Machine Learning 2024 - Sheet 3.2
Block III: SVM and Kernel Methods

Isabel Valera

Exercise 1: Kernel feature representation

Given the kernel $k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^\top \mathbf{z} + c)^2$. Write down, step-by-step, a feature representation $\phi : \mathbb{R}^d \to \mathbb{R}^p$ where $d < p$ such that $\langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle = k(\mathbf{x}, \mathbf{z})$.

Exercise 2: Eigenvalues

Definition Positive Definite Matrix *A complex $m \times m$ matrix K satisfying*

$$\sum_{i,j} c_i \bar{c}_j K_{ij} \geq 0 \tag{1}$$

for all $c_i \in \mathbb{C}$ is called positive definite. The bar in $\bar{c}_j$ denotes complex conjugation; for real numbers, it has no effect. Similarly, a real symmetric $m \times m$ matrix K satisfying (1) for all $c_i \in \mathbb{R}$ is called positive definite.

Prove that a symmetric matrix is positive definite if and only if all its eigenvalues are non-negative.

Exercise 3: Dot products are kernels

Definition Dot Product *A dot product on a vector space $\mathcal{H}$ is a symmetric bilinear form,*

$$\begin{aligned} \langle ., . \rangle : \mathcal{H} \times \mathcal{H} &\to \mathbb{R} \\ (\mathbf{x}, \mathbf{x}') &\mapsto \langle \mathbf{x}, \mathbf{x}' \rangle \end{aligned}$$

that is strictly positive definite; in other words, it has the property that for all $\mathbf{x} \in \mathcal{H}$, $\langle \mathbf{x}, \mathbf{x} \rangle \geq 0$ with equality only for $\mathbf{x} = 0$.
Prove that dot products are positive definite kernels.

Exercise 4: Kernel Logistic Regression

Consider a Logistic Regression (LR) model with the following loss function (cross entropy):

$$L(\theta) = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \log P_\theta(\hat{y} = 1 \,|\, \mathbf{x}_i) + (1 - y_i) \log \left(1 - P_\theta(\hat{y} = 1 \,|\, \mathbf{x}_i)\right) \right] \tag{2}$$

where $P_\theta(\hat{y} = 1 \,|\, \mathbf{x}) = \frac{1}{1+\mathrm{e}^{-\theta^T \mathbf{x}}} = \sigma\left(\theta^T \mathbf{x}\right)$ with sigmoid function $\sigma(\cdot) : \mathbb{R} \to [0, 1]$.

Given a training dataset $\{\mathbf{x}_i, y_i\}_{i=1}^N$, where $\mathbf{x}_i \in \mathbb{R}^L$ and $y_i \in \{0, 1\}$, assume that each of the input vectors can be transformed as $\phi(\mathbf{x}) : \mathbb{R}^L \to \mathbb{R}^D$ with $D \geq L$. Derive step by step the dual loss function for the Kernel Logistic Regression for binary classification. State whether and, if so, where you used the kernel trick and/or the representer theorem. Make sure to explicitly state your assumptions and explain every step in your own words.

Exercise 5: Group error penalty ⚡⚡⚡

Suppose the training data are partitioned into l groups:

$$\begin{matrix} (\boldsymbol{x}_1^1, y_1^1), & ..., & (\boldsymbol{x}_1^{m_1}, y_1^{m_1}) \\ \vdots & & \vdots \\ (\boldsymbol{x}_l^1, y_l^1), & ..., & (x_l^{m_l}, y_l^{m_l}), \end{matrix} \tag{3}$$

where $\boldsymbol{x}_i^j \in \mathcal{H}$ and $y_i^j \in \{-1, +1\}$ with $i = 1, ..., l$ and $j = 1, ..., m_i$.
Suppose, moreover, that we would like to count a point as misclassified already if one point belonging to the same group is misclassified. Design a soft-margin support vector algorithm where each group's penalty equals the slack of the worst point in that group.

Use the following objective and constraints:

$$\begin{aligned} &\min \frac{1}{2}||\mathbf{w}||^2 + \sum_i C_i \xi_i \\ &\text{s.t.} \quad y_i^j \left(\langle w, \boldsymbol{x}_i^j \rangle + b\right) \geq 1 - \xi_i \\ &\qquad \xi_i \geq 0. \end{aligned} \tag{4}$$

Show that the corresponding dual problem is given by:

$$\begin{aligned} &\max W(\boldsymbol{\alpha}) = \sum_{i,j} \alpha_i^j - \frac{1}{2} \sum_{i,j,i',j'} \alpha_i^j \alpha_{i'}^{j'} y_i^j y_{i'}^{j'} \langle \boldsymbol{x}_i^j, \boldsymbol{x}_{i'}^{j'} \rangle \\ &\text{s.t.} \quad \sum_{i,j} \alpha_i^j y_i^j = 0 \\ &\qquad \alpha_i^j \geq 0 \\ &\qquad \sum_j \alpha_i^j \leq C_i \ \forall i = 1, ..., l. \end{aligned} \tag{5}$$

Exercise 6: Margin ⚡⚡

i) Show that the value of ρ of the margin for the hard-margin support vector machine is given by
$$\frac{1}{\rho^2} = ||\mathbf{w}||^2. \tag{6}$$

ii) Show that the value ρ of the margin for the hard-margin support vector machine is given by
$$\frac{1}{\rho^2} = \sum_{i=1}^n \alpha_i, \tag{7}$$
where $\boldsymbol{\alpha}$ is given by maximizing the dual representation of the maximum margin problem subject to constraints as defined in Lecture 14 on slide 18.

Exercise 7: Regression SVM ⚡⚡⚡⚡

Consider the Lagrangian of the regression support vector machine (see [1] chapter 7.1.4 on SVMs for regression):

$$\begin{aligned} L(\mathbf{w}, b, \xi_i, \hat{\xi}_i, \alpha, \hat{\alpha}) =& C \sum_{i=1}^{n} (\xi_i + \hat{\xi}_i) + \frac{1}{2} ||\mathbf{w}||^2 - \sum_{i=1}^{n} (\beta_i \xi_i + \hat{\beta}_i \hat{\xi}_i) \\ & - \sum_{i=1}^{n} \alpha_i (\epsilon + \xi_i + y(\boldsymbol{x}_i) - y_i) - \sum_{i=1}^{n} \hat{\alpha} (\epsilon + \hat{\xi}_i - y(\boldsymbol{x}_i) + y_i), \end{aligned} \tag{8}$$

(recall $y(x_i) = \mathbf{w}^\top \phi(x_i) + b$), where E_ϵ is the epsilon-insensitive error function:

$$E_\epsilon(y(x) - y) = \begin{cases} 0 & \text{if } |y(\boldsymbol{x}) - y| < \epsilon \\ |y(\boldsymbol{x}) - y| - \epsilon & \text{otherwise} \end{cases} \tag{9}$$

with the largest accepted error ϵ. We use Lagrange multipliers $\alpha, \hat{\alpha}$ for the constraints with slack variables $\xi_i, \hat{\xi}_i$:

$$\begin{aligned} y_i &\le y(\boldsymbol{x}_i) + \epsilon + \xi_i \\ y_i &\ge y(\boldsymbol{x}_i) - \epsilon - \hat{\xi}_i \end{aligned}$$

and $\beta_i, \hat{\beta}_i$ to express the positivity constraints for $\xi_i, \hat{\xi}_i$.

By setting the derivatives of the Lagrangian with respect to $\mathbf{w}, b, \xi_i$ and $\hat{\xi}_i$ to zero and then back substituting to eliminate the corresponding variables, show that the dual Lagrangian is given by

$$\begin{aligned} \widetilde{L}(\boldsymbol{\alpha}, \widehat{\boldsymbol{\alpha}}) =& -\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} (\alpha_i - \widehat{\alpha}_i)(\alpha_j - \widehat{\alpha}_j) k(\mathbf{x}_i, \mathbf{x}_j) \\ & - \epsilon \sum_{i=1}^{i} (\alpha_i + \widehat{\alpha}_i) + \sum_{i=1}^{n} (\alpha_i - \widehat{\alpha}_i) y_i. \end{aligned} \tag{10}$$

with respect to $\boldsymbol{\alpha}$ and $\widehat{\boldsymbol{\alpha}}$. The kernel is defined as $k(\mathbf{x}, \mathbf{x}') = \phi(\mathrm{x})^{\mathrm{T}} \phi(\mathrm{x}')$.

References

[1] C. M. Bishop. *Pattern recognition and machine learning.* springer, 2006.

[2] B. Schölkopf and A. J. Smola. *Learning with kernels : support vector machines, regularization, optimization, and beyond.* Adaptive computation and machine learning. MIT Press, 2002.