Homework 4: Unsupervised Learning

MACS 30100: Perspectives on Computational Modeling
University of Chicago

For the most part, I collaberated with Jinfei Zhu, Xi Cheng and Boya Fu, with minor changes.

Overview

For each of the following prompts, produce responses *with* code in-line. While you are encouraged to stage and draft your problem set solutions using any files, code, and data you'd like within the private repo for the assignment, *only the final, rendered PDF with responses and code in-line will be graded.*

Note: take a look at the `hw04.pdf` file to see a better rendering of this problem set (e.g., cleaner looking table, etc.).

Dimension Reduction

Conceptual Problems

1. (5 points) Compute the total variance from the following PCA output.

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
Standard deviation	3.55	2.41	1.82	1.31	1.05	0.86	0.81	0.79	0.72	0.70
Variance	3.45	3.10	1.75	0.98	0.64	0.33	0.31	0.30	0.09	0.05

The total variance is the sum of variances of all individual principal components.

```
sum(3.45, 3.10, 1.75, 0.98, 0.64, 0.33, 0.31, 0.30, 0.09, 0.05)
```

```
## [1] 11
```

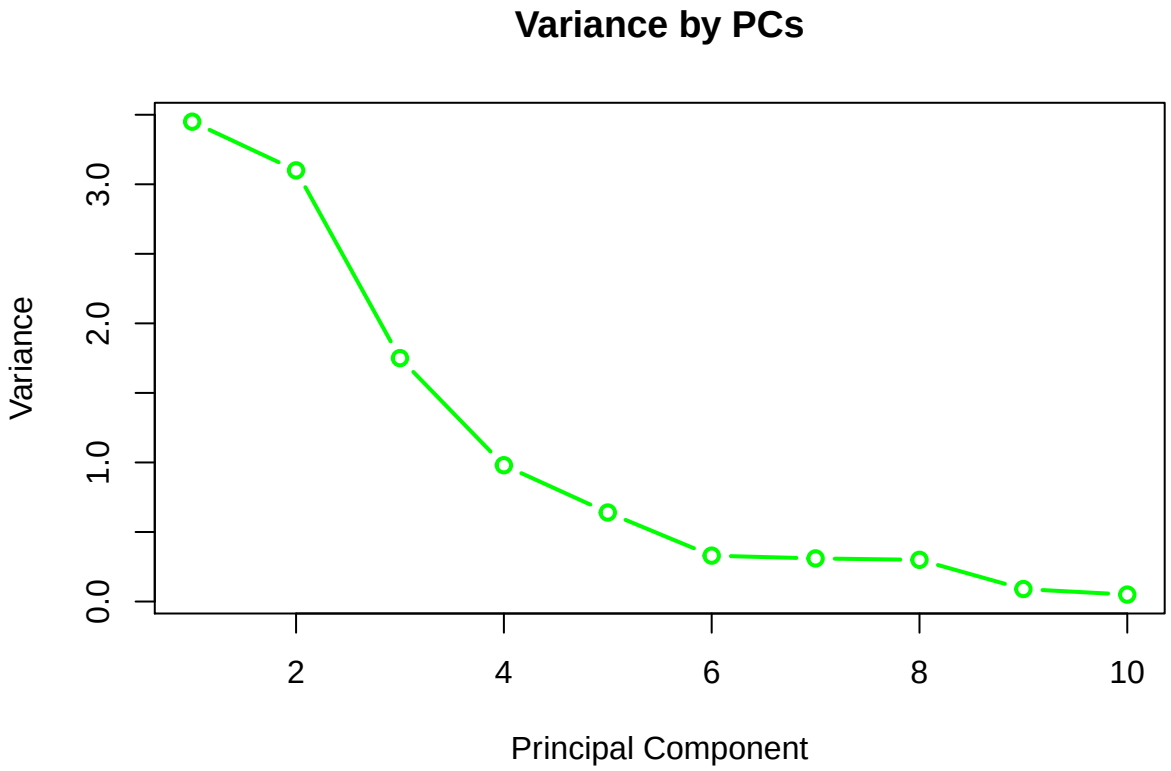
2. (10 points) Make a *manual* scree plot based on these results. That is, *no* canned functions or packages (e.g., `factoextra`). A scree plot shows how much variation each PC captures from the data.

```
library(ggplot2)
library(stringr)
library(tibble)

number_of_PCs <- 1:10
variance <- c(3.45, 3.10, 1.75, 0.98, 0.64, 0.33, 0.31, 0.30, 0.09, 0.05)
PVE = variance / sum(variance) * 100 # in percentage
tibble(number_of_PCs, variance, PVE, cumsum(PVE))
```

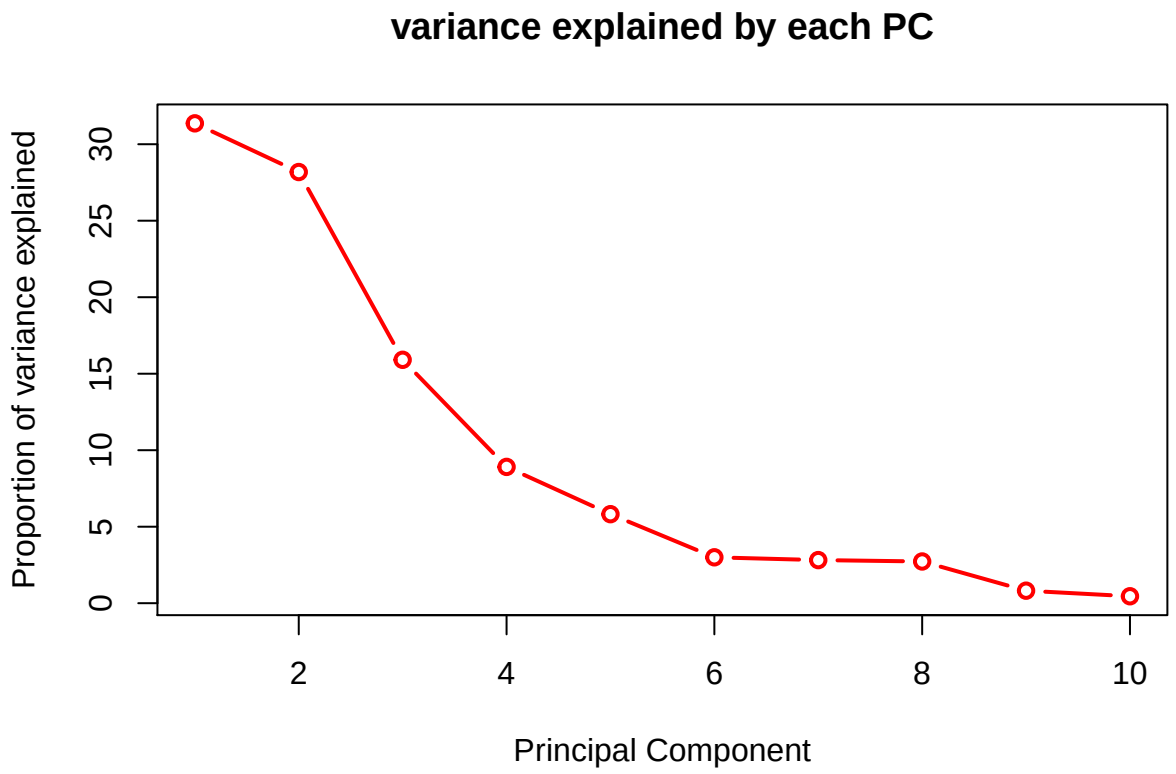
```
## # A tibble: 10 x 4
##    number_of_PCs variance    PVE `cumsum(PVE)`
##            <int>    <dbl>  <dbl>         <dbl>
##  1             1     3.45 31.4           31.4
##  2             2     3.1  28.2           59.5
##  3             3     1.75 15.9           75.5
```

```
##  4             4     0.98  8.91           84.4
##  5             5     0.64  5.82           90.2
##  6             6     0.33  3.             93.2
##  7             7     0.31  2.82           96
##  8             8     0.3   2.73           98.7
##  9             9     0.09  0.818          99.5
## 10            10     0.05  0.455         100
```

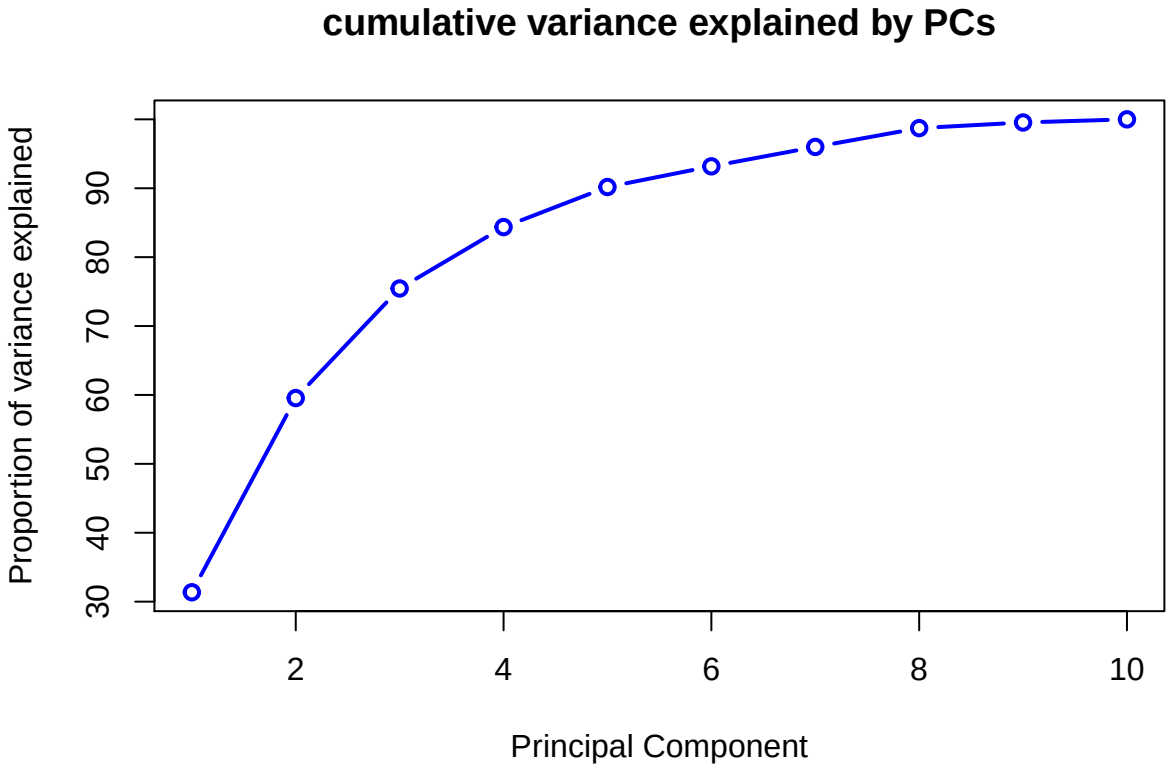
```r
plot(variance, xlab="Principal Component",
     ylab="Variance",
     main="Variance by PCs",
     type='b',
     col="green",
     lwd=2)
```




```r
plot(PVE, xlab="Principal Component",
     ylab="Proportion of variance explained",
     main="variance explained by each PC",
     type='b',
     col="red",
     lwd=2)
```




```
plot(cumsum(PVE), xlab="Principal Component",
     ylab="Proportion of variance explained",
     main="cumulative variance explained by PCs",
     type='b',
     col="blue",
     lwd=2)
```



3. (10 points) Based on your results in the previous question, how many PCs would you suggest characterize these data well? That is, what would the dimensionality of your new reduced data space be?

The cumulative variance explained by the PCs plot provides a good indication of when components hit the point of diminishing returns (i.e., little variance is gained by retaining additional components). Also, the cumulative variance explained by the PCs should be greater than 80%, therefore, I would choose 5 PCs to characterize this example.

4. (10 points) Calculate the Euclidean distance between each of the following observations, i, and some observation at 0 (i.e., x_0) in 4-dimensional space $\forall X \in \{1, 2, 3, 4\}$.

i	X_1	X_2	X_3	X_4	Euclidean Distance
1	2	2	3	1	...
2	1	1	-2	2	...
3	1	-2	-2	-1	...
4	3	3	2	2	...
5	-3	2	-1	1	...

```
m <- matrix(c(0, 0, 0, 0, 2, 2, 3, 1, 1, 1, -2, 2, 1, -2, -2, -1, 3, 3, 2, 2, -3, 2, -1, 1),
            nrow = 6, ncol = 4, byrow = TRUE,
            dimnames = list(c("0", "1", "2", "3", "4", "5"),
                            c("X1", "X2", "X3", "X4")))
m
```

```
##   X1 X2 X3 X4
## 0  0  0  0  0
## 1  2  2  3  1
## 2  1  1 -2  2
## 3  1 -2 -2 -1
## 4  3  3  2  2
```

```
## 5 -3  2 -1  1
```

```
distance.matrix <- dist(m, method = "euclidean", diag = T)
distance.matrix
```

```
##          0        1        2        3        4        5
## 0 0.000000
## 1 4.242641 0.000000
## 2 3.162278 5.291503 0.000000
## 3 3.162278 6.782330 4.242641 0.000000
## 4 5.099020 2.000000 4.898979 7.348469 0.000000
## 5 3.872983 6.403124 4.358899 6.082763 6.855655 0.000000
```

```
distance.matrix <- dist(m, method = "euclidean")
distance.matrix
```

```
##          0        1        2        3        4
## 1 4.242641
## 2 3.162278 5.291503
## 3 3.162278 6.782330 4.242641
## 4 5.099020 2.000000 4.898979 7.348469
## 5 3.872983 6.403124 4.358899 6.082763 6.855655
```

An Applied Problem

For the following applied problem, use the 2019 American National Election Study (ANES) Pilot survey data. These data include, among many other features, a battery of 35 feeling thermometers, which are questions with answers ranging from 1 to 100 for how respondents "rate" some topic (e.g., *How would you rate Obama?* or *How would you rate Japan?*). See the documentation and more detail here.

To make your lives a bit easier, I have preprocessed the data for you, including: 1) feature engineering (via kNN) for missing data, and 2) reduction of the feature space to include only the 35 feeling thermometers and a feature for the respondent's party affiliation (`democrat`), where 1 = Democrat and 0 = non-Democrat (which could be Republican, Independent, or decline to say).

5. (10 points) Fit a PCA model on all 35 feeling thermometers from the 2019 ANES, but be careful to *not* include the party affiliation feature.

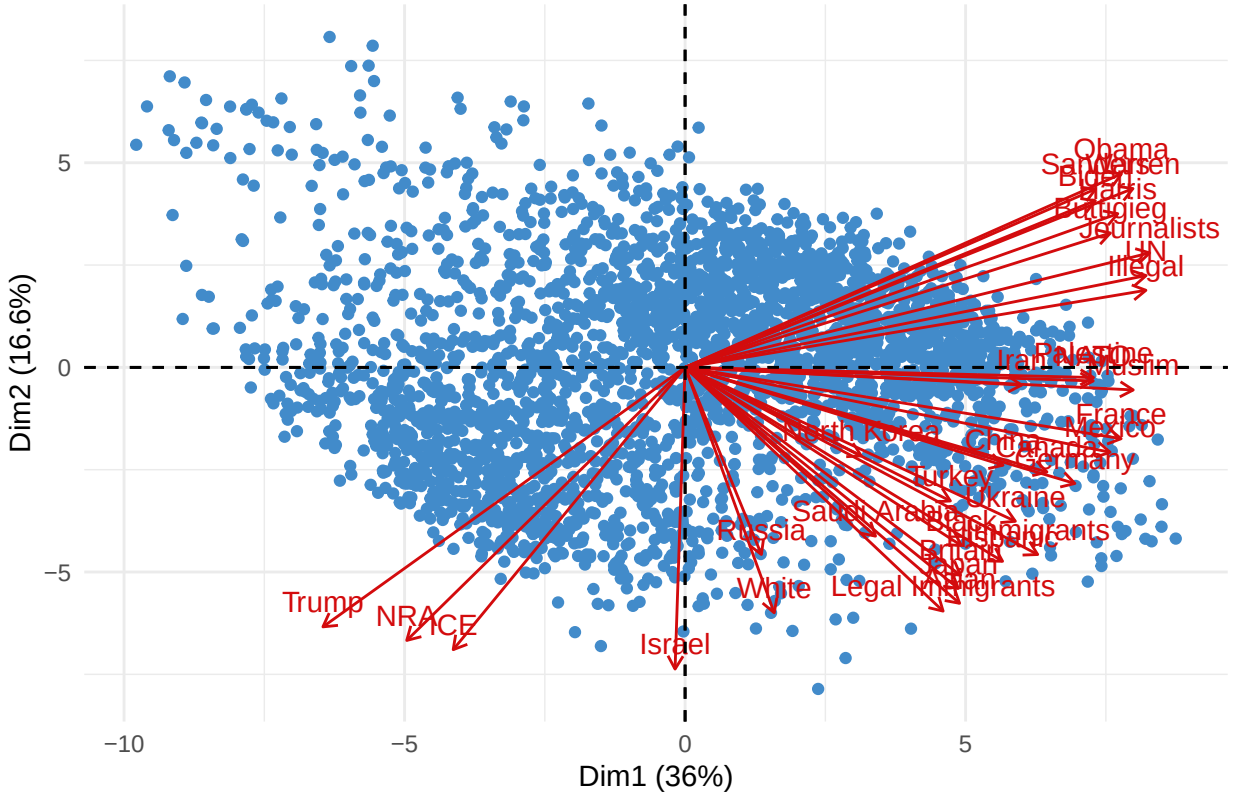
```
library(tidyverse)
library(here)
library(corrr)
library(amerika)
library(factoextra)
library(patchwork)
library(ggrepel)

anes <- read_rds("anes.rds")

pca_fit <- anes[,-36] %>%
  scale() %>%
  prcomp(); summary(pca_fit)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6     PC7
## Standard deviation     3.5491 2.4082 1.82408 1.30799 1.04776 0.86327 0.81279
## Proportion of Variance 0.3599 0.1657 0.09506 0.04888 0.03137 0.02129 0.01888
## Cumulative Proportion  0.3599 0.5256 0.62065 0.66953 0.70090 0.72219 0.74107
##                           PC8     PC9    PC10    PC11    PC12    PC13    PC14
```

```
## Standard deviation     0.78918 0.72418 0.69914 0.68242 0.66830 0.65582 0.63517
## Proportion of Variance 0.01779 0.01498 0.01397 0.01331 0.01276 0.01229 0.01153
## Cumulative Proportion  0.75886 0.77384 0.78781 0.80112 0.81388 0.82616 0.83769
##                           PC15    PC16    PC17    PC18    PC19    PC20    PC21
## Standard deviation     0.62471 0.6176 0.60486 0.5826 0.57746 0.56879 0.55645
## Proportion of Variance 0.01115 0.0109 0.01045 0.0097 0.00953 0.00924 0.00885
## Cumulative Proportion  0.84884 0.8597 0.87019 0.8799 0.88942 0.89866 0.90751
##                           PC22    PC23    PC24    PC25    PC26    PC27    PC28
## Standard deviation     0.55282 0.54164 0.53058 0.52288 0.51509 0.50415 0.48309
## Proportion of Variance 0.00873 0.00838 0.00804 0.00781 0.00758 0.00726 0.00667
## Cumulative Proportion  0.91624 0.92462 0.93267 0.94048 0.94806 0.95532 0.96199
##                           PC29    PC30    PC31    PC32    PC33    PC34    PC35
## Standard deviation     0.47198 0.45893 0.45325 0.43761 0.42993 0.40210 0.39188
## Proportion of Variance 0.00636 0.00602 0.00587 0.00547 0.00528 0.00462 0.00439
## Cumulative Proportion  0.96835 0.97437 0.98024 0.98571 0.99099 0.99561 1.00000
```

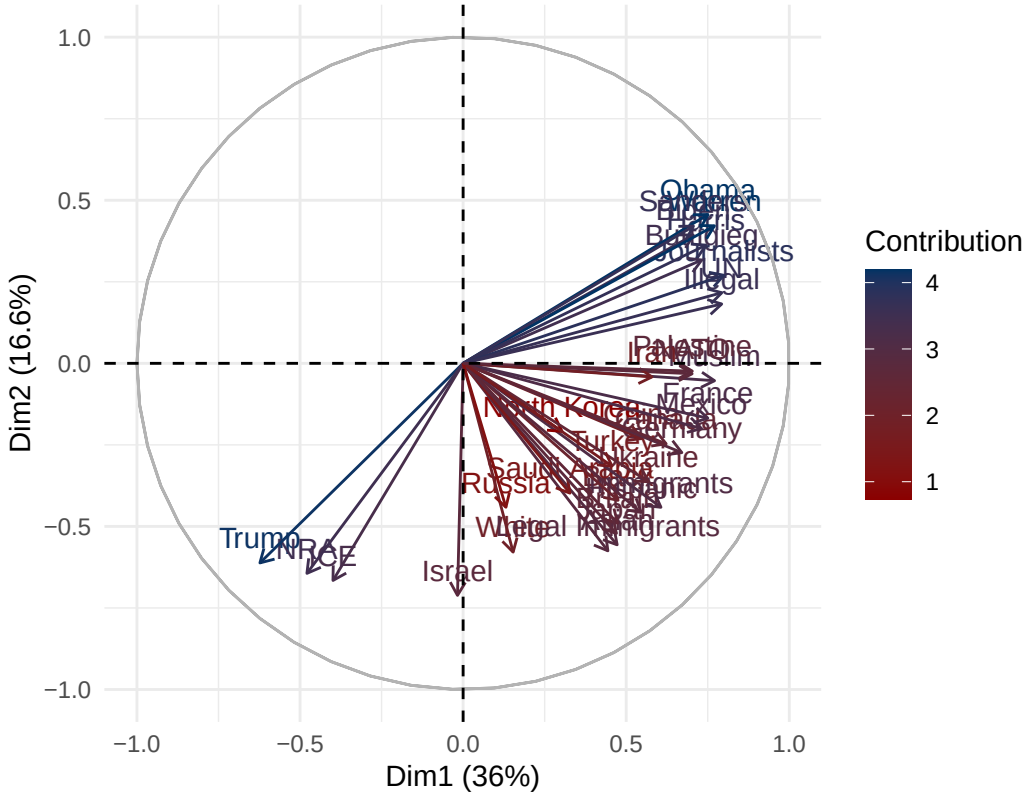
6. (20 points) Plot the feature contributions from each of the feeling thermometers in the first two dimensions (i.e., PC1 and PC2). Describe the patterns, groupings, and structure of the lower-dimensional projections in *substantive* terms.

```
pca_fit %>%
  fviz_pca_biplot(label = "var",
                  col.var = amerika_palettes$Republican[2],
                  col.ind = amerika_palettes$Democrat[3]) +
  labs(title = "") +
  theme_minimal()
```

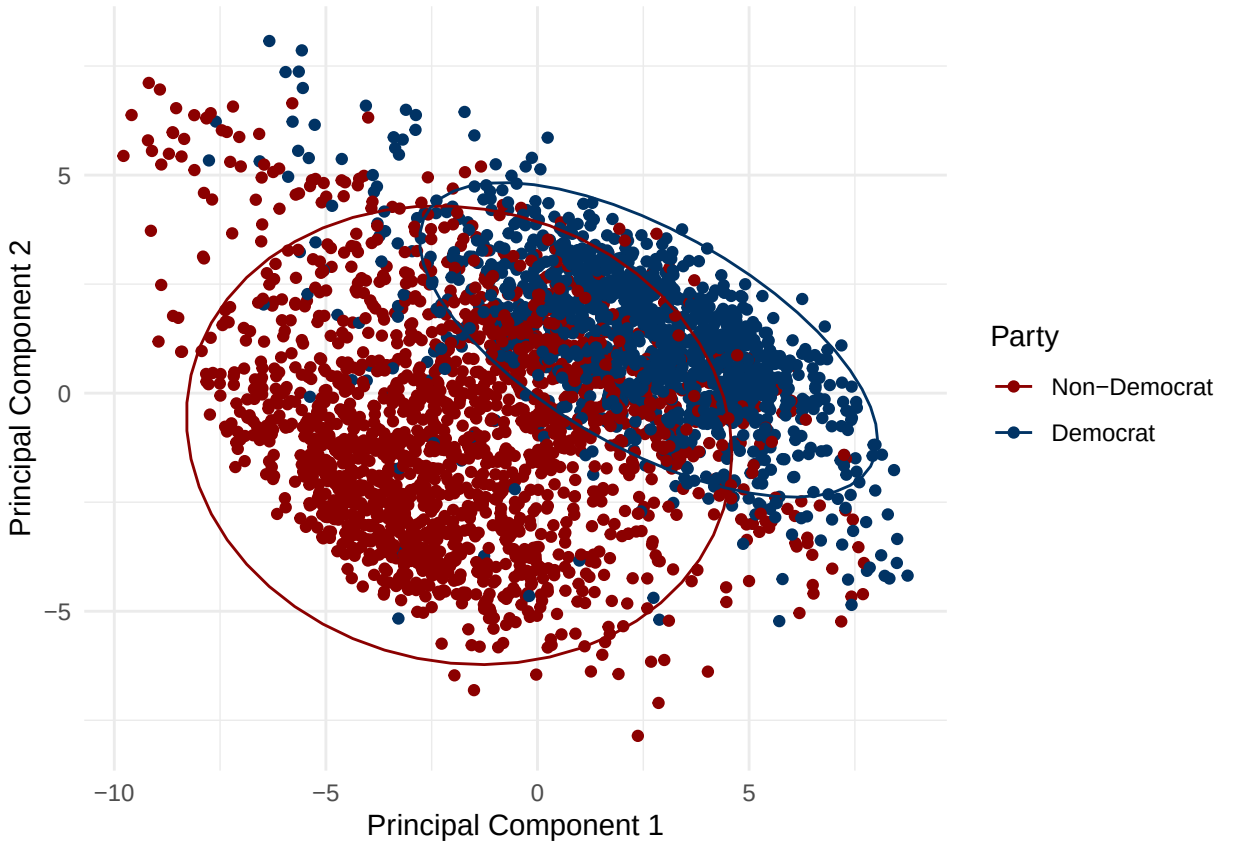



```
# feature loadings/contributions ("contrib")
pca_fit %>%
  fviz_pca_var(col.var = "contrib") +
```

```
  scale_color_gradient(high = amerika_palettes$Democrat[1],
                       low = amerika_palettes$Republican[1]) +
  labs(color = "Contribution",
       title = "") +
  theme_minimal()
```




```
# custom, full viz
anes %>%
  ggplot(aes(pca_fit$x[, 1],
             pca_fit$x[, 2],
             col = factor(democrat))) +
  geom_point() +
  stat_ellipse() +
  scale_color_manual(values=c(amerika_palettes$Republican[1],
                              amerika_palettes$Democrat[1]),
                     name="Party",
                     breaks=c("0", "1"),
                     labels=c("Non-Democrat", "Democrat")) +
  labs(x = "Principal Component 1",
       y = "Principal Component 2") +
  theme_minimal()
```



The first dimension (i.e. the first principal component) explains to 36% of total variances and dimension 2 explains 16.6% of the total variance.

From the biplot, we can see the feature projections of 35 original variables on the first two dimensions. Based on the signal of the first two dimensions, we can roughly divide the features in two 3 groups:

Group 1 (negative on both dimensions): Feeling thermometer of Trump, NRA(National Riffle Association), ICE(Immigration and Customs Enforcement) They are mainly conservative and republican-affiliated politician and organizations.

Group 2 (positive on both dimensions): Feeling thermometer of Obama, Biden, Harris, Warren, Buttigieg, Journalists, illegal, etc. They are mainly democrats and liberal politicians and media.

Group 3 (positive on dimension 1 but negative on dimension 2): Feeling thermometer of Russia, China, France, Mexico, Canada, Germany, Turkey, North Korea, Hispanic, Saudi Arabia, Britain, Japan, Asian, Legal, immigrants, White, Black, etc. They are mainly foreign countries.

Based on the features in each group, we can find that the first dimension are mainly about party affiliation and ideology. When the direction of features on the first dimension is positive, the feature is more democrat, liberal, or global, such as Obama and Canada. When the direction is negative, it's more conservative and republican, for example, Trump and National Riffle Association.

The second dimension are more about democrats against others–against both conservative party, conservative organizations, and foreign countries.

The feature loading of the feeling thermometer toward group 1 features such as Trump, NRA, and ICE are in the the opposite directions with the feeling thermometer toward group 2 features such as Obama, Biden, Harris, journalists, on both PC1 and PC2.

Group 2 features' projections are in the same direction on PC1 with group 3 features' projections, but are in the opposite directions on PC2.

Group 2 and group 3 features share the same direction on PC2, but are in the opposite direction on PC1.

Clustering

A Conceptual Problem

7. (10 points) What are the two properties required for a *hard* partitional solution, and when thus relaxed, give a *soft* partitional clustering solution? Be sure to answer this both formally (with mathematical notation) and substantively (with words). Then, give an example or two of each and how they relate to these two central properties of clustering.

For data $\{1, \ldots, n\}$ and each cluster $C_1 \ldots, C_k$, the two properties required for a hard partitional solution are:

1. Strict assignment: Each observation belongs to one of the kth clusters such that $C_1 \cup C_2 \ldots, C_k = \{1, \ldots, n\}$
2. No overlapping: Clusters are non-overlapping such that $C_k \cap C_k{}' = \phi \forall k \neq k'$

In the case of a soft partitional clustering solution, these two properties are relaxed. Theoretically, an observation can belong to more than one cluster (and even all of the clusters), with varying degrees.

The k-means clustering is an example of the two properties of hard partitional solutions. With the idea of achieving local optimum, the algorithm assigns each observation to one cluster whose centroid is closest. The final k clusters cover the full set without overlapping.

For the Gaussian mixture model, one of the most common soft partitioning solutions, observations are given probabilities of belonging to all clusters and then assigned based on probabilistic similarities. This probabilistic process thus enables overlapping with varying degrees and relaxes the two central properties of hard partitional clustering.

An Applied Problem

In this applied problem, you will again use the 2019 ANES data, but this time to explore the clustering solution from fitting a fuzzy c-means (FCM) algorithm to all feeling thermometers. As with the dimension reduction exercise, derive a clustering solution using *only* the feeling thermometers. The idea here is to explore whether attitudes on these issues, countries, and people map onto natural groupings between major American political parties.

8. (5 points) Load and scale the ANES *feeling thermometer* data.

```
library(tidyverse)
library(e1071)
library(ggplot2)
library(caret)

anes <- read_rds("anes.rds")
ftdata <- anes[,1:35] %>%
  scale()
```

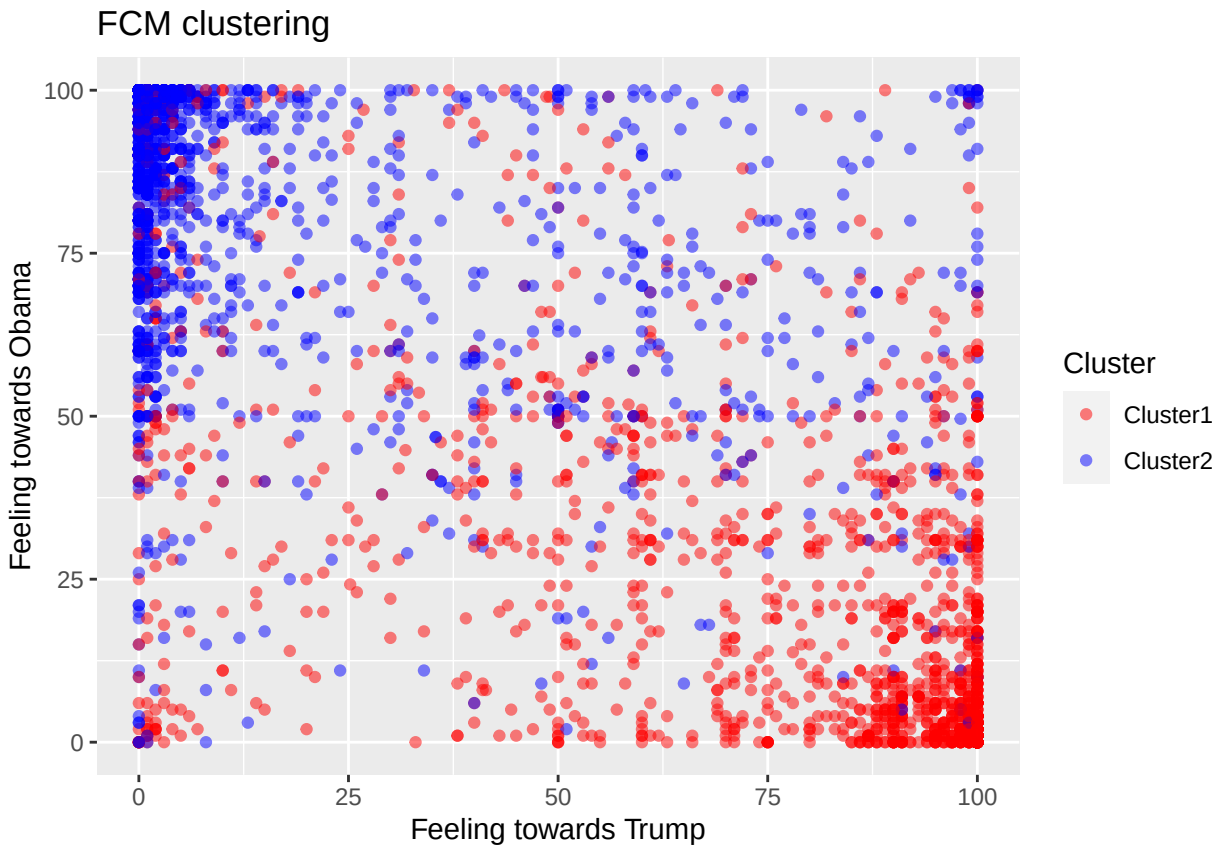
9. (5 points) Fit an FCM algorithm to the scaled data initialized at $k = 2$, driven by the assumption that party affiliation (Democrat or non-Democrat) underlies these data.

```
set.seed(1234)
anes.out <- anes %>%
  mutate(c2 = cmeans(ftdata, 2)$cluster) #fit FCM algorithm and store the cluster
```

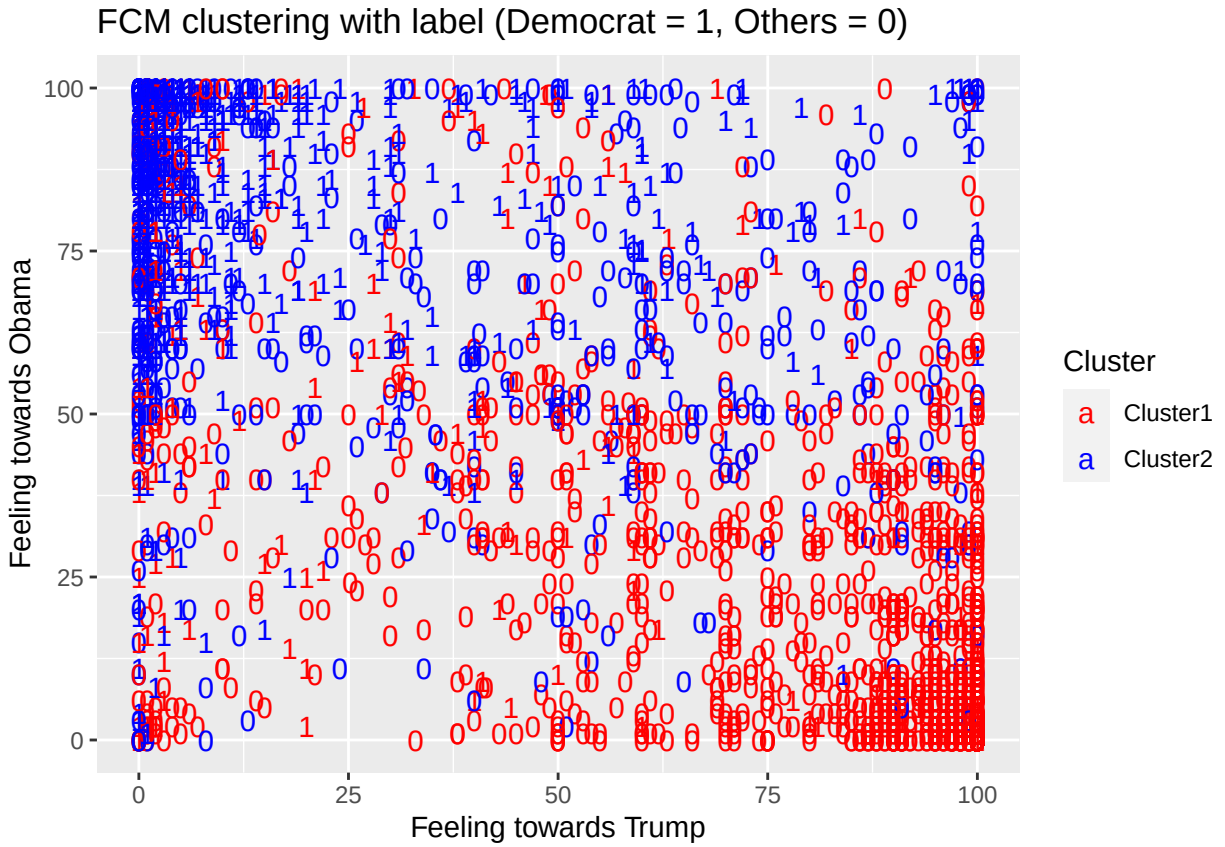
10. (15 points) Visualize the cluster scores from your FCM solution plotted over the range of feelings toward `Trump` and `Obama`, with data points colored by cluster assignment and also labeled by the respondent's true party affiliation (the `democrat` feature). As party wasn't included in your clustering solution,

what can you conclude based on these patterns? Is there a grouping pattern among observations along a partisan dimension, or isn't there? Do respondents group in expected ways (e.g., Trump supporters to the right and Obama supporters to the left)? Do cluster assignments align with the true party affiliation or not? How would you evaluate the effectiveness of FCM for this type of task?

```
#Without label
anes.out %>% ggplot(aes(x = Trump, y = Obama, color = factor(c2))) +
  geom_point(alpha = 0.5, size = 1.5) +
  scale_color_manual(values=c("red", "blue"),
                     name="Cluster",
                     breaks=c("1", "2"),
                     labels=c("Cluster1", "Cluster2")) +
  xlab("Feeling towards Trump") +  ylab("Feeling towards Obama") +
  labs(title = "FCM clustering")
```




```
#With label
anes.out %>% ggplot(aes(x = Trump, y = Obama, color = factor(c2))) +
  geom_text(label = anes.out$democrat, label.size = 0.05) +
  scale_color_manual(values=c("red", "blue"),
                     name="Cluster",
                     breaks=c("1", "2"),
                     labels=c("Cluster1", "Cluster2")) +
  xlab("Feeling towards Trump") +  ylab("Feeling towards Obama") +
  labs(title = "FCM clustering with label (Democrat = 1, Others = 0)")
```




```
#calculate the misalignment rate
misalign <- (count(anes.out %>% filter(c2 == 2 & democrat == 0))
             + count(anes.out %>% filter(c2 == 1 & democrat == 1)))/count(anes.out)
misalign
```

```
##           n
## 1 0.2066351
```

As shown in the two scatter plots: data points belonging to the first cluster (colored red) tend to gather around the lower right side, with high feelings towards Trump score and low feelings towards Obama score; whereas data points of the other cluster (colored blue) locate around the upper left side, with high feelings towards Obama score and low feelings towards Trump score. Therefore, these clusters suggest that there is a grouping patterns among observations along a partisan dimension, and as demostrated in the labelled plot, most of the cluster assignments echo with the true party affiliation. Most red colored points belong to the non-democrat class(democrate = 0), and most most blue colored points belong to the democrat class(democrat = 1). Besides qualitatively observing the pattern, we also calculated the misallignnmenr rate (where cluster assignment does not align with party affliation). The rate - 0.2066 - is relatively low.

In short, we think that FCM did a relatively good job in clustering the ANES dataset and answering the given question that "whether attitudes on these issues, countries, and people map onto natural groupings between major American political parties".