Homework 6

- Daria Laslo -

Exercise 1.a The variant of SVM used in transductive SVM [2] is:

$$\min_{w,b,\xi,\xi^*} \frac{1}{2}||\vec{w}||^2 + C\sum_{i=1}^{n}\xi_i + C_-^*\sum_{j:y_j^*=-1}\xi_j^* + C_+^*\sum_{j:y_j^*=1}\xi_j^*$$

$$\begin{aligned}\text{s.t. } & y_i(\langle\vec{w},\vec{x}_i\rangle + b) \geq 1-\xi_i, \ \ \forall i \in \{1,\dots,n\}\\ & y_j^*(\langle\vec{w},\vec{x}_j\rangle + b) \geq 1-\xi_j^*, \ \ \forall j \in \{1,\dots,k\}\\ & \xi_i \geq 0, \ \ \forall i \in \{1,\dots,n\}\\ & \xi_j^* \geq 0, \ \ \forall j \in \{1,\dots,k\}\end{aligned}$$

To arrive at the dual representation we first have to derive the objective using the method of Lagrange multipliers as following:

(we use the $\alpha_i, \alpha_j, \beta_i, \beta_j$ as Lagrange multipliers for the four constraints)

$$\begin{aligned}\mathcal{L}((w,b,\xi,\xi^*),\alpha,\beta) = & \frac{1}{2}||\vec{w}||^2 + C\sum_{i=1}^{n}\xi_i + C_-^*\sum_{j:y_j^*=-1}^{k}\xi_j^* + C_+^*\sum_{j:y_j^*=1}^{k}\xi_j^*\\ & - \sum_{i=1}^{n}\alpha_i(y_i(\langle w,x_i\rangle + b) - (1-\xi_i)) - \sum_{i=1}^{n}\beta_i\xi_i\\ & - \sum_{j=1}^{k}\alpha_j(y_j^*(\langle w,x_j\rangle + b) - (1-\xi_j^*)) - \sum_{i=j}^{k}\beta_j\xi_j^*\end{aligned} \quad (1)$$

In order to obtain the dual the Lagrangian has to be maximised with respect to w, b, ξ, ξ^*. Therefore the first derivatives with respect to those need to be set to 0.

$$\frac{\partial \mathcal{L}}{\partial w} = 0 \Rightarrow -\sum_{i=1}^{m} \alpha_i y_i \vec{x_i} - \sum_{j=1}^{k} \alpha_j y_j^* \vec{x_j} + \vec{w} = 0$$

$$\vec{w} = \sum_{i=1}^{m} \alpha_i y_i \vec{x_i} + \sum_{j=1}^{k} \alpha_j y_j^* \vec{x_j} = \sum_{i=1}^{m+k} \alpha_i y_i \vec{x_i} \quad (2)$$

$$\frac{\partial \mathcal{L}}{\partial b} = 0 \Rightarrow \sum_{i=1}^{m} \alpha_i y_i + \sum_{j=1}^{k} \alpha_j y_j^* = 0 \text{ (multiplied by } (-1)) \quad (3)$$

$$\frac{\partial \mathcal{L}}{\partial \xi_i} = 0 \Rightarrow C - \alpha_i - \beta_i = 0 \Rightarrow \beta_i = C - \alpha_i \quad (4)$$

$$\frac{\partial \mathcal{L}}{\partial \xi_j^*} = 0 \Rightarrow \begin{cases} C_-^* - \alpha_j - \beta_j = 0 \Rightarrow \beta_j = C_-^* - \alpha_i \text{ (if } j : y_j^* = -1) & (5) \\ C_+^* - \alpha_j - \beta_j = 0 \Rightarrow \beta_j = C_+^* - \alpha_i \text{ (if } j : y_j^* = 1) & (6) \end{cases}$$

Additionally, we know the Lagrange multipliers need to be non negative.

$$\beta_m \geq 0 \quad (7)$$

$$\alpha_m \geq 0$$

(m - arbitrary indexing)

$$(4) + (7) \Rightarrow \beta_i = C - \alpha_i \geq 0 \Rightarrow \alpha_i \leq C, \forall i$$

$$(7) + (5) \Rightarrow \beta_j = C_-^* - \alpha_j \geq 0 \Rightarrow \alpha_j \leq C_-^* \text{ (if } j : y_j^* = -1)$$

$$(7) + (6) \Rightarrow \beta_j = C_+^* - \alpha_j \geq 0 \Rightarrow \alpha_j \leq C_+^* \text{ (if } j : y_j^* = 1)$$

By expanding and rearranging (1) we obtain:

$$\mathcal{L} = \frac{1}{2}\|\vec{w}\|^2 - \vec{w} * \left(\sum_{i=1}^{m} \alpha_i y_i \vec{x_i} + \sum_{j=1}^{k} \alpha_j y_j^* \vec{x_j} \right) -$$

$$- b * \left(\sum_{i=1}^{m} \alpha_i y_i + \sum_{j=1}^{k} \alpha_j y_j^* \right) + \cancel{\sum_{i=1}^{m} (\beta_i + \alpha_i) \xi_i} +$$

$$+ \underbrace{\sum_{j:y_j^*=-1}^{k} (\beta_j + \alpha_j) \xi_j^* + \sum_{j:y_j^*=+1}^{k} (\beta_j + \alpha_j) \xi_j^*}_{\cancel{\sum_{j=1}^{k} (\beta_j + \alpha_j) \xi_j^*} \text{ (considering the two cases, all } j \text{ are covered)}} + \sum_{i=1}^{m} \alpha_i + \sum_{j=1}^{k} \alpha_j$$

$$- \cancel{\sum_{i=1}^{m} \alpha_i \xi_i} - \cancel{\sum_{i=1}^{m} \beta_i \xi_i} - \cancel{\sum_{j=1}^{k} \alpha_j \xi_j^*} - \cancel{\sum_{j=1}^{k} \beta_j \xi_j^*}$$

After simplifying and using (2) and (3) to substitute, we obtain:

$$\mathcal{L} = \frac{1}{2}\|\vec{w}\|^2 - \vec{w} * \vec{w} - b * 0 + \sum_{i=1}^{m} \alpha_i + \sum_{j=1}^{k} \alpha_j$$

If we concatenate our training and test samples we can use one indexing going from 1 to $k+m$.

$$\mathcal{L} = -\frac{1}{2}\|\vec{w}\|^2 + \sum_{i=1}^{m+k} \alpha_i$$

$$\xrightarrow{(2)} = \sum_{i=1}^{m+k} \alpha_i - \frac{1}{2} \left\langle \sum_{i=1}^{m+k} \alpha_i y_i \vec{x_i}, \sum_{i=1}^{m+k} \alpha_i y_i \vec{x_i} \right\rangle$$

$$= \sum_{i=1}^{m+k} \alpha_i - \frac{1}{2} \sum_{i=1}^{m+k} \sum_{j=1}^{m+k} \alpha_i \alpha_j y_i y_j \langle \vec{x_i}, \vec{x_j} \rangle \quad (8)$$

Writing the obtained eq and constraints gives us the dual problem as stated in the task:

$$\max_{\alpha} \; W(\alpha) = \sum_{i=1}^{n+k} \alpha_i - \frac{1}{2} \sum_{i=1}^{n+k} \sum_{j=1}^{n+k} \alpha_i \alpha_j y_i y_j \langle \vec{x_i}, \vec{x_j} \rangle \quad (2)$$

$$\text{s.t.} \quad \sum_{i=1}^{n+k} \alpha_i y_i = 0 \quad (3)$$

$$0 \leq \alpha_i \leq C, \quad \forall i \in \{1, \dots, n\} \quad (4)$$

$$0 \leq \alpha_i \leq C^*_-, \quad \forall i \in \{n+1, \dots, n+k\}, \; y_i = -1 \quad (5)$$

$$0 \leq \alpha_i \leq C^*_+, \quad \forall i \in \{n+1, \dots, n+k\}, \; y_i = 1 \quad (6)$$

$\blacksquare$

Homework 6

Daria Laslo

June 1, 2022

Exercise 1. (a). See pages above.

(b). See code in svm$_l$*inear.py*.

Exercise 2. (a). See code in tsvm$_l$*inear.py*.

(b). C1 and C2 have an important effect on the decision boundary the classifier converges to. The value of these affects how much the misclassification of a training or test point respectively is penalised. A large value means, the margin width will be reduced in order to minimize the number of misclassified points. A small value means that the margin width would be larger, however, at the expense of misclassifying some points.

Hypertuning these parameters attempts to find a trade-off between the two. One possible way of doing that would be by using cross validation for different values of the parameters. Such an approach would be doing a grid search cross validation. In the case of transductive SVMs we need to consider the two data sets so the cross validation should be done for both the training and the test data set.

(c). H is called a reproducing space because of its 'reproducing' property. This property allows the evaluation of a point x to be only expressed in terms of the inner product with a function K_x which is determined by the kernel, rather than having the need to compute the map. Therefore, the mapping is 'reproduced' in the Hilbert space by using the inner product, this being possible, however, only under the condition that the mapping function itself is continuous.