PERGAMON

Pattern Recognition 35 (2002) 2927–2936

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Linear programming support vector machines

Weida Zhou*, Li Zhang, Licheng Jiao

Key Laboratory for Radar Signal Processing, Xidian University, Xi'an, 710071, People's Republic of China

Received 3 May 2001; accepted 30 October 2001

Abstract

Based on the analysis of the conclusions in the statistical learning theory, especially the VC dimension of linear functions, linear programming support vector machines (or SVMs) are presented including linear programming linear and nonlinear SVMs. In linear programming SVMs, in order to improve the speed of the training time, the bound of the VC dimension is loosened properly. Simulation results for both artificial and real data show that the generalization performance of our method is a good approximation of SVMs and the computation complex is largely reduced by our method. © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Statistical learning theory; VC dimension; Support vector machines; Generalization performance; Linear programming

1. Introduction

Since the 1970s', Vapnik et al. have applied themselves to the study of statistical learning theory [1–3]. Until the early of the 1990, a new kind of learning machines, support vector machine (SVM), was presented based on those theories [2,4,5]. The main study of statistical learning theory is the model of learning from examples, which can be described as: there are l random independent identically distributed examples $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_l, y_l)$; $((\mathbf{x}, y) \in (R^n, R))$ drawn according to the uniform probability distribution $P(\mathbf{x}, y)$, $(P(\mathbf{x}, y) = P(\mathbf{x})P(y \mid \mathbf{x}))$. Given a set of functions $f(\mathbf{x}, \alpha), \alpha \in \Lambda$ (where Λ is a parameter set) from which the goal of learning from examples is to select a function $f(\mathbf{x}, \alpha_0)$ that can express the relationship between $\mathbf{x}$ and y in the best possible way. In general, in order to obtain $f(\mathbf{x}, \alpha_0)$, one has to minimize the expected risk functional

$$R(\alpha) = \int L(y, f(\mathbf{x}, \alpha))P(\mathbf{x}, y)\,\mathrm{d}\mathbf{x}\,\mathrm{d}y, \tag{1.1}$$

where $L(y, f(\mathbf{x}, \alpha))$ measures the loss between the response y to a given input $\mathbf{x}$ and the response $f(\mathbf{x}, a)$ provided by the learning machine. The learning problems such as the problems of pattern recognition, regression estimation and density estimation may be taken as the same learning models with different loss functions [2]. In this paper, we only deal with the problem of pattern recognition. Consider the following loss function

$$L(y, f(\mathbf{x}, \alpha)) = \begin{cases} 0 & \text{if } y = f(\mathbf{x}, \alpha), \\ 1 & \text{if } y \neq f(\mathbf{x}, \alpha). \end{cases} \tag{1.2}$$

Due to the probability distribution $P(\mathbf{x}, y)$ in Eq. (1.1) is unknown, the expected risk functional is replaced by the empirical risk functional

$$R_{emp}(\alpha) = \frac{1}{l}\sum_{i=1}^{l} L(y_i, f(\mathbf{x}_i, \alpha)). \tag{1.3}$$

In order to know the quality of the empirical risk $R_{emp}(\alpha)$ to approximate the expected risk, Vapnik presented the following bound theorem [2]. With probability at least $1 - \eta$ $(0 \leqslant \eta \leqslant 1)$, the inequality

$$R(\alpha) \leqslant R_{emp}(\alpha) + \frac{\varepsilon}{2}\left(1 + \sqrt{1 + \frac{4R_{emp}(\alpha)}{\varepsilon}}\right) \tag{1.4}$$

holds true. Where $\varepsilon = 4(h(\ln(2l/h) + 1) - \ln\eta)/l$ and h is the VC dimension of the set of functions $f(\mathbf{x}, \alpha)$, $\alpha \in \Lambda$.

[*] Corresponding author. Fax: +86-29-823-6159.
E-mail address: zhouwd@rsp.xidian.edu.cn (W. Zhou).

PII: S0031-3203(01)00210-2

From Eq. (1.4), we can see that the minimization of the expected risk $R(\alpha)$ is equal to the minimization of the two terms on the right-hand side of Eq. (1.4) at the same time. The first term on the right of Eq. (1.4) $R_{emp}(\alpha)$ is minimized by learning process. The second term varies with the VC dimension h and the number of examples l. The smaller the VC dimension h and the larger the number of examples l, the smaller the value of the second term. In fact, the number of examples is finite. So for the case of a small example set, the minimization of the expected risk is implemented by minimizing the empirical risk and the VC dimension. Generally speaking, a complex target function set or a large hypothesis space is required for minimizing the empirical risk. But a small hypothesis space is requested for minimizing the VC dimension of the target function set. Therefore, the minimization problem is in a dilemma, the best solution of the problem is to take a compromise between them.

Now, restrict the target function to the linear function. Similar to the set of Δ-margin separating hyperplanes defined in Ref. [3], we define the set of m_Δ-margin separating hyperplanes. Let us denote the target functions set by $f(\mathbf{x},\mathbf{w},b)=\mathbf{w}^{\mathrm{T}}\mathbf{x}+b$. If these functions classify an example $\mathbf{x}$ as follows:

$$y=\begin{cases} 1, & \mathbf{w}^{\mathrm{T}}\mathbf{x}+b\geqslant\Delta,\\ -1, & \mathbf{w}^{\mathrm{T}}\mathbf{x}+b\leqslant-\Delta, \end{cases}\qquad \Delta\geqslant 0 \tag{1.5}$$

then the set $f(\mathbf{x},\mathbf{w},b)=\mathbf{w}^{\mathrm{T}}\mathbf{x}+b$ is called the set of m_Δ-margin separating hyperplanes whose margin is

$$m_\Delta=\frac{\Delta}{||\mathbf{w}||_2}, \tag{1.6}$$

where $||\cdot||_2$ denotes l_2-norm, namely Euclidean distance. There is a conclusion about the VC dimension of the set of m_Δ-margin separating hyperplanes. Let vectors $\mathbf{x}\in X$ belong to a sphere of radius R. Then the set of m_Δ-margin separating hyperplanes has the VC dimension h bounded by the inequality:

$$h\leqslant\min\left(\left[\frac{R^2}{m_\Delta^2}\right],n\right)+1, \tag{1.7}$$

where n is the dimension of input space. From Eq. (1.7), we can see that if the VC dimension of the target functions set h is $<n$, then h varies inversely with the margin m_Δ^2. In this way, $f(\mathbf{x},\mathbf{w}_0,b_0)$ can be approached by minimizing the empirical risk functional and maximizing the separating margin m_Δ, which is the structural risk in SVMs introduced by Vapnik:

$$R_{structure}(\alpha)=CR_{emp}(\alpha)+\frac{1}{m_\Delta^2}, \tag{1.8}$$

where the constant $C>0$ is a parameter chosen by the users and α is a parameter of the target function set. For l random independent identically distributed examples $(\mathbf{x}_1,y_1),(\mathbf{x}_2,y_2),\ldots,(\mathbf{x}_l,y_l);((\mathbf{x},y)\in(R^n,R))$, the linear SVMs for pattern recognition have the following optimization problem [4]:

$$\min\quad \frac{1}{2}||\mathbf{w}||_2^2+C\sum_{i=1}^{l}\xi_i, \tag{1.9}$$

$$\text{s.t.}\quad y_i((\mathbf{w}\cdot\mathbf{x}_i)+b)\geqslant 1-\xi_i,$$

$$\xi_i\geqslant 0,\quad i=1,\ldots,l,$$

where $(\cdot,\cdot)$ denotes the inner product. Minimizing the first term in Eq. (1.9) $\frac{1}{2}||\mathbf{w}||_2^2$ plays the role of controlling the capacity of the learning machine and avoiding the overfitting of the machine. While minimizing the second term is to minimize the empirical risk. The Wolfe dual programming of Eq. (1.9) is [4]

$$\max\quad W(\alpha)=\sum_{i=1}^{l}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_j y_i y_j(\mathbf{x}_i\cdot\mathbf{x}_j), \tag{1.10}$$

$$\text{s.t.}\quad \sum_{i=1}^{l}\alpha_i y_i=0, \tag{1.11}$$

$$\alpha_i\in[0,C],\quad i=1,\ldots,l. \tag{1.12}$$

The decision function has the following form:

$$f(\mathbf{x})=\sum_{i=1}^{l}\alpha_i y_i(\mathbf{x}_i\cdot\mathbf{x})+b \tag{1.13}$$

and

$$y=\operatorname{sgn}(f(x)). \tag{1.14}$$

The kernel functions are introduced in linear SVMs, which leads to nonlinear SVMs [4]:

$$\max\quad W(\alpha)=\sum_{i=1}^{l}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_j y_i y_j K(\mathbf{x}_i\cdot\mathbf{x}_j) \tag{1.16}$$

$$\text{s.t.}\quad \sum_{i=1}^{l}\alpha_i y_i=0 \tag{1.17}$$

$$\alpha_i\in[0,C],\quad i=1,\ldots,l. \tag{1.18}$$

And then the decision function can be written as

$$f(\mathbf{x})=\sum_{i=1}^{l}\alpha_i y_i K(\mathbf{x}_i\cdot\mathbf{x})+b. \tag{1.19}$$

In a nutshell, the theory foundation of SVMs (statistical learning theory) is rather perfect. But training a SVM requires the solution of a quadratic programming (QP) optimization that is not easy to implement, in particular for large-scale problems. Based on the statistical learning theory, the linear programming SVMs that are extremely simple without explicitly solving QP problems are proposed.