Assignment #4: Cluster Analysis

Harini Anand

1. Introduction

For assignment 4, we explored the multivariate data partitioning techniques called hierarchical clustering and k-means clustering. We also explored using the output of principal component analysis as input to cluster analysis and using cluster analysis as a predictive model. We applied these techniques on three data sets – the European Employment data set, the USSTATES data set, and the RECIDIVISM data set.

2. Data

The first data set we used for cluster analysis is the European Employment data set. It captures the employment data from thirty European nations in various industry segments. The data set has 11 variables. Table 01 lists the variables and provides the description.

#	Variable	Description
1	Country	Name of the European country
2	Group	Group of countries – Eastern (Eastern European nations), EFTA (European Free Trade Association), EU (European Union), Other
3	AGR	Employment data about Agriculture
4	MIN	Employment data about Mining
5	MAN	Employment data about Manufacturing
6	PS	Employment data about Power and water supply
7	CON	Employment data about Construction
8	SER	Employment data about Services
9	FIN	Employment data about Finance
10	SPS	Employment data about Social and personal services
11	TC	Employment data about Transport and communications

Table 01: Variables in European Employment data set along with their description

In the data set, there are two categorical variables – Country and Group. The rest of the variables – AGR, MIN, MAN, PS, CON, SER, FIN, SPS, TC – use ratio scale variables since the employment data is captured as a percent.

3. Data preparation

As part of the data preparation, we checked for the presence of any records in the data with missing values. The data set shows no records with missing values.

Section A.1. in the Appendix has the R code for data preparation.

4. Basic Exploratory data analysis – Univariate plots

In this section, we present the results of the exploratory data analysis conducted on the European Employment data.

Basic EDA shows the country variable has unique values, one for each of the 30 countries.

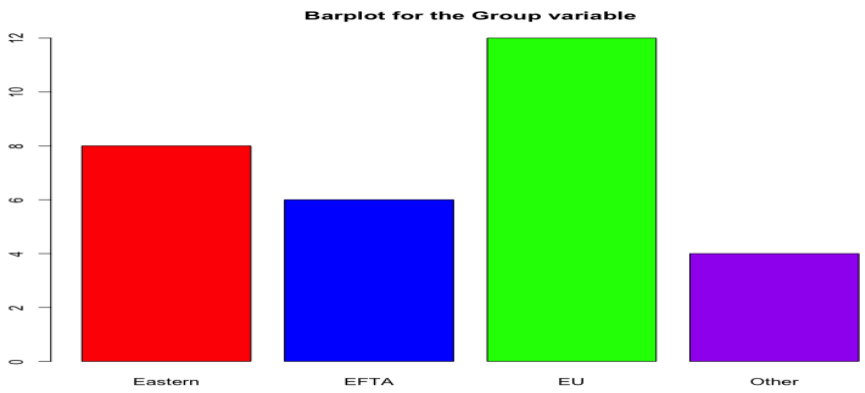

FIG 01: Barplot of the Group variable

Next, we explored the Group variable. From FIG 01, we can glean there are the most records (12 records) from the EU, followed by the Eastern group with 8 records. The EFTA group comes in third with 6 records. The other group has only 4 records.

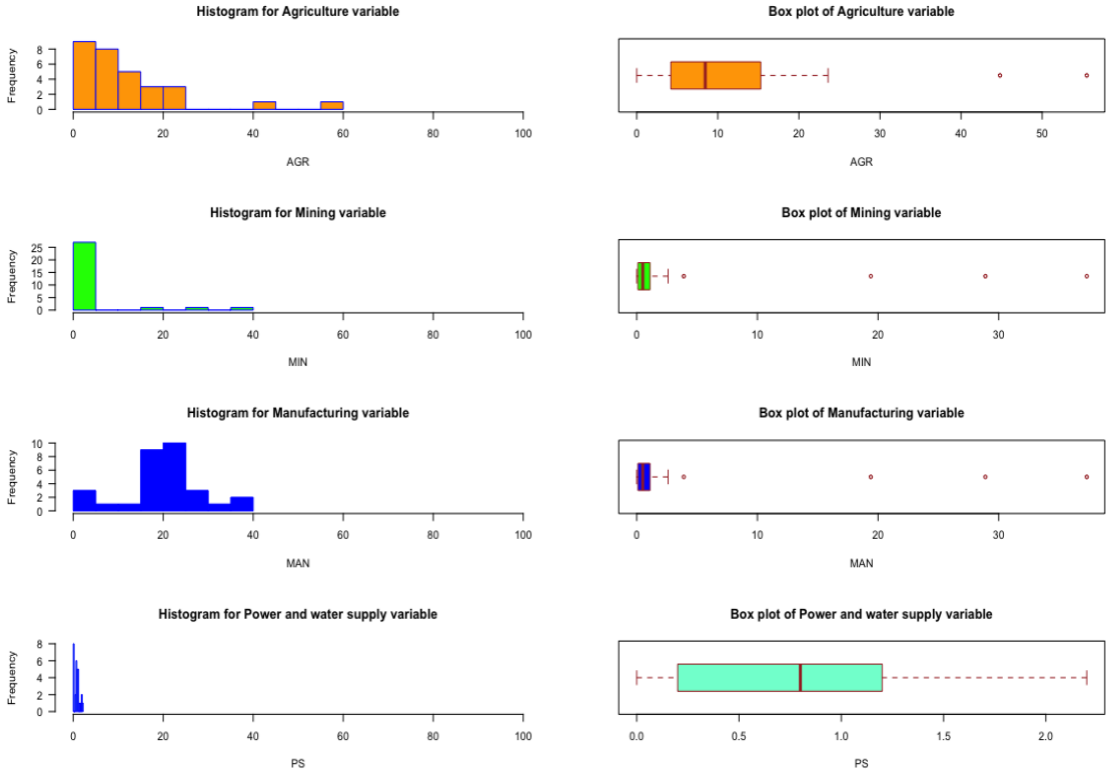

FIG 02: Histogram and boxplots of the variables – AGR, MIN, MAN, PS

Next, we ran the EDA on the variables AGR, MIN, MAN, PS. FIG 02 shows the histogram and the boxplots for the variables AGR, MIN, MAN, and PS.

The mean value for the AGR variable is 12.19, and the median value for the AGR variable is 8.45. AGR has the widest range from 0 to 55.50. The sub-range 0-5 has the most values. The AGR data is positively skewed.

The mean value for the MIN variable is 3.447, and the median value for the MIN is 0.5. The range for MIN lies between 0 and 37.30. However, most of the data lie in the range 0-5. As a result, the data is significantly positively skewed.

Next, we looked at the MAN variable, which ranges in value from 0 to 38.70. The mean is 20.29, and the median is 20.30. There is a slight negative skewness.

We also looked at the PS variable in the first set. The PS variable has the same value for the mean and the median, which is 0.8. The PS variable ranges from 0 to 2.2. But the mode is 0. The PS data is slightly positively skewed. PS has the smallest range of all the variables.

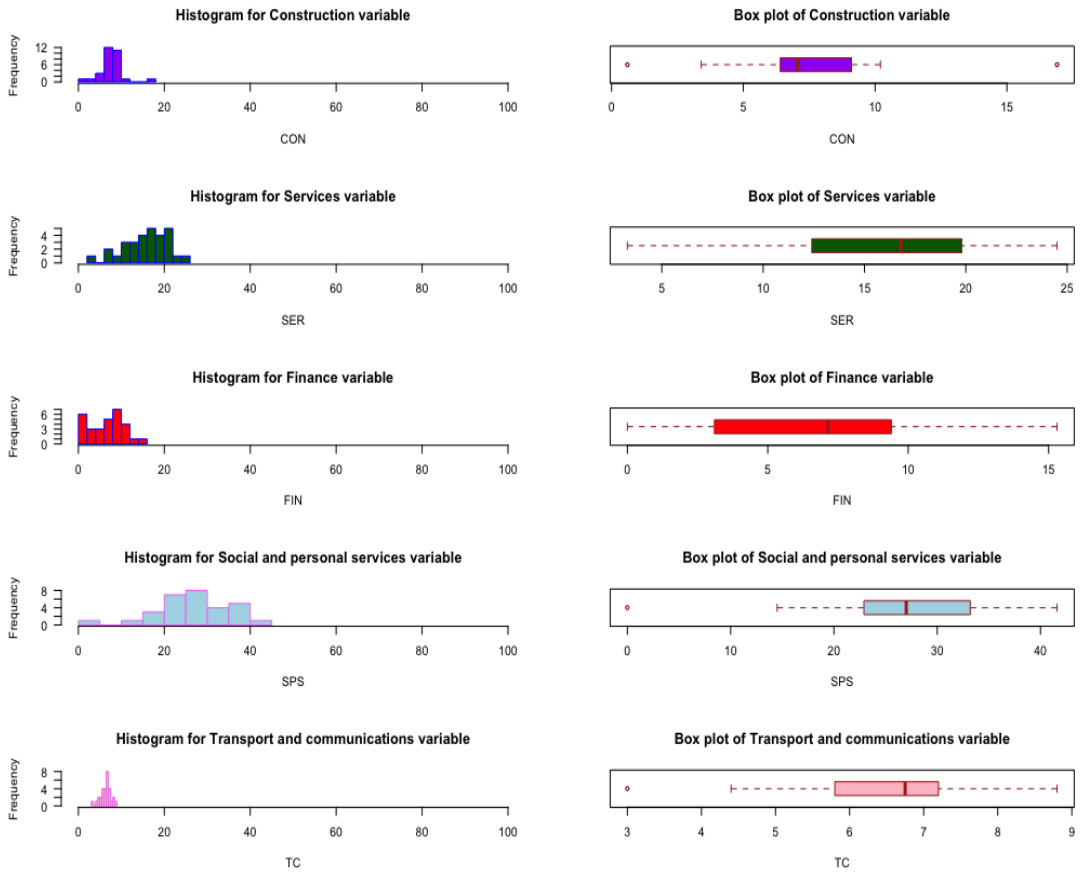

FIG 03: Histogram and boxplots of the variables – CON, SER, FIN, SPS, and TC

Next, we looked at the EDA on the variables CON, SER, FIN, SPS, and TC. FIG 03 shows the histogram and the boxplots for the variables.

The range of the CON variable is 0 to 16.90. The data is slightly positively skewed. The data has two modes. The mean of the CON variable is 7.53, and the mode is 7.05.

The range for the SER variable is 3.30 to 24.50. The SER variable has a mean of 15.64, and the mode is 16.80. The data for SER is slightly negatively skewed.

The FIN variable has a range from 0 to 15.30, with a mean value of 6.65 and a median value of 7.15. The FIN variable has very slight negative skewness.

For the SPS variable, it has the second widest range of the variables from 0 to 41.60. The mean value is 26.99, and the median is 27.00. The data is very slightly negatively skewed.

The last variable we looked at is the TC variable. The TC variable ranges only from 3.0 to 8.8 (the second smallest range of values). The mean is 6.453, and the median is 6.750. The TC variable is also very slightly negatively skewed.

Section A.2. in the Appendix has the R code for basic exploratory data analysis – univariate plots.

5. Basic Exploratory data analysis – Bivariate plots

In this section, we present the results of the pairwise plot. We excluded the categorical variables – Country and Group – from the scatterplot. **Using the pairwise plot, we tried to determine if there are any clusters or groups of points with clear partitions. For cluster analysis, these groups are interesting.** For example, in FIG 04, there are some plots such as AGR vs. FIN, SER vs. TC, MAN vs. SER, and SER vs. FIN, which show some clusters of points. **MAN vs. SER and SER vs. FIN plots look interesting as there appear at least two clusters of points in the plots.** These are explored further in the next section.

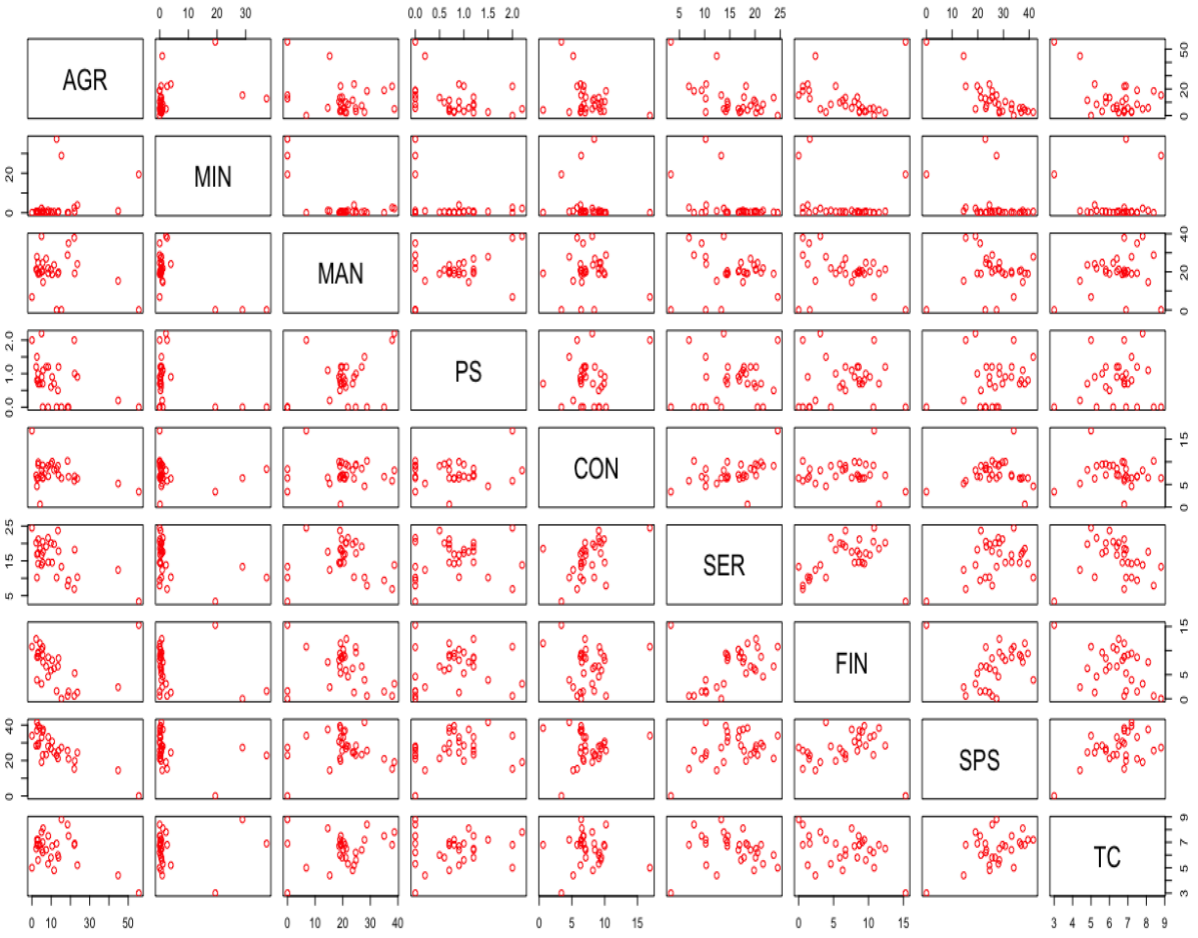

FIG 04: Pairwise Scatterplot of the European Employment continuous variables

Section A.3. in the Appendix has the R code for basic exploratory data analysis – Bivariate plots.

6. Visualizing the Data with Labelled Scatterplots

In this section, we present the results of the specialized plots with added labels and colors. The objective of this was to compress more than two dimensions of information into a two-dimensional plot.

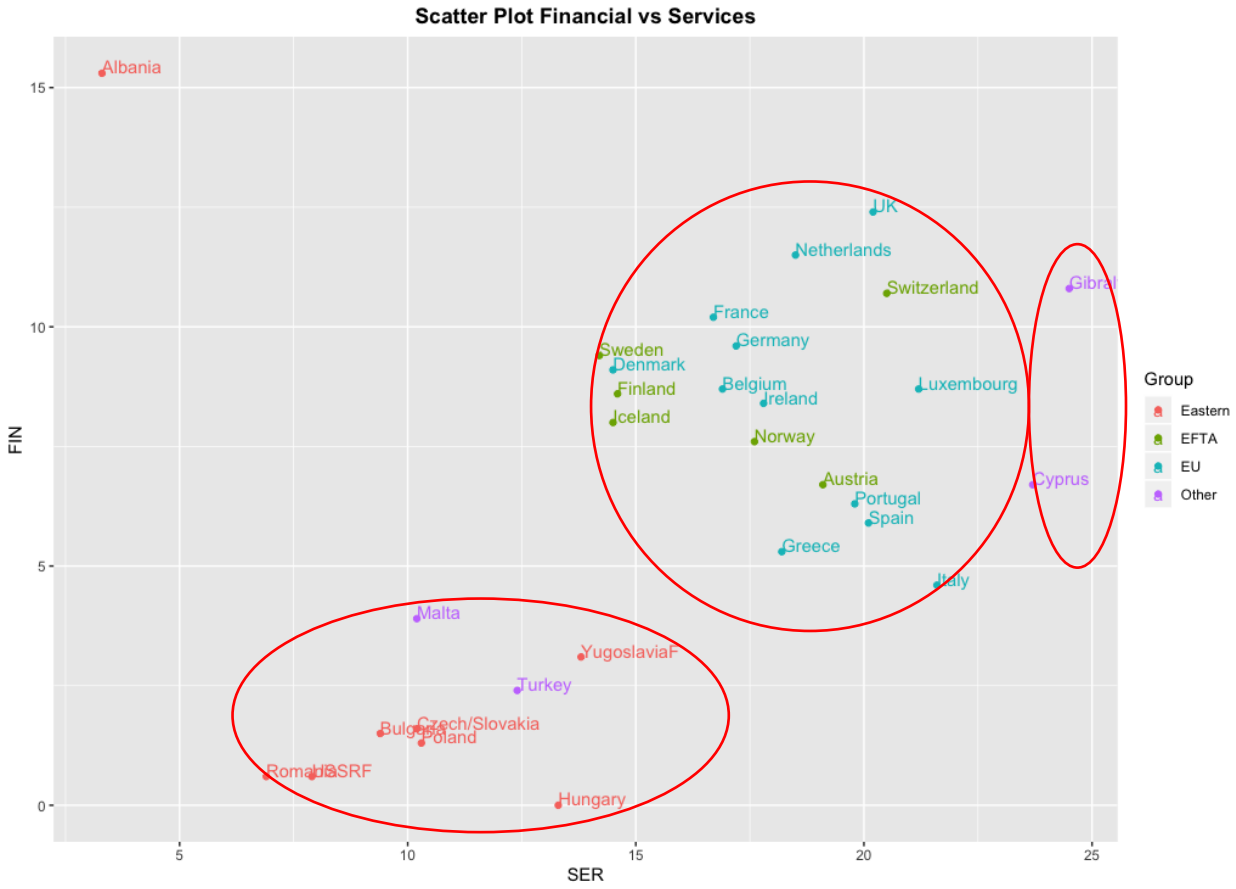

FIG 05: FIN vs SER plot

From FIG 05, we notice three clusters of points. The cluster to the right middle half of the plot has all countries in the EU group and all the countries in the EFTA group. It does not have any countries in the Eastern group or the countries in the Other group. The second cluster to the bottom lower half of the plot (below the FIN=4 line) has all the countries in the Eastern group and half of the countries in the Other group (Malta and Turkey). A third smaller cluster is to the right which includes Gibraltar and Cyprus from the Other group.

Segmentation is the process of putting items into groups based on similarities. If we applied segmentation, we would have five partitions or clusters. The similarity attribute is the value of the Group attribute (to which a country belongs to). One cluster is for only the Eastern countries, one is for only the EU countries, one cluster is for the EFTA countries, and two clusters for the Other group (due to the way the four countries are scattered far apart on the plot).

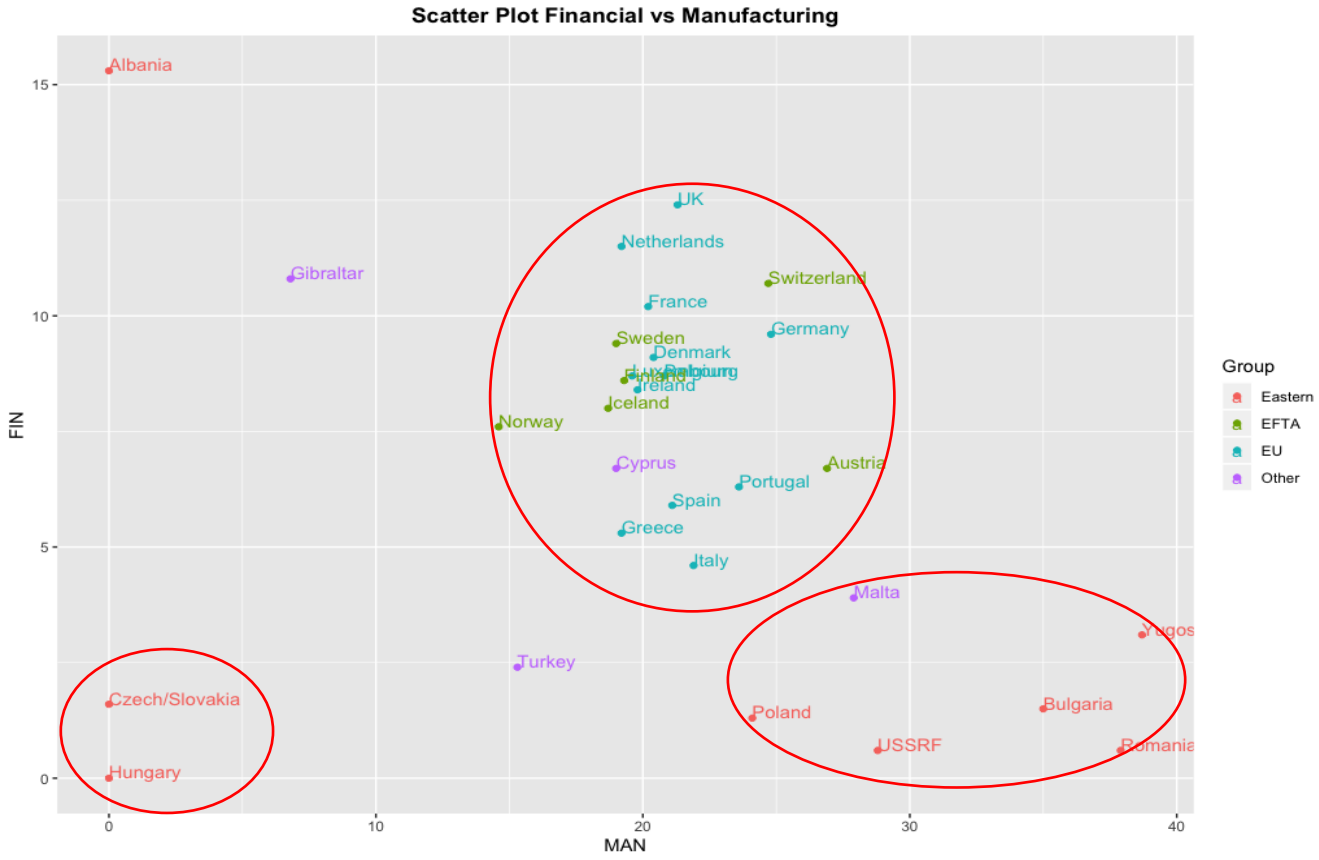

FIG 06: FIN vs MAN plot

In the case of the FIN vs. MAN plot (shown in FIG 06), we notice three clusters of points. One cluster in the center of the plot which encompasses all the countries in EU, EFTA, and Cyprus in the Other group of countries. A second cluster is below towards the right bottom of the graph, which includes Malta in the Other group countries and the Eastern countries excluding Czech/Slovakia and Hungary. The third smaller cluster is below towards the left bottom of the graph, which includes the remaining Eastern countries, namely Czech/Slovakia and Hungary. The clusters are not the same as what we obtained with the previous FIN vs. SER plot.

If the segmentation method is applied, we would have four partition groups or clusters – one with all the EU countries, one with all the EFTA countries, one with all the Eastern countries, and a cluster with the Other countries. The similarity attribute used in the segmentation is the group attribute.

Of the two 2D views of the data, for supervised clustering, the FIN vs. SER provides a better view. This is because the groups of countries can be clearly demarcated in FIN vs. SER than in the case of FIN vs. MAN. In the FIN vs. MAN plot, the EU countries overlap significantly with the EFTA countries. Therefore, in the case of the FIN vs. MAN plot, it would be hard to assign the countries to the correct class/label (group). It is easier to achieve that in the case of FIN vs. SER plot. Based on that, we concluded that FIN vs. SER plot is better for assigning the group/class labels.

Section A.4. in the Appendix has the R code for visualizing the data using labelled scatterplots.

7. Principal Component Analysis

In this section, we present the results of the principal component analysis (PCA), used to reduce the number of dimensions from nine dimensions to two dimensions. The scores of the first two dimensions are used to obtain a new 2D plot of the data.

PCA without standardization

First, we attempted the PCA on the original data without any standardization. The type of data we have is called compositional data. The percent values of all the nine industry segments (dimensions) for a row sum up to 100. Each dimension represents a component of the economy. All the nine dimensions excluding the Country and Group dimensions in the original data are continuous as they are percent values.

The PCA was conducted using the **covariance matrix** and using only the continuous variables (AGR, MIN, MAN, PS, CON, SER, FIN, SPS, TC). We then computed the scores for all the observations corresponding to the first two principal components. We plotted the obtained scores for the observations by adding the colors based on the group variable of the observation and labels based on the country variable of the observation.

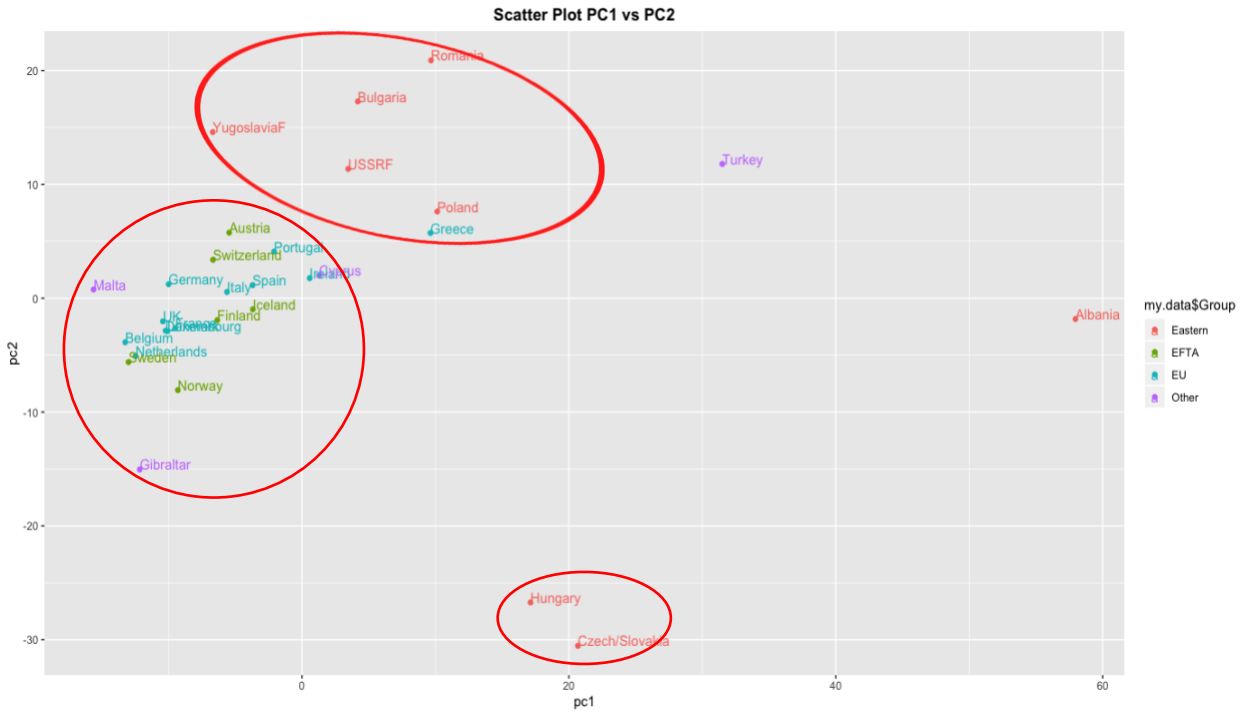

FIG 07: Scatterplot of PC1 vs PC2 scores with no data standardization

In FIG 07 (scatterplot of PC1 vs. PC2 scores with no data standardization), we can glean there are three clusters:

- one cluster with all of the EFTA and most of the EU countries (Greece is excluded), and with Malta, Gibraltar, and Cyprus countries in the Other group,
- a second cluster with Greece in the EU group and all the Eastern countries except for Hungary and Czech/Slovakia
- a third smaller cluster with Hungary and Czech/Slovakia in the Eastern countries.

The clusters identified in FIG 07 are somewhat similar to FIG 06 (FIN vs. MAN plot) in terms of the number of clusters and the positions of the countries but contrasts considerably with FIG 05 (FIN vs. SER plot).

PCA with standardization

Next, we attempted PCA on the original data with data standardization (mean zero with unit variance). Upon obtaining the PCA scores corresponding to the first two principal components, we plotted the scores with the Group variable value as the color and with the Country variable value as the label.

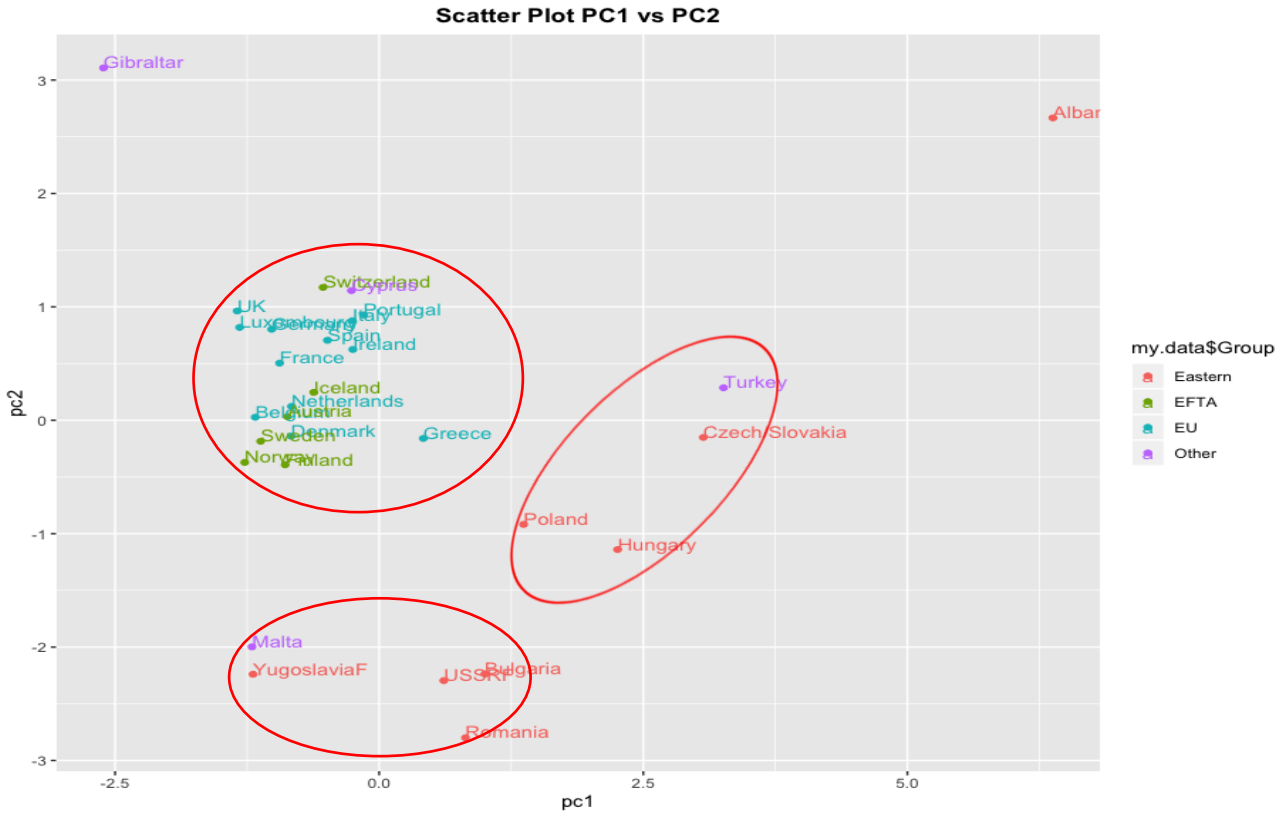

FIG 08: Scatterplot of PC1 vs PC2 scores with data standardization (mean 0 and sd 1)

In FIG 08 (scatterplot of PC1 vs. PC2 scores with data standardization), we can glean there are three clusters:

- one cluster with all of the EFTA and all the EU countries, and with Cyprus in the Other group,

- a second cluster with the most of the Eastern countries (excluding Poland, Hungary, Czech/Slovakia) and Malta in the other group,
- a third smaller cluster with Poland, Hungary, and Czech/Slovakia in the Eastern countries and Turkey in the Other group.

Even with the standardization, the number of clusters noted in the scatterplot of the scores is the same. There is some rearrangement of the countries. Particularly, in FIG 08, the Eastern countries are split more than in the scatter plot of the PCA scores without standardization. The clusters from the scatter plot with PCA scores without standardization are more cohesive. Therefore, in the case of European Employment data set, we will use the PCA components without standardization.

Section A.5. in the Appendix has the R code for principal component analysis.

8. Hierarchical Clustering Analysis

In this section, we present the results of the hierarchical clustering analysis method. We had used this method twice – once with the original data and a second time with the PCA scores obtained without standardization.

Hierarchical clustering using the original data

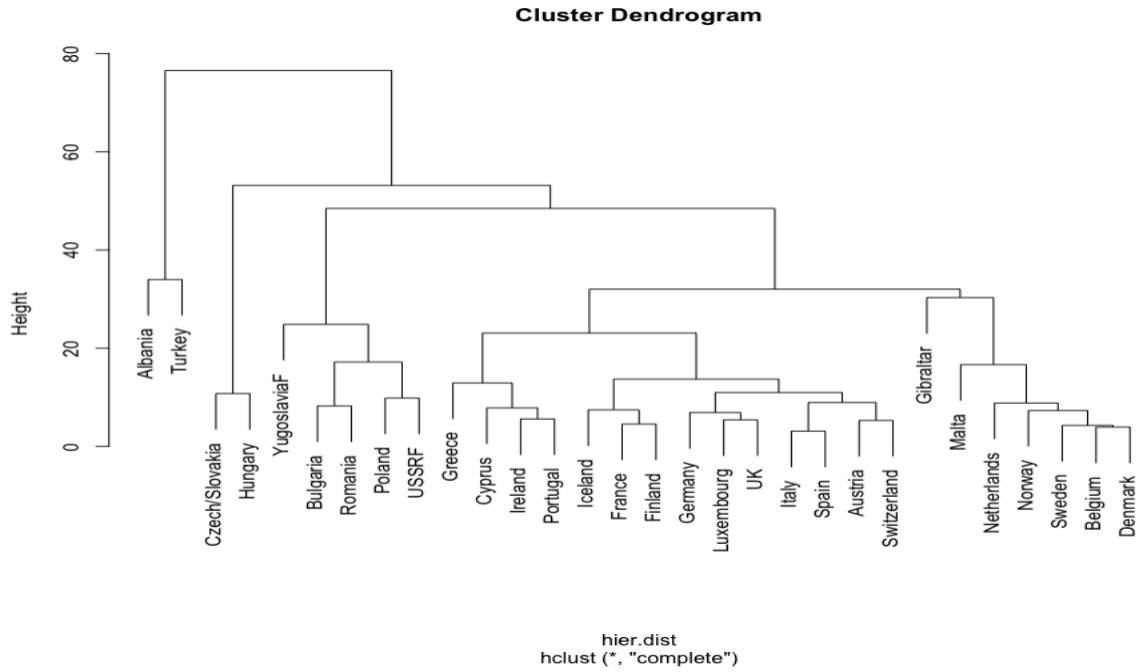

FIG 09: Dendrogram - Hierarchical Clustering on the original European Employment data

In a hierarchical classification, the data are NOT partitioned into a particular number of clusters at a single step. Instead, using a series of merges, the algorithm runs from n clusters, each containing a single individual until a single cluster containing all individuals is obtained. The result is a tree-based representation of the objects, named a ***dendrogram***. This method is called the agglomerative method of hierarchical clustering. Several linkage methods are available. We used the R method ***hclust()*** with ***"complete"*** linkage where the inter-group distances are defined as the largest distance between any two individuals, one from each group. FIG 09 shows the dendrogram created using the hierarchical

clustering on the original data, excluding the Country and the Group variables (only includes the continuous variables).

Using k=3 for cutree()

We used k=3 to cut the dendrogram into three groups. The cut height is 3. The resultant groups define the partitions in such a way that the clusters below that height in the dendrogram are distant from each other by at least that amount. By combining the ***cutree()*** output with the Country and the Group data, we obtained Table 02.

No.	Country	Group	cut.3	No.	Country	Group	cut.3
1	Belgium	EU	1	16	Norway	EFTA	1
2	Denmark	EU	1	17	Sweden	EFTA	1
3	France	EU	1	18	Switzerland	EFTA	1
4	Germany	EU	1	19	Albania	Eastern	2
5	Greece	EU	1	20	Bulgaria	Eastern	1
6	Ireland	EU	1	21	Czech/Slovakia	Eastern	3
7	Italy	EU	1	22	Hungary	Eastern	3
8	Luxembourg	EU	1	23	Poland	Eastern	1
9	Netherlands	EU	1	24	Romania	Eastern	1
10	Portugal	EU	1	25	USSRF	Eastern	1
11	Spain	EU	1	26	YugoslaviaF	Eastern	1
12	UK	EU	1	27	Cyprus	Other	1
13	Austria	EFTA	1	28	Gibraltar	Other	1
14	Finland	EFTA	1	29	Malta	Other	1
15	Iceland	EFTA	1	30	Turkey	Other	2

Table 02: cluster assignment to the observations in the European Employment data set based on k=3 cutree

	1	2	3
Eastern	5	1	2
EFTA	6	0	0
EU	12	0	0
Other	3	1	0

Table 03: cluster assignment to Groups in the data set based on k=3 cutree

Table 02 and Table 03 shows all of the EFTA, EU, all but one country (Turkey) in the Other group, and five of the Eastern group countries as part of the cluster 1. Cluster 2 has two countries – Albania in the Eastern group and Turkey in the Other group. Cluster 3 has two countries (Czech/Slovakia and Hungary) in the Eastern group.

Next, we computed the accuracy of the k=3 cluster solution. For this, we calculated With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and percent BSS.

The definitions for the metrics are:
With-in-Sum-of-Squares (WSS): WSS is the total distance of data points from their respective cluster centroids.
Total-Sum-of-Squares (TSS): TSS is the total distance of data points from the global mean of data. For a given dataset, this quantity is going to be constant. The TSS is independent of the clustering.
Between-Sum-of-Squares (BSS): BSS is the total weighted distance of various cluster centroids to the global mean of data.
Percent BSS provides the total variance in the data set explained by the clustering solution.

Table 04 shows the computed values for WSS, TSS, BSS, and Percent BSS for k=3 hierarchical clustering solution. **For homogenous clusters, WSS should be lower, and BSS should be higher.**

Metric	Values for k=3 (original data - Hierarchical Clustering)
TSS	12981.5
WSS	5331.018
BSS	7650.482
BSS Percent	0.5893374

Table 04: classification accuracy for k=3 cutree

Using k=6 for cutree()

We used k=6 to cut the dendrogram into six groups. The cut height of the dendrogram is 6. The resultant groups define the partitions in such a way that the clusters below that height in the dendrogram are distant from each other by at least 6 units. By combining the ***cutree()*** output with the Country and the Group data, we obtained the Table 05. Table 06 shows the cluster assignment per the Group variable.

No.	Country	Group	cut.6	No.	Country	Group	cut.6
1	Belgium	EU	1	16	Norway	EFTA	1
2	Denmark	EU	1	17	Sweden	EFTA	1
3	France	EU	2	18	Switzerland	EFTA	2
4	Germany	EU	2	19	Albania	Eastern	3
5	Greece	EU	2	20	Bulgaria	Eastern	4
6	Ireland	EU	2	21	Czech/Slovakia	Eastern	5
7	Italy	EU	2	22	Hungary	Eastern	5
8	Luxembourg	EU	2	23	Poland	Eastern	4
9	Netherlands	EU	1	24	Romania	Eastern	4
10	Portugal	EU	2	25	USSRF	Eastern	4
11	Spain	EU	2	26	YugoslaviaF	Eastern	4
12	UK	EU	2	27	Cyprus	Other	2
13	Austria	EFTA	2	28	Gibraltar	Other	1
14	Finland	EFTA	2	29	Malta	Other	1
15	Iceland	EFTA	2	30	Turkey	Other	6

Table 05: cluster assignment to the observations in the data set based on k=6 cutree

	1	2	3	4	5	6
Eastern	0	0	1	5	2	0
EFTA	2	4	0	0	0	0
EU	3	9	0	0	0	0
Other	2	1	0	0	0	1

Table 06: cluster assignment to Groups in the data set based on k=6 cutree

Table 05 and Table 06 show that all of the EFTA and EU countries fall in either cluster 1 or cluster 2. All of the Eastern countries fall in clusters 3, 4, and 5. All the Other group countries fall in clusters 1, 2, and 6.

Next, we computed the accuracy of the k=6 cluster solution. For this, we calculated With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), and Between-Sum-of-Squares (BSS). Table 07 shows the computed values for WSS, TSS, and BSS for k=6 hierarchical clustering solution on the original data.

Metric	Values for k=6 (original data - Hierarchical Clustering)
WSS	2049.701
TSS	12981.5
BSS	10931.8
BSS percent	0.8421061

Table 07: classification accuracy for k=6 cutree

Comparison of the classification accuracy of k=3 and k=6 cluster tree cuts on the original data

For a good clustering solution, WSS should be lower, and BSS should be higher. By comparing the values of the accuracy metrics for k=3 and k=6, we determine that k=6 has better classification accuracy. Table 08 shows the metrics side-by-side for k=3 and k=6 hierarchical clustering solutions on the original data.

Metric	Values for k=3 (original data - Hierarchical Clustering)	Values for k=6 (original data - Hierarchical Clustering)
WSS	5331.018	**2049.701**
TSS	12981.5	12981.5
BSS	7650.482	**10931.8**
BSS percent	0.5893374	**0.8421061**

Table 08: classification accuracy for k=3 and k=6 cutree for comparison

Hierarchical clustering using the PCA components

We have used the scores of the first and second components of PCA created using the covariance matrix. We did not use the PCA results of the correlation matrix since the data for European Employment is of the type called compositional data. As a result, the standardization was not of much use.

Like before, the R method ***hclust()*** is used with ***"complete"*** linkage where the inter-group distances are defined as the largest distance between any two individuals, one from each group. FIG 10 shows the dendrogram created using the hierarchical clustering on the scores of the first and second principal components.

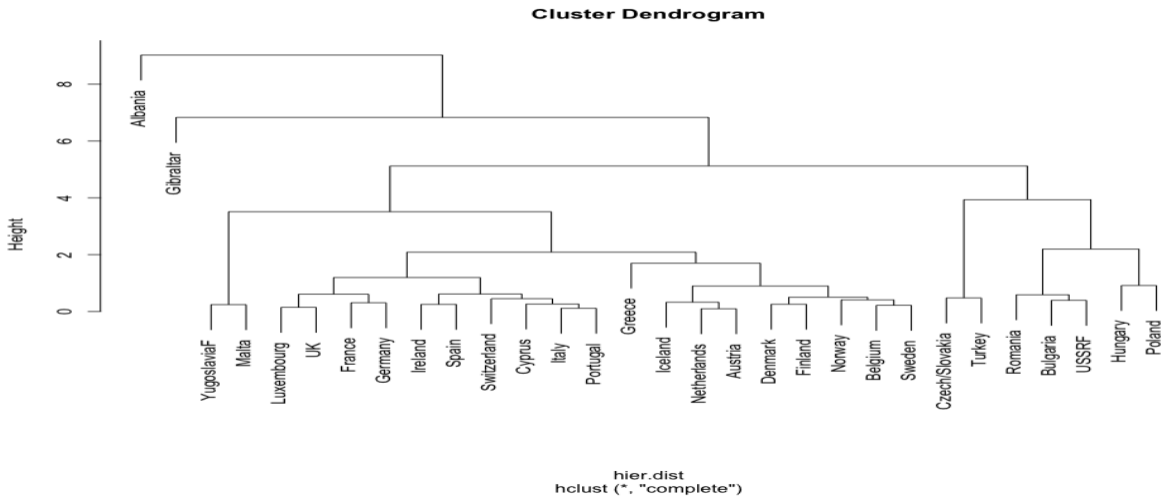

FIG 10: Dendrogram - Hierarchical Clustering – with first and second principal component scores

Using k=3 for cutree()

We used k=3 to cut the above dendrogram into three groups. The cut height is 3.

Table 09 and Table 10 show all of the EFTA, EU, all but one country in the Other group, and all but one country in the Eastern group are part of the cluster 1. Cluster 2 has only one country (Albania) in the Eastern group. Cluster 3 has only one country (Gibraltar) in the Other group. Therefore, Cluster 2 and Cluster 3 are very sparse, and 28 out of 30 countries are in one cluster, Cluster 1.

No.	Country	Group	cut.3	No.	Country	Group	cut.3
1	Belgium	EU	1	16	Norway	EFTA	1
2	Denmark	EU	1	17	Sweden	EFTA	1
3	France	EU	1	18	Switzerland	EFTA	1
4	Germany	EU	1	19	Albania	Eastern	2
5	Greece	EU	1	20	Bulgaria	Eastern	1
6	Ireland	EU	1	21	Czech/Slovakia	Eastern	1
7	Italy	EU	1	22	Hungary	Eastern	1
8	Luxembourg	EU	1	23	Poland	Eastern	1
9	Netherlands	EU	1	24	Romania	Eastern	1
10	Portugal	EU	1	25	USSRF	Eastern	1
11	Spain	EU	1	26	YugoslaviaF	Eastern	1
12	UK	EU	1	27	Cyprus	Other	1
13	Austria	EFTA	1	28	Gibraltar	Other	3
14	Finland	EFTA	1	29	Malta	Other	1
15	Iceland	EFTA	1	30	Turkey	Other	1

Table 09: cluster assignment to PCA data based on k=3 cutree

	1	2	3
Eastern	7	1	0
EFTA	6	0	0
EU	12	0	0
Other	3	0	1

Table 10: cluster assignment to Groups in the PCA data based on k=3 cutree

Next we computed the accuracy for the k=3 cluster solution using the PCA data. For this, we calculated With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and Percent BSS. Table 11 shows the computed values for WSS, TSS, BSS, and Percent BSS.

Metric	Values for k=3 (PCA data Hierarchical Clustering)
TSS	147.6449
WSS	81.69851
BSS	65.94635
BSS percent	0.4466553

Table 11: classification accuracy for k=3 cutree cluster solution using the PCA data

Using k=6 for cutree()

We used k=6 to cut the above dendrogram in FIG 10 into six groups. The cut height is 6. Table 12 and Table 13 show all of the EFTA countries, EU countries, and one country (Cyprus) in the Other group are part of cluster 1. Cluster 2 has only one country (Albania) in the Eastern group. Cluster 3 has five countries in the Eastern group. Cluster 4 has two countries - Czech/Slovakia (Eastern group) and Turkey (the Other group). Cluster 5 also has two countries – YugolsaviaF (Eastern group) and Malta (the Other group). Gibraltar (the Other group) is the only country in cluster 6.

No.	Country	Group	cut.6	No.	Country	Group	cut.6
1	Belgium	EU	1	16	Norway	EFTA	1
2	Denmark	EU	1	17	Sweden	EFTA	1
3	France	EU	1	18	Switzerland	EFTA	1
4	Germany	EU	1	19	Albania	Eastern	2
5	Greece	EU	1	20	Bulgaria	Eastern	3
6	Ireland	EU	1	21	Czech/Slovakia	Eastern	4
7	Italy	EU	1	22	Hungary	Eastern	3
8	Luxembourg	EU	1	23	Poland	Eastern	3
9	Netherlands	EU	1	24	Romania	Eastern	3
10	Portugal	EU	1	25	USSRF	Eastern	3
11	Spain	EU	1	26	YugoslaviaF	Eastern	5
12	UK	EU	1	27	Cyprus	Other	1
13	Austria	EFTA	1	28	Gibraltar	Other	6
14	Finland	EFTA	1	29	Malta	Other	5
15	Iceland	EFTA	1	30	Turkey	Other	4

Table 12: cluster assignment to PCA data based on k=6 cutree

	1	2	3	4	5	6
Eastern	0	1	5	1	1	0
EFTA	6	0	0	0	0	0
EU	12	0	0	0	0	0
Other	1	0	0	1	1	1

Table 13: cluster assignment to Groups using PCA data based on k=6 cutree

Following that, we computed the accuracy of the k=6 cluster solution using the PCA data. We computed the metrics - With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and percent BSS. Table 14 shows the computed values for WSS, TSS, BSS, and percent BSS.

Metric	Values for k=6 (PCA data Hierarchical Clustering)
TSS	147.6449
WSS	13.45312
BSS	134.1918
BSS percent	0.9088819

Table 14: classification accuracy for k=6 cutree cluster solution using the PCA data

Comparison of the classification accuracy of all four hierarchical clustering models

For a good clustering solution, WSS should be lower, and BSS should be higher. Table 15 shows side-by-side the values for WSS, TSS, BSS, BSS percent for all the four cluster models (obtained with original data and with PCA data). From Table 15, we can glean that BSS percent is the best for the k=6 cluster solution for PCA data. **Therefore, we conclude that k=6 cluster solution obtained using the PCA data has the best classification accuracy.**

Metric	Values for k=3 (original data – Hierarchical Clustering)	Values for k=6 (original data – Hierarchical Clustering)	Values for k=3 (PCA data – Hierarchical Clustering)	Values for k=6 (PCA data – Hierarchical Clustering)
TSS	12981.5	12981.5	147.6449	147.6449
WSS	5331.018	2049.701	81.69851	13.45312
BSS	7650.482	10931.8	65.94635	134.1918
BSS percent	0.5893374	0.8421061	0.4466553	**0.9088819**

Table 15: classification accuracy comparison table

Section A.6. in the Appendix has the R code for Hierarchical Clustering Analysis.

9. K-Means Clustering Analysis

In this section, we present the results of the clustering solution for European Employment using the K-means clustering method. We conducted this analysis using both the original data and the PCA scores data.

K-Means clustering using the original data (k=3)

We conducted the K-Means clustering with k=3. Table 16 shows the cluster assignment for all the observations in the original data among the 3 clusters using the K-Means method. Most of the countries are assigned to cluster 3, while cluster 2 has the least number of countries (3 countries – Albania, Czech/Slovakia, Hungary).

No.	Country	Group	k-means (k=3)	No.	Country	Group	k-means (k=3)
1	Belgium	EU	3	16	Norway	EFTA	3
2	Denmark	EU	3	17	Sweden	EFTA	3
3	France	EU	3	18	Switzerland	EFTA	3
4	Germany	EU	3	19	Albania	Eastern	2
5	Greece	EU	1	20	Bulgaria	Eastern	1
6	Ireland	EU	3	21	Czech/Slovakia	Eastern	2
7	Italy	EU	3	22	Hungary	Eastern	2
8	Luxembourg	EU	3	23	Poland	Eastern	1
9	Netherlands	EU	3	24	Romania	Eastern	1
10	Portugal	EU	3	25	USSRF	Eastern	1
11	Spain	EU	3	26	YugoslaviaF	Eastern	1
12	UK	EU	3	27	Cyprus	Other	3
13	Austria	EFTA	3	28	Gibraltar	Other	3
14	Finland	EFTA	3	29	Malta	Other	3
15	Iceland	EFTA	3	30	Turkey	Other	1

Table 16: cluster assignment using original data with K-Means clustering method (k=3)

Following that, we computed the accuracy for the k=3 K-Means cluster solution using the original data. We computed the metrics - With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and percent BSS. Table 17 shows the computed values for WSS, TSS, BSS, and percent BSS.

Metric	Values for k=3 (original data – K-Means)
TSS	12981.5
WSS	5461.366
BSS	7520.139
BSS percent	0.5792964

Table 17: classification accuracy for k=3 K-Means cluster solution using the original data

K-Means clustering using the original data (k=6)

Next, we conducted the K-Means clustering with k=6. Table 18 shows the cluster assignment for all the observations in the original data among the 6 clusters. Most of the countries are assigned to cluster 3. Clusters 1 and 4 have only two countries.

No.	Country	Group	k-means (k=6)	No.	Country	Group	k-means (k=6)
1	Belgium	EU	2	16	Norway	EFTA	2
2	Denmark	EU	2	17	Sweden	EFTA	2
3	France	EU	2	18	Switzerland	EFTA	6
4	Germany	EU	5	19	Albania	Eastern	3
5	Greece	EU	6	20	Bulgaria	Eastern	1
6	Ireland	EU	6	21	Czech/Slovakia	Eastern	4
7	Italy	EU	6	22	Hungary	Eastern	4
8	Luxembourg	EU	5	23	Poland	Eastern	1
9	Netherlands	EU	2	24	Romania	Eastern	1
10	Portugal	EU	6	25	USSRF	Eastern	1
11	Spain	EU	6	26	YugoslaviaF	Eastern	1
12	UK	EU	5	27	Cyprus	Other	6
13	Austria	EFTA	6	28	Gibraltar	Other	5
14	Finland	EFTA	2	29	Malta	Other	2
15	Iceland	EFTA	6	30	Turkey	Other	3

Table 18: cluster assignment using original data with K-Means clustering method (k=6)

Following that, we computed the accuracy for the k=6 K-Means cluster solution using the original data. We computed the metrics - With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and percent BSS. Table 19 shows the computed values for WSS, TSS, BSS, and percent BSS.

Metric	Values for k=6 (original data – K-Means)
TSS	12981.5
WSS	2195.909
BSS	10785.6
BSS percent	0.8308433

Table 19: classification accuracy for k=6 K-Means cluster solution using the original data

Comparison of the classification accuracy of K-Means models with the Hierarchical cluster models

We compared the accuracy metrics of k=3 and k=6 K-Means cluster solutions against the values obtained with Hierarchical Clustering in the previous section. As stated before, a good clustering solution should have a low value for WSS (With-in-Sum-of-Squares) and a high value for BSS. BSS Percent gives the total variance explained by the clustering solution. We chose the model with the highest BSS Percent as the best cluster model. Using values from Table 20**, we can determine that hierarchical clustering conducted on the PCA scores is still the best cluster model.**

Metric	Values for k=3 (original data – Hierarchical clustering)	Values for k=6 (original data - Hierarchical clustering)	Values for k=3 (PCA data - Hierarchical clustering)	Values for k=6 (PCA data - Hierarchical clustering)	Values for k=3 (original data – K-Means)	Values for k=6 (original data – K-Means)
WSS	5331.018	2049.701	81.69851	13.45312	5461.366	2195.909
TSS	12981.5	12981.5	147.6449	147.6449	12981.5	12981.5
BSS	7650.482	10931.8	65.94635	134.1918	7520.139	10785.6
BSS percent	0.5893374	0.8421061	0.4466553	**0.9088819**	0.5792964	0.8308433

Table 20 classification accuracy comparison table

Next, we plotted the K-Means cluster solution for k=3 and k=6. We achieve this using the ***clusplot***() R function of the cluster solution. FIG 11 shows the clusplot for k=3 K-Means cluster model with original labels, assigned clusters, and cluster centers. As noted, before, cluster 3 has the most countries (20 countries) followed by cluster 1 (7 countries). Cluster 2 has only 3 countries (Hungary, Czech/Slovakia, and Albania).

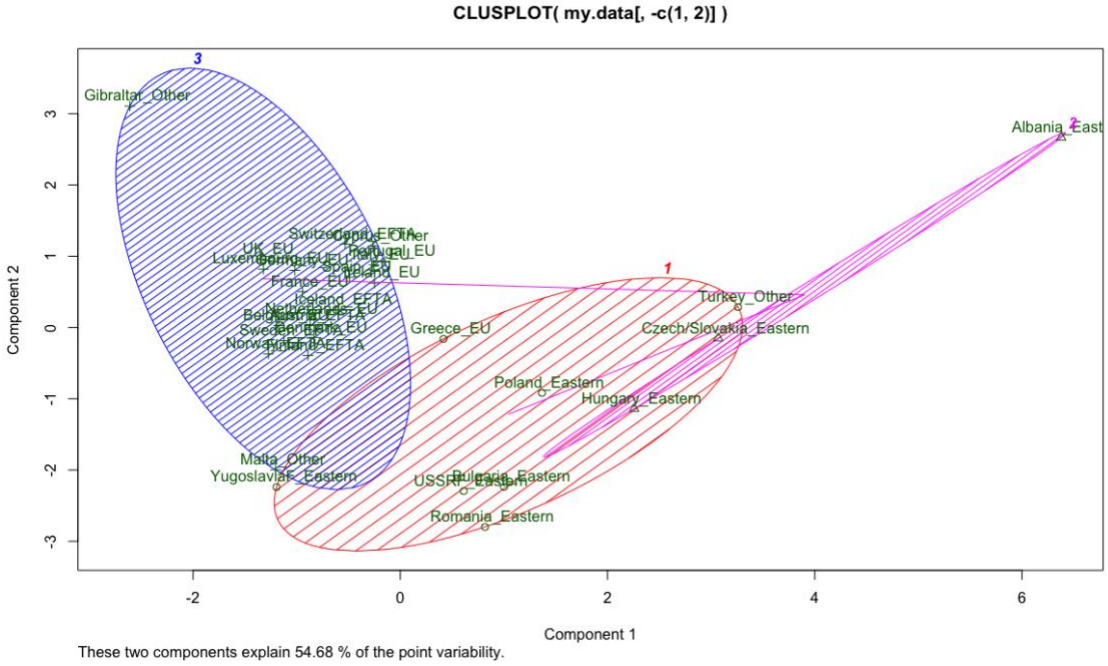

FIG 11: K-Means Clustering Solution (k=3)

Similarly, FIG 12 shows the clusplot for k=6 K-Means cluster model with original labels, assigned clusters (does not show cluster centers). FIG 13 shows the same plot with cluster centers. As previously shown in Table 16, cluster 6 has the most countries (9), followed by cluster 2 with 8 countries. Cluster 1 has 5 countries. Cluster 5 has 4 countries. Cluster 3 and 4 have only 2 countries in this solution.

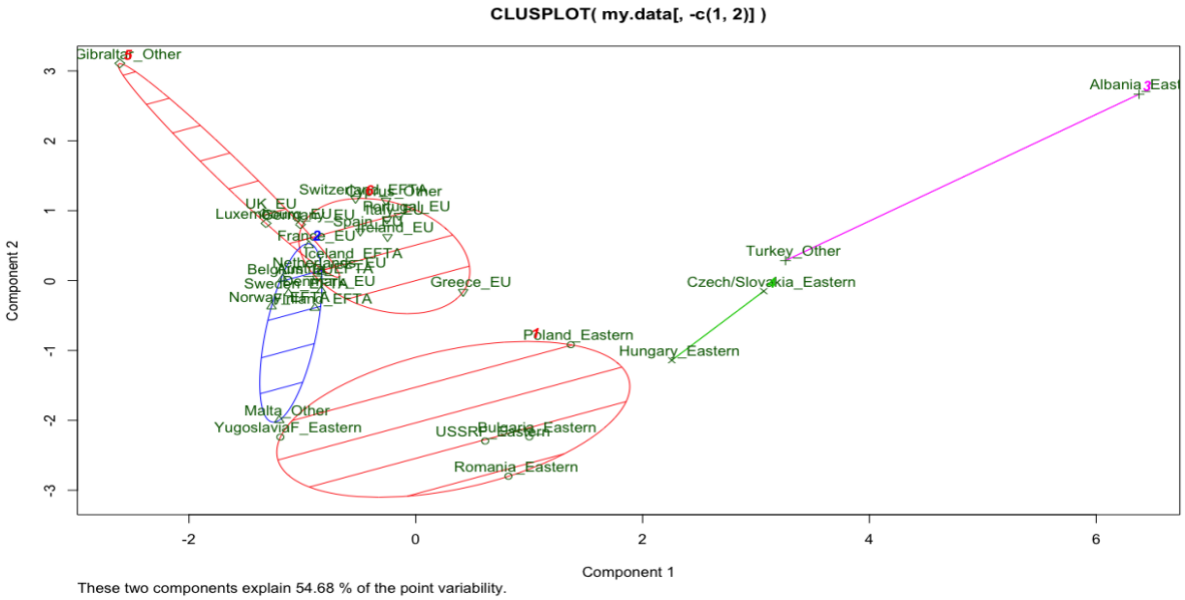

FIG 12: K-Means Clustering Solution (k=6) without cluster centers

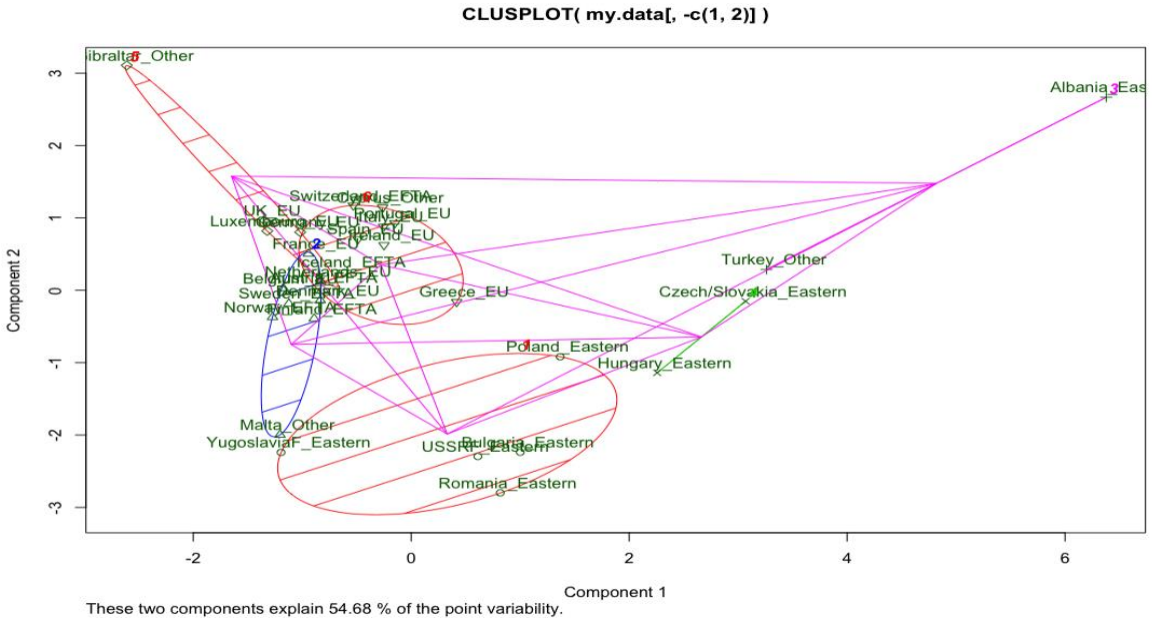

FIG 13: K-Means Clustering Solution (k=6) with lines from cluster centers drawn

K-Means clustering using the PCA scores data (k=3)

Next, we performed K-Means clustering on the PCA scores data obtained using the covariance matrix. First, we used k=3. Table 21 shows the cluster assignment for all the records in the PCA scores data among the 3 clusters. Most of the countries are assigned to cluster 2. Cluster 1 has the least number of countries with Albania, Czech/Slovakia, and Hungary. FIG 14 shows the clusplot for this solution.

No.	Country	Group	k-means (k=3)	No.	Country	Group	k-means (k=3)
1	Belgium	EU	2	16	Norway	EFTA	2
2	Denmark	EU	2	17	Sweden	EFTA	2
3	France	EU	2	18	Switzerland	EFTA	2
4	Germany	EU	2	19	Albania	Eastern	1
5	Greece	EU	3	20	Bulgaria	Eastern	3
6	Ireland	EU	2	21	Czech/Slovakia	Eastern	1
7	Italy	EU	2	22	Hungary	Eastern	1
8	Luxembourg	EU	2	23	Poland	Eastern	3
9	Netherlands	EU	2	24	Romania	Eastern	3
10	Portugal	EU	2	25	USSRF	Eastern	3
11	Spain	EU	2	26	YugoslaviaF	Eastern	2
12	UK	EU	2	27	Cyprus	Other	2
13	Austria	EFTA	2	28	Gibraltar	Other	2
14	Finland	EFTA	2	29	Malta	Other	2
15	Iceland	EFTA	2	30	Turkey	Other	3

Table 21: cluster assignment using PCA scores data with K-Means clustering method (k=3)

Following that, we computed the accuracy for the k=3 K-Means cluster solution using the PCA scores data. We computed the metrics - With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and BSS percent. Table 22 shows the computed values for WSS, TSS, BSS, and BSS percent.

Metric	Values for k=3 (PCA data – K-Means)
TSS	10524.65
WSS	3291.616
BSS	7233.03
BSS percent	0.6872469

Table 22: classification accuracy for k=3 K-Means cluster solution using the PCA scores data

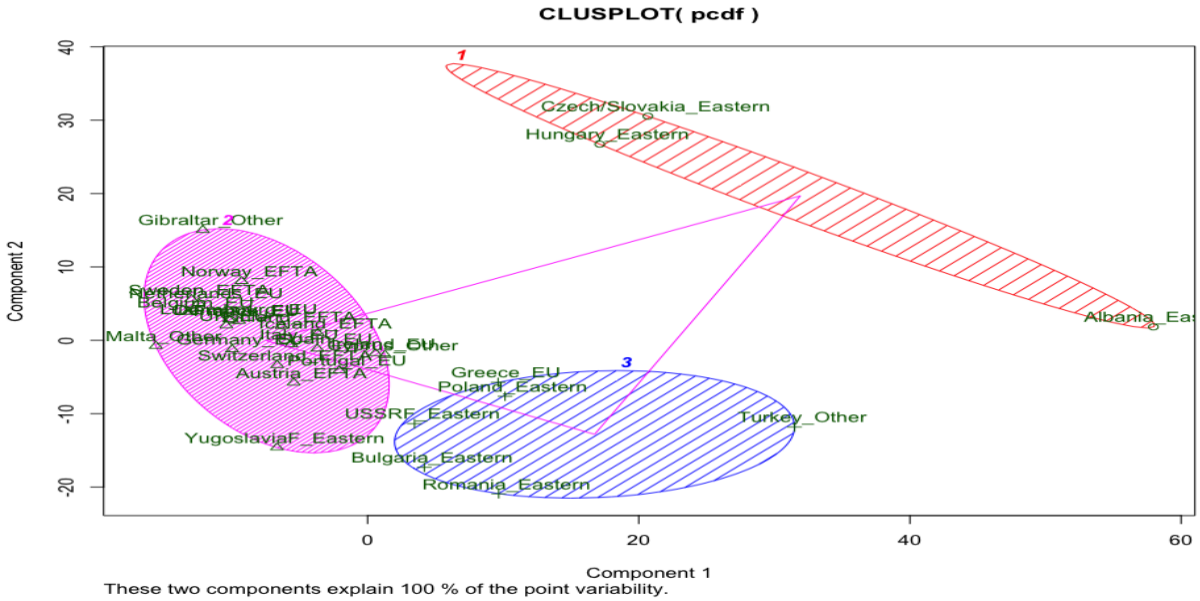

FIG 14: K-Means Clustering Solution (k=3) with PCA scores data

K-Means clustering using the PCA scores data (k=6)

We also performed K-Means clustering on the PCA scores data obtained using the covariance matrix. This time we used k=6. Table 23 shows the cluster assignment for all the records in the PCA scores data among the 6 clusters. In this solution, Cluster 4 shows the most countries (12). Cluster 5 has nine countries. The rest of the clusters are sparse.

No.	Country	Group	k-means (k=6)	No.	Country	Group	k-means (k=6)
1	Belgium	EU	5	16	Norway	EFTA	5
2	Denmark	EU	5	17	Sweden	EFTA	5
3	France	EU	5	18	Switzerland	EFTA	6
4	Germany	EU	5	19	Albania	Eastern	3
5	Greece	EU	2	20	Bulgaria	Eastern	4
6	Ireland	EU	6	21	Czech/Slovakia	Eastern	1
7	Italy	EU	6	22	Hungary	Eastern	1
8	Luxembourg	EU	5	23	Poland	Eastern	2
9	Netherlands	EU	5	24	Romania	Eastern	4
10	Portugal	EU	6	25	USSRF	Eastern	4
11	Spain	EU	6	26	YugoslaviaF	Eastern	4
12	UK	EU	5	27	Cyprus	Other	6
13	Austria	EFTA	6	28	Gibraltar	Other	5
14	Finland	EFTA	6	29	Malta	Other	5
15	Iceland	EFTA	6	30	Turkey	Other	2

Table 23: cluster assignment using PCA scores data with K-Means clustering method (k=6)

We computed the accuracy for the k=6 K-Means cluster solution using the PCA scores data. We computed the metrics – With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and BSS percent. Table 24 shows the computed values for WSS, TSS, BSS, and BSS percent.

Metric	Values for k=6 (PCA data – K-Means)
TSS	10524.65
WSS	888.016
BSS	9636.63
BSS percent	0.9156251

Table 24: classification accuracy for k=6 K-Means cluster solution using the PCA scores data

FIG 15 and FIG 16 show the clusplots for the k=6 K-Means clustering solutions on the PCA data. FIG 15 shows the clusplot with original labels, assigned clusters to the countries but does not show the cluster centers. FIG 16 shows the same plot with cluster centers.

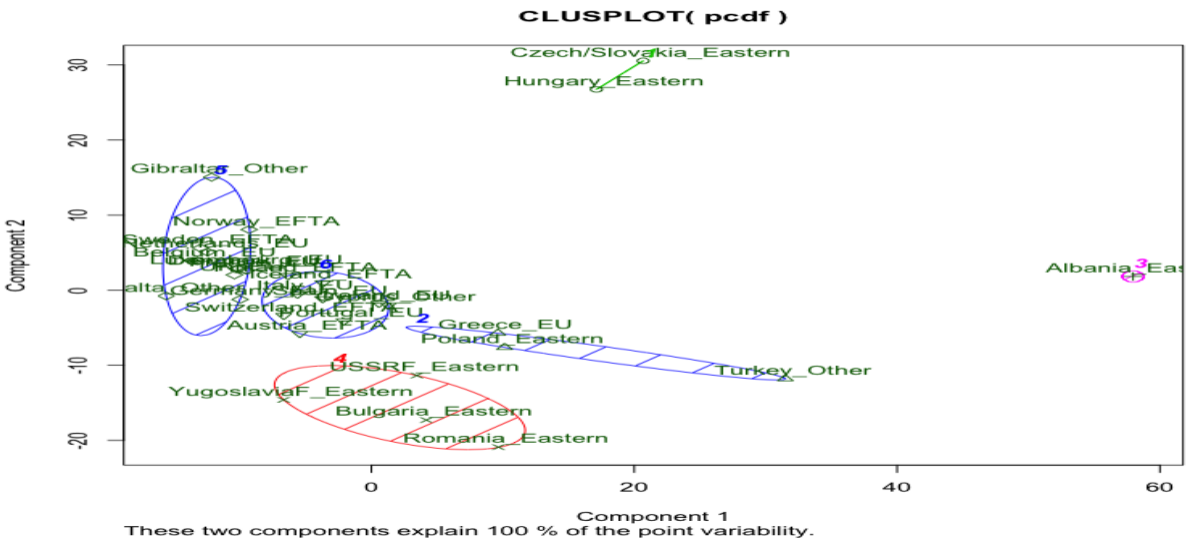

FIG 15: K-Means Clustering Solution (k=6) with PCA scores data without cluster centers drawn

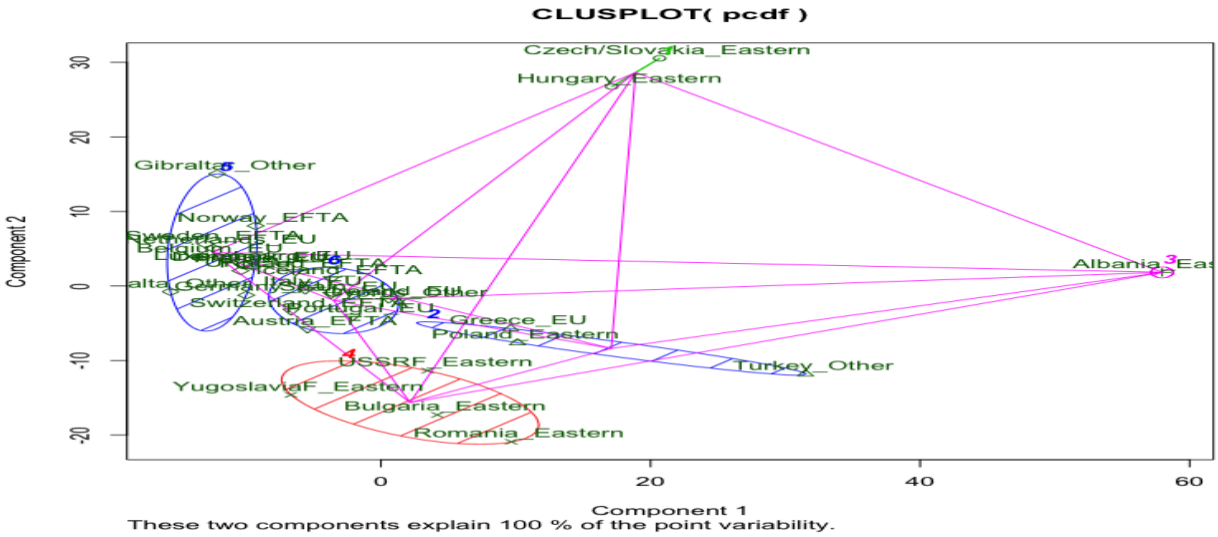

FIG 16: K-Means Clustering Solution (k=6) with PCA scores data with cluster centers drawn

Increasing the number of clusters from k=3 to k=6

Based on the computed classification accuracy metrics, increasing the number of clusters from k=3 to k=6 improves the BSS percent (BSS Percent gives the total variance explained by the clustering solution).

Comparison of classification accuracy the eight cluster models

Table 25 lists all the accuracy metric values – WSS, TSS, BSS, and BSS percent, for all the cluster models created thus far (Hierarchical and K-Means cluster modes using the original and the PCA scores data). We pick the model with the best BSS percent (percent of total variance explained by the solution). **The K-Means clustering model with k=6 generated using the PCA data still has the best BSS percent (classification accuracy) of all the eight models.**

Metric	Values for k=3 (original data – Hierarchical clustering)	Values for k=6 (original data - Hierarchical clustering)	Values for k=3 (PCA data - Hierarchical clustering)	Values for k=6 (PCA data - Hierarchical clustering)	Values for k=3 (original data – K-Means)	Values for k=6 (original data – K-Means)	Values for k=3 (PCA data – K-Means)	Values for k=6 (PCA data – K-Means)
WSS	5331.018	2049.701	81.69851	13.45312	5461.366	2195.909	3291.616	10524.65
TSS	12981.5	12981.5	147.6449	147.6449	12981.5	12981.5	10524.65	888.016
BSS	7650.482	10931.8	65.94635	134.1918	7520.139	10785.6	7233.03	9636.63
BSS percent	0.5893374	0.8421061	0.4466553	0.9088819	0.5792964	0.8308433	0.6872469	**0.9156251**

Table 25: classification accuracy comparison table – eight models

K-Means Cluster comparison based on the original labels (EU, EFTA, Eastern, Other)

Table 26 lists comparisons obtained for K-Means clusters for k=3 and k=6, based on the original data and the PCA scores data, against the Group labels (EU, EFTA, Eastern, Other).

In the case of k=3 cluster solutions, most of the countries are assigned to a single cluster. From Table 26, we can note that Cluster 3 has 20 countries in the k=3 cluster solution with the original data. Likewise, Cluster 2 has 21 countries in the cluster solution obtained with k=3 on the PCA scores data. In both k=3 cluster solutions, the EFTA, EU, and three of the four countries in the Other group are assigned to this one cluster.

When we move to k=6 solutions, we note half of the clusters have more density than the other half. For example, in the case of k=6 original data K-Means cluster solution, clusters 6, 2, and 1 have more than 70% countries (from a mix of groups) assigned. In the same solution, clusters 3, 4, and 5 are sparse. Similarly, if we consider the k=6 PCA scores data K-Means cluster solution, clusters 3 and 4 are dense, with 70% of the countries distributed between the two clusters. These also belong to a mix of the groups. Clusters 1,2,5, and 6 are sparse with only a few countries.

However, none of the cluster solutions clearly align to the original labels.

	1	2	3
Eastern	5	3	0
EFTA	0	0	6
EU	1	0	11
Other	1	0	3

K-Means on original data (k=3)

	1	2	3	4	5	6
Eastern	5	0	1	2	0	0
EFTA	0	3	0	0	0	3
EU	0	4	0	0	3	5
Other	0	1	1	0	1	1

K-Means on original data (k=6)

	1	2	3
Eastern	3	1	4
EFTA	0	6	0
EU	0	11	1
Other	0	3	1

K-Means on PCA data (k=3)

	1	2	3	4	5	6
Eastern	2	1	1	4	0	0
EFTA	0	0	2	4	0	0
EU	0	1	0	4	7	0
Other	0	1	0	0	2	1

K-Means on PCA data (k=6)

Table 26: K-Means clustering assignment per the original label (Group)

Section A.7 in the Appendix has the R code for K-Means Clustering Analysis.

10. Computing the 'Optimal' Number of Clusters by Brute Force

In this section, we run the Hierarchical clustering and the K-Means clustering methods on the original data, and the PCA scores data for k=1 to k=20. We used WSS plots and percent BSS plots. WSS (Within-cluster sum of squares) measures the compactness of the clustering. To determine the optimal number of clusters, we want to reduce or minimize the total intra-cluster variation, which is measured by WSS. We utilize the "Elbow" method for this on the WSS plots, which plots the total WSS as a function of the number of clusters. The second plot shows the graph of the percent of BSS as a function of the number of clusters. Percent BSS explains the total variance explained by the clustering solution. Our criterion is to choose the number of clusters in such a way that adding another cluster does not reduce much the total WSS or improve the percent BSS.

Optimal number of clusters with K-Means clustering method using the original data

FIG 17 shows the WSS plot as a function of the number of clusters and the percent BSS as a function of the number of clusters from k=1 to k=20. These are obtained with the K-Means clustering method using the original data. The WSS plot to the left shows a distinct drop in the within-groups sum of squares when moving from four to five clusters. After six clusters, the reduction in the within-groups

sum of squares is not much. With k=6, the % of Between SS is more than 0.8. **Therefore, we conclude that k=6 gives the optimal number of clusters for the K-Means cluster solution on the original data.**

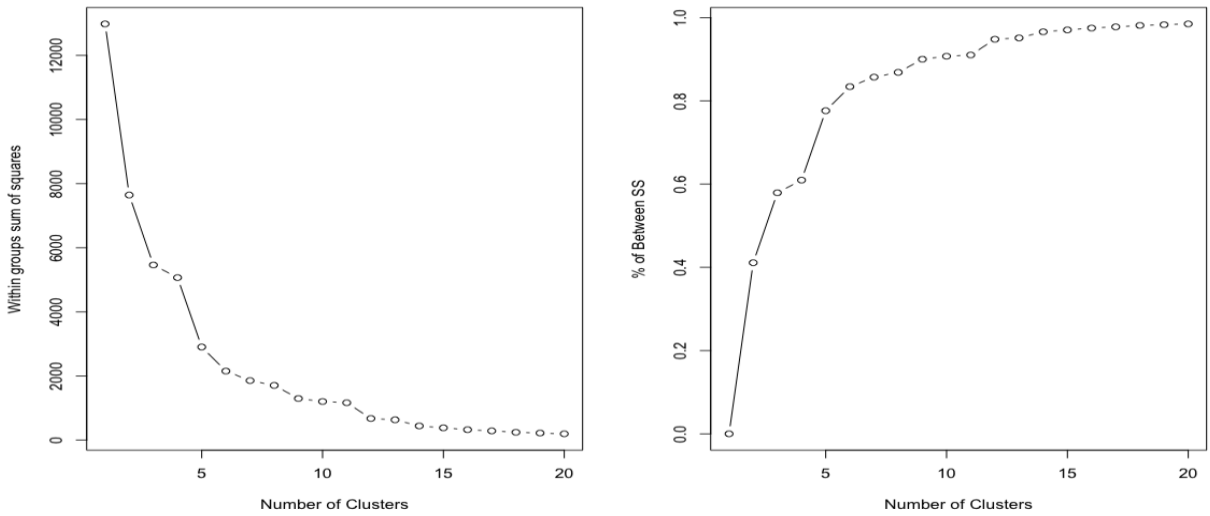

FIG 17: WSS plot and % BSS plot with K-Means Clustering method on the original data (k=1 to k=20)

Optimal number of clusters with K-Means clustering method using the PCA scores data

FIG 18 shows the WSS plot as a function of the number of clusters and the percent BSS as a function of the number of clusters from k=1 to k=20. These are obtained with the K-Means clustering method using the PCA scores data. The WSS plot to the left shows a sharp drop in the within-groups sum of squares when moving from five to six clusters. With k=6, the % of Between SS is more than 0.9. **k=6 gives the optimal number of clusters for the K-Means solution on the PCA scores data.**

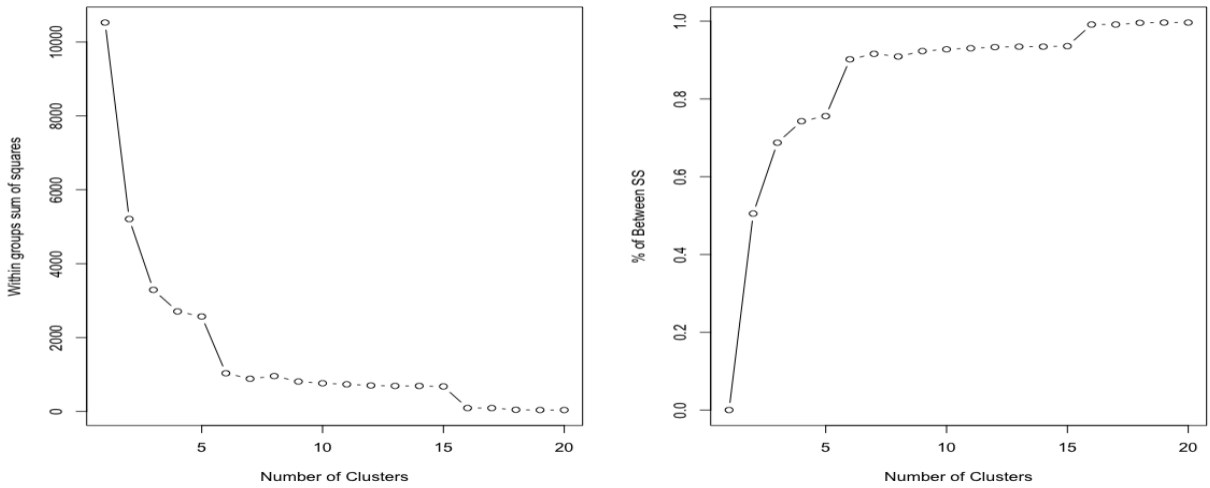

FIG 18: WSS plot and % BSS plot with K-Means Clustering method on the PCA scores data (k=1 to k=20)

Optimal number of clusters with Hierarchical clustering method using the original data

FIG 19 shows the WSS plot as a function of the number of clusters and the percent BSS as a function of the number of clusters from k=1 to k=20. In this case, the Hierarchical clustering method is used on the original data. The WSS plot to the left shows a distinct drop in the within-groups sum of squares when moving from five to six clusters. After seven or eight clusters, the reduction in the within-groups sum of squares is not much. With k=7 or k=8, the % of Between SS is more than 0.85 or so**. Therefore, we conclude that k=7 gives the optimal number of clusters for the Hierarchical clustering solution on the original data.**

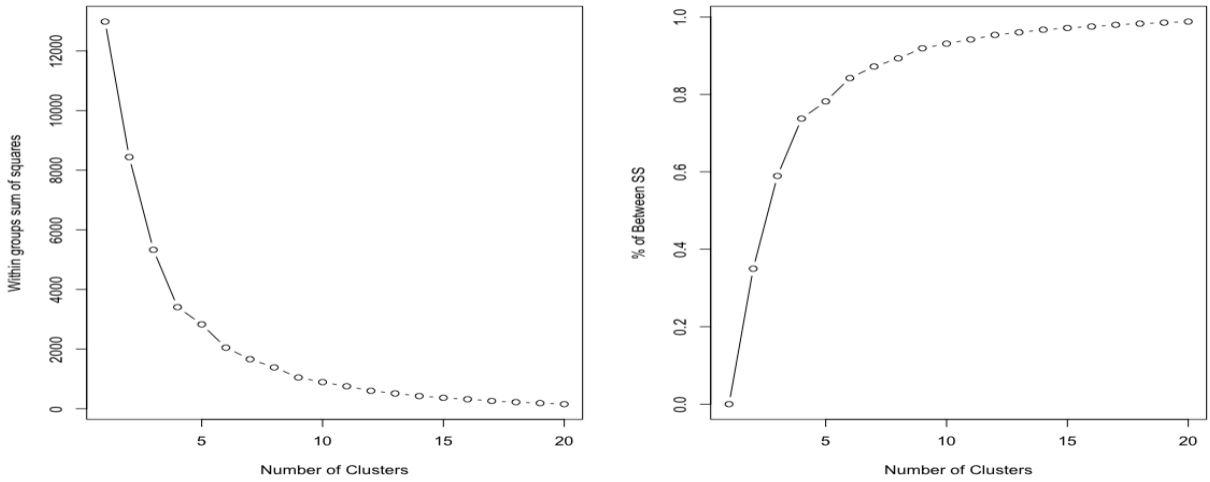

FIG 19: WSS plot and % BSS plot with Hierarchical Clustering method on the original data (k=1 to k=20)

Optimal number of clusters with Hierarchical clustering method using the PCA scores data

FIG 20 shows the WSS plot as a function of the number of clusters and the percent BSS as a function of the number of clusters from k=1 to k=20. In this case, the Hierarchical clustering method is used on the PCA scores data. The WSS plot to the left shows a clear elbow drop in the within-groups sum of squares when moving from five to six clusters. At k=6, the reduction in the within-groups sum of squares is not much. With k=6, the % of Between SS is more than 0.90. **Therefore, we conclude that k=6 gives the optimal number of clusters for Hierarchical clustering solution on the PCA scores data.**

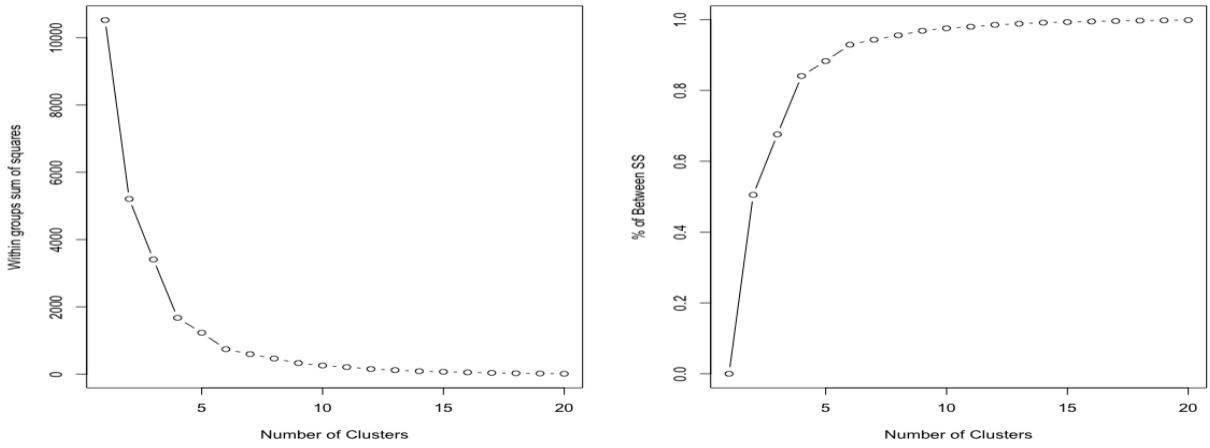

FIG 20: WSS plot and % BSS plot with Hierarchical Clustering method on the PCA scores data (k=1 to k=20)

Section A.8 in the Appendix has the R code for Computing Optimal number of clusters by brute force.

11. On Your Own Modeling 1 – Cluster solution for USSTATES dataset

In this section, we present the results of conducting hierarchical clustering analysis on the USSTATES dataset. The USSTATES dataset has been sourced from the census data and consists of state-wide average or proportion scores for the non-demographic variables. The data set has 50 records. The number of dimensions in the data set is 13. Table 27 lists the variables and their descriptions. The data set has no records with missing values. The data set has two categorical variables – State and Region. The rest of the variables are continuous variables.

No.	Variable	Description
1	State	A record is present for each state in US
2	Region	Four region values – MW, NE, S, W
3	Population	Average or proportion score of population of the state
4	HouseholdIncome	Average or proportion score of household income
5	HighSchool	Average or proportion score of high school attendees
6	College	Average or proportion score of college attendees
7	Smokers	Average or proportion score of smokers
8	PhysicalActivity	Average or proportion score of physical activity of citizens in the state
9	Obese	Average or proportion score of obese citizens
10	NonWhite	Average or proportion score of non-white citizens
11	HeavyDrinkers	Average or proportion score of heavy drinkers in the state
12	TwoParents	Average or proportion score of kids with two parents in the state
13	Insured	Average or proportion score of insured citizens in the state

Table 27: variables in the USSSTATES data set and their description

For EDA, we created a pairwise scatterplot with the USSTATES data. FIG 21 shows the pairwise scatterplot, which shows some patterns or clusters – High School vs. Obese, HighSchool vs. Smokers, etc.

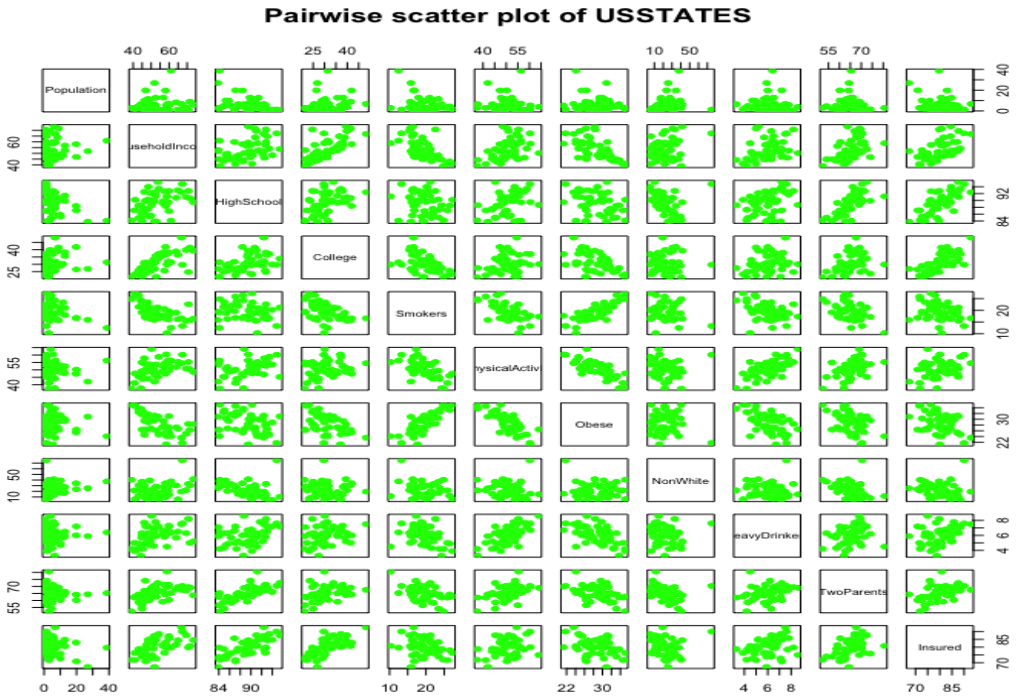

FIG 21: Pairwise plot of USSTATES data set

Prior to attempting the hierarchical clustering, we applied center scaling to standardize the continuous variables to mean 0 and standard deviation 1. The result is stored as a matrix. We then computed the distance matrix using the Euclidean distance ***dist()*** R function. We used complete linkage (distance between groups is defined as that of the most distant pair of individuals). FIG 22 shows the resultant dendrogram for the USSTATES data set with "***complete***" linkage hierarchical clustering method.

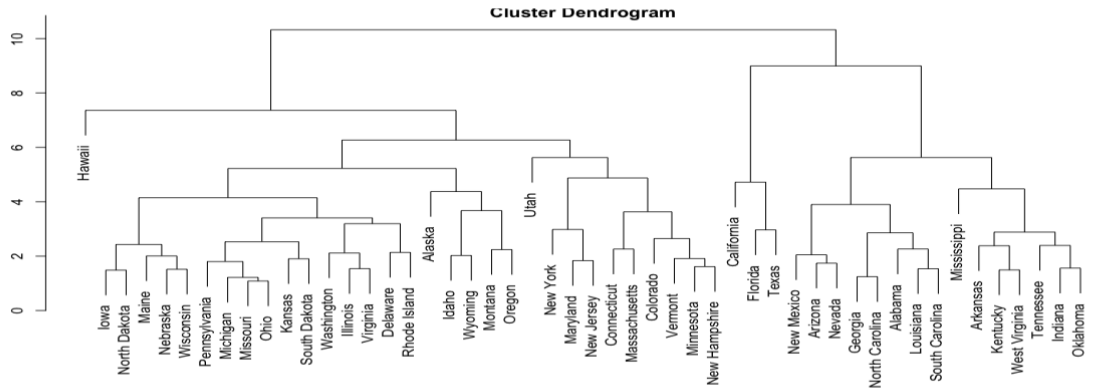

FIG 22: Hierarchical clustering dendrogram – USSTATES Dataset

Using k=3 for cutree()

We used k=3 to cut the above dendrogram into three groups. The cut height is 3.

Table 28 shows the cluster assignment against the Region in the USSTATES data set. Cluster 2 has the most records followed by Cluster 1. Cluster 3 has the fewest with just 3 records.

	1	2	3
MW	2	11	0
NE	0	11	0
S	10	1	2
W	3	9	1

Table 28: cluster assignment to Regions in the USSTATES data based on k=3 cutree

Next, we computed the accuracy for the k=3 cluster solution using the USSTATES data. For this, we calculated With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and percent BSS values. Table 29 shows the computed values for WSS, TSS, BSS, and Percent BSS.

Metric	Values for k=3 (USSTATES data Hierarchical Clustering)
TSS	539
WSS	324.8617
BSS	214.1383
BSS percent	0.3972881

Table 29: classification accuracy for k=3 cutree cluster solution using the USSTATES data

Using k=6 for cutree()

Following that, we used k=6 to cut the above dendrogram into six groups. The cut height is 6. Table 30 shows the cluster assignment against the Region in the USSTATES data set. The resultant groups define the partitions in such a way that the clusters below that height in the dendrogram are distant from each other is at least 6 units. Cluster 2 has the most records followed by Cluster 5. Cluster 6 has the fewest with just 1 record.

	1	2	3	4	5	6
MW	0	10	2	0	1	0
NE	0	4	0	0	7	0
S	5	1	5	2	0	0
W	3	6	0	1	2	1

Table 30: cluster assignment to Regions in the USSTATES data based on k=6 cutree

After that, we computed the accuracy of the k=6 cluster solution using the USSTATES data. For this, we calculated With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and percent BSS values. Table 31 shows the computed values for WSS, TSS, BSS, and Percent BSS.

Metric	Values for k=6 (USSTATES data Hierarchical Clustering)
TSS	539
WSS	220.0264
BSS	318.9736
BSS percent	0.5917878

Table 31: classification accuracy for k=6 cutree cluster solution using the USSTATES data

Using k=9 for cutree()

Next, we used k=9 to cut the above dendrogram into nine groups. The cut height is 9. This is because the percent BSS value, which measures the classification accuracy, of k=6 is only around 0.5.

Table 32 shows cluster assignment against the Region in the USSTATES data set for k=9. In this solution, Cluster 6 has the most records, followed by Cluster 5. Cluster 9 has the fewest with just 1 record.

	1	2	3	4	5	6	7	8	9
MW	0	0	2	0	1	19	0	0	0
NE	0	0	0	0	4	4	0	3	0
S	5	0	5	2	0	1	0	0	0
W	3	5	0	1	1	1	1	0	1

Table 32: cluster assignment to Regions in the USSTATES data based on k=9 cutree

Next we computed the accuracy for the k=9 cluster solution using the USSTATES data. For this, we calculated With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and percent BSS values. Table 33 shows the computed values for WSS, TSS, BSS, and Percent BSS.

Metric	Values for k=9 (USSTATES data Hierarchical Clustering)
TSS	539
WSS	164.4064
BSS	374.5936
BSS percent	0.6949788

Table 33: classification accuracy for k=9 cutree cluster solution using the USSTATES data

Using k=14 for cutree()

Lastly, we used k=14 to cut the above dendrogram into fourteen groups. The cut height is 14 in ***cutree()*** function. This is because the percent BSS value, which measures the classification accuracy, of k=9 is only ~0.69. Percent BSS provides the total variance explained by the clustering solution.

Table 34 shows the cluster assignment against the Region in the USSTATES data set for k=14. The resultant groups are the partitions in the dendrogram, whose distance from each other is at least 7 units. From the table, we can note that Cluster 7 has the most records, followed by Cluster 6. Clusters 2, 5, 9, 13, and 14 have only one record.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
MW	0	0	0	2	0	1	6	0	0	0	4	0	0	0
NE	0	0	0	0	0	4	3	0	0	0	1	3	0	0
S	5	0	0	4	0	0	1	2	0	0	0	0	1	0
W	0	1	3	0	1	1	1	0	1	4	0	0	0	1

Table 34: cluster assignment to Regions in the USSTATES data based on k=14 cutree

Next, we obtained the accuracy metrics for the k=14 cluster solution using the USSTATES data. We computed With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and percent BSS values. Table 35 shows the computed values for WSS, TSS, BSS, and Percent BSS.

Metric	Values for k=9 (USSTATES data Hierarchical Clustering)
TSS	539
WSS	104.5163
BSS	434.4837
BSS percent	0.8060922

Table 35: classification accuracy for k=14 cutree cluster solution using the USSTATES data

So, we conclude to have to use k=14 to obtain a decent classification accuracy if we were to use hierarchical clustering on the original USSTATES data.

Optimal number of clusters using hierarchical clustering on USSTATES data

Next, we used the WSS plot and percent of BSS plot (as a function of the number of clusters) to determine the optimal number of clusters. WSS (Within-cluster sum of squares) measures the compactness of the clustering. To determine the optimal number of clusters, we want to reduce or minimize the total intra-cluster variation, which is measured by WSS. We utilized the "Elbow" method for this by analyzing the WSS plots (which plots total WSS as a function of the number of clusters). We also plotted the percent of BSS as a function of the number of clusters. Percent BSS explains the total variance explained by the clustering solution. Our criterion is to choose the number of clusters in such a way that adding another cluster does not reduce much the total WSS or improve the percent BSS.

FIG 23 shows the WSS plot as a function of the number of clusters and the percent BSS as a function of the number of clusters from k=1 to k=20. These are obtained with the hierarchical clustering method using the USSTATES data. The WSS plot to the left shows a drop in the decrease of the within-groups sum of squares when moving from six to seven clusters. Stated otherwise, after six clusters, the reduction in the within-groups sum of squares is minuscule. **But, with k=6, the % of Between SS is only close to 0.6. However, if we were aiming for a percent BSS value of more than 0.8, we would then leverage the plot to the right and determine we need k=14. Since percent BSS gives the classification accuracy, we chose k=14 as the optimal solution for hierarchical clustering on the original USSTATES data.**

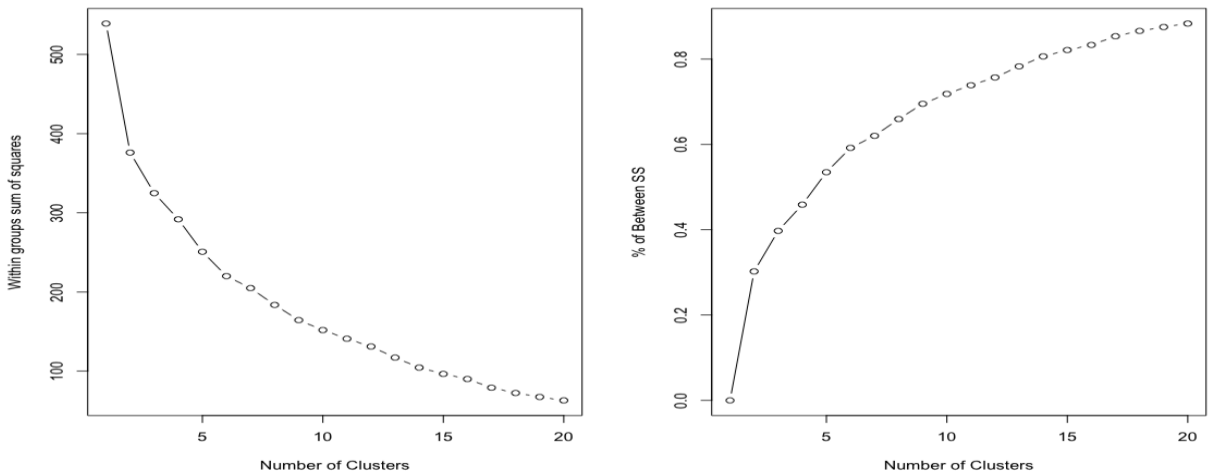

FIG 23: WSS plot and % BSS plot with Hierarchical Clustering method on the USSTATES data (k=1 to k=20)

Using PCA components for hierarchical cluster solution for USSTATES data set

Based on the above plots, we concluded we would need at least 14 clusters to be able to get a better classification accuracy if we were to use the original data. Therefore, we next explored using PCA components for the USSTATES clustering. We used ***princomp()*** R function with cor=TRUE for the correlation matrix so that data is standardized. FIG 24 shows the principal component scores obtained using the USSTATES data set.

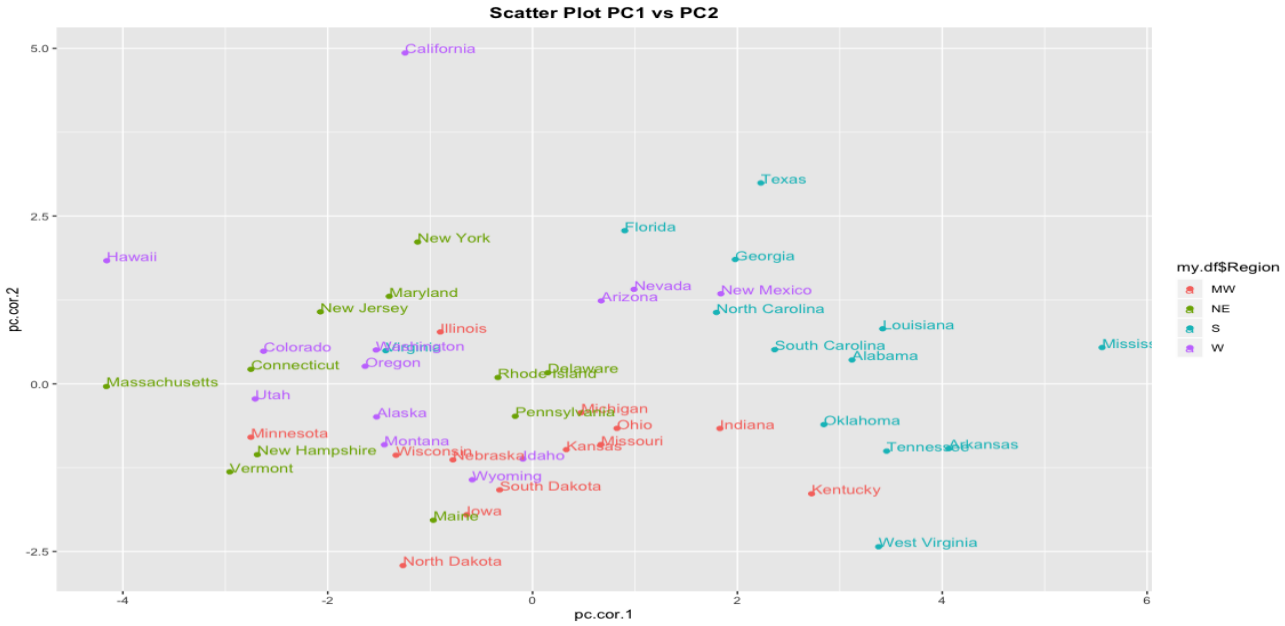

FIG 24: scatter plot of the first and second principal components

FIG 25 shows the dendrogram obtained using hierarchical clustering on the principal component data. The clusters shown in FIG 24 are very different from ones obtained using the USSTATES original data set.

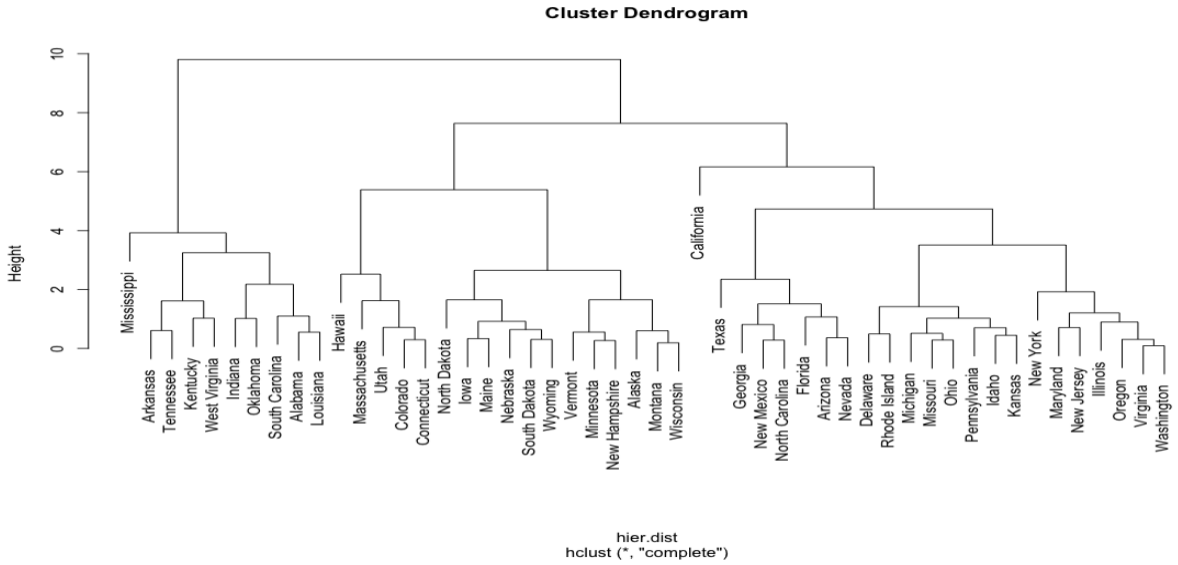

FIG 25: Hierarchical clustering dendrogram – using PC1 and PC2 obtained with USSTATES Dataset

Using k=3 for cutree()

We used k=3 to cut the above dendrogram obtained using PCA data into three groups. The cut height is 3.

Table 36 shows the cluster assignment against the Region in the USSTATES data set. Clusters in the dendrogram below 3 units are at least 3 units apart. Cluster 3 has the most records, followed by Cluster 2. Cluster 1 has the fewest with 10 records.

	1	2	3
MW	2	6	5
NE	0	5	6
S	8	0	5
W	0	6	7

Table 36: cluster assignment to Regions in the USSTATES PCA data based on k=3 cutree

Next, we computed the accuracy for the k=3 cluster solution using the USSTATES data. For this, we calculated With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and percent BSS values. Table 37 shows the computed values for WSS, TSS, BSS, and Percent BSS.

Metric	Values for k=3 (USSTATES PCA data Hierarchical Clustering)
TSS	347.022
WSS	142.3627
BSS	204.6593
BSS percent	0.5897589

Table 37 classification accuracy for k=3 cutree cluster solution using the USSTATES PCA data

Using k=6 for cutree()

We used k=6 to cut the above dendrogram into six groups. The cut height is 6.
Table 38 shows the cluster assignment against the Region in the USSTATES data set. The resultant groups define the partitions in such a way that the clusters below that height in the dendrogram are distant from each other by at least 6 units. Cluster 6 has the most records, followed by Cluster 2. Cluster 4 has the fewest with just 1 record.

	1	2	3	4	5	6
MW	2	6	0	0	0	5
NE	0	3	0	0	2	6
S	8	0	4	0	0	1
W	0	3	3	1	3	3

Table 38: cluster assignment to Regions in the USSTATES PCA data based on k=6 cutree

Next, we computed the accuracy for the k=6 cluster solution using the USSTATES data. For this, we calculated With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and percent BSS values. Table 39 shows the computed values for WSS, TSS, BSS, and Percent BSS.

Metric	Values for k=6 (USSTATES PCA data Hierarchical Clustering)
TSS	347.022
=WSS	67.47677
BSS	279.5452
BSS percent	0.8055547

Table 39: classification accuracy for k=6 cutree cluster solution using the USSTATES PCA data

Using k=8 for cutree()

We used k=8 to cut the above dendrogram into six groups. The cut height is 8.
Table 40 shows cluster assignment against the Region in the USSTATES data set. The resultant groups define the partitions where the clusters below the height 8 in the dendrogram are distant from each other by at least 8 units. Cluster 2 has the most records followed by Cluster 1. Cluster 8 has the fewest with just 1 record.

	1	2	3	4	5	6	7	8
MW	2	6	0	0	0	4	1	0
NE	0	3	0	0	2	3	3	0
S	7	0	4	0	0	0	1	1
W	0	3	3	1	3	1	2	0

Table 40: cluster assignment to Regions in the USSTATES PCA data based on k=8 cutree

Next, we computed the accuracy for the k=8 cluster solution using the USSTATES data. For this, we calculated With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and percent BSS values. Table 41 shows the computed values for WSS, TSS, BSS, and Percent BSS.

Metric	Values for k=8 (USSTATES PCA data Hierarchical Clustering)
TSS	347.022
WSS	41.92513
BSS	305.0969
BSS percent	**0.8791859**

Table 41: classification accuracy for k=8 cutree cluster solution using the USSTATES PCA data

Optimal number of clusters using hierarchical clustering on USSTATES PCA data

We next used the "Elbow" method with the WSS plot to determine the optimal number of clusters. The percent BSS plot as a function of the number of clusters is also plotted to evaluate the classification accuracy. FIG 26 shows the WSS plot as a function of the number of clusters and the percent BSS as a function of the number of clusters from k=1 to k=20. These are obtained with the hierarchical clustering method using the USSTATES PCA data. The WSS plot to the left shows a drop in the decrease of the within-groups sum of squares when moving from eight to nine clusters. Stated otherwise, after eight clusters, the reduction in the within-groups sum of squares is quite small. With k=8, the % of Between SS is close to 0.9. We conclude k=8 as the optimal number of clusters for the hierarchical clustering on the PCA scores data obtained with USSTATES data.

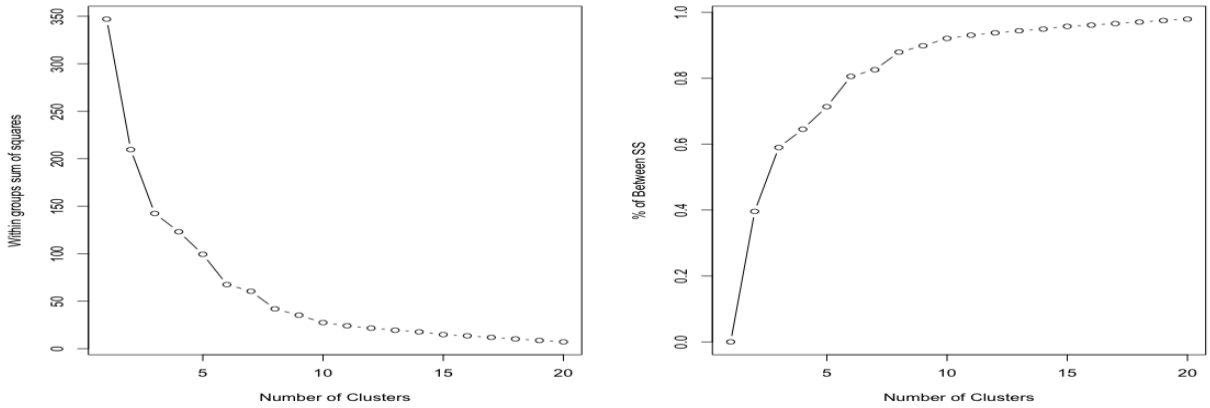

FIG 26: Hierarchical clustering dendrogram – using PC1 and PC2 obtained with USSTATES PCA Data set

To summarize, we conclude that the PCA components should be used in the clustering solution for USSTATES data because it provides a better solution than using the original USSTATES data. The percent BSS value with the PCA solution with k=8 is close to 0.87 indicating ~87% of the total variance is explained by the clustering solution.

Section A.9. in the Appendix has the R code for on your own modeling 1.

12. On Your Own Modeling 2 – Cluster solution for RECIDIVISM data set

In this section, we present the results of conducting K-Means clustering analysis on the RECIDIVISM dataset. The data set used for this analysis pertains to a random sample of convicts released from prison between July 1, 1977, and June 30, 1978. The primary purpose of the data collection was to obtain the time until they return to prison. The information was collected retrospectively in April 1984 using the records. The maximum length of observation is 81 months. There are 1445 observations in the data set with 18 variables. Table 42 lists the variables in the RECIDIVISM data set.

Variable	Description
black	1 if black
alcohol	1 if alcohol problems
drugs	1 if drug history
supervised	1 if release supervised
married	1 if married when incarcerated
felony	1 if felony sentence
workprg	1 if N.C. prison work program
property	1 if property crime
person	1 if crime against person
nbr_piors	No. of prior convictions
education	Years of schooling
nbr_rules	No. of rules violations in prison
age	in months
time_served	time served, rounded to months
follow_up	length follow period, months
duration	max (time until return, follow)
censored	1 if duration right censored
log_duration	log (duration)

Table 42: Variables in RECIDIVISM data set along with their description

In the data set, there are 10 nominal variables – black, alcohol, drugs, supervised, married, felony, workprg, property, person, and censored. The rest of the variables are continuous or quasi-continuous in nature - namely nbr_piors, education, nbr_rules, age, time_served, follow_up, duration, and log_duration. The duration variable captures the time until the convicts returned to prison. log_duration is obtained by taking the natural logarithm of the duration variable. There are no records in the data set with missing values.

Prior to beginning our analysis, we had used ***scale()*** R function on the dataset to standardize the data with mean 0 and standard deviation 1.

K-Means clustering using the RECIDIVISM data (k=11)

We conducted the K-Means clustering with k=11 on the RECIDIVISM data. We chose k=11 because the RECIDIVISM data set has a considerably large number of records. FIG 27 shows the ***clusplot()*** of the k=11 cluster solution. Due to the size of the data (1445 observations), we plotted the clusters without labels.

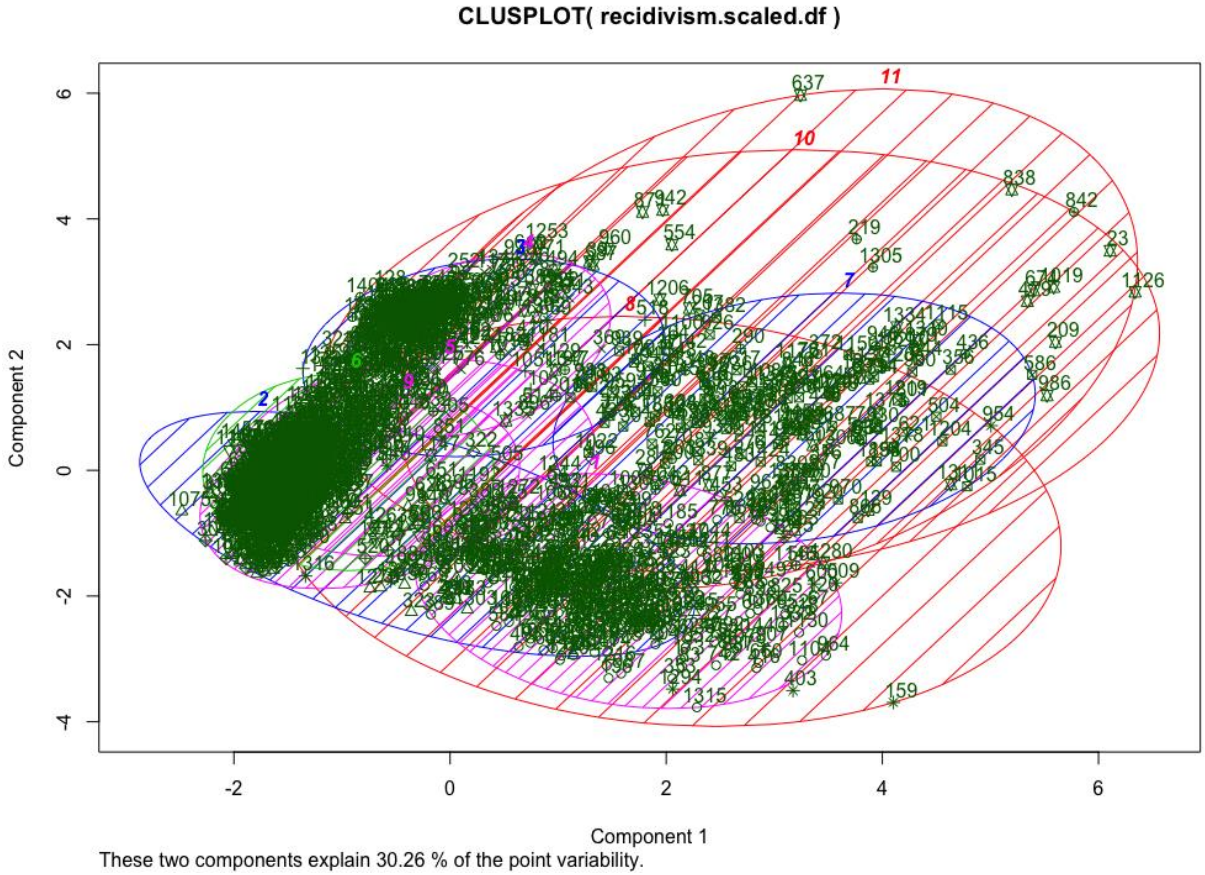

FIG 27: K-Means Clustering Solution (k=11) with scaled RECIDIVISM data

Following that, we computed the accuracy for the k=11 K-Means cluster solution using the original scaled RECIDIVISM data. We computed the metrics - With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and percent BSS. Table 43 shows the computed values for WSS, TSS, BSS, and percent BSS.

Metric	Values for k=6 (original data – K-Means)
TSS	25992
WSS	14534.72
BSS	11457.28
BSS percent	0.4408003

Table 43: classification accuracy for k=11 K-Means cluster solution using the original scaled RECIDIVISM data

K-Means clustering using the RECIDIVISM data (k=20)

We conducted the K-Means clustering with k=20 on the RECIDIVISM data. FIG 28 shows the ***clusplot()*** of the k=20 cluster solution. Here too, due to the size of the data (1445 observations), we plotted the clusters without labels.

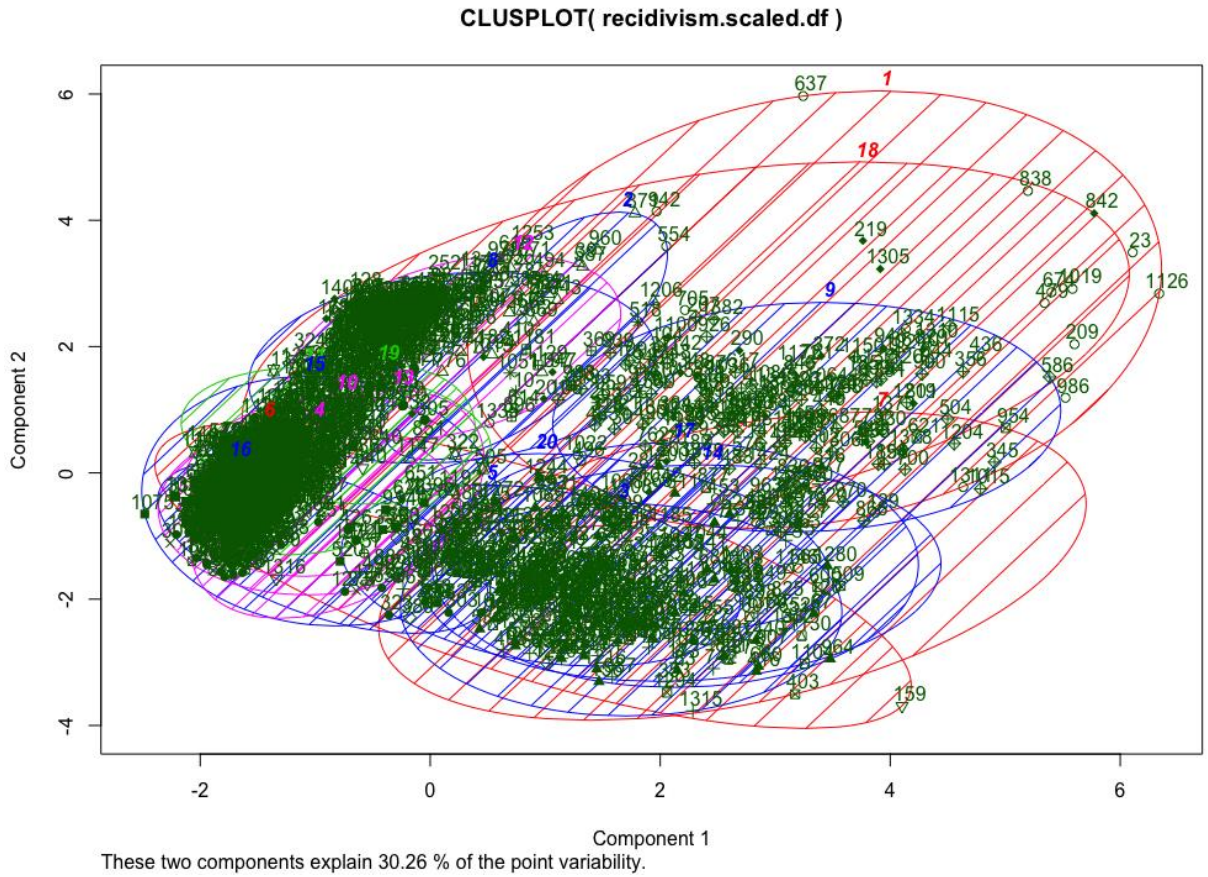

FIG 28: K-Means Clustering Solution (k=20) with scaled RECIDIVISM data

Following that, we computed the accuracy metrics for the k=20 K-Means cluster solution using the original scaled RECIDIVISM data. We computed the metrics - With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and percent BSS. Table 44 shows the computed values for WSS, TSS, BSS, and percent BSS. The clusplot show that the above two solutions only explain 30.26% of the point variability. This differs from the calculated BSS percent values for both k=11 and k=20.

Metric	Values for k=6 (original data – K-Means)
TSS	25992
WSS	12255.28
BSS	13736.72
BSS percent	0.5284981

Table 44: classification accuracy for k=20 K-Means cluster solution using the original scaled RECIDIVISM data

Optimal number of clusters using K-Means clustering on RECIDIVISM data

Since the classification accuracy, even for k=20 is only about 0.53, we needed more clusters to obtain a better solution. So, we used the WSS plot and percent BSS plot to determine the optimal number of clusters. WSS (Within-cluster sum of squares) measures the compactness of the clustering. To determine the optimal number of clusters, we want to reduce or minimize the total intra-cluster variation, which is measured by WSS. Using the "Elbow" method on the WSS plot, which plots total WSS as a function of

the number of clusters, we determine the number of clusters needed. We also plotted the percent of BSS as a function of the number of clusters. Percent BSS explains the total variance explained by the clustering solution. Our criterion is to choose the number of clusters in such a way that adding another cluster does not reduce much the total WSS or improve the percent BSS.

FIG 29 shows the WSS plot as a function of the number of clusters and the percent BSS as a function of the number of clusters from k=1 to k=30. These are obtained with the K-Means clustering method using the scaled RECIDIVISM data. The plot on the left shows a decrease in within-groups sum of squares after eleven clusters. But, with k=11, the percent of Between SS is only close to 0.4. Meaning the cluster solution only explains 40% of the total variance. Even with k=20, we noted that the percent BSS value does not improve much. **This exercise strongly suggests that we should consider PCA for dimension reduction and use that as input to K-Means clustering.**

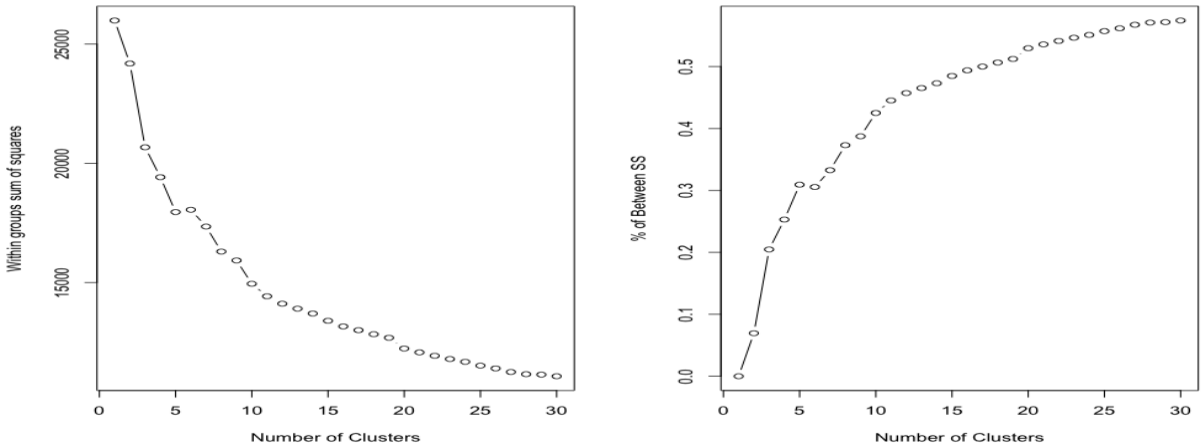

FIG 29: WSS plot and % BSS plot with K-Means Clustering method on the scaled RECIDIVISM data (k=1 to k=30)

K-Means clustering using PCA components obtained with RECIDIVISM data set

Based on the above plots, we concluded we needed to explore PCA as a pre-step to K-Means clustering. Therefore, we next explored using PCA components for the RECIDIVISM clustering solution. We used ***princomp()*** R function with cor=TRUE for the correlation matrix so that data is standardized. FIG 30 shows the principal component scores obtained using the RECIDIVISM data set.

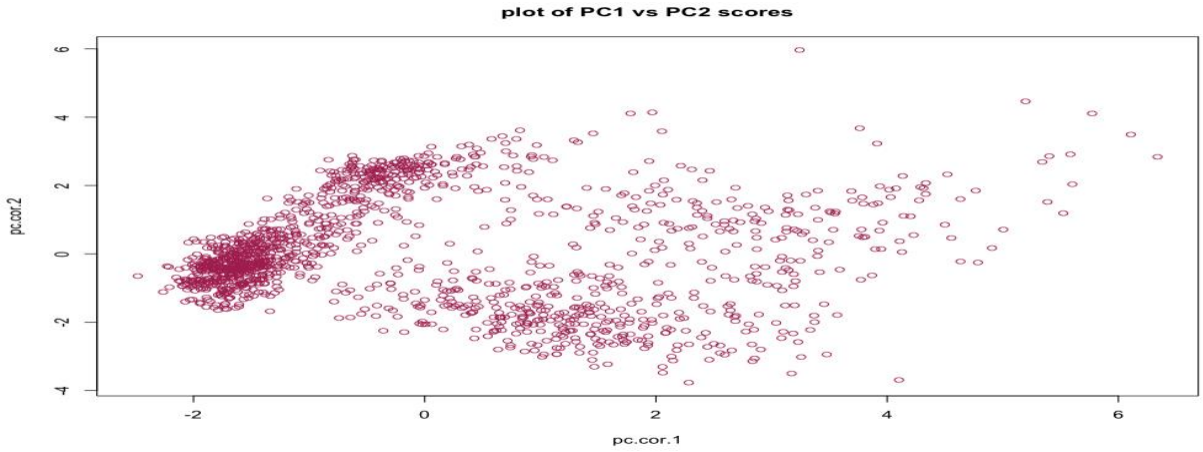

FIG 30: Scatter plot of PC1 vs PC2 obtained using RECIDIVISM data with cor=TRUE

K-Means clustering using the PCA scores RECIDIVISM data (k=5)

We first performed K-Means clustering on the PCA scores data with k=5. FIG 31 shows the plot generated by the **clusplot()** R function for the K-Means solution.

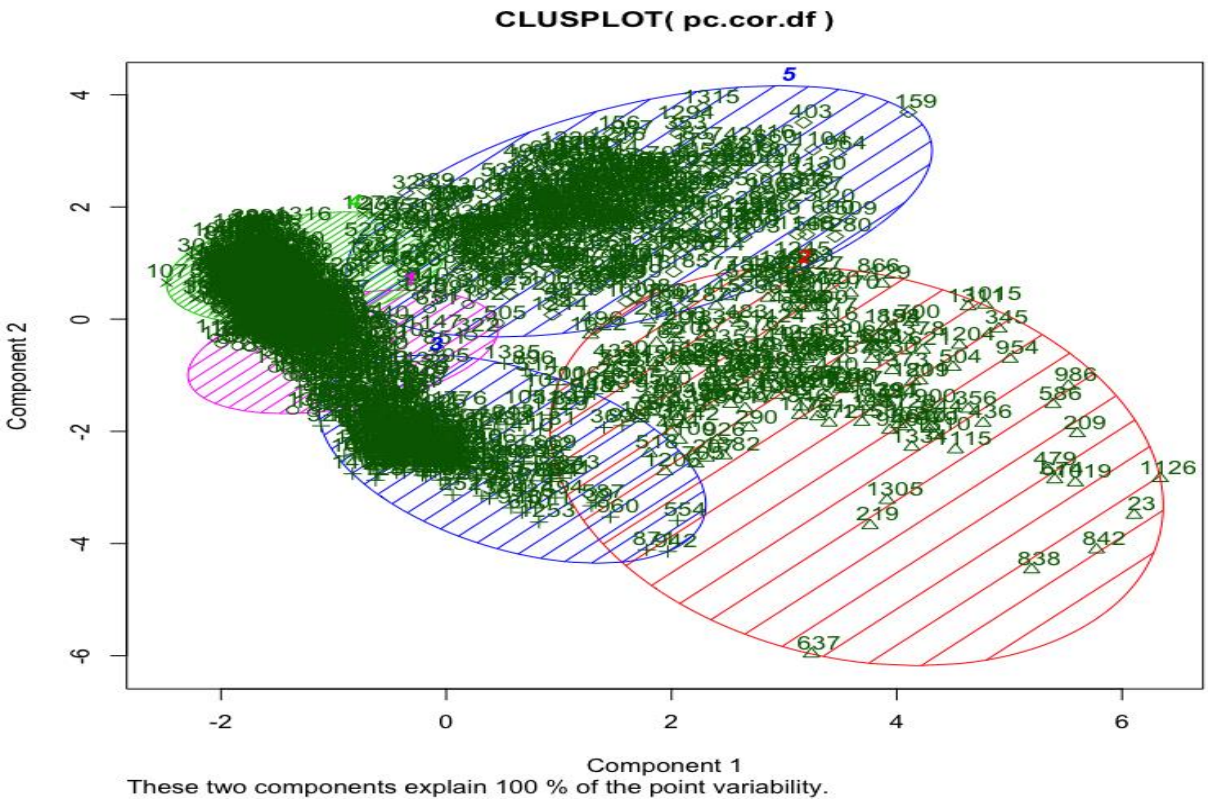

FIG 31: K-Means Clustering Solution (k=5) with PCA scores obtained using the RECIDIVISM data

Next we looked at the classification accuracy metrics for the k=5 K-Means cluster solution on the PCA scores data. For this, we computed the metrics - With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and BSS percent. Table 45 shows the computed values for WSS, TSS, BSS, and BSS percent.

Metric	Values for k=5 (PCA data – K-Means)
TSS	7871.097
WSS	1172.141
BSS	6698.957
BSS percent	**0.851083**

Table 45: classification accuracy for k=5 K-Means cluster solution using the PCA scores of RECIDIVISM data

K-Means clustering using the PCA scores RECIDIVISM data (k=7)

We next performed K-Means clustering on the PCA scores data with k=7. FIG 32 shows the plot generated by the **clusplot()** R function for the K-Means solution.

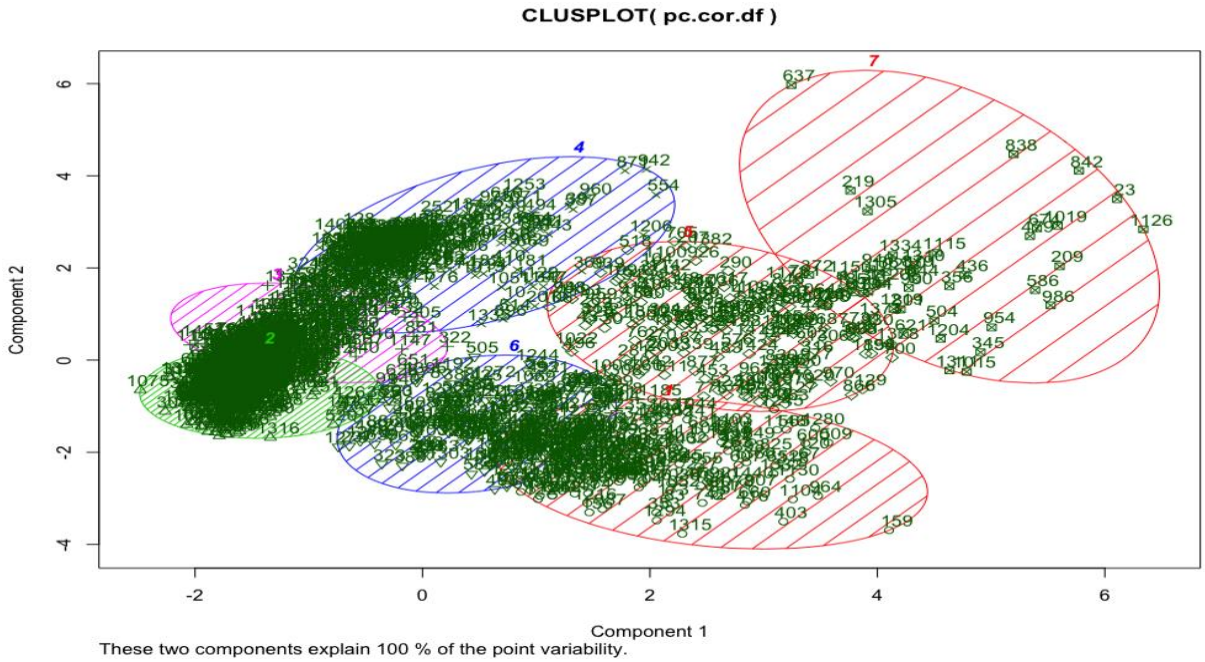

FIG 32: K-Means Clustering Solution (k=7) with PCA scores obtained using the RECIDIVISM data

We computed the accuracy metrics for the k=7 K-Means cluster solution using the PCA scores data. We computed the metrics - With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and BSS percent. Table 46 shows the computed values for WSS, TSS, BSS, and BSS percent.

Metric	Values for k=7 (PCA data – K-Means)
TSS	7871.097
WSS	819.7365
BSS	7051.361
BSS percent	**0.8958549**

Table 46: classification accuracy for k=7 K-Means cluster solution using the PCA scores of RECIDIVISM data

K-Means clustering using the PCA scores RECIDIVISM data (k=9)

Next, we performed K-Means clustering on the PCA scores data obtained using the covariance matrix on the RECIDIVISM dataset. This time we chose k=9. FIG 33 shows the plot generated by the **clusplot()** R function for the K-Means solution.

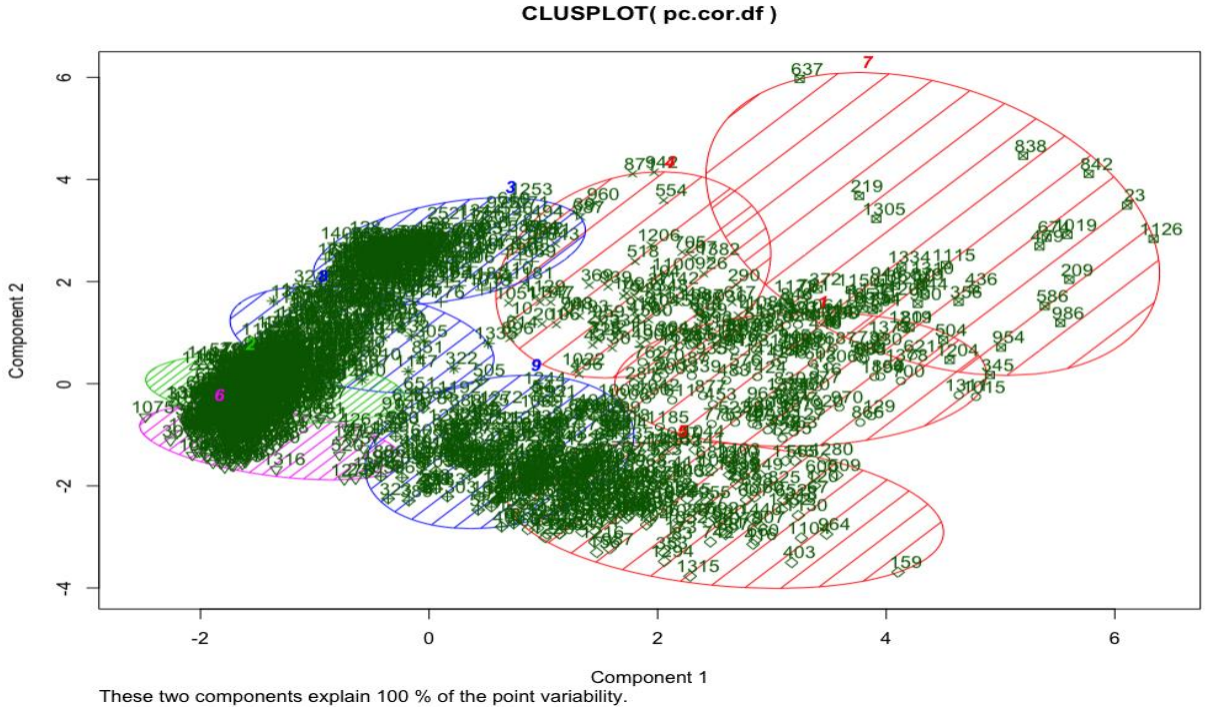

FIG 33: K-Means Clustering Solution (k=9) with PCA scores obtained using the RECIDIVISM data

We computed the accuracy for the k=9 K-Means cluster solution using the PCA scores data. We computed the metrics - With-in-Sum-of-Squares (WSS), Total-Sum-of-Squares (TSS), Between-Sum-of-Squares (BSS), and BSS percent. Table 47 shows the computed values for WSS, TSS, BSS, and BSS percent.

Metric	Values for k=9 (PCA data – K-Means)
TSS	7871.097
WSS	660.4287
BSS	7210.668
BSS percent	**0.9160945**

Table 47: classification accuracy for k=9 K-Means cluster solution using the PCA scores of RECIDIVISM data

Optimal number of clusters using K-Means clustering on PCA scores obtained using RECIDIVISM data

FIG 34 shows the WSS plot as a function of the number of clusters and the percent BSS as a function of the number of clusters from k=1 to k=20. In this case, the K-Means clustering method is used on the PCA data obtained with RECIDIVISM data set. The WSS plot to the left shows a distinct drop in the within-groups sum of squares when moving from five to six clusters. The value of percent BSS for k=5 is ~ 0.85. Therefore, we conclude that k=5 gives the optimal number of clusters.

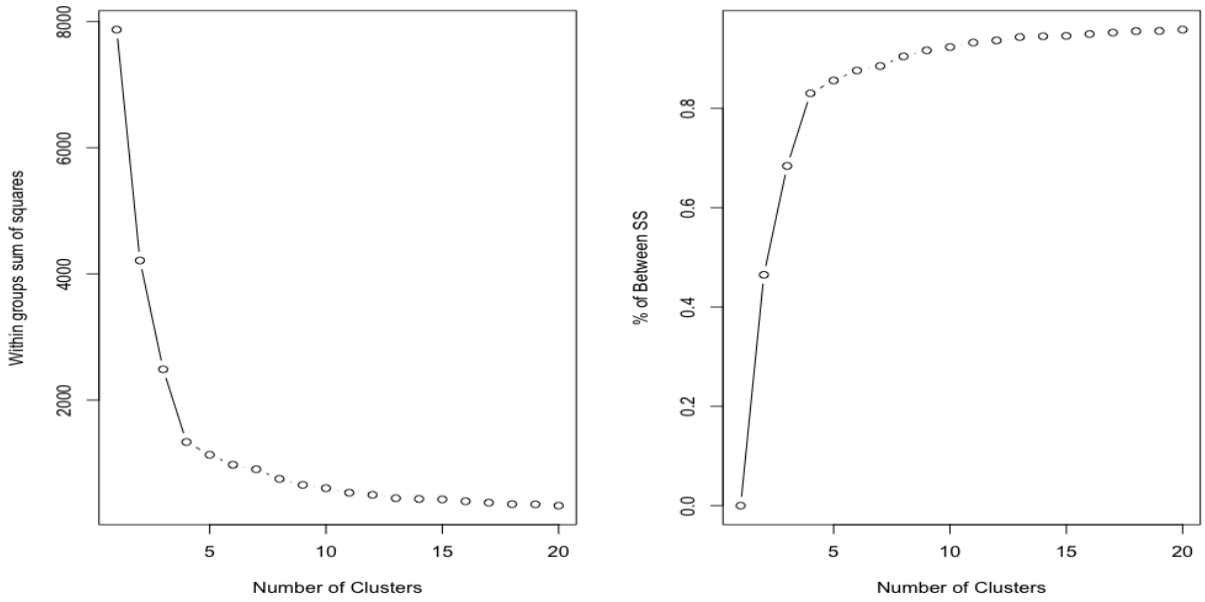

FIG 34: WSS plot and % BSS plot with K-Means Clustering method on the RECIDIVISM data (k=1 to k=20)

To summarize, we conclude that PCA components should be used in the clustering solution for RECIDIVISM data because it provides a better solution than using the original RECIDIVISM data. The percent BSS value with the PCA solution with k=5 is close to 0.85 indicating ~85% of the total variance is explained by the clustering solution.

Section A.10. in the Appendix has the R code for on your own modeling 2.

13. Reflection on cluster modeling experiences

For this assignment on clustering, we looked at three data sets – the European Employment data set, the USSTATES data set, and the RECIVIDISM data set.

With the European Employment data set, upon determining there are no records with missing values, we then conducted the EDA with the pairwise plots. These plots showed some patterns in terms of clusters of points. We then conducted PCA on the data with the covariance matrix and also with the correlational matrix. **Since the data set is of compositional type (the sum of the nine continuous dimensions for each row sum up to 100), we determined that we do not need the standardization done on the data**. **We also learned that otherwise, scaling is an important step in conducting cluster analysis.**

For the clustering, we first started with the Hierarchical clustering method. Using this method, we conducted the analysis first with the original data (only used the continuous variables to compute the Euclidean distance). We used the "completed" linkage method. Upon getting the dendrogram, we used k=3 and k=6 to cut the dendrogram tree. In both the cases, we measured the classification accuracy metrics – TSS, WSS, BSS, and Percent BSS. We learned that a good clustering solution should have a low WSS and a high BSS/Percent BSS. Between the k=3 and k=6 hierarchical cluster solutions on the original

European Employment data, we determined the k=6 solution performs better. Next, we used the scores obtained with PC1 and PC2 components in the Hierarchical clustering method (with "complete" linkage method). Here too, upon getting the dendrogram, we used k=3 and k=6 to cut the dendrogram tree. **When we compared the classification accuracy of all the hierarchical cluster models, among them, we learned that the k=6 hierarchical cluster solution obtained with the PCA scores data performs the best**.

We also explored the K-Means clustering solution for European Employment data. As with Hierarchical clustering, we conducted K-Means clustering with the original data and also with the PCA scores data. We used k=3 and k=6. This resulted in four K-Means cluster models. We learned to plot the K-Means clustering solutions using clusplot to display the clusters along with their centers. **For the European Employment data set, it turned that the k=6 K-Means clustering solution with PCA scores data is the best solution. This conclusion was arrived after comparing all the eight models.** The percent BSS metric for this solution was 0.92, indicating that 92% of the total variance in the data was explained by the cluster solution.

We also learned to use the "Elbow" method on the WSS plot/% BSS plot to select the optimal number of clusters. We have plotted four sets of these plots – with Hierarchical clustering and with K-Means clustering methods applied to the original data and on the PCA data. **In most of the cases, it appeared that k=6 is the optimal number of clusters for the European Employment data.**

Next, for the first "on your own modeling" assignment with USSTATES data set, we followed the same methodology. We first ensured there are no records with missing values. We then conducted a basic EDA analysis. We then extracted the continuous variables in the data and standardized the data to center mean 0 and standard deviation 1. We computed the Euclidean distance on the scaled continuous variables. We next used the Hierarchical clustering method (with "complete" linkage) with several values for k. In all the cases, we computed the classification accuracy. With the scaled USSTATES data, we determined we needed a high value for k (we found out k=14 works well) to obtain a decent classification accuracy. So, we used PCA to reduce the dimensions of the data. We computed the scores based on the first and second principal components. We used them with the Hierarchical clustering method. With the PCA scores data, we only needed k=8 to obtain a good classification accuracy. **We learned that using PCA as a pre-step to clustering helped obtain a better clustering solution for the USSTATES data.**

For the second "on your own modeling" assignment with RECIDIVISM data set, we used the K-Means clustering method. Like with the previous data set, upon ensuring there are no records with missing values, we conducted standardization on the data to center the variables to mean 0 and sd 1. RECIDIVISM data set has a large number of records (1445). With this data set, K-Means needed a high value of k to obtain a good classification accuracy. This was confirmed by plotting the WSS plot and the % BSS plot. Hence, we decided to try the K-Means method with the PCA data. We obtained the PCA scores for the first and second principal components. We tried the K-Means method with k=5,7,9. Using all three values of k, we were able to obtain very good classification accuracy in excess of 0.85. **Therefore, we concluded that k=5 K-Means cluster solution obtained with PCA scores data provided a good solution for the RECIDIVISM data.** This solution explained 85% of the total variance in the data.

Using these three data sets, we learned that clustering is very subjective and that we should always consider standardization and using PCA as pre-steps to clustering. The configurable parameters for Hierarchical clustering and K-Means clustering are different. We learned that we should select the best solution by trying out different values for the parameters.

Appendix

A.1. Data Preparation

```
library(cluster)
require(ggplot2)

setwd("/Users/harini-mac/Desktop/Northwestern University/MSDS-411/Week7/Assignment04/")
# read the European Employment file
my.data <- read.csv("EuropeanEmployment.csv")

# Check the structure and the data set
str(my.data)
head(my.data)
dim(my.data)

# remove any rows with missing values
my.data <-  na.omit(my.data)
dim(my.data)
```

A.2. Basic exploratory data analysis – univariate plots

```
# distribution table for Group
par(mfrow=c(1,1))
barplot(table(my.data$Group), col=c("red","blue","green","purple"), main="Barplot for the Group variable")

par(mfrow=c(4,2))
# EDA for AGR
summary(my.data$AGR)
hist(my.data$AGR,main="Histogram for Agriculture variable",xlab="AGR",border="blue",col="orange",xlim=c(0,100),las=1,breaks=10)
boxplot(my.data$AGR,main = "Box plot of Agriculture variable", xlab = "AGR",ylab = "",col = "orange",border = "brown",horizontal =
TRUE,notch = FALSE)

# EDA for MIN
summary(my.data$MIN)
hist(my.data$MIN,main="Histogram for Mining variable",xlab="MIN",border="blue",col="green",xlim=c(0,100),las=1,breaks=10)
boxplot(my.data$MIN,main = "Box plot of Mining variable", xlab = "MIN",ylab = "",col="green",border = "brown",horizontal =
TRUE,notch = FALSE)

# EDA for MAN
summary(my.data$MAN)
hist(my.data$MAN,main="Histogram for Manufacturing
variable",xlab="MAN",border="blue",col="blue",xlim=c(0,100),las=1,breaks=10)
boxplot(my.data$MIN,main = "Box plot of Manufacturing variable", xlab = "MAN",ylab = "",col="blue",border = "brown",horizontal =
TRUE,notch = FALSE)

# EDA for PS
summary(my.data$PS)
hist(my.data$PS,main="Histogram for Power and water supply
variable",xlab="PS",border="blue",col="aquamarine",xlim=c(0,100),las=1,breaks=10)
boxplot(my.data$PS,main = "Box plot of Power and water supply variable", xlab = "PS",ylab = "",col="aquamarine",border =
"brown",horizontal = TRUE,notch = FALSE)

par(mfrow=c(5,2))
# EDA for CON
summary(my.data$CON)
hist(my.data$CON,main="Histogram for Construction variable",xlab="CON",border="blue",col="purple",xlim=c(0,100),las=1,breaks=10)
boxplot(my.data$CON,main = "Box plot of Construction variable", xlab = "CON",ylab = "",col = "purple",border = "brown",horizontal =
TRUE,notch = FALSE)

# EDA for SER
summary(my.data$SER)
hist(my.data$SER,main="Histogram for Services variable",xlab="SER",border="blue",col="darkgreen",xlim=c(0,100),las=1,breaks=10)
boxplot(my.data$SER,main = "Box plot of Services variable", xlab = "SER",ylab = "",col="darkgreen",border = "brown",horizontal =
TRUE,notch = FALSE)

# EDA for FIN
summary(my.data$FIN)
hist(my.data$FIN,main="Histogram for Finance variable",xlab="FIN",border="blue",col="red",xlim=c(0,100),las=1,breaks=10)
boxplot(my.data$FIN,main = "Box plot of Finance variable", xlab = "FIN",ylab = "",col="red",border = "brown",horizontal = TRUE,notch =
FALSE)

# EDA for SPS
summary(my.data$SPS)
hist(my.data$SPS,main="Histogram for Social and personal services
variable",xlab="SPS",border="violet",col="lightblue",xlim=c(0,100),las=1,breaks=10)
boxplot(my.data$SPS,main = "Box plot of Social and personal services variable", xlab = "SPS",ylab = "",col="lightblue",border =
"brown",horizontal = TRUE,notch = FALSE)

# EDA for TC
summary(my.data$TC)
hist(my.data$TC,main="Histogram for Transport and communications
variable",xlab="TC",border="violet",col="pink",xlim=c(0,100),las=1,breaks=10)
boxplot(my.data$TC,main = "Box plot of Transport and communications variable", xlab = "TC",ylab = "",col="pink",border =
"brown",horizontal = TRUE,notch = FALSE)
```

A.3. Basic exploratory data analysis – Bivariate plots

```
#Pairwise scatterplot
pairs(my.data[,-c(1,2)],col="red")
#pairs(my.data[,-c(1,2)],pch=19, col=as.numeric(my.data$Group)+1)
```

A.4. Visualizing the data using labelled scatterplots

```
# FIN vs SER scatterplot
ggplot(my.data, aes(x=SER, y=FIN, colour = Group, label= Country)) +geom_point() +
  geom_text(aes(label=Country),hjust=0, vjust=0) +
  ggtitle("Scatter Plot Financial vs Services") +
  theme(plot.title=element_text(lineheight=0.8, face="bold", hjust=0.5))

# FIN vs MAN scatterplot
ggplot(my.data, aes(x=MAN, y=FIN, colour = Group, label= Country)) +
  geom_point() + geom_text(aes(label=Country),hjust=0, vjust=0) +
  ggtitle("Scatter Plot Financial vs Manufacturing") +
  theme(plot.title=element_text(lineheight=0.8, face="bold", hjust=0.5))
```

A.5. Principal Component Analysis

```
# Principal component analysis
# PCA with No standardization - compositional data
apply(my.data[,-c(1,2)],MARGIN=1,FUN=sum)
pca.out <- princomp(x=my.data[,-c(1,2)],cor=FALSE);
names(pca.out)
pc.1 <- pca.out$scores[,1];
pc.2 <- pca.out$scores[,2];
str(pc.1)
pcdf = data.frame(pc1=pc.1, pc2=pc.2)
pcdf1 = cbind(pcdf,my.data$Country)
pcdf2 = cbind(pcdf1,my.data$Group)
str(pcdf2)
ggplot(pcdf2, aes(x=pc1, y=pc2, colour = my.data$Group, label=
              my.data$Country)) +
 geom_point() + geom_text(aes(label=my.data$Country),hjust=0, vjust=0) +
 ggtitle("Scatter Plot PC1 vs PC2") +
 theme(plot.title=element_text(lineheight=0.8, face="bold", hjust=0.5))

# PCA with Correlation matrix
pca.cor<- princomp(x=my.data[,-c(1,2)],cor=TRUE);
names(pca.cor)
pc.cor.1 <- pca.cor$scores[,1];
pc.cor.2 <- pca.cor$scores[,2];
str(pc.cor.1)
pc.cor.df= data.frame(pc1=pc.cor.1, pc2=pc.cor.2)
pc.cor.df1 = cbind(pc.cor.df,my.data$Country)
pc.cor.df2 = cbind(pc.cor.df1,my.data$Group)
str(pc.cor.df2)
ggplot(pc.cor.df2, aes(x=pc.cor.1, y=pc.cor.2, colour = my.data$Group, label=
              my.data$Country)) +
 geom_point() + geom_text(aes(label=my.data$Country),hjust=0, vjust=0) +
 ggtitle("Scatter Plot PC1 vs PC2") +
 theme(plot.title=element_text(lineheight=0.8, face="bold", hjust=0.5))
```

A.6. Hierarchical Cluster Analysis

```
#############################################
# Hirerarchical clustering - using raw data##
#############################################
hier.dist = dist(my.data[,-c(1,2)])
require(maptree)
hclustmodel <- hclust(hier.dist, method = 'complete')
par(mfrow=c(1,1))
plot(hclustmodel,labels=my.data$Country)
# choose the number of clusters k = 3
cut.3 <- cutree(hclustmodel, k=3)
head(cut.3)
cut.3
df3 <- cbind(my.data,cut.3)
df3
# cross tab of clusters vs Group
table(df3$Group,df3$cut.3)

# accuracy for k=3 - Between % ss
subdat <- my.data[,-c(1,2)]
TSS <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
TSS
require(fpc)
complete3 <- cutree(hclust(hier.dist),3)
WSS <- cluster.stats(hier.dist,complete3,
                alt.clustering=NULL)$within.cluster.ss
WSS
BetSSPer <- (TSS-WSS)/TSS
BetSSPer

# choose the number of clusters k = 6
cut.6 <- cutree(hclustmodel, k=6)
head(cut.6)
cut.6
df6 <- cbind(my.data,cut.6)
df6
# cross tab of clusters vs Group
table(df6$Group,df6$cut.6)

# accuracy for k=6 - Between % ss
subdat <- my.data[,-c(1,2)]
TSS <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
TSS
complete6 <- cutree(hclust(hier.dist),6)
WSS <- cluster.stats(hier.dist,complete6,
                alt.clustering=NULL)$within.cluster.ss
WSS
BetSSPer <- (TSS-WSS)/TSS
BetSSPer

#####################################################################
# Hirerarchical clustering - using PCA from covariance matrix       #
#####################################################################
hier.dist = dist(pcdf)
require(maptree)
hclustmodel <- hclust(hier.dist, method = 'complete')
par(mfrow=c(1,1))
plot(hclustmodel,labels=my.data$Country)
# choose the number of clusters k = 3
cut.3 <- cutree(hclustmodel, k=3)
head(cut.3)
cut.3
df3 <- cbind(pcdf2,cut.3)
df3
```

```
colnames(df3) <- c("pc1","pc2","Country","Group","cut.3")
# cross tab of clusters vs Group
table(df3$Group,df3$cut.3)

# accuracy for k=3 - Between % ss
subdat <- pcdf
TSS <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
TSS
require(fpc)
complete3 <- cutree(hclust(hier.dist),3)
WSS <- cluster.stats(hier.dist,complete3,
            alt.clustering=NULL)$within.cluster.ss
WSS
BetSSPer <- (TSS-WSS)/TSS
BetSSPer

# choose the number of clusters k = 6
cut.6 <- cutree(hclustmodel, k=6)
head(cut.6)
cut.6
df6 <- cbind(pcdf2,cut.6)
df6
colnames(df6) <- c("pc1","pc2","Country","Group","cut.6")
# cross tab of clusters vs Group
table(df6$Group,df6$cut.6)

# accuracy for k=6 - Between % ss
subdat <- pcdf
TSS <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
TSS
require(fpc)
complete6 <- cutree(hclust(hier.dist),6)
WSS <- cluster.stats(hier.dist,complete6,
            alt.clustering=NULL)$within.cluster.ss
WSS
BetSSPer <- (TSS-WSS)/TSS
BetSSPer
```

A.7. K-Means Cluster Analysis

```
#####################################################################
# k-Means clustering                                                #
#####################################################################
addLabel <- function(x) {
 label <- paste(x['Country'],x['Group'],sep="_")
 return(label)
}
label <- apply(my.data,1,function(x) addLabel(x))

#l1 <- 1:nrow(my.data)
#l2 <- my.data$Group
#label1 <- paste(l1, l2, sep="_")

rownames(my.data) <- label
# kmeans clustering with k=3 clusters
clusterresults.3 <- kmeans(my.data[,-c(1,2)],3)
names(clusterresults.3)
BetSSPer <- clusterresults.3$betweenss/clusterresults.3$totss
BetSSPer
clusterresults.3$totss
clusterresults.3$tot.withinss
clusterresults.3$betweenss

# cluster plots for kmeans (k=3)
clusplot(my.data[,-c(1,2)], clusterresults.3$cluster, color=TRUE,
      shade=TRUE,
      labels=2, lines=1)

# kmeans clustering with k=6 clusters
clusterresults.6 <- kmeans(my.data[,-c(1,2)],6)
names(clusterresults.6)
BetSSPer <- clusterresults.6$betweenss/clusterresults.6$totss
BetSSPer
clusterresults.6$totss
clusterresults.6$tot.withinss
clusterresults.6$betweenss
#plot(clusterresults, data=my.data[,-c(1,2)])

# cluster plots for kmeans
library(cluster)
clusplot(my.data[,-c(1,2)], clusterresults.6$cluster, color=TRUE,
      shade=TRUE,
      labels=2, lines=1)

#####################################################################
#         K-Means clustering with PCA data                          #
#####################################################################
rownames(pcdf) <- label

# kmeans clustering with k=3 clusters
clusterresults.pca.3 <- kmeans(as.matrix(pcdf),3)
names(clusterresults.pca.3)
BetSSPer <- clusterresults.pca.3$betweenss/clusterresults.pca.3$totss
BetSSPer
clusterresults.pca.3$totss
clusterresults.pca.3$tot.withinss
clusterresults.pca.3$betweenss

# cluster plots for kmeans
clusplot(pcdf, clusterresults.pca.3$cluster, color=TRUE,
      shade=TRUE,
      labels=2, lines=1)
```

```
# kmeans clustering with k=6 clusters
clusterresults.pca.6 <- kmeans(as.matrix(pcdf),6)
names(clusterresults.pca.6)
BetSSPer <- clusterresults.pca.6$betweenss/clusterresults.pca.6$totss
BetSSPer
clusterresults.pca.6$totss
clusterresults.pca.6$tot.withinss
clusterresults.pca.6$betweenss
#plot(clusterresults, data=my.data[,-c(1,2)])

# cluster plots for kmeans
#clusplot(pcdf,  clusterresults.pca.6$cluster, color=TRUE,
#        shade=TRUE,
#        labels=2, lines=0)

clusplot(pcdf, clusterresults.pca.6$cluster, color=TRUE,
       shade=TRUE,
       labels=2, lines=1)
```

A.8. Computing the Optimal Number of Clusters by Brute Force

```
###############################################################
# Computing the 'Optimal' Number of Clusters by Brute Force #
###############################################################
# Internal validation
## K means clustering
subdat <- my.data[, -c(1,2)]
wssplot <- function(subdat, nc=20, seed=1234) {
  wss <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
  for (i in 2:nc) {
    set.seed(seed)
    wss[i] <- sum(kmeans(subdat, centers=i)$withinss)}
  rs <- (wss[1] - wss)/wss[1]
  plot(1:nc, wss, type="b", xlab="Number of Clusters",
       ylab="Within groups sum of squares")
  plot(1:nc, rs, type="b", xlab="Number of Clusters",
       ylab="% of Between SS")}
par(mfrow=c(1,2))
wssplot(subdat)

subdat <- pcdf
wssplot <- function(subdat, nc=20, seed=1234) {
  wss <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
  for (i in 2:nc) {
    set.seed(seed)
    wss[i] <- sum(kmeans(subdat, centers=i)$withinss)}
  rs <- (wss[1] - wss)/wss[1]
  plot(1:nc, wss, type="b", xlab="Number of Clusters",
       ylab="Within groups sum of squares")
  plot(1:nc, rs, type="b", xlab="Number of Clusters",
       ylab="% of Between SS")}
par(mfrow=c(1,2))
wssplot(subdat)

## Hierarchical clustering
subdat <- my.data[, -c(1,2)]
wssplot <- function(subdat, nc=20, seed=1234) {
  wss <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
  for (i in 2:nc) {
    require(fpc)
    set.seed(seed)
    hier.dist <- dist(subdat)
    complete3 <- cutree(hclust(hier.dist),i)
    wss[i] <- cluster.stats(hier.dist,complete3,
                  alt.clustering=NULL)$within.cluster.ss}
  rs <- (wss[1] - wss)/wss[1]
  plot(1:nc, wss, type="b", xlab="Number of Clusters",
       ylab="Within groups sum of squares")
  plot(1:nc, rs, type="b", xlab="Number of Clusters",
       ylab="% of Between SS")
  return(wss)}
par(mfrow=c(1,2))
wssplot(subdat)

subdat <- pcdf
wssplot <- function(subdat, nc=20, seed=1234) {
  wss <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
  for (i in 2:nc) {
    require(fpc)
    set.seed(seed)
    hier.dist <- dist(subdat)
    complete3 <- cutree(hclust(hier.dist),i)
    wss[i] <- cluster.stats(hier.dist,complete3,
                  alt.clustering=NULL)$within.cluster.ss}
```

```
  rs <- (wss[1] - wss)/wss[1]
  plot(1:nc, wss, type="b", xlab="Number of Clusters",
      ylab="Within groups sum of squares")
  plot(1:nc, rs, type="b", xlab="Number of Clusters",
      ylab="% of Between SS")
  return(wss)}
par(mfrow=c(1,2))
wssplot(subdat)
```

A.9. On your own Modeling I – USSTATES data set

```
#################################################################
# On Your Own Modeling 1  - CLUSTER SOLUTION USSTATES DS    #
#################################################################
library(readxl)
setwd("/Users/harini-mac/Desktop/Northwestern University/MSDS-411/Week7/Assignment04/")
# read the USSTATES file
my.df <- read_excel("USStates.xlsx")

# Check the structure and the data set
str(my.df)
head(my.df)
dim(my.df)

# remove any rows with missing values
my.df <-  na.omit(my.df)
dim(my.df)

# Scatter plot of USSTATES data
pairs(my.df[,-c(1,2)], main = "Pairwise scatter plot of USSTATES", pch = 19, col="green")

################################################
# Hirerarchical clustering - using raw data##
################################################
# Extract the continuous variables into a matrix
usstates.mat <- as.matrix(my.df[,-c(1,2)])
# Perform center scaling to mean 0 and sd 1
usstates.mat <- scale(usstates.mat, center=TRUE, scale=TRUE)
# Compute the Eucliean distance on the matrix
hier.dist = dist(usstates.mat)
require(maptree)
hclustmodel <- hclust(hier.dist, method = 'complete')
par(mfrow=c(1,1))
plot(hclustmodel,labels=my.df$State)
# choose the number of clusters k = 3
cut.3 <- cutree(hclustmodel, k=3)
head(cut.3)
cut.3
df3 <- cbind(my.df,cut.3)
df3
# cross tab of clusters vs Group
table(df3$Region,df3$cut.3)

# accuracy for k=3 - Between % ss
subdat <- as.data.frame(usstates.mat)
TSS <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
TSS
require(fpc)
complete3 <- cutree(hclust(hier.dist),3)
WSS <- cluster.stats(hier.dist,complete3,
                     alt.clustering=NULL)$within.cluster.ss
WSS
BetSSPer <- (TSS-WSS)/TSS
BetSSPer

# choose the number of clusters k = 6
cut.6 <- cutree(hclustmodel, k=6)
head(cut.6)
cut.6
df6 <- cbind(my.df,cut.6)
df6
# cross tab of clusters vs Group
table(df6$Region,df6$cut.6)
```

```
# accuracy for k=6 - Between % ss
subdat <- as.data.frame(usstates.mat)
TSS <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
TSS
complete6 <- cutree(hclust(hier.dist),6)
WSS <- cluster.stats(hier.dist,complete6,
                alt.clustering=NULL)$within.cluster.ss
WSS
BetSSPer <- (TSS-WSS)/TSS
BetSSPer

# choose the number of clusters k = 6
cut.6 <- cutree(hclustmodel, k=6)
head(cut.6)
cut.6
df6 <- cbind(my.df,cut.6)
df6
# cross tab of clusters vs Group
table(df6$Region,df6$cut.6)

# accuracy for k=6 - Between % ss
subdat <- as.data.frame(usstates.mat)
TSS <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
TSS
complete6 <- cutree(hclust(hier.dist),6)
WSS <- cluster.stats(hier.dist,complete6,
                alt.clustering=NULL)$within.cluster.ss
WSS
BetSSPer <- (TSS-WSS)/TSS
BetSSPer

# choose the number of clusters k = 14
cut.14 <- cutree(hclustmodel, k=14)
head(cut.14)
cut.14
df14 <- cbind(my.df,cut.14)
df14
# cross tab of clusters vs Group
table(df14$Region,df14$cut.14)

# accuracy for k=14 - Between % ss
subdat <- as.data.frame(usstates.mat)
TSS <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
TSS
complete14 <- cutree(hclust(hier.dist),14)
WSS <- cluster.stats(hier.dist,complete14,
                alt.clustering=NULL)$within.cluster.ss
WSS
BetSSPer <- (TSS-WSS)/TSS
BetSSPer

########################################
## Hierarchical clustering
subdat <- scale(as.matrix(my.df[, -c(1,2)]))
wssplot <- function(subdat, nc=20, seed=1234) {
 wss <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
 for (i in 2:nc) {
  require(fpc)
  set.seed(seed)
  hier.dist <- dist(subdat)
  complete3 <- cutree(hclust(hier.dist),i)
  wss[i] <- cluster.stats(hier.dist,complete3,
                alt.clustering=NULL)$within.cluster.ss}
 rs <- (wss[1] - wss)/wss[1]
 plot(1:nc, wss, type="b", xlab="Number of Clusters",
```

```
    ylab="Within groups sum of squares")
 plot(1:nc, rs, type="b", xlab="Number of Clusters",
    ylab="% of Between SS")
 return(wss)}
par(mfrow=c(1,2))
wssplot(subdat)

# PCA with Correlation matrix
pca.cor<- princomp(x=my.df[,-c(1,2)],cor=TRUE);
names(pca.cor)
pc.cor.1 <- pca.cor$scores[,1];
pc.cor.2 <- pca.cor$scores[,2];
str(pc.cor.1)
pc.cor.df= data.frame(pc1=pc.cor.1, pc2=pc.cor.2)
pc.cor.df1 = cbind(pc.cor.df,my.df$State)
pc.cor.df2 = cbind(pc.cor.df1,my.df$Region)
str(pc.cor.df2)
ggplot(pc.cor.df2, aes(x=pc.cor.1, y=pc.cor.2, colour = my.df$Region, label=
                my.df$State)) +
 geom_point() + geom_text(aes(label=my.df$State),hjust=0, vjust=0) +
 ggtitle("Scatter Plot PC1 vs PC2") +
 theme(plot.title=element_text(lineheight=0.8, face="bold", hjust=0.5))

hier.dist = dist(pc.cor.df)
require(maptree)
hclustmodel <- hclust(hier.dist, method = 'complete')
par(mfrow=c(1,1))
plot(hclustmodel,labels=my.df$State)
# choose the number of clusters k = 3
cut.3 <- cutree(hclustmodel, k=3)
head(cut.3)
cut.3
df3 <- cbind(pc.cor.df2,cut.3)
df3
colnames(df3) <- c("pc1","pc2","State","Region","cut.3")
# cross tab of clusters vs Group
table(df3$Region,df3$cut.3)

# accuracy for k=3 - Between % ss
subdat <- pc.cor.df
TSS <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
TSS
require(fpc)
complete3 <- cutree(hclust(hier.dist),3)
WSS <- cluster.stats(hier.dist,complete3,
            alt.clustering=NULL)$within.cluster.ss
WSS
BetSSPer <- (TSS-WSS)/TSS
BetSSPer

# choose the number of clusters k = 6
cut.6 <- cutree(hclustmodel, k=6)
head(cut.6)
cut.6
df6 <- cbind(pc.cor.df2,cut.6)
df6
colnames(df6) <- c("pc1","pc2","State","Region","cut.6")
# cross tab of clusters vs Group
table(df6$Region,df6$cut.6)

# accuracy for k=6 - Between % ss
subdat <- pc.cor.df
TSS <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
TSS
require(fpc)
complete6 <- cutree(hclust(hier.dist),6)
```

```
WSS <- cluster.stats(hier.dist,complete6,
                alt.clustering=NULL)$within.cluster.ss
WSS
BetSSPer <- (TSS-WSS)/TSS
BetSSPer

# choose the number of clusters k = 8
cut.8 <- cutree(hclustmodel, k=8)
head(cut.8)
cut.8
df8 <- cbind(pc.cor.df2,cut.8)
df8
colnames(df8) <- c("pc1","pc2","State","Region","cut.8")
# cross tab of clusters vs Group
table(df8$Region,df8$cut.8)

# accuracy for k=8 - Between % ss
subdat <- pc.cor.df
TSS <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
TSS
require(fpc)
complete8 <- cutree(hclust(hier.dist),8)
WSS <- cluster.stats(hier.dist,complete8,
                alt.clustering=NULL)$within.cluster.ss
WSS
BetSSPer <- (TSS-WSS)/TSS
BetSSPer

## Hierarchical clustering
subdat <- pc.cor.df
wssplot <- function(subdat, nc=20, seed=1234) {
  wss <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
  for (i in 2:nc) {
    require(fpc)
    set.seed(seed)
    hier.dist <- dist(subdat)
    complete3 <- cutree(hclust(hier.dist),i)
    wss[i] <- cluster.stats(hier.dist,complete3,
                      alt.clustering=NULL)$within.cluster.ss}
  rs <- (wss[1] - wss)/wss[1]
  plot(1:nc, wss, type="b", xlab="Number of Clusters",
       ylab="Within groups sum of squares")
  plot(1:nc, rs, type="b", xlab="Number of Clusters",
       ylab="% of Between SS")
  return(wss)}
par(mfrow=c(1,2))
wssplot(subdat)
```

A.10. On your own Modeling II – RECIDIVISM data set

```
#################################################################
# On Your Own Modeling 2  - CLUSTER SOLUTION RECIDIVSIM DS    #
#################################################################
library(readxl)
setwd("/Users/harini-mac/Desktop/Northwestern University/MSDS-411/Week7/Assignment04/")
# read the recidivism file
my.data <- read_excel("recidivism.xlsx")

# Check the structure and the data set
str(my.data)
head(my.data)
dim(my.data)

# remove any rows with missing values
recidivism.df <- na.omit(my.data)
# check the dimensions again
dim(recidivism.df)
names(recidivism.df)

# Apply scale - standardization with mean 0 and sd 1
recidivism.mat <- scale(as.matrix(recidivism.df),center=TRUE,scale=TRUE)
recidivism.scaled.df <- as.data.frame(recidivism.mat)

## K means clustering
subdat <- recidivism.mat
wssplot <- function(subdat, nc=30, seed=1234) {
  wss <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
  for (i in 2:nc) {
    set.seed(seed)
    wss[i] <- sum(kmeans(subdat, centers=i)$withinss)}
  rs <- (wss[1] - wss)/wss[1]
  plot(1:nc, wss, type="b", xlab="Number of Clusters",
       ylab="Within groups sum of squares")
  plot(1:nc, rs, type="b", xlab="Number of Clusters",
       ylab="% of Between SS")}
par(mfrow=c(1,2))
wssplot(subdat)


# kmeans clustering with k=11 clusters
clusterresults <- kmeans(recidivism.mat,11)
names(clusterresults)
BetSSPer <- clusterresults$betweenss/clusterresults$totss
BetSSPer
clusterresults$totss
clusterresults$tot.withinss
clusterresults$betweenss

# cluster plots for kmeans
par(mfrow=c(1,1))
library(cluster)
clusplot(recidivism.scaled.df, clusterresults$cluster, color=TRUE,
         shade=TRUE,
         labels=2, lines=0)

# kmeans clustering with k=20 clusters
clusterresults <- kmeans(recidivism.mat,20)
names(clusterresults)
BetSSPer <- clusterresults$betweenss/clusterresults$totss
BetSSPer
clusterresults$totss
clusterresults$tot.withinss
```

```
clusterresults$betweenss

# cluster plots for kmeans
par(mfrow=c(1,1))
library(cluster)
clusplot(recidivism.scaled.df, clusterresults$cluster, color=TRUE,
       shade=TRUE,
       labels=2, lines=0)

# PCA with Correlation matrix
pca.cor<- princomp(x=recidivism.mat,cor=TRUE);
names(pca.cor)
pc.cor.1 <- pca.cor$scores[,1];
pc.cor.2 <- pca.cor$scores[,2];
str(pc.cor.1)
pc.cor.df= data.frame(pc1=pc.cor.1, pc2=pc.cor.2)
par(mfrow=c(1,1))
plot(pc.cor.1, pc.cor.2,col="maroon",main="plot of PC1 vs PC2 scores")

## K means clustering
subdat <- pc.cor.df
wssplot <- function(subdat, nc=20, seed=1234) {
  wss <- (nrow(subdat)-1)*sum(apply(subdat,2,var))
  for (i in 2:nc) {
    set.seed(seed)
    wss[i] <- sum(kmeans(subdat, centers=i)$withinss)}
  rs <- (wss[1] - wss)/wss[1]
  plot(1:nc, wss, type="b", xlab="Number of Clusters",
      ylab="Within groups sum of squares")
  plot(1:nc, rs, type="b", xlab="Number of Clusters",
      ylab="% of Between SS")}
par(mfrow=c(1,2))
wssplot(subdat)

# kmeans clustering with k=5 clusters on PCA data
clusterresults <- kmeans(pc.cor.df,5)
names(clusterresults)
BetSSPer <- clusterresults$betweenss/clusterresults$totss
BetSSPer
clusterresults$totss
clusterresults$tot.withinss
clusterresults$betweenss

# cluster plots for kmeans for k=5
par(mfrow=c(1,1))
clusplot(pc.cor.df, clusterresults$cluster, color=TRUE,
       shade=TRUE,
       labels=2, lines=0)

# kmeans clustering with k=7 clusters on PCA data
clusterresults <- kmeans(pc.cor.df,7)
names(clusterresults)
BetSSPer <- clusterresults$betweenss/clusterresults$totss
BetSSPer
clusterresults$totss
clusterresults$tot.withinss
clusterresults$betweenss

# cluster plots for kmeans for k=7
par(mfrow=c(1,1))
clusplot(pc.cor.df, clusterresults$cluster, color=TRUE,
       shade=TRUE,
       labels=2, lines=0)

# kmeans clustering with k=9 clusters on PCA data
clusterresults <- kmeans(pc.cor.df,9)
```

```
names(clusterresults)
BetSSPer <- clusterresults$betweenss/clusterresults$totss
BetSSPer
clusterresults$totss
clusterresults$tot.withinss
clusterresults$betweenss

# cluster plots for kmeans for k=9
par(mfrow=c(1,1))
clusplot(pc.cor.df, clusterresults$cluster, color=TRUE,
      shade=TRUE,
      labels=2, lines=0)
```