Programming Assignment 3

Rachit Garg CS14B050

1 Clustering

Data sets were converted to ARFF format and visualized. Analysis On Basis of Visualization

1.1 Part (a)

:

1.1.1 spiral

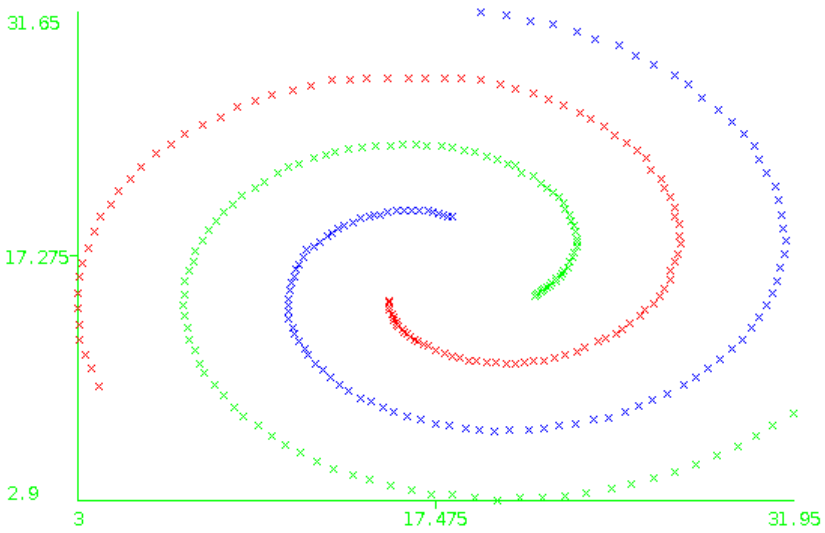

Figure 1: spiral

- K-means
  - K-means would fail as data is not convex in nature.
- DBSCAN:
  - DBSCAN works very well in this scenario. And gives a good spiral cluster as points are uniformly dense.
- Hierarchical clustering:

– **Single link**: It would work well.
– **Complete link**: It would not work well as points in same cluster are very far from each other.

1.1.2 Compound

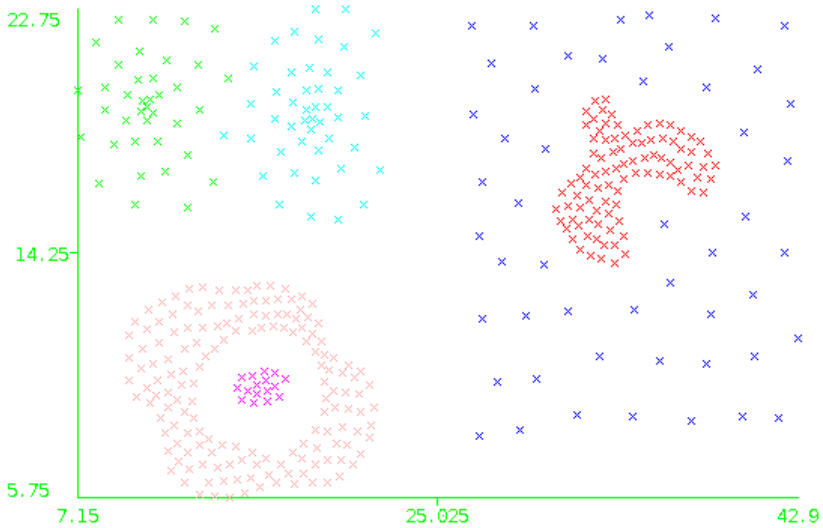

Figure 2: Compound

- K-means
  - As structures are complex, k means doesn't work well.
- DBSCAN:
  - The densities of the classes are not uniform, hence coming up with a suitable radius and minimum number of points parameter is tough.
- Hierarchical clustering:
  - **Single link**: The close enough clusters might not get classified correctly.

1.1.3 Aggregation Data

:

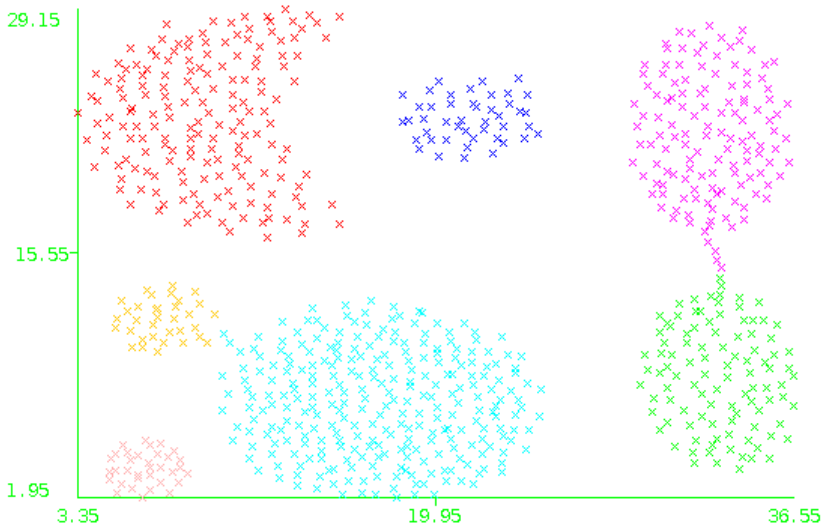

Figure 3: Aggregation

- K-means
  - K-means can run well as clusters are mostly convex in nature. Also they are well separated. A run of k means on number of clusters as 7 gives a good result on weka.
- DBSCAN:
  - With appropriately chosen minPts and ϵ, DBSCAN could correctly cluster all the classes correctly.
  - The minimum number of points was a parameter that worked well in the range 1-5.
  - ϵ was carefully tuned, so that the yellow and blue clusters do not end up in the same cluster.
- Hierarchical clustering:
  - **Single link**: Clusters close together such as pink and green and light blue and yellow reduce the purity of single link clustering.
  - **Complete link**: With complete link clustering the connecting points between the two groups would likely be in a cluster and the rest of the points in the two different classes would be in the same cluster.

1.1.4 D31

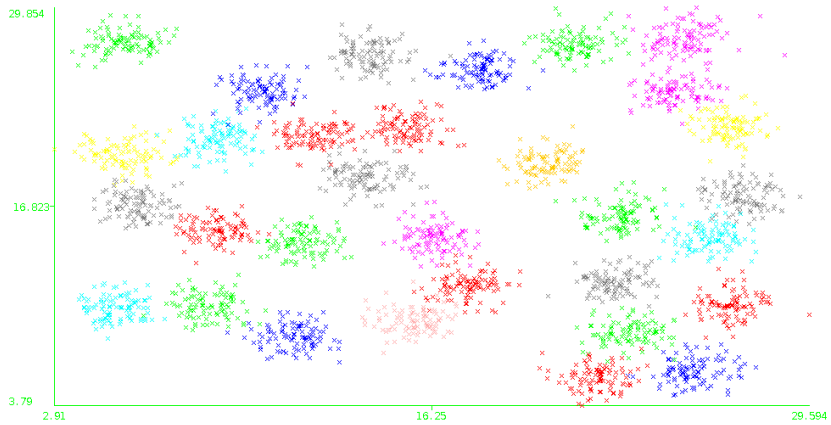

Figure 4: D31

- K-means
  - With k = 31, K-means will be able to cluster the classes properly as all of them approximately come from a spherical distribution
- DBSCAN:
  - DBSCAN is a slow algorithm, hence after a while if we increase complexity in parameters it takes time to cluster.
- Hierarchical clustering:
  - **Single link**: Points are very close to each other and gives problems in clustering.
  - **Complete link**: Connecting points of some clusters get assigned wrong.

1.1.5 flames

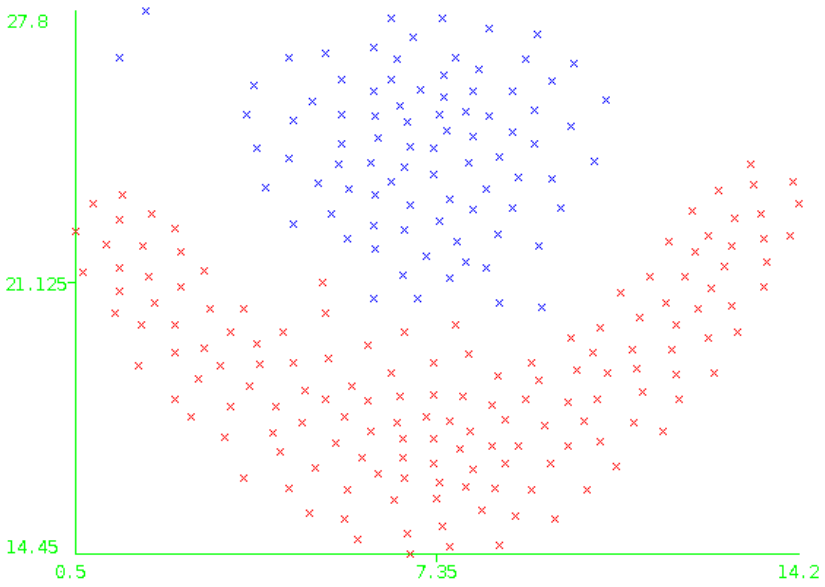

Figure 5: Flames

- K-means
  - Lack of a proper convex structure means k means will struggle.
- DBSCAN:
  - DBSCAN would work well if we carefully select radius and min number of points to separate the clusters.
- Hierarchical clustering:
  - **Single link**: It will perform well on single link.
  - **Complete link**: Some outliers in the red region cause a decrease in purity.

1.1.6 jain

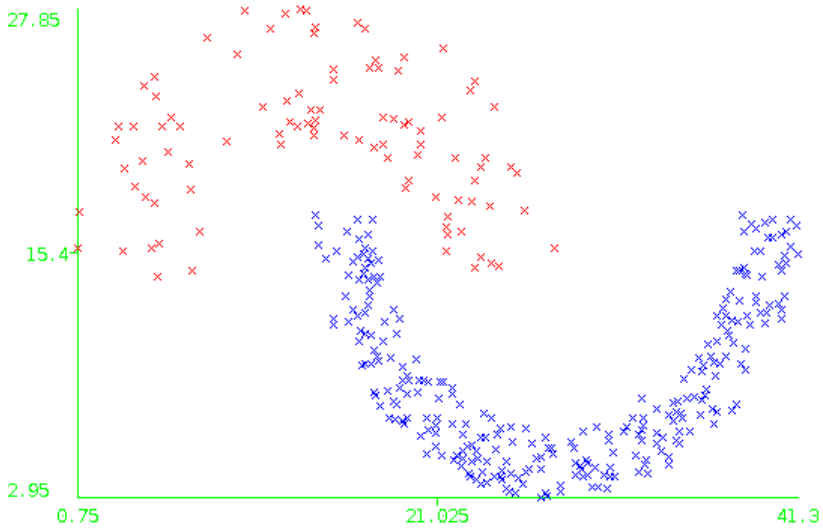

Figure 6: Jain

- K-means
  - Lack of convex structures means k means will struggle.
- DBSCAN:
  - DBSCAN would work well if we carefully select radius and min number of points to separate the clusters.
- Hierarchical clustering:
  - **Single link**: It will report one single cluster on the whole due to close link between classes.
  - **Complete link**: The clusters have points that are far away from each other, hence they will not get clustered correctly.

1.1.7 pathbased

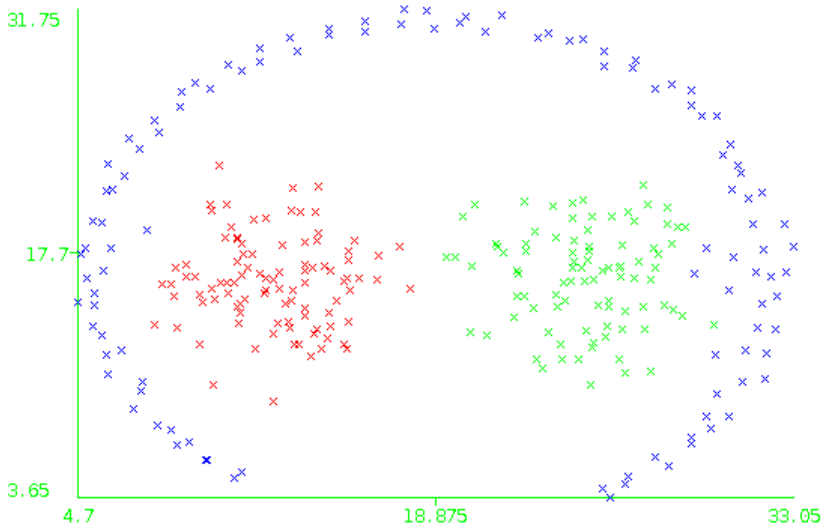

Figure 7: Path Based

1. K-means
   - The outside paths would never be ge classified in one cluster via k-means.
2. DBSCAN:
   - DB Scan doesn't work well in this scenario as if we increase epsilon above 0.05 the two major clusters in between are classified together and if we keep epsilon at 0.05 the outside path isn't getting clustered correctly. It is being incorrectly assigned to no class as the path is not continuous, there are breaks in the outside path.
3. Hierarchical clustering:
   - **Single link**: It should work well.
   - **Complete link**: The whole outside path has points with a huge distance between them, hence they won't be clustered in the same cluster.

1.1.8 R15

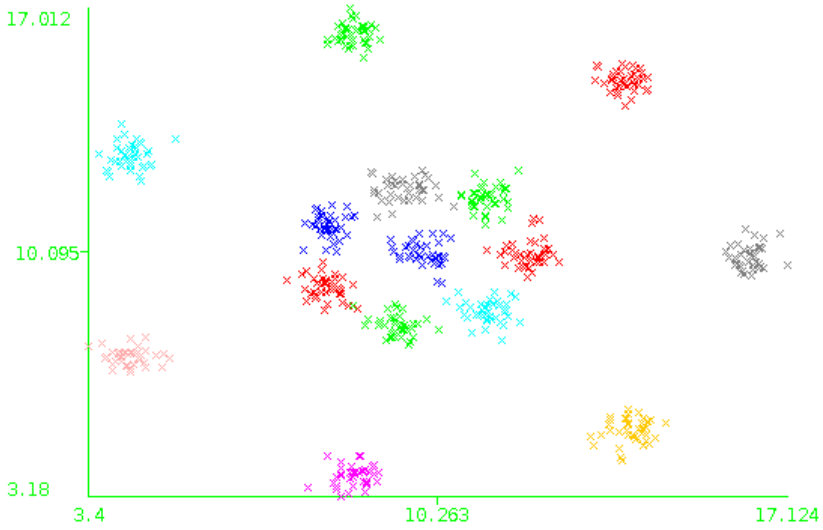

Figure 8: R15

- K-means
  - k-means should work very well on this data set.
- DBSCAN:
  - DBSCAN should work well as all clusters are fairly dense and well separated.
- Hierarchical clustering:
  - **Single link**: It might cluster some with connecting points together in one cluster.
  - **Complete link**: It should work well.

1.2 k vs cluster purity

The cluster purity were calculated by changing the number of clusters in R15 data set. The following observations and plots are as follows:-

K	Purity
1	6.67
2	13.33
3	20
4	26.67
5	33.34
6	40
7	46.67
8	53.33
9	60
10	66.67
11	73.34
12	80
13	86.17
14	84
15	89.67
16	87.834
17	95.67
18	91.5
19	88.17
20	85.17

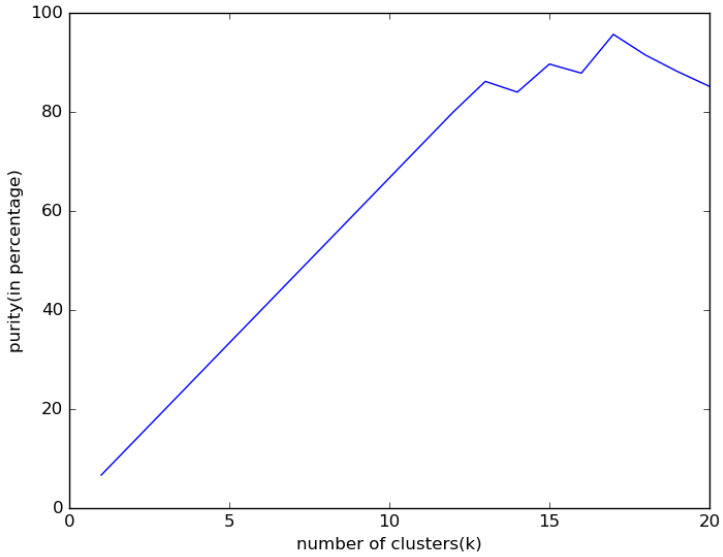

Figure 9: Number of clusters(k) vs Purity(in percentage)

1.3 DBScan on Jain dataset

- We start at epsilon = 0.9 and minpts = 6. Cluster purity that we obtain is 74%.
- If we increased epsilon and decreased or increased min points we observed that everything was getting classified into one cluster which is similar to our starting case.
- If we decrease epsilon and increase min points, we observe that there is an increase in number of unclustered instances, though purity increases and everything was getting classified into one cluster.
- Hence we decreased epsilon and decreased min points. As in data we could see two different clusters, the best result we got was when epsilon = 0.08 and min points = 4. We got two unclustered instances and a 98.2 percent purity. We observed that the two points which were causing the clusters to link are approximately at a distance 0f 0.083. The cluster assignments visualized is also depicted.

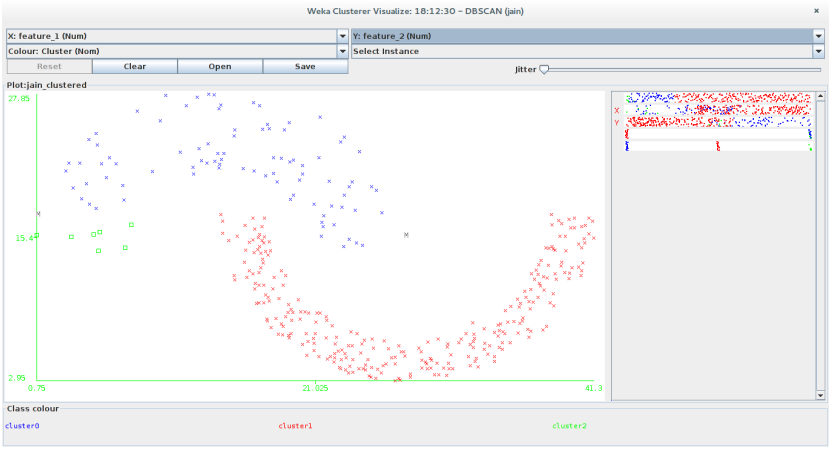

Figure 10: Feature 1 vs Feature2 (diff classes are colored)

1.4 Part(e)

1.4.1 path-based

Link Type	Purity
single link	37
complete link	71
average	73
mean	70
centroid	74
ward	76
adjcomplete	63
neighbour_adjoin	37

- The data is such that ward works best in this scenario. Single link performs bad as it leads to linking of the clusters with the outside path.
- DB Scan doesn't work well in this scenario as if we increase epsilon above 0.05 the two major clusters in between are classified together and if we keep epsilon at 0.05 the outside path isn't getting clustered correctly. It is being incorrectly assigned to no class as the path is not continuous, there are breaks in the outside path.
- Best classification was seen when epsilon was 0.05 and minpts 1 and gave a purity of 67% with no unclustered point. If we increased minpts from this value the purity would increase but the number of unclustered instances also drastically increase.

1.4.2 spiral

- In spiral dataset, in hierarchical clustering we observed that single link gives best 100% purity and classifies everything correctly rest give close to 40 percent purity.
- DB Scan is best suited for this type of clustering as we need a random cluster shape. If we put our epsilon as 0.05 and minpts as 2. We are able to get three perfectly classified clusters.

1.4.3 flames

Link Type	Purity
single link	65
complete link	52
average	64
mean	93
centroid	65
ward	100
adjcomplete	65
neighbour_adjoin	64

- Here we observe that ward and mean type links perform better. Single link tends to classify everything into one cluster and complete link gives arbitrary clusters.
- DB Scan works well after a lot of parameter tuning. For epsilon = 0.06 and min points = 2, we get a good purity of 96.7% with only 10 unclustered points which are outliers in the data.

1.5 DB31

- DB Scan might work well after a lot of parameter tuning, since if we increase min points the algorithm is taking a lot of time to run, results could only be partially recorded. For epsilon = 0.04 and min points = 33, we get a purity of 65% with 60 unclustered points.
- Since number of clusters was large, weka didn't run for k means wih 32 clusters and heirarchical clustering with ward linkage as well.