Pairwise Multi-classification Support Vector Machines: Quadratic Programming (QP-P$_A$MSVM) formulations

THEODORE B. TRAFALIS and OLUTAYO OLADUNNI
School of Industrial Engineering
University of Oklahoma
202 West Boyd, Room 124 Norman, OK 73019
USA

Abstract: - The binary support vector machines (SVMs) have been extensively investigated. However their extension to a multi-classification model is still an on-going research. In this paper we present an extension of the binary support vector machines (SVMs) for the k > 2 class problems. The SVM model as originally proposed requires the construction of several binary SVM classifiers to solve the multi-class problem. We propose a single quadratic optimization problem called a pairwise multi-classification support vector machines (P$_A$MSVMs) for constructing a pairwise linear and nonlinear classification decision functions. A kernel approach is also discussed for nonlinear classification problems. Computational results are presented for two real data sets.

Key Words: - pairwise, SVM, MSVM, multi-class, kernel, classification, quadratic programming

1 Introduction

Support Vector Machines (SVMs) developed by Vapnik [1] are based on statistical learning theory and have been successfully applied to a wide range of problems. They provide tools in modeling and simulation of decision making processes for classification and regression problems. SVMs for classification problems perform a mapping of the input variables into a high dimensional (possibly infinite) feature space. Classification is done in this feature space by the use of a hyperplane. The resulting discriminant function in the input space is generally a nonlinear function. In order to map the variables into a higher dimensional feature space we use implicitly the concept of a kernel function.

Multi-classification SVM is an extension of support vector machines (SVMs), involving three or more classes. There is active research in this area, aiming at the construction of single optimization models for the reduction of the computational effort needed to solve the resulting large scale optimization problems and subproblems. Earlier attempts involved solving *k* SVM models, where *k* is the number of classes and *k(k-1)/2* is the number of SVM classifiers [2, 3]. Other attempts involved the solution of a single optimization problem using all data at once [3, 4, 5]. The latter are arguably the most well constructed multi-class formulations most closely aligned with Vapnik's structural minimization principle [1].

Our main contribution will be a multi-classification formulation which will be expressed as a single optimization problem. We will look at the development of a pairwise multi-classification support vector machine (QP-P$_A$MSVM) expressed as a quadratic optimization problem. Similar to the two-class problem we will formulate the optimal pairwise separator.

This paper is organized as follows. In section 2 we review the SVM for the binary case. In section 3 we present a quadratic programming pairwise multi-classification support vector machine (QP-P$_A$MSVM). In section 4 we give computational results for two real data sets, and finally, section 5 concludes the paper.

2 Support Vector Machines

In this section we consider the two-class classification problem. The SVM avoids overfitting by maximizing the margin between two classes of training data, i.e., maximizing the distance between the separating hyperplane and the training data on either side of it.

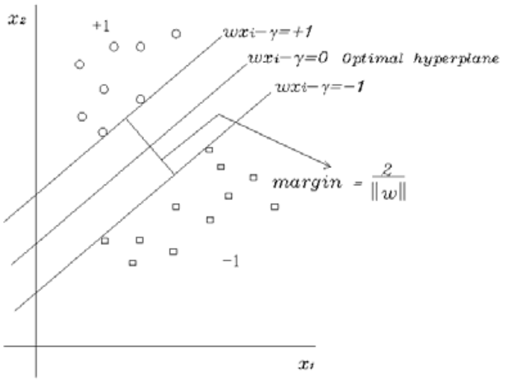

x_2
+1
$wx_i-\gamma=+1$
$wx_i-\gamma=0$ Optimal hyperplane
$wx_i-\gamma=-1$
margin = $\frac{2}{\|w\|}$
−1
x_1

Fig 1: A Support Vector Machine classification problem; the optimal hyperplane is orthogonal to the shortest line connecting the two classes, and intersects it halfway.

The formulation can be written in its primal form [1, 5, 6, 7, 8] as follows:

$$\begin{aligned} &\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l} \xi_i \\ &s.t.\ y_i\left(w \cdot x_i - \gamma\right) + \xi_i \geq 1 \\ &\quad \xi_i \geq 0 \ \ i = 1,\ldots,l \end{aligned} \tag{1}$$

where $x_i \in \Re^d$ are the input training vectors, $y_i \in \{+1,-1\}$ are the corresponding labels, $\|w\|^2 = w^T w$ is the square of the 2-norm of the weight vector defining the separating hyperplane, and ξ_i is a non-negative slack (penalty term) that measures the degree of violation of the constraints. The parameter C is a constant, called the regularization parameter, which controls the trade-off between minimizing training errors and minimizing the norm of the weight vector (generalization ability).

3 Pairwise Multi-classification Support Vector Machines

In pairwise classification, we train a classifier for each possible pair of classes. For *k* classes, this results to *k(k-1)/2* SVM classifiers. For the multi-classification case we express all *k* classes as a single optimization problem that will produce *k(k-1)/2* SVM classifiers.

Given that the data sets in R^n are represented by a matrix $A^i \in R^{m_i \times n}$, where $i = 1,..,k$ (k classes).

Let A^i be an $m_i \times n$ matrix whose rows are points in the ith class.

Let A^j be a $m_j \times n$ matrix whose rows are points in the jth class. Then if $x \in R^n$ can be classified as follows:

$$\begin{aligned} &x^T w^{ij} - \gamma^{ij} > 0, \quad x \in A^i \\ &x^T w^{ij} - \gamma^{ij} < 0, \quad x \in A^j, \quad i < j \end{aligned} \tag{2}$$

In the separable case, the pairwise linear discriminant function between two classes must satisfy the following set of inequalities: Find $w^{ij} \in R^n$ and $\gamma^{ij} \in R$, such that

$$A^i w^{ij} > \gamma^{ij} e, \quad \gamma^{ij} e > A^j w^{ij} \quad i < j,$$

(3)

where e is a vector of ones of appropriate dimension. If such a w^{ij} and γ^{ij} exist, we say that the sets are pairwise linearly separable. To classify a new point x, we employ the “Max Wins” strategy. This is a voting approach [2, 3]. For example, if the sign of [$x^T w^{ij} - \gamma^{ij}$] gives that x is in the *i*th class, then the vote for the *i*th class is increased by one. Otherwise, the *j*th is increased by one. Hence we predict x as being in the class with the largest vote. In the case that those two classes have identical votes, we select the one with the smallest index.

3.1 P$_A$MSVM linear separability formulation

We propose the construction of a pairwise linear and pairwise nonlinear SVM using a single quadratic program (QP). Like in the dichotomous case we formulate the optimal pairwise linear separator for the separable case. For the pairwise separable case there exists a $w^{ij} \in R^n$ and $\gamma^{ij} \in R$, such that

$$A^i w^{ij} > \gamma^{ij} e \quad \gamma^{ij} e > A^j w^{ij} \quad i < j, \tag{4}$$

Since infinitely many w^{ij} and γ^{ij} exist, the optimal solution would provide the largest margin of classification. The margin of separation between classes i and j is $2 / \|w^{ij}\|$.

Therefore, one would minimize $\|w^{ij}\|$ for $i < j$.

Let $A^{ij} = \begin{bmatrix} A^i \\ A^j \end{bmatrix}$ and $y^{ij} = \pm 1$ for classes i and j respectively.

For the pairwise linearly separable problem we formulate the constrained optimization problem as below:

$$\begin{aligned} &\min_{w,\gamma} \ \frac{1}{2}\sum_{i<j}^{k} \|w^{ij}\|^2 \\ &s.t. \ y^{ij}(A^{ij} w^{ij} - e\gamma^{ij}) \ge e, \quad i < j \end{aligned} \tag{5}$$

Here is a 3 classes problem (k = 3) rewritten in matrix notation

Let

$$C = \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{pmatrix},$$

$$A = \begin{pmatrix} A^1 & 0 & 0 \\ -A^2 & 0 & 0 \\ 0 & A^1 & 0 \\ 0 & -A^3 & 0 \\ 0 & 0 & A^2 \\ 0 & 0 & -A^3 \end{pmatrix} \quad E = \begin{pmatrix} -e^1 & 0 & 0 \\ e^2 & 0 & 0 \\ 0 & -e^1 & 0 \\ 0 & e^3 & 0 \\ 0 & 0 & -e^2 \\ 0 & 0 & e^3 \end{pmatrix}$$

where the identity matrix $I \in R^{n \times n}$, $A^i \in R^{m_i \times n}$, $A^j \in R^{m_j \times n}$ $i < j$, $e^i \in R^{m_i \times 1}$ and $e^j \in R^{m_j \times 1}$ $i < j$ are the vectors of ones. So, for k > 2 problem (5) can be expressed in the following form:

$$\begin{aligned} &\min_{w,\gamma} \ \frac{1}{2}\|Cw\|^2 \\ &s.t. \ Aw + E\gamma - e \ge 0, \quad i < j \end{aligned}$$

(6)

where

$$w = \left[w^{12^T}, w^{13^T}, \ldots, w^{(k-1)k^T} \right]^T \ and \ \gamma = \left[\gamma^{12}, \gamma^{13}, \ldots, \gamma^{(k-1)k} \right]^T$$

The constrained optimization is solved by introducing Lagrange multipliers $\alpha \ge 0$ and a Lagrangian

$$L(\alpha, w, \gamma) = \frac{1}{2}\|Cw\|^2 - \alpha^T(Aw + E\gamma - e) \tag{7}$$

Differentiating the Lagrangian with respect to $w \ and \ \gamma$ leads to

$$\frac{dL}{dw} = (C^T C)w - A^T\alpha = 0 \tag{8}$$

$$\frac{dL}{d\gamma} = -E^T\alpha = 0 \tag{9}$$

To eliminate variables $w \ and \ \gamma$ from the Lagrangian, matrix $(C^T C)$ needs to be an invertible matrix. Since $(C^T C)$ is invertible, we have

$$(C^T C)^{-1} A^T = A^T \tag{10}$$

So from equations (8) & (10) and (9) we can obtain the relations

$$\begin{aligned} w &= (C^T C)^{-1} A^T \alpha = A^T \alpha \\ 0 &= E^T \alpha \end{aligned} \tag{11}$$

Using the relationships in (11) we eliminate (w, γ) from the Lagrangian and we obtain the Wolfe dual quadratic optimization problem below:

$$\begin{aligned} L(\alpha, w, \gamma) &= \frac{1}{2}\|Cw\|^2 - \alpha^T(Aw + E\gamma - e) \\ &= \frac{1}{2}\langle Cw, Cw\rangle - \alpha^T(Aw + E\gamma - e) \\ &= \frac{1}{2} w^T C^T C w - \alpha^T(Aw + E\gamma - e) \\ \Rightarrow L(\alpha) &= \frac{1}{2}\alpha^T A A^T \alpha - \alpha^T A A^T \alpha - 0 + \alpha^T e \\ L(\alpha) &= \alpha^T e - \frac{1}{2}\alpha^T A A^T \alpha \end{aligned} \tag{12}$$

In matrix notation the Wolfe dual becomes

$$\begin{aligned} &\max_{\alpha} \ \alpha^T e - \frac{1}{2}\alpha^T A A^T \alpha \\ &s.t. \quad E^T \alpha = 0 \\ &\qquad \alpha \ge 0 \end{aligned} \tag{13}$$

where $\alpha = \left[\alpha^{ij^T}, \alpha^{ji^T}, \ldots, \alpha^{(k-1)k^T}, \alpha^{k(k-1)^T} \right]^T, \ i < j$

Solving for w in matrix notation

$$w^{ij} = A^{i^T} \alpha^{ij} - A^{j^T} \alpha^{ji}$$

Using a summation notation the Wolfe dual of problem (5) is given as

$$\begin{aligned}&\max_{\alpha} \sum_{i<j}^{k}\sum_{c=1}^{m_{ij}}\alpha_c^{ij} - \frac{1}{2}\sum_{i<j}^{k}\sum_{c,d=1}^{m_{ij}}\alpha_c^{ij}\alpha_d^{ij}y_c^{ij}y_d^{ij}A_c^{ij}A_d^{ij^T}\\ &s.t.\ \sum_{c=1}^{m_{ij}}\alpha_c^{ij}y_c^{ij}=0 \qquad (14)\\ &\qquad \alpha_c^{ij}\geq 0,\ \ i<j\end{aligned}$$

Where m_{ij} is the number of training points for pairwise comparison between classes i and j.

Solving for w^{ij} : $w^{ij} = \sum_{c=1}^{m_{ij}}\alpha_c^{ij}y_c^{ij}A_c^{ij}$

3.2 P_AMSVM nonlinear separability formulation

For the pairwise nonlinearly separability, we employ the kernel trick to map the input data into a higher dimension feature space using a kernel function [9].

Replacing $A^{ij}A^{ij^T}$ with a kernel $K(A^{ij}, A^{ij^T})$, the dual problem in (14) becomes

$$\begin{aligned}&\max_{\alpha} \sum_{i<j}^{k}\sum_{c=1}^{m_{ij}}\alpha_c^{ij} - \frac{1}{2}\sum_{i<j}^{k}\sum_{c,d=1}^{m_{ij}}\alpha_c^{ij}\alpha_d^{ij}y_c^{ij}y_d^{ij}K(A_c^{ij}, A_d^{ij^T})\\ &s.t.\ \sum_{c=1}^{m_{ij}}\alpha_c^{ij}y_c^{ij}=0 \qquad (15)\\ &\qquad \alpha_c^{ij}\geq 0,\ \ i<j\end{aligned}$$

In matrix notation the dual problem (14) becomes

$$\begin{aligned}&\max_{\alpha}\ \alpha^T e - \frac{1}{2}\alpha^T K(A, A^T)\alpha\\ &s.t.\ \ E^T\alpha = 0 \qquad (16)\\ &\qquad \alpha\geq 0,\ \ i<j\end{aligned}$$

3.3 P_AMSVM inseparability formulation

To construct a pairwise inseparable classifier, we introduce a parameter λ. This parameter is a constant called the regularization parameter, which controls the trade-off between minimizing training errors and minimizing the norm of the weight vector (generalization ability).

Adding the error criterion $\xi \geq 0$ and weighting it with the regularization parameter λ, then the primal problem (6) becomes

$$\begin{aligned}&\min_{w,\gamma}\ \frac{1}{2}\|Cw\|^2 + \lambda e^T\xi\\ &s.t.\ \ Aw + E\gamma - e + \xi \geq 0 \qquad (17)\\ &\qquad \xi\geq 0,\ \ i<j\end{aligned}$$

and the dual of problem (17) becomes

$$\begin{aligned}&\max_{\alpha}\ \alpha^T e - \frac{1}{2}\alpha^T AA^T\alpha\\ &s.t.\ \ E^T\alpha = 0 \qquad (18)\\ &\qquad 0\leq\alpha\leq\lambda,\ \ i<j\end{aligned}$$

Replacing $A^{ij}A^{ij^T}$ with a kernel $K(A^{ij}, A^{ij^T})$ and using the summation notation the dual problem (18) can be expressed as

$$\begin{aligned}&\max_{\alpha} \sum_{i<j}^{k}\sum_{c=1}^{m_{ij}}\alpha_c^{ij} - \frac{1}{2}\sum_{i<j}^{k}\sum_{c,d=1}^{m_{ij}}\alpha_c^{ij}\alpha_d^{ij}y_c^{ij}y_d^{ij}K(A_c^{ij}, A_d^{ij^T})\\ &s.t.\ \sum_{c=1}^{m_{ij}}\alpha_c^{ij}y_c^{ij}=0 \qquad (19)\\ &\qquad 0\leq\alpha_c^{ij}\leq\lambda,\ \ i<j\end{aligned}$$

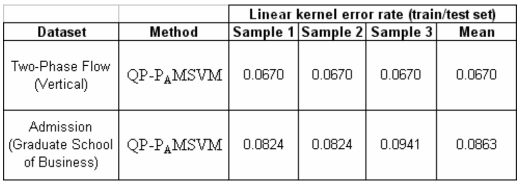
4 Numerical Testing

In this section we present the computational results that utilize multi-class classification formulation of problem (18) for discriminating between k classes. Experiments were carried out on a vertical 1 inch two-phase flow dataset [10] and the admission data for graduate school of business [11]. Description of datasets is as follows:

Vertical Two-Phase Flow Dataset: The two-phase flow dataset [10] uses a pair of flow rates (superficial gas and liquid velocity) to delineate the flow regime. There are 209 instances (points) and 2 attributes (features). The distribution of instances with respect to their class is as follows: 44 instances in class 1 (bubble flow), 102 instances in class 2 (intermittent flow), and 63 instances in class 3 (annular flow).

Admission Data for Graduate School of Business: The admission data dataset [11] uses the undergraduate grade point average (GPA) and graduate management aptitude test (GMAT) scores to help determine which applicants should be admitted to the school's graduate program. There are 85 instances (points) and 2 attributes (features). The distribution of instances with respect to their class is as follows: 28 instances in class 1 (not admitted), 26 instances in class 2 (borderline), and 31 instances in class 3 (admitted).

The QP formulation was implemented using the optimization and matrix decomposition routines in the MATLAB [12] software. The two-phase flow dataset were scaled by taking the natural logarithm of each instance, while the admission data were scaled using the unit vector normalization for the QP formulation. The methods were trained on 50% of the dataset, and tested on the whole dataset (50% training, 50% testing data), all randomly drawn from the dataset to obtain 3 training sample data. We report the results of the quadratic programming (QP) formulation of P_AMSVM ($\lambda = 1$).



		Linear kernel error rate (train/test set)			
Dataset	Method	Sample 1	Sample 2	Sample 3	Mean
Two-Phase Flow (Vertical)	QP-P_AMSVM	0.0670	0.0670	0.0670	0.0670
Admission (Graduate School of Business)	QP-P_AMSVM	0.0824	0.0824	0.0941	0.0863

Table 1: Performance of P_AMSVM

Table 1 contains the results for the QP-P_AMSVM on the Two-Phase flow and Admission dataset. The errors rates are low enough to demonstrate the capability of the model. The linear kernel employed was adequate enough to give a low misclassification error, however further studies could involve the use of nonlinear kernels in problem (19). Note that the quality of the solution is dependent on the choice of λ. For this problem $\lambda = 1$ was sufficient enough to present good results. Further computational studies could be of interest such as varying choices of λ to determine the effect on the solution of the QP MSVM model.

5 Conclusion and Future Work

In this paper we presented an extension of the binary SVMs to the multi-class SVMs. We have used a quadratic programming formulation. The proposed method presents an accurate and good alternative to existing earlier methods based on solving *k* SVM models which can become computationally intensive due to the number of SVM models one would have to solve in order to discriminate between *k* classes.

Formulation (18) was applied to the Two-Phase flow and Admission dataset and the results are very encouraging considering that the kernel used is a linear kernel. The linear kernel indicates that the both datasets are linearly separable but with a tolerable misclassification rate (see Table 1). Future work will involve the investigation of nonlinear kernels to solve nonlinear classification problems. Mercer's condition is applicable, so kernel functions can be incorporated into the MSVM methods. Further computational studies would include the implementation of several preprocessing schemes to normalize the data and varying the choice of meta-parameter λ to help determine the effect on the model performance.

Acknowledgements: The present work has been partially supported by the NSF grant EIA-0205628.

References

[1] Vapnik, V. *Statistical Learning Theory.* John Wiley & Sons, Inc., **1998**.

[2] Santosa, B.; Conway, T.; Trafalis, T. B. Knowledge Based-Clustering and Application of Multi-Class SVM for Genes Expression Analysis, *Intelligent Engineering Systems through Artificial Neural Networks* **2002**, 12, 391 – 395.

[3] Hsu, C-W.; Lin, C-J. A Comparison of Methods for Multi-class Support Vector Machines, *IEEE Transactions on Neural Networks* **2002** 13, 415 – 425.

[4] Bredensteiner E. J.; Bennet, K. P. Multicategory Classification by Support Vector Machines, *Computational Optimization and Applications* **1999**, 12, 53 – 79.

[5] Cristianini, N.; Shawe-Taylor, J. *Support Vector Machines and other kernel-based learning methods.* Cambridge University Press, Cambridge, UK, **2000**.

[6] Burges, C.J.C. A tutorial on support vector machines for pattern classification. *Data Mining and Knowledge Discovery,* 2(2): **1998**, 121-167.

[7] Chang, C-C.; Lin, C-J. *LIBSVM: A Library for Support Vector Machines* **2001**, http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[8] Hsu, C-W.; Chang, C-C.; Lin, C-J. *A Practical Guide to Support Vector Classification*, Technical Report Department of Computer Science and Information Engineering, National Taiwan University, Taipei 106, Taiwan, **2003**.

[9] Scholokopf, B. *Statistical Learning and Kernel Methods.* Technical Report MSR-TR-2000-23, Microsoft Research Ltd., Microsoft Corporation, **2000**

[10] Trafalis, T. B.; Oladunni, O.; Papavassiliou, D. V. Two-Phase Flow Regime Identification with a Multi-Classification SVM Model, School of Industrial Engineering, College of Engineering, University of Oklahoma. (accepted) to appear in the *Industrial & Engineering Chemistry Research*.

[11] Johnson, R. A.; Wichern, D. W. *Applied Multivariate Statistics Analysis*; Prentice Hall: New Jersey, **2002**.

[12] *MATLAB User's Guide*. The Math-Works, Inc., Natwick, MA 01760, 1994-2003. http://www.mathworks.com.

[13] Bazaraa, M.S.; Sherali, H.D.; Shetty, C.M.; *Nonlinear Programming – Theory and Algorithms*; John Wiley & Sons, Inc., **1993**

[14] Reklaitis, G.V.; Ravindran, A.; Ragsdell, K.M. B.; *Engineering Optimization: Methods and Applications*; John Wiley & Sons, Inc., **1983**