Q1 (i) : The variance in the data-set (centered in this case) explained by each of the principal components is :

- variance along component 1 = 54.178 %
- variance along component 2 = 45.822 %

Figure : Output Variance Console

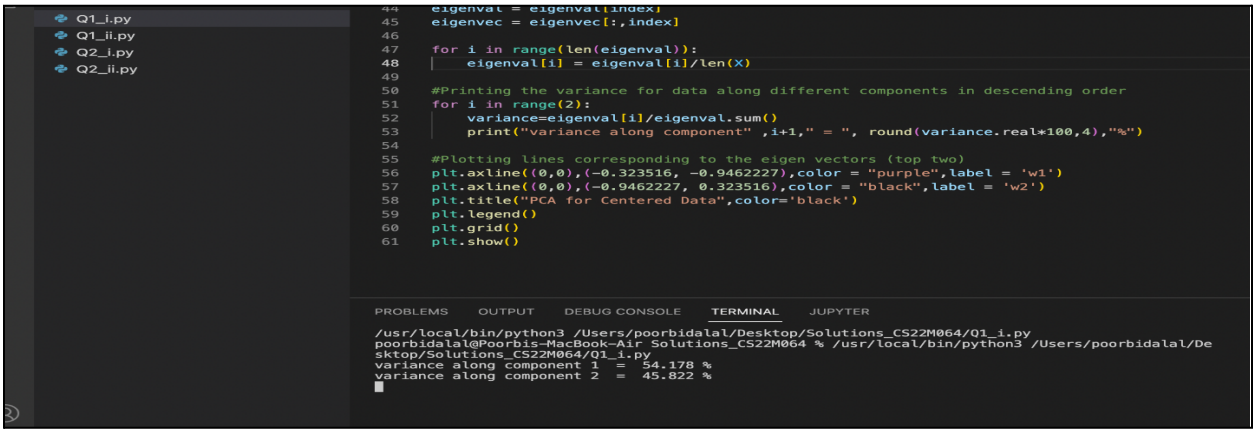
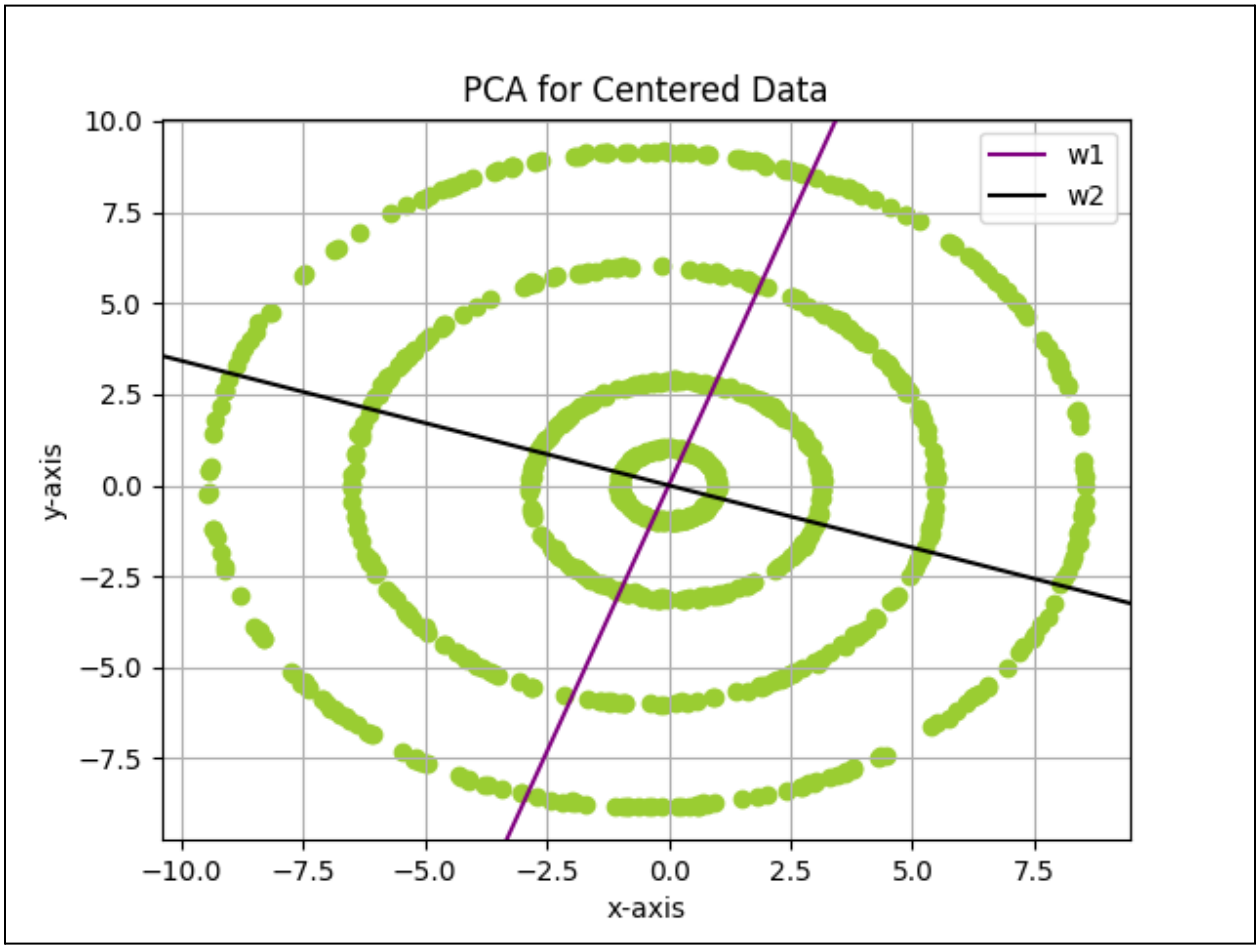


Figure : PCA for Centered Data



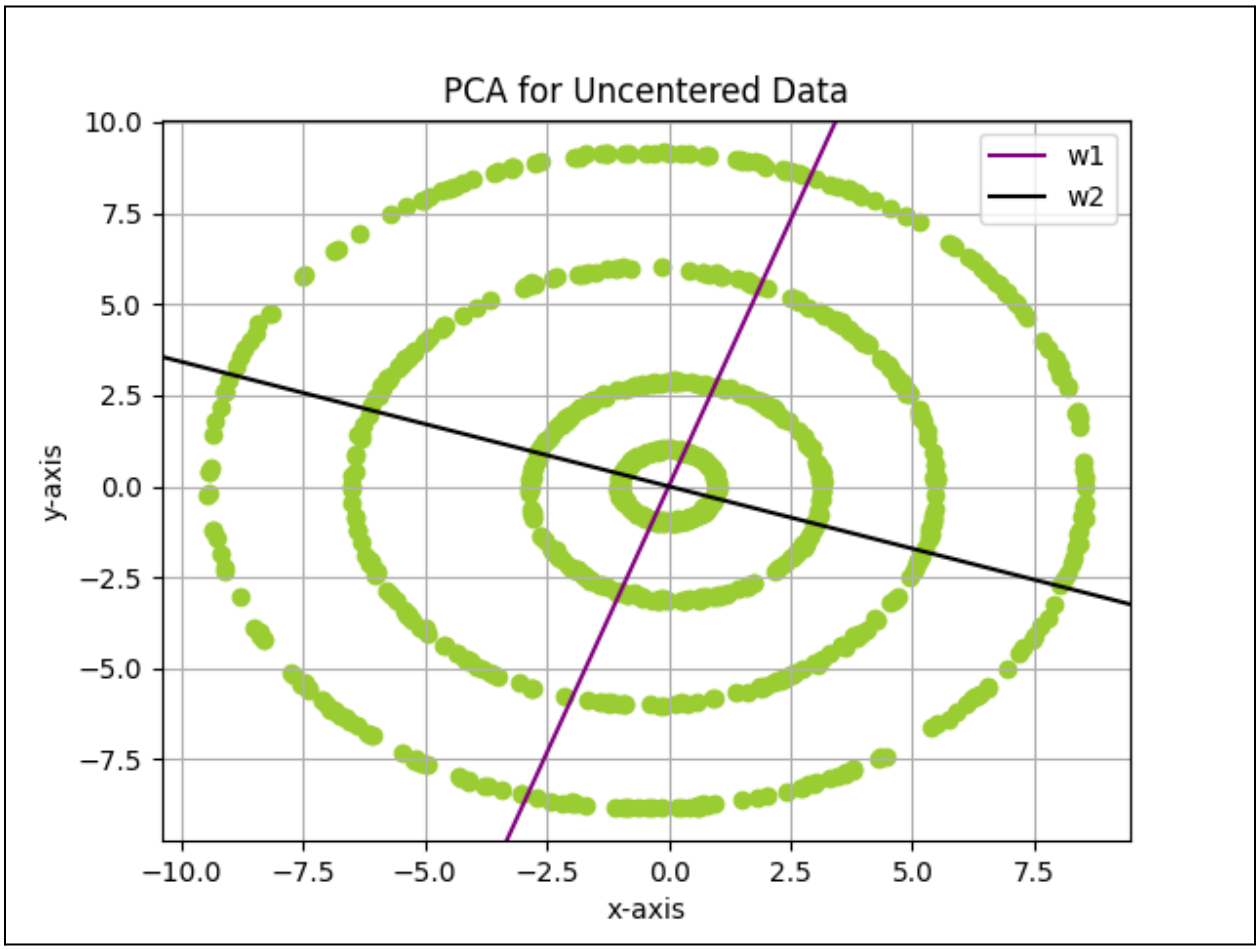
Q1(ii) : The variance in the data-set (not centered in this case) explained by each of the principal components is :

- variance along component 1 = 54.178 %
- variance along component 2 = 45.822 %

Mean of data = [4.0749999997216625e-07, 2.2269999997881483e-07]

As per the mean calculated from the data we can infer that it is nearly negligible. Hence the effect of centering comes out to be insignificant in this case. We can also infer that the plot of the data with and without centering is almost similar.

Figure: PCA for uncentered data



Q1(iii): A.

Figure : Polynomial Function Kernel PCA for degree = 2

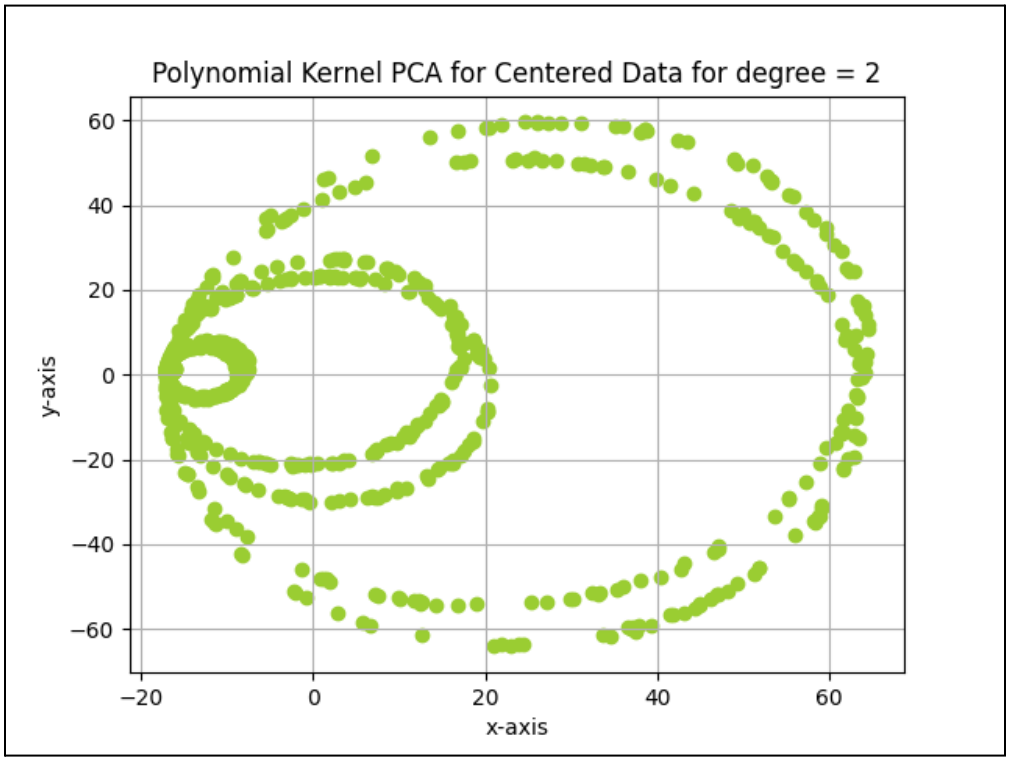
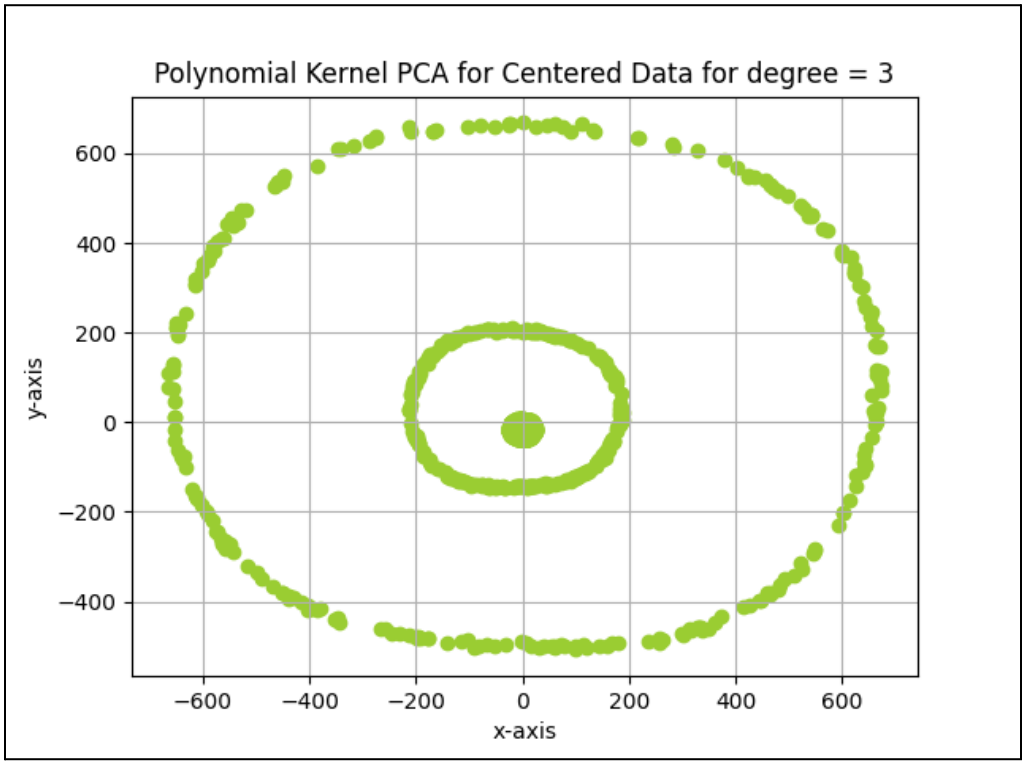


Figure : Polynomial Function Kernel PCA for degree = 3



Q1(iii): B.

Figure : Radial Basis Function Kernel PCA for sigma = 0.1

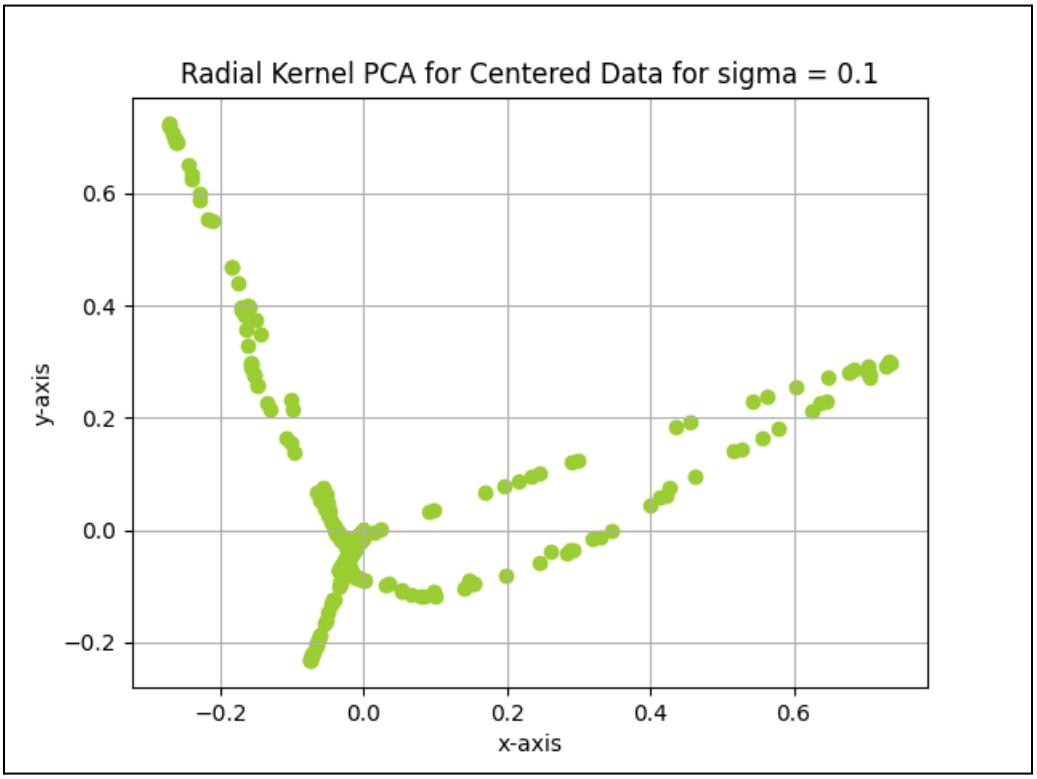


Figure : Radial Basis Function Kernel PCA for sigma = 0.2

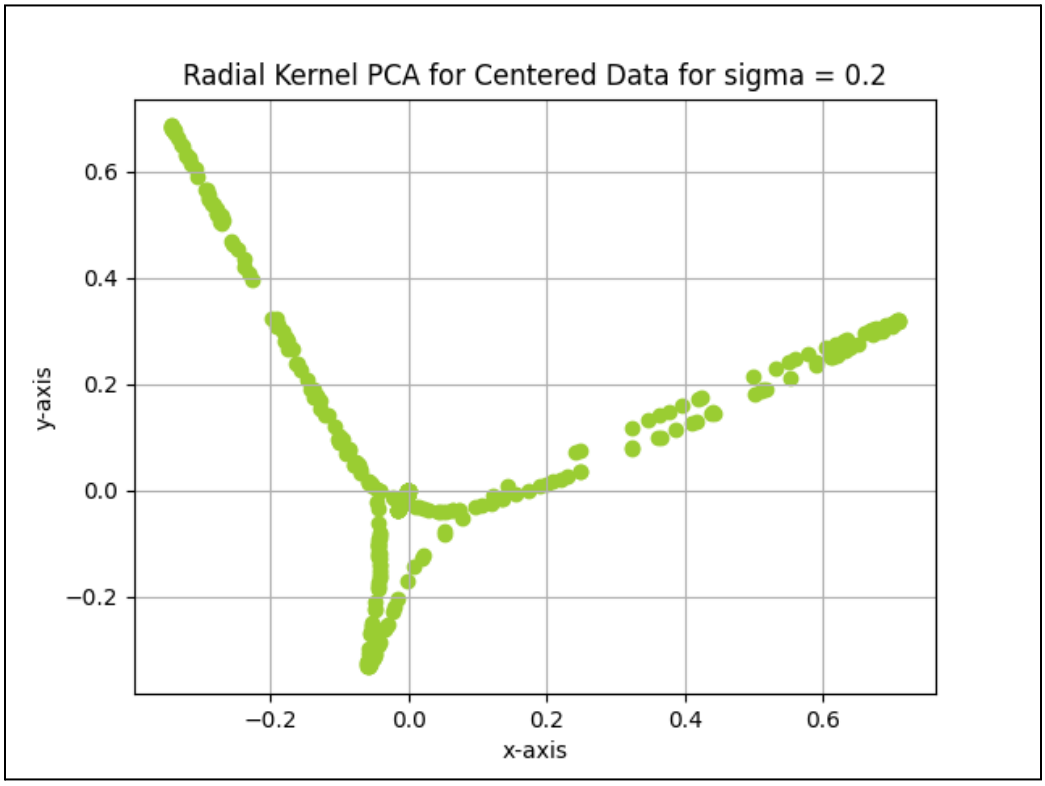


Figure : Radial Basis Function Kernel PCA for sigma = 0.3

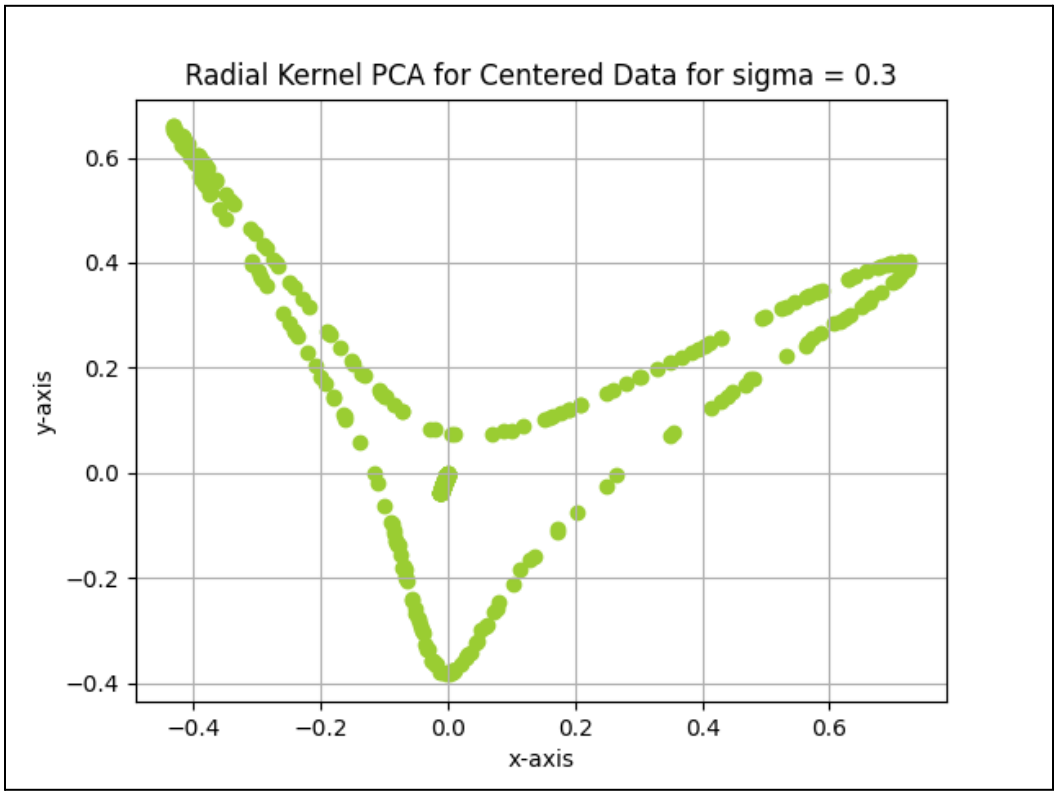


Figure : Radial Basis Function Kernel PCA for sigma = 0.4

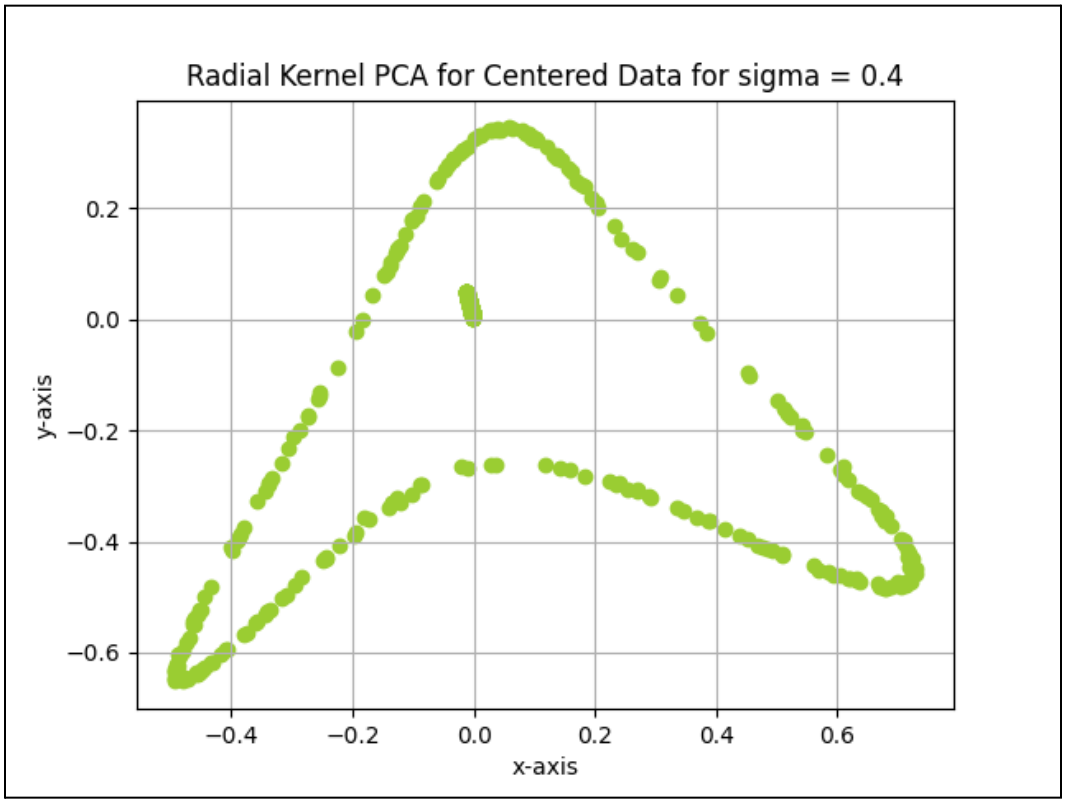


Figure : Radial Basis Function Kernel PCA for sigma = 0.5

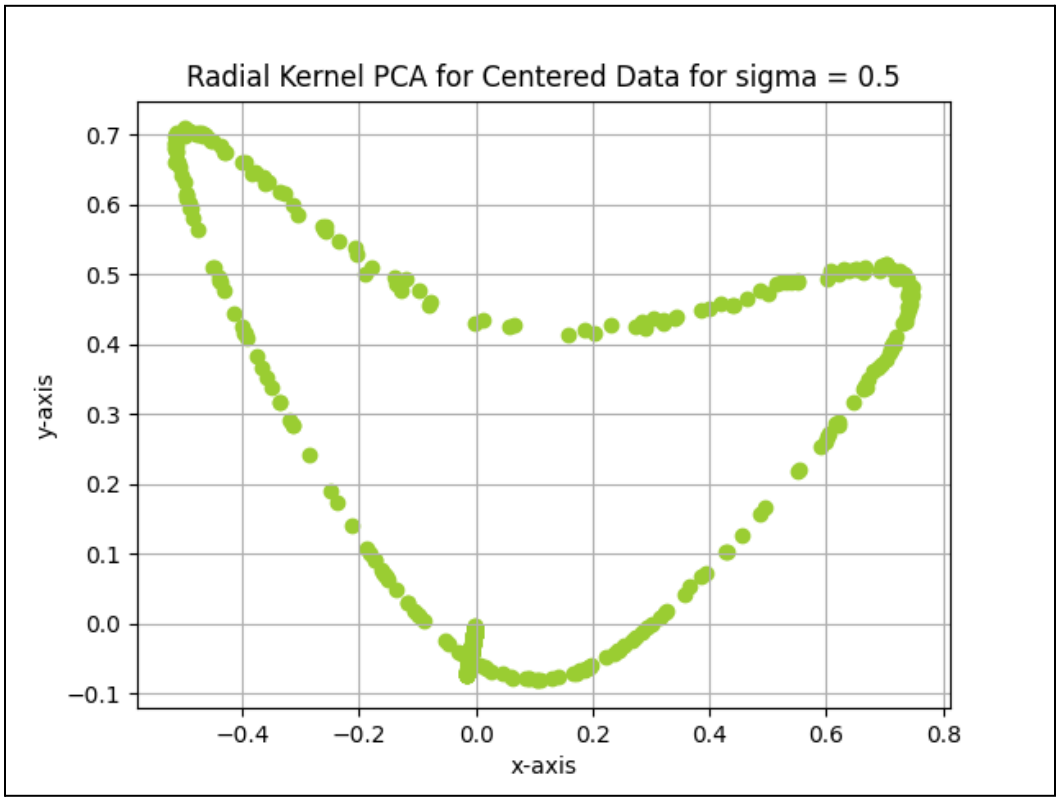


Figure : Radial Basis Function Kernel PCA for sigma = 0.6

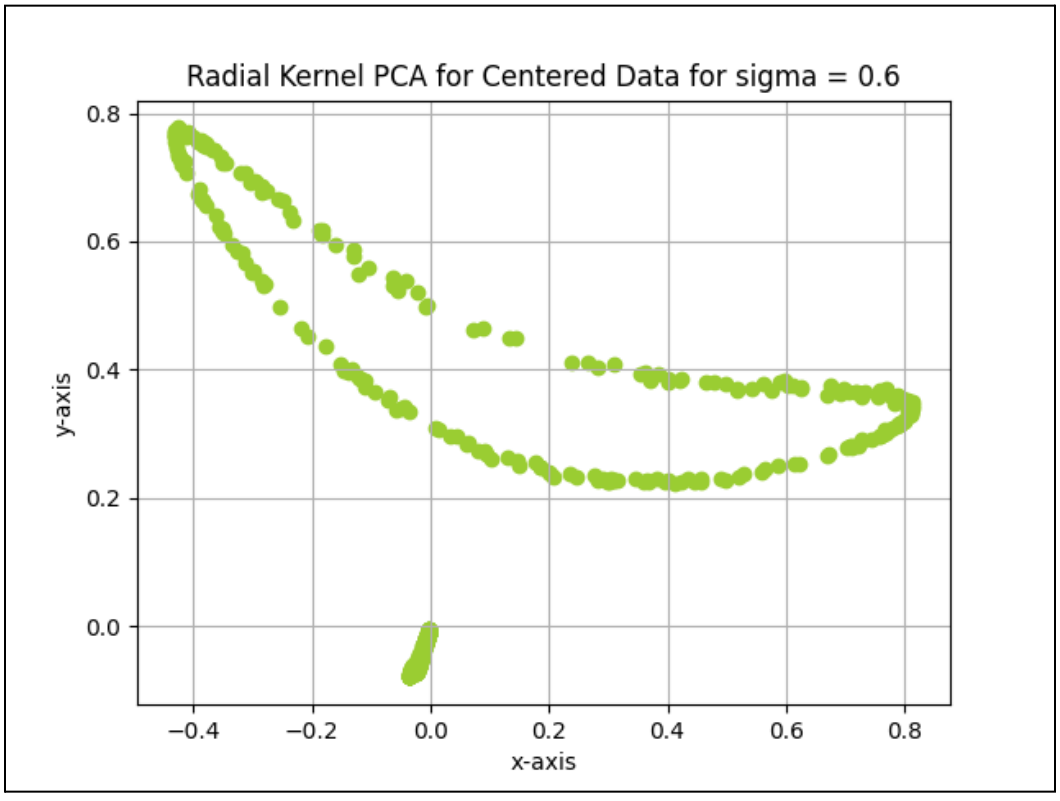


Figure : Radial Basis Function Kernel PCA for sigma = 0.7

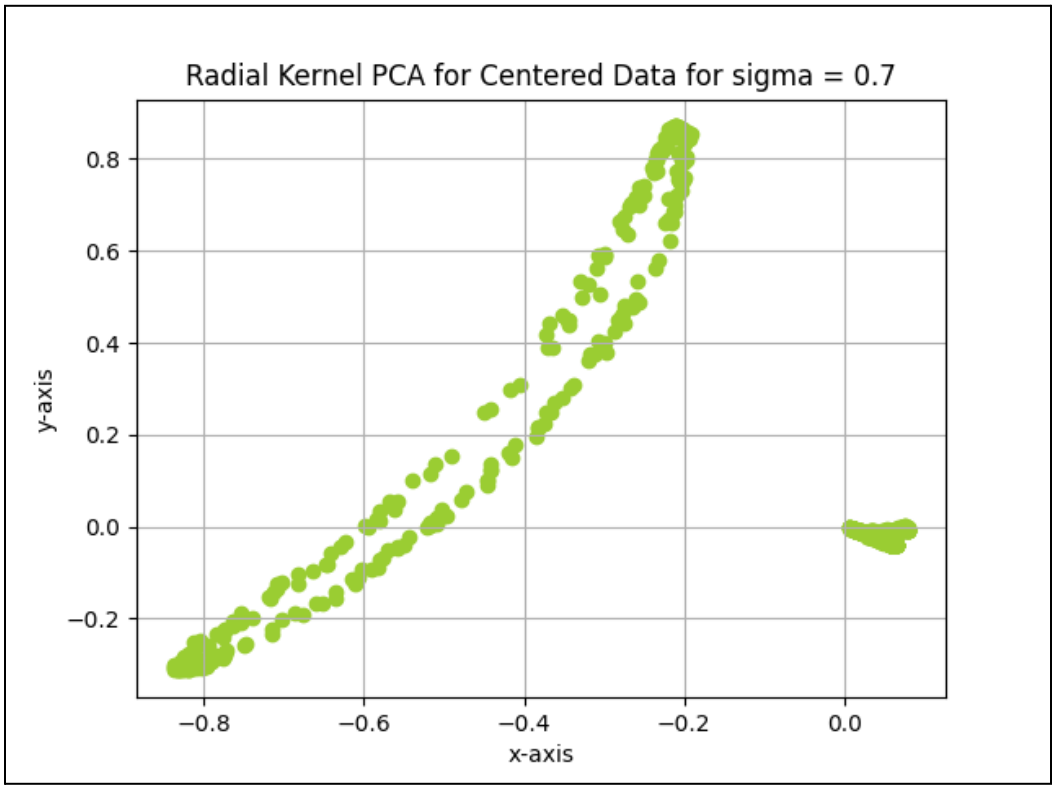


Figure : Radial Basis Function Kernel PCA for sigma = 0.8

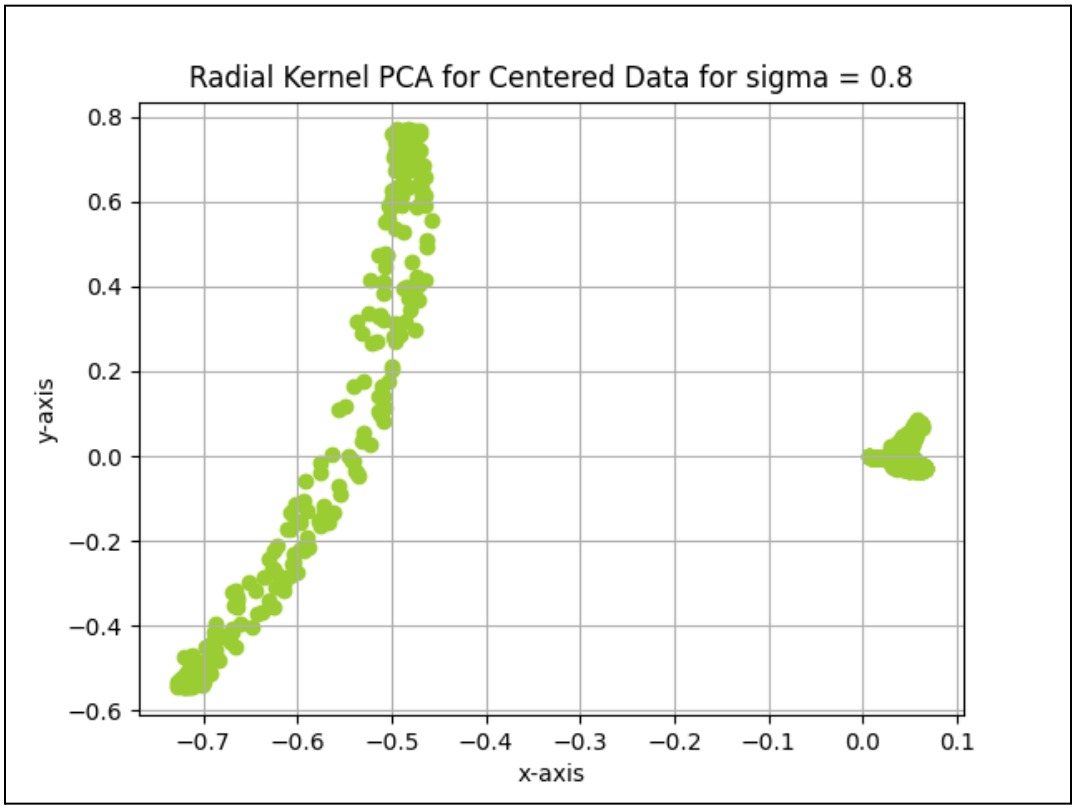


Figure : Radial Basis Function Kernel PCA for sigma = 0.9

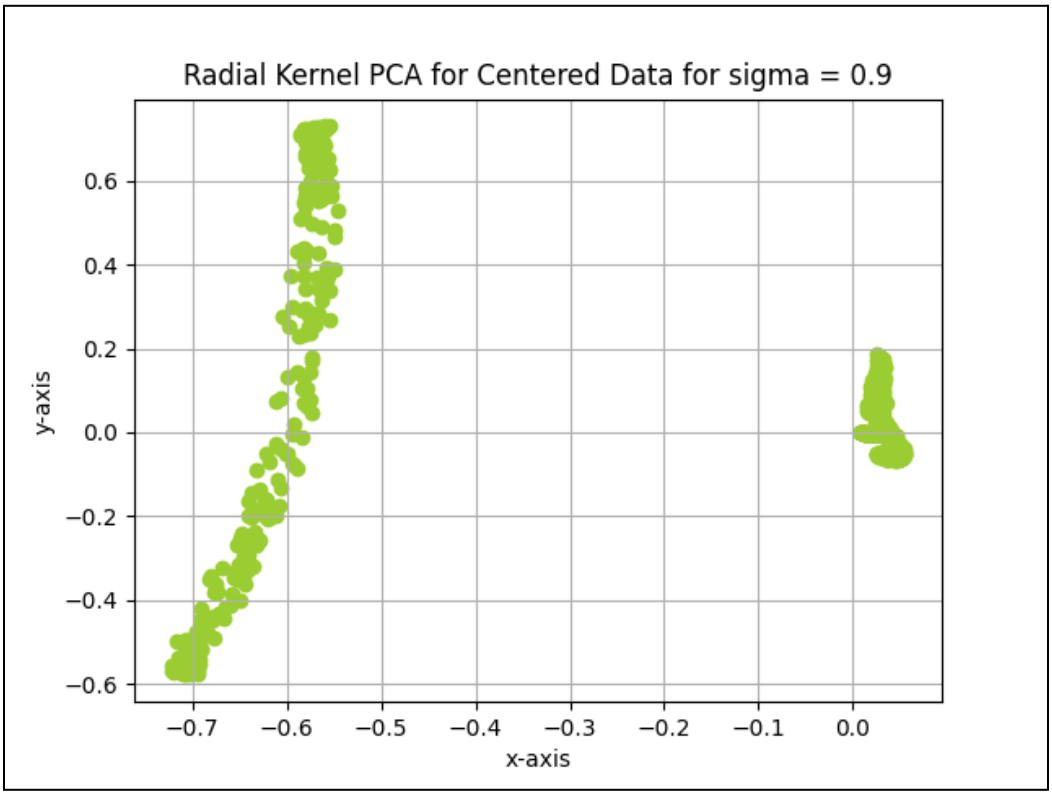
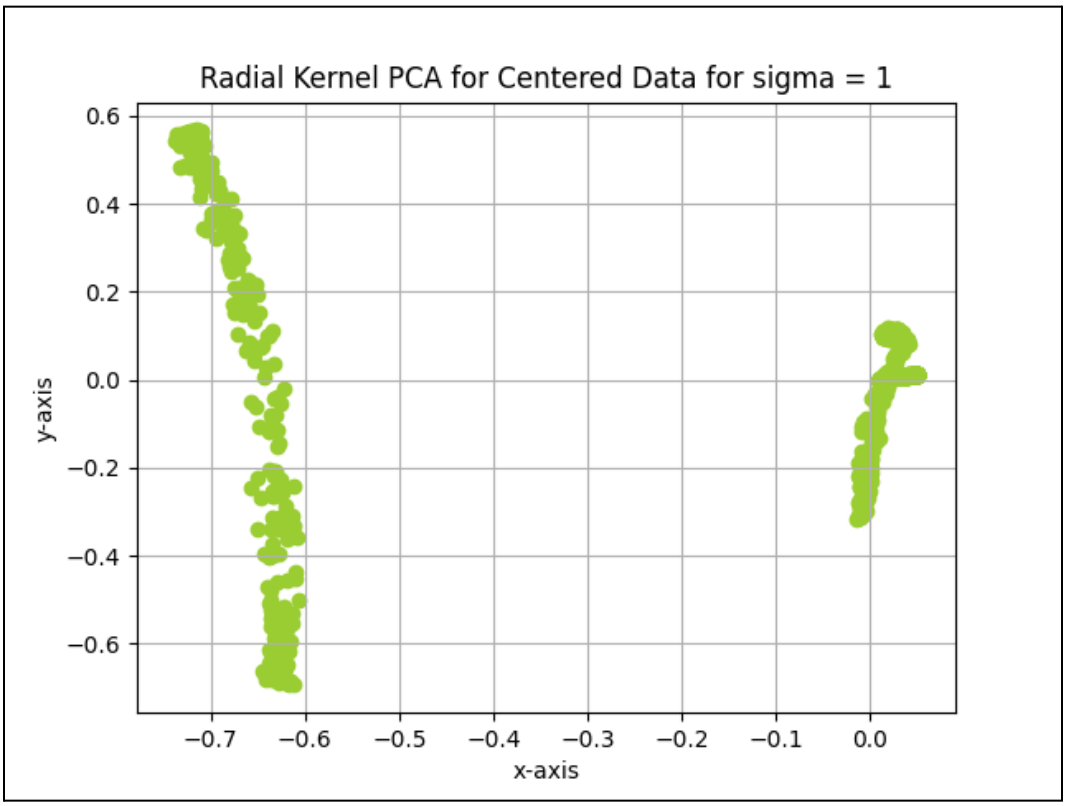


Figure : Radial Basis Function Kernel PCA for sigma = 1



Q1(iv) :

The kernel function that is best suited for this data set is the Polynomial kernel function with degree 3.

The reason for Polynomial Kernel function being better than Radial Basis Kernel function for this data set is that the maximum variance (corresponding to component with highest eigenvalue) obtained from the top two components obtained from Radial Basis Kernel functions is approximately 15% which isn't high enough. However the maximum variance obtained from Polynomial Kernel function is approximately 73%.

The Radial Basis Kernel function does not provide high variance because of which the spread out of plot isn't satisfying and hence original data point identification becomes hard. From the projections of the top two components obtained from Radial Basis Function we are not able to reconstruct the original data set (identify the points of original data) significantly similar.

The best result for this data is shown by the Polynomial Kernel function with degree = 3. It provides the highest variance about 73%, thus providing a high spread out.The Polynomial Kernel function with degree = 2 provides a variance of approximately 68% which is slightly less than one obtained by the Polynomial Kernel function with degree = 3. It also helps in significant reconstruction of the data set.

Q2(i) :

Figure : K-means Clustering for k=4 (1st Random Initialization)

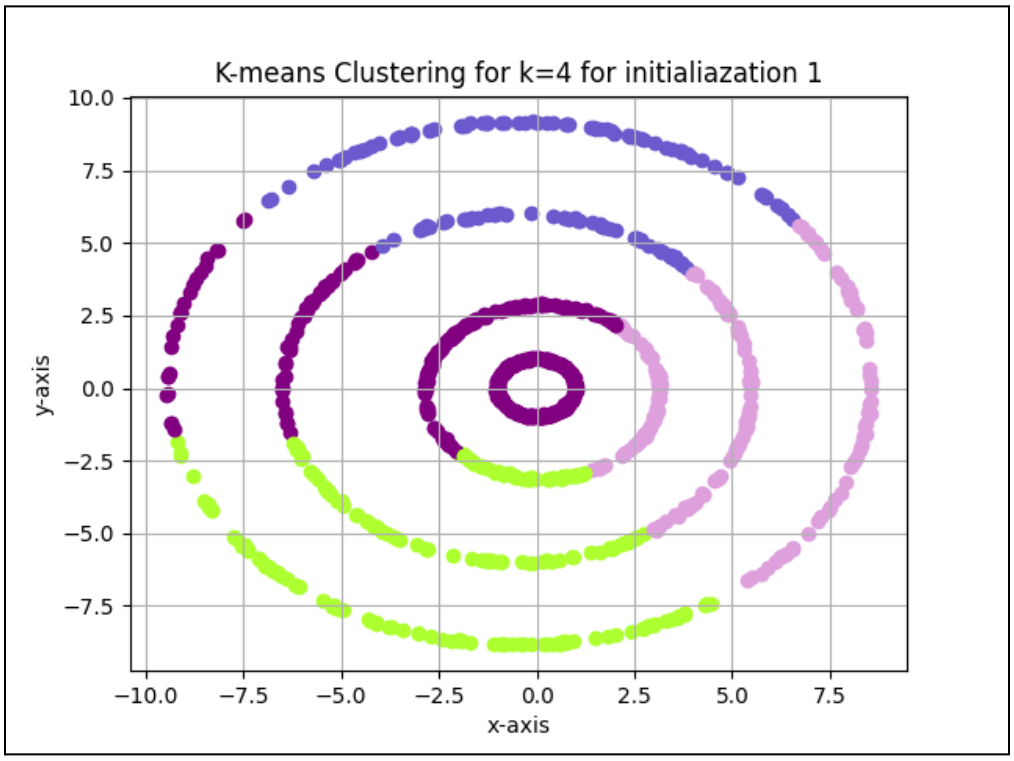


Figure : Error Function (1st Random Initialization)

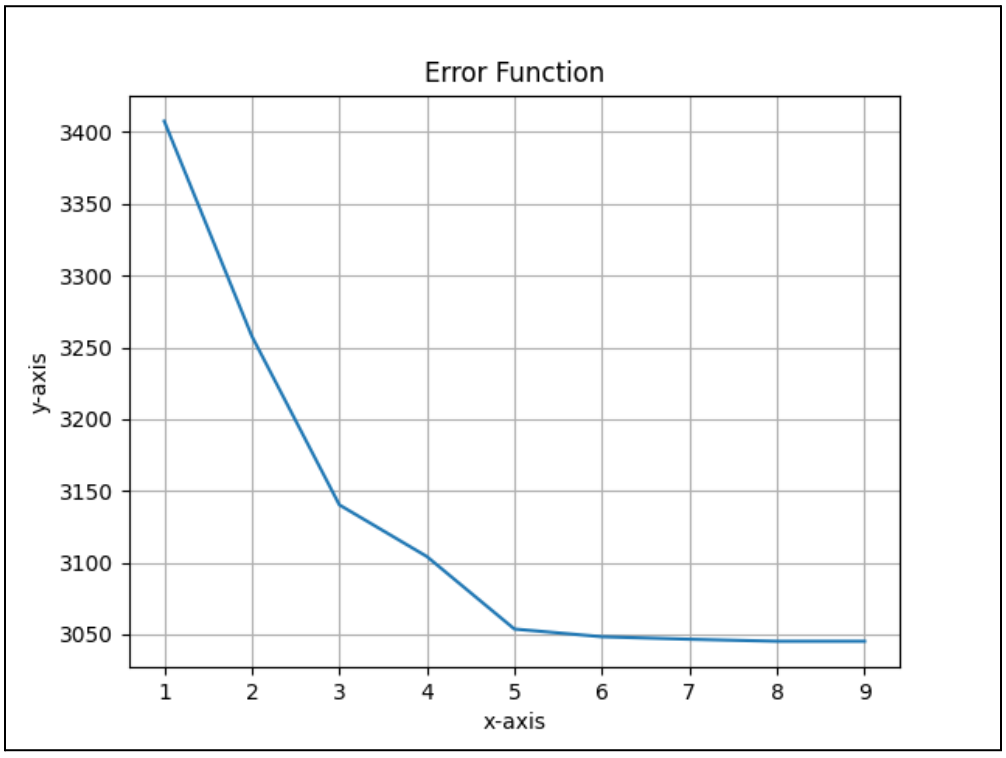


Figure : K-means Clustering for k=4 (2nd Random Initialization)

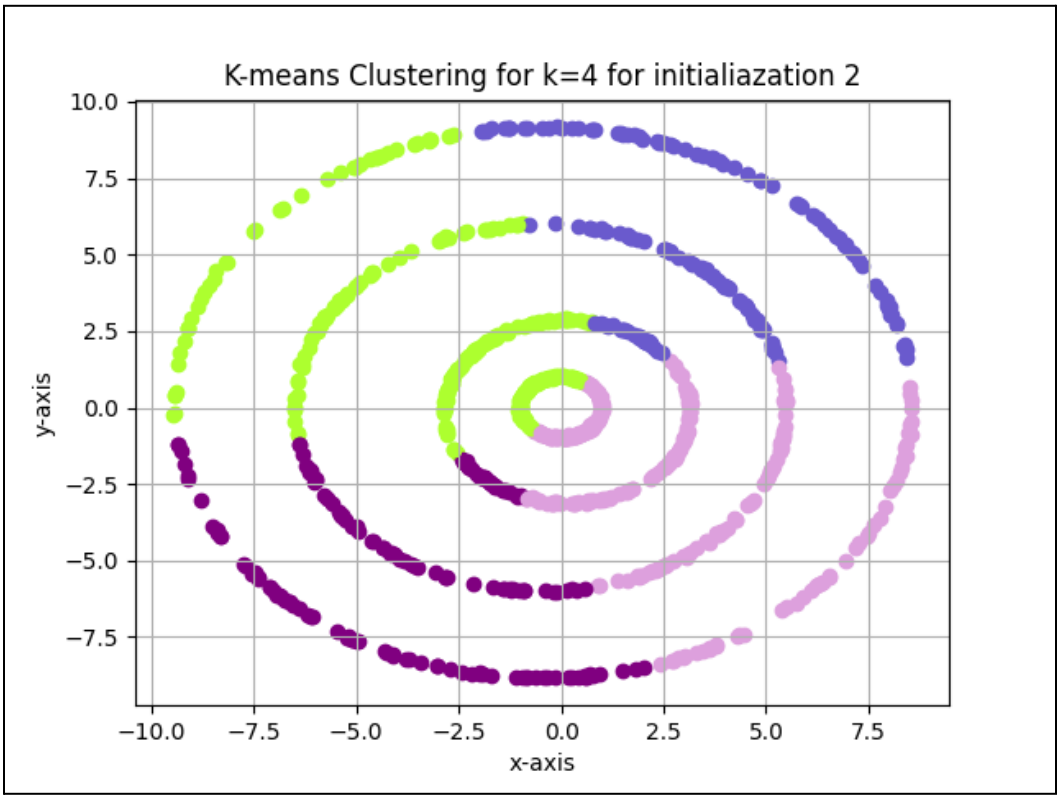


Figure : Error Function (2nd Random Initialization)

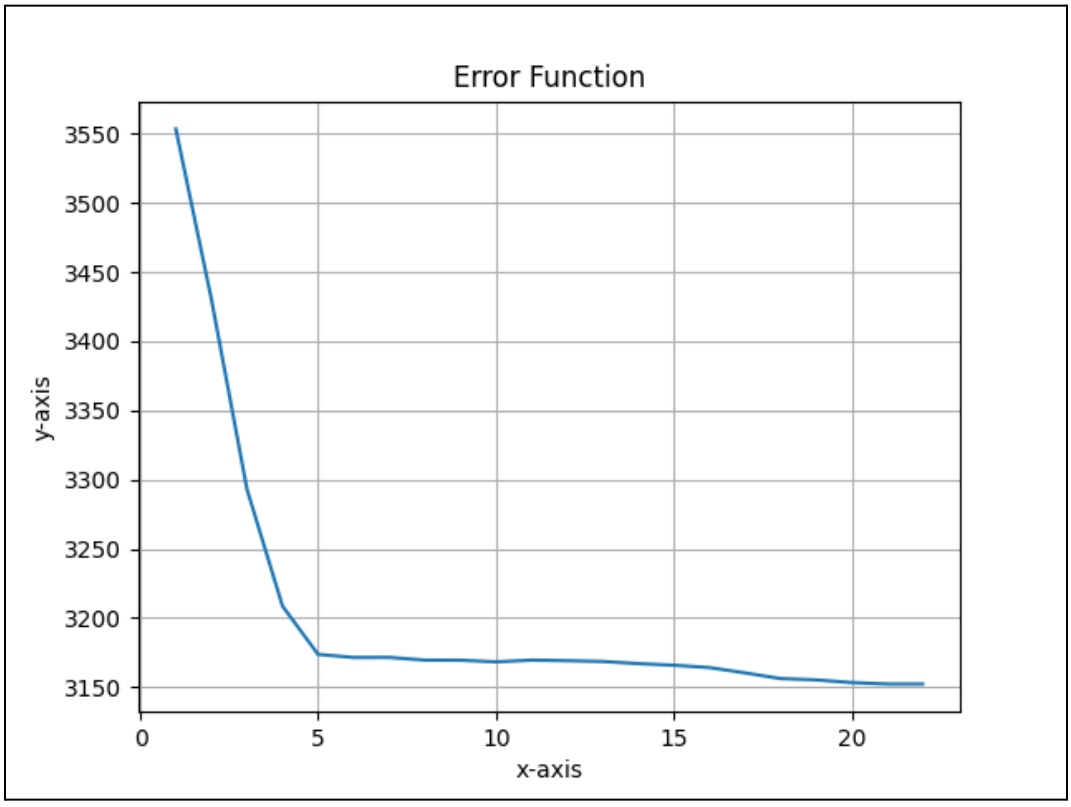


Figure : K-means Clustering for k=4 (3rd Random Initialization)

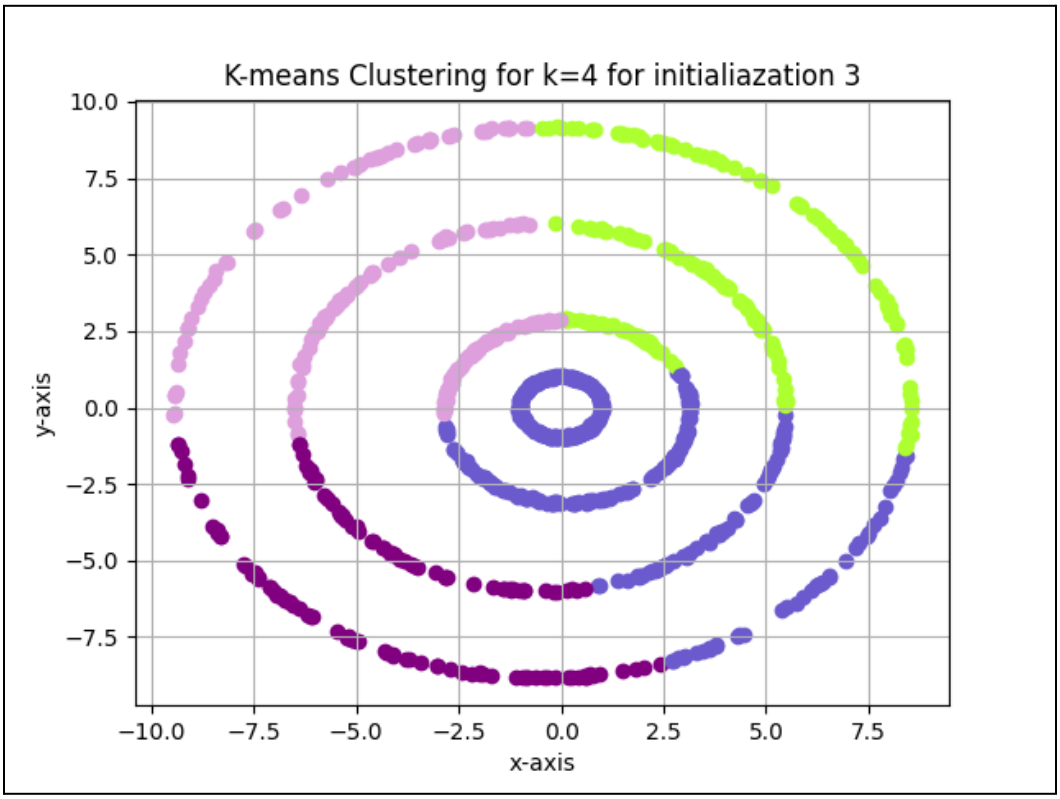


Figure : Error Function (3rd Random Initialization)

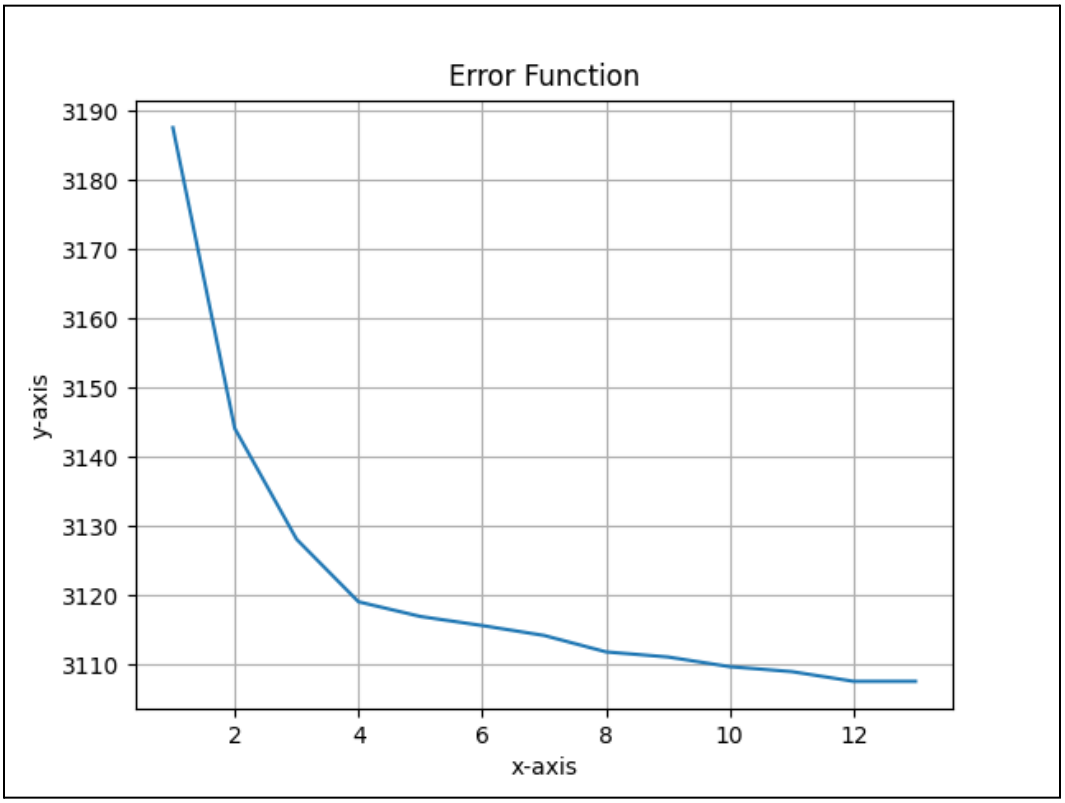


Figure : K-means Clustering for k=4 (4th Random Initialization)

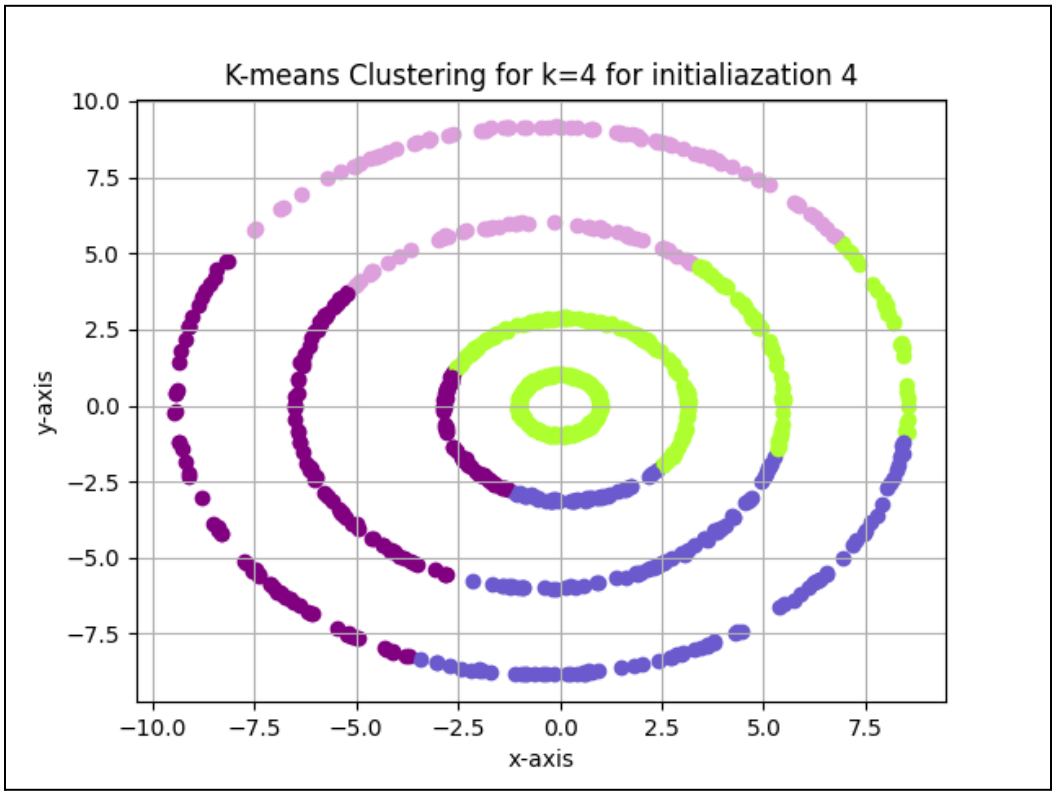


Figure : Error Function (4th Random Initialization)

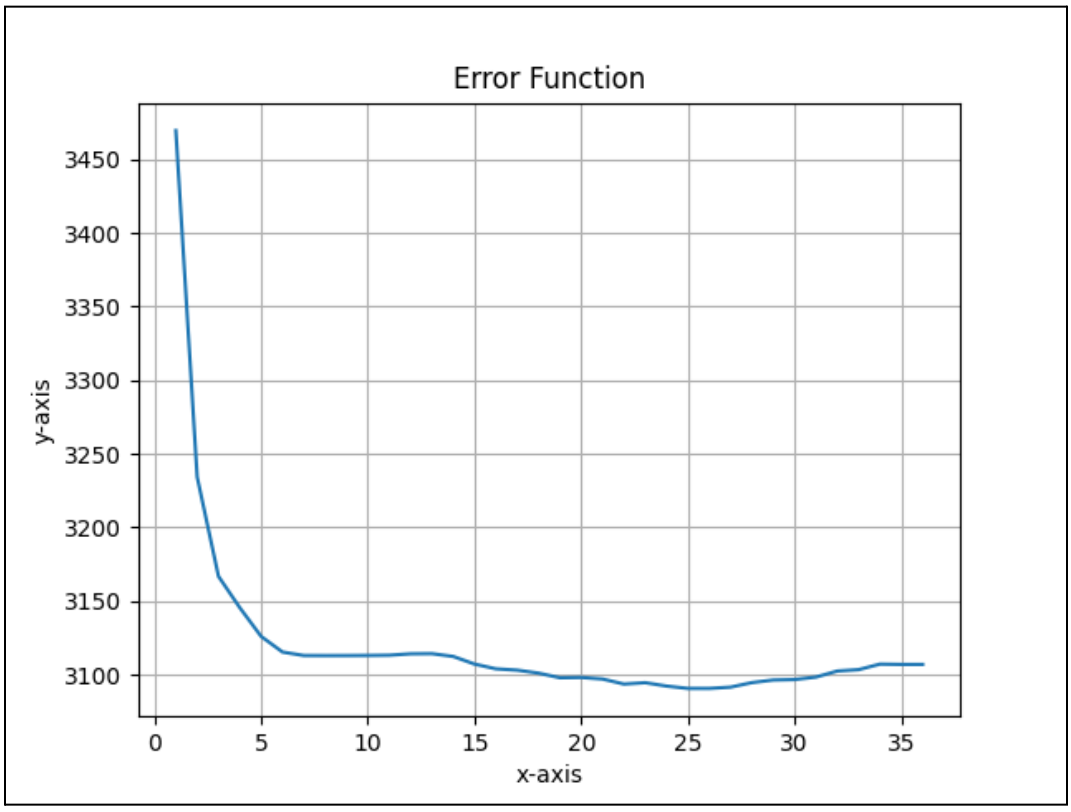


Figure : K-means Clustering for k=4 (5th Random Initialization)

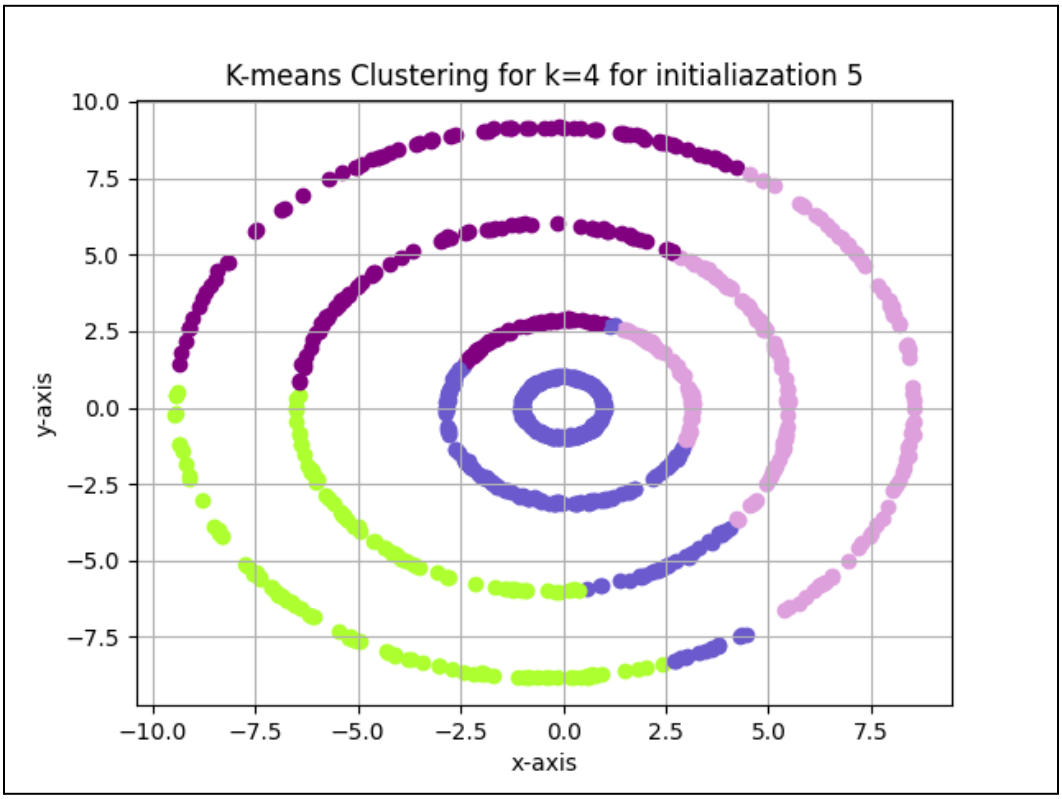
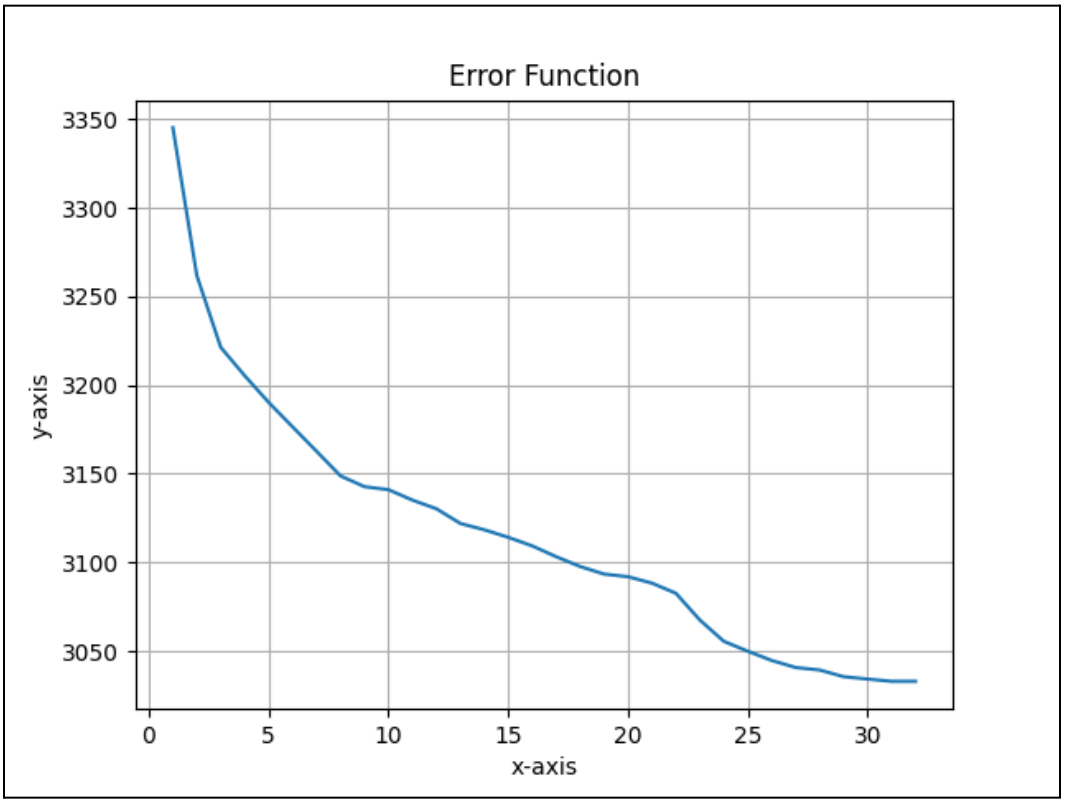


Figure : Error Function (5th Random Initialization)



Q2(iii):

Figure : K-means Clustering for k=2

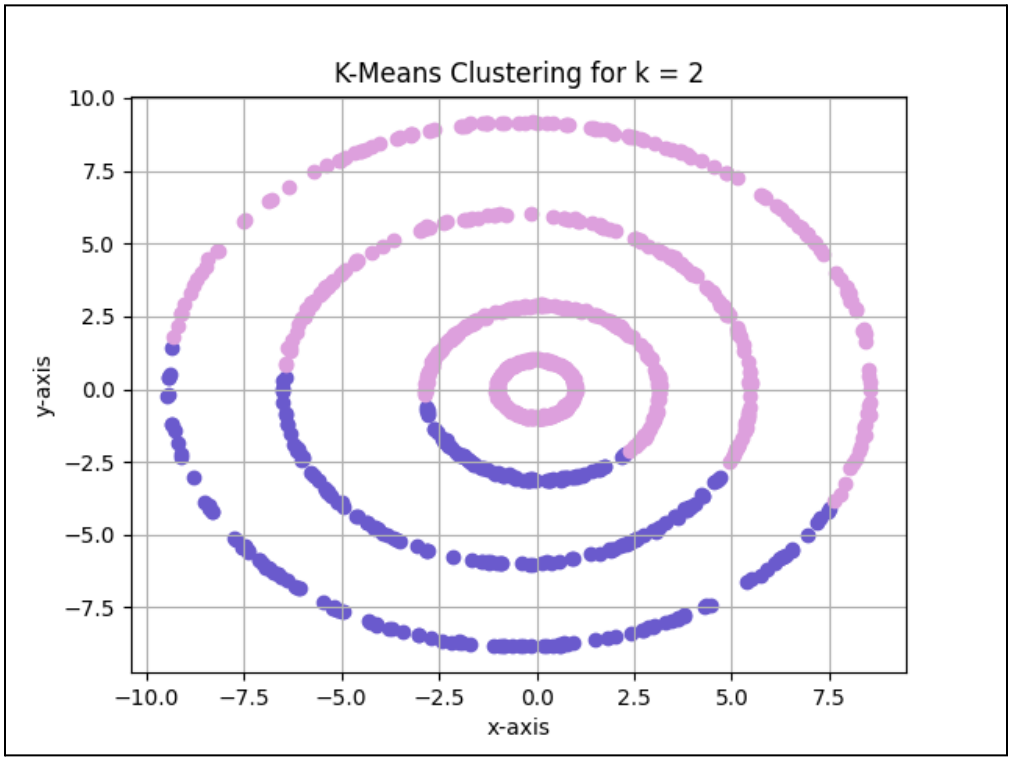


Figure : Error Function of K-means Clustering for k=2

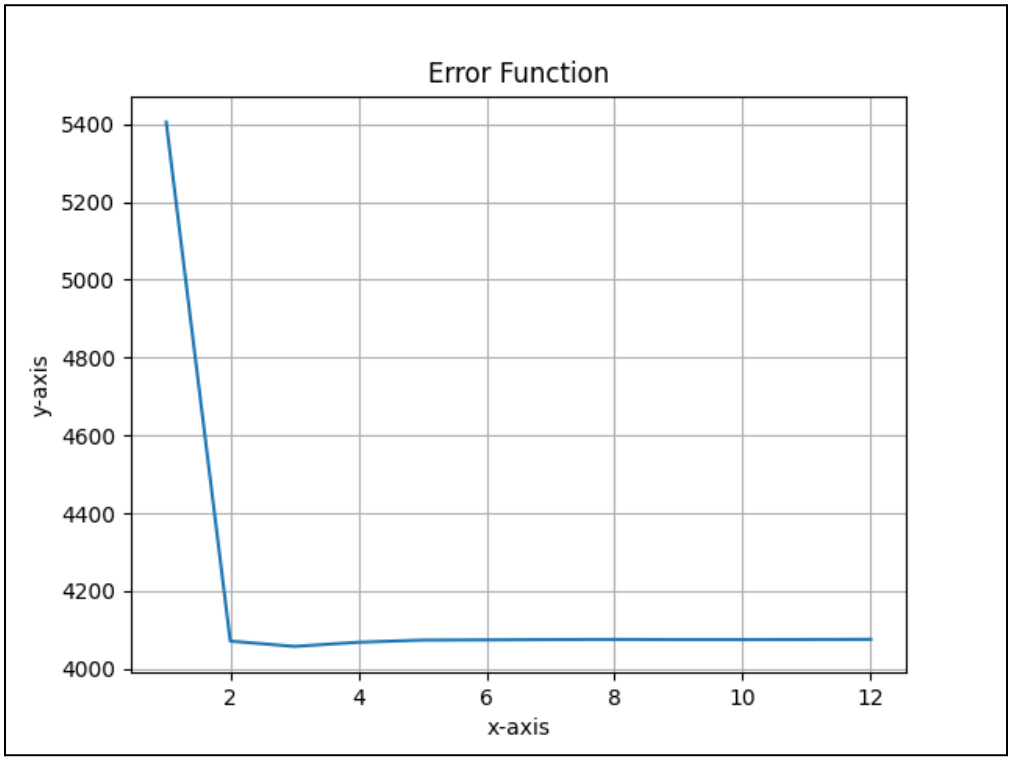


Figure : Voronoi Regions of K-means Clustering for k=2

Figure : K-means Clustering for k=3

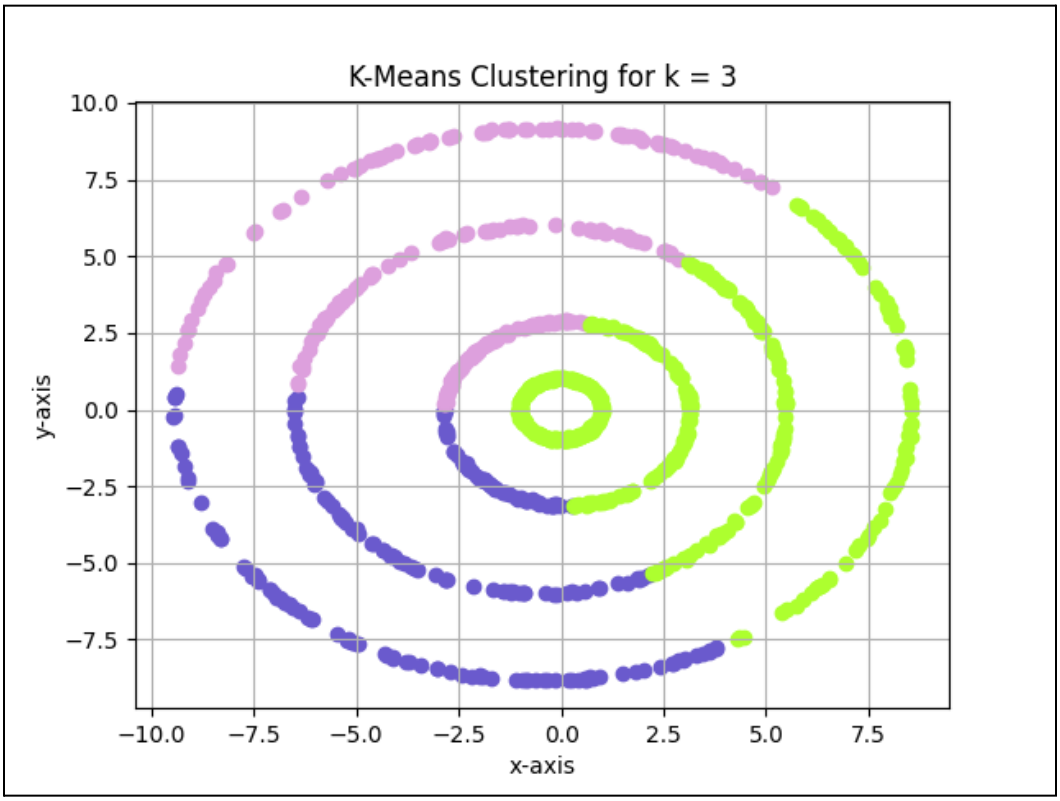


Figure : Error Function of K-means Clustering for k=3

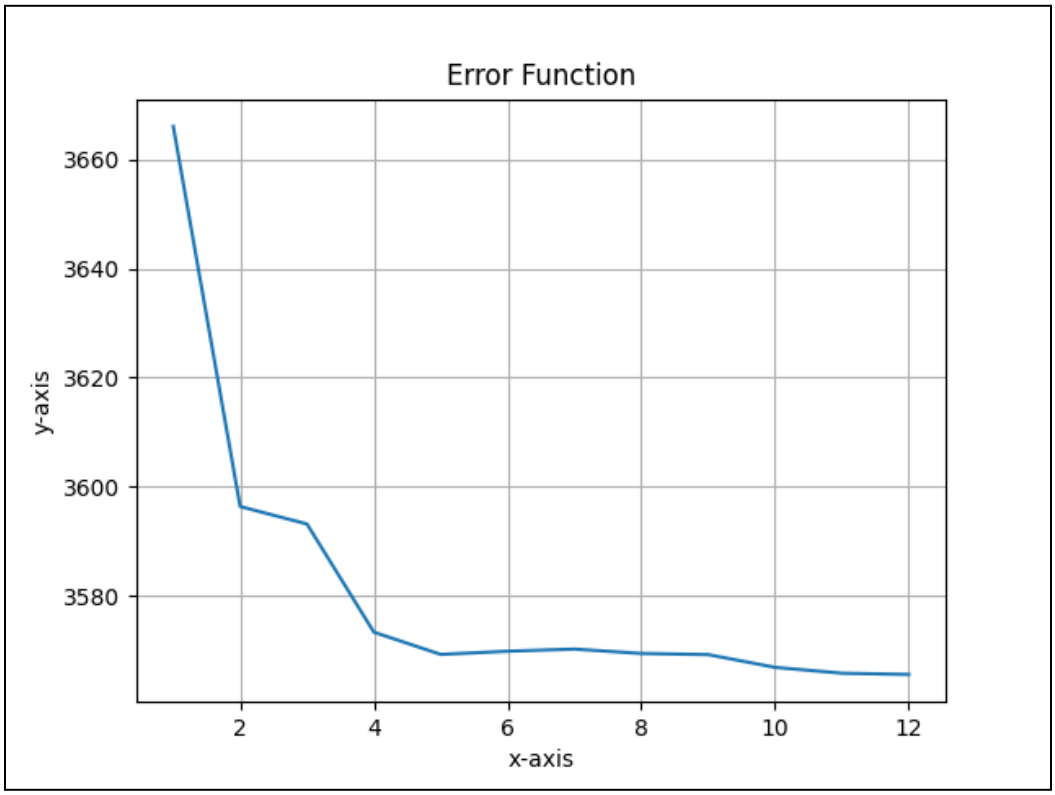


Figure : Voronoi Regions of K-means Clustering for k=3

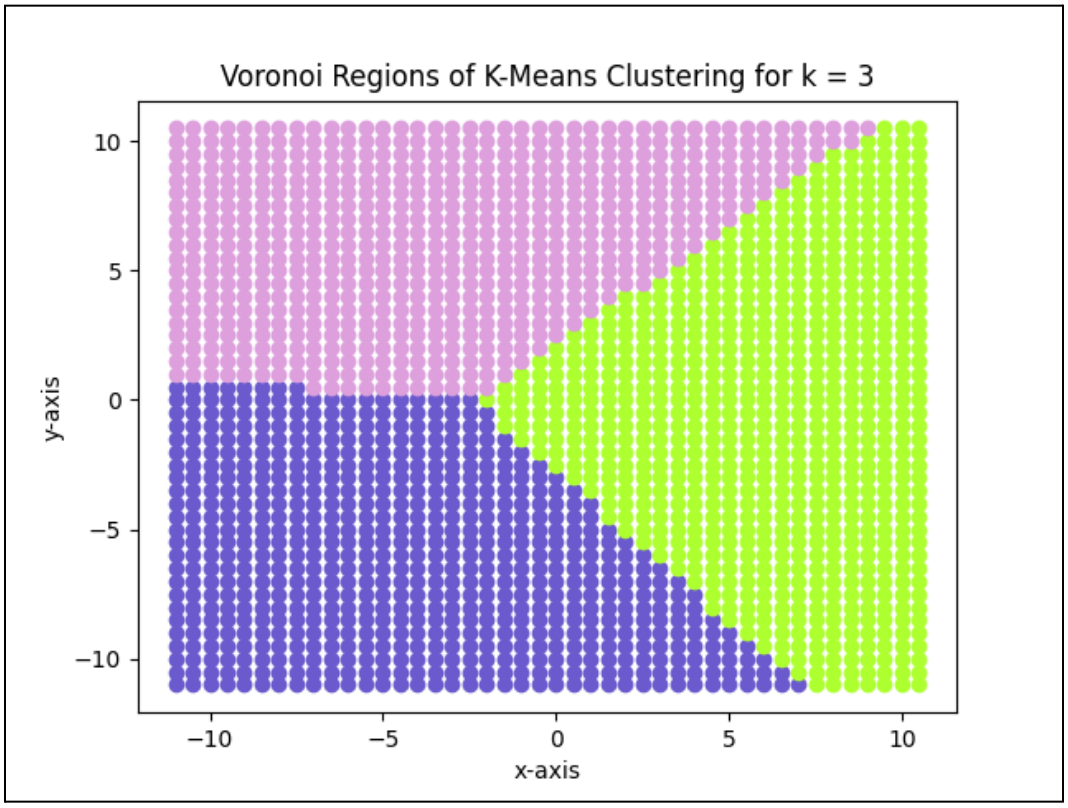


Figure : K-means Clustering for k=4

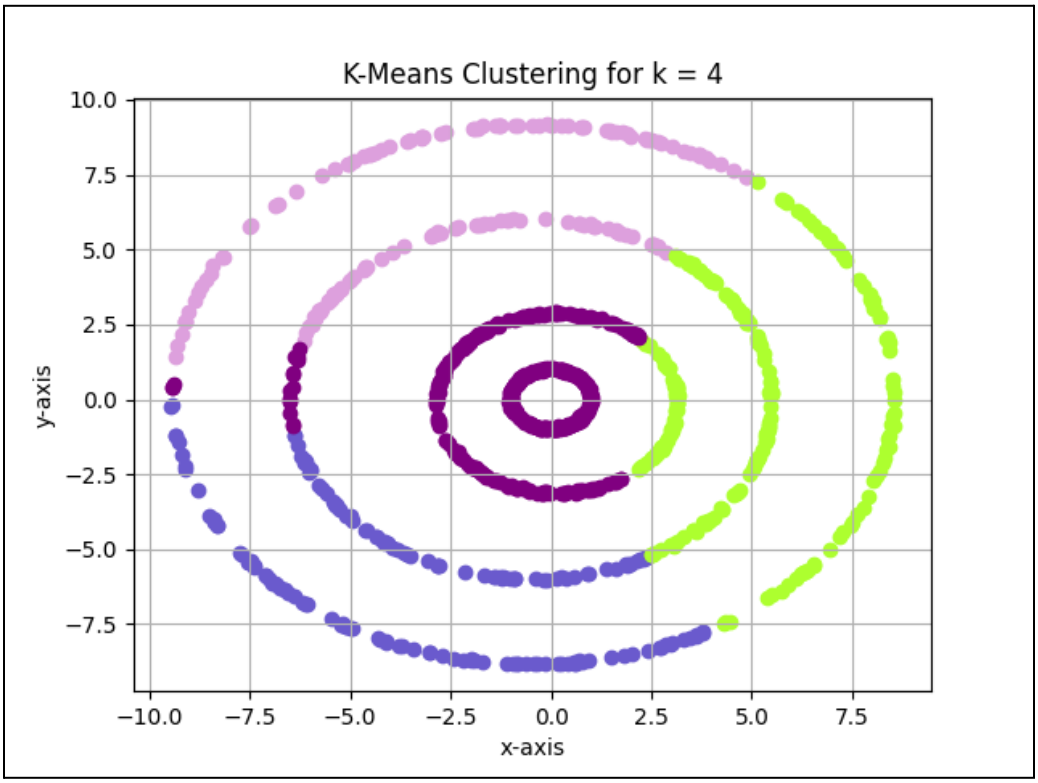


Figure : Error Function of K-means Clustering for k=4

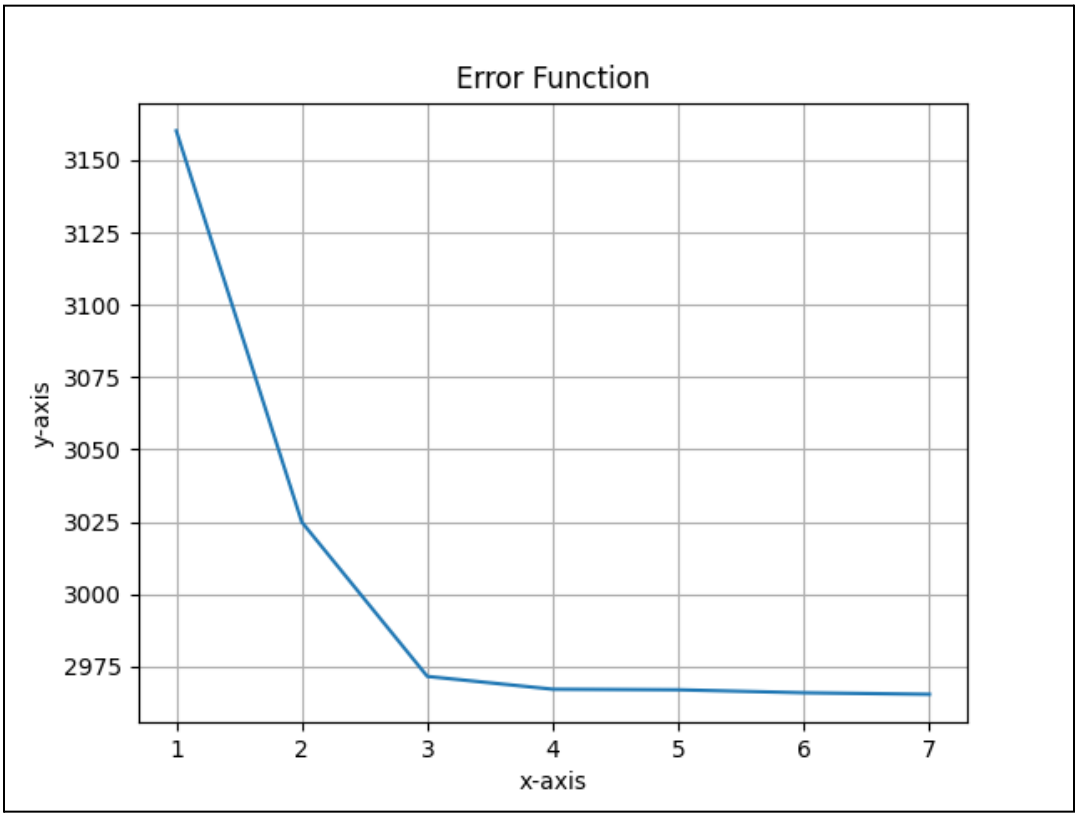


Figure : Voronoi Regions of K-means Clustering for k=4

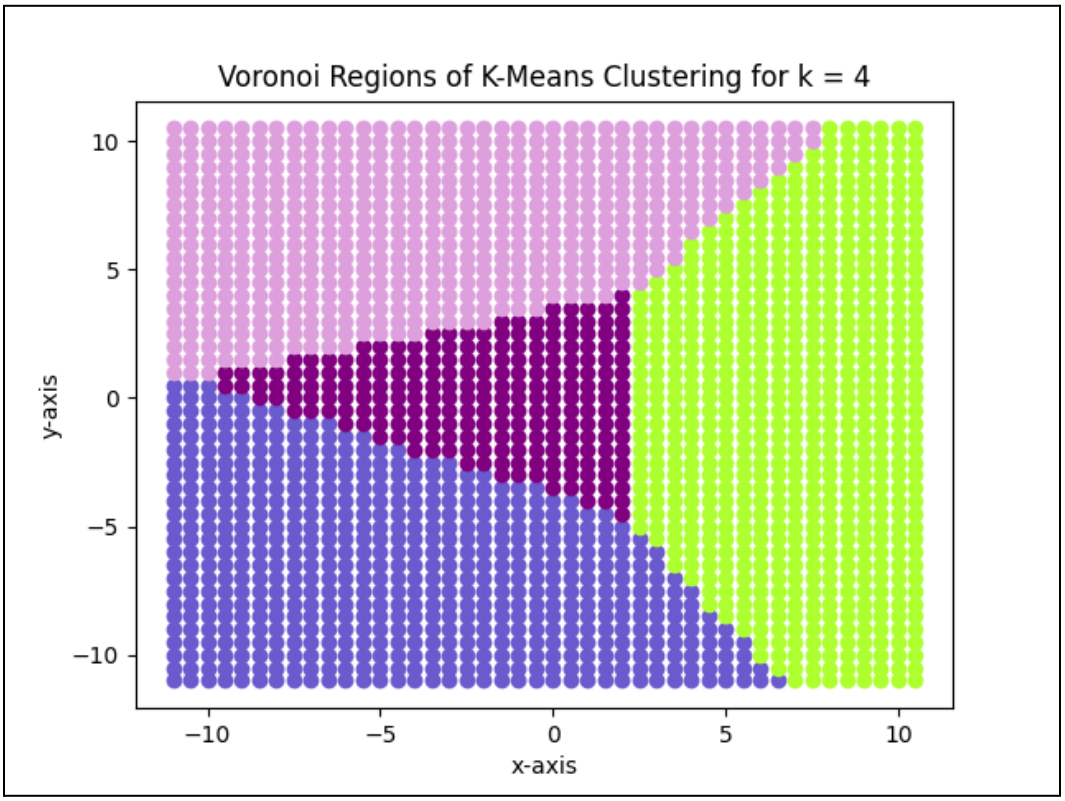


Figure : K-means Clustering for k=5

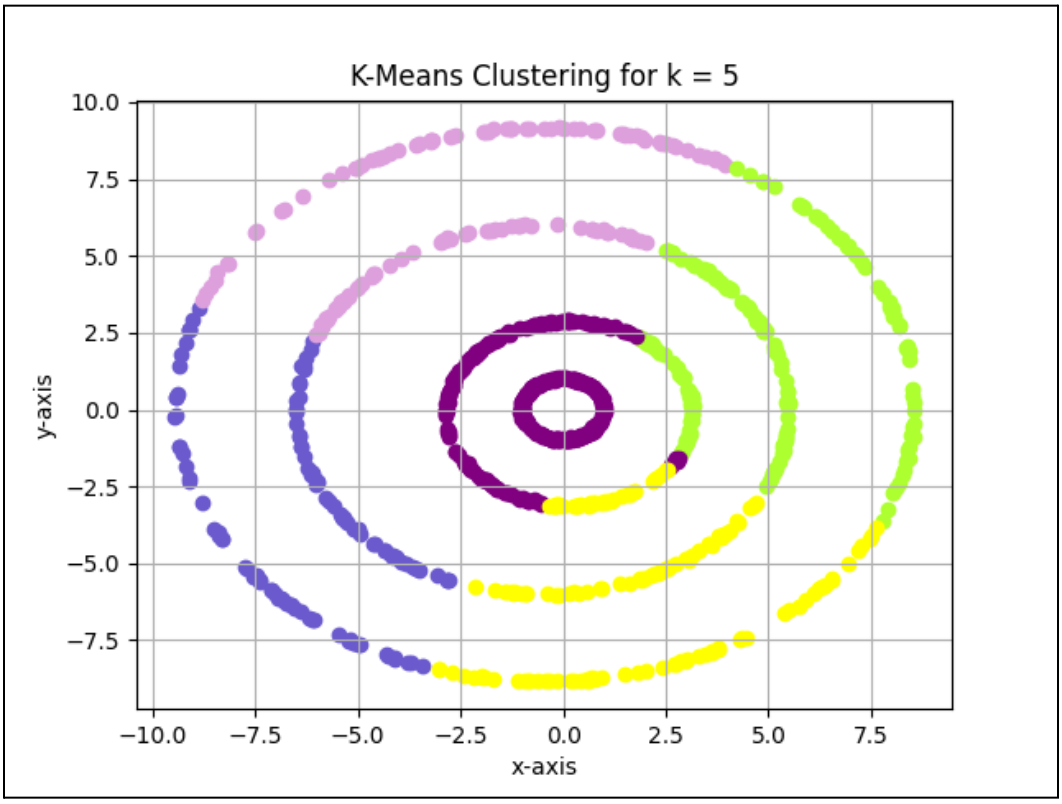


Figure : Error Function of K-means Clustering for k=5

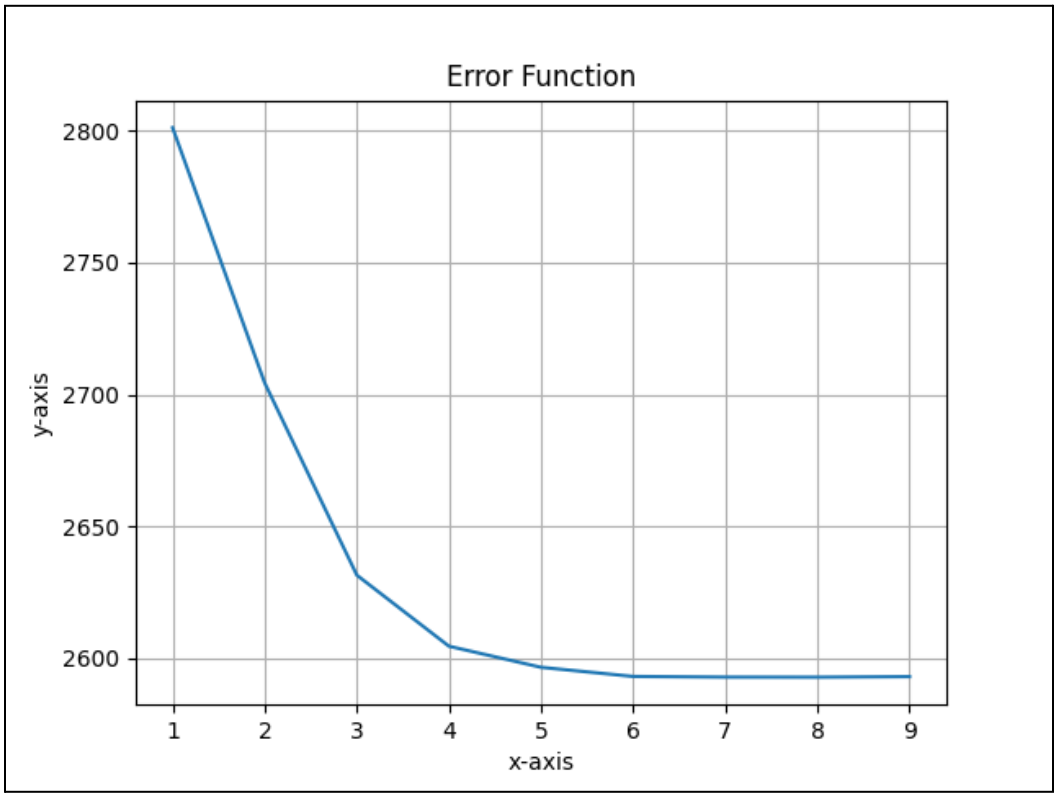
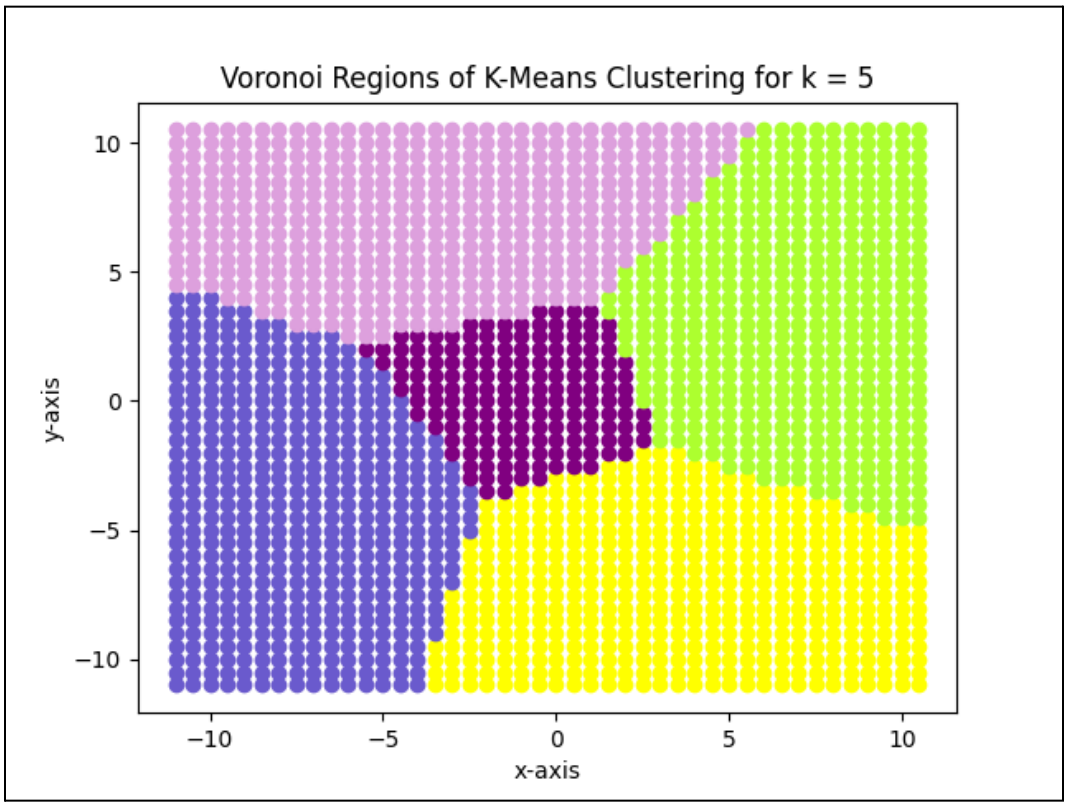


Figure : Voronoi Regions of K-means Clustering for k=5



Q2(iii) :
According to the data, we expect the clusters to be formed in such a way that each ring is a different cluster in spectral clustering. We see that this expectation is somewhat fulfilled in the cluster represented by polynomial kernel function used in spectral clustering with degree = 2.

Hence the best choice of kernel is polynomial function with degree 2.

Figure : Spectral Clustering using Polynomial Kernel Function with degree = 2

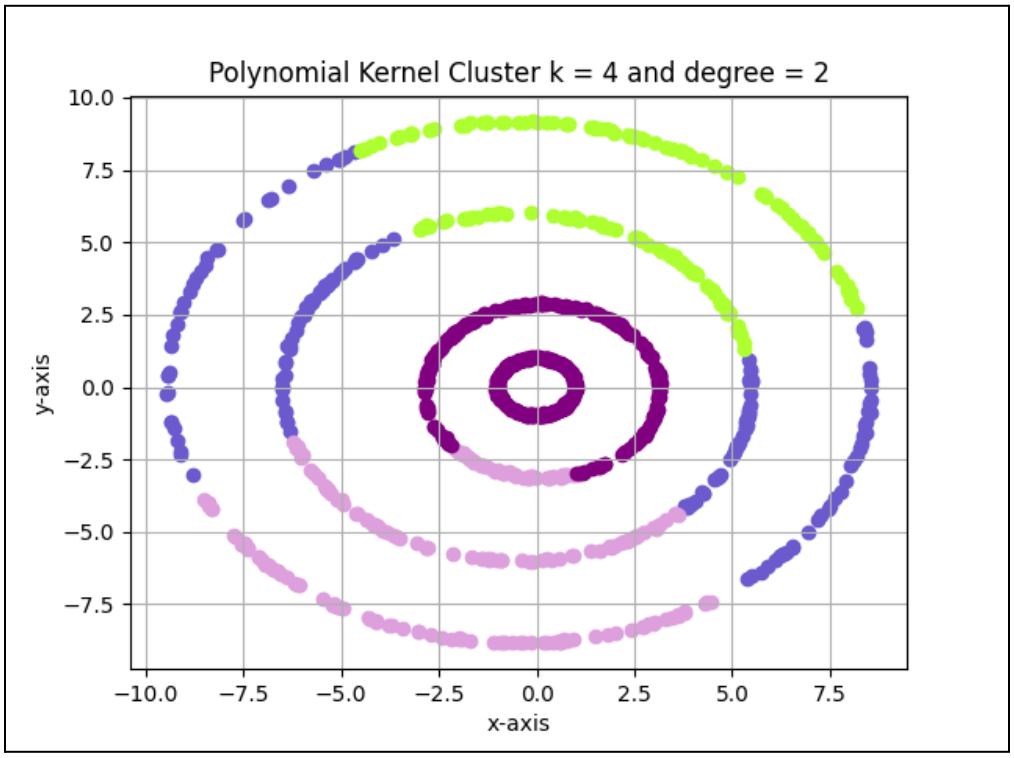


Figure : Error Function for Spectral Clustering

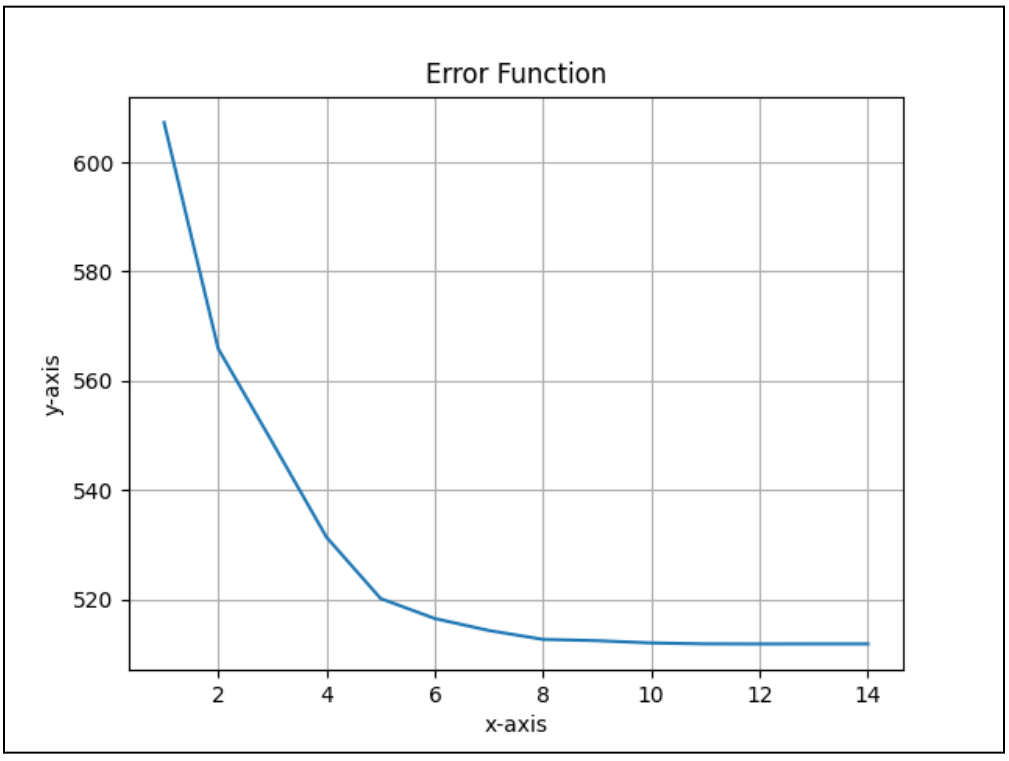


Figure : Spectral Clustering using Polynomial Kernel Function with degree = 3

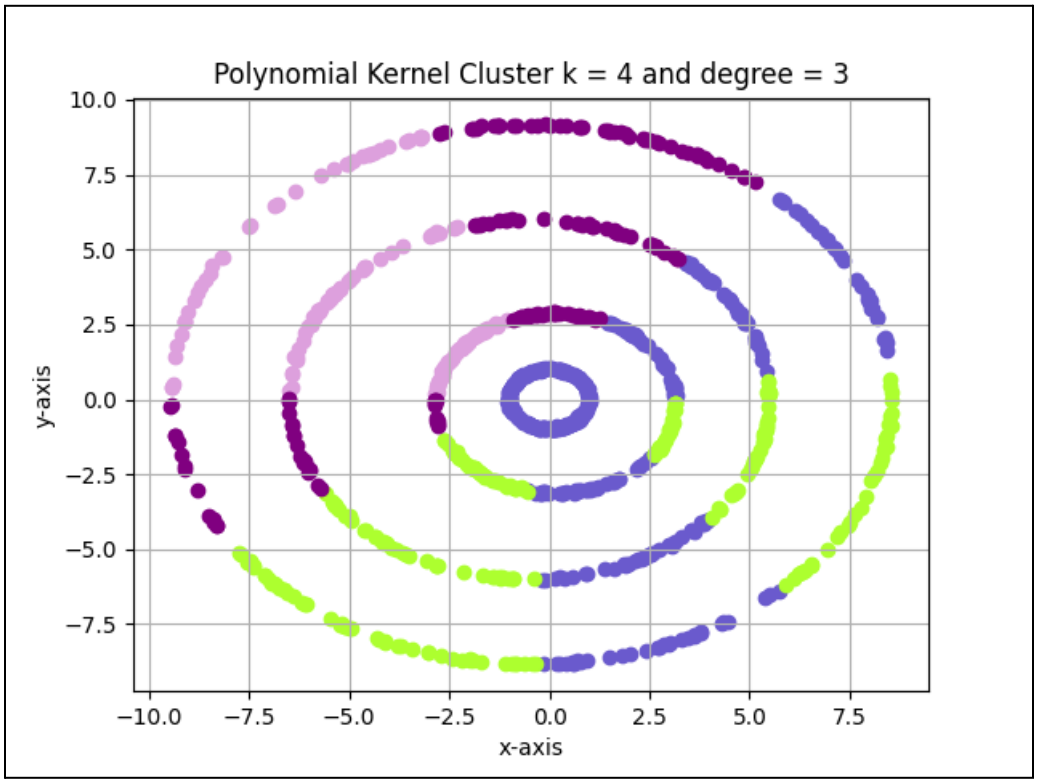


Figure : Error Function for Spectral Clustering

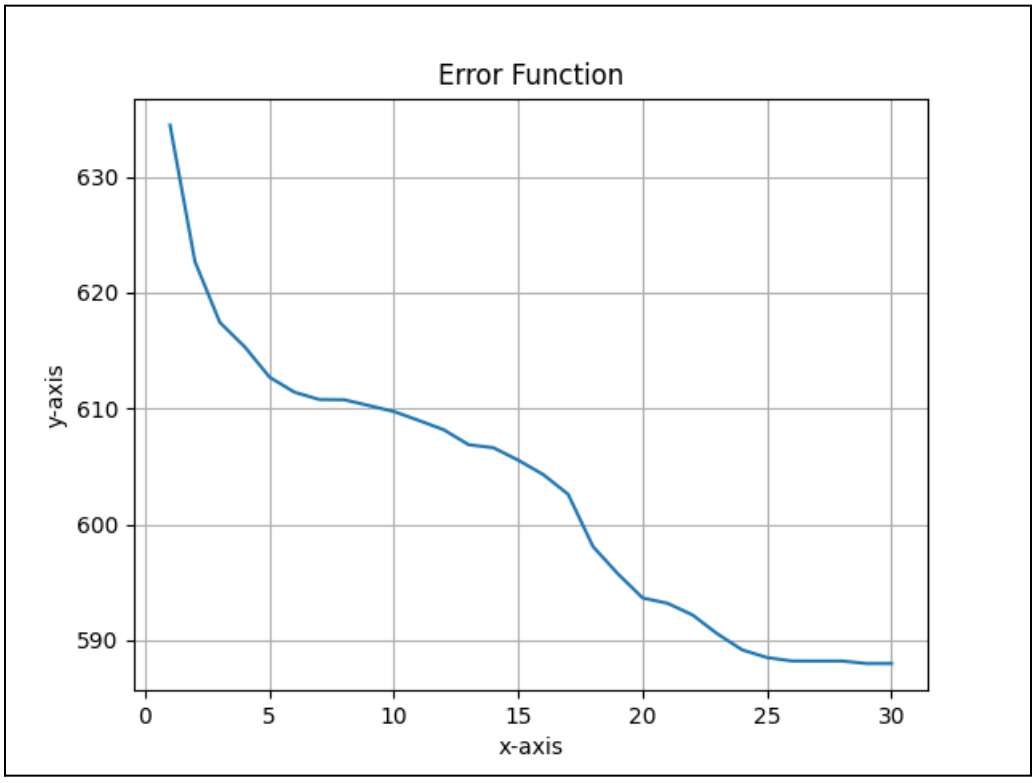


Figure : Spectral Clustering using Radial Basis Kernel Function with sigma = 0.1

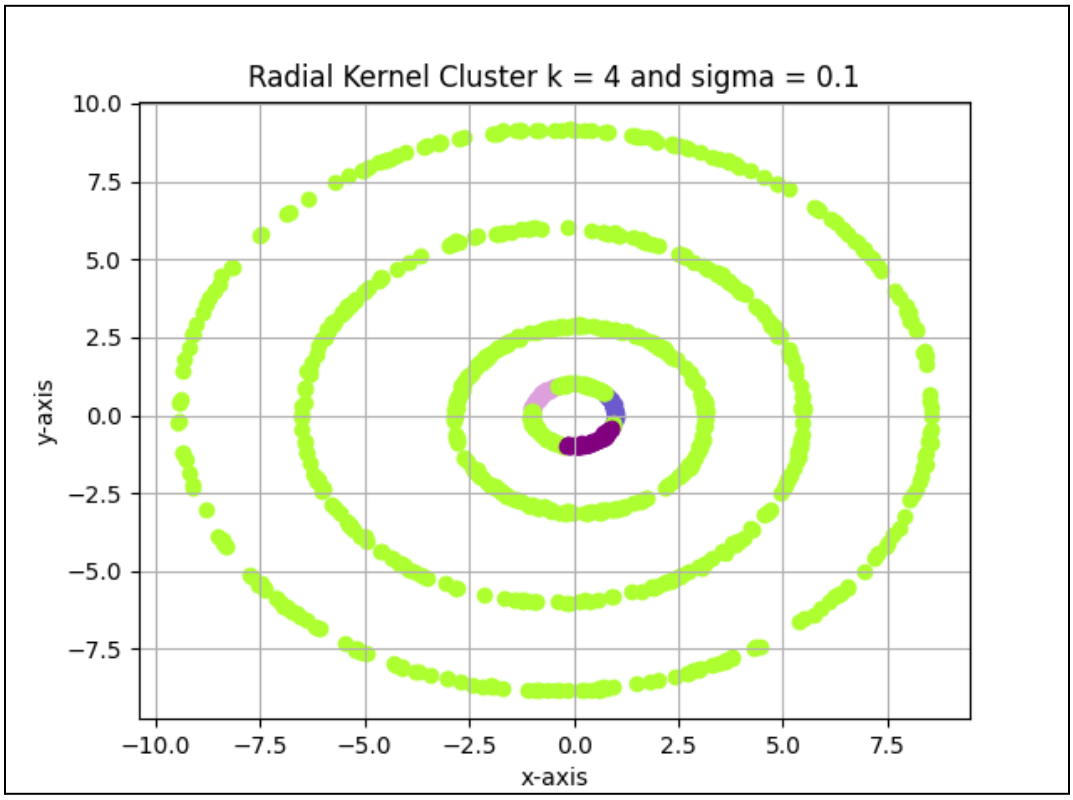


Figure : Error Function for Spectral Clustering

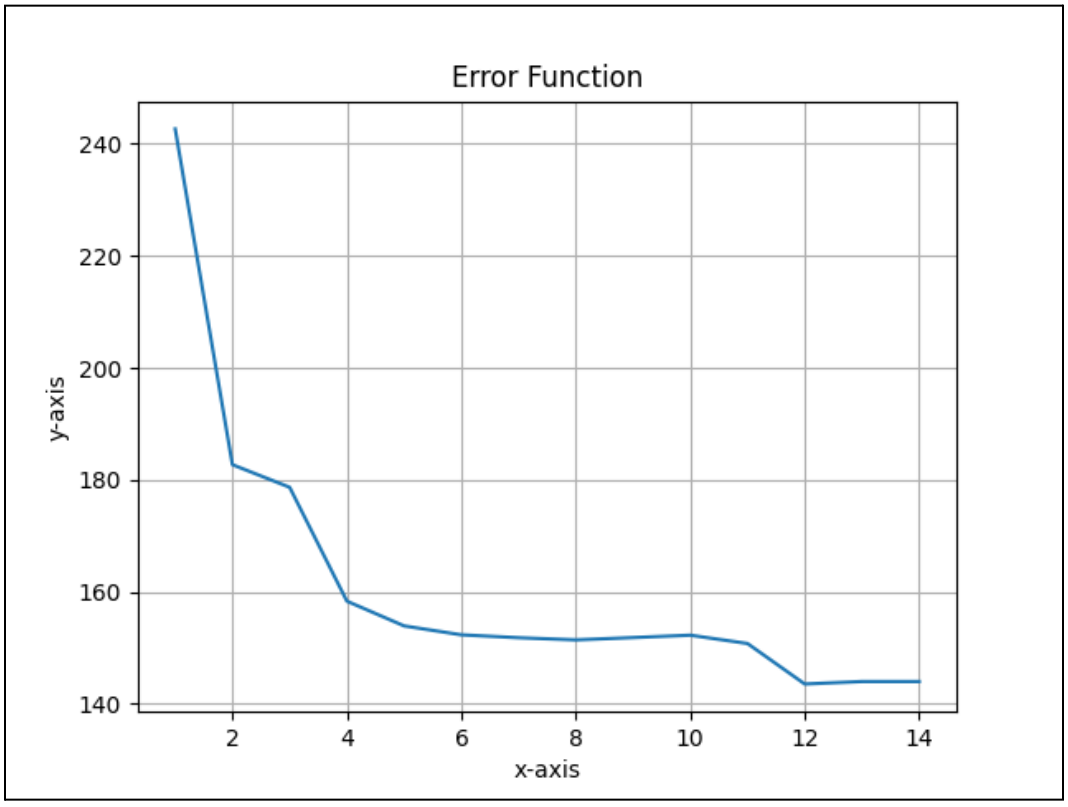


Figure : Spectral Clustering using Radial Basis Kernel Function with sigma = 0.2

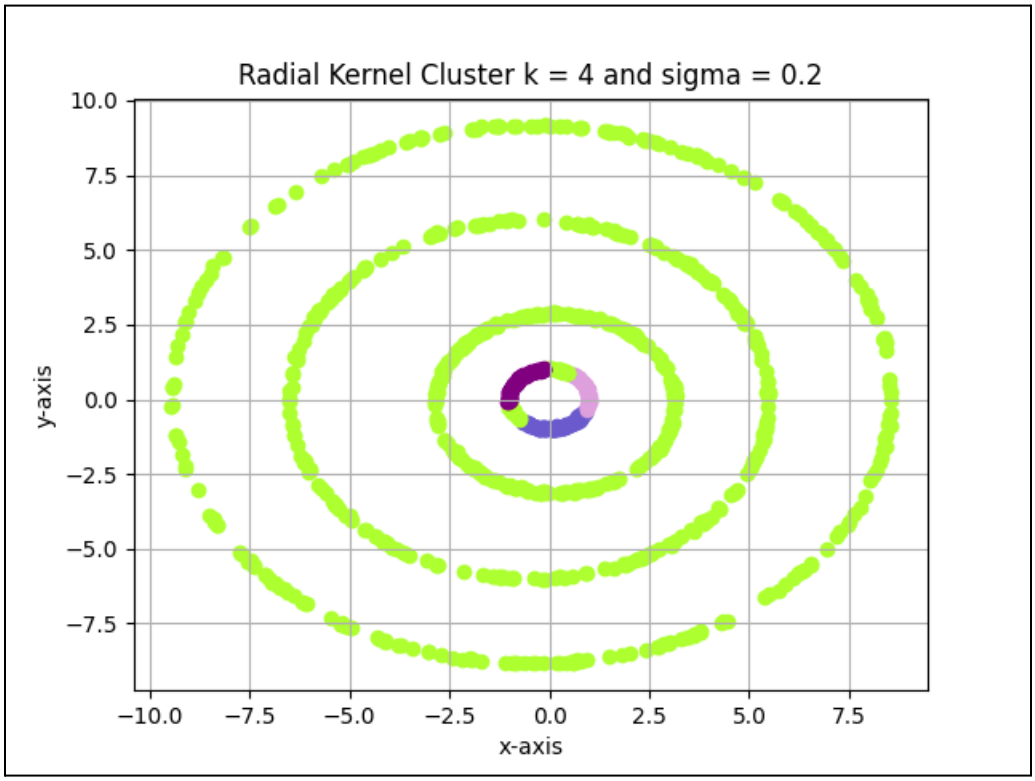


Figure : Error Function for Spectral Clustering

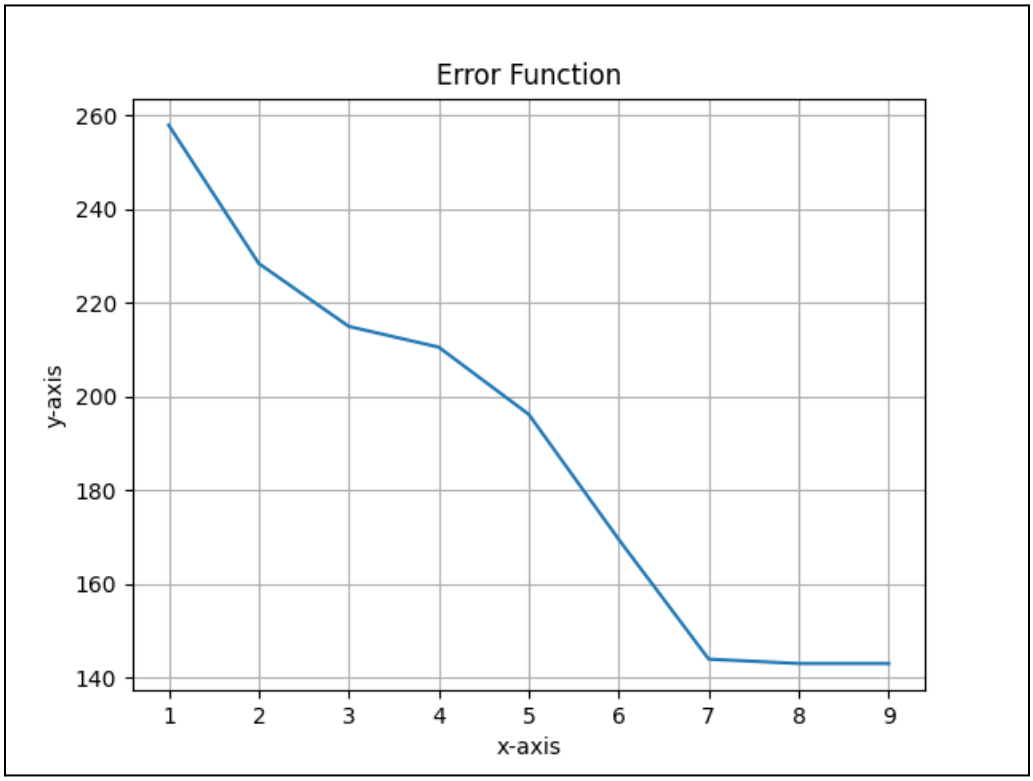


Figure : Spectral Clustering using Radial Basis Kernel Function with sigma = 0.3

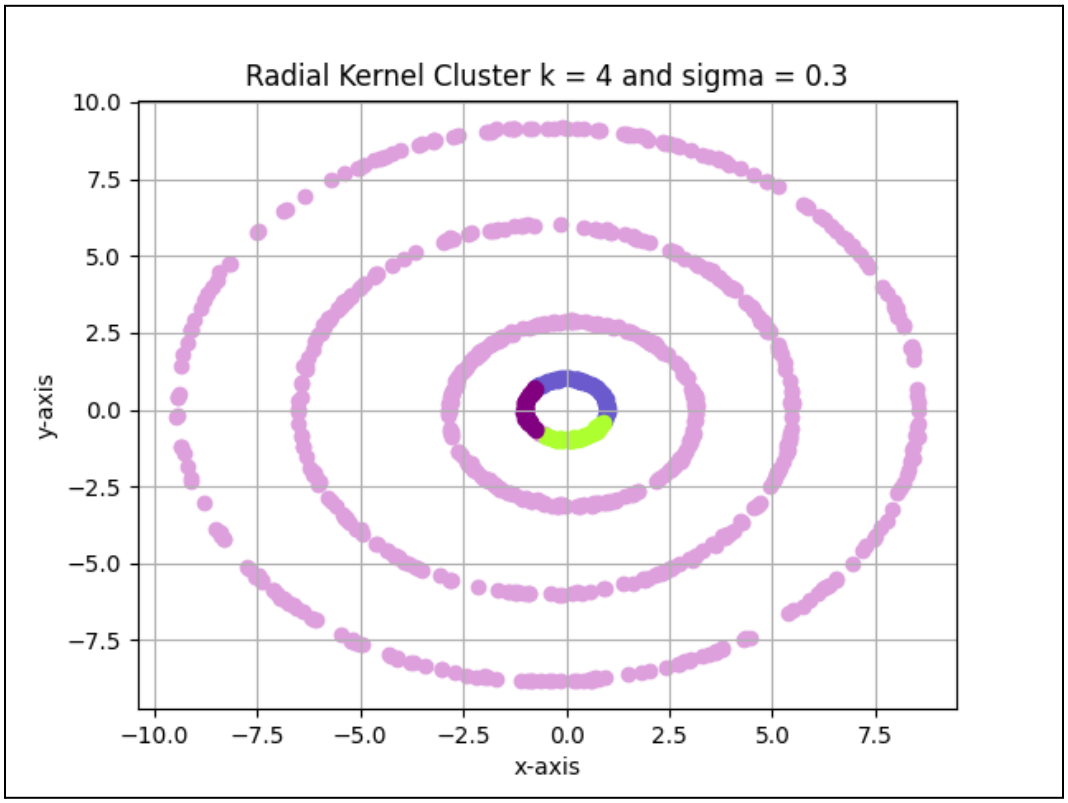


Figure : Error Function for Spectral Clustering

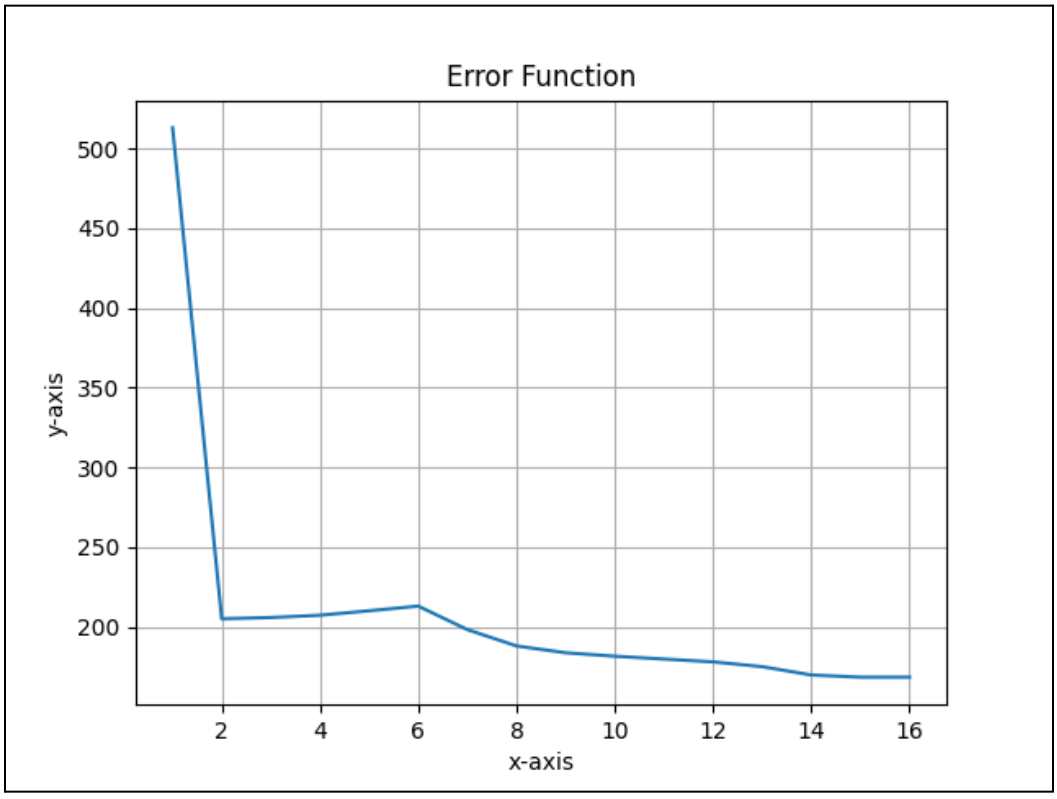


Figure : Spectral Clustering using Radial Basis Kernel Function with sigma = 0.4

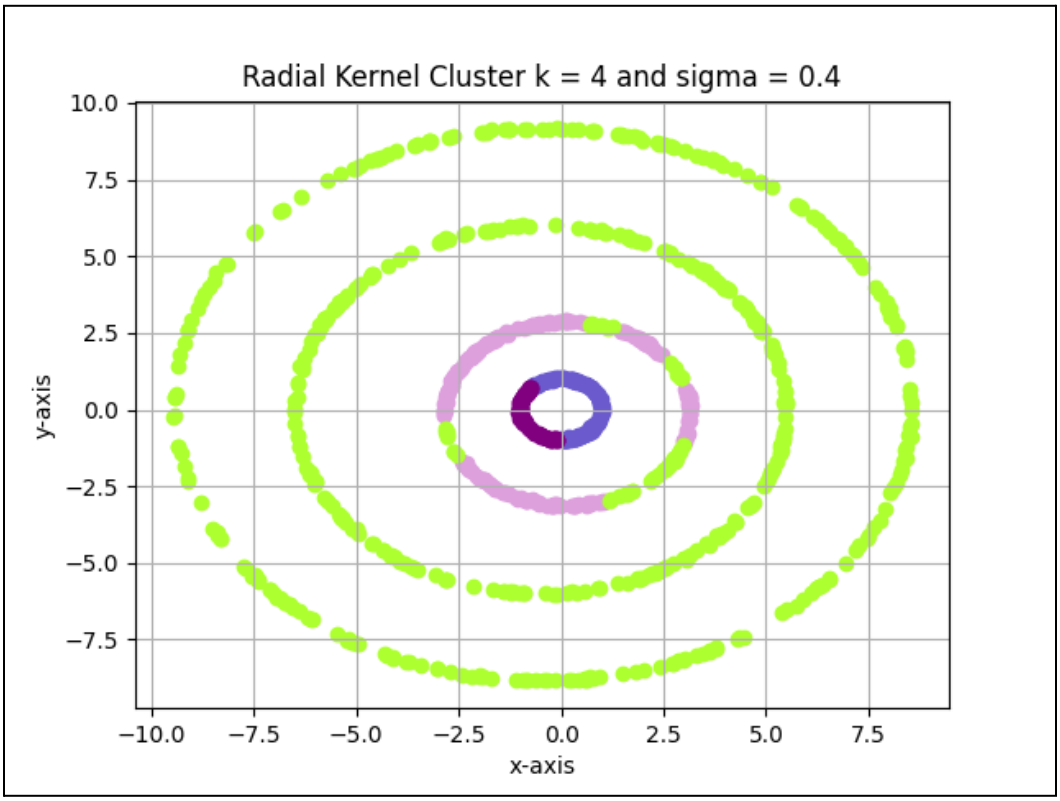


Figure : Error Function for Spectral Clustering

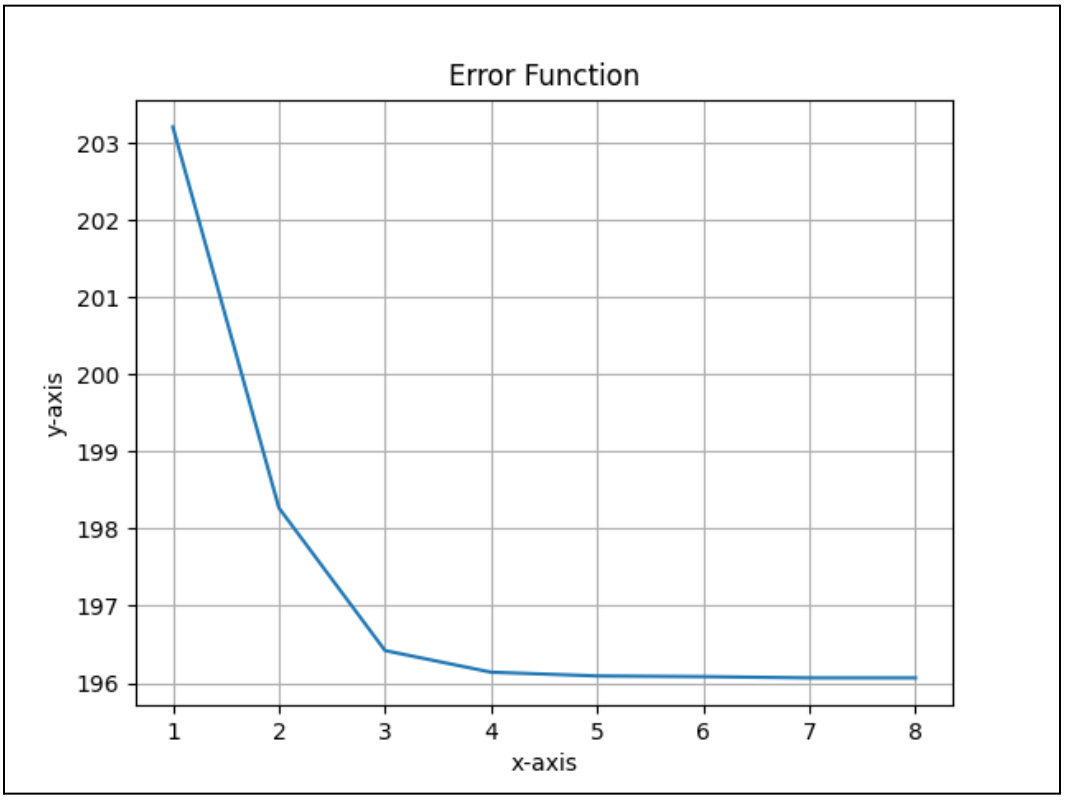


Figure : Spectral Clustering using Radial Basis Kernel Function with sigma = 0.5

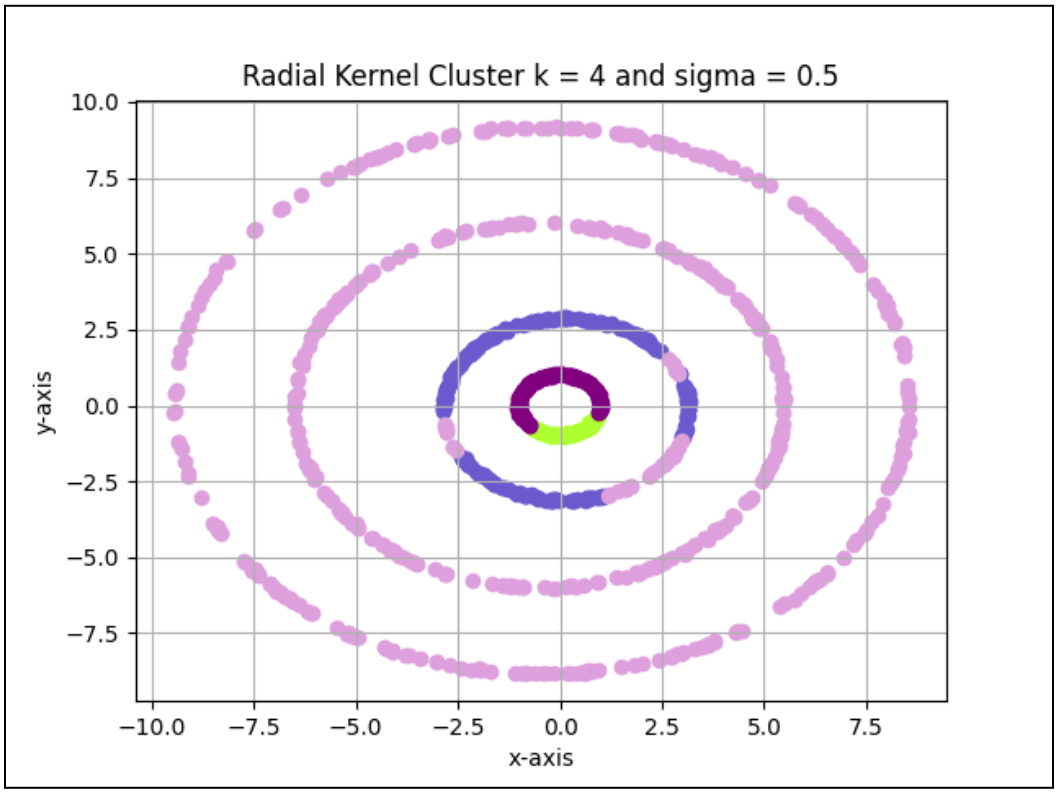


Figure : Error Function for Spectral Clustering

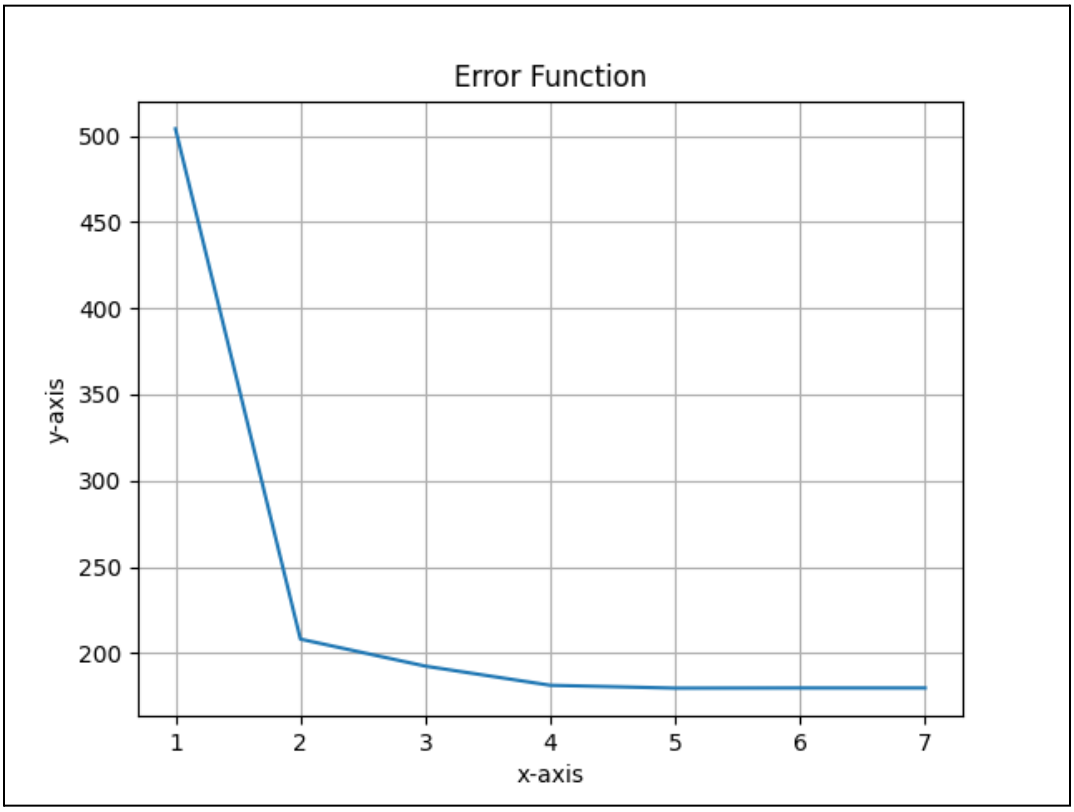


Figure : Spectral Clustering using Radial Basis Kernel Function with sigma = 0.6

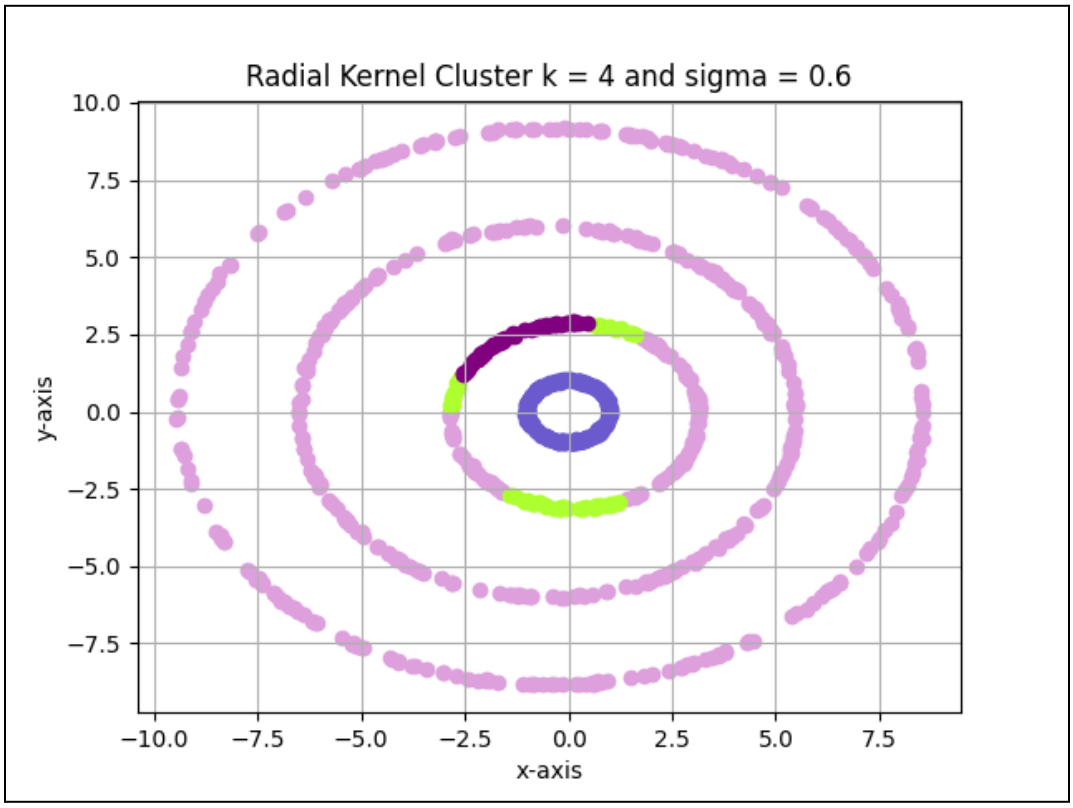


Figure : Error Function for Spectral Clustering

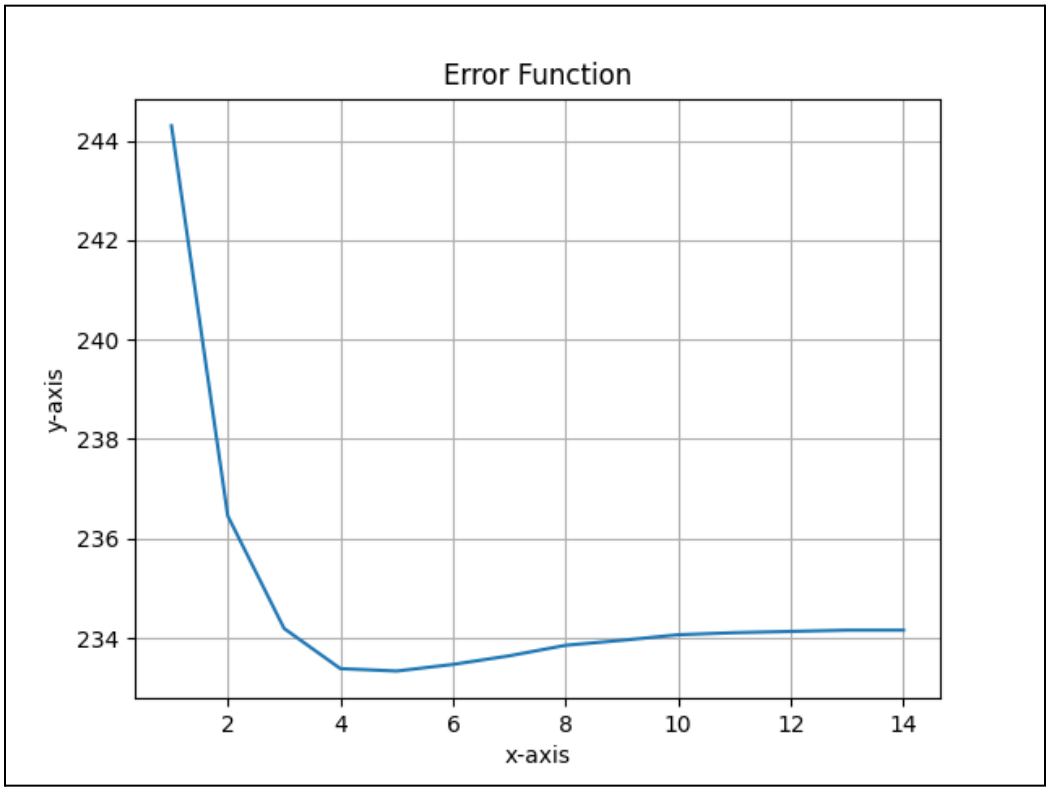


Figure : Spectral Clustering using Radial Basis Kernel Function with sigma = 0.7

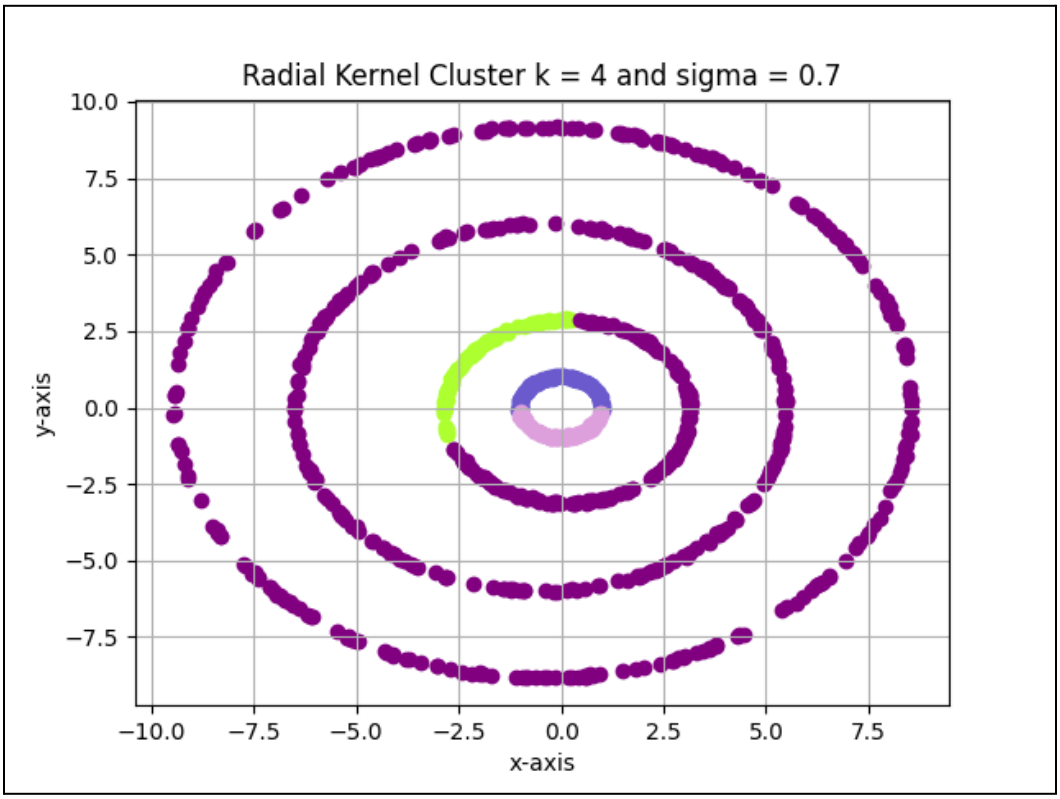


Figure : Error Function for Spectral Clustering

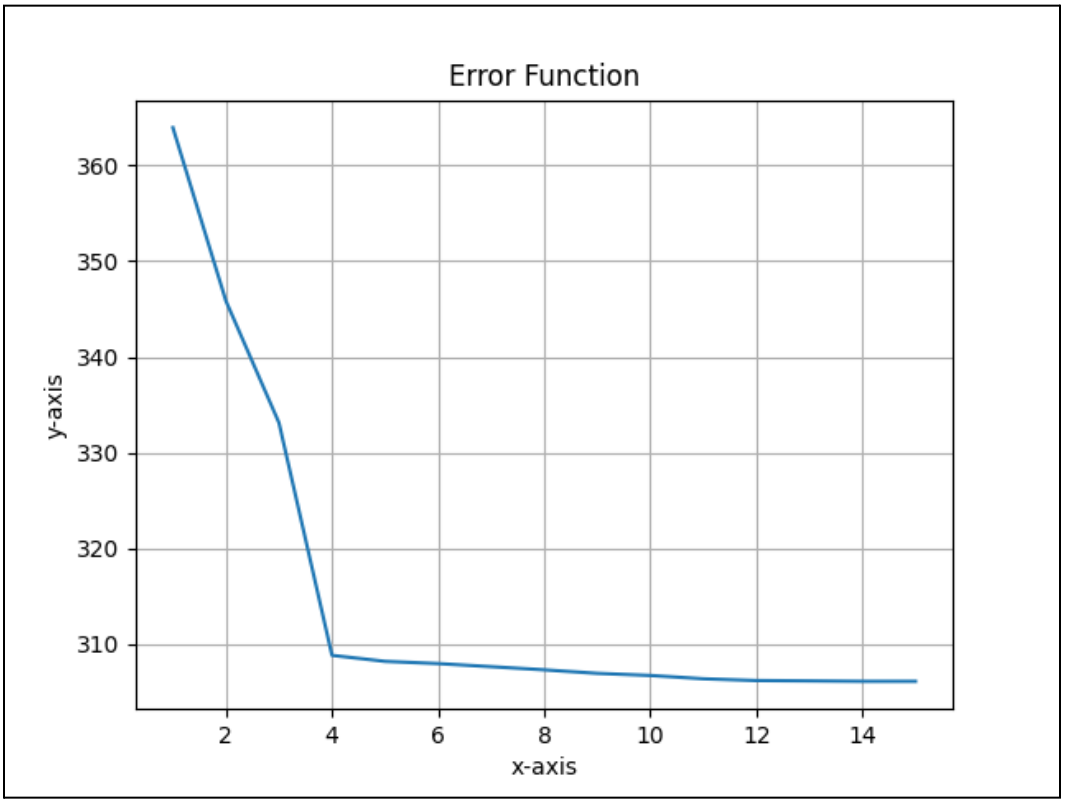


Figure : Spectral Clustering using Radial Basis Kernel Function with sigma = 0.8

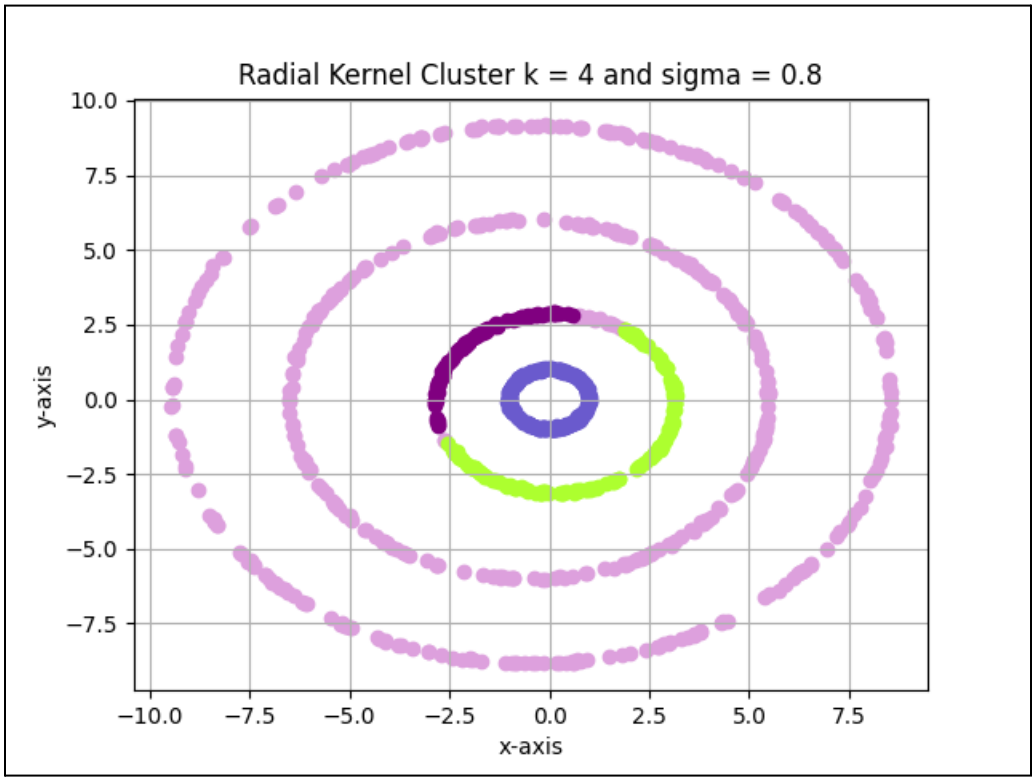


Figure : Error Function for Spectral Clustering

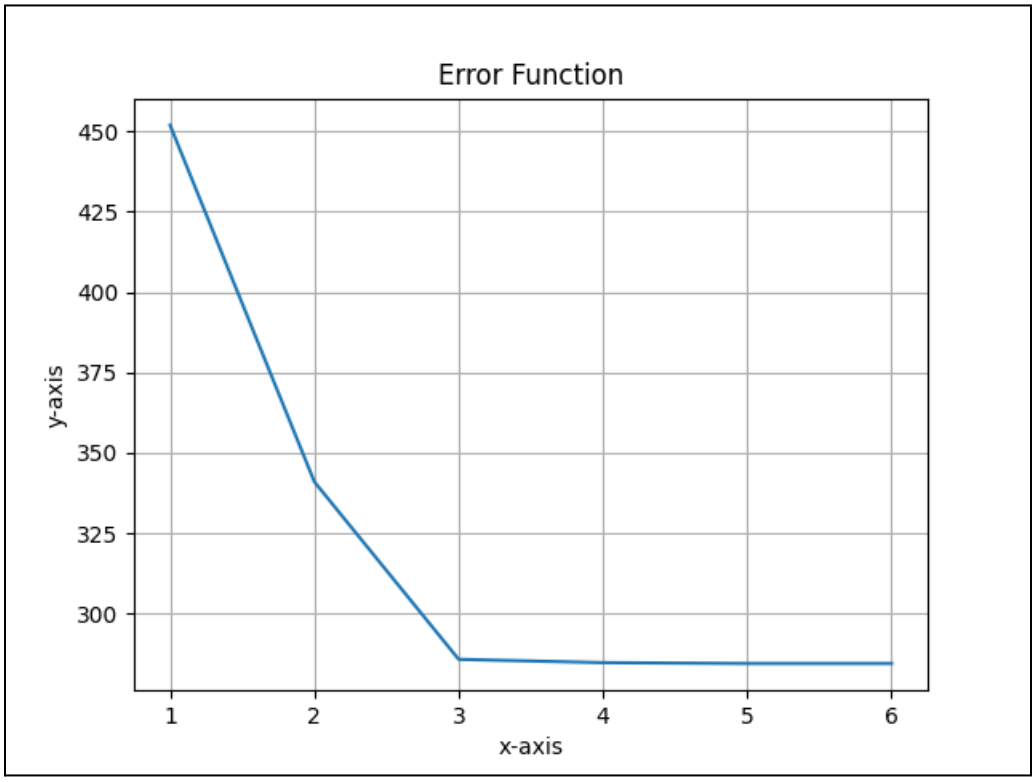


Figure : Spectral Clustering using Radial Basis Kernel Function with sigma = 0.9

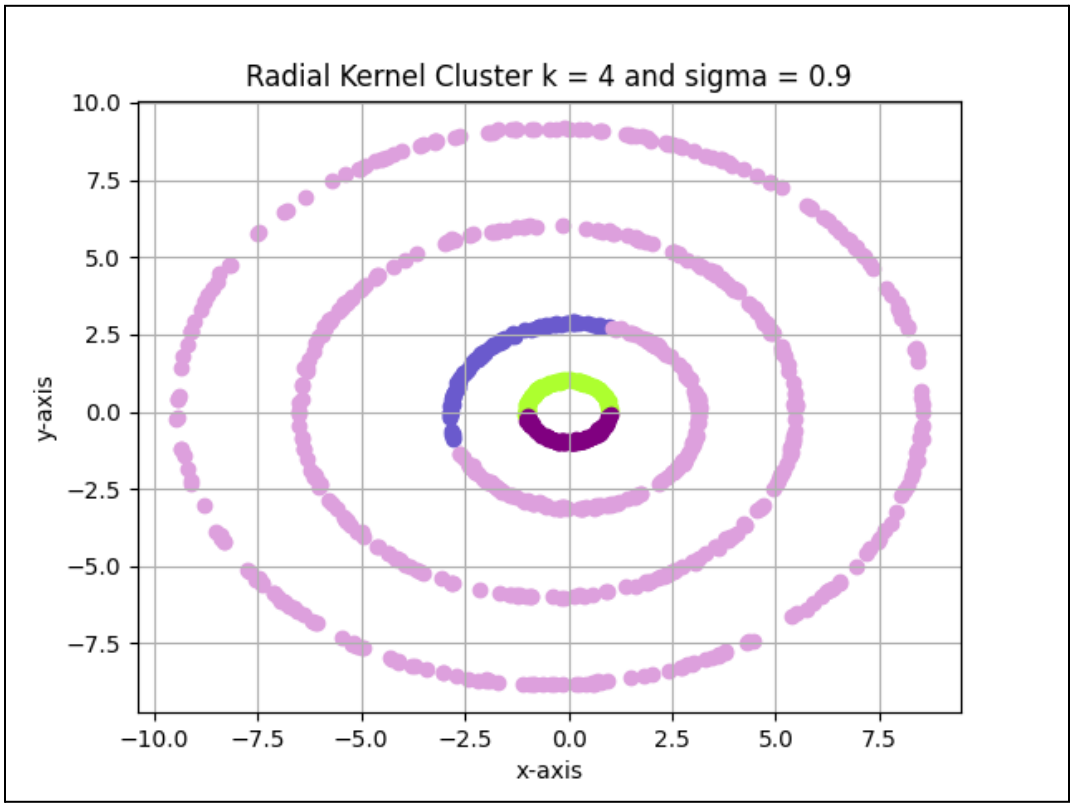


Figure : Error Function for Spectral Clustering

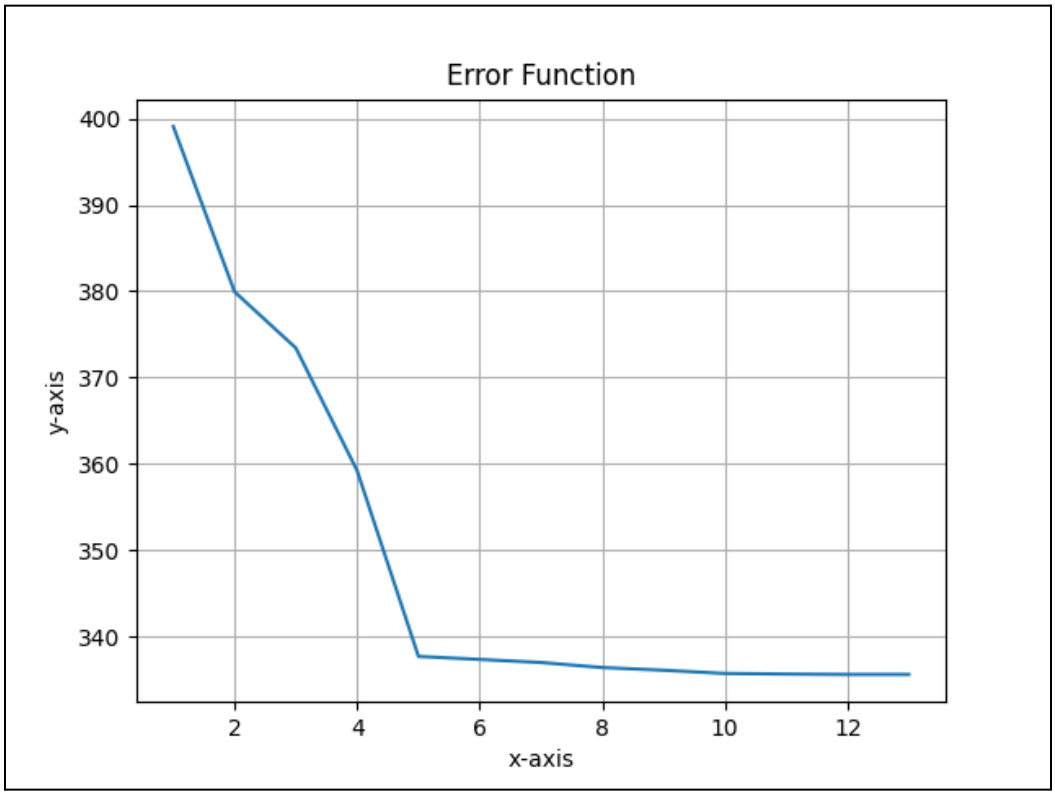


Figure : Spectral Clustering using Radial Basis Kernel Function with sigma = 1

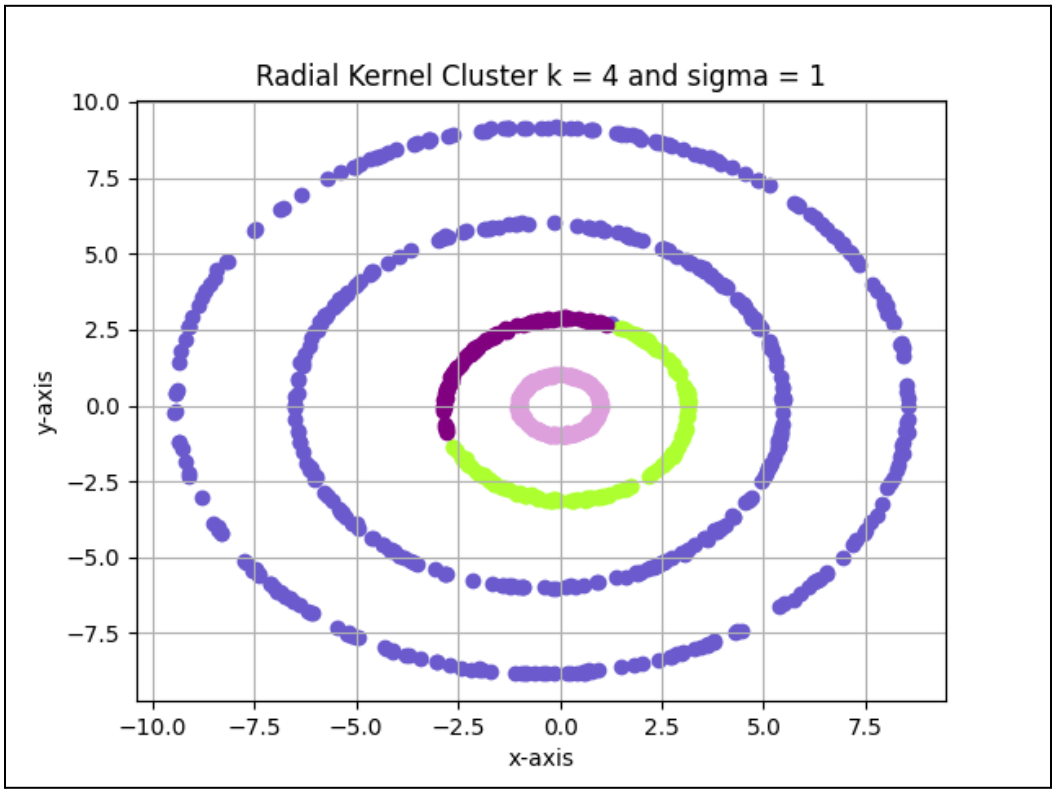
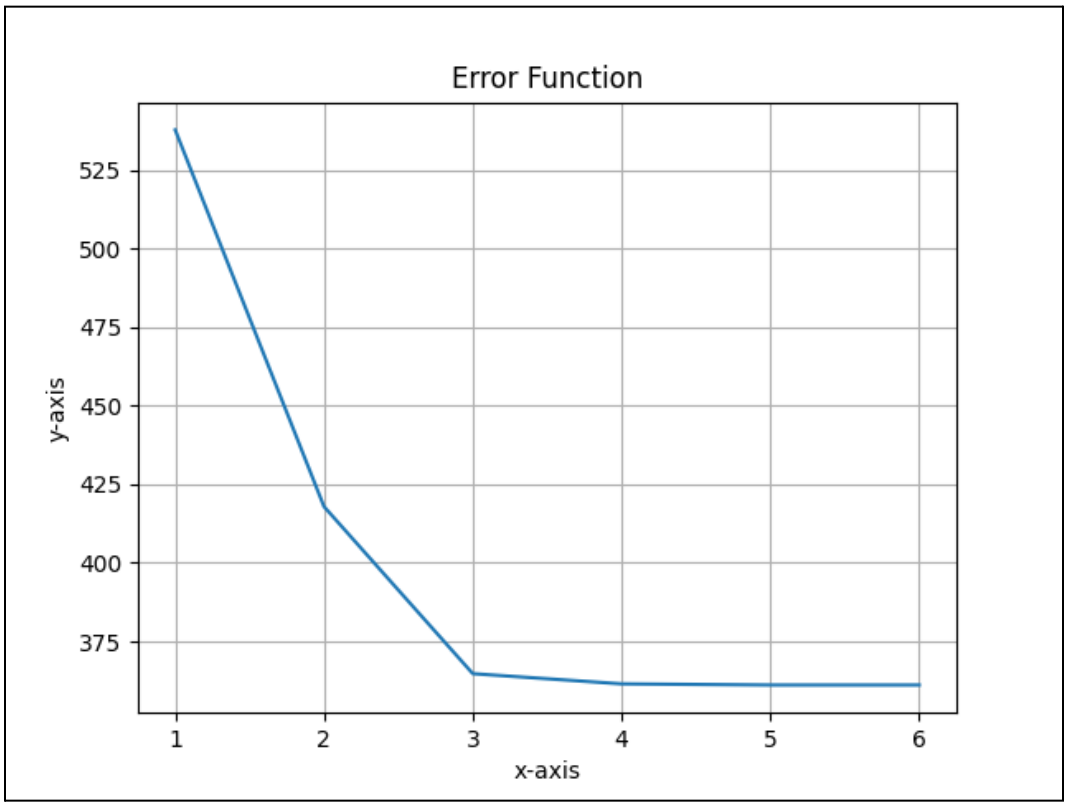


Figure : Error Function for Spectral Clustering



Q2(iv) :

This mapping performs better than k-means clustering with random initialization but not better than k means++ algorithm.

The reason being that k-means clustering uses random initialization. However, for this type of mapping we have a matrix of top k eigenvectors H[n][k] where n = number of data points, k = number of clusters which is normalized. To choose the cluster for each data point we find the maximum value in a row. Hence performing better than k-means clustering.

Figure : Clustering using Polynomial Kernel Function with degree = 2 (argmax)

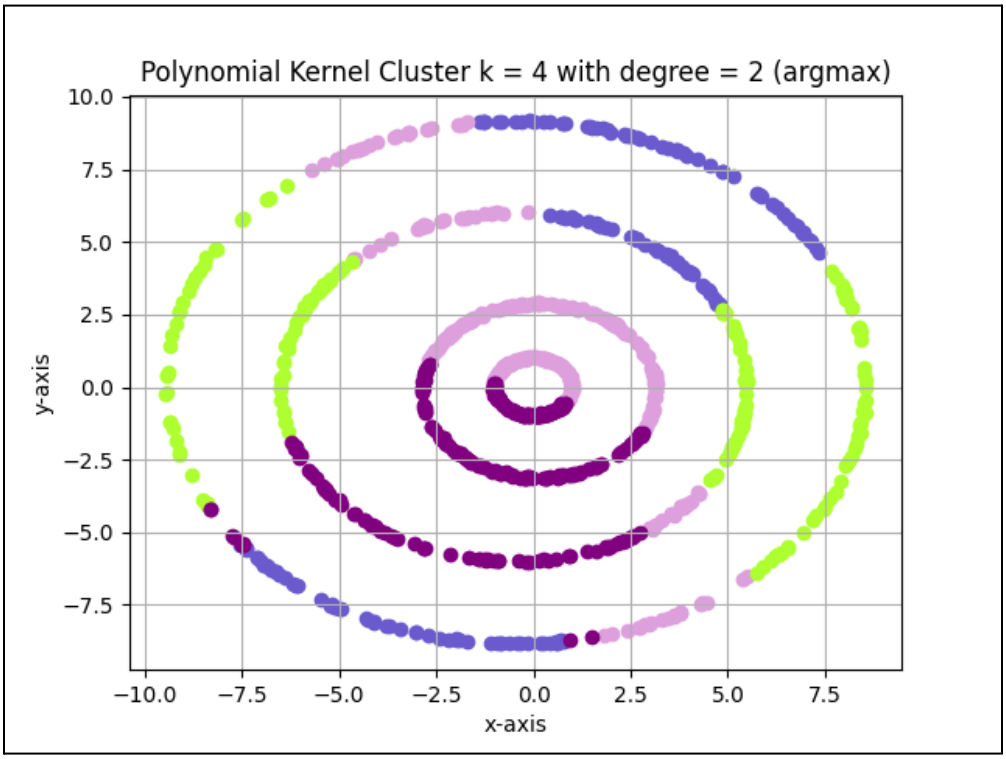


Figure : Clustering using Polynomial Kernel Function with degree = 3 (argmax)

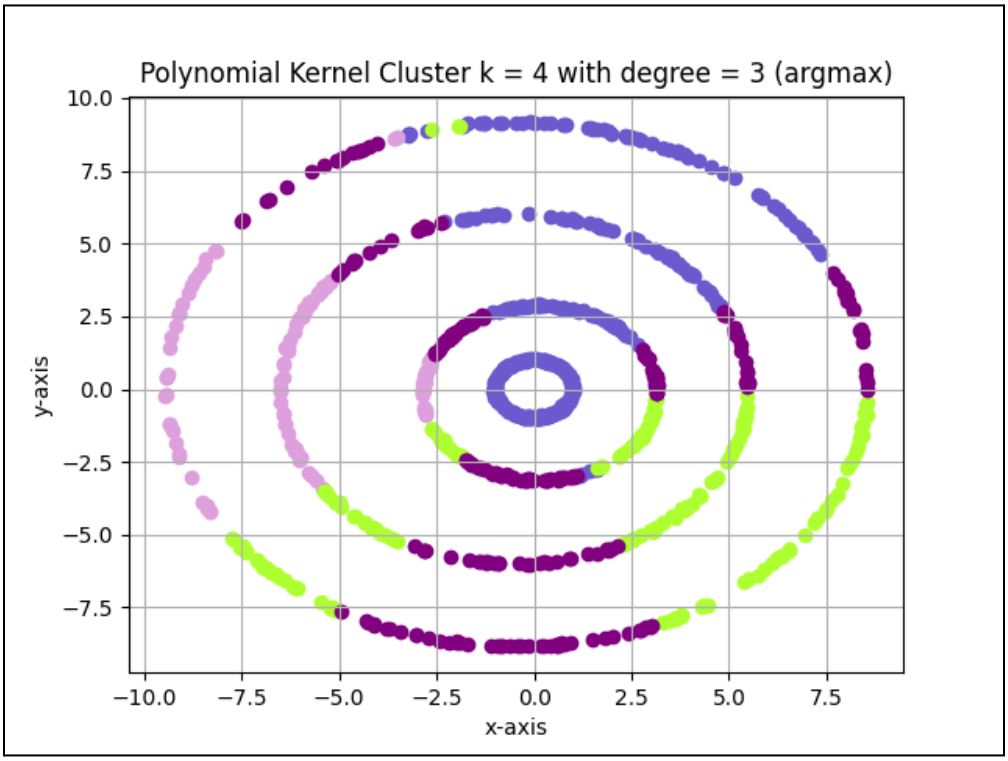


Figure : Clustering using Radial Basis Kernel Function with sigma = 0.1 (argmax)

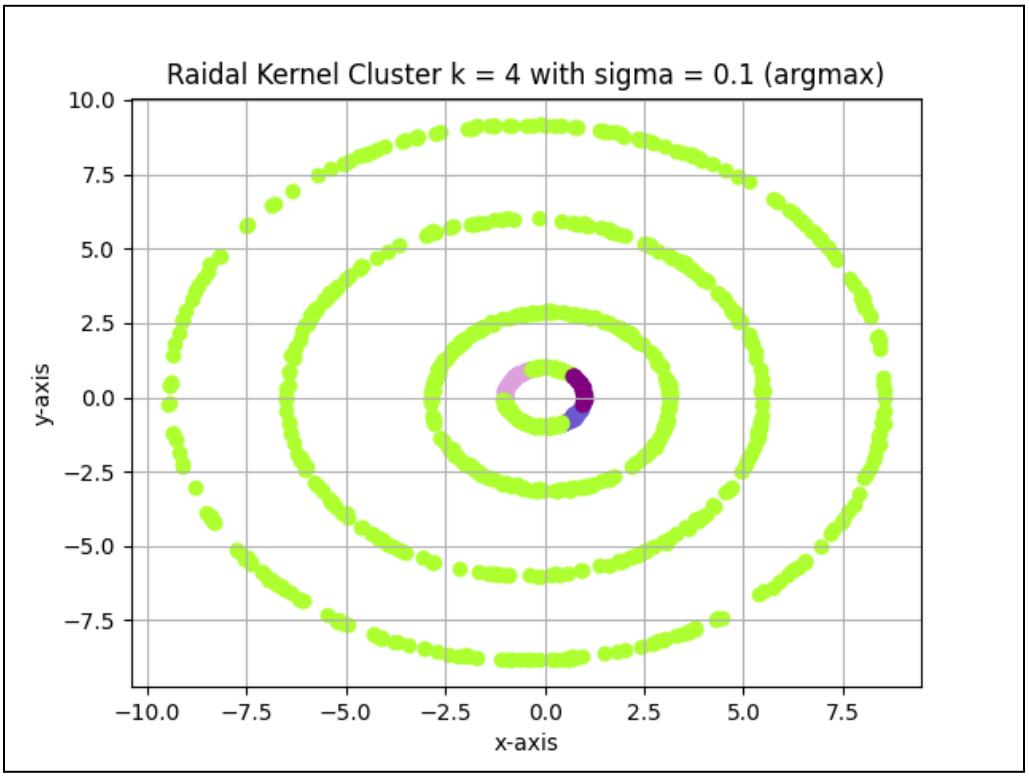


Figure : Clustering using Radial Basis Kernel Function with sigma = 0.2 (argmax)

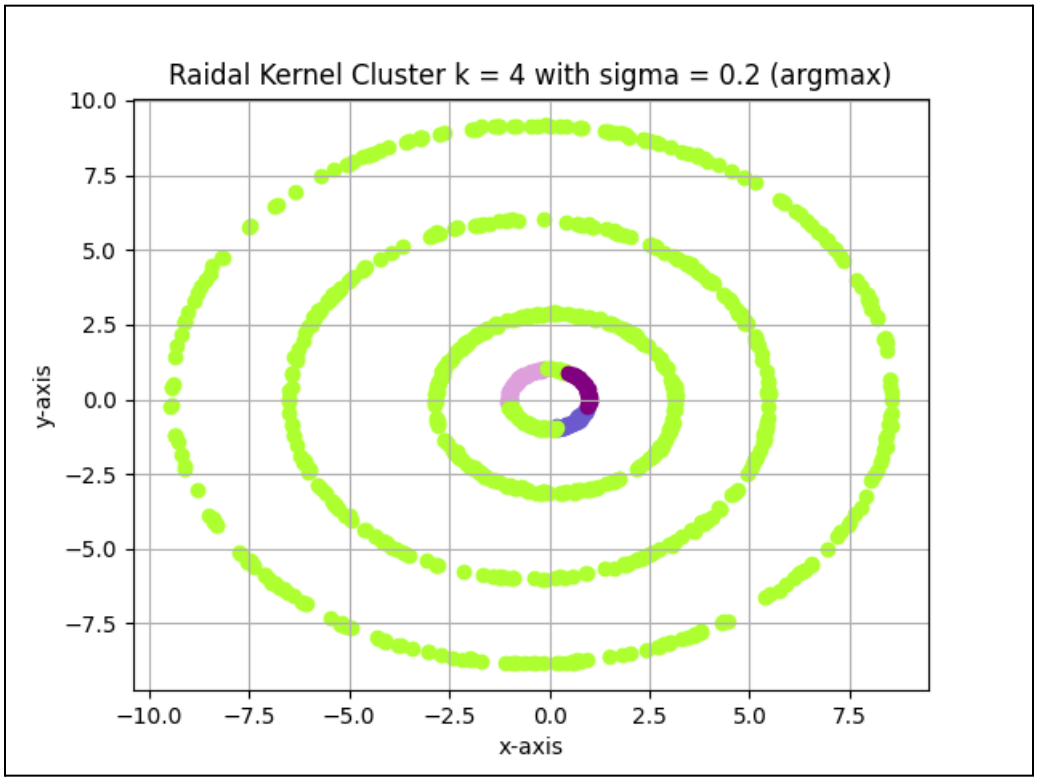


Figure : Clustering using Radial Basis Kernel Function with sigma = 0.3 (argmax)

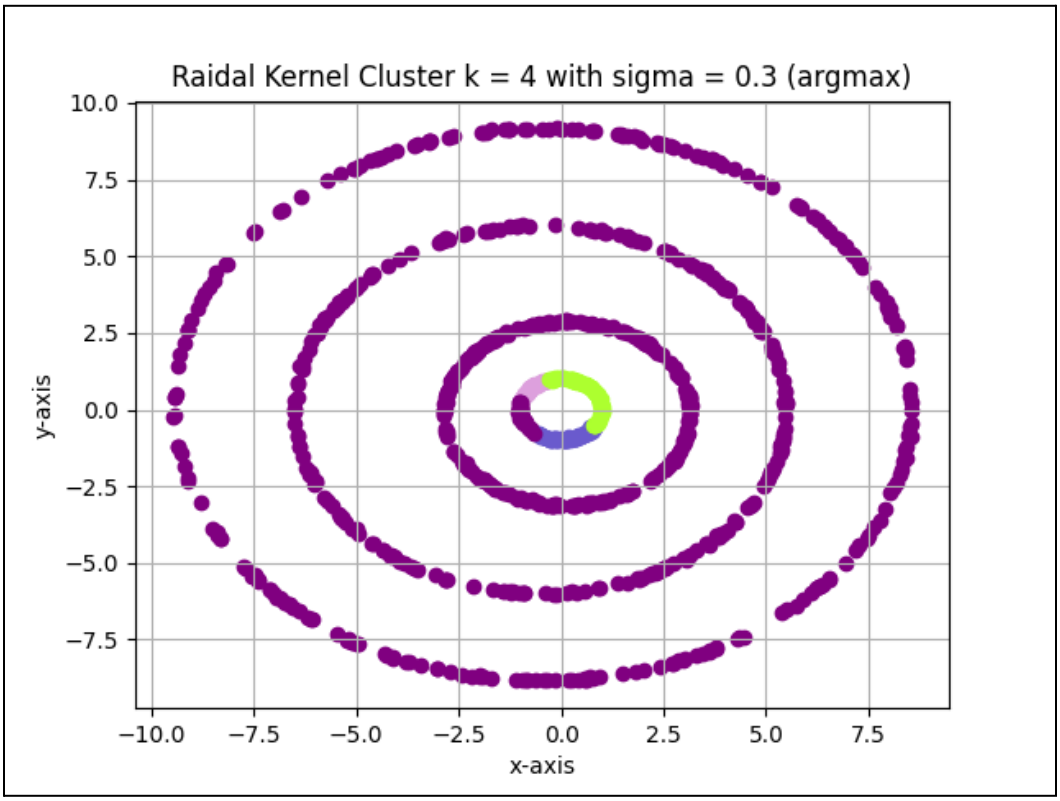


Figure : Clustering using Radial Basis Kernel Function with sigma = 0.4(argmax)

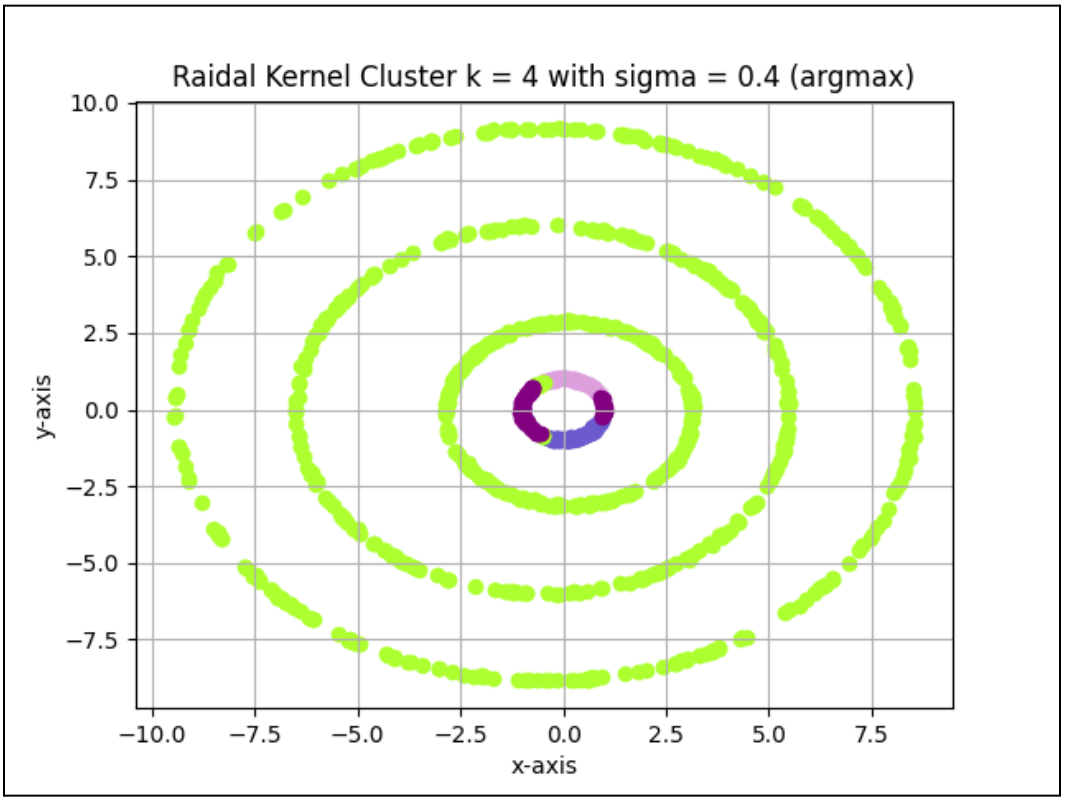


Figure : Clustering using Radial Basis Kernel Function with sigma = 0.5(argmax)

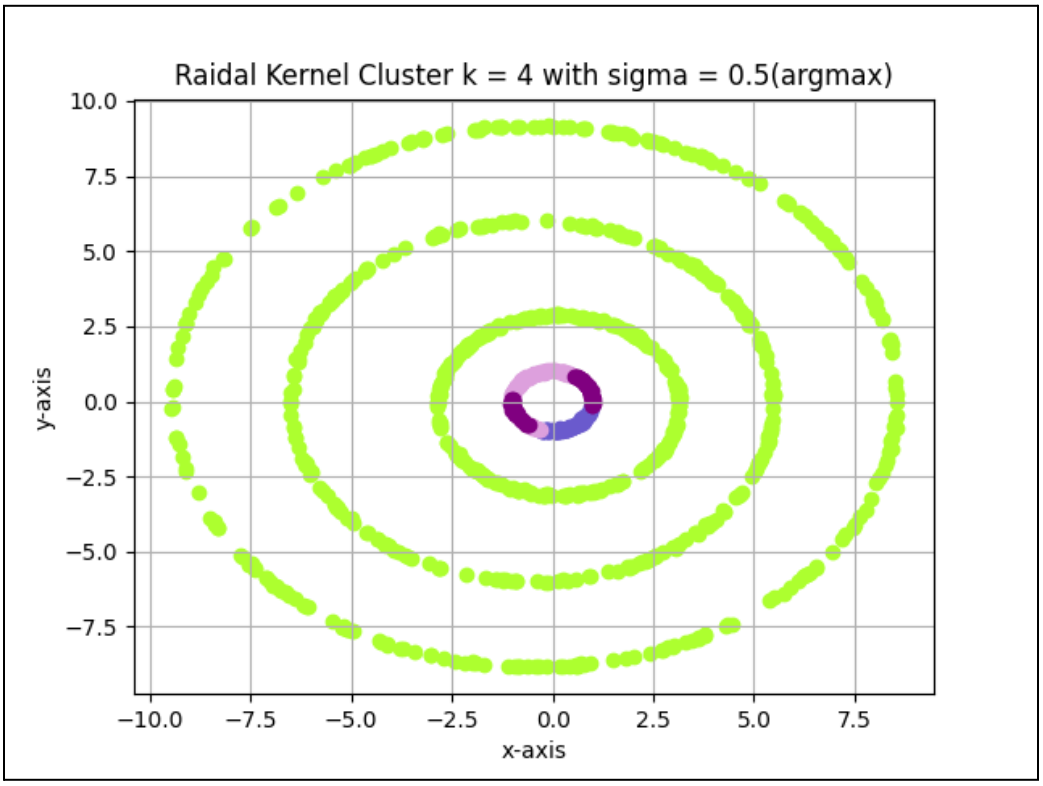


Figure : Clustering using Radial Basis Kernel Function with sigma = 0.6(argmax)

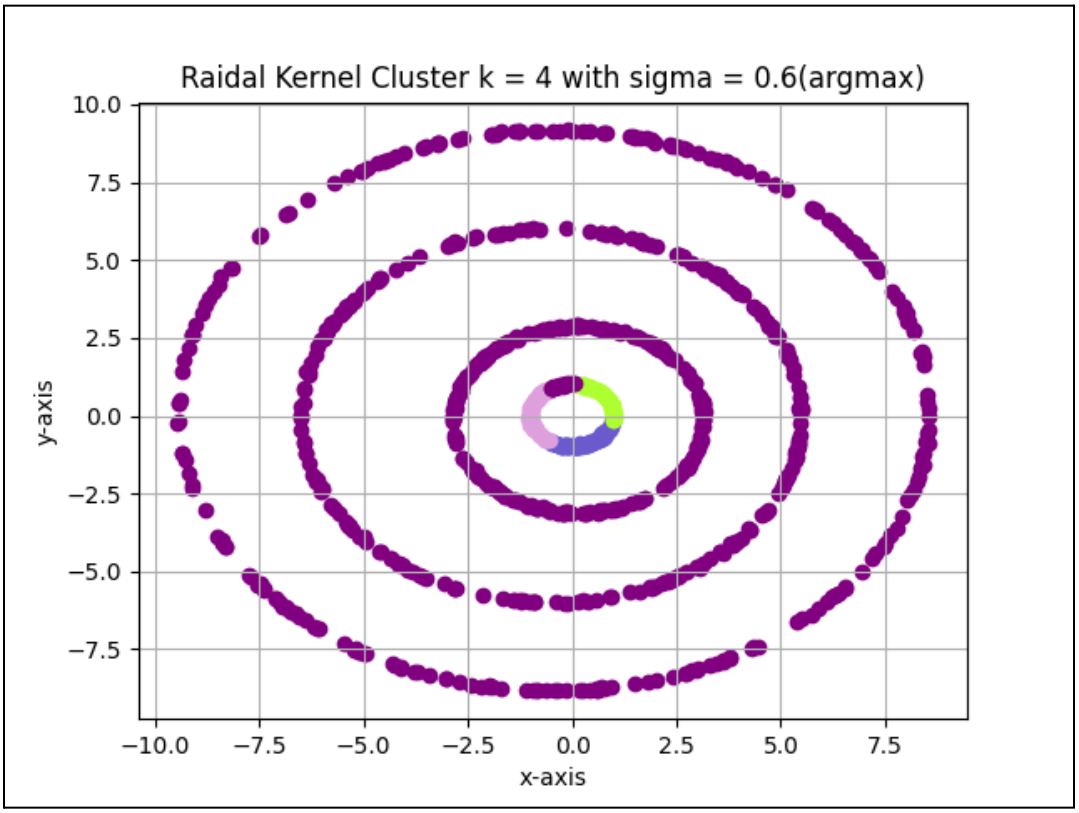


Figure : Clustering using Radial Basis Kernel Function with sigma = 0.7(argmax)

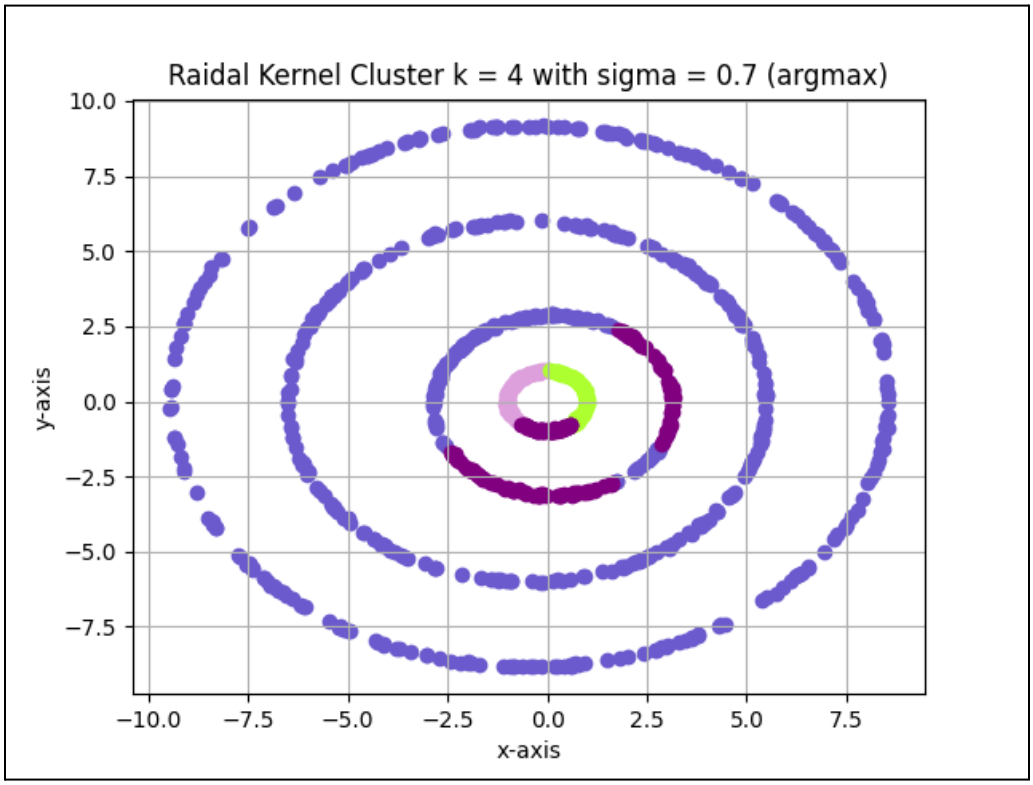


Figure : Clustering using Radial Basis Kernel Function with sigma = 0.8(argmax)

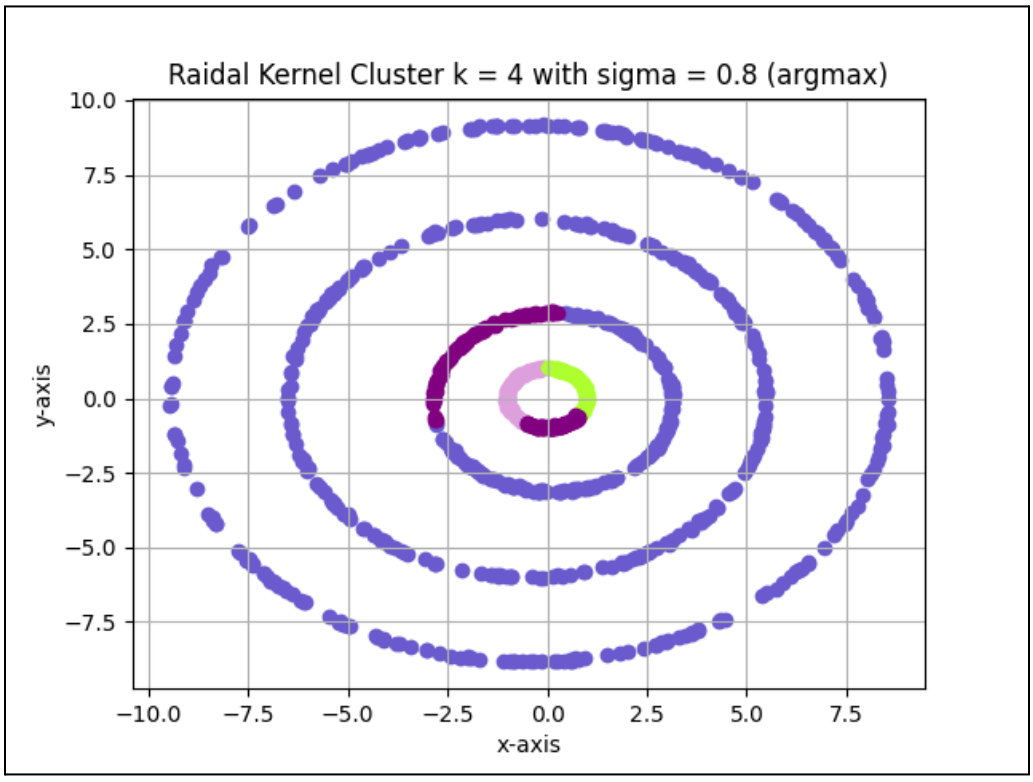


Figure : Clustering using Radial Basis Kernel Function with sigma = 0.9 (argmax)

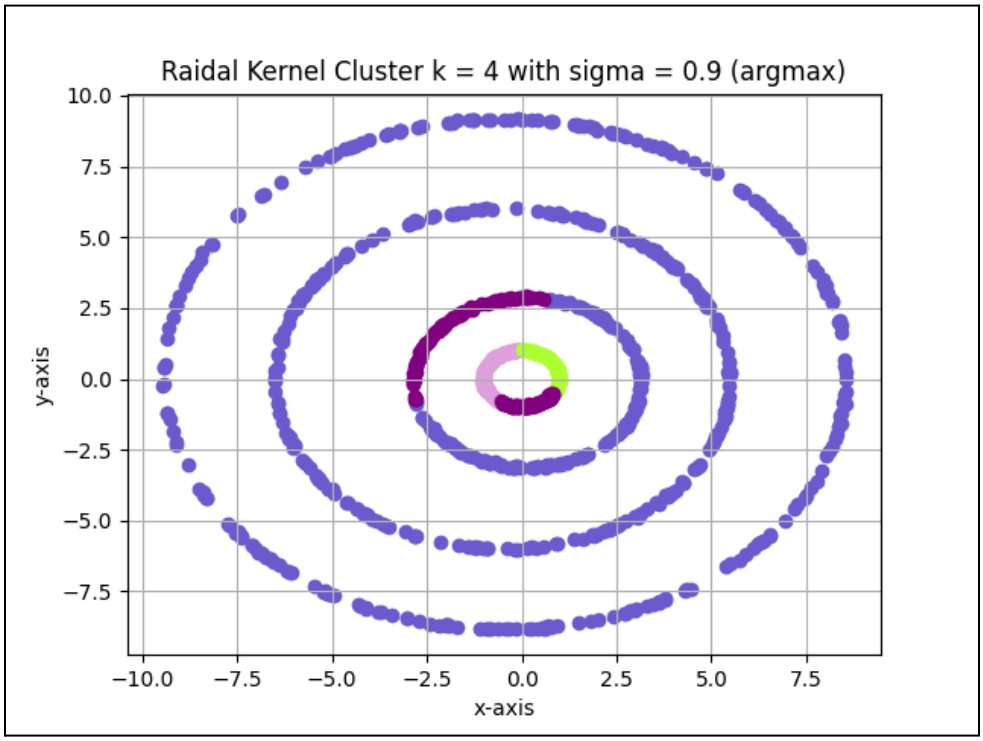


Figure : Clustering using Radial Basis Kernel Function with sigma = 1 (argmax)