# Homework10

Patrick Foster

2025-04-06

### Load packages

```
library(tidyverse)
library(tidyclust)
library(tidymodels)
library(embed)
library(ggrepel)
library(patchwork)
```

# The 2022 ANES Pilot Study

## PCA Analysis

```
library(doParallel)
cl <- makePSOCKcluster(parallel::detectCores(logical = FALSE))
registerDoParallel(cl)
```

## Part A. Setup

```
data <- read_csv('https://gedeck.github.io/DS-6030/datasets/anes_pilot_2022_csv_20221214/anes_pilot_2022
```

### 1.1 Identify the feeling thermometer questions

Here we can use the select function from `dpylr` to only select the columns we want to analyze. Here we want to remove the timing columns, the ord columns, and the columns that contain black and white.

```
ft <- data %>%
  select(caseid,starts_with('ft'),jan6therm) %>%
  select(-contains('timing')) %>%
  select(-contains('white'),-contains('black'))
```

### 1.2 Filter out NA

Since the NAs were recorded as negative values we can use base R, to subset the dataframe to only include positive values and input NA values on the negatives. Then we can use the `drop_na()` function to remove any rows that contain NA values.

```
ft[ft < 0] <- NA
ft <- ft %>%
  drop_na()
```

```r
nrow(ft)
```

```
## [1] 1565
```

We now have approximately 1560 rows with 16 feeling thermometer questions.

## Part B PCA

Now we set up the PCA for the ft data.

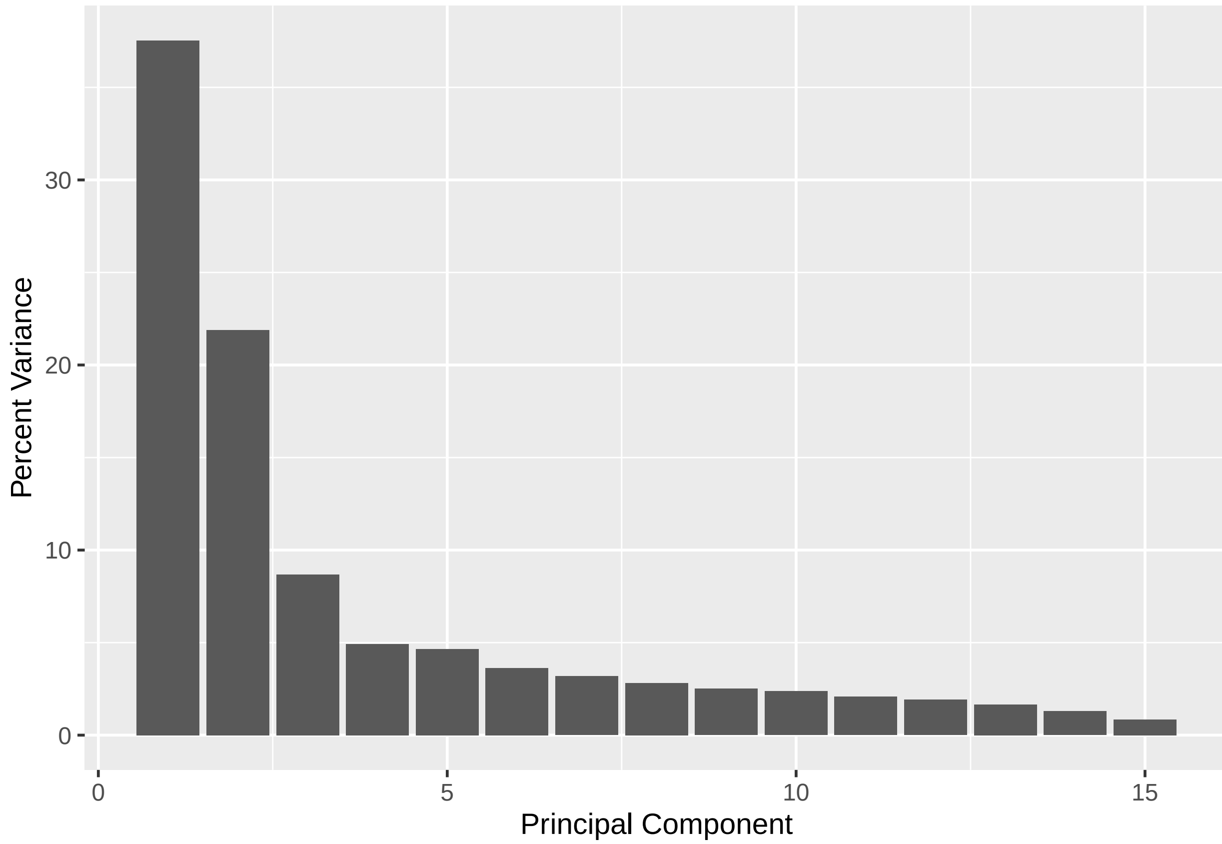
```r
pca_rec <- recipe(data=ft, formula = ~.) %>%
    update_role(caseid,new_role = "id") %>%
    step_normalize(all_numeric_predictors()) %>%
    step_pca(all_numeric_predictors())

ft_pca <- pca_rec %>%
    prep() %>%
    bake(new_data=NULL)
```

### 1.3 Create a Scree plot

```r
explained_variance <- pca_rec %>%
  prep() %>%
  pluck('steps',2) %>%
  tidy(type='variance')

explained_variance %>%
  pivot_wider(id_cols="component", names_from="terms", values_from="value")
```

```
## # A tibble: 15 x 5
##    component variance `cumulative variance` `percent variance`
##        <int>    <dbl>                 <dbl>              <dbl>
##  1         1    5.63                   5.63             37.5
##  2         2    3.28                   8.91             21.9
##  3         3    1.30                  10.2               8.68
##  4         4    0.739                 11.0               4.93
##  5         5    0.698                 11.7               4.65
##  6         6    0.546                 12.2               3.64
##  7         7    0.478                 12.7               3.18
##  8         8    0.424                 13.1               2.83
##  9         9    0.377                 13.5               2.51
## 10        10    0.356                 13.8               2.37
## 11        11    0.311                 14.1               2.07
## 12        12    0.287                 14.4               1.91
## 13        13    0.249                 14.7               1.66
## 14        14    0.194                 14.9               1.29
## 15        15    0.127                 15.0               0.845
## # i 1 more variable: `cumulative percent variance` <dbl>
```

```r
perc_variance <- explained_variance %>% filter(terms == "percent variance")
cum_perc_variance <- explained_variance %>% filter(terms == "cumulative percent variance")

ggplot(explained_variance, aes(x=component, y=value))+
  geom_bar(data = perc_variance, stat = "identity")+
  labs(x="Principal Component",y="Percent Variance")
```




An argument could be made for either 2 or 3 principal components, I am going to use 2 principal components in order as there is a definite "elbow" located there. After three principal components the amount of variance is fairly constant and small.

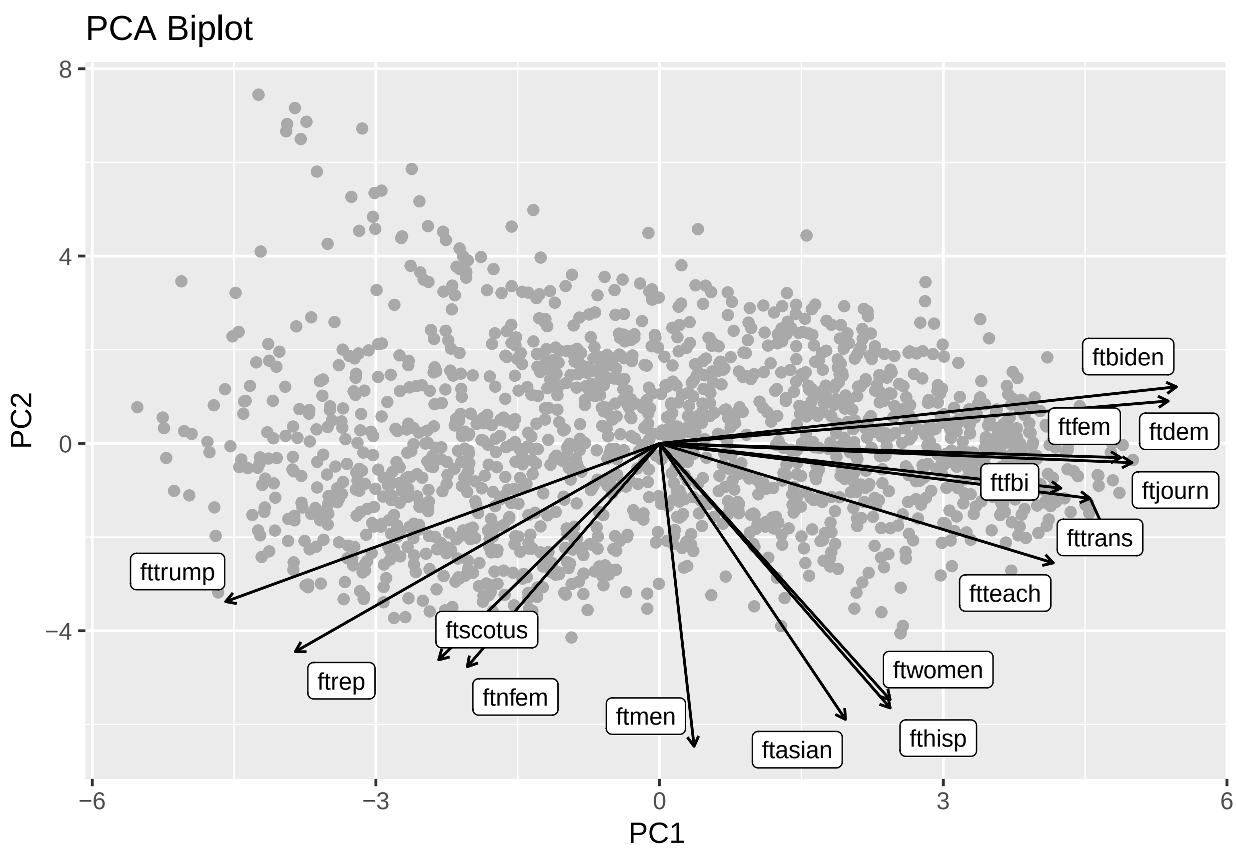
### 1.4 Create A bi-plot

```
loadings <- pca_rec %>%
  prep() %>%
  pluck("steps",2) %>%
  tidy(type = "coef") %>%
  pivot_wider(id_cols = "terms", names_from = "component", values_from = "value")

loadings %>%
  select("terms","PC1","PC2") %>%
  arrange(desc(PC1))
```

```
## # A tibble: 15 x 3
##    terms        PC1     PC2
##    <chr>      <dbl>   <dbl>
##  1 ftbiden    0.364  0.0806
##  2 ftdem      0.359  0.0603
##  3 ftjourn    0.333 -0.0274
##  4 ftfem      0.325 -0.0202
##  5 fttrans    0.304 -0.0782
##  6 ftfbi      0.283 -0.0637
##  7 ftteach    0.278 -0.170
```

```
##  8 fthisp    0.163  -0.377
##  9 ftwomen   0.163  -0.365
## 10 ftasian   0.131  -0.393
## 11 ftmen     0.0243 -0.431
## 12 ftnfem   -0.136  -0.318
## 13 ftscotus -0.156  -0.308
## 14 ftrep    -0.257  -0.296
## 15 fttrump  -0.306  -0.225
```

```
scale <-15
ggplot(ft_pca, aes(x=PC1, y=PC2))+
  geom_point(color= "darkgrey")+
  geom_segment(data=loadings,
               aes(xend=scale*PC1, yend=scale*PC2,x=0,y=0),
               arrow = arrow(length = unit(.15,"cm")))+
  geom_label_repel(data=loadings,
             aes(x=scale*PC1,y=scale*PC2,label=terms),
             size = 3, max.overlaps = 20)+
  labs(title = "PCA Biplot")
```




**1.5 Interpret the two components.**

Component 1 seems to be the traditional left-right partisan split on the Us electorate, The ftbiden, ftdem, ftfem, are all the most positive PC1, whereas the fttrump and ftrep are the most negative values of PC1. PC2 is harder to quantify.

```r
loadings %>%
  select("terms","PC1","PC2") %>%
  arrange(desc(PC2))
```

```
## # A tibble: 15 x 3
##    terms        PC1     PC2
##    <chr>      <dbl>   <dbl>
##  1 ftbiden   0.364   0.0806
##  2 ftdem     0.359   0.0603
##  3 ftfem     0.325  -0.0202
##  4 ftjourn   0.333  -0.0274
##  5 ftfbi     0.283  -0.0637
##  6 fttrans   0.304  -0.0782
##  7 ftteach   0.278  -0.170
##  8 fttrump  -0.306  -0.225
##  9 ftrep    -0.257  -0.296
## 10 ftscotus -0.156  -0.308
## 11 ftnfem   -0.136  -0.318
## 12 ftwomen   0.163  -0.365
## 13 fthisp    0.163  -0.377
## 14 ftasian   0.131  -0.393
## 15 ftmen     0.0243 -0.431
```

Looking at the values of PC2 arranged in descending order it seems that PC2 is more of a distinguisher of group types, where we see that the most negative values are women, hisp, asian, and men.

## Part C. Explore the dataset

### 1.6 Map respondents profile

```r
ft_profile <- data %>%
  select(caseid,gender,educ,marstat)
```

```r
ft_profile <- ft_profile %>%
  mutate(
    gender = factor(gender,levels = c(1,2),labels = c("Male","Female")),
    educ = factor(educ,levels=c(1,2,3,4,5,6),labels = c("No Hs","High School Graduate","Some College",
    marstat = factor(marstat, levels = c(1,2,3,4,5,6),labels=c("Married","Seperated","Divorced","Widowe
  )
```

```r
head(ft_profile)
```

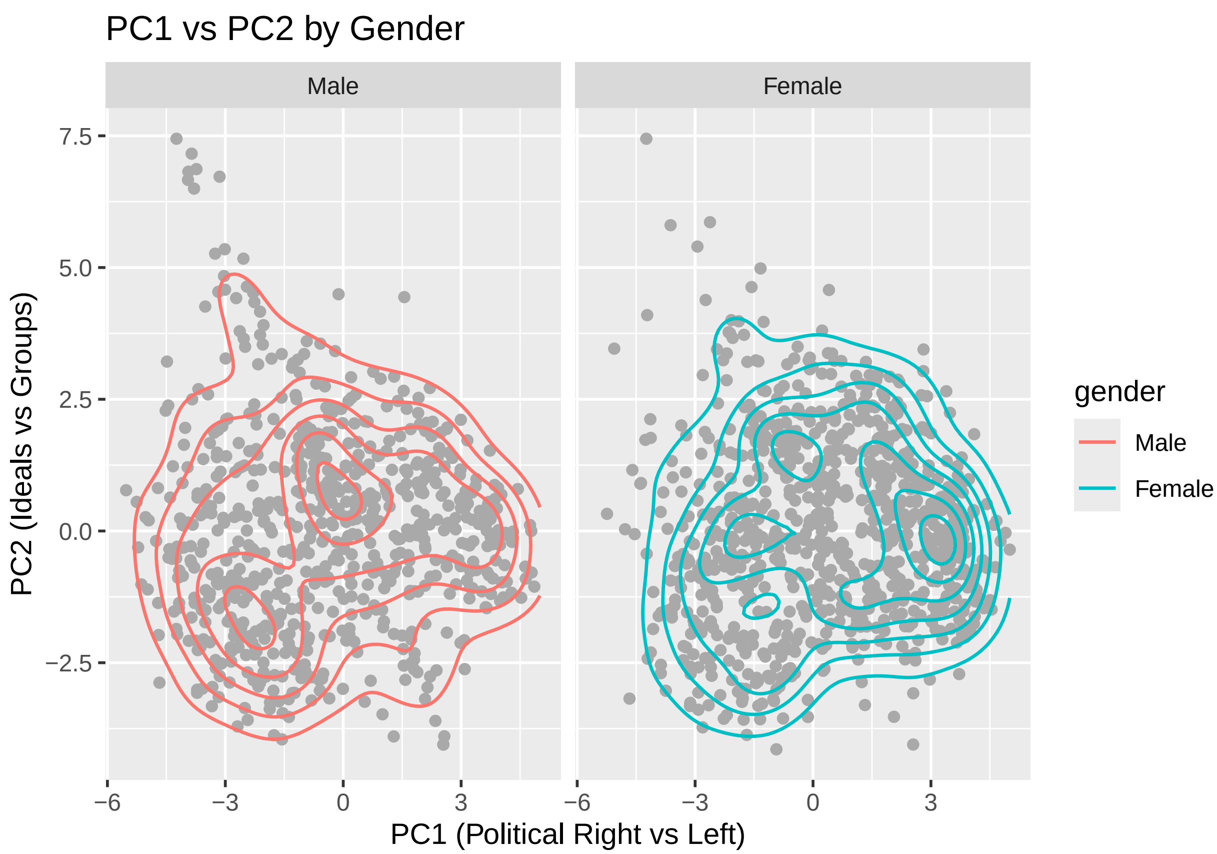
```
## # A tibble: 6 x 4
##   caseid gender educ                 marstat
##    <dbl> <fct>  <fct>                <fct>
## 1      1 Male   2-Year               Divorced
## 2      2 Female Post Grad            Divorced
## 3      3 Male   4-Year               Divorced
## 4      4 Male   High School Graduate Married
## 5      5 Female 4-Year               Married
## 6      6 Female Post Grad            Never Married
```

```r
ft_pca <- ft_pca %>%
  inner_join(ft, by = "caseid")
```

```r
ft_profile <- ft_profile %>%
  inner_join(ft_pca, by="caseid")

head(ft_profile)
```

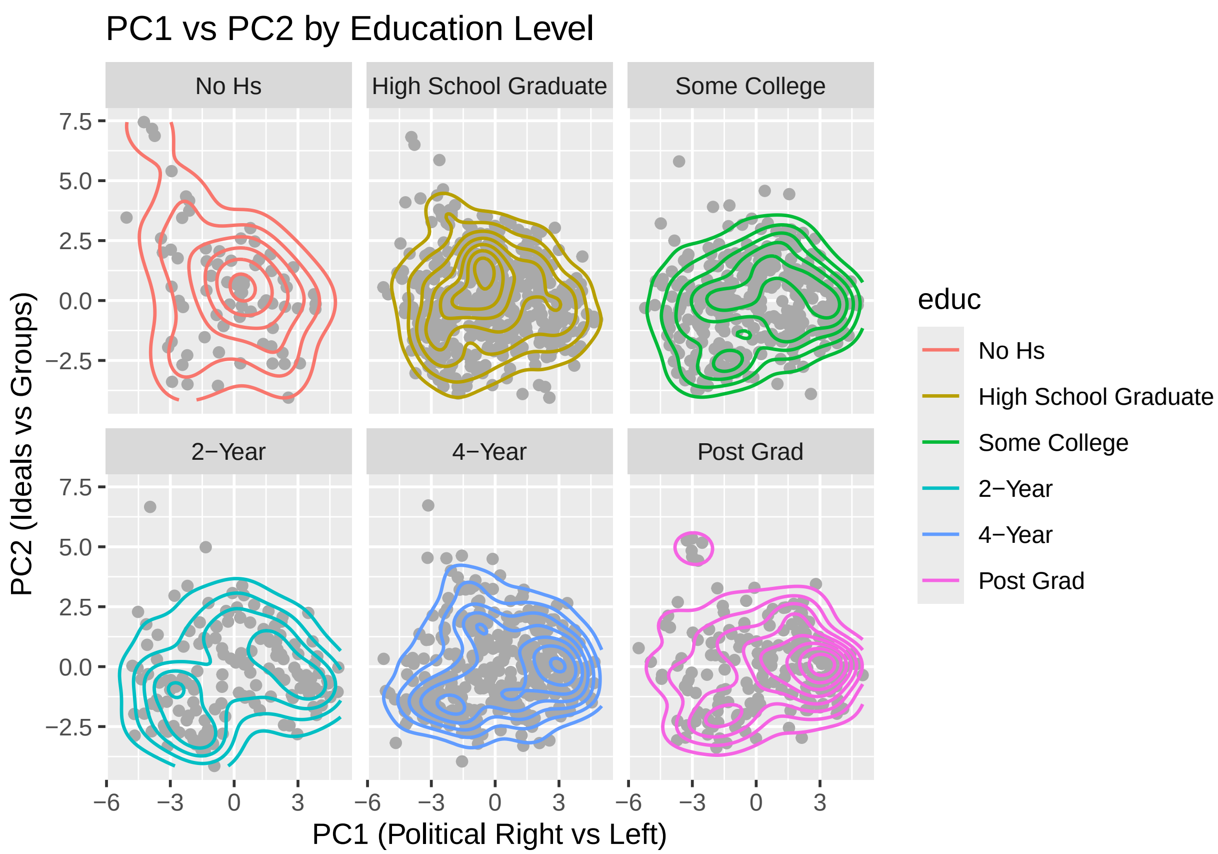
```
## # A tibble: 6 x 24
##   caseid gender educ         marstat    PC1    PC2    PC3     PC4     PC5 fthisp
##    <dbl> <fct>  <fct>        <fct>    <dbl>  <dbl>  <dbl>   <dbl>   <dbl>  <dbl>
## 1      1 Male   2-Year       Divorc~ -4.52   2.28   0.157  0.121  -1.13       32
## 2      2 Female Post Grad    Divorc~ -1.71  -3.00  -1.01  -1.62    0.575      74
## 3      3 Male   4-Year       Divorc~ -0.597  0.738 -1.64   0.0361  0.290      51
## 4      4 Male   High School~ Married  3.14   0.259  0.904 -1.08    0.0360     87
## 5      5 Female 4-Year       Married  4.54  -1.49  -0.146  0.105   0.154     100
## 6      6 Female Post Grad    Never ~  2.92   0.310  1.31   0.973  -0.283     100
## # i 14 more variables: ftasian <dbl>, ftfbi <dbl>, ftscotus <dbl>,
## #   fttrump <dbl>, ftbiden <dbl>, ftdem <dbl>, ftrep <dbl>, ftteach <dbl>,
## #   ftfem <dbl>, ftnfem <dbl>, ftjourn <dbl>, ftmen <dbl>, ftwomen <dbl>,
## #   fttrans <dbl>
```

```r
ft_profile %>%
  ggplot(aes(x=PC1,y=PC2))+
  geom_point(color = "darkgrey")+
  geom_density2d(aes(color=gender),linewidth=.6)+
  facet_wrap(~gender)+
  labs(title= "PC1 vs PC2 by Gender",
       x = "PC1 (Political Right vs Left)",
       y = "PC2 (Ideals vs Groups)")
```



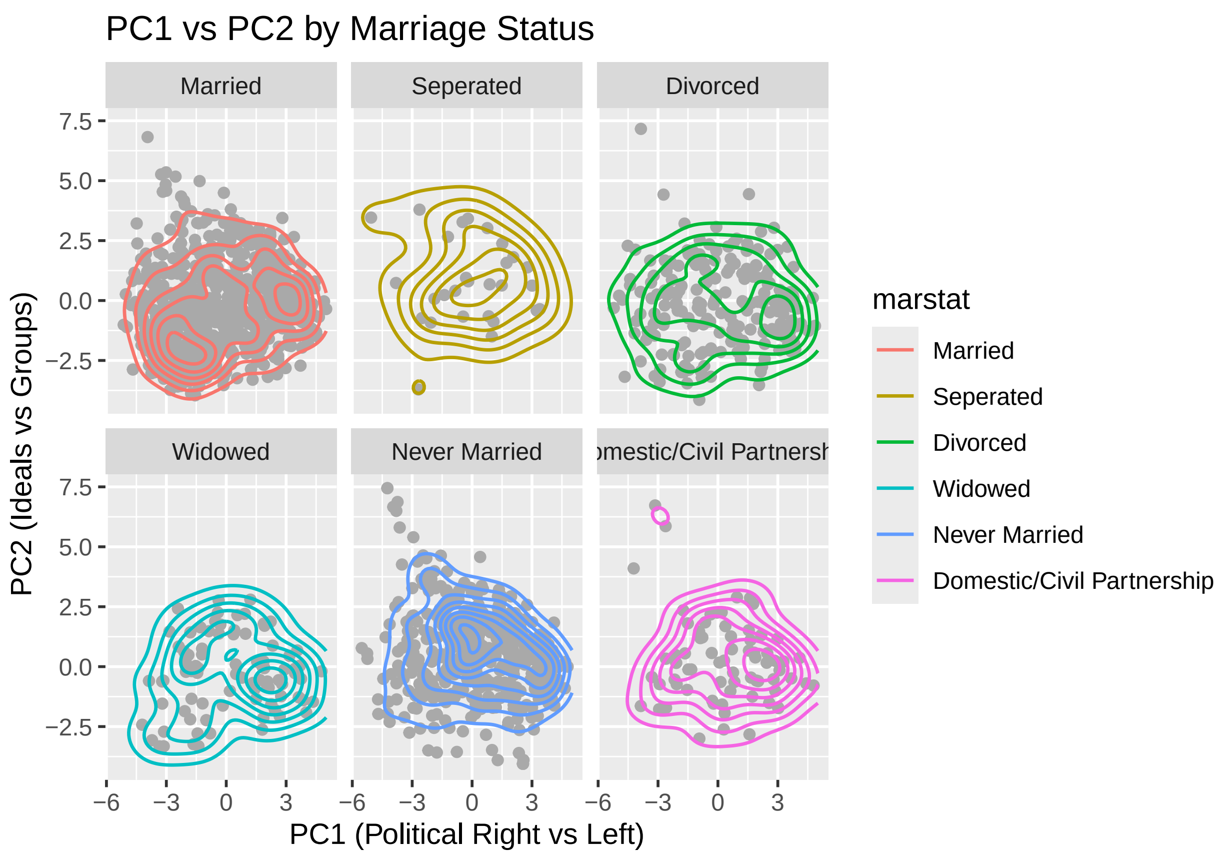

Here we see that in general the females are skewed more to the right of PC1, which is the political left, when compared to the Males. Also There is a distinct pull up on PC2 for males, which we determined was towards ideals vs groups.

```
ft_profile %>%
  ggplot(aes(x=PC1,y=PC2))+
  geom_point(color = "darkgrey")+
  geom_density2d(aes(color=educ),linewidth=.6)+
  facet_wrap(~educ)+
  labs(title= "PC1 vs PC2 by Education Level",
       x = "PC1 (Political Right vs Left)",
       y = "PC2 (Ideals vs Groups)")
```




It seems that as education level increases generally the groups skew more towards the political left.

```
ft_profile %>%
  ggplot(aes(x=PC1,y=PC2))+
  geom_point(color = "darkgrey")+
  geom_density2d(aes(color=marstat),linewidth=.6)+
  facet_wrap(~marstat)+
  labs(title= "PC1 vs PC2 by Marriage Status",
       x = "PC1 (Political Right vs Left)",
       y = "PC2 (Ideals vs Groups)")
```




Interestingly the married group has two distinct peaks, one for political left vs political right. The never married group has one peak it's pretty much right in the middle!

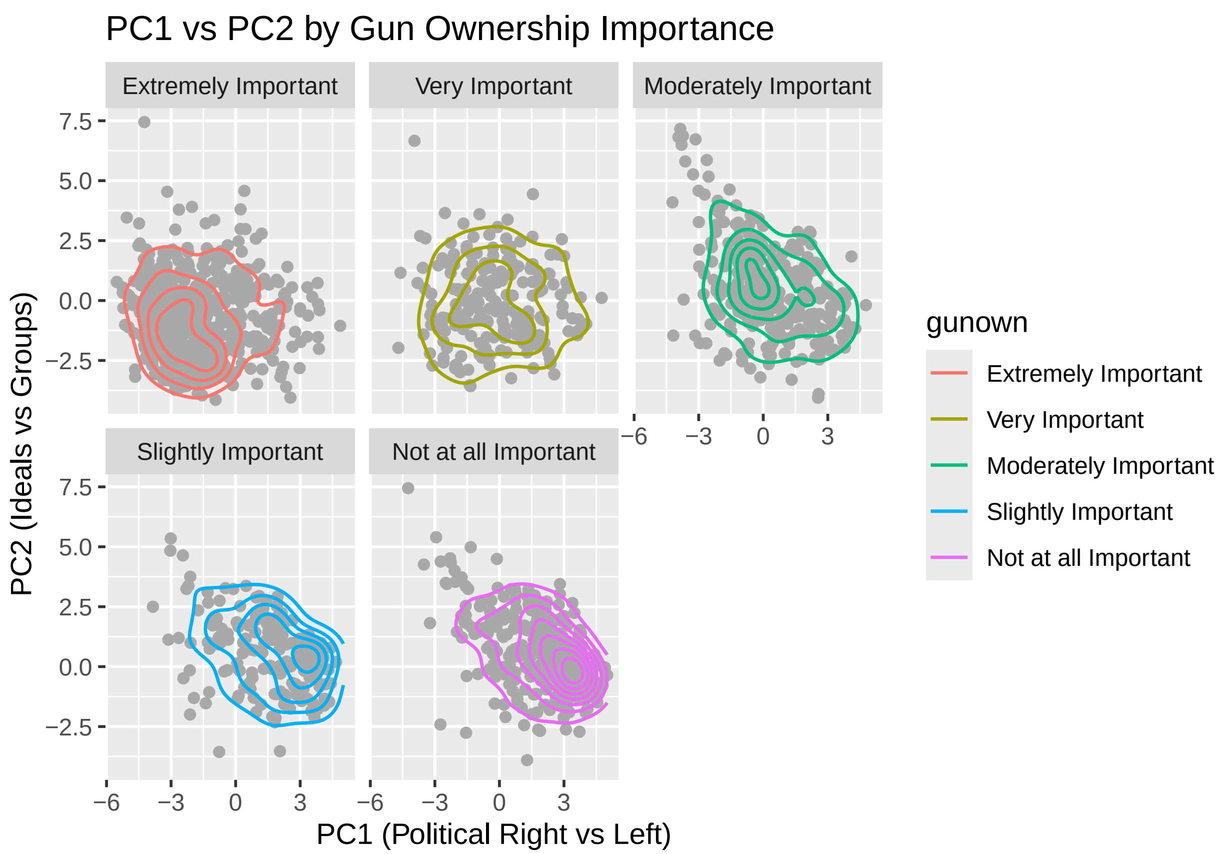
### 1.7 Gun Ownership and PCA Analysis

My hypothesis for gun ownership is that those who say that gun ownership rights are extremely important will have a very negative value of PCA 1, and a negative value of PCA2, so they will be on the political right and have stronger opinions on groups. Whereas the opposite will hold true for those who say it is not important at all.

```
guns <- data %>%
  select(caseid, gunown) %>%
  mutate(
    gunown = factor(gunown, levels=c(1,2,3,4,5),labels=c("Extremely Important","Very Important","Modera
  )
```

```
ft_profile_gun <- ft_profile %>%
  inner_join(guns,by="caseid")
```

```
ft_profile_gun %>%
  ggplot(aes(x=PC1,y=PC2))+
  geom_point(color = "darkgrey")+
  geom_density2d(aes(color=gunown),linewidth=.6)+
  facet_wrap(~gunown)+
  labs(title= "PC1 vs PC2 by Gun Ownership Importance",
       x = "PC1 (Political Right vs Left)",
       y = "PC2 (Ideals vs Groups)")
```




Here we see that the the political left vs political right is spot on. PCA 1 does a very good job at separating this classes. Those who classify Gun Ownership as "Extremely Important" are much more likely to have a lower PCA1 value when compared to those in the "Not at all important".

## Clustering

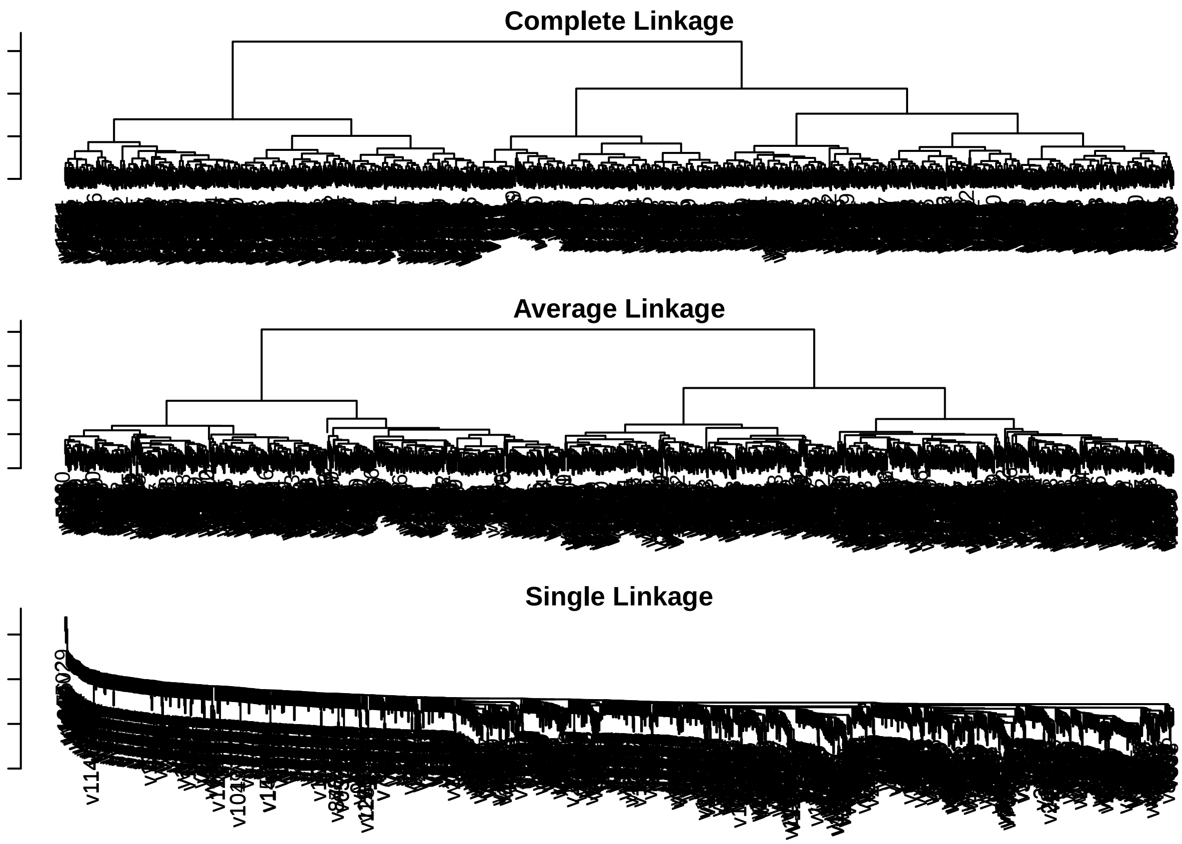
## Part A

### 2.1 Create a hierarchical clustering.

Using the code from class we can visualize the different clustering methods from hierarchical clustering.

```
get_hier_clust_fit <- function(linkage_method) {
  formula = ~.
    hier_ft <- hier_clust(linkage_method=linkage_method) %>%
        set_engine("stats") %>%
        set_mode("partition")
    hier_model <- hier_ft %>% fit(formula, data=ft)
    hier_model
}

par(mfrow=c(3, 1), mar=c(1, 1, 1, 1))
plot(get_hier_clust_fit("complete")$fit,
    main="Complete Linkage", xlab="", sub="", ylab="")
plot(get_hier_clust_fit("average")$fit,
    main="Average Linkage", xlab="", sub="", ylab="")
plot(get_hier_clust_fit("single")$fit,
```

```
    main="Single Linkage", xlab="", sub="", ylab="")
```



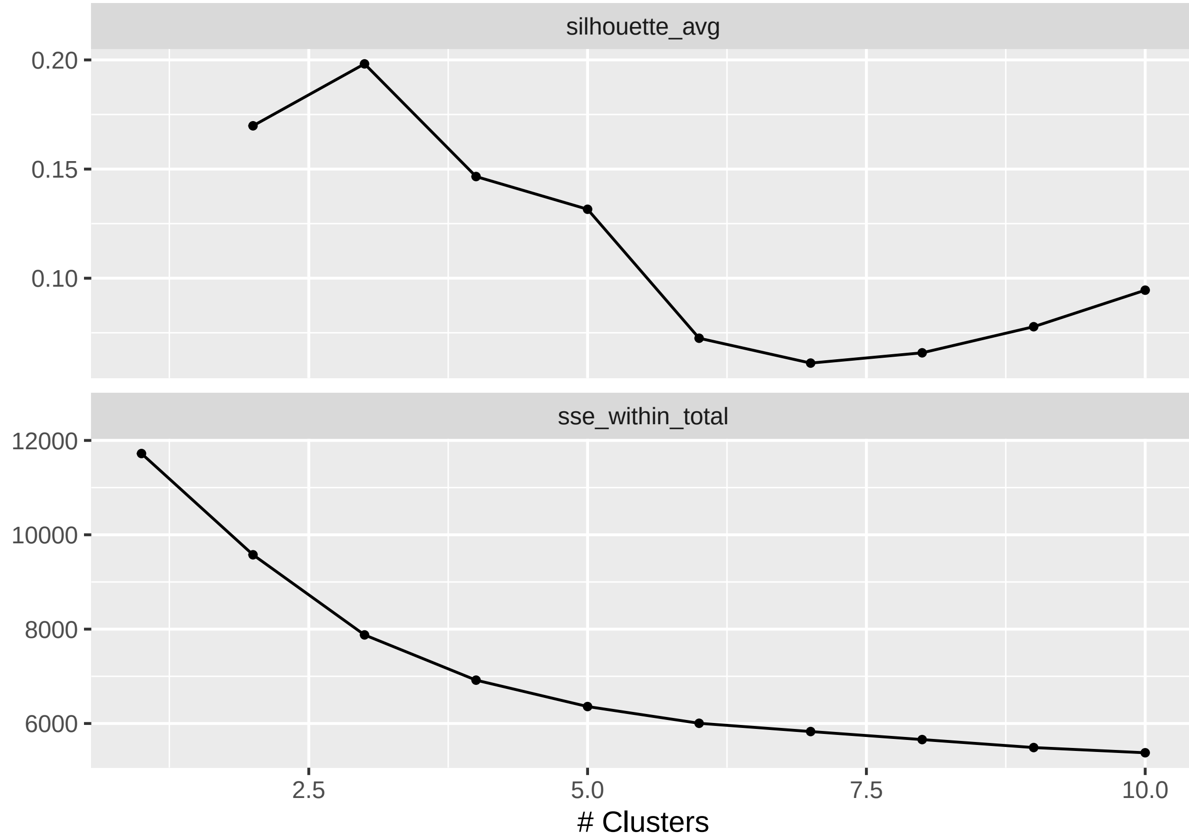
Judging from the above graph I am going to choose the complete linkage for the balanced clustering.

Now we can tune a model to choose the clustering depth.

```
formula <- (~.)

rec_ft <- recipe(formula,data=ft) %>%
  update_role(caseid,new_role = "id") %>%
  step_normalize(all_predictors())

hier_ft <- hier_clust(num_clusters = tune()) %>%
  set_engine("stats") %>%
  set_mode("partition")

hier_wf <- workflow() %>%
  add_recipe(rec_ft) %>%
  add_model(hier_ft)
```

```
registerDoSEQ()
folds <- vfold_cv(ft,v=2)
grid <- tibble(num_clusters=1:10)

result <- tune_cluster(hier_wf,resamples = folds,grid = grid,
                       metrics = cluster_metric_set(sse_within_total,silhouette_avg))
registerDoParallel(cl)
```

```r
autoplot(result)
```




Here we see that the optimal number of clusters is 3.

### 2.2 k-means clustering

We are going to create a k-means cluster using the tidyclust package.

First we define a workflow.

```r
formula <- (~.)

rec_ft <- recipe(formula,data=ft) %>%
  update_role(caseid,new_role = "id") %>%
  step_normalize(all_predictors())

kmeans_ft <- k_means(num_clusters=5) %>%
    set_engine("stats") %>%
    set_mode("partition")

kmeans_wf <- workflow() %>%
    add_recipe(rec_ft) %>%
    add_model(kmeans_ft)
```

Then we can fit the model.

```r
kmeans_model <- fit(kmeans_wf, data = ft)
```

Now we can get the predicted cluster from the model.

```
ft_kmeans <- augment(kmeans_model,new_data = ft) %>%
  select(caseid, .pred_cluster)
```

Here I join it to all the other data.

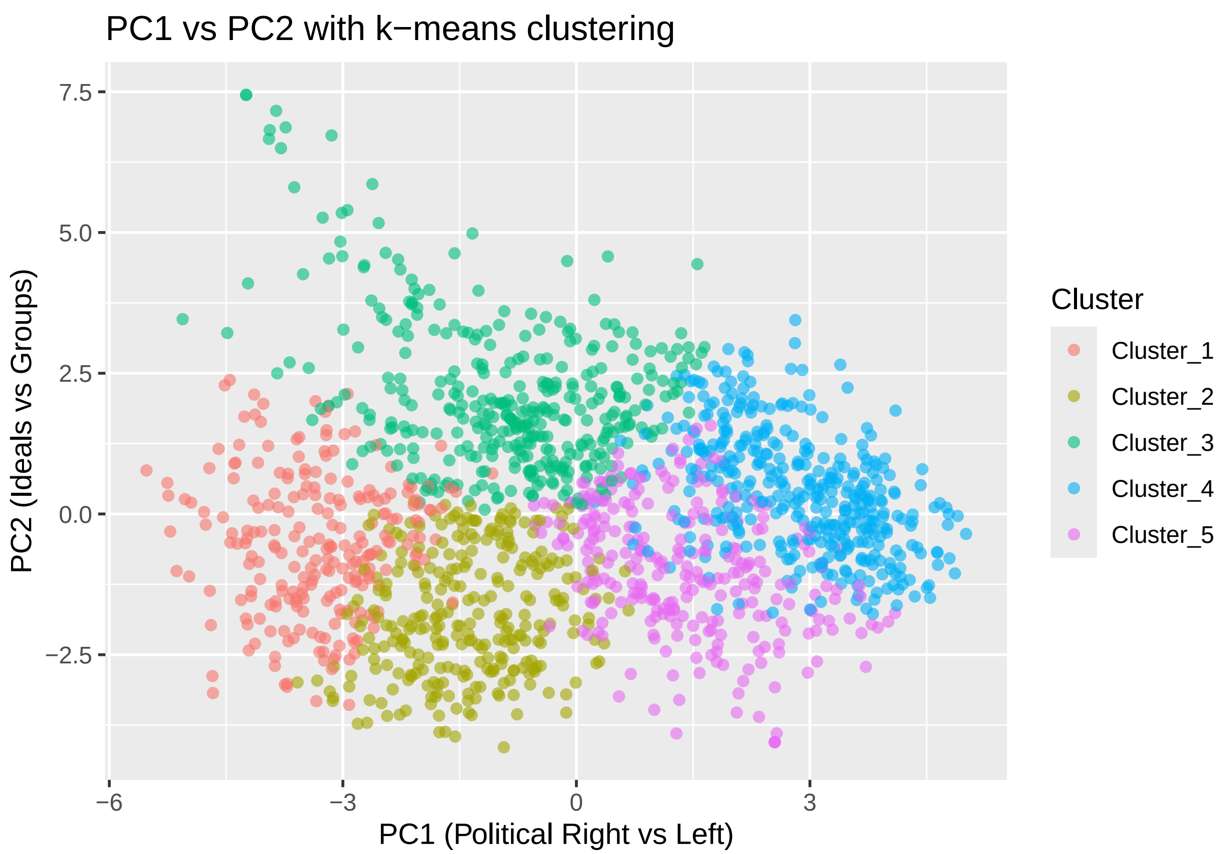
```
ft_kmeans <- ft_kmeans %>%
  inner_join(ft_profile,by="caseid")
```

```
head(ft_kmeans,3)
```

```
## # A tibble: 3 x 25
##   caseid .pred_cluster gender educ   marstat    PC1    PC2    PC3     PC4    PC5
##    <dbl> <fct>         <fct>  <fct>  <fct>    <dbl>  <dbl>  <dbl>   <dbl>  <dbl>
## 1      1 Cluster_1     Male   2-Year Divorc~ -4.52   2.28   0.157  0.121  -1.13
## 2      2 Cluster_2     Female Post ~ Divorc~ -1.71  -3.00  -1.01  -1.62    0.575
## 3      3 Cluster_3     Male   4-Year Divorc~ -0.597  0.738 -1.64   0.0361  0.290
## # i 15 more variables: fthisp <dbl>, ftasian <dbl>, ftfbi <dbl>,
## #   ftscotus <dbl>, fttrump <dbl>, ftbiden <dbl>, ftdem <dbl>, ftrep <dbl>,
## #   ftteach <dbl>, ftfem <dbl>, ftnfem <dbl>, ftjourn <dbl>, ftmen <dbl>,
## #   ftwomen <dbl>, fttrans <dbl>
```

Now we can plot the PC1/PC2 scatter and see the cluster groupings.

```
ft_kmeans %>%
  ggplot(aes(x=PC1,y=PC2))+
  geom_point(aes(color = .pred_cluster), alpha=.6)+
    labs(title= "PC1 vs PC2 with k-means clustering",
       x = "PC1 (Political Right vs Left)",
       y = "PC2 (Ideals vs Groups)",
       color = "Cluster")
```




Here we see that the clusters are definitely separating along some line. They have different clusters for different areas of the scatter plot. The clusters seem to further divide the political spectrum along the PC1/PC2 Axis. Into a Far Right, Middle Right, True Moderate, Middle Left, and Far left.

- Cluster1 : Orange : True-Moderate
- Cluster2 : Yellow : Middle-Right
- Cluster3 : Green : Far-Left
- Cluster4 : Blue : Far-Right
- Cluster5 : Purple: Middle-Left

For further analysis we can create a new variable `Cluster` with this factor in mind.

```
ft_kmeans <- ft_kmeans %>%
  mutate(
    Cluster = factor(.pred_cluster,
   levels=c('Cluster_1','Cluster_2','Cluster_3','Cluster_4','Cluster_5'),
   labels=c("True-Moderate","Middle-Right","Far-Left","Far-Right","Middle-Left"))
  )
```

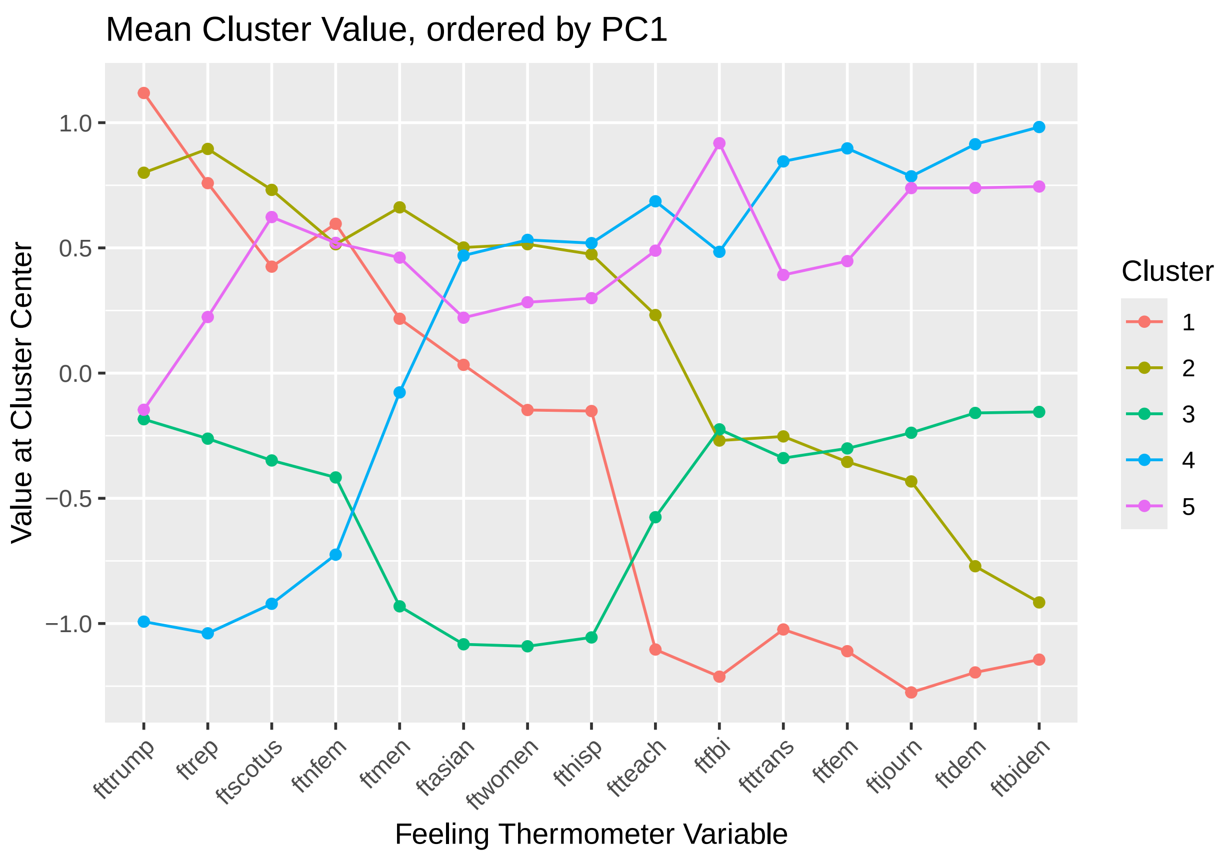
```
tidy(kmeans_model)
```

```
## # A tibble: 5 x 18
##   fthisp ftasian  ftfbi ftscotus fttrump ftbiden  ftdem  ftrep ftteach  ftfem
##    <dbl>    <dbl> <dbl>    <dbl>   <dbl>   <dbl> <dbl>  <dbl>   <dbl>  <dbl>
```

```
## 1 -0.151  0.0330 -1.21     0.425   1.12   -1.14  -1.20   0.759  -1.10  -1.11
## 2  0.475  0.502  -0.269    0.732   0.800  -0.916 -0.771  0.895   0.232 -0.355
## 3 -1.06  -1.08   -0.225   -0.349  -0.184  -0.155 -0.159 -0.262  -0.575 -0.301
## 4  0.519  0.470   0.484   -0.921  -0.992   0.983  0.914 -1.04    0.686  0.897
## 5  0.299  0.221   0.918    0.623  -0.146   0.745  0.740  0.224   0.489  0.447
## # i 8 more variables: ftnfem <dbl>, ftjourn <dbl>, ftmen <dbl>, ftwomen <dbl>,
## #   fttrans <dbl>, size <int>, withinss <dbl>, cluster <fct>
```

In order to create the Parallel Coordinate plot we have to use the tidy command on the kmeans_model. In order to better vizualize the class seperations I ordered the feeling thermometers by PC1.

```r
tidy(kmeans_model) %>%
  pivot_longer(-c(cluster,size,withinss)) %>%
  left_join(loadings %>% select(terms,PC1), by = c("name"="terms")) %>%
  mutate(name=fct_reorder(name,PC1)) %>%
  ggplot(aes(x = name, y = value, group = factor(cluster), color = factor(cluster))) +
  geom_point() +
  geom_line() +
  labs(
    title = "Mean Cluster Value, ordered by PC1",
    x = "Feeling Thermometer Variable",
    y = "Value at Cluster Center",
    color = "Cluster"
  ) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

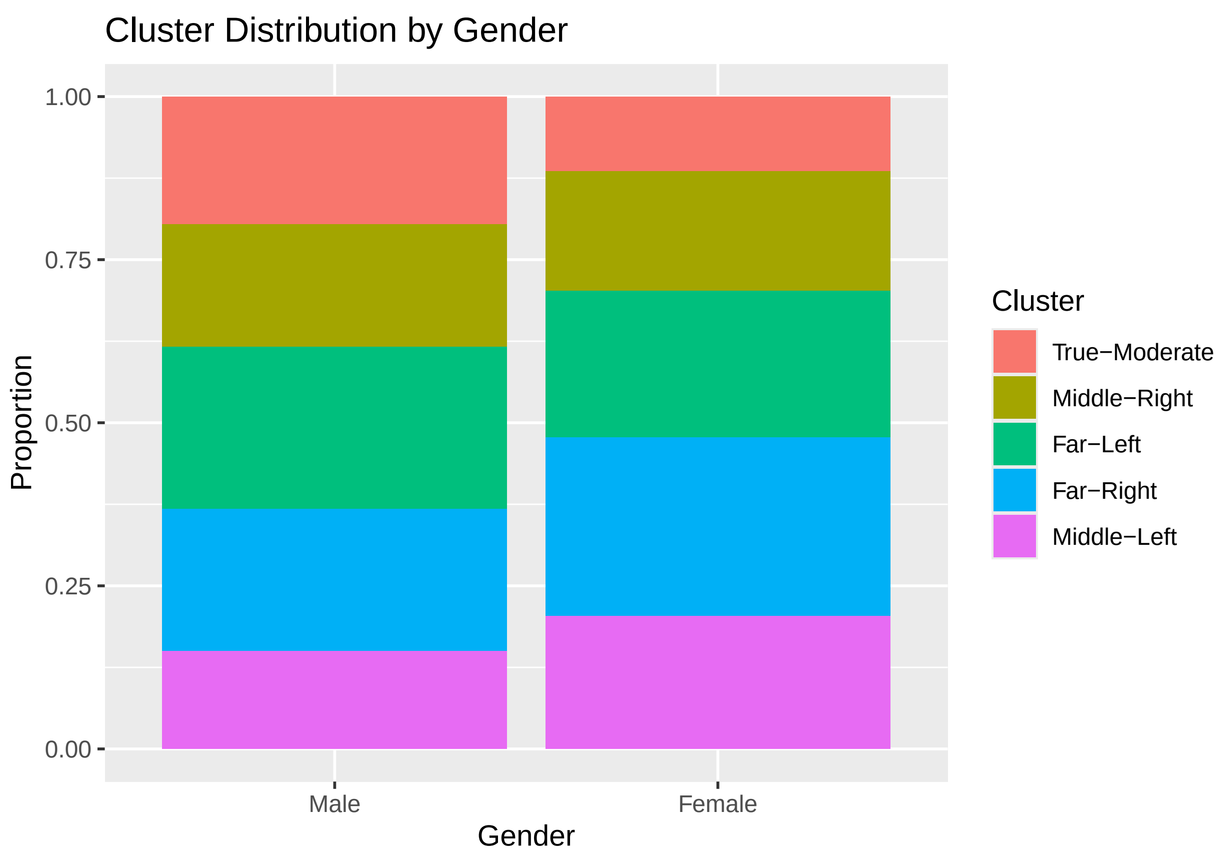



Here we see that the cluster do tend to help separate on the political spectrum.

# C Explore the Dataset

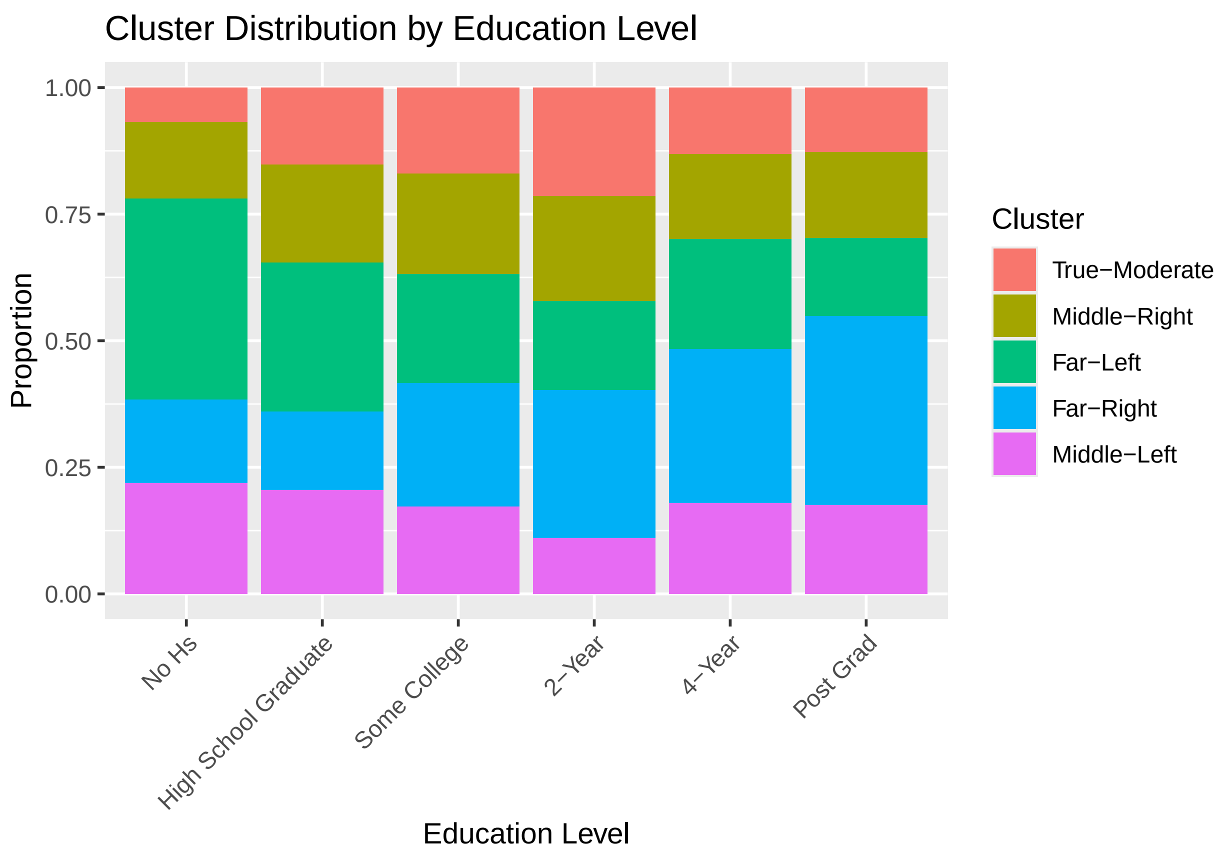
## 2.3 Characterize the clusters

```
ft_kmeans %>%
  ggplot(aes(x=gender, fill = Cluster))+
  geom_bar(position = "fill")+
  labs(
    title = "Cluster Distribution by Gender",
    x="Gender",
    fill = "Cluster",
    y="Proportion"
  )
```



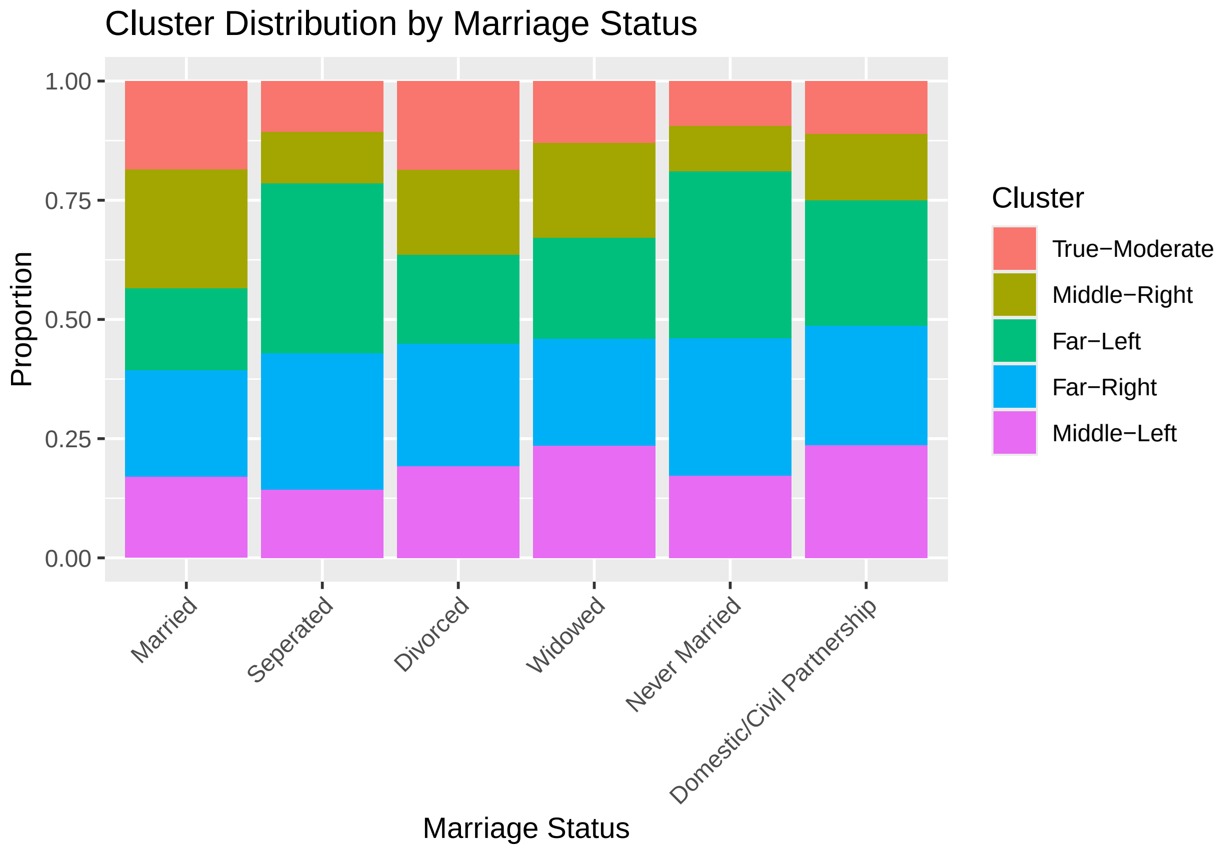

This tends to agree with what I saw in 1.6. Females tend to be more left-leaning than men, their is a higher proportion of Left/Middle-Left women than men.

```
ft_kmeans %>%
  ggplot(aes(x=educ, fill = Cluster))+
  geom_bar(position = "fill")+
  labs(
    title = "Cluster Distribution by Education Level",
    x="Education Level",
    fill = "Cluster",
    y="Proportion"
  )+
  theme(axis.text.x = element_text(angle=45, hjust =1))
```




This also agrees with 1.6. The Far-Left definitely increases as education increases.

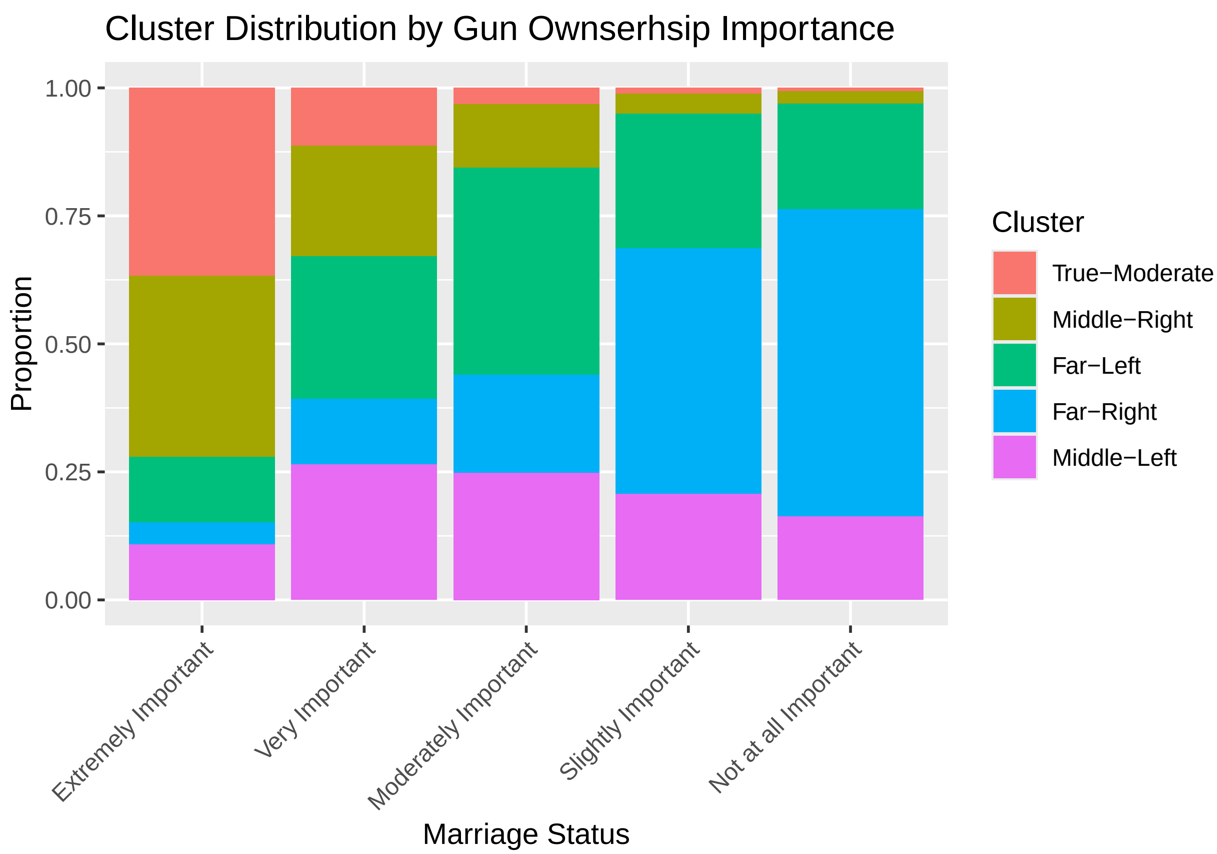
```
ft_kmeans %>%
  ggplot(aes(x=marstat, fill = Cluster))+
  geom_bar(position = "fill")+
  labs(
    title = "Cluster Distribution by Marriage Status",
    x="Marriage Status",
    fill = "Cluster",
    y="Proportion"
  )+
  theme(axis.text.x = element_text(angle=45, hjust =1))
```




These clusters seem to be agreeing with what I saw in 1.6. It was harder to make seperations based on Marriage staus, however the married couples seem to have a fairly even split across all ideologies. The never married seem to have fewer far-right people.

### 2.4 Gun Ownership Clustering

```
ft_kmeans %>%
  left_join(guns, by="caseid") %>%
  ggplot(aes(x=gunown, fill = Cluster))+
  geom_bar(position = "fill")+
  labs(
    title = "Cluster Distribution by Gun Ownserhsip Importance",
    x="Marriage Status",
    fill = "Cluster",
    y="Proportion"
  )+
  theme(axis.text.x = element_text(angle=45, hjust =1))
```




Here it is even more clear that the distributions support the conclusions from 1.7. The Far-Right/Right Finds Gun Ownership to be extremely important. Of the people who say it is "not at all important", the far-left are the majority by far.

```
stopCluster(cl)
registerDoSEQ()
```