Student Name: Mohit Patil.

Registered E-mail ID: mohitz4418@gmail.com

Assignment Based Subjective Questions.

Question 1: Assignment Summary

Briefly describe the "Clustering of Countries" assignment that you just completed within 200-300 words. Mention the problem statement and the solution methodology that you followed to arrive at the final list of countries. Explain your main choices briefly (what EDA you performed, which type of Clustering produced a better result and so on)

Answer:

Problem Statement:

HELP International is an international humanitarian NGO that is committed to fighting poverty and providing the people of backward countries with basic amenities and relief during the time of disasters and natural calamities. It runs a lot of operational projects from time to time along with advocacy drives to raise awareness as well as for funding purposes. After the recent funding programmes, they have been able to raise around $ 10 million. Now the CEO of the NGO needs to decide how to use this money strategically and effectively. The significant issues that come while making this decision are mostly related to choosing the countries that are in the direst need of aid.

Solution Methodology:

1. Data Processing:
    - There was no null values or duplicate rows in the dataset.
    - Then there are 3 columns which has percentile values and other columns has actual values so we converted it into actual values for column exports, health and imports.
    - Checked for outliers
    - Then performed bivariate and univariate analysis on dataset to get relation between columns and its effect on its need of aid.
    - Treatment for outliers: In the dataset has some outliers in each column. But we cannot remove it since we had less data. So instead of removing them we cap the outliers for all column at upper capping except the ‘child_mort’ column. Because we needed that outliers that to find the country which has high child mortality rate.
2. Feature Scaling:
    - The data was standardized by using Min – Max Scaler method.
3. Hopkins Statistics:
    - The Hopkins statistics method was used to know that is our data has tendency to form meaningful clusters.
4. K-means Clustering:
    - For choosing the K value we used Elbow Curve and Silhouette Score method. From both method we got the optimal value of K = 3.
    - By using K-means algorithm we had 3 clusters.
      Cluster_0 – Developing Countries.
      Cluster_1 – Undeveloped Countries.
      Cluster_2 – Developed Countries.

- This clusters are named based mainly on their gdpp, child_mort and income columns.

5. Hierarchical Clustering:
    - We used Single Linkage and Complete Linkage method for this clustering.
    - By using single linkage, the result is not clear but complete linkage provide good result by showing dendrograms.
    - By using complete linkage, we got the optimal cluster number 3 by using cut_tree.
    - By using K-means algorithm we had 3 clusters.

      Cluster_0 – Undeveloped Countries.
      Cluster_1 – Developed Countries.
      Cluster_2 – Developing Countries.
    - This clusters are named based mainly on their gdpp, child_mort and income columns.
6. By using both K-means and Hierarchical clustering method we get same countries which needed aid.

Question 2: Clustering

a) Compare and contrast K-means Clustering and Hierarchical Clustering?

Answer:

K-means Clustering:

k-means, using a pre-specified number of clusters, the method assigns records to each cluster to find the mutually exclusive cluster of spherical shape based on distance.

K Means clustering needed advance knowledge of K i.e. no. of clusters one want to divide your data.

K-means algorithm can use median or mean as a cluster centre to represent each cluster.

K-means clustering used are normally less computationally intensive and are suited with very large datasets.

In K Means clustering, since one start with random choice of clusters, the results produced by running the algorithm many times may differ.

K- means clustering a simply a division of the set of data objects into non- overlapping subsets (clusters) such that each data object is in exactly one subset.

K Means is found to work well when the shape of the clusters is hyper spherical (like circle in 2D, sphere in 3D).

In K-means clustering convergence is guaranteed.

Hierarchical Clustering:

Hierarchical methods can be either divisive or agglomerative.

In hierarchical clustering one can stop at any number of clusters, one finds appropriate by interpreting the dendrogram.

Agglomerative methods begin with ‘n’ clusters and sequentially combine similar clusters until only one cluster is obtained.

Divisive methods work in the opposite direction, beginning with one cluster that includes all the records and Hierarchical methods are especially useful when the target is to arrange the clusters into a natural hierarchy.

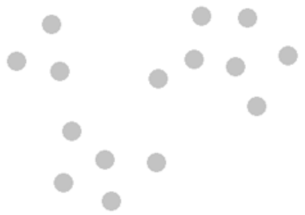
In Hierarchical Clustering, results are reproducible in Hierarchical clustering

A hierarchical clustering is a set of nested clusters that are arranged as a tree.

b) Briefly explain the steps of the K-means clustering algorithm?

Answer:

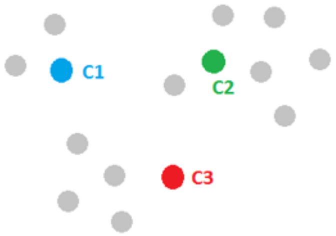
Among all the unsupervised learning algorithms, clustering via k-means might be one of the simplest and most widely used algorithms. Briefly speaking, k-means clustering aims to find the set of k clusters such that every data point is assigned to the closest centre, and the sum of the distances of all such assignments is minimized.



…… Initial data points.

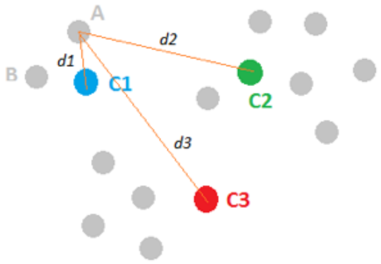
<u>Step 1: Initialize cluster centres - K centroids are created randomly (based on the predefined value of K)</u>

We randomly pick three points C1, C2 and C3, and label them with blue, green and red colour separately to represent the cluster centres.

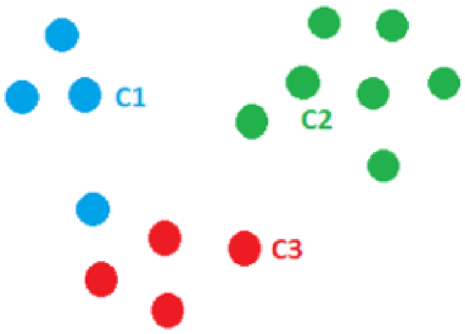


<u>Step 2: Assign observations to the closest cluster center.</u>

K-means allocates every data point in the dataset to the nearest centroid (minimizing Euclidean distances between them), meaning that a data point is considered to be in a particular cluster if it is closer to that cluster's centroid than any other centroid



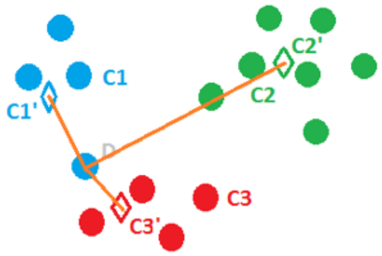
Once we have these cluster centers, we can assign each point to the clusters based on the minimum distance to the cluster center. For the grey point A, compute its distance to C1, C2 and C3, respectively. And after comparing the lengths of *d1*, *d2* and *d3*, we figure out that *d1* is the smallest, therefore, we assign point A to the blue cluster and label it with blue. We then move to point B and follow the same procedure. This process can assign all the points and leads to the following figure.



Step 3: Revise cluster centers as mean of assigned observations.

Then K-means recalculates the centroids by taking the mean of all data points assigned to that centroid's cluster, hence reducing the total intra-cluster variance in relation to the previous step. The "means" in the K-means refers to averaging the data and finding the new centroid

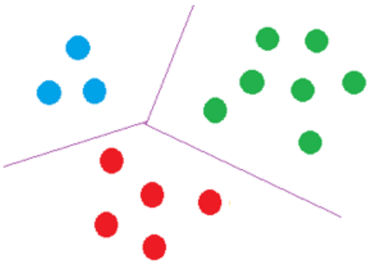
Now we've assigned all the points based on which cluster center they were closest to. Next, we need to update the cluster centers based on the points assigned to them. For instance, we can find the center mass of the blue cluster by summing over all the blue points and dividing by the total number of points, which is four here. And the resulted center mass C1', represented by a blue diamond, is our new center for the blue cluster. Similarly, we can find the new centers C2' and C3' for the green and red clusters.



Step 4: Repeat step 2 and step 3 until convergence.

The algorithm iterates between steps 2 and 3 until some criteria is met (e.g. the sum of distances between the data points and their corresponding centroid is minimized, a maximum number of iterations is reached, no changes in centroids value or no data points change clusters)

The last step of k-means is just to repeat the above two steps. For example, in this case, once C1', C2' and C3' are assigned as the new cluster centers, point D becomes closer to C3' and thus can be assigned to the red cluster. We keep on iterating between assigning points to cluster centers, and updating the cluster centers until convergence. Finally, we may get a solution like the following figure.



c) How is the value of 'k' chosen in K-means clustering? Explain both the statistical as well as the business aspect of it?

Answer:

In K-means algorithm the value of K is identified by using Elbow Curve method, and Silhouette method

Elbow Curve method.

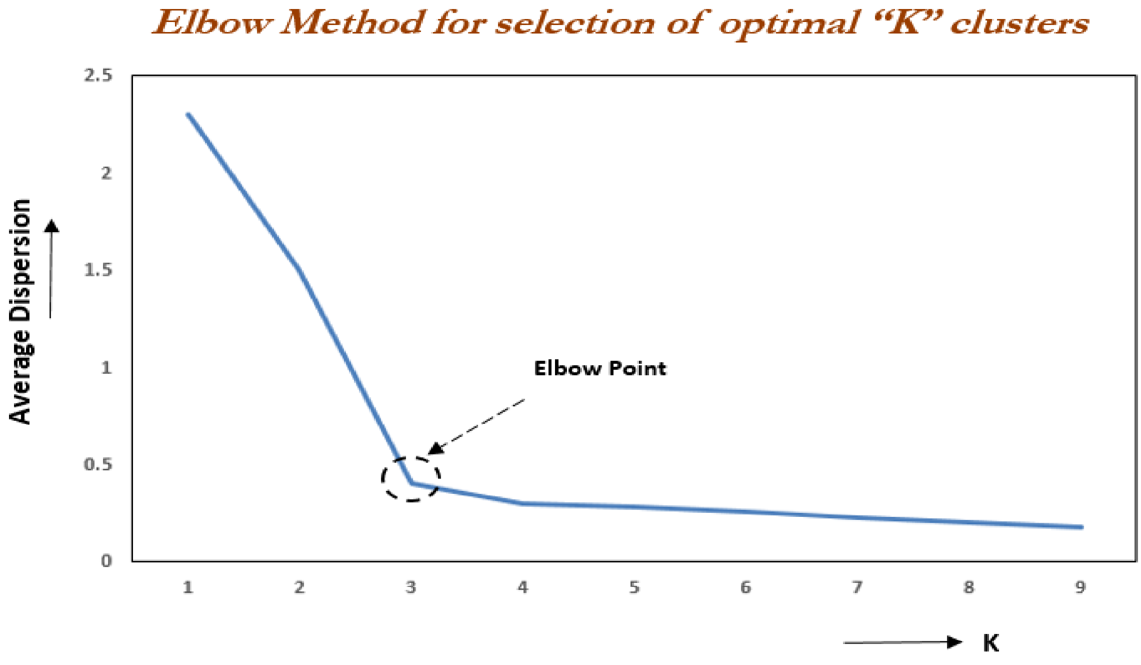
Recall that, the basic idea behind partitioning methods, such as k-means clustering, is to define clusters such that the total intra-cluster variation [or total within-cluster sum of square (WSS)] is minimized. The total WSS measures the compactness of the clustering and we want it to be as small as possible.

The Elbow method looks at the total WSS as a function of the number of clusters: One should choose a number of clusters so that adding another cluster doesn’t improve much better the total WSS.

The elbow method is used to determine the optimal number of clusters in k-means clustering. The elbow method plots the value of the cost function produced by different values of *k*.

As you know, if *k* increases, average distortion will decrease, each cluster will have fewer constituent instances, and the instances will be closer to their respective centroids.

However, the improvements in average distortion will decline as *k* increases. The value of *k* at which improvement in distortion declines the most is called the elbow, at which we should stop dividing the data into further clusters.



Silhouette method.

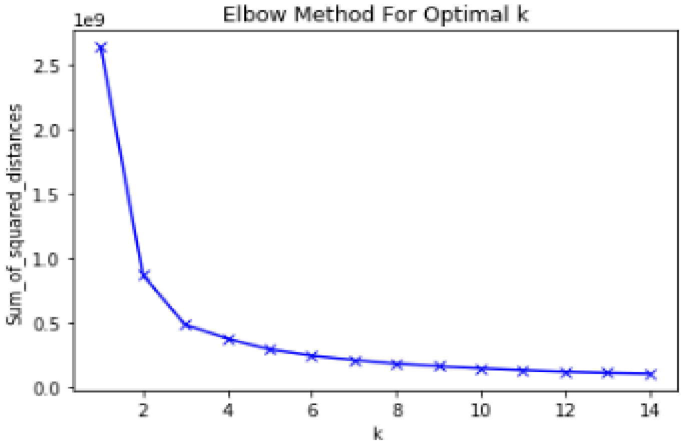
This is a better measure to decide the number of clusters to be formulated from the data. It is calculated for each instance and the formula goes like this:

Silhouette Coefficient = (x - y) / max (x, y)

where, y is the mean intra cluster distance: mean distance to the other instances in the same cluster. x depicts mean nearest cluster distance i.e. mean distance to the instances of the next closest cluster.

The coefficient varies between -1 and 1. A value close to 1 implies that the instance is close to its cluster is a part of the right cluster. Whereas, a value close to -1 means that the value is assigned to the wrong cluster.

As per this method k=3 was a local optimum, whereas k=5 should be chosen for the number of clusters. This method is better as it makes the decision regarding the optimal number of clusters more meaningful and clearer.



But this metric is computation expensive as the coefficient is calculated for every instance. Therefore, decision regarding the optimal metric to be chosen for the number of cluster decision is to be made according to the needs of the product.

d) Explain the necessity for scaling/standardisation before performing Clustering?

Answer:

When we standardize the data prior to performing cluster analysis, the clusters change. We find that with more equal scales, the Percent Native American variable more significantly contributes to defining the clusters. Standardization prevents variables with larger scales from dominating how clusters are defined.

Example:

Let's say that you have two features:

1. weight (in Lbs)
2. height (in Feet)

and we are using these to predict whether a person needs a 'S' or 'L' size shirt.

We are using weight + height for that, and in our trained set let's say we have two people already in clusters:

1. Adam (175Lbs+5.9ft) in 'L'
2. Lucy (115Lbs+5.2ft) in 'S'.

We have a new person - Alan (140Lbs+6.1ft.), and your clustering algo will put it in the cluster which is nearest. So, if we don't scale the features here, the height is not having much effect and Alan will be allotted in 'S' cluster.

In fact, most clustering algorithms are even **highly sensitive to scaling**. Rescaling the data can completely ruin the results.

Bad scaling also appears to be a key reason why people fail with finding meaningful clusters. It is just very easy to do badly.

By no means rely on automatic scaling. It must fit your task and data. Pre-processing is an art, and will require most of the work.

Non-continuous variables are big issue. While you can "hack" data into binary encodings and then pretend they are suitable, the discreteness poses a major issue for the algorithms. For example, many points have the same distance. And the *mean* of such a variable doesn't make a lot of semantic sense anymore. The squared deviation (as used by k-means) is even worse. Results may often be better if you ignore such variables when clustering.

Same goes for bad attributes, such as identifiers, sequence numbers, etc.

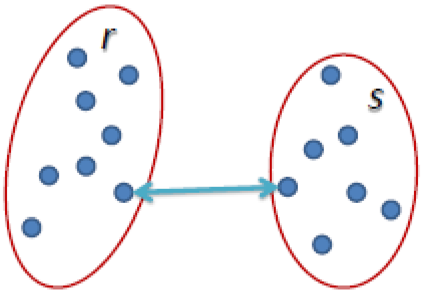
e) Explain the different linkages used in Hierarchical Clustering?

Answer:

There are three methods of linkages in Hierarchical Clustering 1. Single Linkage, 2. Complete Linkage, 3. Average Linkage

Single Linkage:

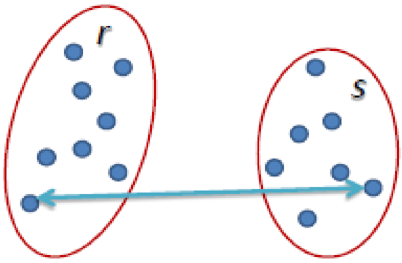
In single linkage hierarchical clustering, the distance between two clusters is defined as the *shortest* distance between two points in each cluster. For example, the distance between clusters “r” and “s” to the left is equal to the length of the arrow between their two closest points.



$$L(r,s) = \min(D(x_{ri}, x_{sj}))$$

Complete Linkage:

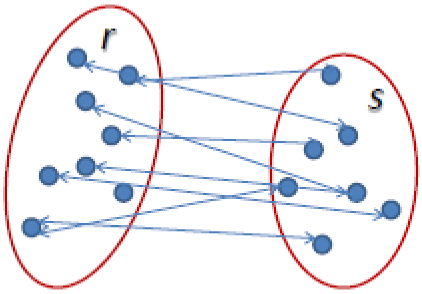
In complete linkage hierarchical clustering, the distance between two clusters is defined as the *longest* distance between two points in each cluster. For example, the distance between clusters “r” and “s” to the left is equal to the length of the arrow between their two furthest points.



$$L(r,s) = \max(D(x_{ri}, x_{sj}))$$

Average Linkage:

In average linkage hierarchical clustering, the distance between two clusters is defined as the average distance between each point in one cluster to every point in the other cluster. For example, the distance between clusters “r” and “s” to the left is equal to the average length each arrow between connecting the points of one cluster to the other.



$$L(r,s) = \frac{1}{n_r n_s} \sum_{i=1}^{n_r} \sum_{j=1}^{n_s} D(x_{ri}, x_{sj})$$