Journal of Universal Computer Science, vol. 13, no. 10 (2007), 1462-1470
submitted: 12/6/06, accepted: 24/10/06, appeared: 28/10/07 © *J.UCS*

An Improved SVM Based on Similarity Metric

Chaoyong Wang
(College of Computer Science and Technology, Jilin University
Key Laboratory of Symbol Computation and Knowledge Engineering of
Ministry of Education, Changchun 130012, China, and
Department of Fundamental Sciences
Jilin Teacher's Institute of Engineering and Technology
Changchun 130021, China
dynasty1188@126.com)

Yanfeng Sun
(College of Computer Science and Technology, Jilin University
Key Laboratory of Symbol Computation and Knowledge Engineering of
Ministry of Education, Changchun 130012, China
sunyf@jlu.edu.cn)

Yanchun Liang
(College of Computer Science and Technology, Jilin University
Key Laboratory of Symbol Computation and Knowledge Engineering of
Ministry of Education, Changchun 130012, China
ycliang@jlu.edu.cn)

Abstract: A novel support vector machine method for classification is presented in this paper. A modified kernel function based on the similarity metric and Riemannian metric is applied to the support vector machine. In general, it is believed that the similarity of homogeneous samples is higher than that of inhomogeneous samples. Therefore, in Riemannian geometry, Riemannian metric can be used to reflect local property of a curve. In order to enlarge the similarity metric of the homogeneous samples or reduce that of the inhomogeneous samples in the feature space, Riemannian metric is used in the kernel function of the SVM. Simulated experiments are performed using the databases including an artificial and the UCI real data. Simulation results show the effectiveness of the proposed algorithm through the comparison with four typical kernel functions without similarity metric.

Key Words: Support vector machine, Riemannian metric, Similarity metric

Category: H.3.7, H.5.4

1 Introduction

Support vector machine (SVM) is a novel machine learning method based on statistical learning theory. SVM is a powerful tool for solving problems with small samples, nonlinearities, high dimension and local minima. The theory of support

vector machines was first introduced by Vapnik and was developed from the theory of the structural risk minimization [Vapnik 1995, Cortes and Vapnik 1995, Cherkassky and Mulier 1998]. In recent years, SVM has been used in many applications successfully, such as pattern recognition, regression analysis, function approximation and signal processing, etc.

Currently, the study on SVM theory concentrates mainly on the following three aspects: the first aspect is that the classical algorithms for SVM are modified to achieve higher computation speed and expand application scope; the second one is that the kernel function is modified to improve the performance of a SVM classifier, and the last one is that the decision functional forms are simplified for SVM. The work in this paper is related to the second topic.

The study on the kernel function is mostly keeping a watchful eye on how to improve its classifying ability of a given kernel function. Steinwart et al. [Steinwart 2001, Cristianini et al. 2000] have studied deeply on the properties of the kernel function. Because of their outstanding work on this subject, their idea becomes almost the main aspect of the current study on the kernel function. But a given kernel function only reflects the inner product measure of the input samples in the feature space. Considering the same measure, there might be many different types of mapping. Therefore, what degree can the properties of the kernel function be analyzed? Is it enough to solve the problem of the kernel function selection based on the analysis? It is worthy to considering that whether the corresponding kernel function can be selected according to different data set in stead of the present method in which the given kernel function does not have any changed to be used to all the data set. At present, the theory of the kernel function selection isn't enough to be used in guiding the selection of the kernel function. In practical, different kernel functions will be tried out according to the given data set. And then the kernel function having the best test result will be selected as the kernel function in SVM. But improving the kernel function based on the properties of kernel function is effective, as we will demonstrate in the following sections.

2 Similarity Metric

In the field of machine learning, what we need to do in most cases is to derive some rules and to discover knowledge from the relational database. Similarity should be considered here. For instance, in a classification problem let the training set be

$$T = \{(x_1, y_1), \ldots, (x_l, y_l)\} \in (X \times Y)^l,$$

where $x_i \in X \subset R^n$, $y_i \in Y = \{-1, 1\}$, $i = 1, 2, \ldots, l$.

A decision function will be derived from the above training samples, and then the corresponding output y should be predicted for an input x according to

this decision function to decide its class label. An intuitionistic idea to solve this problem is to estimate which samples the input x is similar with, the positive or negative samples, and then to decide the corresponding class that it belongs to. If the input x is similar to the positive samples, the corresponding output y should be $+1$, or else-1. This idea is reasonable because the similar inputs lead to the same output values. For this reason, it would be necessarily to describe the similarity metric.

The similarity of points in space R^n can be measured by the distance and cosine of two vectors, i. e.,

$$\|x - x^{'}\| = \sqrt{(x \cdot x) - 2(x \cdot x^{'}) + (x^{'} \cdot x^{'})}, \tag{1}$$

and

$$\cos\beta = \frac{(x \cdot x^{'})}{\|x\| \cdot \|x^{'}\|} = \frac{(x \cdot x^{'})}{\sqrt{(x \cdot x) \cdot (x^{'} \cdot x^{'})}}. \tag{2}$$

Thus it can be seen that we could directly use the inner product of vectors as a similarity measure.

In general, the form of the SVM decision function turns out to be

$$f(x) = \sum_{i \in SV} \alpha_i y_i K(x_i, x) + b. \tag{3}$$

From Eq. (3), we can realize that the final output value of the SVM decision function is only dependent on the inner product in the transformed Hilbert space.

$$K(x_i, x_j^{'}) = (\Phi(x_i) \cdot \Phi(x_j^{'})),\ i, j = 1, 2, \ldots, l. \tag{4}$$

SVM performs a nonlinear mapping of the input vector from the input space $X = R^n$ into a higher dimensional Hilbert space, where the mapping is determined by the kernel function. Selecting different kernel functions or different mapping and the corresponding Hilbert space is equivalent to selecting the different forms of the inner product. It means that the similarity is estimated using different criterions. Thus it can be seen that the nature of the classification problem is a similarity problem. In this sense, we can regard support vector machine as an algorithm for solving similarity problem.

For the samples of homogeneous input, their corresponding output samples should have greater similarity, namely the similarity of the homogeneous samples is stronger than that of the inhomogeneous ones. Thus we can compute a probability p_i for each possible output label i given any new input sample x, which can be performed by taking the weighted average of the known correct outputs of a number of nearest neighbors [Lowe 1995, Cover and Hart 1967].

Let n_j be the weight that is assigned to each of the J ($e.g. J = 15$) nearest neighbors, and s_{ij} be the known output probability (usually 0 or 1) for label i

of each neighbor. Then, we obtain a positive scalar function

$$c(x) = e^{p_i} = exp(\frac{\sum_{j=1}^{J} n_j s_{ij}}{\sum_{j=1}^{J} n_j}) \geq 0, \tag{5}$$

where the weight n_j assigned to each neighbor is determined by a Gaussian kernel centered at x

$$n_j = exp(-d_j^2/2\sigma^2), \tag{6}$$

where d_j is the distance of the neighbor from x

$$d_j^2 = \sum_k (x_k - c_{jk})^2, \tag{7}$$

where k is the dimension of the input sample, and c_j is the input location of each neighbor.

3 Riemannian Metric

In this section we introduce the Riemannian metric [Palais and Terng 1988, Schölkopf et al. 1999]. Riemannian metric is a second-order symmetrical non-degenerate tensor used to measure the distance and angle in a metric space. When a local coordinate system is selected, metric tensor can be expressed in the form of a matrix, denoted as G . The symbol g_{ij} denotes the component of the metric tensor traditionally (that is the matrix elements). From the geometrical point of view, the mapping $\Phi(x)$ defines an embedding of S into F as a curved submanifold. When F is a Euclidean or Hilbert space, a Riemannian metric is thereby induced in the space S , where the length of a small line element dx is defined by the length in the larger space F.

Denote z as the mapped sample in the feature space, i.e., $z = \varphi(x)$. A small vector dx is mapped to

$$dz = \nabla\varphi \cdot dx = \sum_i \frac{\partial}{\partial x_i} dx_i \tag{8}$$

where $\nabla\varphi = (\frac{\partial}{\partial x_i}\varphi(x))$. The squared length of $dz = dz_\alpha$ is written in the quadratic form as

$$\|dz\|^2 = \sum_\alpha (dz_\alpha)^2 = \sum_{i,j} g_{ij}(x) dx_i dx_j \tag{9}$$

where

$$g_{ij}(x) = (\frac{\partial}{\partial x_i}\varphi(x)) \cdot (\frac{\partial}{\partial x_j}\varphi(x)), \tag{10}$$

and the $n \times n$ positive definite matrix $G(x) = (g_{ij}(x))$ is the Riemannian metric tensor induced in S. It can be obtained directly from the kernel

$$G_{ij}(x) = \frac{\partial}{\partial x_i}\frac{\partial}{\partial x_j}K(x, x^{'})|_{x=x'} \tag{11}$$

There are some typical kernel functions. One is radial,

$$K(x, x^{'}) = f(x - x^{'}) \tag{12}$$

which includes the Gaussian RBF kernel,

$$K(x, x^{'}) = e^{-\|x-x^{'}\|/2\sigma^2} \tag{13}$$

The other is the function of the inner product

$$K(x, x^{'}) = f(x \cdot x^{'}) \tag{14}$$

which includes the linear kernel $K(x, x^{'}) = x \cdot x^{'}$, the polynomial kernel of degree d $K(x, x^{'}) = (1 + x \cdot x^{x})^d$, and the multi-layer perceptron kernel $K(x, x^{'}) = \tan(K(x \cdot x^{'}) + \theta)$.

The Riemannian metric for the first case is given by

$$\begin{aligned} g_{ij}(x) &= \frac{\partial}{\partial x_i}\frac{\partial}{\partial x_j}K(x, x^{'}) \\ &= -\delta_{ij}f^{'}(\frac{1}{2}\|x - x^{'}\|^2)|_{x=x'} \\ &\quad -f^{''}(\frac{1}{2}\|x - x^{'}\|^2)(x_i - x_i^{'})(x_j - x_j^{'})\|_{x=x'} \\ &= -f^{'}(0)\delta_{ij} \end{aligned} \tag{15}$$

In particular for the Gaussian RBF kernel, we have

$$g_{ij}(x) = \frac{1}{\sigma^2}\delta_{ij} \tag{16}$$

The metric for the inner product case can be calculated in a similar way, and is given by

$$g_{ij}(x) = f^{'}(0)\delta_{ij} + x_i x_j f^{''}(0) \tag{17}$$

The volume form in a Riemannian space is defined as

$$dV = \sqrt{g(x)}dx_1 \cdots dx_l \tag{18}$$

where $g(x) = det|g_{ij}(x)|$. The factor $\sqrt{g(x)}$ represents how a local area is magnified or reduced in F under the mapping $\Phi(x)$[Wu et al. 2002].

4 Improving Kernel Function Based on the Similarity Metric

In Section 3, some knowledge on the kernel function and similarity metric are described. Now, introducing the similarity metric function Eq. (5) into the kernel function, we could have a new kernel function as follows [Wu et al. 2002]

$$\tilde{K}(x, x^{'}) = c(x)c(x^{'})K(x, x^{'}) \tag{19}$$

Lemma 1. (*Positivity Condition for Mercer kernels [Cristianini et al. 2000]*) *A kernel* $K : R^n \times R^n \rightarrow R$ *: is a Mercer kernel if and only if the matrix* $[K(x_i, x_j)] \in R^{n \times n}$ *is positive semi-definite for all choices of points* $x_1, x_2, \cdots, x_n \subset X$ *(*X *is a compact subset of* R^n*) and all* $n = 1, 2, \cdots$.

Theorem 2. *Let* X *be a compact subset of* R^n. $\forall x \in X$, *a positive scalar function* $c(x)$ *and a kernel function* $K(x, x^{'})$ *on* $X \times X$, *the function* $\tilde{K}(x, x^{'}) = c(x)c(x^{'})K(x, x^{'})$ *is Mercer kernel.*

Proof Since $K(x, x^{'})$ is a kernel function on $X \times X$, there exists a mapping Φ from X to the Hilbert space H, subject to

$$K(x, x^{'}) = (\Phi(x) \cdot \Phi(x^{'})). \tag{20}$$

For $x_1, x_2, \cdots, x_l \in X$, construct Gram matrix

$$(K_{ij})_{i,j=1}^{l} = (K(x_i, x_j))_{i,j=1}^{l} \tag{21}$$

by using kernel $K(\cdot, \cdot)$ on $x_1, x_2, \cdots, x_l$. Because $c(x)$ is a positive scalar function, according to Eq. (20), $\forall \alpha_1, \alpha_2, \cdots, \alpha_l$, we have

$$\begin{aligned}
&\sum_{i,j} \alpha_i \alpha_j \tilde{K}(x_i, x_j) \\
&= \sum_{i,j} \alpha_i \alpha_j (c(x_i)\Phi(x_i)) \cdot (c(x_j)\Phi(x_j)) \\
&= (\sum_{i} \alpha_i c(x_i)\Phi(x_i)) \cdot (\sum_{j} \alpha_j c(x_j)\Phi(x_j)) \\
&= \| \sum_{i} \alpha_i c(x_i)\Phi(x_i) \|^2 \\
&\geq 0
\end{aligned}$$

It is shown that the Gram matrix of $\tilde{K}(x, x^{'})$ with respect to $x_1, \cdots, x_l \in X$ is positive semi-definite. According to the positivity condition for Mercer kernel, $\tilde{K}(x, x^{'})$ is Mercer kernel.

For the Riemannian metric of the new kernel function (19) and Eq. (11), we have

$$\tilde{g}_{ij}(x) = c^2(x)g_{ij}(x) + c_i(x)c_j(x) + 2c_i(x)c(x)K_i(x, x) \tag{22}$$

where $c_i(x) = \partial c(x)/\partial x_i$ and $K_i(x,x) = \partial K(x,x^{'})/\partial x_i|_{x=x'}$.

For the RBF kernel, we have $K_i(x,x) = 0$. For the linear kernel, we have $K_i(x,x) = x_i$. For the polynomial kernel, we have $K_i(x,x) = dx_i$. For the Sigmoid kernel, we have $K_i(x,x) = kx_i \sec^2\theta$. Therefore, if choosing $c(x)$ in this way, we can really achieve the purpose of enlarging the similarity metric of homogeneous samples.

In summary, the main steps of the proposed method can be stated as follows:

1. Find the similarity metric for any training set according to Eqs. (5), (6) and (7);
2. Modify the kernel according to Eq. (19);
3. Train SVM using the modified kernel;
4. Find the similarity metric for any testing set according to Eqs. (5), (6) and (7);
5. Examine the testing samples using the trained SVM and similarity metric.

5 Simulated Experiments

In order to evaluate the performance of the proposed method, we performed simulations on artificial data and UCI standard data set.

Firstly, an artificial two-dimensional data set is used in our method. All samples in the data set are uniformly distributed in the region $[-1,1]\times[-1,1]$. The two classes are separated by a nonlinear boundary determined by $y = \cos(3x)$, as shown in Figure 1.

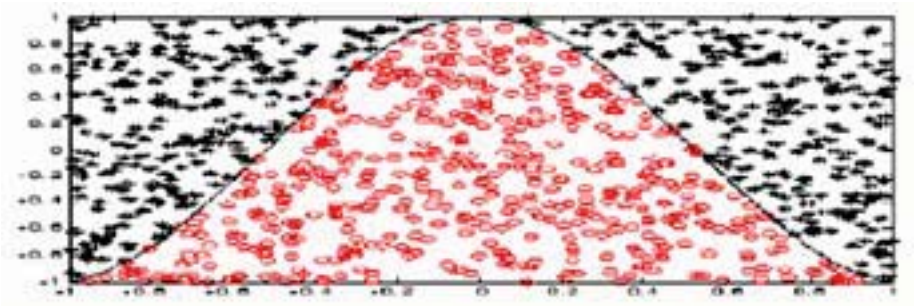

Figure 1: A two-dimensional artificial data set.

In the simulation, there are four training sets where each has 100 samples and one test set having 400 samples. They are generated randomly and uniformly for

each classification problem. The performance of SVM is measured by the test correct rate. Simulation results are displayed in Table 1.

Table 1: Comparison of the testing ratios(%) using artificial data

Test	Line		Poly		RBF		Sigmoid	
	Prototype	Improved	Prototype	Improved	Prototype	Improved	Prototype	Improved
1	67.25	72.00	72.75	85.50	62.00	65.25	67.25	68.25
2	64.75	77.25	76.00	80.25	89.75	93.00	71.50	72.25
3	74.00	81.75	70.25	74.00	90.50	94.00	74.50	82.00
4	63.25	66.00	64.00	64.25	92.00	93.25	69.75	67.00
Average	67.31	74.25	70.75	76.00	83.56	86.38	70.75	72.38

From Table 1, it can be seen that different kernel functions have great influence on the same testing data set. The performance of the modified SVM is improved obviously compared with the original method. The testing correct rate increases by 7%, 7%, 4% and 2% for linear kernel, polynomial kernel, RBF kernel and Sigmoid kernel, respectively.

In order to further illustrate the effectiveness of the proposed method, we selected 4 data sets from the UCI standard data set, which are Iris, Cancer, Sonar and Wdbc respectively. Because the quantity of samples in some of UCI data set is less relatively, we randomly draw 50 data as the training sample and 100 data as the test sample. Table 2 summaries the simulation results.

Table 2: Comparison of the testing ratios(%) using the UCI data set

Kernel	Iris		Cancer		Sonar		Wdbc	
	Prototype	Improved	Prototype	Improved	Prototype	Improved	Prototype	Improved
Poly	97.0	99.00	94.0	96.0	58.0	60.0	68.0	84.0
RBF	99.0	99.00	99.0	99.0	90.0	92.0	96.0	98.0
Linear	92.0	94.00	96.0	96.0	60.0	60.0	60.0	78.0
Sigmoid	94.0	96.00	98.0	98.0	60.0	62.0	60.0	86.0
Average	95.5	97.0	96.8	97.3	67.0	68.5	71.0	86.5

Through analyzing the simulation results in Table 2, we can see that the performance of our method is better in the selected data set. About 5% of test correct rate is increased in the data sets of Iris, Cancer, Sonar and Wdbc, respectively. However, the test results for data sets of Sonar and Wdbc are not ideal except for the RBF kernel.

6 Conclusions

In this paper, a novel method of modifying a kernel function is proposed to improve the performance of the support vector machine classification. The theory of modifying a kernel function is based on the Riemannian metric and similarity metric. The main idea is to enlarge or reduce the similarity among samples so as to increase the homogeneous samples' similarity measure and to increase the ability of classification. Kernel function modifying is performed on four classical kernel functions with artificial data and UCI standard dataset. Simulation results show the effectiveness of the proposed algorithm.

Acknowledgements

The authors are grateful to the support of the National Natural Science Foundation of China (60673023, 60433020), the science technology development project of Jilin Province of China (20050705-2), the doctoral funds of the National Education Ministry of China (20030183060), the graduate innovation lab of Jilin University (503043), 985 project of Jilin University of China, and the support of the European Commission under grant No. TH/Asia Link/010 (111084).

References

[Cherkassky and Mulier 1998] Cherkassky V., Mulier F.: "Learning from Data: Concepts, Theory and Methods"; John Wiley and Sons (1998)

[Cortes and Vapnik 1995] Cortes C., Vapnik V.: "Support Vector Networks"; Machine Learning, 20, (1995), 273-297

[Cover and Hart 1967] Cover T. M., Hart P. E.: "Nearest Neighbour Pattern Classification"; IEEE Transactions on Information Theory, 13, (1967), 21-27

[Cristianini et al. 2000] Cristianini N., Shawe-Taylor J.: "An Introduction to Support Vector Machines and other Kernel-based Learning Methods"; Cambridge University Press, Cambridge (2000)

[Lowe 1995] Lowe D. G.: "Similarity Metric Learning for a Variable-Kernel Classifier"; Neural Computation, 7, (1995), 72-85

[Palais and Terng 1988] Palais R. S., Terng C. L.: "Critical Point Theory and Submanifold Geometry"; Lecture Notes in Mathematics, 1353, Springer-Verlag, Berlin (1988)

[Schölkopf et al. 1999] Schölkopf B., Burges C. J. C., Smola A. J.: "Advances in Kernel Methods C Support Vector Learning"; MIT Press, Cambridge, MA (1999)

[Steinwart 2001] Steinwart I.: "On the Influence of the Kernel on the Generalization Ability of Support Vector Machines"; Journal of Machine Learning Research, 2, (2001), 67-93

[Vapnik 1995] Vapnik V.: "The Nature of Statistical Learning Theory"; Springer-Verlag, New-York (1995)

[Wu et al. 2002] Wu S., Amari S.: "Conformal Transformation of Kernel Function: A Data-Dependent Way to Improved Support Vector Machine Classifiers"; Neural Processing Letters, 15, (2002), 59-67