ASSESMENT -5

21BCE0516

ANUSHKA

anushka.2021a@vitstudent.ac.in (mailto:anushka.2021a@vitstudent.ac.in)

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

1. Understanding the Data

In [2]:
```
df = pd.read_csv('Mall_Customers.csv')
df.head()
```

Out[2]:

	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

In [3]:
```
df.shape
```

Out[3]:
```
(200, 5)
```

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):
 #   Column                  Non-Null Count  Dtype 
---  ------                  --------------  ----- 
 0   CustomerID              200 non-null    int64 
 1   Gender                  200 non-null    object
 2   Age                     200 non-null    int64 
 3   Annual Income (k$)      200 non-null    int64 
 4   Spending Score (1-100)  200 non-null    int64 
dtypes: int64(4), object(1)
memory usage: 7.9+ KB
```

In [5]:
```
df.isnull().sum()
```

Out[5]:
```
CustomerID                0
Gender                    0
Age                       0
Annual Income (k$)        0
Spending Score (1-100)    0
dtype: int64
```

In [6]:
```
df.describe()
```

Out[6]:

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
count	200.000000	200.000000	200.000000	200.000000
mean	100.500000	38.850000	60.560000	50.200000
std	57.879185	13.969007	26.264721	25.823522
min	1.000000	18.000000	15.000000	1.000000
25%	50.750000	28.750000	41.500000	34.750000
50%	100.500000	36.000000	61.500000	50.000000
75%	150.250000	49.000000	78.000000	73.000000
max	200.000000	70.000000	137.000000	99.000000

2. Data Preprocessing

In [7]:
```
from sklearn import cluster
```

In [8]:
```
new_df = df.iloc[:,-2:]
new_df.head()
```

Out[8]:

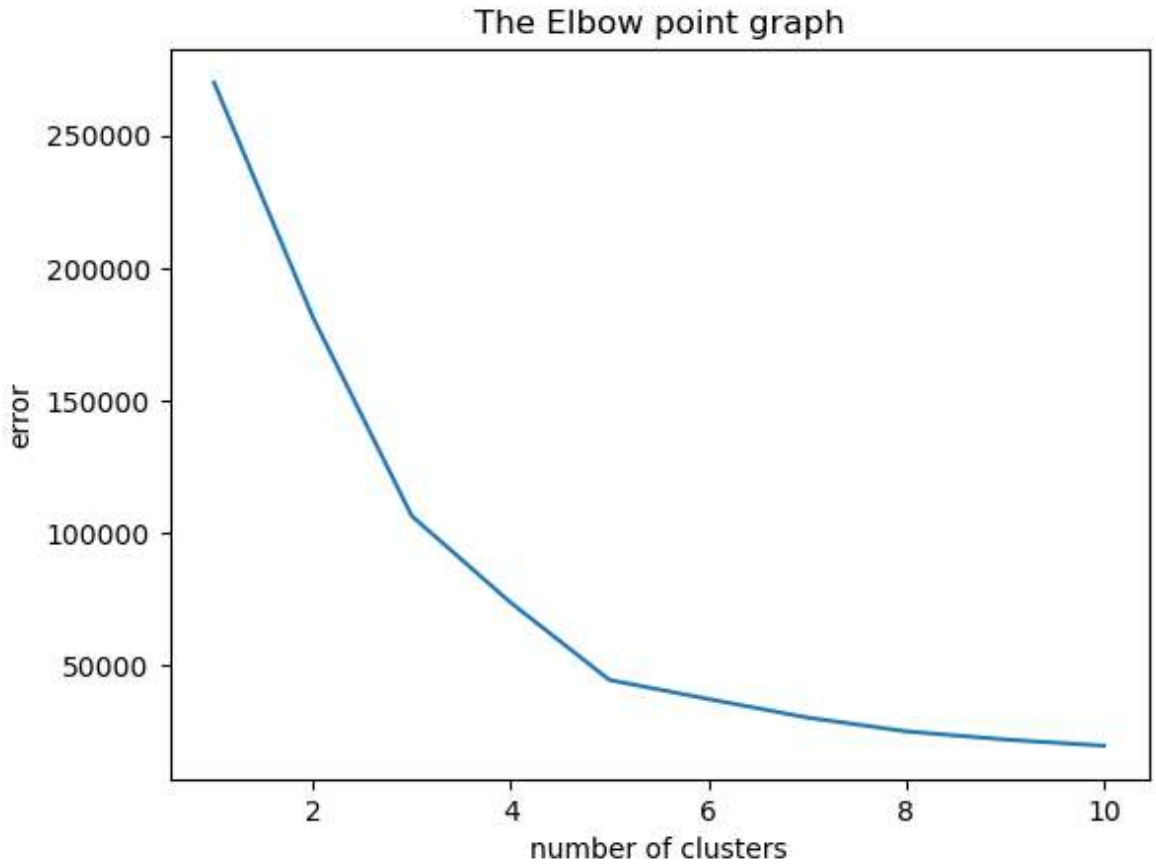
	Annual Income (k$)	Spending Score (1-100)
0	15	39
1	15	81
2	16	6
3	16	77
4	17	40

In [9]:
```
error=[]
for i in range(1,11):
  kmeans = cluster.KMeans(n_clusters=i,init = 'k-means++',random_state=4)
  kmeans.fit(new_df)
  error.append(kmeans.inertia_)
```

```
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:870: 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
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:1382: UserWarning: KMeans is known to have a memory leak on Windows with
MKL, when there are less chunks than available threads. You can avoid it by
setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:870: 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
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:1382: UserWarning: KMeans is known to have a memory leak on Windows with
MKL, when there are less chunks than available threads. You can avoid it by
setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:870: 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
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:1382: UserWarning: KMeans is known to have a memory leak on Windows with
MKL, when there are less chunks than available threads. You can avoid it by
setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:870: 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
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:1382: UserWarning: KMeans is known to have a memory leak on Windows with
MKL, when there are less chunks than available threads. You can avoid it by
setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:870: 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
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:1382: UserWarning: KMeans is known to have a memory leak on Windows with
MKL, when there are less chunks than available threads. You can avoid it by
setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:870: 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
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:1382: UserWarning: KMeans is known to have a memory leak on Windows with
MKL, when there are less chunks than available threads. You can avoid it by
setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:870: 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
```

```
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:1382: UserWarning: KMeans is known to have a memory leak on Windows with
MKL, when there are less chunks than available threads. You can avoid it by
setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:870: 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
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:1382: UserWarning: KMeans is known to have a memory leak on Windows with
MKL, when there are less chunks than available threads. You can avoid it by
setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:870: 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
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:1382: UserWarning: KMeans is known to have a memory leak on Windows with
MKL, when there are less chunks than available threads. You can avoid it by
setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:870: 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
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:1382: UserWarning: KMeans is known to have a memory leak on Windows with
MKL, when there are less chunks than available threads. You can avoid it by
setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
```

In [10]:
```
plt.plot(range(1,11),error)
plt.title('The Elbow point graph')
plt.xlabel('number of clusters')
plt.ylabel('error')
plt.show()
```



3. Machine Learning approach with Clustering Algorithm

In [12]:
```
km_model = cluster.KMeans(n_clusters=5,init = 'k-means++',random_state=0)
km_model.fit(new_df)
```

```
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:870: 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
  warnings.warn(
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\cluster\_kmeans.
py:1382: UserWarning: KMeans is known to have a memory leak on Windows with
MKL, when there are less chunks than available threads. You can avoid it by
setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
```

Out[12]:
```
KMeans
KMeans(n_clusters=5, random_state=0)
```

In [13]:
```
pred = km_model.predict(new_df)
pred
```

Out[13]:
```
array([4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3,
       4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 1,
       4, 3, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 0, 2, 1, 2, 0, 2, 0, 2,
       1, 2, 0, 2, 0, 2, 0, 2, 0, 2, 1, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2,
       0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2,
       0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2,
       0, 2])
```

In [14]:
```
# Testing the model with random observation
km_model.predict([[60,50]])
```

```
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\base.py:439: Use
rWarning: X does not have valid feature names, but KMeans was fitted with fe
ature names
  warnings.warn(
```

Out[14]:
```
array([1])
```

In [15]:
```
km_model.predict([[15,1]])
```

```
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\base.py:439: Use
rWarning: X does not have valid feature names, but KMeans was fitted with fe
ature names
  warnings.warn(
```

Out[15]:
```
array([4])
```

In [17]:
```
km_model.predict([[41,34]])
```

```
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\base.py:439: Use
rWarning: X does not have valid feature names, but KMeans was fitted with fe
ature names
  warnings.warn(
```

Out[17]:
```
array([4])
```

In [18]:
```
km_model.predict([[137,99]])
```

```
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\base.py:439: Use
rWarning: X does not have valid feature names, but KMeans was fitted with fe
ature names
  warnings.warn(
```

Out[18]:
```
array([2])
```

In [19]:
```
km_model.predict([[78,73]])
```

```
C:\Users\dell\Desktop\mincon1\env\Lib\site-packages\sklearn\base.py:439: UserWarning: X does not have valid feature names, but KMeans was fitted with feature names
  warnings.warn(
```

Out[19]:
```
array([2])
```

In [ ]:
```

```