Pattern Recognition Letters 42 (2014) 91–100

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Enhanced multi-weight vector projection support vector machine ☆

Qiaolin Ye [a,*], Ning Ye [a], Tongming Yin [b]

[a] College of Information Science and Technology, Nanjing Forestry University, Nanjing, PR China
[b] College of Wood Science and Technology, Nanjing Forestry University, Nanjing, PR China

ARTICLE INFO

Article history:
Received 13 June 2013
Available online 19 February 2014

Keywords:
Multiple weight vectors
Classification
Support vector machines
Multisurface support vector machine

ABSTRACT

Recently, we have developed an effective classifier, called *Multi-weight vector projection support vector machine* (MVSVM). Like traditional multisurface support vector machine *Generalized-Eigenvalue-based Mulitisurface Support Vector Machine* (GEPSVM), MVSVM can fast complete the computation and simultaneously handle the complex Exclusive Or (XOR) problems well. In addition, MVSVM still shows the more promising results than GEPSVM for different classification tasks. Despite the effectiveness of MVSVM, there is a serious limitation, which is that the number of the projection weight vectors for each class is limited to one. Intuitively, it is not enough to use only one projection weight vector for each class to obtain better classification. In order to address this problem, we, in this paper, develop enhanced MVSVM (EMVSVM), which is based on MVSVM. For a particular class, EMVSVM maximizes the distances from its projected average vector to the projected points from different classes to find better separability, which is different from MVSVM which maximizes the separability between classes by enforcing the maximization of the distances between the average vectors of different classes. Doing so can make EMVSVM obtain more than one discriminative weight-vector projections for each class due to that the rank of the newly-formed between-class scatter matrix is enlarged. From the statistical viewpoint, we analyze the proposed approach. Experimental results on public datasets indicate the effectiveness and efficiency of EMVSVM.

1. Introduction

Support vector machine (SVM) [1] is a widely-used statistical learning method, which has been widely applied to a variety of real-world problems, such as handwritten digit recognition, bioinformatics, face recognition, and text categorization. SVM earns its place as a classification tool because of its effectiveness as well as its solid theoretical foundation.

However, SVM is computationally expensive, especially when handling large-scale data. This is so because its solution follows from solving a *quadratic convex programming problem* (QPP) with numerous constraints. Furthermore, SVM cannot well cope with the complex *Exclusive Or* (XOR) problems. To boost the computing speed of SVM, past years can see many fast SVM algorithms [2–5]. The most representative two are LIBSVM and LIBLINEAR which are always compared with other classifiers [6,7]. However, it is worthwhile to note that the time complexity of most of existing fast SVM algorithms depends on the number of samples [8], thus they may not get very satisfactory efficiency for massive- or large-scale problems. Moreover, these algorithms are to seek for only a separating plane, resulting in the difficulty that the complex XOR problems cannot be well solved [9].

In 2006, Mangasarian and Wild [9] introduced a multisurface classifier for binary data classification, referred to as *multisurface proximal support vector machines via generalized eigenvalues* (GEPSVM). GEPSVM is originally motivated to solve the XOR problems and simultaneously reduce the computing time of SVM. Instead of solving only one plane, it solves two non-parallel planes. Each nonparallel plane is obtained by solving the eigenvector corresponding to a smallest eigenvalue of a generalized eigenvalue problem, such that each plane is as close as possible to the samples for its own class and meanwhile as far as possible from the samples for the other classes [9]. Although GEPSVM has many advantages, it is still confronted with many problems, such as the unstable solution and the insufficiently good generalization for more complex data sets [10]. Therefore, our previous work [10] proposed *multi-weight vector projection support vector machine* (MVSVM). The central idea in the stand-alone classifier MVSVM is to seek for two optimal weight vector projections instead of two planes. Each weight vector projection is obtained as the eigenvector corresponding to the maximum eigenvalue of a standard eigenvalue

☆ This paper has been recommended for acceptance by Jie Zou.
* Corresponding author. Tel./fax: +86 15996460537.
E-mail address: yeqiaolin65620868@163.com (Q. Ye).

http://dx.doi.org/10.1016/j.patrec.2014.02.006
0167-8655/

problem, such that each of two data sets are closest to one of two class means and meanwhile the points sharing different labels are separated as far as possible. For an unseen sample, according to the decision rule of MVSVM, it will be assigned to the closest class mean (when projected onto the found direction) [10]. MVSVM exhibits the better generalization than GPESVM. This algorithm finds only a weight-vector projection for each class, which may be not enough for the acquisition of the optimal classification performance [11]. In the recent research [11], we extended MVSVM to *Recursive Projection Twin Support Vector Machine* (RPTSVM). RPTSVM is based on MVSVM, which simply casts MVSVM as two related SVM-type problems and seeks for more than one weight vectors for each class by employing a recursive procedure. Due to the recursive procedure and the related SVM-type problem, RPTSVM requires considerable computational cost. Moreover, it is not easy to use the efficient techniques in fast SVM algorithms like LIBSVM to solve the RPTSVM problem, due to the different objective function from SVM. These make RPTSVM not practical in real applications.

In this paper, we propose a novel classifier, called *enhanced multi-weight vector projection support vector machine* (EMVSVM). In MVSVM, the between-class scatter matrix is computed as the sum of the distances from the average vector of a particular class to average vectors of other classes. The essential property of this between-class scatter matrix is that its rank is limited to one. As can be seen from the objective functions of MVSVM, the two eigenvalue-based optimization problems in the MVSVM pair are two subproblems of *Linear Discriminat Analysis* (LDA) [13] in two-class settings. It is necessary to point out that MVSVM is used for classification while LDA can be used as the processing tool of MVSVM. According to the previous work [12], the separability information is, generally, hidden in the eigenvectors corresponding to the positive eigenvalues of the MVSVM eigen-equations, and the maximum number of the positive eigenvalues is equal to the rank of the between-class matrix. To enlarge the rank of the between-class matrix, EMVSVM redefines the between-class matrix used for measuring the separability between classes. In this way, the proposed approach is capable of finding more than one weight vectors for each class. EMVSVM finds all weight vectors by solving only two standard eigenvalue problems instead of multiple pairs of related SVM-type problems in the work [11], and thus has comparable time complexity to MVSVM. Experiments conducted on public datasets demonstrate that the efficiency and effectiveness of EMVSVM.

2. Related work

Suppose that we are given a training point set of two classes $\mathbf{X}^{(l)} = \left[\mathbf{x}_1^{(l)}, \mathbf{x}_2^{(l)}, \ldots, \mathbf{x}_{N_l}^{(l)}\right]$ $(l = 1, 2)$ in the n-dimensional input space $\mathbf{R}^n$, denoted by the $m_1 \times n$ matrix $\mathbf{A}$ belonging to class +1 and the $m_2 \times n$ matrix $\mathbf{B}$ belonging to class -1, where $m_1 + m_2 = m$. In the following, we review two eigenvalue-type fast support vector machine classifiers GEPSVM [9] and MVSVM [10].

2.1. Generalized eigenvalue support vector machine classifier

GEPSVM [9] is to determine the following two nonparallel planes in the n-dimensional input space

$$\mathbf{x}^T\mathbf{w}_1 - b_1 = 0, \quad \text{and} \quad \mathbf{x}^T\mathbf{w}_2 - b_2 = 0 \tag{1}$$

so as to minimize the distance from the planes to the points of class +1 and class -1, respectively, where $\mathbf{x}^T$ denotes the transpose of $\mathbf{x}$, and $(\mathbf{w}_i, b_i) \in (\mathbf{R}^n \times \mathbf{R})$ $(i = 1, 2)$. This leads to the following optimization problem

$$\min_{\mathbf{w}_1, b_1} \frac{||\mathbf{A}\mathbf{w}_1 - \mathbf{e}_1 b_1||^2 + \delta \left|\left| \begin{bmatrix} \mathbf{w}_1 \\ b_1 \end{bmatrix} \right|\right|^2}{||\mathbf{B}\mathbf{w}_1 - \mathbf{e}_2 b_1||^2}, \quad \text{and}$$
$$\min_{\mathbf{w}_2, b_2} \frac{||\mathbf{B}\mathbf{w}_2 - \mathbf{e}_2 b_2||^2 + \delta \left|\left| \begin{bmatrix} \mathbf{w}_2 \\ b_2 \end{bmatrix} \right|\right|^2}{||\mathbf{A}\mathbf{w}_2 - \mathbf{e}_1 b_2||^2} \tag{2}$$

where $\mathbf{e}_1$ and $\mathbf{e}_2$ are two column vectors of one of appropriate dimensions, and $\delta > 0$ is a regularized parameter to improve the generalization. Let us define

$$\mathbf{G} = [\mathbf{A} - \mathbf{e}_1]^T[\mathbf{A} - \mathbf{e}_1] + \delta\mathbf{I} \quad \mathbf{H} = [\mathbf{B} - \mathbf{e}_2]^T[\mathbf{B} - \mathbf{e}_2] \; \mathbf{z}_1 = \begin{bmatrix} \mathbf{w}_1 \\ b_1 \end{bmatrix} \tag{3}$$

and

$$\mathbf{L} = [\mathbf{B} - \mathbf{e}_2]^T[\mathbf{B} - \mathbf{e}_2] + \delta\mathbf{I} \quad \mathbf{M} = [\mathbf{A} - \mathbf{e}_1]^T[\mathbf{A} - \mathbf{e}_1] \; \mathbf{z}_2 = \begin{bmatrix} \mathbf{w}_2 \\ b_2 \end{bmatrix} \tag{4}$$

The optimization problem in (2) is simplified as

$$\min_{\mathbf{z}_1} \frac{\mathbf{z}_1^T\mathbf{G}\mathbf{z}_1}{\mathbf{z}_1^T\mathbf{H}\mathbf{z}_1}, \quad \text{and} \quad \min_{\mathbf{z}_2} \frac{\mathbf{z}_2^T\mathbf{L}\mathbf{z}_2}{\mathbf{z}_2^T\mathbf{M}\mathbf{z}_2} \tag{5}$$

The solutions of (5) can be yielded by solving the following generalized eigenvalue problems: $\mathbf{G}\mathbf{z}_1 = \lambda_1\mathbf{H}\mathbf{z}_1$ and $\mathbf{L}\mathbf{z}_2 = \lambda_2\mathbf{M}\mathbf{z}_2$. Once the solutions for $\mathbf{z}_1$ and $\mathbf{z}_2$ are obtained, the planes for class +1 and -1 can be expressed. For an unknown data point $\mathbf{x} \in \mathbf{R}^n$, its class is computed as

$$\text{class}(\mathbf{x}) = \min_{l=1,2} |\mathbf{w}_l^T\mathbf{x} - b_l|, \tag{6}$$

where $|\cdot|$ is the perpendicular distance of the point $\mathbf{x}$ from the plane of a particular class.

2.2. Multi-weight vector projection support vector machine classifier

Like GEPSVM, MVSVM [10] has two eigenvalue formulations, which, however, is different from GEPSVM in spirit. Instead of aiming to finding the specific planes in (1), MVSVM aims to find the weight-vector projections $\mathbf{w}_1$ and $\mathbf{w}_2$ for the respective class.

The optimization criteria of MVSVM are given by

$$\begin{aligned} &\max_{\mathbf{w}_1} \left(\mathbf{w}_1^T \frac{1}{m_1} \sum_{j=1}^{m_1} \mathbf{x}_j^{(1)} - \mathbf{w}_1^T \frac{1}{m_2} \sum_{j=1}^{m_2} \mathbf{x}_j^{(2)} \right)^2 \\ &\quad - \beta \sum_{i=1}^{m_1} \left(\mathbf{w}_1^T \mathbf{x}_i^{(1)} - \mathbf{w}_1^T \frac{1}{m_1} \sum_{j=1}^{m_1} \mathbf{x}_j^{(1)} \right)^2 \\ &\text{s.t. } ||\mathbf{w}_1||^2 = 1 \end{aligned} \tag{7}$$

$$\begin{aligned} &\max_{\mathbf{w}_2} \left(\mathbf{w}_2^T \frac{1}{m_2} \sum_{j=1}^{m_2} \mathbf{x}_j^{(2)} - \mathbf{w}_2^T \frac{1}{m_1} \sum_{j=1}^{m_1} \mathbf{x}_j^{(1)} \right)^2 \\ &\quad - \beta \sum_{i=1}^{m_2} \left(\mathbf{w}_2^T \mathbf{x}_i^{(2)} - \mathbf{w}_2^T \frac{1}{m_2} \sum_{j=1}^{m_2} \mathbf{x}_j^{(2)} \right)^2 \\ &\text{s.t. } ||\mathbf{w}_2||^2 = 1 \end{aligned} \tag{8}$$

where β is a free trade-off parameter. According to the above criteria, MVSVM can find two optimal weight-vector projections (each for a particular class), such that each of two data sets are closest to one of two class means and meanwhile the points sharing different labels are separated as far as possible.

Define

$$\mathbf{S}_1 = \sum_{i=1}^{m_1} \left(\mathbf{x}_i^{(1)} - \frac{1}{m_1} \sum_{j=1}^{m_1} \mathbf{x}_j^{(1)} \right) \left(\mathbf{x}_i^{(1)} - \frac{1}{m_1} \sum_{j=1}^{m_1} \mathbf{x}_j^{(1)} \right)^T \tag{9}$$