Clustering

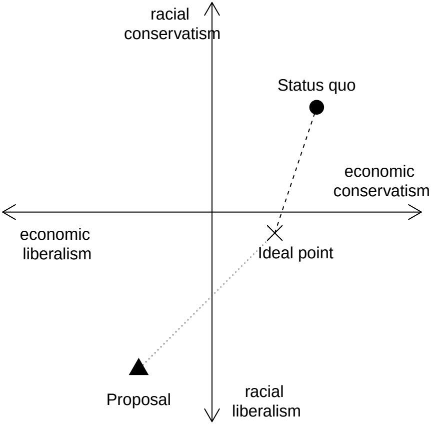
Introduction to Quantitative Social Science

Kosuke Imai
Harvard University / University of Tokyo

Summer 2022

Measuring Political Polarization

- Has the US Congress been polarizing over time?
- Measuring political polarization ⇝measuring ideology
- Analysis of roll call votes using spatial voting model



Item Response Theory

- The probability of voting yes on a proposal is determined by

$$
\begin{aligned}
& \text{distance between Ideal point and Proposal}^2 \\
& \quad - \text{distance between Ideal point and Status quo}^2 \\
= \; & \left\{(x_{\text{ideal}} - x_{\text{proposal}})^2 + (y_{\text{ideal}} - y_{\text{proposal}})^2\right\} \\
& \quad - \; \left\{(x_{\text{ideal}} - x_{\text{status quo}})^2 + (y_{\text{ideal}} - y_{\text{status quo}})^2\right\} \\
= \; & \alpha + \beta \, x_{\text{ideal}} + \gamma \, y_{\text{ideal}}
\end{aligned}
$$

- The model originally developed in educational testing literature
  - test questions ↭ legislative proposals
  - answering the questions ↭ voting on the proposals
  - ability ↭ ideal point
  - α: difficulty parameter
  - β: discrimination parameter

DW-NOMINATE scores

Name	Description
name	name of a Congressional representative
state	state of a Congressional representative
district	district number of a Congressional representative
party	party of a Congressional representative
congress	Congressional session number
dwnom1	DW-NOMINATE score (first dimension)
dwnom2	DW-NOMINATE score (second dimension)

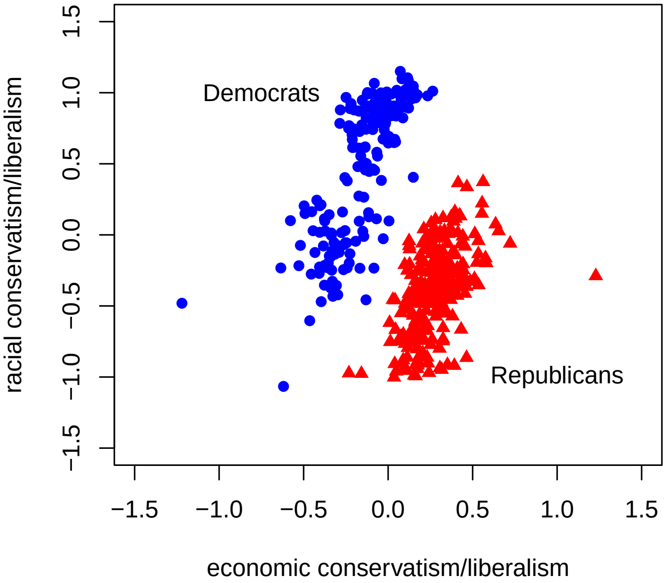
```r
congress <- read.csv("data/congress.csv")
## subset the data by party
rep <- subset(congress, subset = (party == "Republican"))
dem <- congress[congress$party == "Democrat", ]
```

- Ideal points for the 80th (1947-48) and 120th (2011-12) Congresses

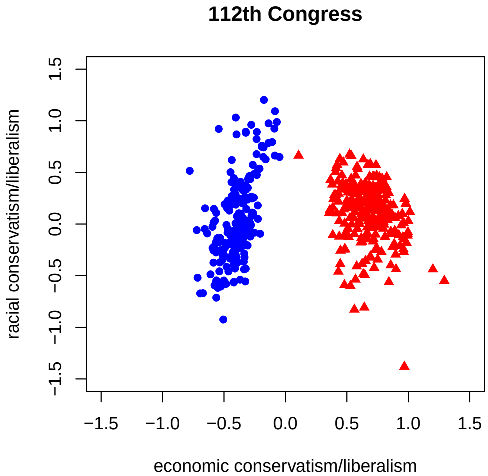
```r
rep80 <- subset(rep, subset = (congress == 80))
dem80 <- subset(dem, subset = (congress == 80))
rep112 <- subset(rep, subset = (congress == 112))
dem112 <- subset(dem, subset = (congress == 112))
```

```r
## preparing labels and axis limits to avoid repetition
xlab <- "economic conservatism/liberalism"
ylab <- "racial conservatism/liberalism"
lim <- c(-1.5, 1.5)
## plot democrats and then republicans
plot(dem80$dwnom1, dem80$dwnom2, pch = 16, col = "blue",
     xlim = lim, ylim = lim, xlab = xlab, ylab = ylab,
     main = "80th Congress")
points(rep80$dwnom1, rep80$dwnom2, pch = 17, col = "red")
text(-0.75, 1, "Democrats")
text(1, -1, "Republicans")
```

80th Congress



```r
plot(dem112$dwnom1, dem112$dwnom2, pch = 16, col = "blue",
     xlim = lim, ylim = lim, xlab = xlab, ylab = ylab,
     main = "112th Congress")
points(rep112$dwnom1, rep112$dwnom2, pch = 17, col = "red")
```



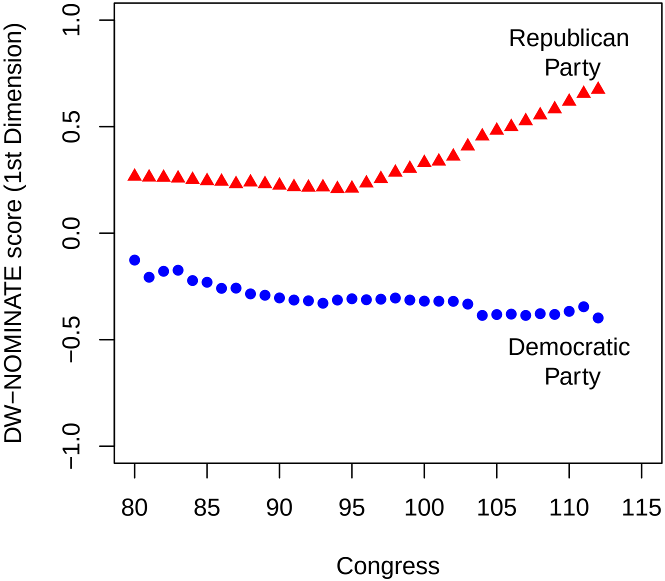
Party Median

- Party median represents a measure of party's ideological center
- Compute party median for each Congress

```r
dem.median <- tapply(dem$dwnom1, dem$congress, median)
rep.median <- tapply(rep$dwnom1, rep$congress, median)
```

- Create a time-series plot

```r
plot(as.integer(names(dem.median)), dem.median,
     col = "blue", pch = 16, xlim = c(80, 115),
     ylim = c(-1, 1), xlab = "Congress",
     ylab = "DW-NOMINATE score (1st Dimension)")
points(as.integer(names(rep.median)), rep.median,
       col = "red", pch = 17)
text(110, -0.6, "Democratic\n Party")
text(110, 0.85, "Republican\n Party")
```


DW−NOMINATE score (1st Dimension)
1.0
0.5
0.0
−0.5
−1.0
Republican Party
Democratic Party
80
85
90
95
100
105
110
115
Congress

Clustering

- Who are clustered (ideologically) with each other in Congress?
- Polarization ⇝ legislators cluster with members of their party
- Are there clusters within each party?

- Clustering algorithm: discover groups of observations similar to each other
- Unsupervised learning vs. supervised learning
- Descriptive and exploratory data analysis
- Applications of clustering algorithms to text and network data

k-means Algorithm Demonstration

1. Start the balls at three different places in the room

2. Students closest to the brown ball are in group 1
3. Students closest to the black ball are in group 2
4. Students closest to the blue ball are in group 3

5. Move the brown ball to the middle of group 1
6. Move the black ball to the middle of group 2
7. Move the blue ball to the middle of group 3

8. Repeat 2–7 until the balls no longer need to move

k-means Clustering Algorithm

1. Choose the initial centroids of k clusters
2. Given the centroids, assign each observation to a cluster whose centroid is the closest (in terms of Euclidian distance) to that observation
3. Choose the new centroid of each cluster whose coordinate equals the within-cluster mean of the corresponding variable
4. Repeat Steps 2 and 3 until cluster assignments no longer change

- Two inputs: number of clusters, starting values
- random multiple starting values
- no direct way of evaluating the performance

Discovering Clusters in Congress

- Create an input matrix to cluster

```
congress <- read.csv("data/congress.csv")
dwnom80 <- cbind(congress$dwnom1[congress$congress == 80],
                 congress$dwnom2[congress$congress == 80])
dwnom112 <- cbind(congress$dwnom1[congress$congress == 112]
                  congress$dwnom2[congress$congress == 112])
```

- cbind() (rbind()) to combine objects by rows (columns)
- Useful operations on matrix: colSums(), rowSums(), colMeans(), rowMeans(), or more generally apply()

```
colMeans(dwnom80)
## [1] 0.087711 0.000585
apply(dwnom80, 2, mean)
## [1] 0.087711 0.000585
```

- Choose the number of clusters and run the *k*-means algorithm

```
k80two.out <- kmeans(dwnom80, centers = 2)
k112two.out <- kmeans(dwnom112, centers = 2)
```

- The output is a list containing multiple elements of different types

```
names(k80two.out)

## [1] "cluster"      "centers"      "totss"
## [4] "withinss"     "tot.withinss" "betweenss"
## [7] "size"         "iter"         "ifault"
```

- The resulting centroids extracted using $

```
k80two.out$centers

##       [,1]   [,2]
## 1  0.1521 -0.344
## 2 -0.0561  0.769
```

```
k112two.out$centers

##      [,1]   [,2]
## 1 -0.391 0.0326
## 2  0.678 0.0906
```

- Clusters by party

```
table(party = congress$party[congress$congress == 80],
      cluster = k80two.out$cluster)

##                cluster
## party          1   2
##   Democrat    59 135
##   Other        2   0
##   Republican 247   3

table(party = congress$party[congress$congress == 112],
      cluster = k112two.out$cluster)

##                cluster
## party          1   2
##   Democrat   200   0
##   Republican   1 242
```

Plot the Results of *k*-means Algorithm

- Clustering for the 80th Congress

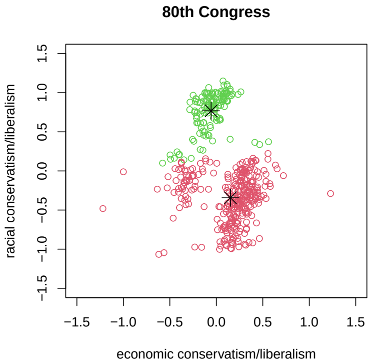
```
plot(dwnom80, col = k80two.out$cluster + 1, xlab = xlab,
     ylab = ylab, xlim = lim, ylim = lim,
     main = "80th Congress")
points(k80two.out$centers, pch = 8, cex = 2)
```

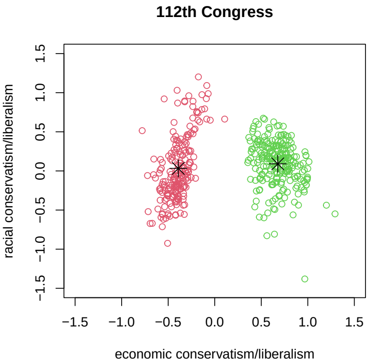
- Clustering for the 112th Congress

```
plot(dwnom112, col = k112two.out$cluster + 1,
     xlab = xlab, ylab = ylab, xlim = lim, ylim = lim,
     main = "112th Congress")
points(k112two.out$centers, pch = 8, cex = 2)
```

- color choice

```
palette()
## [1] "black"   "#DF536B" "#61D04F" "#2297E6" "#28E2E5"
## [6] "#CD0BBC" "#F5C710" "gray62"
## preparing labels and axis limits to avoid repetition
xlab <- "economic conservatism/liberalism"
ylab <- "racial conservatism/liberalism"
lim <- c(-1.5, 1.5)
```





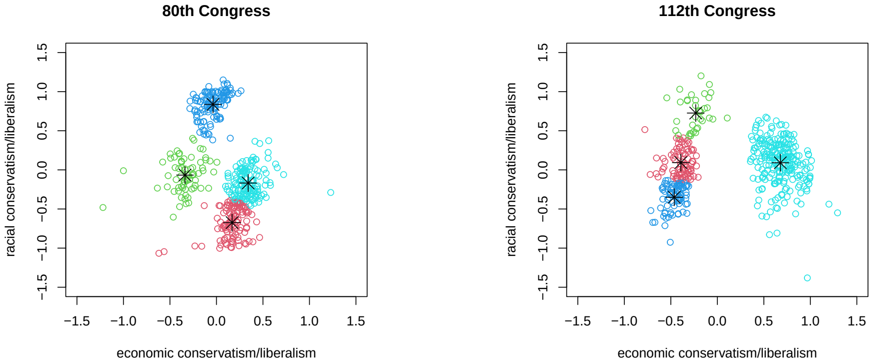
Four Clusters

- Clustering for the 80th Congress

```r
k80four.out <- kmeans(dwnom80, centers = 4)
plot(dwnom80, col = k80four.out$cluster + 1, xlab = xlab,
     ylab = ylab, xlim = lim, ylim = lim,
     main = "80th Congress")
points(k80four.out$centers, pch = 8, cex = 2)
```

- Clustering for the 112th Congress

```r
k112four.out <- kmeans(dwnom112, centers = 4)
plot(dwnom112, col = k112four.out$cluster + 1,
     xlab = xlab, ylab = ylab, xlim = lim, ylim = lim,
     main = "112th Congress")
points(k112four.out$centers, pch = 8, cex = 2)
```


80th Congress
112th Congress
racial conservatism/liberalism
economic conservatism/liberalism
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5