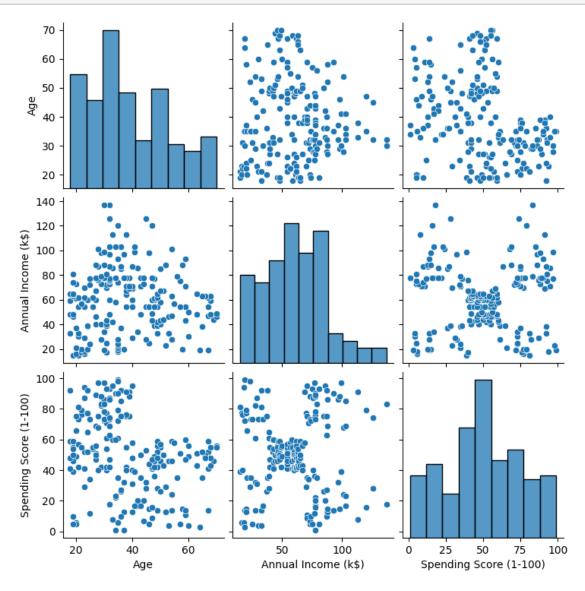
hierarchial

December 27, 2023

```
[1]: import pandas as pd
     import matplotlib.pyplot as plt
     import seaborn as sns
     from sklearn.cluster import AgglomerativeClustering
     from sklearn import preprocessing
     from scipy.cluster.hierarchy import dendrogram,linkage
[2]: df=pd.read_csv('/content/drive/MyDrive/Mall_Customers.csv')
[3]: df
[3]:
          CustomerID
                       Gender
                                     Annual Income (k$)
                                                          Spending Score (1-100)
                               Age
                         Male
                                 19
                                                      15
                         Male
     1
                    2
                                 21
                                                      15
                                                                                81
     2
                    3 Female
                                 20
                                                      16
                                                                                 6
     3
                    4
                       Female
                                 23
                                                                                77
                                                      16
     4
                    5
                       Female
                                 31
                                                      17
                                                                                40
     195
                                 35
                                                     120
                                                                                79
                  196
                       Female
     196
                       Female
                                 45
                                                                                28
                  197
                                                     126
                                                                                74
     197
                  198
                         Male
                                 32
                                                     126
     198
                  199
                         Male
                                 32
                                                     137
                                                                                18
     199
                  200
                         Male
                                 30
                                                     137
                                                                                83
     [200 rows x 5 columns]
[4]: df.head()
[4]:
        CustomerID
                     Gender
                             Age
                                   Annual Income (k$)
                                                        Spending Score (1-100)
                       Male
                               19
                                                    15
                                                                              39
                  2
                       Male
                               21
                                                    15
                                                                              81
     1
                  3 Female
     2
                               20
                                                    16
                                                                              6
     3
                  4 Female
                                                                              77
                               23
                                                    16
                    Female
                               31
                                                    17
                                                                              40
[5]: df.describe()
```

[5]: CustomerID Annual Income (k\$) Spending Score (1-100) Age 200.000000 200.000000 200.000000 200.000000 count 100.500000 38.850000 60.560000 50.200000 mean std 57.879185 13.969007 26.264721 25.823522 18.000000 min 1.000000 15.000000 1.000000 25% 50.750000 28.750000 41.500000 34.750000 100.500000 36.000000 50% 61.500000 50.000000 75% 150.250000 49.000000 78.000000 73.000000 200.000000 70.000000 137.000000 99.000000 max

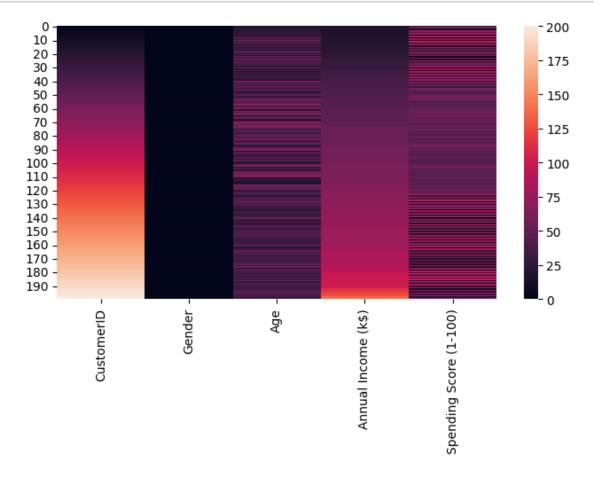
[6]: sns.pairplot(df[['Age','Annual Income (k\$)','Spending Score (1-100)']])
plt.show()



```
[7]: label_encoder=preprocessing.LabelEncoder()
    df['Gender']=label_encoder.fit_transform(df['Gender'])
    df.head()
```

```
[7]:
         CustomerID
                                     Annual Income (k$)
                                                             Spending Score (1-100)
                      Gender
                                Age
                   1
                            1
                                 19
                                                        15
                                                                                    39
                   2
     1
                            1
                                 21
                                                        15
                                                                                    81
     2
                   3
                                                                                     6
                            0
                                 20
                                                        16
     3
                   4
                            0
                                 23
                                                        16
                                                                                    77
                   5
                            0
                                 31
                                                        17
                                                                                    40
```

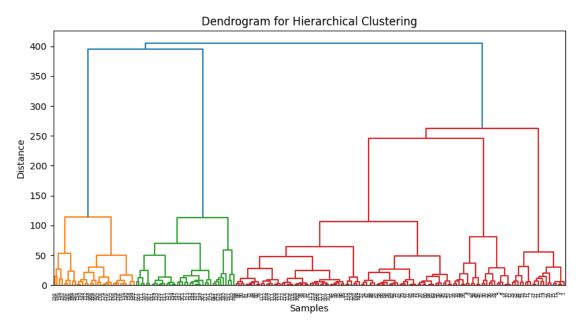
```
[8]: plt.figure(1,figsize= (8,4))
sns.heatmap(df)
plt.show()
```



```
[11]: x=df[['Annual Income (k$)','Spending Score (1-100)']]
```

[12]: linked=linkage(x,method='ward')

```
[13]: plt.figure(figsize=(10,5))
  dendrogram(linked,orientation='top',distance_sort='descending',show_leaf_counts=True)
  plt.title('Dendrogram for Hierarchical Clustering')
  plt.xlabel('Samples')
  plt.ylabel('Distance')
  plt.show()
```



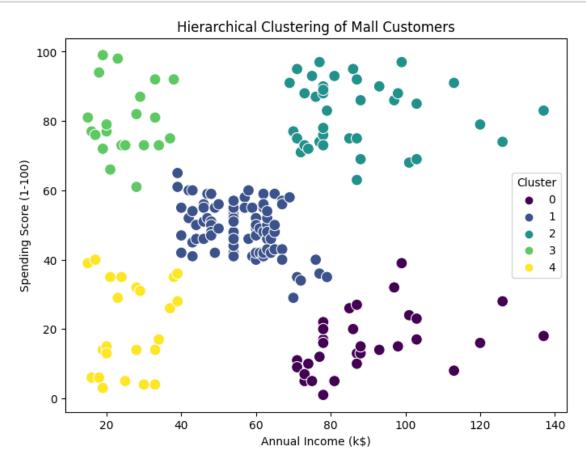
```
[14]: hc=AgglomerativeClustering(n_clusters=5,affinity='euclidean',linkage='ward')
y_pred=hc.fit_predict(x)
```

/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_agglomerative.py:983:
FutureWarning: Attribute `affinity` was deprecated in version 1.2 and will be removed in 1.4. Use `metric` instead warnings.warn(

```
[15]: df['Cluster'] = y_pred
y_pred
```

```
0, 2])
```

```
[16]: plt.figure(figsize=(8, 6))
sns.scatterplot(x='Annual Income (k$)', y='Spending Score (1-100)',
data=df,hue='Cluster', palette='viridis', s=100)
plt.title('Hierarchical Clustering of Mall Customers')
plt.show()
```



```
[18]: from sklearn import metrics
silhouette = metrics.silhouette_score(x, y_pred)
print(f"Silhouette Score: {silhouette}")
```

Silhouette Score: 0.5529945955148897