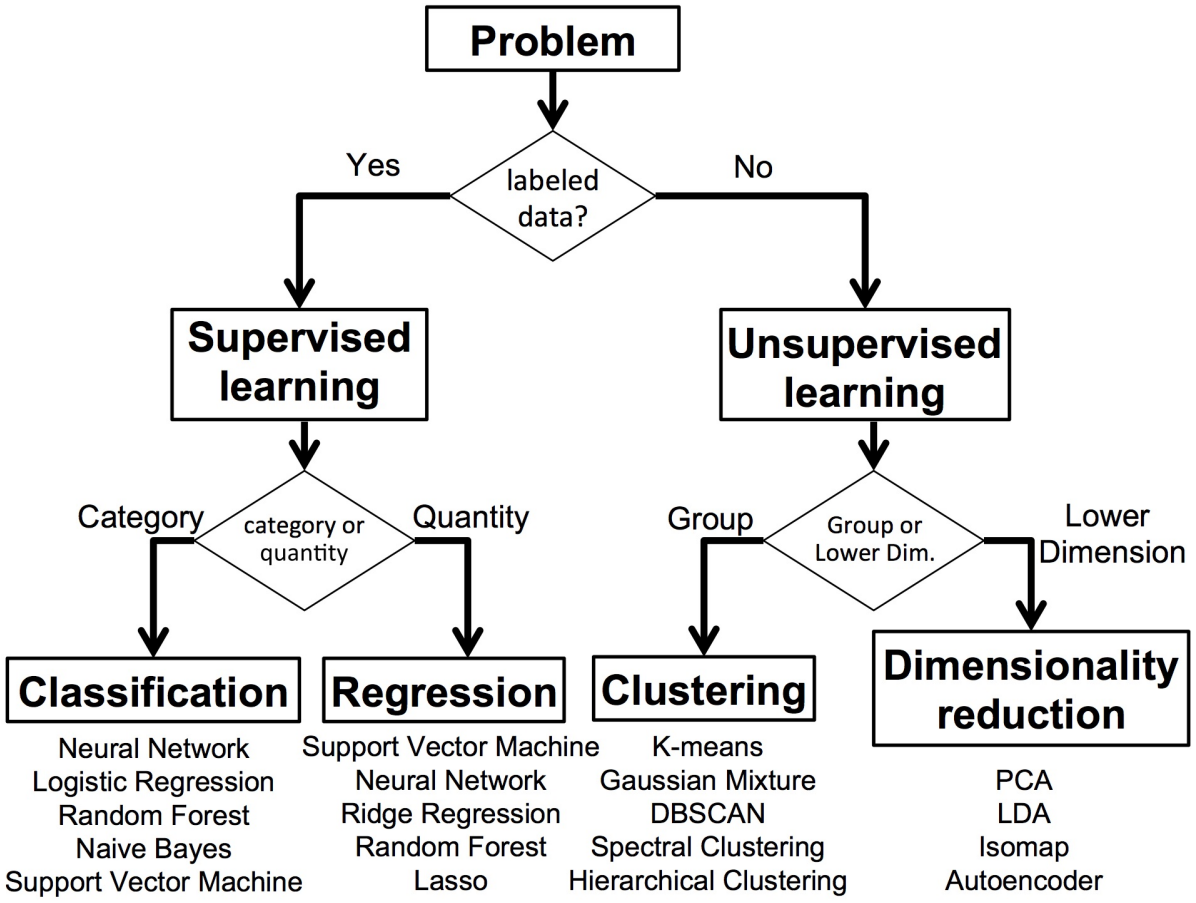
Synapse
Task 2

Unsupervised Learning

What is Unsupervised Learning?

Models find the insights and hidden patterns itself, from the given data, it is like the human brain learning new things. There are no correct answers and its goal is to learn about the data.



.

Question 1: Preprocess the data and handle null values (usually, real world datasets are never clean. Null values can be present because of many reasons ranging from a simple data entry error to a loss of data.)

So as we know data that we receive is not always perfect. Particular elements are missing and hence we need to fix this, and then use our algorithms. So that the result has the highest chance of accuracy and scope of error should be the lowest.

PREPROCESSING

We import the necessary libraries we need after adding it on our system using the command prompt.
Libraries such as Pandas,Matplot, Numpy,Sklearn.preprocessing etc.. are extremely crucial and required
After we import libraries, for example we have a csv file then we will read it using pandas
pd.read_csv('Name_Of_CSV_File')

NULL VALUES

Then find out the null values by using the function *.isnull()*
After this to know the total value of null we add *.sum()* for each variable in the dataset
Depending upon the data we can avail any of these methods:-

1) Deleting
2) Replacing With Mean/Median/Mode

DELETING
-This method is commonly used to handle the null values. Here, we either delete a particular row if it has a null value for a particular feature and a particular column if it has more than 70-75% of missing values.
This method is advised only when there are enough samples in the data set.
data.drop('Name',axis=1,inplace=True)

REPLACING

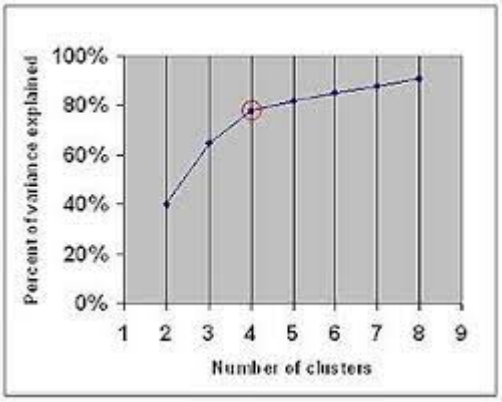
This strategy can be applied on a feature which has numeric data like the age of a person. We can calculate the mean, median or mode of the feature and replace it with the missing values. This is an approximation which can add variance to the data set. But the loss of the data can be negated by this method which yields better results compared to removal of rows and columns

Question 2: What clustering algorithm you'll use and why? Explain what you understand about the algorithm in detail.

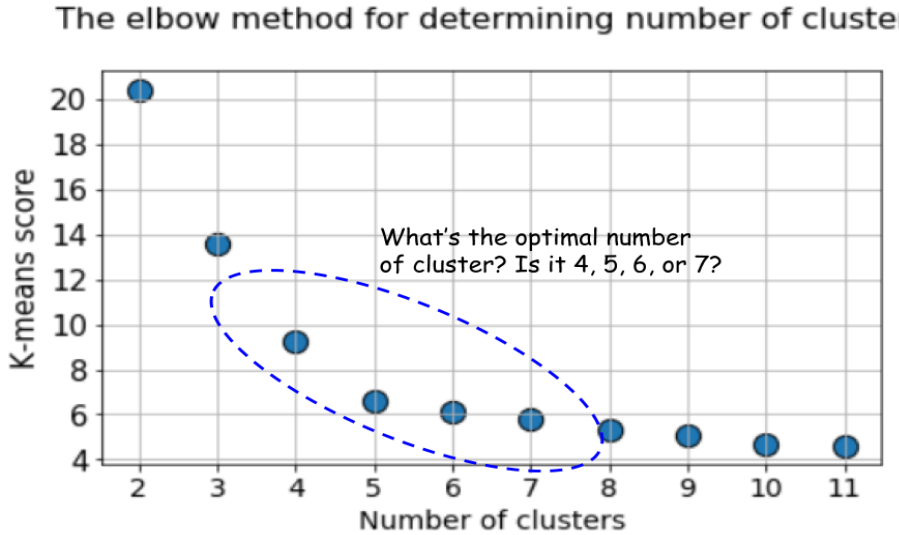
I would choose K-Means because hierarchical clustering excels at discovering embedded structures within the data, and density-based approaches shine at finding an unknown number of clusters of comparable density. However, both fail at finding a consensus across the complete dataset. Hierarchical clustering can put together clusters that appear close, but no information about other points is taken into account . Density-based methods only check out a little neighborhood of nearby points and similarly fail to think about the complete dataset.
And we cannot afford to ignore the data set in this
K-means works by selecting k central points, or means, hence K-Means. These means are then used because the centroid of their cluster: any point that's closest to a given mean is assigned to that mean's cluster.
Once all points are assigned, move through each cluster and take the average of all points it contains. This new 'average' point is the new mean of the cluster.
Just repeat these two steps over and once again until the purpose assignments stop changing
The Forgy Method randomly selects *k* random observations from the data and uses these as the starting points. The Random Partition Method will assign every point in the dataset to a random cluster, then calculate the centroid from these and resume the algorithm.

Question 3: How will you determine the number of clusters to divide the customers into?

In cases with k Means we need to determine the number of clusters, while dbscan and all don't really require it.
Hence there are many ways to determine the number of clusters, such as the Elbow Method, X ray, etc..
We can Use the Elbow method, it is basically like trial and error where the number of clusters ‘k’ is plotted with variance. We plot the share of variance explained by the clusters against the amount of clusters, the primary clusters will add much information (explain tons of variance), but at some point the marginal gain will drop or the loss of variance will drop marginally, giving an angle within the graph. the amount of clusters is chosen at now , hence the "elbow criterion". This "elbow" cannot always be unambiguously identified, making this method very subjective and unreliable.



Now we can plot this with k means score also, score refers to the each value of k computes an average score for all clusters. By default, the distortion score is computed, the sum of square distances from each point to its assigned center.



We can also use reduction of dimensions using PCA to find the number of cluster points, when the data is projected into a lower dimension from a higher space, the dimensions are nothing but the Principal Components that capture most of the variance of your data. In this you lose some of variance, so explained_varaince_raio can be used and generally if it's above 85%, we can use it.