hierarchical

April 3, 2022

1 Hierarical Clustring (Agglomerative-single Linkage)

1.0.1 Importing packages

[ ]:
```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
from scipy.cluster.hierarchy import dendrogram
from sklearn.cluster import AgglomerativeClustering
```

1.1 Reading the dateset & Renaming columns lables

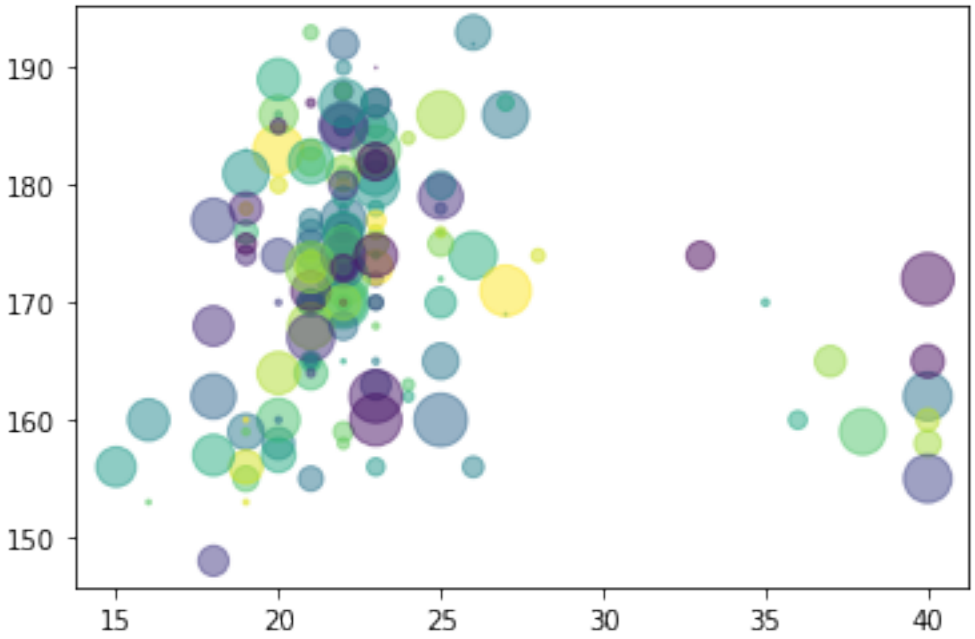
[ ]:
```
col_names = ['timeStamp', 'Gender', 'Grade', 'Age', 'Length', 'Weight',␣
 ↪'ShoesSize']
dataframe = pd.read_csv("../human_features.csv", names = col_names,␣
 ↪skiprows=(0, ))
```

1.2 Visualizing the data

[ ]:
```
x = dataframe.loc[:, 'Age']
y = dataframe.loc[:, 'Length']
```

[ ]:
```
n = x.shape[0]
colors = np.random.rand(n)
size = pow(20 * np.random.rand(n), 2)
plt.scatter(x, y, s=size, c=colors, alpha=0.5)
```

[ ]:
```
<matplotlib.collections.PathCollection at 0x7f422a9afdc0>
```



1.3 Creating Agglomerative Clusting model

[ ]:
```
cluster_model = AgglomerativeClustering(distance_threshold=0, n_clusters=None)
```

1.3.1 Training the model on the data

[ ]:
```
clustring = cluster_model.fit(dataframe.loc[:, ['Age', 'Length']])
```

[ ]:
```
clustring.labels_
```

[ ]:
```
array([120, 125, 157, 156, 155, 118, 153, 117, 152, 154, 116, 151, 150,
       115, 134, 148,  58,  86, 114, 128, 143, 142, 111, 112, 144, 147,
       110, 107, 133, 106,  76, 149, 131,  57,  91, 138, 113,  78, 105,
       132, 104, 108,  99, 139,  89,  85, 103,  71, 123,  55, 141,  70,
        80,  77, 137, 122,  97,  87,  84,  56,  68,  92, 126,  51, 109,
        53, 140,  45, 146, 101,  27,  52, 145,  83, 121,  28,  82, 136,
       119,  93,  90,  41,  95,  81,  62,  25,  54, 135,  75,  79, 102,
        42,  67, 124,  94,  98,  26, 127,  37,  61,  63,  96,  46, 129,
       100,  39, 130,  40,  60,  20,  30,  73,  88,  69,  59,  47,  72,
        74,  49,  66,  36,  44,  29,  50,  24,  13,  64,  65,  48,  43,
        38,  21,  34,  23,  14,  33,  31,  22,  32,  19,  15,  11,  16,
        35,  12,  10,   6,  18,   9,   7,   5,   4,   2,  17,   8,   3,
         1,   0])
```

[ ]:
```
clustring.labels_.shape
```

[ ]: (158,)

[ ]:
```
clustring.children_
```

[ ]:
```
array([[  2,  37],
       [  3,  71],
       [  4,  39],
       [  9,  91],
       [  6,  19],
       [  8,  57],
       [ 11,  30],
       [ 12,  14],
       [ 31, 165],
       [ 15, 105],
       [111, 112],
       [ 68,  73],
       [ 72, 154],
       [ 24, 162],
       [ 20, 107],
       [ 21, 115],
       [ 50, 164],
       [113, 174],
       [ 66, 146],
       [ 35, 139],
       [ 60, 143],
       [ 77, 159],
       [142, 179],
       [157, 180],
       [ 28,  83],
       [138, 166],
       [ 32,  76],
       [126, 172],
       [103, 158],
       [156, 183],
       [100, 132],
       [ 62, 148],
       [ 84, 182],
       [ 99, 190],
       [ 93, 114],
       [ 55,  86],
       [108, 150],
       [  0,  25],
       [ 78, 116],
       [  5,  17],
       [  7, 131],
       [ 10,  34],
       [ 13,  42],
```

```
[ 18,  22],
[ 36, 136],
[ 23,  59],
[ 49,  94],
[ 26,  56],
[ 64, 137],
[ 65,  97],
[ 27,  33],
[ 29,  52],
[ 38,  74],
[ 40,  47],
[ 46,  48],
[ 90, 118],
[ 69, 110],
[104, 109],
[152, 153],
[ 95, 117],
[141, 168],
[133, 170],
[ 82, 178],
[102, 192],
[ 79, 163],
[ 61, 193],
[ 67, 171],
[ 80, 175],
[ 44, 181],
[145, 224],
[173, 176],
[203, 204],
[ 45,  89],
[ 58, 106],
[161, 197],
[167, 216],
[177, 199],
[188, 211],
[206, 228],
[147, 189],
[ 53, 184],
[160, 187],
[ 98, 207],
[205, 225],
[208, 226],
[186, 223],
[169, 218],
[ 51,  81],
[ 43, 130],
[ 54,  92],
```

```
[ 87,  96],
[119, 121],
[140, 196],
[194, 233],
[219, 232],
[  1,  16],
[212, 213],
[209, 210],
[122, 127],
[200, 241],
[ 75, 235],
[185, 220],
[ 70, 215],
[191, 238],
[ 41,  63],
[ 85, 214],
[144, 151],
[123, 129],
[124, 125],
[202, 250],
[101, 198],
[195, 257],
[246, 248],
[201, 245],
[227, 239],
[230, 258],
[236, 259],
[221, 247],
[242, 274],
[231, 268],
[ 88, 155],
[120, 264],
[229, 263],
[135, 253],
[244, 262],
[249, 256],
[149, 255],
[234, 252],
[222, 251],
[237, 254],
[261, 286],
[272, 284],
[266, 283],
[271, 281],
[128, 278],
[243, 280],
[267, 279],
```

```
       [240, 285],
       [217, 277],
       [269, 287],
       [270, 275],
       [273, 295],
       [260, 288],
       [134, 294],
       [265, 290],
       [282, 291],
       [276, 297],
       [301, 303],
       [296, 300],
       [298, 304],
       [289, 293],
       [292, 302],
       [299, 306],
       [307, 308],
       [305, 309],
       [311, 312],
       [310, 313]])
```

[ ]:
```
clustring.children_.shape
```

[ ]:
```
(157, 2)
```

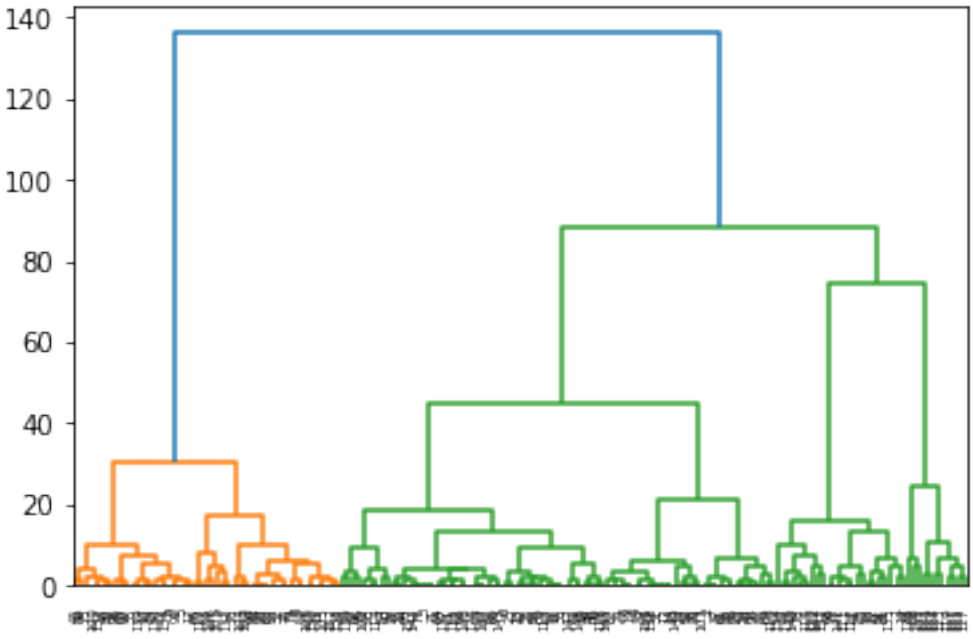
[ ]:
```
def plot_dendrogram(model, **kwargs):
    # Create linkage matrix and then plot the dendrogram

    # create the counts of samples under each node
    counts = np.zeros(model.children_.shape[0])
    n_samples = len(model.labels_)
    for i, merge in enumerate(model.children_):
        current_count = 0
        for child_idx in merge:
            if child_idx < n_samples:
                current_count += 1  # leaf node
            else:
                current_count += counts[child_idx - n_samples]
        counts[i] = current_count

    linkage_matrix = np.column_stack(
        [model.children_, model.distances_, counts]
    ).astype(float)

    # Plot the corresponding dendrogram
    dendrogram(linkage_matrix, **kwargs)
```

[ ]:
```
plot_dendrogram(clustring)
```



[ ]:
```
plot_dendrogram(clustring, truncate_mode="level", p=3 )
```