Statistical learning theory and its application to pattern recognition

Ling Zhang* and Bo Zhang
Sate Key Lab of Intelligent Technology and Systems
Tsinghua University, Beijing, China
*Artificial Intelligence Institute, Anhui University, Anhui, China

ABSTRACT

The problem of pattern recognition is formulated as a classification in the statistic learning theory. Vapnik constructed a class of learning algorithms called support vector machine (SVM) to solve the problem. The algorithm not only has strong theoretical foundation but also provides a powerful tool for solving real-life problems. But it still has some drawbacks. Two of them are (1) the computational complexity of finding the optimal separating hyperplane is quite high in the linearly separable case, and (2) in the linearly non-separable case, for any given sample set it's hard to choose a proper nonlinear mapping (kernel function) such that the sample set is linearly separable in the new space after the mapping. To overcome these drawbacks, we presented some new approaches. The main idea and some experimental results of the approaches are presented.

Keywords: statistic learning, pattern recognition, support vector machine, covering algorithms

1. INTRODUCTION

Vapnik presented a new statistical learning theory recently [1]-[5]. The theory deals with the rather general problem, i.e., the problem of choosing the desired dependence on the basis of empirical data. The pattern recognition, either speech, image or language recognition, etc., can be regarded as a classification problem. It is involved in the general problem above, learning from data. The general model of the pattern recognition can be stated as follows. For simplicity, we only consider the two-class classification problem.

Given the training set

$$K=\{(x^1,y^1),\ldots,(x^m,y^m)\},\qquad x^i\in R^n, y^i\in\{-1,1\}.$$

Where the probability distribution function *F(x, y)* is unknown. The problem is to construct a learning algorithm that minimizes the probability of a classification error based on the given data.

Now we consider the learning problem in neural networks.

A given sample set

$$\cdot K=\{(x^i,y^i), i=1,2,\ldots,m\},\qquad y^i\in\{1,-1\}. \tag{1}$$

Assume a three-layer feed-forward neural network N, *x* is its input, *y* is its output. A set of actuation functions of neurons is as follows.

Image Extraction, Segmentation, and Recognition, Tianxu Zhang, Bir Bhanu, Ning Shu, Editors, Proceedings of SPIE Vol. 4550 (2001) © 2001 SPIE · 0277-786X/01/$15.00

$$y = f\left(\sum_{j=1}^{n} w_j x_j - \theta\right), \quad f(x) \text{ is a sign function.} \qquad (2)$$

The learning problem in neural networks is to construct a procedure for choosing from this set an approximating function using the samples such that if vector x^i belongs to the first class $y^i = 1$ and if vector x^i does not belong to the first class $y^i = -1$. So the learning principle of neural networks is the same as that of the general learning problem (or classification problem).

Vapnik constructed a class of learning algorithms called support vector machine (SVM) to solve the classification problem by using the optimal separating hyperplane. The SVM theory can briefly be stated as follows: If a given sample set is linearly separable, the learning problem is to find the optimal (maximal margin) separating hyperplane by using some optimization technique; If a given sample set is linearly non-separable, the original input vectors x are mapped in high dimensional feature space by a kernel function such that the images of the input vectors in the new space are linearly separable. Then, the problem can be solved in the same way as the linearly separable case. These ideas provide a new tool for solving real-life problems. But there are drawbacks in the SVM algorithms. First, in the linearly separable case, the computational complexity is generally quite high for finding the optimal separating hyperplane. In the linearly non-separable case, for any given sample set it's generally hard to find the proper kernel function. Our research works on these issues and some of its applications are discussed below.

2. FINDING THE OPTIMAL SEPARATING HYPERPLANE

When the sample set K is linearly separable, the goal of SVM is to construct a hyperplane that separates vector x into two classes and has the maximal margin. Vapnik transferred the problem into a quadratic optimization problem, i.e., finding the saddle point of a Lagrange function by using Lagrange multipliers method. But generally, to find the optimal separating hyperplane with the maximal margin, its computational complexity is quite high based on the Lagrange multipliers method. In order to improve the computational complexity, the problem was transformed as follows. From formulas (1) and (2), we know that if the sample set is linearly separable then the following inequalities have solutions (see [6] for more details).

$$\sum_{i=1}^{n} (w_i x_i^k - \theta) y^k > 0$$
$$w = (w_0, w_1, \ldots, w_n), \qquad w_0 = \theta$$
$$x(k) = (x_0^k, x_1^k, \ldots, x_n^k), \qquad x_0^k = -1$$
$$c^k = x(k) y^k$$

Where $x^k = (x_1^k, x_2^k, \ldots, x_n^k)$ is the input vector.

$w = (w_0, w_1, \ldots, w_n)$ is the weight-threshold vector.

Let

$$C = \begin{bmatrix} c^1 \\ c^2 \\ \dots \\ c^m \end{bmatrix}$$

Then, the learning problem, i.e., finding the weight-threshold vector *w*, of neural networks under training sample set *K* can be equivalently transferred into that of finding the *s* of the following inequality system.

$$Cs > 0$$
$$s = (s_0, s_1, \dots, s_n)$$

A specific solution of the above inequality system is one of its several feasible solutions. If one of the performances of neural networks, for example, the generalization capacity, is taken as an objective function and the above inequalities are regarded as constraint conditions, the learning problem can be transformed into some sort of programming (optimization) problems as follows.

Problem 1

Objective: $\max\limits_{s} \min\limits_{i} \{< s, c^i >\}$, where <*x*, *y*> is the inner product of *x* and *y*.

Constraint:

$$Cs > 0, \quad |s| = 1$$
$$C = \begin{bmatrix} c^1 \\ c^2 \\ \dots \\ c^m \end{bmatrix}$$

Problem 1 is a quadratic programming problem and can equivalently be transformed into the following problem 2 and 3.

Problem 2

Let **C** be the convex closure of $\{c^i$, simply the convex closure of samples. Find s^* such that

$$|s^*| = \min_{s \in C} |s|$$

Problem 3

Find λ such that $\min\limits_{\lambda \in R} \lambda^T Q \lambda$

Where $R = \left\{ \lambda \middle| \lambda_i \geq 0, \sum_{i=1}^{m} \lambda_i = 1 \right\}$ and Q is a positive semi-definite matrix.

Problem 2 is a geometric representation of problem 1. By using the geometric intuition of problem 2, some well-known methods such as simplex algorithms can be used to develop some efficient algorithms. We have presented a learning algorithm called iterative simplex algorithm with the polynomial complexity by using this idea.

Problem 3 is an algebraic form of problem 2. Some well-known programming techniques can be used to deal with the problem. For example, in [6] the potential reduction algorithm [7] was used to construct a programming based algorithm with the polynomial complexity.

Therefore, the computational complexity for finding the optimal separating hyperplane is reduced by these new algorithms.

3. KERNEL FUNCTION APPROACH

When the given sample set is linearly non-separable, the idea of the SVM is the following: It maps the input vector *x* into a high-dimensional feature space Z through some nonlinear mapping, chosen a priori. In this space, an optimal separating hyperplane is constructed. The advantage of the kernel function approach presented by Vapnik is that for constructing the optimal separating hyperplane in the feature space Z, one does not need to consider the high-dimensional feature space in explicit form. One only has to deal with a kind of kernel functions defined in the original space. The problem is how to find a proper kernel function such that in the new space the sample set is linearly separable and the separating hyperplane constructed generalizes well. To this end, we presented a new approach called covering algorithms [8][9].

In [8][9], the learning problem (classification problem) of neural networks was transferred into that of the point set covering in the input space of samples. Assume that $K = \{K_1, K_2\}$, $K = \{x^1, x^2, .., x^m$, is a set of input vectors of training samples, a point set in an *n*-dimensional space. Assume that *K* is classified into two classes K_1, K_2. From form (2), it's known that a neuron can be considered as a hyperplane below in the input space geometrically.

$$P : f\left(\sum_{j=1}^{n} w_j x_j - \theta\right) = 0 \qquad (3)$$

It divides the space into two half-spaces. When the input vector falls in the positive half-space, the corresponding output will be 1, otherwise –1. Based on the geometric interpretation, a constructive learning approach may be given. But when the number of hyperplanes (neurons) increases, the geometric intuition will lose, since each hyperplane extends infinitely. To this end, in our approach all points of *K* are projected upward to a hyper-sphere *SP,* where *SP* is a hyper-sphere with radius $r > \max_i |x^i|$ in *(n+1)*-dimensional space. Then the intersection between sphere *SP* and hyperplane *P* is a localized sphere neighborhood or a localized covering C_i. Each covering C_i represents a neuron. Due to the geometric intuition, based on the given sample set a neural network can easily be constructed as follows. That is, a set $C = \{C_1, ..., C_m\}$ of coverings is constructed such that each C_i only covers the points in one of $K_i, i = 1,2$ and the union of C_i covers the whole *K*. The *C* is called a classification cover set of *K*, since each $C_i \in C$ represents the classification (K_1 or K_2) of the input vectors covered by C_i. A neural network performing the classification $K = \{K_1, K_2\}$ can be constructed by using the set *C* of coverings. Then, the learning problem (classification) in neural networks is transformed into the covering problem of input vectors. In the covering algorithm, the classification is performed by several separating hyperplanes in the *(n+1)*-dimensional sphere space rather than by only one separating hyperplane in the high dimensional feature space in SVM. But they are equivalent. The advantages of the covering algorithm are: (1) for any given sample set, the samples can always be separated without classification error by using several separating planes; (2) since the algorithm is performed in rather low dimensional (i.e., *n+1*) space, the separating planes that generalize well can easily be constructed by using the geometric intuition. However, since the dimensionality of the feature space is huge, in the SVM it's hard to choose a proper kernel function to guarantee lower classification error and higher generalization ability.

Let C be a classification cover set. If any C_i is deleted from C $C/\{C_i\}$ will no longer become a classification cover set, then *C* is called a classification cover set without redundant. Based on the approach presented in [7], the following theorem can be proved.

Theorem 1: Assume that $C = \{C_1, ..., C_m\}$ is a classification cover set of *K* without redundant, the first *t* coverings cover K_1 and the last *m-t* coverings cover K_2. Let $D_i = C_i \cap K_1, i = 1, ..., t$, $D_i = C_i \cap K_2, i = t+1, ..., m$. Set E_i is a non-empty subset of D_i and E_i do not intersect with each other. There must exist a kernel function with $E = \cup E_i$ as its support vectors.

The theorem indicates that any subset of input vectors of the given sample set may become the support vectors through some nonlinear mapping (or kernel function). Since the set of support vectors changes with different kernel functions arbitrarily, it means that support vectors do not always represent the essential classification property of the given sample set. Therefore, to construct a separating hyperplane that separates the training data and has maximal margin in the high dimensional feature space Z does not necessarily generalize well. It depends on what type of nonlinear mappings Z is used. Since the distance measure defined in the feature space is quite different from that defined in the original space, choosing a proper kernel function or a proper nonlinear mapping Z is very important in the SVM. To this end, we have proposed the new covering algorithms above. Its basic idea is the following.

Assume that a given sample set is divided into two classes. In the original input vector space (or *n+1*-dimensional sphere space) of the given sample set, finding the representative vectors (called separating points) of the boundary between two classes is easier than in the high dimensional feature space. Then, by using the separating points, a proper mapping Z can be constructed. The covering algorithms just benefit from processing in the lower dimensional space.

3. APPLICATIONS

3.1 Handwritten Chinese character recognition [10]

The number (categories) of characters: 100-700. Each character has 256 features. Each character has 70 training samples and 60 testing samples. Total training samples are 45,000, when the categories are 700. The experiments were implemented by the covering algorithm in Celetron-300 PC.

Written Chinese character Recognition by using the covering learning algorithm

Categories	Training Time (sec.)	Testing Correct Rate %	Training Correct Rate %
100	645	97.0	100
300	5481	95.5	100
500	16786	94.5	100
700	31089	93.6	100

From the above results, we can see that the covering algorithm can always guarantee the training correct rate 100%.

3.2 Content based image retrieval [11][12]

The content-based image retrieval is to retrieve the relevant images from an image database by a query. Given a query represented by an image, all images in the database are ranked according to their distance to the query. The images that are relevant to the query should be ranked top in the list. Relevance feedback is an interactive technique in the retrieval process. It allows users to evaluate the retrieval results and indicate which images he or she thinks are relevant to the query. The images marked as relevant ones are called positive images. Then, the human feedback information is used to retrieve the images again. The relevance feedback process can be carried out several times. Since the user is involved as part of the retrieval process, the retrieval performance can be improved. We transfer the relevance feedback process into a learning problem of neural networks. Each positive image marked is considered as a new training sample to train a neural network. The trained neural network is used to retrieve new relevant images. By using the generalization ability of the neural network, more relevant images can be found. In order to show the efficiency of the new approach, two learning algorithms, i.e., the covering learning algorithm and the SVM, are used. The experimental results shown below are carried out under the following conditions. A color image database has 9918 images, including animals, landscapes, figures, buildings, etc. Each image has 256 features (color auto-correlogram: RGB quantification 4*4*4=64, distance={1,3,5,7}).

The relevance feedback by using the covering learning algorithm

Learning Times	0	1	2	3	4	5
Recall*	0.407	0.563	0.634	0.642	0.643	659
Avg-r	1056.2	756.3	777.9	887.8	920.3	946.8
Avg-p	0.234	0.507	0.620	0.612	0.634	0.627

The relevance feedback by using the SVM algorithm

Learning Times	0	1	2	3	4	5
Recall*	0.407	0.637	0.706	0.733	0.733	0.743
Avg-r	1056.2	1508.7	1035.4	1392.8	867.5	1372.5
Avg-p	0.234	0.606	0.689	0.724	0.676	0.717

Assume that $Q_1,\ldots,Q_q$ are q query (images). For each $Q_\imath$, a_i is the number of the retrieval correct images, $I_1^{(i)},\ldots,I_{a_i}^{(i)}$. And $rank(I_\jmath^{(i)})$ is the ranking of image $I_\jmath^{(i)}$.

Where Recall*=Recall vs. Scope=$\left|\left\{I_\jmath^{(i)} \middle| rank\left(I_\jmath^{(i)}\right)\le S\right\}\right|/a_i$, S is a given integral, the scope.

Avg-r=Average r-measure=$\frac{1}{q}\sum_{\imath=1}^{q}\frac{1}{a_i}\sum_{\jmath=1}^{a_i} rank(I_\jmath^{(i)})$, the average ranking of the retrieval correct images.

Avg-p=Average p measure=$\frac{1}{q}\sum_{\imath=1}^{q}\frac{1}{a_i}\sum_{\jmath=1}^{a_i}\frac{j}{rank(I_\jmath^{(i)})}$, if all a_i retrieval correct images rank in the top a_i images, then Avg-p=1.

From the results, it can be seen that two learning approaches have similar performances, but the computational complexity of the SVM is much higher than that of the covering algorithm.

ACKNOWLEDGMENTS

This work was supported by the National Key Basic Research Program (G1998030509) and the Doctoral Foundation (98000335).

REFERENCES

1. V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag, New York, 1995.
2. V. N. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, Inc., New York, 1998.
3. A. Vidgasagar, A Theory of Learning and Generalization, Springer, New York, 1997.
4. V. Vapnik, S. Golowich, and A. Smola, "Support vector method for function approximation, regression estimation, and signal processing," In *Advances in Neural Information Processing Systems* 9, pp281-287, Cambridge, MA: MIT Press, 1997.
5. B. Scholkopf, S. Mika, et al, "Input space versus feature space in kernel-based method," IEEE Trans. on Neural Networks, vol.10, no.5, pp.1000-1017, 1999.
6. Bo. Zhang, Ling. Zhang, "Programming based learning algorithm of neural networks with self-feedback connections," *IEEE Trans. on Neural Networks*, vol.6, no.3, pp.771-775, 1995.
7. P. M. Pardalos, Y. Ye and C. Han, "Algorithms for the solution of quadratic knapsack problems," Linear Algebra and its Applications, vol.152, pp.69-92, July 1991
8. Ling. Zhang, Bo. Zhang, "A geometrical representation of McCulloch-Pitts neural model and its applications," *IEEE Trans. on Neural Networks,* vol.10, No.4, pp.925-929, July 1999.
9. Zhang Ling, Zhang Bo, and Yin Haifeng, "An alternative design algorithm of multi-layer neural networks," *Chinese Journal of Software*, vol.10, no.7, pp.737-742, 1999.
10. Mingrui Wu, Bo Zhang and Ling Zhang, "A neural network based classifier for Handwritten Chinese character

recognition," Proc. Of 2000 *International Conference on Pattern Recognition*, vol.2, pp.561-564, Bercelona, Spain, Sept. 2000.

11.Lei Zhang, Fuzong Lin and Bo Zhang, "A content based image retrieval (CBIR) method based on color-spatial feature," *IEEE Region 10 Annual International Conference 1999*, pp.166-169, Cheju, Korea, 1999.

12.Lei Zhang, Fuzong Lin and Bo Zhang, "Support vector machine learning for image retrieval," *IEEE International Conference on Image Processing* , Thessalonoki, Greece, October 2001 (to be appeared).