STAT 380 FINAL PROJECT REPORT (GROUP 1)

Analysis of Congressional Voting in the 119th Congress

By: Sid Tekumalla, Sean Dasovich, Lucas Sadoulet, Li Zhu

INTRODUCTION

Politics is tricky when it comes to data analysis because emotions and humans are involved. When these things are involved, no data will be perfect, and we are sure that there will be outliers present. Although it's not perfect data, we still would like to see if there are patterns that emerge in the way these congress members vote. In our modern world, politics is more divided than ever. If you look at the news, there is constantly fighting between party lines. We want to see how true this is. Is America truly as divided as we see, or is that just the news? This project is giving us the opportunity to explore perception vs reality.

For this project, we took a while to select a data set that would pique our interest and be insightful. We have a joint interest in politics, social behavior, and public policy. We landed with a data set from voteview.com which describes how each member of Congress has voted on bills throughout history. We are fascinated by the way congress members vote whether that's crossing party-lines or shifting positions over time. We are also interested in if there are ways one could group together members of Congress to see if there's patterns in ways members vote. Throughout discussions and initial exploratory data analysis we landed on three research questions:

I. Intra-Party Factions: Using the NOMINATE scores, can we identify distinct Republican factions (Freedom Caucus, moderates, Trump-aligned) and Democratic factions (progressives, moderates, Blue Dogs)? How accurate is this identification?

II. Age and Ideology: Using the birth year data, is there a correlation between member age and ideological positioning in 2025, controlling for party? What factors most predict a politician's age? Are younger Republicans/Democrats different from older ones, and in what ways?

III. Party Loyalty: Are politics getting more divided, i.e. is there less cooperation on bill / less breaking rank with each progressing congress? Are there alternative scoring principal components beyond NOMINATE which would provide more accurate models?

DATA DESCRIPTION

The Data set we used came from Vote View which is a public resource that has information on congress voting. We used the member ideology dataset for the 119th Congress. The Data set had one row per member of Congress, indicating whether the member belongs to the House or Senate. The key variables can be broken into several categories: member identification, basic demographics, congressional information, and ideology scores.

Member Identification

- **bioname:** Full legislator name
- **icpsr:** Unique numeric identifier
- **bioguide_id:** Biographical directory ID

Basic Demographics

- **born:** Birth year
- **died:** Year of death (if applicable)
- **state_abbrev:** Two-letter state code
- **district_code:** House district number (0 for Senators)

Congressional Information

- **congress:** Session number (119th)
- **chamber:** House or Senate
- **party_code:**
    - 100 = Democrat
    - 200 = Republican
    - 328 = Independent

Ideology Scores

- Nominate_dim1: First dimension of DW-NOMINATE ideology score
- Nominate_dim2: Second dimension of DW-NOMINATE ideology score
- Nominate_log_likelihood: Long-likelihood of model fit for the member's ideological estimation
- Nominate_geo_mean_probability: Geometric mean of the probabilities that the model predicts each vote correctly
- Nominate_number_of_votes: Total number of roll-call votes used to estimate the position
- Nominate_number_of_errors: Number of incorrect predicted votes by the model
- Conditional: Used internally for DW_NOMINATE conditional estimation
- Nokken_poole_dim1: Alternative ideological dimension score using Nokken-poole method
- Nokken_poole_dim2: Second Nokken-poole dimension score

METHODOLOGY

Intra-Party Factions Methodology

We cluster members within each party using k-means on (Nominate_dim1, Nominate_dim2) and validate with elbow and silhouette diagnostics.

- Nominate_dim1: liberal (−) vs. conservative (+)
- Nominate_dim2: often captures establishment (+) vs. populist (−) tendencies in the modern era

We choose k using elbow (WSS) and average silhouette over k_grid. Then we fit k-means and visualize a 2D cluster map.

Age and Ideology Methodology

We built a multiple linear regression model. We had the response variable as age and the predictors were chambered (House/Senate), state, nominate_dim1, nominate_dim2, nokken_poole_dim1 and nokken_poole_dim2. This linear regression helps us to see if age is associated with ideology.

Party Loyalty with Custom Principal Components Methodology

We derived custom ideological scores for each legislator by performing singular value decomposition (SVD) on the roll-call vote matrix, then reduced the diagnosed product matrix to produce two principal components (X, Y) that summarize voting behavior. The first two singular vectors were scaled by the square roots of their corresponding singular values in order to normalize their importance. These coordinates were then merged with legislator metadata using ICPSR identifiers to align with party affiliation and other attributes.

To evaluate the predictive strength of these custom scores, we compared them to the established NOMINATE scores. For each roll-call vote, we fit separate logistic regression models predicting vote choice using either the Custom or NOMINATE scores as predictors. Predicted probabilities were used to quantify both accuracy (proportion of votes correctly predicted) and confidence (distance of predicted probabilities from 0.5, with higher values indicating more decisive predictions). Distributions of accuracy and confidence across all roll calls were summarized and visualized to facilitate a comparison between the two scoring systems.

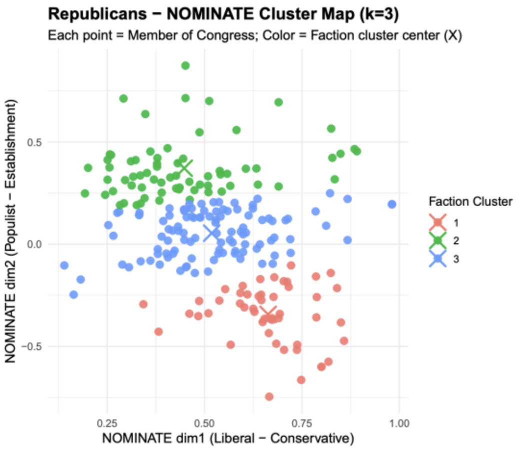
DATA ANALYSIS RESULTS

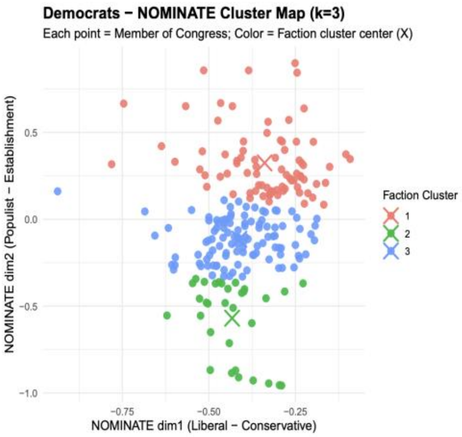
Intra-Party Factions Results

Although the Democrats were split into three clusters in the k-means map (k = 3), this choice was made mainly to stay consistent with the research question. In fact, the silhouette plot shows that k = 2 has the highest average silhouette score, meaning two clusters provide the clearest separation according to the underlying NOMINATE distances. When we examine the numerical cluster centers, the three Democratic clusters are extremely close to each other. All three centers fall within a very narrow range. As a result, the labeling rules classify all three Democratic clusters as “Mainline Moderate,” indicating that the algorithm’s three-way split does not correspond to three meaningfully distinct ideological factions within the caucus.

With k = 3, the Republican NOMINATE map visually splits into three clusters, but the numerical results show that two of the clusters sit extremely close to each other in the ideological space and both represent the same mainstream “Traditional Conservative” group. The third cluster—located lower on the second dimension—is the only one that clearly separates, corresponding to a “Trump-aligned / Populist” faction. So even though the algorithm technically identifies three groups, the substantive interpretation is that Republicans largely divide into one major establishment bloc and a smaller Trump-aligned faction, with the two establishment clusters reflecting minor within-group variation rather than distinct ideological camps.

There is no clear point in either WSS plot where the total within-cluster sum of squares levels off; instead, both show a gradually flattening pattern as k increases. Silhouette scores above 0.7 indicate strong clustering and scores above 0.5 indicate reasonable clustering, yet for both Republicans and Democrats, the highest silhouette value occurs at k = 2 and remains below 0.5. This suggests that neither party exhibits a meaningful clustering structure. Although k = 3 was chosen to match the requirements of the research question, the more appropriate choice based on the metrics would be k = 2. Furthermore, the resulting NOMINATE dimension analysis shows that the ideological separation is not very distinct: while we expected three ideological levels for both parties, the actual clustering reveals fewer than three clearly differentiated ideological group.





Age and Ideology Results

As the EDA showed, controlling party affiliation, there is no visible correlation between congress member age and ideological positioning in the current Congress. This is further supported if we were to try linear regression, as shown in the code. We choose a range of important predictors like chamber, state, and ideological scores, and run a regression on them. The multiple R-squared value is a measly 0.1811, and the adjusted R-squared is an even lower 0.05292. This means that a very small amount of the variance in the data is covered by our predictor variables, making them unfit to explain age.

It is well known that American Congress members skew on the old side. The median age is 59, and plenty of congress members are far older than this. Our analysis shows that this age is not related to location or ideology, party or chamber, suggesting that this high age is built into the institution itself. Further studies would be needed to examine the institution, probably through a study of national/state laws and party election rules. These, however, are beyond the scope of our project.

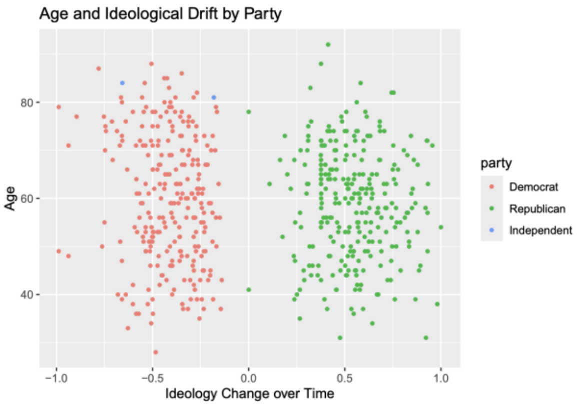


Image 1: A scatterplot comparison of age vs ideological drift, done under the assumption that older politicians will have a greater overall change in their beliefs. This assumption is proven unfounded.

Party Loyalty with Custom Principal Components

Deriving the Custom Scores

We derived custom principal components from the SVD decomposition of the roll-call vote matrix and compared them to the established NOMINATE dimensions. When deciding how many components to retain, there needs to strike a balance between capturing meaningful variance with avoiding overfitting. Since complexity is not an easy metrix to derive in these types of problems, we must locate an elbow point the models Scree plots[1], indicating that the first two components explained most of the variance and were sufficient for producing accuracte models before the diminishing returns of futhure complexity. This conclusion is supported by the fact that the third principal component accounted for only a small fraction of additional variance (~2%) relative to the added complexity (~50%). Additionally, comparing NOMINATE and the custom scoring system is most informative when both systems use the same number of dimensions.

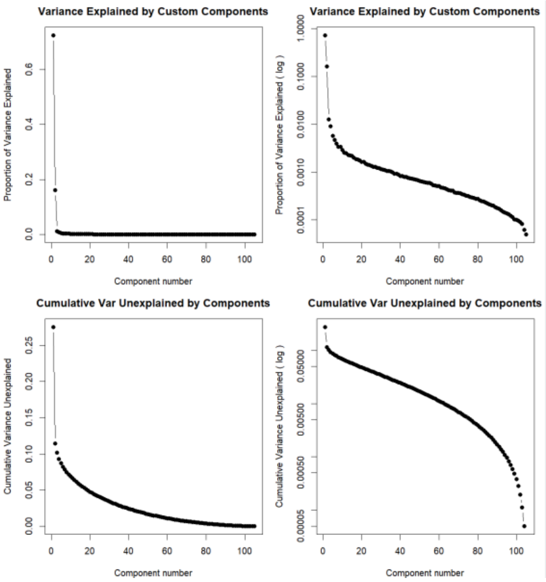


Image 1: Variance and Cumulative Unexplained Variance in linear and log-scale by most to least important Custom Principal Component

Comparison of Custom and NOMINATE Scores

Across this specific Congress, both custom dimensions showed strong correlation with the first NOMINATE dimension, while correlations with the second NOMINATE dimension were noticeably weaker. This suggests that the custom scores capture much of the primary ideological divide represented by NOMINATE 1, reflecting the importance of party alignment. The second NOMINATE dimension, by contrast, appears to capture secondary, intra-party or issue-specific variation, highlighting subtler patterns that are less represented in the custom components.

To compare the predictive strength of the two scoring systems, we went beyond simply assigning party labels, since both systems perfectly classify legislators' political parties. Instead, we fit logistic regression models predicting individual vote choices using either the custom scores or NOMINATE dimensions. Predicted probabilities were used to calculate both accuracy and model confidence. For this Congress, the custom dimensions achieved significantly higher accuracy (~93.4% vs ~93.0%) and confidence (~43.2% vs ~42.4%), as determined by paired t-test, indicating that they were more decisive in predicting voting behavior. Density plots[2] of predicted probabilities illustrated these differences, with the custom scores showing a tighter distribution of high-confidence predictions. In summary, the custom dimensions effectively capture the primary partisan divide and slightly outperform NOMINATE in predicting individual votes for this Congress. However, they are less effective at representing secondary, intra-party variation, which NOMINATE's second dimension continues to capture, and the Custom model may struggle in a congress with lesser party loyalty.

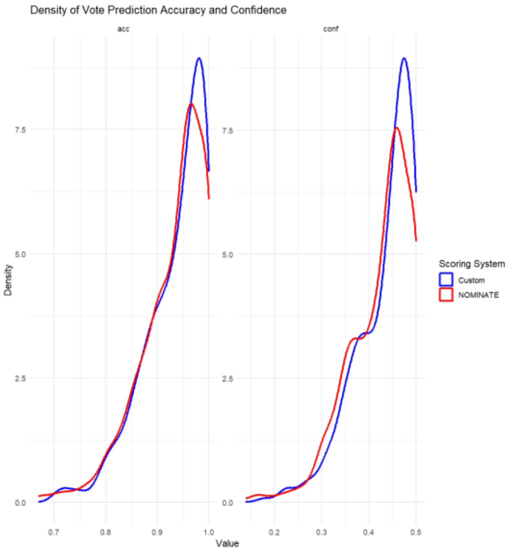


Image 2: Density Graphs comparing the accuracy and confidence of the Custom and NOMINATE scores according to a logistic regression model.

Visualization of Party Clustering

When mapping legislators using the two scoring systems, party affiliation is the dominant factor in shaping the ideological scorings. While both systems achieve similar predictive accuracy, the NOMINATE scores are able to capture greater variance along the second dimension, which largely ignores party alignment. This secondary spread appears intentional, likely designed to prevent overemphasis on party and reduce overfitting to the primary partisan divide. In the results, this means the two scores are uncorrelated[3], meaning that they provide more individual utility to the model. Inversely, the custom components have a higher correlation between one another, and produce a more tightly clustered embedding, with separation primarily reflecting members with partial terms or unusual voting patterns. A change in X would impact Y heavily, and vice versa, indicating that these variables are not able to gain as significant utility as possible. This suggests that the custom scores may be overfitting the specific votes in this Congress and could have weaker utility to new roll calls.

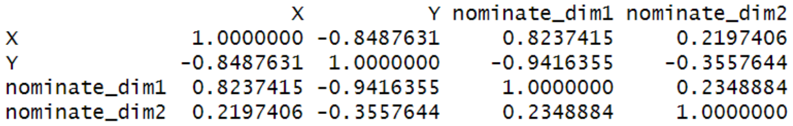

```
                       X          Y nominate_dim1 nominate_dim2
X              1.0000000 -0.8487631     0.8237415     0.2197406
Y             -0.8487631  1.0000000    -0.9416355    -0.3557644
nominate_dim1  0.8237415 -0.9416355     1.0000000     0.2348884
nominate_dim2  0.2197406 -0.3557644     0.2348884     1.0000000
```

Image 3: The correlation table between the principal components of both models: Custom (X, Y) and NOMINATE (nomintate_dim1, nominate_dim2)

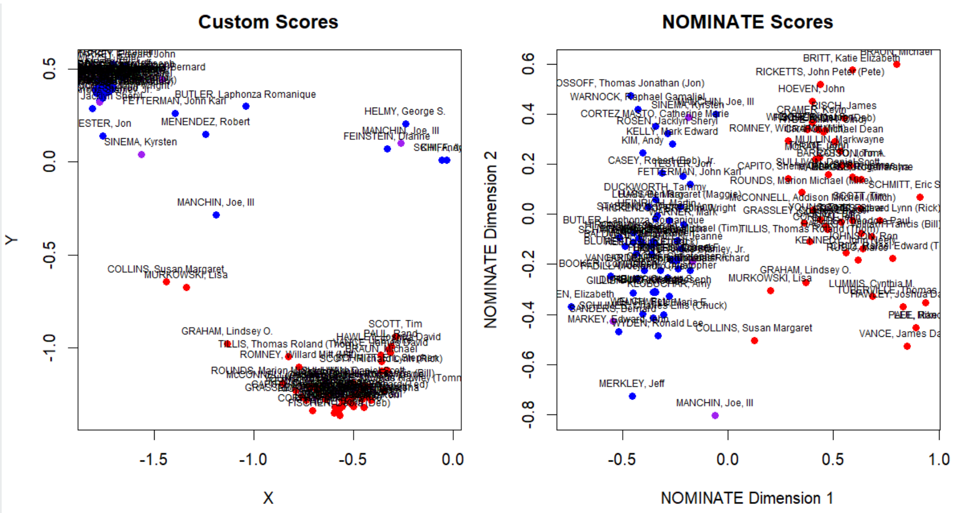


Image 4: The principal component plots of the Custom and NOMINATE Scores by political party.

CONCLUSION

This project used information from Vote View ideology data set to answer our three research questions about congressional behavior and patterns.

We found the Democrats of congress are unified across two dimensions even when we forced 3 clusters. On the other hand, for the Republicans there was a large subgroup, Trump aligned and a smaller traditional conservative subgroup. This shows with voting the Democrats tend to be more unified while there is less unification with the Republicans.

Next, the age and ideology analysis showed no evidence that age can predict ideology. We created multiple scatterplots to find a trend and were unsuccessful. Further analysis proved that our predictor variables were unfit to predict age.

Finally, while party alignment is clearly the dominant factor in congressional voting behavior, deriving secondary components, like intra-party variation and issue-specific, are important to creating strong scoring systems. Overreliance on party as a parameter, risks overfitting a specific Congress, highlighting that party loyalty remains the dominant influence in congressional politics.

Overall, our project uses multiple statistical analytical tools to find trends in congressional politics. While there were a few limitations in the dataset, it was still useful in deriving insightful and differing hypothesis. In the future we could go further back to history, exploring the trends as they change and further supporting our models and analysis across new datasets. There are many other questions which this data set could answer that would be interesting to look at in the future.

AUTHOR CONTRIBUTION STATEMENT:

Sid: age-ideology analysis, built regression model, wrote methodology and results for his research question

LI: cleaned and prepared the data set, performed the intra-party faction analysis, creating silhouette and elbow diagnostics. Wrote methodology and results for his question.

Sean: wrote introduction, data description, conclusion. Integrated everyone's work into the final report and completed this document.

Lucas: derived questions, wrote script, and conducted analysis for party loyalty, strength of NOMINATE system, and offering alterative scoring by using custom principal components.

Code Appendix:

Appendix: Full Code

```
library(tidyverse)
library(cluster)
library(factoextra)  # fviz_nbclust helpers
library(ggrepel)
library(ggplot2)
library(dplyr)
library(caret)
library(tidyr)
set.seed(params$seed)
knitr::opts_chunk$set(comment = NA)
knitr::opts_chunk$set(echo = FALSE, include = FALSE)

data <- read.csv("HS119_members.csv")

#Age And Ideology Setup
# Create age column
data$age <- 2025 - data$born

# Remove rows where age cannot be calculated
data <- data %>% filter(!is.na(age))


# Convert party_code to factor with labels
data$party <- factor(data$party_code,
                     levels = c(100, 200, 328),
                     labels = c("Democrat", "Republican", "Independent"))
#Party Loyalty Setup
S118_votes <- read.csv("S118_votes.csv")
S118_members <- read.csv("S118_members.csv")
S118_rollcalls <- read.csv("S118_rollcalls.csv")
#Intra-Party Factions
infile <- params$infile
which_chamber <- params$which_chamber
k_grid <- params$k_min:params$k_max

df_raw <- readr::read_csv(infile, show_col_types = FALSE)

# Basic sanity check for expected columns
needed <- c("chamber","party_code","bioname","nominate_dim1","nominate_dim2")
missing <- setdiff(needed, names(df_raw))
if (length(missing) > 0) {
  stop(paste("Missing required columns:", paste(missing, collapse=", ")))
}

df <- df_raw %>%
  filter(chamber == which_chamber) %>%
  select(bioname, party_code, chamber, nominate_dim1, nominate_dim2) %>%
  filter(!is.na(nominate_dim1), !is.na(nominate_dim2))

df_dem <- df %>% filter(party_code == 100)   # Democrats
df_gop <- df %>% filter(party_code == 200)   # Republicans
```

```r
tibble(
  party = c("Democrats","Republicans"),
  N = c(nrow(df_dem), nrow(df_gop))
)
fit_party_clusters <- function(df_party, party_label, k_grid = 2:5) {
  if (nrow(df_party) < min(k_grid)) {
    warning(paste0("Too few members in ", party_label, " for k>=2. Skipping."))
    return(NULL)
  }
  X <- df_party %>% select(nominate_dim1, nominate_dim2)

  # Elbow
  wss <- sapply(k_grid, function(k){
    km <- kmeans(X, centers = k, nstart = 50)
    km$tot.withinss
  })

  # Silhouette
  sil_avgs <- sapply(k_grid, function(k) {
    km <- kmeans(X, centers = k, nstart = 50)
    sil <- silhouette(km$cluster, dist(X))
    mean(sil[, "sil_width"])
  })

  #k_best <- k_grid[which.max(sil_avgs)]
  k_best <- 3
  km <- kmeans(X, centers = k_best, nstart = 200)
  df_out <- df_party %>% mutate(cluster = factor(km$cluster))

  centers <- as_tibble(km$centers) %>% mutate(cluster = factor(1:n()))

  p_elbow <- tibble(k = k_grid, wss = wss) %>%
    ggplot(aes(k, wss)) +
    geom_line() + geom_point() +
    labs(title = paste0(party_label, " - Elbow (WSS)"),
         x = "k", y = "Total within-cluster SS") +
    theme_minimal()

  p_sil <- tibble(k = k_grid, silhouette = sil_avgs) %>%
    ggplot(aes(k, silhouette)) +
    geom_line() + geom_point() +
    labs(title = paste0(party_label, " - Average Silhouette by k"),
         x = "k", y = "Average silhouette width") +
    theme_minimal()

  p_map <- df_out %>%
  ggplot(aes(nominate_dim1, nominate_dim2, color = cluster)) +
  geom_point(size = 3, alpha = 0.9) +  # larger, clear points
  # Mark cluster centers with an X
  geom_point(data = centers, aes(nominate_dim1, nominate_dim2, color = cluster),
             size = 6, shape = 4, stroke = 1.5, inherit.aes = FALSE) +
  labs(
    title = paste0(party_label, " - NOMINATE Cluster Map (k=", k_best, ")"),
    subtitle = "Each point = Member of Congress; Color = Faction cluster center (X)",
    x = "NOMINATE dim1 (Liberal - Conservative)",
    y = "NOMINATE dim2 (Populist - Establishment)",
    color = "Faction Cluster"
  ) +
  theme_minimal(base_size = 13) +
  theme(
    plot.title = element_text(face = "bold"),
    legend.position = "right",
    legend.title = element_text(size = 12),
    legend.text = element_text(size = 10)
  )


  list(
    df = df_out,
    centers = centers,
    k_best = k_best,
    silhouette_table = tibble(k = k_grid, avg_silhouette = sil_avgs),
    elbow_plot = p_elbow,
    silhouette_plot = p_sil,
    map_plot = p_map
  )
}

# Heuristic labeling rules - tweak after inspecting centers
label_dem_cluster <- function(center_dim1, center_dim2) {
  if (center_dim1 <= -0.55) return("Progressive")
  if (center_dim1 >= -0.30 && center_dim2 >= 0.05) return("Blue Dog / Moderate-Establishment")
  return("Mainline Moderate")
}

label_gop_cluster <- function(center_dim1, center_dim2) {
  if (center_dim1 >= 0.60 && center_dim2 <= 0.00) return("Trump-aligned / Populist")
  if (center_dim1 <= 0.35 && center_dim2 >= 0.05) return("Moderate / Establishment")
  return("Traditional Conservative")
}

apply_labels <- function(res, party = c("D","R")) {
  party <- match.arg(party)
  if (is.null(res)) return(NULL)
  centers_labeled <- res$centers %>%
    rowwise() %>%
    mutate(
      faction = if (party == "D")
        label_dem_cluster(nominate_dim1, nominate_dim2)
      else
        label_gop_cluster(nominate_dim1, nominate_dim2)
    ) %>%
    ungroup()
  df_labeled <- res$df %>%
    left_join(centers_labeled %>% select(cluster, faction), by = "cluster")
  list(members = df_labeled, centers = centers_labeled)
```

```
}
res_dem <- fit_party_clusters(df_dem, "Democrats", k_grid)
if (!is.null(res_dem)) {
  print(res_dem$elbow_plot)
}
if (!is.null(res_dem)) {
  print(res_dem$silhouette_plot)
}
if (!is.null(res_dem)) {
  print(res_dem$map_plot)
}
if (!is.null(res_dem)) {
  out_dem <- apply_labels(res_dem, "D")

  cat("## Cluster Centers (Democrats)
")
  print(out_dem$centers %>% arrange(faction))
  "~/Documents/GitHub/Stat_380_Final_Proj/HS119_members.csv"
  cat("## Counts by Faction (Democrats)
")
  print(out_dem$members %>% count(faction, sort = TRUE))
}
res_gop <- fit_party_clusters(df_gop, "Republicans", k_grid)
if (!is.null(res_gop)) {
  print(res_gop$elbow_plot)
}
if (!is.null(res_gop)) {
  print(res_gop$silhouette_plot)
}
if (!is.null(res_gop)) {
  print(res_gop$map_plot)
}
if (!is.null(res_gop)) {
  out_gop <- apply_labels(res_gop, "R")

  cat("## Cluster Centers (Republicans)
")
  print(out_gop$centers %>% arrange(faction))

  cat("## Counts by Faction (Republicans)
")
  print(out_gop$members %>% count(faction, sort = TRUE))
}
closest_to_centers <- function(df_members, centers_tbl, top_n = 8) {
  if (is.null(df_members)) return(invisible(NULL))
  out <- list()
  for (cl in centers_tbl$cluster) {
    cen <- centers_tbl %>% filter(cluster == cl) %>%
      select(nominate_dim1, nominate_dim2) %>% as.numeric()
    tmp <- df_members %>%
      filter(cluster == cl) %>%
      mutate(dist_to_center = sqrt((nominate_dim1 - cen[1])^2 + (nominate_dim2 - cen[2])^2)) %>%
      arrange(dist_to_center) %>%
```

```r
      slice_head(n = top_n) %>%
      select(bioname, nominate_dim1, nominate_dim2, faction, dist_to_center)
    out[[as.character(cl)]] <- tmp
  }
  out
}

if (exists("out_dem") && !is.null(out_dem)) {
  cat("## Democrats - Examples Closest to Centers
")
  print(closest_to_centers(out_dem$members, out_dem$centers, 10))
}
if (exists("out_gop") && !is.null(out_gop)) {
  cat("## Republicans - Examples Closest to Centers
")
  print(closest_to_centers(out_gop$members, out_gop$centers, 10))
}
#Age and Ideology Question

# Compare with chamber
ggplot(data, aes(x = chamber, y = age, color = party)) +
  geom_point(size = 1) +
  labs(title = "Age and Chamber by Party",
       x = "Chamber",
       y = "Age",
       color = "Party")

# Compare with State
ggplot(data, aes(x = state_abbrev, y = age, color=party)) +
  geom_point(size = 1) +
  labs(title = "Age and State by Party",
       x = "State",
       y = "Age")

# Compare with Ideological Score
ggplot(data, aes(x = nominate_dim1, y = age, color = party)) +
  geom_point(size = 1) +
  labs(title = "Age and Ideology by Party",
       x = "Ideology Score",
       y = "Age")

# Compare with Ideological Radicality Score
ggplot(data, aes(x = nominate_dim2, y = age,color = party)) +
  geom_point(size = 1) +
  labs(title = "Age and Ideology Radicality by Party",
       x = "Ideology Radicality Score",
       y = "Age")


# Compare with Ideological Score over Time
ggplot(data, aes(x = nokken_poole_dim1, y = age, color = party)) +
  geom_point(size = 1) +
  labs(title = "Age and Ideological Drift by Party",
       x = "Ideology Change over Time",
       y = "Age")


#Party Loyalty
data <- S118_votes
votes <- subset(data, select = -c(congress, chamber, prob))
votes$cast_code = votes$cast_code == 1

mat <- xtabs( cast_code ~ rollnumber + icpsr , data = votes)
mat

decomp <- svd(mat)
decomp$u
decomp$d
decomp$v

k = 3

d <- decomp$d[1:2]
U2 <- decomp$u[, 1:2, drop = FALSE]
V2 <- decomp$v[, 1:2, drop = FALSE]

U_scaled <- U2 %*% diag(sqrt(d))
V_scaled <- V2 %*% diag(sqrt(d))

icpsr_coords <- data.frame(
  icpsr = as.character(colnames(mat)),
  X = V_scaled[,1],
  Y = V_scaled[,2]
)
colnames(mat)


S118_members$icpsr <- as.character(S118_members$icpsr)
head(S118_members)
head(icpsr_coords)
named_icpsr <- left_join(icpsr_coords, S118_members, by = 'icpsr')

named_icpsr$party_color <- ifelse(
  named_icpsr$party_code == 200, "red",
  ifelse(named_icpsr$party_code == 100, "blue", "purple")
)

plot(
  named_icpsr$X, named_icpsr$Y,
  pch = 19,
  col = named_icpsr$party_color,
  xlab = "Component 1",
  ylab = "Component 2",
  main = "Rank-2 Embedding of Legislators (icpsr)"
)
```

```
text(named_icpsr$X, named_icpsr$Y,
     labels = named_icpsr$bioname,
     pos = 3, cex = 0.7)


rows <- grep("CORR", named_icpsr$bioname)
named_icpsr[rows, ]




# testing model strength
d <- decomp$d
pve <- d^2 / sum(d^2)
tot_pve <- cumsum(pve)


# PC scree matrix
options(scipen = 999)
par(mfrow = c(2, 2), mar = c(4, 4, 3, 1))
plot(pve, type = "b", pch = 19,
     xlab = "Component number",
     ylab = "Proportion of Variance Explained",
     main = "Variance Explained by Custom Components")

plot(pve, type = "b", pch = 19,
     xlab = "Component number",
     ylab = "Proportion of Variance Explained ( log )",
     main = "Variance Explained by Custom Components",
     log = "y")

plot(1-tot_pve, type = "b", pch = 19,
     xlab = "Component number",
     ylab = "Cumulative Variance Unexplained",
     main = "Cumulative Var Unexplained by Components")

plot(1-tot_pve, type = "b", pch = 19,
     xlab = "Component number",
     ylab = "Cumulative Variance Unexplained ( log )",
     main = "Cumulative Var Unexplained by Components",
     log = "y")


# corrilation scores
#S118_rollcalls$icpsr <- as.character(S118_rollcalls$icpsr)

score_data <- named_icpsr[, c("X", "Y", "nominate_dim1", "nominate_dim2")]
cor_matrix <- cor(score_data, use = "complete.obs")
print(cor_matrix)

custom_dims <- c("X", "Y")
nom_dims    <- c("nominate_dim1", "nominate_dim2")

for (c_dim in custom_dims) {
```

```r
  for (n_dim in nom_dims) {

    plot(
      named_icpsr[[c_dim]],
      named_icpsr[[n_dim]],
      pch = 19,
      col = named_icpsr$party_color,
      xlab = paste("Custom", c_dim),
      ylab = paste("NOMINATE", n_dim),
      main = paste(c_dim, "vs", n_dim)
    )

    text(
      named_icpsr[[c_dim]],
      named_icpsr[[n_dim]],
      labels = named_icpsr$bioname,
      pos = 3,
      cex = 0.6
    )
  }
}


# ploting scores
plot(named_icpsr$X, named_icpsr$Y,
     pch = 19,
     col = named_icpsr$party_color,
     xlab = "X",
     ylab = "Y",
     main = "Custom Scores")

text(named_icpsr$X, named_icpsr$Y,
     labels = named_icpsr$bioname,
     pos = 3, cex = 0.6)

plot(named_icpsr$nominate_dim1, named_icpsr$nominate_dim2,
     pch = 19,
     col = named_icpsr$party_color,
     xlab = "NOMINATE Dimension 1",
     ylab = "NOMINATE Dimension 2",
     main = "NOMINATE Scores")

text(named_icpsr$nominate_dim1, named_icpsr$nominate_dim2,
     labels = named_icpsr$bioname,
     pos = 3, cex = 0.6)

# getting corrilation
cor(named_icpsr$X, named_icpsr$nominate_dim1, use = "complete.obs")
cor(named_icpsr$X, named_icpsr$nominate_dim2, use = "complete.obs")
cor(named_icpsr$Y, named_icpsr$nominate_dim2, use = "complete.obs")
cor(named_icpsr$Y, named_icpsr$nominate_dim1, use = "complete.obs")

# finding acc and conf
```

```r
S118_votes$icpsr <- as.character(S118_votes$icpsr)
named_icpsr$icpsr <- as.character(named_icpsr$icpsr)

results <- list()

for (roll in unique(S118_votes$rollnumber)) {

  votes_sub <- S118_votes |>
    filter(rollnumber == roll) |>
    left_join(named_icpsr, by = "icpsr") |>
    filter(!is.na(X), !is.na(nominate_dim1))

  if (nrow(votes_sub) == 0) next

  votes_sub$cast_bin <- ifelse(votes_sub$cast_code == 1, 1, 0)

  glm_custom <- glm(cast_bin ~ X + Y, data = votes_sub, family = binomial)
  prob_custom <- predict(glm_custom, type = "response")
  pred_custom <- ifelse(prob_custom > 0.5, 1, 0)
  acc_custom <- mean(pred_custom == votes_sub$cast_bin, na.rm = TRUE)
  conf_custom <- mean(abs(prob_custom - 0.5), na.rm = TRUE)

  glm_nom <- glm(cast_bin ~ nominate_dim1 + nominate_dim2, data = votes_sub, family = binomial)
  prob_nom <- predict(glm_nom, type = "response")
  pred_nom <- ifelse(prob_nom > 0.5, 1, 0)
  acc_nom <- mean(pred_nom == votes_sub$cast_bin, na.rm = TRUE)
  conf_nom <- mean(abs(prob_nom - 0.5), na.rm = TRUE)

  results[[as.character(roll)]] <- data.frame(
    rollnumber = roll,
    acc_custom, acc_nom,
    conf_custom, conf_nom
  )
}

results_df <- do.call(rbind, results)
head(results_df)
summary(results_df)

# acc and conf matrix
results_long <- results_df %>%
  select(acc_custom, acc_nom, conf_custom, conf_nom) %>%
  pivot_longer(
    cols = everything(),
    names_to = c("metric", "system"),
    names_sep = "_",
    values_to = "value"
  )

results_df <- results_df[, !(names(results_df) %in% "rollnumber")]
col_means <- sapply(results_df, mean, na.rm = TRUE)
col_vars <- sapply(results_df, var, na.rm = TRUE)
summary_stats <- data.frame(
  Mean = col_means,
  Variance = col_vars
)

summary_stats

#t-test
t.test(results_df$acc_custom, results_df$acc_nom, paired = TRUE)
t.test(results_df$conf_custom, results_df$conf_nom, paired = TRUE)

# plots for acc and conf
ggplot(results_long, aes(x = value, color = system)) +
  geom_density(size = 1) +
  facet_wrap(~metric, scales = "free") +
  scale_color_manual(values = c("blue", "red"), labels = c("Custom", "NOMINATE")) +
  labs(
    x = "Value",
    y = "Density",
    color = "Scoring System",
    title = "Density of Vote Prediction Accuracy and Confidence"
  ) +
  theme_minimal()
```