Machine Learning: Homework #8

Due on December 18, 2017 at 09:59am

Professor Dr. Stephan Guennemann

Marc Meissner - 03691673

Problem 1

Assume that we have a linearly separable dataset D, on which a soft-margin SVM is fitted. Is it guaranteed that all training samples in D will be assigned the correct label by the fitted model? Explain your answer.

Solution
No, it is not guaranteed. In the formula $y_i(w^T x_i + b) \geq 1 - \epsilon_i$, the slack variables ϵ_i make sure that for any given sample this condition is fulfilled, even if it is on the wrong side of the hyperplane. This means, that the resulting model can sacrifice detection accuracy on the training set for the sake of finding a hyperplane with a bigger margin. The parameter C plays a huge role in this. The higher C is, the more the soft-margin SVM will result in a hard-margin classifier.

Problem 2

Why do we need to ensure that C $>$ 0 in the slack variable formulation of soft-margin SVM? What would happen if this was not the case?

Solution
From $\alpha_i = C - \mu_i$ and dual feasability $\mu_i \geq 0$, $\alpha_i \geq 0$, it follows that:

$0 \leq \alpha_i \leq C$

This condition can not be fulfilled for $C < 0$. Intuitively, it also does not make sense to decrease the cost of a misclassified sample. This would likely lead to an erroneous or wrong solution.

Problem 3

Show that for $c \geq 0$ and $d \in N^+$ the function $K(x, y) = (x^T y + c)^d$ is a valid kernel.

Solution
Using the kernel rules that multiplications and polynomials with nonnegative coefficients preserve valid kernels:

$$K(x, y) = (x^T y + c)^d = \prod_{i=1}^{d} (x^T y + c)$$

$$k(x, y) = (z_1(x^T y) + z_0(x^T y)^0)$$

Problem 3 continued on next page...

With $z_1 = 1$ and $z_0 = c$, it follows:

$$K(x, y) = x^T y$$

, which is a valid kernel.

Problem 4

Can we directly apply this feature transformation to data? Explain why or why not!

Solution

No, an infinite feature space would require infinite computation time and storage capacity.

Problem 5

From the lecture, we know that we can express a linear classifier using only inner products of input vectors in the transformed feature space. It would be great if we could somehow use the feature space obtained by the feature transformation Φ_∞ . However, to do this, we must be able to compute the inner product of samples in this infinite feature space. We define the inner product between two infinite vectors $\Phi_\infty(x)$ and $\Phi_\infty(y)$ as the infinite sum given in the following equation:

$$K(x, y) = \sum_{i=0}^{\infty} \Phi_{\infty,i}(x)\Phi_{\infty,i}(y)$$

What is the explicit form of $K(x, y)$? (Hint: Think of the Taylor series of e^x .) With such a high dimensional feature space, should we be concerned about overfitting?

Solution

$$\begin{aligned}
K(x, y) &= \sum_{i=0}^{\infty} \Phi_{\infty,i}(x)\Phi_{\infty,i}(y) \\
&= \sum_{i=0}^{\infty} e^{-\frac{x^2}{2\sigma^2}} e^{-\frac{y^2}{2\sigma^2}} \left(\frac{x}{\sigma}\right)^i \left(\frac{y}{\sigma}\right)^i \frac{1}{i!} \\
&= e^{-\frac{x^2+y^2}{2\sigma^2}} \sum_{i=0}^{\infty} \left(\frac{xy}{\sigma^2}\right)^i \frac{1}{i!}
\end{aligned}$$

With $e^x = \sum\limits_{i=0}^{\infty} \frac{k^i}{i!}$ it follows that:

Problem 5 continued on next page. . .

$$K(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}} e^{\frac{xy}{\sigma^2}} = e^{-\frac{x^2+y^2-xy}{2\sigma^2}} = e^{-\frac{(x-y)^2}{2\sigma^2}}$$

This corresponds to a gaussian kernel. The higher the dimension of the feature space, the more prone the model is to overfitting. This has to be dealt with, e.g. by setting the hyperparameters (σ here) to a value which minimizes the cross validation error.

Problem 6

Can any finite set of points be linearly separated in the feature space defined by Φ_∞ if σ can be chosen freely?

Solution

Due to its “endless“ complexity, it can explain any statistical properties of the finite set of data points and find a space to separate them linearly in. The only exception is the case of exact copies of data points with differing class labels. However, this case is not explainable by any separation model.

Problem 7

Formulate the k-nearest neighbors algorithm in feature space by introducing the feature map $\Phi_\infty(x)$. Then rewrite the k-nearest neighbors algorithm so that it only depends on the scalar product in feature space $K(x,y) = \Phi(x)^T\Phi(y)$.

Solution

In order to classify points, a kNN classifier calculates the euclidean distance between the new and the remaining data points. Since this is done to rank points (and not for the exact value of the distance), one can use the squared metric:

$$d(x,y)^2 = ||x-y||^2$$

This is equivalent to:

$$d(x,y)^2 = x^Tx - 2x^Ty + y^Ty = K(x,x) - 2K(x,y) + K(y,y)$$