Neurocomputing 275 (2018) 869–880

Contents lists available at ScienceDirect

Neurocomputing

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

Multiclass support matrix machine for single trial EEG classification

Qingqing Zheng[a,*], Fengyuan Zhu[a], Jing Qin[b], Pheng-Ann Heng[a]

[a] Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong
[b] Centre of Smart Health, School of Nursing, The Hong Kong Polytechnic University, Hong Kong

ARTICLE INFO

Article history:
Received 6 February 2017
Revised 23 August 2017
Accepted 11 September 2017
Available online 21 September 2017

Communicated by Dr. Nianyin Zeng

Keywords:
Brain computer interface (BCI)
Electroencephalograph (EEG)
Support vector machine (SVM)
Support matrix machine
Multiclass classification
Alternating direction method of multipliers (ADMM)

ABSTRACT

We propose a novel multiclass classifier for single trial electroencephalogram (EEG) data in matrix form, namely multiclass support matrix machine (MSMM), aiming at improving the classification accuracy of multiclass EEG signals, and hence enhancing the performance of EEG-based brain computer interfaces (BCIs) involving multiple mental activities. In order to construct the MSMM, we propose a novel objective function, which is composed of a multiclass hinge loss term and a combined regularization term. We first formulate the multiclass hinge loss by extending the margin rescaling loss to support matrix-form data. We then devise the regularization term by combining the squared Frobenius norm of tensor-form model parameter and the nuclear norm of matrix-form hyperplanes extracted from the model parameter. While the Frobenius norm prevents over-fitting when training the model, the nuclear norm captures the structural information within the matrix data. We further propose an efficient solver for MSMM based on the alternating direction method of multipliers (ADMM) framework. We conduct extensive experiments on two benchmark EEG datasets. Experimental results show that MSMM achieves much better performance than state-of-the-art classifiers and yields a mean kappa value of 0.880 and 0.648 for dataset IIIa of BCI III and dataset IIa of BCI IV, respectively. To our best knowledge, MSMM is the first classifier that supports multiclass classification for matrix-form EEG data. The proposed MSMM enables easier and more efficient implementation of robust multi-task BCIs, and therefore has potential to promote the wider use of BCI technology.

1. Introduction

Brain computer interfaces (BCIs) have emerged as a new and promising communication mode between human and computers with the development of neuroscience and engineering over the last 20 years [1]. They capture brain signals associated with mental activities and transform them into commands to communicate with or control external machines. BCIs not only benefit people with severe motor impairments caused by various neuromuscular disorders through restoring their communication and movement ability [2,3], but also find a lot of applications for healthy individuals, such as virtual reality systems [4,5] and games [6,7]. The brain signals employed in BCIs can be measured by several techniques, mainly including electroencephalogram (EEG), magnetoencephalogram (MEG), functional magnetic response imaging (fMRI) and functional near-infrared spectroscopy (fNIRS). Among them, EEG, which measures voltage fluctuations from scalps during brain activities, is most widely used in practical applications due to its simplicity and efficiency [8]. As motor imagery (MI) based EEG–BCIs offer promise for motor function recovery and are widely used in rehabilitative applications, in the paper, we focus on motor imagery (MI) based EEG–BCIs.

Fig. 1(a) illustrates a typical pipeline of a MI based EEG–BCI system. When a user is imagining an action (e.g., left hand movement), the EEG acquisition device first obtains the EEG signals. Then the preprocessing is performed to remove the artifacts from the signals. Based on the filtered signals, the system extracts discriminative patterns/features and then based on them trains a classifier and identifies the motion that the user has imagined. While such a MI-based system has been widely used in many applications, most of them mainly focus on two-class motor imagination patterns [9–13]. In practice, MI-based BCI systems supporting multiple tasks are highly demanded [14]. However, implementing a multi-task MI-based BCI system is still a challenging problem, as the involvement of more brain activities may make it quite difficult to precisely identify multiple tasks from single trial EEG signals [15,16].

One of the main limitations for current BCI systems to support multiple motor tasks is that most of them employed classifiers that only accept input EEG signals in the form of vector, such as Bayes classifiers [17,18], support vector machine (SVM) [19,20] and

* Corresponding author.
E-mail address: qqzheng@cse.cuhk.edu.hk (Q. Zheng).

https://doi.org/10.1016/j.neucom.2017.09.030
0925-2312/

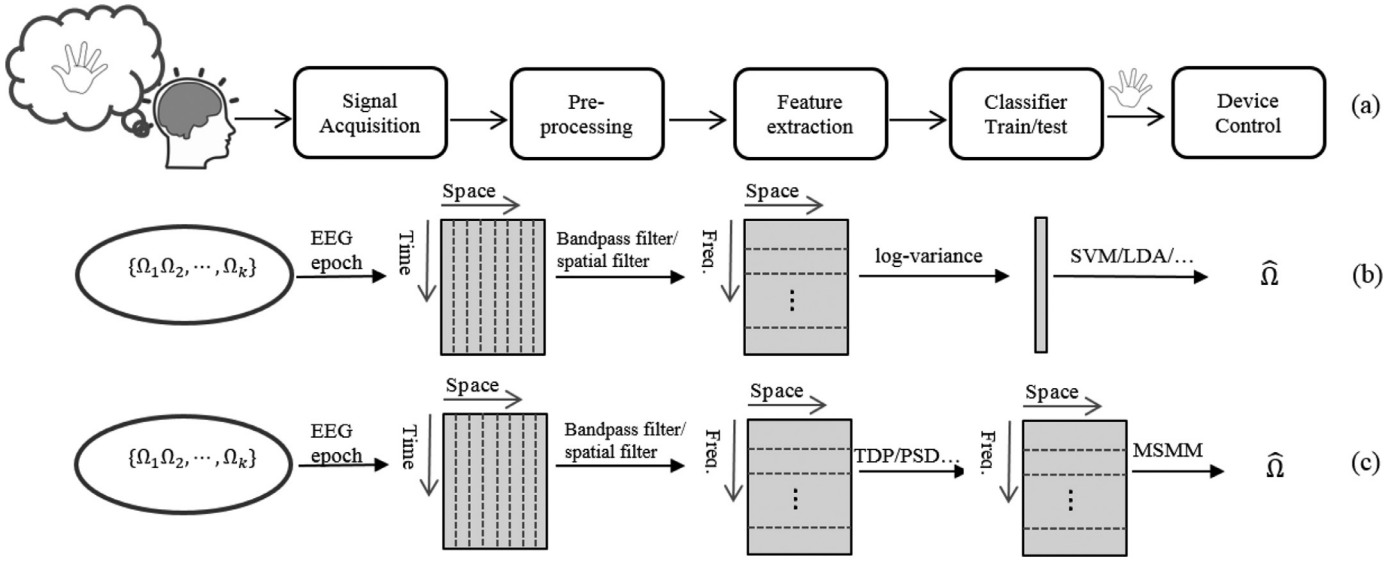


Fig. 1. The schematic illustrations of (a) a general EEG based BCI system, (b) classification workflow based on vector-form features, and (c) classification workflow with the proposed MSMM model supporting multiclass classification of matrix-form EEG data.

linear discriminant analysis (LDA) [10,21]. In practical applications, each single trial EEG signal records voltage fluctuations at several electrodes during a time period. In this regard, it is more natural to be represented as a two-dimensional matrix with strong correlation between rows and columns with respect to certain channels and frequency bands in motor imagery tasks, rather than as a vector. For example, EEG signals from C3, C4, Fz and Cz channels between 7 and 30 Hz are correlated to reflect some motor imagery tasks [22]. To this end, matrix-form features can better preserve the structural information of EEG signals between the temporal and spatial domain whereas vectorization would collapse the topology and loss the structural information.

A straightforward solution for this problem is to concatenate a matrix into a vector to fit into these classifiers, as shown in Fig. 1(b). However, such a solution will suffer from the curse of dimensionality [23], especially for EEG signals. For example, the popular dataset IIa of BCI Competition IV [24] contains 288 training samples and each sample unit is a 750×22 temporal-spatial matrix. The dimension of feature vectors is extremely large as $d = 750 \times 22 = 16,500$, whereas the samples size is only 288, which may lead to severe over-fitting problem. Some efforts have been devoted to suppressing the matrix-form features into vectors [18,25] after preprocessing with common spatial patterns (CSP) like filters [26–30]. These methods, however, still ignore the topological structure latently embedded within single trial EEG data. In this regard, it is of great interest to study classification algorithms that can take full advantage of structural information of EEG signals in matrix form, such as the correlation information over frequencies and channels, to improve the performance of multiclass classification of EEG signals and hence facilitate the implementation of multi-task MI based BCI systems.

Several classifiers have been proposed for classification of data in matrix form. Wolf et al. [31] proposed a rank-k SVM to capture the global structure of data matrices by regularizing the regression matrix as the sum of k rank-one orthogonal matrices; Dyrholm et al. [32] and Pirsiavash et al. [33] decomposed the regression parameter into the product of two rank-k matrices; Kobayashi et al. [34] proposed a similar bilinear SVM framework by regularizing the nuclear norm of the model parameter; and recently Luo et al. [35] proposed a spectral elastic net regularization to constrain the combination of Frobenius norm and nuclear norm of the regression parameter simultaneously. However, all these matrix classifiers are originally built for binary classification problems. Though it is natural to break a multiclass classification task into a series of binary ones by one-versus-rest (OvR) or one-versus-one (OvO) strategies [36,37], this scheme would suffer from several drawbacks. First, it would introduce bias when the scales of confidence values are different between the binary classifiers in the prediction stage [38]. Second, it may result in unbalanced distribution of input samples because the number of negative samples is much larger than that of positive ones with OvR scheme [39]. Finally, it is quite time-consuming to train multiple binary classifiers, especially when the number of motor imagery tasks involved in the BCI system is large.

In this paper, we propose a novel classifier to address the multiclass classification of single trial EEG signals in matrix form, aiming at improving the performance of BCIs supporting multiple motion tasks, as shown in Fig. 1(c). We call our classifier multiclass support matrix machine (MSMM). The MSMM is constructed based on regularized risk minimization framework. We first propose a novel objective function, which consists of two components: a multiclass hinge loss term and a combined regularization term taking structural information of matrix-form data into consideration. We formulate the multiclass hinge loss by extending the margin rescaling loss [40] to support matrix-form data. The regularization term is a combination of the squared Frobenius norm of tensor-form model parameter and nuclear norm of matrix-form hyperplanes extracted from the model parameter. While the Frobenius norm is applied to prevent the over-fitting problem when training the model, the nuclear norm is leveraged to capture the global structure within the matrix data. We further propose a solver for this convex objective function based on the alternating direction method of multipliers (ADMM) framework [41,42]. The solver converges quickly and reaches to the global optimal solution. We extensively evaluate the proposed MSMM on two benchmark single trial EEG datasets: dataset IIIa of BCI competition III [43] and dataset IIa of BCI competition IV [24]. Experimental results show that the proposed MSMM achieves much better classification accuracy on multiclass single trial EEG data than state-of-the-art classifiers, by taking full advantage of the structural information of EEG data.

The contributions of this paper can be summarized as follows.

- We propose a novel classifier for multiclass classification of EEG data in matrix form, namely MSMM. Compared with

existing EEG signal classifiers, which are either based on vector-form data or only capable of coping with two-class classification of matrix data, the proposed MSMM can leverage the inherent structural information of EEG data for more accurate multiclass classification, and hence improve the performance of BCI systems with multiple tasks. To our best knowledge, the proposed MSMM is the first classifier that can support multiclass classification for EEG data in matrix form.
- We propose a novel objective function based on regularized risk minimization framework by regularizing the combination of the squared Frobenius norm of the tensor-form model parameter and nuclear norm of matrix-form hyperplanes extracted from the model parameter, and develop an efficient solver based on ADMM framework to solve it.
- We extensively evaluate the proposed MSMM on real multiclass EEG datasets, and achieve the state-of-the-art performance. Although the proposed method is applied to MI-based BCI systems, it is general enough to be used in other BCI systems involving multiclass matrix-form signals.

The rest of this paper is organized as follows: In Section 2, we review the relevant studies on classification algorithms used in MI based EEG classification. We briefly introduce some preliminaries in Section 3. Then we illustrate the MSMM model and its efficient solver for multiclass matrix classification in Section 4. We conduct experiments to evaluate the performance of our model in Section 5. We conclude this paper in Section 6.

2. Related work

Single trial EEG classification is very challenging due its poor characteristic, such as the low signal-to-noise ratio, the non-stationarity of signals and the presence of noises. Building a classifier of high performance based on limited training EEG signals is essential. This section provides a comprehensive study on the classification algorithms used in motor imagery in EEG based BCIs.

The existing classification algorithms can roughly be divided into two categories with respect to the type of the training data. The first category is the vector-form classifiers, which require the training data to be in vector form. The state-of-the-art vector-form classifiers applied successfully into the EEG classification include various linear or nonlinear classifiers, such as linear discriminant analysis (LDA) [13], support vector machine (SVM) [44], k nearest neighbor (KNN) [43], neural network (NN) [45], etc. Among these algorithms, SVM and LDA are most popular amongst researchers for MI based EEG classification due to their simplicity and robustness [46]. Many studies have focused on how to transform the acquired high-dimensional EEG signals into discriminant features that can be fed into these simple classifiers. Common spatial pattern (CSP) is the most widely used algorithm for EEG feature extraction, which seeks optimal projections such that the filtered variance between two classes are maximized or minimized. Many variants of CSP have been studied in the literature [18]. Thomas et al. [47] proposed filter-bank CSP (FBCSP) to combine features in different frequency bands by using a bank of multiple bandpass filters. Sun and Zhang [48] defined a variability coefficient in the CSP formulation to denote the weighted average of historical covariance. Both algorithms used SVM for classification. Wang and Li [49] proposed an ℓ_1 norm based CSP to alleviate the negative impact of outliers and noises without large deviations. Arvaneh et al. [50] proposed KLCSP by utilizing Kulback–Leibler (KL) divergence to measure the changes in the distribution of data in each class. The extracted features from ℓ_1-CSP and KLCSP are recognized by the LDA algorithm. The work in [51] used a clustering technique to extract representative features and applied SVM for classification. Though these methods have been devoted to efficiently improving the single trial EEG classification, they all suppress the matrix-form features into vectors, resulting in loss of topological structural information within EEG signals and hence degradation of the classification performance.

The second category is matrix-form classifiers, which are motivated by the development of matrix analysis and accept the matrix-form training data. Tomioka and Aihara [52] proposed a spectral ℓ_1 norm regularized logistic regression. The optimization problem is formulated as semi-definite programming (SDP) and solved by interior point method. Dyrholm et al. [32] decomposed the regression parameter into the product of two rank-k matrices, where the rank-k is required to be predetermined. Luo et al. [35] defined a spectral elastic net penalty, which is the linear combination of Frobenius norm and nuclear norm of the regression matrix. Tomioka et al. [53] and Christoforou et al. [54] directly exploited the covariance matrix of the signals as the representation of EEG features and built the matrix logistic regression classifier. Following [53,54], Zeng and Song [55] integrated the within session non-stationary regularization into a convex empirical risk minimization problem and solved it with accelerated proximal gradient-based algorithm. These methods all take advantage of the low-rank assumptions to exploit the correlation between rows and columns within each single trial EEG data. However, all these methods are built for binary classification problems and their extensions for multiclass problem may be difficult. This limits their application to multi-task EEG based BCIs.

3. Preliminaries

In order to facilitate the description of our method, in this section, we introduce some notations that run throughout our formulation and solver. We also give a brief introduction of the multiclass SVM [56], based on which we devise the proposed MSMM supporting the classification of EEG signals in matrix form to take full advantage of correlations among different channels and frequency bands.

3.1. Notations

The singular value decomposition of a matrix $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2}$ is denoted as $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathbf{T}}$, where $\mathbf{U}$ is a unitary matrix, $\mathbf{\Sigma} = diag(\sigma_1, \sigma_2, \ldots, \sigma_r, 0, \ldots, 0)$ is a rectangular diagonal matrix with r (the rank of $\mathbf{X}$) singular values on the diagonal, and $\mathbf{V}^T$ is the conjugate transpose of the unitary matrix $\mathbf{V}$. The Frobenius norm of $\mathbf{X}$ is denoted as $||\mathbf{X}||_F = \sqrt{\sum_{i_1=1}^{I_1}\sum_{i_2=1}^{I_2} x_{i_1 i_2}^2}$; the nuclear (trace) norm is $||\mathbf{X}||_* = \sum_{i=1}^{r}\sigma_i$. For any $\tau \geq 0$, the singular value thresholding operator is defined as $D_\tau(\mathbf{X}) = \mathbf{U}\mathbf{\Sigma}_\tau\mathbf{V}^{\mathbf{T}}$, where $\mathbf{\Sigma}_\tau = diag([\sigma_1 - \tau]_+, \ldots, [\sigma_r - \tau]_+, 0, \ldots, 0)$ and $[\cdot]_+ = max(\cdot, 0)$. The inner product between $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{I_1 \times I_2}$ is the sum of element-wise product, i.e., $< \mathbf{X}, \mathbf{Y} >= \sum_{i_1=1}^{I_1}\sum_{i_2=1}^{I_2} x_{i_1 i_2} \times y_{i_1 i_2}$.

As our method involves processing multiple matrices, we introduce the concept of *tensor* here. Tensor is a generalized array with multiple dimensions. For example, a zero-order tensor is a scalar; a first-order tensor is a vector; a second-order tensor is a matrix and a third-order or higher one is called a high-order tensor. The Frobenius norm of a high-order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_n}$ can be defined as $||\mathcal{X}||_F = \sqrt{\sum_{i_1=1}^{I_1}\sum_{i_2=1}^{I_2}\cdots\sum_{i_n=1}^{I_n} x_{i_1 i_2 \ldots i_n}^2}$. The inner product between two same-dimensional tensors $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_n}$ is denoted as $< \mathcal{X}, \mathcal{Y} >= \sum_{i_1=1}^{I_1}\sum_{i_2=1}^{I_2}\cdots\sum_{i_n=1}^{I_n} x_{i_1 i_2 \ldots i_n} \times y_{i_1 i_2 \ldots i_n}$.

3.2. Multiclass support vector machine

While a lot of methods have been proposed to extend SVM to support multiclass classification of vector-form data by OvR and

OvO strategies, Crammer and Singer proposed a true multiclass SVM [57]. As our method is inspired by this work, we briefly review it here and readers can refer to [56,57] for more details. For a training set $\{\boldsymbol{x}_i, y_i\}_{i=1}^n \in \{\mathcal{X}, \mathcal{Y}\}$ with the i_{th} feature vector $\boldsymbol{x}_i \in \mathbb{R}^m$ and label $y_i \in \{1, 2, 3, \ldots, k\}$, the idea of the multiclass SVM is to learn a discriminant function $f_{\boldsymbol{w}}$ with model parameter $\boldsymbol{w}$ to predict a most possible class label $\hat{y}$ by maximizing $f_{\boldsymbol{w}}$ over all $y \in \mathcal{Y}$ for a testing input $\boldsymbol{x}$ with

$$\hat{y} = \arg\max_{y \in \mathcal{Y}} f_{\boldsymbol{w}}(\boldsymbol{x}, y). \tag{1}$$

Here, $f_{\boldsymbol{w}}(\boldsymbol{x}, y)$ takes the linear form

$$f_{\boldsymbol{w}}(\boldsymbol{x}, y) = \boldsymbol{w}^T \Psi(\boldsymbol{x}, y), \quad y \in \mathcal{Y}, \tag{2}$$

where $\Psi(\boldsymbol{x}, y) \in \mathbb{R}^N$ is a feature mapping between input sample $\boldsymbol{x}$ and output y; $\boldsymbol{w} \in \mathbb{R}^N$ defines a weight for each element in the feature mapping Ψ, and $f_{\boldsymbol{w}}$ is a measurement showing how the input $\boldsymbol{x}$ matches an output y.

Based on the regularized empirical risk minimization, the objective function of multiclass SVM is defined as:

$$\begin{aligned} &\min_{\boldsymbol{w}, \boldsymbol{\xi} \geq \mathbf{0}} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{C}{n}\sum_{i=1}^{n}\xi_i \\ &\text{s.t.} \quad \forall\, i \in \{1, 2, \ldots, n\},\ \hat{y}_i \in \mathcal{Y}: \\ &\Delta(\hat{y}_i, y_i) + \boldsymbol{w}^T\Psi(\boldsymbol{x}_i, \hat{y}_i) - \boldsymbol{w}^T\Psi(\boldsymbol{x}_i, y_i) \leq \xi_i. \end{aligned} \tag{3}$$

where n is the number of training samples; C is a non-negative parameter to balance the loss term and the regularization term; $\boldsymbol{\xi}$ denotes a sequence of slack variables for the hinge loss; Δ is the Hamming loss function; $\hat{y}_i$ is the estimated output of the input $\boldsymbol{x_i}$; and y_i is the ground truth.

Albeit the effectiveness of the multiclass SVM in many applications, it is tailored for vector-form data and incapable of sufficiently taking advantage of the rich structural information hidden in matrix-form EEG data for better multiclass classification.

4. Method

In this section, we first introduce the proposed matrix classifier MSMM, aiming at efficiently capturing the correlation within each EEG matrix for better multiclass classification, and then we provide the details of the proposed solver to minimize the objective function of the proposed MSMM.

4.1. MSMM formulation

As mentioned before, each EEG sample can be naturally represented in matrix form, which can well preserve spatio-temporal information within the sample. However, most existing classifiers aim at coping with features in vector form. While a few matrix classifiers have been proposed, they focus on binary classification and hence are incapable of dealing with multi-task BCIs. This motivates us to develop a new classifier for multiclass classification of EEG signals.

Given a k-class ($k \geq 2$) matrix-form training dataset $\{\mathbf{X}_i, y_i\}_{i=1}^n \in \{\mathcal{X}, \mathcal{Y}\}$, where $\mathbf{X}_i \in \mathbb{R}^{I_1 \times I_2}$ is the ith feature matrix and $y_i \in \{1, 2, 3, \ldots, k\}$ is the corresponding ground truth label, we devise a novel objective function in order to train an efficient multiclass classifier to predict the label of a new observation:

$$\begin{aligned} &\min_{\mathcal{W}, \boldsymbol{\xi} \geq 0} \frac{1}{2}||\mathcal{W}||_F^2 + \tau\sum_{c=1}^{k}||\mathbf{W}_c||_* + \frac{C}{n}\sum_{i=1}^{n}\xi_i, \\ &\text{s.t.} \quad \forall\, i \in \{1, 2, \ldots, n\},\ \hat{y}_i \in \mathcal{Y}: \\ &\Delta(\hat{y}_i, y_i) + \langle \mathcal{W}, \delta\Psi(\mathbf{X}_i, \hat{y}_i, y_i)\rangle \leq \xi_i, \end{aligned} \tag{4}$$

where $\mathcal{W} \in \mathbf{R}^{I_1 \times I_2 \times k}$ denotes the regression parameter in the form of tensor and $||\mathcal{W}||_F$ is the Frobenius norm of $\mathcal{W}$; each frontal slice $\{\mathbf{W}_{:,:,c}\}_{c=1}^k$ in $\mathcal{W}$ (short for $\{\mathbf{W}_c\}_{c=1}^k$) represents the matrix-form hyperplane for the cth class data; τ and C are positive scalars to constrain the nuclear norm and loss term respectively; $\boldsymbol{\xi}$ denotes a sequence of slack variables for the hinge loss; and $\delta\Psi(\mathbf{X}_i, \hat{y}_i, y_i)$ denotes the difference of feature mappings between an arbitrary label $\hat{y}_i$ and the ground truth label y_i for $\mathbf{X}_i$ with the following definition:

$$\delta\Psi(\mathbf{X}_i, \hat{y}_i, y_i) = \Psi(\mathbf{X}_i, \hat{y}_i) - \Psi(\mathbf{X}_i, y_i). \tag{5}$$

Here the feature mapping $\Psi(\mathbf{X}, c) \in \mathbb{R}^{I_1 \times I_2 \times k}$ is a sparse tensor with all zero elements except $\Psi_{:,:,c} = \mathbf{X}$.

Our objective function is devised based on a spectral elastic net regularization, which is a combination of the squared Frobenius norm $||\mathcal{W}||_F^2$ and the nuclear norm $\sum_{c=1}^k ||\mathbf{W}_c||_*$. The squared Frobenius norm has the similar function as its vector-form counterpart $\boldsymbol{w}^T\boldsymbol{w}$ defined in Eq. (3); it controls model complexity and prevents over-fitting problems in the training phase. In order to take full advantage of the structural information of matrix (i.e., the correlation between rows and columns), we harness the nuclear norm to add penalty on the singular values of all the hyperplane $\mathbf{W}_c$ ($c \in 1, 2, \ldots\ldots, k$), which is defined as ($||\mathbf{W}_c||_* = \sum_{i=1}^r \sigma_i$) to approximate the rank of $\mathbf{W}_c$ in a convex manner. Based on the fundamental assumption that the true model parameter is sparse in terms of its rank [23] and the fact that nuclear norm is considered as the best convex approximation of low rank, such a combined regularization term can lead to an optimal solution of $\mathcal{W}$ sufficiently and elegantly encoded the structural information of matrix for better classification performance. Note that the τ is a key parameter in our objective function; it determines how large the penalty added to the nuclear norm is, and hence implicitly reflects how much structural information of the EEG matrix should be involved in the classification. The τ should be set as a positive scalar to guarantee the function of the nuclear norm term.

4.2. Solver for MSMM

Directly solving Eq. (4) can be extremely difficult, because (1) there exist n slack variables to be estimated and (2) both the multiclass hinge loss and the nuclear norm are non-smooth and non-differentiable. In order to tackle these issues, we first reduce the n slack variables to a single one by leveraging the independence of each estimated label and then propose a new algorithm based on the alternating direction method of multipliers (ADMM) [41,42] framework to solve the problem.

The constraints in Eq. (4) indicate that for each training sample, the score of $< \mathcal{W}, \Psi(\mathbf{X}_i, \hat{y}_i) >$ of an arbitrary label $\hat{y}_i$ must be smaller than the score $< \mathcal{W}, \Psi(\mathbf{X}_i, y_i) >$ of the correct label y_i by a required margin $\Delta(\hat{y}_i, y_i)$. If the margin is violated, the slack variable ξ_i of the sample becomes non-zero. In this regard, $\Sigma_i \xi_i$ is an upper bound on the empirical risk on the training samples. Training such a multiclass support matrix machine on large-scale problems is very challenging. We derive an equivalent formulation to reduce the slack variables and reformulate Eq. (4) as follows:

$$\begin{aligned} &\min_{\mathcal{W}, \boldsymbol{\xi} \geq 0} \frac{1}{2}||\mathcal{W}||_F^2 + \tau\sum_{c=1}^{k}||\mathbf{W}_c||_* + C\xi \\ &\text{s.t.} \quad \forall\, (\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n) \in \mathcal{Y}^n: \\ &\frac{1}{n}\sum_{i=1}^{n}\{\Delta(\hat{y}_i, y_i) + \langle \mathcal{W}, \delta\Psi(\mathbf{X}_i, \hat{y}_i, y_i)\rangle\} \leq \xi, \end{aligned} \tag{6}$$

where ξ is an equivalent upper bound of the inequalities in all constraints. In principle, Eq. (6) enlarges the number of constraints to $|\mathcal{Y}|^n$, where each element denotes a possible estimated label combination $(\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n) \in \{1, 2, \ldots, k\}^n$. Compared with Eq. (4), Eq. (6) has only one slack variable ξ, which is shared among all

constraints instead of setting one slack variable for each constraint. This reformulation is obtained based on the following theorem.

Theorem 1. *The optimal solution $\mathcal{W}^*$ of Eq. (4) is equal to the one in Eq. (6), with $\xi^* = \frac{1}{n}\sum_{i=1}^{n}\xi_i^*$. See its proof in Appendix 1(1).*

We further convert Eq. (6) into an unconstrained problem as:

$$\min_{\mathcal{W}} \frac{1}{2}||\mathcal{W}||_F^2 + \tau\sum_{c=1}^{k}||\mathbf{W}_c||_* + \max_{\{\hat{y}_1\ldots\hat{y}_n\}\in\mathcal{Y}^n}\frac{C}{n}\sum_{i=1}^{n}\left\{\Delta(\hat{y}_i,y_i)+\langle\mathcal{W},\delta\Psi(\mathbf{X}_i,\hat{y}_i,y_i)\rangle\right\}. \tag{7}$$

In this case, our MSMM model aims to minimize a regularized loss which maximizes the margins between different categories.

Most existing matrix classifier solvers, such as Nesterov method [23], require the objective function has Lipschitz-continuous derivative. However, both multiclass hinge loss and the nuclear norm in our formulation are non-smooth and non-differentiable. In this regard, existing matrix classifier solvers cannot be employed to solve our formulation.

The Eq. (7) consists of three terms and all of them are convex. The first two terms (the Frobenius norm and the nuclear norm of $\mathcal{W}$, respectively) are convex because they satisfy the triangle inequality and positive homogeneity properties [58]. The third term is the maximum of a set of linear functions and in this case, it is also convex. Based on the observation that all the terms in Eq. (7) are convex, we therefore develop a novel solver based on the (ADMM) framework [41,42], which is a widely used scheme to solve convex optimization problems by breaking them into sub-problems that are much easier to be coped with.

We first introduce an additional decision variable $\mathcal{S}\in\mathbb{R}^{I_1\times I_2\times k}$ to split the primal problem into two parts:

$$\begin{aligned}&\operatorname*{argmin}_{\mathcal{W},\mathcal{S}} H(\mathcal{W})+G(\mathcal{S}),\\ &\text{s.t.}\quad \mathcal{W}-\mathcal{S}=0.\end{aligned} \tag{8}$$

with

$$\begin{aligned}H(\mathcal{W}) &= \max_{\{\hat{y}_1\ldots\hat{y}_n\}\in\mathcal{Y}^n}\frac{C}{n}\sum_{i=1}^{n}\{\Delta(\hat{y}_i,y_i)\\ &\quad+\langle\mathcal{W},\delta\Psi(\mathbf{X}_i,\hat{y}_i,y_i)\rangle\},\\ G(\mathcal{S}) &= \frac{1}{2}||\mathcal{S}||_F^2+\tau\sum_{c=1}^{k}||\mathbf{S}_c||_*,\end{aligned} \tag{9}$$

where $\mathbf{S}_c\in\mathbb{R}^{I_1\times I_2}$ denotes the c_{th} frontal slice of $\mathcal{S}$.

We then reformulate Eq. (8) by using augmented Lagrangian method:

$$\begin{aligned}L(\mathcal{S},\mathcal{W},\mathcal{V}) = \; & H(\mathcal{W})+G(\mathcal{S})+\langle\mathcal{V},(\mathcal{S}-\mathcal{W})\rangle\\ &+\frac{\rho}{2}||(\mathcal{S}-\mathcal{W})||_F^2,\end{aligned} \tag{10}$$

where $\mathcal{V}\in\mathbb{R}^{I_1\times I_2\times k}$ is the Lagrange multiplier and ρ is a positive scalar hyperparameter bounded away from 0.

Next we decouple the objective function into two sub-problems (with respect to $\mathcal{S}$ and $\mathcal{W}$) and solve it in an iterative fashion. At each iteration, our solver first minimizes $\mathcal{S}$ and $\mathcal{W}$ alternatively, and updates the Lagrangian multiplier $\mathcal{V}$ accordingly as

$$\begin{aligned}\mathcal{S}^{(t+1)} &= \operatorname*{arg\,min}_{\mathcal{S}} L(\mathcal{S},\mathcal{W}^{(t)},\mathcal{V}^{(t)}),\\ \mathcal{W}^{(t+1)} &= \operatorname*{arg\,min}_{\mathcal{W}} L(\mathcal{S}^{(t+1)},\mathcal{W},\mathcal{V}^{(t)}),\\ \mathcal{V}^{(t+1)} &= \mathcal{V}^{(t)}+\rho(\mathcal{S}^{(t+1)}-\mathcal{W}^{(t+1)}).\end{aligned} \tag{11}$$

where t and $t+1$ represent the (t)th and $(t+1)$th iteration, respectively.

4.2.1. Solve subproblem of $\mathcal{S}$

Assume $\mathcal{W}$ is fixed, minimizing the objective function is to minimize the sum of all terms related to $\mathcal{S}$, denoted as L_S:

$$\min_{\mathcal{S}} L_S = G(\mathcal{S})+\langle\mathcal{V},\mathcal{S}\rangle+\frac{\rho}{2}||\mathcal{S}-\mathcal{W}||_F^2. \tag{12}$$

We further concern the optimization problem in Eq. (12) to update $\mathcal{S}$, namely, we update $\mathcal{S}$ by minimizing L_S. As L_S is a non-differentiable but convex function, we derive the subgradient of L_S and denote it as a tensor $\mathcal{K}\in\mathbb{R}^{I_1\times I_2\times k}$ with $\forall c$,

$$\mathbf{K}_c = \mathbf{S}_c+\tau\partial||\mathbf{S}_c||_*+\mathbf{V}_c+\rho(\mathbf{S}_c-\mathbf{W}_c), \tag{13}$$

where $\mathbf{K}_c$, $\mathbf{S}_c$, $\mathbf{V}_c$ and $\mathbf{W}_c$ is the c_{th} frontal slice of $\mathcal{K}$, $\mathcal{S}$, $\mathcal{V}$ and $\mathcal{W}$ respectively; $\partial||\mathbf{S}_c||_*$ denotes the sub-gradient set of the nuclear norm of $\mathbf{S}_c$.

In this case, suppose $\mathbf{S}_c^*$ is an optimum of L_S, the subgradient of L_S at point $\mathbf{S}_c^*$ satisfies $\mathbf{0}\in\partial L_S(\mathbf{S}_c^*)$.

To figure out $\mathbf{S}_c^*$, we have the following theorem.

Theorem 2. *For $\tau\geq 0$, one optimal solution for the following problem*

$$\arg\min_{\mathcal{S}} G(\mathcal{S})+\langle\mathcal{V},\mathcal{S}\rangle+\frac{\rho}{2}||\mathcal{S}-\mathcal{W}||_F^2$$

is

$$\mathbf{S}_c^* = \frac{1}{1+\rho}D_\tau(\rho\mathbf{W}_c-\mathbf{V}_c), \tag{14}$$

where $D_\tau(.)$ is the singular value thresholding operator. See its proof in Appendix 1(2).

Based on Eq. (14), each singular value of $\mathbf{S}_c$ will reduce the value of τ or will be set to zero if it is smaller than τ by the singular thresholding operator $D_\tau(\cdot)$. In this regard, τ softly thresholds the rank of $\mathbf{S}_c$.

4.2.2. Solve subproblem of $\mathcal{W}$

Similar to the subproblem of $\mathcal{S}$, we minimize the sum of all terms related to $\mathcal{W}$ in Eq. (10) and denote it as L_W:

$$\min_{\mathcal{W}} L_W = H(\mathcal{W})+\langle-\mathcal{V},\mathcal{W}\rangle+\frac{\rho}{2}||\mathcal{S}-\mathcal{W}||_F^2 \tag{15}$$

L_W is non-negative weighted sum of the hinge loss, a linear function and a square function. It is also convex as all the three terms are convex. In principle, Eq. (15) contains $|\mathcal{Y}|^n$ constraints; hence, it is intractable to be fed into a quadratic solver. To tackle this issue, we apply the cutting plane (CP) algorithm, which has been proved to achieve nice bounded approximated solutions by selecting only a small subset of constraints [40].

In CP, one of the most important steps is to construct an extensible constraint subset Ω during the iterations, starting with $\Omega=\emptyset$. At each iteration, we add the most violated constraint to Ω and optimize the problem in Eq. (15) over all constraints in Ω. The algorithm has also been proved to converge after polynomial iteration independent of the number of training samples, which makes it suitable to handle large scale data [40].

In this case, the most violated constraint with respect to minimizing L_W at each iteration can be defined as:

$$\xi^* = \max_{\{\hat{y}_1\cdots\hat{y}_n\}\in\mathcal{Y}^n}\frac{C}{n}\sum_{i=1}^{n}\{\Delta(\hat{y}_i,y_i)+\langle\mathcal{W},\delta\Psi(\mathbf{X}_i,\hat{y}_i,y_i)\rangle\}, \tag{16}$$

To derive $\hat{y}_i$, we have the following theorem.

Theorem 3. *Each estimated label in the most violated constraint should satisfy*

$$\hat{y}_i = \arg\max_{y\in\mathcal{Y}}\Delta(y,y_i)+\langle\boldsymbol{\mathcal{W}},\delta\Psi(\mathbf{X}_i,y,y_i)\rangle. \tag{17}$$

See its proof in Appendix 1(3).

In this way, we can solve $L_{\mathcal{W}}$ by gradient descent method with a much smaller constraint subset Ω. Since $L_{\mathcal{W}}$ is non-differential but convex at $\mathcal{W}$, the sub-gradient of $L_{\mathcal{W}}$ with respect to $\mathcal{W}$ is

$$\nabla_{\mathcal{W}} = \rho(\mathcal{W} - \mathcal{S}) - \mathcal{V} + \frac{C}{n}\sum_{i=1}^{n}\delta\Psi(\mathbf{X}_i, \hat{y}_i, y_i), \tag{18}$$

where $\hat{y}_i$ is calculated with Eq. (17).

Thus we can update $\mathcal{W}$ using:

$$\mathcal{W}^{(t+1)} = \mathcal{W}^{(t)} - \eta^{(t)}\nabla_{\mathcal{W}}, \tag{19}$$

where $\eta^{(t)}$ denotes the step size for gradient descent method at t_{th} iteration. Instead of fixing or choosing $\eta^{(t)}$ randomly, we determine the step size automatically by employing Pegasos algorithm [59] with:

$$\eta^{(t)} = \frac{C}{t\rho^{(t)}}. \tag{20}$$

Finally the augmented Lagrangian multiplier can be updated with

$$\mathcal{V}^{(t+1)} = \mathcal{V}^{(t)} + \rho^{(t)}(\mathcal{S}^{(t+1)} - \mathcal{W}^{(t+1)}). \tag{21}$$

$$\rho^{(t+1)} = \beta\rho^{(t)} \tag{22}$$

In each iteration, ρ is automatically increased via multiplying by $\beta > 1$ ($\beta = 1.1$ for our experiments). This scheme speeds up the convergence of Lagrangian multiplier $\mathcal{V}$ and thus the whole algorithm. The overall of the algorithm is summarized in Algorithm. 1.

Algorithm 1: The proposed solver for MSMM.

Input : Training data $\{(\mathbf{X}_i, y_i)\}_{i=1}^{n}$, input coefficients C and τ, Lagrangian multiplier $\mathcal{V}$, ρ, β and ϵ

Output: $\mathcal{W}$

Initialize: $\mathcal{W}, \mathcal{S}, \mathcal{V} \leftarrow \mathbf{0}$, $\xi \leftarrow 0$, $\Omega \leftarrow \emptyset$

while $t \leftarrow 1$ ***to*** *maxIter* **do**

  Update each frontal slice of $\mathcal{S}$ with Eq. (14)

  /*~Update $\mathcal{W}$ with cutting plane algorithm~*/

  for $i \leftarrow 1$ ***to*** n **do**

    $\hat{y}_i = \arg\max_{y \in \mathcal{Y}}\{\Delta(y, y_i) + \langle \mathcal{W}, \delta\Psi(\mathbf{X}_i, y, y_i)\rangle\}$

  end

  if $\frac{1}{n}\sum_{i=1}^{n}\{\Delta(\hat{y}_i, y_i) + \langle \mathcal{W}, \delta\Psi(\mathbf{X}_i, \hat{y}_i, y_i)\rangle\} \geq \xi + \epsilon$ **then**

    $\Omega \leftarrow \Omega \cup \{\hat{y}_1 \dots \hat{y}_n\}$

    $\mathcal{W} \leftarrow \underset{\mathcal{W}}{\operatorname{argmin}}\ H(\mathcal{W}) - \langle \mathcal{V}, \mathcal{W}\rangle + \frac{\rho}{2}||\mathcal{S} - \mathcal{W}||_F^2$,

    s.t.,$\forall\{\hat{y}_1 \dots \hat{y}_n\} \in \Omega$.

    $\xi = \max\frac{1}{n}\sum_{i=1}^{n}\{\Delta(\hat{y}_i, y_i) + \langle \mathcal{W}, \delta\Psi(\mathbf{X}_i, \hat{y}_i, y_i)\rangle\}$

  else

    break

  end

  $\mathcal{V} \leftarrow \mathcal{V} + \rho(\mathcal{S} - \mathcal{W})$

  $\rho \leftarrow \beta\rho$

end

return $\mathcal{W}$

5. Experiments

We extensively evaluate the proposed method based on two widely used public single trial EEG datasets of multi-task motor imagery. Firstly, we experimentally evaluate the influence of the key parameter τ in the objective function on the performance of our method. Secondly, as we are not aware of any previous multiclass classifiers for the data in matrix form, we compare our method with two state-of-the-art binary classifiers for matrix data (bilinear SMM (BSMM) [34] and support matrix machine (SMM) [35]), a competitive multiclass classifier for data in vector form (multiclass SVM (MSVM) [40]) and three widely used binary classifiers for vector data (linear discriminant analysis (LDA) [21], k-nearest neighbor (KNN) [60] and multilayer perceptron (MLP) [61]). We also compare our method with some methods that achieved leading performance on both datasets in the competitions.

5.1. EEG datasets

The first dataset is the *Dataset IIIa*[1] *of BCI Competition III* [43]. This dataset contains 60-channel single trial EEG signals from three subjects (denoted as subject *k3b, k6b* and *l1b*) when performing four classes of motor imagery, including left-hand, right-hand, feet and tongue (namely class 1,2,3 and 4, respectively). Both training and testing sets consist of 45 trials per class for subject *k3b*, and 30 trials per class for subject *k6b* and *l1b*. The EEG was sampled with 250 Hz and filtered between 1 and 50 Hz with notch filter. As the the performance of subject *k6b* is not good during the data collection,[2] we only use the data of *k3b* and *l1b* in our experiments. For this dataset, data of all channels during time segment from 3 s to 7 s in each trial is chosen for analysis.

The second dataset is the *Dataset IIa*[3] *of BCI Competition IV* [24]. The dataset records single trial EEG data from nine subjects performing four classes of motor imagery, including left-hand, right-hand, feet and tongue. The training and testing data are acquired in two sessions conducted on two different days. Each session includes six runs with 48 trials in each run. There are 72 trials per motor imagery task and 288 trials in total per session. Besides 22 EEG channels, 3 monopolar EOG channels were also used to record the signals. Then signals were sampled with 250 Hz and bandpass-filtered between 0.5 and 100 Hz. For this dataset, we consider only the EEG channels and the time segment of [1 s,4 s] after onset of the visual cue in each trial.

5.2. Experimental settings

We implement our method in Matlab 2014b on a workstation with Intel Xeon CPU E5-1620 v2 3.70 GHz, 16.0 GB RAM and 64-bit Windows 7 System. Fig. 2 shows the pipeline of the EEG signal processing framework equipped with our MSMM for multiclass single trial EEG classification. In this paper, we focus on the classifier for EEG features in matrix form, namely the last part of the pipeline. The preprocessing and feature extraction techniques are beyond the scope of this work. For preprocessing, based on Ang's et al. work [25], we employed non-overlapping bandpass filters of six-order Butter-worth to filter out the artifacts and unrelated sensorimotor rhythms; and then we applied a traditional multiclass CSP method [26] to select the most dominant channels for each motor imagery task. For feature extraction, we have empirically experimented with a number of existing algorithms to extract features in matrix form, such as band powers (BP) [62], power spectral density (PSD) [63], and time-domain parameters (TDP) [64]. It turned out that the TDP consistently delivered low computational cost and top performance. So in this work, we choose TDP.

For classification, we applied our model to train a classifier with matrix features as input. In our method, C and τ weight the loss and nuclear norm term, respectively, which need to be determined before training.

In the experiment, we employ the grid search algorithm guided by the cross validation to determine the values of C and τ. Specif-

[1] http://www.bbci.de/competition/iii/#download.

[2] http://www.bbci.de/competition/iii/#data_set_iiia.

[3] http://www.bbci.de/competition/iv/#dataset2a.

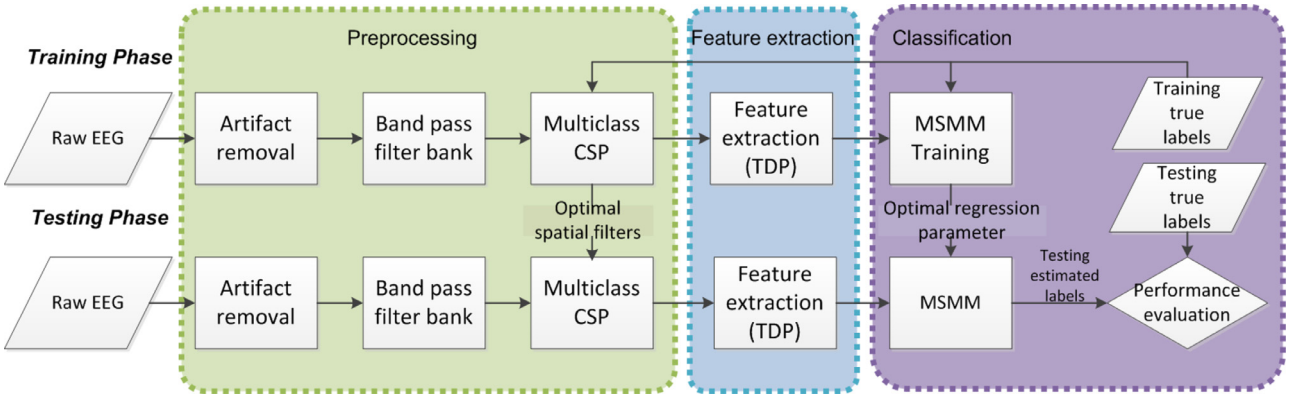


Fig. 2. The pipeline of the EEG signal processing framework equipped with our MSMM.

ically, C and τ were selected from the candidate sets {0.1, 1, 10, 100} and {0, 0.1, 0.5, 1, 2, 5, 10}, which are figured out by some pilot experiments. Then we train a MSMM with each pair (C, τ) in the Cartesian product of these two sets and evaluate their performance on the fivefold cross validation. Finally, we select the settings achieved the highest score in the validation procedure. For competitive classifiers, we employed the same preprocessing and feature extraction schemes for fair comparison. In order to train vector-form classifiers (including MSVM, LDA, KNN and MLP), we concatenate the matrices into vectors and then employ PCA as a preprocessing step to reduce the dimension and prevent memory overflow. For the PCA, we determine the principle components by 90% energy ratio. All compared binary classifiers are extended to cope with multiclass cases with OvR strategy.

5.3. Evaluation metrics

In order to comprehensively measure the classification performance of different classifiers, we employ four evaluation metrics, namely, kappa coefficient κ, precision, recall and F_1 score. We consider kappa coefficient κ rather than accuracy because it is more robust than accuracy by taking into account the accuracy occurring by chance, and it is defined as:

$$\kappa = \frac{acc - p_0}{1 - p_0}. \tag{23}$$

where acc is the classification accuracy and p_0 is the accuracy of random guess (e.g., for a four-class dataset with balanced sample sizes among different classes, $p_0 = \frac{1}{4}$). Note that $\kappa > 0$ means the accuracy we gain is better than the one of random guess and higher κ value means better classification accuracy.

In the classification, precision is a measure of classification relevancy. A low precision can indicate many false positives. Recall can be regarded as a measure of classification completeness. A low recall indicates many false negatives. F_1 score is the harmonic mean of the precision and recall. A system with high recall but low precision returns many results but most of its predicted labels are incorrect when compared to the ground truth. A system with high precision but low recall returns very few results but most of its predicted labels are correct. High scores for both precision and recall indicate that a classifier is of high prediction quality. Since precision, recall and F_1 score are metrics for binary classification, we employ a macro-averaging scheme to average the same measures calculated for each class [65]. These measures for multiclass classification are obtained based a generalization of the measures of Table 1 for k classes; they are defined as:

$$precision = \frac{1}{k}\sum_{c=1}^{k}\frac{tp_c}{tp_c + fp_c},$$

Table 1
Confusion matrix for binary classification.

Data class	Classified as pos	Classified as neg
pos	true positive (tp)	false negative (fn)
neg	false positive (fp)	true negative (tn)

$$recall = \frac{1}{k}\sum_{c=1}^{k}\frac{tp_c}{tp_c + fn_c},$$
$$F_1 = \frac{2 \cdot precision \cdot recall}{precision + recall}, \tag{24}$$

where tp_c, fp_c, fn_c, tn_c are the true positive, false positive, false negative and true negative for class c.

5.4. Results

5.4.1. Experiments for the influence of τ

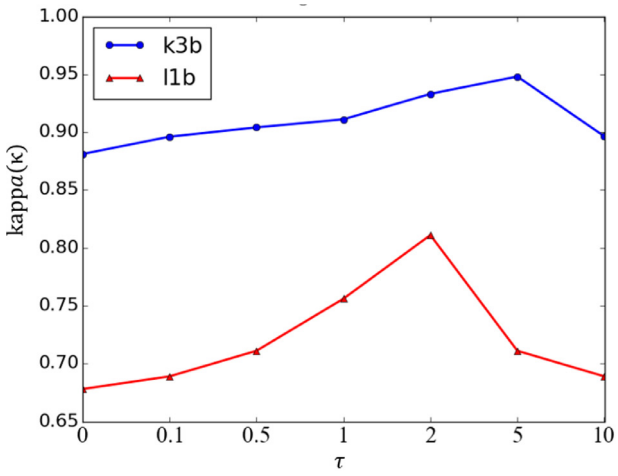
We first delve into the effect of parameter τ in the proposed objective function (Eq. (4)) on the performance of the MSMM. In principle, we can adjust the magnitude of the penalty added on the nuclear norm term by setting different values of τ, and hence determine how much structural information is involved in the classification. In fact, by carefully observing Eq. (14), we can find that the τ manages the penalty by controlling the number of singular value (rank) of the regression parameter. Generally, a larger τ indicates a more powerful penalty on the structure information. Note that when $\tau = 0$, the MSMM degenerates to the MSVM.

To study the effect of τ, we fix C according to the previous cross validation and train MSMM with $\tau = \{0, 0.1, 0.5, 1, 2, 5, 10\}$. The classification performance curves with respect to the changes of τ are presented in Fig. 3. For a clear illustration, we just show the curves of subjects S3, S7 and S9 in Fig. 3(b). In fact, similar trends occur for other six subjects in the second dataset.

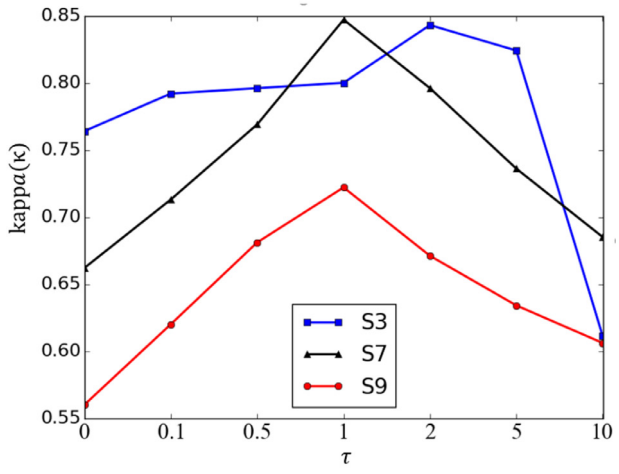
It is observed that, for all the five subjects, when $\tau = 0$, the nuclear norm in the objective function is inactive and hence our results are the same as those of MSVM (please check Tables 2 and 7 for the exact values). With the increase of τ, the κ values of all the five subjects increase with varying rise rate, demonstrating that taking the structural information encoded in the matrix form of EEG data into account can improve the classification accuracy. At a certain value of τ (the value is different for different subject), the κ reaches its optimal value. When τ crosses the optimal value and continues to increase, the κ value decreases. This is because when the τ is too large, most of the singular values in the regression parameter would be set to zero and most structural information embedded in the EEG matrix would be discarded. In this regard, it is necessary to choose a proper τ for each subject to improve the

Table 2
Performance κ (error rate %) comparison of different classification algorithms on dataset IIIa of BCI III.

Subject	BSMM	SMM	MSVM	LDA	KNN	MLP	MSMM
k3b	0.941 (4.4)	0.852 (11.1)	0.889 (8.3)	0.889 (8.3)	0.763 (17.8)	0.807 (14.4)	**0.948** (3.9)
l1b	0.800 (15.0)	0.711 (21.7)	0.678 (24.2)	0.478 (39.2)	0.700 (22.5)	0.489 (38.3)	**0.811** (14.2)
avg	0.871 (9.7)	0.782 (16.4)	0.784 (16.3)	0.683 (23.8)	0.732 (20.1)	0.648 (26.4)	**0.880** (9.0)



(a) Dataset IIIa of BCI III.



(b) Dataset IIa of BCI IV.

Fig. 3. The effect of parameter τ on the classification performance: (a) the classification performance curves of subjects *k3b* and *l1b* in the first dataset with respect to the changes of τ and (b) the classification performance curves of subjects *S3*, *S7* and *S9* in the second dataset.

Table 3
Testing performance (precision) comparison of different classification algorithms on dataset IIIa of BCI III.

Subject	BSMM	SMM	MSVM	LDA	KNN	MLP	MSMM
k3b	0.957	0.899	0.918	0.921	0.836	0.872	**0.962**
l1b	0.862	0.794	0.782	0.616	0.781	0.652	**0.869**
avg	0.910	0.847	0.850	0.768	0.808	0.762	**0.916**

Table 4
Testing performance (recall) comparison of different classification algorithms on dataset IIIa of BCI III.

Subject	BSMM	SMM	MSVM	LDA	KNN	MLP	MSMM
k3b	0.956	0.889	0.917	0.917	0.822	0.856	**0.961**
l1b	0.850	0.783	0.758	0.608	0.775	0.617	**0.858**
avg	0.903	0.836	0.838	0.763	0.799	0.736	**0.910**

Table 5
Testing performance (F_1 score) comparison of different classification algorithms on dataset IIIa of BCI III.

Subject	BSMM	SMM	MSVM	LDA	KNN	MLP	MSMM
k3b	0.956	0.894	0.917	0.919	0.829	0.864	**0.962**
l1b	0.856	0.789	0.770	0.612	0.778	0.634	**0.864**
avg	0.906	0.841	0.844	0.766	0.804	0.749	**0.913**

classification accuracy. We experimentally find that choosing a τ between (0, 2] can achieve appealing results for both datasets.

5.4.2. Comparison of other classifiers

To evaluate the effectiveness of our method, we compare the classification performance of our method with other state-of-the-art or widely used classifiers on the two datasets under different evaluation metrics. Our competitors include bilinear SMM (BSMM) [34], SMM [35], multiclass SVM (MSVM) [40], LDA [21], KNN [60] and MLP [61]. To produce their best performance, we acquire their implementations in public domain, generating a large number of results by trying and fine-tuning their parameters via cross validation for fair comparison. In addition, we also compare our method with methods that achieved leading performance on the two public EEG datasets; they are winner methods reported won the BCI competitions and newly proposed methods ([18,66]) that achieve good performance.

Results of the first dataset. The kappa values of different classifiers on the first dataset are reported in Table 2. We also include the error rate (%) for easy comparison. Among these seven methods, the BSMM, SMM and our MSMM are classifiers for matrix-form EEG data while the rest methods are learned for vector-form EEG data. It is observed that the BSMM and our MSMM achieve much better results than other methods, demonstrating that leveraging the structural information embedded in data matrices is greatly beneficial to the improvement of the classification performance. The SMM achieves mediocre results, though it is also a classifier for data in matrix form. This may be because the unbalanced training data when employing the OvR strategy could result in relatively large variations of the confidence values for different categories. The different scales of confidence values would bias to any specific task, which explains that the high performance on some MI tasks but degraded performance on all MI tasks. Similar phenomena also occur on other binary classifiers like LDA.

We further compute the precision, recall and F_1 score measures on the first dataset and report the results in Tables 3–5, respectively. It is observed that both MSMM and BSMM have an obvious improvement compared with other classifiers. On the contrary, the binary vector-form classifiers like LDA, KNN and MLP have relatively lower average precision and recall compared with the matrix-form classifiers (including MSMM, BSMM and SMM) and multiclass MSVM. In addition, our method, integrating the multiclass formulation and matrix-form classifier, achieves state-of-the-art results for both subjects and yields a mean value of 0.916, 0.910 and 0.913 for precision, recall and F_1 score, respectively. This demonstrates the effectiveness of our method for multi-task EEG signal analysis.

Table 6 shows the performance comparison of the proposed method with those of the winners of the competition and the newly proposed work [66] for this dataset. Although the first win-

Table 6
Performance κ (error rate %) comparison of the proposed MSMM with the winners and a newly published method [66] of the dataset IIIa of BCI III.

Subject	1st winner	2nd winner	3rd winner	[66]	MSMM
k3b	0.822 (13.3)	0.904 (7.2)	**0.948** (3.9)	0.711 (21.7)	**0.948** (3.9)
l1b	0.800 (15.0)	0.711 (21.7)	0.522 (35.8)	0.489 (38.3)	**0.811** (14.2)
avg	0.811 (14.2)	0.808 (14.4)	0.735 (19.9)	0.600 (30)	**0.880** (9.0)

Table 7
Performance κ (error rate %) comparison of different algorithms on dataset IIa of BCI IV.

Subject	BSMM	SMM	MSVM	LDA	KNN	MLP	MSMM
S1	0.727 (20.5)	0.694 (22.9)	0.722 (20.8)	0.671 (24.7)	0.708 (21.9)	0.588 (30.9)	**0.731** (20.1)
S2	0.403 (44.8)	0.230 (57.6)	0.370 (47.2)	0.421 (43.4)	0.398 (45.1)	0.315 (51.4)	**0.426** (43.1)
S3	0.750 (18.8)	0.685 (23.6)	0.764 (17.7)	0.722 (20.8)	0.773 (17.0)	0.639 (27.1)	**0.843** (11.8)
S4	0.505 (37.2)	0.537 (34.7)	0.357 (48.3)	0.509 (36.8)	0.449 (41.3)	0.505 (37.2)	**0.593** (30.6)
S5	0.394 (45.5)	0.315 (51.4)	0.417 (43.8)	0.426 (43.1)	0.380 (46.5)	0.407 (44.4)	**0.495** (37.8)
S6	0.315 (51.4)	0.152 (63.5)	0.185 (61.1)	0.315 (51.4)	0.236 (57.3)	0.171 (62.2)	**0.407** (44.4)
S7	0.810 (14.2)	0.722 (20.8)	0.662 (25.3)	0.565 (32.6)	0.694 (22.9)	0.718 (21.2)	**0.847** (11.5)
S8	0.708 (21.9)	0.708 (21.9)	0.454 (41.0)	0.713 (21.5)	0.620 (28.5)	0.454 (41.0)	**0.769** (17.4)
S9	0.620 (28.5)	0.630 (27.8)	0.560 (33.0)	0.611 (29.2)	0.481 (38.9)	0.500 (37.5)	**0.722** (20.8)
avg	0.581 (31.4)	0.519 (36.0)	0.499 (37.6)	0.550 (33.7)	0.527 (35.5)	0.477 (39.2)	**0.648** (26.4)

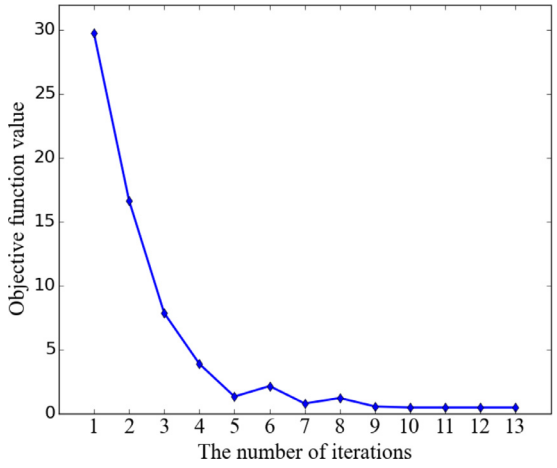


Fig. 4. Curve line of convergence process with MSMM.

ner method [67] achieves relatively good performance for subject *l1b* and the 3rd winner also obtained best performance for subject *k3b* as ours, both of them employ quite complex preprocessing techniques to extract the discriminative vector-form features. The extracted features have lost the inherent structural information, thus the first winner method obtained moderate performance on subject *k3b* and method of 3rd winner have degraded performance on subject *l1b*. In contrast, our method achieves the best results on the first dataset overall, *i.e.*, our method obtains better results than first winner method with a great 6.9% improvement in the mean kappa value.

Fig. 4 shows the convergence process of the proposed MSMM on the subject *k3b*. It shows that our method based on ADMM framework can converge to the global optimum in a few iterations. Similar trend also occurs for other subjects in both datasets.

Results of the second dataset. The classifier comparison results of the second dataset are reported in Table 7. We can observe that for most subjects, the BSMM and our MSMM achieve much better results than other methods, which may result from leveraging the structure information embedded in the data of matrix form. Though KNN is efficient, it ignores the structural information and is very sensitive to the curse-of-dimensionality. MLP even obtains the worst average results on both datasets. This is because MLP is a universal approximator which makes the classifiers sensitive to over-fitting problem, especially for the noisy and non-stationary EEG. Thus, it may fail in the real-world applications of EEG based BCIs [46]. In addition, our MSMM is the winning method, which consistently improves the kappa values across all the subjects for the single trial EEG classification. This is because our method is able to avoid the bias of multiple binary classifiers with the multiclass formulation and leverage the structural information by learning the low rank regularization from the noisy EEG features.

Table 8
Testing performance (precision) comparison of different classification algorithms on dataset IIa of BCI IV.

Subject	BSMM	SMM	MSVM	LDA	KNN	MLP	MSMM
S1	0.804	0.787	0.803	0.772	0.795	0.706	**0.822**
S2	0.542	0.460	**0.659**	0.537	0.563	0.505	0.571
S3	0.827	0.827	0.837	0.803	0.833	0.735	**0.884**
S4	0.643	0.664	0.679	0.646	0.632	0.629	**0.701**
S5	0.629	0.515	0.549	0.588	0.577	0.590	**0.653**
S6	0.514	0.431	0.411	0.505	0.519	0.372	**0.600**
S7	**0.893**	0.856	0.754	0.785	0.789	0.804	**0.893**
S8	0.802	0.797	0.768	0.799	0.763	0.603	**0.833**
S9	0.777	0.729	0.743	0.771	0.680	0.649	**0.805**
avg	0.715	0.674	0.689	0.690	0.684	0.621	**0.751**

Table 9
Testing performance (recall) comparison of different classification algorithms on dataset IIa of BCI IV.

Subject	BSMM	SMM	MSVM	LDA	KNN	MLP	MSMM
S1	0.795	0.771	0.792	0.753	0.781	0.691	**0.799**
S2	0.552	0.424	0.528	0.566	0.549	0.486	**0.569**
S3	0.813	0.764	0.823	0.792	0.830	0.729	**0.882**
S4	0.628	0.653	0.517	0.632	0.587	0.628	**0.694**
S5	0.545	0.486	0.563	0.569	0.535	0.556	**0.622**
S6	0.486	0.365	0.389	0.486	0.427	0.378	**0.556**
S7	0.858	0.792	0.747	0.674	0.771	0.788	**0.885**
S8	0.781	0.781	0.590	0.785	0.715	0.590	**0.826**
S9	0.715	0.722	0.670	0.708	0.611	0.625	**0.792**
avg	0.686	0.640	0.624	0.663	0.645	0.608	**0.736**

We also compute the measures of precision, recall and F_1 score for different classification algorithms on the second dataset. The results are shown in Tables 8–10. From Tables 8 and 10, the proposed method has the highest precision and F_1 score measures across all the subjects but *S2*. While in Table 9, we find that the proposed MSMM achieves best performance across all the subjects for recall evaluation. The high scores for all these measures indicate that MSMM has high prediction quality. Therefore, it again validates the benefits of leveraging the structural information and multiclass hinge loss for the EEG classification problem.

In Table 11, we also summarize the performance comparison of MSMM with top three methods of the competition and the newly

Table 10
Testing performance (F_1 score) comparison of different classification algorithms on dataset IIa of BCI IV.

Subject	BSMM	SMM	MSVM	LDA	KNN	MLP	MSMM
S1	0.799	0.779	0.797	0.763	0.788	0.699	**0.810**
S2	0.547	0.441	**0.586**	0.551	0.556	0.495	0.570
S3	0.820	0.794	0.830	0.797	0.832	0.732	**0.883**
S4	0.636	0.658	0.587	0.639	0.609	0.629	**0.698**
S5	0.584	0.500	0.556	0.579	0.555	0.572	**0.637**
S6	0.500	0.395	0.399	0.496	0.468	0.375	**0.577**
S7	0.875	0.823	0.750	0.725	0.780	0.796	**0.889**
S8	0.791	0.789	0.667	0.792	0.739	0.596	**0.830**
S9	0.745	0.726	0.705	0.738	0.644	0.637	**0.798**
avg	0.700	0.656	0.653	0.676	0.663	0.615	**0.744**

Table 11
Performance κ (error rate %) comparison of the proposed MSMM with the winners and a newly published method [18] of the dataset IIa of BCI IV.

Subject	1st winner	2nd winner	3rd winner	[18]	MSMM
S1	0.676 (24.3)	0.690 (23.3)	0.380 (46.5)	0.623 (28.5)	**0.731** (20.1)
S2	0.417 (43.8)	0.343 (49.3)	0.181 (61.5)	0.277 (54.2)	**0.426** (43.1)
S3	0.745 (19.1)	0.713 (21.5)	0.481 (38.9)	0.658 (25.7)	**0.843** (11.8)
S4	0.481 (38.9)	0.440 (42.0)	0.333 (50.0)	0.326 (50.7)	**0.593** (30.6)
S5	0.398 (45.1)	0.162 (62.8)	0.069 (69.8)	0.146 (63.9)	**0.495** (37.8)
S6	0.273 (54.5)	0.213 (59.0)	0.139 (64.6)	0.256 (55.9)	**0.407** (44.4)
S7	0.773 (17.0)	0.658 (25.7)	0.292 (53.1)	0.407 (44.4)	**0.847** (11.5)
S8	0.755 (18.4)	0.731 (20.1)	0.491 (38.2)	0.595 (30.6)	**0.769** (17.4)
S9	0.606 (25.9)	0.690 (23.3)	0.440 (42.0)	0.659 (25.7)	**0.722** (20.8)
avg	0.569 (32.3)	0.516 (36.5)	0.312 (51.7)	0.439 (42.0)	**0.648** (26.4)

Table 12
p-values of Friedmans and Iman–Davenports test.

Methods	κ	Precision	Recall	F_1 score
Friedman	**5.5998E−6**	**1.9699E−6**	**5.5998E−6**	**2.2359E−6**
Iman–Davenport	**3.5868E−8**	**4.0424E−9**	**3.5868E−8**	**5.3155E−9**

Table 13
Adjusted *p*-values of Holms method (MSMM is the control method).

Methods	κ	Precision	Recall	F_1 score
BSMM	**0.0263**	0.0841	**0.0263**	0.0756
SMM	**2.9053E−4**	**3.5577E−4**	**2.9053E−4**	**3.1561E−4**
MSVM	**1.9368E−4**	**0.0037**	**1.9368E−4**	**5.2981E−4**
LDA	**0.0022**	**4.3466E−4**	**0.0022**	**0.0015**
KNN	**1.0988E−4**	**2.9053E−4**	**1.0988E−4**	**3.1561E−4**
MLP	**1.7198E−6**	**4.5001E−7**	**1.7198E−6**	**5.9119E−7**

Table 14
Average training and average testing time on both datasets between different methods.

Methods	Dataset IIIa of BCI IIIa		Dataset IIa of BCI IV	
	#Train (s)	#Test (s)	#Train (s)	#Test (s)
BSMM	20.381	0.063	43.927	0.245
SMM	18.995	0.059	47.198	0.243
MSMM	22.257	0.054	65.528	0.230

published method [18] for the second dataset. The method of the first winner achieves the top results in the competition, as it employs a complicated method to select discriminative features in vector form for the final classification. Even though, our method achieves the best results for all subjects, demonstrating the effectiveness and robustness of the proposed MSMM. From Tables 7 and 11, we can observe that all the methods including ours have fair kappa values on subject *S2, S5* and *S6*. This may be because the EEG signals in this dataset were collected in two separate sessions and the non-stationarity problem is more obvious for these three subjects. However, compared with the competing methods, our method still obtains rather reasonable results for these cases.

Statistical significance. We further use the hypothesis-testing techniques to find the significant differences among the results obtained by the proposed MSMM and other compared algorithms. We use non-parameter tests because that the initial conditions that guarantee the reliability of the parametric tests may not be satisfied, thus causing incredible statistical analysis with parametric tests. Specifically, we employ the Friedman test as well as the Iman–Davenport test [68] to check whether there are significant differences in the performance among the seven algorithms. If the null hypothesis is rejected, then we can conduct pairwise comparisons between the proposed MSMM (used as the control method) and other compared methods with the Holm method [69] as a post hoc test. A level of significance $\alpha = 0.05$ is used in all the statistical tests.

We first calculate the *p*-values of the Friedman test and Iman–Davenport test on testing results of all 11 subjects with different measures. Table 12 presents the *p*-values, which are highlighted in boldface if the null hypothesis of equivalent performance is rejected. As is shown, the existence of significant differences among the performances of all the algorithms is validated.

Then we perform the Holm method to calculate the adjusted *p*-values (APVs) for pairwise comparisons involving the MSMM as the control method. We present he APVs in Table 13 and highlight the APVs less than 0.05 in boldface. As is shown, it can be safely concluded that the proposed MSMM is statistically better than the remaining methods regarding the measures of kappa and recall, with a significant level of 0.05. The null hypotheses of equivalent performance can also be rejected with a significant level of 0.1 for the measures of precision and F_1 score.

Time performance. We compare the average training and testing time on both datasets between different methods. Since all the vector-form classifiers (i.e., MSVM, LDA, KNN and MLP) would cause out of memory problem without the PCA procedure, we only compare our method with BSMM and SMM due to the same size of the input. The comparison of average training and testing time between these three methods is shown in Table 14. It can be observed that our method has longer training time and shorter testing time compared with the other two methods. For MSMM, it has no need to break the multiclass problem into several binary ones and only train a unified classifier. In the training phase, MSMM is required to train a more sophisticated classifier compared with binary classifiers like BSMM and SMM. In the testing phase, our method can be a little faster without considering the bias term compared with BSMM and SMM. The average testing time of our MSMM on these two datasets is 0.357 ms/trial and 0.798 ms/trial, respectively, which is fast enough for most MI-based BCI applications.

6. Conclusion

We present a novel classifier, namely MSMM, for the multiclass classification of EEG data with matrix form. In order to construct the MSMM, we propose a novel objective function by combining the square Frobenius norm of the tensor-form model parameter and nuclear norm of matrix-form hyperplanes extracted from the model parameter, and develop an efficient solver based on ADMM framework to solve the objective function. Compared with existing EEG signal classifiers, the proposed MSMM can leverage the structural information encoded in EEG matrices for more accurate multiclass classification, and hence improve the performance of BCI systems with multiple tasks. To our knowledge, we are not aware of any previous classifier that can support multiclass classification for EEG data in matrix form. We extensively evaluate the proposed MSMM on two benchmark multiclass EEG datasets. The MSMM has yielded an average kappa value of 0.880 and 0.648 for dataset IIIa of BCI III and dataset IIa of BCI IV, respectively, and achieved the

best results when compared with state-of-the-art or widely used classifiers for EEG data. Although the proposed method is applied to MI-based EEG data, it is general enough to be used in other BCI systems involving multiclass matrix-form EEG signals. Further investigations include assessing our method on more single trial EEG datasets or advanced cross-subject EEG features, and integrating it in BCI systems involving multiple tasks.

Acknowledgments

This work is supported by the National Basic Program of China, 973 Program (Project No. 2015CB351706), the National Natural Science Foundation of China (Project No. 61233012) and a grant form the Research Grants Council of the Hong Kong Special Administrative Region (Project No. CUHK 14225616).

Appendix A

A1. Proof

(1) Proof of Theorem 1. The constraints in Eq. (4) are independent from each other. For each constraint, the slack variable ξ_i that minimizes Eq. (4) should satisfy

$$\xi_i = \max_{\hat{y}_i \in \mathcal{Y}} [\Delta(\hat{y}_i, y_i) + \langle \mathcal{W}, \delta\Psi(\mathbf{X}_i, \hat{y}_i, y_i)\rangle]. \tag{A.1}$$

For optimization problem in Eq. (6), the smallest variable ξ is

$$\xi = \max_{(\hat{y}_1,\ldots,\hat{y}_n)\in\mathcal{Y}^n} [\frac{1}{n}\sum_{i=1}^{n}\Delta(\hat{y}_i, y_i) + \frac{1}{n}\sum_{i=1}^{n}\langle \mathcal{W}, \delta\Psi(\mathbf{X}_i, \hat{y}_i, y_i)\rangle]. \tag{A.2}$$

For a given $\mathcal{W}$, each estimated label in a constraint $(\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n)$ is individually independent, ξ can be decomposed linearly in each $\hat{y}_i$, $(i \in 1, \ldots, n)$ with

$$\begin{aligned}\xi &= \sum_{i=1}^{n}\max_{\hat{y}_i\in\mathcal{Y}}[\frac{1}{n}\Delta(\hat{y}_i, y_i) + \frac{1}{n}\langle \mathcal{W}, \delta\Psi(\mathbf{X}_i, \hat{y}_i, y_i)\rangle] \\ &= \frac{1}{n}\sum_{i=1}^{n}\max_{\hat{y}_i\in\mathcal{Y}}[\Delta(\hat{y}_i, y_i) + \langle \mathcal{W}, \delta\Psi(\mathbf{X}_i, \hat{y}_i, y_i)\rangle] \\ &= \frac{1}{n}\sum_{i=1}^{n}\xi_i. \end{aligned} \tag{A.3}$$

Since both optimization problems have the same regularization, their objective values are equal given any $\mathcal{W}$. This is also applicable for optimal $\mathcal{W}^*$ and the corresponding ξ and ξ_i. □

(2) Proof of Theorem 2. Since the optimal solution $\mathbf{S}_c^*$ of Eq. (12) satisfies $\mathbf{0} \in \partial L_S(\mathbf{S}_c^*)$, we only need to find one $\hat{\mathbf{S}}_c$ subject to

$$\mathbf{0} \in \hat{\mathbf{S}}_c + \tau\partial||\hat{\mathbf{S}}_c||_* + \mathbf{\Lambda} + \rho(\hat{\mathbf{S}}_c - \mathbf{W}_c). \tag{A.4}$$

Let $\mathbf{U}_c\mathbf{\Sigma}_c\mathbf{V}_c^T$ denote the singular value decomposition of an arbitrary matrix $\mathbf{S}_c$. It is known in [70,71] that the sub-gradient set of the nuclear norm $\partial||\hat{\mathbf{S}}_c||_*$ is

$$\begin{aligned}\partial||\hat{\mathbf{S}}_c||_* &= \{\mathbf{U}_c\mathbf{V}_c^T + \mathbf{Z} : \mathbf{Z} \in \mathbb{R}^{I_1\times I_2}, \mathbf{U}_c^T\mathbf{Z} \\ &= \mathbf{0}, \mathbf{Z}\mathbf{V}_c = \mathbf{0}, ||\mathbf{Z}||_F < 1\}. \end{aligned} \tag{A.5}$$

Let $\mathbf{Y}$ denote $(\rho\mathbf{W}_c - \mathbf{\Lambda}_c)$ and decompose it as $\mathbf{Y} = \mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^T + \mathbf{U}_2\mathbf{\Sigma}_2\mathbf{V}_2^T$, where $\mathbf{U}_1$ and $\mathbf{V}_1$ ($\mathbf{U}_2$ and $\mathbf{V}_2$) are the singular vectors associated with singular values greater than τ (smaller than or equal to τ). If $\hat{\mathbf{S}}_c = \frac{\mathbf{U}_1(\mathbf{\Sigma}_1 - \tau\mathbf{I})\mathbf{V}_1^T}{1+\rho}$, according to Eq. (A.4), we have

$$\begin{aligned}\partial||\hat{\mathbf{S}}_c||_* &= \frac{1}{\tau}[\mathbf{Y} - (1+\rho)\hat{\mathbf{S}}_c] \\ &= \mathbf{U}_1\mathbf{V}_1^T + \frac{1}{\tau}\mathbf{U}_2\mathbf{\Sigma}_2\mathbf{V}_2 \end{aligned} \tag{A.6}$$

Let $\mathbf{Z} = \frac{1}{\tau}\mathbf{U}_2\mathbf{\Sigma}_2\mathbf{V}_2$, $\mathbf{U}_c = \mathbf{U}_1$ and $\mathbf{V}_c = \mathbf{V}_1$, we have $\mathbf{0} \in \partial L_S$ when $\mathbf{S}_c^* = \hat{\mathbf{S}}_c$. □

(3) Proof of Theorem 3. The most violated constraint resulting in the largest ξ is

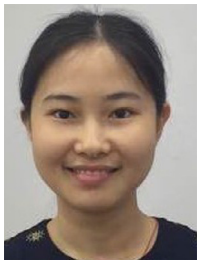
$$\begin{aligned}\xi^* = \max_{\{\hat{y}_1\ldots\hat{y}_n\}\in\mathcal{Y}^n} &\frac{C}{n}\{\sum_{i=1}^{n}\Delta(\hat{y}_i, y_i) \\ &+\langle \mathcal{W}, \delta\Psi(\mathbf{X}_i, \hat{y}_i, y_i)\rangle\} \end{aligned} \tag{A.7}$$

We have proved in Theorem 1 that the original problem satisfies $\xi^* = \frac{1}{n}\sum_{i=1}^{n}\xi_i^*$. Therefore, the most violated constraint $\hat{y}_i$ can be calculated as in Eq. (17). □

References

[1] H. Yuan, B. He, Brain–computer interfaces using sensorimotor rhythms: current state and future perspectives, IEEE Trans. Biomed. Eng. 61 (5) (2014) 1425–1435.
[2] J.J. Shih, D.J. Krusienski, J.R. Wolpaw, Brain-computer interfaces in medicine, Mayo Clin. Proc. 87 (3) (2012) 268–279.
[3] K.K. Ang, K.S.G. Chua, K.S. Phua, C. Wang, Z.Y. Chin, C.W.K. Kuah, W. Low, C. Guan, A randomized controlled trial of EEG-based motor imagery brain–computer interface robotic rehabilitation for stroke, Clin. EEG Neurosci. 46 (4) (2015) 310–320.
[4] S.B. i Badia, A.G. Morgade, H. Samaha, P. Verschure, Using a hybrid brain computer interface and virtual reality system to monitor and promote cortical reorganization through motor activity and motor imagery training, IEEE Trans. Neural Syst. Rehabilit. Eng. 21 (2) (2013) 174–181.
[5] F. Lotte, J. Faller, C. Guger, Y. Renard, G. Pfurtscheller, A. Lécuyer, R. Leeb, Combining BCI with virtual reality: towards new applications and improved BCI, in: Towards Practical Brain-Computer Interfaces, Springer, 2012, pp. 197–220.
[6] L.-D. Liao, C.-Y. Chen, I.-J. Wang, S.-F. Chen, S.-Y. Li, B.-W. Chen, J.-Y. Chang, C.-T. Lin, Gaming control using a wearable and wireless EEG-based brain-computer interface device with novel dry foam-based sensors, J. Neuroeng. Rehabilit. 9 (1) (2012) 1.
[7] R. Leeb, M. Lancelle, V. Kaiser, D.W. Fellner, G. Pfurtscheller, Thinking penguin: multimodal brain–computer interface control of a VR game, IEEE Trans. Comput. Intell. AI Games 5 (2) (2013) 117–128.
[8] M.I. Al-Kadi, M.B.I. Reaz, M.A.M. Ali, Evolution of electroencephalogram signal analysis techniques during anesthesia, Sensors 13 (5) (2013) 6605–6635.
[9] H.-I. Suk, S.-W. Lee, A novel bayesian framework for discriminative feature extraction in brain-computer interfaces, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2) (2013) 286–299.
[10] N. Tomida, T. Tanaka, S. Ono, M. Yamagishi, H. Higashi, Active data selection for motor imagery eeg classification, IEEE Trans. Biomed. Eng. 62 (2) (2015) 458–467.
[11] R. Ameri, A. Pouyan, V. Abolghasemi, Projective dictionary pair learning for EEG signal classification in brain computer interface applications, Neurocomputing 218 (2016) 382–389.
[12] Z. Ma, Z.-H. Tan, J. Guo, Feature selection for neutral vector in eeg signal classification, Neurocomputing 174 (2016) 937–945.
[13] F. Qi, Y. Li, W. Wu, RSTFC: a novel algorithm for spatio-temporal filtering and classification of single-trial EEG, IEEE Trans. Neural Netw. Learn. Syst. 26 (12) (2015) 3070–3082.
[14] Ö.F. AlçÑûn, S. Siuly, V. Bajaj, Y. Guo, A. Şengu, Y. Zhang, et al., Multi-category EEG signal classification developing time-frequency texture features based fisher vector encoding method, Neurocomputing 218 (2016) 251–258.
[15] G. Dornhege, B. Blankertz, G. Curio, K.-R. Muller, Boosting bit rates in non-invasive eeg single-trial classifications by feature combination and multiclass paradigms, IEEE Trans. Biomed. Eng. 51 (6) (2004) 993–1002.
[16] L.F. Nicolas-Alonso, R. Corralejo, J. Gomez-Pilar, D. Álvarez, R. Hornero, Adaptive stacked generalization for multiclass motor imagery-based brain computer interfaces, IEEE Trans. Neural Syst. Rehabilit. Eng. 23 (4) (2015) 702–712.
[17] A. Jordan, On discriminative vs. generative classifiers: a comparison of logistic regression and naive bayes, Adv. Neural Inf. Process. Syst. 14 (2002) 841.
[18] A.S. Aghaei, M.S. Mahanta, K.N. Plataniotis, Separable common spatio-spectral patterns for motor imagery BCI systems, IEEE Trans. Biomed. Eng. 63 (1) (2016) 15–29.
[19] Y. Liu, Q. Zhao, L. Zhang, Uncorrelated multiway discriminant analysis for motor imagery EEG classification, Int. J. Neural Syst. 25 (04) (2015) 1550013.
[20] J.A. Suykens, J. Vandewalle, Least squares support vector machine classifiers, Neural Process. Lett. 9 (3) (1999) 293–300.
[21] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, K.-R. Mullers, Neural Networks for Signal Processing IX, 1999, Proceedings of the 1999 IEEE Signal Processing Society Workshop, 1999, pp. 41–48.
[22] M. Hamedi, S.-H. Salleh, A.M. Noor, Electroencephalographic motor imagery brain connectivity analysis for BCI: a review, Neural Comput. 28 (6) (2016) 999–1041.
[23] H. Zhou, L. Li, Regularized matrix regression, J. R. Stat. Soc. Ser. B (Stat. Methodol.) 76 (2) (2014) 463–483.

[24] C. Brunner, R. Leeb, G. Müller-Putz, A. Schlögl, G. Pfurtscheller, BCI Competition 2008–Graz Data Set A, Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces), Graz University of Technology, 2008, pp. 136–142.
[25] K.K. Ang, Z.Y. Chin, C. Wang, C. Guan, H. Zhang, Filter bank common spatial pattern algorithm on BCI competition IV datasets 2a and 2b, Front. Neurosci. 6 (2012) 39.
[26] S. Lemm, B. Blankertz, G. Curio, K.-R. Muller, Spatio-spectral filters for improving the classification of single trial EEG, IEEE Trans. Biomed. Eng. 52 (9) (2005) 1541–1548.
[27] R. Tomioka, G. Dornhege, G. Nolte, B. Blankertz, K. Aihara, K.-R. Müller, Spectrally Weighted Common Spatial Pattern Algorithm for Single Trial EEG Classification, Technical Report 40, Department of Mathematical Engineering, University of Tokyo, Tokyo, Japan, 2006.
[28] G. Dornhege, B. Blankertz, M. Krauledat, F. Losch, G. Curio, K.-R. Muller, Combined optimization of spatial and temporal filters for improving brain-computer interfacing, IEEE Trans. Biomed. Eng. 53 (11) (2006) 2274–2281.
[29] T.-E. Kam, H.-I. Suk, S.-W. Lee, Non-homogeneous spatial filter optimization for electroencephalogram (EEG)-based motor imagery classification, Neurocomputing 108 (2013) 58–68.
[30] H. Zhang, H. Yang, C. Guan, Bayesian learning for spatial filtering in an EEG-based brain–computer interface, IEEE Trans. Neural Netw. Learn. Syst. 24 (7) (2013) 1049–1060.
[31] L. Wolf, H. Jhuang, T. Hazan, Modeling appearances with low-rank SVM, in: Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2007, pp. 1–6.
[32] M. Dyrholm, C. Christoforou, L.C. Parra, Bilinear discriminant component analysis, J. Mach. Learn. Res. 8 (May) (2007) 1097–1111.
[33] H. Pirsiavash, D. Ramanan, C.C. Fowlkes, Bilinear classifiers for visual recognition, in: Proceedings of the 22nd International Conference on Neural Information Processing Systems, 2009, pp. 1482–1490.
[34] T. Kobayashi, N. Otsu, Efficient optimization for low-rank integrated bilinear classifiers, in: Computer Vision–ECCV 2012, Springer, 2012, pp. 474–487.
[35] L. Luo, Y. Xie, Z. Zhang, W.-J. Li, Support matrix machines, in: Proceedings of the International Conference on Machine Learning (ICML), 2015.
[36] C.-W. Hsu, C.-J. Lin, A comparison of methods for multiclass support vector machines, IEEE Trans. Neural Netw. 13 (2) (2002) 415–425.
[37] N. Robinson, C. Guan, A. Vinod, K.K. Ang, K.P. Tee, Multi-class EEG classification of voluntary hand movement directions, J. Neural Eng. 10 (5) (2013) 056018.
[38] C.M. Bishop, Pattern recognition and machine learning, springer, 2006.
[39] X. He, Z. Wang, C. Jin, Y. Zheng, X. Xue, A simplified multi-class support vector machine with reduced dual optimization, Pattern Recognit. Lett. 33 (1) (2012) 71–82.
[40] T. Joachims, T. Finley, C.-N. J. Yu, Cutting-plane training of structural SVMS, Mach. Learn. 77 (1) (2009) 27–59.
[41] J. Bien, J. Taylor, R. Tibshirani, A lasso for hierarchical interactions, Ann. Stat. 41 (3) (2013) 1111.
[42] B. O'Donoghue, G. Stathopoulos, S. Boyd, A splitting method for optimal control, IEEE Trans. Control Syst. Technol. 21 (6) (2013) 2432–2442.
[43] A. Schlögl, F. Lee, H. Bischof, G. Pfurtscheller, Characterization of four-class motor imagery EEG data for the BCI-competition 2005, J. Neural Eng. 2 (4) (2005) L14.
[44] C.J. Burges, A tutorial on support vector machines for pattern recognition, Data Min. Knowl. Discov. 2 (2) (1998) 121–167.
[45] E. Haselsteiner, G. Pfurtscheller, Using time-dependent neural networks for EEG classification, IEEE Trans. Rehabilit. Eng. 8 (4) (2000) 457–463.
[46] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, B. Arnaldi, A review of classification algorithms for EEG-based brain–computer interfaces, J. Neural Eng. 4 (2) (2007) R1.
[47] K.P. Thomas, C. Guan, C.T. Lau, A.P. Vinod, K.K. Ang, A new discriminative common spatial pattern method for motor imagery brain–computer interfaces, IEEE Trans. Biomed. Eng. 56 (11) (2009) 2730–2733.
[48] S. Sun, C. Zhang, Adaptive feature extraction for eeg signal classification, Med. Biol. Eng. Comput. 44 (10) (2006) 931–935.
[49] H. Wang, X. Li, Regularized filters for l1-norm-based common spatial patterns, IEEE Trans. Neural Syst. Rehabilit. Eng. 24 (2) (2016) 201–211.
[50] M. Arvaneh, C. Guan, K.K. Ang, C. Quek, Optimizing spatial filters by minimizing within-class dissimilarities in electroencephalogram-based brain–computer interface, IEEE Trans. Neural Netw. Learn. Syst. 24 (4) (2013) 610–619.
[51] Y. Li, P.P. Wen, et al., Clustering technique-based least square support vector machine for eeg signal classification, Comput. Methods Progr. Biomed. 104 (3) (2011) 358–372.
[52] R. Tomioka, K. Aihara, Classifying matrices with a spectral regularization, in: Proceedings of the 24th International Conference on Machine learning, ACM, 2007, pp. 895–902.
[53] R. Tomioka, K. Aihara, K.-R. Müller, Logistic regression for single trial EEG classification, Adv. Neural Inf. Process. Syst. 19 (2007) 1377.
[54] C. Christoforou, R. Haralick, P. Sajda, L.C. Parra, Second-order bilinear discriminant analysis, J. Mach. Learn. Res. 11 (Feb) (2010) 665–685.
[55] H. Zeng, A. Song, Optimizing single-trial EEG classification by stationary matrix logistic regression in brain–computer interface, IEEE Trans. Neural Netw. Learn. Syst. 27 (11) (2016) 2301–2313.
[56] I. Tsochantaridis, T. Joachims, T. Hofmann, Y. Altun, Large margin methods for structured and interdependent output variables, J. Mach. Learn. Res. 6 (Sep) (2005) 1453–1484.
[57] K. Crammer, Y. Singer, On the algorithmic implementation of multiclass kernel-based vector machines, J. Mach. Learn. Res. 2 (Dec) (2001) 265–292.
[58] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.
[59] S. Shalev-Shwartz, Y. Singer, N. Srebro, A. Cotter, Pegasos: primal estimated sub-gradient solver for SVM, Math. Program. 127 (1) (2011) 3–30.
[60] S. Bhattacharyya, A. Khasnobish, S. Chatterjee, A. Konar, D. Tibarewala, Performance analysis of LDA, QDA and KNN algorithms in left-right limb movement classification from EEG data, in: Proceedings of the 2010 International Conference on Systems in Medicine and Biology (ICSMB), IEEE, 2010, pp. 126–131.
[61] D. Balakrishnan, S. Puthusserypady, Multilayer perceptrons for the classification of brain computer interface data, in: Proceedings of the IEEE 31st Annual Northeast Bioengineering Conference, 2005, IEEE, 2005, pp. 118–119.
[62] G. Pfurtscheller, C. Neuper, D. Flotzinger, M. Pregenzer, EEG-based discrimination between imagination of right and left hand movement, Electroencephalogr. Clin. Neurophysiol. 103 (6) (1997) 642–651.
[63] F. Rieke, Spikes: exploring the neural code, MIT press, 1999.
[64] C. Vidaurre, N. Krämer, B. Blankertz, A. Schlögl, Time domain parameters as a feature for EEG-based brain–computer interfaces, Neural Netw. 22 (9) (2009) 1313–1319.
[65] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, Inf. Process. Manag. 45 (4) (2009) 427–437.
[66] V.K. Manchala, Human Computer Interface Using Electroencephalography, Arizona State University, 2015 (Ph.D. thesis).
[67] A. Schlögl, Results of the BCI-Competition 2005 for Datasets IIIa and IIIb, Technical Report, Institute of Human-Computer Interface, Graz University of Technology, 2005.
[68] S. García, A. Fernández, J. Luengo, F. Herrera, Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: experimental analysis of power, Inf. Sci. 180 (10) (2010) 2044–2064.
[69] Y. Wen, H. Xu, J. Yang, A heuristic-based hybrid genetic-variable neighborhood search algorithm for task scheduling in heterogeneous multiprocessor system, Inf. Sci. 181 (3) (2011) 567–581.
[70] J.-F. Cai, E.J. Candès, Z. Shen, A singular value thresholding algorithm for matrix completion, SIAM J. Optim. 20 (4) (2010) 1956–1982.
[71] E.J. Candès, B. Recht , Exact matrix completion via convex optimization, Found. Comput. Math. 9 (6) (2009) 717–772.



Qingqing Zheng is currently a Ph.D. student in the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong. Her research interests include machine learning, computer vision and brain computer interfaces.

Fengyuan Zhu received the Ph.D. degree in the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong. His research interests include machine learning theory, computer vision, data mining.

Jing Qin is currently an assistant professor in School of Nursing, The Hong Kong Polytechnic University. His research interests include virtual/augmented reality for healthcare and medicine training, medical image processing, deep learning, visualization and human-computer interaction and health informatics.

Pheng-Ann Heng is currently a professor in the Department of Computer Science and Engineering, CUHK. He is also the director of the Research Center for Human-Computer Interaction, SIAT, Chinese Academy of Sciences. His research interests include virtual reality, visualization, medical imaging, human-computer interfaces, rendering and modeling, interactive graphics, and animation.