Q 1. **Handwritten Digits Recognition with k-NN**

(a)

(i) Accuracy for k = 1, 3, 5, 10, 20, 30, 40, 50, 60:

| k value | Accuracy |
|---|---|
| 1 | 91.95% |
| 3 | 92.2% |
| 5 | 92.118% |
| 10 | 91.2% |
| 20 | 89.666% |
| 30 | 88.55% |
| 40 | 87.51% |
| 50 | 86.98% |
| 60 | 86.25% |

(ii)

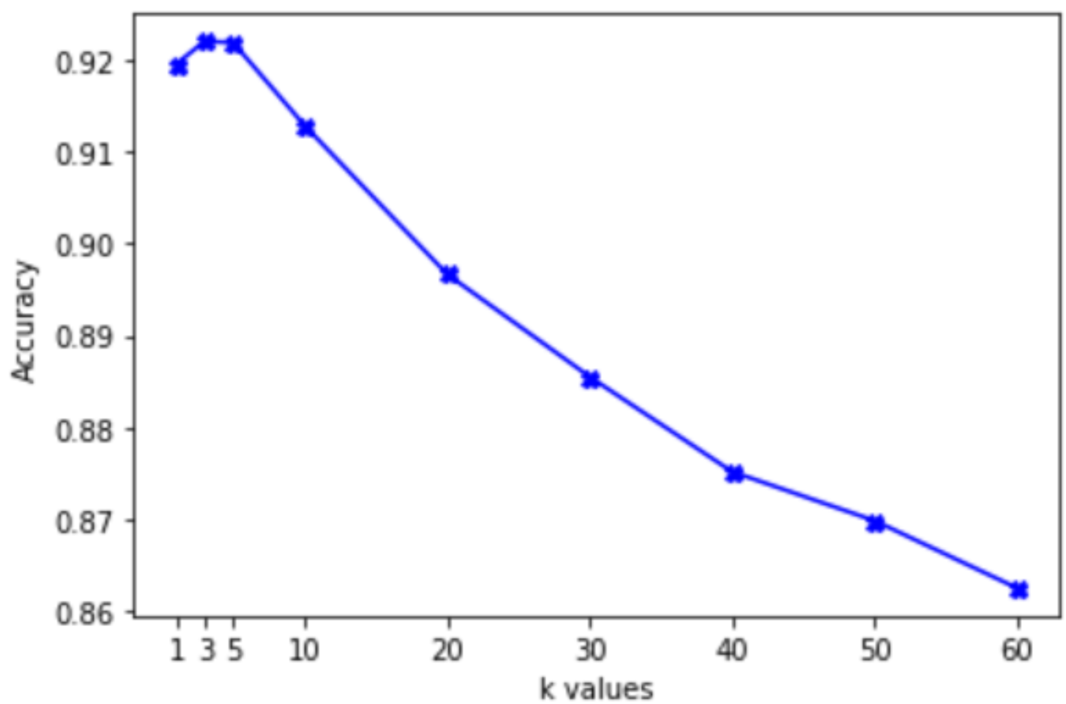
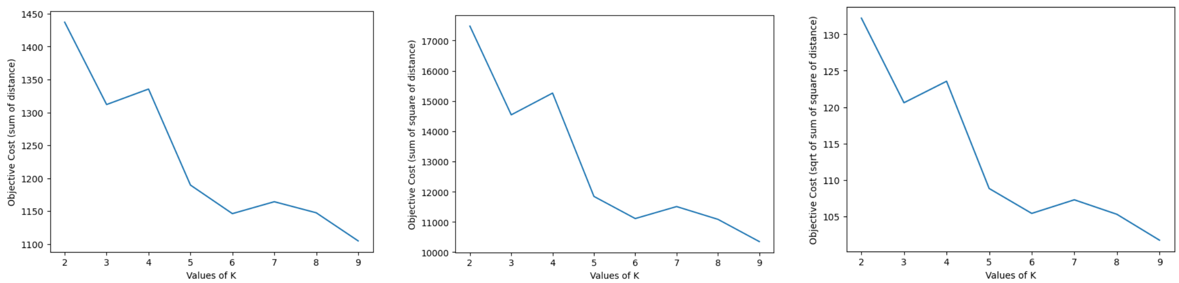



Figure 1: Accuracy v. k-value

(b) Observations:
We see that the accuracy of k-NN peaks at $k = 3$. We also see that there's over-fitting for k-values ¡ 3 due to the influence of noise in the data when considering more neighbors. As the k-value increases, the accuracy of the model decreases, suggesting that larger k-values may result in an overly smoothed decision boundary, leading to a loss in classification performance. The accuracy can be improved if the test data set increases for the same k-value.

## Q 2. K-Means Clustering

(a) Objective function as a function of K




(a) Objective cost (loss function) as Sum of Euclidean distance between each data point and their centroid v. k-values

(b) Objective cost as Sum of Squares of Euclidean distance between each data point and their centroid v. k-values

(c) Objective cost as Square root of Sum of Squares of Euclidean distance between each data point and their centroid v. k-values

Figure 2: Objective costs for different k-values

(b) For K = 2, plot of the points using its first two features. Used two different colors to distinguish the two clusters.

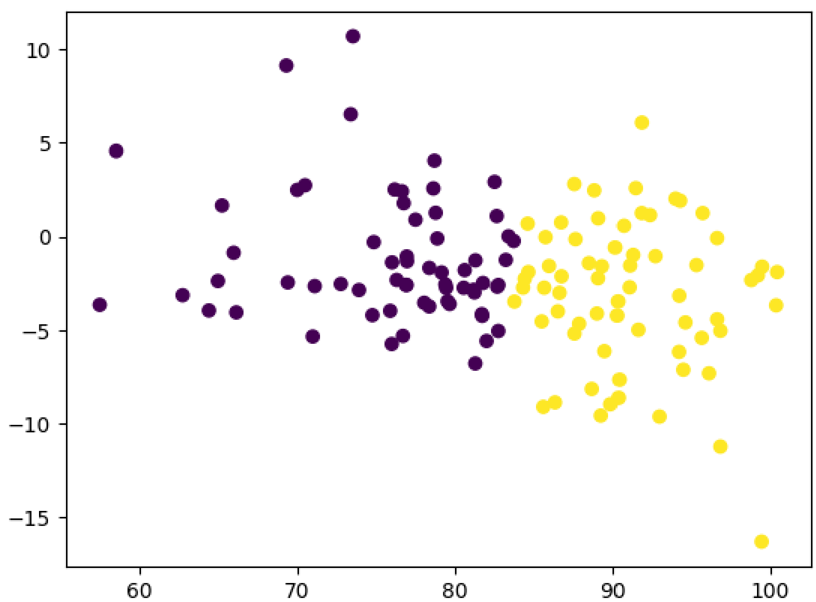


Figure 3: Scatter plot of the K-means clustering for K=2

(c) Observations:
A general observation from the first three figures is that the objective cost or loss function tends to decrease as the k-value increases, although not strictly. A higher number of centroids lead to closer values within clusters and lower loss. However, this may cause overfitting. The final centroids and loss values depend on the initial centroids' random initialization, leading to varying local maxima/minima and steep descents in the plot.

## Q 3. Gaussian Mixture Model (GMM)

(a)

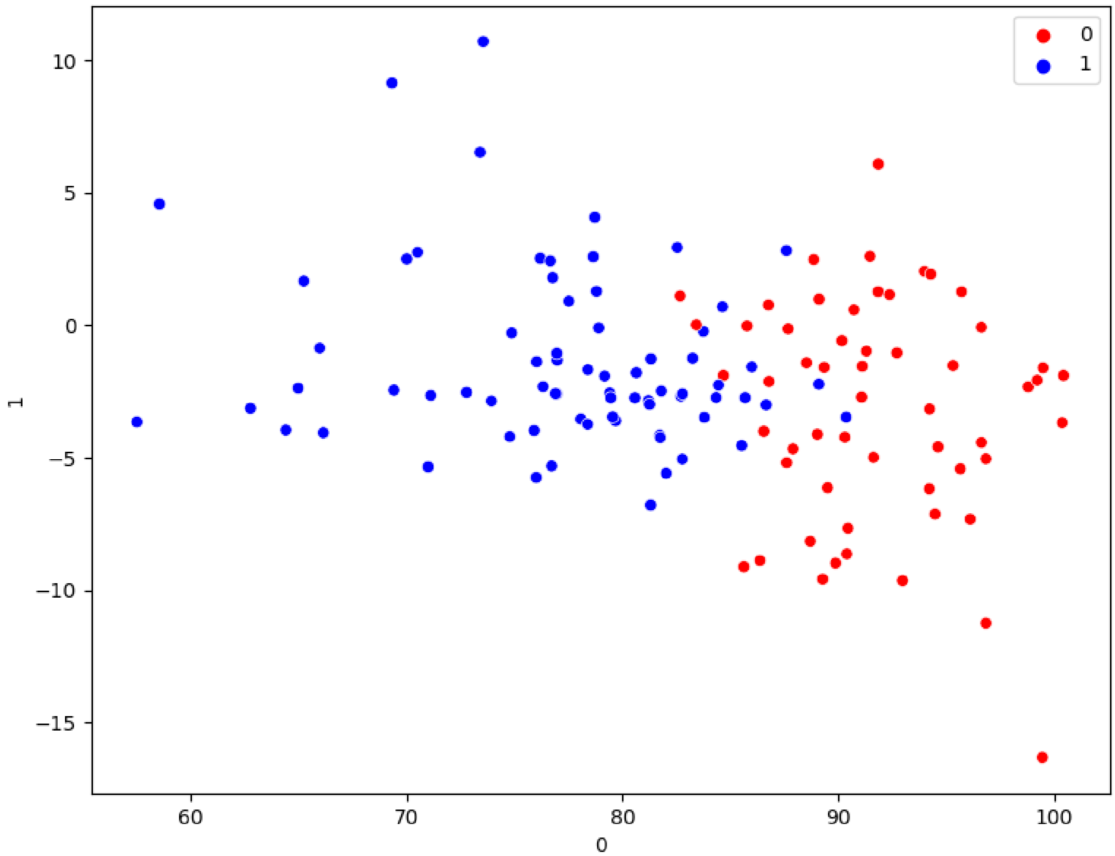



Figure 4: Plot for K = 2

(b) Observations:
We see that GMM and K-means differ in their approaches. K-means produces circular or spherical clusters, while GMM generates elliptical or ellipsoidal clusters. K-means relies on distances from centroids, whereas GMM employs Gaussian distributions, allowing for better control over cluster flexibility. Additionally, GMM's use of covariance makes it superior to K-means, which is based on distance.