Machine Learning Exercise Sheet 08

SVM and Kernels

Homework

$g(\alpha)$ $(1\times N)$ $(N\times N)$ $(N\times 1)$

1×1

Q y X α

$N\times N$ $N\times 1$ $N\times M$ $N\times 1$

1 SVM

Problem 1: Explain the similarities and differences between the SVM and perceptron algorithms.

Problem 2: Recall that the dual function in the setting of the SVM training task (Slide 17) can be written as

$$g(\boldsymbol{\alpha}) = \frac{1}{2}\boldsymbol{\alpha}^T \boldsymbol{Q}\boldsymbol{\alpha} + \boldsymbol{\alpha}^T \mathbf{1}_N \,.$$

$g(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} y_i y_j \alpha_i \alpha_j x_i^T x_j$

(a) Write down the matrix $\boldsymbol{Q}$ using the vector of labels $\boldsymbol{y}$ and feature matrix $\boldsymbol{X}$. Denote the element-wise product between two matrices (in case you want to use it) by $\odot$ (also known as Hadamard product or Schur product).

$Q = yy^T XX^T$

(b) Prove that we can search for a *local* maximizer of g to find its *global* maximum (don't forget to prove properties of $\boldsymbol{Q}$ that you decide to use in this task). $\rightarrow$ prove convexity

Problem 3: Consider training a standard hard-margin SVM on a linearly separable training set of N samples. Let s denote the number of support vectors we would obtain if we would train on the entire dataset. Furthermore, let ε denote the leave-one-out cross validation (LOOCV) misclassification rate. Prove that the following relation holds:

$$\varepsilon \leq \frac{s}{N} \,.$$

$\leftarrow$ points x who lay on the hyperplane

$\nwarrow$ Number of samples

Problem 4: Load the notebook `08_homework_svm_kernels.ipynb` from Piazza. Fill in the missing code and run the notebook. Convert the evaluated notebook to pdf and add it to the printout of your homework.

Note: We suggest that you use Anaconda for installing Python and Jupyter, as well as for managing packages. We recommend that you use Python 3.

For more information on Jupyter notebooks and how to convert them to other formats, consult the Jupyter documentation and nbconvert documentation.

Upload a single PDF file with your homework solution to Moodle by 08.12.2019, 23:59pm CET. We recommend to typeset your solution (using LaTeX or Word), but handwritten solutions are also accepted. If your handwritten solution is illegible, it won't be graded and you waive your right to dispute that.

2 Kernels

Problem 5: Show that for $N \in \mathbb{N}$ and $a_i \geq 0$ for $i = 0, \dots, N$ the following function k is a valid kernel.

$$k(\boldsymbol{x}_1, \boldsymbol{x}_2) = \sum_{i=1}^{N} a_i \left(\boldsymbol{x}_1^T \boldsymbol{x}_2\right)^i + a_0, \text{ with } \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^d .$$

Problem 6: Find the feature transformation $\boldsymbol{\phi}(x)$ corresponding to the kernel

$$k(x_1, x_2) = \frac{1}{1 - x_1 x_2}, \text{ with } x_1, x_2 \in (0, 1) .$$

.

Hint: Consider an infinite-dimensional feature space.

In-Class

3 SVM

Problem 7: What is the connection between soft-margin SVM and logistic regression?

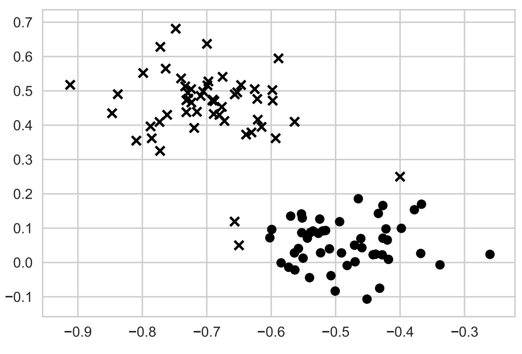
Problem 8: Consider a soft-margin SVM fitted to a linearly separable dataset $\mathcal{D}$ using the Hinge loss formulation of the optimization task.

$$\text{minimize}_{\boldsymbol{w},b} \quad \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N}\max\{0, 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b)\}$$

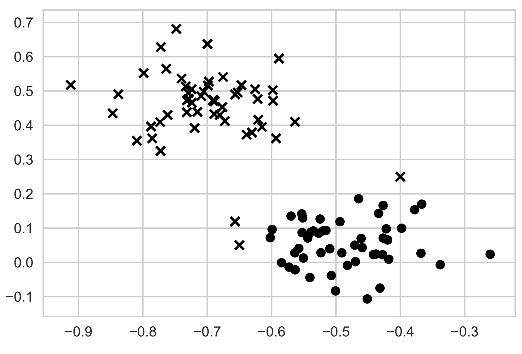
a) Is it guaranteed that all training samples in $\mathcal{D}$ will be assigned the correct label by the model?

b) Prove that if for some $C_0 \geq 0$ the resulting model classifies all training samples correctly then it will also be the case if we train the model with any larger $C \geq C_0$.

Problem 9: Sketch the decision boundary of an SVM with a quadratic kernel (polynomial with degree 2) for the data in the figure below, for two specified values of the penalty parameter C. The two classes are denoted as •'s and ×'s.



(a) $C = 10^{10}$



(b) $C = 10^{-10}$

4 Kernels

Problem 10: Consider the Gaussian kernel

$$k_{\mathrm{G}}(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp\left(-\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2}{2\sigma^2}\right), \text{ with } \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^d .$$

a) Suppose you have found a feature map $\theta : \mathbb{R}^d \to \mathbb{R}^{d_1}$ that transforms your data into a feature space in which a SVM with a Gaussian kernel works well. However, computing $\theta(\boldsymbol{x})$ is computationally expensive and luckily you discover an efficient method to compute the scalar product

$$k(\boldsymbol{x}_1, \boldsymbol{x}_1) = \theta(\boldsymbol{x}_1)^T \theta(\boldsymbol{x}_2)$$

in your feature space without having to compute $\theta(\boldsymbol{x}_1)$ and $\theta(\boldsymbol{x}_2)$ explicitly. Show how you can use the scalar product $k(\boldsymbol{x}_1, \boldsymbol{x}_2)$ to efficiently compute the Gaussian kernel $k_G(\theta(\boldsymbol{x}_1), \theta(\boldsymbol{x}_2))$ in your feature space.

b) One of the nice things about kernels is that new kernels can be constructed out of already given ones. Use the five kernel construction rules from the lecture to prove that k_G is a kernel.

Hint: Use the Taylor expansion of the exponential function to prove that $\exp \circ k_1$ *is a kernel if* k_1 *is a kernel. Also, consider* $k_2(\phi(\boldsymbol{x}_1), \phi(\boldsymbol{x}_2))$ *with the linear kernel* $k_2(\boldsymbol{x}_1, \boldsymbol{x}_2) = \boldsymbol{x}_1^T \boldsymbol{x}_2$ *and a feature map* ϕ *with only one feature.*

c) Can any finite set of points be linearly separated in the feature space of the Gaussian kernel if σ can be chosen freely?

Problem 11: Let $\mathcal{M} = \cup_{n \in \mathbb{N}} \cup_{m \in \mathbb{N}} \mathbb{R}^{n \times m}$ denote the set of all real-valued matrices of arbitrary size. Prove that the function $k : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ is a valid kernel, where

$$k(\boldsymbol{X}, \boldsymbol{Y}) = \min\{\operatorname{rank}(\boldsymbol{X}), \operatorname{rank}(\boldsymbol{Y})\} .$$

Upload a single PDF file with your homework solution to Moodle by 08.12.2019, 23:59pm CET. We recommend to typeset your solution (using LaTeX or Word), but handwritten solutions are also accepted. If your handwritten solution is illegible, it won't be graded and you waive your right to dispute that.