## **Case -1 : Clustering : mtcars**

|  |  |
| --- | --- |
| Problem Statement | Mtcars contains data description of 32 cars eg. mpg, wt, hp  Cluster the cars into groups so that they described in that manner |
| Objectives | Perform Kmeans Clustering for 2 clusters |
| Research Qs | Find mean of mpg, hp, wt of each representative value of 2 clusters |
| Tasks | Ensure that data is numeric  Scale the data  Perform Kmeans with 2 clusters  Find the cluster groups  Calculate the group means of each cluster (mpg, wt, hp) |

|  |
| --- |
| Python Code |
| #Topic: Assignment - Clustering - mtcars  #-----------------------------  #libraries  import matplotlib.pyplot as plt  from kneed import KneeLocator  from sklearn.datasets import make\_blobs  from sklearn.cluster import KMeans  from sklearn.metrics import silhouette\_score  from sklearn.preprocessing import StandardScaler  from pydataset import data  mtcars = data('mtcars')  data = mtcars.copy()  data  #need for scaling : height & weight are in different scales  scaler = StandardScaler()  scaled\_features = scaler.fit\_transform(data)  scaled\_features[:5] #values between -3 to +3  kmeans = KMeans( init = 'random', n\_clusters=?? , n\_init=3, max\_iter=300, random\_state=42)  kmeans  kmeans.fit(scaled\_features)  kmeans.inertia\_  kmeans.cluster\_centers\_ #average or rep values  kmeans.n\_iter\_ #in 6 times, clusters stabilised  kmeans.labels\_[:5]  kmeans.cluster\_centers\_.shape  kmeans.cluster\_centers\_[0:1]  #mean of mpg, hp, wt  data.groupby(kmeans.labels\_).agg({'mpg':'mean','hp':'mean','wt':'mean'})  ##plot scatter wt vs mpg with color cluster  plt.scatter(x=...., y=..., c= ….) |