Final Project Exploratory Analysis

Ankit Mithbavkar

2025-12-18

Load the Data

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.3.3
## Warning: package 'tibble' was built under R version 4.3.3
## Warning: package 'tidyr' was built under R version 4.3.3
## Warning: package 'readr' was built under R version 4.3.3
## Warning: package 'purrr' was built under R version 4.3.3
## Warning: package 'dplyr' was built under R version 4.3.3
## Warning: package 'stringr' was built under R version 4.3.3
## Warning: package 'forcats' was built under R version 4.3.3
## Warning: package 'lubridate' was built under R version 4.3.3
## -- 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   4.0.0     v tibble    3.2.1
## 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
```

```
data <- read_csv("raw_memory_data_study1.csv")
```

```
## Rows: 5372 Columns: 42
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): Word
## dbl (41): both_hit, both_fa, both_h_fa, d_both, c_both, img, Bird2001, Brist...
##
## 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.
```

```
head(data)
```

```
## # A tibble: 6 x 42
##   Word  both_hit both_fa both_h_fa d_both c_both   img Bird2001 Bristol2006
```

```
##   <chr>    <dbl>   <dbl>     <dbl>  <dbl>  <dbl> <dbl>    <dbl>       <dbl>
## 1 ACE       0.71    0.07      0.64   2.03   0.47   4.5    NA          NA
## 2 ACHE      0.81    0.09      0.73   2.26   0.24   3.8    NA           3.11
## 3 ACT       0.68    0.18      0.51   1.41   0.23   3.5    NA          NA
## 4 ADD       0.55    0.1       0.45   1.4    0.57   3.3     5.15       NA
## 5 AGE       0.66    0.08      0.58   1.79   0.48   3.4     5.29       NA
## 6 AID       0.68    0.15      0.53   1.51   0.27   3.5     4.75       NA
## # i 33 more variables: Chiarello1999 <dbl>, Glasgow2018 <dbl>, Cortese <dbl>,
## #   MRC <dbl>, Auditory.mean <dbl>, Gustatory.mean <dbl>, Haptic.mean <dbl>,
## #   Interoceptive.mean <dbl>, Olfactory.mean <dbl>, Visual.mean <dbl>,
## #   Foot_leg.mean <dbl>, Hand_arm.mean <dbl>, Head.mean <dbl>,
## #   Mouth.mean <dbl>, Torso.mean <dbl>, Max_strength.perceptual <dbl>,
## #   Max_strength.action <dbl>, Max_strength.sensorimotor <dbl>,
## #   Minkowski3.perceptual <dbl>, Minkowski3.action <dbl>, ...
```

Find Missing Data

When manually browsing the data, I noticed that some cells in the Object and Body columns were NA. How many total cells are NA and what proportion of the total column is missing?

```
# Check for missing data in the Object component
missing_stats <- data |>
  summarise(
    missing_body = sum(is.na(Body)),
    missing_object = sum(is.na(Object)),
    # Count rows where EITHER is missing
    total_unusable = sum(is.na(Body) | is.na(Object))
  )
missing_stats
```

```
## # A tibble: 1 x 3
##   missing_body missing_object total_unusable
##          <int>          <int>          <int>
## 1           61             61             61
```

These words are missing from the sensorimotor columns. **61** words will be excluded from the final analysis.

```
data_clean <- data |>
  filter(!is.na(Body) & !is.na(Object))
nrow(data_clean)
```

```
## [1] 5311
```

```
nrow(data)
```

```
## [1] 5372
```

61 words are now removed.

Finding the Distribution of Sensitivity

d' is a measure of discriminability/sensitivity. It tells us how easily participants can tell old words apart from new words. The higher d' is the better a participant is at distinguishing these categories of words.

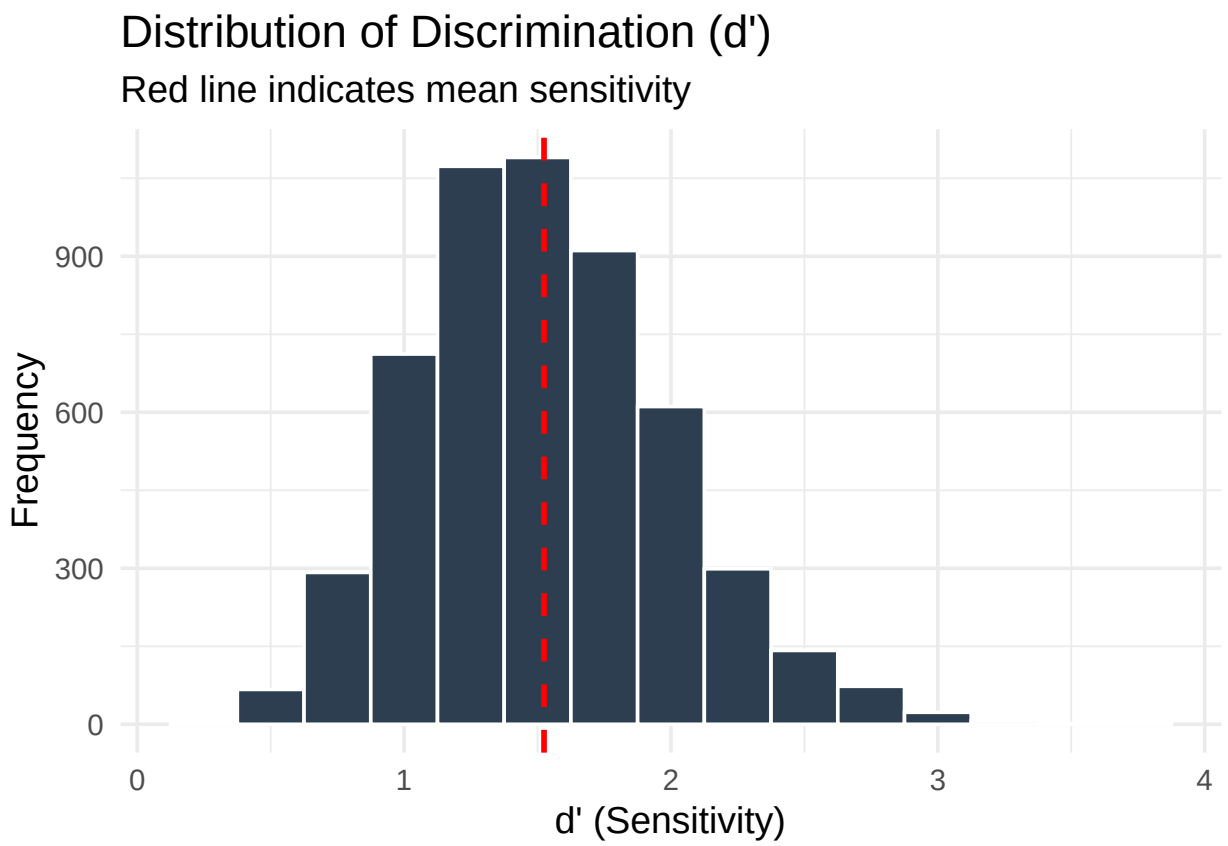
Let's explore the distribution to gauge how well participants performed at this memory task.

```
data_clean |>
  ggplot(aes(x = d_both)) +
  geom_histogram(binwidth = 0.25, fill = "#2c3e50", color = "white") +
```

```r
  geom_vline(aes(xintercept = mean(d_both, na.rm = TRUE)),
             color = "red", linetype = "dashed", size = 1) +
  labs(title = "Distribution of Discrimination (d')",
       subtitle = "Red line indicates mean sensitivity",
       x = "d' (Sensitivity)",
       y = "Frequency") +
  theme_minimal(base_size = 14)
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## Warning: Removed 4 rows containing non-finite outside the scale range
## (`stat_bin()`).
```

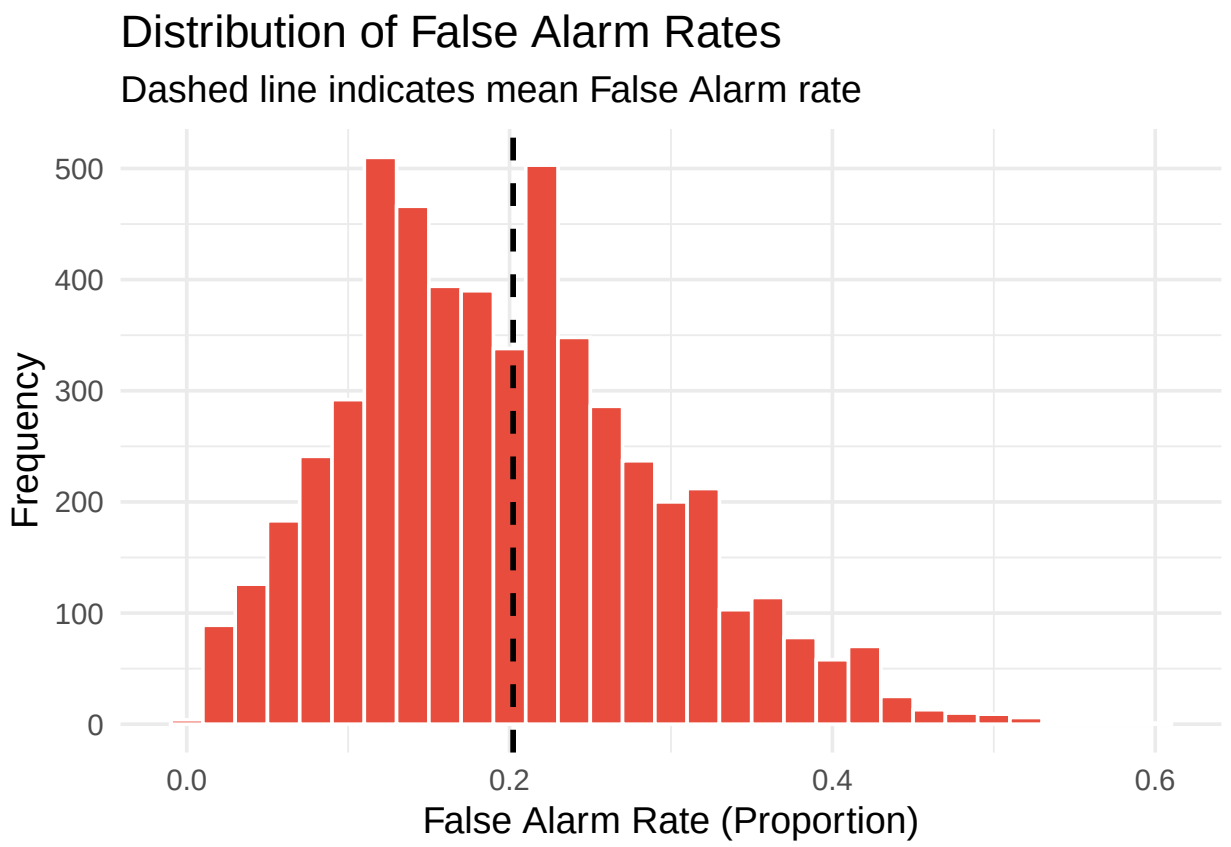


The data is roughly normal but slightly right-skewed. Since it is somewhat symmetric around 1.5, the data indicates participants generally performed decently at distinguishing between studied and new words but still faced some difficulty.

Finding the Distribution of False Alarm Rates

A false alarm means that a participant incorrectly claimed they saw a word that was actually new. This is another metric that can help us gauge the participants' general performance at this task.

```r
data_clean |>
  ggplot(aes(x = both_fa)) +
  geom_histogram(binwidth = 0.02, fill = "#e74c3c", color = "white") +
  geom_vline(aes(xintercept = mean(both_fa, na.rm = TRUE)),
             color = "black", linetype = "dashed", size = 1) +
  labs(title = "Distribution of False Alarm Rates",
       subtitle = "Dashed line indicates mean False Alarm rate",
       x = "False Alarm Rate (Proportion)",
       y = "Frequency") +
  theme_minimal(base_size = 14)
```



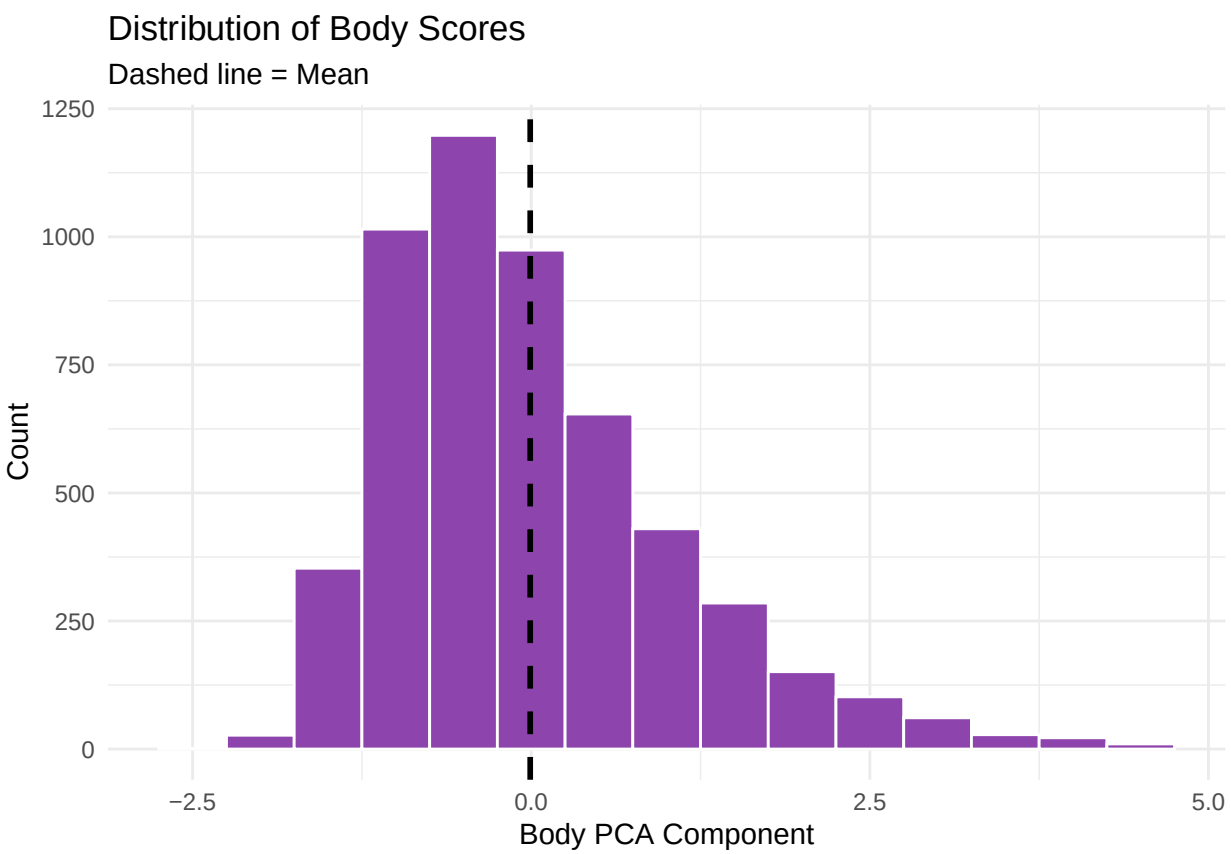
The distribution appears to be bimodal but with the mean rate being **20%**, participants were mostly good at identifying words they haven't seen on their original list of studied words.

Distribution of Body Experience

Now, let's understand the distribution of body experiences. A word with a score of 0 means that it has a "normal" level of association with that specific sensorimotor experience.

```r
data_clean |>
  ggplot(aes(x = Body)) +
  geom_histogram(binwidth = 0.5, fill = "#8e44ad", color = "white") +
  geom_vline(aes(xintercept = mean(Body)),
             color = "black", linetype = "dashed", size = 1) +
  labs(title = "Distribution of Body Scores",
       subtitle = "Dashed line = Mean",
       x = "Body PCA Component", y = "Count") +
```

```
  theme_minimal()
```

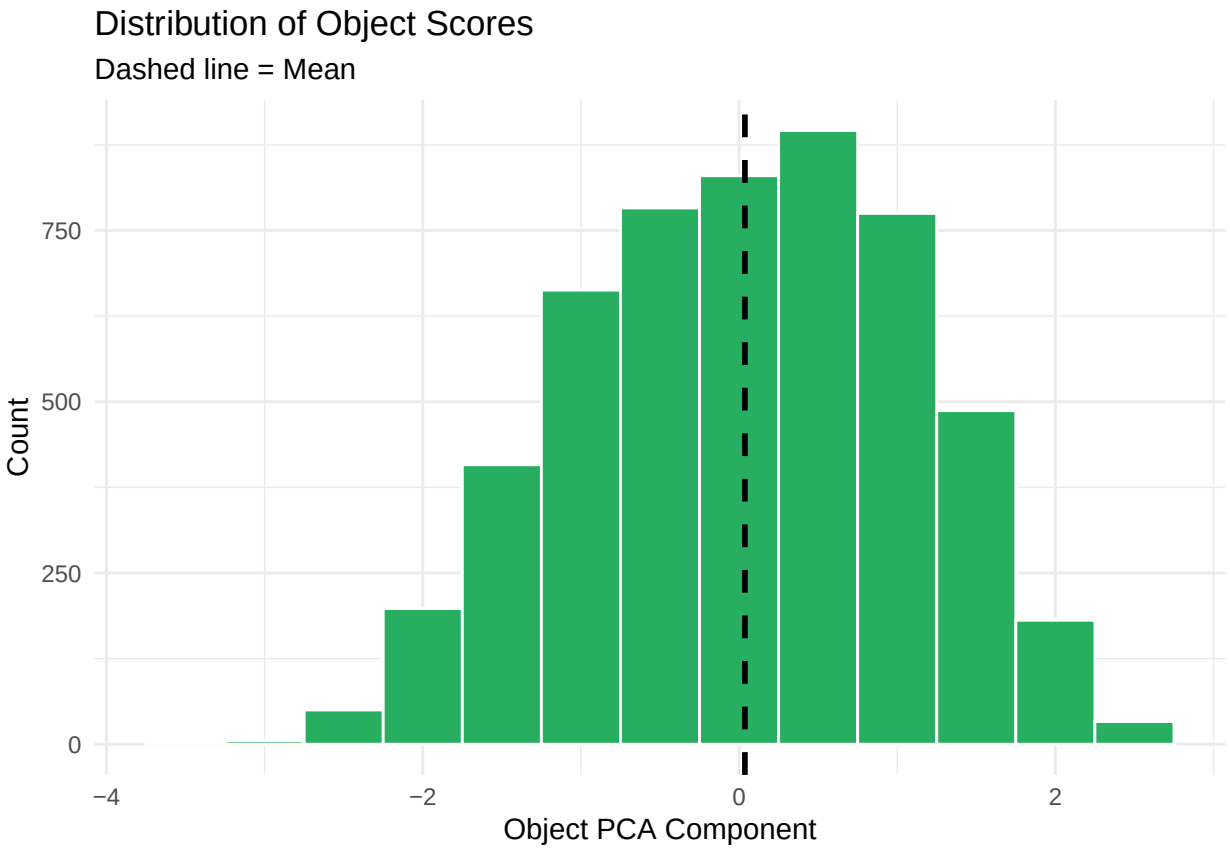


Since the distribution is right-skewed, most words have a stronger than average body association. A word with this score could be "kick" or "shiver" since they are strongly related to body experiences.

Distribution of Object Experience

Now, let's explore the distribution for external or object experiences.

```
data_clean |>
  ggplot(aes(x = Object)) +
  geom_histogram(binwidth = 0.5, fill = "#27ae60", color = "white") +
  geom_vline(aes(xintercept = mean(Object)),
             color = "black", linetype = "dashed", size = 1) +
  labs(title = "Distribution of Object Scores",
       subtitle = "Dashed line = Mean",
       x = "Object PCA Component", y = "Count") +
  theme_minimal()
```



Similar to the Body PCA Component, the average score for the Object PCA Component is 0, meaning that the average word has a roughly normal association with "Object". The distribution is roughly normal, meaning there are around the same amount of words that are highly unrelated and highly related to external experiences.