

```
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt

df = pd.read_csv('hierarchical-clustering-with-python-and-scikit-lear

df.head()
```

	CustomerID	Genre	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.shape

(200, 5)
```

```
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.dtypes
```

```
CustomerID      int64
```

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

```
data = df.iloc[:, 3:5].values
```

```
data
```

```
array([[ 15,  39],
       [ 15,  81],
       [ 16,   6],
       [ 16,  77],
       [ 17,  40],
       [ 17,  76],
       [ 18,   6],
       [ 18,  94],
       [ 19,   3],
       [ 19,  72],
       [ 19,  14],
       [ 19,  99],
       [ 20,  15],
       [ 20,  77],
       [ 20,  13],
       [ 20,  79],
       [ 21,  35],
       [ 21,  66],
       [ 23,  29],
       [ 23,  98],
       [ 24,  35],
       [ 24,  73],
       [ 25,   5],
       [ 25,  73],
       [ 28,  14],
       [ 28,  82],
       [ 28,  32],
       [ 28,  61],
       [ 29,  31],
       [ 29,  87],
       [ 30,   4],
       [ 30,  73],
       [ 33,   4],
       [ 33,  92],
       [ 33,  14],
       [ 33,  81],
       [ 34,  17],
       [ 34,  73],
       [ 37,  26],
       [ 37,  75],
       [ 38,  35],
       ...,
       ...,
       ...])
```

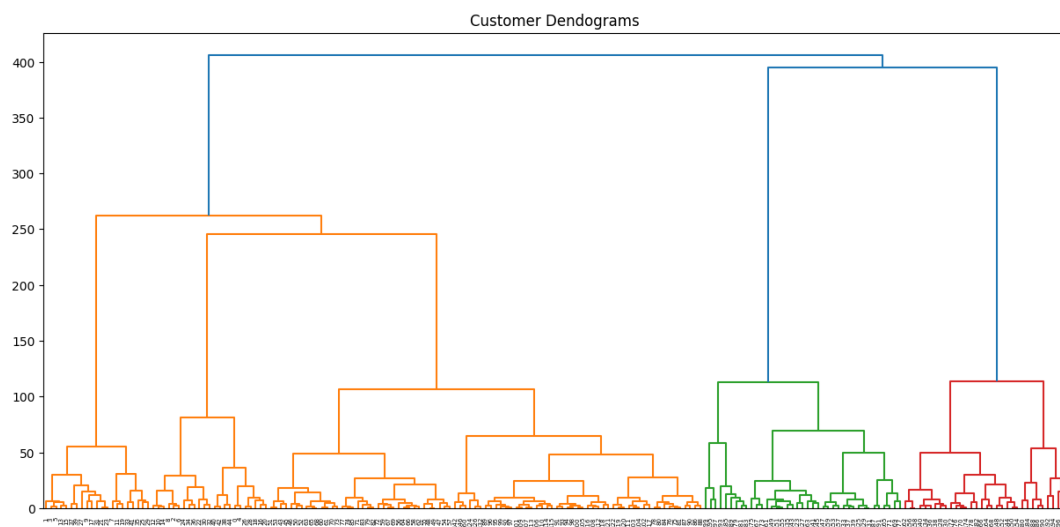
```
[ 38, 92],  
[ 39, 36],  
[ 39, 61],  
[ 39, 28],  
[ 39, 65],  
[ 40, 55],  
[ 40, 47],  
[ 40, 42],  
[ 40, 42],  
[ 42, 52],  
[ 42, 60],  
[ 43, 54],  
[ 43, 60],  
[ 43, 45],  
[ 43, 41],  
[ 44, 50],  
[ 44, 46],
```

```
import scipy.cluster.hierarchy as shc
```

```
plt.figure(figsize=(15, 7))
```

```
plt.title("Customer Dendograms")
```

```
dend = shc.dendrogram(shc.linkage(data, method='ward'))
```



```

from sklearn.cluster import AgglomerativeClustering

cluster = AgglomerativeClustering(n_clusters=5, affinity='euclidean',
labels_=cluster.fit_predict(data)

/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_agglomerative_clustering.py:110: UserWarning:

```

```
labels_
```

```

array([4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4,
3, 4, 3,
       4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4,
3, 4, 1,
       4, 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, 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, 2, 1, 2, 1, 2, 0,
2, 0, 2,
       1, 2, 0, 2, 0, 2, 0, 2, 0, 2, 1, 2, 0, 2, 1, 2, 0, 2, 0,
2, 0, 2,
       0, 2, 0, 2, 0, 2, 1, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0,
2, 0, 2,
       0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0,
2, 0, 2,
       0, 2])

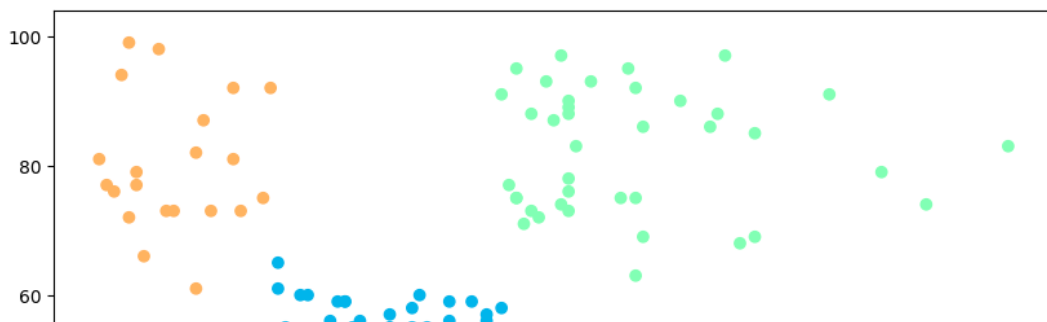
```

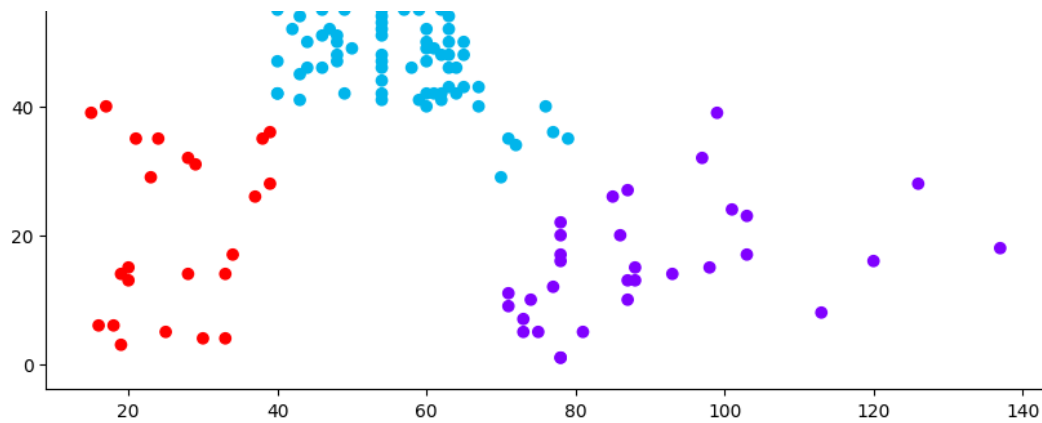
```

plt.figure(figsize=(10, 7))
plt.scatter(data[:,0], data[:,1], c=cluster.labels_, cmap='rainbow')

```

```
<matplotlib.collections.PathCollection at 0x7e379a227940>
```





OUTLIER

data.shape

(200, 2)

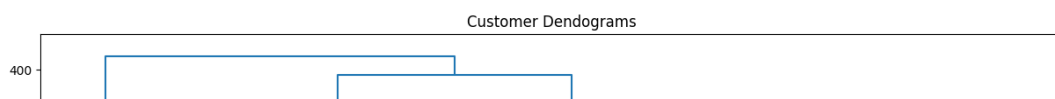
```
new = np.array([[55,122]])
new = np.concatenate((data,new), axis = 0)
```

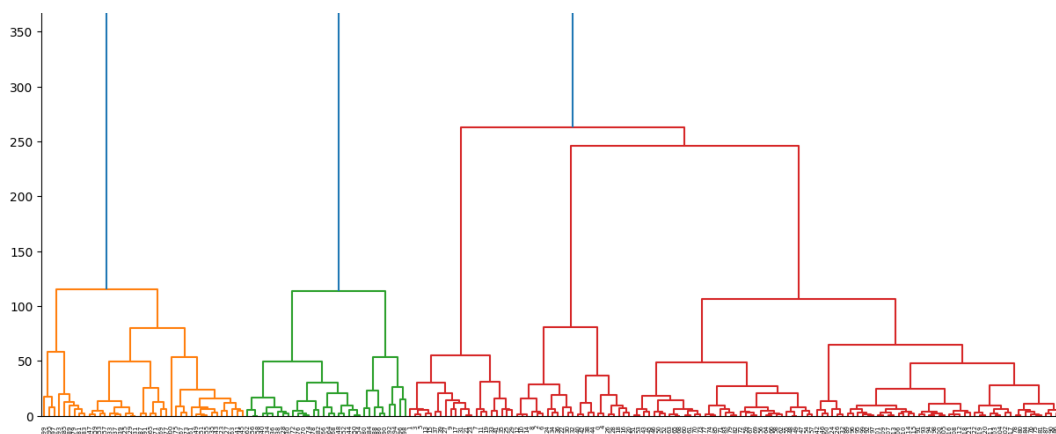
new.shape

(201, 2)

```
import scipy.cluster.hierarchy as shc
```

```
plt.figure(figsize=(15, 7))
plt.title("Customer Dendograms")
dend = shc.dendrogram(shc.linkage(new, method='ward'))
```





```
from sklearn.cluster import AgglomerativeClustering
```

```
cluster = AgglomerativeClustering(n_clusters=5, affinity='euclidean',  
labels_ = cluster.fit_predict(new))
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_agglomerative.py:100: UserWarning: The number of clusters is 5, which is less than the number of data points. This may lead to unexpected results.  
warnings.warn("The number of clusters is %d, which is less than the number of data points. This may lead to unexpected results." % n_clusters)
```

```
labels_
```

```
array([4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4,  
       3, 4, 3,  
       4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4,  
       3, 4, 1,  
       4, 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, 1, 1, 1, 1, 1])
```