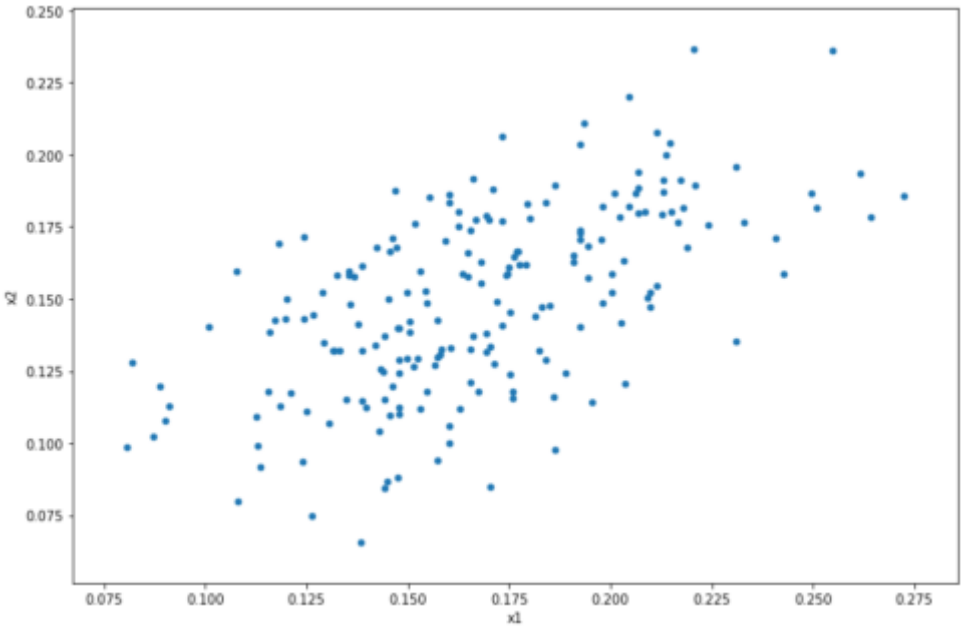
7. Testing algorithms on data

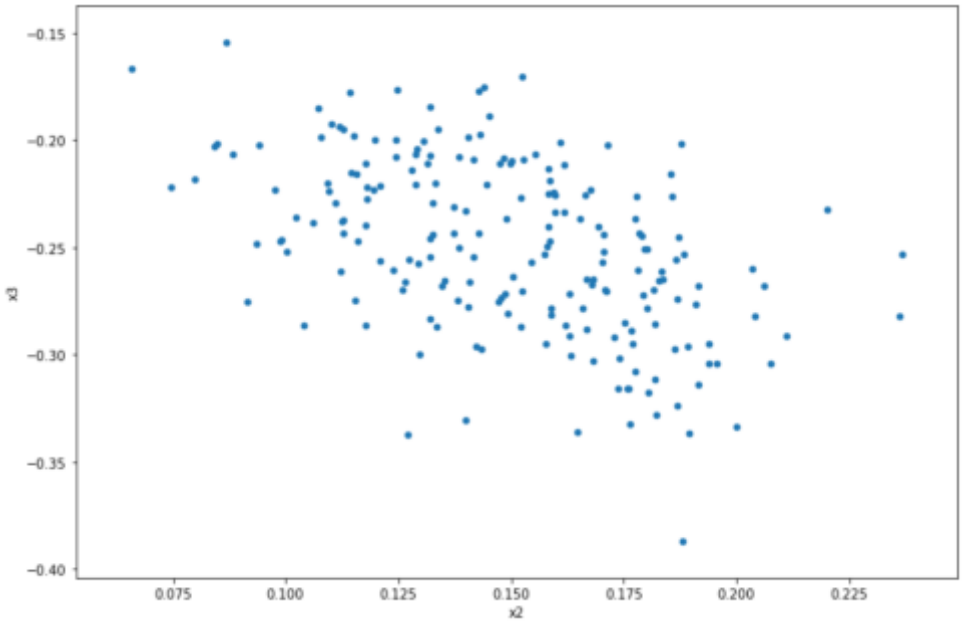
Part A

i. Plotting data using first and second features



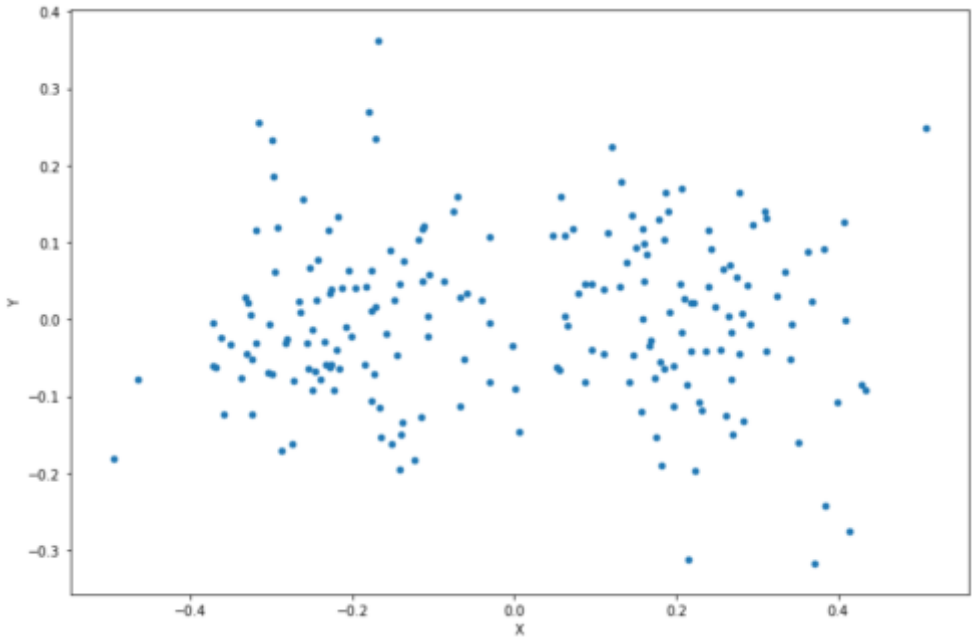
We are randomly choosing the first two features from the given data, but we can't be sure that these features capture the variance among all the points. Thus there is no structure in the plot.

ii. Plotting data using second and third features



Again, the scatter plot between the second and third features does not have any since these features do not capture the information regarding variance within the data.

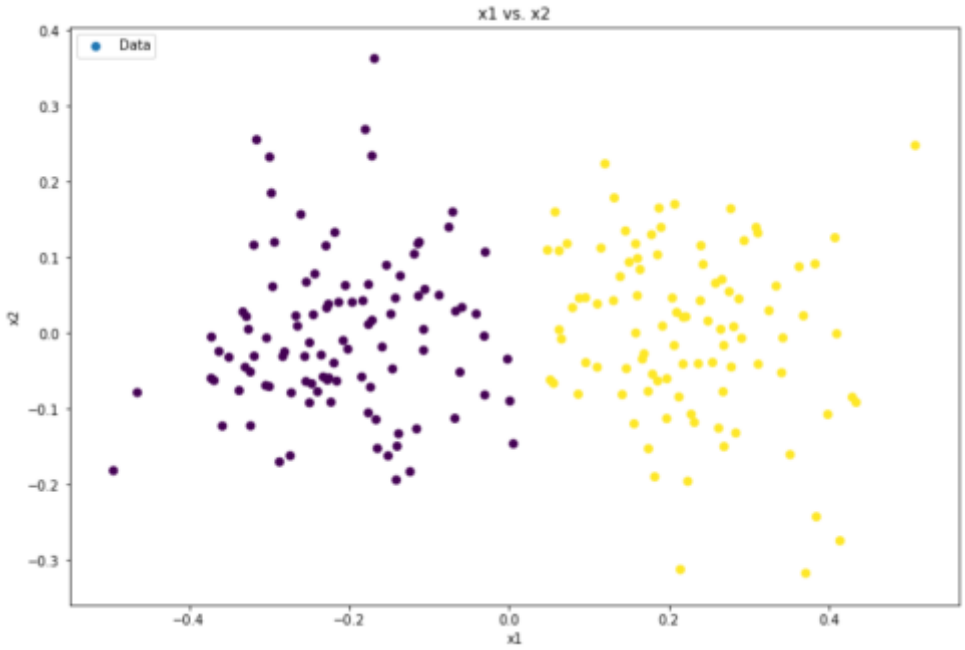
iii. Reducing dimension to 2D using PCA



After applying PCA and reducing the dimensions to 2, the plot of the transformed data shows some structure. This is because PCA found two dimensions that store the maximum variance.

iv. Applying k-means on 40 dimensional data

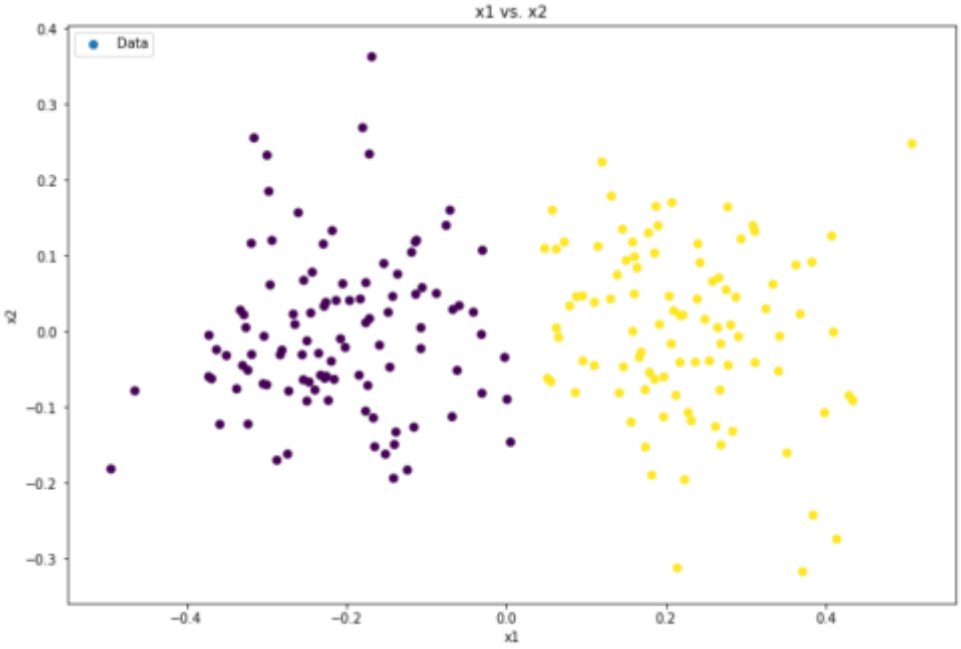
Looking at my PCA plot above, I can see that there are two almost distinct clusters being formed by the data. Hence I keep K = 2 for my K-means algorithm. And it will find me the data divided into two clusters.



Running Kmeans of the data we can see that it is able to recover the true clustering of the data. This is cause we are applying Kmeans of the 40 dimensional data, hence no variance information is lost. Kmeans initializes the cluster centres and then in each iteration it moves the centres till the data is uniformly distributed among the clusters. In the data you can see that the two centres will be far apart from each other, thus Kmeans performs well.

v. Applying k-means on 2 dimensional data

Looking at my PCA plot, I can see that there are two almost distinct clusters being formed by the data. Hence I keep K = 2 for my K-means algorithm. And it will find me the data divided into two clusters.



After reducing the dimensions to 2 by PCA, we retain the maximum variance in data. Hence the clustering information is also retained by these variables. Thus Kmeans performs well here too.

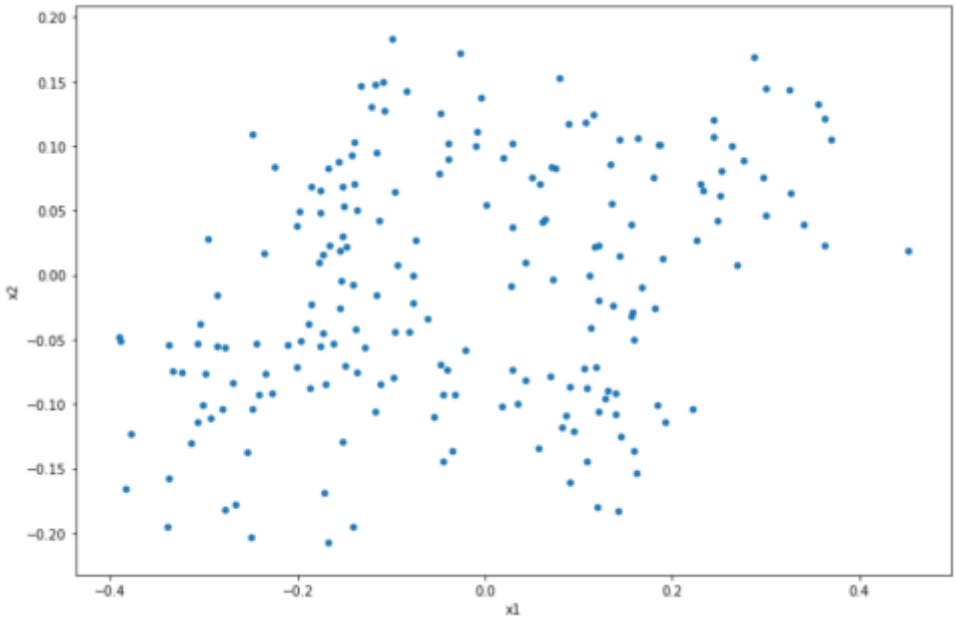
vi. Similarity between (iv) and (v)

Running Kmeans on 40 dimensional and 2 dimensional data gives us similar results. PCA makes sure that the low-dimensional representation of data is representative of the original data. Hence, transforming original data to lower dimension data will not have any adverse effects.

The advantage of applying PCA before K-means is that it saves us a lot of computational cost. As the original data contains a large number of features, running PCA as the first would reduce the dimensionality hence performing operations on low dimension data greatly increases the speed of computation. This can be advantageous when we have limited computing power to process the given data.

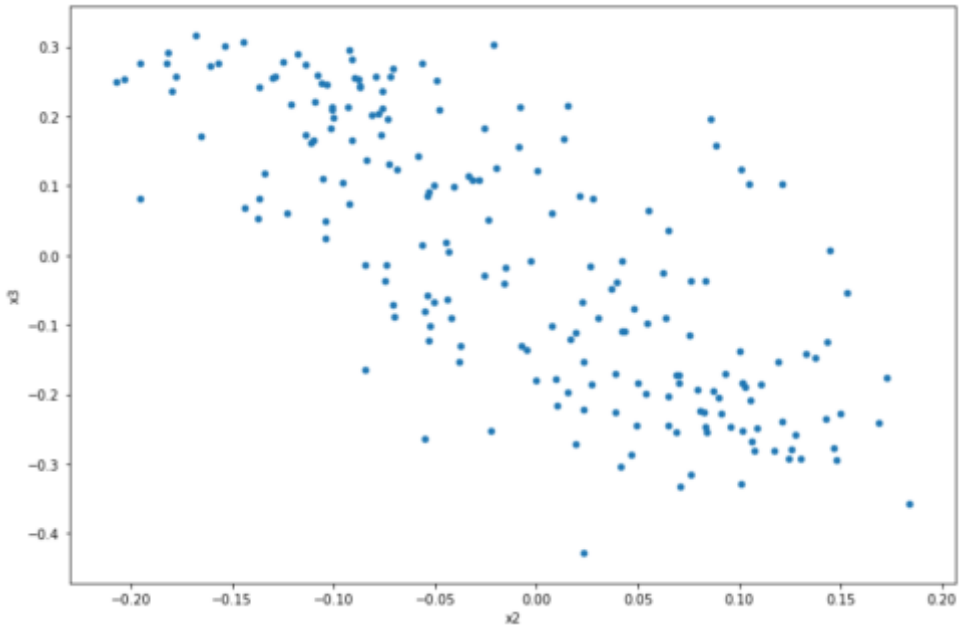
Part B

i. Plotting data using first and second features



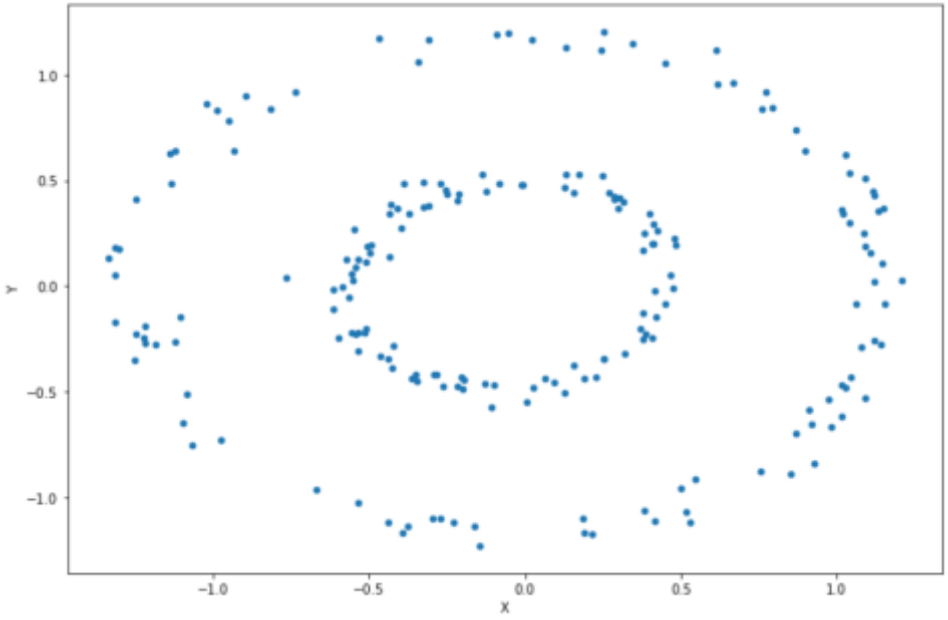
We are randomly choosing the first two features from the given data, but we can't be sure that these features capture the variance among all the points. Thus there is no structure in the plot.

ii. Plotting data using second and third features



Again, the scatter plot between the second and third features does not have any since these features do not capture the information regarding variance within the data.

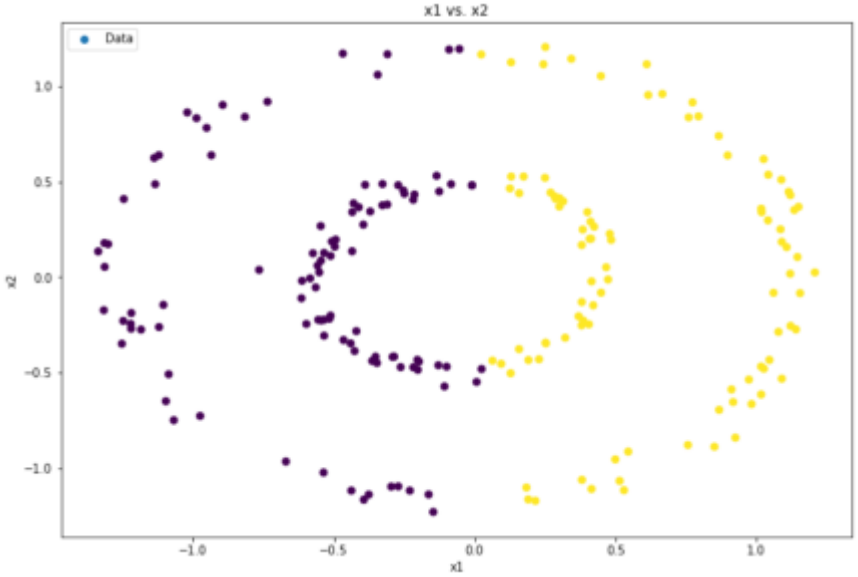
iii. Reducing dimension to 2D using PCA



After applying PCA and reducing the dimensions to 2, the plot of the transformed data shows some structure. This is because PCA found two dimensions that store the maximum variance.

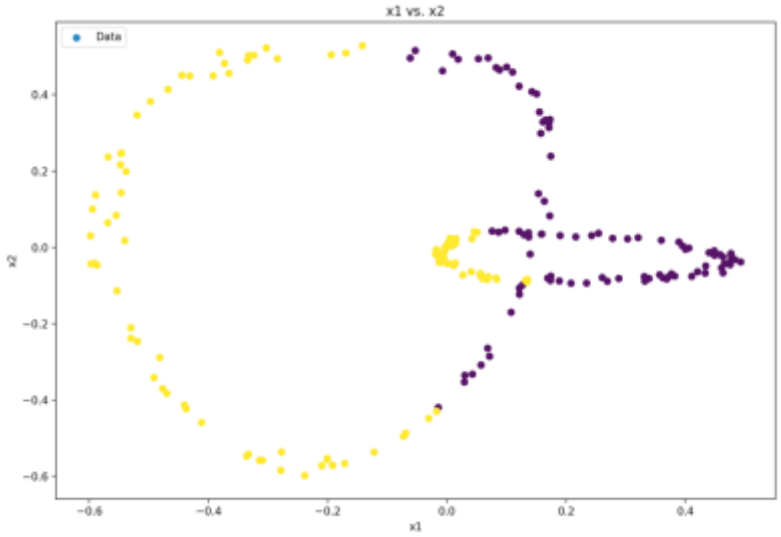
iv. Applying kmeans on 2 dimensional data obtained using PCA

Looking at my PCA plot above, I can see that there are two almost distinct clusters being formed by the data. Hence I keep K = 2 for my Kmeans algorithm. And it will find me the data divided into two clusters. The two clusters here are the outer and inner rings.



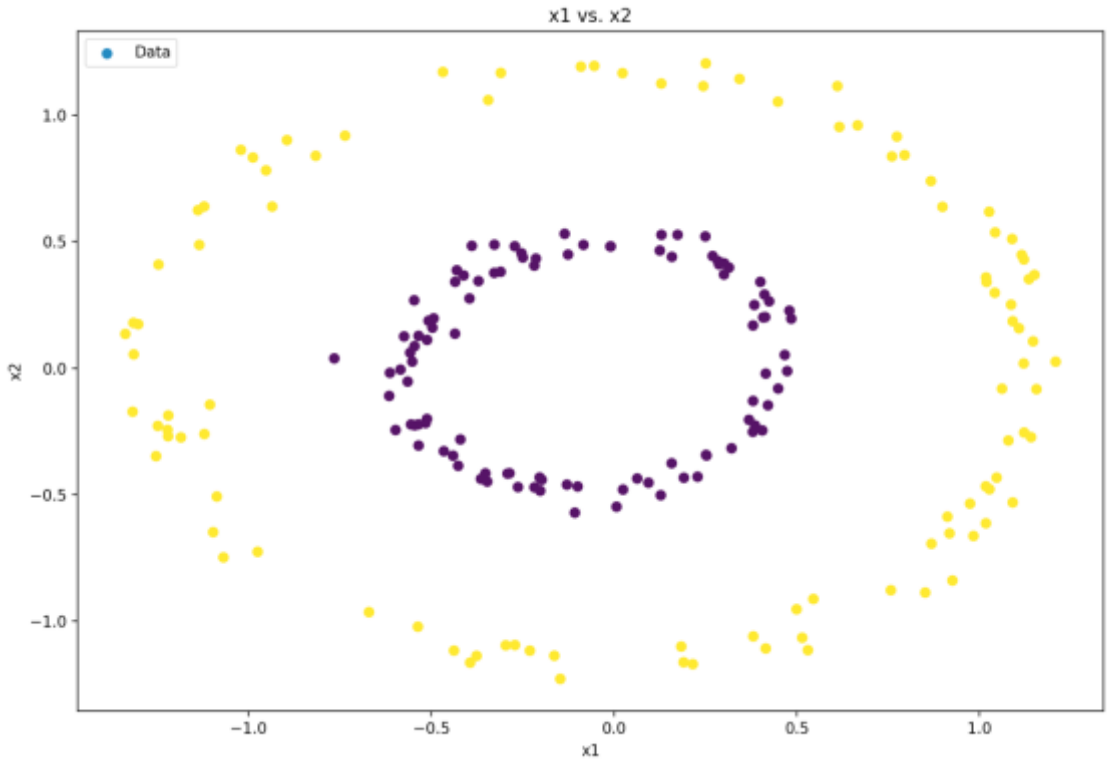
K-means here is not able to extract the true clustering in the data since the mean centres of both the circles are at the same point. As K-means tries to create uniform clusters after initializing the centroids and then moving it away from each other in our case, the true clustering is lost.

v. Applying k-means on 2 dimensional obtained using KPCA



After using KPCA with a variance of 0.3 I got the following output. Since using kernels helps transform the data to higher dimensions which makes the data more linearly separable. Here you can see the transformation to my data. And since the centres are no more at the same place we get a better result.

vi. Applying spectral clustering



Spectral clustering is successful in extracting the true clustering of the given data because when we run KNN on the given data, it generates a similarity matrix that captures the information regarding components of the graph. The two components of our graph here would be the two rings.

PCA results in two concentric circles or rings. Since the similarity graph captures this information using KNN, spectral clustering is able to group the points in outer and inner rings as two different clusters.