

Multicategory Nonparallel Proximal Support Vector Machine

Xubing Yang¹ Songcan Chen¹ Zhisong Pan²

¹ Computer Science and Engineering, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, P. R. China

² Institute of Command Automation, People's Liberation Army University of Science and Technology, Nanjing 210007, P. R. China

Abstract

We propose a multicategory classifier, that is, Multicategory Nonparallel Proximal Support Vector Machine (MNPSVM), which is in the spirit of Proximal SVMs via Generalized Eigenvalues (GEPSVM). Difference from GEPSVM lie in: 1) MNPSVM keeps the genuine rather than approximate geometrical interpretation of the nonparallel proximal SVMs; 2) each nonparallel plane of MNPSVM is generated by its corresponding standard eigenvalue problems, instead of nowadays generalized eigenvalue problems. The effectiveness is demonstrated by tests on synthetic and real data sets. Furthermore, we also discuss its efficiency in experiment section and conclude that MNPSVM is far higher than that of both GEPSVM and SVM.

Keywords: Support vector machine, Multicategory data classification, Generalized eigenvalue

1. Introduction

Standard support vector machines (SVMs) [1] are based on the structural risk minimization (SRM) principle and aim at maximizing the margin between the points of two-category data classification and then assign those points to one of disjoint half-space in either the original input space for linear classifiers, or in a higher feature space for nonlinear classifiers. In recent times, some much simpler classifiers, such as proximal support vector machine (PSVM) [2] and LSSVM [3], were implemented wherein each class of points is assigned to the closet of two parallel planes that are pushed apart as far as possible. Recently, kicking of the parallel constrains of the PSVM, Mangasarian and Fung proposed a nonparallel plane classifier [4]. The classifier attempts to seek two nonparallel planes and requires that each plane should be close to data points of one class and far from the data points of the other class. With this intuition, each

plane is generated by an eigenvector corresponding to a smallest eigenvalue of each of the generalized eigenvalue problems. For computational complexity, SVM requires a solution of either a quadratic or linear programs, and PSVM leads to a single system of linear equations. In contrast, GEPSVM is obtained by solving a generalized eigen-equation.

Due to the simplicity of GEPSVM algorithm, in this paper, we apply this 2-category classifier to construct C-category classification by using a one-against-the-rest separation for each class [5]-[6]. In essence, GEPSVM is derived by a generalized eigenvalue problem through minimizing a kind of Rayleigh quotient. For the two real symmetric matrices appearing in GEPSVM criterion, if both are nonpositive definite (singular), an ill-defined operation will be yielded due to floating-point imprecision. So, GEPSVM adds a perturbation to one of the singular (or semi-definite positive) matrix and the authors claimed that this perturbation acts as a regularization term. Furthermore, FMGEPSVM [7] is a direct extension of GEPSVM through introducing a fuzzy membership into classifier construction. However, such multicategory method is still puzzled by foresaid problems. To keep the genuine but not approximate, geometrical interpretation of the nonparallel plane classifier, a different regularization trick is adopted to adjust the tradeoff between classes of the given data points. So, the proposed multicategory classification method need NOT to care about the singularity of the foresaid matrices in Rayleigh quotient due to adoption of a similar formulation to the Maximum Margin Criterion [8] (MMC), but it is worthwhile noting that MMC just is a dimensionality reduction method rather than a classification method. By the way, the proposed classifier also is easy to extend to its fuzzy version with similarly introducing fuzzy membership as described in [9].

The rest of this paper is organized as follows. In Section 2, we review some basic work about GEPSVM. The proposed approach will be appeared in

section 3. In Section 4, we provide the experimental results on some artificial and public datasets. Finally, we offer our conclusion in Section 5.

2. A brief review on GEPSVM algorithm

To motivate our multicategory approach, we begin with the brief description of two-class GEPSVM [4] algorithm, which has been proved to be an effective approach for binary classification.

Given a training set of two pattern classes $X^{(i)} = [x_1^{(i)}, x_2^{(i)}, \dots, x_{N_i}^{(i)}]$, $i = 1, 2$ with N_i n -dimensional patterns in the i^{th} class. Throughout the paper, superscript “ T ” will denote transposition, and “ e ” is a case-dependent dimensional column vector whose entries are all ones. Denote the training set by a $N_1 \times n$ matrix A (A_i is the i^{th} row of the matrix A corresponding to the i^{th} pattern in Class 1) and the $N_2 \times n$ matrix B (B_i is the i^{th} row of the B , the i^{th} pattern in Class 2), respectively. The GEPSVM attempts to seek two planes in n -dimensional input space respectively corresponding to the two classes:

$$\mathbf{x}^T \mathbf{w}^1 + r^1 = 0, \quad \mathbf{x}^T \mathbf{w}^2 + r^2 = 0, \quad (1)$$

where the two non-parallel planes are closest to the points in its own class and as far from the points in the other one as possible. This leads to the following optimization problem:

$$\min_{(w,r) \neq 0} \frac{\|A \mathbf{w} \mathbf{e}^T\|^2 + \delta \|\mathbf{r}\|^2}{\|B \mathbf{w} \mathbf{e}^T\|^2} \quad (2)$$

where δ is a nonnegative regularization factor, and $\|\cdot\|$ denotes the two-norm. By making the definitions:

$$\begin{aligned} \mathbf{G} &:= [A \mathbf{e}^T]^T [A \mathbf{e}^T] + \delta \mathbf{I}, \\ \mathbf{H} &:= [B \mathbf{e}^T]^T [B \mathbf{e}^T], \quad \mathbf{z} := [\mathbf{w}^T \ r]^T \end{aligned} \quad (3)$$

then, with respect to the first plane of (1), formula (2) becomes:

$$\min_{z \neq 0} r(z) := \frac{\mathbf{z}^T \mathbf{G} \mathbf{z}}{\mathbf{z}^T \mathbf{H} \mathbf{z}} \quad (4)$$

where \mathbf{G} and \mathbf{H} are positive semi-definition matrices when $\delta=0$. Formula (4) can be solved by the following generalized eigenvalue problem

$$\mathbf{G} \mathbf{z} = \lambda \mathbf{H} \mathbf{z}, \quad \mathbf{z} \neq 0 \quad (5)$$

The global minimum of (4) is achieved at an eigenvector of the generalized eigenvalue problem (5) corresponding to the smallest eigenvalue when \mathbf{G} or \mathbf{H} in eq (5) is a positive definite matrix, so in many real-world cases, regularization factor δ must be set to a positive constant, especially in some Small Size Sample (SSS) problems where the number of training samples is far less than the dimensionality of the samples. The 2nd plane can be obtained with a similar process.

To extend this binary classifier to be applicable for multicategory classification, we redefine C -class patterns as the following: $X^{(i)} = [x_1^{(i)}, x_2^{(i)}, \dots, x_{N_i}^{(i)}]$, $i = 1, 2, \dots, C$.

3. The multicategory nonparallel proximal support vector machine (MNPSVM)

With the similar geometric objective, i.e., each plane should be closer to the data points of its own class and be farther to the data points of the other class, the goal of MNPSVM is to estimate C planes that can be well approximated under some constraints by the points of C classes. We first focus on a linear mode. Then, for nonlinear case, we extend those linear methods to corresponding nonlinear ones with the help of kernel tricks [10].

3.1. Linear MNPSVM

Assume C linear planes in n -dimensional space

$$P_i := \{\mathbf{x} \mid \mathbf{w}_i^T \mathbf{x} + b_i = 0, \mathbf{x} \in R^n\}, i = 1, 2, \dots, C \quad (6)$$

where \mathbf{w}_i is normal vector and b_i is the threshold of the linear plane P_i . We hope that the points in the i^{th} class are as close to their corresponding plane P_i as possible and as far from the other planes P_l ($l = 1, \dots, C, l \neq i$) as possible. The optimization criterion and constraints can be defined to determine those planes as follows:

$$\min \sum_{i=1}^C \sum_{j=1}^{N_j} (\mathbf{w}_i^T \mathbf{x}_j^{(i)} + b_i)^2 - \sum_{i=1}^C \eta_i \sum_{\substack{j=1 \\ j \neq i}}^C \sum_{l=1}^{N_l} (\mathbf{w}_i^T \mathbf{x}_l^{(j)} + b_i)^2 \quad (7)$$

$$s.t. \quad \|\mathbf{w}_i\| = 1, \quad i = 1, 2, \dots, C \quad (8)$$

where η_i (>0) is a tradeoff factor. Under minimizing objective (7) and constraint (8), the first term of (7),

$\sum_{j=1}^{N_j} (\mathbf{w}_i^T \mathbf{x}_j^{(i)} + b_i)^2$, can be interpreted as that the points of the i^{th} class must be close to the plane P_i . The

second term $-\sum_{\substack{j=1 \\ j \neq i}}^C \sum_{l=1}^{N_l} (\mathbf{w}_i^T \mathbf{x}_l^{(j)} + b_i)^2$ can be regarded as

that the points of the j^{th} ($j \neq i, j = 1, 2, \dots, C$) class must be far from P_i .

Further simplifying formulas (7) and (8), we obtain the following:

$$\min \sum_{i=1}^C \sum_{j=1}^{N_j} (\tilde{\mathbf{w}}_i^T \tilde{\mathbf{x}}_j^{(i)})^2 - \sum_{i=1}^C \eta_i \sum_{\substack{j=1 \\ j \neq i}}^C \sum_{l=1}^{N_l} (\tilde{\mathbf{w}}_i^T \tilde{\mathbf{x}}_l^{(j)})^2$$

$$\Leftrightarrow \min \sum_{i=1}^C \tilde{\mathbf{w}}_i^T (\mathbf{A}_i \mathbf{A}_i^T - \eta \bar{\mathbf{A}}_i \bar{\mathbf{A}}_i^T) \tilde{\mathbf{w}}_i \quad (9)$$

$$s.t. \tilde{\mathbf{w}}_i^T \mathbf{B} \tilde{\mathbf{w}}_i = 1, \quad i=1,2,\dots,C \quad (10)$$

where $\tilde{\mathbf{w}}_i^T = [\mathbf{w}_i^T \ b_i]$, $(\tilde{\mathbf{x}}_j^{(i)})^T = [(\mathbf{x}_j^{(i)})^T \ 1]$, $\mathbf{A}_i = [\tilde{\mathbf{x}}_1^{(i)}, \tilde{\mathbf{x}}_2^{(i)}, \dots, \tilde{\mathbf{x}}_{N_i}^{(i)}]$ and $\bar{\mathbf{A}}_i$ denotes a matrix whose columns consist of the patterns of the rest $C-1$ classes, i.e., $\bar{\mathbf{A}}_i = [\mathbf{A}_1, \dots, \mathbf{A}_{i-1}, \mathbf{A}_{i+1}, \dots, \mathbf{A}_C]$. Constant matrix $\mathbf{B} = \begin{bmatrix} \mathbf{I}_{n \times n} & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 \end{bmatrix}$, and $\mathbf{I}_{n \times n}$ means an identity matrix with size $n \times n$.

Define Lagrange multipliers function L based on the objective function (9) and equality constraints (10) as follows:

$$L(\tilde{\mathbf{w}}_i, \lambda_i) = \sum_{i=1}^C \tilde{\mathbf{w}}_i^T (\mathbf{A}_i \mathbf{A}_i^T - \eta_i \bar{\mathbf{A}}_i \bar{\mathbf{A}}_i^T) \tilde{\mathbf{w}}_i - \sum_{i=1}^C \lambda_i (\tilde{\mathbf{w}}_i^T \mathbf{B} \tilde{\mathbf{w}}_i - 1) \quad (11)$$

Setting the gradients of L equal to zero gives the following optimality conditions:

$$(\mathbf{A}_i \mathbf{A}_i^T - \eta_i \bar{\mathbf{A}}_i \bar{\mathbf{A}}_i^T) \tilde{\mathbf{w}}_i = \lambda_i \mathbf{B} \tilde{\mathbf{w}}_i, \quad (12)$$

It is obvious that the matrix \mathbf{B} is singular. To overcome this singularity, we decompose the eq. (12) into two expressions as below

$$\begin{aligned} [(\mathbf{X}_i \mathbf{X}_i^T - \eta_i \bar{\mathbf{X}}_i \bar{\mathbf{X}}_i^T) - (\mathbf{X}_i \mathbf{e}_i - \bar{\mathbf{X}}_i \bar{\mathbf{e}}_i) \\ (\mathbf{X}_i \mathbf{e}_i - \bar{\mathbf{X}}_i \bar{\mathbf{e}}_i)^T] \mathbf{w}_i = \lambda_i \mathbf{w}_i \end{aligned} \quad (13)$$

$$b_i = -(\mathbf{X}_i \mathbf{e}_i - \bar{\mathbf{X}}_i \bar{\mathbf{e}}_i)^T \mathbf{w}_i / (N_i - \eta_i (N - N_i)) \quad (14)$$

where \mathbf{X}_i is composed of the patterns of the i^{th} class, and $\bar{\mathbf{X}}_i$, the patterns of the rest $C-1$ classes. As aforementioned definition, \mathbf{e}_i and $\bar{\mathbf{e}}_i$ are case-dependent column vector with all elements are ones, $i=1,2,\dots,C$. N means total number of the given C -class patterns. Thus, the planes just generate from their corresponding standard eigenvalue problems. Next, we turn to the nonlinear case.

3.2. Nonlinear MNPSVM

In real world, the problems encountered can not always be handled using linear methods. For making our method able to accommodate nonlinear cases, we will extend it to nonlinear MNPSVM with well-known kernel trick [10]-[11]. For convenience of description, we define Empirical Kernel Map (for more details please see [11]-[12]) as follows:

$K_{emp} : \mathbf{x} \in R^n \mapsto K_{emp}(\mathbf{x}) \in R^N$
 where $K_{emp}(\mathbf{x}) = [k(\mathbf{x}_1, \mathbf{x}), \dots, k(\mathbf{x}_N, \mathbf{x})]^T$, and $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ are aforesaid N training samples. Function $k(\mathbf{x}, \mathbf{y})$ stands for an arbitrary *kernel*, for

any n -dimension vector \mathbf{x} and \mathbf{y} , which maps the vector \mathbf{x} and \mathbf{y} into a real number in R . A frequently used kernel in nonlinear classification is Gaussian kernel with the expression defined as $k(\mathbf{x}, \mathbf{y}) = \exp(-\gamma \|\mathbf{x} - \mathbf{y}\|^2)$, where γ is a positive constant.

Similarly, we consider the following C kernel-generated nonlinear planes instead of the aforementioned linear ones (6)

$$\mathbf{u}^{(i)T} K_{emp}(\mathbf{x}) + b_i = 0, \quad i=1,2,\dots,C$$

With an entirely similar argument, we consider the following optimization criterion instead of the original one in the input space as (7) and (8).

$$\begin{aligned} \min \sum_{i=1}^C \sum_{j \neq i}^{N_j} ((\mathbf{u}^{(i)})^T K_{emp}(\mathbf{x}_j^{(i)}) + b_i)^2 \\ - \sum_{i=1}^C \eta_i \sum_{j=1}^C \sum_{l \neq i}^{N_j} ((\mathbf{u}^{(i)})^T K_{emp}(\mathbf{x}_l^{(j)}) + b_i)^2 \end{aligned} \quad (15)$$

$$s.t. \|\mathbf{u}^{(i)}\| = 1, \quad i=1,2,\dots,C \quad (16)$$

where $\mathbf{u}^{(i)}$ and b_i denote the normal vector and the threshold of the i^{th} plane P_i^K in the space R^N , respectively.

With the similar manipulation, we have the following C eigen-systems:

$$[(\mathbf{K}_i \mathbf{K}_i^T - \eta_i \bar{\mathbf{K}}_i \bar{\mathbf{K}}_i^T) - (\mathbf{K}_i \mathbf{e}_i - \bar{\mathbf{K}}_i \bar{\mathbf{e}}_i)(\mathbf{K}_i \mathbf{e}_i - \bar{\mathbf{K}}_i \bar{\mathbf{e}}_i)^T] \mathbf{u}^{(i)} = \lambda_i \mathbf{u}^{(i)} \quad (17)$$

$$b_i = -(\mathbf{K}_i \mathbf{e}_i - \bar{\mathbf{K}}_i \bar{\mathbf{e}}_i)^T \mathbf{u}^{(i)} / (N_i - \eta_i (N - N_i)) \quad (18)$$

where $\mathbf{K}_i = [K_{emp}(\mathbf{x}_1^{(i)}), \dots, K_{emp}(\mathbf{x}_{N_i}^{(i)})]$, $\bar{\mathbf{K}}_i = [\mathbf{K}_1, \dots, \mathbf{K}_{i-1}, \mathbf{K}_{i+1}, \dots, \mathbf{K}_C]$. Thus, we obtain C nonlinear planes $\mathbf{u}^{(i)T} K_{emp}(\mathbf{x}) + b_i = 0$.

In what follows, we turn to our experimental tests and some comparisons.

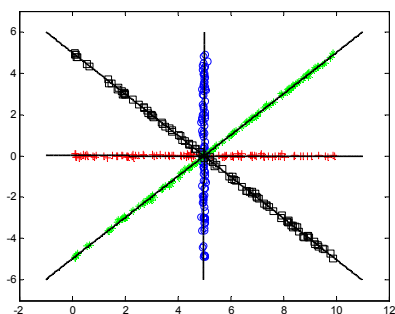
4. Experimental validations and comparisons

In order to test the MNPSVM classification capability, we perform some experiments on one artificial data set and some benchmark datasets from the UCI Repository [13].

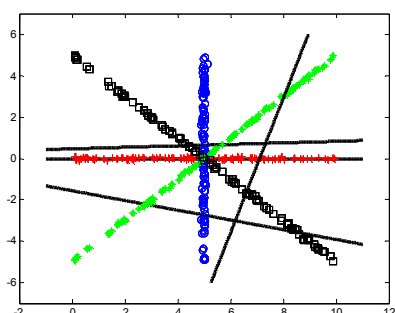
4.1. Illustration on a toy problem

On the artificial problem, a synthetic data set, termed as ‘‘multiple cross planes (MultiPlanes)’’, is designed to visually illustrate the effectiveness of our MNPSVM compared to GEPSVM. The ‘‘multiple cross planes’’ consists of four-class points. Those points in the same class are close to one of four cross planes in R^2 . Fig.1 illustrates their distributions of the

data and the four planes generated by MNPSVM and GEPSVM, respectively.



(a) MNPSVM



(b) GEPSVM

Fig. 1: The “multiple cross planes” learned by MNPSVM and GEPSVM. The figures illustrate the multiple linear planes generated by MNPSVM (a) and GEPSVM (b).

In this paper, when using binary classifiers GEPSVM and SVM for multi-class classification problem, one-against-the-rest approach will be adopted. The “multiple cross planes” consists of four lines which are crossed each other (see Fig1). It is obvious that “multiple cross planes” is a further example of the XOR problems. In this case, the learning correctness of GEPSVM deteriorates greatly, though it is better than that of the SVM on the example “cross planes” in [4]. However, it is undoubted that the training set correctness for MNPSVM is 100% in the examples of “multiple cross planes”. Why does GEPSVM work so poor in this example? The reason may be due to that: 1) after regularization, GEPSVM has lost its original geometrical interpretation; 2) each plane of GEPSVM is generated by all training points with one-against-all technique such that GEPSVM has to face unbalanced classification problem.

After illustrating the toy problem, we are in a position to carry out several experiments on some real-world data sets which are available at [13].

4.2. Test accuracy on public data sets

To demonstrate the performance of our approach, we report our results on some public UCI datasets. Table 1 shows a linear kernel comparison of MNPSVM versus GEPSVM and SVM (SMO algorithm [14-15] for binary classification and one-against-all for multiclass SVM [16]). For a linear kernel, MNPSVM, GEPSVM and SVM have a single parameter: η for MNPSVM, δ for GEPSVM and C for SVM. Similar to [16], the parameters are selected from the values $\{10^i \mid i = -7, -6, \dots, 7\}$ by using 10% percent of each training fold as a tuning set. In addition to report the average testing accuracies across the 10 folds, we also perform the paired t -test comparing GEPSVM to MNPSVM and SVM to MNPSVM. The p -value for each test is probability of the observed or a greater difference between two test accuracy values occurring, under the assumption of the null hypothesis that there is no difference between the test accuracy distributions. Thus, the smaller the p -value, the less likely that the observed difference resulted from identical test accuracy distributions. A typical threshold (confidence level) for p -values is 0.05. For example, in data set MultiPlanes, the p -value of the test comparing MNPSVM and SVM is 0.000, which is less than 0.05, leading us to conclude that MNPSVM and SVM have different accuracies on this data set. In table 1, the p -values are from t-test comparing GEPSVM and SVM to MNPSVM. Best accuracy results are in bold. An asterisk (*) denotes a significant difference from MNPSVM based on p -values less than 0.05.

Data set $m \times n$ Classes	MNPSVM Accuracy	GEPSVM Accuracy p -value	SVM Accuracy p -value
Iris 150×4 3C	92.7%	91.3% 0.678	94.2% 0.134
Sonar 208×60 2C	74.1%	68.4% 0.208	74.35% 0.174
Cmc 1473×8 3C	52.0%	48.4% 0.286	49.8% 0.408
Ionosphere 351×34 2C	85.1%	84.0% 0.785	86.1% 0.773
MultiPlanes 400×2 4C	93.5%	41.2%* 0.000	25.0%* 0.000

Table 1: Linear Kernel Classifiers MNPSVM, GEPSVM and SVM 10-Fold Testing Correctness/Accuracy and p -values.

Table 2 gives MNPSVM, GEPSVM and SVM using Gaussian kernel. The Gaussian kernel parameter σ is chosen from the values $\{10^i \mid i = -4, -3, \dots, 3, 4\}$ for all the three algorithms. The parameters, η_i ($i = 1, 2, \dots, C$) for MNPSVM, δ for GEPSVM and C (tradeoff factor) for SVM, are respectively selected from the set $\{10^i \mid i = -6, -5, \dots, 3, 4\}$. From Table 2, as expected, we note that nonlinear MNPSVM greatly outperforms nonlinear GEPSVM and SVM on the MultiPlanes data set.

Data set $m \times n$ Classes	DMPSVM Accuracy	GEPSVM Accuracy p -value	SVM Accuracy p -value
Wine 178×6 3C	68.7%	63.7% 0.058	93.2%* 0.000
Monk 432×6 2C	97.9%	89.3%* 0.002	98.1% 0.118
Glass 214×9 3C	94.6%	71.0%* 0.002	94.3% 0.161
MultiPlanes 400×2 4C	98.2%	50.3%* 0.000	32.5%* 0.000

Table 2: Gaussian Kernel Classifiers MNPSVM, GEPSVM and SVM 10-Fold Testing Correctness/Accuracy and p -values

Next what we concern is computation time. Time complexity of standard eigenvalue problem for smallest/biggest eigenvalue (power/anti-power method [17, section 8.2]) is of order n^2 , and for all eigenvalues (QR or Jacobi method [17, section 8.3, 8.4]) is of order n^3 , while complexity of the generalized eigenvalue problem is $O(n^3)$ [17, section 7.7]. For an interior point method used for solving a 2-norm SVM quadratic program, its time complexity is of order $n^{3.5}$ [18]. These facts help explain the computation times of the three algorithms: MNPSVM, GEPSVM and SVM. MNPSVM is over one order of magnitude faster than GEPSVM and nearly two orders of magnitude faster than SVM. As final remarks, we note that MNPSVM has revealed its distinct predominance in computation time.

5. Conclusions

We have proposed a novel approach for multicategory classification problem that simultaneously generates multiple planes to classify multicategory task. This approach follows the genuine rather than approximal geometric interpretation of nonparallel plane classifiers. With a different regularization strategy, each plane is easily obtained through solving the standard eigenvalue problem, instead of generalized

eigenvalue problem of GEPSVM and quadratic program of SVM. Furthermore, in most cases, classification accuracy results are comparable to standard SVM. Compared to GEPSVM, the test accuracy of MNPSVM outperforms GEPSVM, especially in multicategory XOR classification. Also, standard eigenvalue problem can more quickly be solved than the generalized eigenvalue problem in GEPSVM and the optimization algorithm needed for SVM. The simple geometric interpretation, computational efficiency, and test accuracy of MNPSVM on real world data indicate that it is an effective and efficient algorithm for multicategory classification.

Acknowledgement

This work is partially supported by National Nature Science Foundation of China (Grant No. 60473035 and 60603029).

References

- [1] V. Vapnik, *The nature of statistical learning theory*, Springer-Verlag, New-York, 1995
- [2] G. Fung and O. L. Mangasarian, Proximal support vector machine classifiers. *Proc. of Knowledge Discovery and Data Mining*, pp. 77-86, 2001.
- [3] J.A. K. Suykens and J. Vandewalle, Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3): 293-300, 1999.
- [4] O.L. Mangasarian and Edward W. Wild, Multisurface proximal support vector machine classification via generalized eigenvalues. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 28(1): 69-74. 2006
- [5] D.M.J. Tax and R.P. W. Duin, Using two-class classifiers for multiclass classification. *Proc. of 16th International Conference on Pattern Recognition*, pp. 124-127, 2002.
- [6] G. Fung and O. L. Mangasarian, Multicategory proximal support vector machine classifiers, *Machine Learning*, 59 (1-2): 77-97, 2005.
- [7] Jayadeva, R. Khemchandani and S. Chandra, Fuzzy multi-category proximal support vector classification via generalized eigenvalues, *Soft Computation*, 11: 679-685, 2007.
- [8] H.Li, T. Jiang and K. Zhang, Efficient and robust feature extraction by maximum margin criterion. *Proc. Of Conference on Advances in Neural Information Processing Systems*, pp. 97-104, 2004.
- [9] J. M. Keller and D. J. Hunt, Incorporating fuzzy membership function into the perceptron algorithm, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 7: 693-699, 1985.

- [10]K. R. Müller, S. Mika and G. Rätsch, An introduction to kernel-based learning algorithms, *IEEE Transaction on Neural Network*, 12(2):181-202, 2001.
- [11]B. Scholkopf and A. J. Smola, Learning with kernels, MIT Press, 2002.
- [12]Min Wang and Songcan Chen, Enhanced FMAM based on empirical kernel map, *IEEE Trans. on Neural Networks*, 16(3): 557-564, 2005
- [13]<http://www.isc.uci.edu/~mlearn/MLRepository.html>.
- [14]<http://www.datalab.uci.edu/people/xge/svm/>
- [15]J. C. Platt, Fast training of support vector machines using sequential minimal optimization. *In Advances in Kernel Methods - Support Vector Learning*, the MIT Press, MA, 1999.
- [16]<http://asi.insarouen.fr/~arakotom/toolbox/index.html>, 2005.12
- [17]G. H. Golub and C. F. V. Loan, *Matrix Computations*, 3rd edition, the Johns Hopkins University Press, Baltimore, 1996.
- [18]M. Kojima, S. Mizuno, T. Noma and A. Yoshise, A unified approach to interior point algorithms for linear complementarity problems, *Lecture Notes in Computer Science*, Springer-Verlag, Berlin, 1991.