Sherry You
1005990849
October 12th, 2021

RSM338 Assignment #2 Report

Brief Introduction: High-Risk Countries

According to the given data, we know that the corruption index and the legal index are highly correlated. From this, we can define high-risk countries as countries that have a low value in the legal index and a high value in the corruption index.

Due to the difference in measurements across the indexes, we first needed to normalize all the values before implementing the KMeans algorithm. After normalizing the values, a country with a legal normalized value of -0.878158 would mean that its legal index value is -0.878158 standard deviations below the average value. Therefore, we know that high-risk countries would have a normalized legal index value closer to -1.

After fitting the data and plotting the results, we can see that in the first scatter plot, “Legal VS Peace”, higher-risk countries would be labelled in blue and would be closer to the bottom right corner where the legal values are the lowest, and peace values highest. We can match the abbreviations to the labelled outcome to find that high-risk countries are labelled with the number 2, while the low-risk is 1, and moderate risk is 0.

(b) Change in n_init

After trying several different values of n_init and computing the results, we can see that aside from when n_init = 2, all countries that were initially labelled as high-risk stayed in the high risk cluster as n_init increased from 10 to 100. From n_init = 2, we can see that AR, IR, NI, and ZW changed from high-risk to moderate risk. If we look at the inertia for each iteration, we would find that the inertia for n_init =2 is higher than the rest, while the rest of the values are all equal. This shows that having fewer iterations than the default amount can result in larger inertia, hence a less optimal outcome. We can assume that as you increase the n_init towards the default amount, the inertia will decrease, however, we also know that as we increase beyond the default amount, there will be no changes in inertia or the resulting clusters.

(c) **KMeans with four features**

After running the KMeans algorithm using four features and displaying the resulting labels in a chart next to the original labels, we can see that all of the countries initially labeled as high-risk stayed as high risk countries. However, five countries initially labelled as low risk (HU, JM, JO, KW, and RO), ended up changing into moderate risk when four features were used instead of three. Furthermore, the inertia for when four features are used is higher than when three features are used. This shows that adding more features to the KMeans algorithm may decrease its performance and outcome.

(d) **Agglomerative Clustering**

From the AC package, we can see that the resulting labels vary greatly depending on what kind of linkage you are using.

First, when the linkage is set to ‘ward’, we can see that some of the countries that were originally labelled as high risk were changed to moderate risk. We can also see that some of the countries that were originally low risk were changed into moderate risk. As explained in the Agglomerative Clustering documentation, this exemplifies how ‘ward’ tries to minimize the variance between the clusters that are being merged.

Next, looking at the results when the linkage is set to ‘complete’, we can see that only three of the countries that were originally labelled as high-risk remained as a high-risk country. The rest of the countries were split between moderate and low-risk. From the “linkage = complete GDP growth VS Peace” we can see three very distinct clusters, with the high-risk cluster located far from the other two clusters. The documentation for ‘complete’ notes that this linkage uses the maximum distance between all the observations. Consequently, the resulting clusters are very distinct and sometimes even distanced from one another.

The third linkage was ‘average’. From the scatter plot, “linkage = average Legal VS Peace”, we can see that a majority of the countries that were originally labelled as moderate risk, were relabelled as low-risk. Only a few countries that were originally labelled as high-risk remained as high risk countries, with the rest being changed to either moderate or low-risk. The outcome is similar to the ‘complete’ linkage, as the resulting clusters are very distinct.

Finally, the last linkage, ‘single’, showed very different results than the original labels. From the scatter plot, we can see that the majority of the data points belong to the moderate risk cluster. High-risk countries now only include three countries, NR, IR, and ZW, while low-risk countries only include US and IL. Therefore, we know that aside from the more extreme values, most of the countries that were previously labelled as high or low-risk were relabelled as moderate. By the documentation, this linkage uses the minimum distance between the three clusters, which could explain why a majority of the data points now fall in a single cluster.

(e) **Impact of Outliers**

After adding Venezuela to the dataset, we noticed that all the countries that were previously labelled as high risk were moved into the moderate risk cluster. Due to Venezuela's extreme values, Venezuela ended up being in its own cluster while the rest of the countries were divided into the other two clusters. This is due to the way the KMeans algorithm runs. The goal of the algorithm is to decrease the inertia or total distance from the samples to the nearest cluster center. If Venezuela has very extreme values, then a cluster center may be moved closer to Venezuela's data point, to decrease the distance. Since Venezuela's values are so extreme, the other data points may be clustered into the remaining clusters that are relatively closer, which will help reduce the inertia. Therefore, we can conclude that k-means is very sensitive to outliers. Outliers tend to end up in their own cluster as their values are too extreme in comparison to the other countries, which can result in a decrease in the effectiveness of the clustering algorithm.

Sources:

“Sklearn.cluster.AgglomerativeClustering.” *Scikit-Learn*, https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html.