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(why you took that many numbers of principal components, which type of Clustering produced a better result and so on)

Note: You don't have to include any images, equations or graphs for this question. Just text should be enough.

The dataset given has financial, health, population growth, mortality related data of 167 countries of the world. HELP international need to allocate $10 million fund to the countries, who deserves the most. The challenges before us is to identify which 5 countries deserve this fund the most.

So this is a clustering problem. In data science this is part of unsupervised learning. For that purpose we should cluster these countries based on the given data. After we are done with clustering we need to select 5 countries from the created clusters.

Second problems which I tried to solve is if some new country is added in this dataset or values of given parameters of the existing country change then predict in which cluster that particular country should be placed.

Broadly we know 2 methods/algorithms of clustering. 1- KMeans 2- Hierarchical clustering. Both of these algorithms works differently on the given data to identify cluster. And they produce different results. Within Hierarchical clustering there are broadly 2 methods one is Divisive (top down), second is Agglomerative (bottom up).

In the hierarchical clustering there are 3 kinds of linkage between data points and clusters. 1- Single, 2-Complete, 3-Average. As per the assignment we should use 2 types of linkage (Single, Complete).

One of the biggest challenges there is out of given 9 parameters which one should be used for modeling purpose. Either use RFE or manual elimination we lose information with every single feature drop. On top of that it is not easy to identify importance/value order of these 9 given parameters. In this situation PCA (Principal Component Analysis) comes to our help. PCA methods compresses the given data and creates different columns. This removes collinearity problem so that no PCA column represent any redundant information. On top of that we can reduce the number of parameters/features, so instead of 9 features we can create our model using 4 PCA features because in our case 91% of information is compressed in 4 PCA features.

Question 2: Clustering
a) Compare and contrast K-means Clustering and Hierarchical Clustering.

Need of Predefined Clusters

- In KMeans we need to tell in advance how many cluster we want. This number is called k. Without K kmean algorithm will not start.
- Hierarchical clustering builds hierarchy connects data points using this hierarchy and does not require a pre-specified number of clusters like k means.

Output (Visual)

- KMeans does not give hierarchy of the data points. It just generate cluster number.
- Hierarchical Algorithm give a hierarchy. There are two approaches of developing this namely top-down and bottom-up.

In top-down hierarchical clustering, whole data set is considered as one cluster. First, we divide the data into 2 clusters (k=2). Then, for each cluster, we keep repeating this process, until all the clusters are too small or too similar for further clustering, or until we reach a preset number of clusters.

In bottom-up hierarchical clustering, we assume each data item is in its own cluster. We then look for the two items that are most similar and combine them in a larger cluster. We keep repeating until all the clusters we have left are too dissimilar to be gathered together, or until we reach a preset number of clusters.

Speed

- KMeans algorithm runs faster.
- Hierarchical clustering is very slow. If we do not have much memory and data size is big then Hierarchical clustering can eat all the memory and processing capacity.

Reuse

- Once Hierarchical clustering is done, we can use the same tree to generate different clusters
- In KMeans, if our requirements of number of cluster changes then we need to rerun the algorithm again.

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

1. Identify number of clusters required, let us say K
2. Randomly select K point in the dataset and treat them as center (centroid)

3. Calculate the *distance from each data point to each of these selected centroid
4. Assign the these datapoint to that centroid from which they have minimum distance. Thus we form the cluster of data points around the centroid.
5. Calculate the centroid using the new data points (post assignment)
6. Repeat from step 3 with the new the centroid
7. Keep doing this till centroids are changing

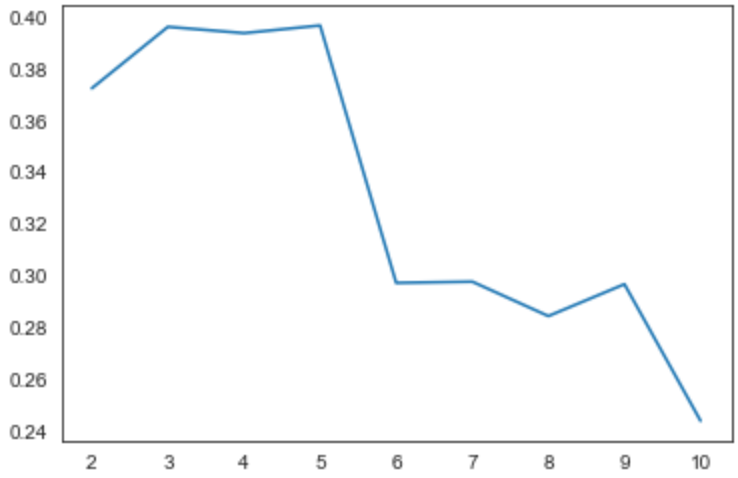
Once centroid stop changing we get the stable clusters.

***There are 2 popular ways of calculating the distance between data points**
1- Euclidean distance - Sum of Squares of difference between corresponding data points. ED = (a1-a2)^2 + (b1-b2)^2 + (c1-c2)^2
2- Manhattan Distance. Sum of absolute difference between corresponding data points. MD = abs(a1-a2) + abs(b1-b2) + abc(c1-c2)

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

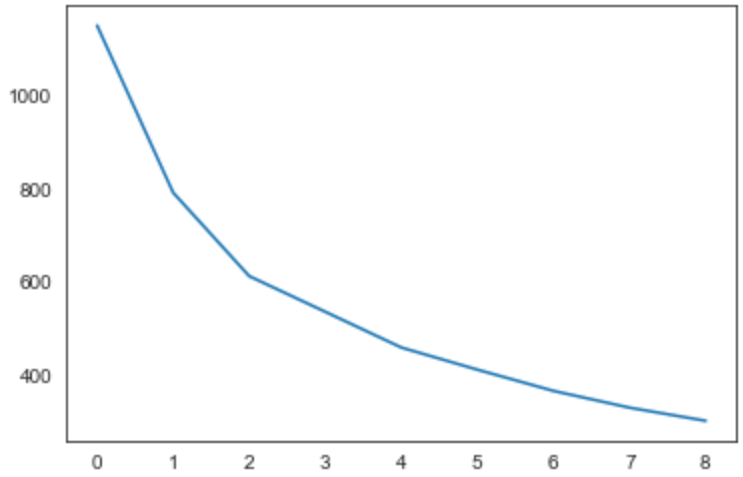
Statistical Methods

1. **Dendogram Horizontal Cut**: Once we create a dendogram we can see at what level we can cut the tree so that we get logical number of clusters. Logical means which can be explained, understood and acted upon. It should not happen that data points between different clusters are hugely disproportionate. There is no benchmark for this "hugely" but by seeing the dendrogram you can sense that easily.
2. **Silhoutte Coefficient**. It is measure of cohesiveness & separation. It shows the strength of cluster. There are two measures for this. One, all the elements of a cluster should have min average distance (cohesiveness). Two, the distance between the elements of near cluster should be maximum (separation)



Silhoutte Coefficient = (b(i)-a(i)) / max(b(i), a(i)) where b(i) >> a(i)

3. **Elbow Curve**. Wherever we find the sharp bent in this curve we can take that as k. In the below image we can take 2 or 4.



4.

Business Aspect

Business managers know from their business domain that how many clusters can be created from this dataset or how many clusters they can manage. For example, if a data set has sales information then it can be divided based on regions like east, west, north, south. It can also be divided using states or metro cities etc. What we want to choose as a basis to create cluster depends upon business, how they want to strategize or how many actions/plan they can manage or how many resources they have.

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

A dataset can have different kinds of information like profit, sales value, quantity, tax. Although all is quantitative information but they vary from each other either in unit of measurement and scale of measurement. For example quantity may be in unit or box or kg but money is in either Rupee or dollar. On the other hand sale can be Rs. 10,000 but profit may be 400.

With this kind of data if you create a model then coefficient will be highly distorted. Similarly if you want to perform the clustering with this kind of data then you will not be able to form stable and reliable clusters. Therefore it makes sense to scale the data of each column/feature/parameter either using standard scale or min-max scale.

In min-max scale data will be will replaced with a new value which is between 0 to 1. Zero is min, 1 is max. Problem with min-max scale is outliers get compressed within the scale. Which in turn affects the quality of cluster.

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Using standard scale we assign the measure of deviation from the mean. The mean of the distribution in this case is is zero and standard deviation is 1

$$x' = \frac{x - x_{mean}}{\sigma}$$

e) Explain the different linkages used in Hierarchical Clustering.

Hierarchical clustering can be done either using either agglomerative technique (bottom up) or divisive technique (top down). The concept of linkage helps us to understand which points of the clusters are taken to measure the distance between clusters. This also impacts the stability of a cluster.

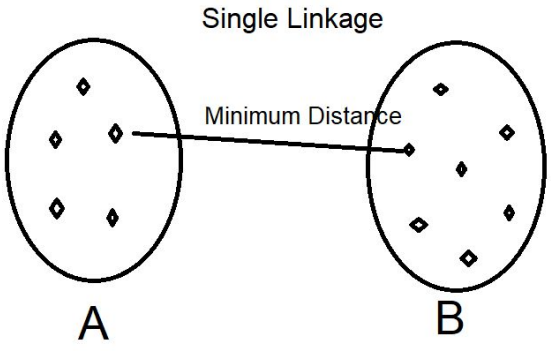
There are three kinds of linkage between clusters.

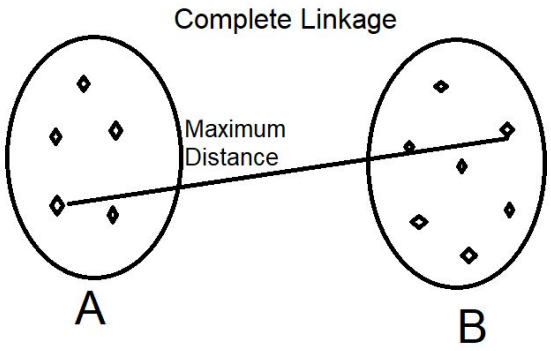
Suppose there are 2 clusters A & B. Cluster A has 5 data points and Cluster B has 7 data points. Let us calculate the distance from all 5 individual points of cluster A to all 7 data points of cluster B.

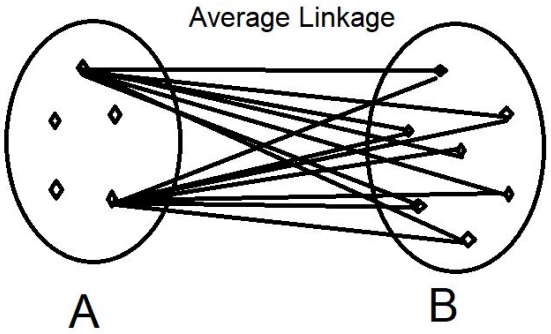
Single Linkage : Whatever is the minimum distance from above calculation is represented by Single Linkage.

Complete Linkage : Whatever is the maximum distance from above calculation is represented by Complete Linkage.

Average Linkage : If we take the average of above all the distances, then that distance is represented by Average Linkage







This way calculate distance between all points of cluster A to all point of cluster B. Then Calculate average

Question 3: Principal Component Analysis

a) Give at least three applications of using PCA.

1. **Feature reduction**: When number of features are very high and there is multicollinearity in the data we can use PCA to compress the information. Using PCA we can have a dataset with lesser number of features which contains almost the same amount of information which is available with original features.
2. **EDA**: When number of features are very high and we are studying relationships among different features using pair feature method then it is highly impractical to extract meaningful information after doing EDA. In this situation if we perform PCA and then do EDA, we will find that all the relationships among different features will be almost zero.
3. **Creating new feature** which is not related to any of the existing feature.
4. **Uncovering Latent information**: Sometimes many features together in a dataset forms a theme, and it is not possible to know that theme if you look these features individually. PCA helps us in combining these features and creating a theme based feature. Creating theme or category of product from millions of movies or products.

5. **Noise Reduction**. In a large features dataset there are some features which are just noise, not adding much value for prediction. It is not easy to remove this kind of noise. PCA helps doing this noise reduction.

b) Briefly discuss the 2 important building blocks of PCA – Basis transformation and variance as information.

Basis Transformation:
Basis is set of linearly independent vectors or direction (they are orthogonal or perpendicular to each other). Let's say we have two dimension space which has 2 basis vectors i (along the x axis) and j (along the y-axis). Using these 2 basis vectors we can represent any point in the 2 dimensional space.

Similarly if space is 3 D and we have 3 basis vectors i, j, k along x,y,z direction respectively then using these 3 basis vectors any point can be represented in the 3 D space.

In relation to database basis vectors can be compared to columns or features. And row/tuples can be compared to point in the space represented by these basis vectors.

Variance as Information
First and foremost important thing to note is, it applies only to numerical data. Every column has some numerical information. There is min number and max number (value) for any given column. This is called range. If we take the mean of a column and measure the square of difference between mean and particular value of column then we know the variance of that value from mean. When we sum all the variances of the column we know the variance of the column/feature. More the variance better it is for modeling purpose because that variation is explaining something about each row/record in the dataset.

***c) State at least three shortcomings of using Principal Component Analysis*.**

Linearity Assumption: It assumes that data in the space are on some linear line and pointing towards a direction (vector). So, it does not work well when data shape is not linear.

Orthogonality: It assumes that basis vectors are perpendicular (orthogonal) to each and there is no relationship between two basis vectors. But in reality we know there is some kind of relationship between the columns (it may be extremely weak, but it exists). So, more the multicolearnity between different features worse the

performance using PCA. Sometimes non-orthogonal useful variables are there is dataset and transforming that information into PCA gives poor results.

Variance: Largers the variances of information in any column or lesser the covariance between the columns higher the importance of features. This property of PCA is not always useful. Problems like BSS (Blind source separation) can not be solved by PCA.

Scale: If data is not scaled then PCA will fail.