Jashaswimalya Acharjee
CS22E005
September 11, 2022

CS 5691: Pattern Recognition and Machine Learning
Assignment: 1
Course Instructor : Arun Rajkumar

Question: 1

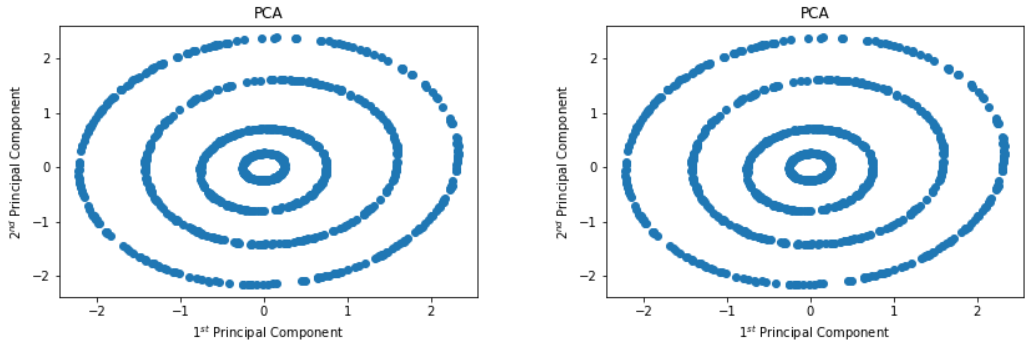
You are given a data-set with 1000 data points each in R2.

(i): Write a piece of code to run the PCA algorithm on this data-set. How much of the variance in the data-set is explained by each of the principal components?

Ans: Due to the data set provided to us for the assignment being non-linear in nature *(Concentric circular or elliptical pattern)*, normal PCA Algorithm fails to produce satisfiable results. Thus, it fails to capture the key features of the data. PCA only guarantees, dimensionality reduction in the context of linear data. Hence, the goal of variance maximization along Principal Components completely fails in this case.

(ii): Study the effect of running PCA without centering the data set. What are your observations? Does Centering help?

Ans: Generally, computing covariance matrix implicitly performs centering. Variance, by definition, is the average squared deviation from the mean. Thus, both Centered and Non-Centered data will have identical covariance matrices. Henceforth, centering does not make any difference. Moreover, the data set given to us is already centered in nature — and produces the same results whatsoever.



(a) PCA Before Centering (b) PCA After centering

Figure 1: PCA Comparison before and after centering

NOTE: Respective Source Codes for all the programs given in the Assignment Question can be found inside the zipped archive containing this **Report.pdf**:

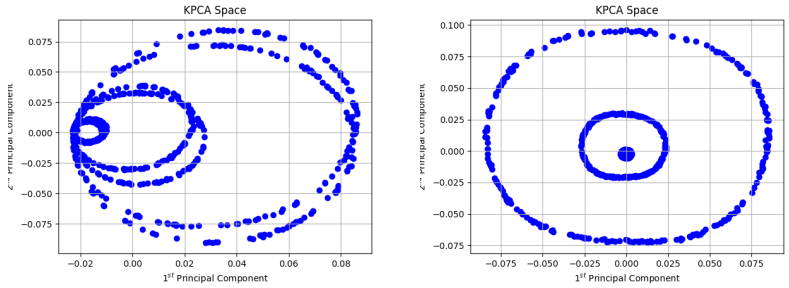
(iii): Write a piece of code to implement the Kernel PCA algorithm on this data set. Use the following kernels :

a $k(x, y = (1 + x^T)^d)$ **for** $d = 2, 3$

b $k(x, y) = e^{\frac{-(x-y)^T(x-y)}{2\sigma^2}}$ **for** $\sigma = 0.1, 0.2, \ldots, 1$

Plot the projection of each point in the dataset onto the top-2 components for each kernel. Use one plot for each kernel and in the case of (B), use a different plot for each value of σ.

Ans: From a visually observational point of view, it can be concluded that the Gaussian Kernel with $\sigma = 1$, performs the best for this dataset. `Ref:` Fig: 6 and Fig: 2 for a visual depiction of the same.



(a) For $d = 2$ (b) For $d = 3$

Figure 2: Kernel PCA with Polynomial Kernel

(iv): Which Kernel do you think is best suited for this dataset and why?

Ans: From the observational study, it can be concluded that the Gaussian Kernel with $\sigma = 1$, performs the best for this dataset. This is because the given dataset is non-linearly distributed, and Gaussian Kernel is suitable for such cases, instead of Polynomial kernel. A Polynomial kernel should ideally be the right choice for decision boundaries which are polynomial in shape, on the other hand, a Gaussian kernel is typically a good choice for distinguishing data points based on the distance from a common centre — In our case the dataset is in a concentric circular pattern.

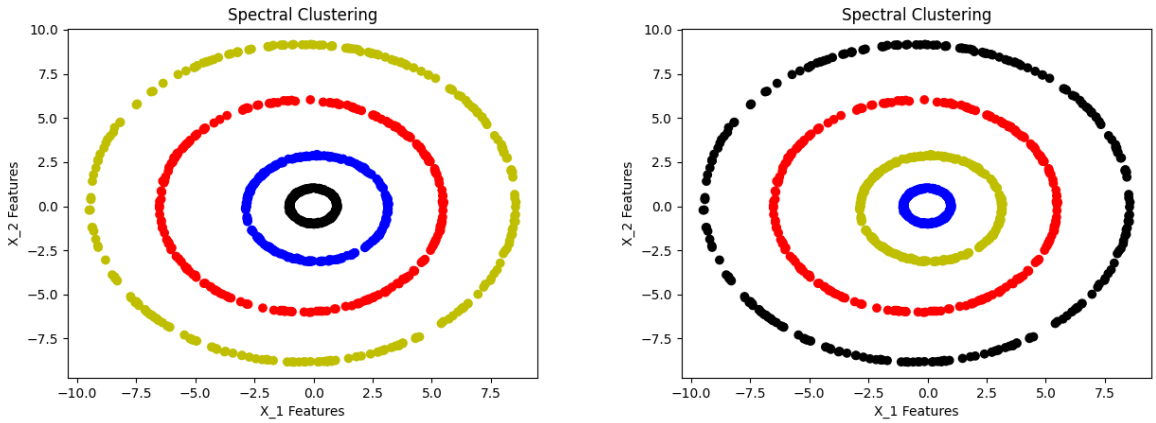


Figure 3: Spectral Clustering

Question: 2

You are given a data-set with 1000 data points each in $\mathbb{R}^2$.

(i): Write a piece of code to run the algorithm studied in class for the K-means problem with $k = 4$. Try 5 different random initialization and plot the error function w.r.t iterations in each case. In each case, plot the clusters obtained in different colors.

Ans: For $k = 4$, with 5 different random initialization, the error function or SSE (Sum of Squares Error) has been plotted. The varies from run-to-run, but doesn't fluctuate by a huge margin. The plots in Fig: 4, depict the visualizations for the same.

(ii): Fix a random initialization. For $K = 2, 3, 4, 5$, obtain cluster centers according to K-means algorithm using the fixed initialization. For each value of K, plot the Voronoi regions associated to each cluster center. (You can assume the minimum and maximum value in the data-set to be the range for each component of $\mathbb{R}^2$).

Ans: The plots in Fig: 5, help in visualizing the plots for Different values of K with clear separation of Voronoi regions.

(iii): Run the spectral clustering algorithm (spectral relaxation of K-means using Kernel-PCA) $k = 4$. Choose an appropriate kernel for this data-set and plot the clusters obtained in different colors. Explain your choice of kernel based on the output you obtain.

Ans: For the Spectral Clustering Assignment:

- First a Similarity Matrix has been constructed.
- Then the unnormalized Laplacian Matrix has been computed.
- Followed by Eigenvalue Decomposition of the Laplacian Matrix to obtain top k generalized Eigenvectors.

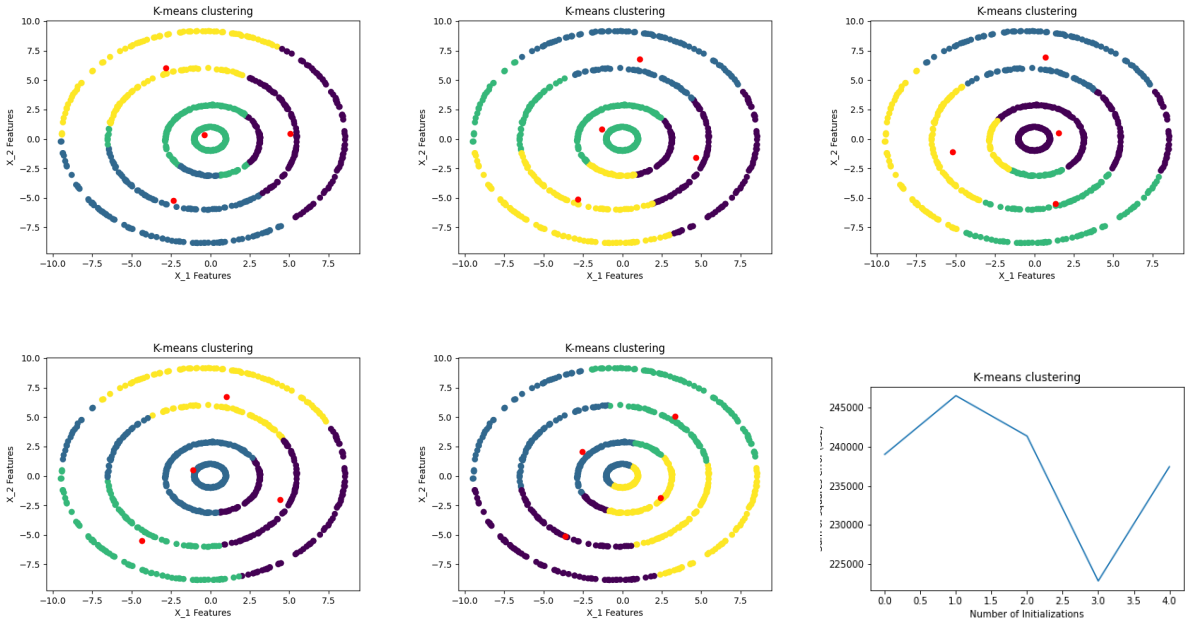
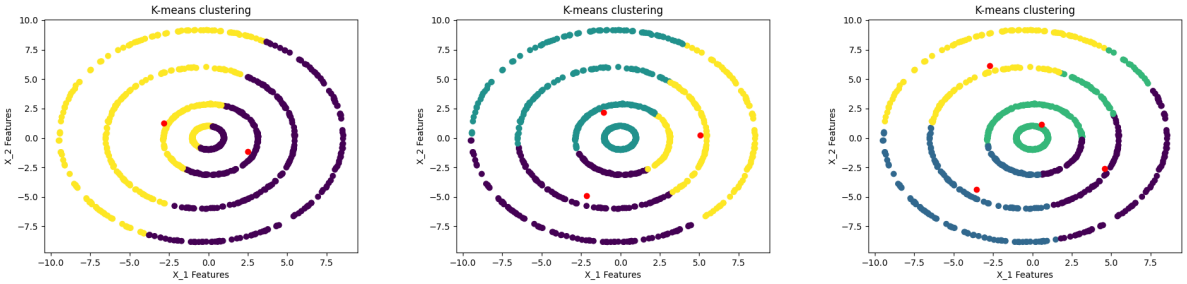


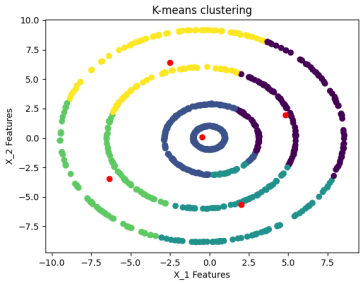
Figure 4: Kmeans Clustering Comparison between 5 random initialization run with $k = 4$.



(a) For $K = 2$.

(b) For $K = 3$.

(c) For $K = 4$.

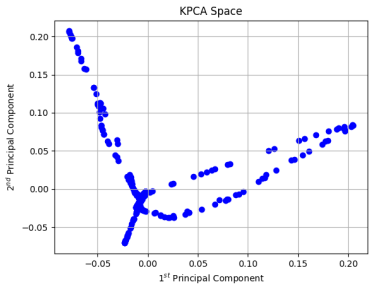


(d) For $K = 5$.

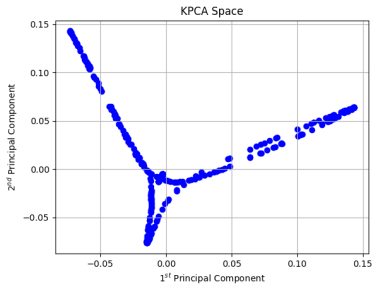
Figure 5: KMeans Clustering Comparison for $k \in \{2, 3, 4, 5\}$.

- Let the vectors, $w_1, w_2, w_3, \ldots, w_k$ be the eigenvectors, as the columns of a matrix $U \in \mathbb{R}^{n \times k}$.
  - Map $y_i \in \mathbb{R}^k$ to the i^{th} row of U.
- Cluster the points in $y_i \in \mathbb{R}^k$ with Kmeans Algorithm into Clusters $z_1, z_2, \ldots, z_k$.

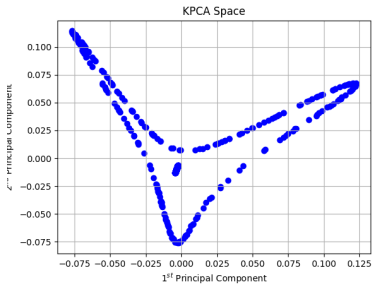
For the Kernel used in KMeans we have used a Gaussian Kernel, which gives somewhat clear separation between individual rings. A Gaussian kernel is typically a good choice for distinguishing data points based on the distance from a common centre.



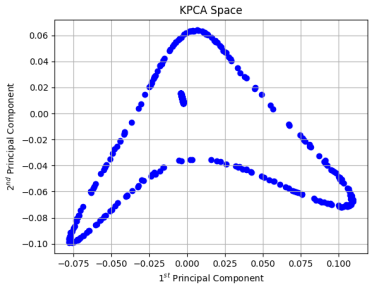
(a) For $\sigma = 0.1$



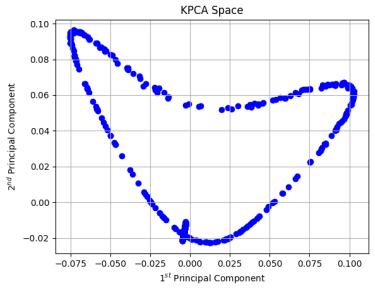
(b) For $\sigma = 0.2$



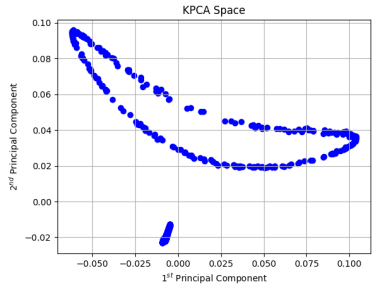
(c) For $\sigma = 0.3$



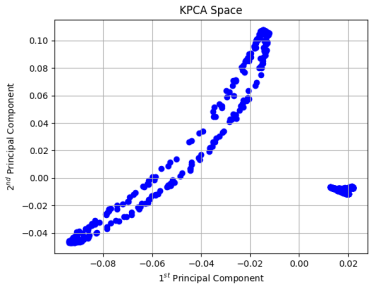
(d) For $\sigma = 0.4$



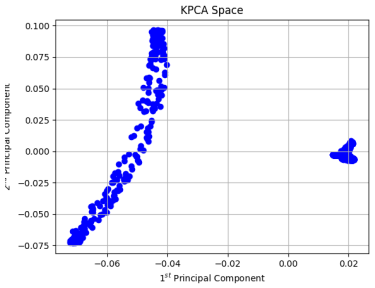
(e) For $\sigma = 0.5$



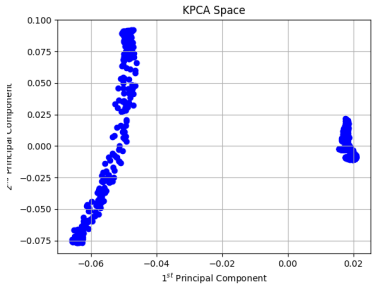
(f) For $\sigma = 0.6$



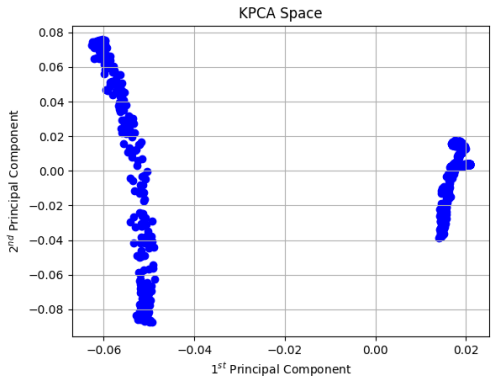
(g) For $\sigma = 0.7$



(h) For $\sigma = 0.8$



(i) For $\sigma = 0.9$



(j) For $\sigma = 1$

Figure 6: Kernel PCA with Exponential (or Gaussian) Kernel