

Agglomerative for "Mall_Customers.csv"

```
In [24]: import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import AgglomerativeClustering
from sklearn.preprocessing import StandardScaler
from scipy.cluster.hierarchy import dendrogram
from sklearn.metrics import silhouette_score
```

```
In [25]: # Load the dataset
data = pd.read_csv('Mall_Customers.csv')
```

```
In [26]: # Select the features for clustering ('Annual Income' and 'Spending Score')
X = data[['Annual Income (k$)', 'Spending Score (1-100)']]
```

```
In [27]: # Standardize the data (important for hierarchical clustering)
scaler = StandardScaler()
data_scaled = scaler.fit_transform(X)
```

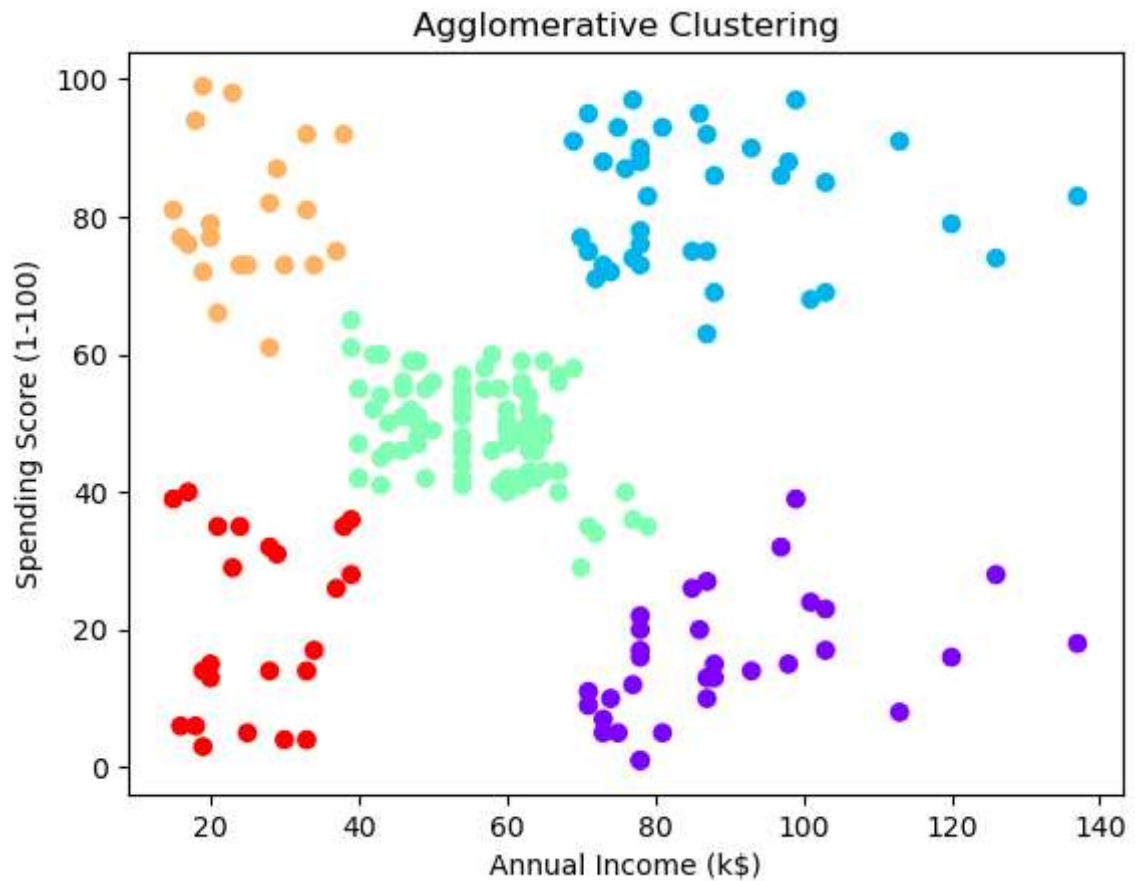
```
In [28]: # Choose the number of clusters
n_clusters = 5
```

```
In [29]: # Perform Agglomerative Clustering
agg_clustering = AgglomerativeClustering(n_clusters=n_clusters)
labels = agg_clustering.fit_predict(data_scaled)
```

```
In [30]: # Calculate the silhouette score
silhouette_avg = silhouette_score(data_scaled, labels)
print(f"Silhouette Score: {silhouette_avg:.2f}")
```

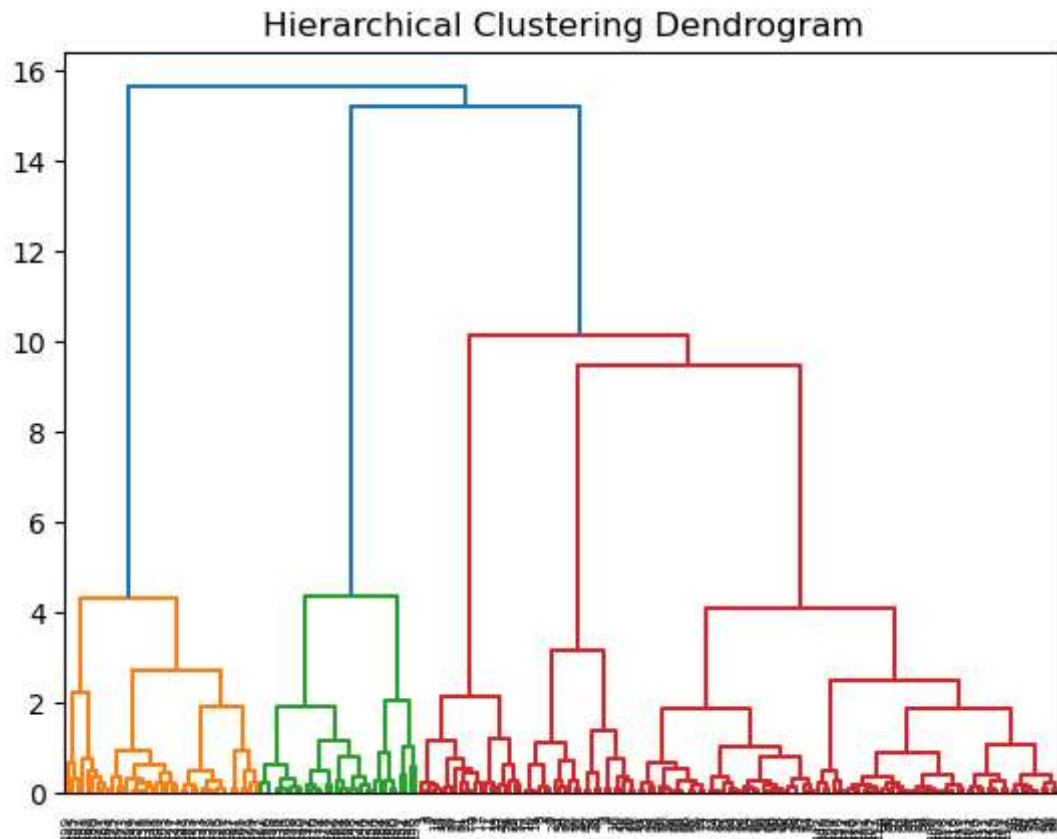
Silhouette Score: 0.55

```
In [31]: # Visualize the clustering results
plt.scatter(X['Annual Income (k$)'], X['Spending Score (1-100)'], c=labels, cm=
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.title('Agglomerative Clustering')
plt.show()
```



```
In [32]: # Create a dendrogram for hierarchical clustering visualization
from scipy.cluster.hierarchy import linkage, dendrogram

linkage_matrix = linkage(data_scaled, method='ward')
dendrogram(linkage_matrix)
plt.title('Hierarchical Clustering Dendrogram')
plt.show()
```



Agglomerative code for "Basic1.csv"

```
In [33]: import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import AgglomerativeClustering
from sklearn.preprocessing import StandardScaler
from scipy.cluster.hierarchy import dendrogram
from sklearn.metrics import silhouette_score
```

```
In [34]: # Load the dataset
data = pd.read_csv('basic1.csv')
```

```
In [35]: # Standardize the data (important for hierarchical clustering)
scaler = StandardScaler()
data_scaled = scaler.fit_transform(data)
```

```
In [36]: # the number of clusters  
n_clusters = 5
```

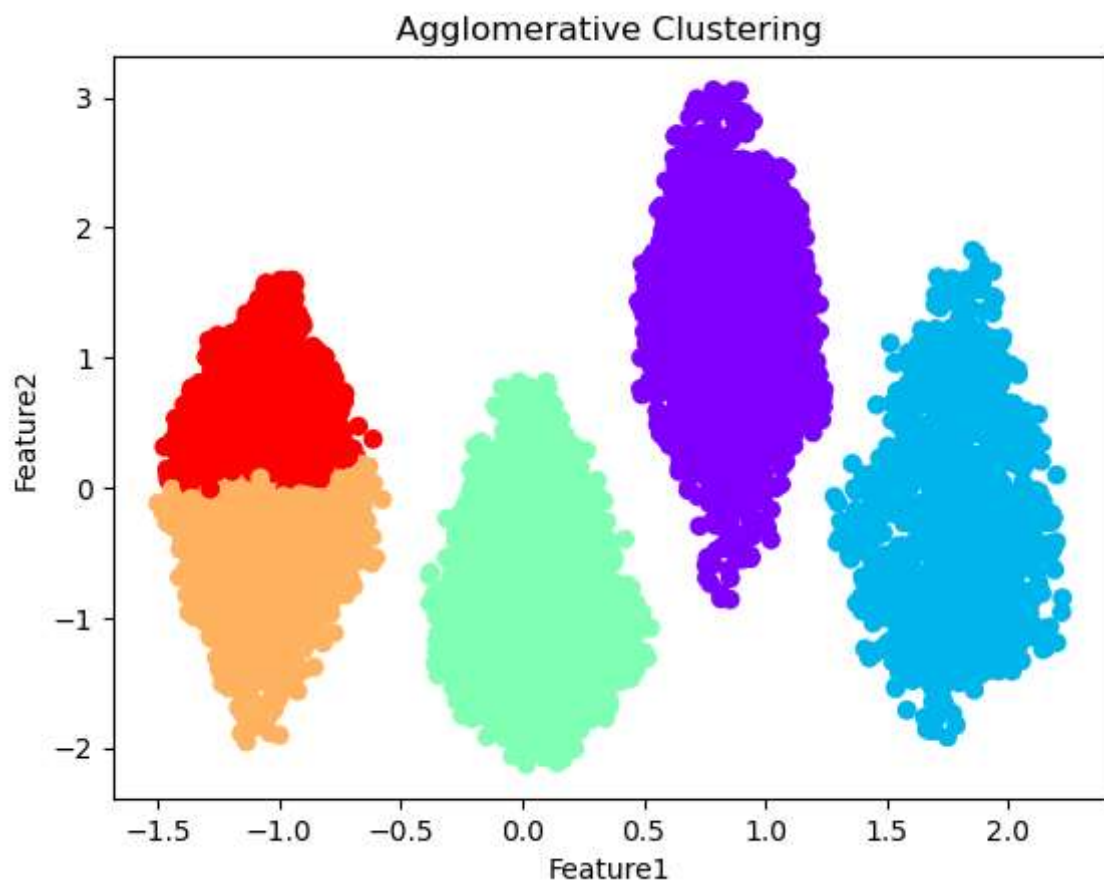
```
In [37]: # Perform Agglomerative Clustering  
agg_clustering = AgglomerativeClustering(n_clusters=n_clusters)  
labels = agg_clustering.fit_predict(data_scaled)
```

```
In [38]: # Calculate the silhouette score  
silhouette_avg = silhouette_score(data_scaled, labels)  
print(f"Silhouette Score: {silhouette_avg:.2f}")
```

Silhouette Score: 0.52

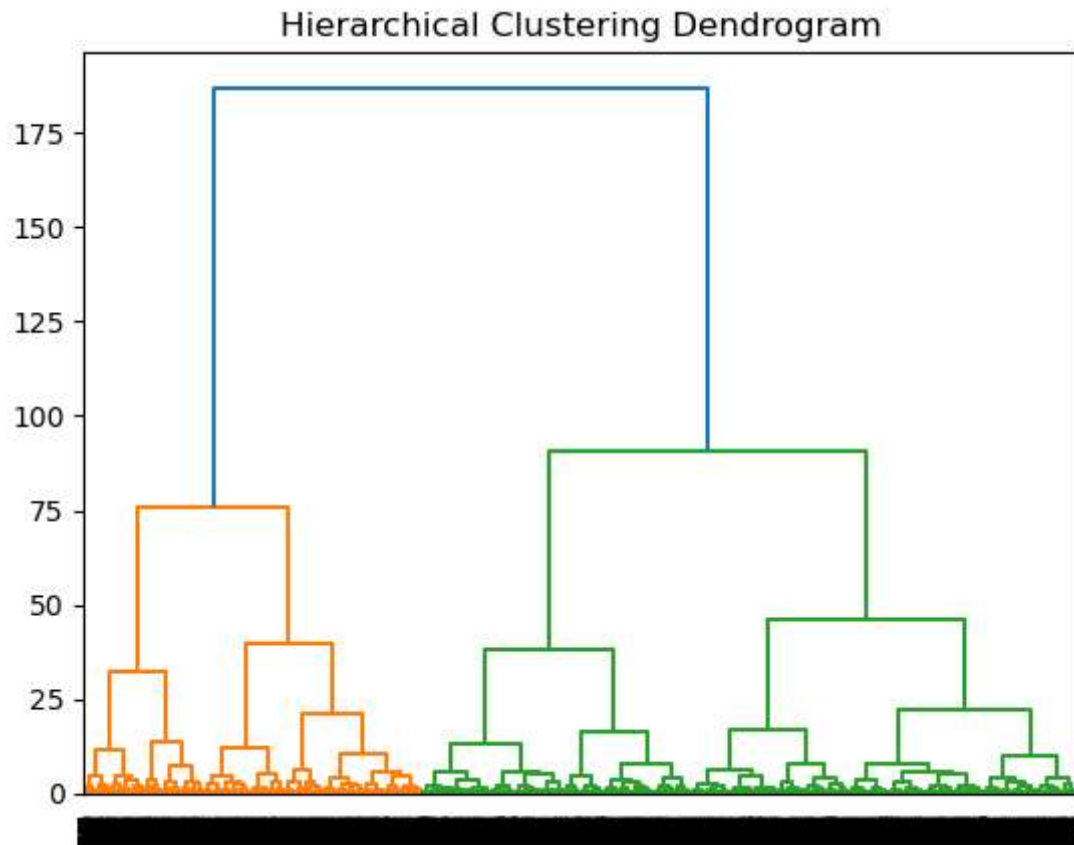
```
In [39]: # Visualize the clustering results (for two selected features, or customize as  
feature1 = 'Feature1' # Replace with an actual feature name  
feature2 = 'Feature2' # Replace with another actual feature name
```

```
In [40]: plt.scatter(data_scaled[:, 0], data_scaled[:, 1], c=labels, cmap='rainbow')  
plt.xlabel(feature1)  
plt.ylabel(feature2)  
plt.title('Agglomerative Clustering')  
plt.show()
```



```
In [41]: # Create a dendrogram for hierarchical clustering visualization
from scipy.cluster.hierarchy import linkage, dendrogram

linkage_matrix = linkage(data_scaled, method='ward')
dendrogram(linkage_matrix)
plt.title('Hierarchical Clustering Dendrogram')
plt.show()
```



In []: