Multivariate Data Analysis on U.S. Counties in the Northeast

Caroline Useda

Introduction

In my project, I plan on studying U.S. counties to see if counties are most similar to those in their own state, or if there are other factors that can describe counties better. I will be using a dataset that gives information about counties, provided by the U.S. Census. (Hammer 2020) For my analysis, I will use specifically the Northeast states (defined by the Census as NY, NJ, CT, PA, MA, RI, VT, NH, ME). I am particularly interested in this region as I have lived here all my life, and have noticed the differences in the counties in CT, my home state. This lead me to wonder if those differences are consistent with the rest of the region, or if other states are more similar.

The first method I will include is Clustering, which aims to find natural groups within a dataset. I will see if the groups found align with the states that the counties belong. Next, I will perform Classification to see if the characteristics of counties in these states are distinct enough that a set of qualifications can correctly determine the state a county belongs to.

Preliminary Analysis

```
#loading data
mydata <- read_csv("UsedaCData.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   area_name = col_character(),
##   state_abbreviation = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
#mydata <- read_csv("~/stat 240 project/UsedaCData.csv")

#renaming variables to names that are easier to use
mydata <- dplyr::rename(mydata, pop = PST045214, p_children = AGE295214, p_elderly = AGE775214,
                        p_female = SEX255214, p_white = RHI125214, p_black = RHI225214,
                        p_asian = RHI425214,
                        p_hispanic = RHI725214, med_income = INC110213, pop_sqmile = POP060210,
                        homeownership = HSG445213, state= state_abbreviation)

#adjusting dataset to only include these variables
mydata <- select(mydata, pop, p_children, p_elderly, p_female, p_white, p_black, p_asian, p_hispanic,
                 med_income, pop_sqmile, homeownership, area_name, state)

#adjusting dataset to only include NE counties
mydata <- filter(mydata, state=="NY" |state=="CT" | state=="NJ" | state=="PA" | state=="MA" |
                   state=="RI" | state=="NH" | state=="VT" | state=="ME")

#creating dataset that can be used in clustering/classifcaton
mydata2 <- select(mydata, -area_name)
mydata2 <- mutate(mydata2, state=factor(state))

#creating dataset for classification with just NY and PA
mydata3 <- filter(mydata2, state=="NY" | state=="PA")
mydata3 <- mutate(mydata3, state=factor(state))
```

I will start by giving a brief look at the datset.

```
glimpse(mydata)
```

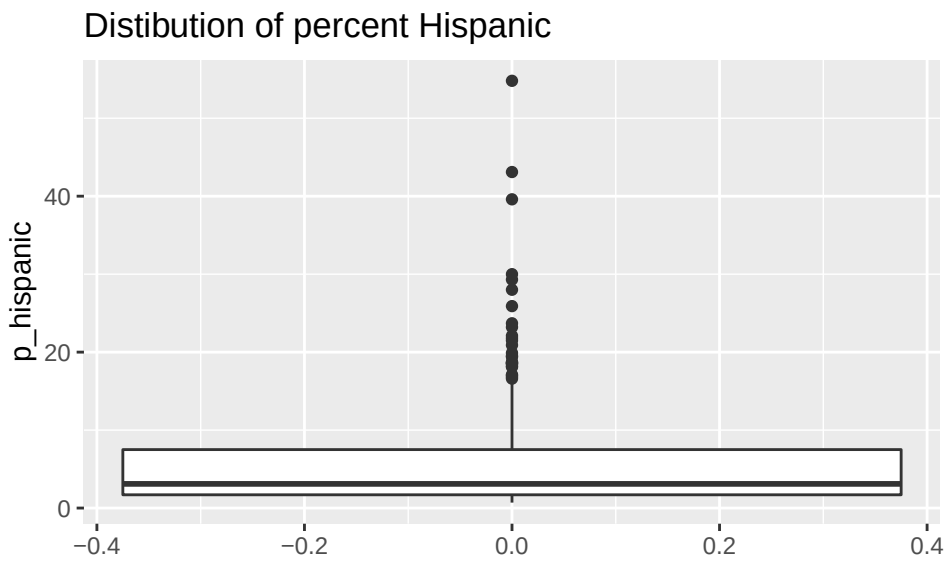
```
## Rows: 217
## Columns: 13
## $ pop           <dbl> 945438, 897985, 184993, 164943, 861277, 273676, 15136...
## $ p_children    <dbl> 23.6, 21.7, 19.4, 19.4, 21.1, 20.3, 18.5, 20.7, 22.0,...
## $ p_elderly     <dbl> 14.4, 15.6, 18.4, 17.7, 15.6, 16.0, 14.0, 14.7, 16.0,...
## $ p_female      <dbl> 51.2, 51.5, 50.7, 51.2, 51.8, 49.9, 49.8, 50.4, 50.9,...
## $ p_white       <dbl> 79.8, 76.8, 94.4, 89.4, 78.8, 83.9, 90.3, 92.8, 92.8,...
## $ p_black       <dbl> 12.2, 15.2, 1.8, 5.4, 14.2, 6.9, 3.6, 2.9, 3.8, 0.8, ...
## $ p_asian       <dbl> 5.5, 5.1, 1.8, 3.0, 4.2, 4.5, 4.1, 1.4, 0.8, 0.5, 2.2...
## $ p_hispanic    <dbl> 18.7, 17.0, 5.5, 5.8, 16.8, 9.9, 5.0, 10.9, 1.7, 1.1,...
## $ med_income    <dbl> 82283, 64967, 71338, 76994, 61996, 66583, 80529, 5933...
## $ pop_sqmile    <dbl> 1467.2, 1216.2, 206.3, 448.6, 1426.7, 412.2, 372.2, 2...
## $ homeownership <dbl> 69.0, 65.5, 78.1, 76.2, 63.7, 67.5, 74.4, 70.1, 64.6,...
```

```
## $ area_name     <chr> "Fairfield County", "Hartford County", "Litchfield Co...
## $ state         <chr> "CT", "CT", "CT", "CT", "CT", "CT", "CT", "CT", "ME",...
```

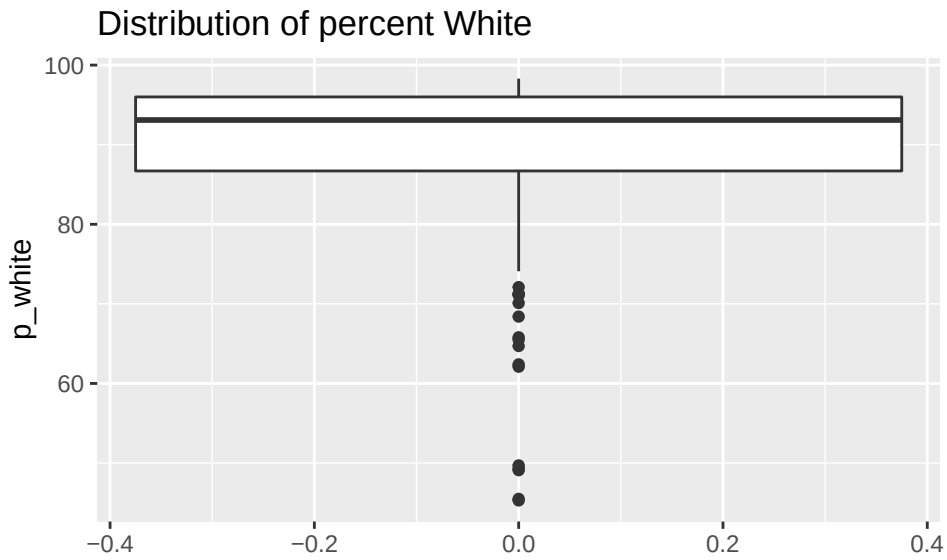
The dataset contains information on U.S. counties using information from the U.S. Census (Hammer 2020). I have filtered it to contain only observations from the Northeast states. I then modified the variables to reduce the size of the dataset to make it easier to work with, focusing on demographic variables such as percent white (p_white) and population (pop). Given these modifications, the dataset has 13 variables and 217 observations. All of my variables are numerical, except area_name, which will only be used to identify observations, and state which will be used to identify observations in clustering and classify observations in the classification analysis. It is important to note here that when given a choice, variable values were taken from the 2014 analysis, but data from the 2014 Census was not present in the dataset for all variables. Thus, some variables, specifically pop_sqmile, homeownership, and med_income, are from different years.

To prepare for analysis, I will first look at the distribution of each variable.

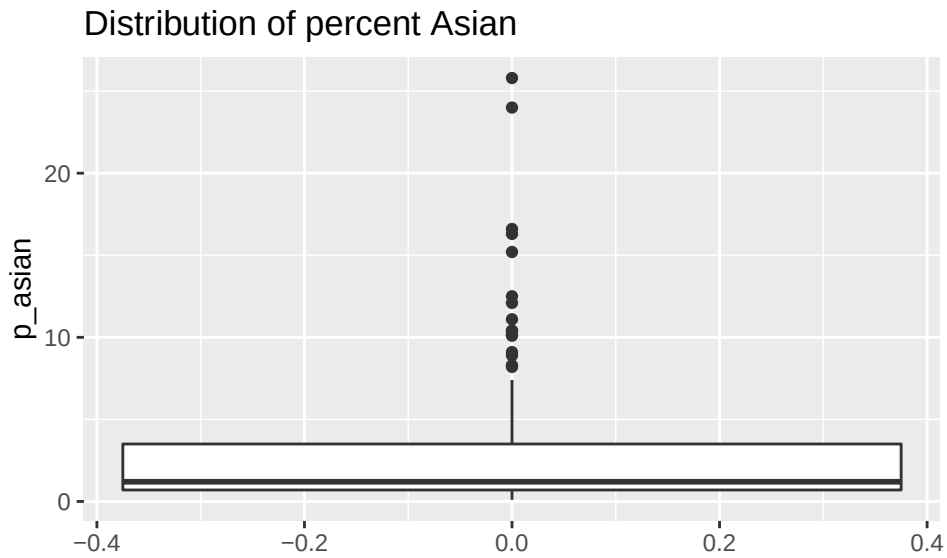
```
#univariate analysis
gf_boxplot(~p_hispanic, data=mydata) %>% gf_labs(title = "Distibution of percent Hispanic")
```



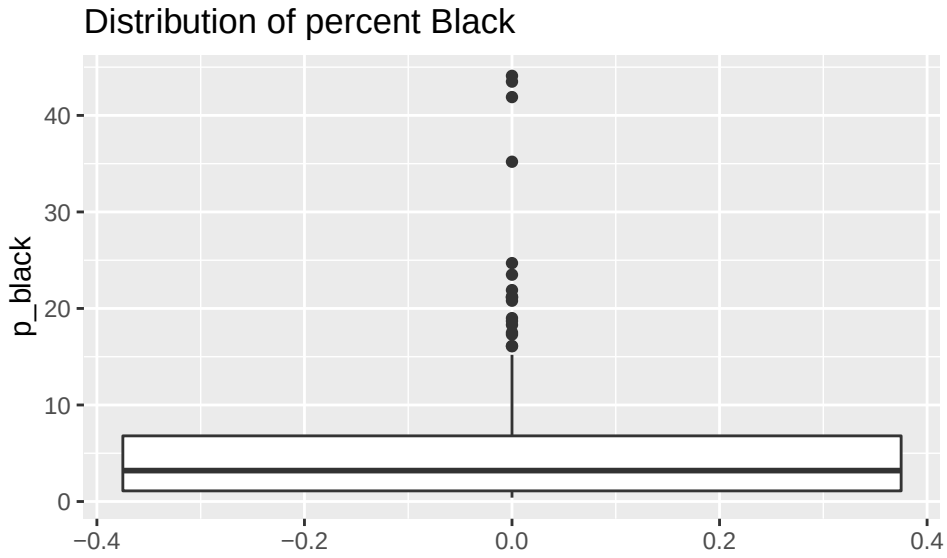

```
gf_boxplot(~p_white, data=mydata)%>% gf_labs(title = "Distribution of percent White")
```



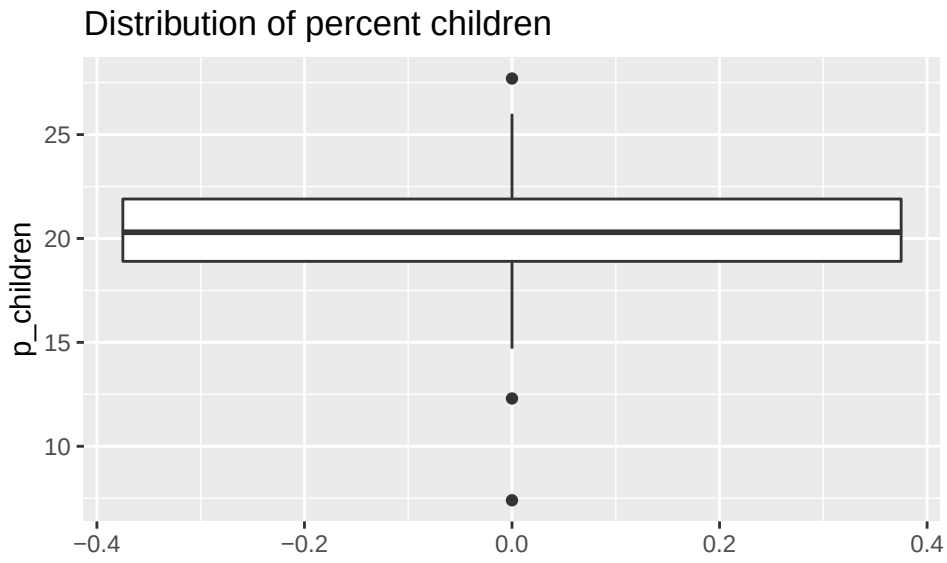
```
gf_boxplot(~p_asian, data=mydata)%>% gf_labs(title = "Distribution of percent Asian")
```



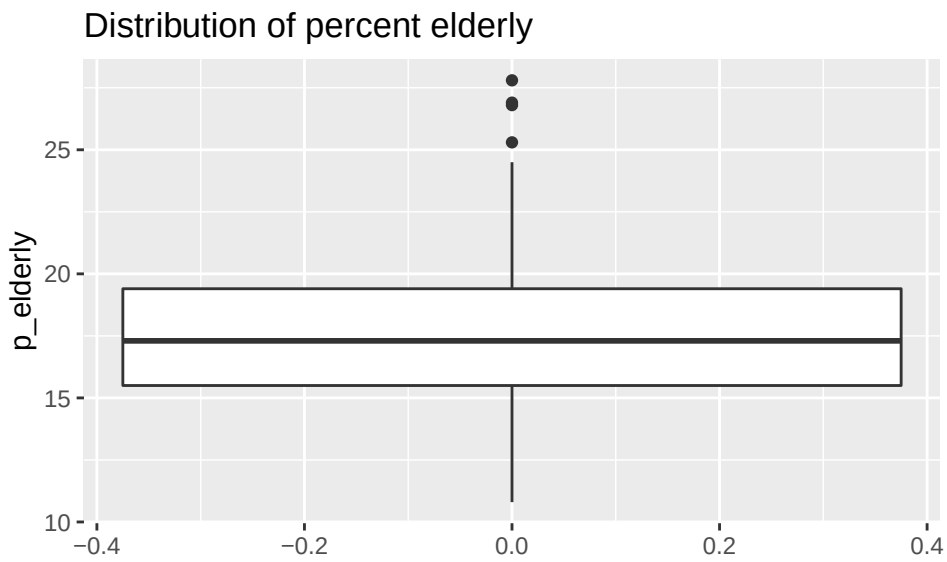

```
gf_boxplot(~p_black, data=mydata)%>% gf_labs(title = "Distribution of percent Black")
```



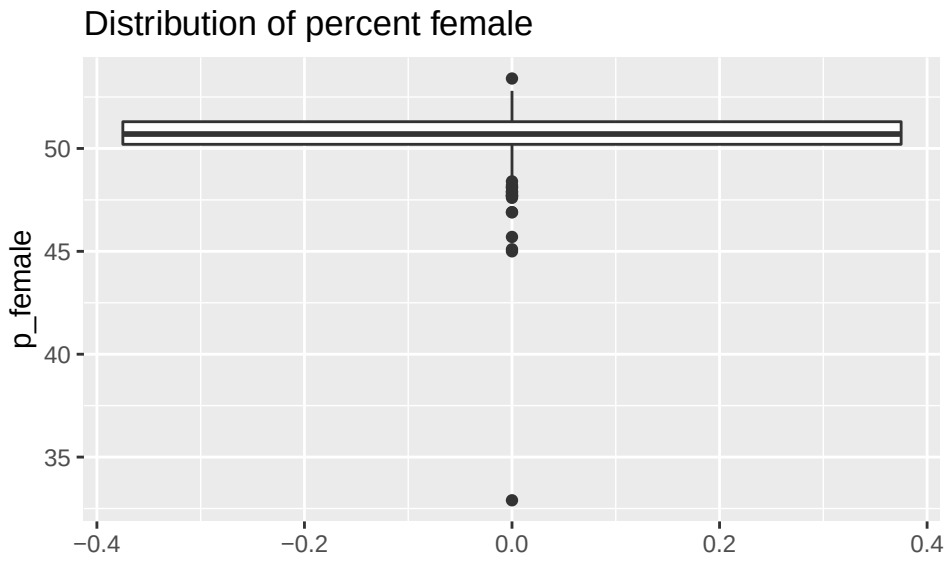

```
gf_boxplot(~p_children, data=mydata)%>% gf_labs(title = "Distribution of percent children")
```



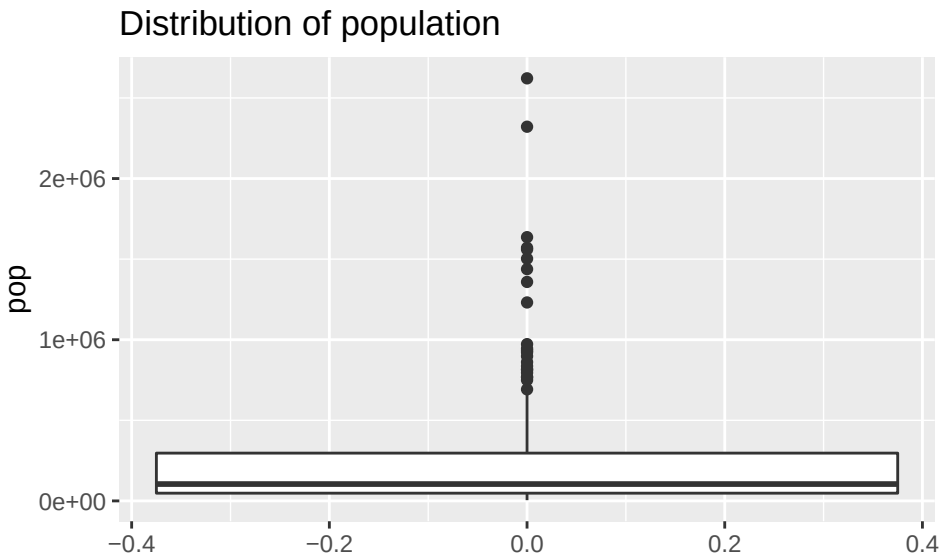

```
gf_boxplot(~p_elderly, data=mydata)%>% gf_labs(title = "Distribution of percent elderly")
```



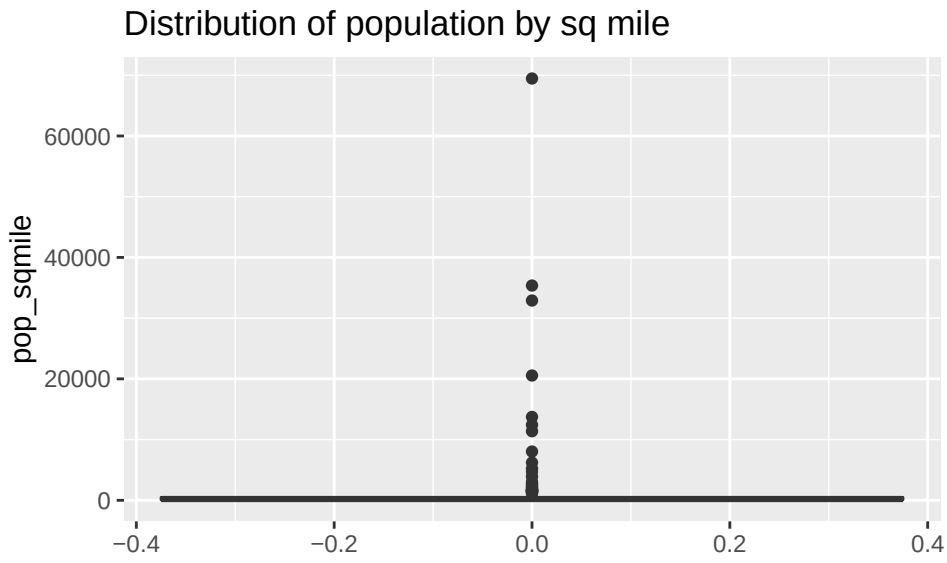

```
gf_boxplot(~p_female, data=mydata)%>% gf_labs(title = "Distribution of percent female")
```



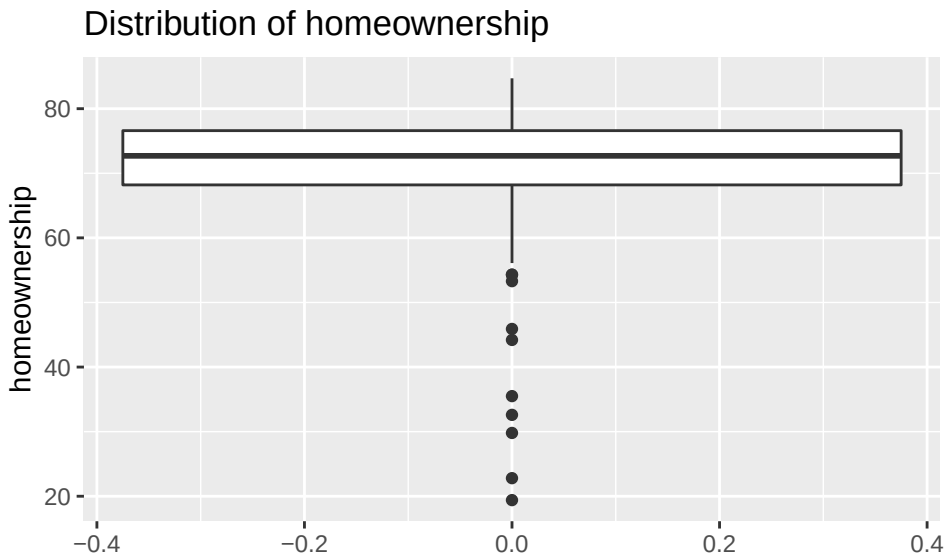

```
gf_boxplot(~pop, data=mydata)%>% gf_labs(title = "Distribution of population")
```



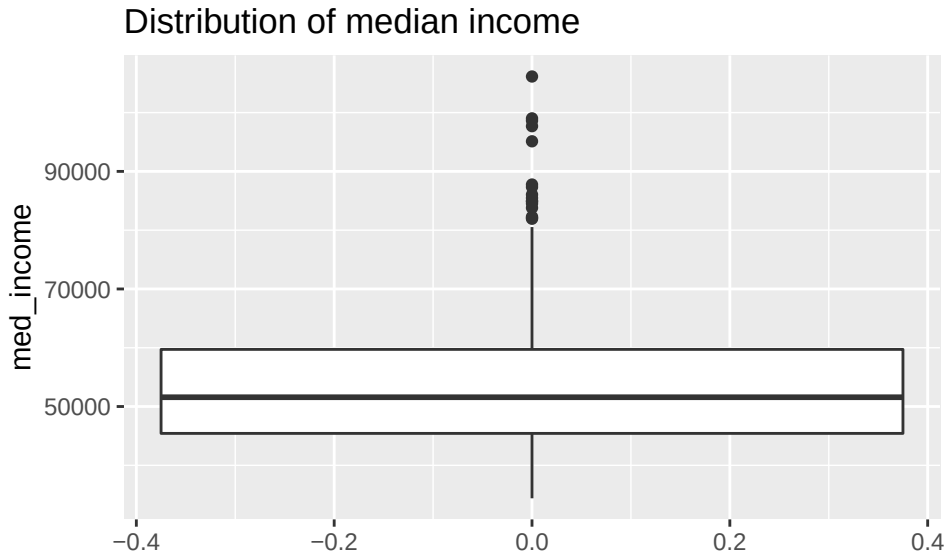

```
gf_boxplot(~pop_sqmile, data=mydata)%>% gf_labs(title = "Distribution of population by sq mile")
```




```
gf_boxplot(~homeownership, data=mydata)%>% gf_labs(title = "Distribution of homeownership")
```




```
gf_boxplot(~med_income, data=mydata)%>% gf_labs(title = "Distribution of median income")
```




```
tally(~state, data=mydata)
```

```
## state
## CT MA ME NH NJ NY PA RI VT
##  8 14 16 10 21 62 67  5 14
```

Overall, the distribution of the numerical variables shows that most counties in the NE are majority white, with about half females. The majority of variables appear to have pretty similar values (since most variables have a small range), but there are many outliers for almost all of them. The response variable for my classification, state does not have an equal distribution since it is based on how many counties are in the state, which can differ.

Next, I will investigate relationships between variables.

```
#bivariate analysis
cor(select(mydata2, -state))
```

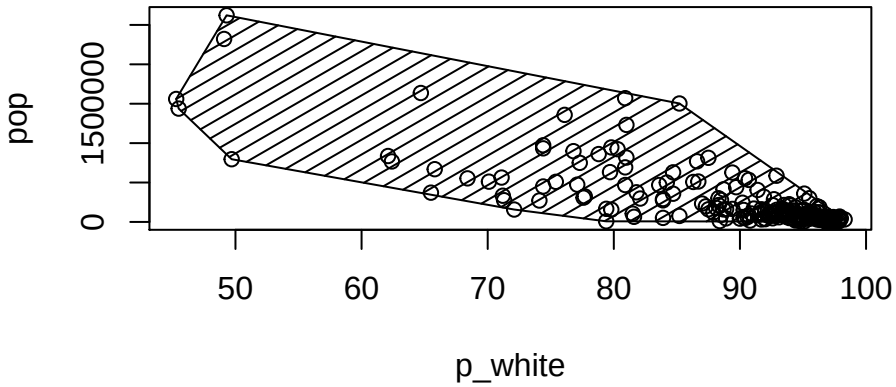
```
##                    pop p_children p_elderly  p_female   p_white   p_black
## pop            1.000000  0.2966691 -0.473359  0.330203 -0.776825  0.685970
## p_children     0.296669  1.0000000 -0.507037  0.417209 -0.314458  0.315189
## p_elderly     -0.473359 -0.5070367  1.000000 -0.157685  0.611854 -0.537449
## p_female       0.330203  0.4172089 -0.157685  1.000000 -0.189014  0.116636
## p_white       -0.776825 -0.3144582  0.611854 -0.189014  1.000000 -0.945045
## p_black        0.685970  0.3151891 -0.537449  0.116636 -0.945045  1.000000
## p_asian        0.707894  0.1942919 -0.520266  0.261595 -0.755308  0.513834
## p_hispanic     0.619248  0.4067944 -0.553624  0.182688 -0.783900  0.728193
## med_income     0.344468  0.2917035 -0.417391  0.241946 -0.302913  0.159628
## pop_sqmile     0.635780  0.0122724 -0.290335  0.205458 -0.568867  0.519212
## homeownership -0.618077 -0.1585837  0.565402 -0.278242  0.706781 -0.654623
##                  p_asian p_hispanic med_income pop_sqmile homeownership
## pop            0.707894   0.619248  0.3444679  0.6357803    -0.6180772
## p_children     0.194292   0.406794  0.2917035  0.0122724    -0.1585837
## p_elderly     -0.520266  -0.553624 -0.4173905 -0.2903353     0.5654018
## p_female       0.261595   0.182688  0.2419462  0.2054575    -0.2782417
## p_white       -0.755308  -0.783900 -0.3029131 -0.5688670     0.7067807
## p_black        0.513834   0.728193  0.1596278  0.5192119    -0.6546235
## p_asian        1.000000   0.594101  0.5055407  0.4624145    -0.5260501
## p_hispanic     0.594101   1.000000  0.3207983  0.5160630    -0.6592901
```

```
## med_income     0.505541   0.320798  1.0000000  0.0628576     0.0388888
## pop_sqmile     0.462415   0.516063  0.0628576  1.0000000    -0.7102143
## homeownership -0.526050  -0.659290  0.0388888 -0.7102143     1.0000000
```

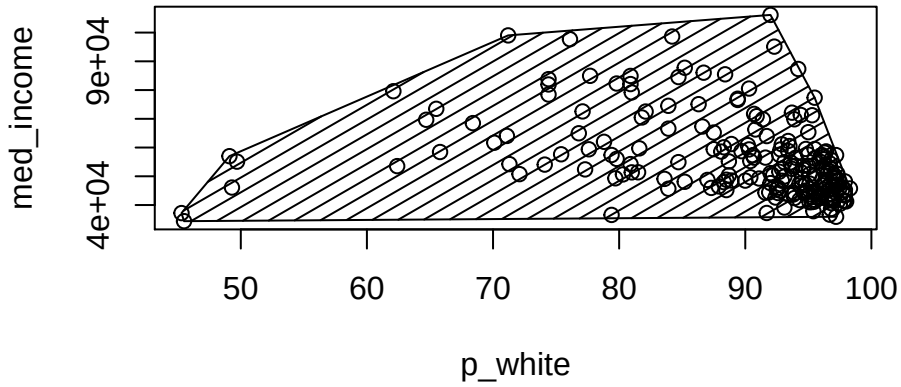
From these graphs, we see that many of the variables are correlated, specifically with the race variables. High percentage of white people is associated with higher homeownerhsip rates, smaller populations, and low percentage of minorities.

Next, I will investigate the presence of outliers within my dataset using a convex hull. The points on the edge of the hull represent observations that could be removed without changing the data significantly. I decided to use 3 of the variables in the datset that have significant relationships to investigate this.

```
#investigating outliers
#hull with first set of variables
hull <- with(mydata, chull(p_white, pop))
with(mydata, plot(p_white, pop, pch = 1))
with(mydata, polygon(p_white[hull], pop[hull], density = 15, angle = 30))
```




```
#hull with second set of variables
hull2 <- with(mydata, chull(p_white, med_income))
with(mydata, plot(p_white, med_income, pch = 1))
with(mydata, polygon(p_white[hull2], med_income[hull2], density = 15, angle = 30))
```



There are some possible outliers, but I am not too concerned since it is expected that some counties could have different characteristics, especially when assuming racial bias. Additionally, since some states have less counties than others, removing outliers could be problematic for later analysis, so I will choose to keep the dataset as is.

Methods

The first method that I will be using is Clustering. Clustering is a statistical technique in which a set of observations (in this case, counties) with similar characteristics are grouped together in clusters. I made multiple models using kmeans clustering, which aims to partition observations into k clusters in which each observation belongs to the cluster with the nearest mean. I chose k means as I wanted to be able to change k, the number of clusters, based on previous solutions. This will help with my initial research question as clustering will allow me to see which factors characterize the clusters, as well the observations within those clusters. I will use all 9 NE states in my clustering solutions. For choosing k, I started with 9 to investigate if the model will find that clusters align with states. I will then see how well that solution helps with my analysis, and adjust k based on both my intuition as well as the WGSS (within group sum of squares). I will then see how well these solutions perform using silhouette values, which measures how well observations fit into their own cluster. I will then pick a final model(s).

The second method I will be using is Classification. Classification is a statistical technique that identifies which of a set of classes (in our case, states) an observation belongs, using a training set of data containing observations whose class we know. This will also assess how well observations belong to their own classes, as observations that the model finds it difficult to classify might not belong well. I will use Random Forests, which will generate many predictions (classification trees) for each observation and use majority voting to determine the final class. RF allows you to see which variables are important as well, which will be useful for analysis, and is additonally my reason for picking RF to begin with. Because I am using RF, a train test split is not needed on the data, and I will look at the OOB error rate, or estimated TER, for analyzing how well the solution is performing. I will experiment with different mtry (number of variables tried at each split) and n (number of trees) to see which model yields the best result. I will first attempt to classify all observations, but that will be difficult as some states have significantly less counties than others. I will then create a model that just focuses on NY and PA, as they have similar numbers of counties.

Results

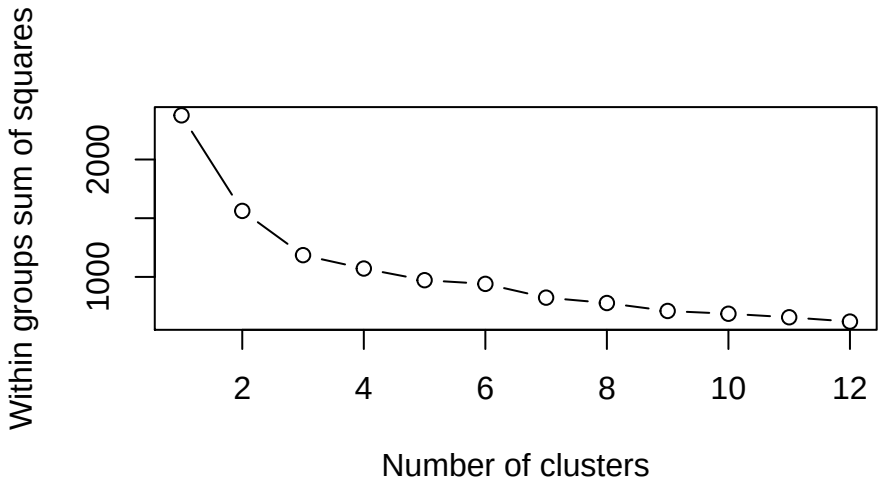
Clustering

For the first cluster analysis using kmeans clustering, I chose 9 clusters, to see if the clusters found by the models aligned with the states the counties belonged in.

```
#k=9
set.seed(240)wss <- rep(0, 12)
for(i in 1:12){wss[i] <- sum(kmeans(scale(select(mydata, -state, -area_name)),
centers = i)$withinss)}
plot(1:12, wss, type = "b", xlab = "Number of clusters", ylab = "Within groups sum of squares")
mykm <- kmeans(scale(select(mydata, -state, -area_name)), centers = 9)
list(mykm)
```

I found this not to be the case, noting that one county was its own cluster. I then looked at the scree plot for WGSS to see what k would be good to start with. WGSS is the within cluster sum of squares, so we want to minimize it. Thus, I will look for an elbow in the plot to find a value where the WGSS is minimized without having too many clusters (since the lowest WGSS would be every observation within it's own cluster). This point is referred to as an elbow.

```
#WGSS plot
wss <- rep(0, 12)
for(i in 1:12){wss[i] <- sum(kmeans(scale(select(mydata, -state, -area_name)),
centers = i)$withinss)}
plot(1:12, wss, type = "b", xlab = "Number of clusters", ylab = "Within groups sum of squares")
```



I saw that there was an elbow of 3 in the WGSS plot, so I decided to try that as k instead. This k proved to be more useful for analysis.

```
#k=3
set.seed(240)
wss <- rep(0, 12)
for(i in 1:12){wss[i] <- sum(kmeans(scale(select(mydata, -state, -area_name)),
centers = i)$withinss)}
mykm2 <- kmeans(scale(select(mydata, -state, -area_name)), centers = 3)
list(mykm2)
```

```
## [[1]]
## K-means clustering with 3 clusters of sizes 8, 150, 59
##
## Cluster means:
```

```
##         pop p_children p_elderly  p_female   p_white   p_black   p_asian
## 1  3.158138   0.295558 -1.776867  0.914429 -3.461831  3.399342  2.417464
## 2 -0.437225  -0.307217  0.414143 -0.176183  0.490794 -0.439622 -0.421652
## 3  0.683367   0.740985 -0.811975  0.323933 -0.778380  0.656756  0.744205
##   p_hispanic med_income pop_sqmile homeownership
## 1   2.788504  -0.293958   3.947229     -3.853761
## 2  -0.451470  -0.393267  -0.204179      0.281685
## 3   0.769703   1.039691  -0.016118     -0.193604
##
## Clustering vector:
##   [1] 3 3 2 2 3 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 3 2 3 2 3 3 3 3 1
##  [38] 3 2 2 2 2 2 3 2 2 2 2 3 3 3 3 2 3 1 3 1 3 3 3 3 3 2 3 3 3 3 3 3 3 2 1 2 2
##  [75] 2 2 2 2 2 2 2 2 3 3 2 2 2 2 2 2 2 3 1 2 2 2 3 2 3 1 2 2 3 2 3 2 2 2 3 1 2
## [112] 3 3 2 2 3 2 2 2 2 3 2 2 3 2 2 2 2 3 2 2 2 3 2 2 2 3 2 2 3 2 2 2 2 2 3 2 2
## [149] 2 2 2 2 3 3 2 2 2 2 2 2 2 2 2 2 2 2 3 2 2 3 2 2 2 2 2 3 3 2 3 2 2 1 2 2 2
## [186] 2 2 2 2 2 2 2 2 2 2 2 2 3 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##
## Within cluster sum of squares by cluster:
## [1] 214.264 564.750 406.324
##  (between_SS / total_SS =  50.1 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
#tally(area_name~mykm$cluster, data=mydata)
#area_name was looked at, but code is commented out due to long output
```

The 3 cluster solution has clusters containing 8, 150, and 59 observations respectively. The first point of interest in that solution was the 1st cluster, as it had significantly less counties than the other 2. Looking at the counties, I saw these counties seemed to include large cities, such as Philadelphia county, NYC boroughs/counties, and Suffolk county, where Boston is located. I then turned my attention to the other 2 clusters to observe differences. As a CT native, I observed that that more urban CT counties (but no where near as urban as cluster 1) were located in cluster 3, while the more rural ones were located in cluster 2. Thus, the clusters can be characterized as big city(1), suburban/rural(2) and small city/suburban(3). I then looked to see the distributions of the demographic variables to see if they aligned with my suspicions.

```
##looking at clusters k=3
favstats(p_white ~ mykm2$cluster, data = mydata)
```

```
##   mykm2$cluster  min     Q1 median     Q3  max    mean      sd   n missing
## 1             1 45.3 48.200   49.5 62.975 65.8 53.9750 8.75716   8       0
## 2             2 79.4 92.825   95.1 96.675 98.3 94.2687 3.12092 150       0
## 3             3 62.1 76.950   81.0 87.100 94.2 81.3305 7.22281  59       0
```

```
favstats(med_income ~ mykm2$cluster, data = mydata)
```

```
##   mykm2$cluster   min      Q1  median      Q3    max    mean       sd   n
## 1             1 34388 43861.8 54317.5 57361.2  69659 51425.2 11662.54   8
## 2             2 35916 44162.8 48471.5 54888.5  77348 50058.9  8254.76 150
## 3             3 46484 56272.0 68507.0 82018.0 106143 69774.1 15078.73  59
##   missing
## 1       0
## 2       0
## 3       0
```

```
favstats(homeownership ~ mykm2$cluster, data=mydata)
```

```
##   mykm2$cluster  min     Q1 median     Q3  max    mean       sd   n missing
## 1             1 19.4 28.050  34.05 44.625 53.3 35.4375 11.72335   8       0
## 2             2 59.6 70.925  73.70 76.800 84.2 73.7393  4.54848 150       0
## 3             3 54.3 65.350  68.30 74.750 84.7 69.3373  7.28913  59       0
```

```
favstats(pop_sqmile ~ mykm2$cluster, data=mydata)
```

```
##   mykm2$cluster    min       Q1   median        Q3     max      mean        sd
## 1             1 6211.5 12156.65 17142.50 33519.975 69467.5 25253.987 20669.058
## 2             2    2.8    52.80    98.75   192.875  2064.0   159.885   221.575
## 3             3   91.6   469.85   881.40  1634.800  8030.3  1296.663  1384.372
##     n missing
## 1   8       0
## 2 150       0
## 3  59       0
```

Here, I can see that the big city cluster had the lowest values for p_white, homeownership, and the highest value for pop_sqmile, which lines up knowing cities such as NYC tend to be more diverse, with many apartments instead of houses. The rural/suburban cluster was the most white, and similar income to the big city cluster. The small city/suburban cluster had the highest income, with similar homeownership to the rural cluster.

I found this solution to be useful, but I was interested in seeing if I could further classify the clusters so that suburban was its own cluster, instead of being incorporated into both the small city and rural clusters, although noting that since our observations here are counties, that it is possible that some will inevitably be both suburban, rural, and have small cities since municipalities within a county can differ. I then tried to create a solution with 4 clusters to see if that could create a suburban cluster.

```
#k=4
set.seed(240)
mykm3 <- kmeans(scale(select(mydata, -state, -area_name)), centers = 4,iter.max=100)
list(mykm3)
```

```
## [[1]]
## K-means clustering with 4 clusters of sizes 52, 135, 8, 22
##
## Cluster means:
##         pop p_children p_elderly   p_female   p_white   p_black   p_asian
## 1  0.775033  0.7722876 -0.815194  0.3806781 -0.896069  0.761051  0.851234
## 2 -0.396064 -0.0714157  0.143985 -0.0147259  0.471851 -0.436776 -0.384648
## 3  3.158138  0.2955577 -1.776867  0.9144287 -3.461831  3.399342  2.417464
## 4 -0.549917 -1.4946500  1.689410 -1.1419405  0.481383 -0.354757 -0.530748
##   p_hispanic med_income  pop_sqmile homeownership
## 1   0.853954   1.080251  0.00215272     -0.225245
## 2  -0.416141  -0.283106 -0.19941088      0.203379
## 3   2.788504  -0.293958  3.94722888     -3.853761
## 4  -0.478847  -0.709185 -0.21678652      0.685761
##
## Clustering vector:
##   [1] 1 1 2 2 1 1 2 2 2 2 2 2 4 2 4 4 2 2 4 2 2 2 4 2 4 2 2 2 1 2 1 2 1 1 1 1 3
##  [38] 1 2 4 2 4 2 1 2 2 2 2 1 1 1 1 4 1 3 1 3 1 1 1 1 1 2 1 1 1 2 1 2 1 2 3 2 2
##  [75] 2 2 2 2 2 2 2 4 1 1 4 2 2 2 4 4 2 2 3 2 2 2 1 2 1 3 2 2 1 2 1 2 2 2 1 3 2
## [112] 1 1 2 2 1 2 2 2 2 1 2 2 1 2 2 2 2 1 2 2 2 1 2 2 2 1 2 2 1 2 2 4 2 2 1 2 4
## [149] 2 2 2 2 1 1 2 2 2 4 2 2 2 4 2 2 2 2 1 2 2 1 2 2 2 2 2 2 1 1 2 2 2 2 3 2 2 2
```

```
## [186] 2 4 4 2 2 4 2 2 2 4 2 2 2 2 2 2 1 2 2 2 2 2 4 2 2 2 2 2 2 2 2 2
##
## Within cluster sum of squares by cluster:
## [1] 355.739 326.227 214.264 176.315
##  (between_SS / total_SS =  54.9 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
#tally(area_name~mykm3$cluster, data=mydata)
#area_name was looked at, but code is commented out due to long output
```

The 4 cluster solution had clusters of 55, 141, 13, and 8 observations, respectively. The 4th cluster is the same as the the 1st in the previous solution, so I will continue to describe it as big city. I continued to use CT as a reference point for looking at the other clusters for a number of reasons. It has only 8 counties so it easy to look at, I am familiar with all of those counties, and lastly since CT is such a small state, no county can truly be rural. I then looked at the 3 CT counties that I noted earlier as being urban/small city. As they were located in the 3rd cluster, I determined that it was possible this cluster could be described as small city. I then looked to the 5 other CT counties, and noted that they were in the 1st cluster, so I determined that Cluster 1 was suburban, and thus cluster 2 was rural. This lines up with what I was looking to achieve, as both the small city and rural cluster have decreased in size to give room for the new cluster. It is also interesting to note that most of the observations that moved into the new cluster came from the rural/suburban cluster from solution 2. I looked at summary statistics to confirm this, refraining from commenting on the big city cluster since it is the same.

```
##looking at clusters k=4
favstats(p_white ~ mykm3$cluster, data = mydata)
```

```
##   mykm3$cluster  min     Q1 median     Q3  max    mean      sd   n missing
## 1             1 62.1 75.925  80.90 84.700 92.3 80.1308 6.81589  52       0
## 2             2 83.9 92.650  94.80 96.400 98.3 94.0756 3.02752 135       0
## 3             3 45.3 48.200  49.50 62.975 65.8 53.9750 8.75716   8       0
## 4             4 79.4 92.825  95.35 96.875 97.8 94.1727 4.12278  22       0
```

```
favstats(med_income ~ mykm3$cluster, data = mydata)
```

```
##   mykm3$cluster   min      Q1  median      Q3    max    mean       sd   n
## 1             1 49094 56377.5 69168.0 82138.2 106143 70332.1 15197.10  52
## 2             2 37855 44604.5 48900.0 57007.0  87335 51574.6  9265.10 135
## 3             3 34388 43861.8 54317.5 57361.2  69659 51425.2 11662.54   8
## 4             4 35916 41125.0 45925.0 49730.0  60526 45712.4  6523.48  22
##   missing
## 1       0
## 2       0
## 3       0
## 4       0
```

```
favstats(homeownership ~ mykm3$cluster, data=mydata)
```

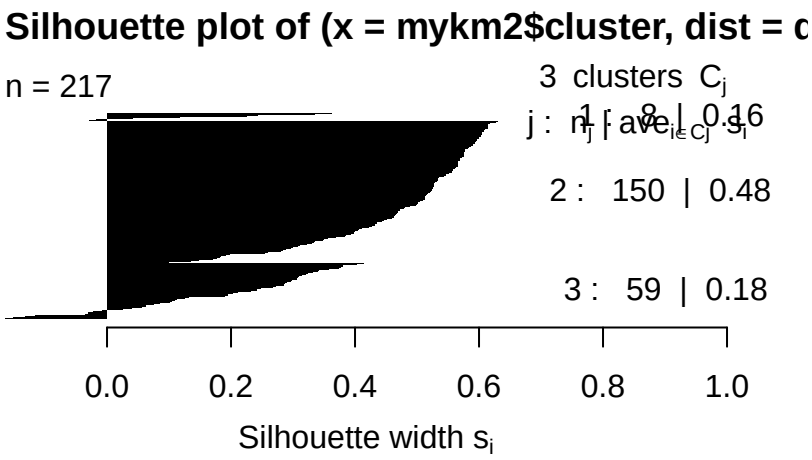
```
##   mykm3$cluster  min     Q1 median     Q3  max    mean       sd   n missing
## 1             1 54.3 65.375  67.85 73.700 84.5 69.0442  7.04513  52       0
## 2             2 56.1 70.450  73.30 76.350 84.7 73.0141  4.66492 135       0
## 3             3 19.4 28.050  34.05 44.625 53.3 35.4375 11.72335   8       0
## 4             4 71.0 74.375  76.85 80.375 83.9 77.4818  3.99245  22       0
```

```
favstats(pop_sqmile ~ mykm3$cluster, data=mydata)
```

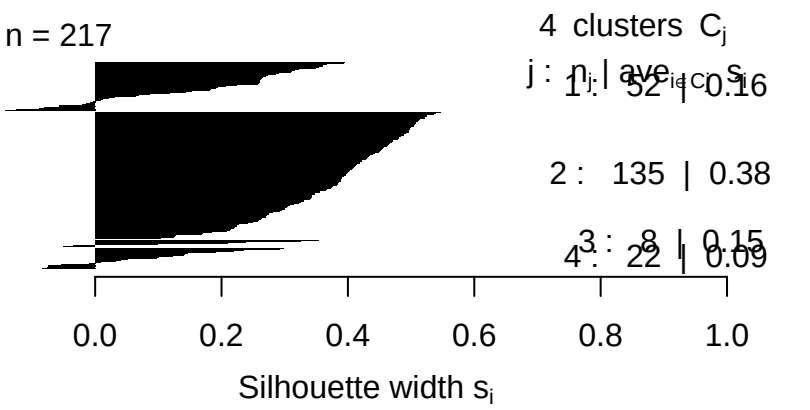
```
##   mykm3$cluster    min        Q1   median       Q3     max       mean        sd
## 1             1  199.1   506.475  1023.60  1660.83  8030.3  1407.1038  1436.314
## 2             2   10.8    61.800   109.60   224.15  2064.0   188.7081   244.468
## 3             3 6211.5 12156.650 17142.50 33519.97 69467.5 25253.9875 20669.058
## 4             4    2.8    15.250    42.85    74.90   548.3    83.6773   131.828
##     n missing
## 1  52       0
## 2 135       0
## 3   8       0
## 4  22       0
```

While the suburb cluster has the highest values for p_white, which might be expected to be a characteristic of the rural cluster, the rural cluster has the lowest values for med_income, pop_sqmile, and female, which aligns with expectations. Again, the small city cluster has the highest income, and 2nd highest pop_sqmile (big city is highest). It seems as though these clusters define the counties better than their initial states may be able to. I then looked to see how well these solutions performed.

```
##cluster validation
#code for k=3
kmeansSil <- silhouette(mykm2$cluster, dist(scale(select(mydata2, -state))))
silsum <- summary(kmeansSil)
plot(kmeansSil, col = "black")
```




```
#code for k=4
kmeansSil2 <- silhouette(mykm3$cluster, dist(scale(select(mydata2, -state))))
silsum <- summary(kmeansSil2)
plot(kmeansSil2, col = "black")
```



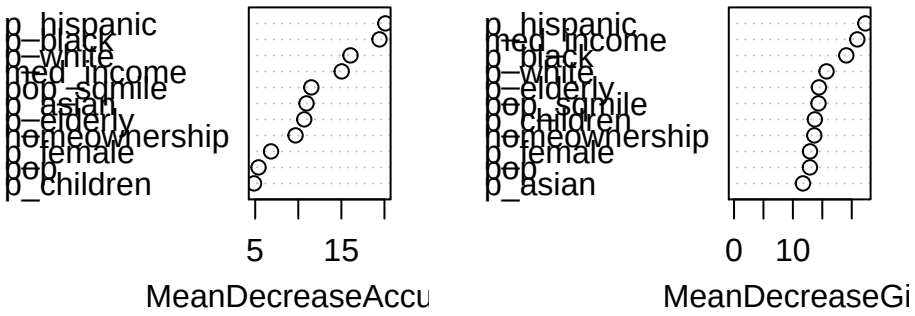
Here, we can see that the second solution performs better than the 3rd. With an average silhouette width (scale from -1 to 1 where 1 is best) of .39, we know that observations belong better in their own cluster than in another. The solution is still moderate at best. It is also interesting to note the high silhouette value for the 2nd cluster (suburban/rural) in comparison to the other 2, and is the reason the average silhouette width is as high as it is. Since the goal of the 3rd solution was to separate this cluster, it makes sense that the 3rd solution has a lower average silhouette width. Thus, model 2, with 3 clusters, is my final model.

Classification

```
##trying RF w all
set.seed(4)
g.rf <- randomForest(state ~ ., data = mydata2, mtry = 5, ntree = 300,
                     importance = T, proximity = T)
g.rf
```

```
##
## Call:
##  randomForest(formula = state ~ ., data = mydata2, mtry = 5, ntree = 300,      importance = T, proxi
##                Type of random forest: classification
##                      Number of trees: 300
## No. of variables tried at each split: 5
##
##         OOB estimate of  error rate: 47%
## Confusion matrix:
##    CT MA ME NH NJ NY PA RI VT class.error
## CT  1  0  0  0  0  4  2  1  0    0.875000
## MA  0  2  0  0  1  6  5  0  0    0.857143
## ME  0  0  9  2  0  0  4  0  1    0.437500
## NH  0  1  4  0  0  3  0  0  2    1.000000
## NJ  0  0  0  0 10  6  5  0  0    0.523810
## NY  1  0  0  0  7 41 11  0  2    0.338710
## PA  0  2  0  0  3 16 45  0  1    0.328358
## RI  0  2  0  0  0  1  1  1  0    0.800000
## VT  0  0  1  2  0  2  3  0  6    0.571429
```

```
varImpPlot(g.rf)
```

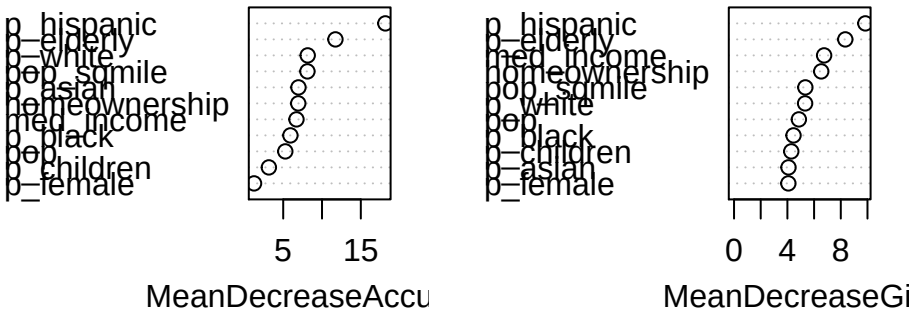


For classificaton, I first tried running RF using all of the states to see how well the solution will perform. I ran it a number of times using different mtry and n values, and found this solution to have the lowest TER corresponding to the least amount of states that were not classified correctly at all. Across all solutions, CT, NH, MA, and RI were difficult to classify as they all have a small number of counties. The TER for this model is .4700, which means 47% of objects were misclassified. Additonally, the variables that proved to be important across all solutions were p_hispanic, p_black, p_white, and med_income. Most solutions had pop_sqmile as important as well.

```
##trying RF w just NY and PA
set.seed(40)
g.rf <- randomForest(state ~ ., data = mydata3, mtry = 4, ntree = 500,
                     importance = T, proximity = T)
g.rf
```

```
##
## Call:
##  randomForest(formula = state ~ ., data = mydata3, mtry = 4, ntree = 500,      importance = T, proxi
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 4
##
##         OOB estimate of  error rate: 26.36%
## Confusion matrix:
##    NY PA class.error
## NY 48 14    0.225806
## PA 20 47    0.298507
```

```
varImpPlot(g.rf)
```



After experimenting with different values for mtry and n, I settled upon a model that with values of 4 and 500, respectively, that yielded an TER of 26.36%, the lowest I observed. Some of the variables that are consistent with the solution including all states, such as p_hispanic, med_income, pop_sqmile, and p_white. However, for NY and PA on their own, homeownership and p_elderly showed more importance. The solution performs relatively well, given that we know both states are similar in that they both contain large, urban areas (NYC and Philly), in addition to having a lot of land and more rural communities.

Conclusion

In this report, I set out to investigate whether or not counties were most similar to those in their own state, or if there are other factors that could describe them better. Using both k-means clustering and random forest classification, we observed that variables such as p_white, p_hispanic, pop_sqmile, and med_income are more useful in describing counties. In k-means, I was able to create clusters that seemed to describe whether counties were mostly big cities, small cities, suburban, or rural. With RF, we saw that it was difficult to classify counties as their own states, so we opted to try classifying only NY and PA due to their similarities instead. Here we saw a better result, but the misclassification rate was still moderate. Thus, with both the k-means and RF solutions, it appears that demographic variables such as race, income, and population by square mile are better at describing counties, and a possible grouping for counties is based on the extent to which they can be described as urban. This helps me to conclude that while the Northeast has many similarities overall, the counties within each state are diverse.

Additionally, since we noted that race was a factor in determining counties, there could be history that could help explain my findings. Segregation, although mostly associated to be an issue in the South, exists in the Northeast as well. In fact, it is noted in an article featured in the conversation that "since the late 1960s, the Northeast has experienced a steady increase in the percentage of black students enrolled in schools with fewer than 10% white students." (Frankenberg 2020) While the article mainly focuses on public schools, it is relevant as public schools in the U.S. are based on location. Additionally, it is known that desegregation efforts were more focused in the South than they were in the North, as the North's de facto segregation isn't illegal/forced. Thus, we can conclude that the racial makeup of the NE could be a contributing factor to our results.

Although these findings are especially interesting to note given the current dialogue in the U.S. surrounding race, it is important to note that the techniques that I used are exploratory, and thus my results cannot imply causation, for which an experiment is necessary. Another issue is that not all the variable values are from the same census year, and that could affect the results. Additionally, my kmeans solution for k=4 is not converging, so that needs to be investigated.

Citations

Hammer, B., 2020. 2016 US Election. [online] Kaggle.com. Available at: https://www.kaggle.com/benhamner/2016-us-election [Accessed 3 November 2020].

Frankenberg, E., 2020. What School Segregation Looks Like In The US Today, In 4 Charts. [online] The Conversation. Available at: https://theconversation.com/what-school-segregation-looks-like-in-the-us-today-in-4-charts-120061 [Accessed 17 November 2020].