Question 1: Assignment Summary

Below I have briefly described the "Clustering of Countries" assignment that I have just completed:

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 programs, 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.

Objective:

- Objective is to segment the countries using some socio-economic and health factors that determine the overall development of the country.
- Then we need to suggest the countries which the CEO needs to focus on the most for considering for NGO Aid.

Method followed:

- **Data Understanding and Processing:**

  - Checked the total number of rows and columns in data frame
  - Checked datatypes of each columns and found that there is no need to perform datatype conversion as all columns were having correct datatype
  - It was found that there were no null values **(NAN)** in the dataset.
  - There were also no duplicate values or redundancy in given countries
  - There were outliers found and they were treated later before clustering
  - Performed soft capping (0.01 to 0.99) for few features which are having very few extreme outliers like (health, income, import, export etc. columns) and kept some outliers as it is for child mortality, life expectancy, inflation as those outliers will be considered for NGO aid.

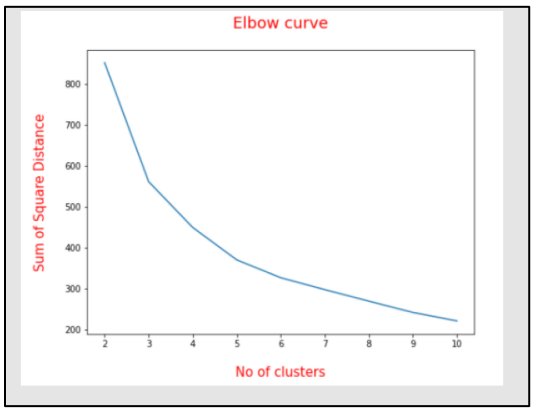
- We have performed scaling /Standardization on all numerical data columns, so that all features will get equal weightage during clustering

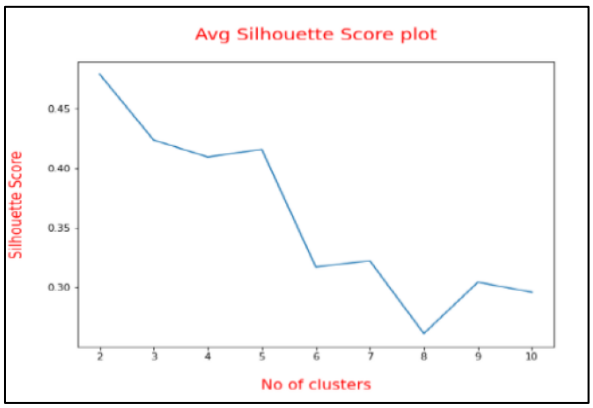
• Data visualization (Exploratory Data Analysis):

- We have performed Univariate analysis like **Distribution plot to check data distribution, Box plot** for outlier analysis
- We have performed Bivariate analysis like **Bar plot between country vs different numerical features** to get some understanding of which country requires more NGO AID

• Clustering (K-Mean & Hierarchical):

- Checked **Hopkin statistics for cluster tendency** and got Hopkins value close to 1 each time. As we know, if Hopkins score closes to 1, there is a good cluster tendency
- Plot **Elbow curve and Silhouette score** and got the optimal value of cluster **K= 3**





- Perform both K-Means clustering and Hierarchical clustering and able to segment country dataset among **under Developed, Developing and Developed country.**
- After cluster profiling I got almost similar top **10 Under Developed country names**, which need NGO AID, from both K-Means and hierarchical clustering.
- I have observed Hierarchical Clustering is more prone to Outliers. Presence of any extreme outliers, which is not capped has effect on cluster formation.
- The ***Result of K-Means is more stable to be considered for final NGO AID*.**

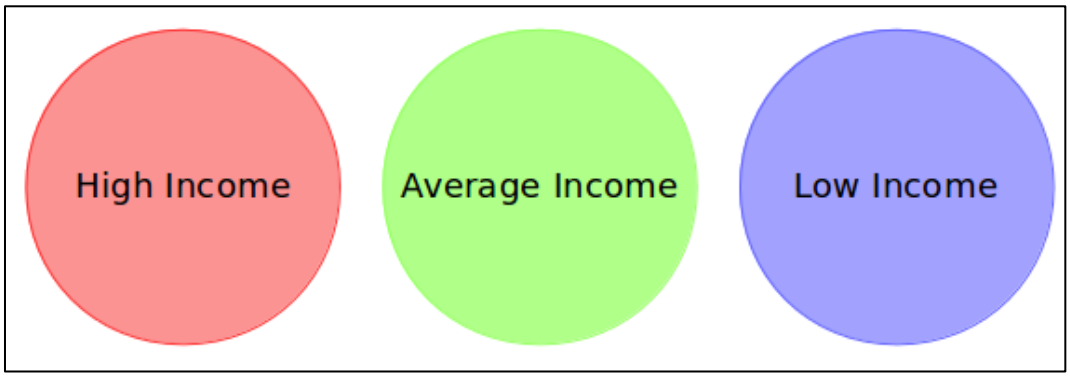
Question 2: Clustering

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

First let us discuss what is clustering?

Clustering: Clustering is the process of dividing the entire data into groups (also known as clusters) based on the patterns in the data.

For example, the bank can group the customers based on their income:



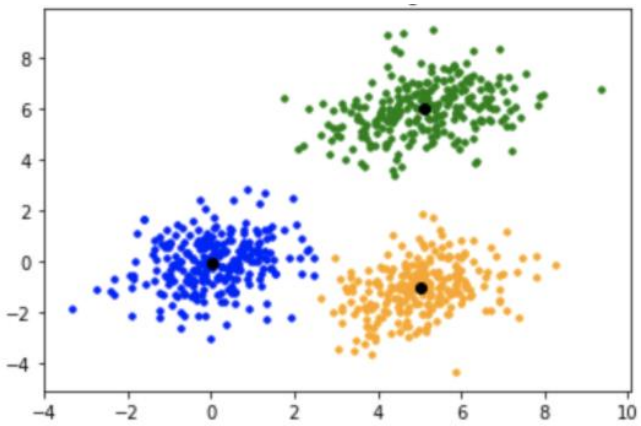
The groups shown above are known as clusters and the process of creating these groups is known as clustering.

Clustering is **Unsupervised machine learning algorithm.** Unsupervised algorithms make inferences from datasets using only input vectors without referring to known, or labelled, outcomes.

There are different types of clustering technique, two of them we will discuss below:

K-Means Clustering: K-means clustering is one of the simplest and popular unsupervised machine learning algorithms. K-means is a centroid-based algorithm, or a distance-based algorithm, where we calculate the distances to assign a point to a cluster. In K-Means, each cluster is associated with a centroid. ***The main objective of the K-Means algorithm is to minimize the sum of square of distances between the points and their respective cluster centroid.***

Clusters formed by K-Mean:



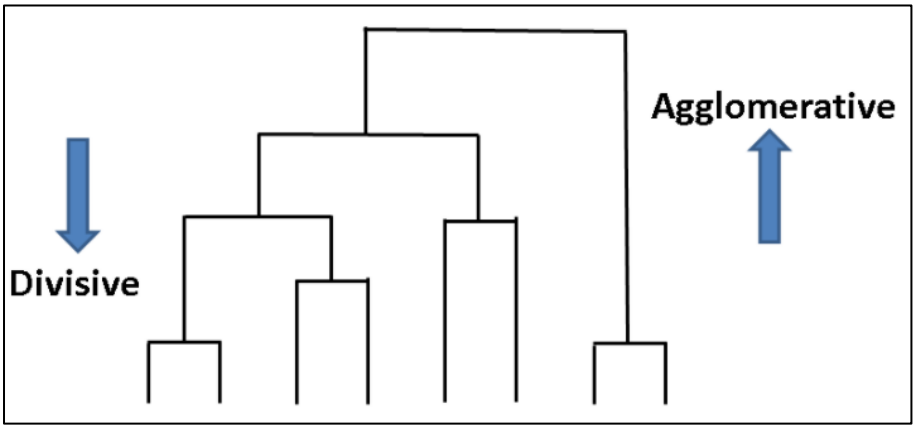
Hierarchical Clustering: The hierarchical clustering Technique is also one of the popular Clustering techniques in Machine Learning ***which involves creating clusters that have predominant ordering from top to bottom.***

This clustering technique is divided into two types:

1) Agglomerative Hierarchical Clustering
2) Divisive Hierarchical Clustering

Agglomerative Hierarchical clustering Technique: In this technique, initially each data point is considered as an individual cluster. At each iteration, the similar clusters merge with other clusters until one cluster or K clusters are formed.

Divisive Hierarchical clustering: All the data points are to be considered as a single cluster and in each iteration, we separate the data points from the cluster which are not similar. Each data point which is separated is considered as an individual cluster. In the end, we'll be left with n clusters.
Dividing the single clusters into n clusters, it is named as ***Divisive Hierarchical clustering.***



Now the **comparison between K-Means and Hierarchical Clustering:**

K-Mean Clustering	Hierarchical Clustering
K Means clustering needed advance knowledge of K i.e. no. of clusters one wants to divide your data.	In hierarchical clustering one can stop at any number of clusters, one finds appropriate by interpreting the dendrogram. Hence advance knowledge of K not required.
One can use median or mean as a cluster center to represent each cluster.	Agglomerative methods begin with 'n' clusters and sequentially combine similar clusters until only one cluster is obtained.
Methods used are normally less computationally intensive and are suited with very large datasets.	This method is normally highly computationally intensive and are suited for with small dataset. Hierarchical methods are especially useful when the target is to arrange the clusters into a natural hierarchy.
In K Means clustering, since one start with random choice of clusters, the results produced by running the algorithm many times may differ.	In Hierarchical Clustering, results are reproducible in Hierarchical clustering. Initial cluster centers are not required in Hierarchical clustering.
SSE is the objective function for K-means.	There exists no global objective function for hierarchical clustering. It considers proximity locally before merging two clusters.

Advantages and Disadvantages of K-Means Algorithm:

Advantages

- Easy to implement
- With many variables, K-Means may be computationally faster than hierarchical clustering (if K is small).
- K-Means may produce Higher clusters than hierarchical clustering
- An instance can change cluster (move to another cluster) when the centroids are recomputed.

Disadvantages

- Difficult to predict the number of clusters (K-Value)
- Initial centroid has a strong impact on the final results
- Sensitive to scale: rescaling your datasets (normalizing or standardizing) will completely change results.
- Very much sensitive to outliers.

Advantages and Disadvantages of Hierarchical Clustering:

Advantages

- Hierarchical clustering outputs a hierarchy, i.e. a structure that is more informative than the unstructured set of flat clusters returned by k-means. Therefore, it is easier to decide on the number of clusters by looking at the dendrogram.

- Easy to implement

Disadvantages

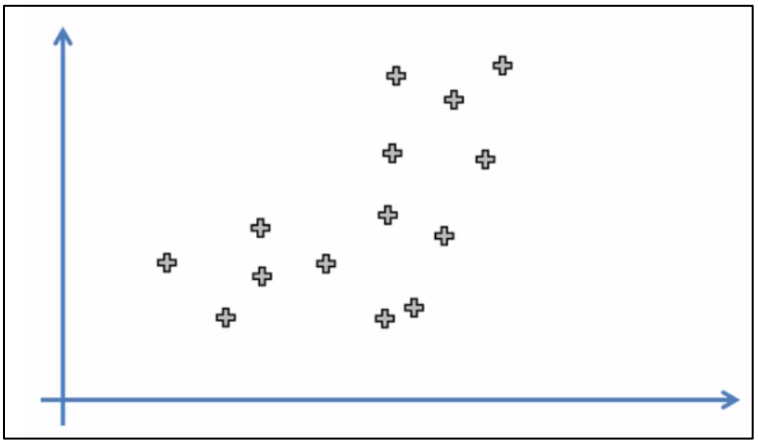
- It is not possible to undo the previous step: once the instances have been assigned to a cluster, they can no longer be moved around.
- Time and space complexity high: not suitable for large datasets
- Very sensitive to outliers

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

The K-Means algorithm steps is described below:

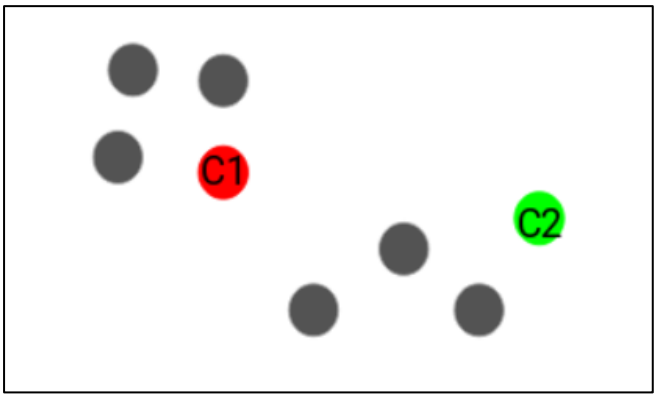
- It begins with **choosing the number of K clusters**. The K signifies the number of clusters that the algorithm would find in the dataset.

Step 1: Choose the number of clusters k



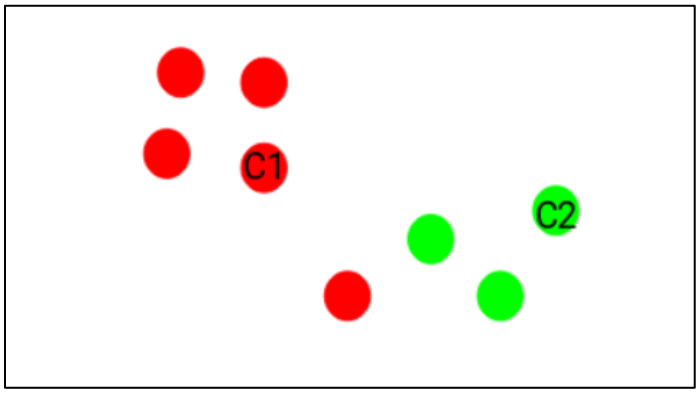
- The second step is to allocate K random points as centroids. These K points could be points from the dataset or outside. There's one thing to note however. The random initialization of centroids can sometimes cause random initialization trap

Step 2: Select k random points from the data as centroids (C1 & C2)



- In the third step the dataset points would be allocated to the centroid which is closest to them

Step 3: Assign all the points to the closest cluster centroid (C1 & C2)



Assignment Step:

The equation for the assignment step is as follows:

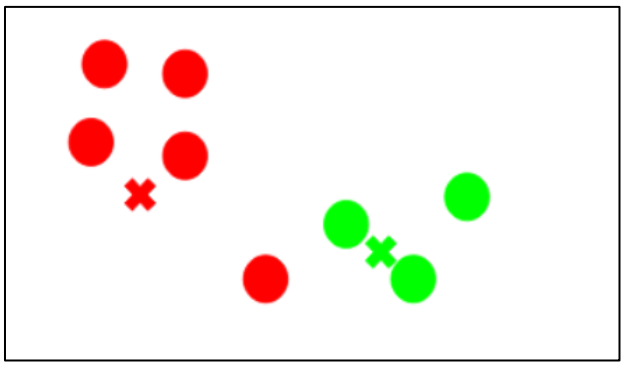
$$Z_i = argmin||X_i - \mu_k||^2$$

Xi is the datapoint

μk cluster centroid

- The next step is to compute the centroids of newly formed clusters. *The algorithm calculates the average of all the points in a cluster and moves the centroid to that average location*.

  Step 4: Recompute the centroids of newly formed clusters



Here, the **red and green crosses** are the new centroids.

Optimization Step:

The equation for optimization is as follows:

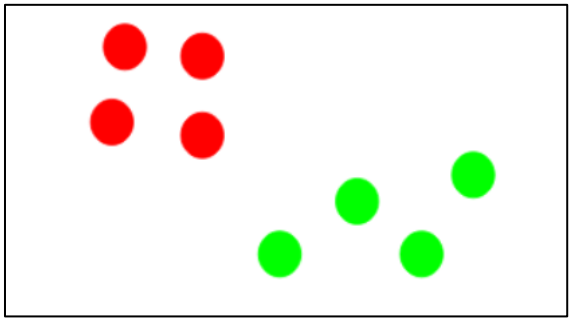
$$\mu_k = \frac{1}{n_k} \sum_{i:z_i=k} X_i$$

Xi is the datapoint

μk new cluster centroid

- The fifth step is to reassign points like we did in step 3. If reassignment takes place, then we need to go back to step four. If no reassignment takes place, then we can say that our model has converged and its ready.

  Step 5: Repeat steps 3 and 4



The **cost function for the K-Means** algorithm is given as:

$$J = \sum_{i=1}^{n} ||X_i - \mu_{k(i)}||^2 = \sum_{k=1}^{K} \sum_{i \epsilon C_k} ||X_i - \mu_k||^2$$

Xi is the datapoint

μk cluster centroid

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

The number of clusters that you want to divide your data points into, i.e. the value of K must be pre-determined.

There are two methods that can be useful to find this predetermined value of k in k-Means.

These methods are:

- The Elbow Method
- The Silhouette Method

The Elbow Method: This is probably the most well-known method for determining the optimal number of clusters. Calculate the ***Within-Cluster-Sum of Squared Errors (WSS) for different values of k*** and choose the k for which WSS becomes first starts to diminish. So, the point where this distortion declines the most is the **elbow point**.

Within-Cluster-Sum of Squared Errors:

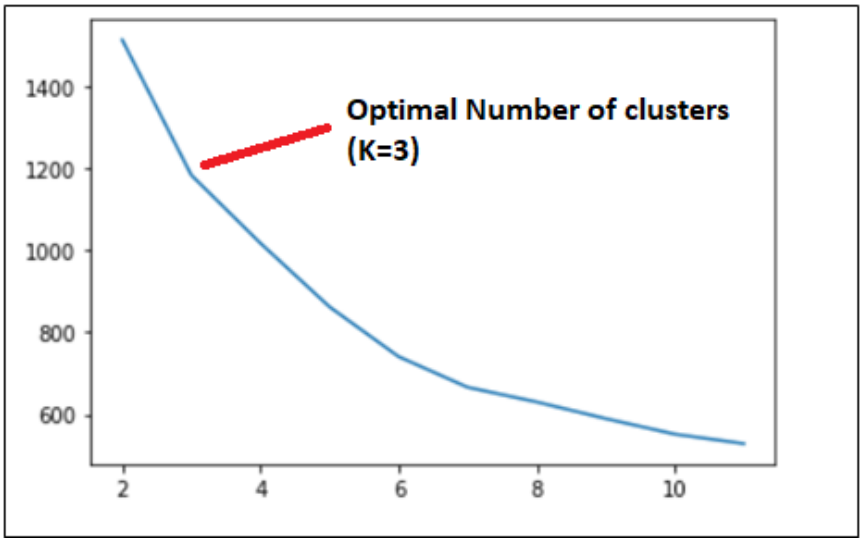
- The Squared Error for each point is the square of the distance of the point from its representation i.e. its predicted cluster center.
- The WSS score is the sum of these Squared Errors for all the points.

Python representation for Elbow Method:

```
clusters=[2,3,4,5,6,7,8,9,10,11]
ssd=[]
for i in clusters:
  kmeans=KMeans(n_clusters=i,max_iter=50)
  kmeans.fit(glass_df[cols])
  ssd.append(kmeans.inertia_)

plt.plot(clusters,ssd)
plt.show()
```

Elbow curve after plotting it:



The plot looks like an ***arm with a clear elbow at k = 3.***

The Silhouette Method: The silhouette value measures how similar a point is to its own cluster (cohesion) compared to other clusters (separation). ***The range of the Silhouette value is between +1 and -1.*** A high value is desirable.

To compute silhouette metric, we need to compute two measures i.e. a(i) and b(i)where,

a(i) is the average distance from own cluster (Cohesion).

b(i) is the average distance from the nearest neighbor cluster (Separation).

The **Silhouette Value s(i) for each data point i** is defined as follows:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, \text{ if } |C_i| > 1$$

and

$$s(i) = 0, \text{ if } |C_i| = 1$$

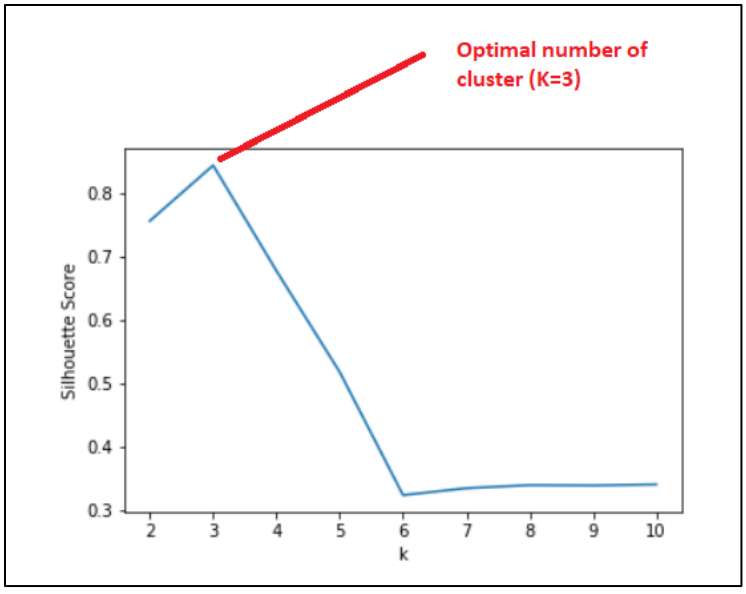
a(i) is the average distance from own cluster (Cohesion).

b(i) is the average distance from the nearest neighbor cluster (Separation).

Python representation for **Silhouette** Method:

```
clusters=[2,3,4,5,6,7,8,9,10,11]
for i in clusters:
  kmeans=KMeans(n_clusters=i,max_iter=50)
  kmeans.fit(glass_df[cols])
  sil=kmeans.labels_
  avg_sil=silhouette_score(glass_df[cols],sil)
  print('for cluster{0} avg silhoutte score{1}'.format(i,avg_sil))
```

After plotting the **Silhouette score, that was obtained from above**



There is a clear peak at k = 3.

Now comes the business aspect of it, means how many numbers of cluster 'k' are needed to choose with respect to business domain understanding.

Cluster analysis is a technique used in machine learning which groups data points together based on the similarities between them. One can use various clustering algorithms which provide valuable insights about your business. The information generated from clustering can be used across your business functions to create a profitable consumer response.

As far as clustering algorithms go, it is simple and flexible to use in retail business.

It needs to specify the number of clusters, which can be time-consuming or detrimental to business if they don't follow a **statistical or knowledge-backed method.**

Working with the optimal number of clusters for retail data and market environment will facilitate the use of resources in a more efficient and effective manner. One can select ***the number of clusters using industry-related knowledge or different statistical methods that is already discussed above.***

Below are **some of the business aspect based on which optimal number of clusters** can be chosen:

- We can use demographic, psychographic and behavioral data as well as performance data to cluster the consumers for a product category. *This is a part of consumer segmentation*.
- The delivery routes and patterns of trucks and drones have been monitored to find the optimal launch locations, routes and destinations for the company. *This is a part of delivery optimization*.
- You can use variables such as frequency of purchases, how recently the consumer visited the store, average spend per trip and basket composition to analyses and predict retention rates of customer segments, clustering or segmentation can be done based on RFM (recency, frequency, monetary value) analysis. *This is a part of customer retention*.

Our goal shouldn't be to just create clusters from the data. It should be to create meaningful, accurate clusters that one can use to generate insights about your business.

d) Explain the necessity for scaling/standardization before performing Clustering.

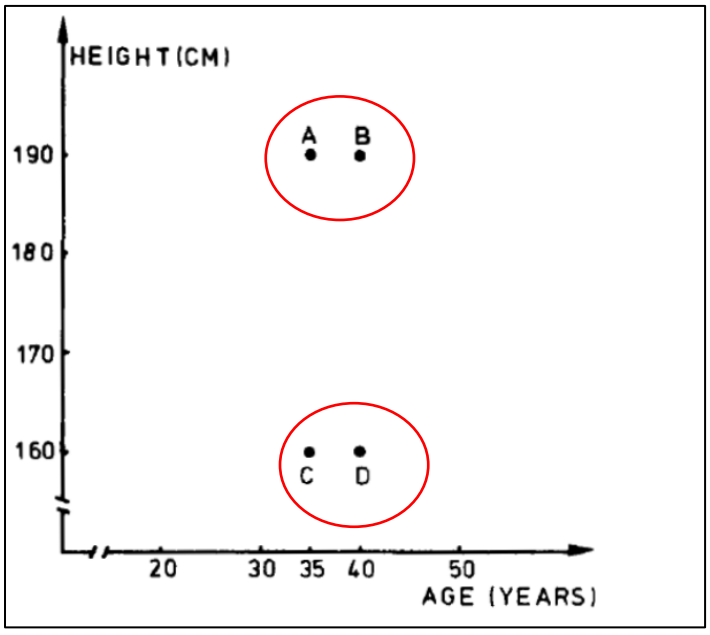
Since the distance metric used in the ***clustering process is the Euclidean distance***, we need to bring all the attributes on the same scale. This can be achieved through standardization.

Standardizing data is recommended because otherwise the range of values in each feature will act as a weight when determining how to cluster data, which is typically undesired.

In some applications, ***changing the measurement units may even lead one to see a very different clustering structure.***

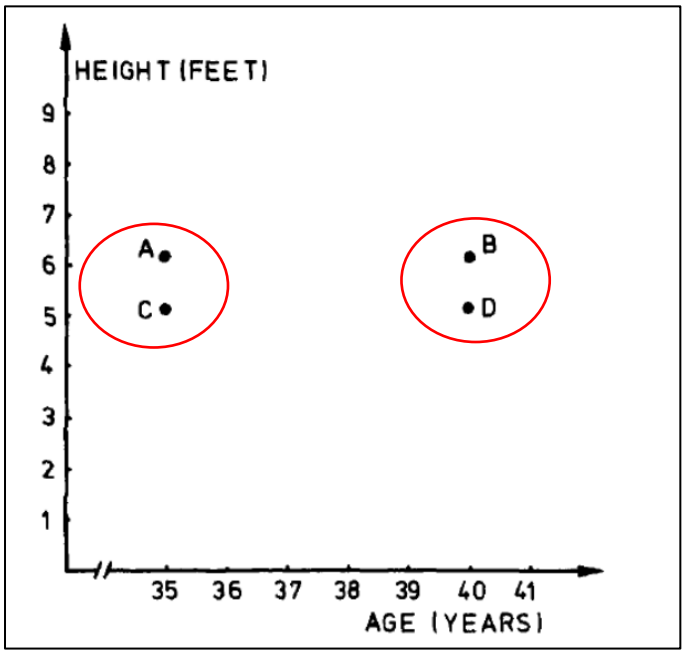
For example, the **age (in years) and height (in centimeters)** of four imaginary people are given and plotted. It appears that **{A, B) and {C, D}** are two well-separated clusters as shown below.

	Person	Age(Yr)	Height(cm)	cluster
0	A	35	190	0
1	B	40	190	0
2	C	35	160	1
3	D	40	160	1



On the other hand, when **height is expressed in feet** one obtains where the obvious clusters are now **{A, C} and {B, D}.** This partition is completely different from the first because each subject has received another companion.

	Person	Age (Yr)	Height(ft)	cluster
0	A	35	6.2	1
1	B	40	6.2	0
2	C	35	5.2	1
3	D	40	5.2	0



Hence from the above example ***it is very clear that to avoid this dependence on the choice of measurement units, one has the option of standardizing the data.***

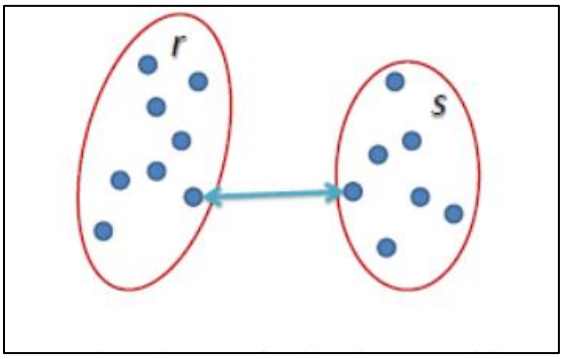
e) Explain the different linkages used in Hierarchical Clustering.

The process of Hierarchical Clustering involves either clustering sub-clusters into larger clusters in a bottom-up manner (Agglomerative clustering) or dividing a larger cluster into smaller sub-clusters in a top-down manner (Divisive clustering). During both the types of hierarchical clustering, the distance between two sub-clusters needs to be computed. ***The different types of linkages describe the different approaches to measure the distance between two sub-clusters of data points.***

The different types of linkages are: -

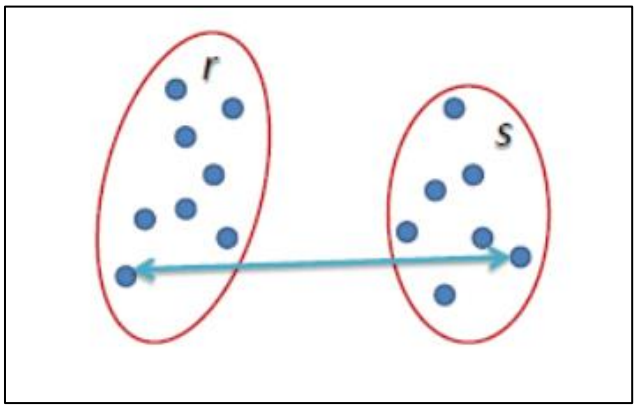
- **Single Linkage:** For two clusters r and s, ***the single linkage returns the minimum distance between two points i and j*** such that i belongs to r and j belongs to s.

$$L(R,S) = min(D(i,j)), i \epsilon R, j \epsilon S$$



- **Complete Linkage:** For two clusters r and s, ***the complete linkage returns the maximum distance between two points i and j*** such that i belongs to r and j belongs to s.

$$L(R,S) = max(D(i,j)), i \epsilon R, j \epsilon S$$



- **Average Linkage:** For two clusters r and s, first for the ***distance between any data-point i in r and any data-point j in s and then the arithmetic mean of these distances are calculated.*** Average Linkage returns this value of the arithmetic mean.

$$L(R,S) = \frac{1}{n_R + n_S} \sum_{i=1}^{n_R} \sum_{j=1}^{n_S} D(i,j), i \epsilon R, j \epsilon S$$