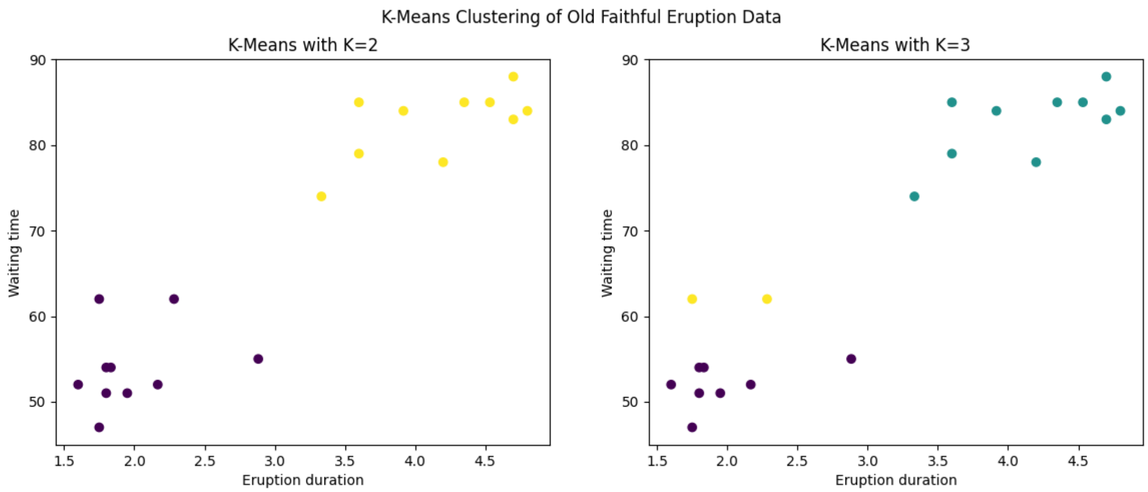
5.4

Left image - Average Entropy: 0.7898771282802609, Purity: 0.7272727272727273
Right image - Average Entropy: 1.0659908817276442, Purity: 0.4090909090909091

Code in folder

5.5



Code in folder

5.11a.

```
[0.49195951 0.50804049]
 [0.80463082 0.19536918]
 [0.74022639 0.25977361]
 [0.53901936 0.46098064]
 [0.67421456 0.32578544]
 [0.24226202 0.75773798]
 [0.79915876 0.20084124]
 [0.80126764 0.19873236]
 [0.21546454 0.78453546]
 [0.56464583 0.43535417]
 [0.82621164 0.17378836]
 [0.2948166  0.7051834 ]
 [0.43207498 0.56792502]
 [0.30341062 0.69658938]
 [0.82162591 0.17837409]
 [0.81173615 0.18826385]
 [0.19450526 0.80549474]
 [0.7749443  0.2250557 ]
 [0.17595207 0.82404793]
 [0.83238042 0.16761958]
```

Code in folder

5.11b.

Mean1: [2.57353054 62.37963562]
S1: [[1.01146201 12.50430356], [12.50430356 183.63335668]]

Mean2: [3.73738501 75.9378535]
S1: [[1.12176073 12.67931378], [12.67931378 166.13496871]]

T1: 0.5670253692542391
T2: 0.43297463074576087

Code in folder

5.13a.

Initial Means
[2.5, 65.0], [3.5, 70.0]

Cluster1:
Mean: 1.98160168, 54.00001977
Covariance: [0.12797041, 0.41133545], [0.41133545, 20.40047935]

Cluster2:
Mean: 4.17329924, 82.49999211
Covariance: [0.25380784, 1.26527997], [1.26527997, 15.85030896]

Code in folder

5.13b.

Initial Means
[2.0, 50.0], [4.0, 80.0]

Cluster1:
Mean: 1.98160168, 54.00001977
Covariance: [0.12797041, 0.41133545], [0.41133545, 20.40047934]

Cluster2:
Mean: 4.17329924, 82.49999211
Covariance: [0.25380784, 1.26527997], [1.26527997, 15.85030897]

Code in folder

5.13c.

Initial Means
[2.0, 60.0], [4.0, 80.0], [3.0, 70.0]

Cluster1:

Mean: 1.98160000, 54.00000001

Covariance: [0.12796804, 0.41130002], [0.41130002, 20.40000132]

Cluster2:

Mean: 4.26666666, 83.44444442

Covariance: [0.19483189, 0.52403706], [0.52403706, 8.69135928]

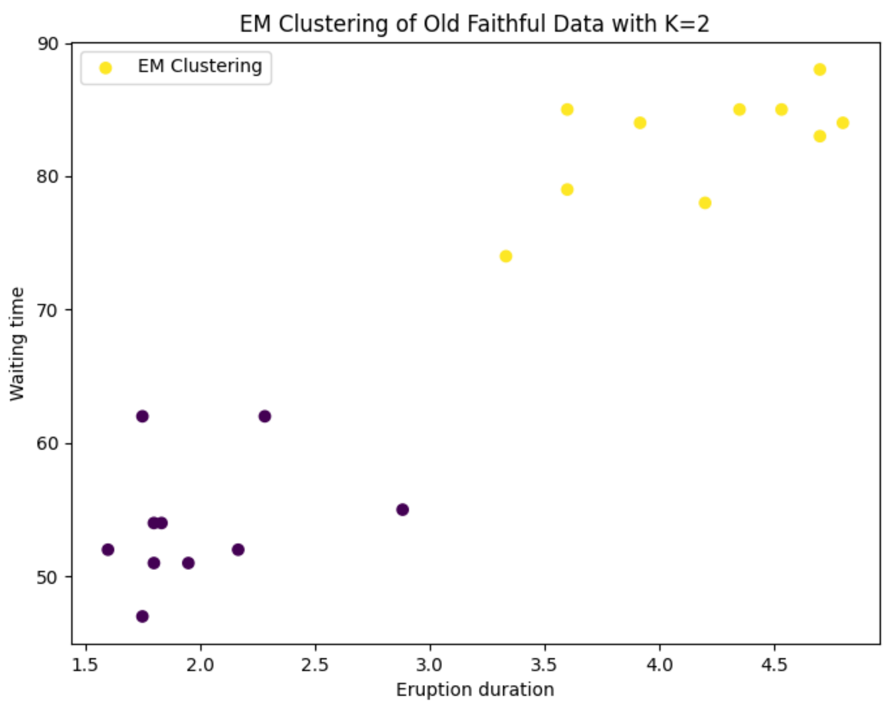
Cluster3:

Mean: 3.33300000, 74.00000000

Covariance: [0.00000100, 0.00000000], [0.00000000, 0.00000100]

Code in folder

5.13d.



EM provides a probabilistic model which can result in a more nuanced clustering when the underlying data distribution is more complex than what K-Means assumes.

Code in folder

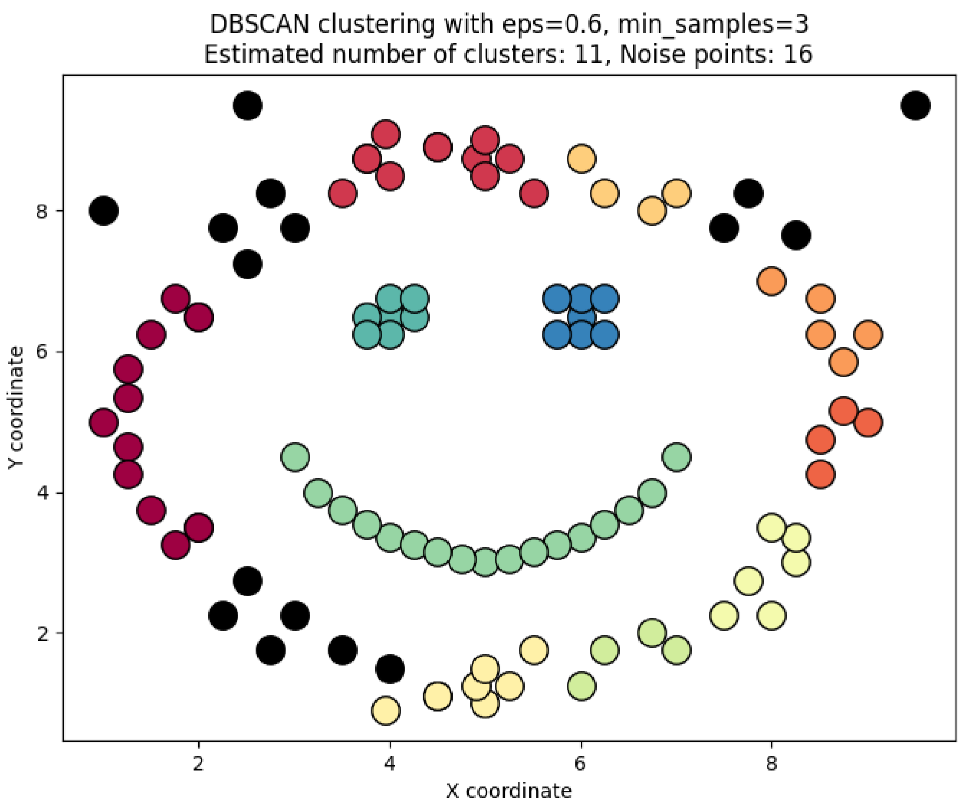
5.14a.

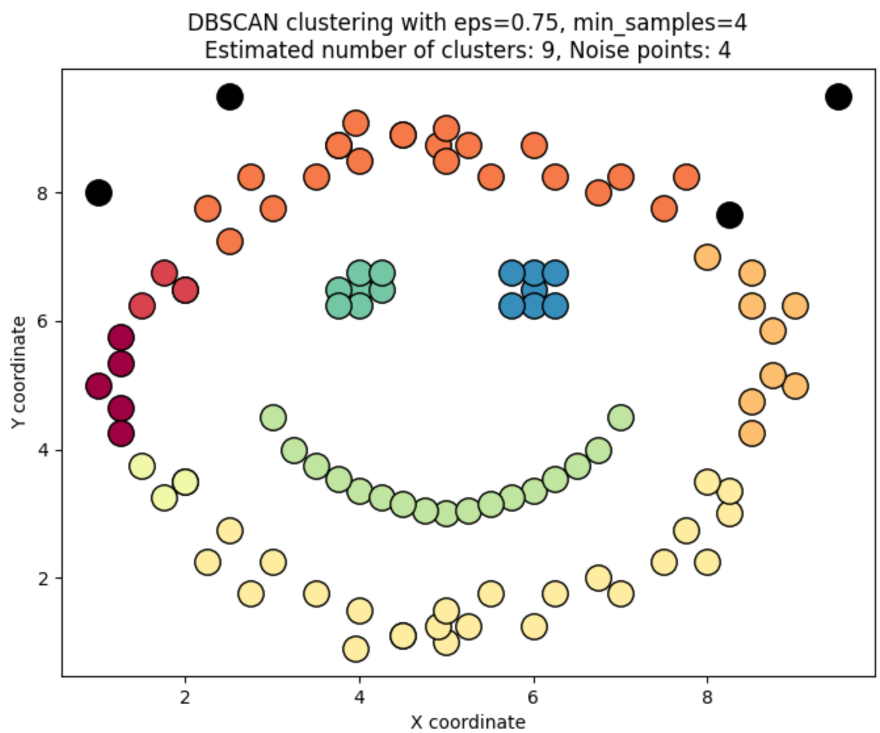
A previously visited point can become part of a cluster if it was not initially identified as a core point but later falls within the ε neighborhood of a new core point that is being explored. This can happen because when a point is first visited, it may not meet the minimum number of neighbors to be considered a core point. However, as the cluster grows and more points are added, the density around this point may increase, and it may become a part of the ε neighborhood of a different core point, thereby being included in a cluster.

5.14b.

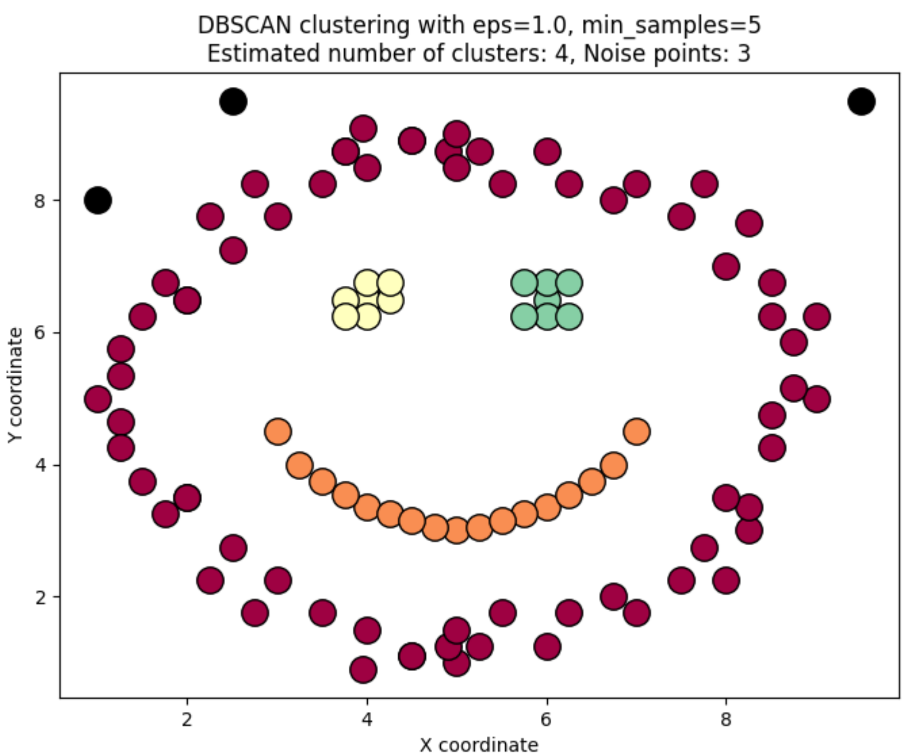
The DBSCAN algorithm is deterministic in terms of the final clusters formed for a given dataset and parameters only if all ties are broken consistently. In practice, this can be ensured by processing points in a specific fixed order (like the order of their indexes). However, if the order of processing is random, then the resulting clusters can indeed differ depending on the order in which points are selected and processed.

5.14c.

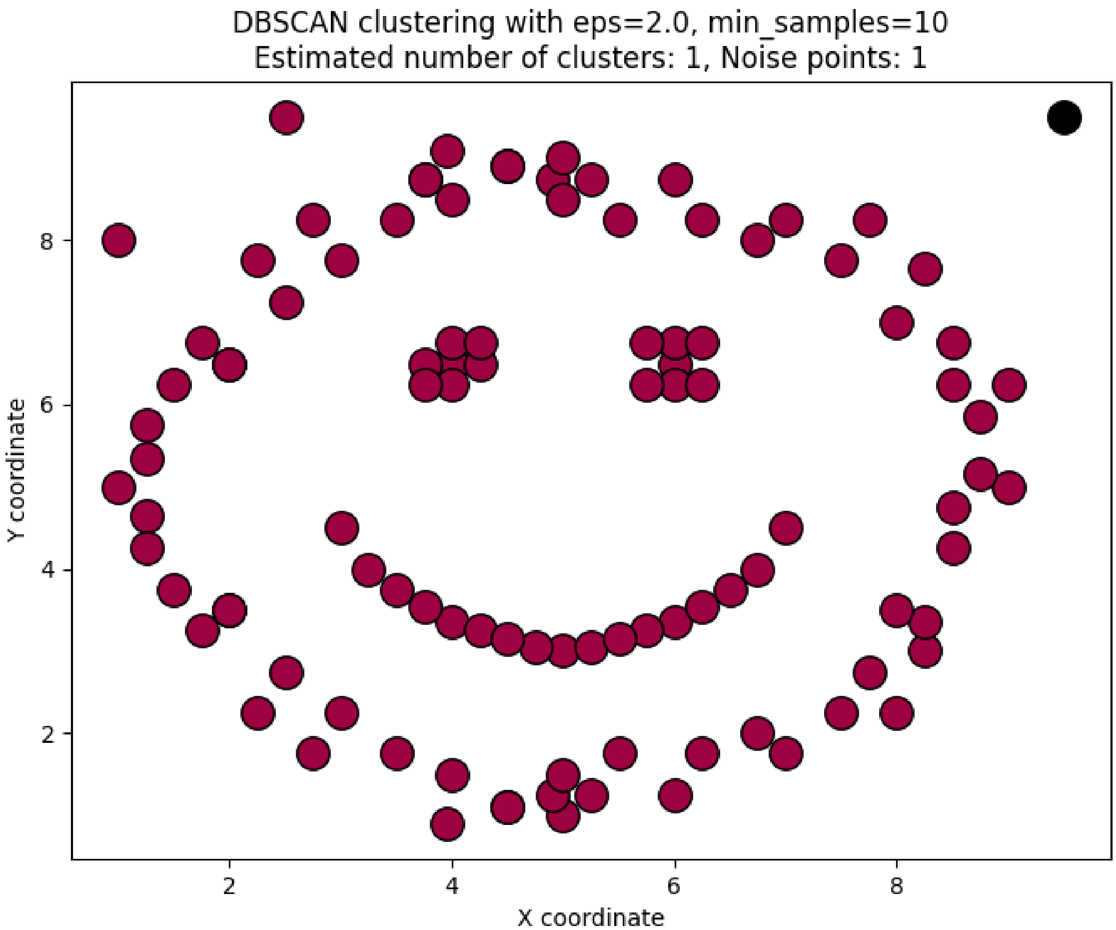




DBSCAN clustering with eps=0.75, min_samples=4
Estimated number of clusters: 9, Noise points: 4
Y coordinate
X coordinate
2
4
6
8
2
4
6
8



DBSCAN clustering with eps=1.0, min_samples=5
Estimated number of clusters: 4, Noise points: 3
Y coordinate
X coordinate
2
4
6
8
2
4
6
8


DBSCAN clustering with eps=2.0, min_samples=10
Estimated number of clusters: 1, Noise points: 1
Y coordinate
X coordinate
2
4
6
8
2
4
6
8