Bhuvi Ghosh
60009210191

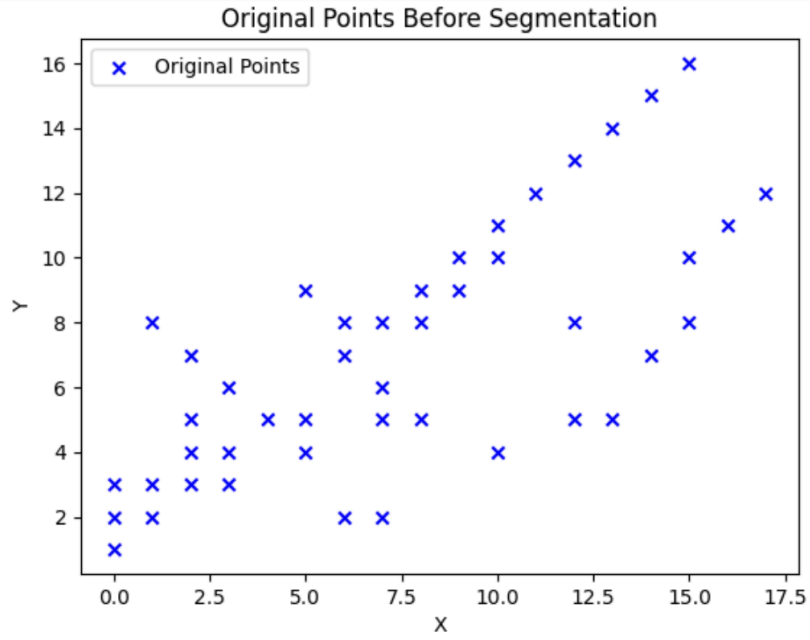
ML-IV Experiment-10

[1] 11s
```
import numpy as np
import matplotlib.pyplot as plt
from scipy.spatial.distance import cdist
from sklearn.cluster import AgglomerativeClustering
```

[13] 0s
```
points = np.array([
    [1, 2], [2, 3], [3, 4], [8, 8], [9, 9], [10, 10],
    [4, 5], [7, 6], [6, 7], [11, 12], [12, 13], [13, 14],
    [14, 15], [15, 16], [5, 5], [8, 9], [9, 10], [10, 11],
    [3, 6], [2, 7], [1, 8], [6, 8], [5, 9], [7, 8], [12, 8],
    [0, 1], [0, 2], [0, 3], [15, 10], [16, 11], [17, 12],
    [5, 4], [8, 5], [13, 5], [7, 2], [6, 2], [7, 5], [10, 4],
    [12, 5], [14, 7], [15, 8], [2, 5], [3, 3], [2, 4], [1, 3]
])
```

0s
```
plt.scatter(points[:, 0], points[:, 1], color='blue', label='Original Points', marker='x')
plt.title("Original Points Before Segmentation")
plt.xlabel("X")
plt.ylabel("Y")
plt.legend()
plt.show()
```

[14] 0s



[15] 0s
```
representatives = np.array([[1, 2], [2, 3], [8, 8], [9, 9]])
```

```
def shrink_representative_points(representatives, points, alpha=0.5):
    distances = cdist(representatives, points)
    closest_points = np.argmin(distances, axis=1)

    new_representatives = []
    for i, rep in enumerate(representatives):
        center = points[closest_points[i]]
        new_rep = alpha * center + (1 - alpha) * rep
        new_representatives.append(new_rep)

    return np.array(new_representatives)
representatives = shrink_representative_points(representatives, points)
```

```
[20] clustering = AgglomerativeClustering(n_clusters=3, linkage='single')
     labels = clustering.fit_predict(points)
```

```
plt.scatter(points[:, 0], points[:, 1], c=labels, cmap='viridis', marker='o')
plt.title("Clustering using CURE Algorithm")
plt.xlabel("X")
plt.ylabel("Y")
plt.show()
```