BT3041: Analysis and Interpretation of Biological Data

Assignment-1

Siddharth Betala

BE19B032

Question-1

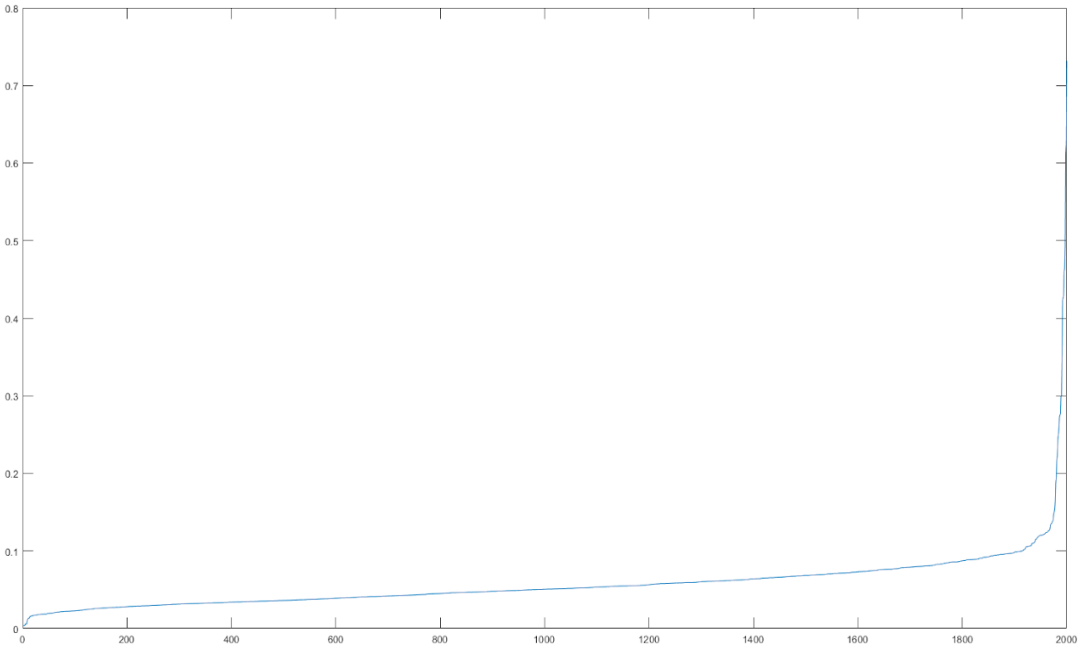
1) Explanation of the DBSCAN Algorithm:
    a) We create an algorithm called DBSCAN as given in the problem statement. Using this algorithm, we can cluster the points into separate clusters. The values of minimum points and the epsilon are not given and we need to find them.
    b) Firstly, points are classified into core points, border points, and noise as the algorithm creates a circle of radius epsilon around every data point.
    c) For each point, if its Euclidean distance from another point is less than epsilon, that is,
    If d < **ϵ**, then the neighbor counter for each of the two points is incremented by 1.
    d) A data point is called a core point if the circle of epsilon radius around it contains at least 'min_pts' number of points.
    If the number of points in this circle is less than 'min_pts', then the data point is called a border point, given that it has a core point within the circle.
    The rest are noise points.
    e) Going over every non-noise point (points that can be put into a cluster), a point is given a cluster label. All of its neighbours, which are found iteratively (neighbours of neighbours and so on) (density reachable points), are given the same cluster label.
    f) If a point has not been visited yet and is not a part of the current cluster label, the cluster label is increased by 1, and the next point not visited and labelled yet is given the same cluster label and the neighbours of it are found.
    g) The same process is repeated until all the core and border points (non-noise) have been labelled.

Corresponding code file: BE19B032_Q1_main.m

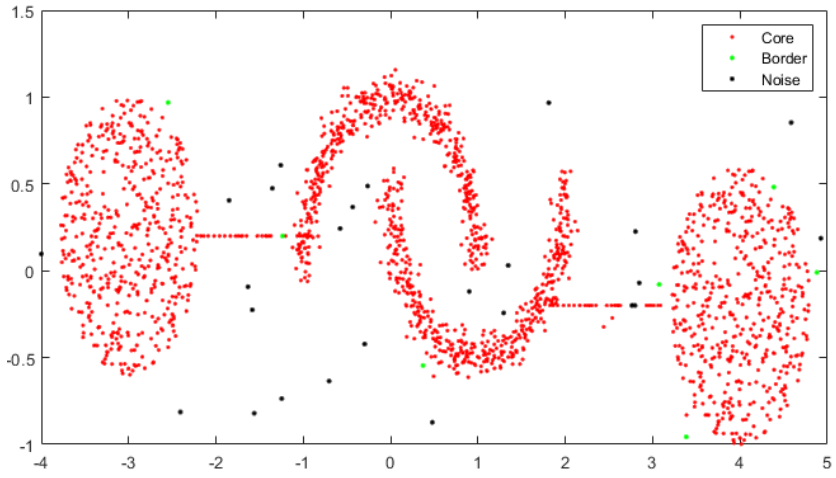
Dataset imported: dbscan2000.m

First I chose, the min_pts = 4 in accordance with the Introduction to Data Mining textbook, which states that for two-dimensional datasets a choice of four minimum points gives good results.

After this, I found an approximate value of epsilon using the 'approximating_eps_for_q1.m' file. We draw a graph of the distance of k-nearest neighbours from each point and then sort it, and after plotting it we obtain a graph as follows:



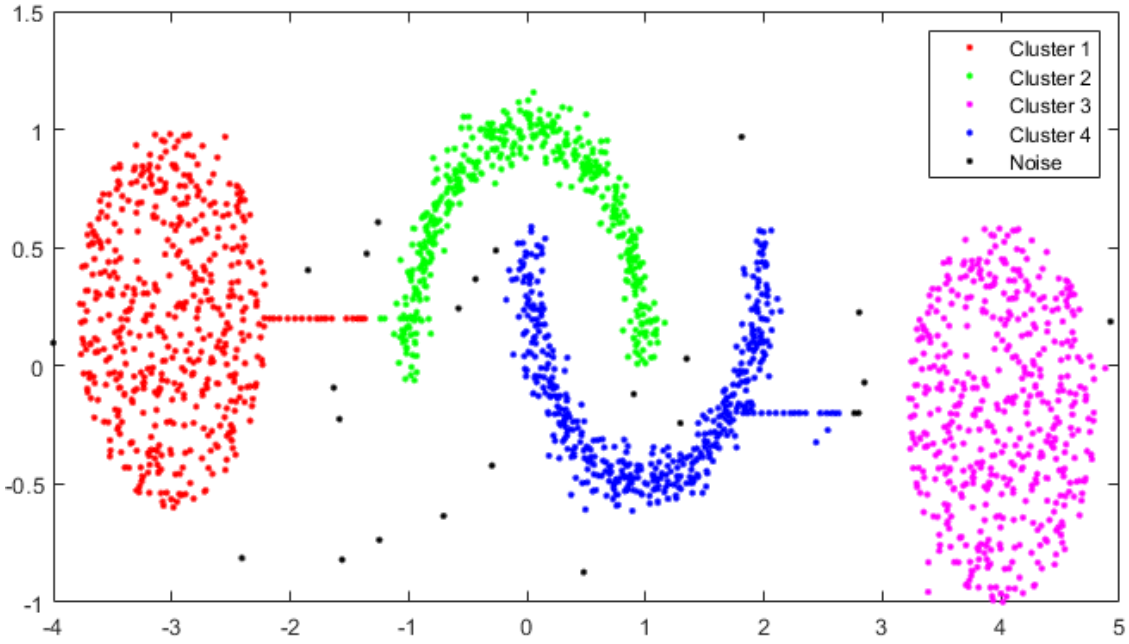
As it can be seen, the epsilon value can be approximated to be around **ϵ = 0.132,** around which the 'elbow bend' or the steep rise happens. For **min_pts = 4**, we get the following plot:



Plot showing the labelling of points as core, border, and noise

Type	Number of points
Core (red)	1968
Border (green)	7
Noise (black)	**25**

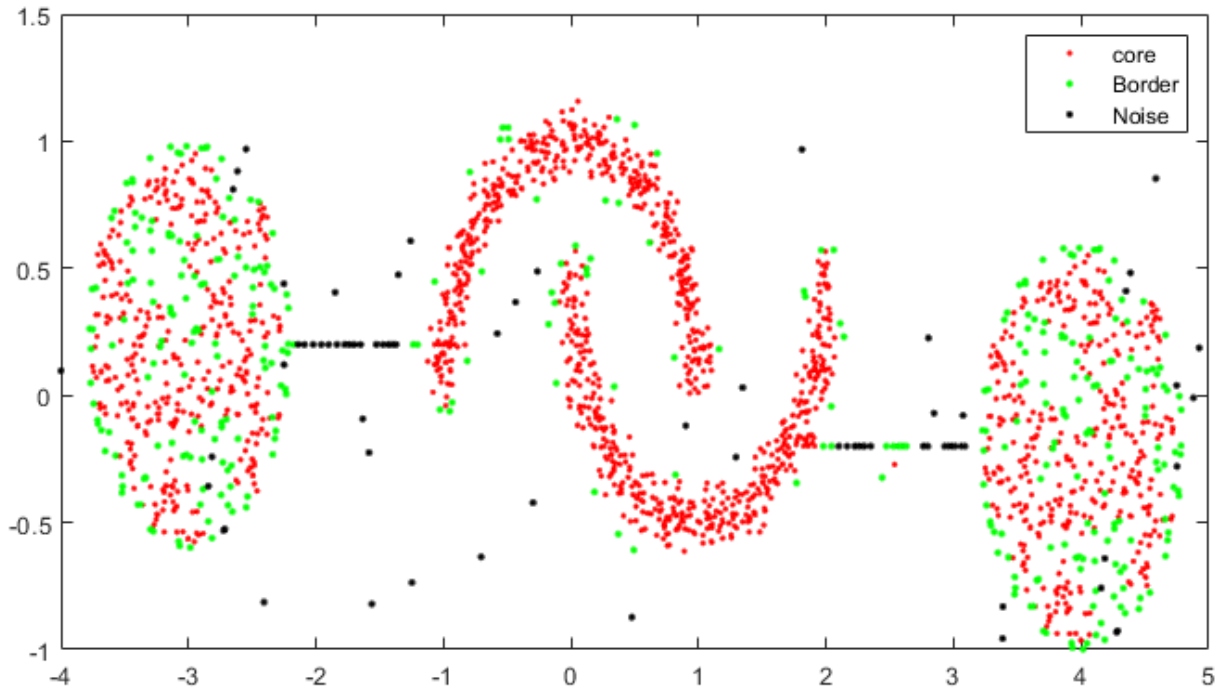
These parameters give us good and well-defined (to some extent) clusters as the following plot will show:



Plot showing the labelling of points into clusters after running the DBSCAN algorithm

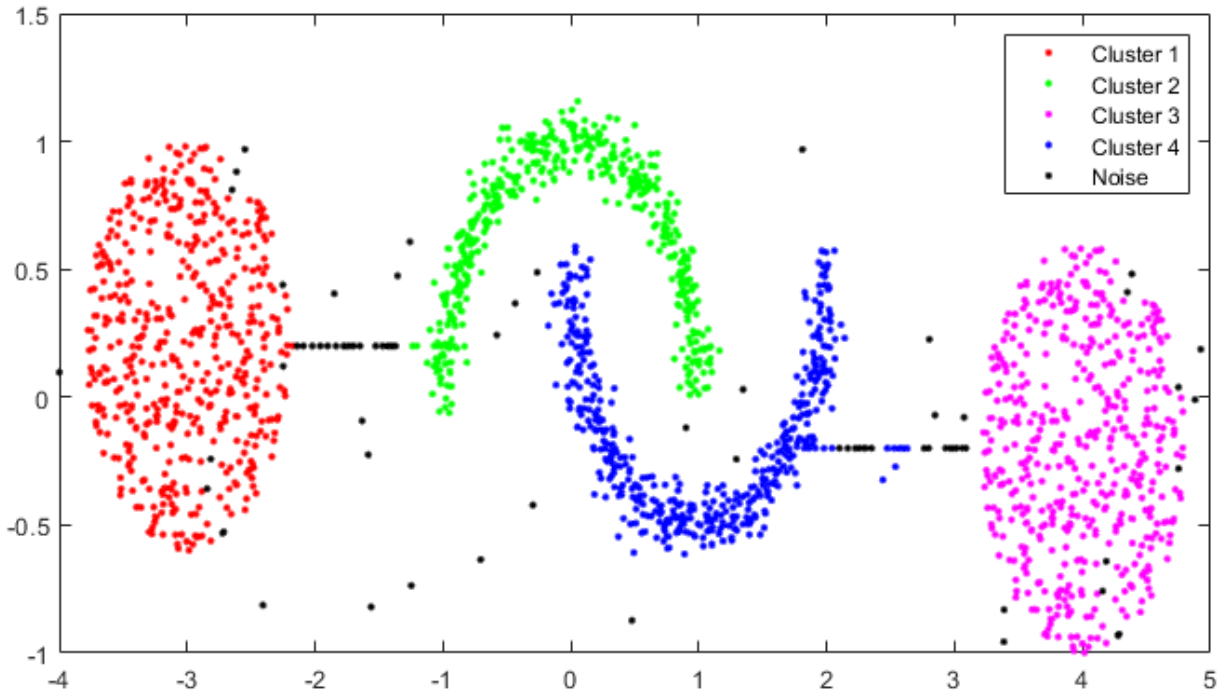
Cluster	Number of points
1 (red)	527
2 (green)	457
3 (magenta)	507
4 (dark blue)	477
Noise (black)	**25**

However, I wanted to look how the clusters would change if we varied the number of minimum points for the same epsilon radius. Here are the plots for **min_pts = 12 and ϵ = 0.132**, for example.



Plot showing the labelling of points as core, border, and noise

Type	Number of points
Core (red)	1576
Border (green)	344
Noise (black)	**80**



Plot showing the labeling of points into clusters after running the DBSCAN algorithm

It can be seen that this value of min_pts classifies some points well within the cluster as border points and noise. However, this value does classify the points on the line between two clusters as noise against when the value of min_pts was taken to be 4. Weighing the two inferences, we can say that the comment made in the Introduction to Data Mining book about using k = 4 for 2D datasets seems to make sense.

Cluster	Number of points
1 (red)	497
2 (green)	457
3 (magenta)	496
4 (dark blue)	470
Noise (black)	**80**

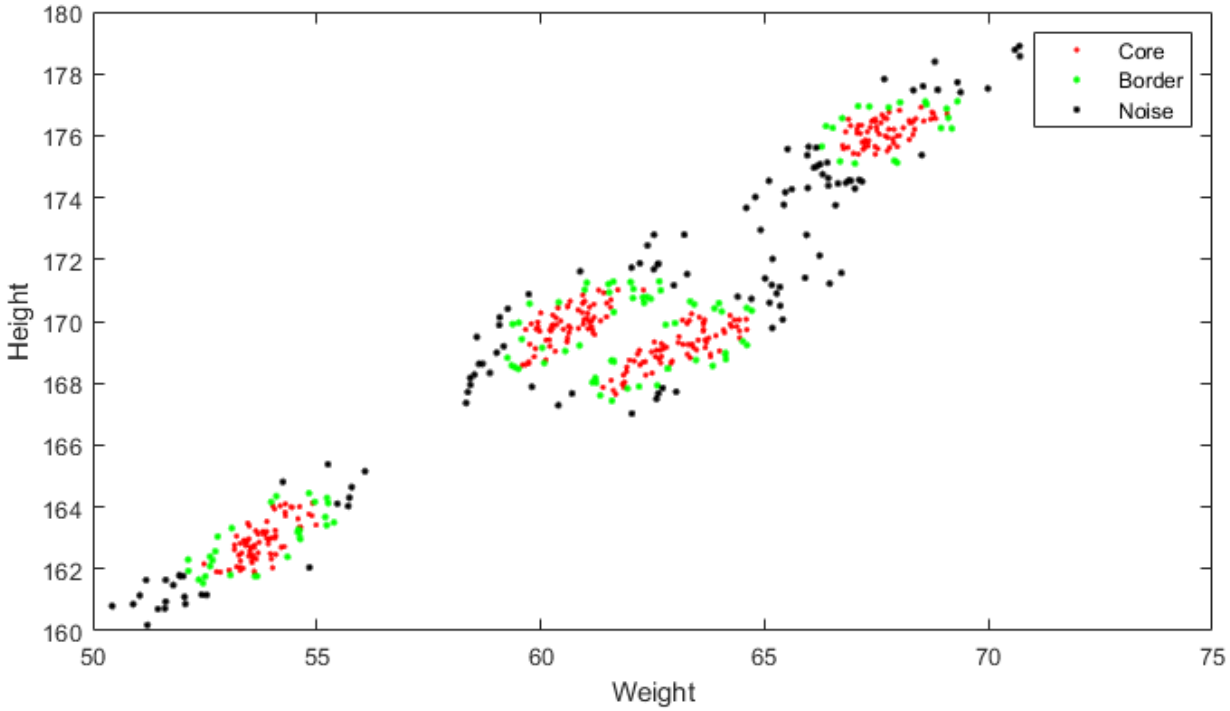
In either case, the number of noise points obtained either by classification of points into the categories of core, border, and noise or by clustering, is the same. This validates the correctness of the program.

Question-2

Algorithm was already explained in the answer to the previous question.

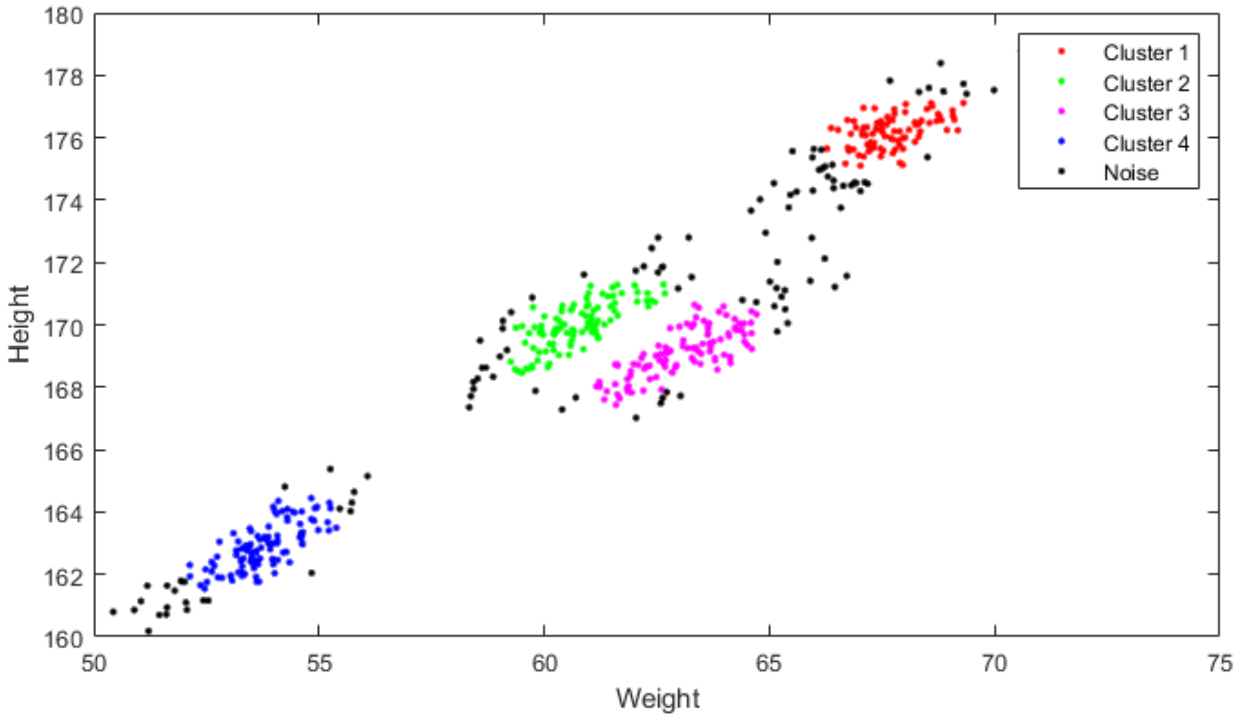
Given parameters: **k = 10, ϵ = 0.5**

Plots observed are as follows:



Plot showing the labelling of points as core, border, and noise

Type	Number of points
Core (red)	279
Border (green)	106
Noise (black)	**115**



Plot showing the labeling of points into clusters after running the DBSCAN algorithm

As, it can be seen that for these parameters, we get 4 clusters. The points labelled in black are noise.

Cluster	Number of points
1 (red)	85
2 (green)	98
3 (magenta)	102
4 (dark blue)	100
Noise (black)	**115**

Once again, the number of points classified as noise using through the two ways is the same.

Conclusions

1) Using the DBSCAN algorithm, alongwith the appropriate parameters, it is possible to get non-globular clusters and clusters of different shapes, which was not possible earlier with k-means clustering.
2) For two-dimensional datasets, k = 4 gives good results. However, for data that seems to be noisy, one needs to use a higher value for k.
3) Plotting a sorted graph of the distance of k-nearest neighbours from each point for given number of minimum points (k), gives a good value of epsilon (radius). This value is around the point at which there is a sudden rise in the value of the distances. This point can be located by looking out for a resemblance of the folded elbow.

References

1) Class Lectures of BT3041: Analysis and Interpretation of Biological Data delivered and curated by Dr. Srinivasa Chakravarthy. V
2) Introduction to Data Mining by Pang-Ning Tan, Michael Steinbach, Vipin Kumar
3) Understanding DBSCAN Algorithm and Implementation from Scratch: Towards Data Science
4) DBSCAN: Wikipedia