Assignment 2

FMAN45 - Machine Learning

AUTHOR:
Dahlberg, Vilmer
`vi8808da-s@student.lu.se`

April 30, 2021

1 Introduction

Task T1

Given the data in the table below, and the transformation $\phi(x) = \begin{pmatrix} x \\ x^2 \end{pmatrix}$ compute the kernel K whos elements are given by $k_{i,j} = \{\phi(x_i)^T \phi(x_j)\}$.

i	1	2	3	4
x_i	-2	-1	1	2
y_i	1	-1	-1	1

Solution

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

Task T2

Solve the Lagrangian dual problem for the (hard margin) SVM numerically using the data from task 1.

Solution

The Lagrangian Dual function is given by

$$\Theta(\boldsymbol{\alpha}) = \sum_{i=1}^{4} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{4} \alpha_i \alpha_j y_i y_j k(x_i, x_j)$$

with constraints $\alpha_i \ \geq 0 \ \forall i$ and $\sum_{i=1}^{4} y_i \alpha_i = 0$. The solution satisfies $\alpha_i = \alpha_j \ \forall \ i, j \in [1, 2, 3, 4]$, which allows the dual function to be rewritten as

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

further, the second constraint is automatically satisfied. Given the data from the previous task, we have $-\frac{1}{2} \sum_{i,j=1}^{4} y_i y_j k(x_i, x_j) = -18$, meaning

$$\Theta(\alpha) = 4\alpha - 18\alpha^2.$$

The maximum of the dual function is then found by differentiating with respect to α and setting the result to zero, $4 - 36\hat{\alpha} = 0$, yielding the optimal solution $\hat{\alpha} = \frac{1}{9}$.

Task T3

> Using the same data-target pairs, reduce the classifier function to the most simple of the simplest possible form, leading to a polynomial in x.

Solution

Begin by computing the term $k(x_j, x)$, which is given by

$$k(x_j, x) = \phi(x_j)^T \phi(x) = \begin{bmatrix} x_j & (x_j)^2 \end{bmatrix} \begin{bmatrix} x \\ x^2 \end{bmatrix} = xx_j + (xx_j)^2$$

The classifier can then easily be computed

$$g(x) = \sum_{j=1}^{4} \alpha_j y_j k(x_j, x) + b = x\left(\sum_{j=1}^{4} \alpha_j y_j x_j\right) + x^2\left(\sum_{j=1}^{4} \alpha_j y_j x_j^2\right) + b.$$

With the data points used in the first task, and the solution $\hat{\alpha} = \frac{1}{9}$ gives $\sum_{j=1}^{4} \alpha_j y_j x_j = 0$, and $\sum_{j=1}^{4} \alpha_j y_j x_j^2 = 2/3$ and reduces the expression to

$$g(x) = \frac{2}{3}x^2 + b.$$

The constant b can be computed by studying the following relation which holds for any support vector x_s

$$y_s\left(\sum_{i=1}^{4} \alpha_i y_i k(x_j, x_s) + b\right) - 1 = 0.$$

Studying the data points, we find that all vectors are support vectors. Taking any data-target pair (x_s, y_s), for example let s = 1,

$$\frac{1}{9}(20 - 6 - 2 + 12) + b - 1 = 0 \iff b = 1 - \frac{24}{9} = -\frac{5}{3}.$$

Finally, we have an expression for the classifier

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

Evaluating at the data points we see that all data points are support vectors,

$$g([-2, -1, 1, 2]) = [1, -1, -1, 1].$$

Task T4

> Using the same kernel as above, what is the solution g(x) of the nonlinear kernel SVM with hard constraint on the data set below?

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

Solution

Begin by studying the image of the data when mapped onto feature space.

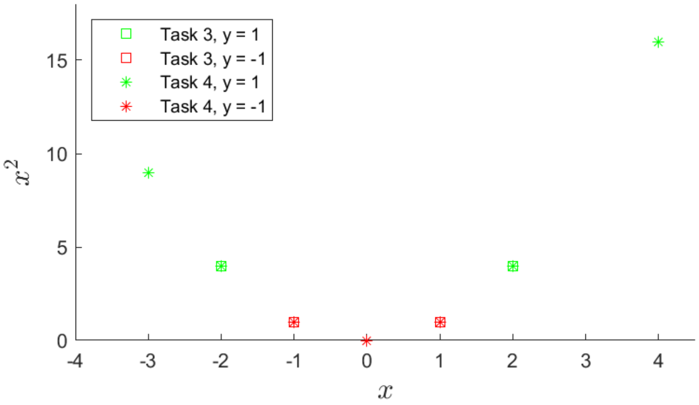

Figure 1: Data set used in this task versus data set used in the previous task.

Even though this data set contains 3 new points, the support vectors are the same, meaning the same classifier can be used for this data. The classifier is given by

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

Task T5

The primal formulation of the linear soft margin classifier is given by

$$\begin{aligned} \underset{\boldsymbol{w},b,\boldsymbol{\xi}}{\text{minimize}} \quad & \frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1}^{n}\xi_i \\ \text{subject to} \quad & y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \xi_i \\ & \xi_i \geq 0, \end{aligned}$$

find the Lagrangian dual problem.

Solution

Begin by assembling the Lagrangian using the Lagrangian multipliers $\alpha_i > 0$

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\xi}; \boldsymbol{\alpha}) = \frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i\Big(y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) - (1 - \xi_i)\Big).$$

After some simplification we get

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\xi}; \boldsymbol{\alpha}) = \frac{1}{2}||\boldsymbol{w}||^2 - \sum_{i=1}^{n}\alpha_i\Big(y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) - 1\Big) + \xi_i(\alpha_i - C).$$

The dual function is given by minimizing the Lagrangian with respect to the variables $\boldsymbol{w}, b, \boldsymbol{\xi}$, meaning

$$\Theta(\boldsymbol{\alpha}) = \underset{\boldsymbol{w},b,\boldsymbol{\xi}}{\text{minimize}}\ \mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\xi}; \boldsymbol{\alpha}).$$

This can be done by computing the gradient of the Lagrangian, and setting the result to zero. Start by considering the derivative with respect to w_j

$$\frac{\partial \mathcal{L}}{\partial w_j} = w_j - \sum_{i=1}^{n} \alpha_i y_i \frac{\partial \boldsymbol{w}}{\partial w_j}^T \boldsymbol{x}_i = w_j - \sum_{i=1}^{n} \alpha_i y_i (\boldsymbol{x}_i)_j = 0 \iff \boldsymbol{w} = \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i.$$

Insert the expression into the Lagrangian.

$$\begin{aligned}
\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\xi}; \boldsymbol{\alpha}) &= \frac{1}{2}||\boldsymbol{w}||^2 - \sum_{i=1}^{n} \alpha_i \Big(y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) - 1 \Big) + \xi_i(\alpha_i - C) \\
&= \frac{1}{2}\Big(\sum_{j=1}^{n} \alpha_j y_j \boldsymbol{x}_j^T \Big)\Big(\sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i \Big) - \sum_{i=1}^{n} \alpha_i \Big(y_i(\sum_{j=1}^{n} \alpha_j y_j \boldsymbol{x}_j^T \boldsymbol{x}_i + b) - 1 \Big) + \xi_i(\alpha_i - C) \\
&= \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i \boldsymbol{x}_j^T - \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^T \boldsymbol{x}_j - \sum_{i=1}^{n} \alpha_i y_i b - \alpha_i + \xi_i(\alpha_i - C) \\
&= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i \boldsymbol{x}_j^T + \sum_{i=1}^{n} \xi_i(\alpha_i - C).
\end{aligned}$$

Next consider the derivative with respect to b,

$$\frac{\partial \mathcal{L}}{\partial b} = \sum_{i=1}^{n} \alpha_i y_i = 0.$$

Lastly, the contribution from ξ is studied in three cases: $\alpha_j > C$, $\alpha_j = C$, $\alpha_j < C$. In first case, the Lagrangian can grow arbitrarily small by letting $\xi \longrightarrow \infty$ and the value of the dual function approaches $-\infty$. In the second case the choice of ξ_j is arbitrary, and in the last case the Lagrangian is minibymized when $\xi_j = 0$. The dual function can now be created by inserting the expressions gathered above

$$\Theta(\boldsymbol{\alpha}) = \begin{cases} -\infty & \text{if } \ \alpha_j > C \\ \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i \boldsymbol{x}_j^T & \text{if } \ 0 \leq \alpha_j \leq C \end{cases}$$

The dual problem is to maximize the dual function with respect to the Lagrangian multipliers. The case $\alpha_j > C$ can be discarded, and the problem can instead be confined to

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

Task T6

Show that support vectors with $y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) < 1$ have coefficient $\alpha = C$.

Solution

By the complementary slackness principle, we have

$$\alpha_i(1 - \xi_i - y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b)) = 0.$$

Since $\boldsymbol{x}_i$ is a support vector $\alpha_i \neq 0$, which means $1 - \xi_i - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) = 0$. Isolating the error on the left hand side $\xi_i = 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) > 0$, since $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) < 1$. The components of the Lagrangian that are dependant on α_i are

$$-\alpha_i(y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) - 1) - \xi_i\alpha_i = -2\xi_i\alpha_i.$$

Thus, the Lagrangian is minimized if we choose the largest possible value for α_i (since $\xi_i > 0$), which is C.

Task E1

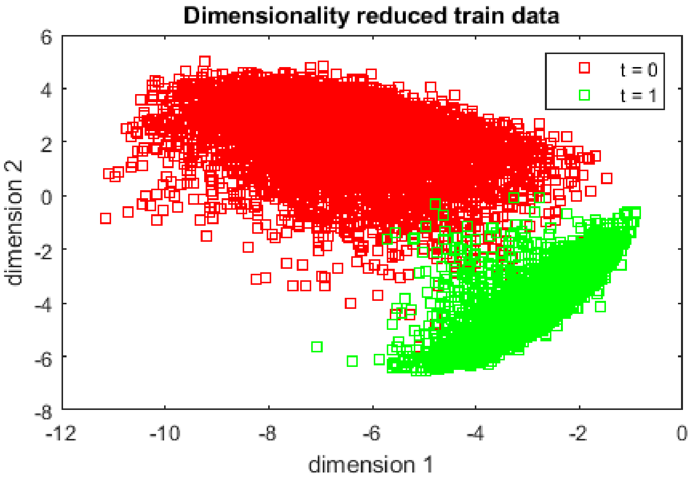

Figure 2: The training data is projected onto the space spanned by the two left singular vectors with the largest singular values. There are two pretty clear clusters, with some overlap.

Task E2

In this task the K-means clustering algorithm was used to cluster the training data. The 2-norm was used both to compute the distance between a data point and centroids, as well as to compute the distance between two centroids.

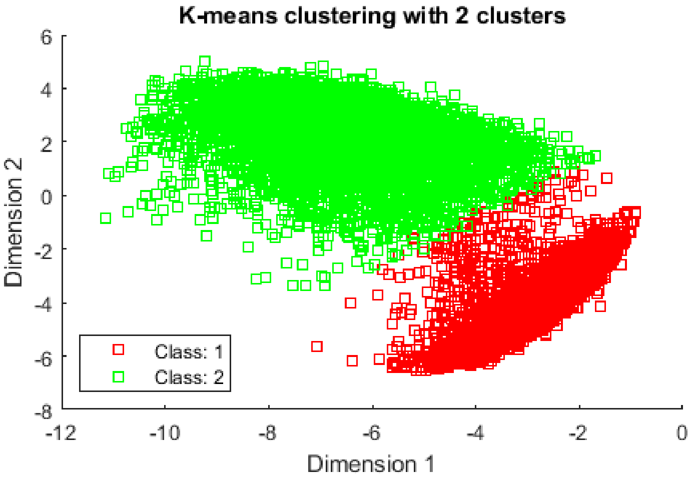

Figure 3: The data is clustered into two clusters using a K-means algorithm, and are labeled according to their cluster. The training data is then projected onto the space spanned by the two left singular vectors used in the previous task.

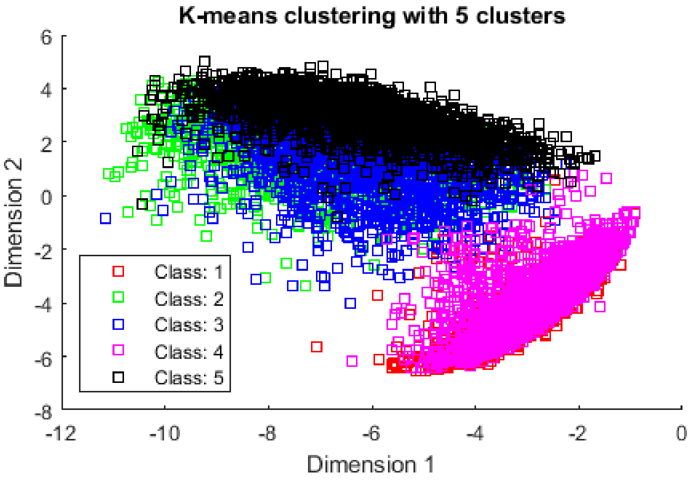

Figure 4: The data is clustered into fives clusters using a K-means algorithm, and are labeled according to their cluster. The training data is then projected onto the space spanned by the two left singular vectors used in the previous task.

In the 784 dimensions of the original data set there is no overlap, since the minimum distance is used to classify the data. However, when the data is projected onto two dimensions some overlap is introduced. This is easily realized if you consider projecting two circles in two dimensions down to one dimension. It is easy to create cases where the circles do not overlap in two dimensions, but the projections onto a straight line overlap.

Task E3

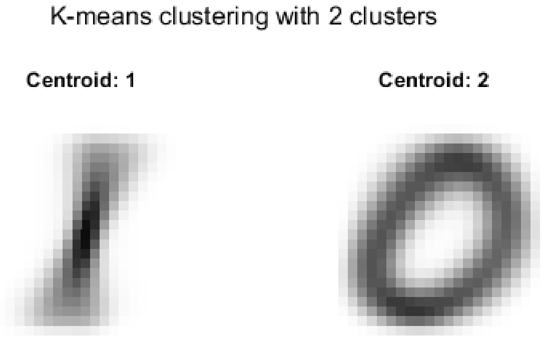

Figure 5: Images of the centroids of the two clusters when a K-means clustering method is used.

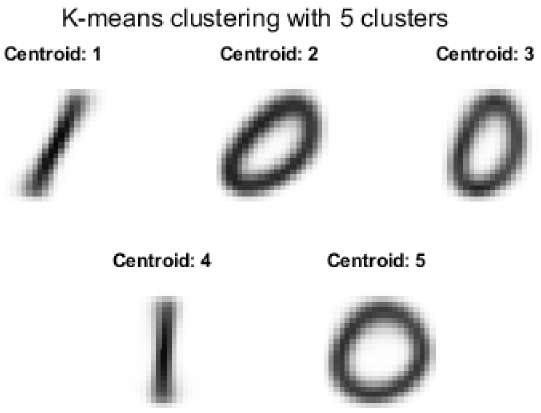

Figure 6: Images of the centroids of the five clusters when a K-means clustering method is used.

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_{train} = 12,665				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_{test} = 2115				Sum misclassified:	12
				Misclassification rate (%):	0.57

Task E5

After some experimentation, using six clusters had good results.

Table 2: K-means classification results

Training data	Cluster	# ’0’	# ’1’	Assigned to class	# misclassified
	1	27	3687	1	27
	2	1629	0	0	0
	3	1552	0	0	0
	4	1497	6	0	6
	5	1205	4	0	4
	6	13	3045	1	13
$N_{\text{train}} = 12{,}665$				Sum misclassified:	50
				Misclassification rate (%):	0.39
Testing data	Cluster	# ’0’	# ’1’	Assigned to class	# misclassified
	1	2	659	1	2
	2	280	0	0	0
	3	235	0	0	0
	4	297	0	0	0
	5	163	0	0	0
	6	3	476	1	3
$N_{\text{test}} = 2115$				Sum misclassified:	5
				Misclassification rate (%):	0.24

Task E6

Table 3: Linear SVM classification results

Training data	Predicted class	True class:	# ’0’	# ’1’
	’0’		5923	0
	’1’		0	6742
$N_{\text{train}} = 12{,}665$			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 (%):	0.09

Task E7

After a couple of tries, using a beta-value of 5 reduced the misclassification rate to 0%.

Table 4: Gaussian kernel SVM classification results

Training data	Predicted class	True class:	# ’0’	# ’1’
	’0’		5923	0
	’1’		0	6742
$N_{\text{train}} = 12{,}665$		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

Task E8

Since the testing data is used as a validation set, we cannot expect the same missrate on new images. The model will be overfit to the test data, and we will end up in the same situation as in the previous assignment.