Machine Learning
Exercise Sheet 4 Solution

Alper Hamza Ari
5771973

Chenqi Hao
5781375

Danil Skokov
5779466

Said Orfan Haidari
5781295

November 15, 2023

1 Support Vector Machines

1.1 Explain the "kernel trick" and why we use it in SVMs.

Kernek trick is a method that allows us to compute dot product of two vectors in a higher dimension without actually transforming them into that higher dimension. It reduces computation. We use kernel trick when data is not linearly seperable, not because of outlier data nor noise, but because of the data itself.

1.2 What is the difference between hard- and soft-margin SVM?

`Hard-margin SVM`: It doesn't tolerate noise and outliers. So it has narrower margin. It is strict on linearizability constraints.
`Soft-margin SVM`: It can tolerate some noise and outlier data by expanding the margin. It relaxes linear separability.

1.3 SVM Problem as average hinge loss and regularization

1.3.1 Compute the gradient of the training objective w.r.t. w and describe the pseudocode for the gradient descent algorithm.

$$L_i = \begin{cases} 1 - y_i(w^T x_i) & \text{if } \; y_i(w^T x_i) < 1 \\ 0 & \text{if } \; y_i(w^T x_i) \geq 1 \end{cases}$$

$$\frac{\partial L_i}{\partial w} = \begin{cases} -y_i x_i & \text{if } \; y_i(w^T x_i) < 1 \\ 0 & \text{if } \; y_i(w^T x_i) \geq 1 \end{cases}$$

$$\frac{\partial R}{\partial w} = \lambda \sum_{j=1}^{m} w_j$$

$$\frac{\partial f(w)}{\partial w} = \frac{1}{n} \sum_{i=1}^{n} \begin{Bmatrix} -y_i x_i & \text{if } \; y_i \hat{y}_i < 1 \\ 0 & \text{if } \; y_i \hat{y}_i \geq 1 \end{Bmatrix} + \lambda \sum_{j=1}^{m} w_j$$

Figure 1: Computation of gradient

for $1, \ldots, \mathcal{I}$ **do**
    for $i = 1, \ldots, N$ **do**
        $w \leftarrow w - \eta \frac{\partial f(w)}{\partial w}$
    end for
end for

Figure 2: Pseudocode for gradient descent algorithm

1.3.2 Use gradient descent to update the weights

$$w^{(t+1)} = w^{(t)} - \eta \frac{\partial f(w)}{\partial w}$$

$$= w^{(t)} - \eta \cdot \lambda \cdot \sum_{j=1}^{m} w_j$$

x_1	x_2	y
1	0	1

$$w^{(0)} = \langle 1, 1 \rangle$$

$$\hat{y}_1 = y_i \left(w^T x_i + b \right)$$

$$= 1 \left[(1,1) \cdot (1,0) + 0 \right]$$

$$= 1 \left[1 + 0 + 0 \right] = 1$$

see that $y_1 \hat{y}_1 = 1$

$$\frac{\partial f(w)}{\partial w} = 0 + \lambda \sum_{j=1}^{m=2} w_j$$ because $y_1 \hat{y}_1 = 1$

$$w^{(1)} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - (0{,}005) \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$= \begin{pmatrix} 0{,}995 \\ 0{,}995 \end{pmatrix}$$

Figure 3: New weights

2 Linear Separability

2.1 Deriving update rule for gradient descent

2.1.1 Sketch the given dataset

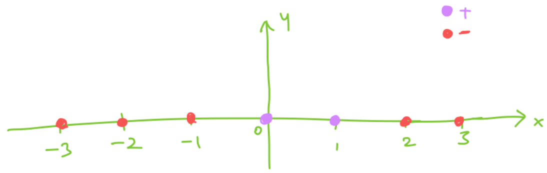

Figure 4: Sketch of 1D dataset

2.1.2 Find a separating hyperplane, find its parameters and plot it

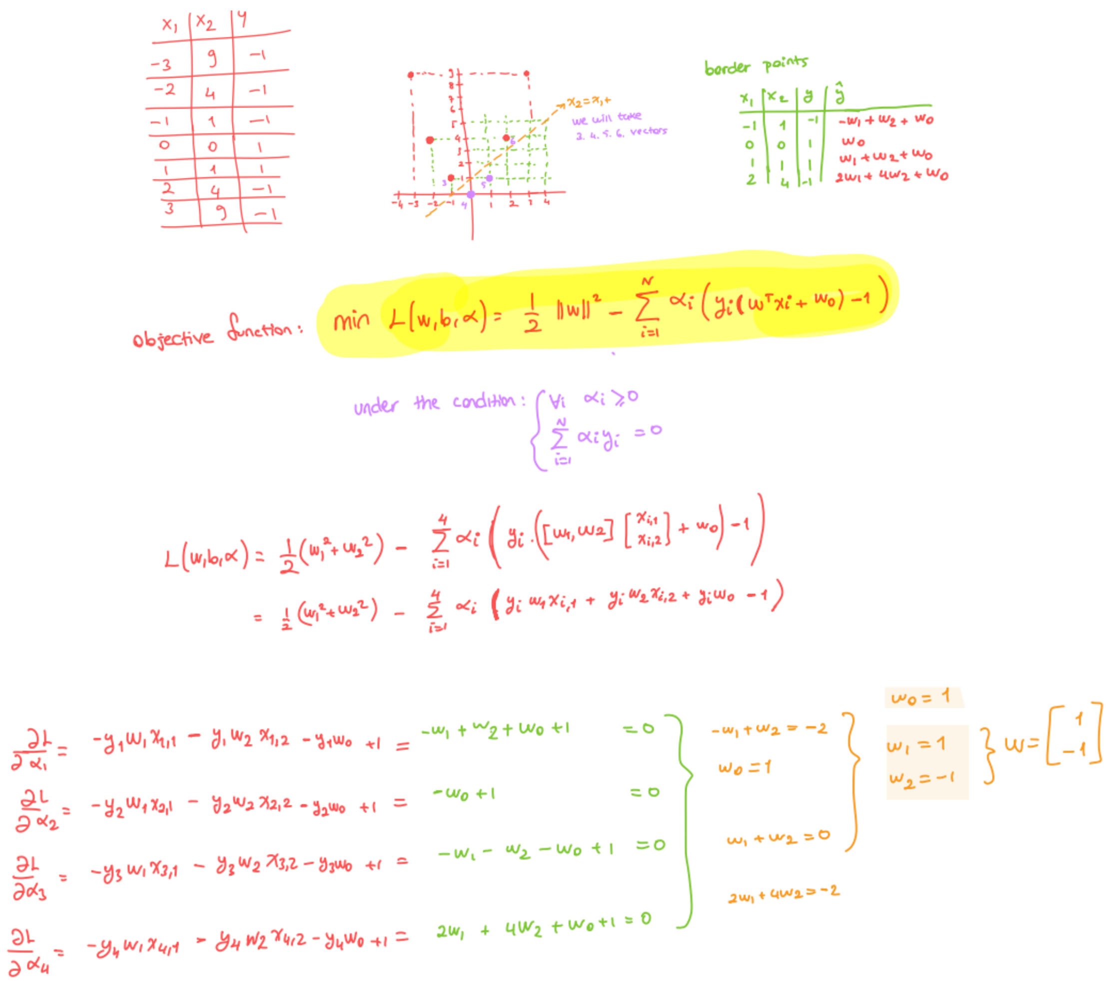

Figure 5: Finding a hyperplane and sketch of the dataset in a new space

2.1.3 Use the computed hyperplane to compute output for $x = \frac{1+\sqrt{5}}{2}$

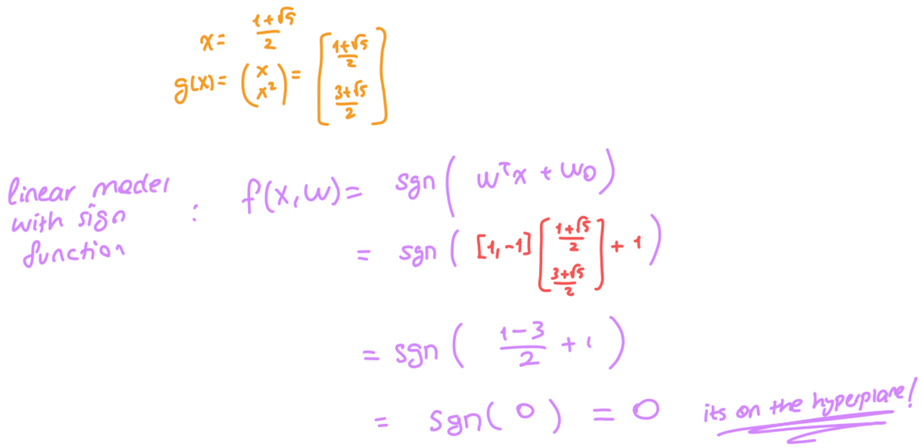

Figure 6: Output for $x = \frac{1+\sqrt{5}}{2}$