

CS5691: Pattern Recognition and Machine Learning

Assignment 1

K Kamalesh Kumar
CE20B054

March 1, 2023

This report explains the outcomes of the observed results while implementing unsupervised learning algorithms like PCA, KernelPCA, Naive K-Means Clustering and Spectral Clustering.

1 Question 1

1.1 Part 1

Figure 1 show the visualization of the top-10 principal components. The variance in the data on each principal component will be contained in their corresponding eigen vectors. Since,

$$\lambda_i = \frac{1}{n} \sum_{j=1}^n (x_j^T w_i)^2 \quad (1)$$

where w_i is the eigen vector corresponding to λ_i . Figure 2 plots the variance contributed by each principal component

1.2 Part 2

Figure 3 displays a few images on the test data reconstructed for different top-d principal components. To choose a d for a compressed representation of the images, it is safe to consider the top- d eigenvectors which explain atleast 95% of the total variance in all the principal directions. In other words,

$$d \text{ s.t. } \frac{\sum_{k=1}^d \lambda_k}{\sum_{k=1}^n \lambda_k} \geq 0.95 \quad (2)$$

For the given dataset, $d = 154$. Figure 4 explains the same result.

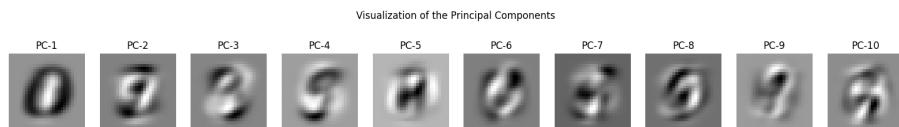


Figure 1: Visualizing the top-10 principal components

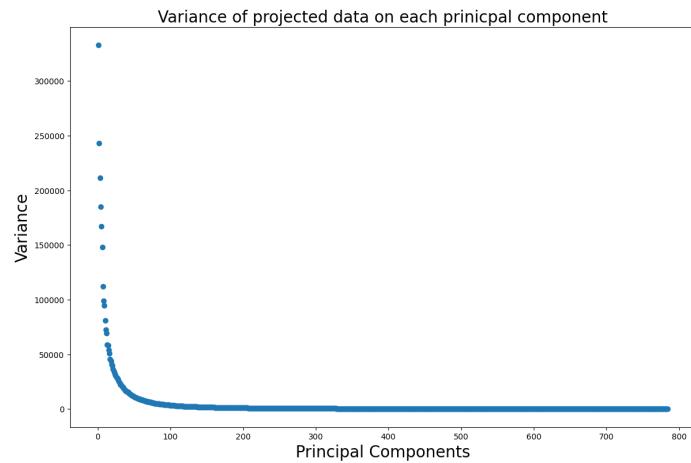


Figure 2: Variance explained by each principal component

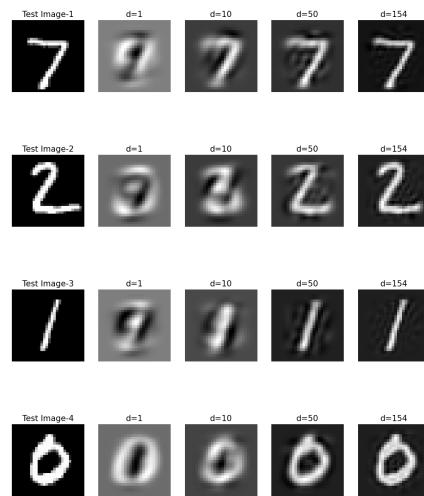


Figure 3: Sample images reconstructed for different top-d principal components

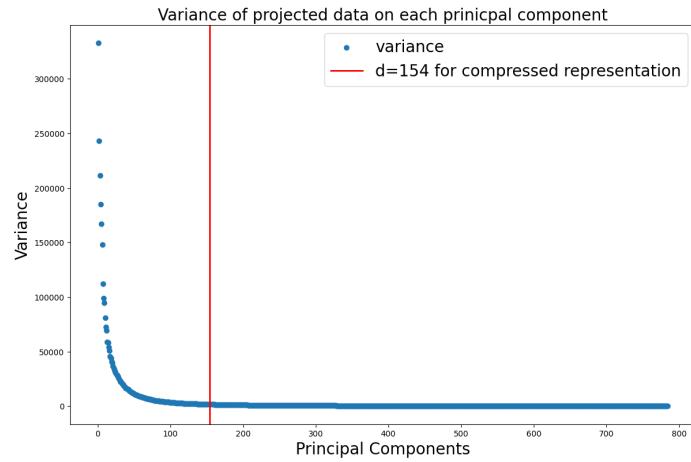


Figure 4: Optimal dimension for a compressed representation

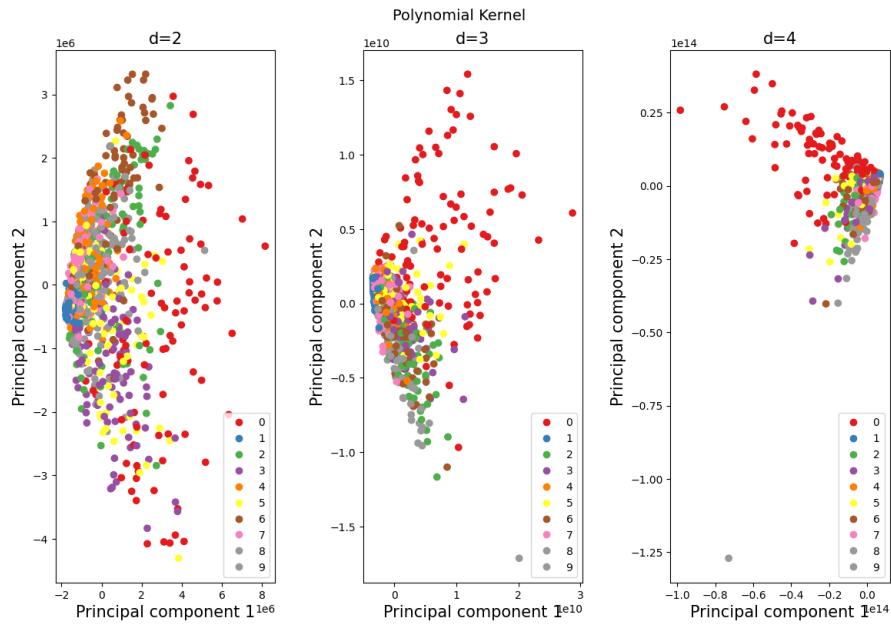


Figure 5: Kernel PCA top-2 component projection for a polynomial kernel

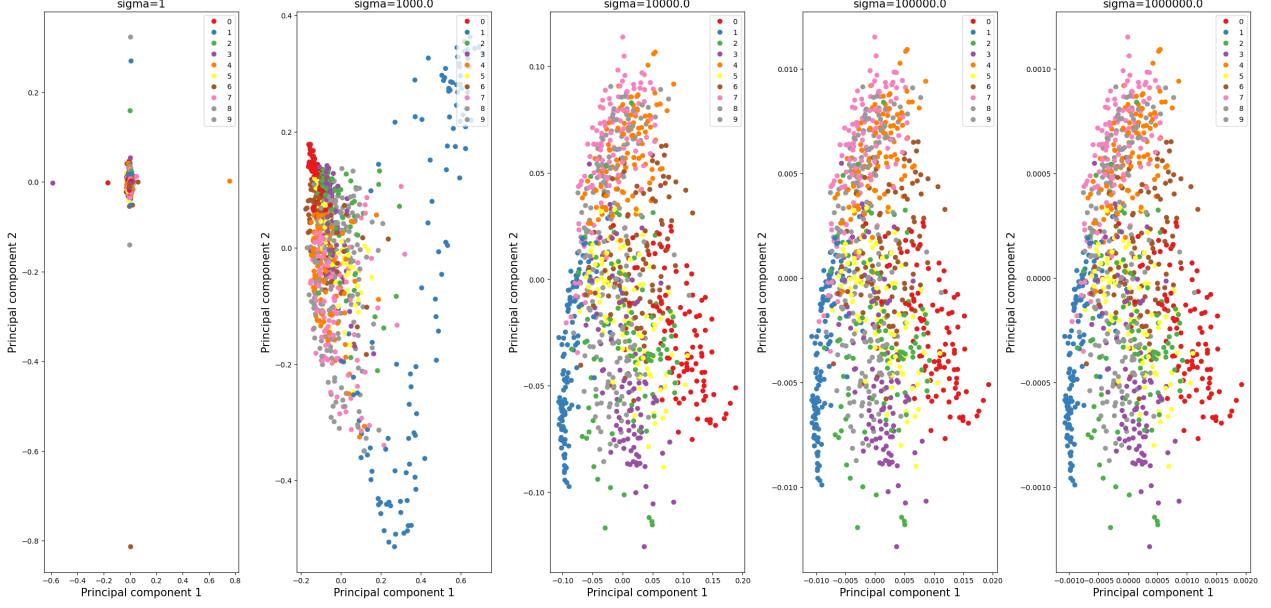


Figure 6: Kernel PCA top-2 component projection for a Gaussian kernel

1.3 Part 3

Figure 5 plots the projection on the top-2 principal components for the polynomial kernel for different degrees. Figure 6 explains the same for the Gaussian kernel for different values of σ . Since the images are of 784 dimensions, the norm-squared difference $\|x - y\|^2$ term present in the gaussian kernel makes the kernel entries to be exponentially very large. Since this goes beyond the precision tolerated by matrix libraries, the plots for low values of $\sigma = \{0.1, 0.2, \dots, 1\}$ are not meaningful. Since the variance in $\|x - y\|^2$ is also large, it requires us to explore larger values of σ to produce ideal principal directions. Thus in the experiments, $\sigma = 10^4, 10^5$ orders were found to be more suitable as portrayed by Figure 6

1.4 Part 4

The optimal kernel was selected on the same rule as stated in expression 2, and that kernel which needed the least number of principal components to explain 95% variance should be selected. From Figures 7 and 8, it is clear that the Gaussian Kernel with $\sigma = 10^5$ needed the least number of dimensions ($d = 129$, which is lesser than what was required in Naïve PCA). It can also be seen in comparison with Polynomial Kernel that, RBF kernels give rise to more spread in the projected representations, whereas the former is dense with lower variance.

2 Question 2

2.1 Part 1

Figure 9 shows the obtained clustering for the naïve K-Means with each datapoint initialized randomly to a cluster after which the Lloyd's algorithm is run. Clearly, the final clustering does not converge to the desired clusters where the crescent is one cluster and the full moon is another. This is expected as this version of K-Means only learns linear-separable boundaries. Figure 10 shows the reduction in the

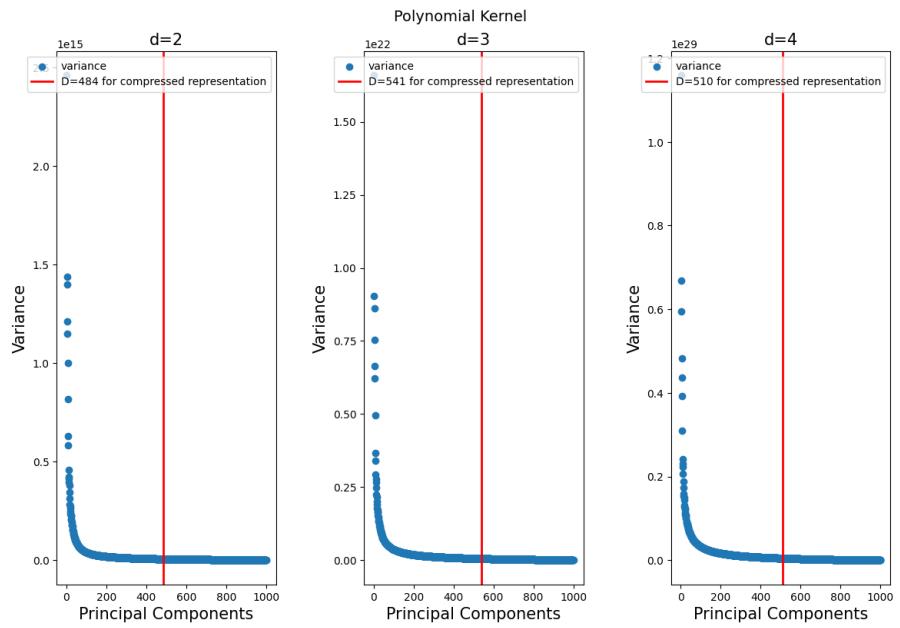


Figure 7: Required number of principal components for 95% explanation in variance- Polynomial kernel

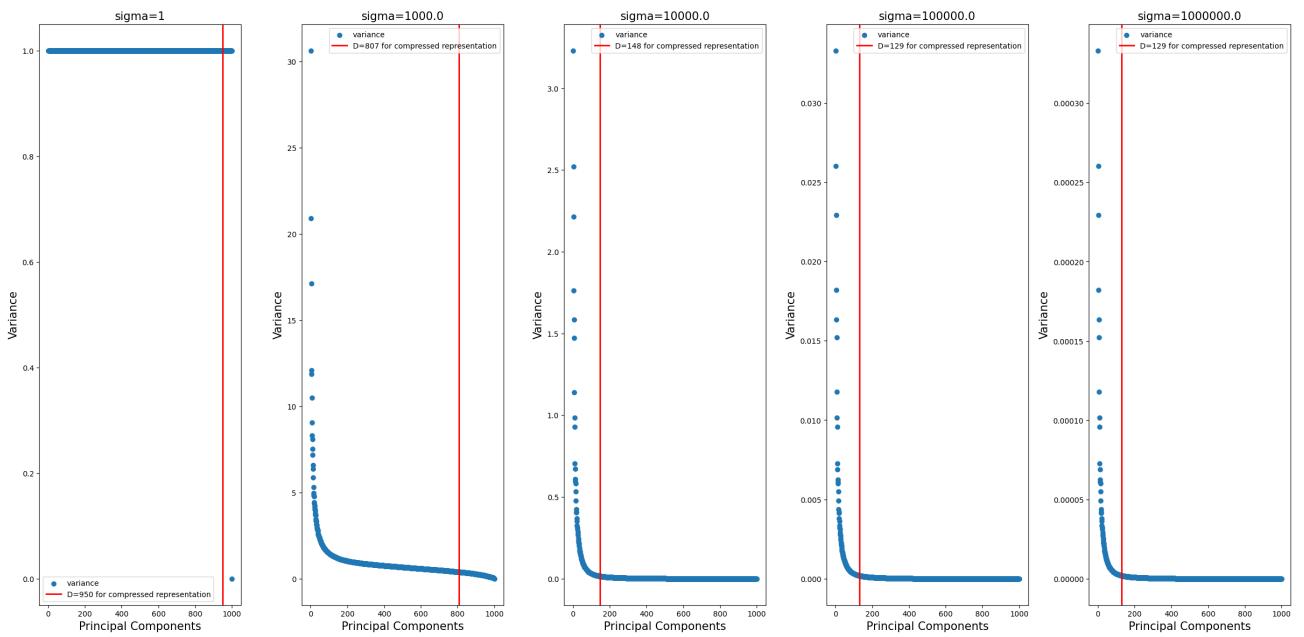


Figure 8: Required number of principal components for 95% explanation in variance- RBF kernel

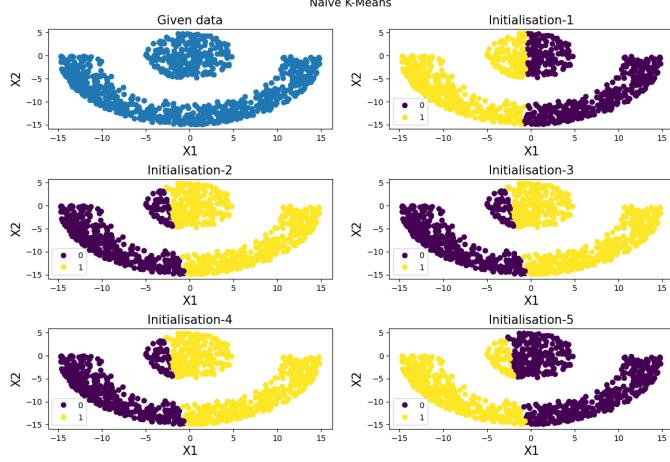


Figure 9: Naïve K-Means for 5 different initializations

error function w.r.t the iterations, where the error is defined as below,

$$E_t = \sum_{i=1}^n \|x_i - \mu_{Z_i}^t\|^2 \quad (3)$$

where t denotes the iteration number, Z_i is the cluster x_i is assigned at iteration t and $\mu_{Z_i}^t$ is the mean of the corresponding cluster at iteration t

2.2 Part 2

The Voronoi regions obtained for different cluster configurations "k" is plotted in Figure 11.

2.3 Part 3

For this experiment, two kernels, specifically polynomial and gaussian have been explored. To analyze the performance of each kernel, it is also important to visualize the mapping that is generated for all datapoints by choosing the two i^{th} components of the top-2 eigenvectors. This mapping is represented by the matrix H . In the existing approach for spectral clustering, the rows of H are normalized before running the Lloyds algorithm on it, but this step was not ideal in some cases, as it made the normalized representation not linearly separable. Thus two versions of the spectral clustering algorithm have been tried out, one in which the rows of H are normalized, and the other case were they remain unnormalized.

Since we know what the expected clustering we desire (as explained in part-1), we can visualize where each of the clusters is getting mapped in the space of the eigenvectors of the Kernel matrix. The normalized case is depicted in Figure 12 and 14, for Polynomial and the gaussian kernels, with different degrees and σ respectively.

Note that the purple and yellow colors do not indicate what the algorithm has clustered, which is to be discussed next, but rather the expected clusters and how they are distributed in the row space of H .

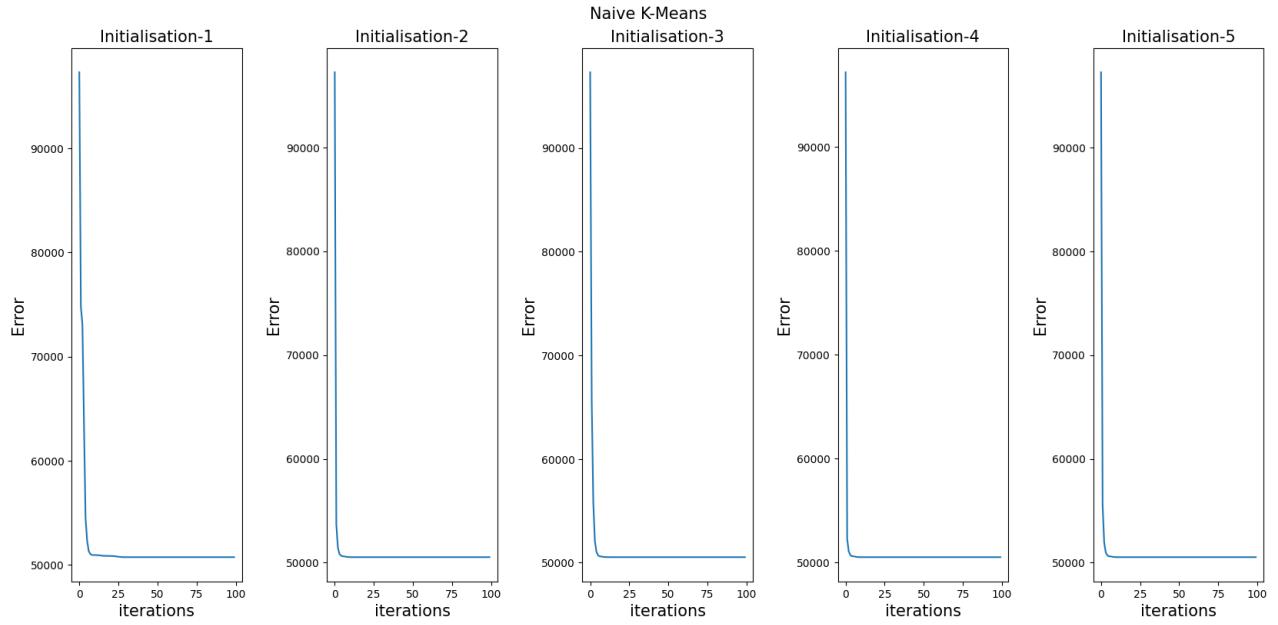


Figure 10: Plot of the error function w.r.t iterations for each initialization

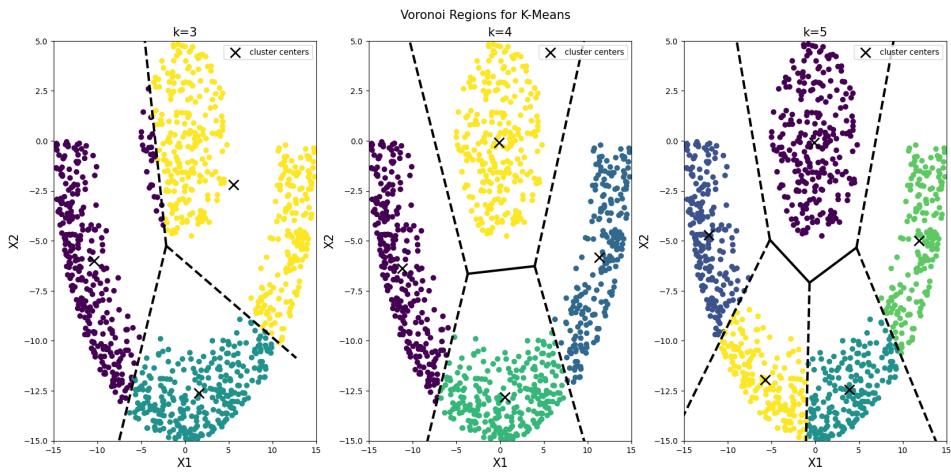


Figure 11: Voronoi regions for different cluster configurations k.

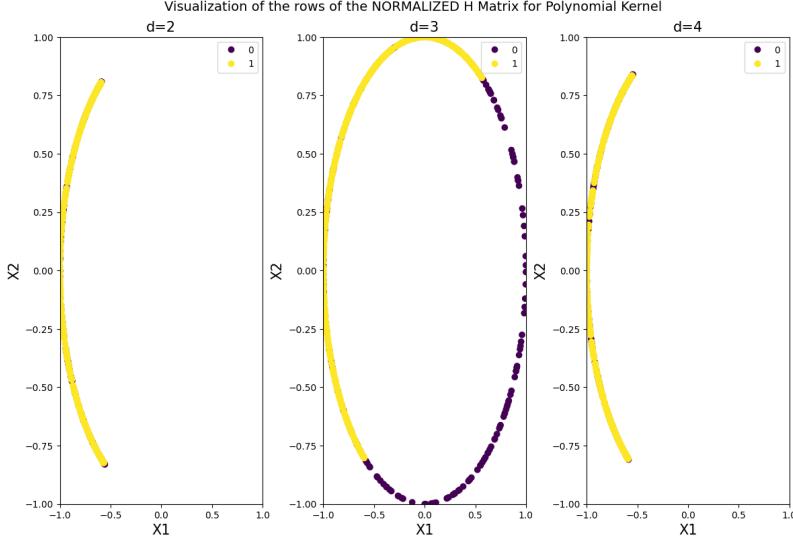


Figure 12: Visualization of the rows of the normalized H Matrix for Polynomial Kernel

Figures 13 and 15 represent the unnormalized scenario, and it is clear that some cases possess linear separability in the unnormalized version and not in the normalized case. The polynomial kernel with degree 2 clearly represents this distinction.

Similarly, we can also visualize the obtained clusterings after running the Lloyds algorithm for each of the cases, in different spaces. The figures following after this present the results of each case. In the majority of the cases, the final clustering did not converge to the expected clustering, both in the random initialization setting, as well as the initialization followed in K-Means++ (Initialisation here, assigns the rows of H as the cluster means before starting the Lloyds algorithm). Figures 24 and 25 present two specific initializations in which, the spectral clustering algorithm actually did converge to the expected clustering. It can also be seen that the ideal kernel for a dataset of this kind would be polynomial and not gaussian, since the mapping to the row space of H is not linearly separable for any σ value and thus would not converge to the expected clustering.

Thus, the ideal kernel choice would be a polynomial kernel with degree 2.

2.4 Part 4

The mapping and the obtained clusters still face the same outcomes as in the previous version of the spectral clustering algorithms. Essentially, in the row space of H , if a particular data point is more prominent in one principal direction than the other, it is assigned to a cluster corresponding to which direction is more principal. But this still has issues for the given dataset as such an assignment does not completely separate the two expected cluster groups. Thus, running Lloyds after this assignment, still makes it converge to clusters as in previous cases.

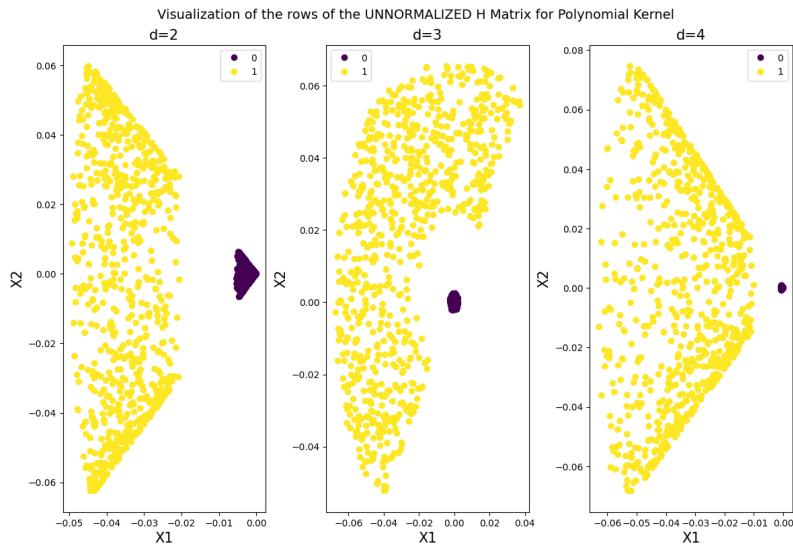


Figure 13: Visualization of the rows of the unnormalized H Matrix for Polynomial Kernel

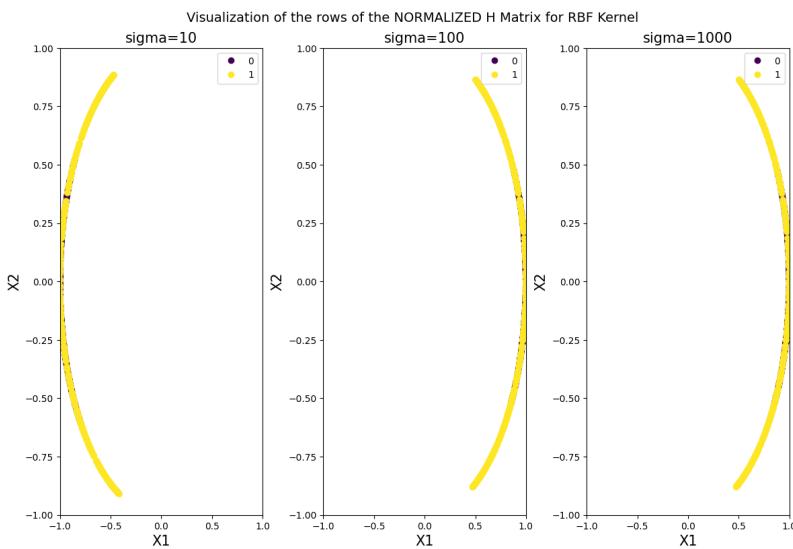


Figure 14: Visualization of the rows of the normalized H Matrix for RBF Kernel

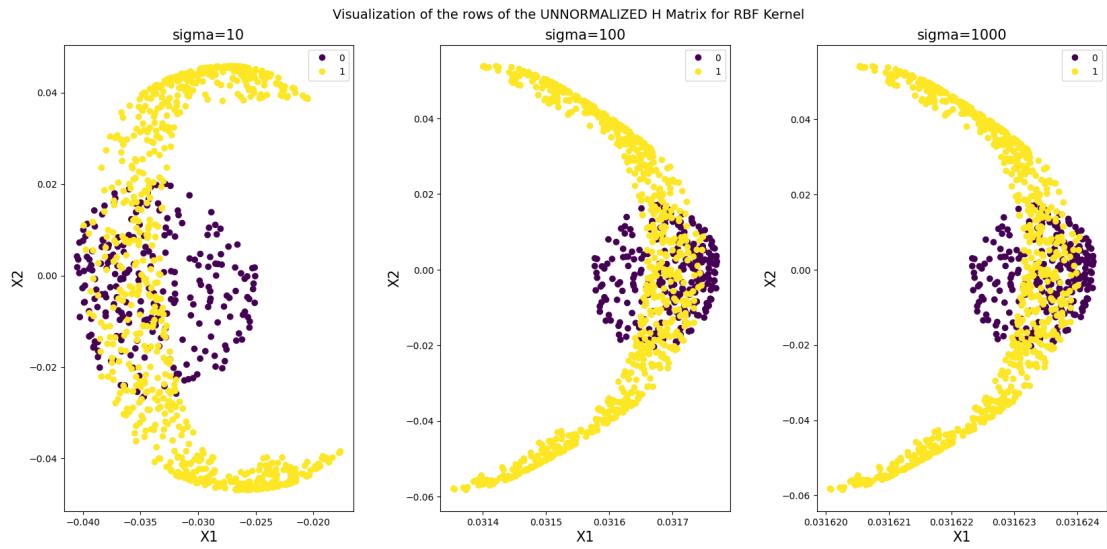


Figure 15: Visualization of the rows of the unnormalized H Matrix for RBF Kernel

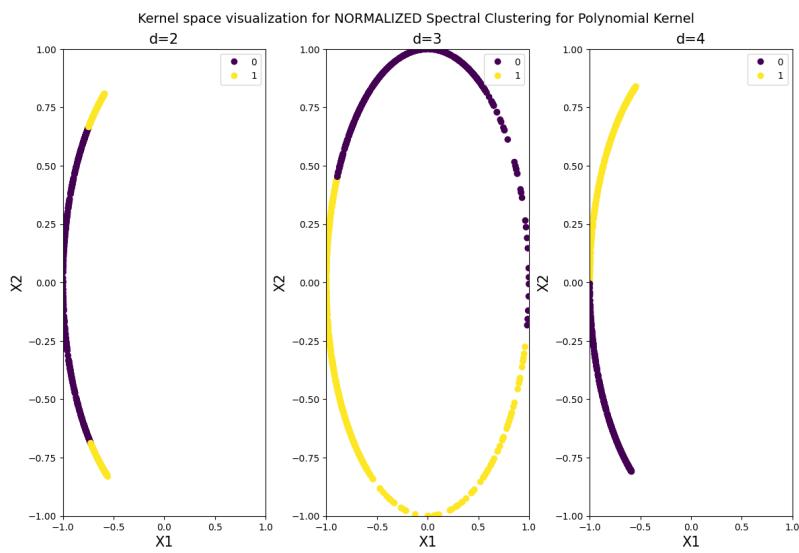


Figure 16: Obtained clusters after Lloyds algorithm, in the row space of H , for the normalized polynomial kernel

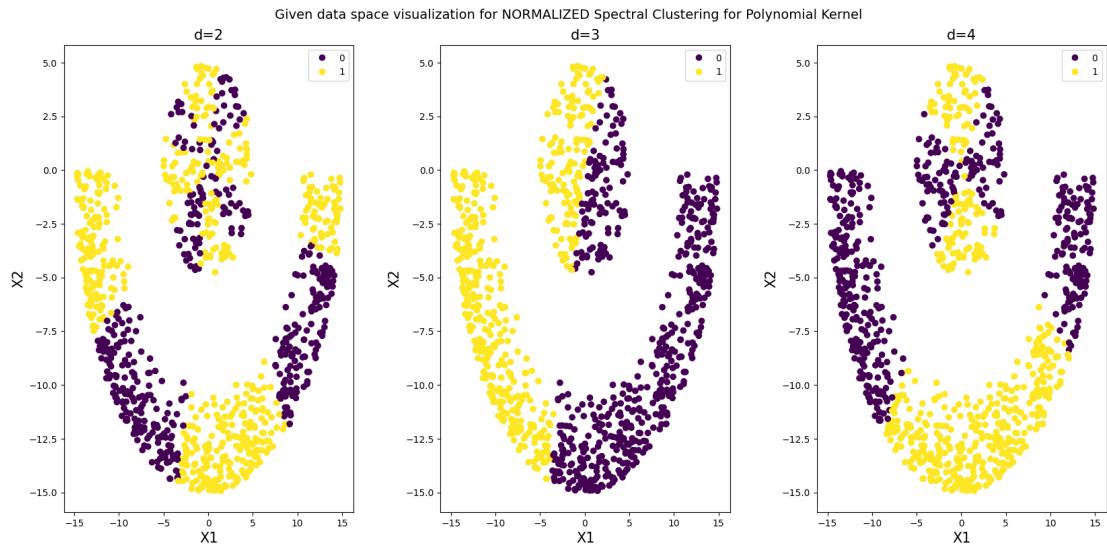


Figure 17: Obtained clustering in the given data, for the normalized polynomial kernel

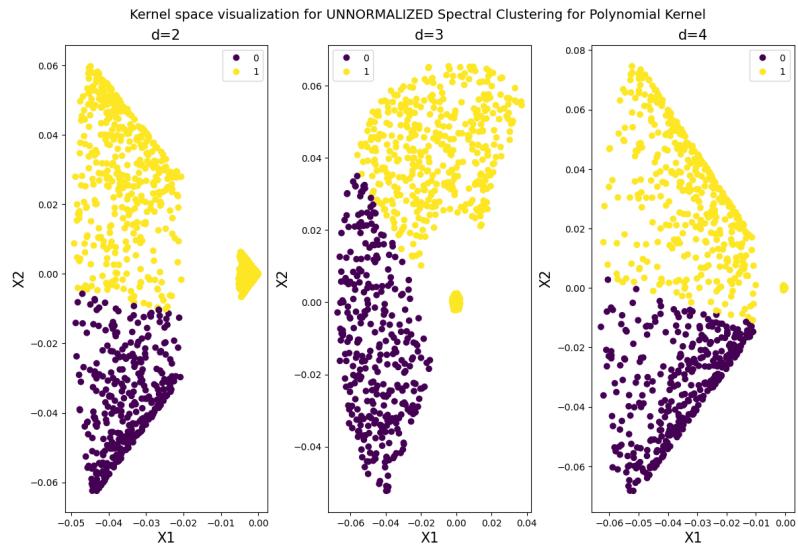


Figure 18: Obtained clusters after Lloyds algorithm, in the row space of H , for the unnormalized polynomial kernel

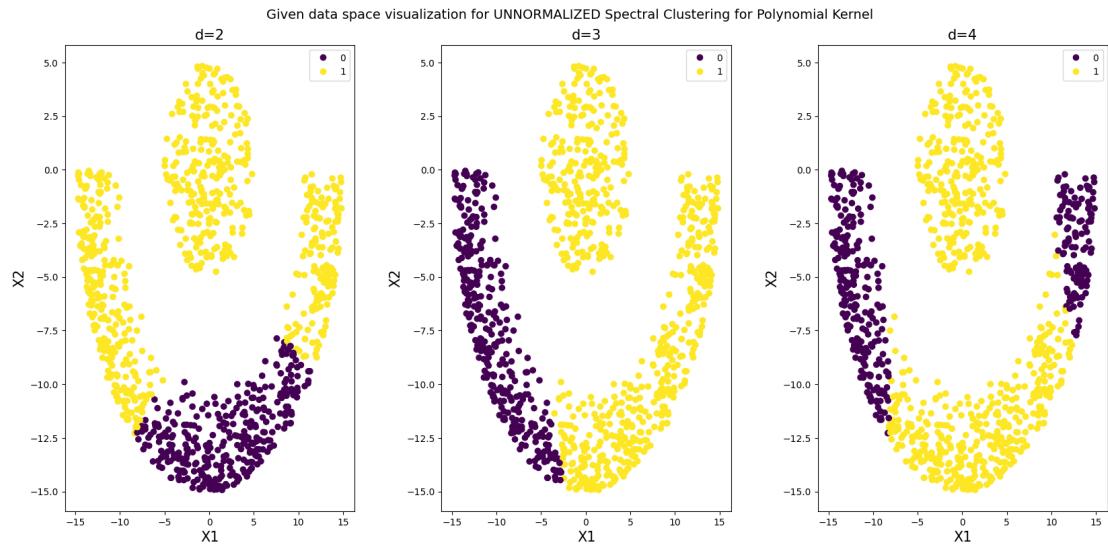


Figure 19: Obtained clustering in the given data, for the unnormalized polynomial kernel

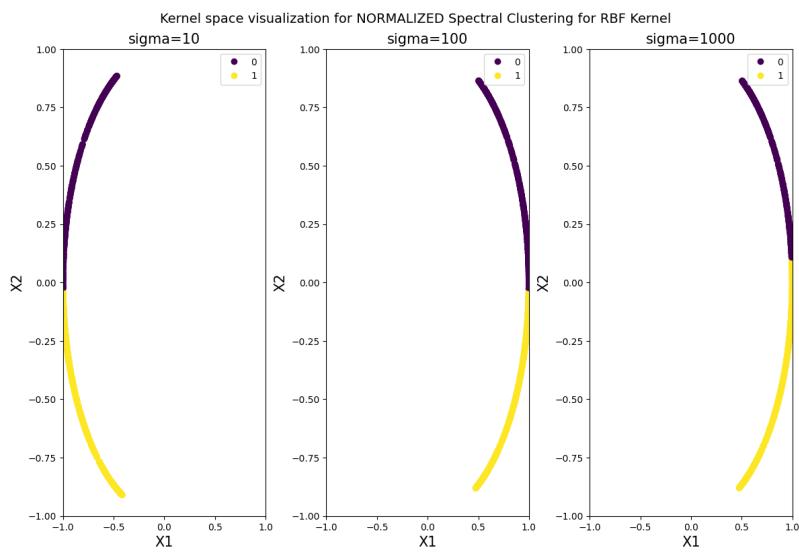


Figure 20: Obtained clusters after Lloyds algorithm, in the row space of H , for the normalized RBF kernel

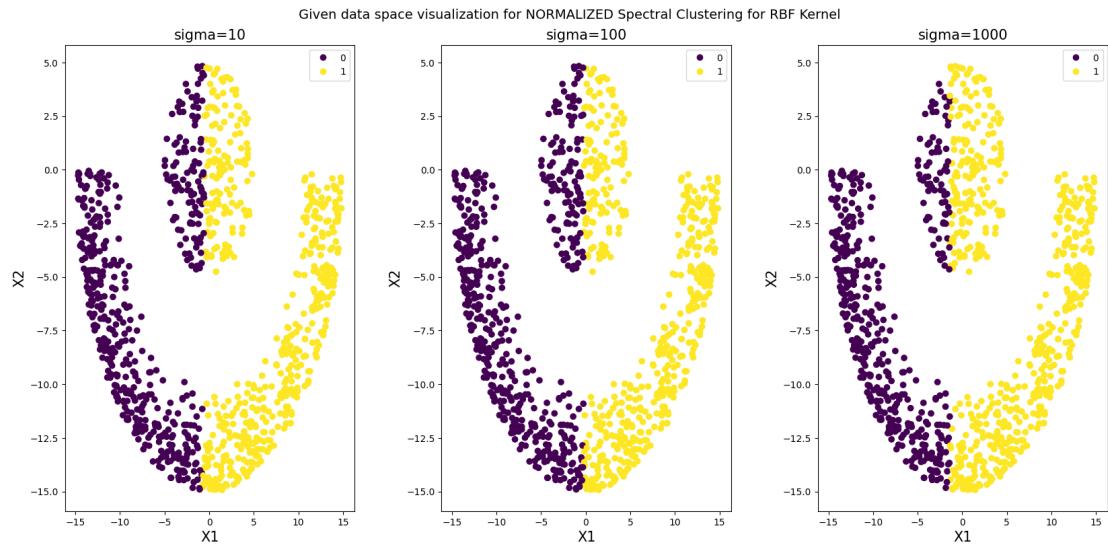


Figure 21: Obtained clustering in the given data, for the normalized RBF kernel

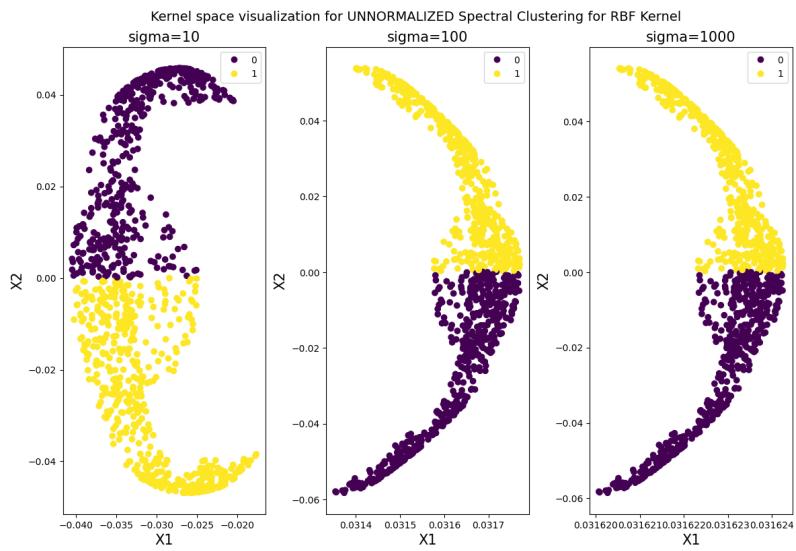


Figure 22: Obtained clusters after Lloyds algorithm, in the row space of H , for the unnormalized RBF kernel

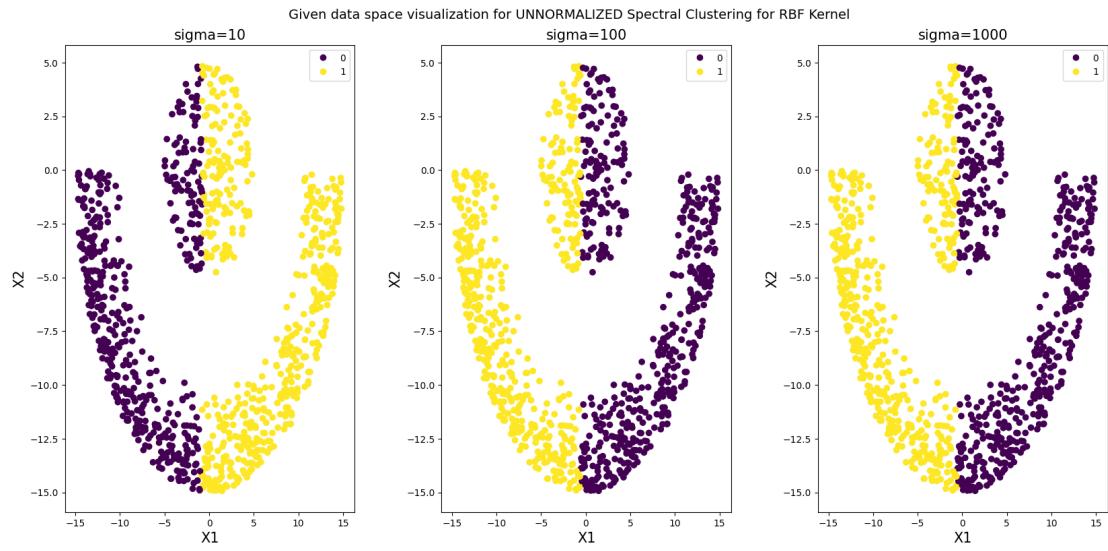


Figure 23: Obtained clustering in the given data, for the unnormalized RBF kernel

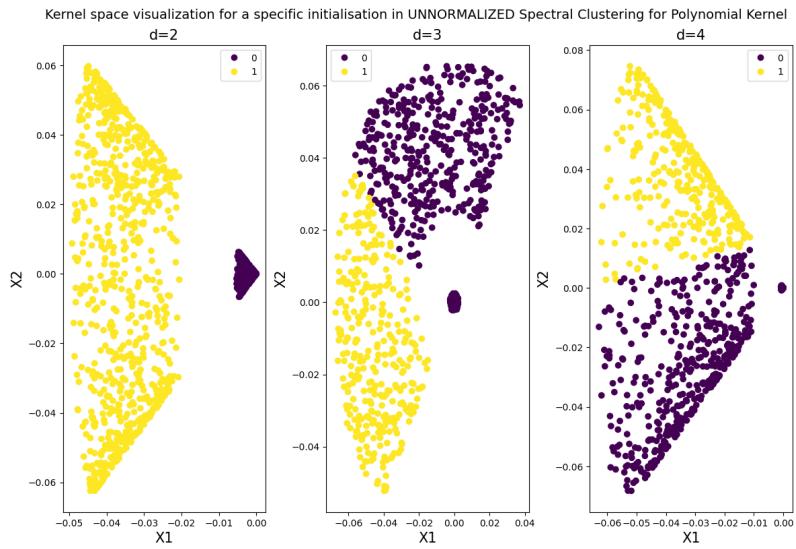


Figure 24: Obtained clusters for a specific initialization of unnormalized Polynomial Kernel, in the row space of H

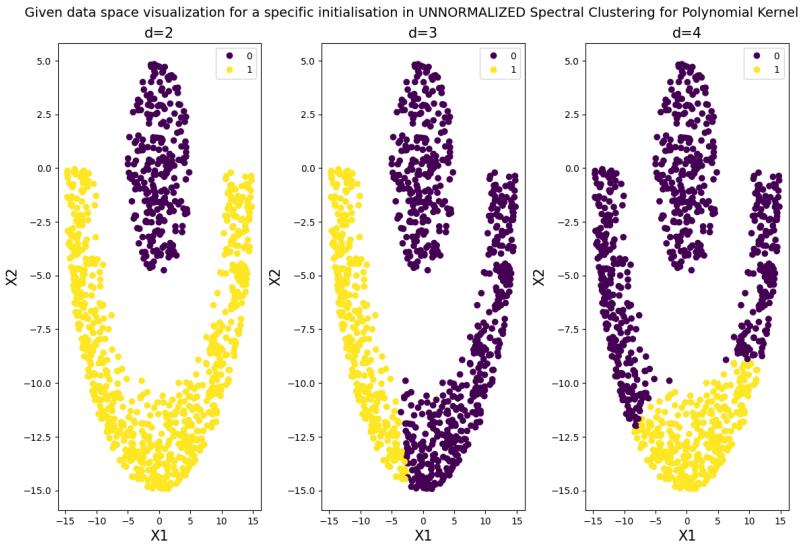


Figure 25: Obtained clusters for a specific initialization in the given data space, for unnormalized Polynomial Kernel.

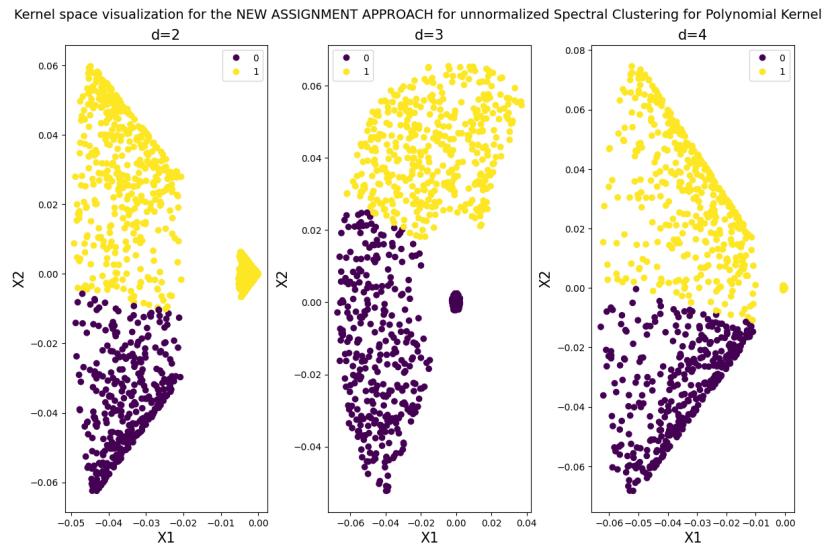


Figure 26: Visualization of the new assignment approach for unnormalized Polynomial Kernel, in row space of H

Given data space visualization for the NEW ASSIGNMENT APPROACH for unnormalized Spectral Clustering for Polynomial Kernel

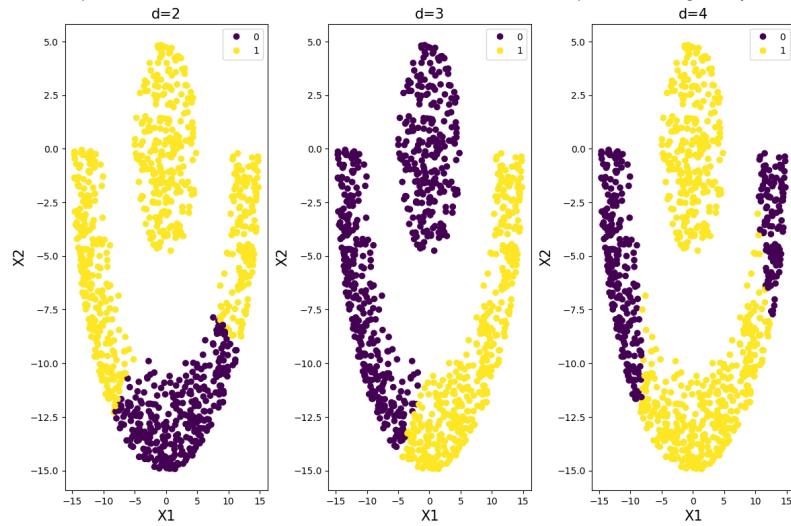


Figure 27: Visualization of the new assignment approach for unnormalized Polynomial Kernel, in the given data space