Assignment 2 in Machine Learning FMAN45

Eliot Montesino Petrén, el6183mo-s, 990618-9130, LU Faculty of Engineering LTH 19 , eliot.mp99@gmail.com

I. Exercises

Exercise 1

The kernel matrix using the data from the table is, with the possible combinations $\forall 1 \leq i, k \leq 4$:

$$\mathbf{K} = k(x_i, x_j) = \phi(x_i)^T \phi(x_j) = \\ = x_i x_j + (x_i x_j)^2 =$$

$$= \begin{pmatrix} 20 & 6 & 2 & 12 \\ 6 & 2 & 0 & 2 \\ 2 & 0 & 2 & 6 \\ 12 & 2 & 6 & 20 \end{pmatrix} \tag{1}$$

Exercise 2

The maximization optimization problem can be rewritten when $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4$.

$$\max_{\alpha} \left(4\alpha - \frac{\alpha^2}{2} \sum_{i,j=1}^{4} y_i y_j k(x_i, x_j) \right) \tag{2}$$

$$\text{Subject to } \alpha \sum_{i=1}^{4} y_i = 0, \alpha \geq 0.$$

Take the kernel values from (1) into (2) yielding (3)

$$\sum_{i,j=1}^{4} y_i y_j k(x_i, x_j) = 68 - 32 = 36 \tag{3}$$

Insert (3) into (2)

$$\max_{\alpha}(4\alpha - 18\alpha^2) = \max_{\alpha} 2\alpha(2 - 9\alpha) \tag{4}$$

Eq (4) is a concave function shown by the second derivative being strictly negative.

$$(2\alpha(2 - 9\alpha))''_{\alpha\alpha} = ... = -36 < 0 \tag{5}$$

Concanve functions have maximum stationary points, maximum is found at.

$$(2\alpha(2 - 9\alpha))'_{\alpha} = 4 - 36\alpha = 0 \Leftrightarrow \alpha = \frac{1}{9} \tag{6}$$

Marking the end of this exercise.

Exercise 3

The task is to reduce the classifier function to the simplest possible form and we expect a simple polynomial in x. Firstly, the classifier function can be written as.

$$g(x) = \sum_{j=1}^{4} \alpha_j y_j k(x_j, x) + b = \frac{1}{9} \sum_{j=1}^{4} y_j k(x_j, x) + b$$

Expand the sum and continue to fill in value obtained from earlier the table denoted (2) in instructions.

$$g(x) = \frac{1}{9} \left(-2x + 4x^2 - x + x^2 - x + x^2 + 2x + 4x^2 \right) + b \\ = \frac{2}{3}x^2 + b \tag{7}$$

We still want to get rid of b. Another expression from instructions denoted (6) gives an equation where we solve for b.

$$1 = y_s \left(\sum_{j=1}^{4} \alpha_j y_j k(x_j, x_s) + b \right) = y_s \left(\frac{2}{3}x^2 + b \right) = \tag{8}$$

$$= 1 \left(\frac{2}{3}(-2)^2 + b \right) = \frac{8}{3} + b \tag{9}$$

$$\Leftrightarrow \quad b = -\frac{5}{3} \tag{10}$$

Finally obtaining an expression only variable in x.

$$g(x) = \frac{1}{3}(2x^2 - 5) \tag{11}$$

Exercise 4

Given the new dataset:

i	x_i	y_i
1	-3	$+1$
2	-2	$+1$
3	-1	-1
4	0	-1
5	1	-1
6	2	$+1$
7	4	$+1$

We need to determine the solution $g(x)$ of the nonlinear kernel SVM with hard constraints on this dataset. We will use the same kernel $k(x, y)$ as in previous exercises, $k(x, y) = x \cdot y + (x \cdot y)^2$.

First, we recall the classifier function derived from the previous dataset:

$$g(x) = \frac{1}{3}(2x^2 - 5)$$

For the new dataset, we need to check if the support vectors have changed. Support vectors are the data points that lie on

the margin boundaries and significantly impact the decision boundary.

From the previous dataset, the support vectors were $x = -2, -1, 1, 2$. Let's examine if adding the new points $x = -3, 0, 4$ affects the support vectors.

Analysis of the New Points:

- Point $x = -3$: This point lies further away from the origin compared to $x = -2$. However, since its label is the same as $x = -2$ and it is farther away, it does not affect the decision boundary.

- Point $x = 0$: This point lies at the origin and belongs to the negative class. Since it is closer to the origin than the other negative class points, it could potentially affect the support vectors.

- Point $x = 4$: This point lies further away from the origin compared to $x = 2$. Its positive label is the same as $x = 2$ and it being farther away means it does not affect the decision boundary.

Re-evaluating the Support Vectors: We will now determine if $x = 0$ should be a new support vector. We calculate the margin for $x = 0$:

$$k(x_i, 0) = 0 \quad \text{for all } x_i$$

The classifier function at $x = 0$:

$$g(0) = \frac{1}{3}(2 \cdot 0^2 - 5) = -\frac{5}{3}$$

Since $y = -1$ for $x = 0$, and $g(0) = -\frac{5}{3}$, which is not on the margin boundary ($y_i \cdot g(x_i) \neq 1$), $x = 0$ does not become a support vector.

Conclusion: Since the newly added points $x = -3, 0, 4$ do not change the support vectors from the previous dataset, the classifier function remains unchanged.

Thus, the solution for the nonlinear kernel SVM with hard constraints on the new dataset is:

$$g(x) = \frac{1}{3}(2x^2 - 5)$$

Marking the end of this exercise.

Exercise 5

Given that the primal formulation of the linear soft margin classifier is given by

$$\min_{w,b,\xi} \left(\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i \right) \tag{12}$$

$$\text{Subject to } y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \quad \forall i \tag{13}$$

$$\xi_i \geq 0, \quad \forall i \tag{14}$$

I am to show that the Lagrangian dual problem is given by

$$\begin{aligned} \text{maximize} \quad & \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j \\ \text{subject to} \quad & 0 \leq \alpha_i \leq C, \quad \forall i \\ & \sum_{i=1}^{n} \alpha_i y_i = 0, \quad \forall i \end{aligned}$$

The corresponding Lagrangian function is.

$$L(w, b, \xi) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i - \tag{15}$$

$$- \sum_{i=1}^{n} \alpha_i(s_i - 1 + y_i(\mathbf{w}^T x_i + b)) - \sum_{i=1}^{n} \lambda_i\xi_i \tag{16}$$

$$\text{where } (\alpha_i, \lambda_i \geq 0).$$

Minimizing the Lagrangian we state the first order conditions (FOC) basically corresponding to taking the gradient of ∇L.

$$\frac{\partial L}{\partial w} = 0 \leftrightarrow w - \sum_{i=1}^{n} \alpha_i y_i x_i = 0 \leftrightarrow w = \sum_{i=1}^{n} \alpha_i y_i x_i \tag{17}$$

$$\frac{\partial L}{\partial b} = 0 \leftrightarrow \sum_{i=1}^{n} \alpha_i y_i = 0 \tag{18}$$

$$\frac{\partial L}{\partial \xi} = 0 \leftrightarrow C - \alpha_i - \lambda_i = 0 \leftrightarrow \lambda_i = C - \alpha_i \tag{19}$$

Simplify the eq divided in (15) and (16) with eq (17) that gives the Lagrangian dual.

$$\max_{\alpha_1,\ldots,\alpha_n} L = \frac{1}{2}\left\|\sum_{i=1}^{n} \alpha_i y_i x_i\right\|^2 + C\sum_{i=1}^{n} \xi_i -$$

$$- \sum_{i=1}^{n} \alpha_i(\xi_i - 1 + u_i(\sum_{i=1}^{n} \alpha_i y_i x_i)^T x_i + b)) - \sum_{i=1}^{n} \lambda_i\xi_i = \tag{20}$$

$$= -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j x_i^T x_j +$$

$$+ C\sum_{i=1}^{n} \xi_i(C - \alpha_i - \lambda_i) + \sum_{i=1}^{n} \alpha_i - \sum_{i=1}^{n} \alpha_i y_i b \tag{21}$$

$$= \max_{\alpha_1,\ldots,\alpha_n} \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j x_i^T x_j \tag{22}$$

Given that $\alpha_i, \lambda_i \geq 0$ and $\lambda_i = C - \alpha_i$, the constraint is.

$$0 \leq \alpha_i \leq C \text{ for all } i \tag{23}$$

Thus marking the end of this exercise.

Exercise 6

The exercise is to show that support vectors

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) < 1 \tag{24}$$

have coefficient $\alpha_i = C$. The complementary slackness is

$$\alpha_i \left(y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 + \xi_i\right) = 0 \tag{25}$$

$$\lambda_i\xi_i = 0 \tag{26}$$

Using (19) rewrite (26)

$$(C - \alpha_i)\xi_i = 0 \tag{27}$$

conditioned on $\alpha_i = C$ and $\xi_i > 0$. Eq (25) holds if

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 + \xi_i = 0 \iff \xi_i = 1 - y_i(\mathbf{w}^T\mathbf{x}_i + b)$$

If $\xi_i > 0$ we get.

$$1 - y_i(\mathbf{w}^T\mathbf{x}_i + b) > 0 \iff y_i(\mathbf{w}^T\mathbf{x}_i + b) > 1 \qquad (28)$$

Marking the end of this exercise

Exercise 7

To visualize the high-dimensional MNIST data, Principal Component Analysis (PCA) was used for dimensionality reduction. Before using PCA the data was made sure to be zero mean. The training data was projected onto its first two principal components, as illustrated in Figure 1. The scatter plot clearly demonstrates the separation between the two classes ('0' and '1'), with minimal overlap and well executed separation.

Exercise 8

The K-means clustering algorithm was applied to the MNIST training data for K=2 and K=5 clusters. Figures 2 and 3 present the clustering results, visualized using the first two principal components from the PCA performed in Exercise 7. For K=2 (Figure 2), the clusters appear to correspond well with the two digit classes, suggesting that the algorithm has effectively captured the inherent structure of the data, there is a little overlap only.

However, for K=5 (Figure 3), the clusters do not make any sense any longer. This is obviously because of the unreasonable choice of K = 5 for this dataset. This showcases the necessity of choosing an appropiate K (or more interestingly implementing an automatic method that chooses an appropiate K).

Exercise 9

Figures 4 and 5 display the cluster centroids as 28x28 pixel images for K=2 and K=5, respectively. These visualizations provide insight into the average characteristics of each cluster. For K=2 (Figure 4), the centroids clearly resemble the digits '0' and '1', indicating that the clustering algorithm has effectively captured the two main classes in the dataset. This aligns with the clear separation observed in the PCA visualization from Exercise 8. In the case of K=5 (Figure 5), the centroids reveal representations with more variation. All of them resemble distinct digits still. It is difficult to tell if the algorithm has identified subgroups within the main classes representing different styles or variations of writing the digits '0' and '1', maybe cluster 3 is different because the 1 is diagonal. The main takeaway is that there are too many clusters and too few classes.

Exercise 10

See table 1

Exercise 11

Best way to approach this was to just try many different K and plot the results for it and find the minimizing K. See figure 6. The minimizer K was actually given from K = 8, but they are fairly equal for values 8 to 10. Best misclassification rate (validation data): 0.19% Corresponding training misclassification rate: 0.3%. As K increases, the clusters become more homogeneous, meaning each cluster contains more similar data points. This increased homogeneity allows for better classification and lower misclassification rates for both training and validation data sets as K grows.

Exercise 12

See table 2. The SVM classifier with Matlab was applied on the data. The SVM classifier is clearly outperforming the K-means classifier used earlier.

Exercise 13 ***and*** *Exercise 14*

See table 3. In my (short) analysis of the Gaussian kernel SVM choosing beta, I found that the optimal beta value was 10, resulting in impressively low misclassification rates of 0.02% for training data and 0.05% for test data. These results also outperform the linear SVM, demonstrating the Gaussian kernel's effectiveness for this MNIST binary classification task. I chose a logarithmic scale for testing different beta values from 0.1 to 10, but with only five points due to time constraints on this report and code running slow. A logarithmic scale is suitable I thought for exploring parameters like beta, as it allows for efficient coverage of multiple orders of magnitude, and because why not, let's try it.

However, the near-perfect accuracy on time constrained testing of beta values suggests a risk of overfitting, and performance on completely new data might not be quite as exceptional. Overfitting could still be true even when applying on validation data, in this case we are using validation data to get an understanding what parameters we should choose. Given new data, then the lack of regularization might show. To avoid this one can use another data set that should basically never be touched until the very last point, given a large enough data set of course.

II. References

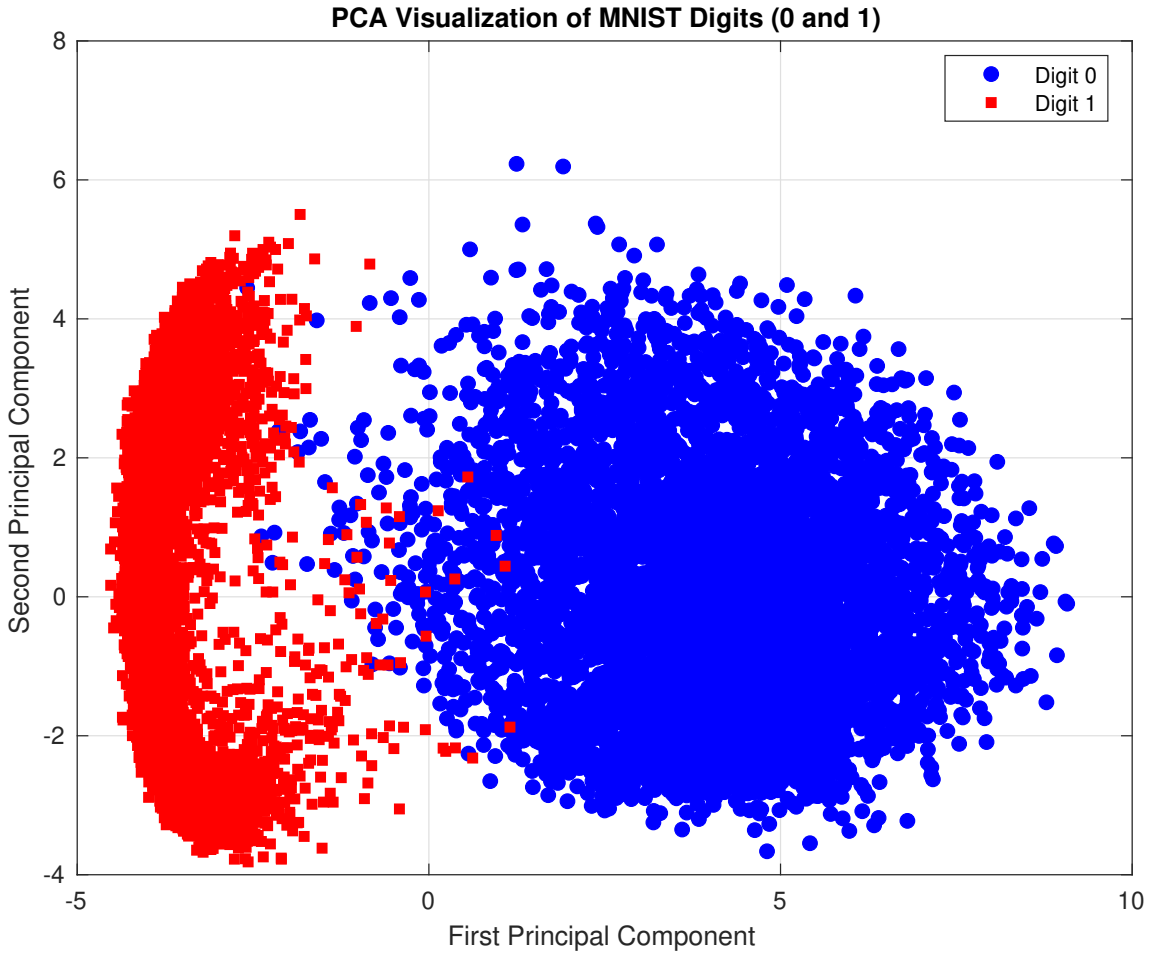

Fig. 1. Visualization of MNIST training data projected onto its first two principal components. The data points are color-coded and marked differently for each class ('0' and '1').

III. Appendix

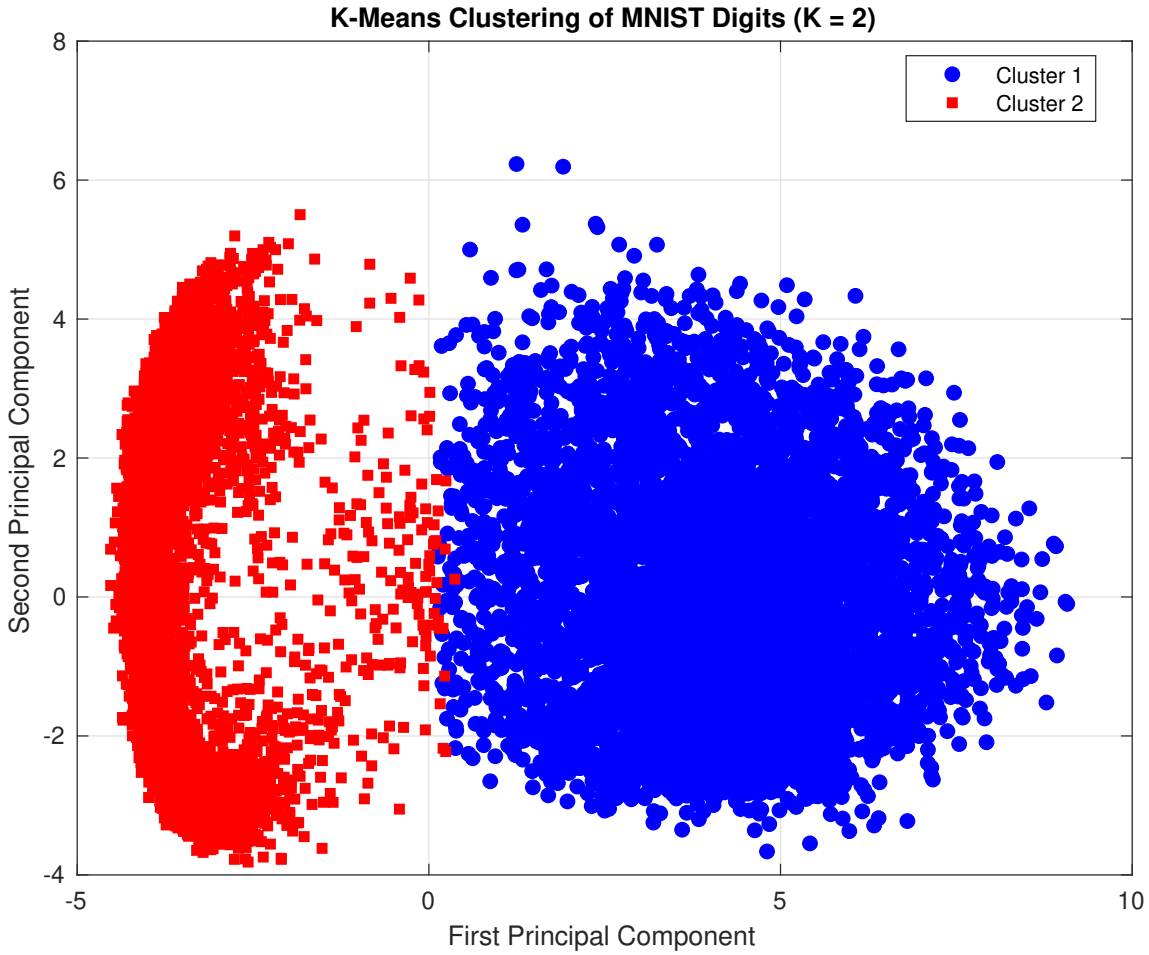

Fig. 2. K-means clustering results for K=2, visualized using the first two principal components. Each cluster is represented by a distinct color and marker.

TABLE I
K-MEANS CLASSIFICATION RESULTS

Training data	**Cluster**	**# '0'**	**# '1'**	**Assigned to class**	**# misclassified**
	1	112	6736	1	112
	2	5811	6	0	6
$N_{\text{train}} = 12665$				**Sum misclassified:**	118
				Misclassification rate (%):	0.93
Testing data	**Cluster**	**# '0'**	**# '1'**	**Assigned to class**	**# misclassified**
	1	12	1135	1	12
	2	968	0	0	0
$N_{\text{test}} = 2115$				**Sum misclassified:**	12
				Misclassification rate (%):	0.57

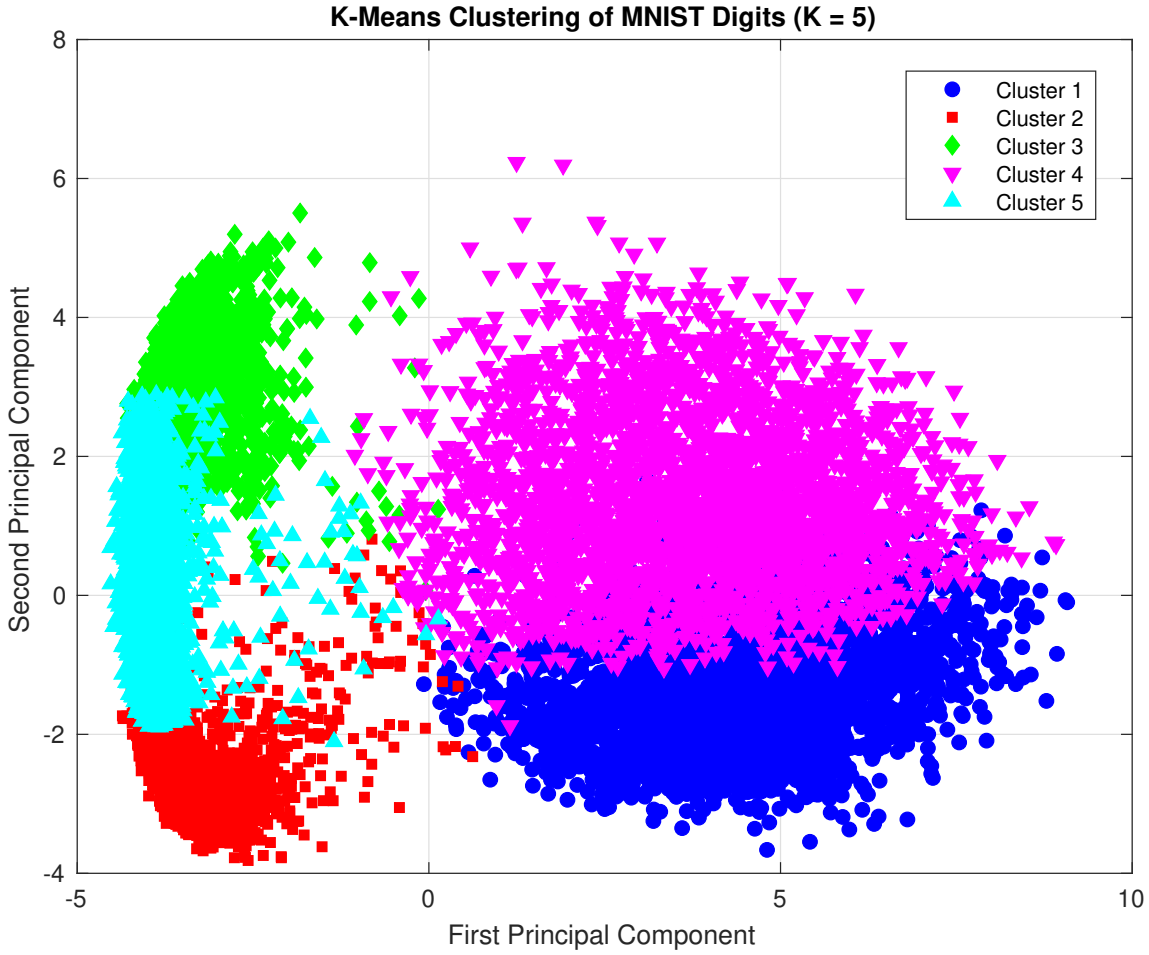

Fig. 3. K-means clustering results for K=5, visualized using the first two principal components. Each cluster is represented by a distinct color and marker.

TABLE II
LINEAR SVM CLASSIFICATION RESULTS

Training data	**Predicted class**	**True class:**	**# '0'**	**# '1'**
	'0'		5923	0
	'1'		0	6742
$N_{\text{train}} = 12665$		**Sum misclassified:**		0
		Misclassification rate (%):		0.00
Testing data	**Predicted class**	**True class:**	**# '0'**	**# '1'**
	'0'		979	1
	'1'		1	1134
$N_{\text{test}} = 2115$		**Sum misclassified:**		2
		Misclassification rate (%):		0.09

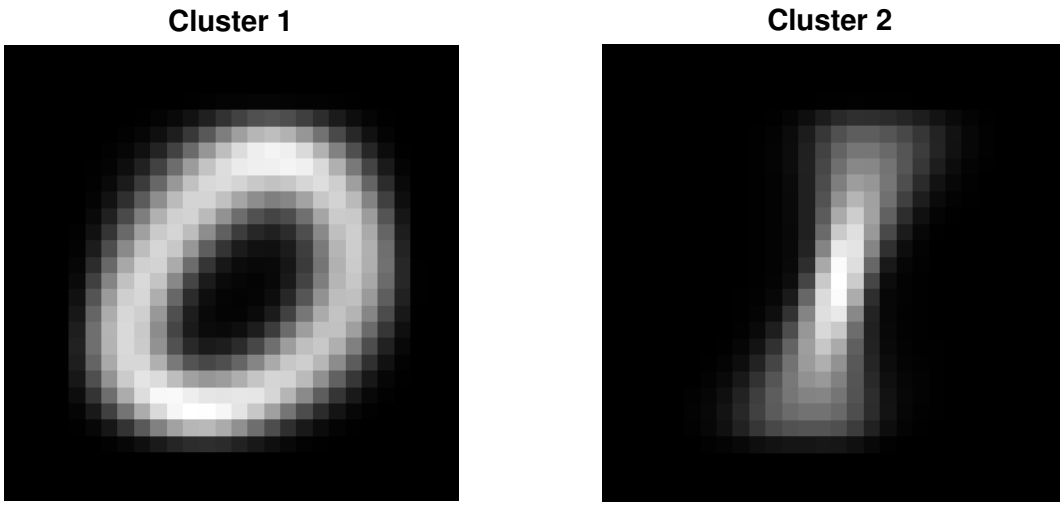

Fig. 4. Visualization of K-means cluster centroids for K=2, represented as 28x28 pixel images.

TABLE III
GAUSSIAN KERNEL SVM CLASSIFICATION RESULTS (OPTIMAL BETA)

Training data	**Predicted class**	**True class:**	**# '0'**	**# '1'**
	'0'		5921	1
	'1'		2	6741
$N_{\text{train}} = 12665$		**Sum misclassified:**		3
		Misclassification rate (%):		0.02
Testing data	**Predicted class**	**True class:**	**# '0'**	**# '1'**
	'0'		979	0
	'1'		1	1135
$N_{\text{test}} = 2115$		**Sum misclassified:**		1
		Misclassification rate (%):		0.05

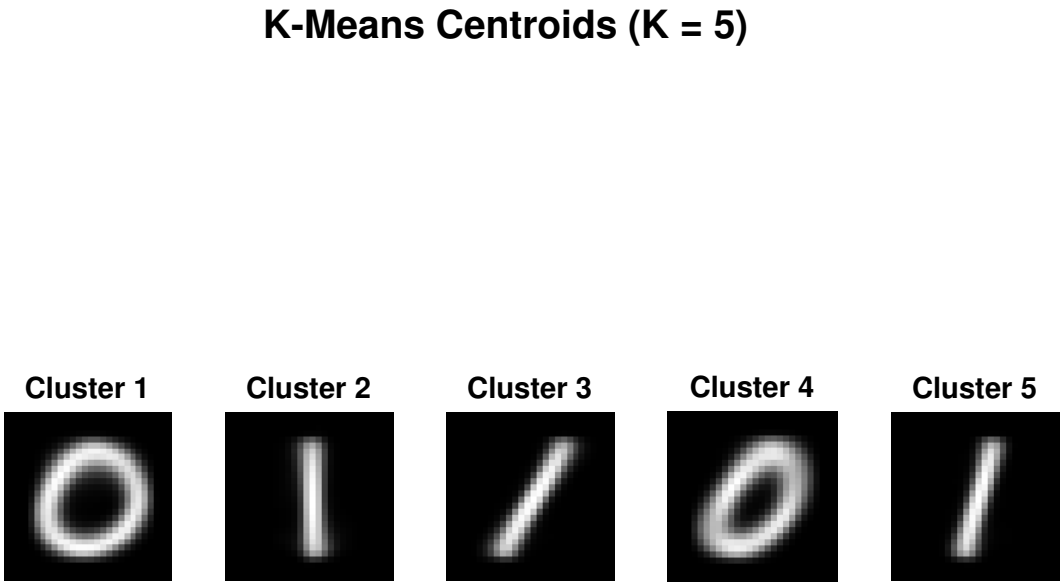

Fig. 5. Visualization of K-means cluster centroids for K=5, represented as 28x28 pixel images.

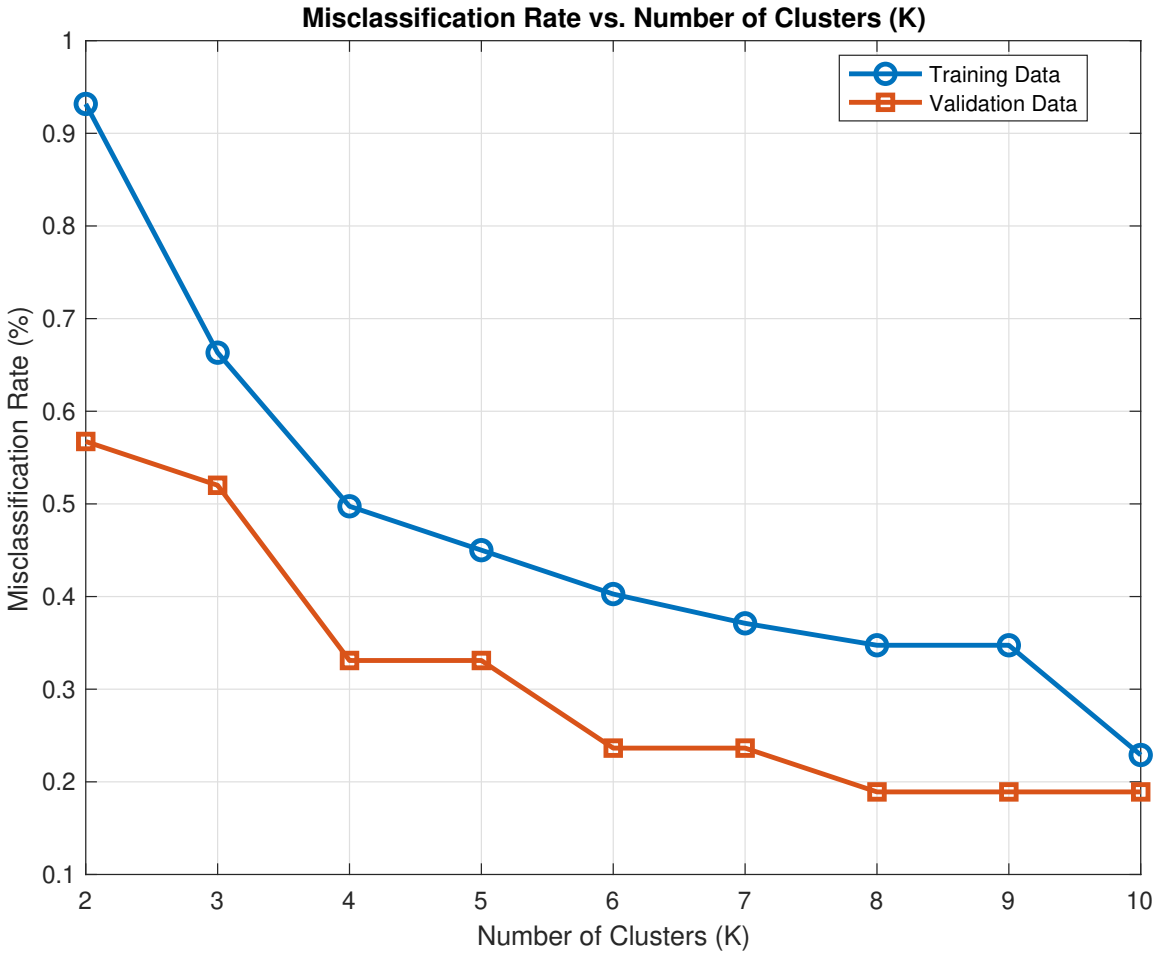

Fig. 6. Misclassification rates for training and validation data sets as a function of the number of clusters (K) in K-means clustering, showing a decreasing trend as K increases.