Q VI

VI) Clustering Algorithms (Euclidean distance may be used)

1. Understand the working of k-means clustering algorithm. Give a pseudo code for the same and trace it for a sample dataset of your choice, clearly showing the centroid updates.

Pseudo Code

1. Choose random K Samples (initial centroid)
2. For specified no of iterations,
   1. Form k clusters by assigning items to their closest mean/centroid
   2. Update the mean points by taking mean of each cluster
   3. Repeat

Trace

Dataset

|  |  |  |
| --- | --- | --- |
|  | X | Y |
| 0 | 3 | 4 |
| 1 | 7 | 5 |
| 2 | 2 | 6 |
| 3 | 3 | 1 |
| 4 | 8 | 2 |
| 5 | 7 | 3 |
| 6 | 4 | 4 |
| 7 | 6 | 6 |
| 8 | 7 | 4 |
| 9 | 6 | 7 |

Choose random K = 2 means,

Let they be C1 – (1, 1) and C2 – (7, 7)

Each item is assigned to whichever cluster has least distance

So, C1 – 0, 2, 3, 6, and C2 – 1, 4, 5, 7, 8, 9

Now find new mean of the clusters

C1(new) = ((3+2+3+4) / 4 , (4+6+1+4) / 4) = (3, 3.75)

C2(new) = ((7+8+7+6+7+6) / 6 , (5+2+3+6+4+7) / 6) = (6.83, 4.5)

Thus, means has been updated

Now, Repeat same process

1. Understand the working of k-medoids clustering algorithm. Give a pseudo code for the same and trace it for the sample dataset used for VI-(a), clearly showing the centroid updates.

Pseudo Code

1. Choose random K items from the given dataset (initial medoids)
2. For specified no of iterations,
   1. Form k clusters by assigning items to their closest medoid
   2. Calculate the total dissimilarity
   3. Now, select a random non-medoid item as medoid and repeat (a) and (b)
   4. If cost/total dissimilarity is more for previous set of medoids than for the new medoid, then replace the old set with new set of medoid and Repeat from (a)
   5. If cost/total dissimilarity is more for new set of medoids than for the previous medoid, then rollback and keep the old set of medoid and Repeat (c)

Trace

Dataset

|  |  |  |
| --- | --- | --- |
|  | X | Y |
| 0 | 3 | 4 |
| 1 | 7 | 5 |
| 2 | 2 | 6 |
| 3 | 3 | 1 |
| 4 | 8 | 2 |
| 5 | 7 | 3 |
| 6 | 4 | 4 |
| 7 | 6 | 6 |
| 8 | 7 | 4 |
| 9 | 6 | 7 |

Choose random K = 2 medoids,

Let they be C1 – (3, 1) (I3) and C2 – (6, 6) (I7)

Each item is assigned to whichever cluster has least dissimilarity

So, C1 – 0, 4, 6 and C2 – 1, 2, 5, 8, 9

Total Diss = (3 + 6 + 4) + (2 + 4 + 4 + 3 + 1) = 13 + 14 = 27

Now, we choose randomly, (I8) = (7, 4) as medoid

C1 – (3, 1) and C2 – (7, 4)

Now, assignment is, C1 – 0, 2 and C2 – 1, 4, 5, 6, 7, 9

Total Diss = (3 + 6) + (1 + 3 + 1 + 3 + 3 + 4) = 9 + 15 = 24

As 24 < 27, we change medoids to new set – C1 – (3, 1) and C2 – (7, 4)

Repeat above process

1. Understand the working of hierarchical clustering algorithm- Agglomerative, Divisive and trace them for the dataset used in VI-(a). You may trace the algorithm for both the approaches and use the dendrogram to represent the clustering process pictorially as well.

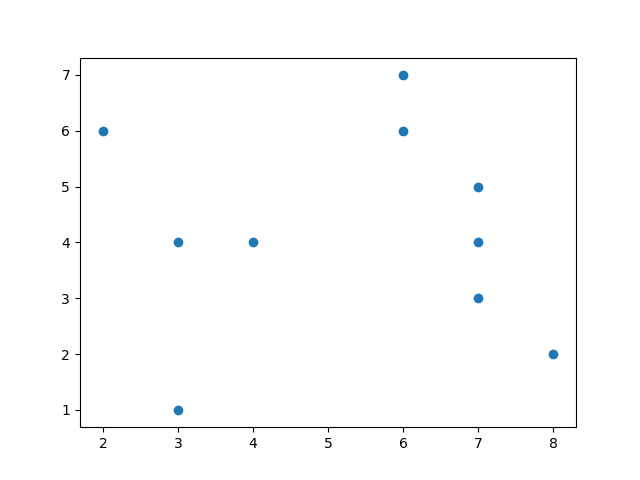
Trace

Agglomerative Clustering

Dataset

|  |  |  |
| --- | --- | --- |
|  | X | Y |
| 0 | 3 | 4 |
| 1 | 7 | 5 |
| 2 | 2 | 6 |
| 3 | 3 | 1 |
| 4 | 8 | 2 |
| 5 | 7 | 3 |
| 6 | 4 | 4 |
| 7 | 6 | 6 |
| 8 | 7 | 4 |
| 9 | 6 | 7 |

Initially, every item is a cluster – C0, C1, … C9



Closest 2 items are (3, 4) and (4, 4) (C0 and C6)

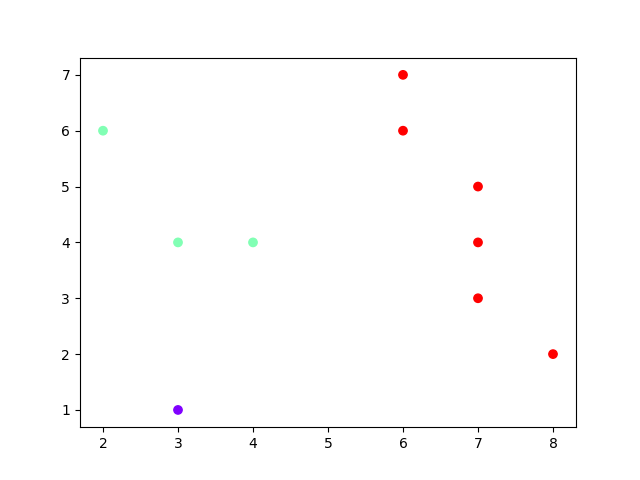
So, Merge them into 1 cluster C06 – (3.5, 4.0)

Next Closest 2 items are (7, 5) and (7, 4) (C1 and C8)

So, Merge them into 1 cluster C18 – (7.0, 4.5)

Simlarly merging,

Finally 3 Clusters

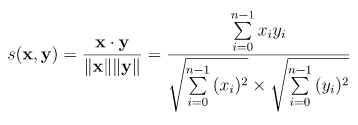


Q VII

VII) Survey the various distance measures used by clustering algorithms eg: Cosine, Jaccard similarity measures etc.

Explore for a minimum of 5 measures (non-Euclidean distance measures) and trace them to measure distance 2 data points.

Cosine Distance

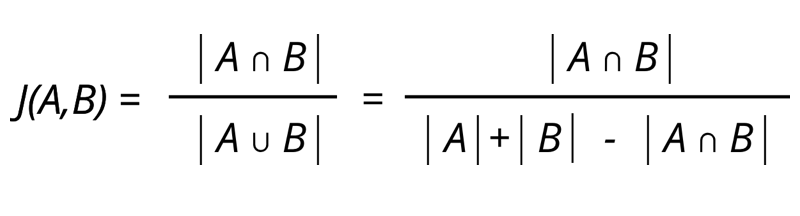


X = (2, 3)

Y = (4, 1)

Distance = = = = 0.739

Jaccard Distance



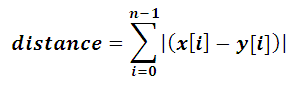
A = {1, 2, 3, 4}

B = {1, 2}

J(A, B) = 2 / 4 = 0.5

Jaccard Distance = 1 – Jaccard Similarity = 1 – 0.5 = 0.5

Manhattan Distance

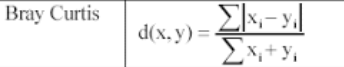


X = (4, 3)

Y = (8, 2)

Manhattan Distance = |(4 - 8)| + |(3 - 2)| = 4 + 1 = 5

Bray-Curtis Distance

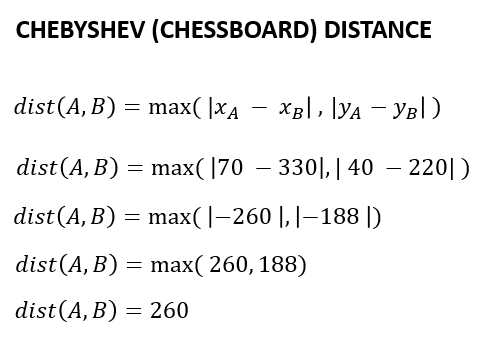


X = (3, 1)

Y = (7, 3)

Bray-Curtis Distance = = = 0.429

Chebychev Distance



A = (3, 6)

B = (1, 7)

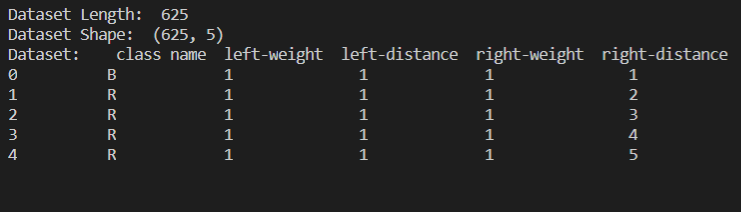
Chebychev Distance = max (|3-1|, |6-7|) = max (2, 1) = 2

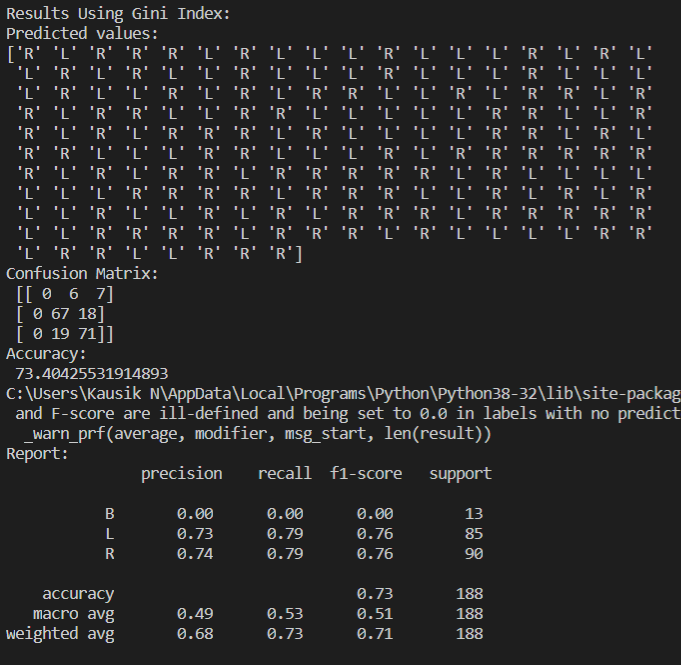
Q VII

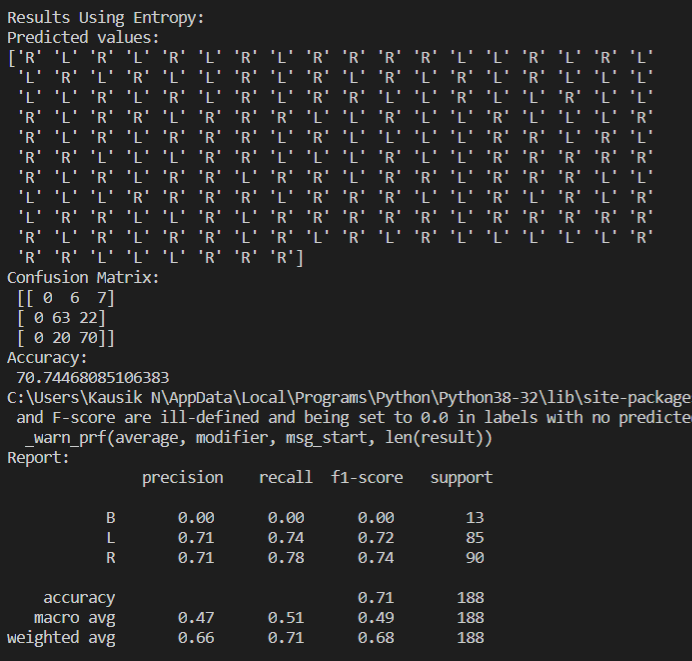
VIII)

a) Test drive/Implement Decision tree, Naive Bayes, BPN, k-means, hierarchical clustering in a platform of your choice.

Decision Tree

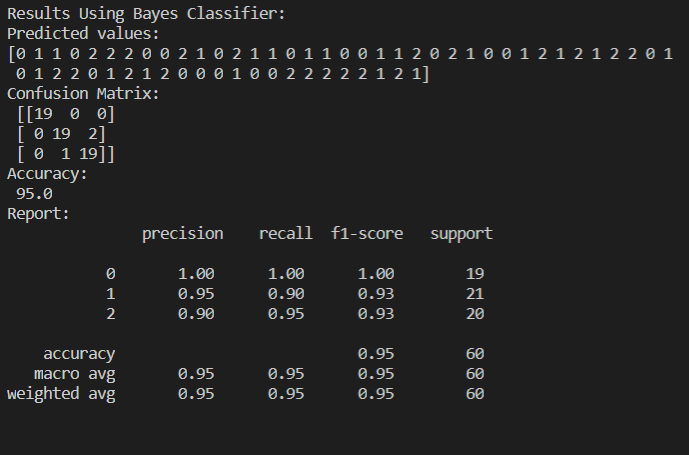






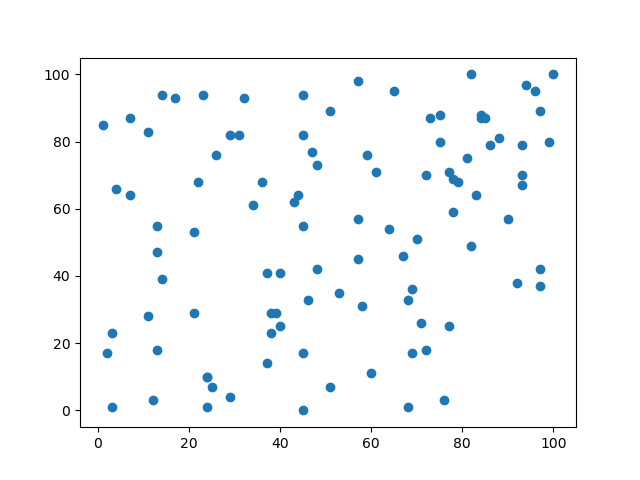
Naive Bayes Classifier

Using Iris Dataset

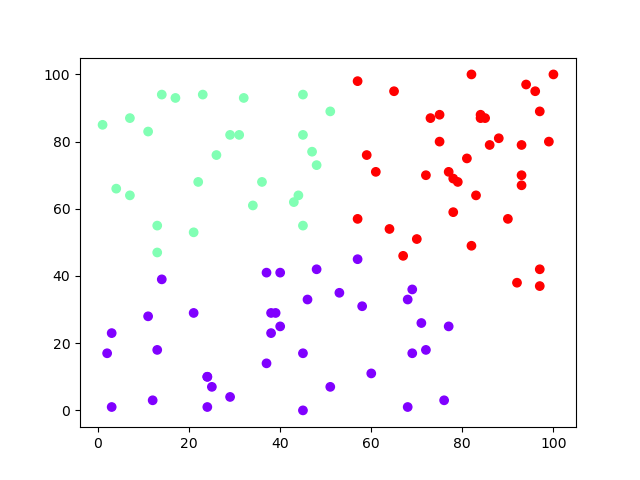


K-Means Clustering

100 Random Points between 0 to 100 values

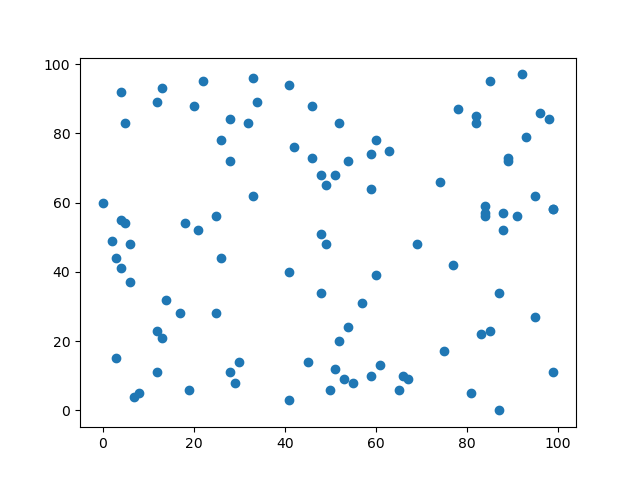


3 Clusters

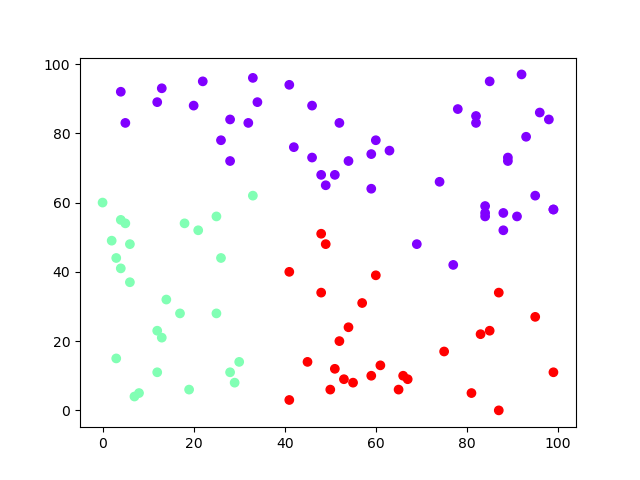


Agglomerative Clustering

100 Random Points between 0 to 100



3 Clusters



b) Test drive/Implement optimization using GA operator in a platform of your choice (Not mandatory)

Inputs

equation\_inputs = [4, -2, 3.5, 5, -11, -4.7, 2.5, 0.1]

num\_weights = len(equation\_inputs) # Number of the weights we are looking to optimize.

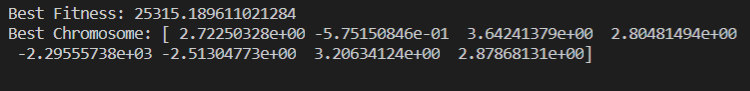
sol\_per\_pop = 200 # Defining the population size.

pop\_size = (sol\_per\_pop, num\_weights) # The population will have sol\_per\_pop chromosomes where each chromosome has num\_weights genes.

num\_generations = 5000

num\_parents\_mating = 100

After Optimising,



c) Test drive/Implement the bucket brigade classifier (Not mandatory)