Kernel Projection Algorithm for Large-Scale SVM Problems

WANG Jiaqi (王家琦), TAO Qing (陶 卿) and WANG Jue (王 珏)

Institute of Automation, The Chinese Academy of Sciences, Beijing 100080, P.R. China

E-mail: wang_jq@yeah.net; wangj@iamail.ia.ac.cn

Received December 28, 2001; revised May 28, 2002.

Abstract Support Vector Machine (SVM) has become a very effective method in statistical machine learning and it has proved that training SVM is to solve Nearest Point pair Problem (NPP) between two disjoint closed convex sets. Later Keerthi pointed out that it is difficult to apply classical excellent geometric algorithms directly to SVM and so designed a new geometric algorithm for SVM. In this article, a new algorithm for geometrically solving SVM, Kernel Projection Algorithm, is presented based on the theorem on fixed-points of projection mapping. This new algorithm makes it easy to apply classical geometric algorithms to solving SVM and is more understandable than Keerthi's. Experiments show that the new algorithm can also handle large-scale SVM problems. Geometric algorithms for SVM, such as Keerthi's algorithm, require that two closed convex sets be disjoint and otherwise the algorithms are meaningless. In this article, this requirement will be guaranteed in theory by using the theoretic result on universal kernel functions.

Keywords SVM, NPP, MNP, feature mapping, projection, fixed-point, universal kernel

1 Introduction

Recently, Support Vector Machine (SVM) has become a very effective method in statistical machine learning[1]. Firstly, SVM is based on the statistical learning theory and minimizes structural risk rather than the empirical one. Secondly, the maximal margin algorithm is given, which is only concerned with the operation of dot product. Finally, the kernel method is applied to solve nonlinear problems. Then training SVM is converted into finding the solution of Quadratic Programming (QP).

However, the classical algorithms about QP are too slow to handle large-scale problems. Although Sequential Maximal Optimization (SMO) can solve SVM very fast, it is still necessary to explore geometric explanation and algorithms for SVM because there have been many intuitive geometric algorithms for large-scale problems. While QP conceals the nature of problems, the geometric method for learning problems may be regarded as a vehicle for understanding profound ideas and more efficient methods can be easily explored.

In 1996, Bennett proved that training SVM is to solve the Nearest Point pair Problem (NPP) between two disjoint closed convex sets[2]. Later this idea was also given in [3–5]. Keerthi pointed out that it is difficult to apply the classical Gilbert algorithm, which does well in the Minimal Normal Problem (MNP), to solving NPP. In fact, MNP is the special case of NPP and thus solving MNP is simpler. Keerthi designed a new algorithm for NPP motivated by the Gilbert algorithm. Compared with SMO, Keerthi's algorithm is intuitive and understandable.

In this article, we point out that the solution of linear SVM is a fixed-point of projection mapping by using the result in [6]. Based on this idea and Swap algorithm in [6], solving SVM is further converted into doing MNP. Then the Gilbert algorithm can be easily applied into solving SVM and so the difficulty pointed out by Keerthi is overcome. Our algorithm is more understandable than Keerthi's because it is only required to understand Gilbert's algorithm and the concept of fixed-point. Some experiments show that our algorithm can also handle large-scale SVM problems.

This research is supported by the NKBRSF of China (Grant No.G1998030508), the National Natural Science Foundation of in China (Grant No.60175032) and the Pilot Program of the Knowledge Innovation Project of Chinese Academy of Sciences.

The geometric algorithms for SVM require that two closed convex sets be disjoint, and otherwise the algorithms are meaningless. This requirement cannot be guaranteed when Keerthi combines his algorithm on NPP with the classical kernel function for nonlinear SVM. By using the theorems in topology and functional analysis, we have proved that some kernel functions are universal, namely, correctly classifying two arbitrary sample sets[12]. In this article, this theoretical result will be explained geometrically and by using universal kernel functions the requirement mentioned above would be guaranteed in theory such that the geometric algorithms for SVM become reasonable.

In all, the primary contributions of this article are:

1) A new way to geometrically solve SVM is pioneered by using the concept of fixed-points of projection mapping such that many excellent geometric algorithms, such as Gilbert algorithm, can be applied to SVM.

2) Based on the analysis of universal kernel functions, it is guaranteed that two closed convex sets are disjoint in theory. Therefore the geometric algorithms for SVM become reasonable.

In addition, we will define Relative Maximal Margin in this article since we find that the computing cost is related to it rather than Maximal Margin. But Maximal Margin is regarded as the benchmark of computation cost in [3]. The nature of support vectors will also be more deeply understood. Support vectors are trivial for linear SVM, but crucial for nonlinear SVM. Support vectors representing the classifier are not unique while the classifier is unique.

Table 1

Notations	Meaning of notations
R^d	d-dimensional Euclidean space
F	Feature space
P, Q	Point sets in sample space
I, J	Index sets of P and Q respectively and $\|I\| = n$, $\|J\| = m$
U, V	Bounded, closed and convex subsets of R^d
$\hat{U}$, $\hat{V}$	Bounded, closed and convex subsets of F
Z	Minkowski set difference of U and V, $Z = U - V \equiv \{u - v : u \in U, v \in V\}$
π_{UV}	$\pi_U \circ \pi_V$
π_U, π_V	Projection operators on U and V respectively
$\pi_{\hat{U}\hat{V}}$	$\pi_{\hat{U}} \circ \pi_{\hat{V}}$
$\pi_{\hat{U}}$, $\pi_{\hat{V}}$	Projection operators on $\hat{U}$ and $\hat{V}$ respectively
$d(\cdot, \cdot)$	Euclidean distance

The remainder of this paper is arranged as follows. Geometric interpretation of linear SVM is given in Section 2. Section 3 introduces the Projection algorithm combining the Swap algorithm with the NPP algorithm. Section 4 is devoted to the definition of feature mappings and kernel functions and the explanation of polynomial and Gaussian Radius Basis Function (RBF) kernel. The Kernel Projection algorithm, three experiments and some discussions will be given in Sections 5 and 6. Conclusions are summarized in the last section. Notation, which will be used in this paper, can be found in Table 1.

2 Linear SVM

Linear SVM is a hyperplane that separates positive samples from negative samples with Maximum Margin. In the linear case, the margin is defined by the distance between the nearest positive and negative samples to the hyperplane (See [7] and left figure in Fig.1).

Maximizing the margin can be expressed by the following optimization problem

$$\min \frac{1}{2}\|\omega\|^2 \quad \text{s.t. } \forall i \in I \cup J, y_i(\langle \omega, p_i \rangle + b) \geq 1. \tag{1}$$

This optimization problem can be converted

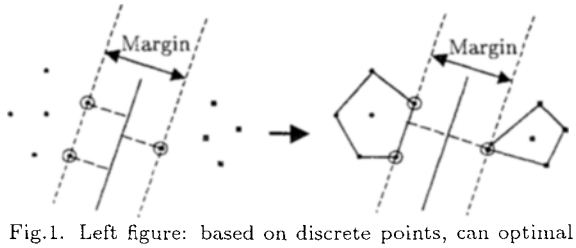

Fig.1. Left figure: based on discrete points, can optimal hyperplane and support vectors be found without the QP method? Right figure: it is easy to find them via the NPP algorithm (geometric method).

into the following dual form

$$\max \sum_{i\in I\cup J} \alpha_i - \frac{1}{2} \sum_{i,j\in I\cup J} \alpha_i\alpha_j y_i y_j \langle p_i, q_j\rangle \quad \text{s.t.} \sum_{i\in I\cup J} y_i\alpha_i = 0,\ \alpha_i \geq 0,\ \forall i \in I\cup J \tag{2}$$

It seems that there is no choice but the QP method for the optimization problem (2). In fact, there are better methods for SVM. In this section, the maximal margin algorithm in the linear case based on the shortest distance between two disjoint closed convex hulls spanned by samples will be described.

The following theorem is elementary in convex analysis.

Theorem 2.1. *Let $p_1, p_2, \ldots, p_n$ be vectors in R^d. Let U be the smallest closed convex set containing $p_1, p_2, \ldots, p_n$, then $U = \left\{u = \sum_{i=1}^n \alpha_i p_i : \sum_{i=1}^n \alpha_i = 1,\ \alpha_i \geq 0,\ i = 1, 2\ldots, n\right\}$. U is also called the closed convex hull of $p_1, p_2, \ldots, p_n$ or the closed convex set spanned by $p_1, p_2, \ldots, p_n$.*

From Hahn-Banach separation theorem, the following theorem is obtained.

Theorem 2.2. *P and Q are linearly separable if and only if their closed convex hulls are disjoint. The separation hyperplane decided by their closed convex hulls is in fact a linear classifier between P and Q.*

Obviously, the existence problem of linear classifier is solved by Theorem 2.2. However, how to get the linear classifier? Except solving the optimization problem (2), the following idea may be better.

Assume that U and V are the closed convex hull of sample sets $p_1, p_2, \ldots, p_n$ and $q_1, q_2, \ldots, q_m$, respectively and the two sample sets are linearly separable. Consider the following quadratic programming problem:

$$\min \left\| \sum_{i\in I} \alpha_i p_i - \sum_{j\in J} \alpha_j q_j \right\|^2 \quad \text{s.t.} \sum_{i\in I} \alpha_i = \sum_{j\in J} \alpha_j = 1,\ \alpha_i \geq 0,\ \forall i \in I\cup J \tag{3}$$

Problem (3) can be strictly proved to be equivalent to (2)[2,3].

If $\alpha_1^*\alpha_2^*, \ldots, \alpha_n^*, \beta_1^*, \beta_2^*, \ldots, \beta_m^*$ is a solution of (3) and let $u^* = \sum_{i=1}^n \alpha_i^* p_i$, $v^* = \sum_{i=1}^m \beta_i^* q_i$ and $\omega = u^* - v^*$, u^* and v^* is the nearest point pair between U and V. $\|u^* - v^*\| = \min_{u\in U, v\in V} \|u - v\|$, which is denoted as $d(U, V)$, is called the shortest distance between U and V. The linear classifier between two sample sets is a hyperplane, which passes the center of u^* and v^* with normal vector ω. This classifier is just the same as the one determined by (2) since the problems (2) and (3) are equivalent. Those points for $\alpha_i^* \neq 0$ $(i = 1, 2, \ldots, n)$ and $\beta_j^* \neq 0$ $(j = 1, 2, \ldots, m)$ are just support vectors (See the right figure in Fig.1).

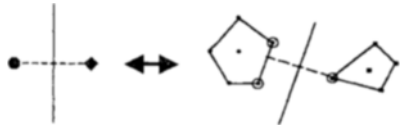

Fig.2. Left figue: the classification of two samples. Right figure: the classification of more than two samples.

According to the above geometric description, one can understand SVM from another viewpoint. If the prior knowledge is unknown, it is reasonable to equally separate two sample sets. For the classification of two samples in the two-dimensional space, the perpendicular bisector should be regarded as the classifier. If there are more than two samples, the classification between two sample sets is equivalent to that between their convex hulls according to the Hahn-Banach theorem. Here the classifier determined by SVM is the perpendicular bisector between the two convex hulls, which is in fact that between the two nearest points. (See Fig.2)

In addition, one can easily understand why only the operation of dot product is concerned in the algorithms for SVM. For example, if only $\langle a, a\rangle$, $\langle a, b\rangle$, $\langle b, b\rangle$, $\langle x, a\rangle$, $\langle x, b\rangle$ are known, then $\langle x, a\rangle - \langle x, b\rangle - \frac{\langle a,a\rangle - \langle b,b\rangle}{2} = 0$ is a classifier. Here "a" and "b" are trained samples and "x" is a test sample. Therefore, the dot products of every two samples are sufficient for implementing the classification and the value of each sample is unnecessary.

From the view of SVM, the classifiers determined by (2) and (3) are the same. However, their description about SVM is different intrinsically. The optimization problem (2) is based on the margin of two discrete point sets while (3) is based on the shortest distance between two disjoint convex hulls. Although QP can still be applied in (3), there are many efficient and intuitive geometric methods for

(3). Moreover, the geometric description makes the nature of learning problems clearer. For example, support vectors are trivial in the linear case since the classifier only depends on the nearest point pair of two disjoint convex hulls naturally. However, support vectors are crucial if kernel is applied in the nonlinear case because the nearest points in the feature space cannot be represented without them. In addition, the solution of SVM is not unique since the representation of the nearest points may be different, but the classifier decided by SVM is fixed. For example, the nearest point u^* may be represented by either support vectors P_1 and P_2 or support vectors P_1 and P_3 (See Fig.3).

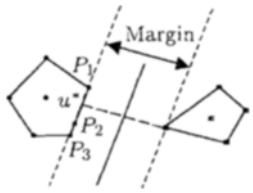

Fig.3. Support vectors representing the nearest points are not unique.

3 Projection Algorithm for Linear SVM

Definition 3.1 (Minimal Normal Problem). *Let U be a bounded and closed convex set in E^d, where o an origin in R^d and $o \notin U$. Solving MNP is to find a $u \in U$ such that $\forall x \in U$, $\|u - o\| \leq \|x - o\|$.*

As pointed out in [3], solving NPP between U and V is equivalent to the MNP of Z, which is the Minkowski set difference of U and V. It seems that MNP of Z is simpler than NPP, but MNP of Z is not simpler because of too many vertices of Z. Thus Keerhi pointed out that it is difficult to apply some excellent geometric algorithms for MNP, such as the Gilbert algorithm, to solving SVM directly. Keerthi designed a geometric algorithm for SVM motivated by the Gilbert algorithm.

Recently, it has been proved that the nearest point pair between U and V is the fixed-point of projection mapping in the Hilbert space. Then the corresponding algorithm on finding this fixed-point named Swap algorithm is presented[6]. According to this result we further point out that the solution of SVM is the fixed-point of projection mapping and apply the Swap algorithm to SVM. Then solving SVM is converted into doing MNP such that a new way to geometrically solve SVM is pioneered. This new way makes it possible to apply many excellent geometric algorithms of MNP such as the Gilbert algorithm, and the difficulty pointed out by Keerthi will be overcome. Then by combining the Swap algorithm with the Gilbert algorithm, a new geometric algorithm for SVM, named Projection algorithm, is given in this article. Some definitions and lemmas in [6] will be introduced in the following in order to make it easy to understand that SVM is just the fixed-point of projection mapping.

Definition 3.2. *Let U be a bounded and closed convex set in R^d. Let o be a point in R^d and $o \notin U$. u is the projection of o on U if and only if $u \in U$ and $\forall x \in U$, $\|u - o\| \leq \|x - o\|$. $\pi_U : o \to u$ is called a projection operator on U.*

Definition 3.3. *Let U and V be two disjoint, bounded and closed convex subsets in R^d and π_U and π_V projection operations on U and V, respectively. Let $\pi_{UV} \equiv \pi_U \circ \pi_V$ and $\pi_{UV} : U \to U$ is named by the alternate projection operation. $\pi_{UV}(u)$ is a alternate projection of u on U if $u \in U$.*

Lemma 3.1. *$\pi_{UV} : U \to U$ is a non-expansive operator and there exists a fixed-point of the operator $\pi_{UV} : U \to U$.*

Lemma 3.2. *Let U and V be two disjoint, bounded and closed convex subsets in R^d and π_U and π_V projection operations on U and V, respectively.*

(1) If $u \in U$ satisfies that there exists $v \in V$ such that $d(u, v) = d(U, V)$ then u is a fixed-point of the operator: $\pi_{UV} : U \to U$.

(2) Conversely, if $u \in U$ and u is a fixed-point of π_{UV}, then $d(u, \pi_V(u)) = d(U, V)$.

Theorem 3.1. *The Swap algorithm converges to a point $u \in U$ such that u and $v = \pi_V(u)$ satisfy $d(u, v) = d(U, V)$.*

The idea of the Swap algorithm is shown in Fig.4 and the details can be found in [6].

To use the Swap algorithm, a fast algorithm projecting a point on a convex hull is needed and a local search procedure has been introduced in [6]. Unfortunately, it can only be applied in a three-dimensional space. In the higher space, we will apply Gilbert algorithm[8], MDM algorithm[9] or

Hybrid algorithm[10] to Swap algorithm. The details about these algorithms can be found in [3]. Here we only introduce Gilbert algorithm for simplicity and its process is shown in Fig.5.

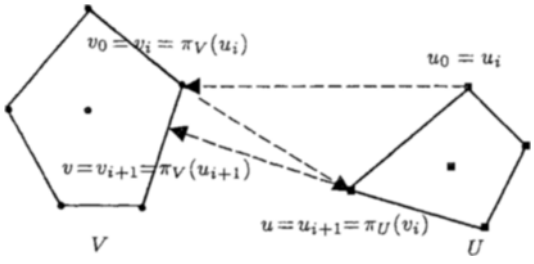

Fig.4. Swap algorithm.

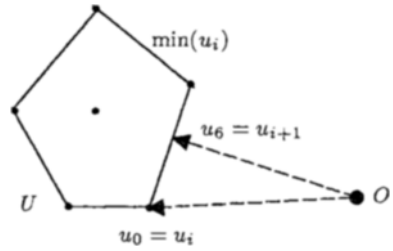

Fig.5. Gilbert algorithm.

Gilbert algorithm is described as follows:

Step 1. Choose $u_0 \in U$ and err;

Step 2. Solve $\min_{x \in P}\langle x - o, u_i - o\rangle$ and $\min(u_i) = x$;

Step 3. Analytically solve u_{i+1} and u_{i+1} is the nearest point of o on the segment of u_i and $\min(u_i)$;

Step 4. If $\|u_{i+1}\|^2 - \langle u_{i+1}, \min(u_{i+1})\rangle \leq err$, then $u = u_{i+1}$ and stop, otherwise $u_i = u_{i+1}$ and go to Step 2.

From the above analysis, the solution of SVM is a fixed-point of alternate projection operation π_{UV} and linear SVM will be solved geometrically by the Projection algorithm combining the Swap algorithm with the Gilbert algorithm. The difference between the Projection algorithm and Keerthi's algorithm is that the Projection algorithm converts NPP between U and V into a simpler MNP of Z and the vertex number of the new convex hull Z does not increase. Therefore a new way to geometrically solve SVM is opened and the Gilbert algorithm can be used directly.

Although no firm conclusion can be drawn from the limited experiments, the Swap algorithm has exhibited sub-linear empirical computational cost[6]. Moreover, the Swap algorithm opens a way to get faster algorithms if a fast projection method required in the Swap algorithm is employed.

4 Nonlinear SVM

In Section 3 a geometric algorithm for linear SVM is presented. Recently the feature mapping and kernel method are very popular and efficient for solving nonlinear SVM. However, the kernel method does not guarantee that the mapped points are linearly separable in the feature space. Linearly inseparable mapped points mean that two corresponding closed convex sets intersect. Obviously, the algorithms on NPP including Keerthi's algorithm and our algorithm will become meaningless if two closed convex sets intersect.

Fortunately, we have proved the existence of universal kernel functions in [12]. In this article, the theoretical result will be used to solve the above problem. Here, we will give the intuitive analysis on the theoretical result and give an example on universal kernel functions. First, it is necessary to understand the nature of feature mappings and the kernel method from the following example (quoted from [11], see Fig.6).

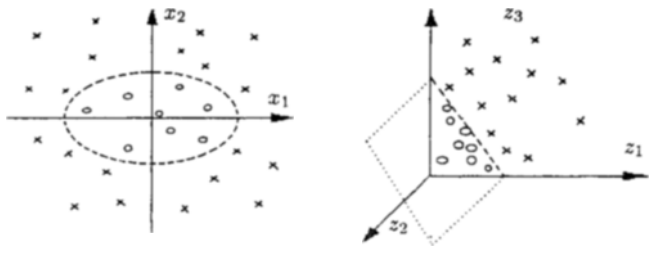

Fig.6. Feature mapping $\Phi : R^2 \to R^3$[11].

Example 1. A nonlinear classifier is required to separate two classes in a two-dimensional sample space, while only a hyperplane is needed in the three-dimensional feature space via the feature mapping $\Phi: \underset{(x_1,x_2)\mapsto(z_1,z_2,z_3)}{R^2 \to R^3}$, here $(z_1, z_2, z_3) \equiv (x_1^2, \sqrt{2}x_1x_2, x_2^2)$.

From this simple example, the nonlinear classification problem can be converted into a linearly separable one in the high dimensional space if the classifier is a polynomial function $a_0 + a_1x_1^2 + \sqrt{2}a_2x_1x_2 + a_3x_2^2 = 0$. The Tietze extension theorem in Topology solves the existence problem of nonlinear classifier, which is just like the Hahn-Banach separation theorem for the linear one. Here we will give a new explanation about the kernel method by using the Tietze extension theorem.

Definition 4.1. *Let Q_1 and Q_2 be two disjoint and closed sets in R^d. A mapping $\Phi: R^d \to H$ is called a feature mapping about classification of Q_1 and Q_2, if H is a Hilbert space, such that $\Phi(Q_1)$ and $\Phi(Q_2)$ are linearly separable in H, and H is called a feature space about classification of Q_1 and Q_2. Here, the dominant role of feature mappings, which guarantees linear separability, is obviously emphasized.*

Definition 4.2. *Let $\Phi: R^d \to H$ be a feature mapping about classification of Q_1 and Q_2. A mapping $k: H \times H \to R$, $k(x, y) = \langle \Phi(x), \Phi(y) \rangle$ is called a kernel mapping corresponding to the feature mapping Φ about classification of Q_1 and Q_2.*

Definition 4.3. *Let H be Banach space, a mapping $\Phi: R^d \to H$ is called a universal feature mapping, if $\Phi(Q_1)$ and $\Phi(Q_2)$ are linearly separable in H for arbitrary disjoint closed sets Q_1 and Q_2 in R^d. The corresponding kernel k is called a universal kernel mapping.*

From the Tietze extension theorem in Topology, the following theorem is obtained.

Theorem 4.1. *Let X be a metric space, then for each two disjoint closed sets Q_1 and Q_2, there exists a continuous function $f(x): X \to [-1, 1]$, such that $f(x) = \begin{cases} 1 & \forall x \in Q_1 \\ -1 & \forall x \in Q_2 \end{cases}$.*

The proof can be found in [12].

Since Q_1 and Q_2 are not closed convex but only closed, their empty intersection only means that they are linearly inseparable, or roughly no inconsistent samples. Theorem 4.1 is in fact an existence theorem for nonlinear classifiers, guaranteeing that there is a nonlinear classifier for linear inseparable problems without inconsistent samples.

When we discuss a classification problem, Q_1 and Q_2 are generally two bounded sets in a finite dimensional Euclidean space. The Tietze extension theorem guarantees that there is a continuous function f separating Q_1 and Q_2. According to the Stone-Weierstrass theorem, for $\forall \varepsilon > 0$, there is an n-order polynomial $a_0 + a_1x + \cdots + a_nx^n$ such that $\|f - (a_0 + a_1x + \cdots + a_nx^n)\| < \varepsilon$. If ε is small enough, $a_0 + a_1x + \cdots + a_nx^n = 0$ can be approximately regarded as a classifier. Therefore, the feature mapping can be constructed and nonlinear problems are converted into linearly separable problems in the high dimensional space similar to the above example.

In fact, RBF can be extended to infinite terms according to Taylor series extension. This means that the dimension of the feature space is infinite. According to the above analysis, RBF can approach arbitrary continuous functions because RBF can always correspond an n-order polynomial function, where n is an arbitrary natural number. Therefore RBF is just a universal kernel function guaranteeing two disjoint closed convex sets, which is the requirement of the algorithm for NPP. More theoretical details on universal kernel functions can be found in [12].

5 Kernel Projection Algorithm for Nonlinear SVM

The Projection algorithm can solve linear SVM. In order to solve linearly inseparable problems in the sample space, the Kernel Projection algorithm is given in this section by combining the Projection algorithm with the kernel method. The reason for this combination is that only the operation of dot product is concerned in the Projection algorithm. In addition, the theoretical analysis in Section 4 guarantees that this geometric algorithm is reasonable.

The Kernel Projection algorithm is described as follows:

Step 1. Choose $x_1 \in U$, then $\Phi(x_1) \in \hat{U}$ and the stopping criterion *EPS*;
Step 2. $\Phi(x_{n+1}) = \pi_{\hat{U}\hat{V}}(\Phi(x_n))$;
Step 3. If $\|\Phi(x_{n+1}) - \Phi(x_n)\|^2 \leq EPS$, then go to Step 4, else $x_n = x_{n+1}$, go to Step 2;
Step 4. $(\Phi(x_n), \pi_{\hat{V}}(\Phi(x_n)))$ is an approximate solution pair and stop.

Here, the projection operator π can be solved through Gilbert, MDM or Hybrid algorithms mentioned in Section 3. Each point in the feature space is represented only by the linear combination of feature mapping in all samples and all the algebraic computing in the high dimensional feature space can be avoided entirely. For example, $\|\Phi(x) - \Phi(y)\|^2$ can be computed by $k(x,x) - 2k(x,y) + k(y,y)$.

From the Kernel Projection algorithm, the Swap algorithm is employed as a shell program for solving NPP in the feature space. It converts NPP into a simpler MNP and the number of outer loops is usually small. In our experiment on the two-spiral problem, two loops are enough for the classification. In addition, it has been reported that an empirical computational time of the Swap algorithm is sub-linear[6]. From our experiments, it is obvious that the computation time increases linearly with the size of samples and the number of support vectors (See Fig.7 and Fig.8). So we think that the Kernel Projection algorithm can also handle large-scale SVM problems and its geometric meaning is more obvious.

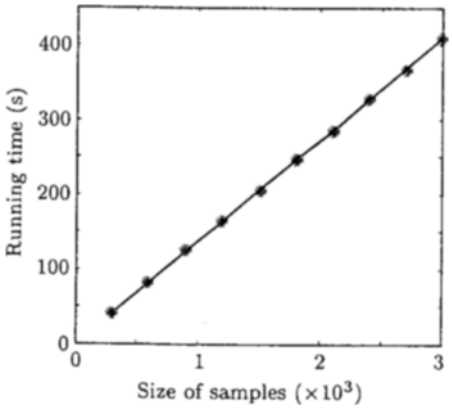

Fig.7. The relation between computational cost and size of samples.

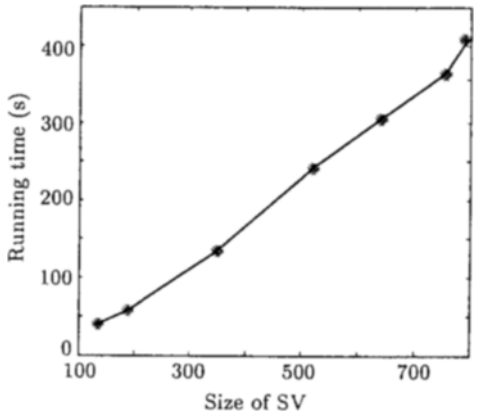

Fig.8. The relation between computational cost and size of SV.

6 Experiments and Further Discussion

The following experiment results are obtained with a PIII 600MHz processor and 256M bytes memory. Two spirals K_1 and K_2 are given in polar coordinates: $K_1 : \rho = \theta$, $K_2 : \rho = \theta + \pi$. In the experiments, RBF $c_0 e^{\frac{-\|x-y\|^2}{2\sigma^2}}$ is used. We find that the appropriate parameters σ and c_0 can ensure the feature space linear separable, thus the Kernel Projection algorithm can be applied.

Experiment 1. $0 \leq \theta \leq 10\pi$. The parameters $2\sigma^2$ and c_0 are taken to be 3 and 1.5, respectively. We find that the number of support vectors almost does not change when the size of samples increases. The relation between the computing cost and the size of samples is shown in Fig.7.

Experiment 2. The size of samples is always 63,000. The parameters $2\sigma^2$ and c_0 are taken to be 0.7 and 4, respectively. We find out that the number of support vectors increases when the range of θ becomes larger. The relation between the computing time and the size of support vectors is shown in Fig.8.

Experiment 3. $0 \leq \theta \leq 10\pi$. The size of samples is 30,000, and $2\sigma^2 = 3$. We find that the running time depends on Relative Maximal Margin rather than Maximal Margin and the larger Maximal Margin does not necessarily reduce the running time. The details can be found in Table 2.

The classical QP based on the Hessian matrix need to compute the dot product of each two points. However, the SMO and geometric method show that this computation is unnecessary. SMO is to solve

a sequence QP of size two and this is done analytically. Similarly, geometric methods convert NPP of two disjoint convex sets into that of two or three points and this can also be done analytically. The computational cost of SVM is reduced since the Hessian matrix is avoided. Geometric methods and SMO are all competitive for solving large-scale SVM problems[3]. Although the objectives of SMO and geometric methods are equivalent, their descriptions are different. The former is to maximize the margin of two discrete point sets and the latter is to solve the shortest distance of two disjoint closed convex sets. While their computational costs are similar, the latter can be more easily understood than the former.

Table 2. Experiments on Relative Margin

Parameter in RBF c_0	1.5	2	2.5	3
Maximal margin	0.1491	0.1614	0.1743	0.1873
Relative maximal margin	0.0994	0.0807	0.0697	0.0624
Running time (s)	45.596	54.939	61.358	66.125
Support vectors	264	315	356	387

Although the existence problem of universal kernels is solved theoretically and RBF $e^{\frac{-\|x-y\|^2}{2\sigma^2}}$ is just a universal kernel that can convert arbitrary distribution samples into linear separable sets, there are still linearly inseparable cases in the feature space because of the rounding error in computers. The feature mapping $e^{\frac{\|x\|^2}{2\sigma^2}}(1, \frac{x}{\sigma}, \frac{1}{\sqrt{2!}}(\frac{x}{\sigma})^2, \ldots, \frac{1}{\sqrt{n!}}(\frac{x}{\sigma})^n, \ldots)$, by which the RBF kernel can be constructed, cannot be extended to infinite terms because of the rounding error and thereby the dimension number of the feature space is not infinite. The smaller σ, the higher the dimension of the feature space. Linearly inseparable problems may occur in the feature space if the dimension is not large enough. On the other hand, we find that the maximal margin lessens if the smaller σ is chosen. Thus it is unavoidable to adjust the parameters in kernel functions in order to solve both generalization and nonlinear problems.

In the view of feature mapping, the new feature mapping $c_0\frac{\Phi(x)}{\|\Phi(x)\|}$ is used such that all the mapped points are on a hypersphere in the feature space. The corresponding kernel function is $c_0\frac{k(x,y)}{\sqrt{k(x,x)\times k(y,y)}}$ and c_0 is the radius of the hypersphere. Experiments can be analyzed easily if mapped points are located in a hypersphere. RBF $e^{\frac{-\|x-y\|^2}{2\sigma^2}}$ just satisfies the above property and maps all sample points onto a hypersphere.

In this paper, a coefficient c_0 is used in RBF and thus $k(x,y) = c_0 e^{\frac{-\|x-y\|^2}{2\sigma^2}}$. This means that each point is mapped onto a hypersphere with the radius c_0 in the feature space. If c_0 is selected appropriately, the number of support vectors will become less and running time will also be reduced. A new concept, Relative Maximal Margin is defined as $\frac{m}{c_0}$, where m is Maximal Margin in the feature space. In [3], Maximal Margin is regarded as the benchmark of convergence speed, but we think that the convergence speed may be proportionally related to $\frac{m}{c_0}$ rather than m (See Table 2). It is interesting that $\frac{c_0}{m}$ is just related to the bound of the VC dimension. Thus Relative Maximal Margin should be more natural than Margin.

7 Conclusion

The Kernel Projection algorithm is presented in this paper, which combines the kernel method with the Projection algorithm. Experiments show that this geometric algorithm can handle large-scale SVM problems and its geometric meaning is more obvious. What is more important is that solving SVM is converted into doing MNP and thus many excellent geometric algorithms can be applied.

In addition, the theoretical analysis on universal kernel functions guarantees that the geometric algorithms for SVM are reasonable in theory. Although the soft margin idea has been applied to the NPP algorithm for SVM in [3], we think that there is a flaw. How to combine the Kernel Projection algorithm with the soft margin idea is further reported in [13].

Acknowledgement We would like to thank the referees for their valuable comments on this paper.

References

[1] Vapnik V. Statistical Learning Theory. Addison-Wiley, 1998.
[2] Bennett R, Bredensteiner E J. Geometry in Learning. Tech. Report, Department of Mathematical Sciences, Rennselaer Polytechnic Institute, New York, 1996.
[3] Keerthi S S, Shevade S K, Bhattacharyya C *et al.* A fast iterative nearest point algorithm for support vector machine classifier design. *IEEE Transactions on Neural Networks,* 2000, 11(1): 124–136.
[4] Crisp D J, Burges C J C. A geometric interpretation of μ-SVM classifiers. *NIPS* 12, 2000, pp.244–250.
[5] Tao Qing, Sun Demin, Fan Jinsong. The maximal margin linear classifier based on the contraction of the closed convex hull. To appear in *Journal of Software.*
[6] Llanas B, de Sevilla M F. An iterative algorithm for finding a nearest pair of points in two convex subsets of R^n. *Computers and Mathematics with Applications,* 2000, 40: 971–983.
[7] Platt J. Fast training of support vector machines using sequential minimal optimization in Advances in Kernel Methods—Support Vector Learning. Scholkopf B, Burges C J C, Smola A J (eds.), Cambridge, MA: MIT Press, 1999, pp.185–208.
[8] Gilbert E G. Minimizing the quadratic form on a convex set. *SIAM J. Contr.,* 1966, 4: 61–79.
[9] Mitchell B F, Dem'yanov V F, Malozemov V N. Finding the point of a polyhedron closest to the origin. *SIAM J. Contr.,* 1974, 12: 19–26.
[10] Gilbert E G, Johnson D W, Keerthi S S. A fast procedure computing the distance between complex objects in three dimension space. *IEEE J. Robot. Automat.,* 1988, 4: 193–203.
[11] Muller K R *et al.* An introduction to kernel-based learning algorithms. *IEEE Trans. Neural Networks,* 2001, 12(2): 181–201.
[12] Tao Qing, Wu Gaowei, Wang Jue. The theoretical analysis of kernel technique and kernel covering approach. Technique Report, Institute of Automation, Chinese Academy of Sciences, 2001.
[13] Tao Qing, Wang Jiaqi, Wang Jue. Soft kernel projection algorithm for support vector machines. Technique Report, Institute of Automation, Chinese Academy of Sciences, 2001.

WANG Jiaqi received his B.S. degree from Beijing Polytechnic University in 1998. He is currently a graduate student in Institute of Automation, Chinese Academy of Sciences, P.R. China. His research interests are data mining, machine learning and kernel method.

TAO Qing received the M.S. degree in mathematics from Southwest Normal University in 1989 and the Ph.D. degree from the University of Science and Technology of China in 1999. From June 1999 to June 2001, he was a Postdoctoral Fellow in the University of Science and Technology of China. He is currently a Postdoctoral Fellow in Institute of Automation, Chinese Academy of Sciences. His research interests are neural networks, nonlinear function analysis and SVM theory.

WANG Jue is a professor in Institute of Automation, Chinese Academy of Sciences. His research interests are ANN, GA, multi-agent system, machine learning and data mining.