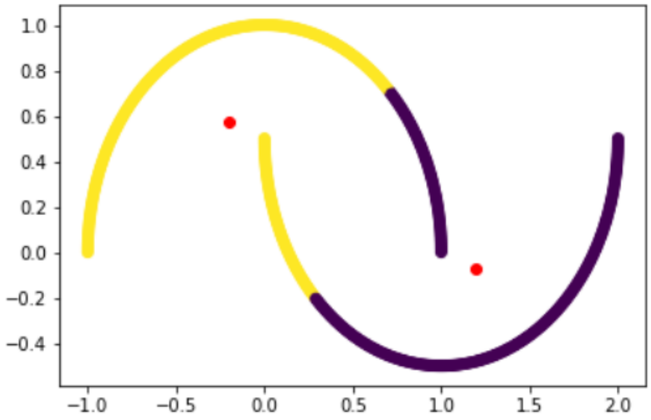
Report

1a.

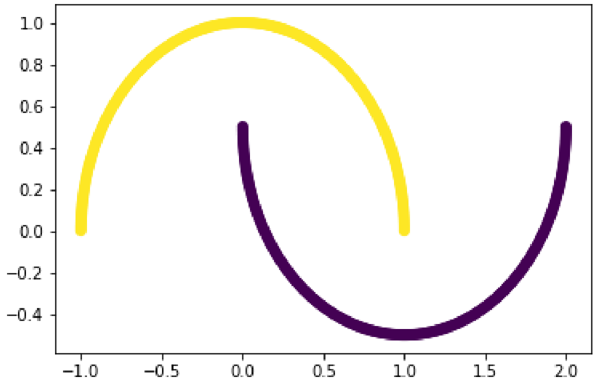


-- In the above plot, we have plotted the points and colored the points according to the labels predicted by the K-Means Algorithm. We have marked the cluster centers with red color.

-- The clusters are formed around the centers. And this is expected since K-Means clustering works by classifying the points using a distance measure.

-- Since the ground truth is not available, we cannot comment on the nature of the classification. But by plotting the points in 2D, we see the points are forming two semi-circles, and could be the case that the two semi-circles are two different clusters. So, with this assumption, we may conclude that the actual nature of the points is not reflected by the PCA algorithm.

1b.



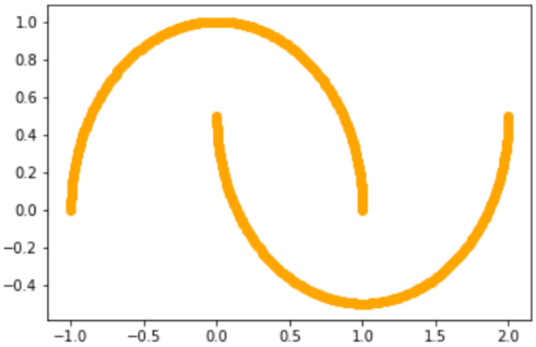
-- In the above plot, we have plotted the points and colored the points according to the labels predicted by the DBSCAN Algorithm.

-- Since clustering is done by considering the distance between two points and using the fact that 'n' number of minimum samples is required to make it a core point. We expect the points situated closer to the same cluster.

-- From graph, we see that the points closer are grouped into the same cluster. And we do not observe any noise points from the graph

-- Since the ground truth is not available, we cannot comment on the nature of the classification. But by plotting the points in 2D we see the points form two semi-circles and could be the case that the the two semi-circles are two different clusters. So, with this assumption, we may conclude that the actual nature of the points is captured by DBSCAN algorithm.

1c.



By plotting the points on 2D we see the points are forming two semi-circles. Since ground truth is not available, we cannot decide on the best classification algorithm for the given dataset.

1. K_Means: In the plot the clustered around the centers of the two cluster. This is expected since the K-means works by clustering the set of points in R radius as a cluster.

2. DBSCAN: In this plot the clusters can be seen as two groups each forming a cluster.

This happens because the DBSCAN algorithm model uses the relative distance between two points and assigns a set of ts (Core point) as a representative of the cluster and does not consider a single global representative of a cluster unlike the K-Means. The set of representative points for a cluster in DBSCAN is set by considering the neighboring points in a fixed radius from the point. And the core points are actual points from the dataset. Whereas in DBSCAN, the cluster representative point is outside the dataset.

1d.

DBSCAN

Pros:

This returns the clusters formed and shows the noise data.

Cons:

Need to optimize the parameters to get results. While elbow method is available for K-Means to find the K- cluster value.

K Means

Pros:

The number of clusters existing in the dataset can be found by identifying the elbow from the graph using ELBOW method.

Cons:

It does not recognize the noise. It also classifies the noise as a cluster

2a.

- When it comes to the number of iterations needed for tSNE to converge, the simplest recommendation can be the more iterations the better. However, this is not feasible for big data sets as one might have to wait for days to reach e.g., 10000 iterations. In contrast, if you use too few iterations the clusters might not be visible, and you typically discover a huge clump of data points in the center of your tSNE plot just like the plot when number of iterations = 250.
- Looking carefully at the tSNE plots, we notice that the largest distance between data points is about ~100. This simple rule of thumb indicates that the algorithm reached convergence and further increasing the number of iterations will only marginally change the plot. Hence, experiments with no. of iterations = 1000 and 2000 produce the same results.

2b.

- t-SNE algorithm starts by calculating the probability of similarity of points in high-dimensional space and calculating the probability of similarity of points in the corresponding low-dimensional space. The similarity of points is calculated as the conditional probability that a point A would choose point B as its neighbor if neighbors were picked in proportion to their probability density under a Gaussian (normal distribution) centered at A. It then tries to minimize the difference between these conditional probabilities (or similarities) in higher-dimensional and lower-dimensional space for a perfect representation of data points in lower-dimensional space. To measure the minimization of the sum of difference of conditional probability t-SNE minimizes the sum of Kullback-Leibler divergence of overall data points using a gradient descent method.
- t-SNE is a stochastic model. By allowing for random variation in inputs, stochastic models are used to estimate the probability of various outcomes. Creating stochastic models involves a set of equations with input that represents uncertainties over time. Moreover. t-SNE is a cost function that is not convex (with different initializations-different results). Therefore, stochastic models will produce different results when running at various times.