[FMAN45] - Assignment2

Filip Kalkan (fi1231ka-s)

May 2022

1 Task T1

Using the values and the function ϕ given in Task T1 in the assignment, the resulting kernel is

$$K = k(x_i, x_j) = \phi(x_i)^T \phi(x_j) = x_i x_j + (x_i x_j)^2 = \tag{1}$$

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

2 Task T2

Using that $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4$ the maximization problem can be rewritten as

$$\max_{\alpha} \left(4\alpha - \frac{\alpha^2}{2} \sum_{i,j=1}^{4} y_i y_j k(x_i, x_j) \right) \tag{3}$$

$$\text{subject to } \alpha \geq 0 \text{ and } \alpha \sum_{i=1}^{4} y_i = 0 \tag{4}$$

Inserting the values from the kernel into the summation in (3) gives

$$\sum_{i,j=1}^{4} y_i y_j k(x_i, x_j) = (2 \cdot 20 + 2 \cdot 12 + 2 \cdot 2 + 2 \cdot 0) - (4 \cdot 6 + 4 \cdot 2) = 36 \tag{5}$$

This, inserted into (3) yields

$$\max_{\alpha}(4\alpha - 18\alpha^2) \tag{6}$$

Now, we can verify that the function is concave by investigating the second derivative

$$\frac{d^2}{d\alpha^2}(4\alpha - 18\alpha^2) = \frac{d}{d\alpha}(4\alpha - 36\alpha^2) = -36 < 0 \tag{7}$$

As the second derivative is constantly negative, we can guarantee that the maximum of the function is

$$\frac{d}{d\alpha}(4\alpha - 18\alpha^2) = 4\alpha - 36\alpha^2 = 0 \iff \tag{8}$$

$$\iff \alpha = \frac{1}{9} \tag{9}$$

3 Task T3

In order to reduce the classifier function

$$g(x) = \sum_{j=1}^{4} \alpha y_j k(x_j, x) + b = \frac{1}{9} \sum_{j=1}^{4} y_j k(x_j, x) + b \tag{10}$$

to its simplest form, we begin by expanding it and eliminating terms.

$$g(x) = \frac{1}{9}((-2x + 4x^2) - (-x + x^2) - (x + x^2) + (2x + 4x^2)) + b = \tag{11}$$

$$= \frac{2}{3}x^2 + b \tag{12}$$

The unknown b can be found from

$$1 = y_s\bigg(\sum_{j=1}^{4} \alpha_j y_j k(x_j, x_s) + b\bigg) = \tag{13}$$

$$= y_s\bigg(\frac{2}{3}x^2 + b\bigg) = 1\bigg(\frac{2}{3}(-2)^2 + b\bigg) = \frac{8}{3} + b \iff \tag{14}$$

$$\iff b = 1 - \frac{8}{3} = -\frac{5}{3} \tag{15}$$

Inserting the result into (12) results in the simplest form;

$$g(x) = \frac{2}{3}x^2 - \frac{5}{3} \tag{16}$$

4 Task T4

Looking at the new dataset, it appears that it is a super set of the previously used dataset. Specifically x_2, x_3, x_5 and x_6 have corresponding data-target values in the previous dataset. Because of this, the classifier equation is equal to the one derived above (16).

5 Task T5

Given the primal formulation of the linear soft margin classifier

$$\min_{w,b,\xi} \frac{1}{2}||w||^2 + C\sum_{i=1}^{n} \xi_i \tag{17}$$

$$\text{subject to } y_i(w^T x_i + b) \geq 1 - \xi_i \ \forall i, \tag{18}$$

$$\xi_i \geq 0 \ \forall i \tag{19}$$

we want to derive the Lagrangian dual problem. We start off by stating the corresponding Lagrangian function.

$$L(w, b, \xi) = \frac{1}{2}||w||^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i(\xi - 1 + u_i(w^T x_i + b)) - \sum_{i=1}^{n} \lambda_i \xi_i \tag{20}$$

$$\alpha_i, \lambda_i \geq 0 \tag{21}$$

Now, minimizing L w.r.t w, b and ξ requires differentiation as follows.

$$\frac{dL}{dw} = 0 \iff w - \sum_{i=1}^{n} \alpha_i y_i x_i = 0 \iff w = \sum_{i=1}^{n} \alpha_i y_i x_i \tag{22}$$

$$\frac{dL}{db} = 0 \iff -\sum_{i=1}^{n} \alpha_i y_i = 0 \iff \sum_{i=1}^{n} \alpha_i y_i = 0 \tag{23}$$

$$\frac{dL}{d\xi} = 0 \iff C - \alpha_i - \lambda_i = 0 \iff \lambda_i = C - \alpha_i \tag{24}$$

Using (22), we can simplify (20) and produce the Lagrangian dual.

$$\max_{a_1,\ldots,a_n} L = \frac{1}{2}||\sum_{i=1}^{n} \alpha_i y_i x_i||^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i(\xi - 1 + u_i((\sum_{i=1}^{n} \alpha_i y_i x_i)^T x_i + b)) - \sum_{i=1}^{n} \lambda_i \xi_i \tag{25}$$

$$= -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j + C\sum_{i=1}^{n} \xi_i(C - \alpha_i - \lambda_i) + \sum_{i=1}^{n} \alpha_i - \sum_{i=1}^{n} \alpha_i y_i b \tag{26}$$

$$= \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j \tag{27}$$

Using that

$$a_i, \lambda_i \geq 0, \ \lambda_i = C - \alpha_i \tag{28}$$

the constrant can be stated as

$$0 \geq \alpha_i \geq C \ \forall i \tag{29}$$

6 Task T6

In order to show that support vectors with

$$y_i(w^T x_i + b) < 1 \tag{30}$$

have coefficient $\alpha_i = C$, we begin by stating the complementary slackness.

$$\alpha_i \Big(y_i(w^T x_i + b) - 1 + \xi_i \Big) = 0 \tag{31}$$

$$\lambda_i \xi_i = 0 \tag{32}$$

Using (24), we can rewrite (32) as

$$(C - \alpha_i)\xi_i = 0 \tag{33}$$

which holds if $\alpha_i = C$ given that $\xi_i > 0$. Furthermore, (31) holds if

$$y_i(w^T x_i + b) - 1 + \xi_i = 0 \iff \xi_i = 1 - y_i(w^T x_i + b) \tag{34}$$

With $\xi_i > 0$, this results in

$$1 - y_i(w^T x_i + b) > 0 \iff y_i(w^T x_i + b) > 1 \tag{35}$$

which is what we wanted to show.

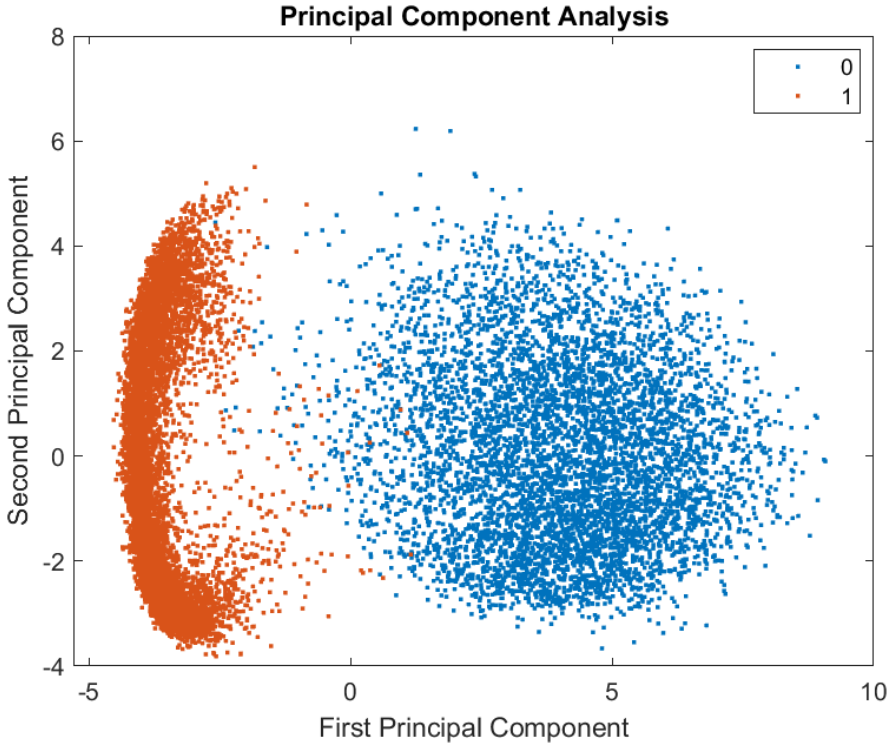

Figure 1: MNIST data set projected onto its first and second principal component

7 Task E1

In order to compute a linear PCA for the given data set, the data was firstly mapped to a corresponding zero mean data set. Using the new data set, singular value decomposition was performed. The two first left singular vectors were found as the first and second principal components. The zero mean data was the projected onto these vectors and colored by label. The result can be seen in figure 1.

8 Task E2

In this task, the data was classified using K-means clustering. Firstly, with 2 clusters and secondly with 5 clusters.

As seen in figures 2 and 3 some seemingly faulty assignments appear as a data point might be assigned to a cluster of which the centroid is not the closest one. This is due to the classification being carried out in a higher dimension. Once the dimensions are reduced via PCA some cluster overlap occurs as there is loss of information. In figure 2 the clusters seem to contain data points which have been reasonably classified. The reason is that the underlying data is sampled from 2 distinct classes. Thus, the algorithm manages to separate these fairly well.

Figure 3 on the other hand, shows unreasonable classification of the data points. The reason for this is that the algorithm was passed a hyperparameter ($K = 5$) which did not make sense given the data.

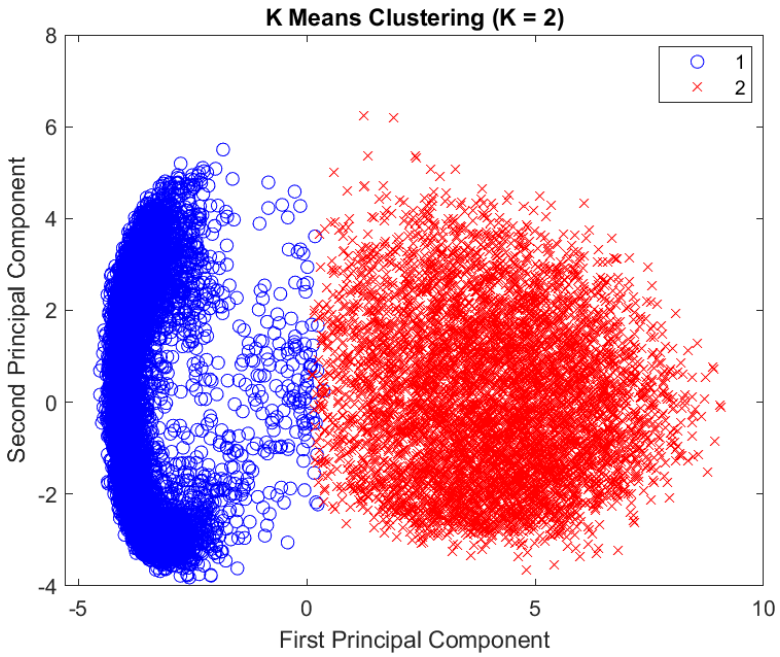

Figure 2: Classification of provided data using K = 2

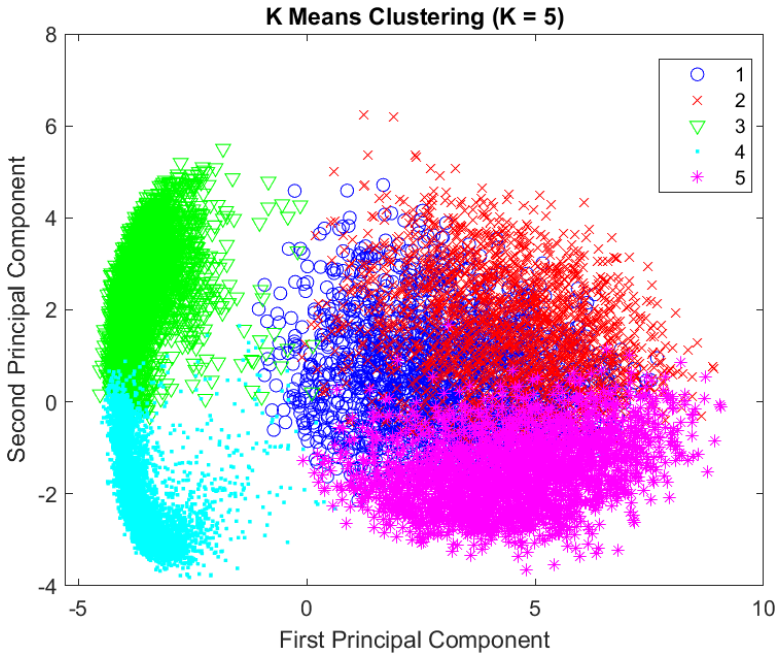

Figure 3: Classification of provided data using K = 5

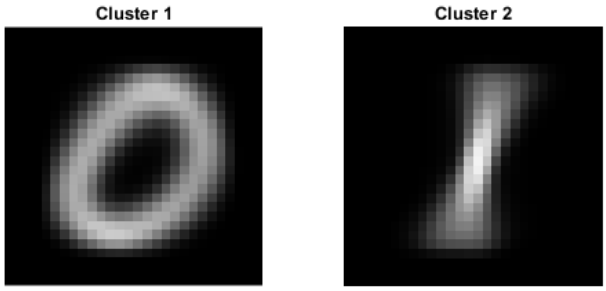

Figure 4: Cluster centroids using K = 2

9 Task E3

Figures 4 and 5 show images of the cluster centroids for $K = 2$ and $K = 5$ respectively. A distinct difference in the characteristics of each setting is that more centroids yield less blurry images. The reason for this is that the variance of the samples within a cluster is smaller with more clusters as each cluster contains fewer samples.

10 Task E4

Table 1: 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	10	1135	1	10
	2	970	0	0	0
$N_{\text{test}} = 2115$				Sum misclassified:	10
				Misclassification rate (%):	0.47

11 Task E5

Figure 6 shows the misclassification rate in log-scale for $K \in [1, 10]$.

The result shows a clear downward trend in misclassification rate as K increases. This is due to the homogeneity increasing along with K.

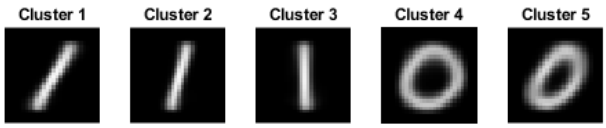

Figure 5: Cluster centroids using K = 5

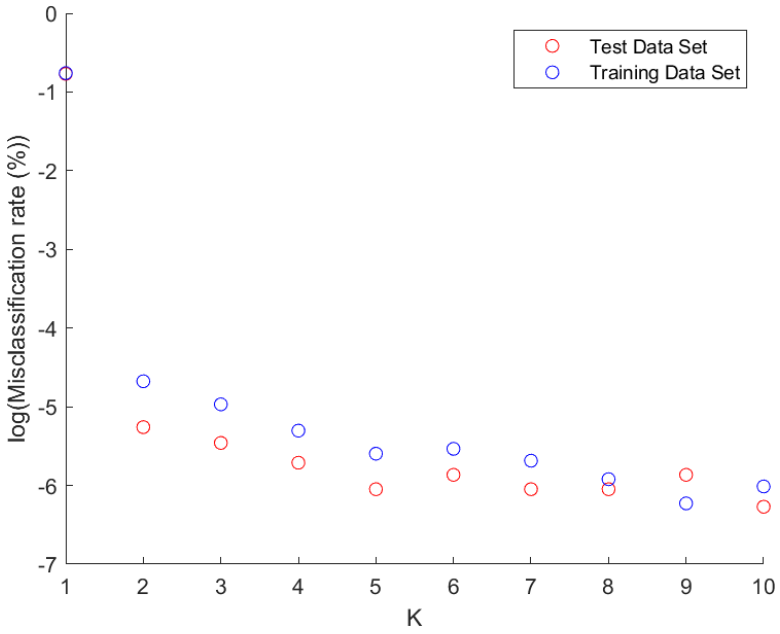

Figure 6: Misclassification rate with K-means classifier

12 Task E6

Table 2 presents the performance of a linear SVM classifier on our data. As shown, classifying the given data appears to be more performant with a linear SVM than with a K-means classifier (results shown in table 1).

Table 2: 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
Testing data	Predicted class	True class: # '0'	# '1'
	'0'	979	1
	'1'	1	1134
$N_{\text{test}} = 2115$		Sum misclassified:	2
		Misclassification rate (%):	0.095

13 Task E7

Table 3: Gaussian kernel SVM classification results

In order to find the optimal β i.e. β^*, the training and evaluation was run with $\beta \in \{1, 1.2, 1.4, ..., 8\}$. Out of these values $\beta^* = 5$ (among others) was found to yield 0 misclassifications.

Table 3 shows classification results with $\beta = \beta^* = 5$

Training data	Predicted class	True class: # '0'	# '1'
	'0'	5923	0
	'1'	0	6742
$N_{\text{train}} = 12665$		Sum misclassified:	0
		Misclassification rate (%):	0
Testing data	Predicted class	True class: # '0'	# '1'
	'0'	980	0
	'1'	0	1135
$N_{\text{test}} = 2115$		Sum misclassified:	0
		Misclassification rate (%):	0

14 Task E8

Using a hyper parameter adjusted to give perfect or near perfect results during the construction of a model is rarely a good idea. The reason for this is that this method of choosing hyper parameters tends to lead to overfitting. In other words, this risks adapting the model excessively to the available training/test data, which leads to overfitting.

Another way to explain it is using the bias/variance trade off which is present when training models. Optimizing the model with respect to the available data introduces a high bias while reducing variance. Eventually, this results in the model generalizing poorly.