M11-L1-P3

November 26, 2023

0.1 M11-L1 Problem 3

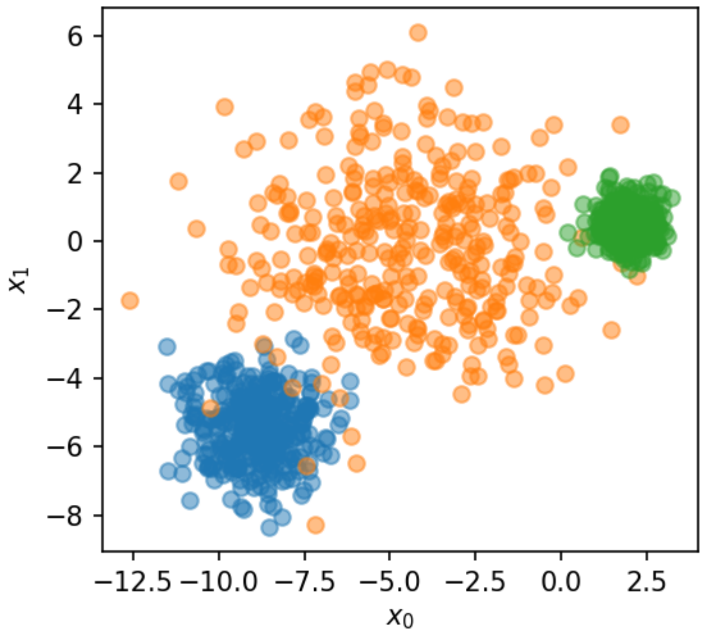
In this problem you will use the `sklearn` implementation of hierarchical clustering with three different linkage criteria (`'single'`, `'complete'`, `'average'`) to clusters two datasets: a “blob” shaped dataset with three classes, and a concentric circle dataset with two classes.

[ ]:
```
import numpy as np
import matplotlib.pyplot as plt

from sklearn.datasets import make_blobs, make_circles
from sklearn.cluster import AgglomerativeClustering

## DO NOT MODIFY
def plotter(x, labels = None, ax = None, title = None):
    if ax is None:
        _, ax = plt.subplots(dpi = 150, figsize = (4,4))
        flag = True
    else:
        flag = False
    for i in range(len(np.unique(labels))):
        ax.scatter(x[labels == i, 0], x[labels == i, 1], alpha = 0.5)
    ax.set_xlabel('$x_0$')
    ax.set_ylabel('$x_1$')
    ax.set_aspect('equal')
    if title is not None:
        ax.set_title(title)
    if flag:
        plt.show()
    else:
        return ax
```

First we will consider the “blob” dataset, generated below. Visualize the data using the provided `plotter(x, labels)` function.

[ ]:
```
## DO NOT MODIFY
x, labels = make_blobs(n_samples = 1000, cluster_std=[1.0, 2.5, 0.5],␣
 ↪random_state = 170)
```

[ ]:
```
plotter(x, labels)
```



Using the `AgglomerativeClustering()` function, generate 3 side-by-side plots using `plt.subplots()` and the provided `plotter(x, labels, ax, title)` function to visualize the results of the following three linkage criteria `['single', 'complete', 'average']`.

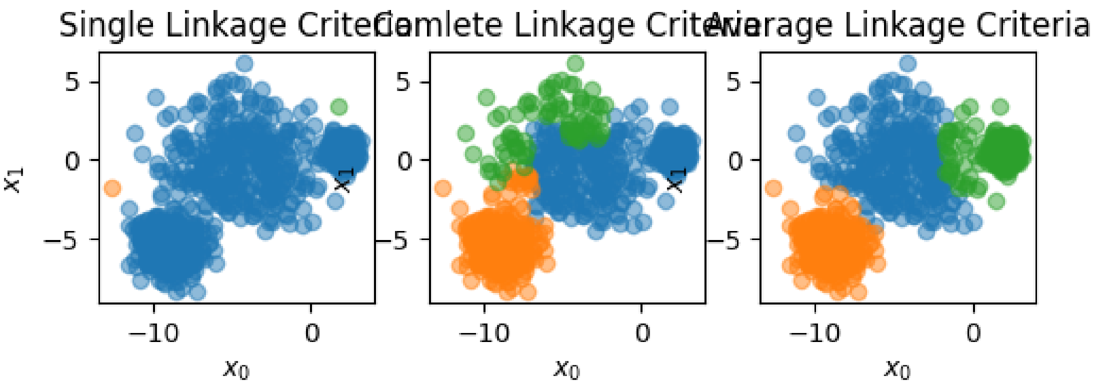
Note: the `plt.subplots()` function will return `fig, ax`, where `ax` is an array of all the subplot axes in the figure. Each individual subplot can be accessed with `ax[i]` which you can then pass to the `plotter()` function's `ax` argument.

[ ]:

```
fig, ax = plt.subplots(1,3)
model1 = AgglomerativeClustering(n_clusters=3,linkage='single').fit(x)
plotter(x, model1.labels_, ax[0], "Single Linkage Criteria")

model2 = AgglomerativeClustering(n_clusters=3,linkage='complete').fit(x)
plotter(x, model2.labels_, ax[1], "Comlete Linkage Criteria")

model3 = AgglomerativeClustering(n_clusters=3,linkage='average').fit(x)
plotter(x, model3.labels_, ax[2], "Average Linkage Criteria")
```

[ ]:

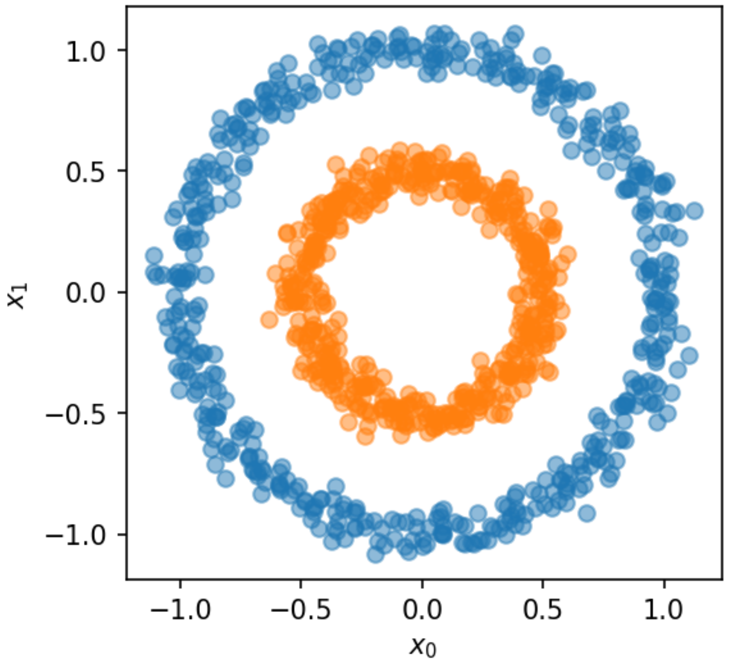
```
<Axes: title={'center': 'Average Linkage Criteria'}, xlabel='$x_0$',
ylabel='$x_1$'>
```



Now we will work on the concentric circle dataset, generated below. Visualize the data using the provided `plotter(x, labels)` function.

[ ]:
```
## DO NOT MODIFY
x, labels = make_circles(1000, factor = 0.5, noise = 0.05, random_state = 0)
```

[ ]:
```
plotter(x, labels)
```



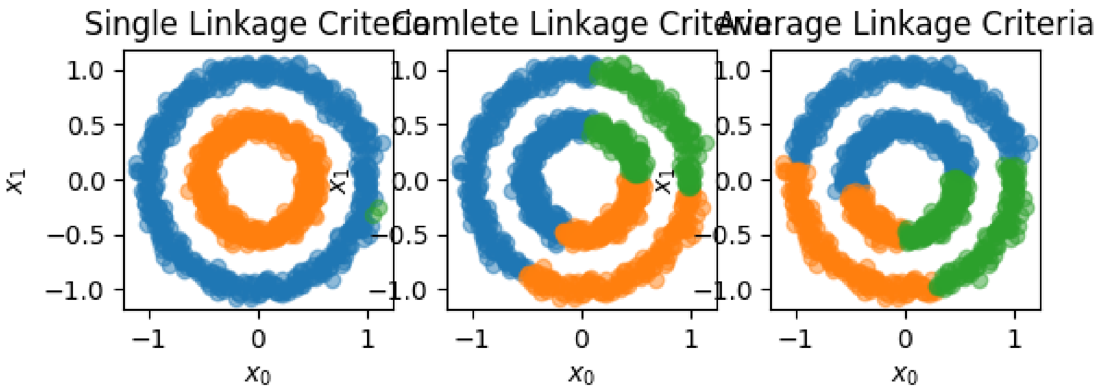
Again, use the `AgglomerativeClustering()` function to generate 3 side-by-side plots using `plt.subplots()` and the provided `plotter(x, labels, ax, title)` function to visualize the results of the following three linkage criteria `['single', 'complete', 'average']` for the concentric circle dataset.

[ ]:

```
fig, ax = plt.subplots(1,3)
model1 = AgglomerativeClustering(n_clusters=3,linkage='single').fit(x)
plotter(x, model1.labels_, ax[0], "Single Linkage Criteria")

model2 = AgglomerativeClustering(n_clusters=3,linkage='complete').fit(x)
plotter(x, model2.labels_, ax[1], "Comlete Linkage Criteria")

model3 = AgglomerativeClustering(n_clusters=3,linkage='average').fit(x)
plotter(x, model3.labels_, ax[2], "Average Linkage Criteria")
```

[ ]:

```
<Axes: title={'center': 'Average Linkage Criteria'}, xlabel='$x_0$',
ylabel='$x_1$'>
```



1 Discussion

Discuss the performance of the three different linkage criteria on the “blob” dataset, and then on the concentric circle dataset. Why do some linkage criteria perform better on one dataset, but worse on others?

The average linkage criteria performed the best on the blob dataset while the single linkage criteria performed best on the concectric dataset. Some linkage criteria perform better on different types of data because of the types of distances used and how they apply to the features of the data. For example, the blob data sets center around an average point and thus lend well to the average linkage criteria as demonstrated above.