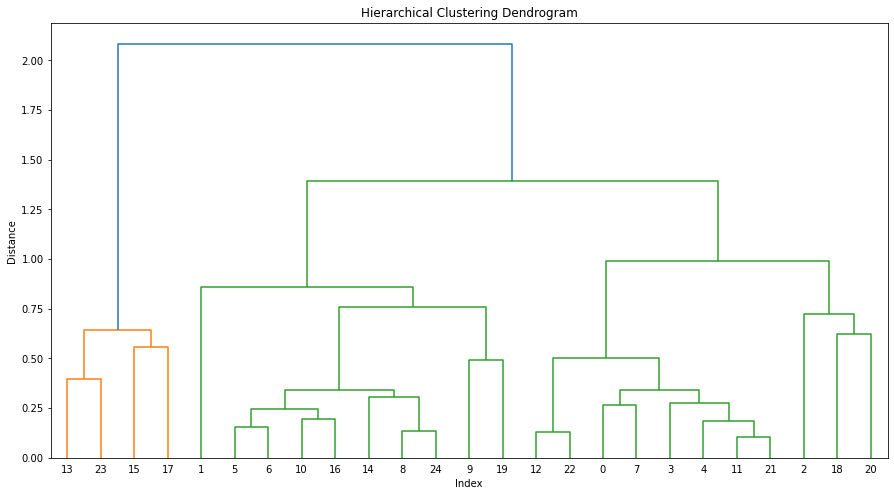
**Code:**

import pandas as pd  
import matplotlib.pylab as plt  
  
Univ1 = pd.read\_excel("C:\\Users\cse-18\Downloads\University\_Clustering.xlsx")  
  
Univ1.describe()  
Univ1.info()  
  
Univ = Univ1.drop(["State"], axis=1)  
  
# Normalization function  
def norm\_func(i):  
    x = (i-i.min()) / (i.max()-i.min())  
    return (x)  
  
# Normalized data frame (considering the numerical part of data)  
df\_norm = norm\_func(Univ.iloc[:, 1:])  
df\_norm.describe()  
  
# for creating dendrogram  
from scipy.cluster.hierarchy import linkage  
import scipy.cluster.hierarchy as sch  
  
z = linkage(df\_norm, method = "complete", metric = "euclidean")  
  
# Dendrogram  
plt.figure(figsize=(15, 8));plt.title('Hierarchical Clustering Dendrogram');plt.xlabel('Index');plt.ylabel('Distance')  
sch.dendrogram(z,  
    leaf\_rotation = 0,  # rotates the x axis labels  
    leaf\_font\_size = 10 # font size for the x axis labels  
)  
plt.show()  
  
  
# Now applying AgglomerativeClustering choosing 5 as clusters from the above dendrogram  
from sklearn.cluster import AgglomerativeClustering  
  
h\_complete = AgglomerativeClustering(n\_clusters = 3, linkage = 'complete', affinity = "euclidean").fit(df\_norm)  
h\_complete.labels\_  
  
cluster\_labels = pd.Series(h\_complete.labels\_)  
  
Univ['clust'] = cluster\_labels # creating a new column and assigning it to new column  
  
Univ1 = Univ.iloc[:, [7,0,1,2,3,4,5,6]]  
Univ1.head()  
  
# Aggregate mean of each cluster  
Univ1.iloc[:, 2:].groupby(Univ1.clust).mean()  
  
# creating a csv file  
Univ1.to\_csv("University.csv", encoding = "utf-8")  
  
import os  
os.getcwd()

**Output:**

****

**Hierarchical Clustering Dendrogram**