

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

```
In [2]: df=pd.read_excel(r'DataFinal17-20.xlsx',sheet_name='2020')
df1=df.iloc[:,2:16]
df1=df1.drop(['Total'], axis=1)
df1.head()
```

```
Out[2]:
```

	Homicide/Murder	Causing death by negligence	Hurt	Assault on woman	Kidnapping and abduction	Human trafficking	Rape	Offence against public tranquility	Offences against property	Offences relating to documents and property marks
0	254	620	1826	543	369	0	92	905	3104	220
1	114	116	1345	245	76	0	28	111	946	86
2	201	354	2133	461	190	0	97	144	2198	250
3	139	386	1869	335	242	0	80	428	3821	265
4	206	308	1233	248	91	0	50	508	1002	104

```
In [3]: from sklearn.cluster import KMeans

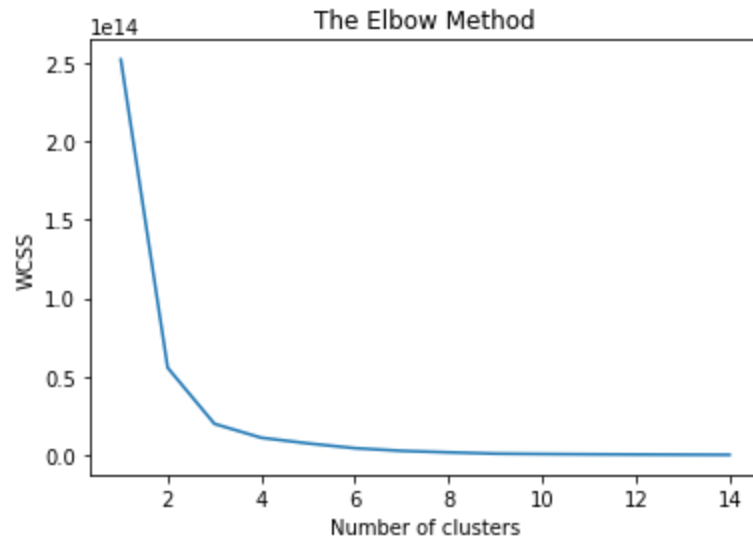
#create a list for the wcss parameter
wcss = []
#test with 14 clusters
for i in range(1, 15):
    kmeans = KMeans(n_clusters = i, init = 'k-means++', random_state = 0)
    kmeans.fit(df1)
    wcss.append(kmeans.inertia_)
```

```
In [4]: wcss
```

```
Out[4]: [251991555908767.72,
55692833341133.74,
19924824817824.41,
11065108262943.742,
7470720341378.248,
4418872317352.032,
2716133886907.053,
1760690819680.5554,
955934227092.7084,
716673512599.725,
528485592493.5999,
406730790121.76666,
301135915319.76666,
194466396974.37378]
```

```
In [5]: plt.plot(range(1, 15), wcss)
plt.title('The Elbow Method')
plt.xlabel('Number of clusters')
```

```
plt.ylabel('WCSS')
plt.show()
```



```
In [6]: km=KMeans(n_clusters=6)
y_pred=km.fit_predict(df1)
```

```
In [7]: y_pred
```

```
Out[7]: array([3, 0, 2, 3, 2, 0, 2, 2, 0, 0, 0, 0, 3, 0, 3, 2, 1, 3, 2, 0, 5, 0,
        0, 0, 4, 2, 0, 2, 2, 0, 3, 1, 0, 0, 2])
```

```
In [8]: df['cluster']=y_pred
df.head()
```

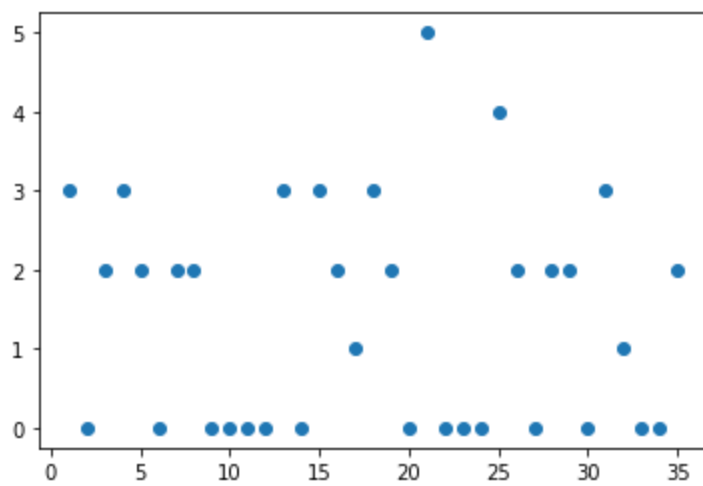
```
Out[8]:
```

	SrNo	District	Homicide/Murder	Causing death by negligence	Hurt	Assault on woman	Kidnapping and abduction	Human trafficking	Rape	Offence against public tranquility	...
0	1	Ahmednagar	254	620	1826	543	369	0	92	905	...
1	2	Akola	114	116	1345	245	76	0	28	111	...
2	3	Amravati	201	354	2133	461	190	0	97	144	...
3	4	Aurangbad	139	386	1869	335	242	0	80	428	...
4	5	Beed	206	308	1233	248	91	0	50	508	...

5 rows × 23 columns

```
In [9]: plt.scatter(df['SrNo'],df['cluster'])
#for col in df.columns:
#    print(col)
```

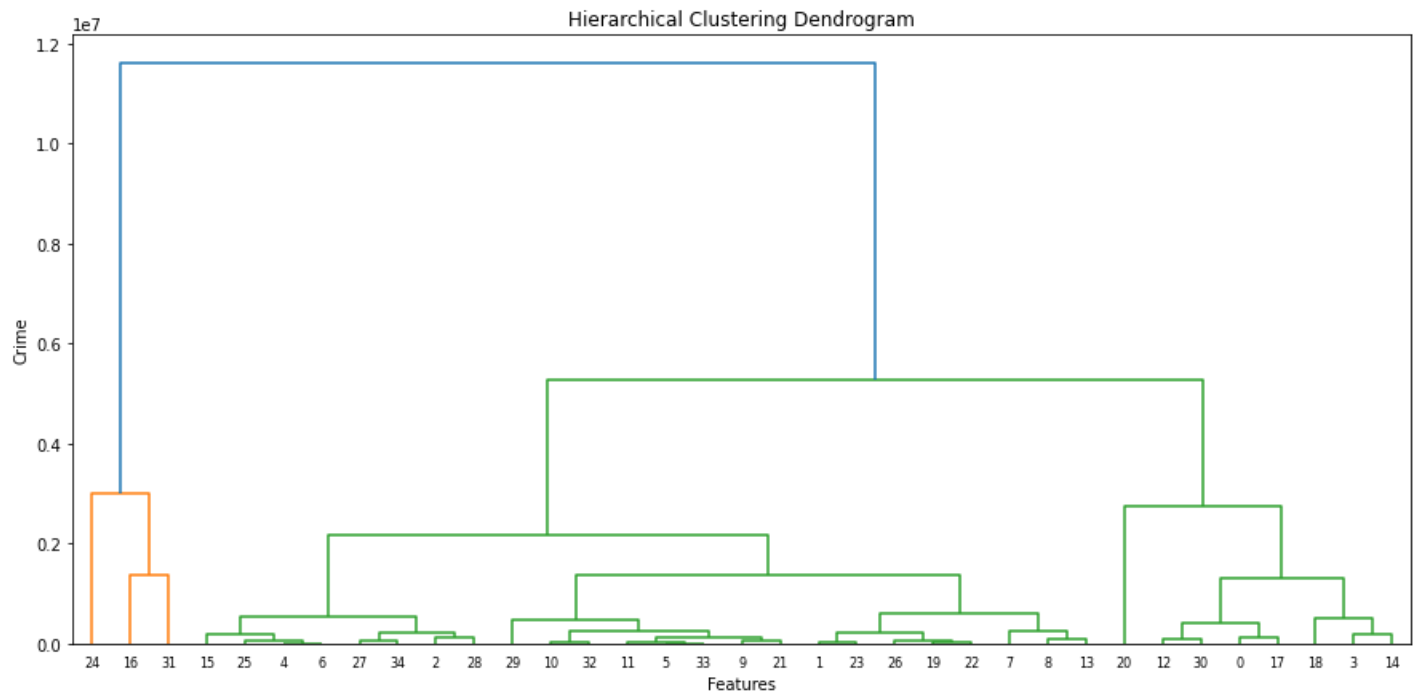
```
Out[9]: <matplotlib.collections.PathCollection at 0x1f22ef52f40>
```



```
In [10]: from scipy.cluster.hierarchy import linkage
import scipy.cluster.hierarchy as sch # for creating dendrogram
```

```
In [11]: z = linkage(dfl, method="complete",metric="euclidean")
```

```
In [12]: plt.figure(figsize=(15,7))
plt.title('Hierarchical Clustering Dendrogram')
plt.xlabel('Features')
plt.ylabel('Crime')
sch.dendrogram(z,
    leaf_rotation=0., # rotates the x axis labels
    leaf_font_size=8., # font size for the x axis labels
)
plt.show()
```



```
In [13]: #df = pd.read_excel (r'Path where the Excel file is stored\File name.xlsx', sheet_name='y')
```

```
In [ ]:
```

