs & deleterious SNVs identified from SCHEMA gene list.
- Genes grouped into functional categories (Synaptic, Chromatin, Protein Regulation, Cell Cycle).
- SCHEMA Genes Analyzed: CACNA1B, CACNA1G, GRIN2A, MAGI2, NLGN2, SP4, TRIO, SETD1A, JARID2, HDAC9, ZNF318, TOP2B, CUL1, HERC1, PSMA3, STAG1, CDK13, AKAP11, ATP9A, NBEA.

Cognitive Domains

- General Cognition
- Facial Memory
- Processing Speed (Digit Symbol)
- Verbal Learning (CVLT)
- Verbal Fluency
- Verbal Memory

Data Cleaning

```
# Load necessary libraries
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
```

```
## v dplyr      1.1.4      v readr      2.1.5
```

```
## v forcats    1.0.0      v stringr   1.5.1
```

```
## v ggplot2     3.5.2      v tibble    3.3.0
```

```
## v lubridate  1.9.4      v tidyr     1.3.1
```

```
## v purrr      1.0.4
```

```
## -- Conflicts ----- tidyverse_conflicts() --
```

```
## x dplyr::filter() masks stats::filter()
```

```
## x dplyr::lag()     masks stats::lag()
```

```
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(ggplot2)
```

```
library(readr)
```

```
# Import data files
```

```
all_gScores <- read_csv("epimex_g_10june2025.csv") # all g scores
```

```
## Rows: 2680 Columns: 2
```

```
## -- Column specification -----
```

```
## Delimiter: ","
```

```
## chr (1): studyid
```

```
## dbl (1): g
```

```
##
```

```
## i Use 'spec()' to retrieve the full column specification for this data.
```

```
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
all_cogTests <- read_csv("epimex_gorilla_10june2025.csv") # all cog tests
```

```
## Rows: 2680 Columns: 20
```

```
## -- Column specification -----
```

```
## Delimiter: ","
```

```
## chr (1): studyid
```

```
## dbl (19): matrixreasoning_trials, matrixreasoning_correct, cvlt_correct, cvl...
```

```
##
```

```
## i Use 'spec()' to retrieve the full column specification for this data.
```

```
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
all_phenotype <- read_csv("epimex_12feb2025_sciddx_corrected_LMS.csv") # all phenotype data
```

```
## Warning: One or more parsing issues, call 'problems()' on your data frame for details,
```

```
## e.g.:
```

```
## dat <- vroom(...)
```

```
## problems(dat)
```

```

## Rows: 2280 Columns: 104
## -- Column specification -----
## Delimiter: ","
## chr (16): family_id2, info_yearsofedcomments, recruitment_site_other, info_d...
## dbl (87): studyid, studyid_2, relation_id, family_id, proband_2, control, co...
## lgl (1): cohab_sib
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.

probands <- read.table("epimex_probands.fam", header = FALSE, sep = "", stringsAsFactors = FALSE) # pro
colnames(probands) <- c("FamilyID", "studyid", "PaternalID", "MaternalID", "sex", "phenotype") # update
probands$group <- "Proband"

controls <- read.table("epimex_controls.fam", header = FALSE, sep = "", stringsAsFactors = FALSE) # con
colnames(controls) <- c("FamilyID", "studyid", "PaternalID", "MaternalID", "sex", "phenotype") # update
controls$group <- "Control"

# Filter data

# filter phenotype data down to ID and age
all_ages <- all_phenotype[, c("studyid", "age_2")]
all_ages <- all_ages %>%
  rename(
    age = age_2
  )

# filter proband and control data down to ID, sex, and phenotype
probands <- probands[, c("studyid", "sex", "phenotype", "group")]
controls <- controls[, c("studyid", "sex", "phenotype", "group")]

# combine proband and control data
combined_phenotype <- bind_rows(controls, probands)

# combine data with age
combined_phenotype <- combined_phenotype %>%
  left_join(all_ages, by = "studyid")

# filter g scores and cog data down to the probands and controls
filtered_gScores <- all_gScores[all_gScores$studyid %in% combined_phenotype$studyid, ]
filtered_cogTests <- all_cogTests[all_cogTests$studyid %in% combined_phenotype$studyid, ]

# Clean Cognitive Data

## Matrix Reasoning Test

# calculate accuracy
filtered_cogTests$matrixreasoning_accuracy <- filtered_cogTests$matrixreasoning_correct / filtered_cogT

# standardize accuracy score (z-score)
filtered_cogTests$matrixreasoning_z <- scale(filtered_cogTests$matrixreasoning_accuracy)

```

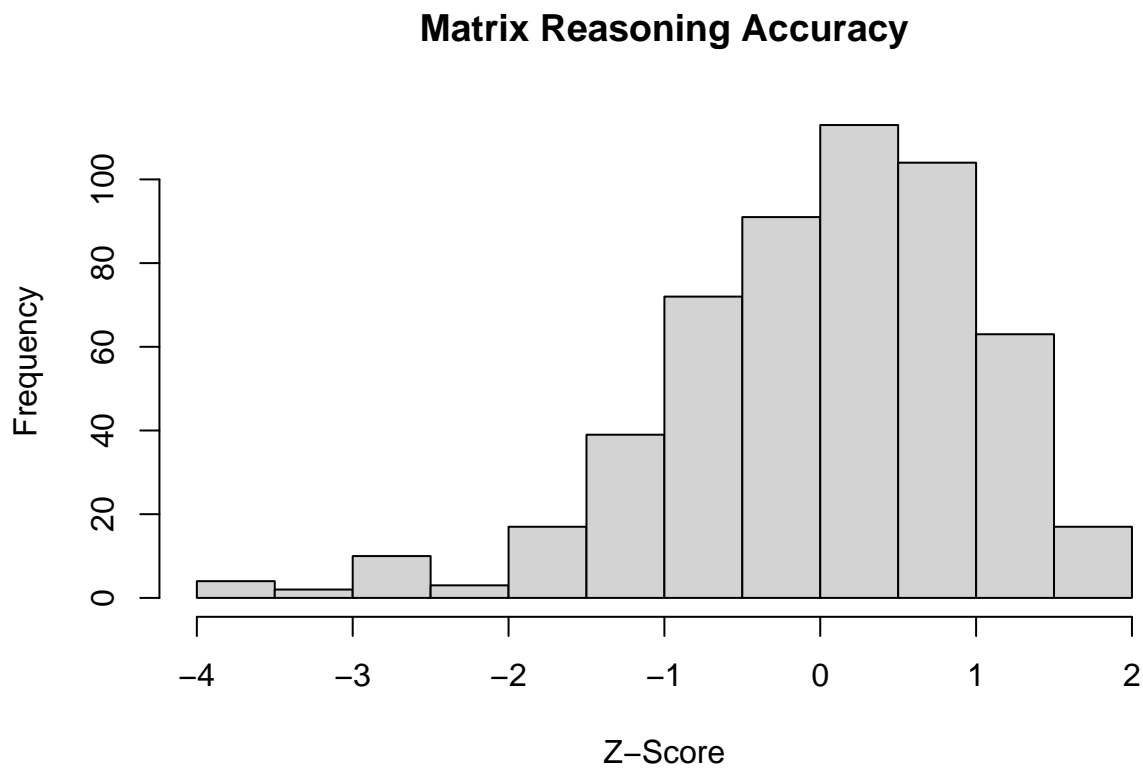
```
# check work and visualize distribution
summary(filtered_cogTests$matrixreasoning_accuracy)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.1667 0.5904 0.6970 0.6788 0.7812 0.9429
```

```
summary(filtered_cogTests$matrixreasoning_z)
```

```
##      V1
## Min.   :-3.8359
## 1st Qu.: -0.6622
## Median : 0.1359
## Mean    : 0.0000
## 3rd Qu.: 0.7671
## Max.    : 1.9774
```

```
hist(filtered_cogTests$matrixreasoning_z, main = "Matrix Reasoning Accuracy", xlab = "Z-Score")
```



```
## CVLT (California Verbal Learning Test)
```

```
# cult_correct - total number of correctly recalled words
# cult_dprime - ability to distinguish targets from distractors
```

```
# standardize scores (z-score)
filtered_cogTests$cvlt_correct_z <- scale(filtered_cogTests$cvlt_correct)
filtered_cogTests$cvlt_dprime_z <- scale(filtered_cogTests$cvlt_dprime)

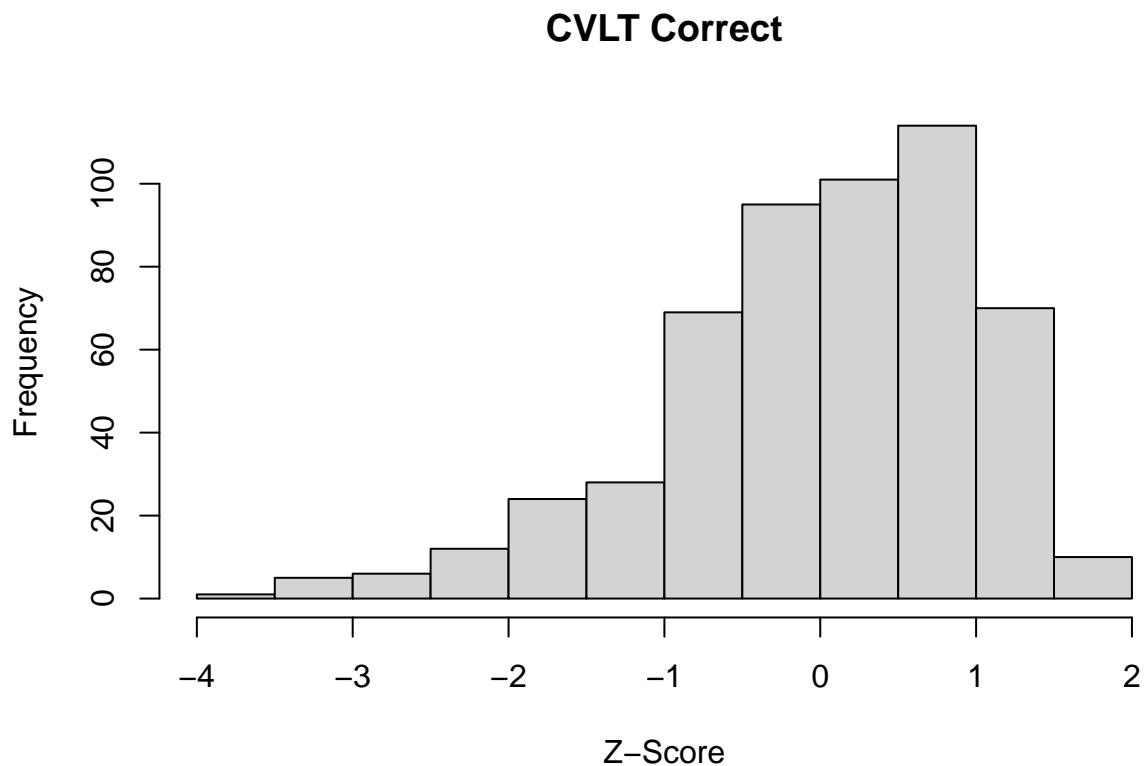
# check summaries and visualize distribution
summary(filtered_cogTests$cvlt_correct)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      8.00  48.00   56.00   54.85  64.00   78.00
```

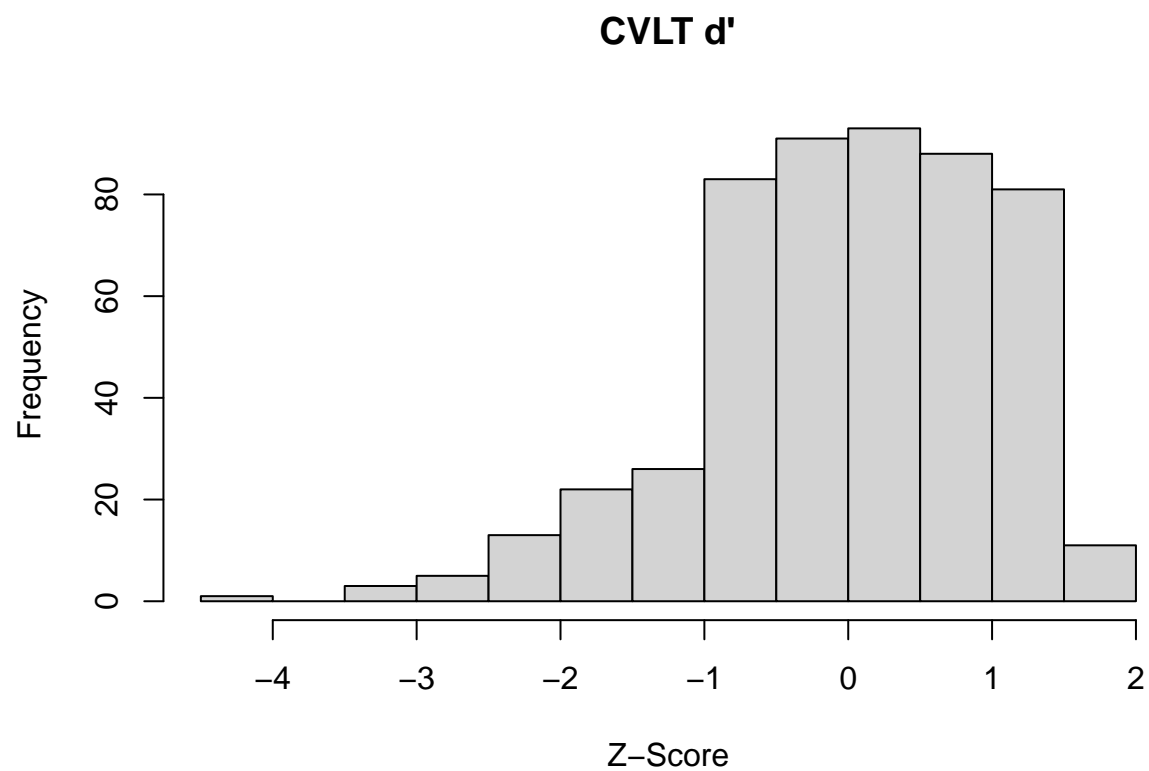
```
summary(filtered_cogTests$cvlt_dprime)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
##  0.5431  2.1536  2.4585  2.4207  2.7702  3.1787     18
```

```
hist(filtered_cogTests$cvlt_correct_z, main = "CVLT Correct", xlab = "Z-Score")
```

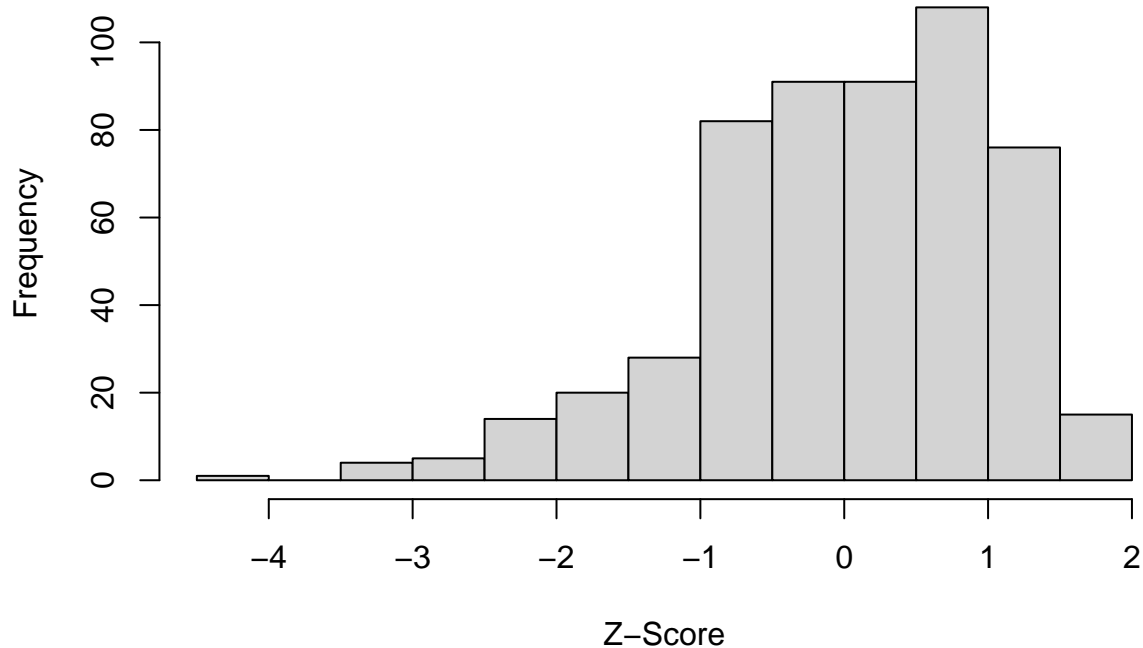


```
hist(filtered_cogTests$cvlt_dprime_z, main = "CVLT d'", xlab = "Z-Score")
```



```
filtered_cogTests$verbal_learning_z <- rowMeans(  
  filtered_cogTests[, c("cvlt_correct_z", "cvlt_dprime_z")],  
  na.rm = TRUE  
)  
hist(filtered_cogTests$verbal_learning_z, main = "Verbal Learning Composite", xlab = "Z-Score")
```

Verbal Learning Composite



```
## Verbal Fluency
```

```
# pmr - phonemic fluency
# animal - semantic fluency
```

```
# check summaries
```

```
summary(filtered_cogTests$verbalfluency_es_pmr)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
##      3.00  22.00   29.00   29.81  37.00   70.00     1
```

```
summary(filtered_cogTests$verbalfluency_es_animal)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
##      0.00  17.00   21.00   20.48  25.00   42.00     1
```

```
# standardize (z-score)
```

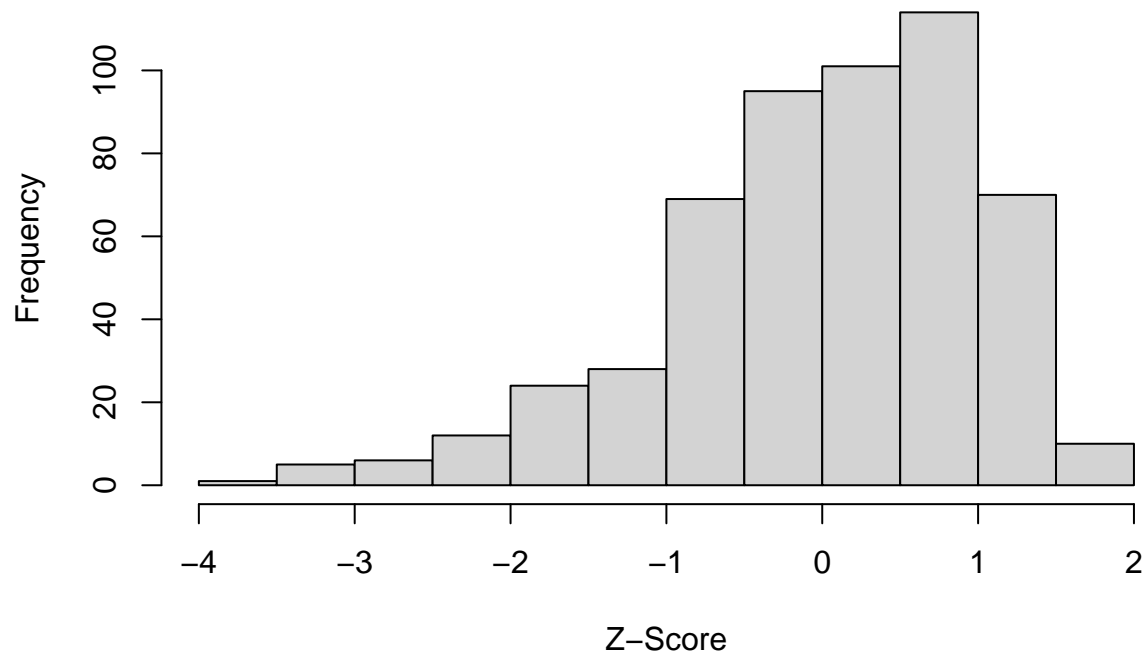
```
filtered_cogTests$verbalfluency_es_pmr_z <- scale(filtered_cogTests$verbalfluency_es_pmr)
```

```
filtered_cogTests$verbalfluency_es_animal_z <- scale(filtered_cogTests$verbalfluency_es_animal)
```

```
# visualize
```

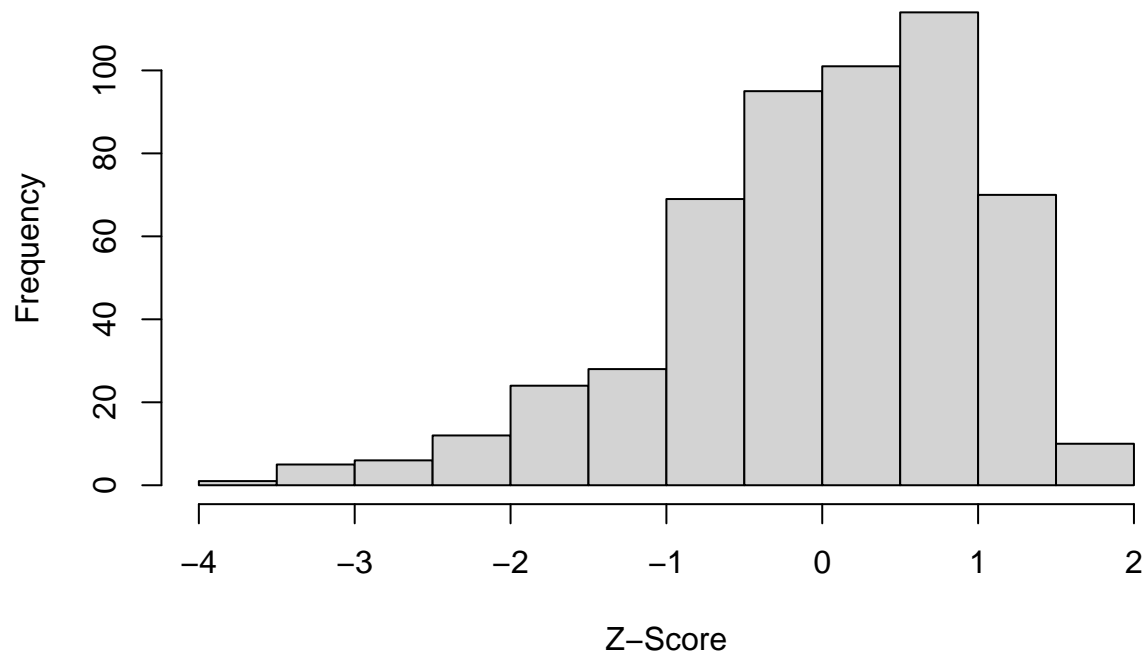
```
hist(filtered_cogTests$cvlt_correct_z, main = "Phonemic Verbal Fluency", xlab = "Z-Score")
```

Phonemic Verbal Fluency



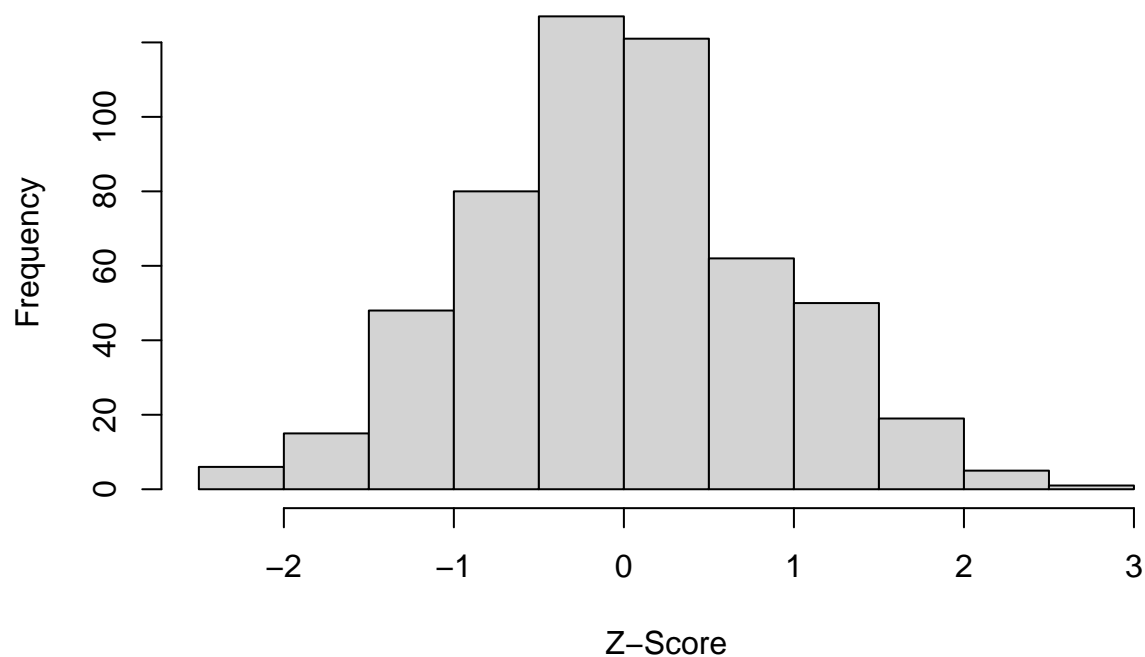
```
hist(filtered_cogTests$cvlt_correct_z, main = "Semantic Verbal Fluency", xlab = "Z-Score")
```


Semantic Verbal Fluency



```
# composite verbal fluency (takes avg per person across both columns)
filtered_cogTests$verbalfluency_composite <- rowMeans(
  filtered_cogTests[, c("verbalfluency_es_pmr_z", "verbalfluency_es_animal_z")],
  na.rm = TRUE
)
hist(filtered_cogTests$verbalfluency_composite, main = "Verbal Fluency Composite", xlab = "Z-Score")
```

Verbal Fluency Composite



```
## Facial Memory
```

```
# check summaries
```

```
summary(filtered_cogTests$facialmemory_correct)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's  
##      11.00   51.00   57.50   56.75   64.00   77.00     1
```

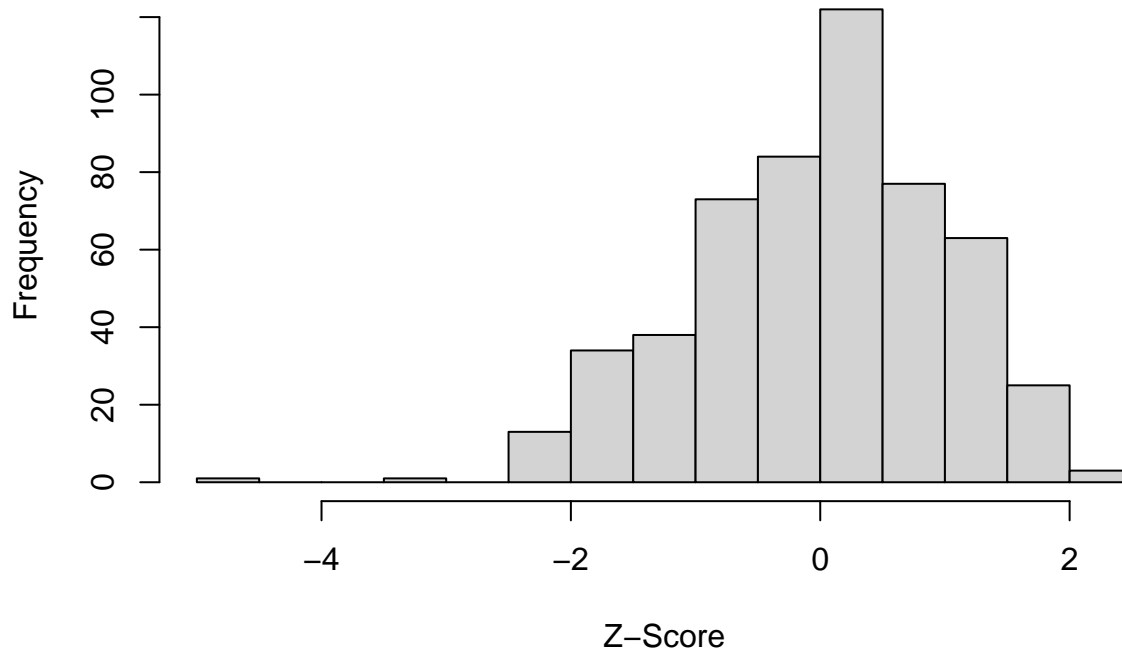
```
# standardize (z-score)
```

```
filtered_cogTests$facialmemory_z <- scale(filtered_cogTests$facialmemory_correct)
```

```
# visualize
```

```
hist(filtered_cogTests$facialmemory_z, main = "Facial Memory", xlab = "Z-Score")
```

Facial Memory



```
## Verbal Working Memory
```

```
# forward_mns - simple span
```

```
# backward_mns - reversed span
```

```
# lns_mns - letter number sequencing
```

```
# summaries
```

```
summary(filtered_cogTests$verbalworkingmemory_forward_mns)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      2.000  4.000  4.500  4.578  5.000  7.500
```

```
summary(filtered_cogTests$verbalworkingmemory_backward_mns)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
##      2.00  3.00  3.50  3.34  3.50  7.00    24
```

```
summary(filtered_cogTests$verbalworkingmemory_lns_mns)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
##      1.000  2.667  3.333  3.197  3.667  5.667     2
```

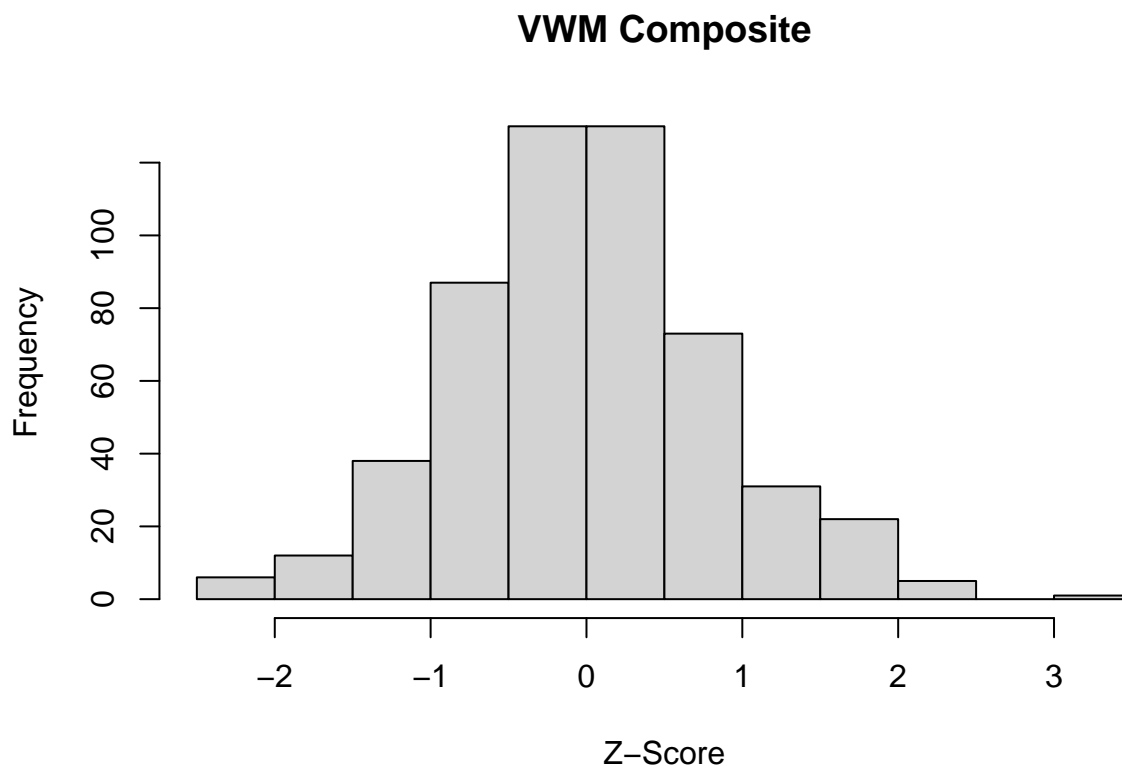
```

# scale
filtered_cogTests$vwm_forward_z <- scale(filtered_cogTests$verbalworkingmemory_forward_mns)
filtered_cogTests$vwm_backward_z <- scale(filtered_cogTests$verbalworkingmemory_backward_mns)
filtered_cogTests$vwm_lns_z <- scale(filtered_cogTests$verbalworkingmemory_lns_mns)

# composite scores
filtered_cogTests$vwm_composite <- rowMeans(
  filtered_cogTests[, c("vwm_forward_z", "vwm_backward_z", "vwm_lns_z")],
  na.rm = TRUE
)

hist(filtered_cogTests$vwm_composite, main = "VWM Composite", xlab = "Z-Score")

```



```
summary(filtered_cogTests$vwm_composite)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.    Max.
## -2.460126 -0.548849 -0.033826 -0.000664  0.495027  3.049111
```

```
## Digit Symbols - processing speed
```

```

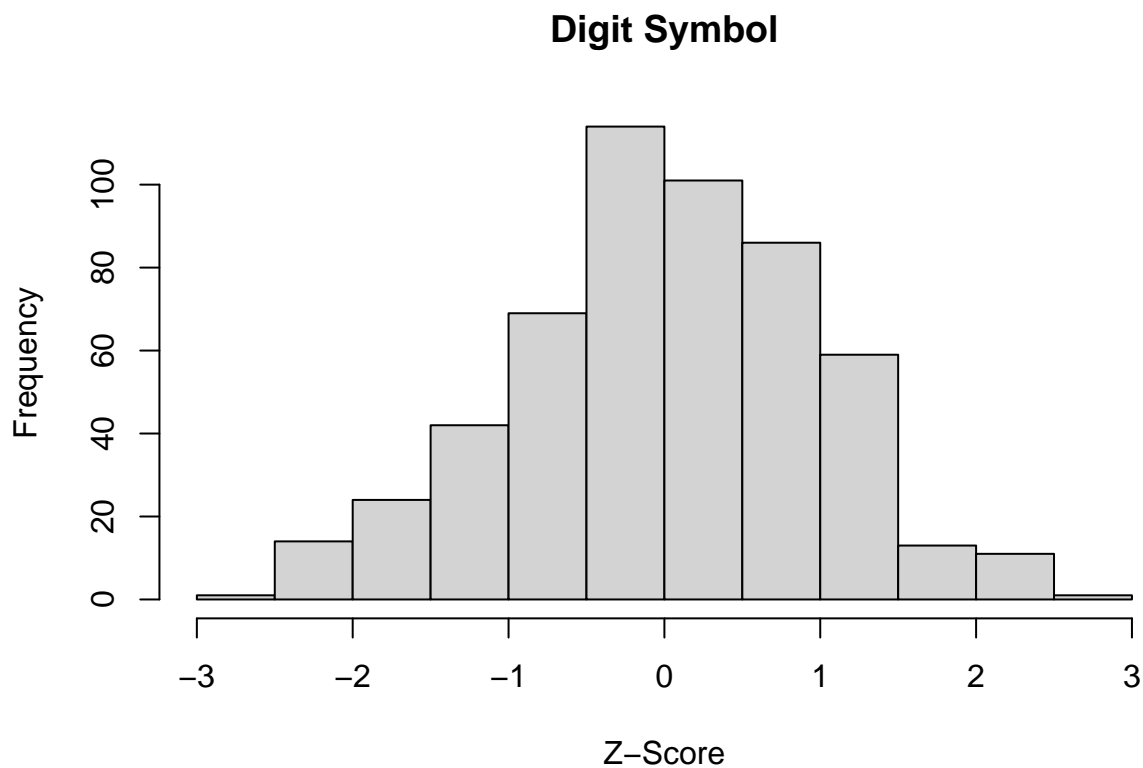
# standardize
filtered_cogTests$digitymbol1_score_z <- scale(filtered_cogTests$digitymbol1_score)
filtered_cogTests$digitymbol2_score_z <- scale(filtered_cogTests$digitymbol2_score)
filtered_cogTests$digitymbol_score_z <- scale(filtered_cogTests$digitymbol_score)

```

```
# composite score
filtered_cogTests$digitymbol_composite <- rowMeans(
  filtered_cogTests[, c("digitymbol1_score_z", "digitymbol2_score_z", "digitymbol_score_z")],
  na.rm = TRUE
)
summary(filtered_cogTests$digitymbol_composite)

##      Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
## -2.53553 -0.62410  0.01985  0.00000  0.65322  2.51523

hist(filtered_cogTests$digitymbol_composite, main = "Digit Symbol", xlab = "Z-Score")
```



```
## Combine cognitive data into a new data frame
composite_cogTests <- filtered_cogTests %>%
  select(studyid, digitymbol_composite, facialmemory_z, verbalfluency_composite, verbal_learning_z, ma)
view(composite_cogTests)

## Combine general cognition scores with the other cognitive tests
final_cogData <- composite_cogTests %>%
  left_join(filtered_gScores, by = "studyid")
view(final_cogData)

final_cogData$studyid <- as.character(final_cogData$studyid)
```

```

combined_phenotype$studyid <- as.character(combined_phenotype$studyid)

## Combine cognitive and phenotype data
pheno_cog <- final_cogData %>%
  left_join(combined_phenotype, by = "studyid")
view(pheno_cog)

# Import carrier data
carriers <- read_csv("~/Downloads/cnv_snv_carriers(Sheet1).csv")

## Rows: 159 Columns: 7
## -- Column specification -----
## Delimiter: ","
## chr (4): gene, variant_type, group, carrier_type
## dbl (3): studyid, is_cnv_carrier, is_snv_carrier
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.

carriers <- carriers %>%
  mutate(
    is_carrier = 1
  )

# filter carrier data down to the probands and controls
filtered_carriers <- carriers[carriers$studyid %in% combined_phenotype$studyid, ]

filtered_carriers$studyid <- as.character(filtered_carriers$studyid)
pheno_cog$studyid <- as.character(pheno_cog$studyid)

n_occur <- data.frame(table(filtered_carriers$studyid))

# add carrier status to the combined phenotype and cognitive data
carriers_combined <- pheno_cog %>%
  left_join(filtered_carriers, by = "studyid")

carriers_combined$is_carrier[is.na(carriers_combined$is_carrier)] <- 0

```

Exploratory Visualizations Comparing Cognitive Scores for Probands and Controls

```

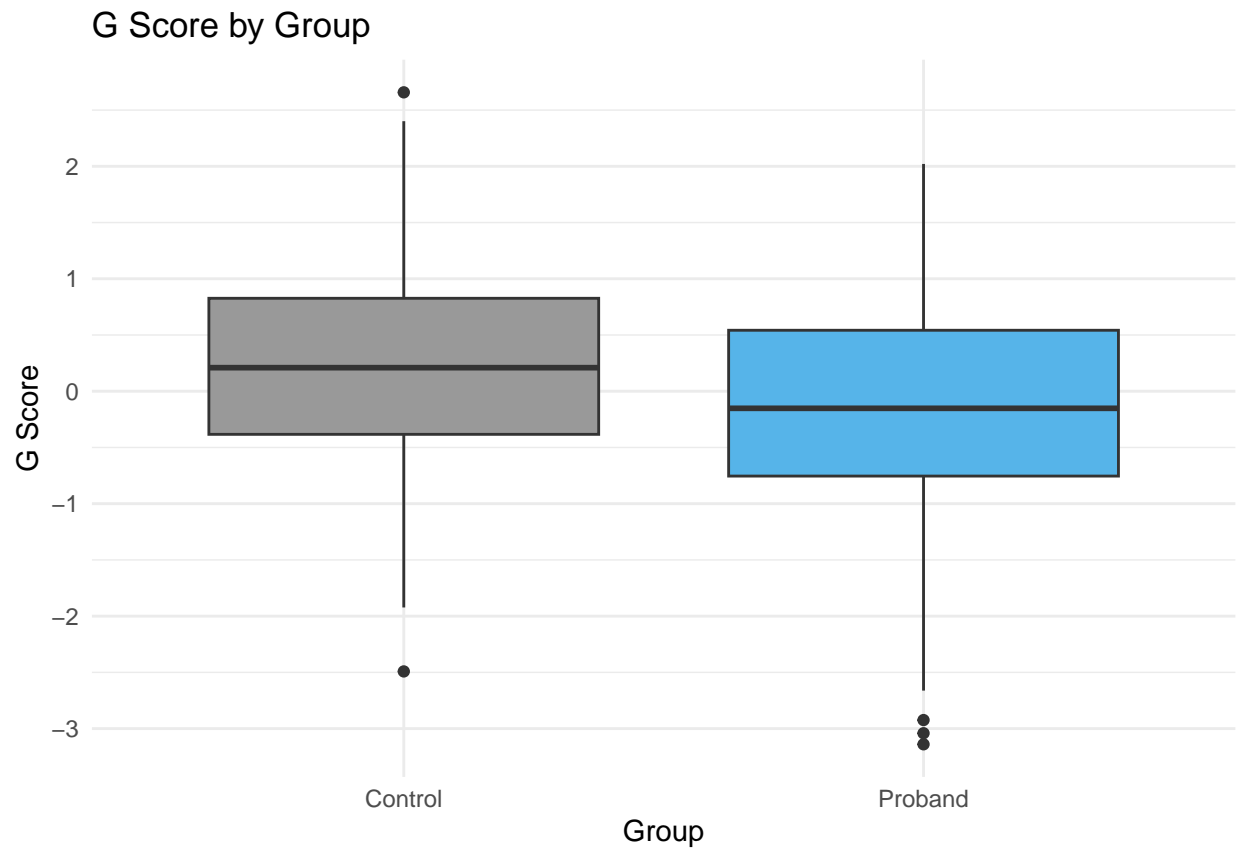
ggplot(pheno_cog, aes(x=factor(group), y = g)) +
  geom_boxplot(fill = c("#999999", "#56B4E9")) +
  labs(x = "Group", y = "G Score", title = "G Score by Group") +
  theme_minimal()

```

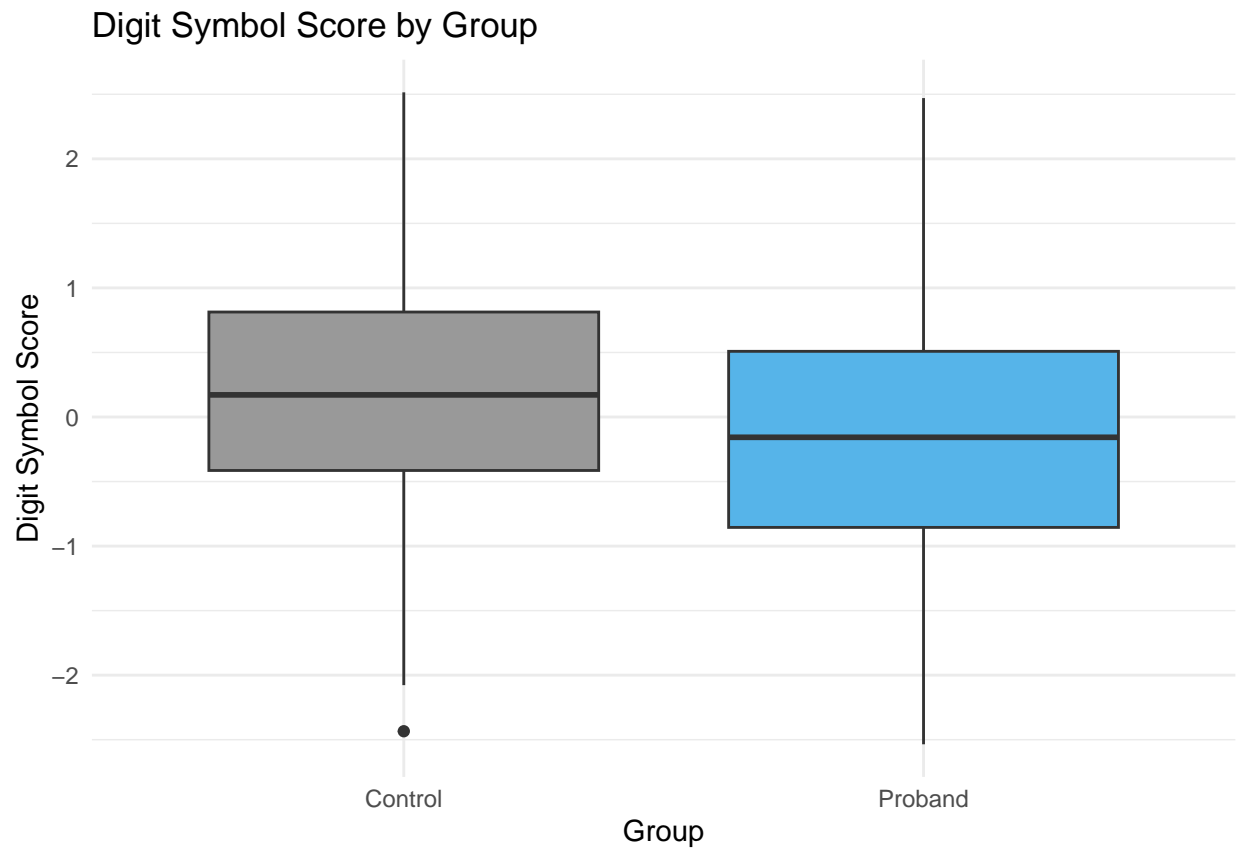
```

## Warning: Removed 46 rows containing non-finite outside the scale range
## ('stat_boxplot()').

```

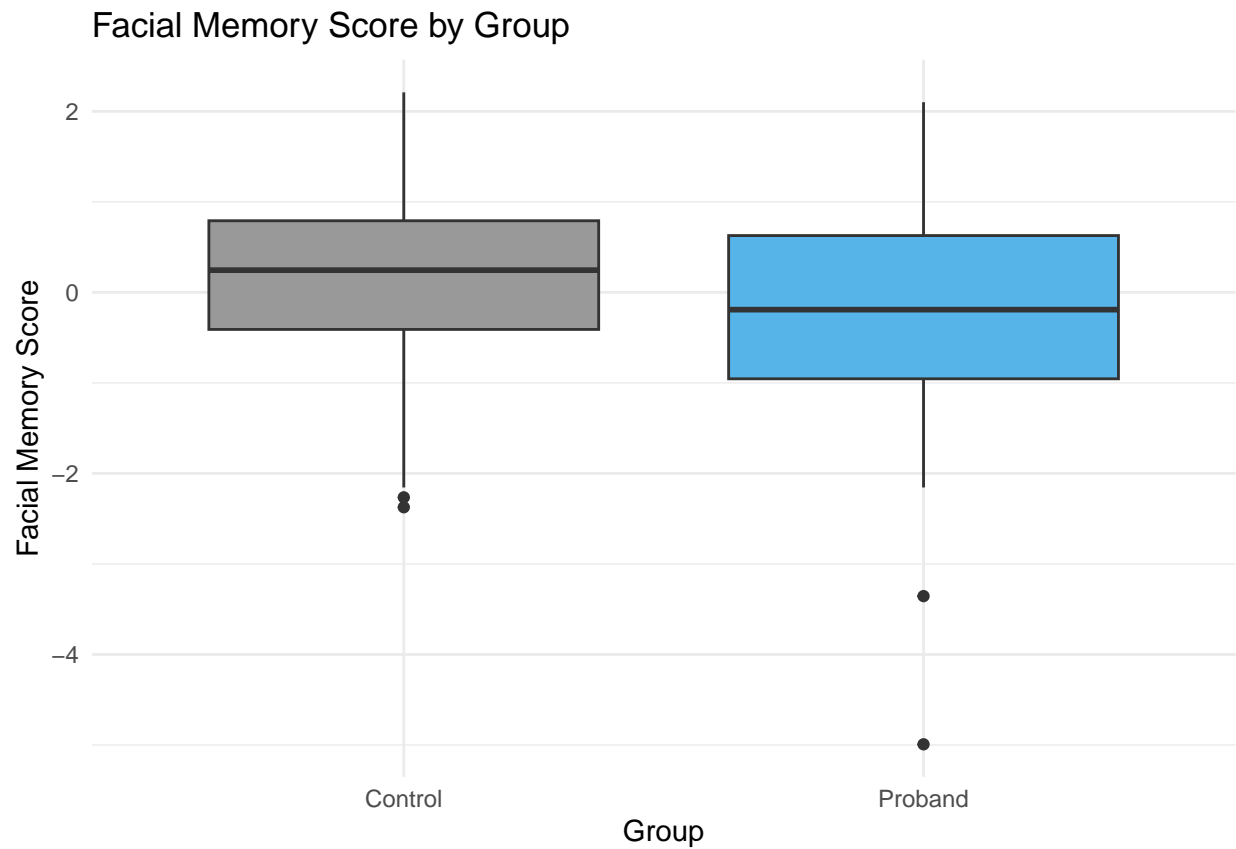


```
ggplot(pheno_cog, aes(x=factor(group), y = digitsymbol_composite)) +  
  geom_boxplot(fill = c("#999999", "#56B4E9")) +  
  labs(x = "Group", y = "Digit Symbol Score", title = "Digit Symbol Score by Group") +  
  theme_minimal()
```



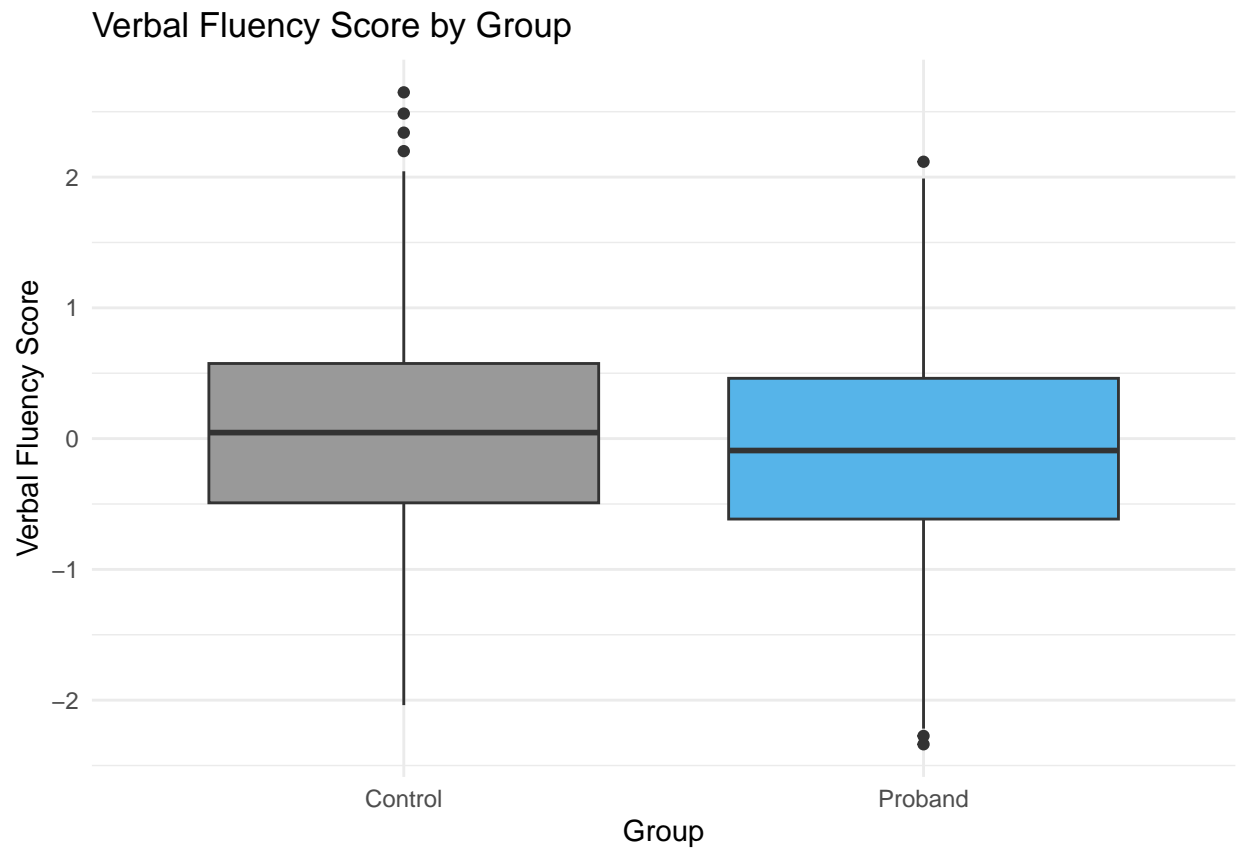
```
ggplot(pheno_cog, aes(x=factor(group), y = facialmemory_z)) +  
  geom_boxplot(fill = c("#999999", "#56B4E9")) +  
  labs(x = "Group", y = "Facial Memory Score", title = "Facial Memory Score by Group") +  
  theme_minimal()
```

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

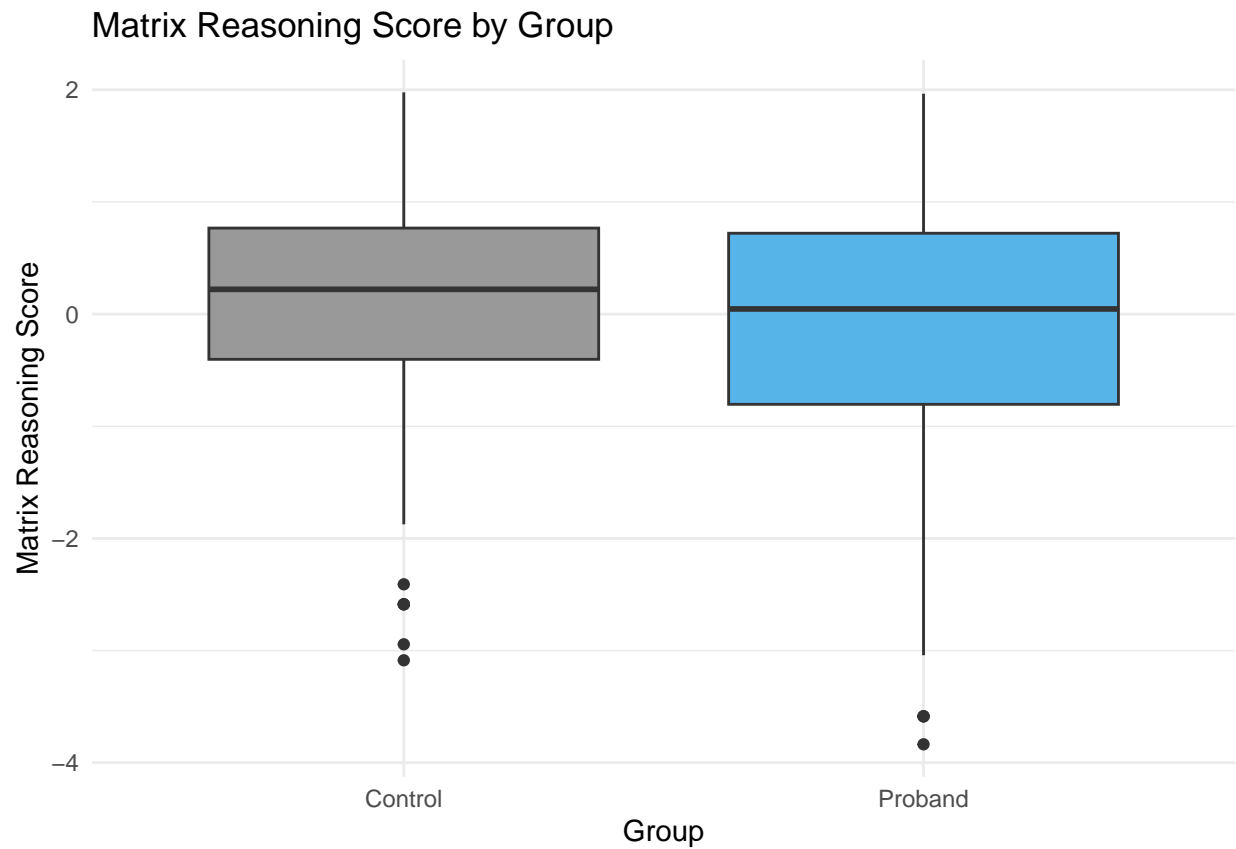



```
ggplot(pheno_cog, aes(x=factor(group), y = verbalfluency_composite)) +
  geom_boxplot(fill = c("#999999", "#56B4E9")) +
  labs(x = "Group", y = "Verbal Fluency Score", title = "Verbal Fluency Score by Group") +
  theme_minimal()
```

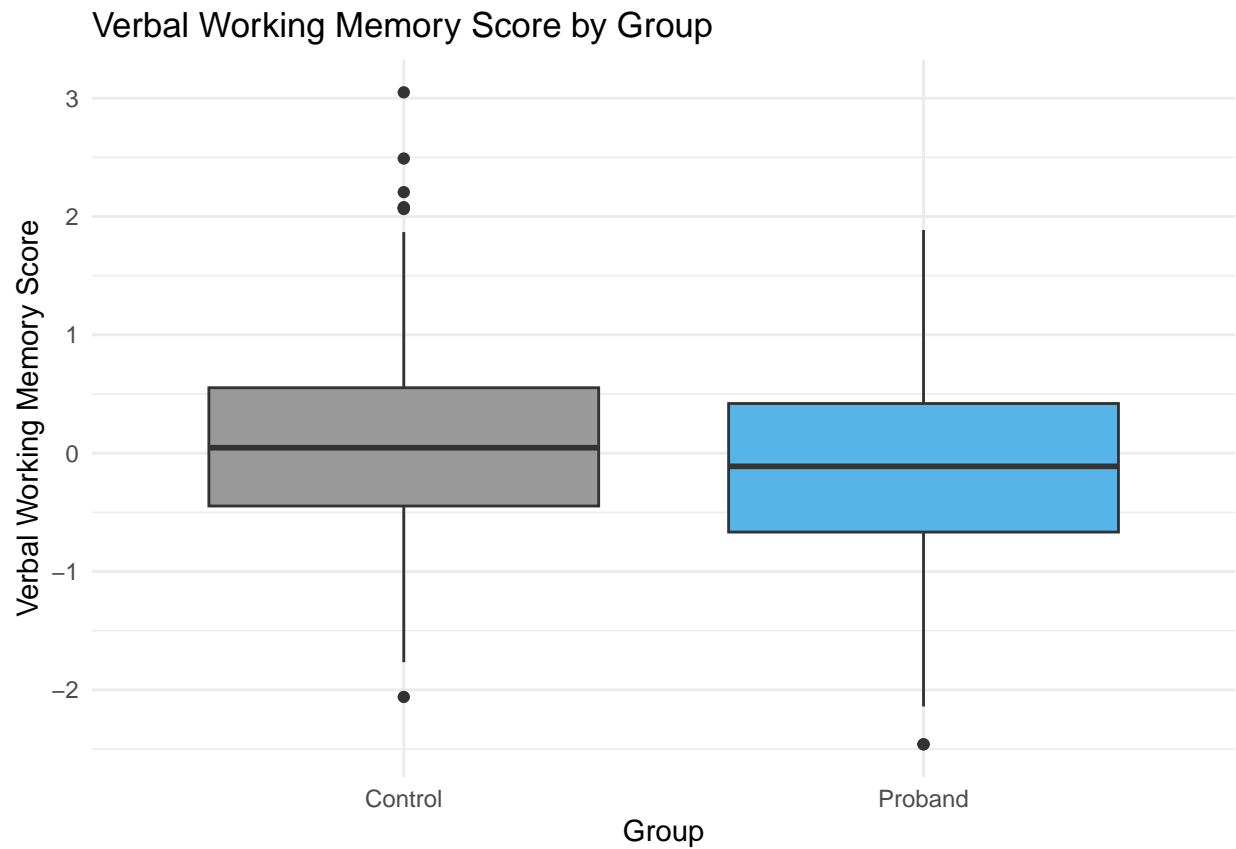
```
## Warning: Removed 1 row containing non-finite outside the scale range
## ('stat_boxplot()').
```



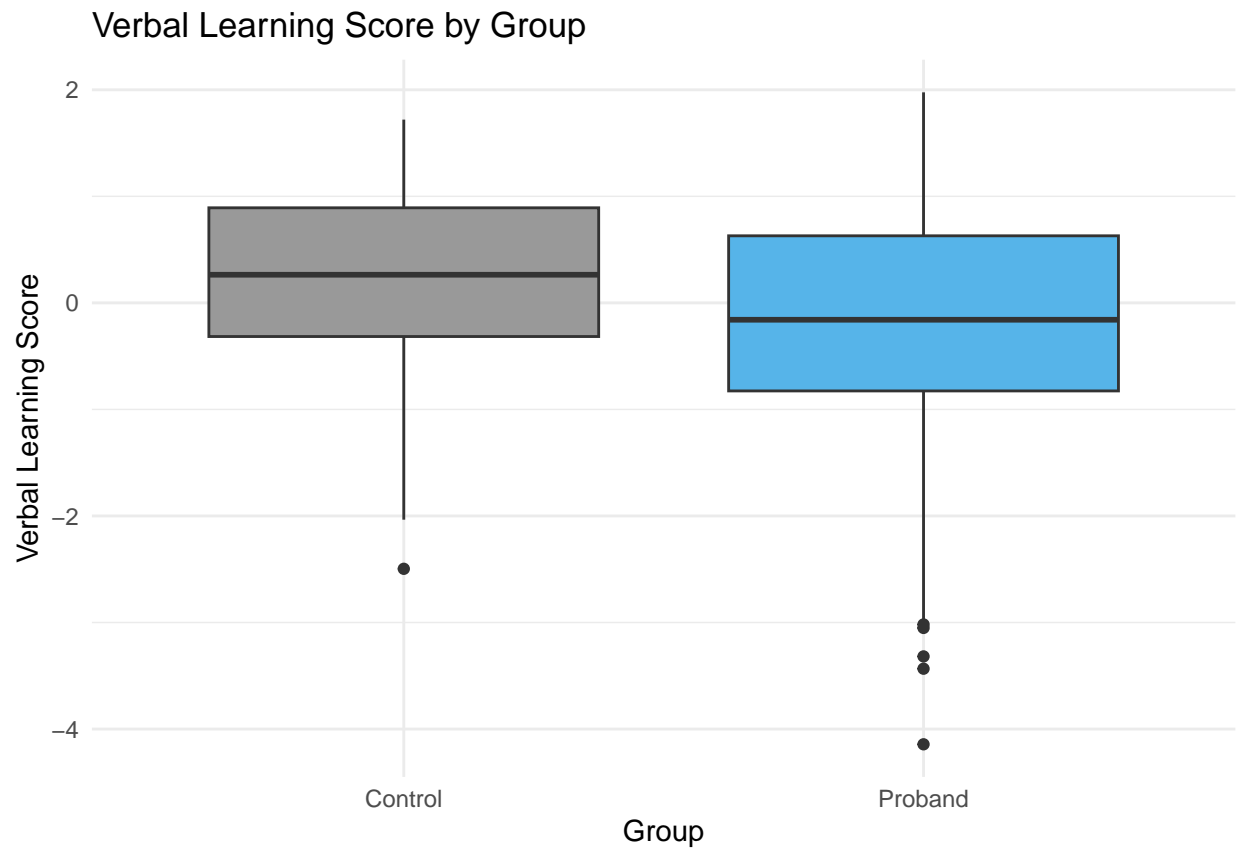
```
ggplot(pheno_cog, aes(x=factor(group), y = matrixreasoning_z)) +  
  geom_boxplot(fill = c("#999999", "#56B4E9")) +  
  labs(x = "Group", y = "Matrix Reasoning Score", title = "Matrix Reasoning Score by Group") +  
  theme_minimal()
```



```
ggplot(pheno_cog, aes(x=factor(group), y = vwm_composite)) +  
  geom_boxplot(fill = c("#999999", "#56B4E9")) +  
  labs(x = "Group", y = "Verbal Working Memory Score", title = "Verbal Working Memory Score by Group") +  
  theme_minimal()
```

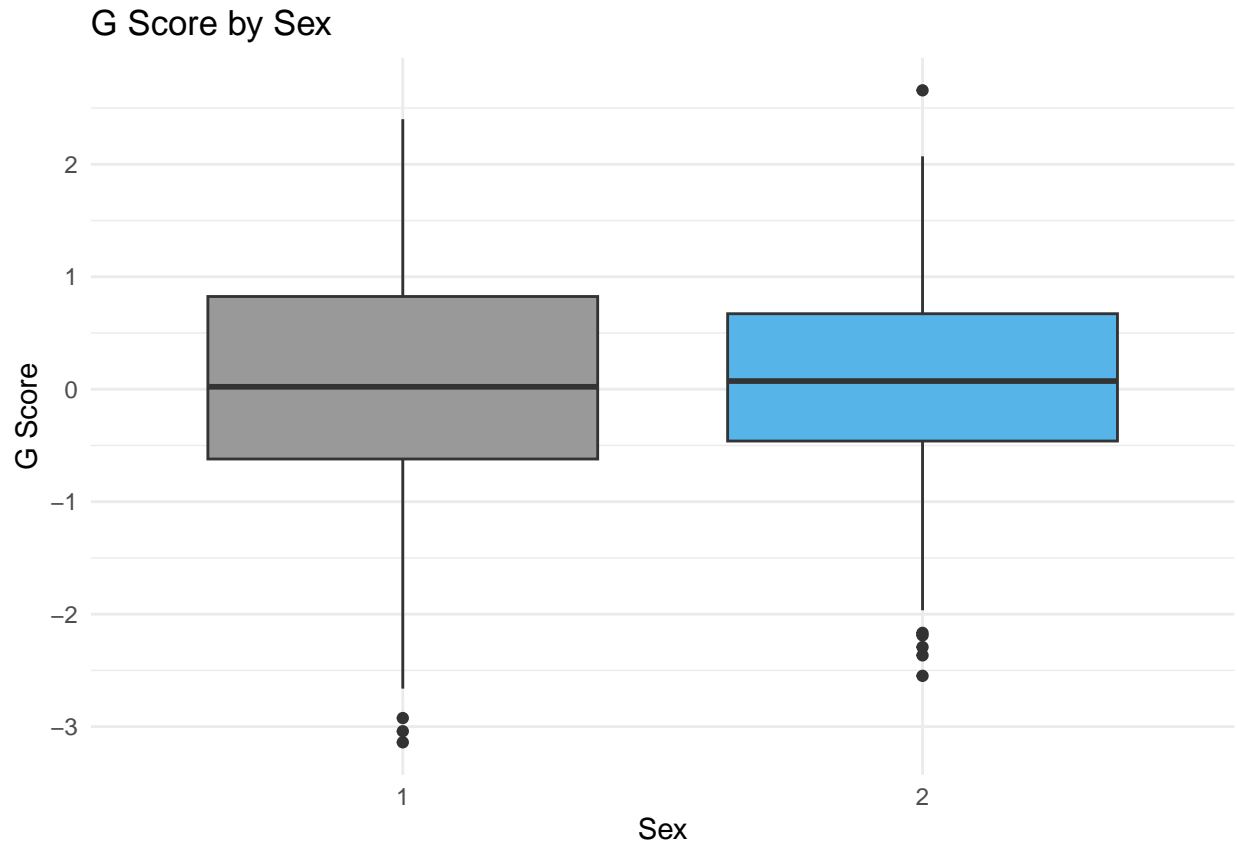


```
ggplot(pheno_cog, aes(x=factor(group), y = verbal_learning_z)) +  
  geom_boxplot(fill = c("#999999", "#56B4E9")) +  
  labs(x = "Group", y = "Verbal Learning Score", title = "Verbal Learning Score by Group") +  
  theme_minimal()
```



```
ggplot(pheno_cog, aes(x=factor(sex), y = g)) +  
  geom_boxplot(fill = c("#999999", "#56B4E9")) +  
  labs(x = "Sex", y = "G Score", title = "G Score by Sex") +  
  theme_minimal()
```

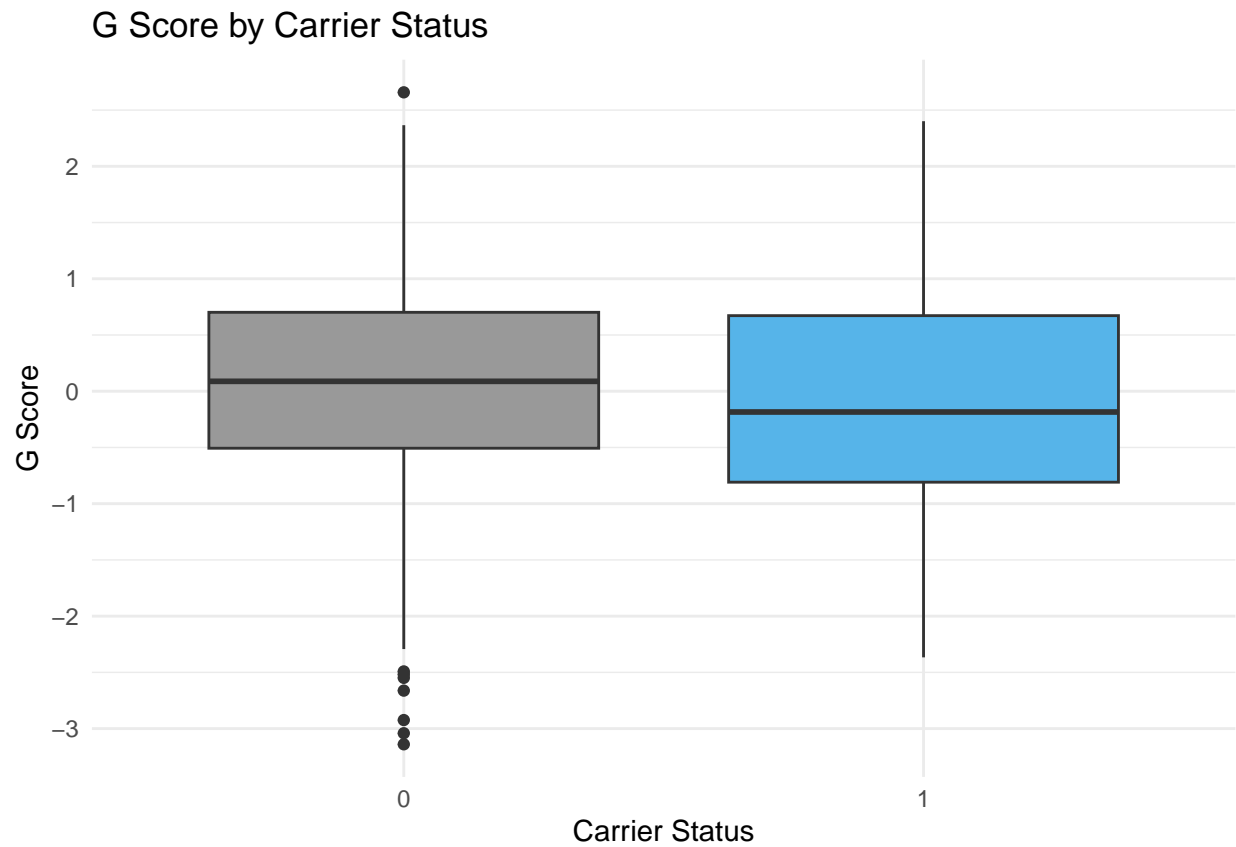
```
## Warning: Removed 46 rows containing non-finite outside the scale range  
## ('stat_boxplot()').
```



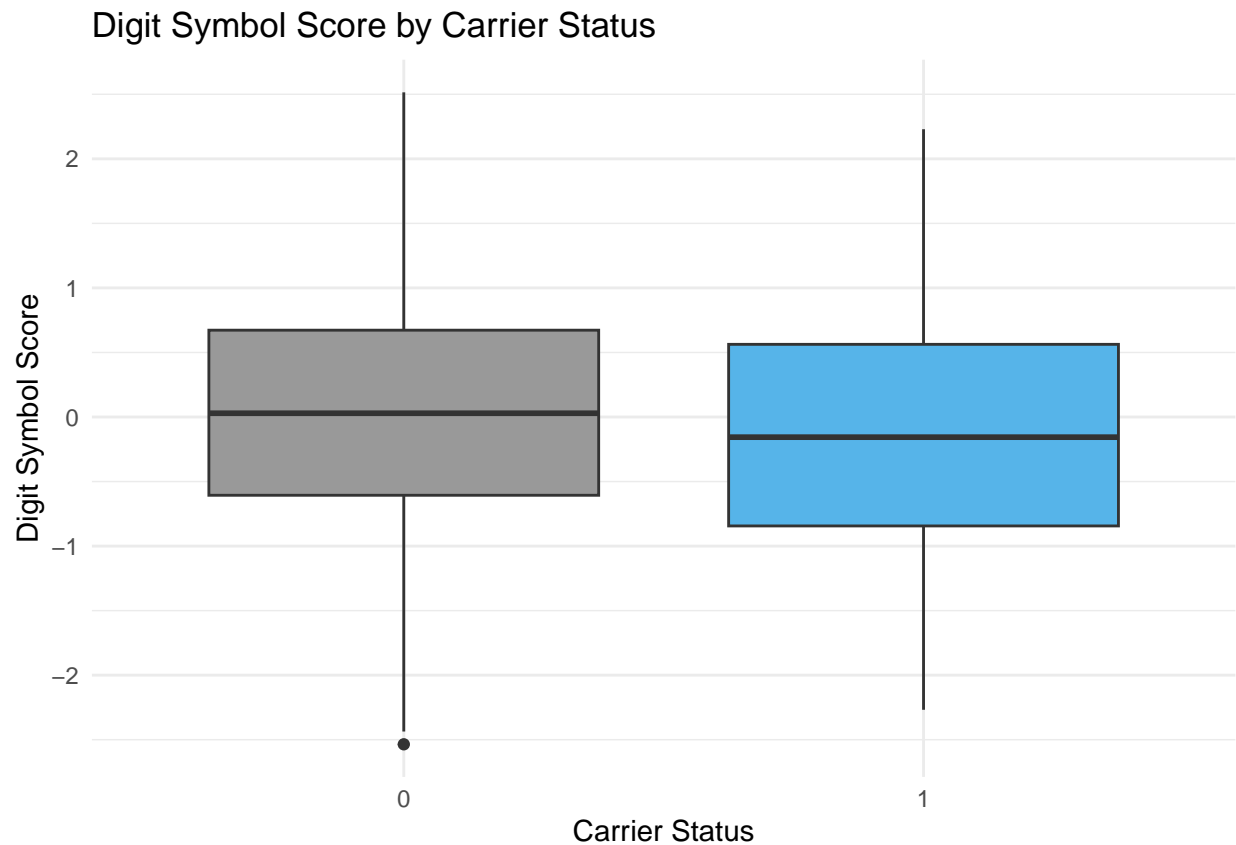
Exploratory Visualizations Comparing Cognitive Scores for Carrier Status

```
ggplot(carriers_combined, aes(x=factor(is_carrier), y = g)) +  
  geom_boxplot(fill = c("#999999", "#56B4E9")) +  
  labs(x = "Carrier Status", y = "G Score", title = "G Score by Carrier Status") +  
  theme_minimal()
```

```
## Warning: Removed 47 rows containing non-finite outside the scale range  
## ('stat_boxplot()').
```

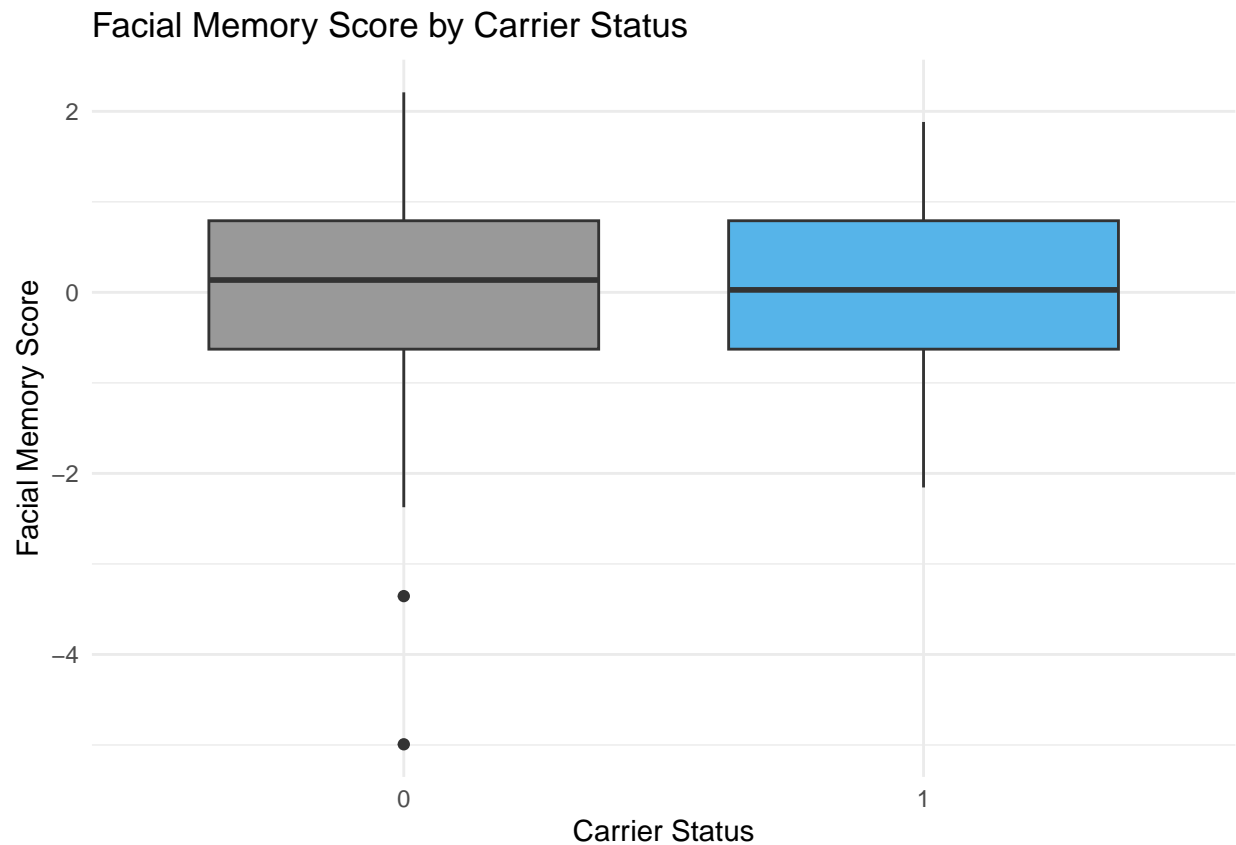


```
ggplot(carriers_combined, aes(x=factor(is_carrier), y = digit_symbol_composite)) +  
  geom_boxplot(fill = c("#999999", "#56B4E9")) +  
  labs(x = "Carrier Status", y = "Digit Symbol Score", title = "Digit Symbol Score by Carrier Status") +  
  theme_minimal()
```



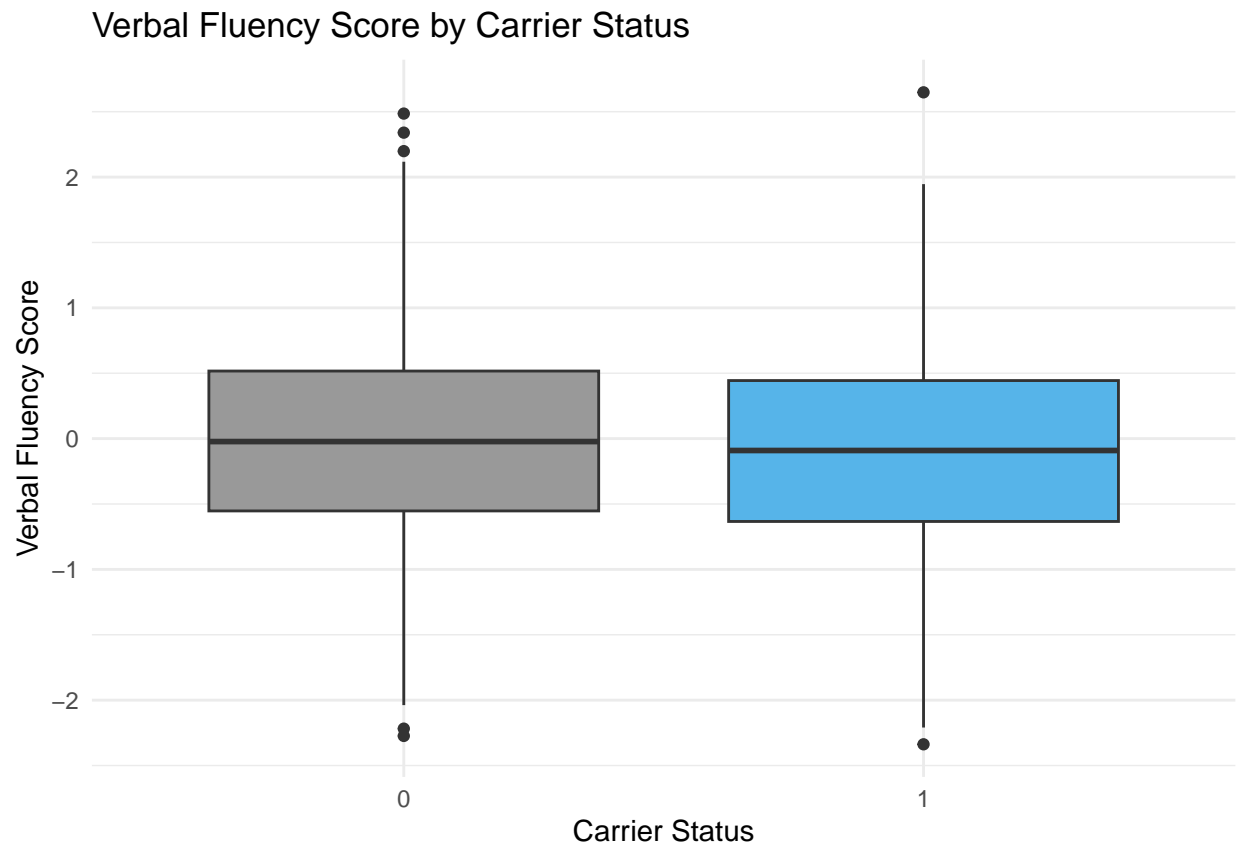
```
ggplot(carriers_combined, aes(x=factor(is_carrier), y = facialmemory_z)) +  
  geom_boxplot(fill = c("#999999", "#56B4E9")) +  
  labs(x = "Carrier Status", y = "Facial Memory Score", title = "Facial Memory Score by Carrier Status")  
  theme_minimal()
```

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

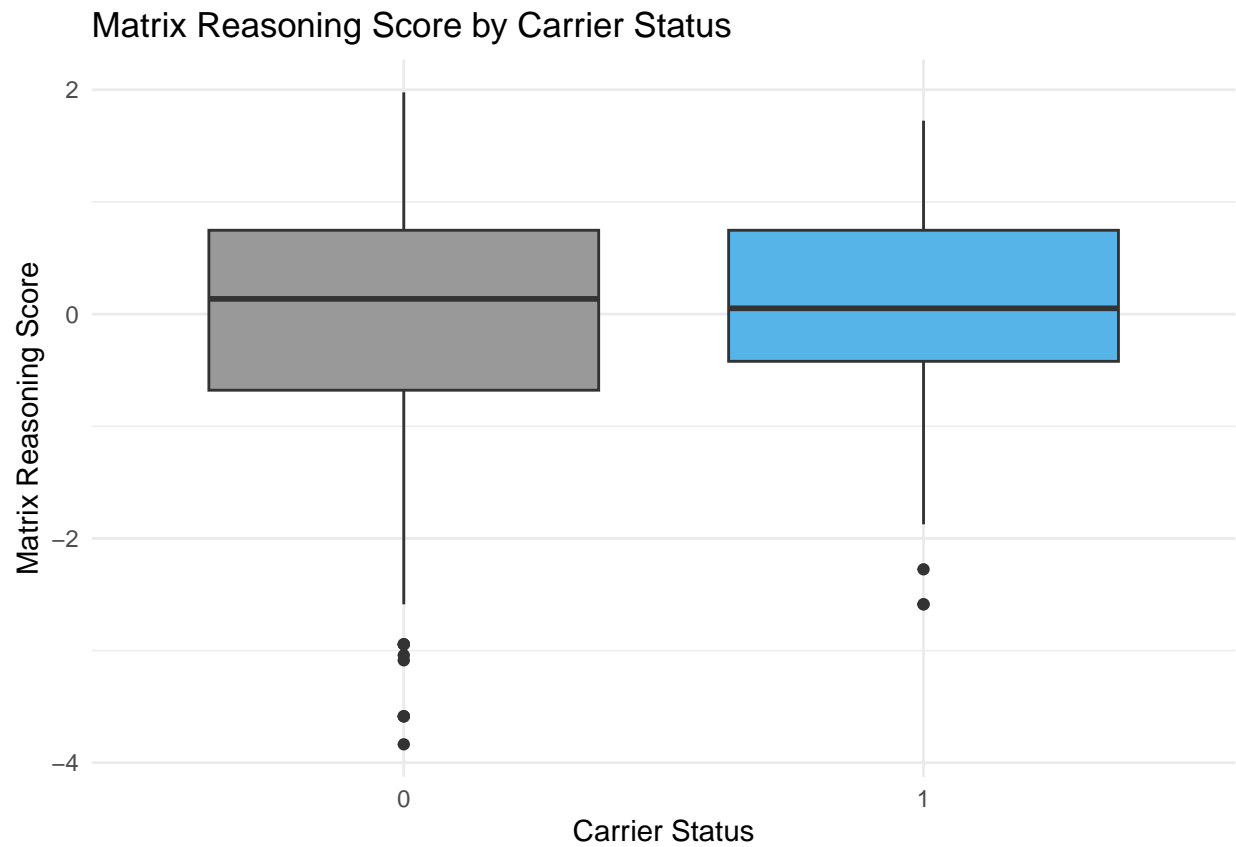



```
ggplot(carriers_combined, aes(x=factor(is_carrier), y = verbalfluency_composite)) +  
  geom_boxplot(fill = c("#999999", "#56B4E9")) +  
  labs(x = "Carrier Status", y = "Verbal Fluency Score", title = "Verbal Fluency Score by Carrier Status") +  
  theme_minimal()
```

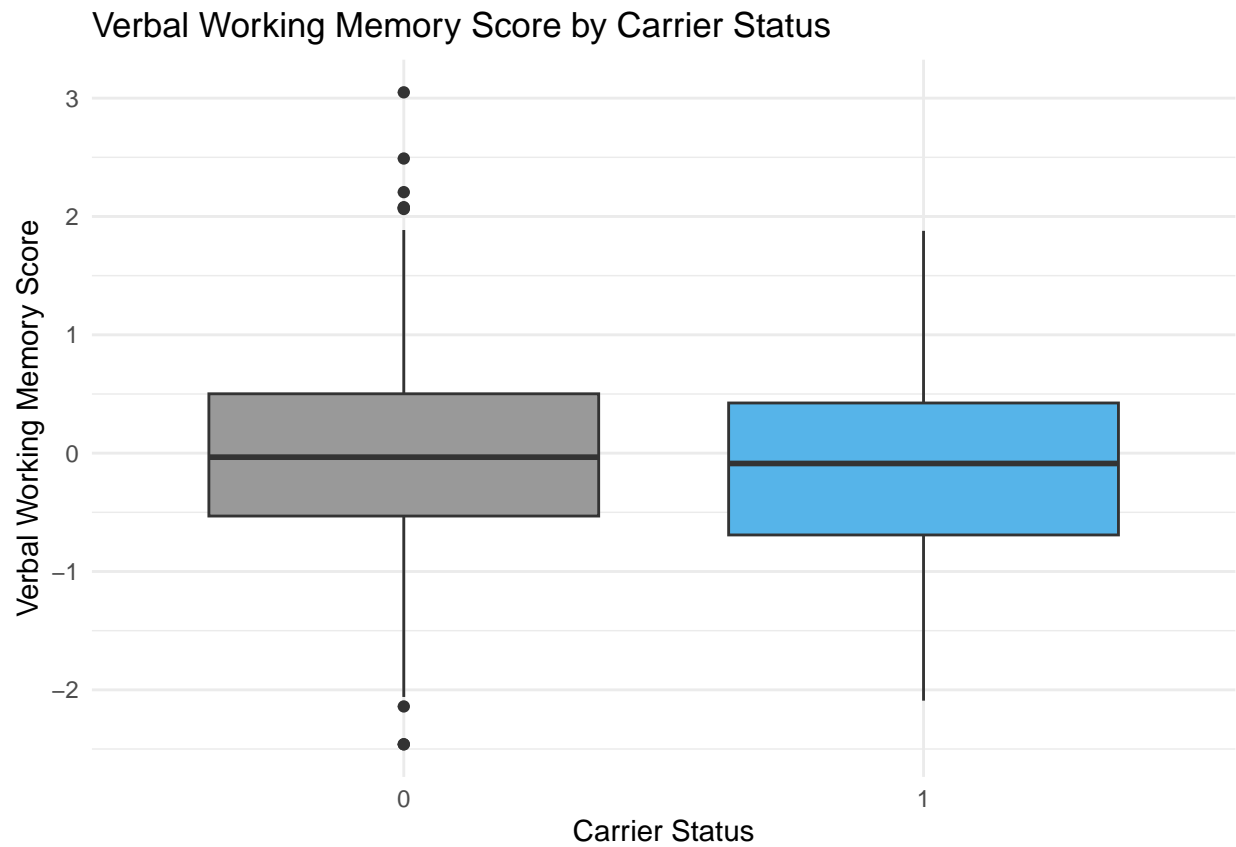
```
## Warning: Removed 1 row containing non-finite outside the scale range  
## ('stat_boxplot()').
```



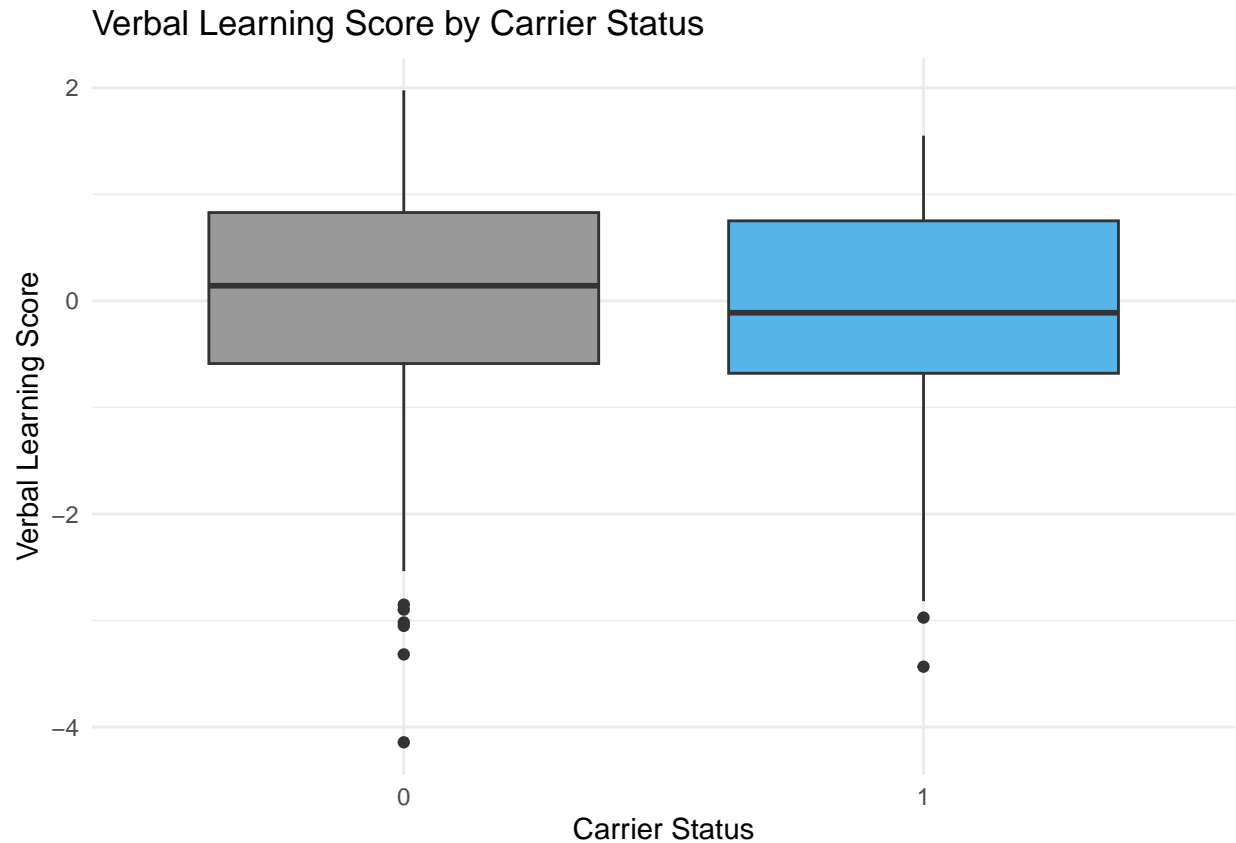
```
ggplot(carriers_combined, aes(x=factor(is_carrier), y = matrixreasoning_z)) +  
  geom_boxplot(fill = c("#999999", "#56B4E9")) +  
  labs(x = "Carrier Status", y = "Matrix Reasoning Score", title = "Matrix Reasoning Score by Carrier S  
  theme_minimal()
```



```
ggplot(carriers_combined, aes(x=factor(is_carrier), y = vwm_composite)) +  
  geom_boxplot(fill = c("#999999", "#56B4E9")) +  
  labs(x = "Carrier Status", y = "Verbal Working Memory Score", title = "Verbal Working Memory Score by  
  theme_minimal()
```



```
ggplot(carriers_combined, aes(x=factor(is_carrier), y = verbal_learning_z)) +  
  geom_boxplot(fill = c("#999999", "#56B4E9")) +  
  labs(x = "Carrier Status", y = "Verbal Learning Score", title = "Verbal Learning Score by Carrier Sta") +  
  theme_minimal()
```



```
# Table of the sample size distributions across variant type and gene
table(filtered_carriers$variant_type)
```

```
##
##          DEL          DUP    frameshift    missense splice acceptor
##          42          25           1          41           2
## stop-gained
##          1
```

```
table(filtered_carriers$gene)
```

```
##
## AKAP11  ATP9A  CACNA1B  CACNA1G  CAGNA1G  CDK13  CUL1  GRIN2A  HDAC9  HERC1
##      1      3      16      3      2      2      3      1      10      15
## JARID2  MAGI2   NBEA    NGLN2   NLGN2   PSMA3  SETD1  SP4     STAG1  TOP2B
##      4      4      16      2      2      1      1      7      9      3
##  TRIO   ZNF318
##      5      2
```

Variant type and Gene group analysis

```

# Group variant types by DUP + LOF, DEL, Missense
grouped_types <- carriers_combined %>%
  mutate(
    variant_group = case_when(
      variant_type %in% c("stop-gained", "frameshift", "splice acceptor", "DUP") ~ "LOF",
      variant_type %in% c("missense") ~ "Missense",
      variant_type %in% c("DEL") ~ "DEL",
      TRUE ~ "none"
    )
  )

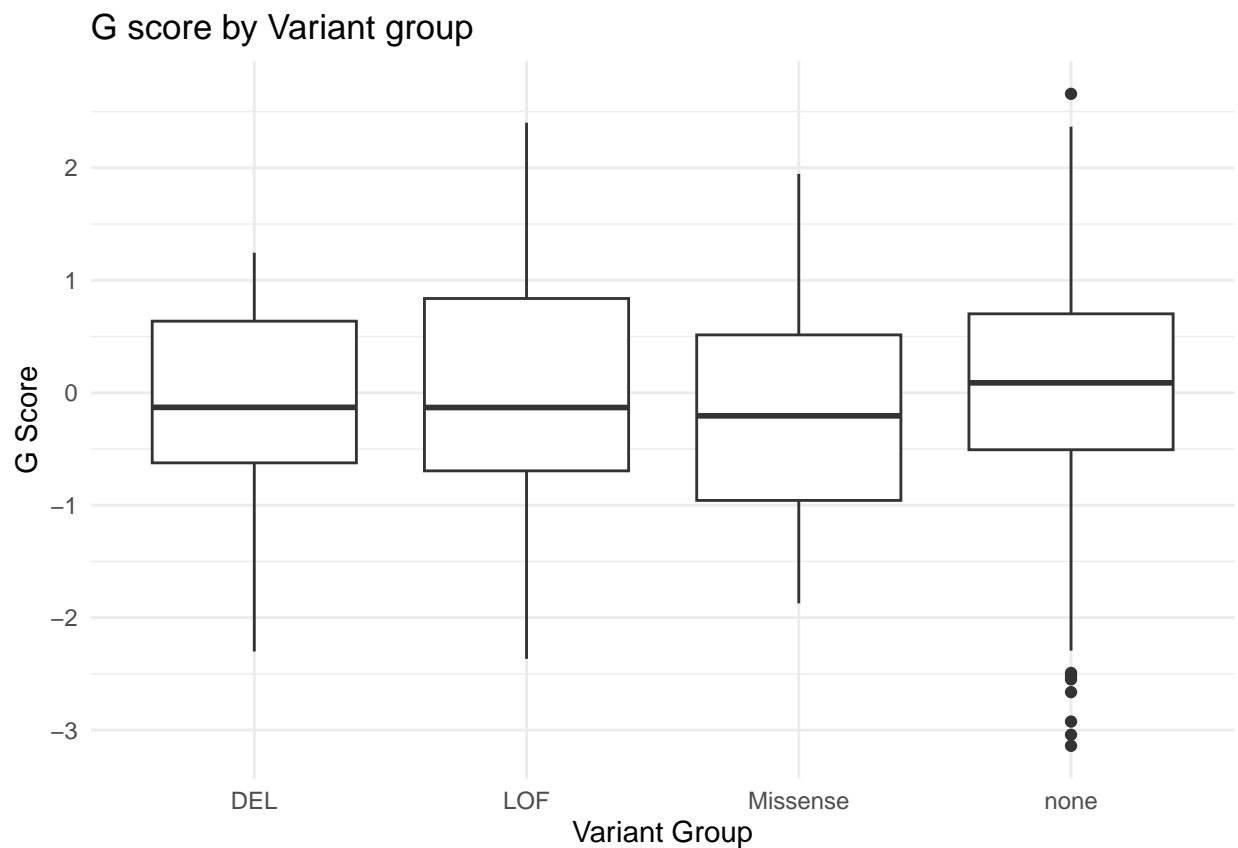
ggplot(grouped_types, aes(x=factor(variant_group), y = g)) +
  geom_boxplot( ) +
  labs(x = "Variant Group", y = "G Score", title = "G score by Variant group") +
  theme_minimal()

```

```

## Warning: Removed 47 rows containing non-finite outside the scale range
## ('stat_boxplot()').

```



```

# Table of the distribution of sample size in the variant groups
table(grouped_types$variant_group)

```

```

##
##      DEL      LOF Missense      none
##      42       29       41      436

```

```
# Group individual genes by biological function
```

```
grouped_types <- carriers_combined %>%
```

```
  mutate(
```

```
    gene_group = case_when(
```

```
      gene %in% c("CACNA1B", "CACNA1G", "CAGNA1G", "GRIN2A", "MAGI2", "NGLN2", "NLGN2", "SP4", "TRIO") ~
```

```
      gene %in% c("CUL1", "HERC1", "PSMA3") ~ "Protien Regulation",
```

```
      gene %in% c("CDK13", "NBEA", "STAG1", "AKAP11", "ATP9A") ~ "Cell Cycle",
```

```
      gene %in% c("SETD1", "HDAC9", "JARID2", "TOP2B", "ZNF318") ~ "Chromatin",
```

```
      TRUE ~ "Control"
```

```
    )
```

```
  )
```

```
ggplot(grouped_types, aes(x=factor(gene_group), y = g)) +
```

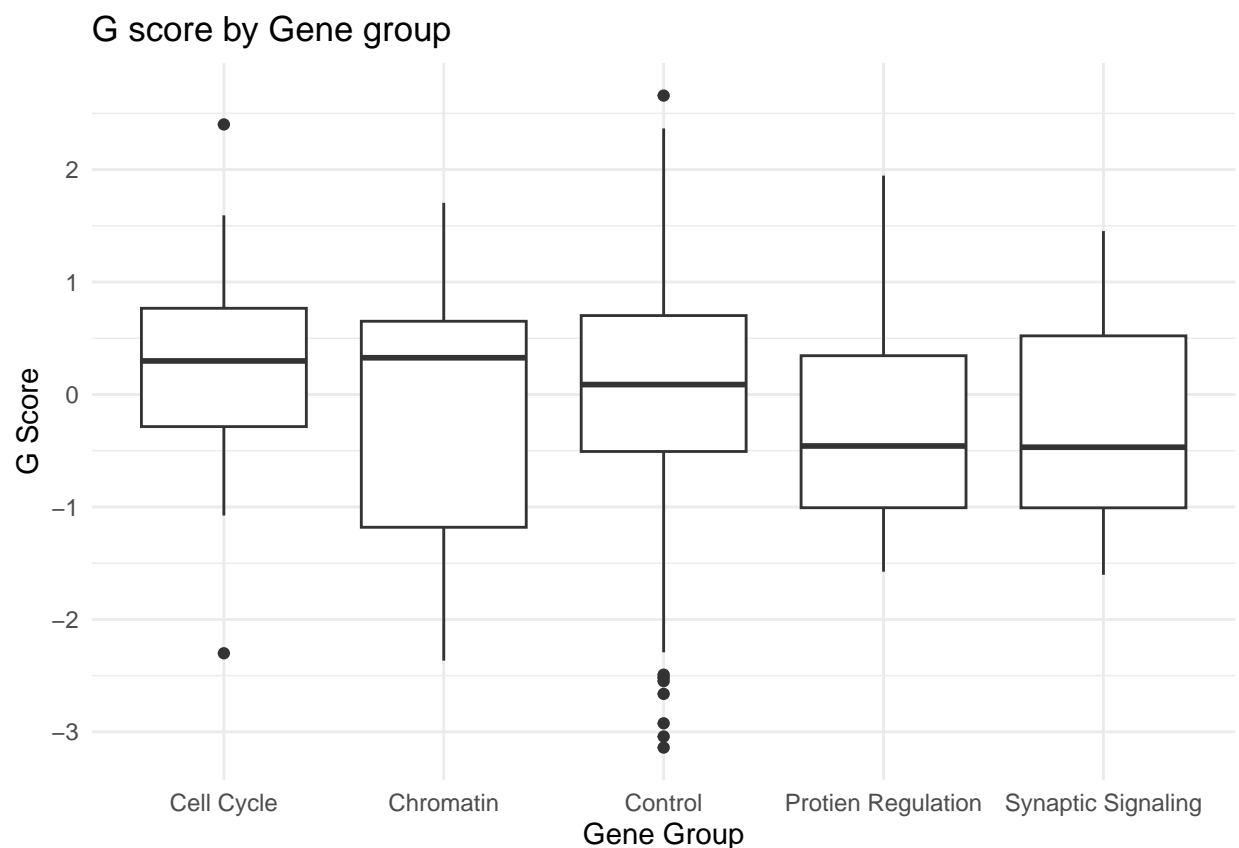
```
  geom_boxplot( ) +
```

```
  labs(x = "Gene Group", y = "G Score", title = "G score by Gene group") +
```

```
  theme_minimal()
```

```
## Warning: Removed 47 rows containing non-finite outside the scale range
```

```
## ('stat_boxplot()').
```



```
# Table of the distribution of sample size in the gene groups
```

```
table(grouped_types$gene_group)
```

```
##
```

```
##           Cell Cycle           Chromatin           Control Protien Regulation
##           31                   20                   436                   19
## Synaptic Signaling
##           42
```

Statistical Modeling

```
carriers_cog <- grouped_types %>%
  filter(is_carrier == 1)

view(carriers_cog)

# pairwise t test
pairwise.t.test(
  x = carriers_cog$g,
  g = carriers_cog$gene_group,
  p.adjust.method = "bonferroni"
)
```

```
##
## Pairwise comparisons using t tests with pooled SD
##
## data: carriers_cog$g and carriers_cog$gene_group
##
##           Cell Cycle Chromatin Protien Regulation
## Chromatin      1.00      -      -
## Protien Regulation 0.64      1.00      -
## Synaptic Signaling 0.30      1.00      1.00
##
## P value adjustment method: bonferroni
```

```
pairwise.t.test(
  x = carriers_cog$digitssymbol_composite,
  g = carriers_cog$gene_group,
  p.adjust.method = "bonferroni"
)
```

```
##
## Pairwise comparisons using t tests with pooled SD
##
## data: carriers_cog$digitssymbol_composite and carriers_cog$gene_group
##
##           Cell Cycle Chromatin Protien Regulation
## Chromatin      1      -      -
## Protien Regulation 1      1      -
## Synaptic Signaling 1      1      1
##
## P value adjustment method: bonferroni
```



```
model <- lm(digitsymbol_composite ~ group.x + age + sex + is_carrier, data = grouped_types)
summary(model)
```

```
##
## Call:
## lm(formula = digitsymbol_composite ~ group.x + age + sex + is_carrier,
##     data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.57035 -0.59209  0.02366  0.64312  2.29600
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.51256    0.24372  -6.206 1.08e-09 ***
## group.xProband -0.42412    0.07797  -5.440 8.09e-08 ***
## age           0.11985    0.01400   8.561 < 2e-16 ***
## sex          -0.06603    0.07727  -0.855  0.393
## is_carrier    -0.14209    0.09566  -1.485  0.138
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9002 on 543 degrees of freedom
## Multiple R-squared:  0.1495, Adjusted R-squared:  0.1433
## F-statistic: 23.87 on 4 and 543 DF,  p-value: < 2.2e-16
```

```
model <- lm(facialmemory_z ~ group.x + age + sex + is_carrier, data = grouped_types)
summary(model)
```

```
##
## Call:
## lm(formula = facialmemory_z ~ group.x + age + sex + is_carrier,
##     data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.7678 -0.6041  0.0546  0.6858  2.0670
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.709451    0.254342  -6.721 4.58e-11 ***
## group.xProband -0.391472    0.081452  -4.806 2.00e-06 ***
## age           0.100629    0.014615   6.885 1.60e-11 ***
## sex           0.233632    0.080695   2.895  0.00394 **
## is_carrier     0.005556    0.099848   0.056  0.95565
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9394 on 542 degrees of freedom
## (1 observation deleted due to missingness)
## Multiple R-squared:  0.1179, Adjusted R-squared:  0.1114
## F-statistic: 18.11 on 4 and 542 DF,  p-value: 5.689e-14
```

```
model <- lm(verbalfluency_composite ~ group.x + age + sex + is_carrier, data = grouped_types)
summary(model)
```

```
##
## Call:
## lm(formula = verbalfluency_composite ~ group.x + age + sex +
##     is_carrier, data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.20311 -0.47024 -0.00782  0.48068  2.33733
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.21295    0.20786 -10.647 < 2e-16 ***
## group.xProband -0.26988    0.06657  -4.054 5.76e-05 ***
## age           0.13259    0.01194  11.101 < 2e-16 ***
## sex           0.21386    0.06595   3.243 0.00126 **
## is_carrier    -0.10843    0.08160  -1.329 0.18448
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7677 on 542 degrees of freedom
## (1 observation deleted due to missingness)
## Multiple R-squared:  0.2098, Adjusted R-squared:  0.204
## F-statistic: 35.98 on 4 and 542 DF,  p-value: < 2.2e-16
```

```
model <- lm(verbal_learning_z ~ group.x + age + sex + is_carrier, data = grouped_types)
summary(model)
```

```
##
## Call:
## lm(formula = verbal_learning_z ~ group.x + age + sex + is_carrier,
##     data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.9339 -0.5854  0.0645  0.7171  2.0493
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.96327    0.25695  -3.749 0.000197 ***
## group.xProband -0.52061    0.08220  -6.333 5.04e-10 ***
## age           0.06377    0.01476   4.320 1.85e-05 ***
## sex           0.19046    0.08146   2.338 0.019751 *
## is_carrier    -0.25392    0.10086  -2.518 0.012100 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9491 on 543 degrees of freedom
## Multiple R-squared:  0.1027, Adjusted R-squared:  0.0961
## F-statistic: 15.54 on 4 and 543 DF,  p-value: 4.812e-12
```

```
model <- lm(matrixreasoning_z ~ group.x + age + sex + is_carrier, data = grouped_types)
summary(model)
```

```
##
## Call:
## lm(formula = matrixreasoning_z ~ group.x + age + sex + is_carrier,
##     data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.7790 -0.5227  0.0902  0.6654  1.9989
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.935849   0.248432  -7.792 3.37e-14 ***
## group.xProband -0.337395   0.079477  -4.245 2.57e-05 ***
## age           0.136646   0.014271   9.575 < 2e-16 ***
## sex           0.003224   0.078762   0.041  0.967
## is_carrier    -0.003209   0.097512  -0.033  0.974
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9176 on 543 degrees of freedom
## Multiple R-squared:  0.158, Adjusted R-squared:  0.1518
## F-statistic: 25.47 on 4 and 543 DF, p-value: < 2.2e-16
```

```
model <- lm(vwm_composite ~ group.x + age + sex + is_carrier, data = grouped_types)
summary(model)
```

```
##
## Call:
## lm(formula = vwm_composite ~ group.x + age + sex + is_carrier,
##     data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.44535 -0.48877 -0.03714  0.49475  2.88863
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.62428   0.20748  -7.829 2.60e-14 ***
## group.xProband -0.31547   0.06638  -4.753 2.57e-06 ***
## age           0.10278   0.01192   8.624 < 2e-16 ***
## sex           0.14021   0.06578   2.131  0.0335 *
## is_carrier    -0.10992   0.08144  -1.350  0.1777
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7664 on 543 degrees of freedom
## Multiple R-squared:  0.1484, Adjusted R-squared:  0.1421
## F-statistic: 23.65 on 4 and 543 DF, p-value: < 2.2e-16
```

```
model <- lm(g ~ group.x + age + sex + is_carrier, data = grouped_types)
summary(model)
```

```
##
## Call:
## lm(formula = g ~ group.x + age + sex + is_carrier, data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.15691 -0.52611  0.04435  0.59914  2.14119
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.05468    0.25059  -8.199 2.08e-15 ***
## group.xProband -0.50281    0.08032  -6.260 8.32e-10 ***
## age            0.14771    0.01441  10.253 < 2e-16 ***
## sex            0.06359    0.07943   0.801  0.4238
## is_carrier     -0.18197    0.09895  -1.839  0.0665 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8861 on 496 degrees of freedom
## (47 observations deleted due to missingness)
## Multiple R-squared:  0.2119, Adjusted R-squared:  0.2055
## F-statistic: 33.34 on 4 and 496 DF, p-value: < 2.2e-16
```

```
model <- lm(g ~ group.x * is_carrier + age + sex, data = grouped_types)
summary(model)
```

```
##
## Call:
## lm(formula = g ~ group.x * is_carrier + age + sex, data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.1762 -0.5417  0.0380  0.5962  2.1609
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.08278    0.25221  -8.258 1.36e-15 ***
## group.xProband -0.46461    0.08917  -5.211 2.77e-07 ***
## is_carrier     -0.10156    0.12821  -0.792  0.429
## age            0.14809    0.01441  10.275 < 2e-16 ***
## sex            0.06621    0.07948   0.833  0.405
## group.xProband:is_carrier -0.19909    0.20185  -0.986  0.324
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8861 on 495 degrees of freedom
## (47 observations deleted due to missingness)
## Multiple R-squared:  0.2134, Adjusted R-squared:  0.2055
## F-statistic: 26.86 on 5 and 495 DF, p-value: < 2.2e-16
```

```
model <- lm(g ~ group.x + age + sex + is_carrier, data = grouped_types)
summary(model)
```

```
##
## Call:
## lm(formula = g ~ group.x + age + sex + is_carrier, data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.15691 -0.52611  0.04435  0.59914  2.14119
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.05468    0.25059  -8.199 2.08e-15 ***
## group.xProband -0.50281    0.08032  -6.260 8.32e-10 ***
## age            0.14771    0.01441  10.253 < 2e-16 ***
## sex            0.06359    0.07943   0.801  0.4238
## is_carrier    -0.18197    0.09895  -1.839  0.0665 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8861 on 496 degrees of freedom
## (47 observations deleted due to missingness)
## Multiple R-squared:  0.2119, Adjusted R-squared:  0.2055
## F-statistic: 33.34 on 4 and 496 DF,  p-value: < 2.2e-16
```

```
model <- lm(g ~ group.x, data = grouped_types)
summary(model)
```

```
##
## Call:
## lm(formula = g ~ group.x, data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.94896 -0.61332  0.01368  0.65486  2.46223
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.19576    0.05930   3.301  0.00103 **
## group.xProband -0.38655    0.08753  -4.416 1.23e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9763 on 499 degrees of freedom
## (47 observations deleted due to missingness)
## Multiple R-squared:  0.03762, Adjusted R-squared:  0.03569
## F-statistic: 19.5 on 1 and 499 DF,  p-value: 1.231e-05
```

```
model <- lm(g ~ group.x + age + sex, data = grouped_types)
summary(model)
```

```
##
## Call:
## lm(formula = g ~ group.x + age + sex, data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.12609 -0.52915  0.02373  0.57410  2.18287
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.09744    0.25011   -8.386 5.22e-16 ***
## group.xProband -0.49208    0.08029   -6.128 1.81e-09 ***
## age           0.14777    0.01444   10.233 < 2e-16 ***
## sex           0.06370    0.07962    0.800  0.424
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8882 on 497 degrees of freedom
## (47 observations deleted due to missingness)
## Multiple R-squared:  0.2065, Adjusted R-squared:  0.2017
## F-statistic: 43.12 on 3 and 497 DF, p-value: < 2.2e-16
```

```
model <- lm(g ~ is_carrier + age + sex + group.x, data = grouped_types)
summary(model)
```

```
##
## Call:
## lm(formula = g ~ is_carrier + age + sex + group.x, data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.15691 -0.52611  0.04435  0.59914  2.14119
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.05468    0.25059   -8.199 2.08e-15 ***
## is_carrier    -0.18197    0.09895   -1.839  0.0665 .
## age           0.14771    0.01441   10.253 < 2e-16 ***
## sex           0.06359    0.07943    0.801  0.4238
## group.xProband -0.50281    0.08032   -6.260 8.32e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8861 on 496 degrees of freedom
## (47 observations deleted due to missingness)
## Multiple R-squared:  0.2119, Adjusted R-squared:  0.2055
## F-statistic: 33.34 on 4 and 496 DF, p-value: < 2.2e-16
```

```
model <- lm(digitsymbol_composite ~ is_carrier + age + sex, data = grouped_types)
summary(model)
```

```
##
## Call:
```

```
## lm(formula = digitsymbol_composite ~ is_carrier + age + sex,
##     data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.78527 -0.55237  0.02812  0.68976  2.48793
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.55759    0.24990  -6.233 9.19e-10 ***
## is_carrier   -0.10165    0.09785  -1.039   0.299
## age          0.11121    0.01427   7.793 3.33e-14 ***
## sex          -0.08333    0.07921  -1.052   0.293
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9236 on 544 degrees of freedom
## Multiple R-squared:  0.1032, Adjusted R-squared:  0.09825
## F-statistic: 20.87 on 3 and 544 DF, p-value: 8.278e-13
```

```
model <- lm(g ~ gene_group + age + sex, data = grouped_types)
summary(model)
```

```
##
## Call:
## lm(formula = g ~ gene_group + age + sex, data = grouped_types)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.4089 -0.5716  0.0676  0.6398  2.3710
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -1.89465    0.31025  -6.107 2.06e-09 ***
## gene_groupChromatin    -0.40542    0.27554  -1.471   0.1418
## gene_groupControl     -0.21740    0.18255  -1.191   0.2343
## gene_groupProtien Regulation -0.57022    0.28407  -2.007   0.0453 *
## gene_groupSynaptic Signaling -0.48395    0.23106  -2.095   0.0367 *
## age              0.13723    0.01480   9.271 < 2e-16 ***
## sex              0.04839    0.08253   0.586   0.5579
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.917 on 494 degrees of freedom
## (47 observations deleted due to missingness)
## Multiple R-squared:  0.1594, Adjusted R-squared:  0.1492
## F-statistic: 15.62 on 6 and 494 DF, p-value: < 2.2e-16
```

```
## strong positive effect - older individuals score higher
```

```
## carriers score \-0.14 points lower on g but this is not significant
```

```
## Model is overall statistically significant - age adds meaningful predictive power
```

```
## sex has no significant effect on g
```

Visualizations

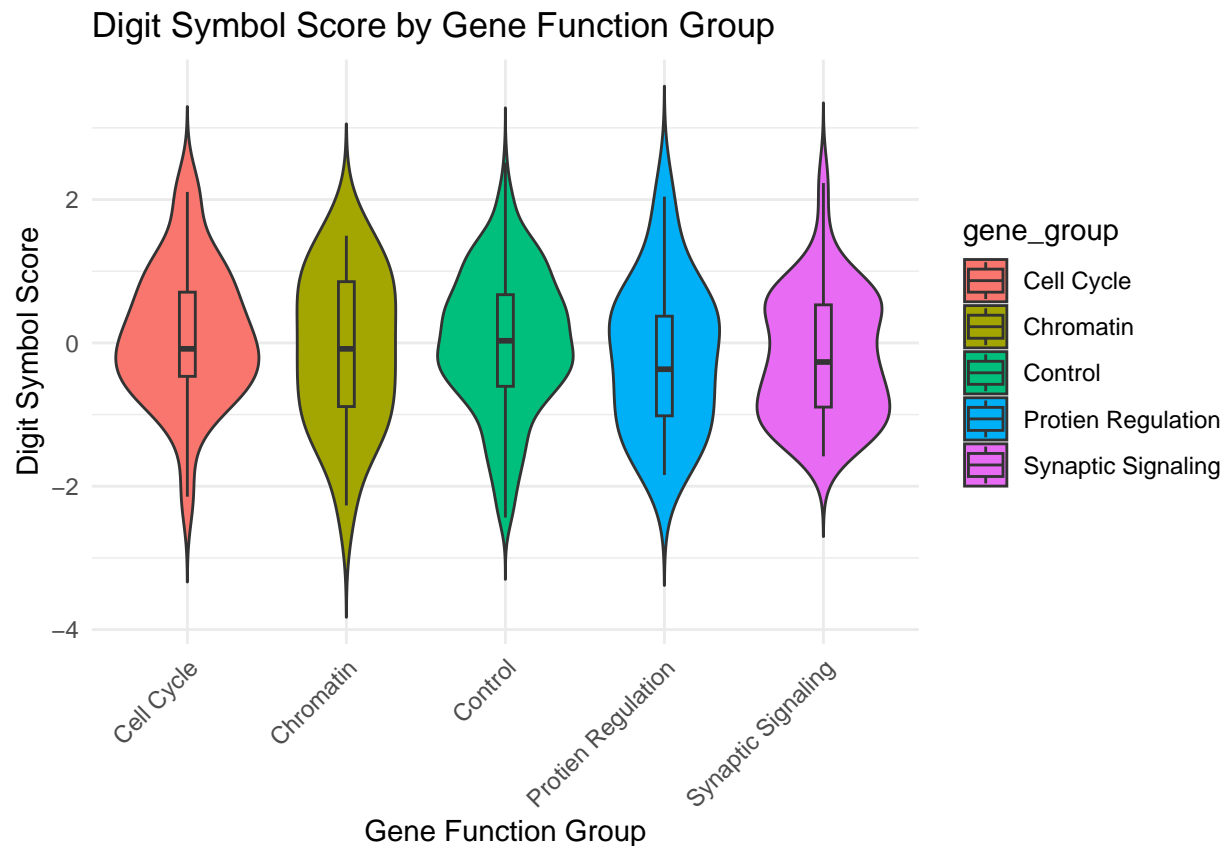
```
# Violin plot
ggplot(grouped_types, aes(x= gene_group, y = g, fill = gene_group)) +
  geom_violin(trim = FALSE) +
  geom_boxplot(width = 0.1, outlier.shape = NA) +
  labs(title = "General Cognition Score by Gene Group",
       x = "Gene Function Group",
       y = "G Score") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```
## Warning: Removed 47 rows containing non-finite outside the scale range
## ('stat_ydensity()').
```

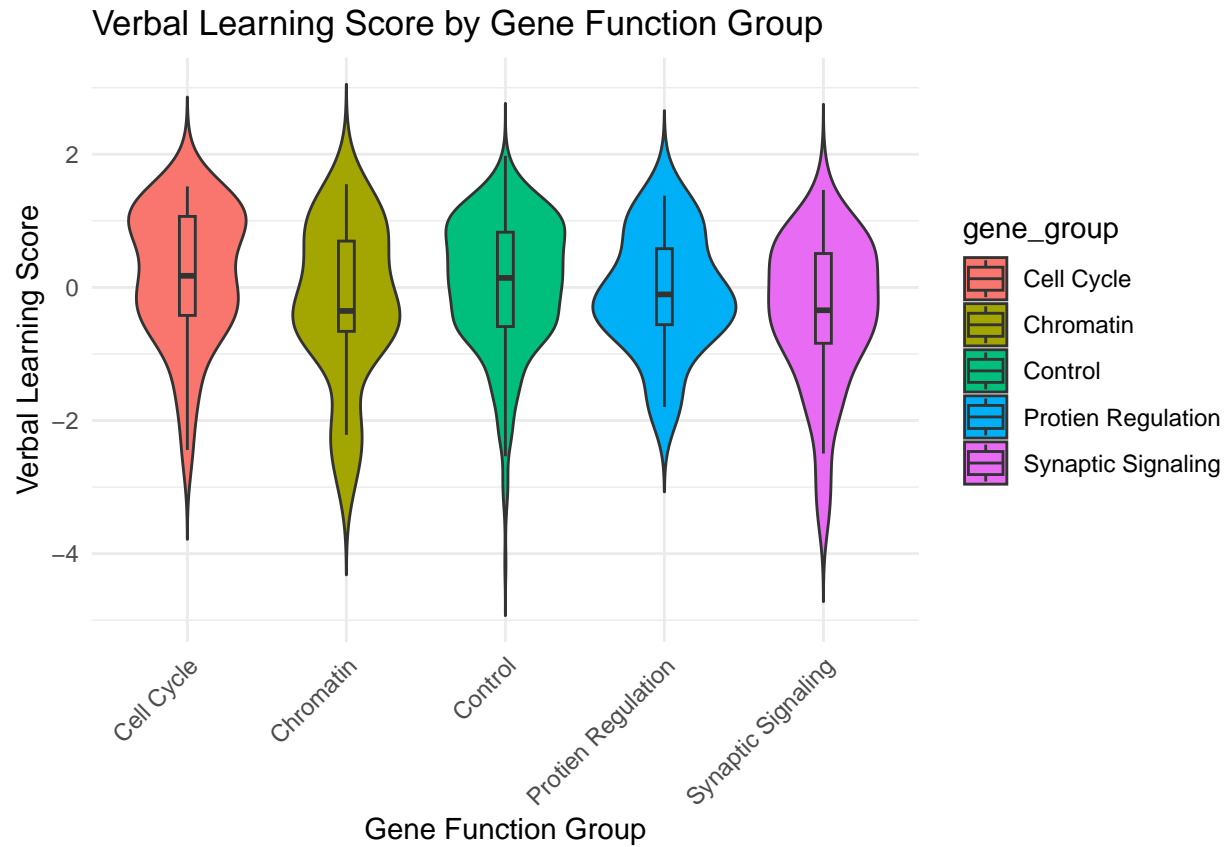
```
## Warning: Removed 47 rows containing non-finite outside the scale range
## ('stat_boxplot()').
```



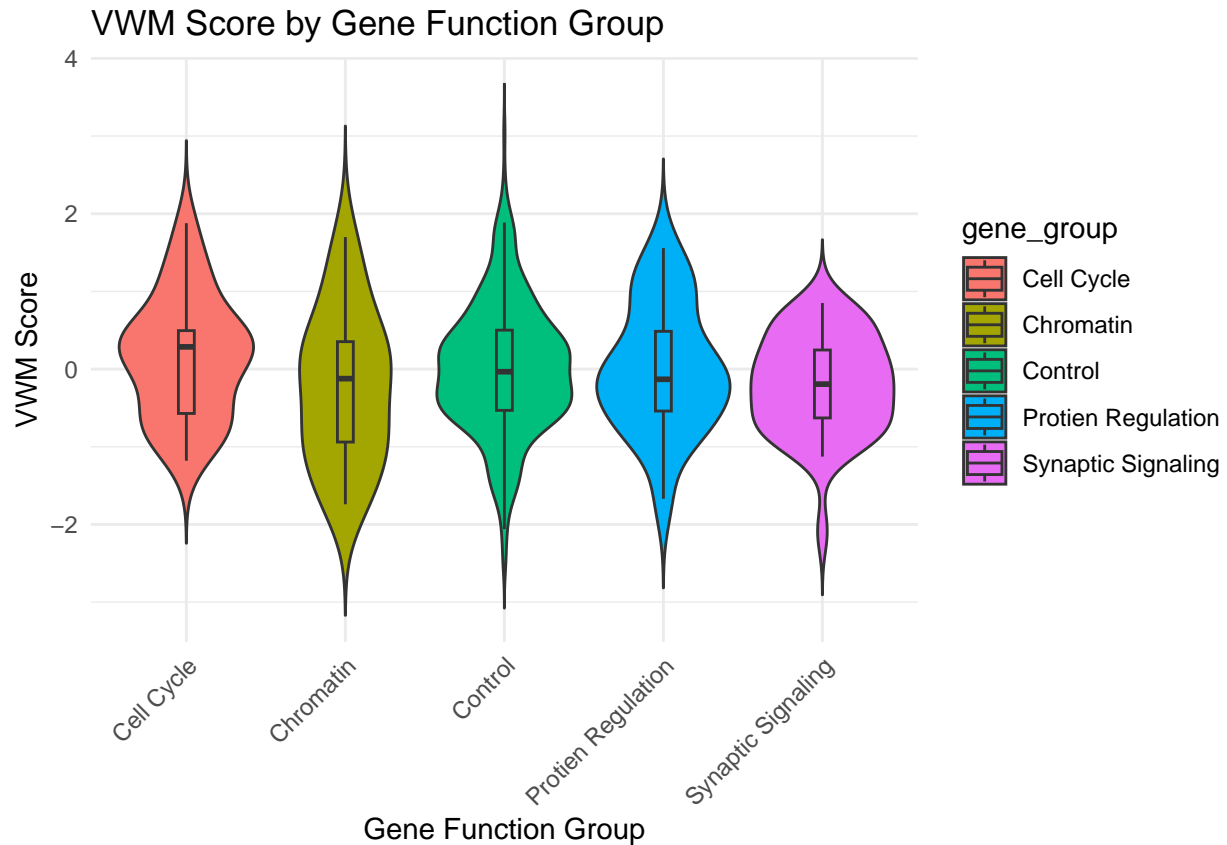

```
ggplot(grouped_types, aes(x= gene_group, y = digitsymbol_composite, fill = gene_group)) +
  geom_violin(trim = FALSE) +
  geom_boxplot(width = 0.1, outlier.shape = NA) +
  labs(title = "Digit Symbol Score by Gene Function Group",
       x = "Gene Function Group",
       y = "Digit Symbol Score") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



```
ggplot(grouped_types, aes(x= gene_group, y = verbal_learning_z, fill = gene_group)) +
  geom_violin(trim = FALSE) +
  geom_boxplot(width = 0.1, outlier.shape = NA) +
  labs(title = "Verbal Learning Score by Gene Function Group",
       x = "Gene Function Group",
       y = "Verbal Learning Score") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



```
ggplot(grouped_types, aes(x= gene_group, y = vwm_composite, fill = gene_group)) +
  geom_violin(trim = FALSE) +
  geom_boxplot(width = 0.1, outlier.shape = NA) +
  labs(title = "VWM Score by Gene Function Group",
       x = "Gene Function Group",
       y = "VWM Score") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



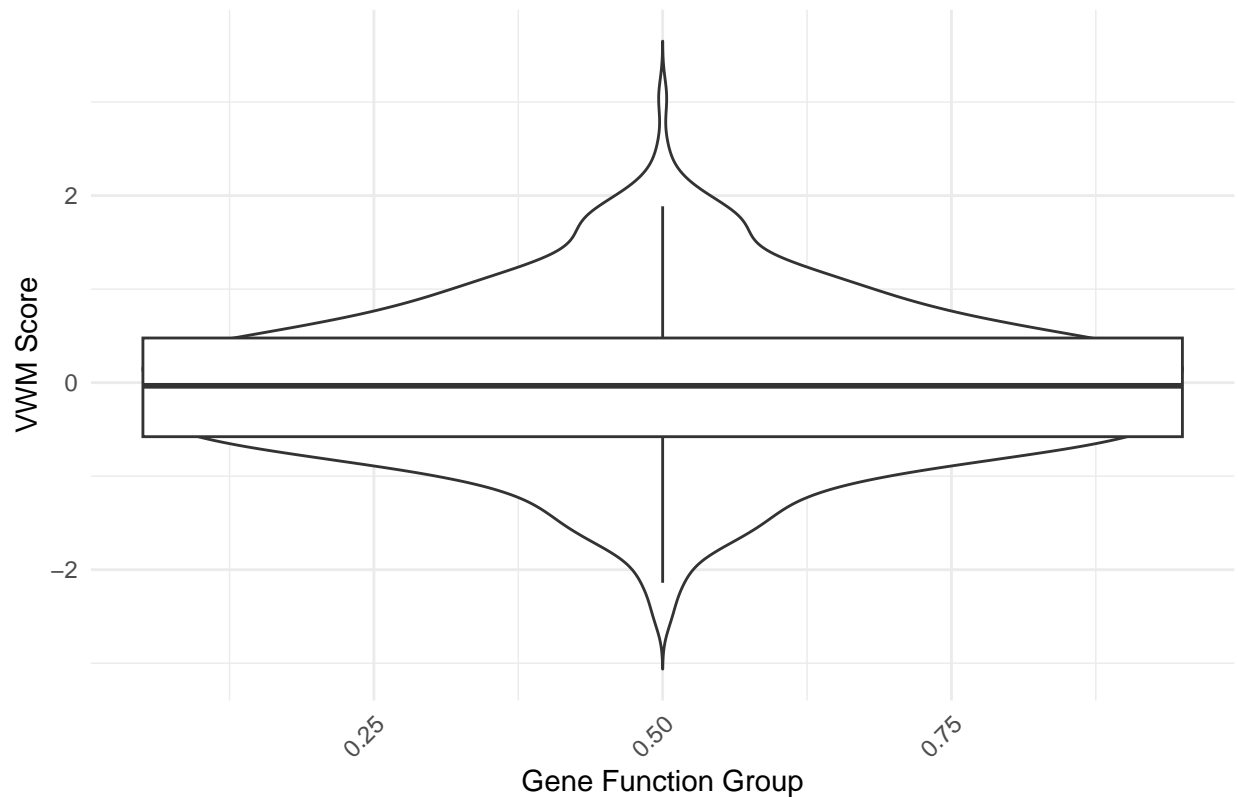
```
ggplot(grouped_types, aes(x= is_carrier, y = vwm_composite, fill = is_carrier)) +
  geom_violin(trim = FALSE) +
  geom_boxplot(width = .1, outlier.shape = NA) +
  labs(title = "VWM Score by Gene Function Group",
       x = "Gene Function Group",
       y = "VWM Score") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```
## Warning: The following aesthetics were dropped during statistical transformation: fill.
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a 'group' aesthetic or to convert a numerical
##   variable into a factor?
```

```
## Warning: Continuous x aesthetic
## i did you forget 'aes(group = ...)'?
```

```
## Warning: The following aesthetics were dropped during statistical transformation: fill.
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a 'group' aesthetic or to convert a numerical
##   variable into a factor?
```

VWM Score by Gene Function Group



Stratified Visualizations

```
# grouped bar chart for group by cog domains
```

```
grouped_data_phenotype <- grouped_types %>%
```

```
  group_by(phenotype) %>%
```

```
  summarise(
```

```
    Processing_Speed = mean(digitsymbol_composite, na.rm = TRUE),
```

```
    Facial_Memory = mean(facialmemory_z, na.rm = TRUE),
```

```
    Verbal_Learning = mean(verbal_learning_z, na.rm = TRUE),
```

```
    Matrix_Reasoning = mean(matrixreasoning_z, na.rm = TRUE),
```

```
    Verbal_Memory = mean(vwm_composite, na.rm = TRUE),
```

```
    Verbal_Fluency = mean(verbalfluency_composite, na.rm = TRUE),
```

```
    General_Cognition = mean(g, na.rm = TRUE),
```

```
)
```

```
long_data_phenotype <- grouped_data_phenotype %>%
```

```
  pivot_longer(cols = c(Processing_Speed, Facial_Memory, Verbal_Learning, Matrix_Reasoning, Verbal_Memory, Verbal_Fluency, General_Cognition),
```

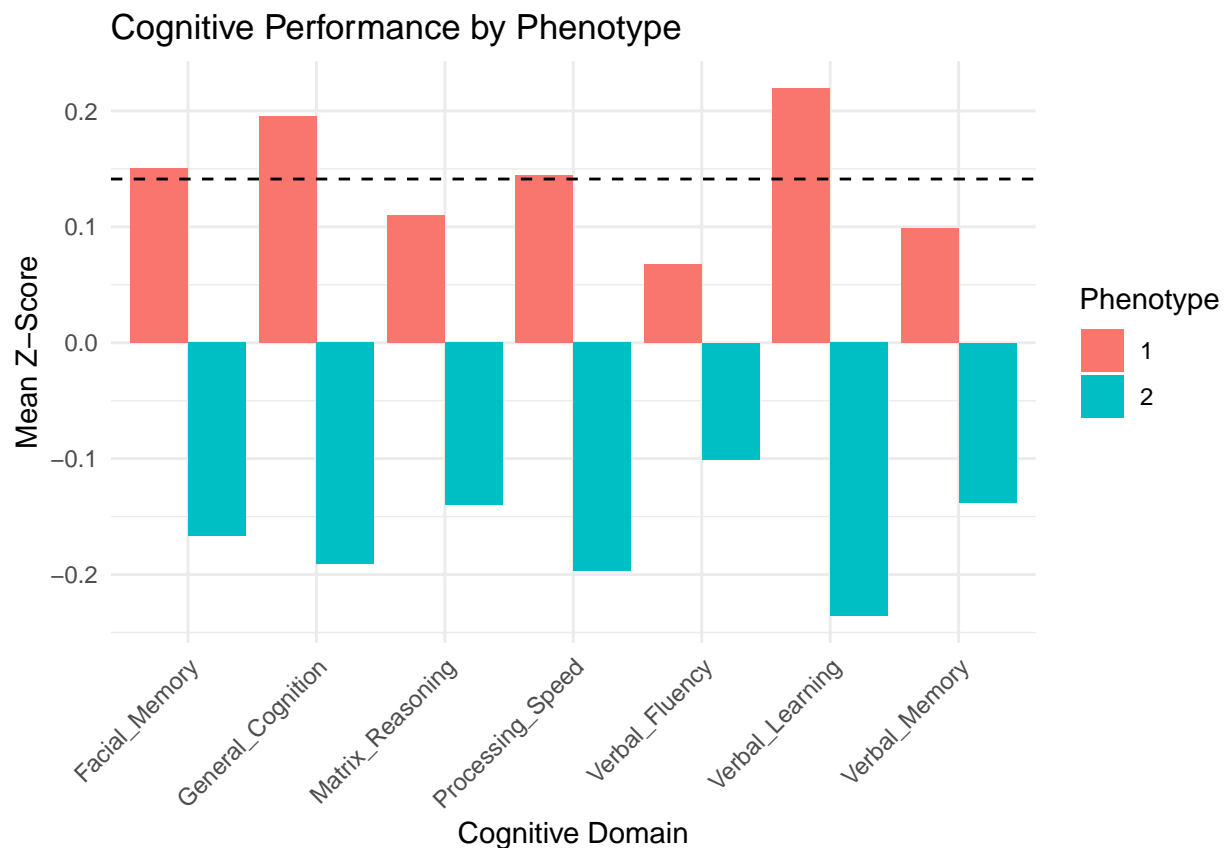
```
    names_to = "domain", values_to = "score")
```

```
control_mean <- mean(long_data_phenotype$score[long_data_phenotype$phenotype == "1"])
```

```
print(control_mean)
```

```
## [1] 0.141239
```

```
ggplot(long_data_phenotype, aes(x = domain, y = score, fill = factor(phenotype))) +  
  geom_bar(stat = "identity", position = "dodge") +  
  labs(title = "Cognitive Performance by Phenotype", x = "Cognitive Domain", y = "Mean Z-Score",  
  geom_hline(yintercept = control_mean, linetype = "dashed", color = "black") +  
  theme_minimal() +  
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



```
# grouped bar chart for carrier status by cog domains
```

```
grouped_data <- grouped_types %>%  
  group_by(is_carrier) %>%  
  summarise(  
    Processing_Speed = mean(digitsymbol_composite, na.rm = TRUE),  
    Facial_Memory = mean(facialmemory_z, na.rm = TRUE),  
    Verbal_Learning = mean(verbal_learning_z, na.rm = TRUE),  
    Matrix_Reasoning = mean(matrixreasoning_z, na.rm = TRUE),  
    Verbal_Memory = mean(vwm_composite, na.rm = TRUE),  
    Verbal_Fluency = mean(verbalfluency_composite, na.rm = TRUE),  
    General_Cognition = mean(g, na.rm = TRUE),  
  )  
  
long_data <- grouped_data %>%  
  pivot_longer(cols = c(Processing_Speed, Facial_Memory, Verbal_Learning, Matrix_Reasoning, Verbal_Memory,
```

```

names_to = "domain", values_to = "score") %>%
mutate(domain = gsub("_", " ", domain))

non_carrier_mean <- mean(long_data$score[long_data$is_carrier == "0"])
print(non_carrier_mean)

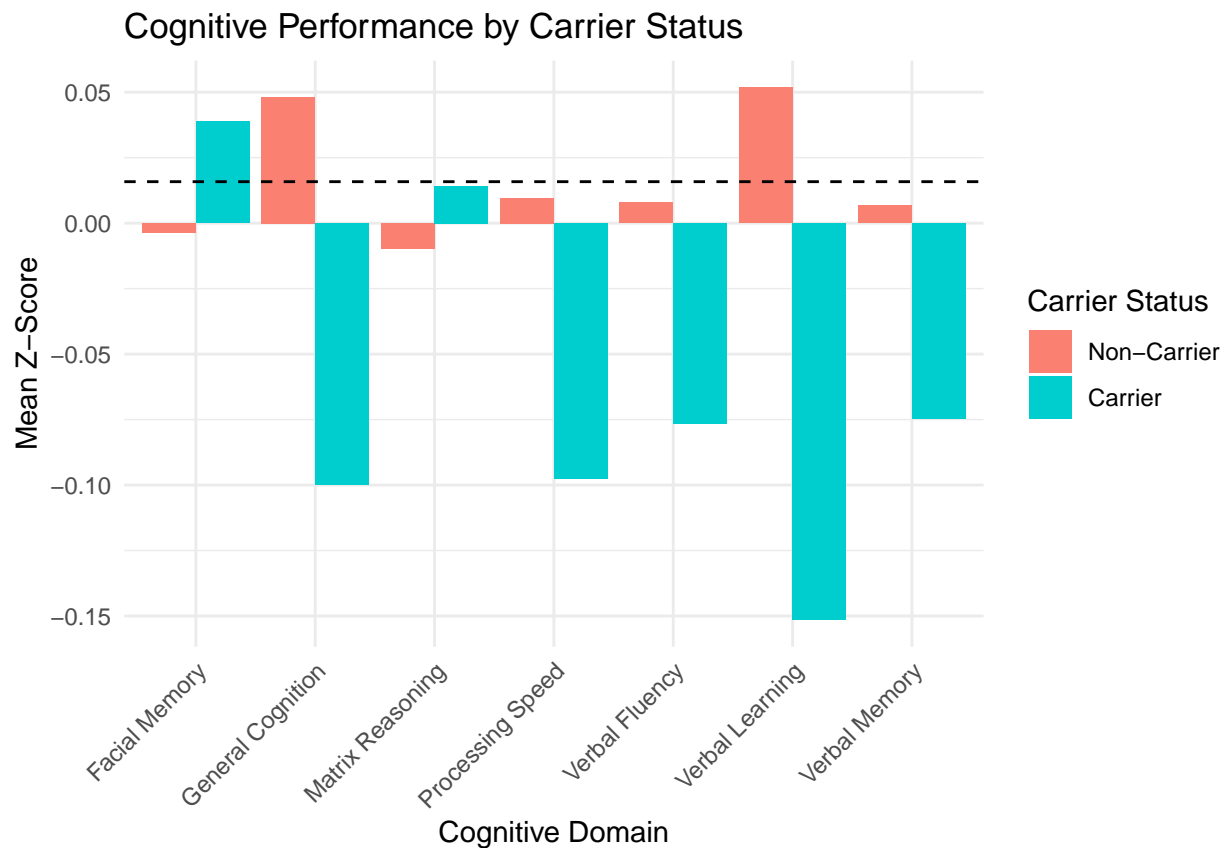
```

```
## [1] 0.01584164
```

```

ggplot(long_data, aes(x = domain, y = score, fill = factor(is_carrier))) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Cognitive Performance by Carrier Status", x = "Cognitive Domain", y = "Mean Z-Score") +
  scale_fill_manual(values = c("0" = "salmon", "1" = "cyan3"),
                    labels = c("Non-Carrier", "Carrier")) +
  geom_hline(yintercept = non_carrier_mean, linetype = "dashed", color = "black") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

```



```

# group genetic data
grouped_gene_data <- grouped_types %>%
  group_by(gene_group) %>%
  summarise(
    Processing_Speed = mean(digitsymbol_composite, na.rm = TRUE),
    Facial_Memory = mean(facialmemory_z, na.rm = TRUE),
    Verbal_Learning = mean(verbal_learning_z, na.rm = TRUE),

```

```

    Matrix_Reasoning = mean(matrixreasoning_z, na.rm = TRUE),
    Verbal_Memory = mean(vwm_composite, na.rm = TRUE),
    Verbal_Fluency = mean(verbalfluency_composite, na.rm = TRUE),
    General_Cognition = mean(g, na.rm = TRUE),
  )

rownames(grouped_gene_data) <- grouped_gene_data$gene_group

## Warning: Setting row names on a tibble is deprecated.

grouped_gene_data <- grouped_gene_data %>%
  select(-gene_group)

rownames(grouped_gene_data) <- c("Cell Cycle", "Chromatin", "Control", "Protein Regulation", "Synaptic S

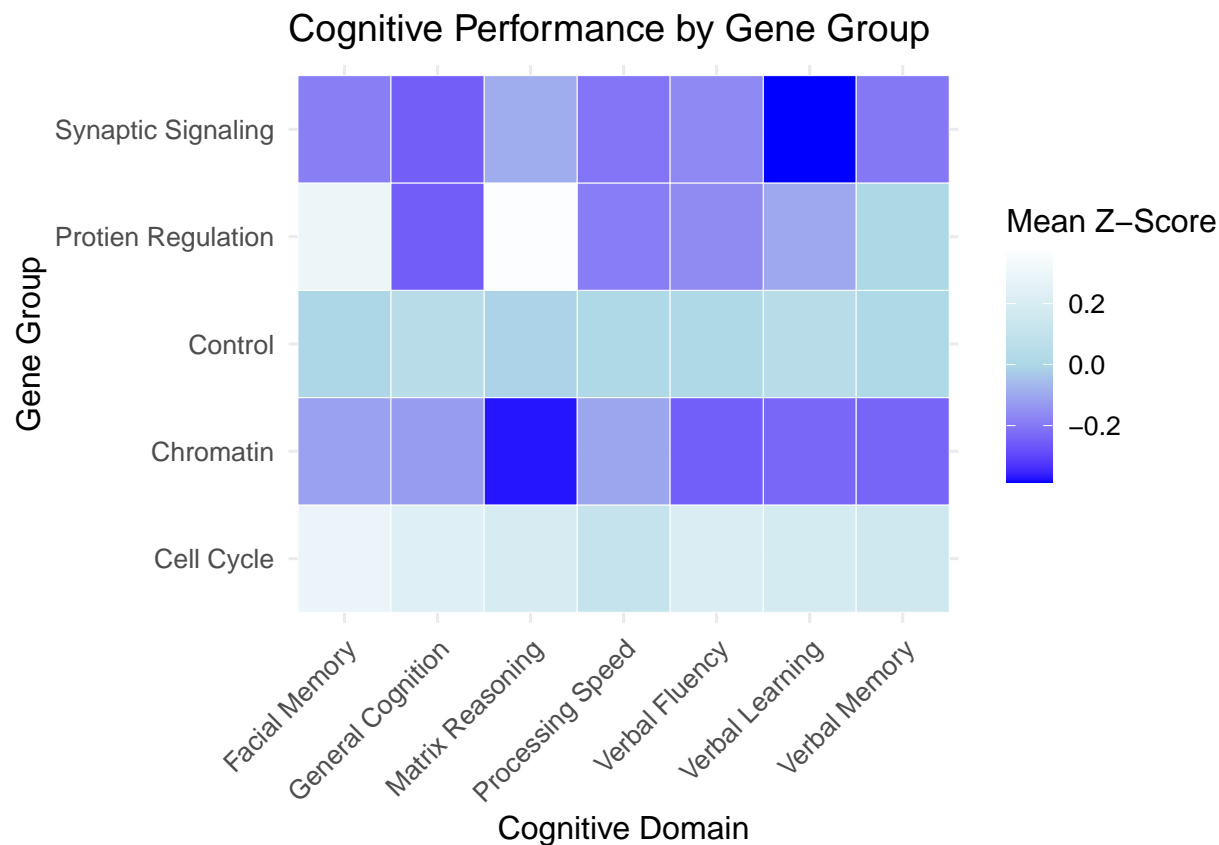
## Warning: Setting row names on a tibble is deprecated.

# heat map for cog domain and z score by gene group
grouped_gene_data2 <- grouped_types %>%
  group_by(gene_group) %>%
  summarise(
    Processing_Speed = mean(digitsymbol_composite, na.rm = TRUE),
    Facial_Memory = mean(facialmemory_z, na.rm = TRUE),
    Verbal_Learning = mean(verbal_learning_z, na.rm = TRUE),
    Matrix_Reasoning = mean(matrixreasoning_z, na.rm = TRUE),
    Verbal_Memory = mean(vwm_composite, na.rm = TRUE),
    Verbal_Fluency = mean(verbalfluency_composite, na.rm = TRUE),
    General_Cognition = mean(g, na.rm = TRUE),
  )

long_heatmap_data <- grouped_gene_data2 %>%
  pivot_longer(cols = -gene_group, names_to = "CognitiveDomain", values_to = "Z_Score") %>%
  mutate(CognitiveDomain = gsub("_", " ", CognitiveDomain))

ggplot(long_heatmap_data, aes(x = CognitiveDomain, y = gene_group, fill = Z_Score)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "blue", mid = "lightblue", high = "white", midpoint = 0) +
  labs(title = "Cognitive Performance by Gene Group",
       x = "Cognitive Domain",
       y = "Gene Group",
       fill = "Mean Z-Score") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title = element_text(face = "plain", size = 14),
    axis.text.x = element_text(angle = 45, , hjust = 1),
    axis.title.x = element_text(size = 12),
    axis.title.y = element_text(size = 12),
    legend.title = element_text(size = 12)
  )

```



```
# radar chart for cog domain and z score by carrier status
# install.packages("fmsb")
# library(fmsb)
#
# radar_data <- grouped_data %>%
#   select(-is_carrier)
#
# rownames(radar_data) <- c("Non-Carrier", "Carrier")
#
# radar_data <- rbind(
#   rep(.2, ncol(radar_data)),
#   rep(-.2, ncol(radar_data)),
#   radar_data
# )
#
# radarchart(
#   radar_data,
#   axistype = 1,
#   pcol = c("blue", "red"),
#   pfcol = c(scales::alpha("blue", 0.3), scales::alpha("red", 0.3)),
#   plwd = 2,
#   cglcol = "grey",
#   cglty = 1,
#   axislabcol = "black",
#   vlce = 0.8
# )
```



```
#  
#   legend("topright", legend = rownames(radar_data)[3:4], fill = c("blue", "red"))
```

Key Findings

- Carriers of rare variants showed trends of lower cognitive performance across domains.
- Proband status is significantly associated with lower cognitive scores independent of age and sex.
- Specific cognitive domains (e.g., Verbal Learning) showed more pronounced carrier effects.

Limitations

- Current sample size limits power for detecting small effects.
- SCHEMA gene list is based on adult-onset schizophrenia; relevance to EOP needs further investigation.
- Psychosis status in controls may introduce confounds.

Future Directions

- Expand cohort as genotyping continues.
- Incorporate whole genome sequencing.
- Analyze gene-environment interactions.
- Domain-specific modeling for individual high-impact genes.