Data Analysis: Homework 3

Adam Wells

10/23/2020

---

1. (5 points) Write a short paragraph describing what unsupervised learning is and how it differs from supervised learning.

Both supervised and unsupervised learning involve the creation of models using machine learning. Supervised learning involves an outcome (or dependent variable). The model ultimately specifies a particular relationship between the outcome and predictor variables. Alternately, unsupervised learning does not involve an outcome variable. Unsupervised learning uses particular algorithms, depending on the type of unsupervised learning, to categorize data into groups or to reveal patterns in the data. Unsupervised learning is, therefore, helpful framing research questions.

2. Perform a scaled PCA on the data

```
pr.out <- prcomp(USArrests, scale = T)
```

a. (4 points) Explain in your own words what information is provided by the following elements that are the results of the prcomp function: "sdev", "center", "scale" and "x."

"Sdev" is is the standard deviation of each of the principal components. Since the first principal component runs in the direction of the highest variance, one would expect it to have the highest standard deviation. "Center" equals the variable means that are subtracted from the original data to "center" it. "Scale" equals the standard deviations of each variable that are used to scale the observations. By scaling and centering the observations, data on different scales can be analyzed on the same scale – namely, the number of standard deviations from the mean (Z-score). "X" equals the centered and squared data multiplied by the rotation matrix. In other words, x equals the data for each state "plugged in" to the four principal components, which are themselves linear combinations of the variables.

b. (2 points) Calculate the PVE for each of the principal components and the PVE by the first 2 PCs combined. Show your calculations.

'The first two principal components account for 86.7% of the variance in the data.

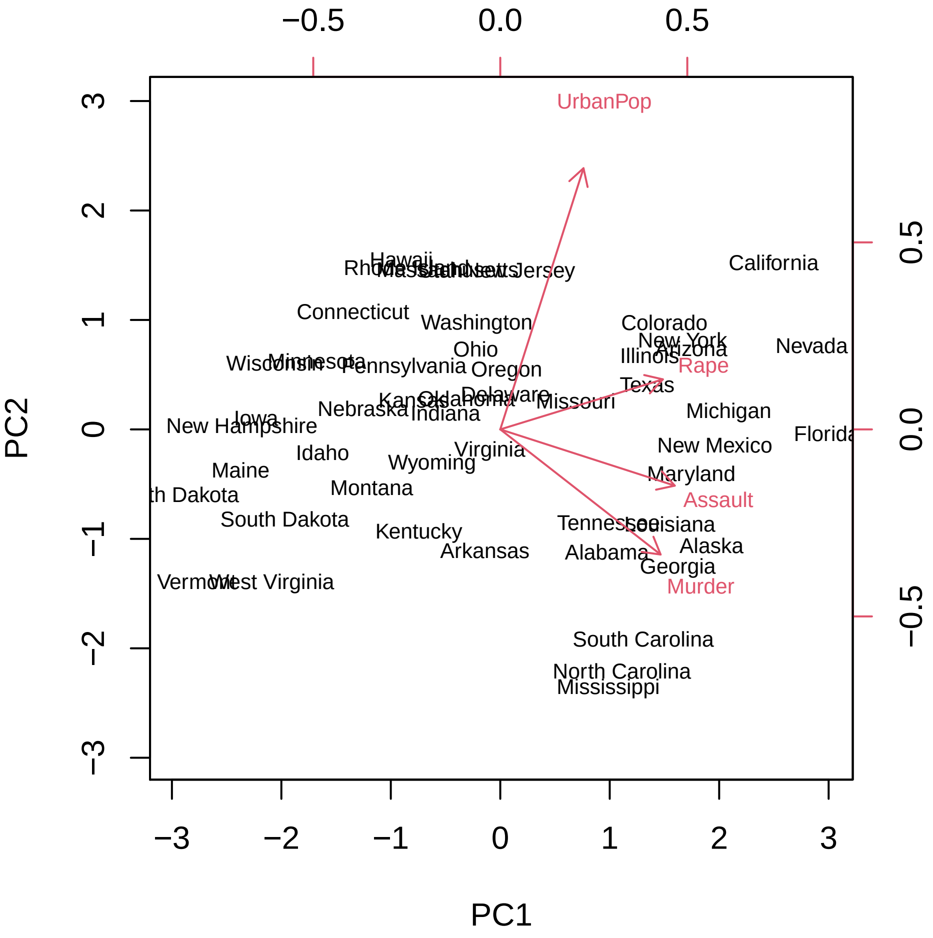
```
pr.var <- pr.out$sdev^2
pve <- round(pr.var / sum(pr.var),3)
as.data.frame(cbind(PC=c("PC1","PC2","PC3","PC4"),PVE = pve, CUM.PVE = cumsum(pve)))%>%
  kbl()%>%
  kable_styling(bootstrap_options = c("striped", "condensed"))
```

PC	PVE	CUM.PVE
PC1	0.62	0.62
PC2	0.247	0.867
PC3	0.089	0.956
PC4	0.043	0.999

c. (5 points) Display a biplot of the first 2 PCs and interpret and the describe the plot in your own words. That is, tell a story about what the plot says to you about patterns or groupings in the data.

The following plot displays two principal components. PC1 emphasizes variables having to do with crime statistics (Rape, Assault, Murder). PC2 emphasizes urban population. So, the states on the right side of the biplot have high crime rates, while the states toward the top of the biplot have high percentages of urban population. Southern states (e.g., North Carolina, South Carolina, and Mississippi) tend to fall in a high crime rate, low urban population group. Northern states (e.g., New Jersey, Rhode Island, Connecticut) tend to fall in a low crime rate, high urban population group. California has both high crime rates and a high percentage of urban population. West Virginia has low crime rates and a low percentage of urban population.

```r
pr.out$x <- -pr.out$x
pr.out$rotation <- -pr.out$rotation
biplot(pr.out, scale = 0,cex=.65)
```



d. (3 points) Describe in your own words what PCA can be used for and how it accomplishes this.

PCA can be helpful in determining underlying patterns in the data. It finds a low dimensional representation that accounts for most of the variation in the observations. The first loading vector runs in the direction of the greatest variation in the data, and it is closest to the data points (as measured by squared euclidean distance). The second loading vector is orthogonal to the first, the third vector is orthogonal to the second, and so on. In this way, relatively few loading vectors can account for most of the variation in the data.

3. Perform a hierarchical clustering on the data using complete linkage. Standardize the data before you do the clustering.

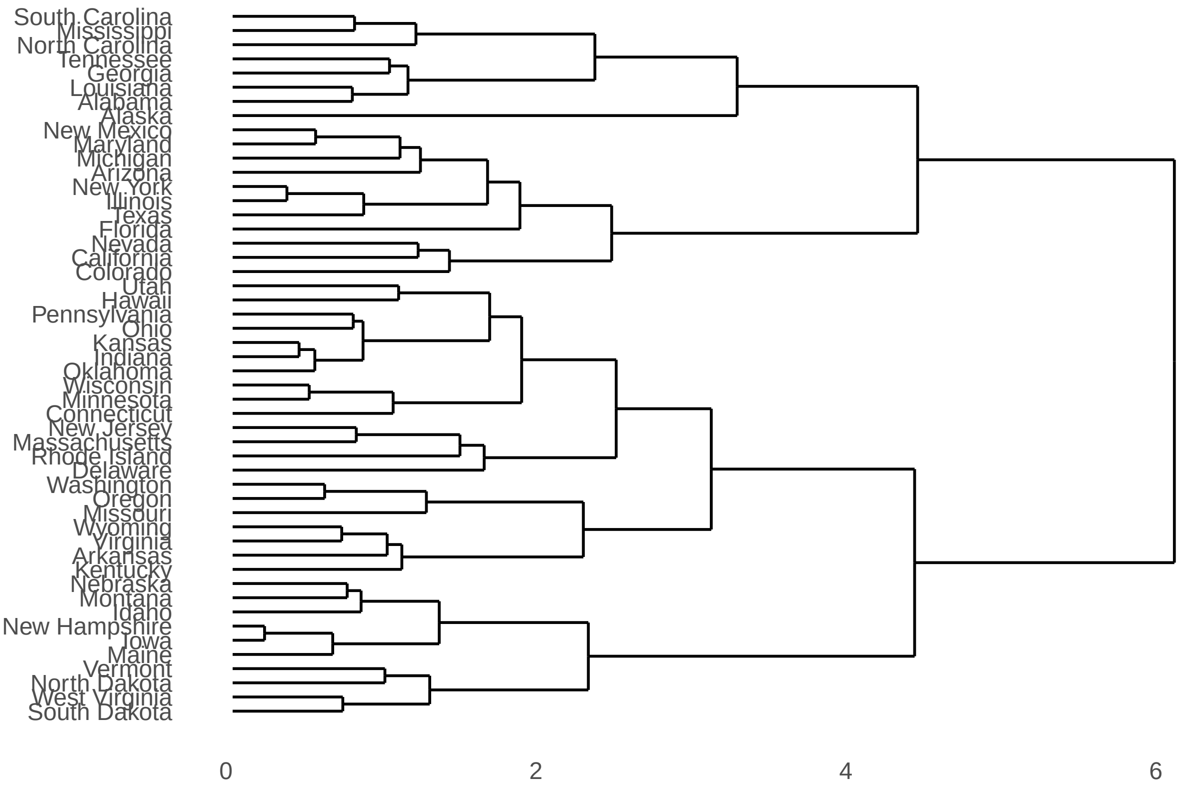
```
hc.complete <- hclust(dist(scale(USArrests)), method = "complete")
```

a. (3 points) Hierarchical clustering requires you to choose a linkage method. Explain in your own words the concept of linkage.

"Linkage" refers to various ways of assessing the dissimilarity between groups of observations. In this problem, we used "complete" linkage, which records the largest of all the dissimilarities between pairs of observations in two groups. "Single" linkage refers to the smallest pairwise dissimilarity. "Average" linkage refers to the average pairwise dissimilarity.

b. (1 point) Display a dendrogram using the ggdendrogram function.

```
ggdendrogram(hc.complete, rotate = T)
```



c. (5 points) Explain in your own words what information a dendrogram displays and how you can use it to decide what level of clustering you would use. Refer to the dendrogram to illustrate your discussion. (Remember, there is no "right" answer to number of clusters.)

A dendrogram displays the the level of similarity/dissimilarity (usually measured by Euclidean distance) between various observations. Starting with a particular observation, one calculates all the pairwise dissimilarities and selects the two observations that are the least dissimilar. The one repeats that process by measuring dissimilarities from the new cluster. The "leaves" of the dendrogram show the smallest clusters (the most similar observations). The branches show similarities between clusters. So, for example, West Virginia and South Dakota are similar observations. That cluster (WV and SD) is most similar to the cluster containing Vermont and North Dakota, and so on.

4. Perform a hierarchical clustering on the data using only the first 2 PCs from question 2 as the features. Use complete linkage.

a. (5 points) Create the dataframe or matrix that will be the input to the hclust function for this part. If nothing else, you should have 2 feature columns corresponding to PC1 and PC1 and 50 rows corresponding to the 50 states. Describe the data structure you have created. I.e., what's in the dataframe?

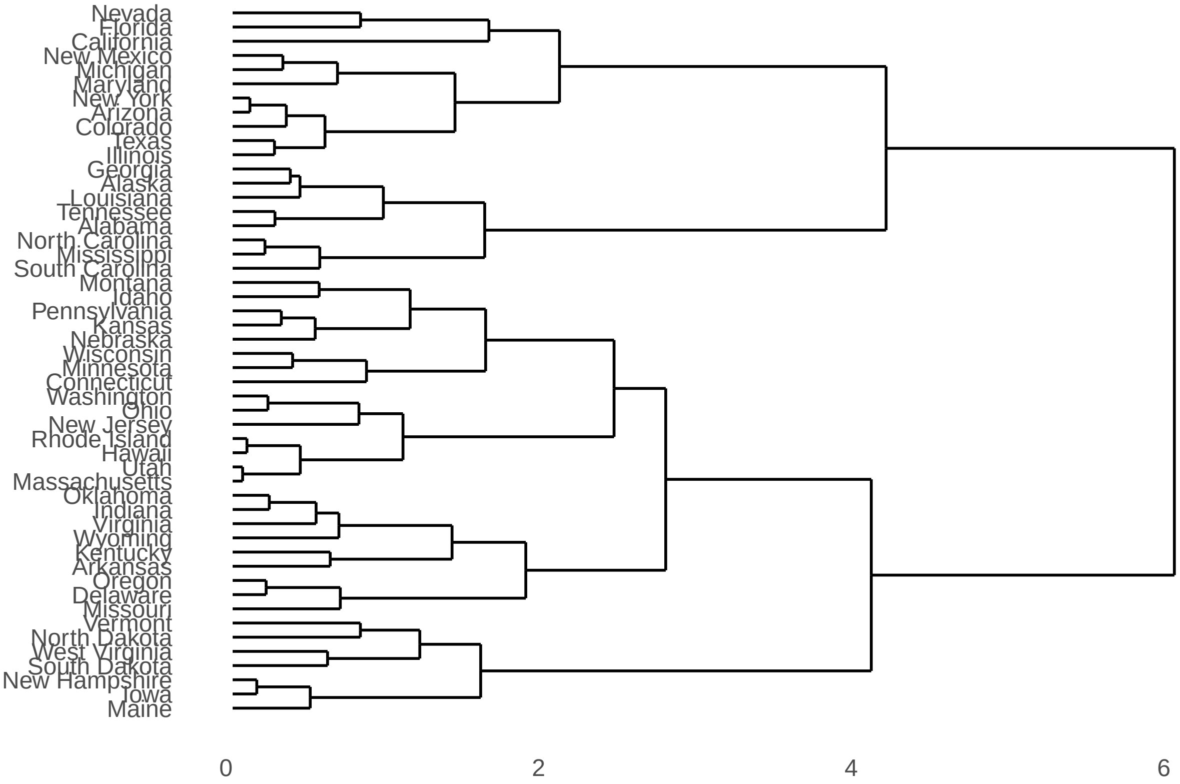
The dataframe contains 50 rows for the 50 states and two columns for PC1 and PC2. The principal components are linear combinations of the available variables. Thus the data for each state can be "plugged in" to each principle component to find a value for each PC for each state. As the CRAN literature on the prcomp function puts it, the values listed in columns 1 and 2 are "the data multiplied by the rotation matrix," though, in this case,the values for PC3 and PC4 were discarded.

```
Features<-as.data.frame(pr.out$x)%>%
  select(PC1,PC2)

hc.complete <- hclust(dist(Features), method = "complete")
```

b. (2 points) Display a dendrogram of the results using the ggdenrogram function. Make sure that the leaves of the dendrogram are labeled with the State names.

```
ggdendrogram(hc.complete, rotate = T)
```



c. (5 points) Compare this dendrogram to the one generated in question 3. Describe any similarities or dissimilarities you notice or find interesting. For example, is the pattern of clustering substantially the same, or are there noticeable changes? Explain.

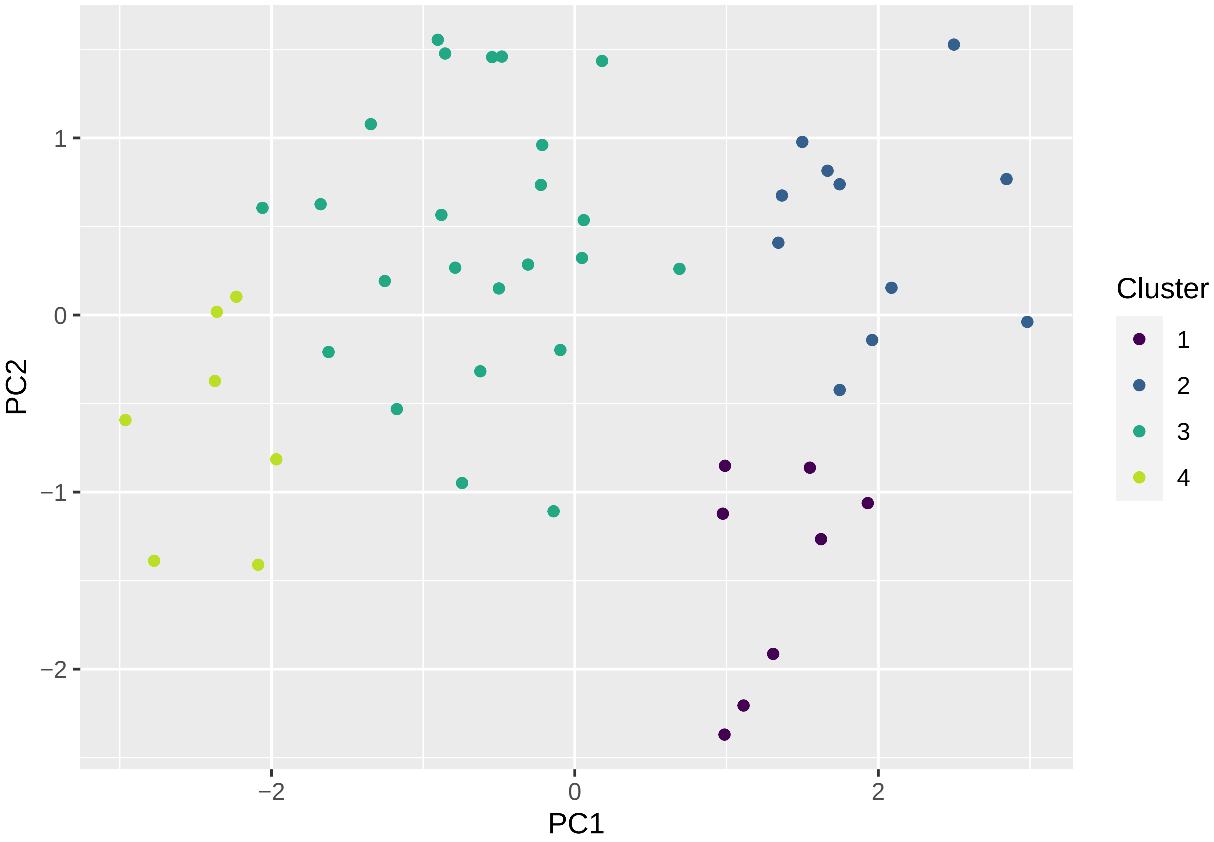
The groupings in this dendrogram are substantially the same as those in question 3. So, Nevada, Florida, California, New Mexico, Michigan, Maryland, New York, Arizona, Colorado, Texas, and Illinois are in the same group in both dendrograms. Similarly, Georgia, Alaska, Louisiana, Tennessee, Alabama, North Carolina, Mississippi, and South Carolina are in the same group.

d. (2 points) Select the number clusters, K, you think best represents the structure of the data and explain why you chose that number.

There is no “right” choice for K, but since there are four distinct “branches” or clusters in the dendrogram, I would choose K=4. K=2 would not provide much differentiation between various groups of states. K>4 might group states in ways that are too complex or less interpretable.

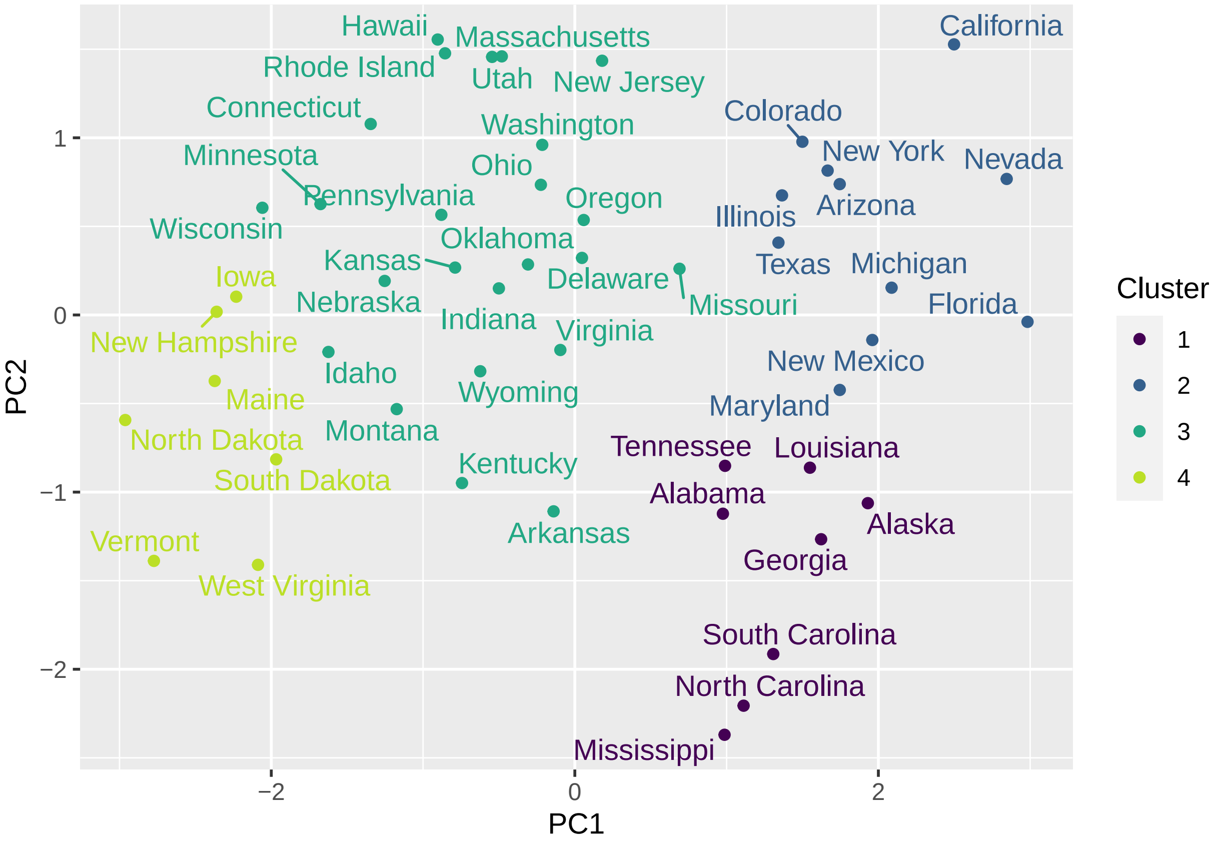
e. (5 points) Using ggplot, display a plot similar to the example below where x1 is PC 1 and x2 is PC 2, the first 2 PCs you should be using. The number of clusters you should display is the number K you selected in c above.

```
Plot<-Features%>%
  mutate(Cluster=as.factor(cutree(hc.complete,4)))%>%
  ggplot(mapping=aes(x=PC1,y=PC2,color=Cluster))+
  geom_point()+
  scale_color_viridis(discrete=T,end=.9)
Plot
```



### f. (5 points) For each of the K clusters you chose name the States that are members of the cluster. You can do this by literally listing the names, or if you know how, by labeling the points on the plot itself. (Don't worry too much about overplotting of the names.)

```
Plot+geom_text_repel(aes(label=rownames(Features)),show.legend = FALSE)
```



g. (5 points) Finally, write a paragraph comparing what you see or hypothesize from this clustering compared to that which you saw and discussed in the biplot you produced in parts 2 c and d above. (You are looking for patterns, groupings, similarities, possibilities, etc.)

Given that this cluster was created using the first two principle components from problem 2, one expects to see the same patterns. Southern states (e.g., North Carolina, South Carolina, Mississippi, Georgia) rank high on PC1 (crime rates) and low on PC2 (urban population). Northern states (e.g., Massachusetts, Rhode Island, Connecticut, New Jersey) rank high on PC2 (urban population) and low on PC1 (crime rates). California ranks high on both PC1 and PC2, while West Virginia ranks low on both PC1 and PC2.