Enhancing Generalization Capability of SVM Classifiers with Feature Weight Adjustment

Xizhao Wang and Qiang He

College of Mathematics and Computer Science,
Hebei University, Baoding 071002, Hebei, China
wangxz@mail.hbu.edu.cn

Abstract. It is well recognized that support vector machines (SVMs) would produce better classification performance in terms of generalization power. A SVM constructs an optimal separating hyper-plane through maximizing the margin between two classes in high-dimensional feature space. Based on statistical learning theory, the margin scale reflects the generalization capability to a great extent. The bigger the margin scale takes, the better the generalization capability of SVMs will have. This paper makes an attempt to enlarge the margin between two support vector hyper-planes by feature weight adjustment. The experiments demonstrate that our proposed techniques in this paper can enhance the generalization capability of the original SVM classifiers.

1 Introduction

Statistical learning theory (SLT) [1], a new theory for small-sample learning problems, was introduced in the 1960's by Vapnik et al, which can deal with the situation where the samples are limited.

Most machine-learning methods perform empirical risk minimization (ERM) induction principle [1], which is effective when samples are enough. However, in most cases of real world, the samples are limited. It is difficult to apply expected risk minimization [2] directly to classification problems. Most expected risk minimization problems are converted to minimize the empirical risk. Unfortunately empirical risk minimization is not always equivalent to expected risk minimization. It implies that ERM cannot lead to a good generalization capability but expected risk minimization can. The statistical learning theory has shown a clear relationship between expected risk and empirical risk, and shown that the generalization can be controlled by the capacity of learning machine.

Support vector machines (SVMs) are a new classification technique based on SLT [2]. Due to its extraordinary generalization, SVMs have been a powerful tool for solving classification problems with two classes. A SVM first maps the original input space into a high-dimensional feature space through some predefined nonlinear mapping and then constructs an optimal separating hyper-plane maximizing the margin between two classes in the feature space. Based on SLT, we know that, the bigger the margin scale takes, the better the generalization capability of SVMs will have. Therefore we always expect that the margin is as large as possible for two-class problems.

M.Gh. Negoita et al. (Eds.): KES 2004, LNAI 3213, pp. 1037–1043, 2004.

Feature weight learning, which assigns a weight to each feature to indicate the importance degree of feature, is an extension of feature selection. This paper adopts a feature weight learning technique introduced in [4]. We expect to enlarge the margin between two support vector hyper-planes by using this technique for generalization capability improvement.

The rest of this paper is organized as follows. In section 2, we review the relationship between the margin of two hyper-planes and the generalization capability of SVMs. In section 3, the detailed technique of feature weight learning is introduced. Two experiments, which show that the feature weight learning's effect to margin enlargement, are presented in section 4. Section 5 gives some remarks and concludes our paper.

2 Relationship Between the Margin and the Generalization Capability

Firstly, we consider a function, i.e., the growth function of the set of indicator functions [2]

$$G^{\Lambda}(l) = \ln \sup_{z_1,\cdots,z_l} N^{\Lambda}(z_1, z_2, \cdots, z_l) \tag{1}$$

where $N^{\Lambda}(z_1, z_2, \cdots, z_l)$ evaluates how many different separations of the given sample $z_1, z_2, \cdots, z_l$ can be done using functions from the set of indicator functions. From SLT, we know the following conclusion: any growth function satisfies

$$G^{\Lambda}(l) = l \ln 2 \text{ or } G^{\Lambda}(l) < h(\ln \frac{l}{h} + 1) \tag{2}$$

where h is an integer for which

$$G^{\Lambda}(h) = h \ln 2 \quad and \quad G^{\Lambda}(h+1) \neq (h+1)\ln 2 \tag{3}$$

We now review the definition of VC dimension [3]: The VC dimension of the set of indicator function $Q(z,\alpha), \alpha \in \Lambda$ equal h if the growth function is bounded by a logarithmic function with coefficient h. VC dimension is a pivotal concept in the statistical learning theory.

For a two-class classification problem, the Δ-margin separating hyper-plane is defined in SLT [1]. The following theorems are valid for the set of Δ-margin separating hyper-planes [2].

Theorem 1. Let vector $x \in X$ belong to a sphere of radius R. then the set of Δ-margin separating hyper-plane has the VC dimension h bounded by the inequality

$$h \leq \min\left(\left[\frac{R^2}{\Delta^2}\right], n\right) + 1 \tag{4}$$

From inequality (4), we know that larger the margin of the set of functions is, the more less VC dimension is.

Theorem 2. With probability at least $1-\eta$, the inequality

$$R(\alpha) \le R_{emp}(\alpha) + \frac{B\varepsilon}{2}\left(1+\sqrt{1+\frac{4R_{emp}(\alpha)}{B\varepsilon}}\right) \tag{5}$$

holds true, where $R(\alpha)$, $R_{emp}(\alpha)$ are the expected risk and the empirical risk functional respectively [1], l is the size of training set, h is VC dimension of the set of functions. ε is formulated in [2]. For simplicity, we write (5) as follow:

$$R(\alpha) \le R_{emp}(\alpha) + \Phi\left(h/l\right) \tag{6}$$

Φ is called confidence interval, which is monotonic increasing function of h, and monotonic decreasing function of l.

Inequality (6) gives bounds on the generalization ability of learning machine. Our purpose is to minimize the left side of (6), i.e., $R(\alpha)$. Minimization of $R(\alpha)$ implies the optimal generalization capability of learning machines. In fact, minimizing $R(\alpha)$ is not feasible due to its integral formulation. Practically, in place of minimizing $R(\alpha)$, we do minimize $R_{emp}(\alpha)$. Inequality (6) shows the relationship between $R(\alpha)$ and $R_{emp}(\alpha)$. Due to the existence of the second term of the right side of (6), it is clear that the minimization of $R_{emp}(\alpha)$ cannot guarantee the minimization of $R(\alpha)$. It is expected that the second term of the right side of (6) is as small as possible since the small confidence interval possibly implies the small actual risk $R(\alpha)$, i.e., possibly implies a good generalization capability.

Noting that Φ is a monotonic increasing function of h which is decreasing with the increase of Δ where Δ is the margin of separating hyper-planes (see Inequation (4)), enlarging Δ possibly leads to an improvement of generalization capability of the learning machine.

The traditional ERM principle only minimizes empirical risk without considering confidence interval, so it perhaps cannot get good generalization capability. SVMs based on SLT perform structural risk minimization (SRM) induction principle [1][2], not only minimizes the empirical risk, but also considers confidence interval by maximizing the margin between two classes in high-dimensional feature space.

3 Feature-Weight Learning

Each feature is considered to have an importance degree called feature-weight. Feature-weight assignment is an extension of feature selection. The latter has only either 0-weight or 1-weight value, while the former can have weight values in the interval [0,1].

The feature-weight learning depends on the similarity between samples. There are many ways to define the similarity measure, such as the related coefficient and Euclidean distance, etc. Here the similarity measure ρ_{ij} is defined as follows:

$$\rho_{ij}^{(w)} = \frac{1}{1+\beta * d_{ij}^{(w)}} \tag{7}$$

where β is a positive parameter in the interval [0,1]. It can adjust the similarity degrees distributed around 0.5, i.e., β is required to satisfy the following:

$$\frac{2}{n(n-1)} \sum_{p<q} \frac{1}{1+\beta * d_{ij}} = 0.5 \tag{8}$$

where d_{ij} is commonly used Euclidean distance, and $d_{ij}^{(w)}$ is the weighted Euclidean distance defined as follows:

$$d_{ij}^{(w)} = \sqrt{\left(\sum_{k=1\cdots s} w_k^2 (x_{ik} - x_{jk})^2 \right)} \tag{9}$$

where $w = (w_1, \cdots, w_s)$ is the feature-weight vector. Its component is the importance degree corresponding to each feature. Larger w_k is, more important the $k-th$ feature is. When $w = (1, \cdots, 1)$, the space $\{ \left\| d_{ij}^{(w)} \right\| \le r \}$ is a hyper-sphere with radius r (called original space) and $d_{ij}^{(w)}$ is d_{ij} and $\rho_{ij}^{(w)}$ is ρ_{ij}. When $w \ne (1, \cdots, 1)$, it means that the axes would be extended or shrunk in accordance with w_k. Thus the space $\{ \left\| d_{ij}^{(w)} \right\| \le r \}$ is hyper-ellipse, called the transformed space. The lower the value of w_k is, the higher the flattening extent is. A good partition should have the following property: the samples within one cluster are more similar and dissimilar samples are more separate, which implies that the fuzziness of the partition is low. Therefore we hope that by adjusting w, $\rho_{ij}^{(w)}$ tends to one or zero if ρ_{ij} is greater or less than 0.5, respectively.

Based on above discussion, we learn the feature-weight value by minimizing an evaluation function $E(w)$ [4] defined as follows:

$$E(w) = \frac{2}{n(n-1)} \sum_i \sum_{j \ne i} \frac{1}{2} \left(\rho_{ij}^{(w)} (1 - \rho_{ij}) + \rho_{ij} (1 - \rho_{ij}^{(w)}) \right) \tag{10}$$

where n is the number of samples.

From equation (10), we can say that $E(w)$ decreases as the similarity degree $\rho_{ij}^{(w)}$ tends to 0 or 1 if $\rho_{ij} < 0.5$ or $\rho_{ij} > 0.5$. Therefore we expect that feature-weight learning by minimizing $E(w)$ can lead to $\rho_{pq}^{(w)}$ close to 0 or 1, which obviously can improve the performance of FCM.

To minimize the evaluation function $E(w)$, we use the gradient descent technique. First of all, let Δw_k be the change of w_k, compute as follows:

$$\Delta w_k = -\eta \frac{\partial E(w)}{\partial w_k} \tag{11}$$

where η is the learning rate. An appropriate value of η could speed up the convergence of the algorithm since too small η leads to low computational efficiency but too big η results in divergence of the algorithm. Through one dimensional searching technique [5], η is determined by:

$$\begin{aligned} & E(w_1 - \eta \frac{\partial E(w)}{\partial w_1}, \ldots, w_n - \eta \frac{\partial E(w)}{\partial w_n}) \\ & = Min_{\lambda > 0} E(w_1 - \lambda \frac{\partial E(w)}{\partial w_1}, \ldots, w_n - \lambda \frac{\partial E(w)}{\partial w_n}) \end{aligned} \tag{12}$$

The derivate of $E(w)$ can obtained from following:

$$\frac{\partial E(w)}{\partial w_k} = \frac{1}{N(N-1)} \sum_{j<i} (1-2\rho_{ij}) \frac{\partial \rho_{ij}^{(w)}}{\partial d_{ij}^{(w)}} \frac{\partial d_{ij}^{(w)}}{\partial w_k} \tag{13}$$

$$\frac{\partial \rho_{ij}^{(w)}}{\partial d_{ij}^{(w)}} = \frac{-\beta}{(1+\beta \cdot d_{ij}^{(w)})^2} \tag{14}$$

$$\frac{\partial d_{ij}^{(w)}}{\partial w_k} = \frac{w_k (x_{ik} - x_{jk})}{d_{ij}^{(w)}} \tag{15}$$

The algorithm is described briefly as follows:

(1) Initialize all weight values with 1. And solve β from equation (8);

(2) Compute $\rho_{ij}^{(w)}$ by equation (7) and $d_{ij}^{(w)}$ by equation (9).

(3) Let Δw_k be the change of w_k. Compute $\Delta w_k = -\eta \frac{\partial E(w)}{\partial w_k}$ by (13-15);

(4) If $1 \geq w_k + \Delta w_k \geq 0$ is satisfied, then $w_k = w_k + \Delta w_k$;

(5) Go to (3) until $E(w)$ is less than a given threshold or the times of iteration reach the user specified number.

4 Enlarging the Margin by Feather Weight Adjustment for Generalization Capability Improvement

In this section, for the purpose of generalization capability improvement, we would like to experimentally demonstrate the enlargement of margin between two hyperplanes by the feature weight adjustment mentioned in previous section. We select two databases to complete the demonstration.

The first one is the well-known toy example, i.e., Iris database from UCI, which includes 150 cases with 3 classes. Since the SVM is only for two-class classification problems, we delete the cases belonging to class one and the remaining 100 cases with two classes are used to demonstrate. The second one, called Pima India Diabetes, is also from UCI. The two databases' characters are shown in table 1. According to the feature weight learning algorithm given in the end of section 3, we can learn the weights for the selected two databases. Table 2 shows the result of feature weight leaning for the two selected databases.

Applying SVM Toolbox (http://www.isis.ecs.soton.ac.uk/isystems/kernel/svm.zip) to the original data of the two selected databases, one can obtain the optimal separating hyper-planes and the corresponding margin. Due to the limit of paper length, we omit the formulation of hyper-planes and margins. For details, one can refer to [1][2]. Similarly, using the same method, one can evaluate the margins among the separating hyper-planes after the learned weighs are incorporated into the original databases. Tables 3 and 4 show the size of margin of separating hyper-planes, the training accuracy, and the testing accuracy for the two databases, where 70% of the databases are randomly selected as the training sets and the remaining 30% as the testing sets. It is worth noting that the experimental results depend on the parameters chosen in the SVM Toolbox.

From Tables 3 and 4, one can see that the margins are indeed enlarged. However, the improvement for training and testing accuracy is not significant. We speculate that the reason is that (1) the data is not enough for Iris database and (2) Pima database is non-linear separable very much. The further investigation to the generalization capability improvement is in progress.

Table 1. The characters of databases

Database Name	Number of samples	Number of features	Category of features
Iris	100	4	Numerical
Pima	668	8	Numerical

Table 2. The results of feature-weight learning

Database Name	Results of feature-weight learning
Iris	0.0001，0.0002，1.0，0.164
Pima	0.410686,0.000000,0.000000,0.000000, 1.000000,0.001334,0.986665,0.000000

Table 3. Margin enlargement for Iris database

Iris database	Margin	Training Accuracy	Testing Accuracy
Before feature weight adjustment	0.11136	0.98561	0.92856
After feature weight adjustment	0.16365	0.96666	0.93686

Table 4. Margin enlargement for Pima database

Pima database	Margin	Training Accuracy	Testing Accuracy
Before feature weight adjustment	0.38258	0.61412	0.6638
After feature weight adjustment	0.98609	0.68585	0.6638

5 Conclusions

According to SLT, the enlargement of margin of separating hyper-plane can enhance the generalization capability of the learning machine. This paper makes an attempt to enlarge the margin by an approach of feature weight adjustment. Initial experiments show the approach's effectiveness. We have the following remarks:

(1) Whether the feature weight learning approach can be mathematically proved to enlarge the margin?
(2) Whether the approach has a difference between linear separable and non-linear separable data sets?
(3) When evaluating the margin of separating hyper-plane, how to choose the optimal values of parameters in the quadratic program?

Acknowledgement

This research is supported by the key project of Chinese Educational Ministry (03016), the Natural Science Foundation of HeBei Province (603136), and the Doctoral Foundation of Hebei Educational Committee (B2003116)

References

1. Vapnik V. N, The Nature of statistical learning theory, New York: Springer-Verlag, 1995.
2. Vapnik V. N, Statistical learning theory. New York: A Wiley-Interscience Publication, 1998.
3. Vladimir N. Vapnik, An Overview of Statistical Learning Theory, IEEE Transactions on Neural Networks, 10, (1999) 988-999.
4. Basak J, De R. K, Pal S. K, Unsupervised feature selection using a neuro-fuzzy approach, Pattern Recognition Letters 19 (1998) 996-1006.
5. Rao, Optimization theory and applications (Wiley eastern limited, 1985).