```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
data=pd.read_csv("/content/Mall_Customers.csv")
data.head()
```

```
   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
```

```
df=pd.DataFrame(data)
df.describe()
```

```
       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
```

```
df=df.drop(["CustomerID"],axis=1)
df.isnull()
```

```
     Gender    Age  Annual Income (k$)  Spending Score (1-100)
0     False  False               False                   False
1     False  False               False                   False
2     False  False               False                   False
3     False  False               False                   False
4     False  False               False                   False
..      ...    ...                 ...                     ...
195   False  False               False                   False
196   False  False               False                   False
197   False  False               False                   False
198   False  False               False                   False
```

```
199   False  False              False                  False

[200 rows x 4 columns]
```

```
df.isna().sum()
```

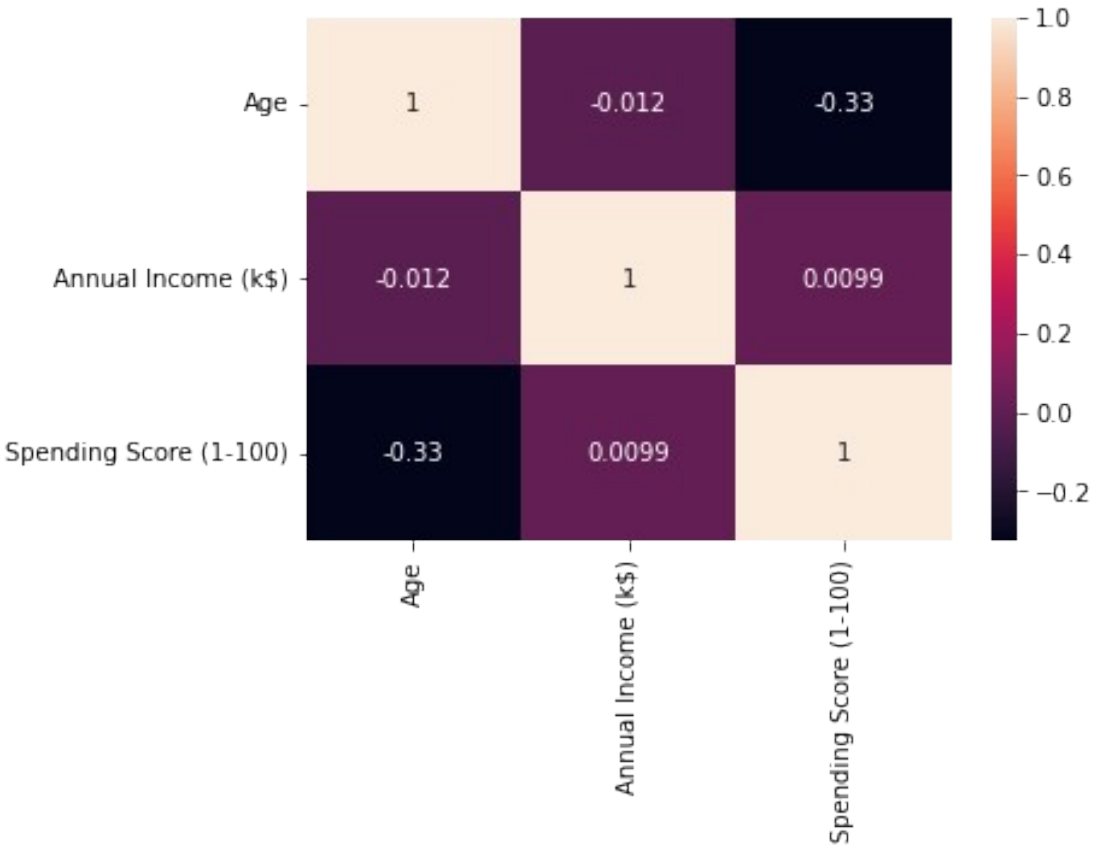
```
Gender                    0
Age                       0
Annual Income (k$)        0
Spending Score (1-100)    0
dtype: int64
```

```
df.corr()
```

```
                            Age  Annual Income (k$)  Spending Score
(1-100)
Age                    1.000000           -0.012398               -
0.327227
Annual Income (k$)     -0.012398            1.000000
0.009903
Spending Score (1-100) -0.327227            0.009903
1.000000
```

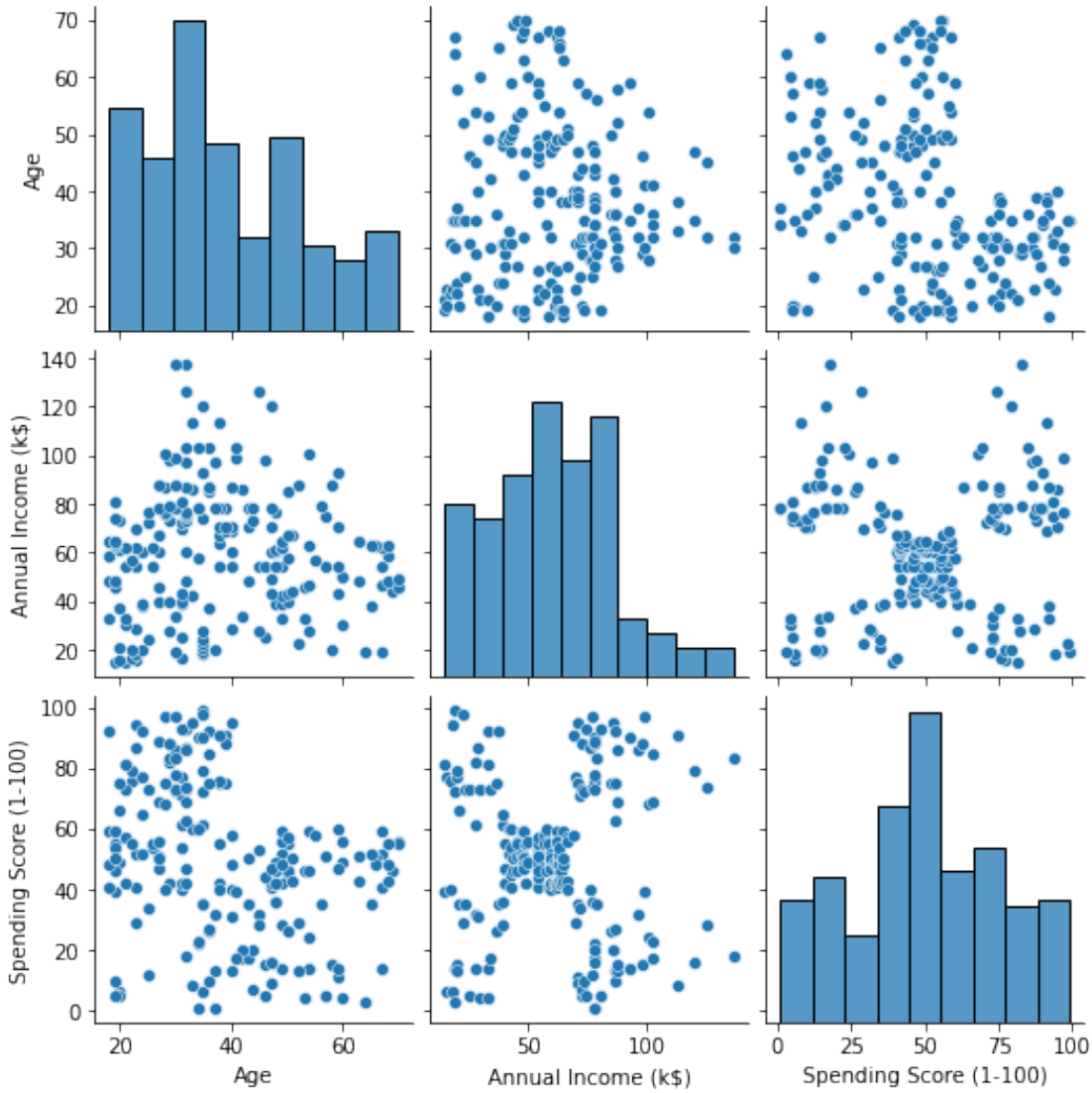
```
sns.heatmap(df.corr(),annot=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fbed436c590>
```

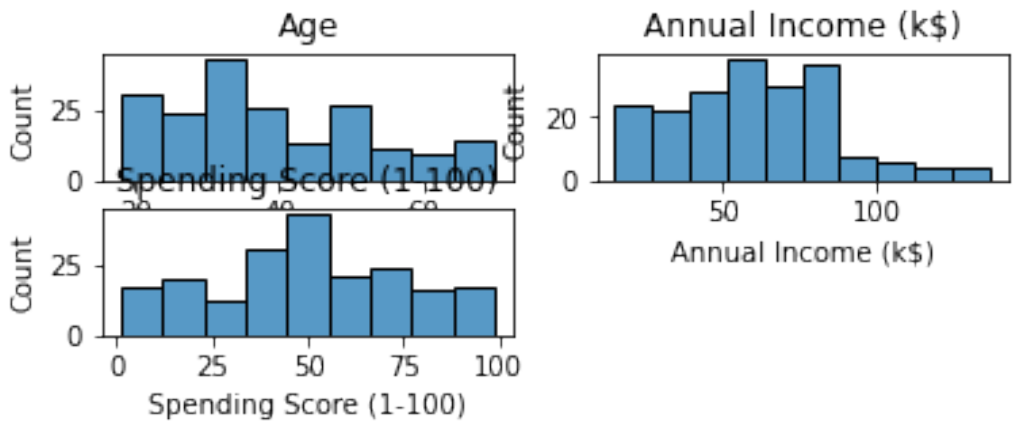



```
#Mivariate Analysis
sns.pairplot(df)
```

```
<seaborn.axisgrid.PairGrid at 0x7fbed1a3a790>
```



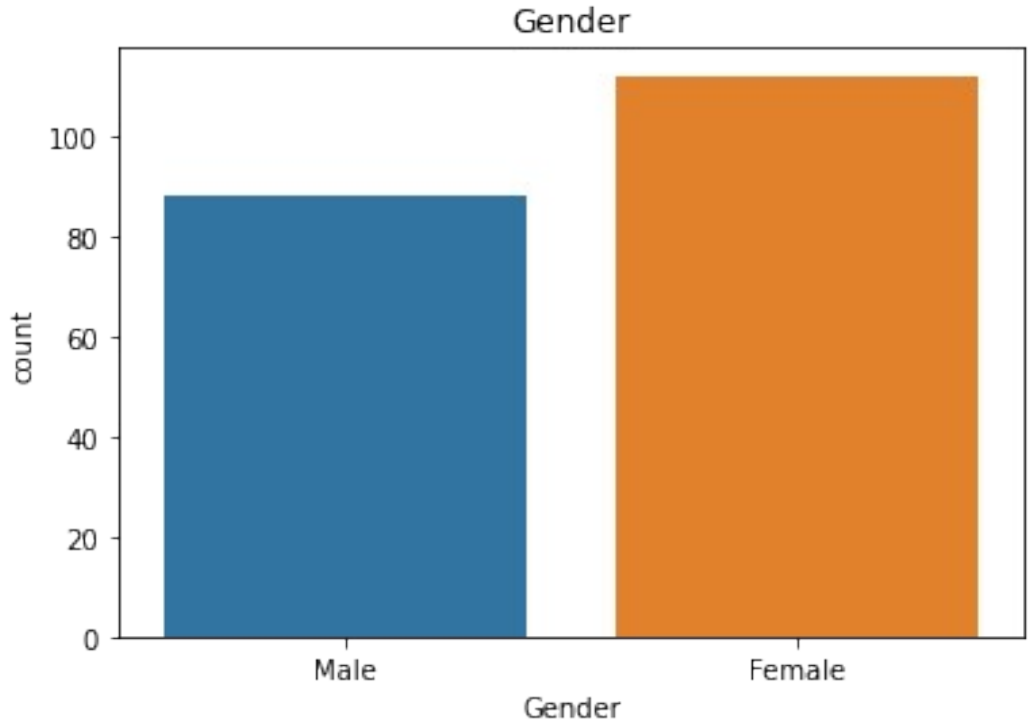

```
#Univariate Analysis of Continous Variables
l=list(df.columns)
l1=l[1:]
for i in range(len(l1)):
  plt.subplot(4,2,i+1)
  sns.histplot(df[l1[i]])
  plt.title(f'{l1[i]}')
```




```
#Univariate Analysis of Categorical Values
sns.countplot(df['Gender'])
plt.title('Gender')
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43:
FutureWarning: Pass the following variable as a keyword arg: x. From
version 0.12, the only valid positional argument will be `data`, and
passing other arguments without an explicit keyword will result in an
error or misinterpretation.
  FutureWarning
```

```
Text(0.5, 1.0, 'Gender')
```



```
#Bivariate Analysis of Continous Variables
df[list(df.columns[2:])].corr()
```

	Annual Income (k$)	Spending Score (1-100)
Annual Income (k$)	1.000000	0.009903
Spending Score (1-100)	0.009903	1.000000

```
#Label Encoding For Categorical Values
from sklearn import preprocessing
label_encoder=preprocessing.LabelEncoder()
df['Gender']=label_encoder.fit_transform(df['Gender'])
df.head()
```

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

```
#Outiers Identification And Replacing Using Quantile-Based Flooring
and Capping for Continuous Variables

q1=df['Age'].quantile(0.10)
q2=df['Age'].quantile(0.90)
df['Age']=np.where(df['Age']<q1,q1,df['Age'])
df['Age']=np.where(df['Age']>q2,q2,df['Age'])
print(df['Age'].skew())

q1=df['Annual Income (k$)'].quantile(0.10)
q2=df['Annual Income (k$)'].quantile(0.90)
df['Annual Income (k$)']=np.where(df['Annual Income
(k$)']<q1,q1,df['Annual Income (k$)'])
df['Annual Income (k$)']=np.where(df['Annual Income
(k$)']>q2,q2,df['Annual Income (k$)'])
print(df['Annual Income (k$)'].skew())

q1=df['Spending Score (1-100)'].quantile(0.10)
q2=df['Spending Score (1-100)'].quantile(0.90)
df['Spending Score (1-100)']=np.where(df['Spending Score (1-
100)']<q1,q1,df['Spending Score (1-100)'])
df['Spending Score (1-100)']=np.where(df['Spending Score (1-
100)']>q2,q2,df['Spending Score (1-100)'])
print(df['Spending Score (1-100)'].skew())
```

```
0.281242452772514
-0.11114517936178386
-0.03744215109217243
```

```
df
```

```
     Gender   Age  Annual Income (k$)  Spending Score (1-100)
0         1  21.0                23.9                    39.0
1         1  21.0                23.9                    81.0
2         0  21.0                23.9                    13.0
3         0  23.0                23.9                    77.0
4         0  31.0                23.9                    40.0
..      ...   ...                 ...                     ...
195       0  35.0                93.4                    79.0
196       0  45.0                93.4                    28.0
197       1  32.0                93.4                    74.0
198       1  32.0                93.4                    18.0
199       1  30.0                93.4                    83.0

[200 rows x 4 columns]
```

```
#Scaling Variables
import pandas as pd
from sklearn.preprocessing import StandardScaler
scaler=StandardScaler()
scaled=scaler.fit_transform(df)
print(scaled)
```

```
[[ 1.12815215 -1.39030724 -1.599128   -0.47229142]
 [ 1.12815215 -1.39030724 -1.599128    1.28936397]
 [-0.88640526 -1.39030724 -1.599128   -1.56284   ]
 [-0.88640526 -1.2301336  -1.599128    1.12158726]
 [-0.88640526 -0.58943902 -1.599128   -0.43034725]
 [-0.88640526 -1.31022042 -1.599128    1.07964309]
 [-0.88640526 -0.26909172 -1.599128   -1.56284   ]
 [-0.88640526 -1.2301336  -1.599128    1.54522344]
 [ 1.12815215  1.6610007  -1.599128   -1.56284   ]
 [-0.88640526 -0.66952584 -1.599128    0.91186639]
 [ 1.12815215  1.6610007  -1.599128   -1.52089582]
 [-0.88640526 -0.26909172 -1.599128    1.54522344]
 [-0.88640526  1.5729052  -1.599128   -1.47895164]
 [-0.88640526 -1.15004677 -1.599128    1.12158726]
 [ 1.12815215 -0.10891808 -1.599128   -1.56284   ]
 [ 1.12815215 -1.31022042 -1.599128    1.20547562]
 [-0.88640526 -0.26909172 -1.599128   -0.64006812]
 [ 1.12815215 -1.39030724 -1.599128    0.66020133]
 [ 1.12815215  1.09238426 -1.599128   -0.89173318]
 [-0.88640526 -0.26909172 -1.599128    1.54522344]
 [ 1.12815215 -0.26909172 -1.59462848 -0.64006812]
 [ 1.12815215 -1.06995995 -1.59462848  0.95381056]
 [-0.88640526  0.61186333 -1.54963332 -1.56284   ]
 [ 1.12815215 -0.58943902 -1.54963332  0.95381056]
 [-0.88640526  1.25255791 -1.41464784 -1.52089582]
 [ 1.12815215 -0.74961266 -1.41464784  1.33130814]
 [-0.88640526  0.5317765  -1.41464784 -0.76590065]
 [ 1.12815215 -0.26909172 -1.41464784  0.45048045]
```

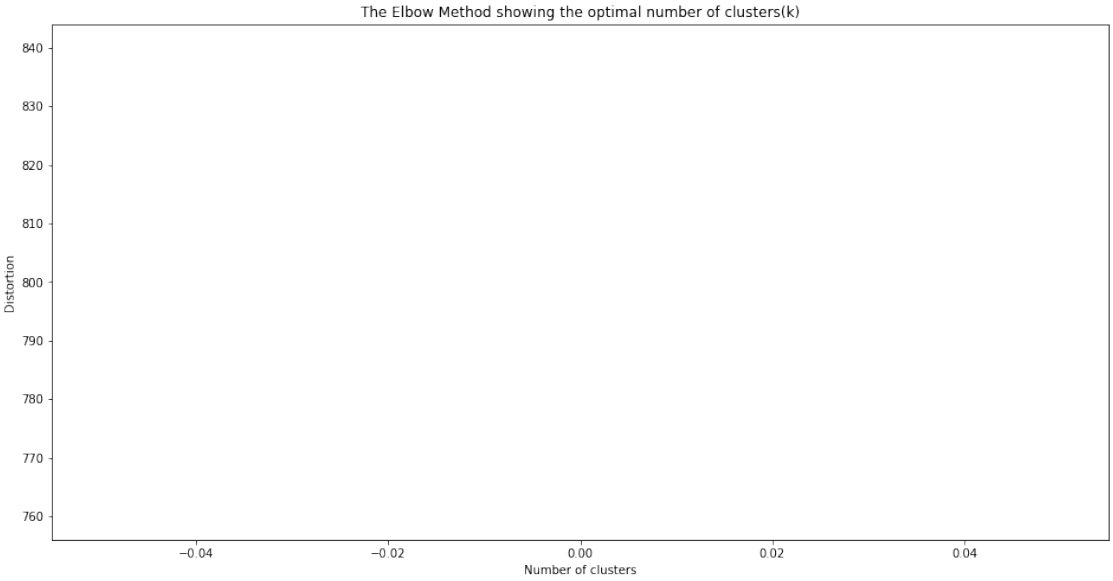
```
[-0.88640526  0.13134239 -1.36965268 -0.80784483]
[-0.88640526 -1.2301336  -1.36965268  1.54102902]
[ 1.12815215  1.6610007  -1.32465752 -1.56284   ]
[-0.88640526 -1.39030724 -1.32465752  0.95381056]
[ 1.12815215  1.17247108 -1.18967204 -1.56284   ]
[ 1.12815215 -1.39030724 -1.18967204  1.54522344]
[-0.88640526  0.85212379 -1.18967204 -1.52089582]
[-0.88640526 -1.39030724 -1.18967204  1.28936397]
[-0.88640526  0.29151603 -1.14467688 -1.39506329]
[-0.88640526 -0.66952584 -1.14467688  0.95381056]
[-0.88640526 -0.1890049  -1.0096914  -1.01756571]
[-0.88640526 -1.39030724 -1.0096914   1.03769891]
[-0.88640526  1.6610007  -0.96469624 -0.64006812]
[ 1.12815215 -1.15004677 -0.96469624  1.54522344]
[ 1.12815215  0.77203697 -0.91970108 -0.59812395]
[-0.88640526 -0.58943902 -0.91970108  0.45048045]
[-0.88640526  0.85212379 -0.91970108 -0.93367736]
[-0.88640526 -1.15004677 -0.91970108  0.61825715]
[-0.88640526  0.93221062 -0.87470592  0.19881539]
[-0.88640526 -0.90978631 -0.87470592 -0.13673801]
[-0.88640526 -0.74961266 -0.87470592 -0.34645889]
[-0.88640526 -0.58943902 -0.87470592 -0.34645889]
[-0.88640526  0.85212379 -0.78471559  0.07298287]
[ 1.12815215 -0.42926537 -0.78471559  0.40853627]
[-0.88640526 -0.58943902 -0.73972043  0.15687122]
[ 1.12815215  1.65299202 -0.73972043  0.40853627]
[-0.88640526  0.93221062 -0.73972043 -0.22062637]
[ 1.12815215  0.69195015 -0.73972043 -0.38840307]
[-0.88640526  1.01229744 -0.69472527 -0.01090549]
[ 1.12815215  1.6610007  -0.69472527 -0.17868219]
[-0.88640526 -0.90978631 -0.60473495  0.03103869]
[ 1.12815215  1.17247108 -0.60473495 -0.17868219]
[ 1.12815215  1.6610007  -0.60473495  0.24075957]
[ 1.12815215 -1.39030724 -0.60473495  0.19881539]
[-0.88640526  1.6610007  -0.55973979  0.07298287]
[-0.88640526  1.25255791 -0.55973979  0.3665921 ]
[ 1.12815215  1.6610007  -0.51474463  0.03103869]
[ 1.12815215 -1.39030724 -0.51474463  0.3665921 ]
[-0.88640526  0.37160286 -0.51474463 -0.01090549]
[-0.88640526  1.6610007  -0.51474463 -0.09479384]
[ 1.12815215 -1.39030724 -0.51474463  0.3665921 ]
[-0.88640526 -0.50935219 -0.51474463 -0.13673801]
[ 1.12815215  1.6610007  -0.46974947  0.19881539]
[-0.88640526  0.69195015 -0.46974947 -0.34645889]
[-0.88640526  1.6610007  -0.42475431 -0.05284966]
[-0.88640526  1.6610007  -0.42475431  0.24075957]
[ 1.12815215  1.65299202 -0.24477367 -0.13673801]
[ 1.12815215 -0.98987313 -0.24477367  0.15687122]
[-0.88640526  0.5317765  -0.24477367  0.11492704]
[ 1.12815215  0.13134239 -0.24477367 -0.09479384]
```

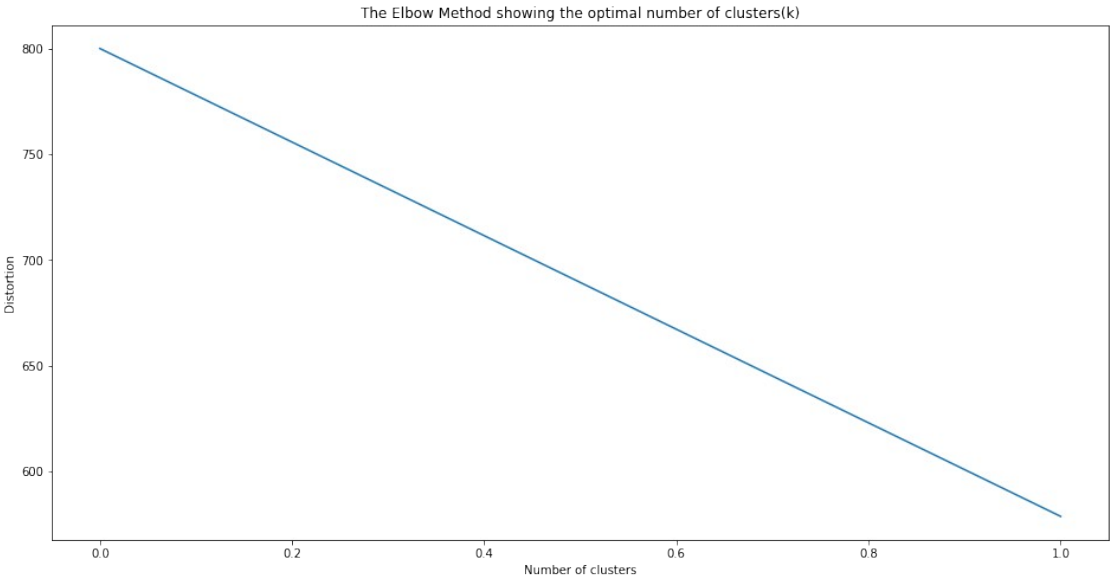
```
[-0.88640526 -1.2301336  -0.24477367  0.07298287]
[-0.88640526  0.85212379 -0.24477367 -0.34645889]
[ 1.12815215  1.49281838 -0.24477367  0.03103869]
[ 1.12815215 -0.02883126 -0.24477367  0.19881539]
[ 1.12815215  1.6610007  -0.24477367 -0.38840307]
[-0.88640526  0.61186333 -0.24477367 -0.26257054]
[-0.88640526 -1.39030724 -0.24477367  0.28270375]
[ 1.12815215  0.77203697 -0.24477367 -0.17868219]
[-0.88640526  1.33264473 -0.10978819  0.32464792]
[-0.88640526 -1.31022042 -0.10978819  0.19881539]
[-0.88640526 -0.34917855 -0.06479303  0.40853627]
[-0.88640526  0.93221062 -0.06479303 -0.17868219]
[-0.88640526  1.6610007  -0.01979787  0.19881539]
[ 1.12815215 -1.39030724 -0.01979787 -0.38840307]
[ 1.12815215  0.77203697  0.02519729 -0.05284966]
[-0.88640526  0.13134239  0.02519729 -0.43034725]
[-0.88640526 -0.50935219  0.02519729 -0.34645889]
[ 1.12815215 -1.15004677  0.02519729  0.07298287]
[-0.88640526  0.69195015  0.02519729 -0.13673801]
[-0.88640526 -0.90978631  0.02519729 -0.01090549]
[ 1.12815215  0.77203697  0.07019245 -0.34645889]
[ 1.12815215 -1.39030724  0.07019245 -0.05284966]
[-0.88640526 -1.2301336   0.11518761 -0.38840307]
[-0.88640526  0.85212379  0.11518761 -0.09479384]
[ 1.12815215  1.6610007   0.11518761  0.3665921 ]
[ 1.12815215 -0.98987313  0.11518761  0.19881539]
[ 1.12815215  0.85212379  0.11518761  0.24075957]
[-0.88640526 -1.39030724  0.11518761 -0.34645889]
[-0.88640526  1.6610007   0.16018277 -0.01090549]
[ 1.12815215  1.25255791  0.16018277 -0.17868219]
[ 1.12815215  1.6610007   0.16018277 -0.30451472]
[ 1.12815215  1.6610007   0.16018277 -0.09479384]
[ 1.12815215  1.6610007   0.16018277  0.07298287]
[-0.88640526 -1.39030724  0.16018277  0.15687122]
[-0.88640526 -0.02883126  0.20517793 -0.34645889]
[ 1.12815215 -1.39030724  0.20517793 -0.17868219]
[-0.88640526 -1.39030724  0.25017309 -0.09479384]
[-0.88640526 -1.39030724  0.25017309 -0.01090549]
[-0.88640526  1.6610007   0.25017309 -0.30451472]
[-0.88640526  0.85212379  0.25017309  0.3665921 ]
[-0.88640526  1.01229744  0.34016341 -0.30451472]
[-0.88640526  0.93221062  0.34016341  0.28270375]
[ 1.12815215 -0.90978631  0.34016341  0.24075957]
[-0.88640526 -0.02883126  0.34016341 -0.43034725]
[-0.88640526  0.13134239  0.43015373  0.32464792]
[ 1.12815215  0.05125557  0.43015373  1.54522344]
[-0.88640526 -1.2301336   0.47514889 -0.89173318]
[-0.88640526 -0.58943902  0.47514889  1.12158726]
[ 1.12815215  0.37160286  0.52014405 -0.64006812]
[ 1.12815215  0.13134239  0.52014405  1.54522344]
```

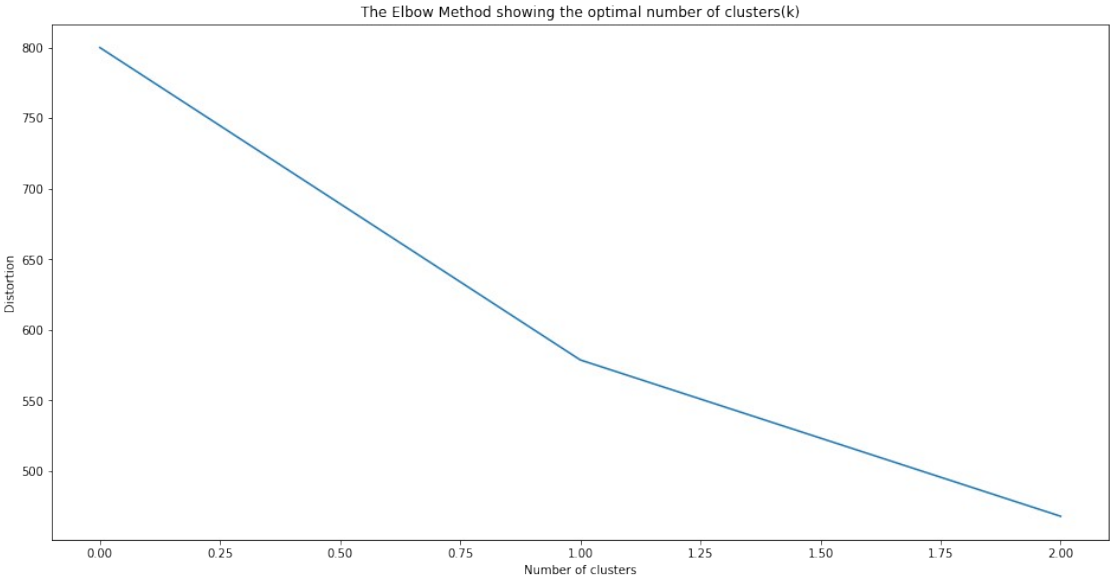
```
[ 1.12815215  1.65299202  0.52014405 -1.56284   ]
[ 1.12815215 -0.02883126  0.52014405  1.03769891]
[ 1.12815215  0.69195015  0.52014405 -1.56284   ]
[ 1.12815215  0.05125557  0.52014405  1.03769891]
[-0.88640526 -1.06995995  0.56513921 -0.6820123 ]
[-0.88640526 -0.58943902  0.56513921  0.86992221]
[ 1.12815215 -1.39030724  0.61013437 -1.56284   ]
[-0.88640526 -0.74961266  0.61013437  1.54522344]
[-0.88640526  0.45168968  0.61013437 -1.56284   ]
[ 1.12815215 -0.50935219  0.61013437  0.95381056]
[ 1.12815215 -1.39030724  0.65512953 -1.56284   ]
[-0.88640526 -0.26909172  0.65512953  0.91186639]
[-0.88640526  1.49281838  0.70012469 -1.56284   ]
[ 1.12815215 -0.50935219  0.70012469  1.54522344]
[-0.88640526 -0.82969948  0.74511985 -0.43034725]
[-0.88640526 -0.50935219  0.74511985  1.54102902]
[ 1.12815215 -1.06995995  0.79011501 -1.56284   ]
[ 1.12815215 -0.82969948  0.79011501  1.54522344]
[ 1.12815215  0.77203697  0.79011501 -0.59812395]
[-0.88640526 -0.50935219  0.79011501  0.99575474]
[-0.88640526 -0.34917855  0.83511017 -1.18534241]
[ 1.12815215 -0.34917855  0.83511017  1.54522344]
[ 1.12815215  0.37160286  0.83511017 -1.39506329]
[ 1.12815215  0.05125557  0.83511017  1.54522344]
[-0.88640526  0.45168968  0.83511017 -1.26923076]
[-0.88640526 -0.02883126  0.83511017  1.07964309]
[-0.88640526  0.69195015  0.83511017 -1.43700747]
[-0.88640526 -0.90978631  0.83511017  1.54522344]
[ 1.12815215 -0.10891808  0.83511017 -1.56284   ]
[-0.88640526 -0.66952584  0.83511017  1.16353144]
[ 1.12815215 -0.34917855  0.83511017 -1.56284   ]
[-0.88640526 -0.66952584  0.83511017  0.95381056]
[-0.88640526  1.41273155  0.88010533 -0.64006812]
[-0.88640526 -0.74961266  0.88010533  1.37325232]
[ 1.12815215 -1.39030724  0.97009565 -1.56284   ]
[-0.88640526 -0.58943902  0.97009565  1.54522344]
[ 1.12815215  0.93221062  1.1500763  -1.01756571]
[-0.88640526 -0.1890049   1.1500763   1.03769891]
[ 1.12815215  0.29151603  1.19507146 -1.26923076]
[-0.88640526 -0.42926537  1.19507146  1.54522344]
[-0.88640526 -0.1890049   1.24006662 -0.97562153]
[ 1.12815215 -0.50935219  1.24006662  0.5343688 ]
[ 1.12815215  0.13134239  1.24006662 -1.56284   ]
[ 1.12815215 -0.82969948  1.24006662  1.03769891]
[ 1.12815215 -0.1890049   1.24006662 -1.56284   ]
[ 1.12815215 -0.1890049   1.24006662  1.54522344]
[-0.88640526  1.09238426  1.28506178 -1.56284   ]
[-0.88640526 -0.66952584  1.28506178  1.49908485]
[ 1.12815215  1.5729052   1.28506178 -1.47895164]
[ 1.12815215 -0.90978631  1.28506178  0.78603386]
```

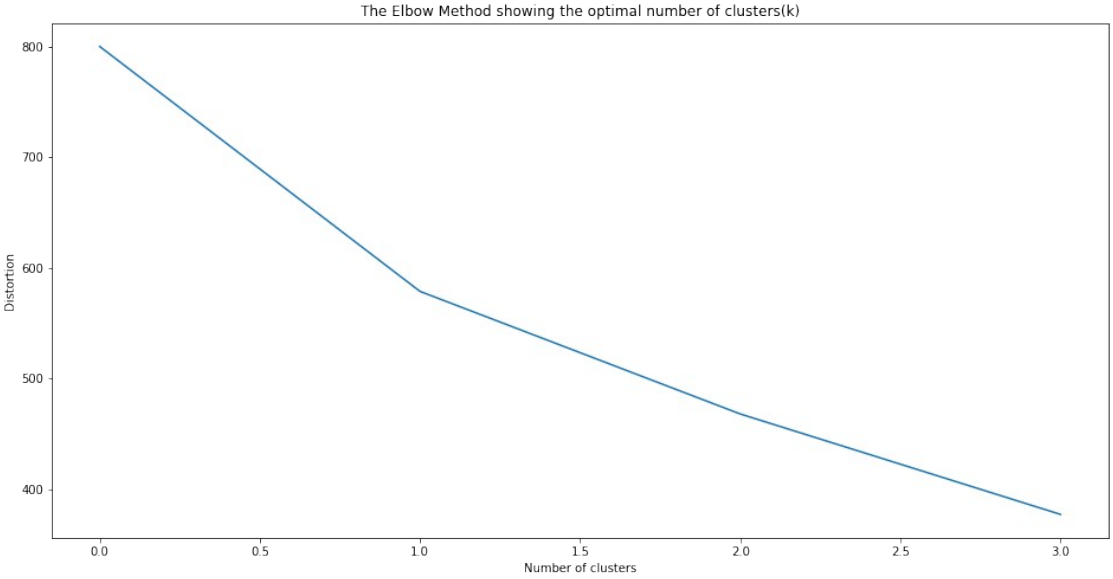
```
 [ 1.12815215  1.65299202  1.51003758 -1.52089582]
 [ 1.12815215 -0.26909172  1.51003758  1.54522344]
 [-0.88640526 -0.10891808  1.52803564 -0.76590065]
 [-0.88640526 -0.50935219  1.52803564  1.49908485]
 [ 1.12815215  0.61186333  1.52803564 -1.47895164]
 [-0.88640526 -0.74961266  1.52803564  1.54522344]
 [-0.88640526  0.21142921  1.52803564 -0.47229142]
 [ 1.12815215 -0.66952584  1.52803564  1.54522344]
 [-0.88640526  1.25255791  1.52803564 -1.10145406]
 [ 1.12815215 -0.82969948  1.52803564  0.74408968]
 [-0.88640526  0.21142921  1.52803564 -1.39506329]
 [-0.88640526 -0.1890049   1.52803564  1.45714067]
 [-0.88640526 -0.34917855  1.52803564 -1.14339824]
 [-0.88640526 -0.50935219  1.52803564  0.78603386]
 [ 1.12815215 -0.42926537  1.52803564 -1.56284   ]
 [-0.88640526 -0.02883126  1.52803564  1.54522344]
 [-0.88640526  0.69195015  1.52803564 -1.43700747]
 [-0.88640526 -0.26909172  1.52803564  1.20547562]
 [-0.88640526  0.5317765   1.52803564 -0.93367736]
 [ 1.12815215 -0.50935219  1.52803564  0.99575474]
 [ 1.12815215 -0.50935219  1.52803564 -1.35311912]
 [ 1.12815215 -0.66952584  1.52803564  1.37325232]]
```

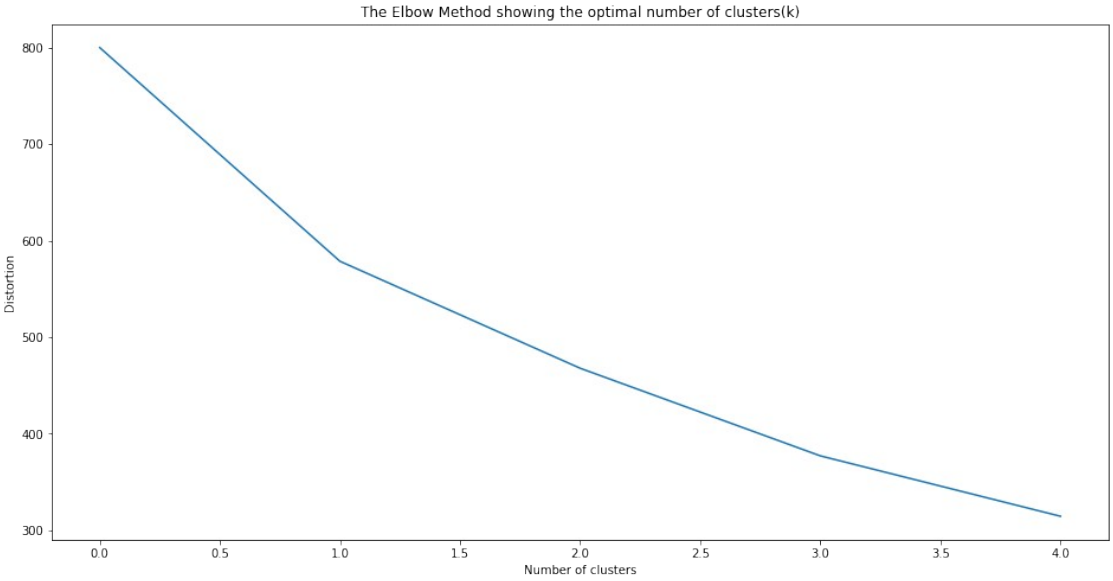
```
#Elbow Method To Find Number Of Clusters
w=[]
K=range(1,10)
for k in K:
  kmeanModel=KMeans(n_clusters=k)
  kmeanModel.fit(scaled)
  w.append(kmeanModel.inertia_)

  plt.figure(figsize=(16,8))
  plt.plot(w)
  plt.xlabel('Number of clusters')
  plt.ylabel('Distortion')
  plt.title('The Elbow Method showing the optimal number of
clusters(k)')

  plt.show()
```

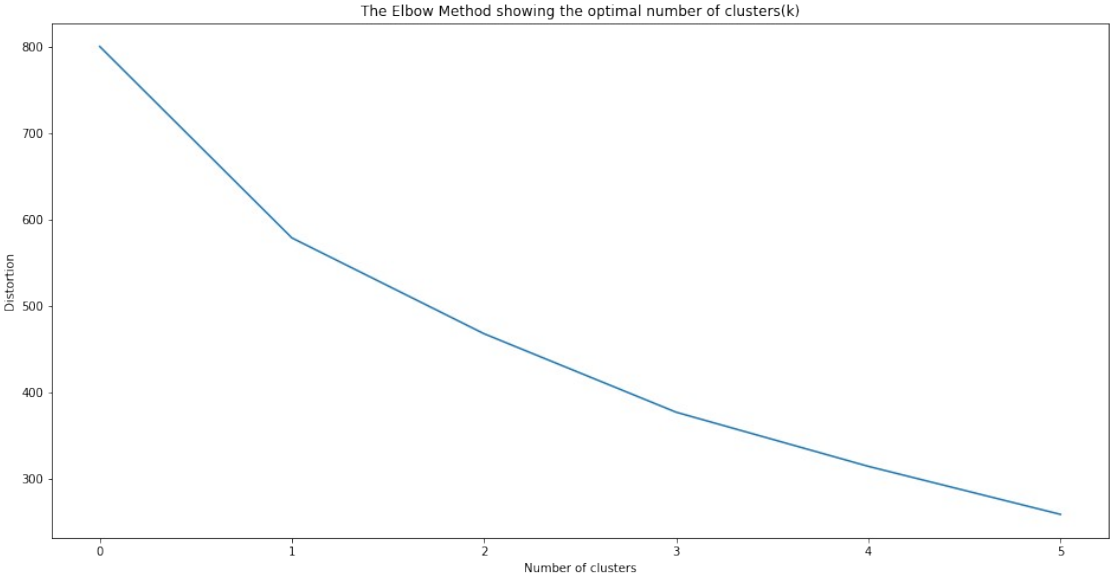

The Elbow Method showing the optimal number of clusters(k)
Distortion
840
830
820
810
800
790
780
770
760
−0.04
−0.02
0.00
0.02
0.04
Number of clusters

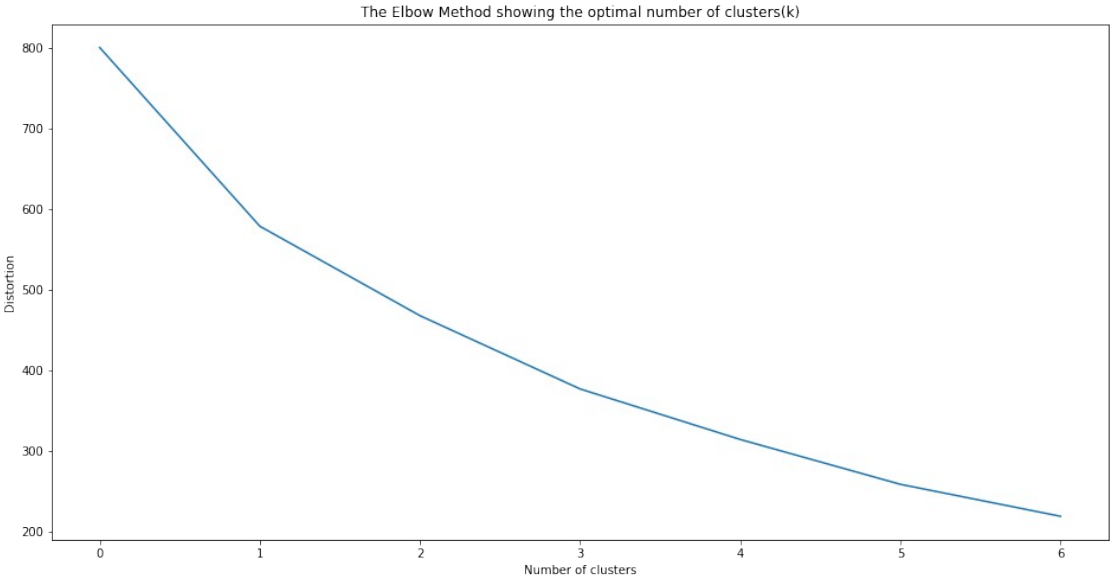

The Elbow Method showing the optimal number of clusters(k)
Distortion
800
750
700
650
600
0.0
0.2
0.4
0.6
0.8
1.0
Number of clusters

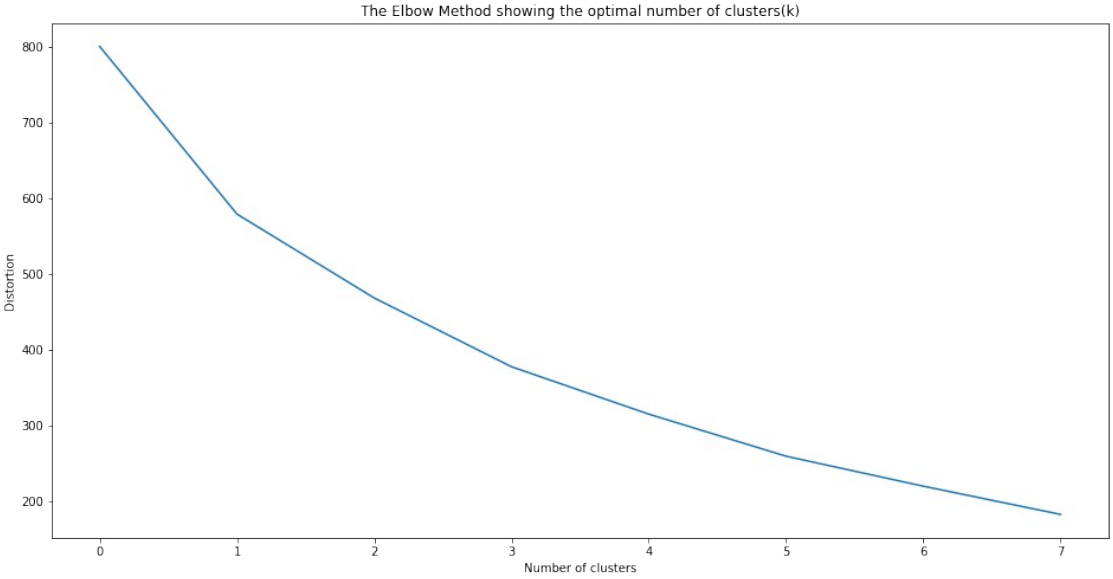

The Elbow Method showing the optimal number of clusters(k)
Distortion
800
750
700
650
600
550
500
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
Number of clusters

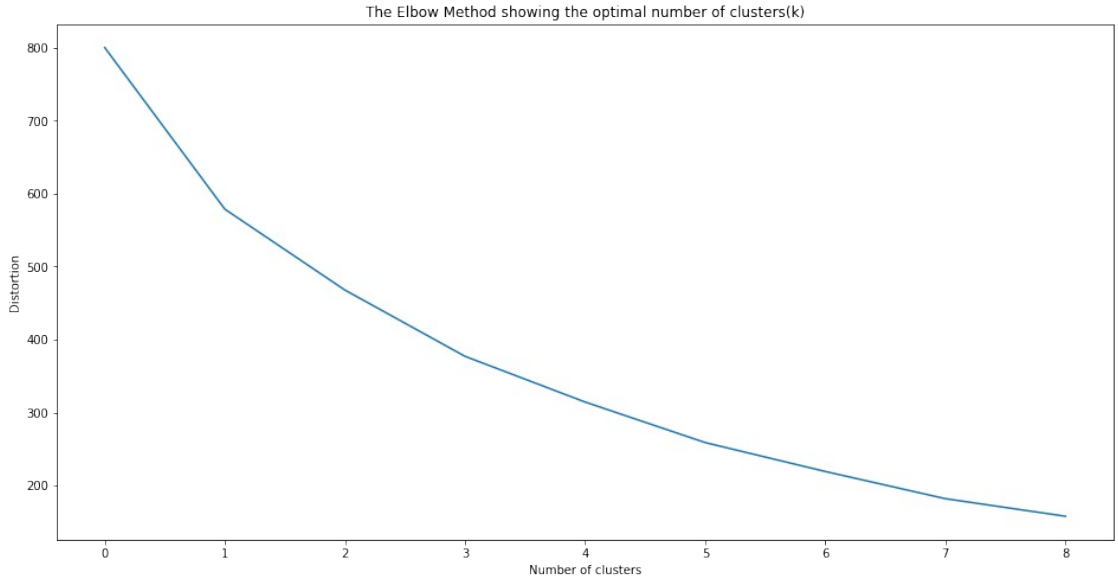

The Elbow Method showing the optimal number of clusters(k)
Distortion
800
700
600
500
400
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Number of clusters


The Elbow Method showing the optimal number of clusters(k)
Distortion
800
700
600
500
400
300
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Number of clusters


The Elbow Method showing the optimal number of clusters(k)
Distortion
800
700
600
500
400
300
0
1
2
3
4
5
Number of clusters


The Elbow Method showing the optimal number of clusters(k)
Distortion
800
700
600
500
400
300
200
0
1
2
3
4
5
6
Number of clusters


The Elbow Method showing the optimal number of clusters(k)
Distortion
800
700
600
500
400
300
200
0
1
2
3
4
5
6
7
Number of clusters




```
#Training the model-support vectom regression
from sklearn.cluster import KMeans
kmean=KMeans(n_clusters=3)
kmean.fit(scaled)
y_pred=kmean.predict(scaled)
df['pred_cluster']=y_pred
df.head
```

```
<bound method NDFrame.head of      Gender   Age  Annual Income (k$)
Spending Score (1-100)  pred_cluster
0         1  21.0                23.9                    39.0
1
1         1  21.0                23.9                    81.0
1
2         0  21.0                23.9                    13.0
0
3         0  23.0                23.9                    77.0
1
4         0  31.0                23.9                    40.0
0
..      ...   ...                 ...                     ...
...
195       0  35.0                93.4                    79.0
1
196       0  45.0                93.4                    28.0
0
197       1  32.0                93.4                    74.0
1
198       1  32.0                93.4                    18.0
2
199       1  30.0                93.4                    83.0
1
```

```
[200 rows x 5 columns]>
```

```
#Independent Variables
l2=list(df.columns)
df2=df[l2[:len(l2)-1]]
df2.head()
```

```
   Gender   Age  Annual Income (k$)  Spending Score (1-100)
0       1  21.0                23.9                    39.0
1       1  21.0                23.9                    81.0
2       0  21.0                23.9                    13.0
3       0  23.0                23.9                    77.0
4       0  31.0                23.9                    40.0
```

```
#Dependent Variables
df1=df['pred_cluster']
df1
```

```
0      1
1      1
2      0
3      1
4      0
      ..
195    1
196    0
197    1
198    2
199    1
Name: pred_cluster, Length: 200, dtype: int32
```

```
#Scaling Variables
import pandas as pd
from sklearn.preprocessing import StandardScaler
scaler=StandardScaler()
scaled=scaler.fit_transform(df)
print(scaled)
```

```
[[ 1.12815215 -1.39030724 -1.599128   -0.47229142  0.10027894]
 [ 1.12815215 -1.39030724 -1.599128    1.28936397  0.10027894]
 [-0.88640526 -1.39030724 -1.599128   -1.56284    -1.2367736 ]
 [-0.88640526 -1.2301336  -1.599128    1.12158726  0.10027894]
 [-0.88640526 -0.58943902 -1.599128   -0.43034725 -1.2367736 ]
 [-0.88640526 -1.31022042 -1.599128    1.07964309  0.10027894]
 [-0.88640526 -0.26909172 -1.599128   -1.56284    -1.2367736 ]
 [-0.88640526 -1.2301336  -1.599128    1.54522344  0.10027894]
 [ 1.12815215  1.6610007  -1.599128   -1.56284     1.43733148]
 [-0.88640526 -0.66952584 -1.599128    0.91186639  0.10027894]
 [ 1.12815215  1.6610007  -1.599128   -1.52089582  1.43733148]
 [-0.88640526 -0.26909172 -1.599128    1.54522344  0.10027894]
```

```
[-0.88640526  1.5729052  -1.599128   -1.47895164 -1.2367736 ]
[-0.88640526 -1.15004677 -1.599128    1.12158726  0.10027894]
[ 1.12815215 -0.10891808 -1.599128   -1.56284     1.43733148]
[ 1.12815215 -1.31022042 -1.599128    1.20547562  0.10027894]
[-0.88640526 -0.26909172 -1.599128   -0.64006812 -1.2367736 ]
[ 1.12815215 -1.39030724 -1.599128    0.66020133  0.10027894]
[ 1.12815215  1.09238426 -1.599128   -0.89173318  1.43733148]
[-0.88640526 -0.26909172 -1.599128    1.54522344  0.10027894]
[ 1.12815215 -0.26909172 -1.59462848 -0.64006812  1.43733148]
[ 1.12815215 -1.06995995 -1.59462848  0.95381056  0.10027894]
[-0.88640526  0.61186333 -1.54963332 -1.56284    -1.2367736 ]
[ 1.12815215 -0.58943902 -1.54963332  0.95381056  0.10027894]
[-0.88640526  1.25255791 -1.41464784 -1.52089582 -1.2367736 ]
[ 1.12815215 -0.74961266 -1.41464784  1.33130814  0.10027894]
[-0.88640526  0.5317765  -1.41464784 -0.76590065 -1.2367736 ]
[ 1.12815215 -0.26909172 -1.41464784  0.45048045  0.10027894]
[-0.88640526  0.13134239 -1.36965268 -0.80784483 -1.2367736 ]
[-0.88640526 -1.2301336  -1.36965268  1.54102902  0.10027894]
[ 1.12815215  1.6610007  -1.32465752 -1.56284     1.43733148]
[-0.88640526 -1.39030724 -1.32465752  0.95381056  0.10027894]
[ 1.12815215  1.17247108 -1.18967204 -1.56284     1.43733148]
[ 1.12815215 -1.39030724 -1.18967204  1.54522344  0.10027894]
[-0.88640526  0.85212379 -1.18967204 -1.52089582 -1.2367736 ]
[-0.88640526 -1.39030724 -1.18967204  1.28936397  0.10027894]
[-0.88640526  0.29151603 -1.14467688 -1.39506329 -1.2367736 ]
[-0.88640526 -0.66952584 -1.14467688  0.95381056  0.10027894]
[-0.88640526 -0.1890049  -1.0096914  -1.01756571 -1.2367736 ]
[-0.88640526 -1.39030724 -1.0096914   1.03769891  0.10027894]
[-0.88640526  1.6610007  -0.96469624 -0.64006812 -1.2367736 ]
[ 1.12815215 -1.15004677 -0.96469624  1.54522344  0.10027894]
[ 1.12815215  0.77203697 -0.91970108 -0.59812395  1.43733148]
[-0.88640526 -0.58943902 -0.91970108  0.45048045  0.10027894]
[-0.88640526  0.85212379 -0.91970108 -0.93367736 -1.2367736 ]
[-0.88640526 -1.15004677 -0.91970108  0.61825715  0.10027894]
[-0.88640526  0.93221062 -0.87470592  0.19881539 -1.2367736 ]
[-0.88640526 -0.90978631 -0.87470592 -0.13673801  0.10027894]
[-0.88640526 -0.74961266 -0.87470592 -0.34645889 -1.2367736 ]
[-0.88640526 -0.58943902 -0.87470592 -0.34645889 -1.2367736 ]
[-0.88640526  0.85212379 -0.78471559  0.07298287 -1.2367736 ]
[ 1.12815215 -0.42926537 -0.78471559  0.40853627  0.10027894]
[-0.88640526 -0.58943902 -0.73972043  0.15687122  0.10027894]
[ 1.12815215  1.65299202 -0.73972043  0.40853627  1.43733148]
[-0.88640526  0.93221062 -0.73972043 -0.22062637 -1.2367736 ]
[ 1.12815215  0.69195015 -0.73972043 -0.38840307  1.43733148]
[-0.88640526  1.01229744 -0.69472527 -0.01090549 -1.2367736 ]
[ 1.12815215  1.6610007  -0.69472527 -0.17868219  1.43733148]
[-0.88640526 -0.90978631 -0.60473495  0.03103869  0.10027894]
[ 1.12815215  1.17247108 -0.60473495 -0.17868219  1.43733148]
[ 1.12815215  1.6610007  -0.60473495  0.24075957  1.43733148]
[ 1.12815215 -1.39030724 -0.60473495  0.19881539  0.10027894]
```

```
[-0.88640526  1.6610007  -0.55973979  0.07298287 -1.2367736 ]
[-0.88640526  1.25255791 -0.55973979  0.3665921  -1.2367736 ]
[ 1.12815215  1.6610007  -0.51474463  0.03103869  1.43733148]
[ 1.12815215 -1.39030724 -0.51474463  0.3665921   0.10027894]
[-0.88640526  0.37160286 -0.51474463 -0.01090549 -1.2367736 ]
[-0.88640526  1.6610007  -0.51474463 -0.09479384 -1.2367736 ]
[ 1.12815215 -1.39030724 -0.51474463  0.3665921   0.10027894]
[-0.88640526 -0.50935219 -0.51474463 -0.13673801 -1.2367736 ]
[ 1.12815215  1.6610007  -0.46974947  0.19881539  1.43733148]
[-0.88640526  0.69195015 -0.46974947 -0.34645889 -1.2367736 ]
[-0.88640526  1.6610007  -0.42475431 -0.05284966 -1.2367736 ]
[-0.88640526  1.6610007  -0.42475431  0.24075957 -1.2367736 ]
[ 1.12815215  1.65299202 -0.24477367 -0.13673801  1.43733148]
[ 1.12815215 -0.98987313 -0.24477367  0.15687122  0.10027894]
[-0.88640526  0.5317765  -0.24477367  0.11492704 -1.2367736 ]
[ 1.12815215  0.13134239 -0.24477367 -0.09479384  1.43733148]
[-0.88640526 -1.2301336  -0.24477367  0.07298287  0.10027894]
[-0.88640526  0.85212379 -0.24477367 -0.34645889 -1.2367736 ]
[ 1.12815215  1.49281838 -0.24477367  0.03103869  1.43733148]
[ 1.12815215 -0.02883126 -0.24477367  0.19881539  1.43733148]
[ 1.12815215  1.6610007  -0.24477367 -0.38840307  1.43733148]
[-0.88640526  0.61186333 -0.24477367 -0.26257054 -1.2367736 ]
[-0.88640526 -1.39030724 -0.24477367  0.28270375  0.10027894]
[ 1.12815215  0.77203697 -0.24477367 -0.17868219  1.43733148]
[-0.88640526  1.33264473 -0.10978819  0.32464792 -1.2367736 ]
[-0.88640526 -1.31022042 -0.10978819  0.19881539  0.10027894]
[-0.88640526 -0.34917855 -0.06479303  0.40853627  0.10027894]
[-0.88640526  0.93221062 -0.06479303 -0.17868219 -1.2367736 ]
[-0.88640526  1.6610007  -0.01979787  0.19881539 -1.2367736 ]
[ 1.12815215 -1.39030724 -0.01979787 -0.38840307  0.10027894]
[ 1.12815215  0.77203697  0.02519729 -0.05284966  1.43733148]
[-0.88640526  0.13134239  0.02519729 -0.43034725 -1.2367736 ]
[-0.88640526 -0.50935219  0.02519729 -0.34645889 -1.2367736 ]
[ 1.12815215 -1.15004677  0.02519729  0.07298287  0.10027894]
[-0.88640526  0.69195015  0.02519729 -0.13673801 -1.2367736 ]
[-0.88640526 -0.90978631  0.02519729 -0.01090549  0.10027894]
[ 1.12815215  0.77203697  0.07019245 -0.34645889  1.43733148]
[ 1.12815215 -1.39030724  0.07019245 -0.05284966  0.10027894]
[-0.88640526 -1.2301336   0.11518761 -0.38840307  0.10027894]
[-0.88640526  0.85212379  0.11518761 -0.09479384 -1.2367736 ]
[ 1.12815215  1.6610007   0.11518761  0.3665921   1.43733148]
[ 1.12815215 -0.98987313  0.11518761  0.19881539  0.10027894]
[ 1.12815215  0.85212379  0.11518761  0.24075957  1.43733148]
[-0.88640526 -1.39030724  0.11518761 -0.34645889  0.10027894]
[-0.88640526  1.6610007   0.16018277 -0.01090549 -1.2367736 ]
[ 1.12815215  1.25255791  0.16018277 -0.17868219  1.43733148]
[ 1.12815215  1.6610007   0.16018277 -0.30451472  1.43733148]
[ 1.12815215  1.6610007   0.16018277 -0.09479384  1.43733148]
[ 1.12815215  1.6610007   0.16018277  0.07298287  1.43733148]
[-0.88640526 -1.39030724  0.16018277  0.15687122  0.10027894]
```

```
[-0.88640526 -0.02883126  0.20517793 -0.34645889 -1.2367736 ]
[ 1.12815215 -1.39030724  0.20517793 -0.17868219  0.10027894]
[-0.88640526 -1.39030724  0.25017309 -0.09479384  0.10027894]
[-0.88640526 -1.39030724  0.25017309 -0.01090549  0.10027894]
[-0.88640526  1.6610007   0.25017309 -0.30451472 -1.2367736 ]
[-0.88640526  0.85212379  0.25017309  0.3665921  -1.2367736 ]
[-0.88640526  1.01229744  0.34016341 -0.30451472 -1.2367736 ]
[-0.88640526  0.93221062  0.34016341  0.28270375 -1.2367736 ]
[ 1.12815215 -0.90978631  0.34016341  0.24075957  0.10027894]
[-0.88640526 -0.02883126  0.34016341 -0.43034725 -1.2367736 ]
[-0.88640526  0.13134239  0.43015373  0.32464792 -1.2367736 ]
[ 1.12815215  0.05125557  0.43015373  1.54522344  0.10027894]
[-0.88640526 -1.2301336   0.47514889 -0.89173318 -1.2367736 ]
[-0.88640526 -0.58943902  0.47514889  1.12158726  0.10027894]
[ 1.12815215  0.37160286  0.52014405 -0.64006812  1.43733148]
[ 1.12815215  0.13134239  0.52014405  1.54522344  0.10027894]
[ 1.12815215  1.65299202  0.52014405 -1.56284     1.43733148]
[ 1.12815215 -0.02883126  0.52014405  1.03769891  0.10027894]
[ 1.12815215  0.69195015  0.52014405 -1.56284     1.43733148]
[ 1.12815215  0.05125557  0.52014405  1.03769891  0.10027894]
[-0.88640526 -1.06995995  0.56513921 -0.6820123  -1.2367736 ]
[-0.88640526 -0.58943902  0.56513921  0.86992221  0.10027894]
[ 1.12815215 -1.39030724  0.61013437 -1.56284     1.43733148]
[-0.88640526 -0.74961266  0.61013437  1.54522344  0.10027894]
[-0.88640526  0.45168968  0.61013437 -1.56284    -1.2367736 ]
[ 1.12815215 -0.50935219  0.61013437  0.95381056  0.10027894]
[ 1.12815215 -1.39030724  0.65512953 -1.56284     1.43733148]
[-0.88640526 -0.26909172  0.65512953  0.91186639  0.10027894]
[-0.88640526  1.49281838  0.70012469 -1.56284    -1.2367736 ]
[ 1.12815215 -0.50935219  0.70012469  1.54522344  0.10027894]
[-0.88640526 -0.82969948  0.74511985 -0.43034725 -1.2367736 ]
[-0.88640526 -0.50935219  0.74511985  1.54102902  0.10027894]
[ 1.12815215 -1.06995995  0.79011501 -1.56284     1.43733148]
[ 1.12815215 -0.82969948  0.79011501  1.54522344  0.10027894]
[ 1.12815215  0.77203697  0.79011501 -0.59812395  1.43733148]
[-0.88640526 -0.50935219  0.79011501  0.99575474  0.10027894]
[-0.88640526 -0.34917855  0.83511017 -1.18534241 -1.2367736 ]
[ 1.12815215 -0.34917855  0.83511017  1.54522344  0.10027894]
[ 1.12815215  0.37160286  0.83511017 -1.39506329  1.43733148]
[ 1.12815215  0.05125557  0.83511017  1.54522344  0.10027894]
[-0.88640526  0.45168968  0.83511017 -1.26923076 -1.2367736 ]
[-0.88640526 -0.02883126  0.83511017  1.07964309  0.10027894]
[-0.88640526  0.69195015  0.83511017 -1.43700747 -1.2367736 ]
[-0.88640526 -0.90978631  0.83511017  1.54522344  0.10027894]
[ 1.12815215 -0.10891808  0.83511017 -1.56284     1.43733148]
[-0.88640526 -0.66952584  0.83511017  1.16353144  0.10027894]
[ 1.12815215 -0.34917855  0.83511017 -1.56284     1.43733148]
[-0.88640526 -0.66952584  0.83511017  0.95381056  0.10027894]
[-0.88640526  1.41273155  0.88010533 -0.64006812 -1.2367736 ]
[-0.88640526 -0.74961266  0.88010533  1.37325232  0.10027894]
```

```
 [ 1.12815215 -1.39030724  0.97009565 -1.56284     1.43733148]
 [-0.88640526 -0.58943902  0.97009565  1.54522344  0.10027894]
 [ 1.12815215  0.93221062  1.1500763  -1.01756571  1.43733148]
 [-0.88640526 -0.1890049   1.1500763   1.03769891  0.10027894]
 [ 1.12815215  0.29151603  1.19507146 -1.26923076  1.43733148]
 [-0.88640526 -0.42926537  1.19507146  1.54522344  0.10027894]
 [-0.88640526 -0.1890049   1.24006662 -0.97562153 -1.2367736 ]
 [ 1.12815215 -0.50935219  1.24006662  0.5343688   0.10027894]
 [ 1.12815215  0.13134239  1.24006662 -1.56284     1.43733148]
 [ 1.12815215 -0.82969948  1.24006662  1.03769891  0.10027894]
 [ 1.12815215 -0.1890049   1.24006662 -1.56284     1.43733148]
 [ 1.12815215 -0.1890049   1.24006662  1.54522344  0.10027894]
 [-0.88640526  1.09238426  1.28506178 -1.56284    -1.2367736 ]
 [-0.88640526 -0.66952584  1.28506178  1.49908485  0.10027894]
 [ 1.12815215  1.5729052   1.28506178 -1.47895164  1.43733148]
 [ 1.12815215 -0.90978631  1.28506178  0.78603386  0.10027894]
 [ 1.12815215  1.65299202  1.51003758 -1.52089582  1.43733148]
 [ 1.12815215 -0.26909172  1.51003758  1.54522344  0.10027894]
 [-0.88640526 -0.10891808  1.52803564 -0.76590065 -1.2367736 ]
 [-0.88640526 -0.50935219  1.52803564  1.49908485  0.10027894]
 [ 1.12815215  0.61186333  1.52803564 -1.47895164  1.43733148]
 [-0.88640526 -0.74961266  1.52803564  1.54522344  0.10027894]
 [-0.88640526  0.21142921  1.52803564 -0.47229142 -1.2367736 ]
 [ 1.12815215 -0.66952584  1.52803564  1.54522344  0.10027894]
 [-0.88640526  1.25255791  1.52803564 -1.10145406 -1.2367736 ]
 [ 1.12815215 -0.82969948  1.52803564  0.74408968  0.10027894]
 [-0.88640526  0.21142921  1.52803564 -1.39506329 -1.2367736 ]
 [-0.88640526 -0.1890049   1.52803564  1.45714067  0.10027894]
 [-0.88640526 -0.34917855  1.52803564 -1.14339824 -1.2367736 ]
 [-0.88640526 -0.50935219  1.52803564  0.78603386  0.10027894]
 [ 1.12815215 -0.42926537  1.52803564 -1.56284     1.43733148]
 [-0.88640526 -0.02883126  1.52803564  1.54522344  0.10027894]
 [-0.88640526  0.69195015  1.52803564 -1.43700747 -1.2367736 ]
 [-0.88640526 -0.26909172  1.52803564  1.20547562  0.10027894]
 [-0.88640526  0.5317765   1.52803564 -0.93367736 -1.2367736 ]
 [ 1.12815215 -0.50935219  1.52803564  0.99575474  0.10027894]
 [ 1.12815215 -0.50935219  1.52803564 -1.35311912  1.43733148]
 [ 1.12815215 -0.66952584  1.52803564  1.37325232  0.10027894]]
```

```
#Splitting Dataset
import numpy as np
from sklearn.model_selection import train_test_split
y=df['pred_cluster'].to_numpy()
X_train,X_test,y_train,y_test=train_test_split(scaled,y,test_size=0.10
,random_state=42)

X_train.shape
```

```
(180, 5)
```

```
X_test.shape
```

```
(20, 5)
```

```
y_train.shape
```

```
(180,)
```

```
y_test.shape
```

```
(20,)
```

```
#Training the model
from sklearn.svm import SVR
regrassor=SVR(kernel='rbf')
regrassor.fit(X_train,y_train)
```

```
SVR()
```

```
#Testing the model
y_pred=regrassor.predict((X_test))
```

```
df3=pd.DataFrame({'Predicted value':y_pred,'Real Value':y_test})
df3
```

	Predicted value	Real Value
0	1.067228	1
1	0.965805	1
2	1.938242	2
3	2.032868	2
4	1.996615	2
5	0.950163	1
6	0.056714	0
7	1.994406	2
8	0.117154	0
9	1.000021	1
10	-0.008916	0
11	1.949340	2
12	0.904566	1
13	0.957596	1
14	0.154222	0
15	1.044386	1
16	-0.006081	0
17	-0.070443	0
18	2.050848	2
19	1.000664	1

```
#performance metrics
from sklearn.metrics import mean_squared_error
import math
print(mean_squared_error(y_test,y_pred))
print(math.sqrt(mean_squared_error(y_test,y_pred)))
```

```
0.0038471231954710695
0.06202518194629557
```