Data Mining

ASSIGNMENT 4

Jeswin W - 19IT040

Jeya Ganesh A V - 19IT041

Rahul Hariesh B - 19IT074

In [12]:

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

In [15]:

```
df=pd.read_csv("Mall_Customers.csv")
df.head(10)
```

Out[15]:

	CustomerID	Age	Annual Income	Spending Score
0	1	19	15	39
1	2	21	15	81
2	3	20	16	6
3	4	23	16	77
4	5	31	17	40
5	6	22	17	76
6	7	35	18	6
7	8	23	18	94
8	9	64	19	3
9	10	30	19	72

In [16]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 4 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   CustomerID      200 non-null    int64
 1   Age             200 non-null    int64
 2   Annual Income   200 non-null    int64
 3   Spending Score  200 non-null    int64
dtypes: int64(4)
memory usage: 6.4 KB
```

In [17]:

```
df.describe()
```

Out[17]:

	CustomerID	Age	Annual Income	Spending Score
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

In [18]:

```
df.isnull().sum()
```

Out[18]:

```
CustomerID        0
Age               0
Annual Income     0
Spending Score    0
dtype: int64
```

In [20]:

```
df.drop_duplicates(inplace=True)
```

In [22]:

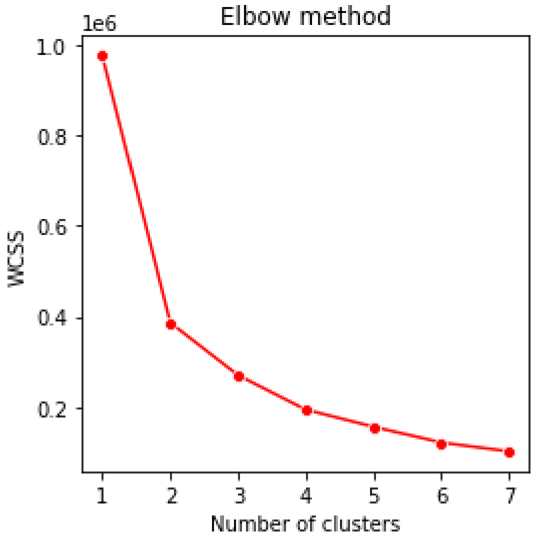
```
from sklearn.cluster import KMeans
wcss=[]
for i in range(1,8):
    kmeans=KMeans(n_clusters=i,init='k-means++')
    kmeans.fit(df)
    wcss.append(kmeans.inertia_)

plt.figure(figsize=(4,4))
sns.lineplot(range(1,8),wcss,marker='o',color='red')
plt.title('Elbow method')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS')
plt.show
```

```
C:\Users\Jeswin\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:881:
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 th
e environment variable OMP_NUM_THREADS=1.
  warnings.warn(
C:\Users\Jeswin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: Futur
eWarning: Pass the following variables as keyword args: x, y. From version
0.12, the only valid positional argument will be `data`, and passing other a
rguments without an explicit keyword will result in an error or misinterpret
ation.
  warnings.warn(
```

Out[22]:

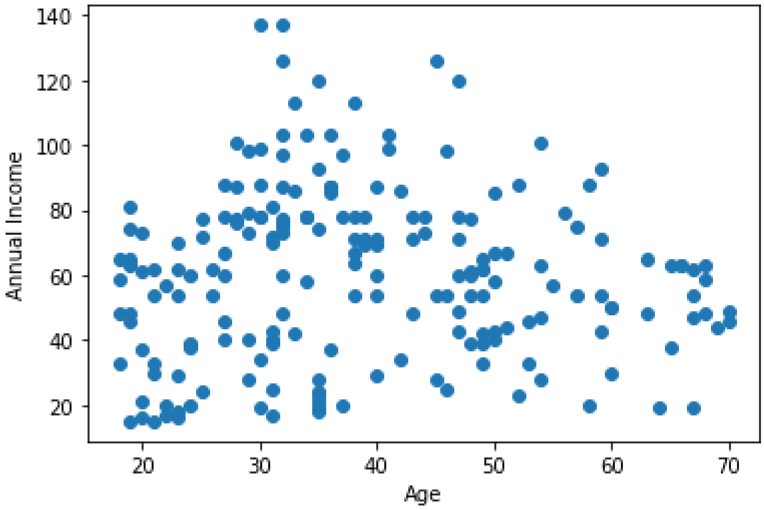
```
<function matplotlib.pyplot.show(close=None, block=None)>
```



K-Means

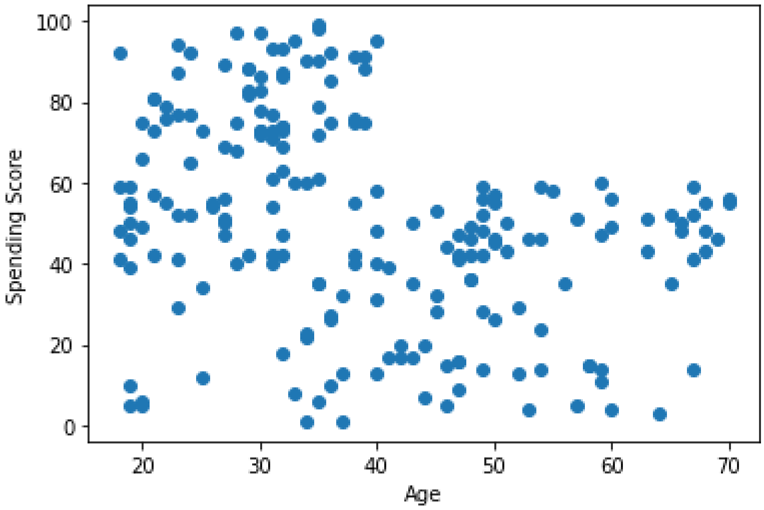
In [23]:

```
plt.scatter(df['Age'], df['Annual Income'])
plt.xlabel('Age')
plt.ylabel('Annual Income')
plt.show()
```



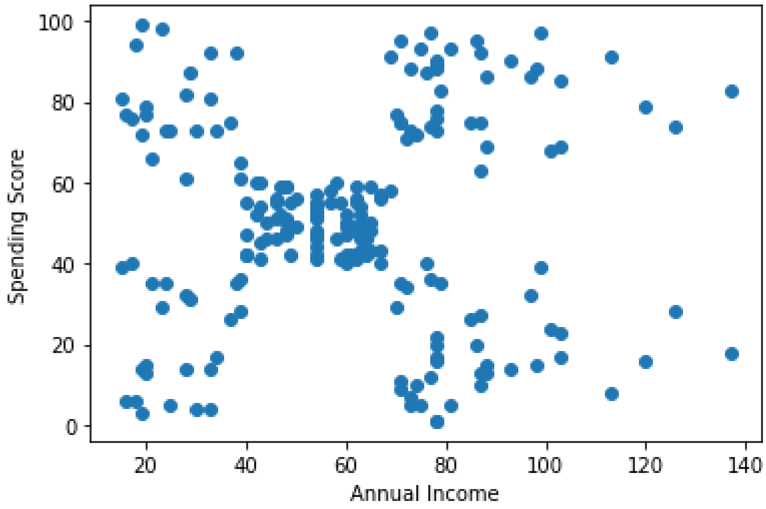
In [24]:

```
plt.scatter(df['Age'], df['Spending Score'])
plt.xlabel('Age')
plt.ylabel('Spending Score')
plt.show()
```



In [25]:

```
plt.scatter(df['Annual Income'], df['Spending Score'])
plt.xlabel('Annual Income')
plt.ylabel('Spending Score')
plt.show()
```

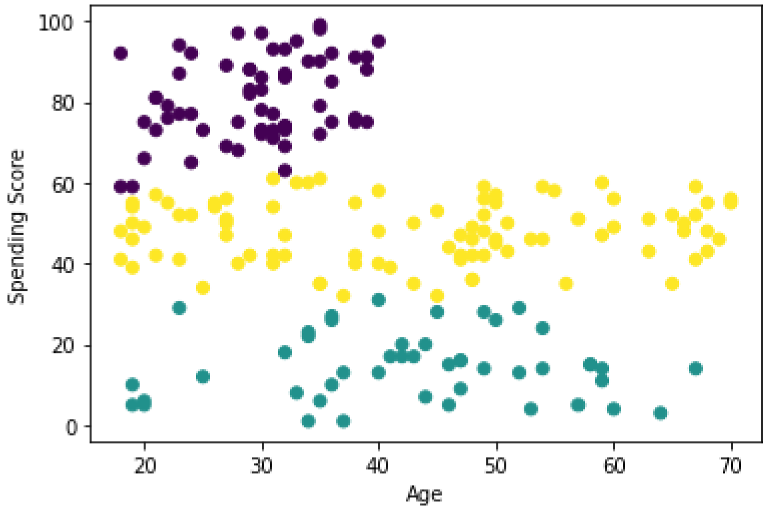


In [26]:

```
df_1=df.loc[:,['Age','Spending Score']]
kmeans_1=KMeans(n_clusters=3)
kmeans_1.fit(df_1)
labels_1=kmeans_1.predict(df_1)
```

In [27]:

```
plt.scatter(df['Age'],df['Spending Score'],c=labels_1)
plt.xlabel('Age')
plt.ylabel('Spending Score')
plt.show()
```



In [28]:

```
kmeans_1.cluster_centers_
```

Out[28]:

```
array([[29.56451613, 80.74193548],
       [42.95744681, 14.59574468],
       [43.05494505, 47.78021978]])
```

In [29]:

```
kmeans_1.n_iter_
```

Out[29]:

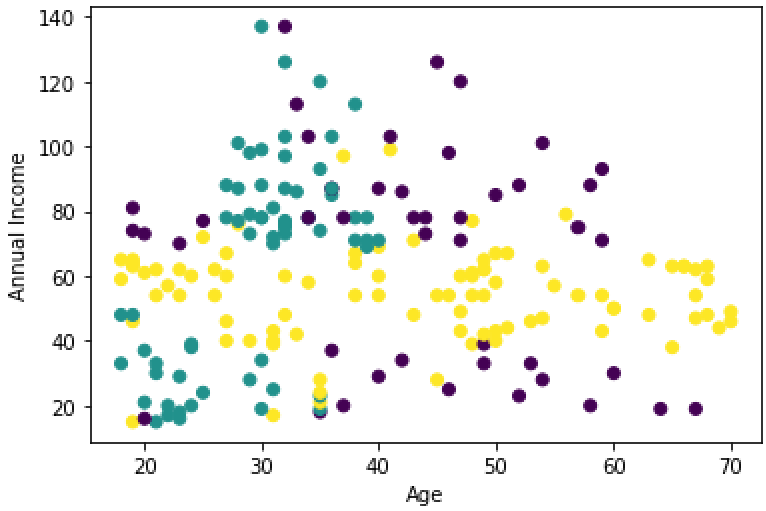
```
8
```

In [30]:

```
df_2 = df.loc[:, ['Age', 'Annual Income']]
kmeans_2 = KMeans(n_clusters=3)
kmeans_2.fit(df_1)
labels_2 = kmeans_2.predict(df_1)
```

In [31]:

```
plt.scatter(df['Age'], df['Annual Income'], c = labels_2)
plt.xlabel('Age')
plt.ylabel('Annual Income')
plt.show()
```



In [33]:

```
kmeans_2.cluster_centers_
```

Out[33]:

```
array([[42.95744681, 14.59574468],
       [29.56451613, 80.74193548],
       [43.05494505, 47.78021978]])
```

In [34]:

```
kmeans_2.n_iter_
```

Out[34]:

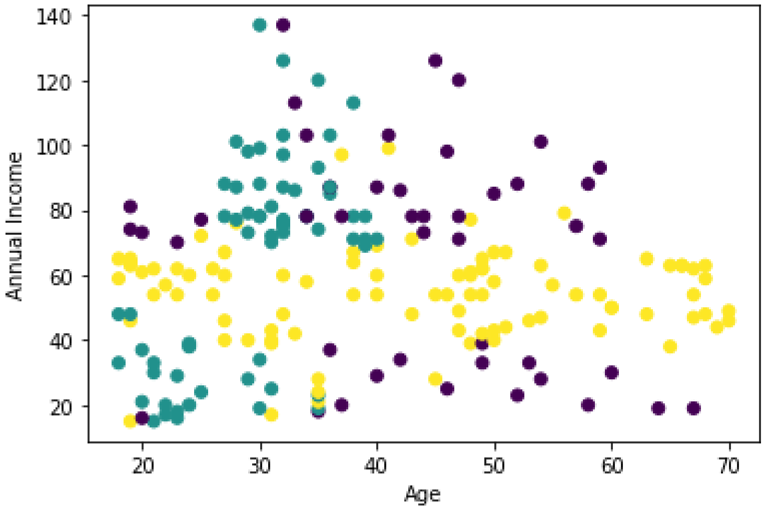
```
11
```

In [35]:

```
df_3 = df.loc[:, ['Age', 'Annual Income']]
kmeans_3 = KMeans(n_clusters=3)
kmeans_3.fit(df_1)
labels_3 = kmeans_3.predict(df_1)
```

In [36]:

```
plt.scatter(df['Age'], df['Annual Income'], c = labels_3)
plt.xlabel('Age')
plt.ylabel('Annual Income')
plt.show()
```



In [37]:

```
kmeans_3.cluster_centers_
```

Out[37]:

```
array([[42.95744681, 14.59574468],
       [29.56451613, 80.74193548],
       [43.05494505, 47.78021978]])
```

In [38]:

```
kmeans_3.n_iter_
```

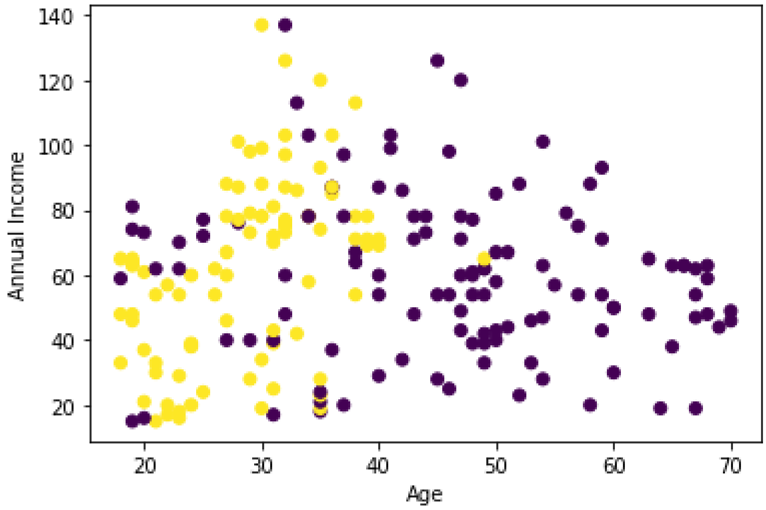
Out[38]:

```
8
```

Number of clusters = 2

In [39]:

```
df_4 = df.loc[:, ['Age', 'Annual Income']]
kmeans_4 = KMeans(n_clusters=2)
kmeans_4.fit(df_1)
labels_4 = kmeans_4.predict(df_1)
plt.scatter(df['Age'], df['Annual Income'], c = labels_4)
plt.xlabel('Age')
plt.ylabel('Annual Income')
plt.show()
```



K-Medoids

In [47]:

```
conda install -c conda-forge scikit-learn-extra
```

```
Collecting package metadata (current_repodata.json): ...working... done
Solving environment: ...working... done

## Package Plan ##

  environment location: C:\Users\Jeswin\anaconda3

  added / updated specs:
    - scikit-learn-extra


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    conda-4.11.0               |   py38haa244fe_0        16.9 MB  conda-forg
e
    python_abi-3.8             |           2_cp38           4 KB  conda-forg
e
    scikit-learn-extra-0.2.0   |   py38h60cbd38_0         312 KB  conda-forg
e
    ------------------------------------------------------------
                                           Total:        17.2 MB

The following NEW packages will be INSTALLED:

Note: you may need to restart the kernel to use updated packages.
  python_abi         conda-forge/win-64::python_abi-3.8-2_cp38
  scikit-learn-extra conda-forge/win-64::scikit-learn-extra-0.2.0-py38h60cbd
38_0

The following packages will be UPDATED:




==> WARNING: A newer version of conda exists. <==
  current version: 4.10.1
  latest version: 4.11.0

  conda              pkgs/main::conda-4.10.1-py38haa95532_1 --> conda-forg
e::conda-4.11.0-py38haa244fe_0



Downloading and Extracting Packages

python_abi-3.8       | 4 KB      |            |   0%
python_abi-3.8       | 4 KB      | ########## | 100%
python_abi-3.8       | 4 KB      | ########## | 100%

conda-4.11.0         | 16.9 MB   |            |   0%
conda-4.11.0         | 16.9 MB   | 1          |   1%
conda-4.11.0         | 16.9 MB   | 6          |   7%
conda-4.11.0         | 16.9 MB   | #3         |  13%
```

```
conda-4.11.0         | 16.9 MB   | #9         |  20% 
conda-4.11.0         | 16.9 MB   | ##6        |  27% 
conda-4.11.0         | 16.9 MB   | ###3       |  33% 
conda-4.11.0         | 16.9 MB   | ####       |  40% 
conda-4.11.0         | 16.9 MB   | ####6      |  47% 
conda-4.11.0         | 16.9 MB   | #####3     |  53% 
conda-4.11.0         | 16.9 MB   | ######     |  60% 
conda-4.11.0         | 16.9 MB   | ######6    |  66% 
conda-4.11.0         | 16.9 MB   | #######3   |  73% 
conda-4.11.0         | 16.9 MB   | ########   |  80% 
conda-4.11.0         | 16.9 MB   | ########6  |  87% 
conda-4.11.0         | 16.9 MB   | #########3 |  94% 
conda-4.11.0         | 16.9 MB   | ########## | 100% 

scikit-learn-extra-0 | 312 KB    |            |   0% 
scikit-learn-extra-0 | 312 KB    | ########## | 100% 
scikit-learn-extra-0 | 312 KB    | ########## | 100% 
Preparing transaction: ...working... done
Verifying transaction: ...working... done
Executing transaction: ...working... done


Please update conda by running

    $ conda update -n base -c defaults conda
```

In [48]:

```
from sklearn_extra.cluster import KMedoids
```

In [49]:

```
kmed = KMedoids(n_clusters=3)
y_kmed = kmed.fit_predict(df)
print(y_kmed)
```

```
[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 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 2 0 2 0 2 0 2 0 2 2 2 0 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2]
```

In [50]:

```
kmed.cluster_centers_
```

Out[50]:

```
array([[ 97,  47,  60,  47],
       [ 28,  35,  28,  61],
       [170,  32,  87,  63]], dtype=int64)
```

In [51]:

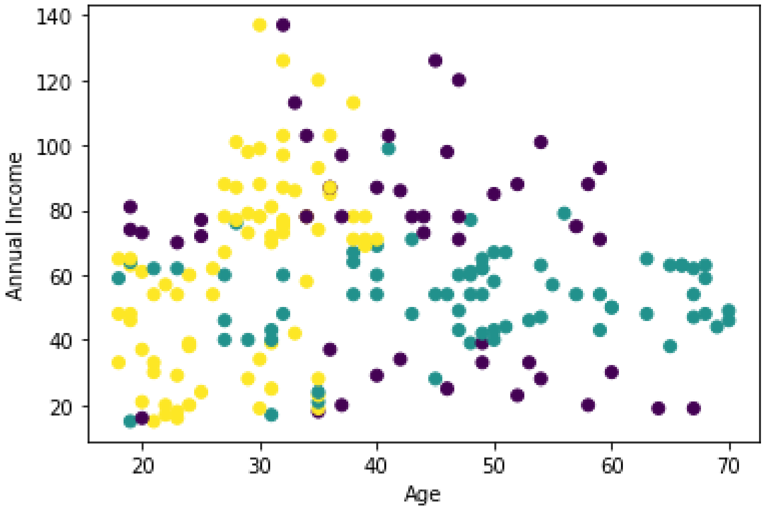
```
kmed.n_iter_
```

Out[51]:

```
4
```

In [52]:

```
df_kmed = df.loc[:, ['Age', 'Annual Income']]
kmed_1 = KMedoids(n_clusters=3)
kmed_1.fit(df_1)
labels_kmed = kmed_1.predict(df_1)
plt.scatter(df['Age'], df['Annual Income'], c = labels_kmed)
plt.xlabel('Age')
plt.ylabel('Annual Income')
plt.show()
```



Performance Analysis:-

Number of iteration in K-Mean Algorithm : 3

Number of iteration in K-Medoids Algorithm : 5

In [ ]: