Support Matrix Machines

Luo Luo RICKY@SJTU.EDU.CN
Yubo Xie YUBOTSE@SJTU.EDU.CN
Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China

Zhihua Zhang ZHIHUA@SJTU.EDU.CN
Institute of Data Science, Department of Computer Science and Engineering, Shanghai Jiao Tong University, China

Wu-Jun Li LIWUJUN@NJU.EDU.CN
National Key Laboratory for Novel Software Technology, Collaborative Innovation Center of Novel Software Technology and Industrialization, Department of Computer Science and Technology, Nanjing University, China

Abstract

In many classification problems such as electroencephalogram (EEG) classification and image classification, the input features are naturally represented as matrices rather than vectors or scalars. In general, the structure information of the original feature matrix is useful and informative for data analysis tasks such as classification. One typical structure information is the correlation between columns or rows in the feature matrix. To leverage this kind of structure information, we propose a new classification method that we call *support matrix machine* (SMM). Specifically, SMM is defined as a hinge loss plus a so-called *spectral elastic net* penalty which is a spectral extension of the conventional elastic net over a matrix. The spectral elastic net enjoys a property of grouping effect, i.e., strongly correlated columns or rows tend to be selected altogether or not. Since the optimization problem for SMM is convex, this encourages us to devise an alternating direction method of multipliers (ADMM) algorithm for solving the problem. Experimental results on EEG and image classification data show that our model is more robust and efficient than the state-of-the-art methods.

1. Introduction

Classical classification methods such as support vector machines (SVMs) (Cortes & Vapnik, 1995) and logistic regression (Hastie et al., 2001) have been originally built on the case that input samples are represented as vectors or scalars. However, it is also often met that input samples are naturally represented as two-dimensional matrices or tensors. When using classical classification methods to data of matrix representation, we have to reshape the input matrices into vectors. However, this would destroy the structure information of the data matrix, e.g., the correlation of different channels of electroencephalogram (EEG) data (Zhou & Li, 2014) and the spatial relationship of the nearby pixels of image data (Wolf et al., 2007). Moreover, if the data matrix is stacked (reshaped) into a vector, the dimensionality of the resulting vector typically becomes very high, which in turn leads to the curse of dimensionality.

There has been some work on classification methods which attempt to exploit the correlation between the columns or rows of the data matrix. Usually, such a classification method introduces a matrix of regression coefficients to leverage the correlation within the data matrix. For example, Wolf et al. (2007) proposed a rank-k SVM, which models the regression matrix as the sum of k rank-one orthogonal matrices. Pirsiavash et al. (2009) devised a bilinear SVM by factorizing the regression matrix into two low-rank matrices. Cai et al. (2006) proposed a similar bilinear framework called support tensor machines for text categorization. These methods essentially take advantage of the low-rank assumption, which can be used for describing the correlation within a matrix. However, their treatments result in non-convex optimization problems.

In this paper we are also concerned with the classification problems on a set of data matrices. Our work is motivated by the use of the nuclear norm (a.k.a., the trace norm) in low-rank matrix approximation (Srebro & Shraibman, 2005), matrix completion (Candès & Recht, 2009; Liu et al., 2013; Salakhutdinov & Srebro, 2010; Huang et al., 2013), and multi-task learning problems (Pong et al., 2010;

Proceedings of the 32^{nd} International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37.

Harchaoui et al., 2012; Kang et al., 2011). The cornerstone of these methods is to use the nuclear norm of a matrix as a convex alternative of the matrix rank. Since the nuclear norm is the best convex approximation of the matrix rank over the unit ball of matrices, this makes it more tractable to solve the resulting optimization problem. Moreover, some nice properties such as the consistency results of the nuclear norm minimization have been studied by Bach (2008). There has been some work which applies the nuclear norm penalization with least square loss function to matrix regression problems (Hunyadi et al., 2012; Signoretto et al., 2014). Recently, Zhou & Li (2014) applied the nuclear norm penalization to matrix regression problems based on generalized linear models (GLMs).

In this paper we propose a new model to address the matrix classification problem. Our model includes two principal ingredients. First, we consider the hinge loss due to its widely deployed ability in sparseness and robustness modeling. Second, we employ a *spectral elastic net* penalty for the regression matrix. The spectral elastic net is a spectral extension of the conventional elastic net over a matrix. In parallel to the conventional elastic net (Zou & Hastie, 2005) which is the combination of the ridge penalty and lasso penalty, our spectral elastic net is the combination of the squared Frobenius matrix norm and nuclear norm. We prove that the spectral elastic net enjoys the property of grouping effect which is similar to the conventional elastic net, while keeping a low-rank representation. We show that the regression matrix in our model is indeed a combination of a set of support matrices. We thus refer to our model as a *support matrix machine* (SMM).

The optimization problem for SMM is convex but the hinge loss function is not smooth. Fortunately, we can resort to an alternating direction method of multipliers (ADMM) studied in (Goldstein et al., 2012). Specifically, we develop an iteration algorithm, which is mainly built on ADMM and a singular value thresholding (SVT) operator (Candès & Recht, 2009; Cai et al., 2010). The algorithm converges quickly and promises to get the global optimal solution. It is worth pointing out that the algorithm requires repeatedly computing singular value decomposition (SVD) of matrices with the same size as the matrix of input features. However, when represented as a matrix, the size of an input matrix is usually not too large.

Finally, we apply our SMM to EEG and image classification problems. We see that classification methods directly working on data matrices outperform those on vectors such as the conventional SVM. When data are contaminated with non-Gaussian noise or outliers, our SMM has significant improvements over baselines. This implies that SMM is robust and has potential applications in matrix classification problems with noises. Moreover, the experiments show that our proposed training algorithm for SMM is efficient.

The remainder of the paper is organized as follows. In Section 2, we give the notation and preliminaries. In Section 3, we review our concerned problem. In Section 4 we present our model and the learning algorithm. In Section 5, we conduct experimental analysis to justify our methods. Finally, we conclude our work in Section 6.

2. Notation and Preliminaries

In this section we give the notation and preliminaries which will be used in this paper. We let $\mathbf{I}_p$ denote the $p\times p$ identity matrix. For a vector $\mathbf{a}\in\mathbb{R}^p$, the Euclidean norm is denoted as $||\mathbf{a}|| = \sqrt{\sum_{i=1}^p a_i^2}$. For a matrix $\mathbf{A}\in\mathbb{R}^{p\times q}$ of rank r where $r\le\min(p,q)$, we let the condensed singular value decomposition (SVD) of $\mathbf{A}$ be $\mathbf{A}=\mathbf{U}_A\mathbf{\Sigma}_A\mathbf{V}_A^T$ where $\mathbf{U}_A\in\mathbb{R}^{p\times r}$ and $\mathbf{V}_A\in\mathbb{R}^{q\times r}$ satisfy $\mathbf{U}_A^T\mathbf{U}_A=\mathbf{I}_r$ and $\mathbf{V}_A^T\mathbf{V}_A=\mathbf{I}_r$, and $\mathbf{\Sigma}_A=\mathrm{diag}(\sigma_1(\mathbf{A}),\cdots,\sigma_r(\mathbf{A}))$ with $\sigma_1(\mathbf{A})\ge\cdots\ge\sigma_r(\mathbf{A})>0$. Obviously, the rank of $\mathbf{A}$ is equal to the number of nonzero singular values of $\mathbf{A}$. Additionally, we let $\|\mathbf{A}\|_F=\sqrt{\sum_{i,j}A_{ij}^2}=\sqrt{\sum_{i=1}^r\sigma_i(\mathbf{A})^2}$ be the Frobenius norm, $\|\mathbf{A}\|_*=\sum_{i=1}^r\sigma_i(\mathbf{A})$ be the nuclear norm, and $\|\mathbf{A}\|_2=\sigma_1(\mathbf{A})$ be the spectral norm.

For any $\tau>0$, we let $\mathcal{D}_\tau[\mathbf{A}]=\mathbf{U}_A S_\tau[\mathbf{\Sigma}_A]\mathbf{V}_A^T$ where $S_\tau[\mathbf{\Sigma}]=\mathrm{diag}([\sigma_1(\mathbf{A})-\tau]_+,\cdots,[\sigma_r(\mathbf{A})-\tau]_+)$ and $[z]_+=\max(z,0)$. In the literature (Candès & Recht, 2009; Cai et al., 2010), $\mathcal{D}_\tau[\mathbf{A}]$ is called the *singular value thresholding* (SVT) operator.

It is well known that the nuclear norm $\|\mathbf{A}\|_*$, as a function from $\mathbb{R}^{p\times q}$ to $\mathbb{R}$, is not differentiable. Alternatively, one considers the subdifferential of $\|\mathbf{A}\|_*$, which is the set of subgradients and denoted by $\partial\|\mathbf{A}\|_*$. It follows from the literature (Candès & Recht, 2009; Lewis, 2003; Watson, 1992) that for a $p\times q$ real matrix $\mathbf{A}$ of rank r,

$$\partial\|\mathbf{A}\|_*=\Big\{\mathbf{U}_A\mathbf{V}_A^T+\mathbf{Z}:\mathbf{Z}\in\mathbb{R}^{p\times q},\ \mathbf{U}_A^T\mathbf{Z}=\mathbf{0},\ \mathbf{Z}\mathbf{V}_A=\mathbf{0},\ \|\mathbf{Z}\|_2\le 1\Big\}. \quad (1)$$

3. Problem Formulation and Related Work

In this paper we study a regularized matrix classifier. We are given a set of training samples $\mathcal{T}=\{\mathbf{X}_i,y_i\}_{i=1}^n$, where $\mathbf{X}_i\in\mathbb{R}^{p\times q}$ is the ith input sample and $y_i\in\{-1,1\}$ is its corresponding class label. As we have seen, $\mathbf{X}_i$ is represented in matrix form. To fit a classifier, a commonly used approach is to stack $\mathbf{X}_i$ into a vector. Let $\mathbf{x}_i\triangleq\mathrm{vec}(\mathbf{X}_i^T)=([\mathbf{X}_i]_{11},\ldots,[\mathbf{X}_i]_{1q},[\mathbf{X}_i]_{21},\ldots,[\mathbf{X}_i]_{pq})^T\in\mathbb{R}^{pq}$. The soft margin SVM is defined as

$$\min_{\mathbf{w},b}\quad\frac{1}{2}\mathbf{w}^T\mathbf{w}+C\sum_{i=1}^n[1-y_i(\mathbf{w}^T\mathbf{x}_i+b)]_+, \quad (2)$$

where $[1-u]_+$ is called the hinge loss function, $\mathbf{w} \in \mathbb{R}^{pq}$ is a vector of regression coefficients, $b \in \mathbb{R}$ is an offset term, and C is a regularization parameter.

When reshaped into vector $\mathrm{vec}(\mathbf{X}_i^T)$, the correlation among columns or rows in the matrix is ignored. However, it would be more reasonable to exploit the correlation information in developing a classifier, because the correlation is helpful and useful in improving the classification performance.

Intuitively, we consider the following formulation:

$$\min_{\mathbf{W},b} \quad \frac{1}{2}\mathrm{tr}(\mathbf{W}^T\mathbf{W})+C\sum_{i=1}^{n}\{1-y_i[\mathrm{tr}(\mathbf{W}^T\mathbf{X}_i)+b]\}_+, \quad (3)$$

where $\mathbf{W} \in \mathbb{R}^{p\times q}$ is the matrix of regression coefficients. However, this formulation is essentially equivalent to Problem (2) when $\mathbf{w} = \mathrm{vec}(\mathbf{W}^T)$, because $\mathrm{tr}(\mathbf{W}^T\mathbf{X}_i) = \mathrm{vec}(\mathbf{W}^T)^T\mathrm{vec}(\mathbf{X}_i^T) = \mathbf{w}^T\mathbf{x}_i$ and $\mathrm{tr}(\mathbf{W}^T\mathbf{W}) = \mathrm{vec}(\mathbf{W}^T)^T\mathrm{vec}(\mathbf{W}^T) = \mathbf{w}^T\mathbf{w}$. This implies that the formulation in (3) cannot directly address our concern.

To capture the correlation, a natural approach is to consider the dependency of the regression matrix $\mathbf{W}$. In particular, one can impose a low-rank constraint on $\mathbf{W}$ to leverage the structure information within $\mathbf{X}_i$. For example, in the bilinear SVM (Pirsiavash et al., 2009), the authors assumed that $\mathbf{W} = \mathbf{W}_y\mathbf{W}_x^T$ where $\mathbf{W}_x \in \mathbb{R}^{q\times d}$, $\mathbf{W}_y \in \mathbb{R}^{p\times d}$ and $d < \min(p, q)$. Accordingly, they defined the following problem without the offset term

$$\begin{aligned}\underset{\mathbf{W}_x,\mathbf{W}_y}{\mathrm{argmin}} \quad & \frac{1}{2}\mathrm{tr}(\mathbf{W}_x\mathbf{W}_y^T\mathbf{W}_y\mathbf{W}_x^T) \\ & +C\sum_{i=1}^{n}[1-y_i\mathrm{tr}(\mathbf{W}_y^T\mathbf{X}_i\mathbf{W}_x)]_+. \quad (4)\end{aligned}$$

However, the resulting problem is nonconvex in both $\mathbf{W}_x$ and $\mathbf{W}_y$. Thus, the authors resorted to an alternately iterative update scheme for $\mathbf{W}_x$ and $\mathbf{W}_y$; that is, they updated either of $\mathbf{W}_x$ and $\mathbf{W}_y$ while keeping the other fixed.

Since the dependency of $\mathbf{W}$ can be revealed by its rank $\mathrm{rank}(\mathbf{W})$, it is also natural to directly impose the rank constraint to $\mathbf{W}$. However, the resulting matrix rank minimization is usually NP-hard (Vandenberghe & Boyd, 1996). Zhou & Li (2014) suggested a function of the singular values of $\mathbf{W}$ as an alternative penalization technique. Based on this idea, they proposed a regularized GLM (R-GLM):

$$\underset{\mathbf{W}}{\mathrm{argmin}} \quad J(\mathbf{W}) + P(\mathbf{W}), \quad (5)$$

where $J(\mathbf{W})$ is a loss function obtained from the negative log-likelihood and $P(\mathbf{W})$ is a penalty function defined on the singular values of $\mathbf{W}$. Typically, $P(\mathbf{W}) = \lambda\|\mathbf{W}\|_*$ for $\lambda > 0$ because the nuclear norm $\|\mathbf{W}\|_*$ is the best convex approximation of $\mathrm{rank}(\mathbf{W})$ over the unit ball of matrices. Assuming that the loss function J is smooth and its derivative has the Lipschitz continuity, Zhou & Li (2014) devised the Nesterov method (Nesterov, 1983) to solve (5).

4. The Support Matrix Machine

It is well known that the hinge loss enjoys the *large margin principle*. Moreover, it embodies *sparseness* and *robustness*, which are two desirable properties for a good classifier. This thus motivates us to employ the hinge loss function in (5) instead. In particular, we present the following formulation:

$$\begin{aligned}\underset{\mathbf{W},b}{\mathrm{argmin}} \quad & \frac{1}{2}\mathrm{tr}(\mathbf{W}^T\mathbf{W}) + \tau||\mathbf{W}||_* \\ & +C\sum_{i=1}^{n}\{1-y_i[\mathrm{tr}(\mathbf{W}^T\mathbf{X}_i)+b]\}_+, \quad (6)\end{aligned}$$

which defines a matrix classification model that we call the *support matrix machine* (SMM). Recall that the hinge loss is not smooth, so our model is not a trivial variant of the regularized GLM. On the other hand, SMM is in fact based on a penalty function, which is the combination of the squared Frobenius norm $\|\mathbf{W}\|_F^2$ and the nuclear norm $\|\mathbf{W}\|_*$. We call this penalty the *spectral elastic net* because $\mathrm{tr}(\mathbf{W}^T\mathbf{W}) = \|\mathbf{W}\|_F^2 = \sum_{i=1}^{\min(p,q)}\sigma_i^2(\mathbf{W})$ and $\|\mathbf{W}\|_* = \sum_{i=1}^{\min(q,p)}\sigma_i(\mathbf{W})$. As we see, the spectral elastic net is parallel to the elastic net of Zou & Hastie (2005).

Again recall that $\mathrm{tr}(\mathbf{W}^T\mathbf{W}) = \mathrm{vec}(\mathbf{W}^T)^T\mathrm{vec}(\mathbf{W}^T)$ and $\mathrm{tr}(\mathbf{W}^T\mathbf{X}_i) = \mathrm{vec}(\mathbf{W}^T)^T\mathrm{vec}(\mathbf{X}_i^T)$, so SMM degenerates to the conventional linear SVM when $\tau = 0$. However, the nuclear norm can not be equivalently defined as a vector norm. This implies that we cannot formulate Problem (6) in an equivalent of the vector form. Thus, our SMM is able to capture the correlation within the input data matrix.

4.1. Theoretical Justification

We now show that SMM possesses some elegant benefits from the conventional SVM (Cortes & Vapnik, 1995) as well as the conventional elastic net (Zou & Hastie, 2005). Without loss of generality, we suppose that each feature of the training data is normalized to have unit length. That is, it holds that $\|\mathbf{f}_{kl}\| = 1$ where $\mathbf{f}_{kl} \triangleq ([\mathbf{X}_1]_{kl}, \ldots, [\mathbf{X}_n]_{kl})^T$ for $k = 1, \ldots, p$ and $l = 1, \ldots, q$.

Theorem 1. *Suppose the minimizer of Problem (6) is* $(\tilde{\mathbf{W}}, \tilde{b})$. *Then*

$$\tilde{\mathbf{W}} = \mathcal{D}_\tau\Big(\sum_{i=1}^{n}\tilde{\beta}_i y_i \mathbf{X}_i\Big),$$

where $0 \leq \tilde{\beta}_i \leq C$.

Denote $\mathbf{\Omega} = \sum_{i=1}^{n} \tilde{\beta}_i y_i \mathbf{X}_i$. We see that $\mathbf{\Omega}$ is the combination of those $\mathbf{X}_i$ associated with nonzero $\tilde{\beta}_i$, while $\tilde{\mathbf{W}}$ is the SVT of $\mathbf{\Omega}$. In fact, we will see from Eqn. (12) in Theorem 5 that $\tilde{\mathbf{W}}$ is indeed the linear combination of a set of support matrices $\{\mathbf{X}_i\}$.

Lemma 1. *The difference between* $[\mathbf{\Omega}]_{k_1 l_1}$ *and* $[\mathbf{\Omega}]_{k_2 l_2}$ *meets the following inequality*

$$([\mathbf{\Omega}]_{k_1 l_1} - [\mathbf{\Omega}]_{k_2 l_2})^2 \leq 2nC^2(1 - \mathbf{f}_{k_1 l_1}^T \mathbf{f}_{k_2 l_2}).$$

Recall that $\|\mathbf{f}_{k_1 l_1}\| = 1$ and $\|\mathbf{f}_{k_2 l_2}\| = 1$, so $\mathbf{f}_{k_1 l_1}^T \mathbf{f}_{k_2 l_2} \in [-1, 1]$ is the correlation coefficient of input features at positions (k_1, l_1) and (k_2, l_2) over the n training samples. Lemma 1 says that $\mathbf{\Omega}$ has the element-wise grouping effect. Specifically, higher (lower) correlation between two elements of $\mathbf{\Omega}$ leads to smaller (larger) difference. Based on this lemma, we also have the following theorem.

Theorem 2. *Let* $[\tilde{\mathbf{W}}]_{:,l}$ *be the* l*th column of* $\tilde{\mathbf{W}}$. *Then*

$$\left\|[\tilde{\mathbf{W}}]_{:,l_1} - [\tilde{\mathbf{W}}]_{:,l_2}\right\|^2 \leq 2nC^2\Big(p - \sum_{k=1}^{p} \mathbf{f}_{kl_1}^T \mathbf{f}_{kl_2}\Big).$$

Especially, if $\mathbf{f}_{kl_1} = \mathbf{f}_{kl_2}$ *for any* $k = 1, \ldots, p$, *then* $[\tilde{\mathbf{W}}]_{:,l_1} = [\tilde{\mathbf{W}}]_{:,l_2}$.

Theorem 2 reflects the relationship of $\tilde{\mathbf{W}}$ with the training input matrices $\mathbf{X}_i$. Interestingly, the columns of the regression matrix $\tilde{\mathbf{W}}$ have grouping effect in our model if the corresponding features have strong correlation. The similar conclusion also applies to the rows of $\tilde{\mathbf{W}}$. Note that Theorem 2 can not be extended to the element-wise case, because even if $\mathbf{f}_{k_1 l_1} = \mathbf{f}_{k_2 l_2}$, we cannot obtain that $[\tilde{\mathbf{W}}]_{k_1 l_1} = [\tilde{\mathbf{W}}]_{k_2 l_2}$. We will present an empirical illustration for the grouping effect problem in Section 5.1.

4.2. Learning Algorithm

The Nesterov method for R-GLM (Zhou & Li, 2014) requires the derivative of the loss function in question to be Lipschitz-continuous. However, both the hinge loss and the nuclear norm are not smooth. Thus, it is hard to develop the Nesterov method for finding the SMM solution. Since the objective function of SMM is convex in both $\mathbf{W}$ and b, we here derive a learning algorithm based on ADMM with the restart rule (Goldstein et al., 2012). The problem in (6) can be equivalently written as follows:

$$\begin{aligned} \underset{\mathbf{W},b,\mathbf{S}}{\operatorname{argmin}} \quad & H(\mathbf{W}, b) + G(\mathbf{S}), \\ \text{s.t.} \quad & \mathbf{S} - \mathbf{W} = \mathbf{0}, \end{aligned} \tag{7}$$

where

$$H(\mathbf{W}, b) = \frac{1}{2}\mathrm{tr}(\mathbf{W}^T\mathbf{W}) + C\sum_{i=1}^{n}\{1 - y_i[\mathrm{tr}(\mathbf{W}^T\mathbf{X}_i) + b]\}_+$$

and $G(\mathbf{S}) = \tau||\mathbf{S}||_*$.

ADMM solves (7) by using the augmented Lagrangian function:

$$\begin{aligned} L_1(\mathbf{W}, b, \mathbf{S}, \mathbf{\Lambda}) =& H(\mathbf{W}, b) + G(\mathbf{S}) + \mathrm{tr}[\mathbf{\Lambda}^T(\mathbf{S} - \mathbf{W})] \\ &+ \frac{\rho}{2}||\mathbf{S} - \mathbf{W}||_F^2, \end{aligned}$$

where $\rho > 0$ is a hyperparameter.

The ADMM learning procedure for our SMM is summarized in Algorithm 1. The key steps of Algorithm 1 are the computations of $\mathbf{S}^{(k)}$ and $(\widehat{\mathbf{W}}^{(k)}, b^{(k)})$, the derivation of which is based on Theorems 3 and 4 below.

Theorem 3. *For positive numbers* τ *and* ρ, *let the matrix* $\rho\mathbf{W} - \mathbf{\Lambda}$ *have SVD of the form:*

$$\rho\mathbf{W} - \mathbf{\Lambda} = \mathbf{U}_0\mathbf{\Sigma}_0\mathbf{V}_0^T + \mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^T, \tag{8}$$

where $\mathbf{\Sigma}_0$ *is the diagonal matrix whose diagonal entries are greater than* τ, $\mathbf{U}_0$ *and* $\mathbf{V}_0$ *are matrices of the corresponding left and right singular vectors;* $\mathbf{\Sigma}_1$, $\mathbf{U}_1$ *and* $\mathbf{V}_1$ *correspond the rest parts of the SVD whose singular values are less than or equal to* τ. *Define*

$$\mathbf{S}^* \triangleq \frac{1}{\rho}\mathcal{D}_\tau(\rho\mathbf{W} - \mathbf{\Lambda}) = \frac{1}{\rho}\mathbf{U}_0(\mathbf{\Sigma}_0 - \tau\mathbf{I})\mathbf{V}_0^T. \tag{9}$$

Then we have $\mathbf{0} \in \partial\, G_1(\mathbf{S}^*)$, *where* $G_1(\mathbf{S}) = G(\mathbf{S}) + \mathrm{tr}(\mathbf{\Lambda}^T\mathbf{S}) + \frac{\rho}{2}||\mathbf{W} - \mathbf{S}||_F^2$.

Since G_1 is convex with respect to $\mathbf{S}$, we obtain the update equation of $\mathbf{S}$ from Theorem 3. That is,

$$\mathbf{S}^{(k+1)} = \underset{\mathbf{S}}{\operatorname{argmin}}\, G(\mathbf{W}^{(k)}, \mathbf{S}, \widehat{\mathbf{\Lambda}}^{(k)}) = \frac{1}{\rho}\mathcal{D}_\tau(\rho\mathbf{W}^{(k)} - \widehat{\mathbf{\Lambda}}^{(k)}).$$

Theorem 4. *One of the solution of the following problem*

$$\underset{(\mathbf{W},b)}{\operatorname{argmin}}\, H(\mathbf{W}, b) - \mathrm{tr}(\mathbf{\Lambda}^T\mathbf{W}) + \frac{\rho}{2}||\mathbf{W} - \mathbf{S}||_F^2$$

is

$$\begin{aligned} \mathbf{W}^* &= \frac{1}{\rho + 1}\Big(\mathbf{\Lambda} + \rho\mathbf{S} + \sum_{i=1}^{n}\alpha_i^* y_i \mathbf{X}_i\Big), \\ b^* &= \frac{1}{|\mathcal{S}^*|}\sum_{i \in \mathcal{S}^*}\Big\{y_i - \mathrm{tr}[(\mathbf{W}^*)^T\mathbf{X}_i]\Big\}, \end{aligned} \tag{10}$$

where $\mathcal{S}^* = \{i : 0 < \alpha_i^* < C\}$, *and* $\boldsymbol{\alpha}^* \in \mathbb{R}^n$ *is the solution of the following box constraint quadratic programming problem:*

$$\begin{aligned} \underset{\boldsymbol{\alpha}}{\operatorname{argmax}} \quad & -\frac{1}{2}\boldsymbol{\alpha}^T\mathbf{K}\boldsymbol{\alpha} + \mathbf{q}^T\boldsymbol{\alpha}, \\ \textit{s.t.} \quad & \mathbf{0} \leq \boldsymbol{\alpha} \leq C\mathbf{1}_n, \\ & \sum_{i=1}^{n}\alpha_i y_i = 0. \end{aligned} \tag{11}$$

Algorithm 1 ADMM for SMM

Initialize $\mathbf{S}^{(-1)} = \widehat{\mathbf{S}}^{(0)} \in \mathbb{R}^{p\times q}, \mathbf{\Lambda}^{(-1)} = \widehat{\mathbf{\Lambda}}^{(0)} \in \mathbb{R}^{p\times q}, \rho > 0, t^{(1)} = 1, \eta \in (0,1)$.
for $k = 1, 2, 3 \ldots$ **do**
$(\mathbf{W}^{(k)}, b^{(k)}) = \underset{(\mathbf{W},b)}{\operatorname{argmin}} H(\mathbf{W}, b) - \operatorname{tr}(\widehat{\mathbf{\Lambda}}^{(k)T}\mathbf{W}) + \frac{\rho}{2}||\mathbf{W}-\widehat{\mathbf{S}}^{(k)}||_F^2$
$\mathbf{S}^{(k)} = \underset{\mathbf{S}}{\operatorname{argmin}} G(\mathbf{S}) + \operatorname{tr}(\widehat{\mathbf{\Lambda}}^{(k)T}\mathbf{S}) + \frac{\rho}{2}||\mathbf{W}^{(k)} - \mathbf{S}||_F^2$
$\mathbf{\Lambda}^{(k)} = \widehat{\mathbf{\Lambda}}^{(k)} - \rho(\mathbf{W}^{(k)} - \mathbf{S}^{(k)})$
$c^{(k)} = \rho^{-1}||\mathbf{\Lambda}^{(k)} - \widehat{\mathbf{\Lambda}}^{(k)}||_F^2 + \rho||\mathbf{S}^{(k)} - \widehat{\mathbf{S}}^{(k)}||_F^2$
if $c^{(k)} < \eta c^{(k-1)}$ **then**
$t^{(k+1)} = \frac{1+\sqrt{1+4t^{(k)2}}}{2}$
$\widehat{\mathbf{S}}^{(k+1)} = \mathbf{S}^{(k)} + \frac{t^{(k)}-1}{t^{(k+1)}}(\mathbf{S}^{(k)} - \mathbf{S}^{(k-1)})$
$\widehat{\mathbf{\Lambda}}^{(k+1)} = \mathbf{\Lambda}^{(k)} + \frac{t^{(k)}-1}{t^{(k+1)}}(\mathbf{\Lambda}^{(k)} - \mathbf{\Lambda}^{(k-1)})$
else
$t^{(k+1)} = 1$
$\widehat{\mathbf{S}}^{(k+1)} = \mathbf{S}^{(k-1)}$
$\widehat{\mathbf{\Lambda}}^{(k+1)} = \mathbf{\Lambda}^{(k-1)}$
$c^{(k)} = \eta^{-1}c^{(k-1)}$
end if
end for

Here $\mathbf{K} = [K_{ij}] \in \mathbb{R}^{n\times n}$ *and* $\mathbf{q} \in \mathbb{R}^n$ *are independent of* $\boldsymbol{\alpha}$*; specifically,*

$$K_{ij} = \frac{y_i y_j \operatorname{tr}(\mathbf{X}_i^T\mathbf{X}_j)}{\rho+1},$$
$$q_i = 1 - \frac{y_i \operatorname{tr}[(\mathbf{\Lambda}+\rho\mathbf{S})^T\mathbf{X}_i]}{\rho+1}.$$

By Theorem 4, updating $\mathbf{W}^{(k)}$ and $b^{(k)}$ can be done by solving Problem (11). Several methods can be used, such as the sequential minimization optimization algorithm. (Platt et al., 1998; Keerthi & Gilbert, 2002)

Theorem 5. *Suppose the optimal solution of Problem (7) is* $(\tilde{\mathbf{W}}, \tilde{b}, \tilde{\mathbf{S}})$*. Then*

$$\tilde{\mathbf{W}} = \tilde{\mathbf{S}} = \tilde{\mathbf{\Lambda}} + \sum_{\tilde{\alpha}_i > 0} \tilde{\alpha}_i y_i \mathbf{X}_i. \tag{12}$$

Theorem 5 can be obtained directly by Algorithm 1 through Eqn. (10). If $\tilde{\alpha}_i > 0$, we call the corresponding $\mathbf{X}_i$ a support matrix. Theorem 5 shows that the solution $\tilde{\mathbf{W}}$ of our SMM can be written as the linear combination of support matrices plus an offset. This is the reason that we call our model the *support matrix machine*.

Since the hinge loss and nuclear norm are weakly convex, the convergence property of Algorithm 1 can be proved immediately based on the result in (Goldstein et al., 2012; He & Yuan, 2012). That is, we have

Theorem 6. *For any* $\rho > 0$ *and* $\eta \in (0,1)$*, the iteration sequence given by Algorithm 1 converges to the optimal solution of Problem (7).*

5. Experiments

In this section we conduct the experimental analysis of our proposed SMM [1]. We first analyze the group effect property of SMM. Then we study the classification performance on synthetic and real-world data sets. All experiments are implemented in Matlab R2011b on a workstation with Intel Xeon CPU X5675 3.06GHz (2×12 cores), 24GB RAM, and 64bit Windows Server 2008 system.

5.1. Group Effect Property on Synthetic Data

To intuitively demonstrate the grouping effect property described in Theorem 2, we design a simulated experiment to visualize it. We generate a synthetic data set of n samples as follows. First, we generate V orthogonal n-dimensional basis vectors $\boldsymbol{\nu}_1, \boldsymbol{\nu}_2, \cdots, \boldsymbol{\nu}_V$ with the unit Euclidean length, respectively. Second, we construct pq feature vectors of length n by the following process:

$$\tilde{\mathbf{f}}_{kl} = \boldsymbol{\nu}_{\lceil l/(0.2q)\rceil} + \boldsymbol{\epsilon}_{kl},$$
$$\boldsymbol{\epsilon}_{kl} \overset{i.i.d}{\sim} \mathcal{N}(\mathbf{0}, \delta^2\mathbf{I}_n),$$

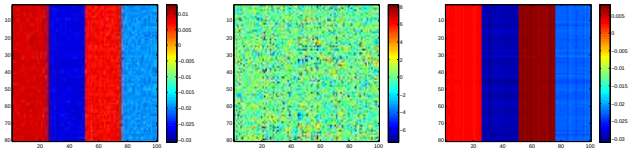
for $k = 1, \ldots, p$, and $l = 1, \ldots, q$. Here $\lceil l/(0.2q)\rceil$ denotes the smallest integer no smaller than $l/(0.2q)$. In other words, the elements in each sample matrix $\mathbf{X}_i$ $(i = 1, \ldots, n)$ can be roughly partitioned into four groups (vertical blocks). The features within the same group have high correlation, while the features between different groups have low correlation. Then we generate a $p \times q$ matrix $\mathbf{W}$ of rank $0.2q$, and the label for each sample is generated by $y_i = \operatorname{sign}[\operatorname{tr}(\mathbf{W}^T\mathbf{X}_i)]$. We set $n = 1000$, $V = 4$, $p = 80$, $q = 100$ and $\delta = 10^{-3}$ in this simulation.

The values of the regression matrix obtained respectively from the bilinear SVM (B-SVM) (Pirsiavash et al., 2009), regularized GLM (R-GLM) (Zhou & Li, 2014) and SMM are shown in Figure 1. It is easy to see that the regression matrix of SMM is clearly partitioned into four pure color blocks, while the blocks of B-SVM have higher noise. R-GLM fails to obtain the groups structure totally. The simulation results show that SMM has a better grouping effect property than other baselines, which also implies that SMM is able to capture the structure information in the feature matrices.

5.2. Classification Accuracy on Synthetic Data

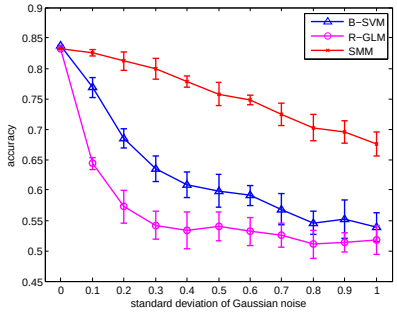
We now conduct the performance of SMM on synthetic data sets. We use the same data generation process as in the previous subsection, but we set $V = 10$ and $\delta = 10^{-2}$ to obtain more complicated data. We use 1000 samples for training, and other 500 samples for testing. All the hyper-parameters involved are selected via cross validation.

[1] The code is available in http://bcmi.sjtu.edu.cn/~luoluo/code/smm.zip

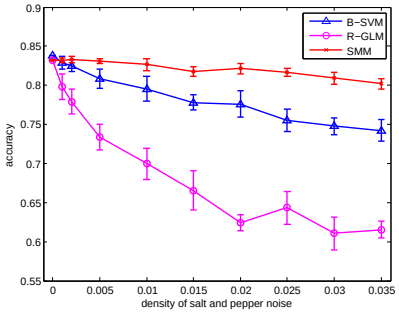


(a) B-SVM (b) R-GLM (c) SMM

Figure 1. (a), (b) and (c) display the values of normalized regression matrix of B-SVM, R-GLM and SMM respectively.



(a) Synthetic data with Gaussian noise



(b) Synthetic data with salt and pepper noise

Figure 2. Classification accuracy on synthetic data with different levels of noises. We use Gaussian noise with 0 mean and standard derivation from 0.01 to 1 in (a), and salt and pepper noise with density from 0.001 to 0.035 in (b).

We add different levels of Gaussian noise and salt and pepper noise on the test data, and repeat this procedure ten times to compute the mean and standard deviation of classification accuracy. The results are shown in Figure 2. It is clear that all methods achieve comparable performance on clean data, but SMM is more robust with respect to high level of noises.

5.3. Classification Accuracy on Real-World Data

We apply SMM to EEG and image classification problems, and compare its performance with B-SVM (Pirsiavash et al., 2009), R-GLM (Zhou & Li, 2014), and the standard linear SVM (L-SVM) (Cortes & Vapnik, 1995). We use four real-world matrix classification data sets: *the EEG alcoholism, the EEG emotion, the students face* and *INRIA person*.

Table 1. Summary of four data sets

Data sets	#positive	#negative	dimension
EEG alcoholism	77	45	256×64
EEG emotion	1286	1334	31×10
students face	200	200	200×200
INRIA person	607	1214	160×96

The *EEG alcoholism* data set[2] arises to examine EEG correlates of genetic predisposition to alcoholism. It contains two groups of subjects: alcoholic and control. For each subject, 64 channels of electrodes are placed and the voltage values are recorded at 256 time points.

The *EEG emotion* data set (Zhu et al., 2014; Zheng et al., 2014) focuses on EEG emotion analysis, which is obtained by showing some positive and negative emotional movie clips to persons and then recording the EEG signal via ESI NeuroScan System from 31 pairs. Each pair contain 10 data points (two channels for one pair, and each channel contains five frequency bands). There are 2620 movie clips chosen to evoke the target emotion, such as Titanic, Kung Fu Panda and so on.

The *student face* data set contains 400 photos of Stanford University medical students (Nazir et al., 2010), which consists of 200 males and 200 females. Each sample is a 200×200 gray level image.

The *INRIA person* data set[3] was collected to detect whether there exist people in the image. We normalize the samples into 160×96 gray images and remove the same person with different aspects. Combining with the negative samples, we obtain 1821 samples in total.

We summarize the main information of these data sets in Table 1. For the student face and INRIA person data sets, we directly use the pixels as input features without any advanced visual features.

For each of the compared methods, we randomly sample 70% of the data set for training and the rest for testing. All the hyperparameters involved are selected via cross validation. More specifically, we select C from $\{1 \times 10^{-3}, 2 \times 10^{-3}, 5 \times 10^{-3}, 1 \times 10^{-2}, 2 \times 10^{-2}, 5 \times 10^{-2} \ldots, 1 \times 10^{3}, 2 \times 10^{3}\}$. For each C, we tune τ manually to make the rank of classifier matrix varied from 1 to the size of the matrix. We repeat this procedure ten times to compute the mean and standard deviation of the classification accuracy. Table 2 shows the classification accuracy of the four methods. We can see that SMM achieves the best performance on all the four data sets.

[2] http://kdd.ics.uci.edu/databases/eeg/eeg.html

[3] http://pascal.inrialpes.fr/data/human/

Table 2. The classification accuracy on four data sets (in %)

Data sets	L-SVM	B-SVM	R-GLM	SMM
EEG alcoholism	71.11 ($\pm$ 8.30)	71.67 ($\pm$ 7.83)	71.39 ($\pm$ 6.55)	**73.33** ($\pm$ 5.89)
EEG emotion	88.76 ($\pm$ 1.16)	87.73 ($\pm$ 1.18)	82.26 ($\pm$ 1.65)	**90.01** ($\pm$ 0.98)
students face	91.67 ($\pm$ 1.57)	95.42 ($\pm$ 1.72)	94.25 ($\pm$ 2.76)	**96.83** ($\pm$ 1.66)
INRIA person	84.88 ($\pm$ 1.98)	85.09 ($\pm$ 1.46)	84.65 ($\pm$ 1.38)	**85.95** ($\pm$ 0.77)

Table 3. The training time on the four data sets (in second)

Data sets	B-SVM	R-GLM	SMM
EEG alcoholism	86.30 ($\pm$ 163.73)	407.59 ($\pm$ 100.93)	**1.36** ($\pm$ 0.09)
EEG emotion	292.89 ($\pm$ 248.47)	33.32 ($\pm$ 3.38)	**6.57** ($\pm$ 6.73)
students face	23.88 ($\pm$ 10.53)	121.14 ($\pm$ 87.40)	**7.20** ($\pm$ 0.22)
INRIA person	19.36 ($\pm$ 9.23)	580.06 ($\pm$ 229.14)	**6.61** ($\pm$ 2.44)

We are also interested in the computational efficiency of the three matrix classification models: B-SVM, R-GLM and SMM. We report the training time on the four data sets in Table 3. Recall that R-GLM is solved by the Nesterov method (Zhou & Li, 2014). We can find that R-GLM is the slowest method on EEG alcoholism, students face and INRIA person. This is because the main step of the Nesterov method is affected by the dimension of the input sample (Zhou & Li, 2014). However, the main step of B-SVM and SMM is a quadratic programming problem whose time complexity is mainly affected by the number of training samples. So B-SVM and SMM are more efficient than R-GLM on the data sets with high-dimension samples. Furthermore, we find that the running time of B-SVM are unstable on different data sets, usually with higher variance than that of SMM. The reason might be that B-SVM is a non-convex problem, the training procedure of which relies heavily on the initial value of the parameter.

6. Conclusion

In this paper we have proposed a novel matrix classification method called support matrix machine (SMM). SMM can leverage the structure of the data matrices and has the grouping effect property. We have derived an iteration algorithm based on ADMM for learning, and applied our method to EEG and image classification with better performance than the baselines such as B-SVM and R-GLM. Specifically, our method is more robust than B-SVM and R-GLM to model noisy data. Furthermore, our method is more efficient than B-SVM and R-GLM, and more numerically stable than B-SVM.

7. Acknowledgement

Luo Luo and Zhihua Zhang are supported by the Natural Science Foundation of Shanghai City of China (No. 15ZR1424200). Wu-Jun Li is supported by the NSFC (No. 61472182) and the Fundamental Research Funds for the Central Universities (No. 20620140510).

Appendix A: The Proof of Lemma 1

Proof. Let $\tilde{\boldsymbol{\beta}}' = [\tilde{\beta}_1 y_1, \ldots, \tilde{\beta}_n y_n]^T$, then we have

$$\begin{aligned}
&([\boldsymbol{\Omega}]_{k_1 l_1} - [\boldsymbol{\Omega}]_{k_2 l_2})^2 \\
= &\Big(\sum_{i=1}^n \tilde{\beta}_i y_i [\mathbf{X}_i]_{k_1 l_1} - \sum_{i=1}^n \tilde{\beta}_i y_i [\mathbf{X}_i]_{k_2 l_2}\Big)^2 \\
= &[\tilde{\boldsymbol{\beta}}'^T (\mathbf{f}_{k_1 l_1} - \mathbf{f}_{k_2 l_2})]^2 \\
\leq &||\tilde{\boldsymbol{\beta}}'||^2 ||\mathbf{f}_{k_1 l_1} - \mathbf{f}_{k_2 l_2}||^2 \\
\leq &2nC^2(1 - \mathbf{f}_{k_1 l_1}^T \mathbf{f}_{k_2 l_2}).
\end{aligned}$$

□

Appendix B: The Proof of Theorem 2

Proof. Suppose $\boldsymbol{\Omega}$ has condensed SVD $\boldsymbol{\Omega} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$, where $\mathbf{U} \in \mathbb{R}^{p \times r}$, $\boldsymbol{\Sigma} = \mathrm{diag}(\sigma_1, \ldots, \sigma_q)$ and $\mathbf{V} \in \mathbb{R}^{q \times r}$ satisfy $\mathbf{U}\mathbf{U}^T = \mathbf{I}_p$ and $\mathbf{V}\mathbf{V}^T = \mathbf{I}_q$. Denote $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_r]$, where for $k = 1, \ldots, r$, $\mathbf{u}_k$ is the kth column of $\mathbf{U}$. Since the columns of $\mathbf{U}$ are orthogonal, we have

$$\begin{aligned}
&\frac{\big\|[\tilde{\mathbf{W}}]_{:,l_1} - [\tilde{\mathbf{W}}]_{:,l_1}\big\|^2}{\big\|[\boldsymbol{\Omega}]_{:,l_1} - [\boldsymbol{\Omega}]_{:,l_1}\big\|^2} \\
= &\frac{\big\|\sum_{k=1}^q (\sigma_k - \tau)_+ ([\mathbf{V}]_{l_1 k} - [\mathbf{V}]_{l_2 k})\mathbf{u}_k\big\|^2}{\big\|\sum_{k=1}^q \sigma_k ([\mathbf{V}]_{l_1 k} - [\mathbf{V}]_{l_2 k})\mathbf{u}_k\big\|^2} \\
= &\frac{\sum_{k=1}^q [(\sigma_k - \tau)_+]^2 ([\mathbf{V}]_{l_1 k} - [\mathbf{V}]_{l_2 k})^2 ||\mathbf{u}_k||^2}{\sum_{k=1}^q \sigma_k^2 ([\mathbf{V}]_{l_1 k} - [\mathbf{V}]_{l_2 k})^2 ||\mathbf{u}_k||^2} \\
= &\frac{\sum_{k=1}^q [(\sigma_k - \tau)_+]^2 ([\mathbf{V}]_{l_1 k} - [\mathbf{V}]_{l_2 k})^2}{\sum_{k=1}^q \sigma_k^2 ([\mathbf{V}]_{l_1 k} - [\mathbf{V}]_{l_2 k})^2} \leq 1.
\end{aligned}$$

Then we can obtain the following bound based on Lemma 1

$$\begin{aligned} & \left\|[\tilde{\mathbf{W}}]_{:,l_1} - [\tilde{\mathbf{W}}]_{:,l_2}\right\|^2 \\ \leq & \left\|[\mathbf{\Omega}]_{:,l_1} - [\mathbf{\Omega}]_{:,l_2}\right\|^2 \\ = & \sum_{k=1}^{p} \left\|[\mathbf{\Omega}]_{kl_1} - [\mathbf{\Omega}]_{kl_2}\right\|^2 \\ \leq & 2nC^2\Big(p - \sum_{k=1}^{p} \mathbf{f}_{kl_1}^T \mathbf{f}_{kl_2}\Big). \end{aligned}$$

□

Appendix C: The Proof of Theorem 3

Proof. Let $\mathbf{Z}_0$ be $\frac{1}{\tau}\mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1$. Recall that $\mathbf{U}_0$, $\mathbf{U}_1$, $\mathbf{V}_0$ and $\mathbf{V}_1$ are column orthogonal. So we have $\mathbf{U}_0^T\mathbf{Z}_0 = \mathbf{0}$ and $\mathbf{Z}_0\mathbf{V}_0 = \mathbf{0}$. By the SVD form of $\widehat{\mathbf{S}}$, formulation (9) and using Eqn. (1) we have:

$$\partial G_1(\mathbf{S})|_{\mathbf{S}=\mathbf{S}^*} = \mathbf{\Lambda} - \rho\mathbf{W} + \mathcal{D}_\tau(\rho\mathbf{W} - \mathbf{\Lambda}) + \tau\,\partial||\mathbf{S}||_*|_{\mathbf{S}=\mathbf{S}^*}.$$

Thus, we have $\mathbf{0} \in \partial G_1(\mathbf{S}^*)$. □

Appendix D: The Proof of Theorem 4

Proof. We denote $H_1(\mathbf{W}, b) = H(\mathbf{W}, b) - \mathrm{tr}(\mathbf{\Lambda}^T\mathbf{W}) + \frac{\rho}{2}||\mathbf{W}-\mathbf{S}||_F^2$. Finding the minimizer of $H_1(\mathbf{W}, b)$ is equivalent to solving the following problem:

$$\begin{aligned} \min_{\mathbf{W},b,\boldsymbol{\xi}} \quad & \frac{1}{2}\mathrm{tr}(\mathbf{W}^T\mathbf{W}) + C\sum_{i=1}^{n}\xi_i \\ & -\mathrm{tr}(\mathbf{\Lambda}^T\mathbf{W}) + \frac{\rho}{2}||\mathbf{W}-\mathbf{S}||_F^2 \\ \text{s.t.} \quad & y_i[\mathrm{tr}(\mathbf{W}^T\mathbf{X}_i) + b] \geq 1 - \xi_i \\ & \xi_i \geq 0. \end{aligned} \tag{13}$$

To solve problem (13), we construct the following Lagrangian function

$$\begin{aligned} & L(\mathbf{W}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\gamma}) \\ =& \frac{1}{2}\mathrm{tr}(\mathbf{W}^T\mathbf{W}) + C\sum_{i=1}^{n}\xi_i - \mathrm{tr}(\mathbf{\Lambda}^T\mathbf{W}) + \frac{\rho}{2}||\mathbf{W}-\mathbf{S}||_F^2 \\ & - \sum_{i=1}^{n}\alpha_i\{y_i[\mathrm{tr}(\mathbf{W}^T\mathbf{X}_i) + b] - 1 + \xi_i\} - \sum_{i=1}^{n}\gamma_i\xi_i. \end{aligned} \tag{14}$$

Setting the derivative of L with respect to $\boldsymbol{\xi}$ to be $\mathbf{0}$, we have

$$\gamma_i = C - \alpha_i \geq 0, \quad i = 1, \ldots, n. \tag{15}$$

Setting the derivative of L with respect to b be 0, we have

$$\sum_{i=1}^{n}\alpha_i y_i = 0. \tag{16}$$

Substituting (15) and (16) into (14) to eliminate γ_i and ξ_i, we obtain

$$\begin{aligned} & L(\mathbf{W}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\gamma}) \\ =& \frac{1}{2}\mathrm{tr}(\mathbf{W}^T\mathbf{W}) - \mathrm{tr}(\mathbf{\Lambda}^T\mathbf{W}) + \frac{\rho}{2}||\mathbf{W}-\mathbf{S}||_F^2 \\ & - \sum_{i=1}^{n}\alpha_i\{y_i[\mathrm{tr}(\mathbf{W}^T\mathbf{X}_i) + b] - 1\}. \end{aligned} \tag{17}$$

Setting the derivative of L with respect to $\mathbf{W}$ to be $\mathbf{0}$, we have the optimal value

$$\mathbf{W}^* = \frac{1}{\rho+1}\Big(\mathbf{\Lambda} + \rho\mathbf{S} + \sum_{i=1}^{n}\alpha_i y_i \mathbf{X}_i\Big). \tag{18}$$

Substituting (18) into (17), we obtain

$$\begin{aligned} & L(\mathbf{W}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\gamma}) \\ =& \sum_{i=1}^{n}\Big(1 - \frac{y_i\mathrm{tr}[(\mathbf{\Lambda} + \rho\mathbf{S})^T\mathbf{X}_i]}{\rho+1}\Big)\alpha_i \\ & - \frac{1}{2(\rho+1)}\sum_{i,j=1}^{n}\alpha_i\alpha_i y_i y_j \mathrm{tr}(\mathbf{X}_i^T\mathbf{X}_j) + D, \end{aligned}$$

where $D = \frac{\rho}{2}\mathrm{tr}(\mathbf{S}^T\mathbf{S}) - \frac{1}{2(\rho+1)}||\mathbf{\Lambda} + \rho\mathbf{S}||_F^2$. Thus, finding the minimizer of $H(\mathbf{W}, b)$ is equivalent to solving Problem (11) by KKT conditions. Let the optimal solution of (11) be $\boldsymbol{\alpha}^*$, we can obtain (10) from (18) directly. The KKT conditions also provide

$$\begin{aligned} \alpha_i^*\{y_i\{\mathrm{tr}[(\mathbf{W}^*)^T\mathbf{X}_i] + b^*\} - 1 + \xi_i^*\} = 0 \\ \gamma_i^*\xi_i^* = 0, \end{aligned}$$

which means for any $0 < \alpha_i^* < C$, the corresponding $\gamma_i^* > 0, \xi_i^* = 0$ and $y_i\{\mathrm{tr}[(\mathbf{W}^*)^T\mathbf{X}_i] + b^*\} - 1 = 0$. Then the optimal b^* can be calculated by

$$b^* = y_i - \mathrm{tr}[(\mathbf{W}^*)^T\mathbf{X}_i].$$

In practice, we choose the optimal b^* by averaging these solutions

$$b^* = \frac{1}{|\mathcal{S}^*|}\sum_{i\in\mathcal{S}^*}\{y_i - \mathrm{tr}[(\mathbf{W}^*)^T\mathbf{X}_i]\}.$$

□

References

Bach, Francis R. Consistency of trace norm minimization. *The Journal of Machine Learning Research*, 9: 1019–1048, 2008.

Cai, Deng, He, Xiaofei, Wen, Ji-Rong, Han, Jiawei, and Ma, Wei-Ying. Support tensor machines for text categorization. Technical report, University of Illinois at Urbana-Champaign, 2006.

Cai, Jian-Feng, Candès, Emmanuel J, and Shen, Zuowei. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

Candès, Emmanuel J and Recht, Benjamin. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

Cortes, Corinna and Vapnik, Vladimir. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

Goldstein, Tom, ODonoghue, Brendan, and Setzer, Simon. Fast alternating direction optimization methods. *CAM report*, pp. 12–35, 2012.

Harchaoui, Zaid, Douze, Matthijs, Paulin, Mattis, Dudik, Miroslav, and Malick, Jérôme. Large-scale image classification with trace-norm regularization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3386–3393, 2012.

Hastie, Trevor, Robert, Tibshirani, and Jerome, Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer-Verlag, 2001.

He, Bingsheng and Yuan, Xiaoming. On non-ergodic convergence rate of douglas–rachford alternating direction method of multipliers. *Numerische Mathematik*, pp. 1–11, 2012.

Huang, Jin, Nie, Feiping, and Huang, Heng. Robust discrete matrix completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 424–430, 2013.

Hunyadi, Borbála, Signoretto, Marco, Van Paesschen, Wim, Suykens, Johan AK, Van Huffel, Sabine, and De Vos, Maarten. Incorporating structural information from the multichannel eeg improves patient-specific seizure detection. *Clinical Neurophysiology*, 123(12): 2352–2361, 2012.

Kang, Zhuoliang, Grauman, Kristen, and Sha, Fei. Learning with whom to share in multi-task feature learning. In *Proceedings of the International Conference on Machine Learning*, pp. 521–528, 2011.

Keerthi, S. Sathiya and Gilbert, Elmer G. Convergence of a generalized smo algorithm for svm classifier design. *Machine Learning*, 46(1-3):351–360, 2002.

Lewis, Adrian S. The mathematics of eigenvalue optimization. *Mathematical Programming*, 97(1-2):155–176, 2003.

Liu, Ji, Musialski, Przemyslaw, Wonka, Peter, and Ye, Jieping. Tensor completion for estimating missing values in visual data. In *IEEE Tansactions on Pattern Analysis and Machine Intelligence*, volume 35, pp. 208–220, 2013.

Nazir, M, Ishtiaq, Muhammad, Batool, Anab, Jaffar, M Arfan, and Mirza, Anwar M. Feature selection for efficient gender classification. In *Proceedings of the WSEAS international conference, Wisconsin*, pp. 70–75, 2010.

Nesterov, Yurii. A method of solving a convex programming problem with convergence rate o(1/k2). *Soviet Mathematics Doklady*, 27(2):372–376, 1983.

Pirsiavash, Hamed, Ramanan, Deva, and Fowlkes, Charless C. Bilinear classifiers for visual recognition. In *Proceedings of the Advances in Neural Information Processing Systems*, pp. 1482–1490, 2009.

Platt, John et al. Sequential minimal optimization: A fast algorithm for training support vector machines. *Technical report msr-tr-98-14, Microsoft Research*, 1998.

Pong, Ting Kei, Tseng, Paul, Ji, Shuiwang, and Ye, Jieping. Trace norm regularization: reformulations, algorithms, and multi-task learning. *SIAM Journal on Optimization*, 20(6):3465–3489, 2010.

Salakhutdinov, Ruslan and Srebro, Nathan. Collaborative filtering in a non-uniform world: Learning with the weighted trace norm. *arXiv preprint arXiv:1002.2780*, 2010.

Signoretto, Marco, Dinh, Quoc Tran, De Lathauwer, Lieven, and Suykens, Johan AK. Learning with tensors: a framework based on convex optimization and spectral regularization. *Machine Learning*, 94(3):303–351, 2014.

Srebro, Nathan and Shraibman, Adi. Rank, trace-norm and max-norm. In *Proceedings of the Conference on Learning Theory*, pp. 545–560. 2005.

Vandenberghe, Lieven and Boyd, Stephen. Semidefinite programming. *SIAM review*, 38(1):49–95, 1996.

Watson, G Alistair. Characterization of the subdifferential of some matrix norms. *Linear Algebra and its Applications*, 170:33–45, 1992.

Wolf, Lior, Jhuang, Hueihan, and Hazan, Tamir. Modeling appearances with low-rank SVM. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–6, 2007.

Zheng, Wei-Long, Zhu, Jia-Yi, Peng, Yong, and Lu, Bao-Liang. Eeg-based emotion classification using deep belief networks. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pp. 1–6, 2014.

Zhou, Hua and Li, Lexin. Regularized matrix regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(2):463–483, 2014.

Zhu, Jia-Yi, Zheng, Wei-Long, Peng, Yong, Duan, Ruo-Nan, and Lu, Bao-Liang. Eeg-based emotion recognition using discriminative graph regularized extreme learning machine. In *Proceedings of the International Joint Conference on Neural Networks*, pp. 525–532, 2014.

Zou, Hui and Hastie, Trevor. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2): 301–320, 2005.