Centroid Analysis

calculate centroids

In this analysis, we first calculated the individual centroids of each participant. (Here, we take the centroid to represent the average conceptual position of everything that a specific participant shared.) (also I did not need to do this bc I do this again but whatever)

```
# calculate the mean position (centroid) for each participant
participant_centroids <- self_responses_ids %>%
  filter(!is_skip) %>%
  group_by(subject_id, age_group) %>%
  summarise(across(starts_with("Dim"), mean, na.rm = TRUE), .groups = "drop")

# 1 centroid (average vector of all vectorized responses) per participant
print(participant_centroids)
```

```
## # A tibble: 268 x 770
##    subject_id age_group Dim1_texts Dim2_texts Dim3_texts Dim4_texts Dim5_texts
##    <chr>      <chr>          <dbl>      <dbl>      <dbl>      <dbl>      <dbl>
##  1 001fafba   adult         0.155     -0.0606     0.142     -0.0209    0.00514
##  2 002e49b2   adult         0.0725    -0.0791    -0.0852     0.0845    0.0845
##  3 00982e47   adult         0.0699    -0.244      0.0238     0.274    -0.117
##  4 00a14d68   adult         0.0339    -0.0593    -0.0785     0.0217    0.159
##  5 01b0b64a   adult         0.0943    -0.218     -0.0584     0.230     0.0824
##  6 024a0cf3   child         0.0397    -0.0818     0.0741     0.0696    0.107
##  7 02a8f8f6   adult         0.100     -0.231     -0.107      0.146     0.236
##  8 039b6c84   adult         0.148     -0.120     -0.0802     0.137     0.195
##  9 03f84c9b   adult         0.0276    -0.152     -0.100      0.235     0.0351
## 10 043f55bb   adult         0.154     -0.214     -0.0957    -0.0295    0.137
## # i 258 more rows
## # i 763 more variables: Dim6_texts <dbl>, Dim7_texts <dbl>, Dim8_texts <dbl>,
## #   Dim9_texts <dbl>, Dim10_texts <dbl>, Dim11_texts <dbl>, Dim12_texts <dbl>,
## #   Dim13_texts <dbl>, Dim14_texts <dbl>, Dim15_texts <dbl>, Dim16_texts <dbl>,
## #   Dim17_texts <dbl>, Dim18_texts <dbl>, Dim19_texts <dbl>, Dim20_texts <dbl>,
## #   Dim21_texts <dbl>, Dim22_texts <dbl>, Dim23_texts <dbl>, Dim24_texts <dbl>,
## #   Dim25_texts <dbl>, Dim26_texts <dbl>, Dim27_texts <dbl>, ...
```

```
dim(participant_centroids) # [1] 268 770
```

```
## [1] 268 770
```

Then, we calculated the distance of each participant's response to their own centroid.

```r
intra_participant_distances <- self_responses_ids %>%
  filter(!is_skip) %>%
  group_by(subject_id, age_group) %>%
  group_modify(~ {

    # get all vectors for this participant
    vecs <- as.matrix(.x %>% select(starts_with("Dim")))

    # normalize em
    norm_vecs <- vecs / sqrt(rowSums(vecs^2))

    # calculate this participant's centroid
    centroid <- colMeans(norm_vecs)

    # calculate distance of each response to participant's centroid
    dists_to_centroid <- 1 - (norm_vecs %*% centroid) # (1 - cosine similarity)

    tibble(mean_dist_to_self = mean(dists_to_centroid))
  })

intra_participant_distances
```

```
## # A tibble: 268 x 3
## # Groups:   subject_id, age_group [268]
##    subject_id age_group mean_dist_to_self
##    <chr>      <chr>                 <dbl>
##  1 001fafba   adult                0.103
##  2 002e49b2   adult                0.104
##  3 00982e47   adult                0.0977
##  4 00a14d68   adult                0.108
##  5 01b0b64a   adult                0.0887
##  6 024a0cf3   child                0.0727
##  7 02a8f8f6   adult                0.0971
##  8 039b6c84   adult                0.108
##  9 03f84c9b   adult                0.115
## 10 043f55bb   adult                0.0950
## # i 258 more rows
```

```r
t.test(mean_dist_to_self ~ age_group, data = intra_participant_distances)
```

```
##
##  Welch Two Sample t-test
##
## data:  mean_dist_to_self by age_group
## t = 7.2913, df = 19.936, p-value = 4.831e-07
## alternative hypothesis: true difference in means between group adult and group child is not equal to
## 95 percent confidence interval:
##  0.01880999 0.03389008
## sample estimates:
## mean in group adult mean in group child
##          0.09505152          0.06870148
```

TODO–WRITE UP T TEST REPORT!!!

```r
get_mean_group_distance <- function(centroid_df) {

  # extract vectors
  vecs <- as.matrix(centroid_df %>% select(starts_with("Dim")))

  # normalize vectors
  norm_vecs <- vecs / sqrt(rowSums(vecs^2))

  # dot product between vectors
  sim_matrix <- norm_vecs %*% t(norm_vecs)

  # convert similarity to distance: distance = 1 - similarity
  dist_matrix <- 1 - sim_matrix

  # only take the values above the diagonal (to avoid self-distance and duplicates)
  return(mean(dist_matrix[upper.tri(dist_matrix)], na.rm = TRUE))
}

adult_distances <- get_mean_group_distance(participant_centroids %>% filter(age_group == "adult"))
child_distances <- get_mean_group_distance(participant_centroids %>% filter(age_group == "child"))

cat("Adult Inter-participant Distance:", adult_distances, "\n") # 0.02628904
```

```
## Adult Inter-participant Distance: 0.02628904
```

```r
cat("Child Inter-participant Distance:", child_distances, "\n") # 0.03238887
```

```
## Child Inter-participant Distance: 0.03238887
```

moment of truth... between group comparison (bootstrapped)

Then, we compared the distance of each participant's centroid from other participants' within their respective age group.

```r
set.seed(13126) # for reproducibility
adult_subsample_size <- 18
n_iterations <- 10000

boot_adult_dists <- replicate(n_iterations, {
  participant_centroids %>%
    filter(age_group == "adult") %>%
    sample_n(adult_subsample_size) %>%
    get_mean_group_distance()
})

cat("Mean Adult Dist (Bootstrapped):", mean(boot_adult_dists), "\n")
```

```
## Mean Adult Dist (Bootstrapped): 0.02624849
```

```r
cat("Actual Child Distance:", child_distances, "\n")
```

```
## Actual Child Distance: 0.03238887
```

```r
cat("Empirical p-value:", sum(boot_adult_dists >= child_distances) / 1000, "\n")
```

```
## Empirical p-value: 0.876
```

```r
plot_boot_dist <- data.frame(dist = boot_adult_dists)

ggplot(plot_boot_dist, aes(x = dist)) +
  geom_histogram(bins = 50, fill = "#648FFF", alpha = 0.7) +
  geom_vline(xintercept = child_distances, color = "#FE6100", linetype = "dashed", size = 1) +
  theme_minimal() +
  labs(title = "Null Distribution of Adult Inter-participant Distances",
       subtitle = "The orange line represents the actual mean child distance",
       x = "Mean Pairwise Distance",
       y = "Frequency")
```

To assess semantic distance between participants' centroids, we first L2-normalized the centroids to account for differences in response magnitude. We then calculated the pairwise Euclidean distances between these normalized vectors, a transformation that is monotonically related to cosine distance and ensures that our dispersion analysis reflects differences in conceptual direction rather than vector length. (not sure if I have to mention this previous sentence)

```r
# pull each participant's centroid
vec_matrix <- as.matrix(participant_centroids %>% select(starts_with("Dim")))
age_labels <- participant_centroids$age_group

# L2 (Euclidean) normalize participants' centroids (scales the vector so that its total length/magnitud
norm_vecs <- vec_matrix / sqrt(rowSums(vec_matrix^2))
# technically, Euclidean distance, but normalized above so equiv. to cosine distance
dist_mat <- dist(norm_vecs)

disp_test <- betadisper(dist_mat, age_labels)
permutest(disp_test)
```

```
##
## Permutation test for homogeneity of multivariate dispersions
## Permutation: free
## Number of permutations: 999
##
## Response: Distances
##            Df Sum Sq   Mean Sq      F N.Perm Pr(>F)
## Groups      1 0.0045 0.0044973 2.1412    999  0.124
## Residuals 266 0.5587 0.0021004
```

To examine whether the groups differed in their overall semantic variability, we conducted a permutation test for homogeneity of multivariate dispersions (PERMDISP) on the participant centroids. The analysis indicated that there was no significant difference in the multivariate dispersion between children and adults ($F(1, 266) = 2.14, p = .13$, based on 999 permutations). This suggests that the groups occupy a similar 'semantic volume' in the embedding space, despite their different conceptual locations.

visualize

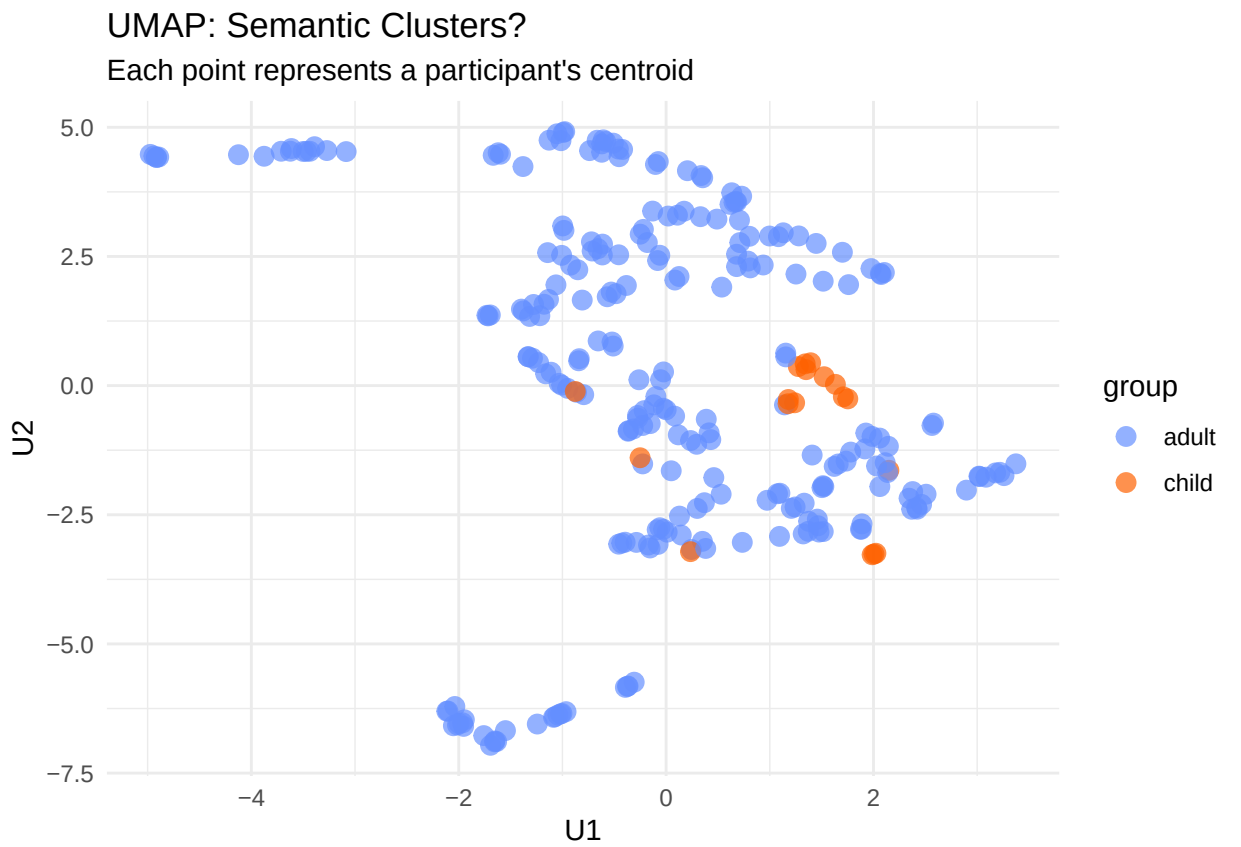
So, here is a UMAP visualization of the participant centroids... not sure what to make of it. Does not look very meaningful to me.

```
# generate coordinates
umap_res <- umap(vec_matrix, n_neighbors = 4, min_dist = 0.005, metric = "cosine")

umap_df <- data.frame(
  U1 = umap_res[,1],
  U2 = umap_res[,2],
  subject_id = participant_centroids$subject_id,
  group = age_labels)

# parse out clusters
clusters <- dbscan(umap_df %>% select(U1, U2), eps = 0.2, minPts = 3)
umap_df$cluster <- as.factor(clusters$cluster)

ggplot(umap_df, aes(x = U1, y = U2, color = group)) +
  geom_point(size = 3, alpha = 0.7) +
  theme_minimal() +
  scale_color_manual(values = c("adult" = "#648FFF", "child" = "#FE6100")) +
  labs(title = "UMAP: Semantic Clusters?",
       subtitle = "Each point represents a participant's centroid")
```



IF WE INCLUDE UMAP SHOULD IT BE USING RAW VECTORS? ### UMAP of raw vectors

```r
# pull raw vectors
raw_vec_matrix <- as.matrix(self_responses_ids %>%
                            filter(!is_skip) %>%
                            select(starts_with("Dim")))

# run UMAP on raw vectors
umap_raw <- umap(raw_vec_matrix,
                 n_neighbors = 15,
                 min_dist = 0.1,
                 metric = "cosine")

umap_raw_df <- data.frame(
  U1 = umap_raw[,1],
  U2 = umap_raw[,2],
  subject_id = (self_responses_ids %>% filter(!is_skip))$subject_id,
  age_group = (self_responses_ids %>% filter(!is_skip))$age_group
)
```

```r
ggplot(umap_raw_df, aes(x = U1, y = U2)) +

  # adults in gray
  geom_point(data = filter(umap_raw_df, age_group == "adult"),
             color = "gray80", alpha = 0.2, size = 0.5) +

  # children colored by subject_id
  geom_point(data = filter(umap_raw_df, age_group == "child"),
             aes(color = subject_id), alpha = 0.8, size = 2) +
  theme_void() +
  theme(legend.position = "none") +
  labs(title = "UMAP of raw participant responses",
       subtitle = "Each point represents a response: child responses colored by sub_id; adults responses
```

UMAP of raw participant responses

Each point represents a response: child responses colored by sub_id; adults responses in gray

```
set.seed(13126)

# subset 18 adults
sampled_adult_ids <- umap_raw_df %>%
  filter(age_group == "adult") %>%
  pull(subject_id) %>%
  unique() %>%
  sample(18)


ggplot(umap_raw_df, aes(x = U1, y = U2)) +

  # children in dark gray
  geom_point(data = filter(umap_raw_df, age_group == "child"),
             color = "black", alpha = 0.4, size = 1) +

  # non sub-sampled adults in light gray
  geom_point(data = filter(umap_raw_df, age_group == "adult", !(subject_id %in% sampled_adult_ids)),
             color = "gray90", alpha = 0.2, size = 0.8) +

  # sampled adults colored by subject_id
  geom_point(data = filter(umap_raw_df, subject_id %in% sampled_adult_ids),
             aes(color = as.factor(subject_id)), alpha = 0.8, size = 2) +
  theme_void() +
  theme(legend.position = "none",
        plot.background = element_rect(fill = "white", color = NA)) +
```

```
  labs(title = "Adult responses",
       subtitle = "gray for children; colors represent individual adults")
```