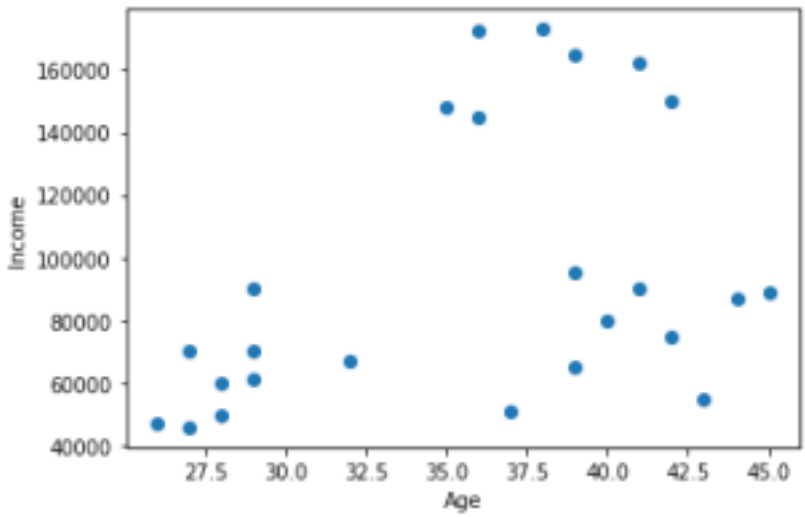
```
from sklearn.cluster import KMeans
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from matplotlib import pyplot as plt
%matplotlib inline
```

```
df = pd.read_csv("/content/income.csv")
df.head()
```

	Name	Age	Income
0	A	27	70000
1	B	29	90000
2	C	29	61000
3	D	28	60000
4	E	42	150000

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

```
Text(0, 0.5, 'Income')
```




```
km = KMeans(n_clusters=3)
y_predicted = km.fit_predict(df[['Age','Income']])
y_predicted
```

```
array([0, 2, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 2, 2, 2, 2, 0,
2, 2, 0], dtype=int32)
```

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

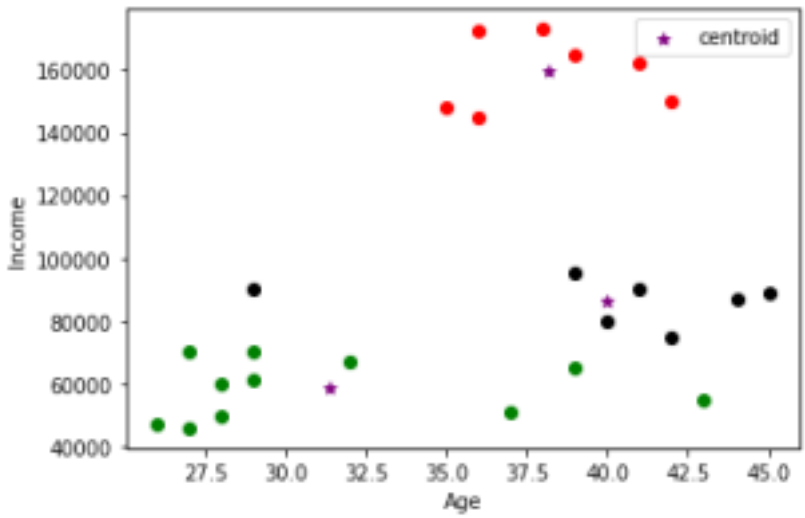
	Name	Age	Income	cluster
0	A	27	70000	0
1	B	29	90000	2
2	C	29	61000	0
3	D	28	60000	0
4	E	42	150000	1

```
km.cluster_centers_
```

```
array([[3.13636364e+01, 5.83090909e+04],
       [3.81428571e+01, 1.59328571e+05],
       [4.00000000e+01, 8.65714286e+04]])
```

```
df1 = df[df.cluster==0]
df2 = df[df.cluster==1]
df3 = df[df.cluster==2]
plt.scatter(df1.Age,df1['Income'],color='green')
plt.scatter(df2.Age,df2['Income'],color='red')
plt.scatter(df3.Age,df3['Income'],color='black')
plt.scatter(km.cluster_centers_[:,0],km.cluster_centers_[:,1],color='purple',marker='*',la
plt.xlabel('Age')
plt.ylabel('Income')
plt.legend()
```

```
<matplotlib.legend.Legend at 0x7f95a19f7d10>
```



Preprocessing using min max scaler

```
scaler = MinMaxScaler()

scaler.fit(df[['Income']])
df['Income'] = scaler.transform(df[['Income']])

scaler.fit(df[['Age']])
df['Age'] = scaler.transform(df[['Age']])
```

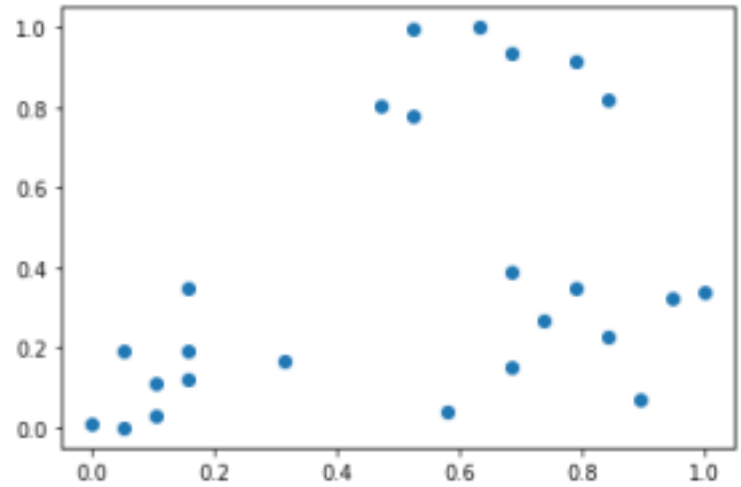


```
df.head()
```

	Name	Age	Income	cluster
0	A	0.052632	0.190252	0
1	B	0.157895	0.347484	2
2	C	0.157895	0.119497	0
3	D	0.105263	0.111635	0
4	E	0.842105	0.819182	1

```
plt.scatter(df.Age,df['Income'])
```

```
<matplotlib.collections.PathCollection at 0x7f95a1931e10>
```




```
km = KMeans(n_clusters=3)
y_predicted = km.fit_predict(df[['Age','Income']])
y_predicted
```

```
array([1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 2, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2,
2, 2, 2], dtype=int32)
```

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

	Name	Age	Income	cluster
0	A	0.052632	0.190252	1
1	B	0.157895	0.347484	1
2	C	0.157895	0.119497	1
3	D	0.105263	0.111635	1
4	E	0.842105	0.819182	0

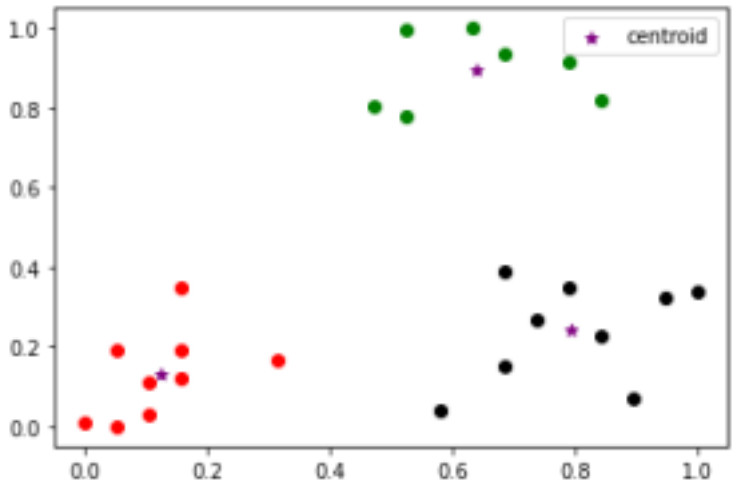
```
km.cluster_centers_
```



```
array([[0.63909774, 0.89252022],
 [0.12280702, 0.12945493],
 [0.79532164, 0.24004193]])
```

```
df1 = df[df.cluster==0]
df2 = df[df.cluster==1]
df3 = df[df.cluster==2]
plt.scatter(df1.Age,df1['Income'],color='green')
plt.scatter(df2.Age,df2['Income'],color='red')
plt.scatter(df3.Age,df3['Income'],color='black')
plt.scatter(km.cluster_centers_[:,0],km.cluster_centers_[:,1],color='purple',marker='*',la
plt.legend()
```

```
<matplotlib.legend.Legend at 0x7f95a18ade90>
```



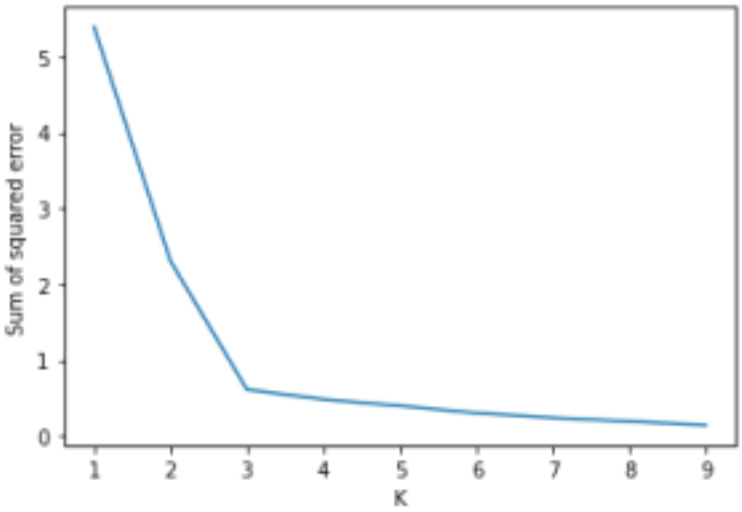
Elbow Plot

```
sse = []
k_rng = range(1,10)
for k in k_rng:
    km = KMeans(n_clusters=k)
    km.fit(df[['Age','Income']])
    sse.append(km.inertia_) # inertia will calculate SSE
```

```
plt.xlabel('K')
plt.ylabel('Sum of squared error')
plt.plot(k_rng,sse)
```



```
[<matplotlib.lines.Line2D at 0x7f95a1838c50>]
```



Colab paid products - Cancel contracts here