

KMEANS CLUSTERING MODEL ¶

In [48]:

```
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
import scipy.cluster.hierarchy as sch
```

In [49]:

```
df= pd.read_csv('/Users/anjali98/lc.csv')
df.head()
df.describe()
```

Out[49]:

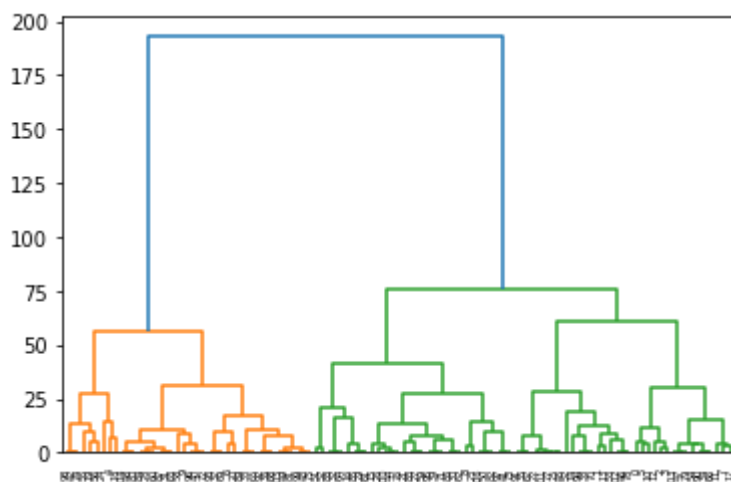
	Age	Smokes	Alcohol	Result
count	107.000000	107.000000	107.000000	107.000000
mean	43.635514	16.046729	3.280374	0.485981
std	15.229931	6.724193	2.188281	0.502155
min	18.000000	0.000000	0.000000	0.000000
25%	32.000000	12.000000	2.000000	0.000000
50%	42.000000	20.000000	3.000000	0.000000
75%	56.000000	20.000000	4.500000	1.000000
max	77.000000	34.000000	8.000000	1.000000

In [50]:

```
y= df['Result']
X= df.drop(['Result'], axis=1)
```

In [51]:

```
dendrogram = sch.dendrogram(sch.linkage(X, method='ward'))
```



In [52]:

```
y= df['Result']
X= df.drop(['Result'], axis=1)
```

In [53]:

```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test = train_test_split(X,y, test_size=0.4, random_state=42)
```

In [54]:

```
from sklearn.cluster import KMeans
kmeans = KMeans(n_clusters=56, random_state=0)
kmeans.fit(X)
kmeans.cluster_centers_

print (kmeans.inertia_)
```

4.1

In [55]:

```
identified_clusters = kmeans.fit_predict(X)
identified_clusters
```

Out[55]:

```
array([[33, 32, 38, 38,  7,  0, 26, 47, 10, 51, 41, 40, 22, 52, 30, 49,
48,
      18, 29, 36, 45, 35, 28, 46, 27, 28,  4, 25, 34, 14, 44, 54, 12,
23,
      5,  8, 11,  3, 21, 19,  1, 15,  6,  9, 39, 28,  2, 42, 24, 53,
20,
      16, 13, 43, 50, 37, 17, 55, 31, 25, 34, 14, 44, 54, 12, 23,  5,
8,
      11,  3, 21, 19,  1, 15,  6,  9, 39, 28,  2, 42, 24, 53, 20, 25,
34,
      14, 44, 54, 12, 23,  5,  8, 11,  3, 21, 19,  1, 15,  6,  9, 39,
28,
      2, 42, 24, 53, 20], dtype=int32)
```

In [56]:

```
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
X_train, X_test,y_train,y_test = train_test_split(X,y,test_size=0.40,random_state=70)
k_means = KMeans(n_clusters=2)
k_means.fit(X_train)
```

Out[56]:

KMeans(n_clusters=2)

FINAL ACCURACY SCORE

In [58]:

```
score = accuracy_score(y_test, k_means.predict(X_test))  
print('Accuracy:{0:f}'.format(score))
```

Accuracy:0.093023

In []: