**Documentation Of KMeans Clustering Using Python**

1)Perform K means clustering for the airlines data to obtain optimum number of clusters.

Draw the inferences from the clusters obtained.

Business Objective:

Analyze the information given in the following ‘EastWestAirlines dataset’ to prepare KMeans Clustering model.

solution:

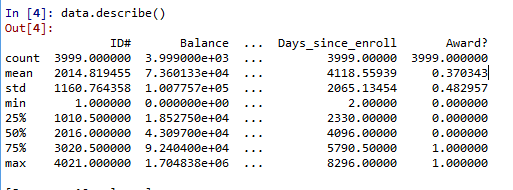
step1:

load the EastWestAirlines.xlsx dataset to the airlines variable

Exploring data

After you have loaded the dataset, you might want to know a little bit more about it.

for that describe function is used as



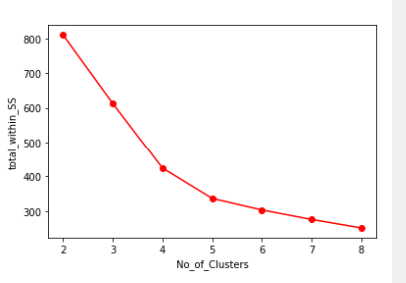
To cluster the data we should normalize the data for proper clustering

normalization is done by using customized function of norm() which takes data as parameter.

later norm() is applied on numerical data columns

Scree plot or elbow plot

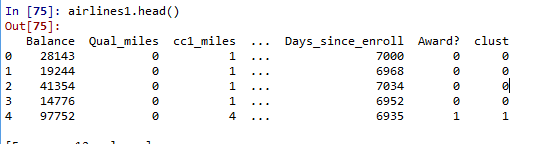
For k value ranging from 3 to 9 the scree plot is drawn



By observing the above scree plot the steep is observed between 3 to 5

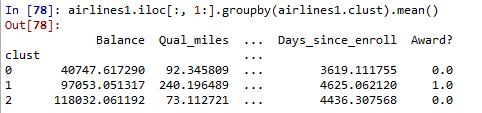
so k value is 3

the clusters are formed and inspecting first few rows after clustering

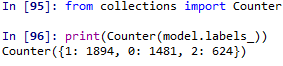


Aggregate mean of each cluster is calculated to be grouped

airlines1.iloc[:, 1:].groupby(airlines1.clust).mean()



to determine each has how many values Counter module is used which is imported from collections



cluster 0:1481

cluster2: 1894

cluster 3: 624

for k=4



cluster 0=1897

cluster1=673

cluster2=808

cluster3=621

for k=5



cluster0: 673

cluster1: 1032

cluster2: 868

cluster3: 618

cluster 4: 808

2.)Perform clustering for the crime data and identify the number of clusters formed and draw inferences. Refer to crime\_data.csv dataset.

Business Objective:

Analyze the information given in the following crime\_data.csv ‘dataset’ to prepare KMeans Clustering model.

solution:

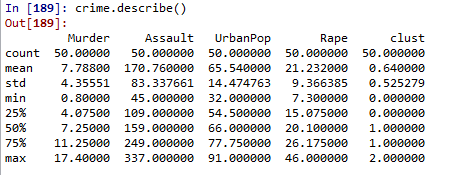
step1:

load the crime\_data.csv dataset to the airlines variable

Exploring data

After you have loaded the dataset, you might want to know a little bit more about it.

for that describe function is used as



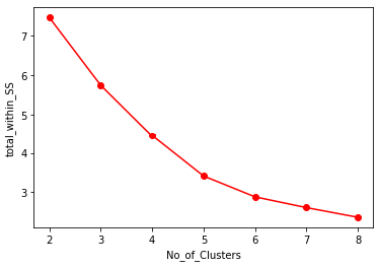
To cluster the data we should normalize the data for proper clustering

normalization is done by using customized function of norm() which takes data as parameter.

later norm() is applied on numerical data columns

Scree plot or elbow plot

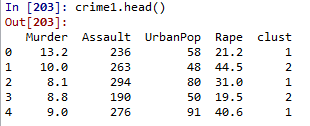
For k value ranging from 3 to 9 the scree plot is drawn



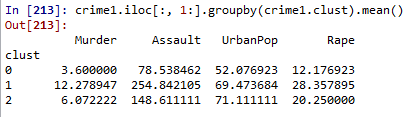
By observing the above scree plot the steep is observed between 3 to 5

so k value is 3

the clusters are formed and inspecting first few rows after clustering



Aggregate mean of each cluster is calculated to be grouped



to determine each has how many values Counter module is used which is imported from collections



cluster 0: 13

cluster1: 18

cluster 2: 13

for k=4



cluster 0=12

cluster1=18

cluster2=13

cluster3=7

for k=5



cluster0: 7

cluster1: 13

cluster2: 12

cluster3: 1

cluster 4: 17

1. Analyze the information given in the following ‘Insurance Policy dataset’ to create clusters of persons falling in the same type. Refer to Insurance Dataset.csv

Business Objective:

Analyze the information given in the following Insurance Dataset.csv ‘dataset’ to prepare KMeans Clustering model.

solution:

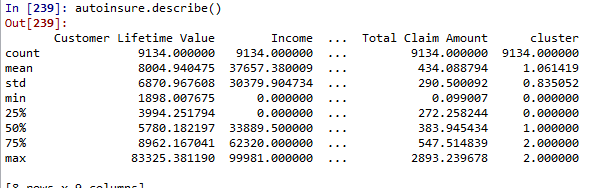
step1:

load the Insurance Dataset.csv dataset to the airlines variable

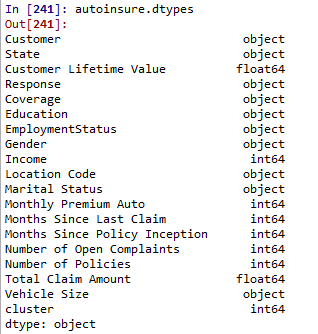
Exploring data

After you have loaded the dataset, you might want to know a little bit more about it.

for that describe function is used as



the unwanted columns are removed and then data type of each column is determined by



as observed most of the data points are of categorical so Euclidean distance between two data points is not gives better results so we find growers distance . for that

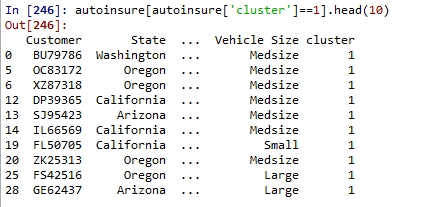
from kmodes.kprototypes import KPrototypes

initially k for 3 values kprototypes are formed and that model is fitted on autoinsure data

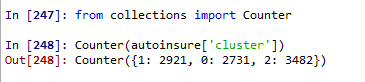
cluster=kproto.fit\_predict(ainsure\_array,categorical=[0,1,3,4,5,6,7,9,10,17])

Best run was number 1

after clustering the first few rows of cluster 0 are



the count of data points fall under each cluster is determined by Counter() which is imported from collections



cluster0 =2731

cluster1=2921

cluster2=3482