Ashpak Sheikh
KMEANS CLUSTERING
BATCH : DSWDMOD 020421

Business Problem:- Perform clustering on mixed data convert the categorical variables to numeric by using dummies or Label Encoding and perform normalization techniques. The data set consists details of customers related to auto insurance

About data:- We have been given data about the auto insurance of customers about there coverage , customer ID, offers etc.

Analysis with Python:-

```
import pandas as pd

import matplotlib.pyplot as plt

auto=pd.read_csv("D:/DataScience/Class/assignment working/h_clustering/AutoInsurance.csv")
```

Checking descreption

```
auto.describe()
```

```
In [247]: #checking descreption

In [248]: auto.describe()
Out[248]:
       Customer Lifetime Value  ...  Total Claim Amount
count              9134.000000  ...         9134.000000
mean               8004.940475  ...          434.088794
std                6870.967608  ...          290.500092
min                1898.007675  ...            0.099007
25%                3994.251794  ...          272.258244
50%                5780.182197  ...          383.945434
75%                8962.167041  ...          547.514839
max               83325.381190  ...         2893.239678

[8 rows x 8 columns]
```

Removing unwanted columns

```
auto_1=auto.drop(["Customer","State","Vehicle Size","Effective To Date","Location Code"],axis=1)


#creating dummy variables for categorical data
```

auto_1.isna().sum() no null values

```
auto_dummies=pd.get_dummies(auto_1,drop_first=True).astype(int)
```

Ashpak Sheikh

KMEANS CLUSTERING

BATCH : DSWDMOD 020421

EDA

```
auto_dummies.agg(["mean","median","var","std","skew","kurt"])
```

```
In [255]: auto_dummies.agg(["mean","median","var","std","skew","kurt"])
Out[255]:
        Customer Lifetime Value  ...  Vehicle Class_Two-Door Car
mean               8.004437e+03  ...                    0.206481
median             5.780000e+03  ...                    0.000000
var                4.721020e+07  ...                    0.163865
std                6.870968e+03  ...                    0.404802
skew               3.032282e+00  ...                    1.450502
kurt               1.382353e+01  ...                    0.103978

[6 rows x 43 columns]
```

```
def normalize(x):
    w=(x-x.min())/(x.max()-x.min())
    return w

norm_auto=normalize(auto_dummies)
```

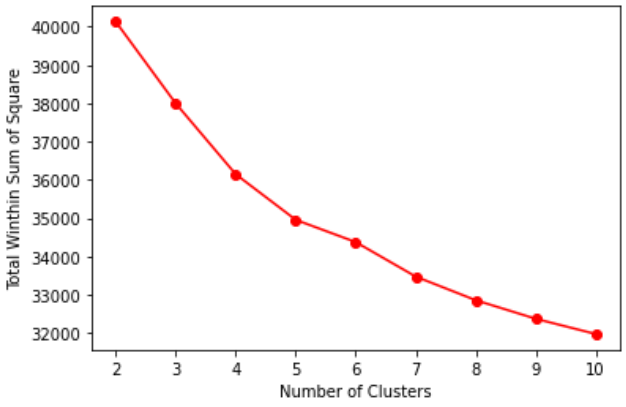
Kmean clusters formation

```
from sklearn.cluster import KMeans
#calculating inertia for different numbers of clusters
twss=[]
i=range(2,11)
for x in i:
    kmeans=KMeans(n_clusters=x)
    kmeans.fit(norm_auto)
    twss.append(kmeans.inertia_)

twss

#plotting scree plot
plt.plot(i,twss,"ro-");plt.xlabel("Number of Clusters");plt.ylabel("Total Winthin Sum of Square")
```

Ashpak Sheikh
KMEANS CLUSTERING
BATCH : DSWDMOD 020421



#at cluster 5 therenis maximum bend ,so choosing 5 clusters

KMeans clustering

kmeans=KMeans(n_clusters=5)

kmeans.fit(norm_auto)

auto["clusters"]=kmeans.labels_

#grouping cluters

auto.groupby(by="clusters").mean()

clusters	ımer Lifetime '	Income	hly Premium	ths Since Last C	ıince Policy I	ɛr of Open Com	ımber of Polici	Total Claim Amount
0	8001.49	46244…	93.64	14.9346	47.3883	0.337561	3.13171	395.834
1	7970.16	46182…	92.5057	15.3209	47.5005	0.365524	2.97352	378.311
2	8214.97	51265…	93.9698	14.9541	48.0209	0.386967	2.92156	376.99
3	7601.43	842.2…	94.0078	15.114	49.0378	0.399482	3.0487	629.523
4	8190.1	44448…	91.8359	15.1296	48.0702	0.411998	2.83475	383.947

Summary and Inference :-

- Data has been grouped into five categories based on the different weightage of columns
- We have five type of customers amongst which 4th group of customers have maximun number of claim amount
- As the number of policies increases, number of claims also increases
- Maximum the costumer lifetime minimum the claim