

customer-clustering

December 1, 2023

1 Customer classification on the basis of spending and earning

1.1 1) Importing libraries

```
[1]: from sklearn.cluster import KMeans
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from matplotlib import pyplot as plt
```

1.2 2) Loading data

```
[17]: df= pd.read_csv('Mall.csv')
df
```

```
[17]:
```

	CustomerID	Gender	Age	Annual Income (k\$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40
..
195	196	Female	35	120	79
196	197	Female	45	126	28
197	198	Male	32	126	74
198	199	Male	32	137	18
199	200	Male	30	137	83

[200 rows x 5 columns]

```
[18]: df.info
```

```
[18]: <bound method DataFrame.info of
```

	CustomerID	Gender	Age	Annual Income (k\$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77

4	5	Female	31	17	40
..
195	196	Female	35	120	79
196	197	Female	45	126	28
197	198	Male	32	126	74
198	199	Male	32	137	18
199	200	Male	30	137	83

[200 rows x 5 columns]>

1.3 3) Elbow methode

```
[20]: sse = []
      k_rng = range(1,10)
      for k in k_rng:
          km = KMeans(n_clusters=k)
          km.fit(df[['Annual Income (k$)', 'Spending Score (1-100)']])
          sse.append(km.inertia_)
      sse
```

```
C:\Users\luhar\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:1416:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  super()._check_params_vs_input(X, default_n_init=10)
C:\Users\luhar\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:1416:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  super()._check_params_vs_input(X, default_n_init=10)
C:\Users\luhar\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:1416:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  super()._check_params_vs_input(X, default_n_init=10)
C:\Users\luhar\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:1416:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  super()._check_params_vs_input(X, default_n_init=10)
C:\Users\luhar\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:1416:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  super()._check_params_vs_input(X, default_n_init=10)
C:\Users\luhar\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:1416:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  super()._check_params_vs_input(X, default_n_init=10)
```

```

    super()._check_params_vs_input(X, default_n_init=10)
C:\Users\luhar\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:1416:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
    super()._check_params_vs_input(X, default_n_init=10)
C:\Users\luhar\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:1416:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
    super()._check_params_vs_input(X, default_n_init=10)

```

```

[20]: [269981.28,
      181363.59595959596,
      106348.37306211119,
      73679.78903948834,
      44448.45544793371,
      37265.86520484346,
      30241.343617936585,
      25011.839349156595,
      21818.114588452183]

```

```

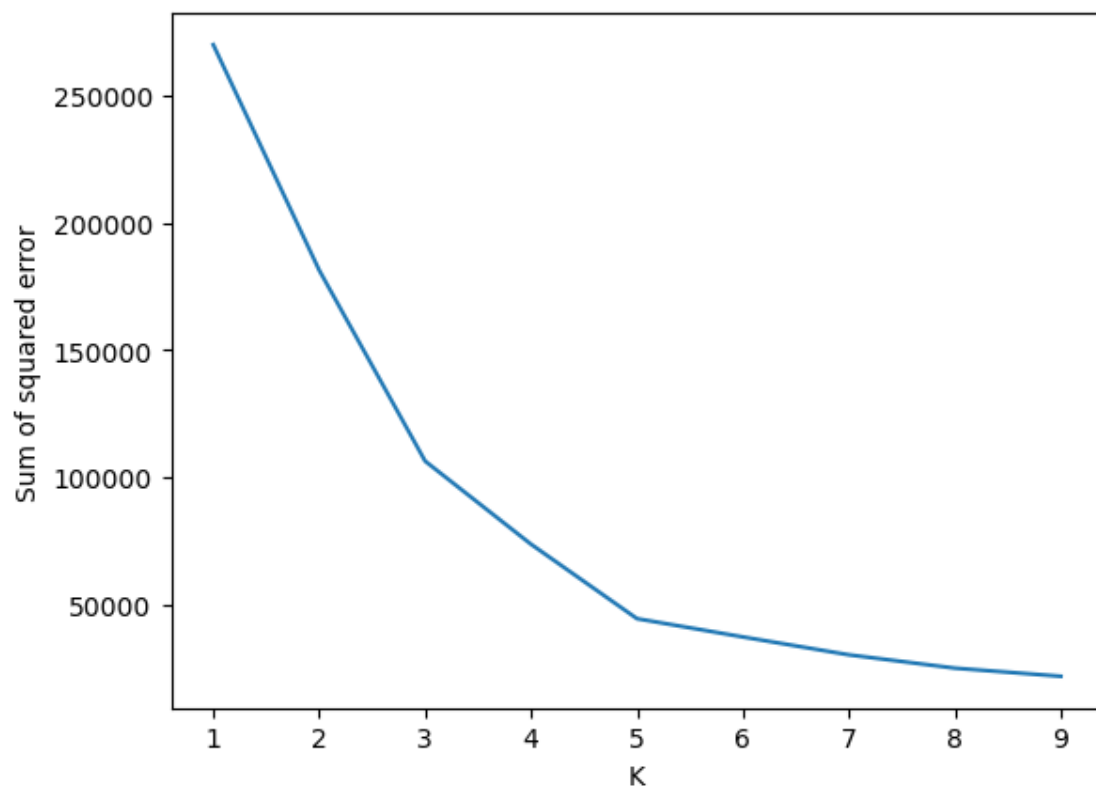
[21]: plt.xlabel('K')
      plt.ylabel('Sum of squared error')
      plt.plot(k_rng,sse)

```

```

[21]: [<matplotlib.lines.Line2D at 0x15b42fa78e0>]

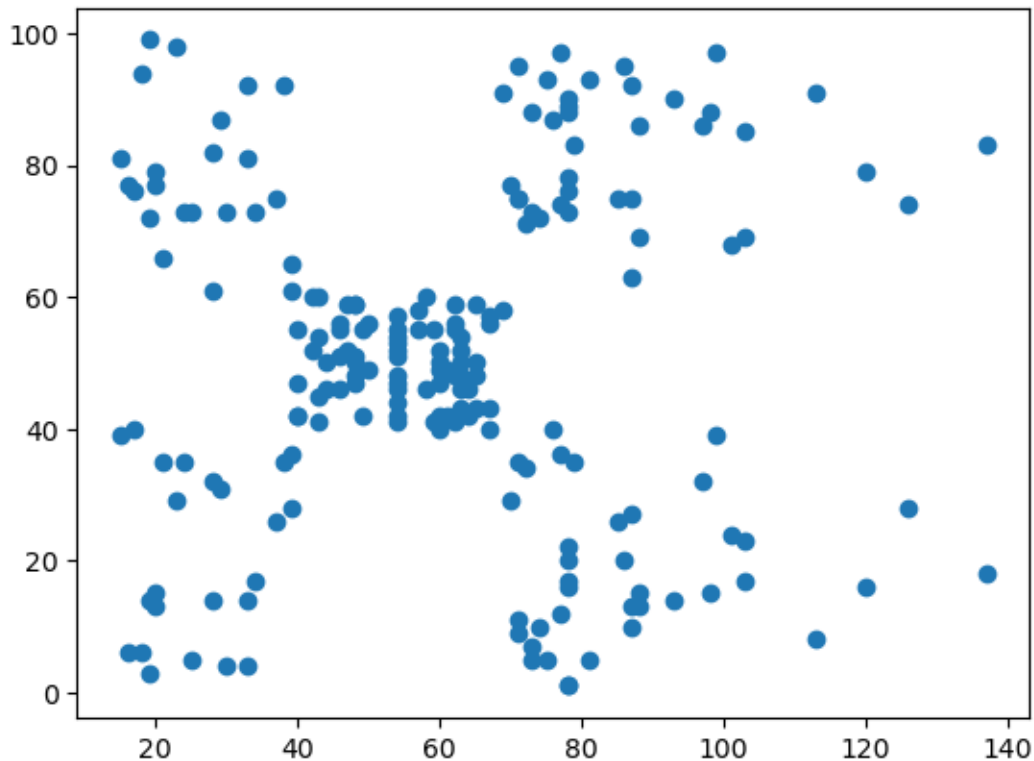
```



1.4 Clustering

```
[22]: plt.scatter(df['Annual Income (k$)'],df['Spending Score (1-100)'])
```

```
[22]: <matplotlib.collections.PathCollection at 0x15b41d236a0>
```



```
[23]: km = KMeans(n_clusters=5)
y_predicted = km.fit_predict(df[['Annual Income (k$)', 'Spending Score (1-100)']])
y_predicted
```

```
C:\Users\luhar\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:1416:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
    super()._check_params_vs_input(X, default_n_init=10)
```

[illegible]

```
[24]: df['cluster']=y_predicted
df.head()
```

```
[24]:
```

	CustomerID	Gender	Age	Annual Income (k\$)	Spending Score (1-100)	\
0	1	Male	19	15	39	
1	2	Male	21	15	81	
2	3	Female	20	16	6	
3	4	Female	23	16	77	
4	5	Female	31	17	40	


```

cluster
0      4
1      2
2      4
3      2
4      4

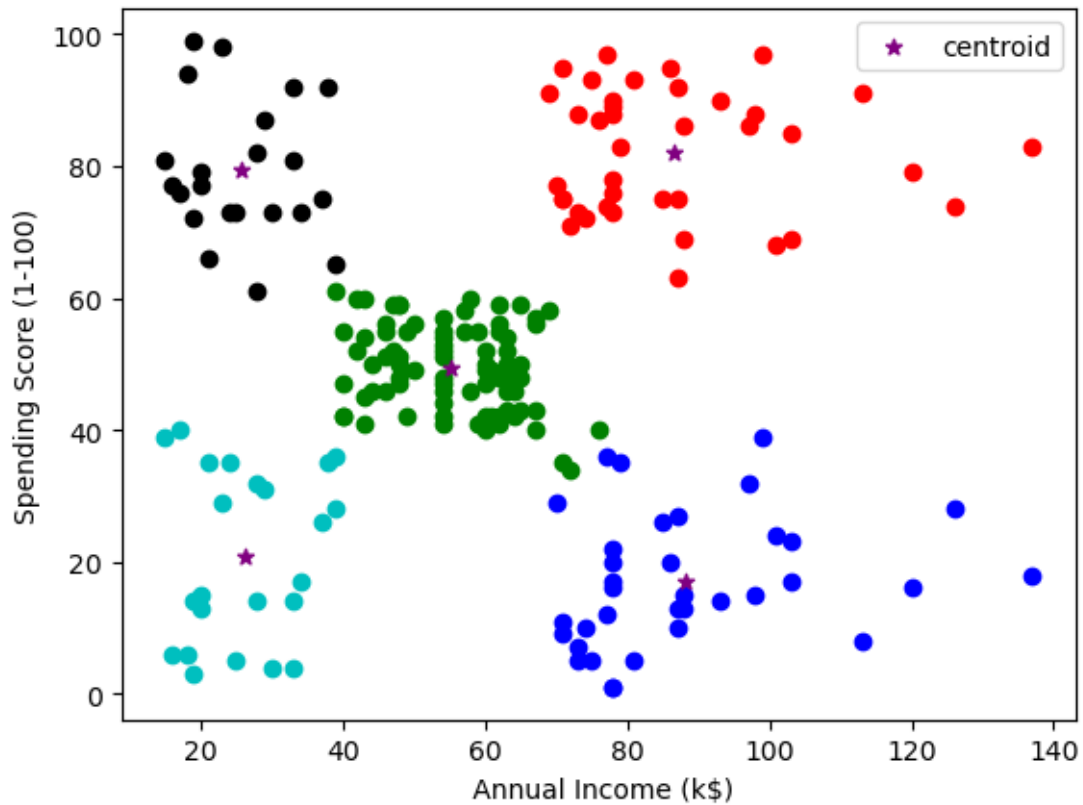
```

```
[25]: km.cluster_centers_
```

```
[25]: array([[55.2962963 , 49.51851852],
            [86.53846154, 82.12820513],
            [25.72727273, 79.36363636],
            [88.2       , 17.11428571],
            [26.30434783, 20.91304348]])
```

```
[26]: df1 = df[df.cluster==0]
df2 = df[df.cluster==1]
df3 = df[df.cluster==2]
df4 = df[df.cluster==3]
df5 = df[df.cluster==4]
plt.scatter(df1['Annual Income (k$)'],df1['Spending Score_
↪(1-100)'],color='green')
plt.scatter(df2['Annual Income (k$)'],df2['Spending Score (1-100)'],color='red')
plt.scatter(df3['Annual Income (k$)'],df3['Spending Score_
↪(1-100)'],color='black')
plt.scatter(df4['Annual Income (k$)'],df4['Spending Score_
↪(1-100)'],color='blue')
plt.scatter(df5['Annual Income (k$)'],df5['Spending Score (1-100)'],color='c')
plt.scatter(km.cluster_centers_[ :,0],km.cluster_centers_[ :
↪,1],color='purple',marker='*',label='centroid')
plt.xlabel(' Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.legend()
```

```
[26]: <matplotlib.legend.Legend at 0x15b42f0e1c0>
```



1.5 Model Interpretation

- 1.5.1 Cluster 1 which is in blue color is a group of customers earning high but spending less
- 1.5.2 cluster 2 which is green Color is a group with average in terms of earning and spending
- 1.5.3 cluster 3 red color is earning high and also spending high [TARGET SET]
- 1.5.4 cluster 4 black color earning less but spending more
- 1.5.5 Cluster 5 cyan color Earning less , spending less

[]: