CS5691 Assignment 1

Pattern Recognition and Machine Learning

Arjun Vikas Ramesh

CS21B008

March 10, 2024

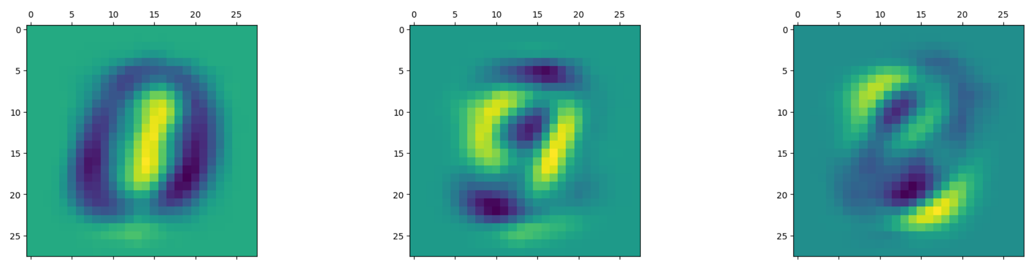

Figure 1: images of principal components

1 Question 1

1.1 PCA

We implemented PCA by performing the following steps:

- Center the data by finding mean.
- Compute the Covariance matrix
- Find eigenvalues and corresponding eigenvectors for the covariance matrix using numpy.
- Sort by the largest eigenvalues and the corresponding eigenvectors.

We obtained the following images of three principal components: $figure$ 1

Each Eigenvector explains the following percentage of variance:

1 explains 9.70 % of the variance.
2 explains 6.95 % of the variance.
4 explains 6.08 % of the variance.
5 explains 4.91 % of the variance.
6 explains 4.11 % of the variance.
7 explains 3.38 % of the variance.
8 explains 2.95 % of the variance.
9 explains 2.72 % of the variance.
...so on

As given in table 1 there is a great reduction in number of components as we reduce our variance coverage.

Percentage of variance explained	**Number of eigenvectors**
85	55
95	132
99	278
99.9	426
99.99	506
100	784

Table 1: Variance explained by number of eigenvectors

1.2 Dimensionality Reduction

From above table we can see that the data can be reduced to a lower dimension and still retain much information.

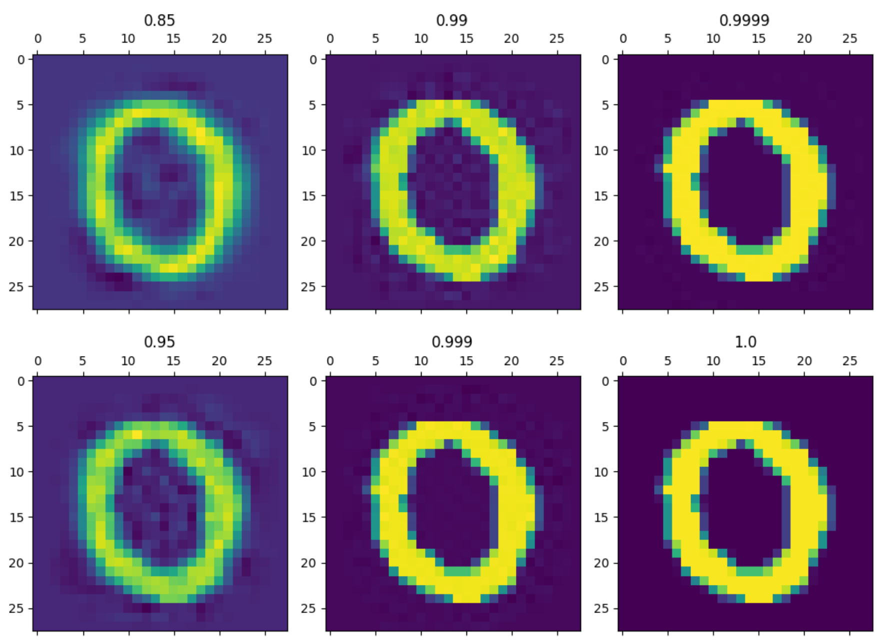

Figure 2: Reconstruction of a zero

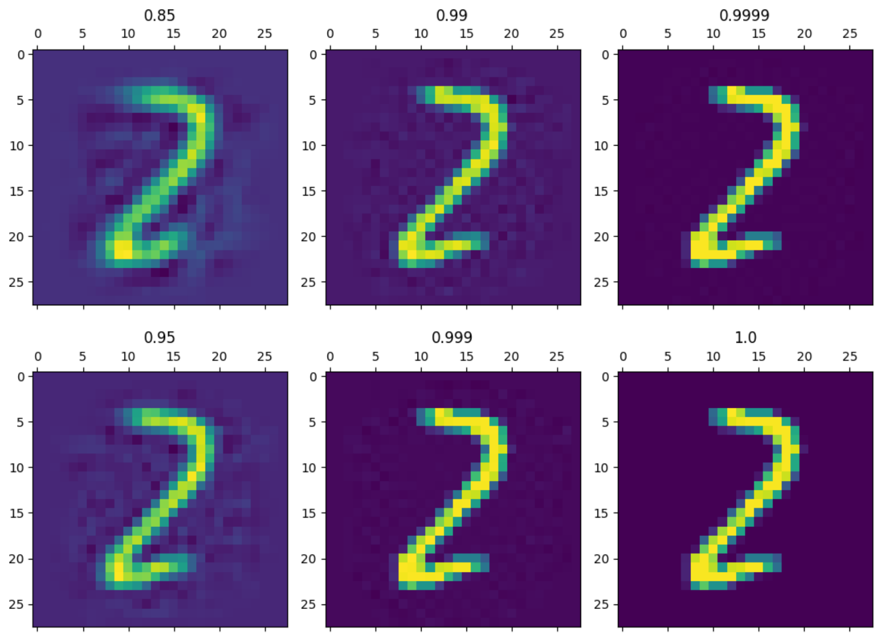

Figure 3: Reconstruction of a two

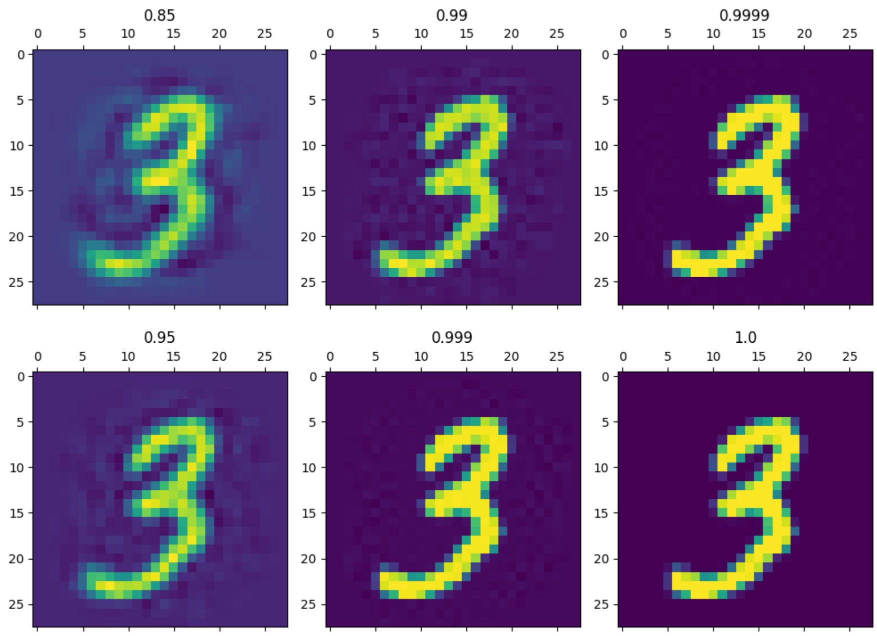

Figure 4: Reconstruction of a three

- We can see that the reconstructed data for 95% variance coverage requires only 132 principal components, while still being reasonably legible images.
- A dimension of 132 would be preferred over 278 since the benefits of reducing the dimension by half will outweigh the slight increase in accuracy of choosing the higher dimensional representation.

1.3 Kernel PCA

We implemented Kernel PCA by performing the following steps:

- Evaluate K matrix as a centered data.
- perform PCA on K.

$$\textbf{(A)}k(x, y) = (1 + x^T y)^d \textbf{ for } d = 2, 3, 4$$

Gives us the following graphs upon plotting the projections of data upon top 2 components: figures 5,6,7

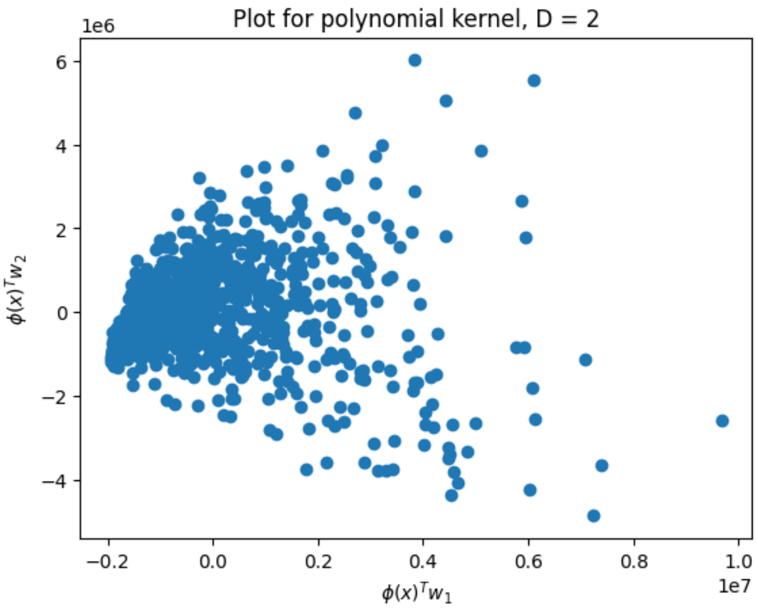

Figure 5: d = 2

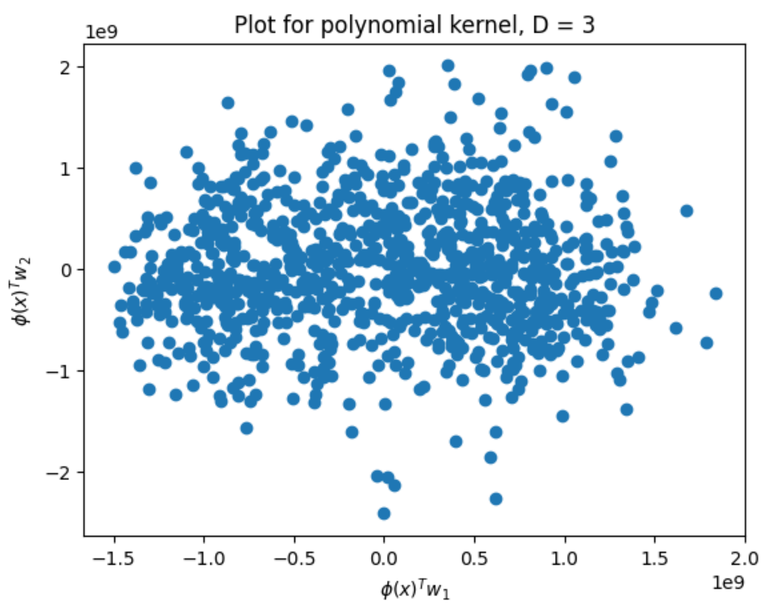

Figure 6: d = 3

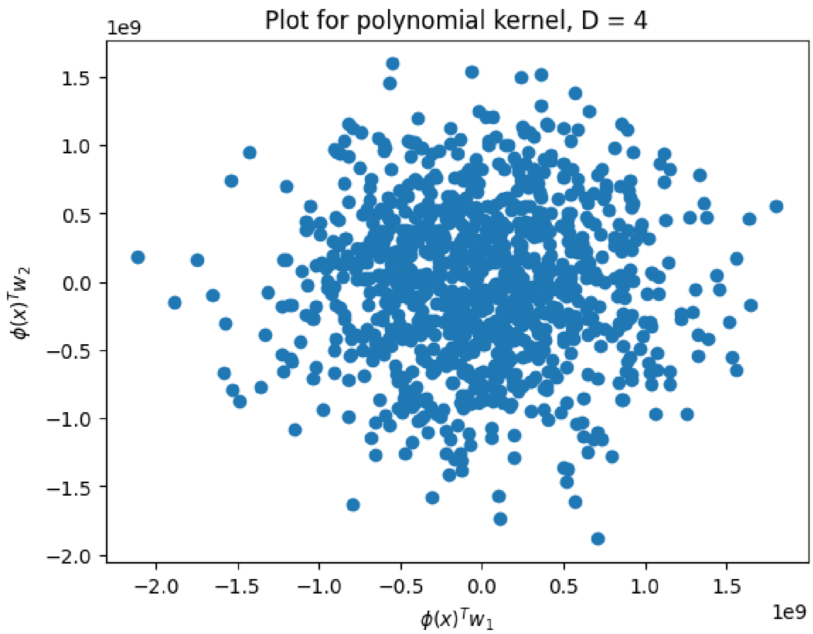

Figure 7: d = 4

$$\textbf{(B)}(x,y) = exp\frac{(xy)^T(xy)}{2\sigma^2}$$

Gives us the following graphs upon plotting the projections of data upon top 2 components: figures 8,9,10,11

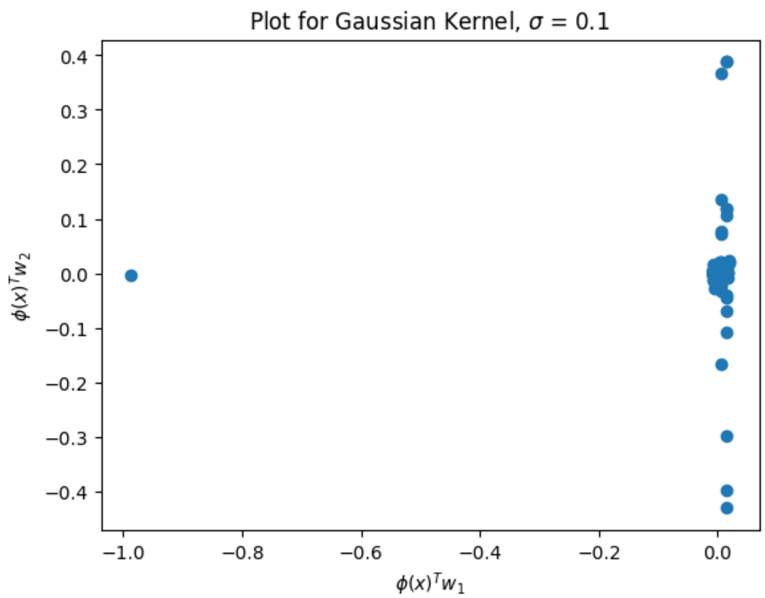

Figure 8: Enter Caption

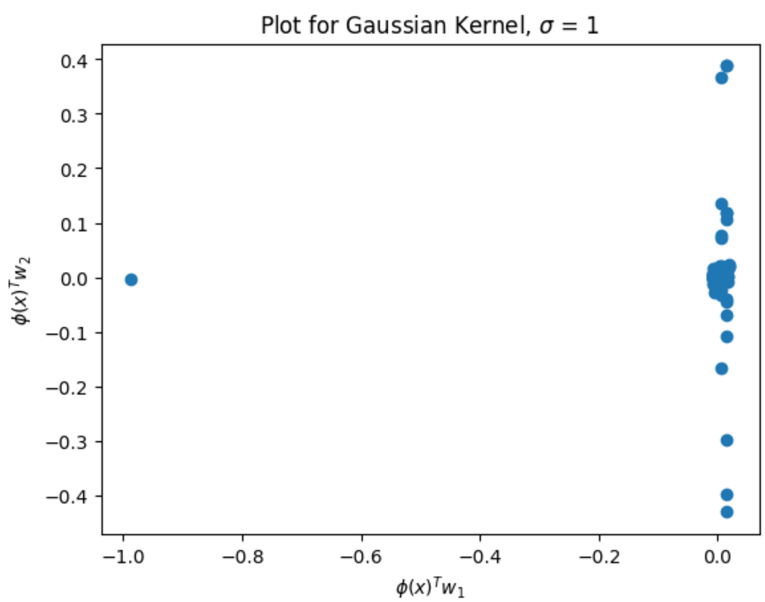

Figure 9: Enter Caption

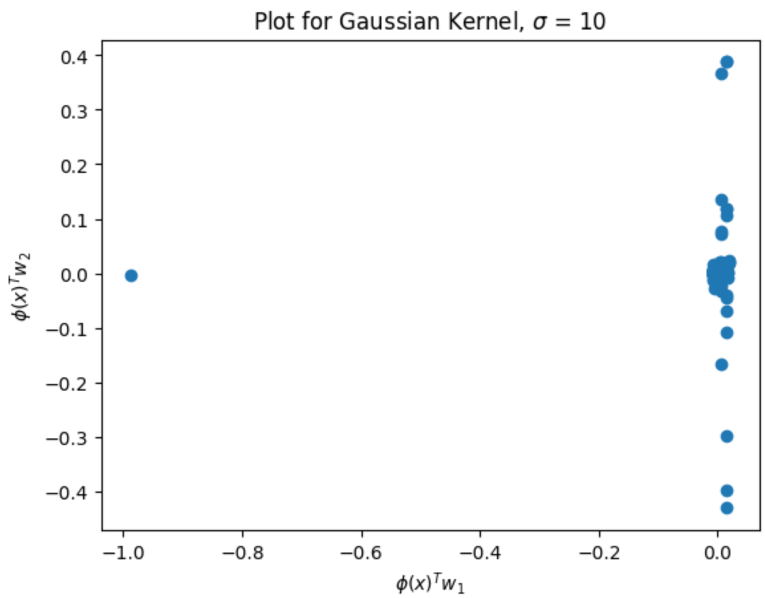

Figure 10: Enter Caption

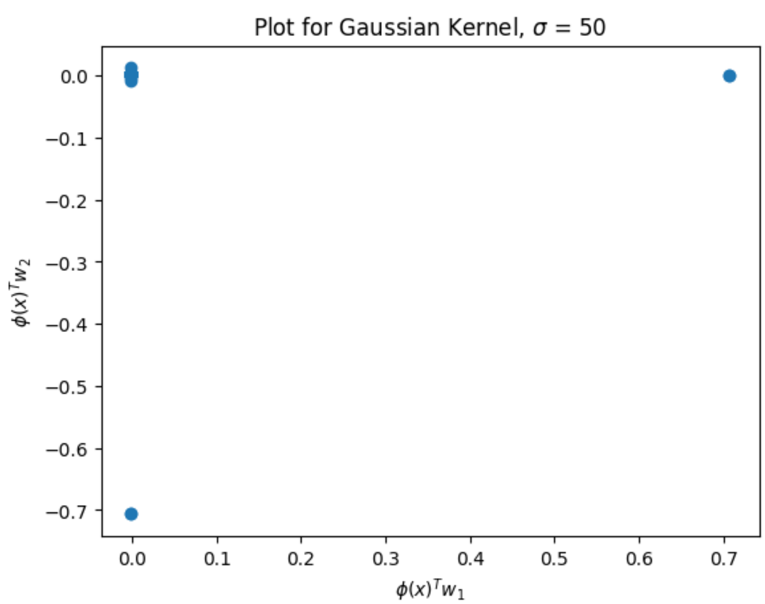

Figure 11: Enter Caption

1.4 The Better Kernel function

- By plotting the projections along two components, we can see that the polynomial kernel resulted in **no decorrelation** of features, where the plot is scattered around the origin.
- However in the case of Gaussian kernel, for $\sigma = 0.1, 1, 10$, the two components are **decorrelated** and focused along the y axis.
- However $\sigma = 50$ gives a more separated plot since it may nullify some numerators in the exponential equation.
- **Gaussian Kernel** is a better fit for the data, because of the above reasons.

2 Question 2

We implement the Llyod's algorithm in the following steps:

- Initialize the means of each cluster *(at random)*.
- Compute the error function for that iteration and if there exists a data point which has a lower MSE with a different cluster mean, we transfer that point to the other cluster.
- Recompute means and MSE and repeat until convergence.

2.1 5 random iterations of *K-means* Algorithm with k=2

We obtained the following graphs: Figures 12-16

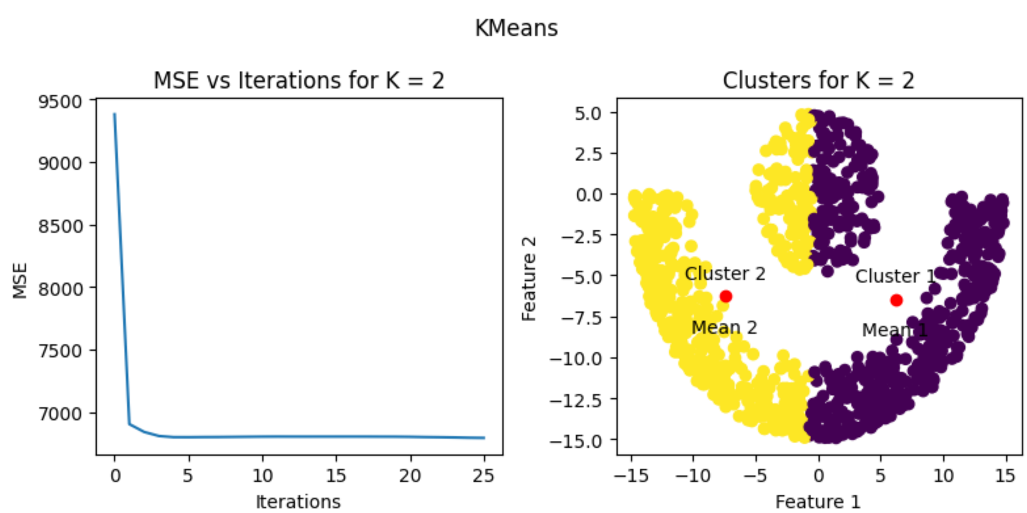

Figure 12: RANDOM run = 1

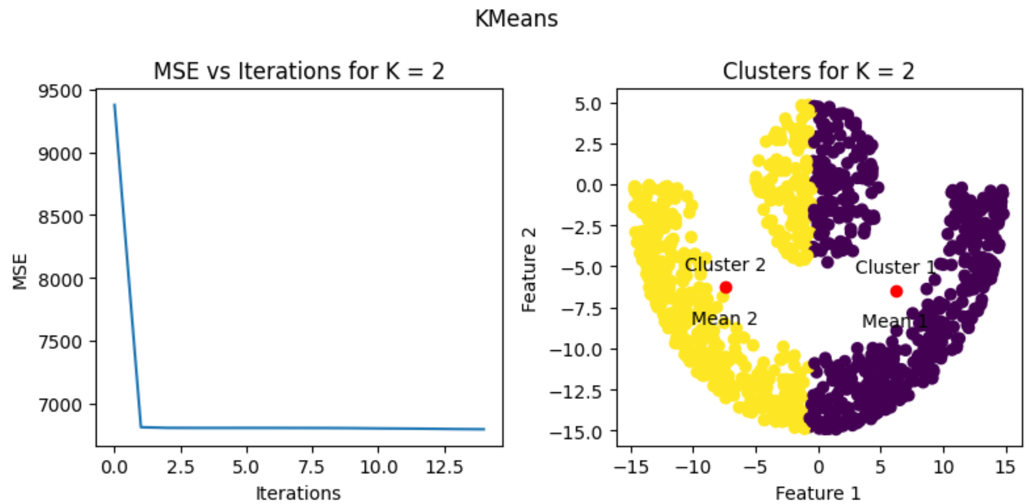

Figure 13: RANDOM run = 2

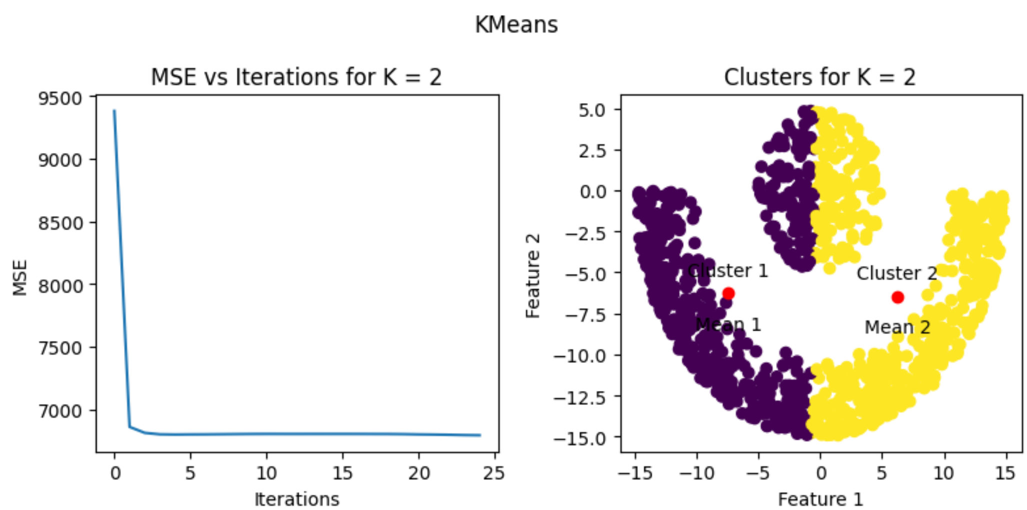

Figure 14: RANDOM run = 3

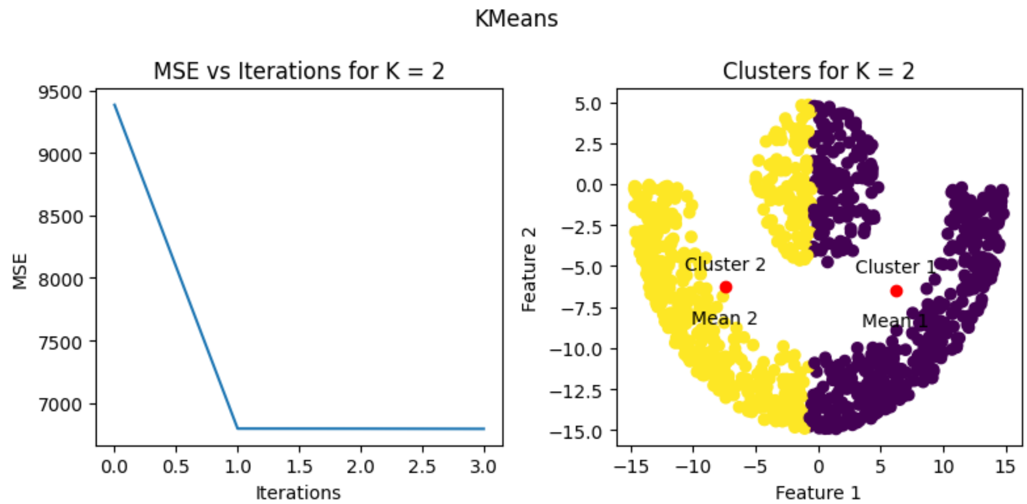

Figure 15: RANDOM run = 4

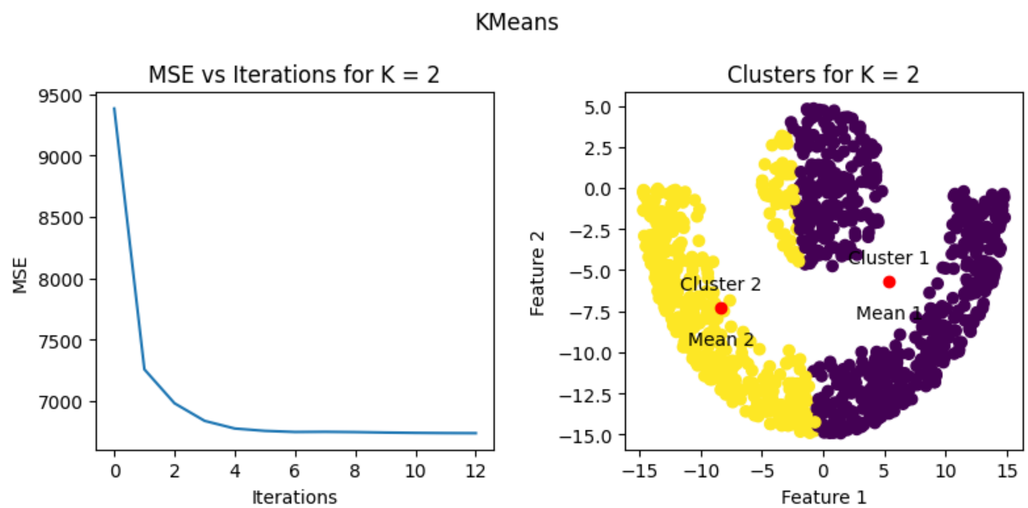

Figure 16: RANDOM run = 5

2.2 Voronoi regions

We obtain the following graphs with voronoi region for $k = 2, 3, 4, 5$

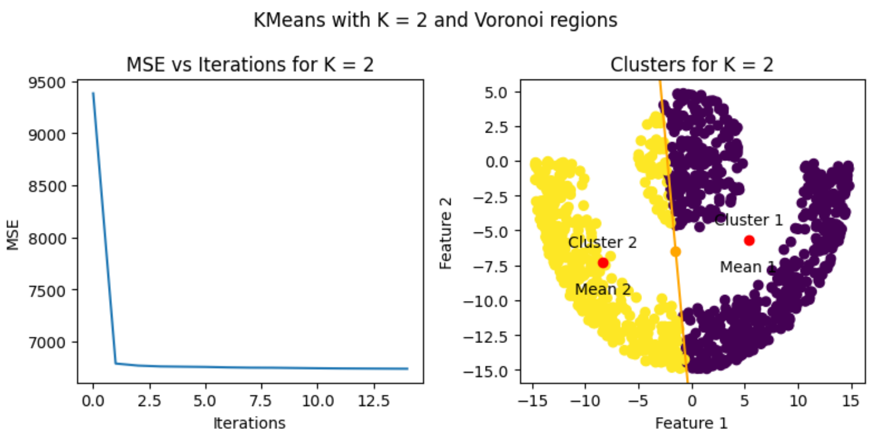

Figure 17: K = 2

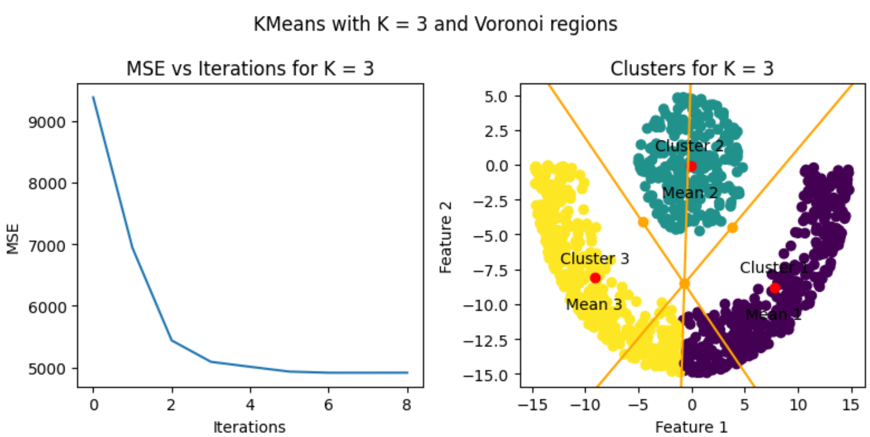

Figure 18: K = 3

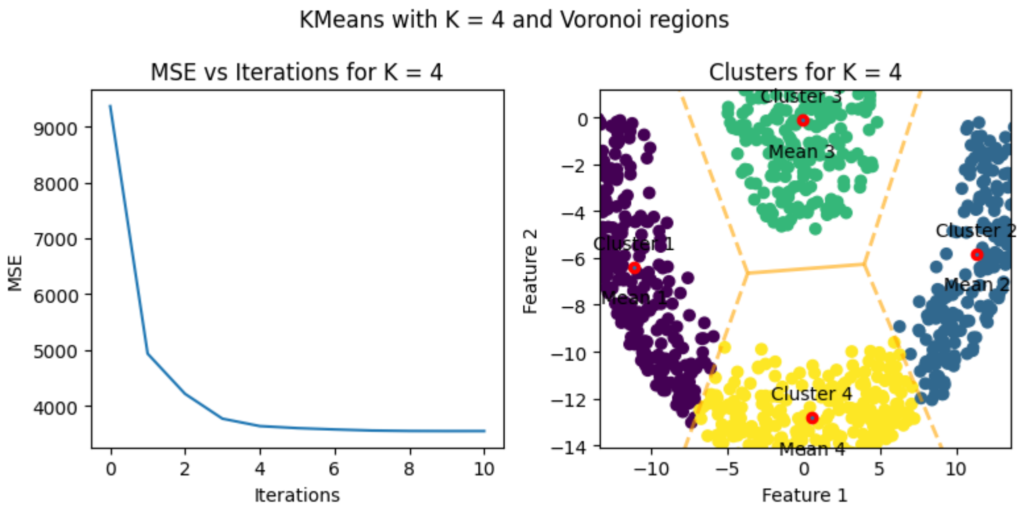

Figure 19: K = 4

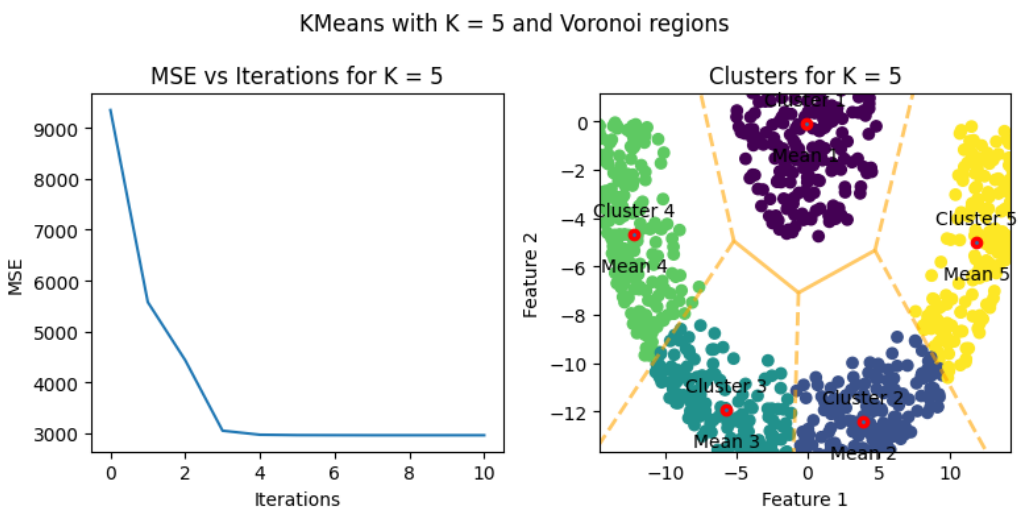

Figure 20: K = 5

2.3 Spectral Clustering

Let us observe the clustering when choosing a polynomial kernel.Therefore we will use the kernel:

$$k(x, y) = (1 + x^T y)^d$$

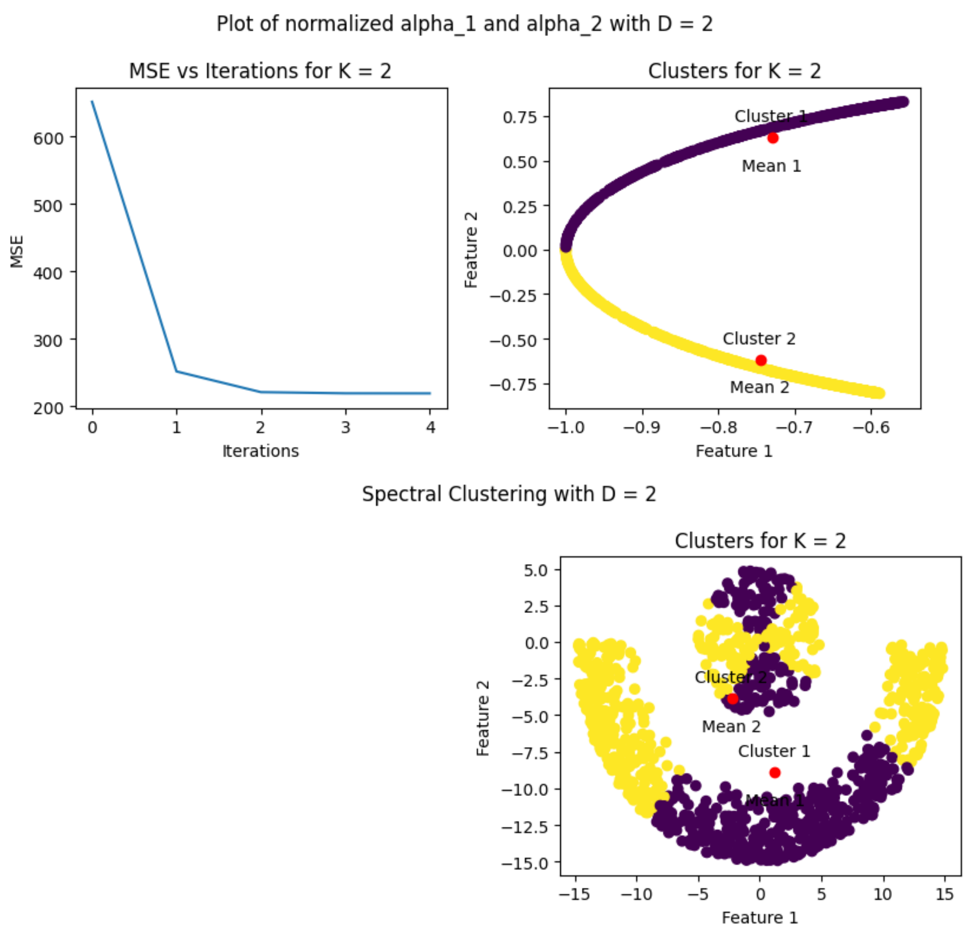

Figure 21: D = 2

Let us observe the clustering when choosing a Gaussian kernel.Therefore we will use the kernel:

$$k(x,y) = e^{\frac{(x-y)^T(x-y)}{2\sigma^2}}$$

Choice of Kernel:
- A gaussian distribution with small σ gives us a poor clustering of data whereas a higher σ gives us good differentiation of the centre accumulation.
- In comparison to the polynomial kernel, the radial basis function gives us a better demarcation of the central accumulation.

2.4 Max arg clustering

- We Observe that the above method produces Clustering that fits reasonably for gaussian function of $\sigma = 25{,}50$. - The method given is analogous to row normalizing H. - So, when H is of the form $H = ZL^{\frac{1}{2}}$, the above will produce same result as normalizing rows of H. - Therefore, for sigma 25, and 50, H is of the form $H = ZL^{\frac{1}{2}}$.

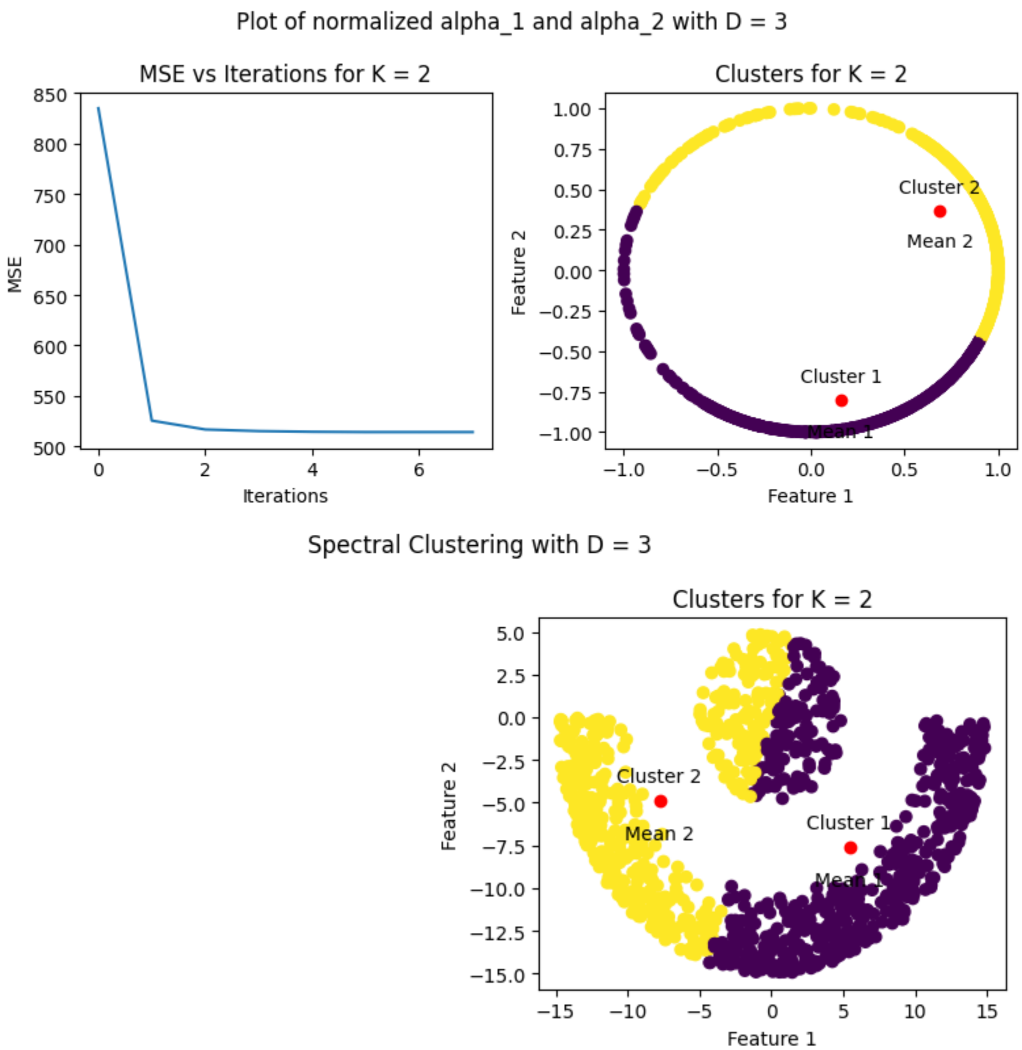
Plot of normalized alpha_1 and alpha_2 with D = 3
MSE vs Iterations for K = 2
MSE
Iterations
Clusters for K = 2
Feature 1
Feature 2
Cluster 2
Mean 2
Cluster 1
Mean 1
Spectral Clustering with D = 3
Clusters for K = 2
Feature 1
Feature 2
Cluster 2
Mean 2
Cluster 1
Mean 1

Figure 22: D = 3

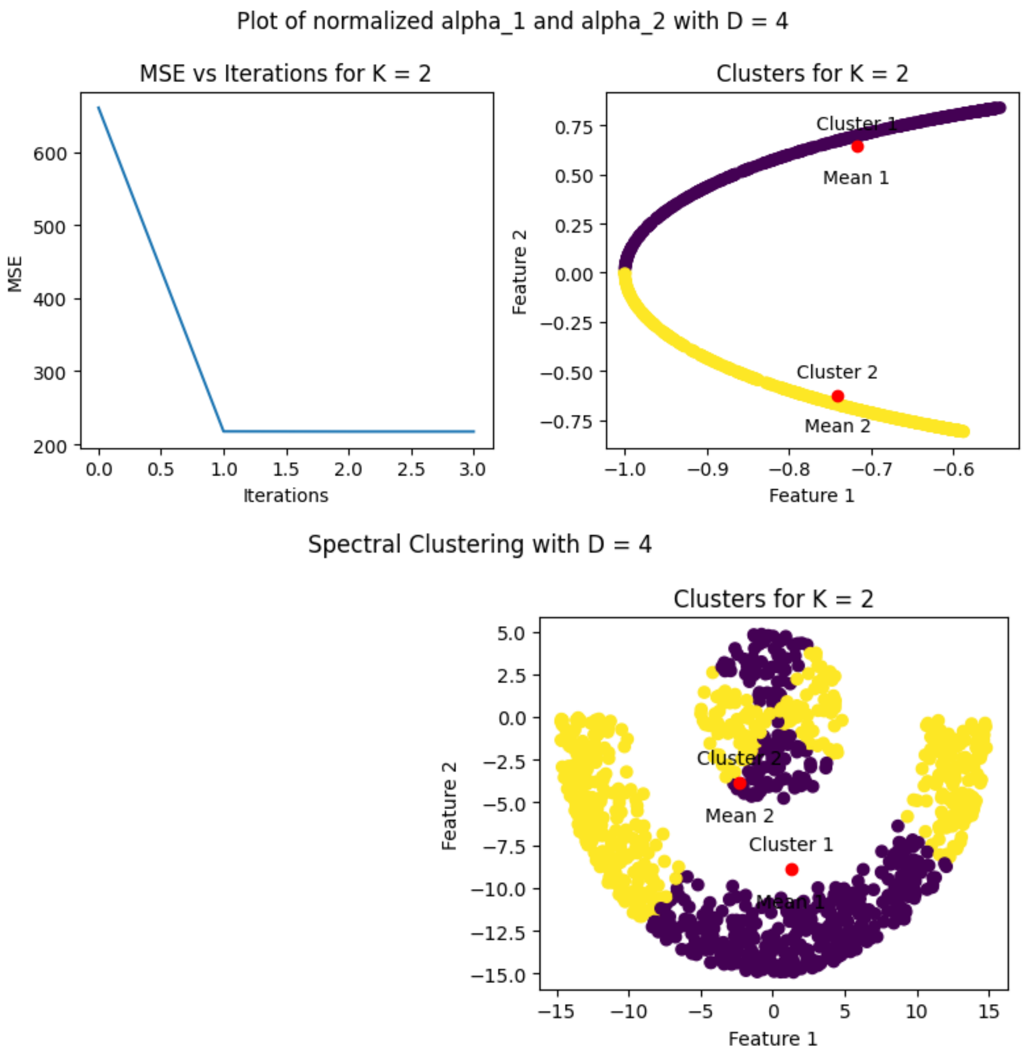
Plot of normalized alpha_1 and alpha_2 with D = 4
MSE vs Iterations for K = 2
MSE
Iterations
Clusters for K = 2
Cluster 1
Mean 1
Cluster 2
Mean 2
Feature 1
Feature 2
Spectral Clustering with D = 4
Clusters for K = 2
Cluster 2
Mean 2
Cluster 1
Mean 1
Feature 1
Feature 2

Figure 23: D = 4

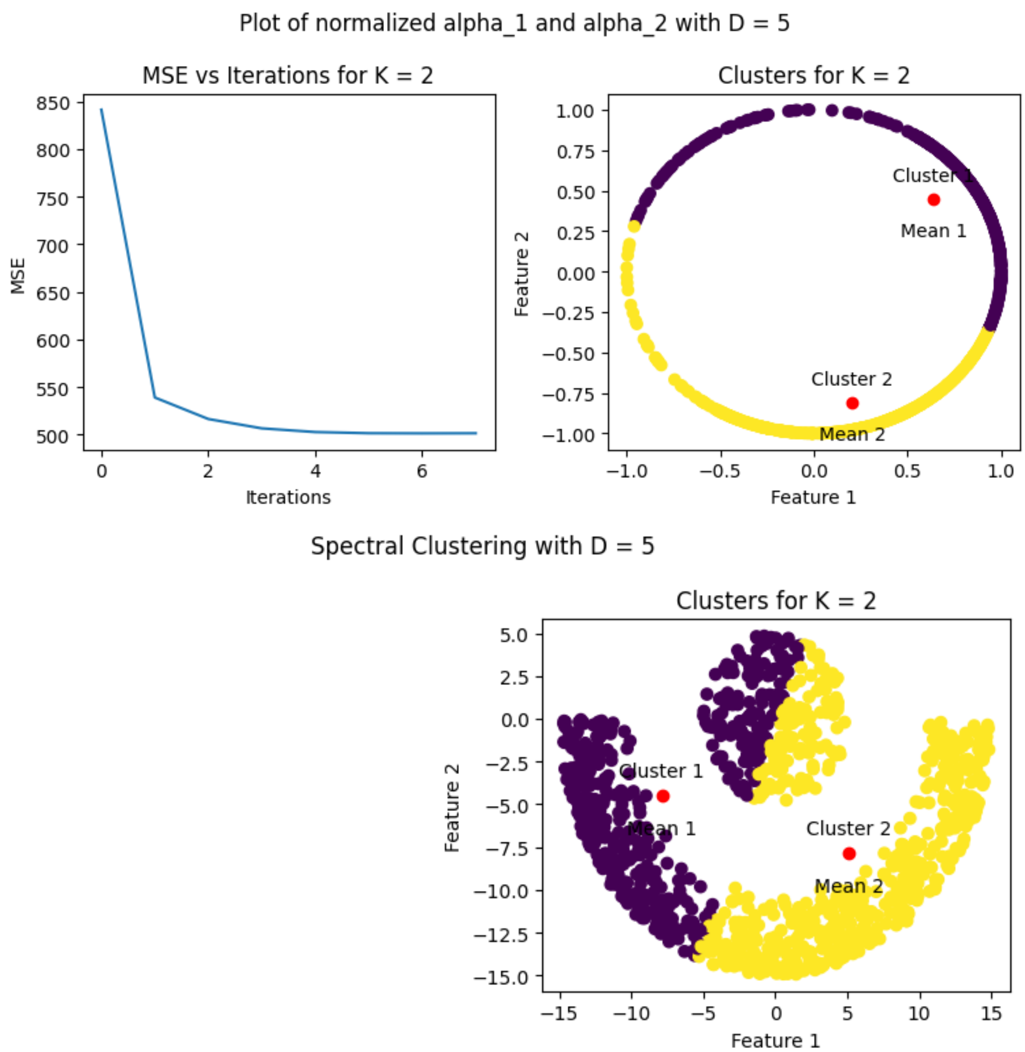
Plot of normalized alpha_1 and alpha_2 with D = 5
MSE vs Iterations for K = 2
MSE
Iterations
Clusters for K = 2
Cluster 1
Mean 1
Cluster 2
Mean 2
Feature 2
Feature 1
Spectral Clustering with D = 5
Clusters for K = 2
Cluster 1
Mean 1
Cluster 2
Mean 2
Feature 2
Feature 1

Figure 24: D = 5

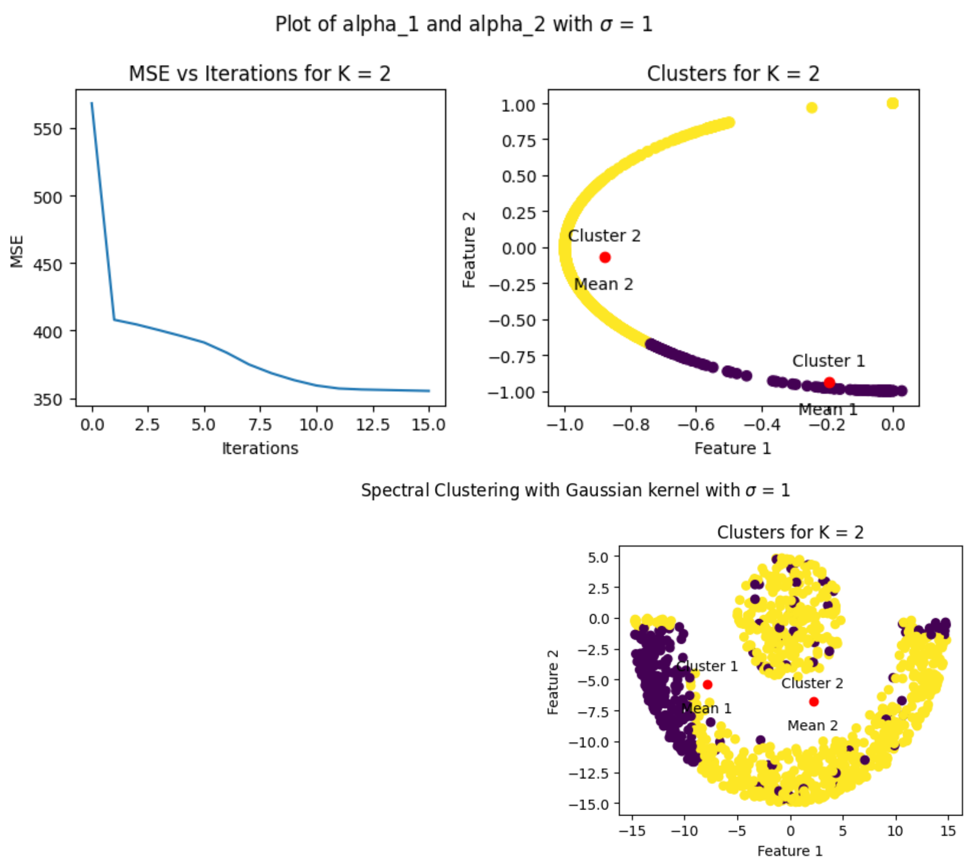

Figure 25: $\sigma = 1$

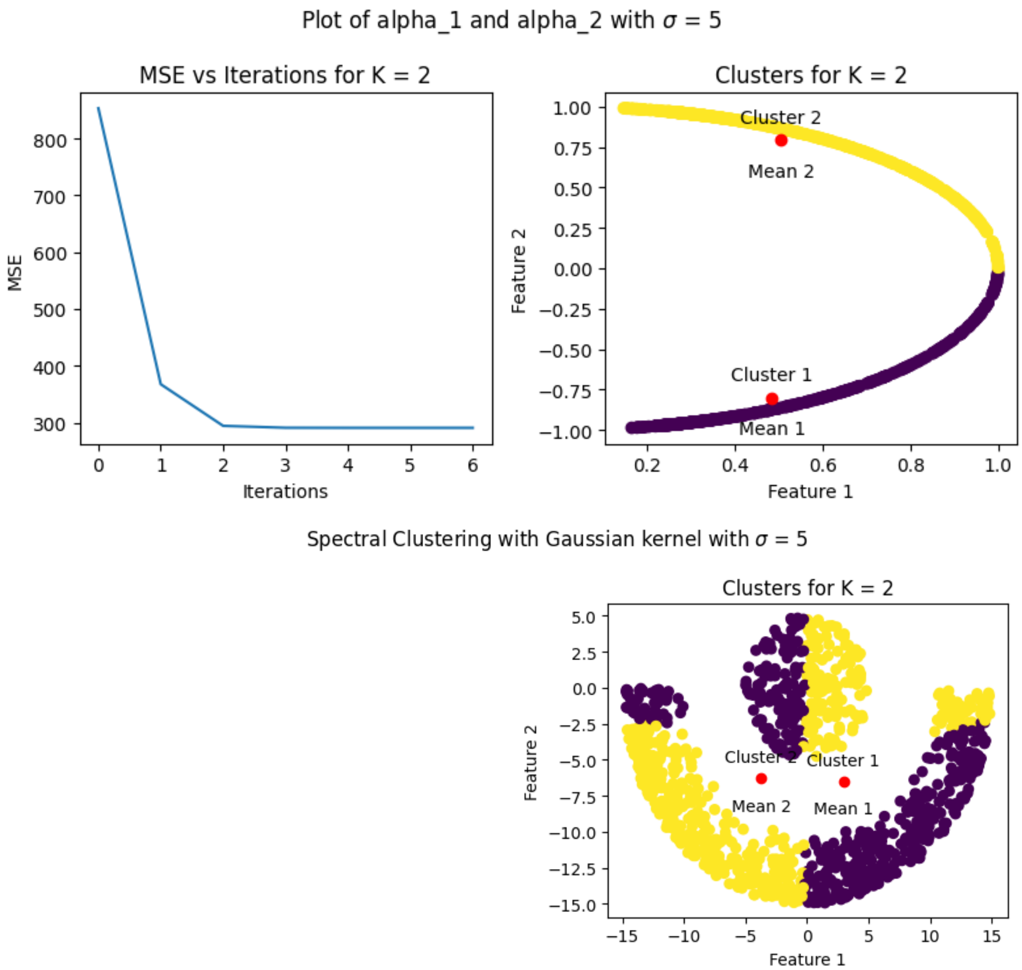

Figure 26: $\sigma = 5$

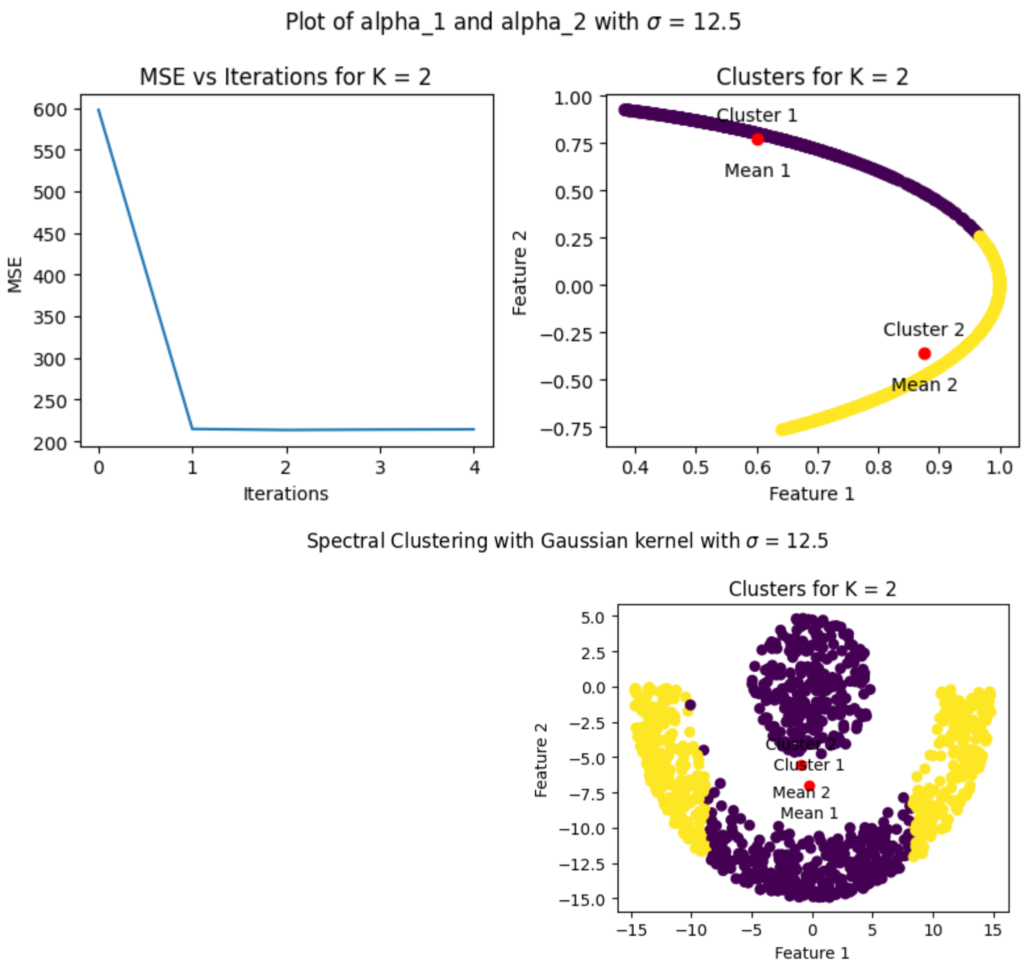

Figure 27: $\sigma = 12.5$

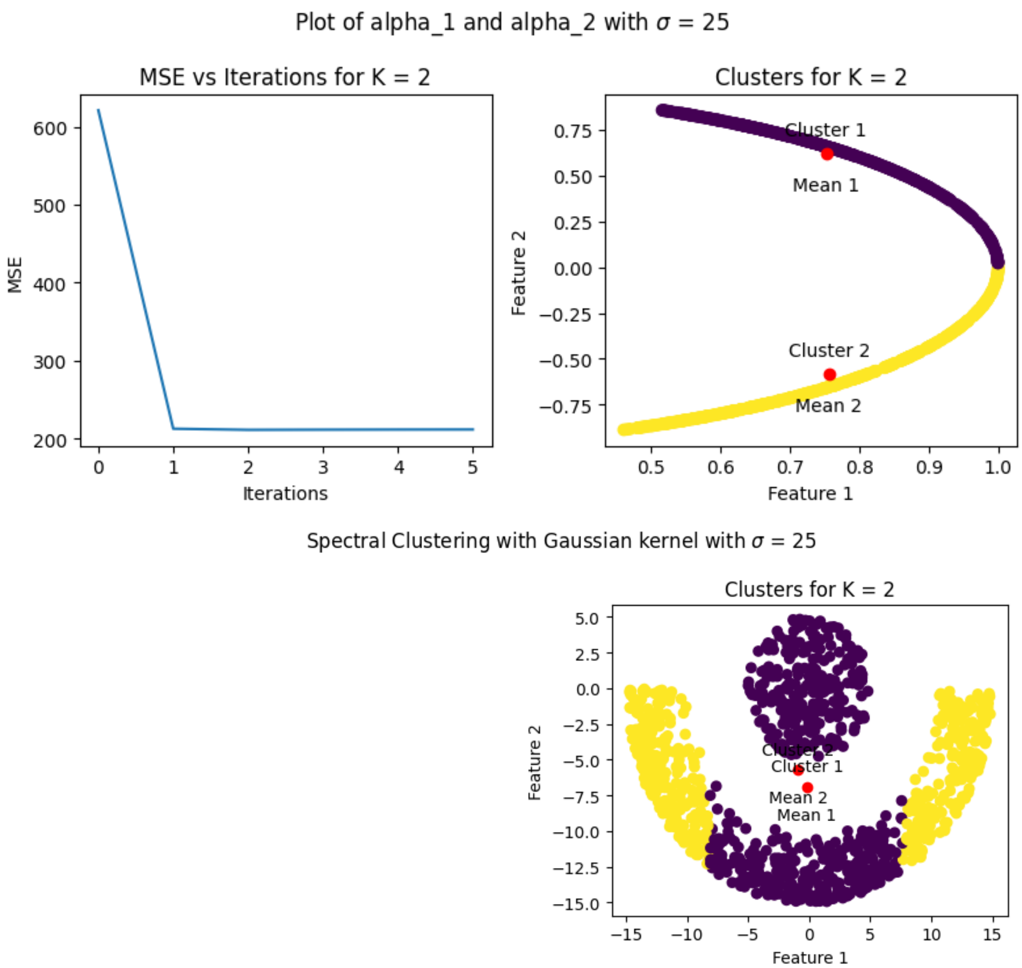

Figure 28: $\sigma = 35$

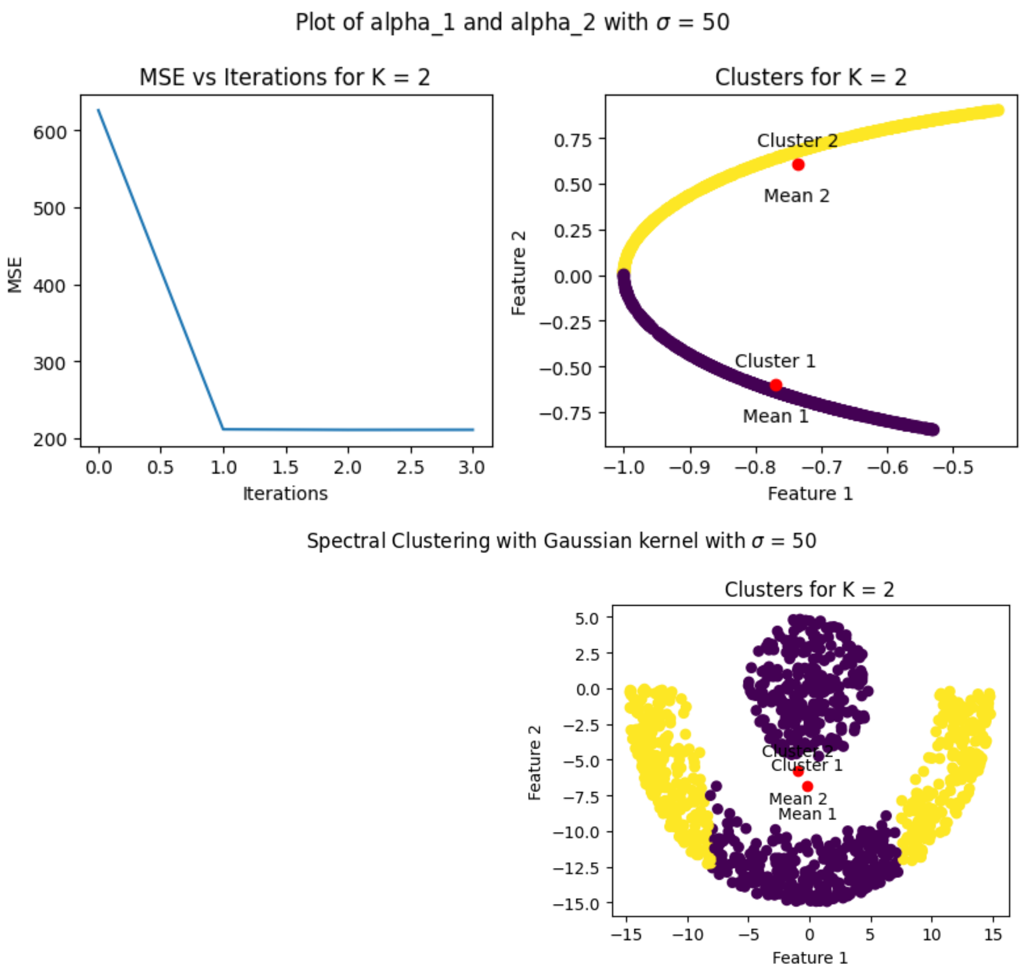

Figure 29: $\sigma = 50$

Maximum argument Clustering with D =2

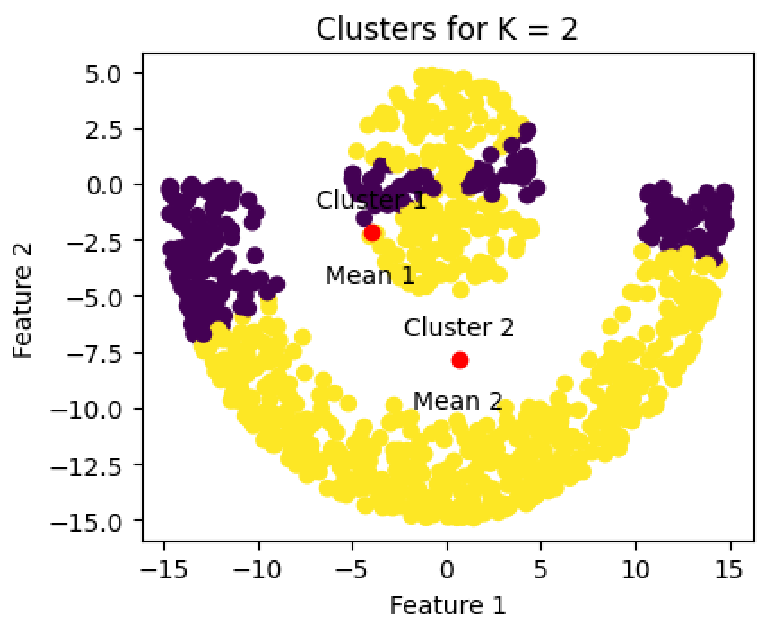

Figure 30: Enter Caption

Maximum argument Clustering with D =3

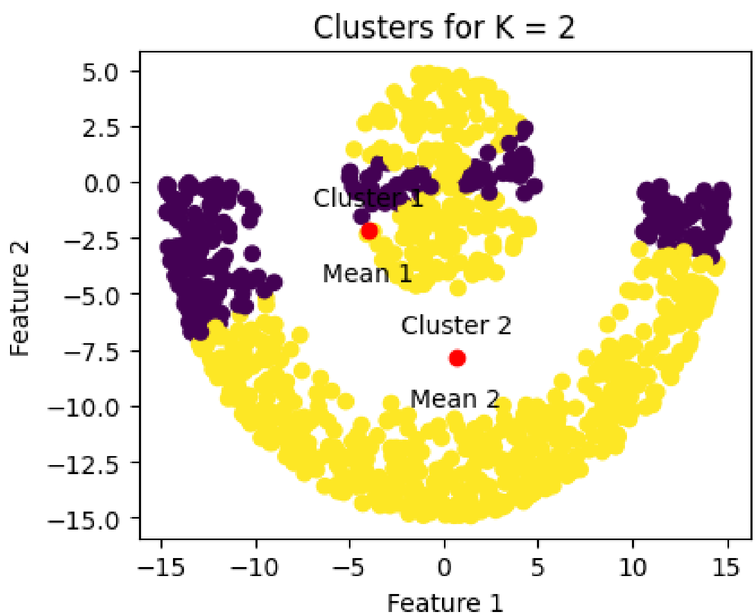

Figure 31: Enter Caption

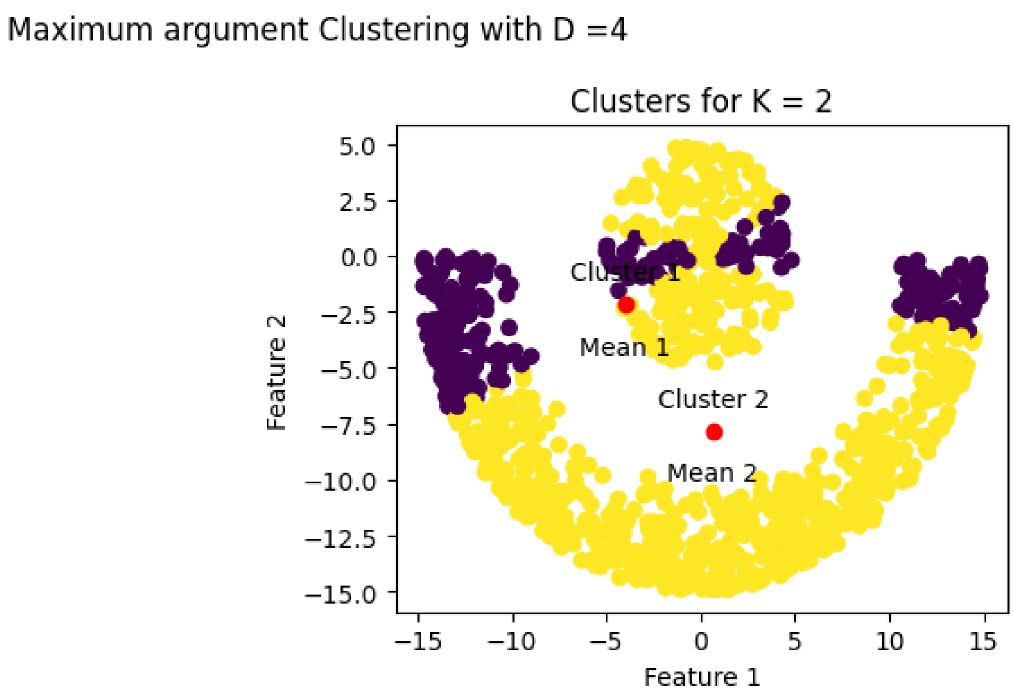

Figure 32: Enter Caption

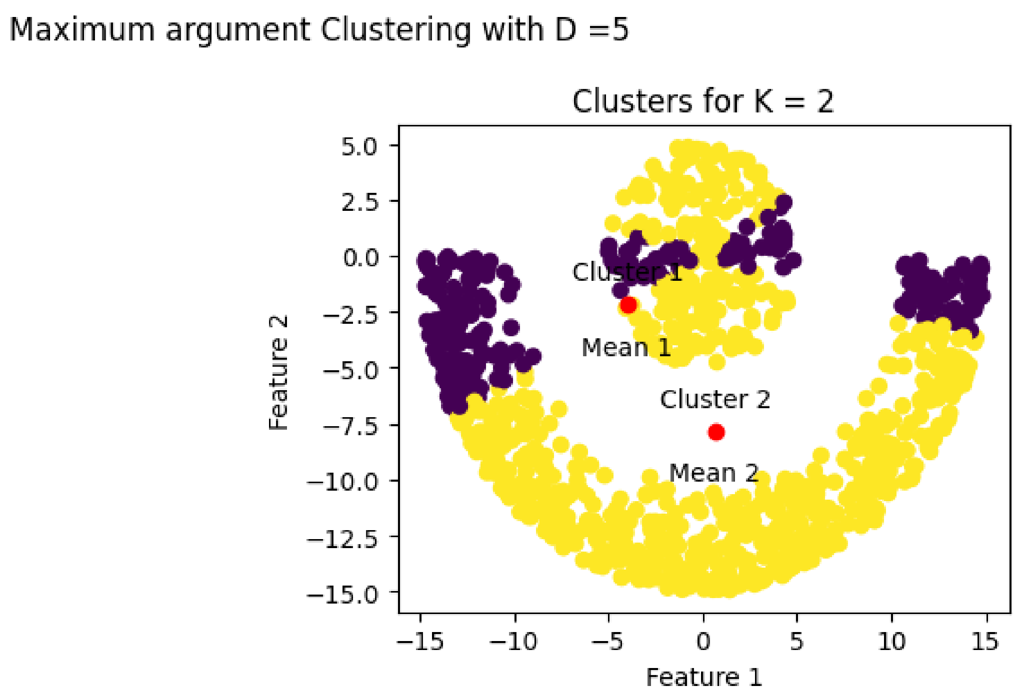

Figure 33: Enter Caption

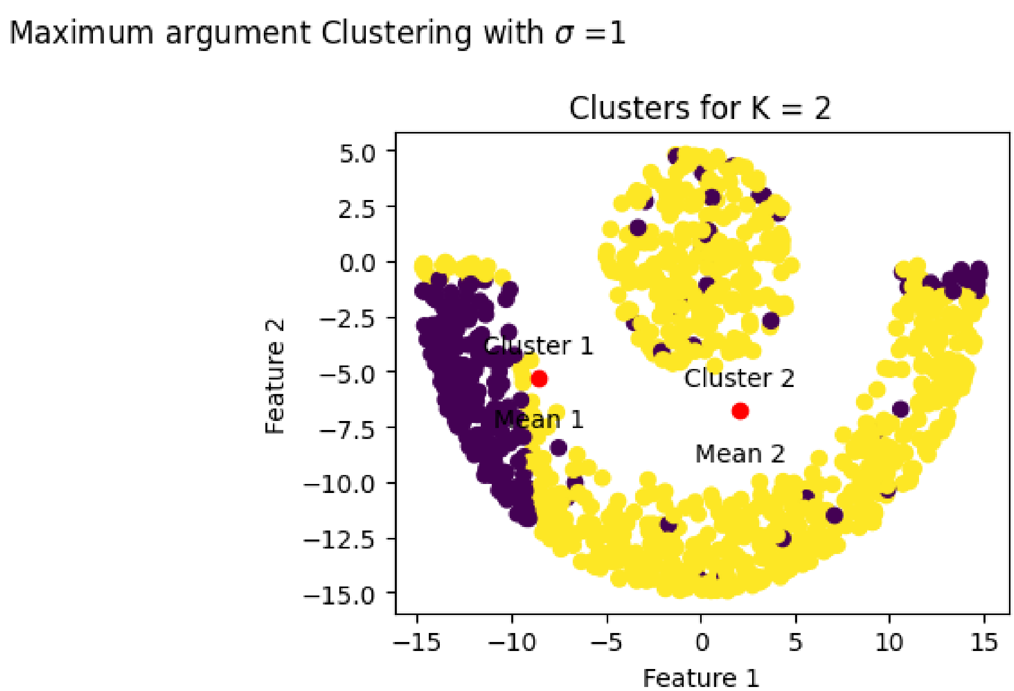

Figure 34: Enter Caption

Maximum argument Clustering with σ =5

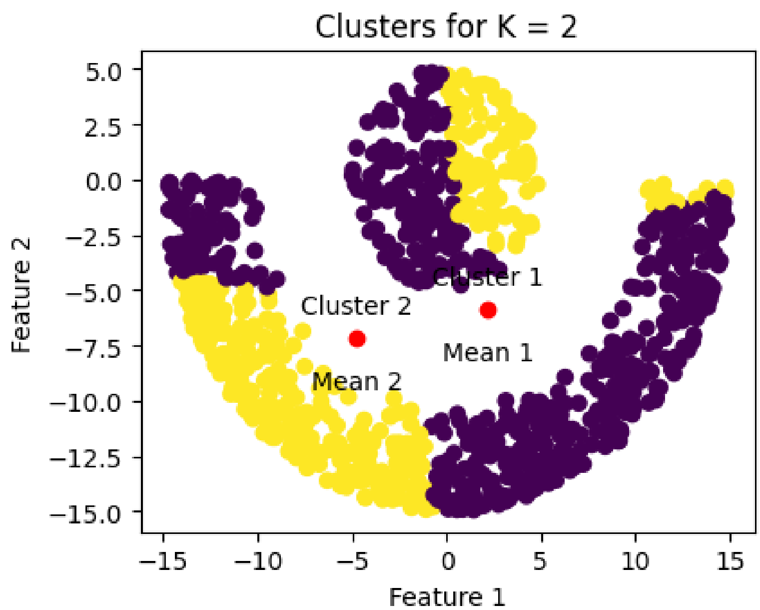

Figure 35: Enter Caption

Maximum argument Clustering with σ =25

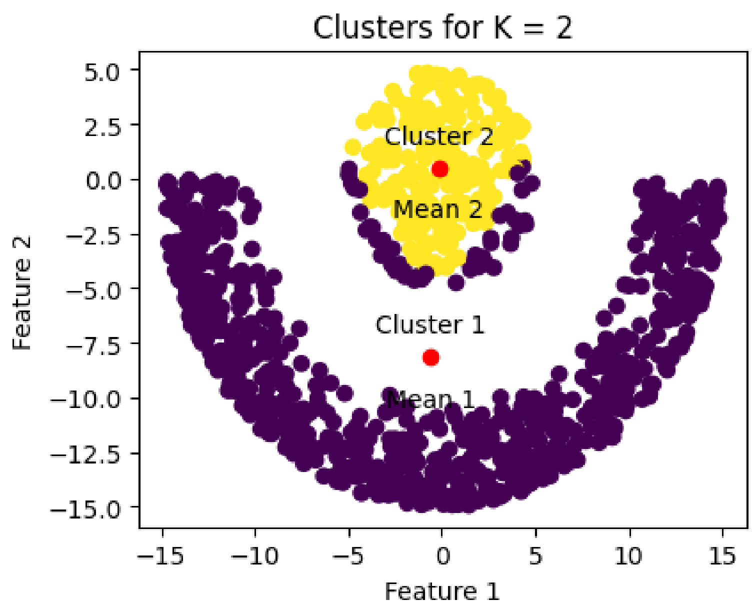

Figure 36: Enter Caption

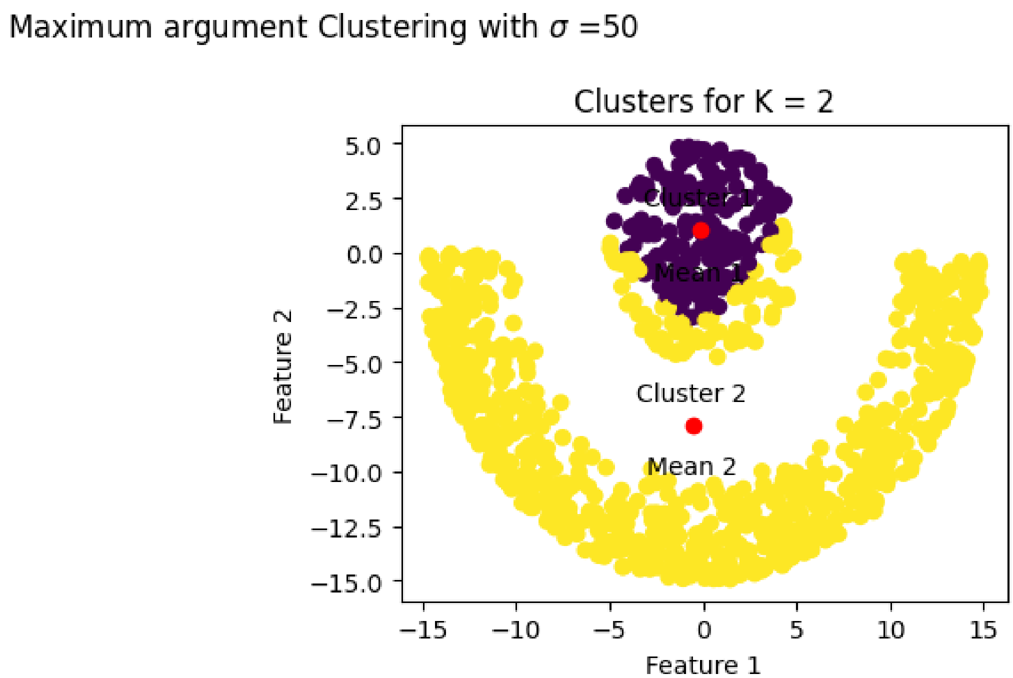

Figure 37: Enter Caption