IC 272: DATA SCIENCE - III
LAB ASSIGNMENT – VII
Clustering

Student's Name: Sumit Kumar Yadav

Roll Number: B19119

Mobile No: 9474029241

Branch:CSE

IC 272: DATA SCIENCE - III
LAB ASSIGNMENT – VII
Clustering

1 a.

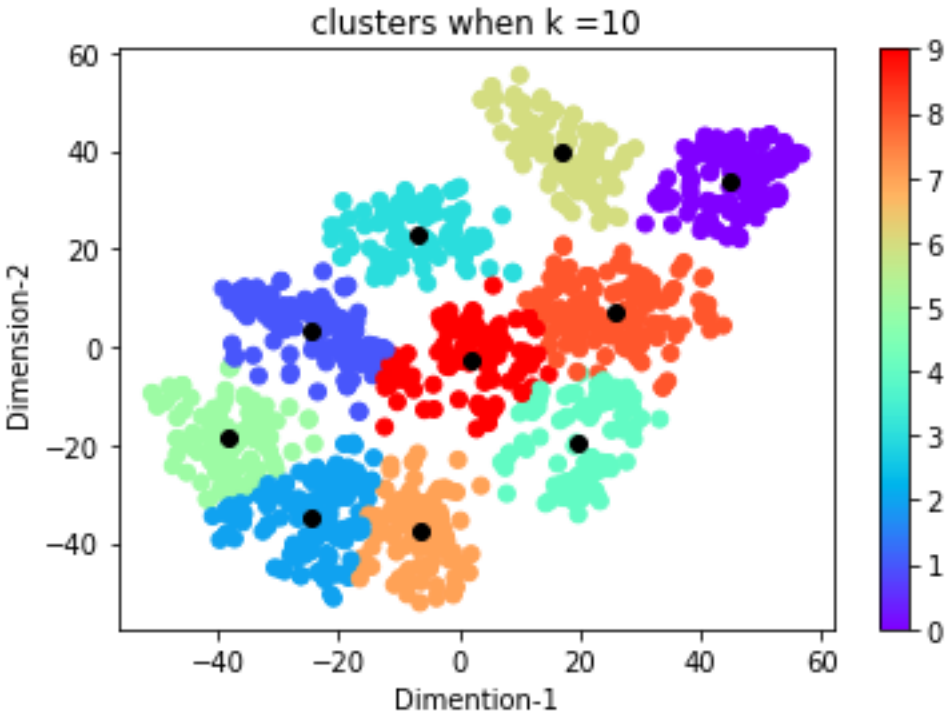


Figure 1 K-means (K=10) clustering on the mnist tsne training data

Inferences:

1. K-Means clustering divides the data into predefined distinct non overlapping clusters based upon the distances, those having minimum squared Euclidean distance with cluster centroid are assigned to that cluster until the convergence criteria is satisfied.
2. K-means algorithm assumes cluster boundaries to be circular in 2D. From the output, we can say that some of the clusters are having circular boundary while some are linear in shape. Possible reasons for this behavior are , the classes don't have the same variance and there may not be enough data points so as to form circular boundary

b.

The purity score after training examples are assigned to the clusters is 0.691

c.

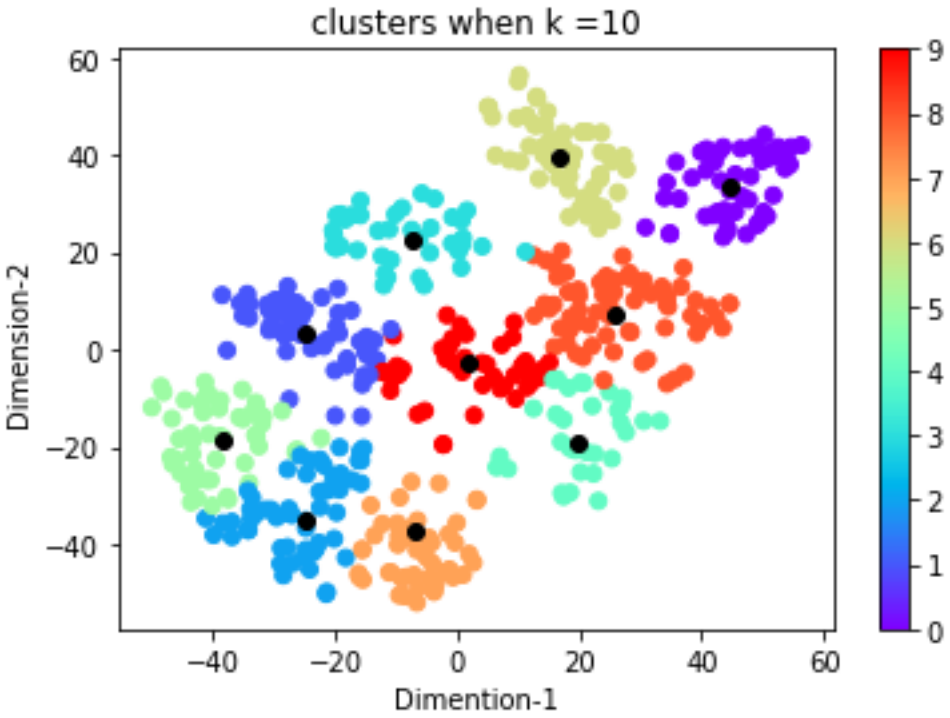


Figure 2 K-means (K=10) clustering on the mnist tsne test data

Inferences:

1. There are no observable differences between the plots of train and test data this is because the test plot is based on centroid of training data.

d.

The purity score after test examples are assigned to the clusters is 0.678

Inferences:

1. Purity score is higher for train data as compared to test data this is because the error reduced.
2. The K-Means clustering method do not work for numerical data. Boundary of clusters are assumed to be circular that can fail in many cases where boundary is of any arbitrary shape. The method is also very sensitive towards the outliers.

2 a.

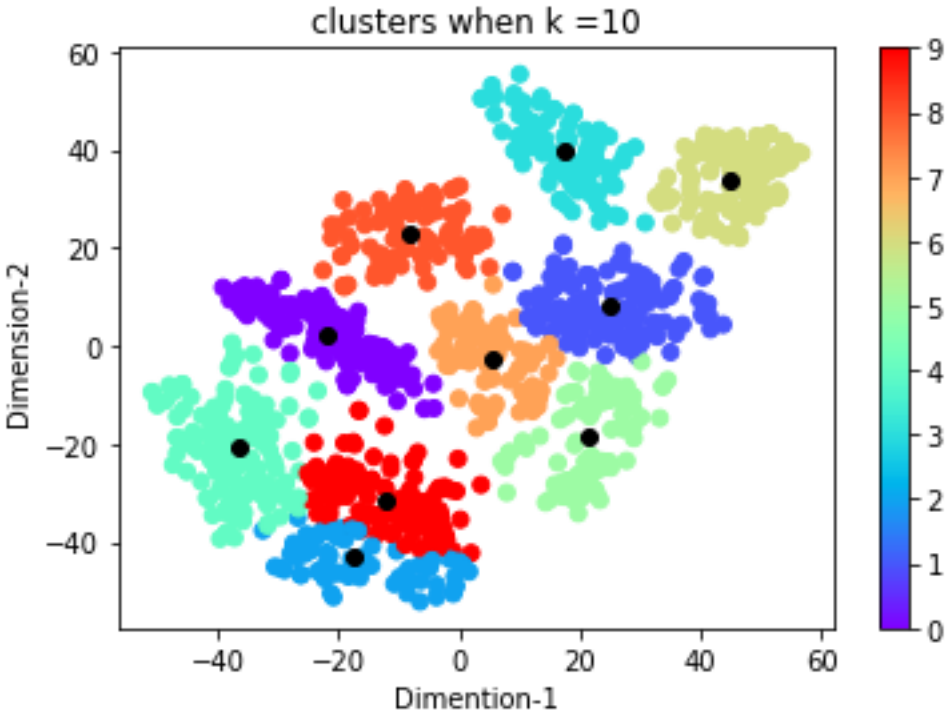


Figure 3 GMM clustering on the mnist tsne training data

Inferences:

1. Inferring from the clusters formed in the above plot, comment on the clustering prowess of the algorithm.
2. GMM algorithm constraints cluster boundaries to be elliptical in 2D. From the output, we can say that the boundaries are elliptical.
3. The clusters that are formed in the plot of (2a)(GMM) are the best way to predict test data than the clusters of K-Means (1a), but looking at plots there isn't much difference in the clusters and cluster centers formed in both clustering. In GMM, we used the means as the cluster representatives and as the cluster centers.

b.

The purity score after training examples are assigned to the clusters is 0.712

c.

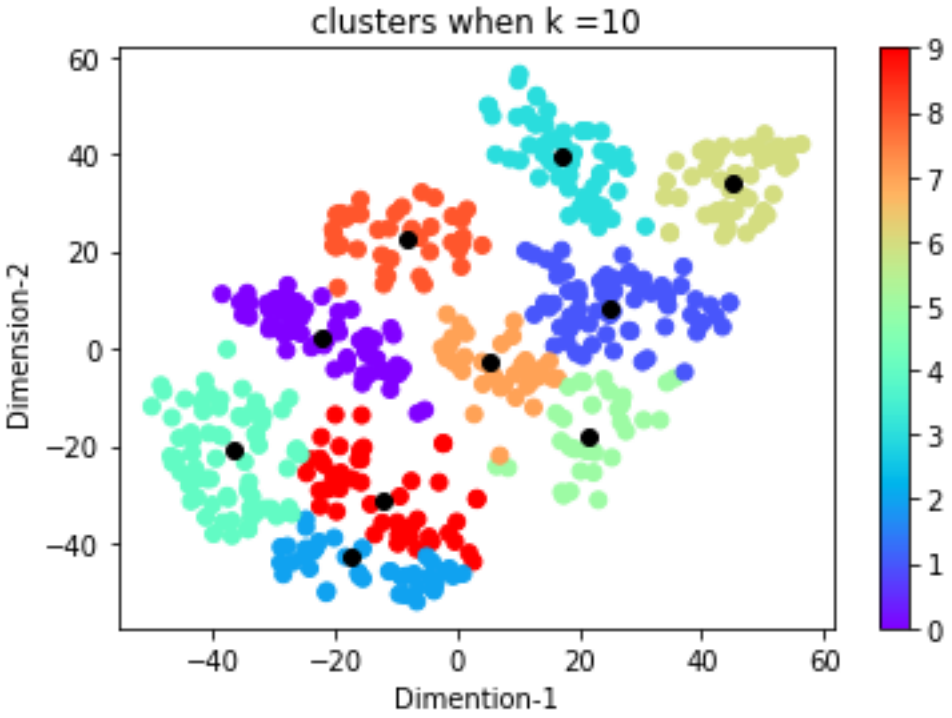


Figure 4 GMM clustering on the mnist tsne test data

Inferences:

1. There are no observable differences between the plots of train and test data this is because both of the plots are formed on the same parameters(mean and covariance).

d.

The purity score after test examples are assigned to the clusters is 0.688

Inferences:

1. Purity score is higher for the train data as compared to the test data. This might be because the train data is predicted well compared to test data.
2. Some of the limitation are
   (i) It cannot form clusters with data of higher dimension
   (ii) It is computationally expensive
   (iii) It doesn't work for clusters with arbitrary shape.

3 a.

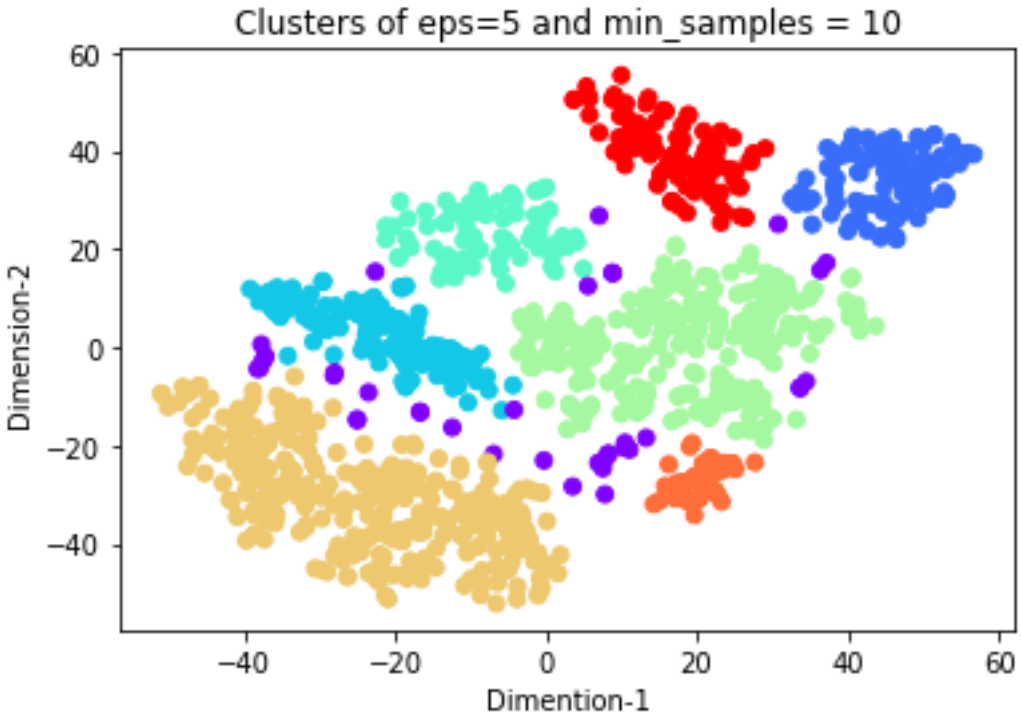


Figure 5 DBSCAN clustering on the mnist tsne training data

Inferences:

1. The number of cluster need not be specified. The algorithm divides data points on notion of density. The algorithm also takes care about the outliers in the data. The clustering process is not so good because some left out points are there where the density is less.
2. In both GMM and KMeans there were more clusters than this case. In this case some left out points are there too. GMM is accurate and more effective than DBSCAN and KMeans.

b.

The purity score after training examples are assigned to the clusters is 0.585

c.

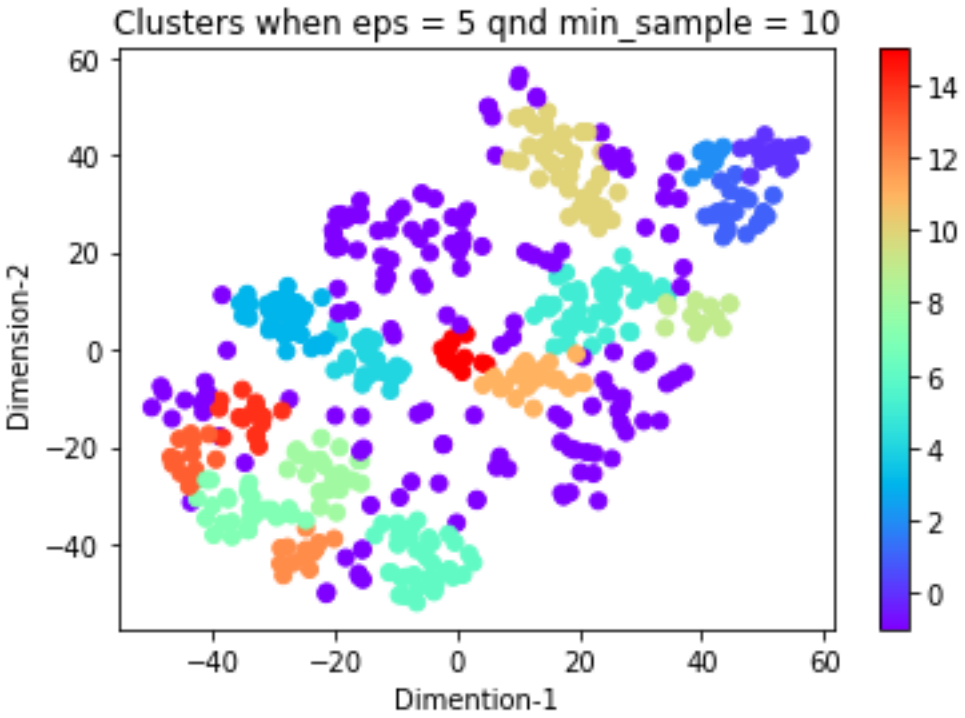


Figure 6 DBSCAN clustering on the mnist tsne test data

Inferences:

1. The clusters are less dense in train data.

d.

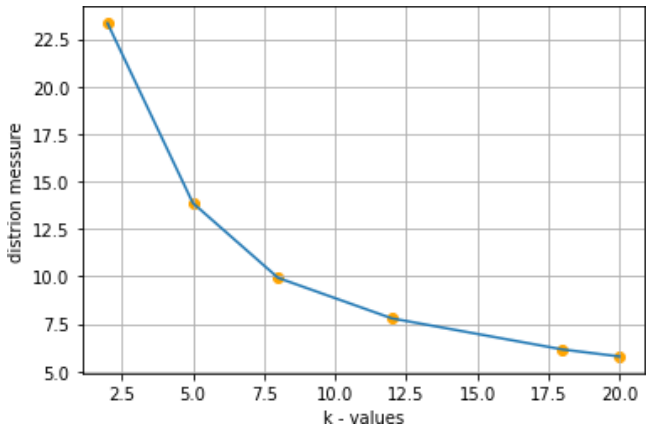
The purity score after test examples are assigned to the clusters is 0.484

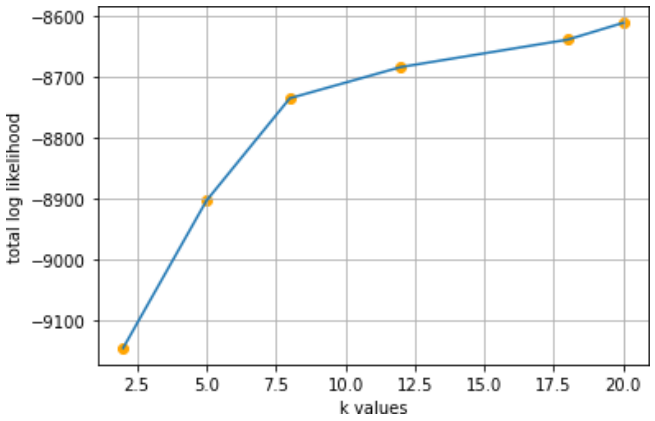
Inferences:

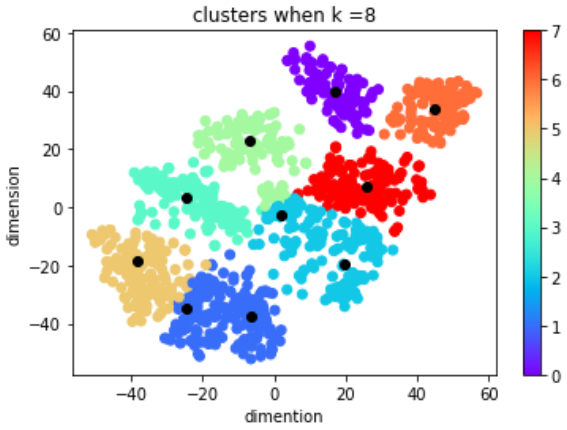
1. The purity score is higher for the train data. In this model the model decides for the number of cluster. It is possible that the distance between two boundaries of two class is very small such that they cannot be distinguished by the given values of 'epsilon' and 'min_samples'.
2. The model fails if there isn't significant difference of density between classes.

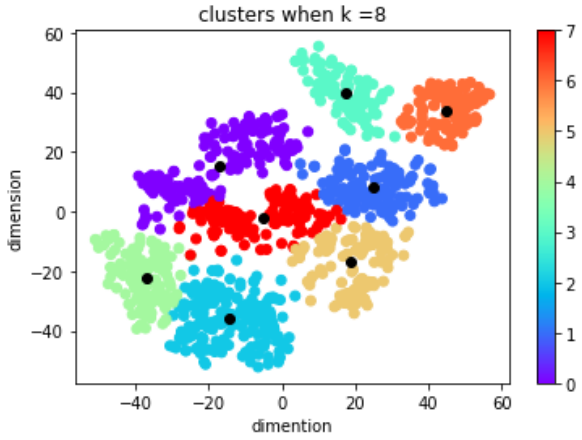
Bonus Questions:

IC 272: DATA SCIENCE - III
LAB ASSIGNMENT – VII
Clustering









PART-1

K Values	Kmean purity score	GMM purity score
2	**0.2**	**0.2**
5	**0.392**	**0.471**
8	**0.63**	**0.588**
12	**0.61**	**0.63**
18	**0.478**	**0.472**
20	**0.453**	**0.424**

The above two plots are the plots of different values k vs the distortion (k-means) and k vs the log likelihood.

We can see from 1st plot that the at the point of elbow is the optimized k value. From this point onwards the distortion becomes almost linear i.e., same. So, the optimized k-value there is 8. Similarly, for 2nd plot at the elbow we chose the value k, where it is 8 again. From this point onwards the value of log likelihood becomes almost linear.

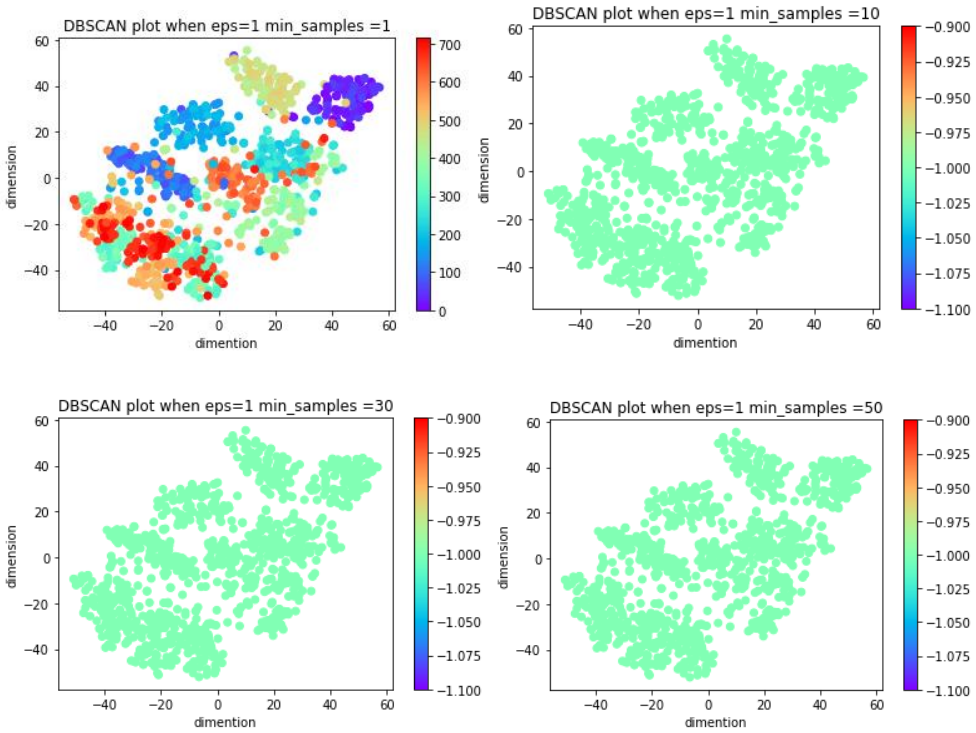
8

Above are the scatterplot for optimum number for k value which is 8.

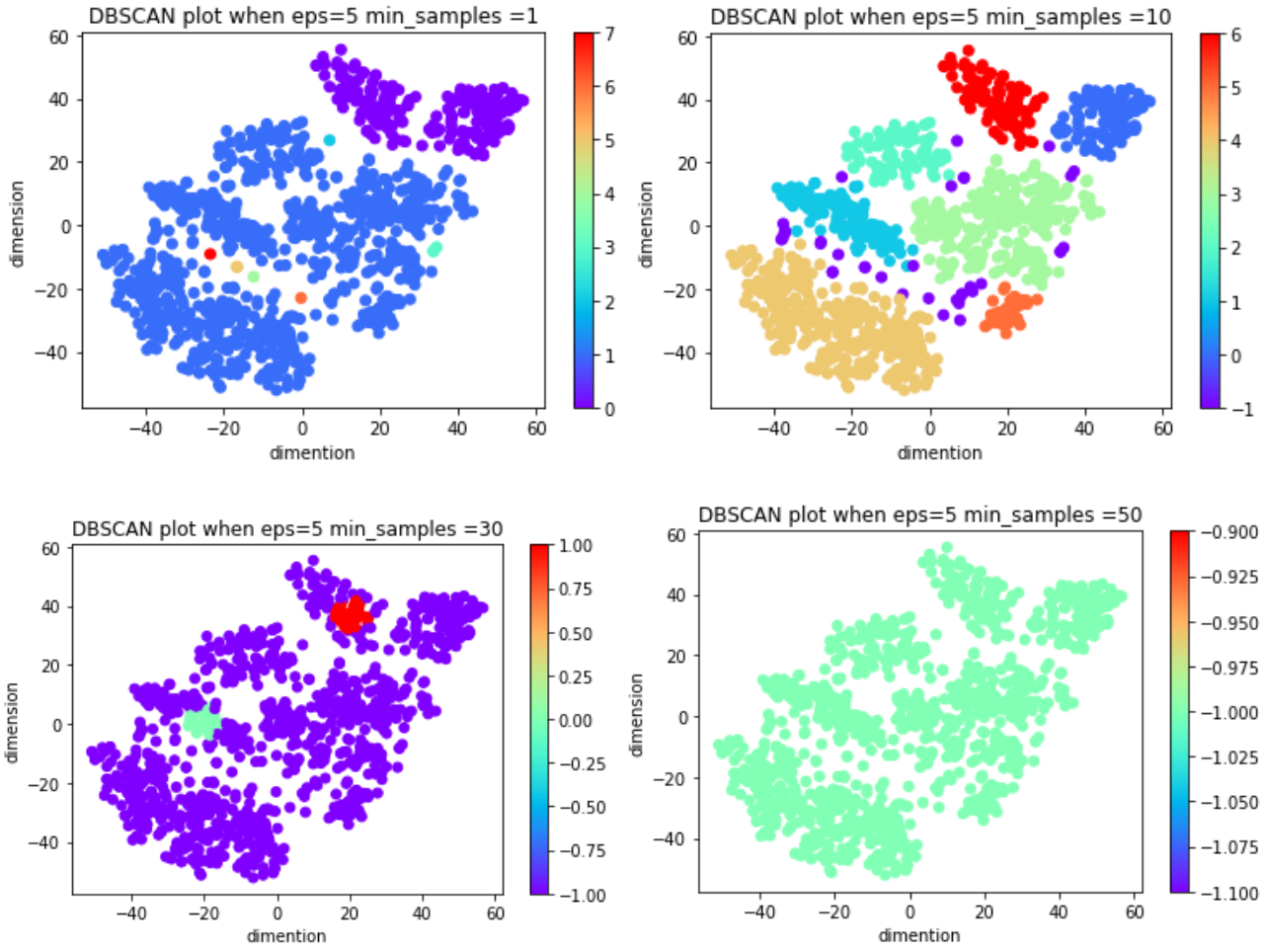
3.1 PART-2

Purity score	Eps=1	Eps=5	Eps=10
Min samples=1	**0.036**	**0.208**	**0.1**
Min samples=10	**0.1**	**0.585**	**0.1**
Min samples=30	**0.1**	**0.158**	**0.1**
Min samples=50	**0.1**	**0.1**	**0.503**

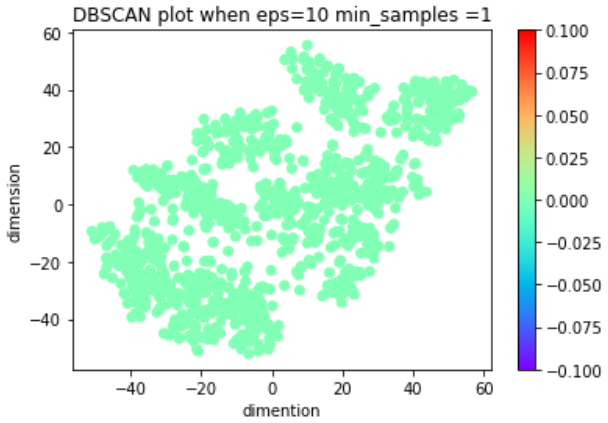
Plot for train data eps: 1 and minimum samples: 1,10,30,50

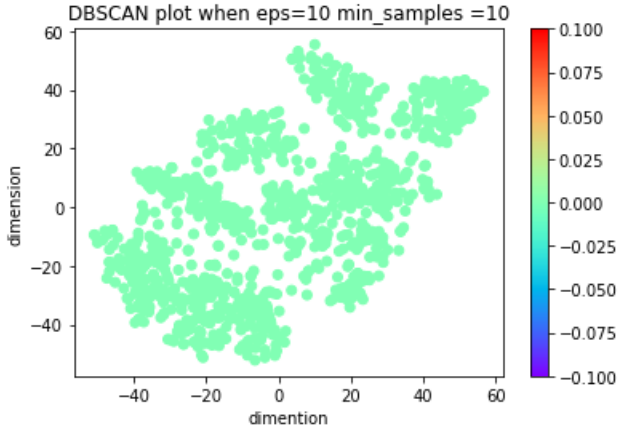


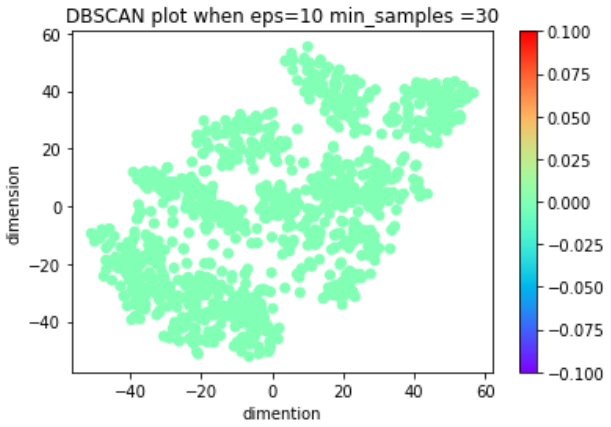
Here in this plot, we can observe that there are no proper clusters formed and there are very large in number, which is not suitable for clustering, the chosen epsilon and minimum samples are not suitable for DBSCAN clustering. This have the least purity score is 0.036, hence the clusters are clumsy. Where for eps=1 and min samples =1 than there many clusters are formed (700) . as keeping eps 1 and increasing the min samples the data is combining into a single cluster

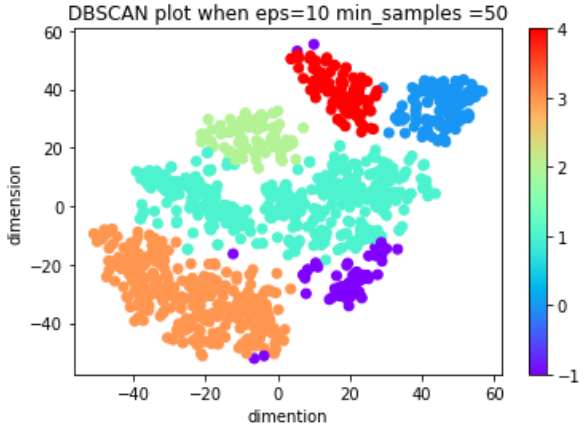


This plot seems to be the best plot compared to above because we got few numbers of clusters (8) though there are few left out noise points where the density is less and those doesn't have a core point in the boundary and don't even have 10 samples in the neighborhood. This has scatterplot of 0.585 which is highest of all. After that changing the value of min samples are mixing the clusters as single cluster this because the makes many as boundary points, so the cluster are not sepratable by low dense by region


DBSCAN plot when eps=10 min_samples =1
dimension
dimention


DBSCAN plot when eps=10 min_samples =10
dimension
dimention


DBSCAN plot when eps=10 min_samples =30
dimension
dimention


DBSCAN plot when eps=10 min_samples =50
dimension
dimention

The first three clusters are not good as remaining but the last cluster also seems to be good enough for predicting because there are at least few numbers of clusters in the plot. So, from the overall plots, the best suitable plot is for epsilon=5 and minimum samples=10; and epsilon=10 and minimum samples=50. Out of these, the best suitable values for epsilon and minimum samples would be 5 and 10 because that gave us maximum number of clusters (8) and best plot for predicting compared to all other values of epsilon and minimum samples. This last plot has the purity score of 0.503. Only two plots of all have optimum purity scores on test data.