Learning with Kernels

Proseminar Data Mining

Matthieu Bulté
Fakultät für Mathematik
Technische Universität München
Email: matthieu.bulte@tum.de

***Abstract*— Text der Kurzfassung**
***Index Terms*— support vector machines, machine learning, statistics**

I. Introduction

todo

II. First steps: hard margin classifier

The hard margin classifier learns the parameters $\mathbf{w} \in \mathbb{R}^n$ and $b \in \mathbb{R}$ of a hyperplane $\mathscr{H} \subset \mathbb{R}^n$, separating two classes of observations $\mathbf{x}$ of dimension n. To train it, the algorithm takes a set of m observations $\mathbf{x_i}$ together with a vector of labels $y_i = \pm 1$ for each of the observations:

$$(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), \dots, (\mathbf{x_m}, y_m)$$

The learned parameters are then used to define a decision function f, assigning to points of $\mathbb{R}^n$ a label ± 1 corresponding to the side of the hyperplane on which the points stands:

$$f(\mathbf{x}) = sgn(\langle \mathbf{w}, \mathbf{x} \rangle + b) \tag{1}$$

A. Margin maximization

One particularity about the learning algorithm used for Support Vector Machines is the loss function the algorithm tries to optimize. Other learning algorithms typically chose a loss function based on the empirical risk, defined as following for a function f and a training set $(\mathbf{x}, \mathbf{y})$ of size m:

$$R_{emp}[f] = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{2} |f(\mathbf{x_i}) - y_i| \tag{2}$$

The margin maximization algorithm instead, searches within the space of hyperplanes properly classifying the training set, for the hyperplane with the largest distance from the training points to the hyperplane. This can be translated in the following constrained optimization problem:

NOTE: before jumping into equations, maybe explain the reason why we want to maximize the margin. argument using the noise around sample.

$$\begin{aligned} \underset{\mathbf{w} \in \mathbb{R}^n, b \in \mathbb{R}}{\text{maximize}} \quad & M := \frac{1}{\|\mathbf{w}\|} \underset{i}{min}\ y_i(\langle \mathbf{w}, \mathbf{x_i} \rangle + b) \\ \text{subject to} \quad & y_i(\langle \mathbf{w}, \mathbf{x_i} \rangle + b) \geq M \text{ for all } i = 1 \dots m \end{aligned} \tag{3}$$

Assuming that a solution to this optimization problem exisits, it still has the issue that although the solution hyperplane is unique, there is an inifity of parameter vectors $\|\mathbf{w}\|$ resulting in the solution hyperplane, only differing in their length. Uniqueness of $\mathbf{w}$ can be ensured by adding to its length the follwoing contraint:

$$M\|\mathbf{w}\| = 1$$

We can thus reformulate the optimization problem into the following quadratic optimization problem which is known to have a numerical solution:

$$\begin{aligned} \underset{\mathbf{w} \in \mathbb{R}^n, b \in \mathbb{R}}{\text{minimize}} \quad & \|\mathbf{w}\| \\ \text{subject to} \quad & y_i(\langle \mathbf{w}, \mathbf{x_i} \rangle + b) \geq 1 \text{ for all } i = 1 \dots m \end{aligned} \tag{4}$$

B. Support vectors

Although the previous formulation of the problem helped us to better understand the ideas of support vector machines, the resulting quadratic problem will become impractical when later introducing kernels methods. We will introduce in this section an equivalent formulation of the optimization problem (4)can latex handle references instead of hard coding? that will not only solve computational issues, but will also give us more insights about the resulting solution.

In order to get to these benefits, we introduce the Lagrangian L together with Lagrange multipliers $\alpha_i \geq 0$:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{m} \alpha_i [y_i(\langle \mathbf{w}, \mathbf{x_i} \rangle + b)) - 1] \tag{5}$$

This is called the dual formulation of our original optimization problem, called the primal problem, in which $\mathbf{w}$ and b are called primal variables, and $\boldsymbol{\alpha}$ the dual variable. It can be shown ref? that the promal optimization problem is equivalent to finding a saddle point of L, minimizing with respect to $\mathbf{w}$ and b, while maximizing it with respect to $\boldsymbol{\alpha}$.

Because the solution of this dual problem is a saddle point, thus an extremum with respect to all variables, following equalities must hold:

$$\frac{\partial}{\partial b} L(\mathbf{w}, b, \boldsymbol{\alpha}) = 0, \quad \frac{\partial}{\partial \mathbf{w}} L(\mathbf{w}, b, \boldsymbol{\alpha}) = 0 \tag{6}$$

leading to

$$\sum_{i=1}^{m} \boldsymbol{\alpha}_i y_i = 0 \tag{7}$$

and

$$\begin{aligned} \mathbf{w} - \sum_{i=1}^{m} \alpha_i y_i \mathbf{x_i} &= 0 \\ \Rightarrow \mathbf{w} &= \sum_{i=1}^{m} \alpha_i y_i \mathbf{x_i} \end{aligned} \tag{8}$$

Further, by replacing (7) and (8) into the Lagrangian, we obtain an optimization problem free of primal variables:

$$\begin{aligned} \underset{\boldsymbol{\alpha} \in \mathbb{R}^n}{\text{maximize}} \quad & W(\boldsymbol{\alpha}) = \sum_{i=1}^{m} \boldsymbol{\alpha}_i - \frac{1}{2} \sum_{i,j=1}^{m} \boldsymbol{\alpha}_i \boldsymbol{\alpha}_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \\ \text{subject to} \quad & \boldsymbol{\alpha} \geq 0 \text{ for all } i = 1 \dots m \\ & \sum_{i=1}^{m} \boldsymbol{\alpha}_i y_i = 0 \end{aligned} \tag{9}$$

Finally, the Karush-Kuhn-Tucker theorem states that the solution $\boldsymbol{\alpha}$ satisfies the following equality for all $i = 1 \dots m$

$$\boldsymbol{\alpha}_i [y_i (\langle \mathbf{w}, \mathbf{x_i} \rangle + b) - 1] = 0 \tag{10}$$

The importance of this equality is twofold. First, it allows us to compute b from an observation i for whichs $\boldsymbol{\alpha}_i \neq 0$. Second and most importantly, this equality implies that every observations not lying on the margin must have a vanishing Lagrangian coefficient. These observations are called *support vectors*.

III. Non linearly separable

A. Noisy training data

B. Non linear target boundary

IV. Kernel methods

A. The kernel trick

B. Some useful kernels

V. Zusammenfassung und Ausblick

blabla