Sleep Staging Using Plausibility Score: A Novel Feature Selection Method Based on Metric Learning

Tao Zhang, Zhonghui Jiang, Dan Li, Xiao Wei, Bing Guo, Wu Huang, and Guobiao Xu

***Abstract*—As an effective method, feature selection can reduce computational complexity and improve classification performance. A number of criteria exist for feature selection using labeled data, unlabeled data and pairwise constraints, most of which are based on the Euclidean distance. In this paper, we propose a filter method for feature selection with pairwise constraints, aiming to jointly evaluate a feature subset based on metric learning. Two criteria are designed based on the well-known Kullback-Leibler divergence for measuring the difference between must-link constraints and cannot-link constraints that can indicate the feature subset discrimination based on Keep It Simple and Straightforward (KISS) metric learning and Cross-view Quadratic Discriminant Analysis (XQDA) metric learning. To address the challenging feature selection problem, we formulate a sequential search algorithm guided by indicators that are simplified from the proposed criteria. Furthermore, we conducted several experiments on sleep staging based on electroencephalogram (EEG) recordings from the Sleep-EDF Database Expanded. The experimental results demonstrate the effectiveness of the proposed method compared with nine representative feature selection methods. On the data set from healthy volunteers and the data set from volunteers that had mild difficulty falling asleep, the classification average accuracies achieve 97.66% and 93.57% by using the proposed method, respectively.**

***Index Terms*—Euclidean metric, feature selection, KISS metric learning, plausibility scores, sleep staging, XQDA metric learning.**

I. Introduction

IN CLASSIFICATION, it is generally known that a tremendous number of features would result in the curse of dimensionality and unexplainability [1]. As an effective method, feature selection aims to select a useful subset of original features to reduce dimensionality and improve classification performance [2], [3]. Typically, feature selection methods can be divided into three categories: filter methods [4]–[8], wrapper methods [9] and embedded methods [10]. The results of filter methods depend only on the characteristics of the data. In contrast, wrapper methods require the participation of classification algorithms. Embedded methods combine the former two methods. Normally, wrapper methods are more accurate than filter methods, but their computational costs are often more expensive than those of filter methods.

In supervised filter methods, label information and pairwise constraints have been explored and applied for feature selection [1], [3]–[8]. Pairwise constraints as a type of side information specify whether a pair of instances belongs to the same class or different classes. In many cases, obtaining pairwise constraints is easier than obtaining class labels. Recently, pairwise constraints as another alternative have been emphasized for feature selection in [6], which showed that a constraint score based on less information can be similar to or even better than a Fisher Score based on full information. In addition to focusing on supervised information, some studies focus on two strategies that have been frequently employed to evaluate a subset of features. The first strategy is evaluating every feature independently, e.g., the Fisher score [1], ReliefF [5] and constraint score [6]. In this case, all candidate features are ranked, and the features with high ranking scores are selected. However, this strategy neglects the combination effects and redundancy of features, thereby preventing further optimization. To solve this problem, the second strategy proposes to integrally evaluate a feature subset, e.g., the generalized Fisher score [7] and the trace ratio criterion [4]. The disadvantage of the second strategy is that a more complicated and expensive algorithm is often required for feature selection.

In addition to feature selection, how to define a distance over inputs would obviously impact the performance of some classifiers, such as nearest-neighbor classifiers and support vector machines. Thus, distance metric learning was introduced aiming to make additional improvements in classification and clustering [11]. In recent decades, some state-of-the-art Mahalanobis metric learning algorithms have been proposed, such as Neighbourhood Components Analysis (NCA) [12], Large Margin Nearest Neighbor (LMNN) classification [13], Information-Theoretic Metric Learning (ITML) [14], Keep It Simple and

Manuscript received September 25, 2019; revised March 6, 2020, April 16, 2020, and May 4, 2020; accepted May 5, 2020. Date of publication May 11, 2020; date of current version February 4, 2021. This research is supported by the 13th Five-Year Plan for National Education Science in 2017 Grant DLA170428. *(Corresponding author: Wu Huang.)*

Tao Zhang, Zhonghui Jiang, Dan Li, and Xiao Wei are with the Chengdu Techman Software Co., Ltd., Chengdu, Sichuan, China (e-mail: ztuestc@outlook.com; zhonghui.jiang@126.com; ld120608@126.com; weixiao890623@163.com).

Bing Guo and Wu Huang are with the Sichuan University, Chengdu, Sichuan, China (e-mail: guobing@scu.edu.cn; huangwu@scu.edu.cn).

Guobiao Xu is with the Civil Aviation Flight University of China, Sichuan, China (e-mail: mis03@126.com).

Digital Object Identifier 10.1109/JBHI.2020.2993644

Straightforward (KISS) metric learning [15] and Cross-view Quadratic Discriminant Analysis (XQDA) [16]. Goldberger *et al.* proposed NCA by using the k-nearest neighbors (KNN) classification algorithm [12]. In this work, the Mahalanobis metric is obtained by gradually optimizing the leave-one-out performance of data training. Weinberger *et al.* proposed LMNN to learn a good Mahanalobis distance metric in KNN classification [13]. By using LMNN, the distances between similarly labeled instances decrease, and those between differently labeled instances grow to a certain extent. Davis *et al.* formulated the ITML and showed that it closely relates to the Kullback-Leibler divergence between two multivariate Gaussians [14]. To avoid over-fitting, ITML learns a Mahalanobis distance metric as close as possible to the Euclidean metric under the given constraint conditions. For a large-scale data set, the computational costs of some metric learning algorithms, e.g., the LMNN algorithm, are very expensive. Kostinger *et al.* proposed the KISS metric learning, which can efficiently obtain a Mahalanobis distance over large-scale inputs without tedious iteration [15]. This method is realized based on a likelihood-ratio test. Liao *et al.* extended the KISS metric learning method and proposed XQDA [16]. By using XQDA, the dimension of the feature subspace can be reduced.

The method of applying a metric learning algorithm in multi-class classification can be global or class-dependent [17]. The class-dependent methods learn different distance metrics for different classes, and the global method learns a single distance metric over all classes. In addition, the weight metrics can be full or sparse [17]–[19]. Full weight matrices are the most flexible models for learning, but their computational cost may be expensive. To solve this problem, the weight matrices were considered sparse or even diagonal. Nevertheless, some sparse weight matrices are competitive in some cases, e.g., the class-dependent diagonal weight matrices for time series classification [17]. The disadvantage of diagonal weight matrices is that the combination effects and redundancy of features are neglected. To address the problem, some sparse weight matrices with more parameters, e.g., the block diagonal weight matrices [18] and the shrinkage weight matrices [17], were proposed.

Classification of sleep stages is important in sleep studies and disorder diagnosis. In addition to the conventional discrimination of wakefulness and sleep, the Rechtschaffen and Kales standard (R&K) [20] enables the sleep stages of humans to be recognized as six different types: wakefulness, NREM stages (S1, S2, S3 and S4) and REM stages. At present, some machine learning algorithms are used for automatic sleep staging by using a set of features extracted from electroencephalogram (EEG), electrooculogram (EOG) and/or electromyogram (EMG) signals, which include the time-domain features, the frequency-domain features and the non-linear features [21]–[27]. To reduce the dimension of high-dimensional data, a variety of feature selection algorithms have been developed based on these biomedical signals [25]–[30]. On the other hand, Phan *et al.* noted that the Euclidean distance would not be the best distance metric over inputs for sleep staging and introduced the LMNN algorithm into the classification task [31]. Therefore, both metric learning and feature selection have been regarded as effective methods to improve sleep staging performance. However, a feature selection method may not match a learned Mahalanobis distance metric if they are used simultaneously, because the feature selection method is probably developed based on the Euclidean distance. The mismatch in some cases may reduce the effectiveness of feature selection.

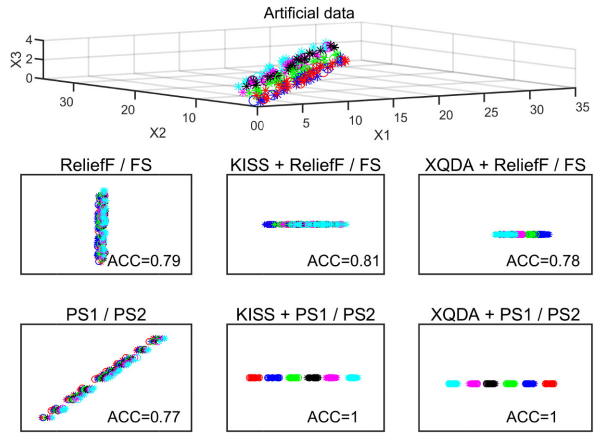


Fig. 1. Artificial data with features X1, X2, and X3. The 6-class data are divided into training data (asterisk) and test data (circle). Based on the training data, X2 and X3 are selected by ReliefF [5] and Fisher score (FS) [1], and X1 and X2 are selected by our methods PS1 or the PS2. The Accuracy (ACC) in each case is obtained by using the 1-NN classifier. Our methods are much better than ReliefF and FS based on KISS metric or XQDA metric.

To address this problem, in this work, we focus on how to select an optimal feature subset based on distance metric learning for sleep staging. We propose a filter method called the Plausibility Score for feature selection, using pairwise constraints rather than more expensive label information. Two feature selection criteria will be designed based on a well-known conception, the Kullback-Leibler (KL) divergence, for evaluating an information divergence of occurrence between a pair of samples belonging to the same class (must-link constraint) and a pair of samples belonging to different classes (cannot-link constraint). This information divergence indicates the feature subset discrimination based on an appropriate Mahalanobis distances learned by either of two comparatively low computational cost algorithms, the KISS and the XQDA metric learning algorithms [15], [16]. Subsequently, we will adopt a novel sequential search strategy to solve the complicated feature selection problems. The goal of the proposed method is recognizing the optimal feature set based on KISS and XQDA metrics that would be difficult to achieve by using other methods, illustrated in Fig. 1. In previous work, KISS and XQDA metric learning algorithms were mainly used in the person re-identification field rather than in the health care field. In these experiments, we plan to investigate the feature selection problem based on these metric learning algorithms for sleep staging. The effectiveness of our methods is validated by the experimental results.

The rest of the content is organized as follows. We derive the feature selection criteria based on the KL divergence and propose the feature selection algorithm in Section 2. We conduct

the experiments on sleep staging in Section 3 and discuss the experimental results in Section 4. Finally, we conclude the work in Section 5.

II. Proposed Plausibility Score

In this section, we first revisit KISS and XQDA metric learning. Afterwards, we design two feature selection criteria based on the KL divergence and formulate a feature selection algorithm based on the proposed criteria.

A. KISS and XQDA Metric Learning

Kostinger *et al.* proposed the KISS metric learning for large-scale metric learning [15]. This method came from a view of statistical inference that whether an instance pair is similar can be determined by the likelihood-ratio test [15]

$$L\left(\mathbf{\Delta}_{ij}\right)=\log\left(\frac{P\left(\mathbf{\Delta}_{ij}\left|D\right.\right)}{P\left(\mathbf{\Delta}_{ij}\left|U\right.\right)}\right). \tag{1}$$

where

$$P\left(\mathbf{\Delta}_{ij}\left|D\right.\right)=\frac{1}{\left(2\pi\right)^{n/2}\left|\Sigma_D\right|^{1/2}}\exp\left(-\frac{1}{2}\mathbf{\Delta}_{ij}^T\Sigma_D^{-1}\mathbf{\Delta}_{ij}\right), \tag{2}$$

$$P\left(\mathbf{\Delta}_{ij}\left|U\right.\right)=\frac{1}{\left(2\pi\right)^{n/2}\left|\Sigma_U\right|^{1/2}}\exp\left(-\frac{1}{2}\mathbf{\Delta}_{ij}^T\Sigma_U^{-1}\mathbf{\Delta}_{ij}\right). \tag{3}$$

In Eq. 1, $\mathbf{\Delta}_{ij}=\mathbf{x}_i-\mathbf{x}_j$ where $\mathbf{x}_i,\mathbf{x}_j\in\mathbb{R}^{d\times1}$ correspond to two arbitrary instances i and j, respectively, and $\mathbf{\Delta}_{ij}\in U$ and $\mathbf{\Delta}_{ij}\in D$ denote the instances i and j sharing the same and different labels, respectively. In Eqs. 2 and 3, $\Sigma_D=\sum_{\mathbf{\Delta}_{ij}\in D}(\mathbf{x}_i-\mathbf{x}_j)(\mathbf{x}_i-\mathbf{x}_j)^T$ and $\Sigma_U=\sum_{\mathbf{\Delta}_{ij}\in U}(\mathbf{x}_i-\mathbf{x}_j)(\mathbf{x}_i-\mathbf{x}_j)^T$ are the covariance matrics in the likelihoods $P(\mathbf{\Delta}_{ij}|D)$ and $P(\mathbf{\Delta}_{ij}|U)$, respectively, which are proven to have zero means. Therefore, Eq. 1 can be simplified to

$$L\left(\mathbf{\Delta}_{ij}\right)=\frac{1}{2}\left[\mathbf{\Delta}_{ij}^T\left(\Sigma_U^{-1}-\Sigma_D^{-1}\right)\mathbf{\Delta}_{ij}+\log\left(\frac{\left|\Sigma_U\right|}{\left|\Sigma_D\right|}\right)\right], \tag{4}$$

where the second term in the right part is a constant. Suppose that M is obtained by re-projecting $\Sigma_U^{-1}-\Sigma_D^{-1}$ onto the cone of positive semi-definite matrices; the distance based on the KISS metric is obtained from the first term in the right part of Eq. 4,

$$d_{M,KISS}^2\left(\mathbf{x}_i,\mathbf{x}_j\right)=\mathbf{\Delta}_{ij}^TM\mathbf{\Delta}_{ij}. \tag{5}$$

Liao *et al.* extended KISS metric learning and proposed the XQDA [16]. Similar to Principal Component Analysis (PCA), the XQDA strategy searches for some useful components in initial feature spaces. XQDA is supervised, however, and the supervised information includes $\mathbf{\Delta}_{ij}\in U$ and $\mathbf{\Delta}_{ij}\in D$. The difference between U and D projecting to a one-dimensional space was assessed by the variance ratio of two Gaussian distributions and used as a criterion to evaluate the components. The component selection problem was introduced as follows:

$$\mathbf{w}^1=\arg\max_{\mathbf{w}}\mathbf{w}^T\Sigma_D\mathbf{w},\mathrm{s.t.}\mathbf{w}^T\Sigma_U\mathbf{w}=1. \tag{6}$$

In Eq. 6, the optimal solution $\mathbf{w}^1$ is the eigenvector of matrix $\Sigma_U^{-1}\Sigma_D$ corresponding to its maximum eigenvalue. The solution $\mathbf{w}^2$ orthogonal to $\mathbf{w}^1$ is the eigenvector corresponding to the second largest eigenvalue, et cetera. A series of components $W=\{\mathbf{w}^1,\mathbf{w}^2,\mathbf{w}^3,...\}$ are selected as the new subspace. If some eigenvalues are less than 1, their corresponding eigenvectors would not provide useful information for discrimination. Therefore, these eigenvectors are not included in W. Finally, the distance based on the XQDA metric is

$$d_{M,XQDA}^2\left(\mathbf{x}_i,\mathbf{x}_j\right)=\mathbf{\Delta}_{ij}^TW\left(\Sigma_U^{'-1}-\Sigma_D^{'-1}\right)W^T\mathbf{\Delta}_{ij}, \tag{7}$$

where $\Sigma_U^{'}=W^T\Sigma_UW$ and $\Sigma_D^{'}=W^T\Sigma_DW$.

B. Feature Selection Criteria

Based on the KISS metric or the XQDA metric, the discrimination of a feature subset can be indicated by the difference between $P(\mathbf{\Delta}_{ij}|D)$ and $P(\mathbf{\Delta}_{ij}|U)$. Therefore, to find the feature subsets with high discrimination, we can make simple changes to the KL divergence and enable it to measure this difference. Suppose that n-feature subset W is selected from initial d-feature set S. Let $KL_{D,U}^W$ denotes the KL divergence $D_{KL}(P(\mathbf{\Delta}_{ij}|D,W)\|P(\mathbf{\Delta}_{ij}|U,W))$ that

$$\begin{aligned}KL_{D,U}^W&=\int P\left(\mathbf{\Delta}_{ij}|D,W\right)\log\left(\frac{P\left(\mathbf{\Delta}_{ij}\left|D,W\right.\right)}{P\left(\mathbf{\Delta}_{ij}\left|U,W\right.\right)}\right)d\mathbf{\Delta}_{ij}\\&=\mathbb{E}_{\mathbf{\Delta}_{ij}\sim P_{D,W}}\left(\log\left(\frac{P\left(\mathbf{\Delta}_{ij}\left|D,W\right.\right)}{P\left(\mathbf{\Delta}_{ij}\left|U,W\right.\right)}\right)\right),\end{aligned} \tag{8}$$

where $P(\mathbf{\Delta}_{ij}|D,W)$ and $P(\mathbf{\Delta}_{ij}|U,W)$ are the $P(\mathbf{\Delta}_{ij}|D)$ and $P(\mathbf{\Delta}_{ij}|U)$ in Eqs. 2 and 3 under the feature subset W, respectively, and $\mathbb{E}_{\mathbf{\Delta}_{ij}\sim P_{D,W}}(\cdot)$ is the expectation of a random variable obeying the probability distribution with $P(\mathbf{\Delta}_{ij}|D,W)$. Recall that the means of $P(\mathbf{\Delta}_{ij}|D,W)$ and $P(\mathbf{\Delta}_{ij}|U,W)$ are zero; the KL divergence can be simply calculated as

$$\begin{aligned}KL_{D,U}^W&=\frac{1}{2}\left[\log\frac{\left|\Sigma_{U,W}\right|}{\left|\Sigma_{D,W}\right|}+\mathbb{E}_{\mathbf{\Delta}_{ij}\sim P_{D,W}}\left(\mathbf{\Delta}_{ij}^T\Sigma_{U,W}^{-1}\mathbf{\Delta}_{ij}\right.\right.\\&\quad\left.\left.-\mathbf{\Delta}_{ij}^T\Sigma_{D,W}^{-1}\mathbf{\Delta}_{ij}\right)\right]\\&=\frac{1}{2}\left[\log\frac{\left|\Sigma_{U,W}\right|}{\left|\Sigma_{D,W}\right|}+\mathrm{tr}\left(\mathbb{E}_{\mathbf{\Delta}_{ij}\sim P_{D,W}}\left(\mathbf{\Delta}_{ij}^T\Sigma_{U,W}^{-1}\mathbf{\Delta}_{ij}\right)\right)\right.\\&\quad\left.-\mathrm{tr}\left(\mathbb{E}_{\mathbf{\Delta}_{ij}\sim P_{D,W}}\left(\mathbf{\Delta}_{ij}^T\Sigma_{D,W}^{-1}\mathbf{\Delta}_{ij}\right)\right)\right]\\&=\frac{1}{2}\left[\log\frac{\left|\Sigma_{U,W}\right|}{\left|\Sigma_{D,W}\right|}+\mathrm{tr}\left(\Sigma_{U,W}^{-1}\Sigma_{D,W}\right)\right.\\&\quad\left.-\mathrm{tr}\left(\Sigma_{D,W}^{-1}\Sigma_{D,W}\right)\right]\\&=\frac{1}{2}\left[\mathrm{tr}\left(\Sigma_{U,W}^{-1}\Sigma_{D,W}\right)-\log\left|\Sigma_{U,W}^{-1}\Sigma_{D,W}\right|-n\right].\end{aligned} \tag{9}$$

where $\Sigma_{D,W}$ and $\Sigma_{U,W}$ are the Σ_D and Σ_U with the feature set W, respectively. By means of the relationship between the trace and the eigenvalues that $\mathrm{tr}(\Sigma_{U,W}^{-1}\Sigma_{D,W})=\sum_{l=1}^n\lambda_W^l$ and the

relationship between the determinant and the eigenvalues that $\log|\Sigma_{U,W}^{-1}\Sigma_{D,W}| = \log(\prod_{l=1}^{n}\lambda_W^l) = \sum_{l=1}^{n}\log\lambda_W^l$, where all of λ_W^l are the eigenvalues of matrix $\Sigma_{U,W}^{-1}\Sigma_{D,W}$, the KL divergence can be expressed as

$$KL_{D,U}^{W} = \frac{1}{2}\sum_{l=1}^{n}\left(\lambda_W^l - \log\lambda_W^l - 1\right). \tag{10}$$

It is demonstrable that the KL divergence in Eq. 10 increases as λ_W^l increases if each $\lambda_W^l > 1$ and increases as λ_W^l decreases if each $\lambda_W^l < 1$ (here log refers to the natural logarithm). In addition, as every $\lambda_W^l = 1$, the KL divergence is zero, and the probability distributions of $\boldsymbol{\Delta}_{ij} \in U$ and $\boldsymbol{\Delta}_{ij} \in D$ are the same.

Based on the KL divergence, we will design two feature selection criteria according to two aspects. On the one hand, we hope that the plausibility scores increase as each λ_W^l increases, because a large λ_W^l indicates a high degree of discrimination [16]. On the other hand, we consider both cases that the components of feature subspace corresponding to $\lambda_W^l < 1$ contribute and do not contribute to the discrimination. We design the first plausibility score (PS1) as

$$score_1(W) = \frac{1}{2}\sum_{l=1}^{n}\left(\Lambda_W^l - \log\Lambda_W^l - 1\right) \tag{11}$$

where

$$\Lambda_W^l = \begin{cases} \lambda_W^l & \lambda_W^l \geq 1 \\ 1 & \lambda_W^l < 1, \end{cases} \tag{12}$$

and design the second plausibility score (PS2) as

$$score_2(W) = \frac{1}{2}\sum_{l=1}^{n}\lambda_W^l - \log(\lambda_W^l + 1). \tag{13}$$

Equation 11 implies that the $\lambda_W^l < 1$ have no influence on $score_1(W)$, indicating that the components of feature subspace corresponding to them have no contributions for discriminating. In contrast, Eq. 13 implies that the $\lambda_W^l < 1$ have influence on the $score_2(W)$, which increases as each $\lambda_W^l < 1$ increases. Because W with a low value of $score_1(W)$ or $score_2(W)$ would be not useful for discriminating, the feature selection problem is how to select a feature subset $W \subset S$ to maximize $score_1(W)$ or $score_2(W)$ while giving the number of features n.

C. Feature Selection Algorithm

Generally, selecting the optimal subset of features is NP-hard, which remains an open problem. Compared with independently selecting features, designing efficient selection algorithms for jointly selecting features causes more difficulties. In practice, one often settles for second best, i. e., searching a suboptimal solution using heuristic algorithms [7], [32]–[34]. For our problem, it is not easy to maximize PS1 and PS2 by using some methods such as those that require calculating gradient. Instead, we propose a simple and effective local search method to address this problem. In this method, a simplified version of the two plausibility scores is proposed to guide the search order. Specifically, feature subsets with higher simplified PS1 or simplified PS2 are searched preferentially. Finally, the algorithm is terminated by limiting the number of iterations, and the feature set that has the highest PS1 or PS2 among all the searched feature sets is selected. For the two simplified plausibility scores, the features are supposed to be mutually independent, and thus the two covariance matrices $\Sigma_{U,W}$ and $\Sigma_{D,W}$ in the two simplified plausibility scores are diagonal.

Theorem: Suppose that $\Sigma_{U,W}$ and $\Sigma_{D,W}$ are the diagonal matrices, $score_1(W)$ in Eq.11 can be written as

$$SIMscore_1(W) = \frac{1}{2}\sum_{l=1}^{n}\left(\Lambda_W^l - \log\Lambda_W^l - 1\right) \tag{14}$$

where

$$\Lambda_W^l = \begin{cases} \frac{\sigma_W^{l,D}}{\sigma_W^{l,U}} & \frac{\sigma_W^{l,D}}{\sigma_W^{l,U}} \geq 1 \\ 1 & \frac{\sigma_W^{l,D}}{\sigma_W^{l,U}} < 1, \end{cases} \tag{15}$$

and $score_2(W)$ in Eq.13 can be written as

$$SIMscore_2(W) = \frac{1}{2}\sum_{l=1}^{n}\frac{\sigma_W^{l,D}}{\sigma_W^{l,U}} - \log\left(\frac{\sigma_W^{l,D}}{\sigma_W^{l,U}} + 1\right), \tag{16}$$

where $\sigma_W^{l,D}$ and $\sigma_W^{l,U}$ are the variances of $\boldsymbol{\Delta}_{ij} \in D$ and $\boldsymbol{\Delta}_{ij} \in U$ under feature l, respectively.

Proof: Suppose that u_l and d_l are the lth diagonal elements of the diagonal matrices $\Sigma_{U,W}$ and $\Sigma_{D,W}$, respectively. Because $\Sigma_{U,W}$ and $\Sigma_{D,W}$ are the covariance matrices of n features, u_l and d_l are equal to the variances $\sigma_W^{l,U}$ and $\sigma_W^{l,D}$, respectively. In addition, it can be easily proved that $\lambda_W^l = d_l/u_l$ with arbitrary $l \in \{1, 2, ..., n\}$. Therefore, $SIMscore_1(W)$ and $SIMscore_2(W)$ are obtained by replacing λ_W^l with $\sigma_W^{l,D}/\sigma_W^{l,U}$ in $score_1(W)$ and $score_2(W)$, respectively.

As a feature subset W^i is searched, how to search the next one W^{i+1} is equal to solving the problem

$$\begin{aligned} W^{i+1} = \arg\max_{W\subset S} & SIMscore_{1(2)}(W) \\ \text{s.t. } & SIMscore_{1(2)}(W) < SIMscore_{1(2)}(W^i) \\ & |W| = n, \end{aligned} \tag{17}$$

where $SIMscore_{1(2)}(W)$ denotes the $SIMscore_1(W)$ or the $SIMscore_2(W)$,and $|W|$ denotes the cardinal number of W. The first searched subset W^1 is obtained by assembling the features with the n-largest $SIMscore_1$ or $SIMscore_2$ at the single-feature level. Because $SIMscore_{1(2)}(W^i)$ is known, Problem 17 is a typical integer linear programming problem that can be efficiently solved by Cut Generation [36], Branch and Bound [37], Relaxation Induced Neighborhood Search (RINS) [38] and Diving Heuristic [39]. Here, we plan to solve the problem with the Diving Heuristic. The algorithm procedure is formally presented in Algorithm 1.

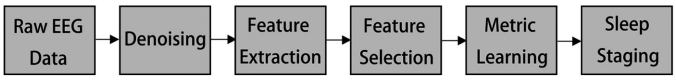


Fig. 2. Overview of sleep staging using Plausibility Score-based feature selection.

Algorithm 1: Feature Selection Based on The Proposed PS1 or PS2.

Input: Feature set S, pairwise constraints in U and D, iteration upper bound I;
Output: Feature subset W;
Calculate $score_1(l)$ ($score_2(l)$) of each single feature l by using Eq.11 (Eq.13), and rank these features in descending order according to their $score_1(l)$ ($score_2(l)$);
Initialize $SCORE = 0$;
Select n features with the largest $score_1(l)$ ($score_2(l)$) as the initial feature subset W^1;
for $i = 1$ to I **do**
  Calculate $score_1(W^i)$ ($score_2(W^i)$) by using Eq.11 (Eq.13);
  if $score_1(W^i)$ ($score_2(W^i)$)$> SCORE$
    $SCORE = score_1(W^i)$ ($score_2(W^i)$);
    $W = W^i$;
  end if
  Solve Problem 17 for obtaining W^{i+1} according to W^i by using the Diving Heuristic [39];
end for

III. Experiments and Results

In this section, we conduct experiments on sleep staging according to the procedure shown in Fig. 2. We first introduce the data set used in our experiments and briefly illustrate the de-noising method. Second, we present all the features that need to be used. Third, we introduce the experimental details and the metric used for testing the effectiveness of the proposed method. Finally, we present the experimental results.

A. Experimental Data and De-Noising

Two experimental data sets are obtained from the Sleep-EDF Database Expanded that is open source for the public [40]–[42]. The first data set (Dataset I) was provided by 15 Caucasian males and females, which were healthy volunteers from 25 to 34 years old. The Dataset I contains PSGs of about 20 hours recorded during day-night period from the SC* files (SC = Sleep Cassette). The second data set (Dataset II) were provided by 15 volunteers from 18 to 48 years old who had mild difficulty falling asleep but were otherwise healthy. The Dataset II contains PSGs of about 9 hours recorded during night from the ST* files (ST = Sleep Telemetry). Compared with using combinations of EEG, EOG and EMG channels for automatic sleep staging, using a single EEG channel is not only feasible but also has more advantages in practical applications [23]. Thus, in these experiments, we plan to use the data from the single EEG channel Fpz-Cz for automatic sleep staging. The sampling frequency of recordings from the Fpz-Cz is 100 Hz. All recordings were divided into 30 second segments, and each was scored according to the R&K standard [20] and labeled as one of the states (AWA, S1, S2, S3, S4, REM, MVT (movement time) and UNS (unknown state)). In these experiments, the segments labeled as AWA, S1, S2, S3, S4 or REM will be used as the labeled instances for classification of sleep stages, and the segments labeled as MVT and UNS are not used in our study. The size of each class is presented in Table I.

TABLE I
Sizes of the Six Classes

Data sets	AWA	S1	S2	S3	S4	REM	Total
Dataset I	28609	976	6849	1139	884	2622	41079
Dataset II	1190	1041	6901	1201	1173	3115	14621

EEG signals are weak and susceptible to external interference, such as EMG interference, powerline interference and white noise. These noises degrade the quality of EEG signals and further exert adverse influence on extracted features. Here, we use the method of wavelet decomposition for de-noising. After re-sampling, the sampling frequency increases to 128 Hz, and the maximum frequency of the EEG signal is 64 Hz according to the Naquist theorem. In addition, the available information contained in the EEG signals is present in the sub-band from 0.5 Hz to 40 Hz. Thus, we set the levels of wavelet decomposition as seven and the coefficients of wavelets mainly contributed by noises (<0.5 Hz and >40 Hz) as zero. We use Daubechies wavelet (db4) as the mother wavelet that is effective for de-noising EEG signals [44]. After that, we further de-noise the EEG signals based on db4, using the soft-thresholding rule and the universal threshold [45].

B. Feature Extraction

In this step, a total of 77 features that have been investigated in previous work were extracted from one single EEG signal [23], [35], [43], [46]–[48]. The feature set includes the time-domain features, the frequency-domain features, the time-frequency features and the non-linear features. All 77 features used in these experiments are listed in Table II, whose computing methods can be found in the corresponding references. Here we simply introduce these features.

As conventional frequency-domain features, the relative spectral powers in the sub-bands of δ (0.5-4 Hz), θ (4-8 Hz), α (8-12 Hz), σ (12-16 Hz), β (16-30 Hz) and γ (30-40 Hz) are extracted via the Fourier transformation of the time-domain signal [43], [46]. The relative spectral power is calculated by dividing the absolute power in each frequency sub-band by the total absolute power in the 0.5-40 Hz frequency range. In addition to the relative power, the ratios of the relative spectral powers can also be explored as effective features [46]. In addition to those in the frequency-domain features, the statistical features in the time domain are commonly used [43]. Furthermore, Bajaj *et al.* proposed three time-frequency features based on the smoothed pseudo Wigner-Ville distribution (SPWVD)-based

TABLE II
SEVENTY-SEVEN FEATURES EXTRACTED FROM THE SINGLE EEG SIGNAL

No.	Feature Name	Reference
1-6	Relative powers P in sub-bands of δ (0.5-4Hz), θ (4-8Hz), α (8-12Hz), σ (12-16Hz), β (16-30Hz) and γ (30-40Hz).	[46]
7-21	The ratios between 1 - 6 features: P_δ/P_θ, P_δ/P_α, P_δ/P_σ, P_δ/P_β, P_δ/P_γ, P_θ/P_α,P_θ/P_σ, P_θ/P_β, P_θ/P_γ, P_α/P_σ, P_α/P_β, P_α/P_γ, P_σ/P_β,P_σ/P_γ and P_β/P_γ.	[46]
22-25	Mean of the absolute values of the wavelet coefficients in D2, D3, D4 and A4.	[47]
26-29	Average powers of the wavelet coefficients in D2, D3, D4 and A4.	[47]
30-33	Standard deviation of the wavelet coefficients in D2, D3, D4 and A4.	[47]
34-37	Ratios of average absolute values of the wavelet coefficients between adjacent sub-bands: D2/D1, D3/D2, D4/D3 and A4/D4.	[47]
38-42	Maximum counts of pixel intensity in the histogram of time-frequency sub-images (0.5-4Hz, 4-8Hz, 8-12Hz, 12-30Hz, 30-50Hz) based on the smoothed pseudo Wigner-Ville distribution (SPWVD).	[48]
43-47	Spreads in the histogram of time-frequency sub-images (0.5-4Hz, 4-8Hz, 8-12Hz, 12-30Hz,30-50Hz) based on SPWVD.	[48]
48-52	Aspect ratios in the histogram of time-frequency sub-images (0.5-4Hz, 4-8Hz, 8-12Hz, 12-30Hz,30-50Hz) based on SPWVD.	[48]
53-66	Features obtained from segments in the time domain: minimum value, maximum value, arithmetic mean, mode, first quartile, second quartile, third quartile, range, standard deviation, variance, variation, skewness, kurtosis and zero crossings.	[43], [47]
67-72	Spectral edge frequencies and their differences: SEF50(0.5-50Hz), SEF95(0.5-50Hz), SEF95(0.5-50Hz)-SEF50(0.5-50Hz), SEF50(8-16Hz), SEF95(8-16Hz) and SEF95(8-16Hz) - SEF50(8-16Hz).	[23]
73-77	Sample entropy, permutation entropy, approximate entropy, spectral entropy and fractal dimension.	[35], [47]

time-frequency representation of the EEG signal [48]. These features show the effectiveness for sleep staging. Since the REM stage involves the chin muscle and eye activity, EOG and EMG are normally required. However, Imtiaz *et al.* proposed spectral edge frequencies as features that can be used for REM detection based on recordings from a single EEG channel [23]. In addition, entropy-based features are important for measuring the degree of regularity of EEG signals. Baha *et al.* proposed wavelet-based features that show the frequency distribution of the signal and the amount of transformation in the distribution of the frequency [47]. According to this work, we plan to use the discrete wavelet transform (DWT) for EEG signal decomposition, where the used mother wavelet is the db4, and extract features based on the wavelet coefficients D2(16- 32 Hz), D3(8-16 Hz), D4(4-8 Hz) and A4(0-4 Hz) after four-level wavelet decomposition of EEG signal segments. After extraction, the data set under each feature is normalized using the Z-score method. To avoid that the test set can be seen, the procedure is first normalizing the training data and obtaining a regulation, and then using the same regulation to normalize the test data.

C. Experimental Design

In this section, we briefly introduce how to design the experiments. Five-fold cross validation is used for testing, and the size of any class in each fold is the same. In each fold, some instances are randomly selected from the training set, and the pairwise constraints are created between these instances and their k-nearest neighbors with the same and different labels. A constraint created by any pair of instance i and j contains $\mathbf{\Delta}_{ij}$ and $\mathbf{\Delta}_{ji}$. In these experiments, the must-link constraints and the cannot-link constraints are chosen as the same number, which would be more useful than the unbalanced must-link and cannot-constraints [6]. It should be noted that the pairwise constraints are fixed in each fold except when the pairwise constraints need to be discussed. In addition, features selection and metric learning are achieved based on these must-link constraints and cannot-link constraints that are created from the training data in each fold.

K-NN is applied as the main classifier, whose hyperparameter K is tuned using grid search with cross validation to better predict the labels of the test data. The classification performance is mainly evaluated by the classification accuracy, which is defined as

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{18}$$

where TP, TN, FP and FN are the true positive, the true negative, the false positive and the false negative, respectively.

Besides the classification accuracy, the other two measurements, F_1-score and Kappa coefficient, are used for evaluation of classification performance. F_1-score is defined as the harmonic mean of precision and recall

$$F_1 = \frac{2 \times precision \times recall}{precision + recall}, \tag{19}$$

where

$$precision = \frac{TP}{TP + FP}, \tag{20}$$

and

$$recall = \frac{TP}{TP + FN}. \tag{21}$$

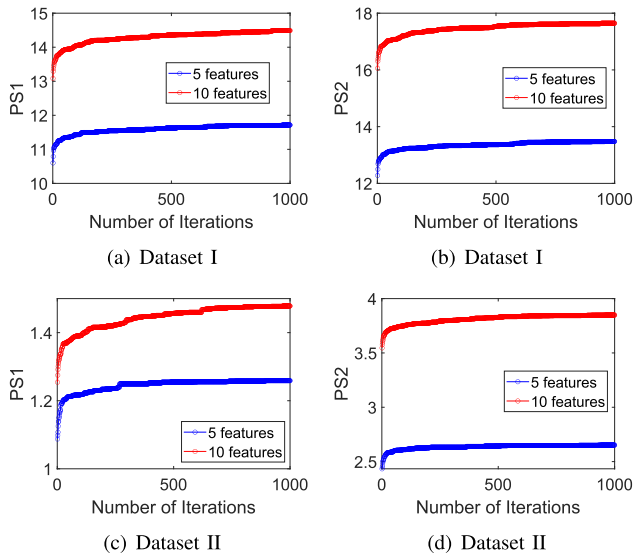
Kappa coefficient is used for measuring the agreement between two individuals, which is defined as

$$\kappa = \frac{p_o - p_e}{1 - p_e}, \tag{22}$$

where p_o is the relative observed agreement and p_e is the probability that agreement is due to chance. We independently run the program 5 times, and the three measurements obtained from each time are averaged as the final results, respectively.

D. Influence of Number of Iterations

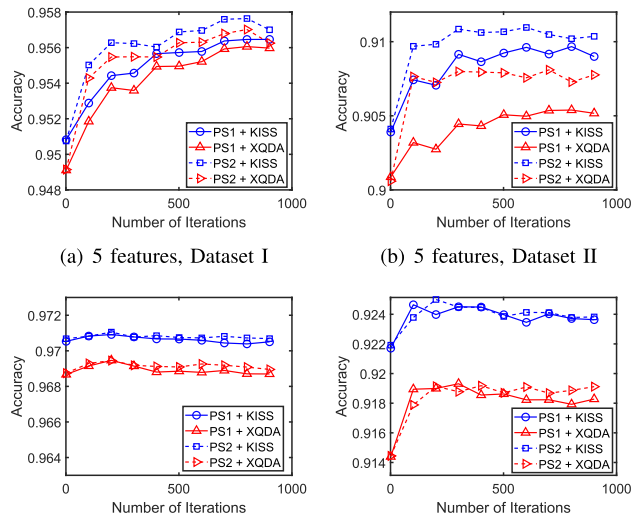
To efficiently and effectively apply Algorithm 1 for selecting an optimal feature set, the number of iterations should be set to an appropriate value. In Fig. 3, the growth rates of the PS1 and the PS2 versus the number of iterations with different number of selected features are investigated based on Dataset I and Dataset II, respectively. Figures 3(a) to 3(d) show that the trends of the



(a) Dataset I (b) Dataset I

(c) Dataset II (d) Dataset II

Fig. 3. Values of PS1 and PS2 versus the number of iterations, respectively. In these cases, the number of constraints is 10000.



(a) 5 features, Dataset I (b) 5 features, Dataset II

(c) 10 features, Dataset I (d) 10 features, Dataset II

Fig. 4. Classification accuracy versus the number of iterations by using PS1 + KISS, PS2 + KISS, PS1 + XQDA and PS2 + XQDA, respectively. In these cases, the number of constraints is 10000.

two scores using 5 or 10 features present rapid grow at first and then slow grow as the number of iterations increases. These results show that the feature set with the near-optimal PS1 and PS2 can be found by using Algorithm 1 within a limited number of iterations.

Further, the classification accuracies obtained by using PS1 and PS2 versus different numbers of iterations are discussed, as shown in Fig. 4. Because our approach is built on the metric learning, in all of the following experiments, the combinations PS1 + KISS, PS2 + KISS, PS1 + XQDA and PS2 + XQDA will be analyzed. In Figs. 4(a), 4(b) and 4(d), we observe that the trends of all the accuracy curves are similar to those in Fig. 3 regardless of the slight fluctuations. In these cases, when the number of iterations is less than 100, the performance can be significantly improved by increasing the number of iterations. In Figs. 4(c), all the accuracy almost stable as the number of iterations increases. From Figs. 4(a) to 4(d), we observe that the improvement of classification performance via increasing the number of iterations is more obvious when 5 features are selected. and the maximum growth rate of accuracy reaches about 0.7%. The results suggest that, to keep the accuracy at a high level, the number of iterations should be set large enough in most cases.

E. Influence of Constraints

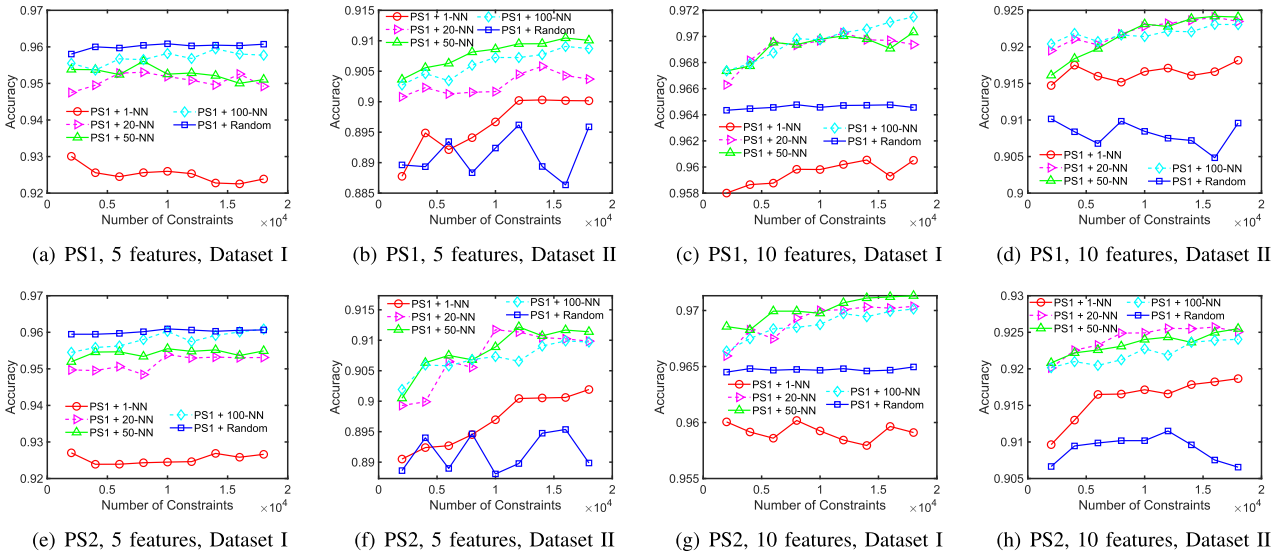
In previous work, the constraints were often created randomly, and their effectiveness was not fully considered. Thus, we discuss the influence of some different constraint-created methods on the classification performance. In this work, we created the constraints between some random samples and their nearest neighbors, which is similar to that of ReliefF. The comparison between our created methods with different numbers of k-nearest neighbors and the random selection method is shown in Fig. 5. It should be noted that a fixed value on the horizontal ordinate in Fig. 5 indicates a total number of constraints regardless of the method.

In general, Figs. 5(a) to 5(h) show that the accuracies by using the proposed created methods with $k = 20, 50, 100$ are higher than those using the proposed created method with $k = 1$. The random selection method shows different performances on Dataset I and Dataset II. On the Dataset I, it performs best when 5 features are selected and only better than the proposed created method with $k = 1$ when 10 features are selected. On the Dataset II, it performs worst in all cases. In addition, we observe that the performances of proposed created methods with $k = 20, 50, 100$ are generally similar with each other. Nevertheless, the proposed created method with $k = 100$ shows the performance closest to the best while 5 features are selected on the Dataset I, and the proposed created method with $k = 50$ is the best while PS1 is used and 5 featured are selected on the Dataset II. Therefore, in order to achieve a good and stable classification performance, it is considerable to use the proposed constraint created method with an appropriate k (i.e., k = 50, 100) for feature selection.

In Fig. 5(a), 5(c), 5(e) and 5(g), on the Dataset I, the proposed created methods with $k = 20, 50, 100$ show a growth trend while 10 features instead of 5 features are selected. The number of constraints has no significant impact on the performance of the created methods with $k = 1$ and the random selection method regardless of their fluctuations. In Fig. 5(b), 5(d), 5(f) and 5(h), on the Dataset II, all the proposed created methods show a growth trend as the number of constraints increases, and the random selection method does not show such trend but only the obvious fluctuation.

F. Feature Selection Performance

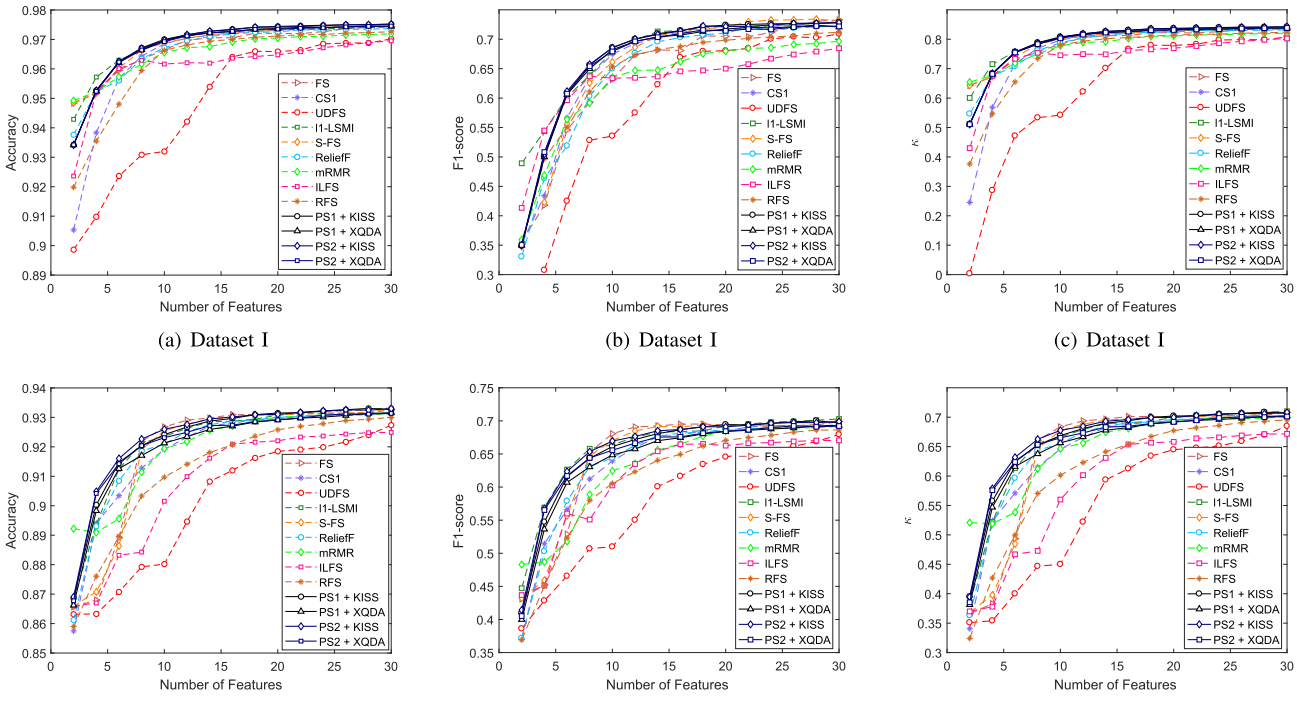
In this section, we compare the proposed method with nine widely used feature selection methods including Infinite



(a) PS1, 5 features, Dataset I (b) PS1, 5 features, Dataset II (c) PS1, 10 features, Dataset I (d) PS1, 10 features, Dataset II

(e) PS2, 5 features, Dataset I (f) PS2, 5 features, Dataset II (g) PS2, 10 features, Dataset I (h) PS2, 10 features, Dataset II

Fig. 5. Comparison of different constraint-created methods using KISS metric learning. The number of iterations is 200.



(a) Dataset I (b) Dataset I (c) Dataset I

(d) Dataset II (e) Dataset II (f) Dataset II

Fig. 6. (a, d) Classification accuracy, (b, e) F_1-score and (c, f) Kappa coefficient versus different numbers of features based on Dataset I and Dataset II, respectively. In these cases, the number of iterations of the proposed algorithm is 200 and number of labeled data or constraints are 10000. The constraint-created methods of 100-NN and 50-NN are used for Dataset I and Dataset II, respectively.

Latent Feature Selection (ILFS) [50], Unsupervised Discriminative Feature Selection (UDFS) [51], Robust Feature Selection (RFS) [52], l_1-penalized squared-loss mutual information (l1-LSMI) [53], Fisher Score (FS) [1], minimal-redundancy-maximal-relevance criterion (mRMR) [49], ReliefF [5], Constraint Score1(CS1) [6] and trace ratio criterion [4]. For the trace ratio criterion, the subset-level Fisher score (S-FS) is used. Among all the methods, our method and CS1 are constraint-guided and share the same pairwise constraints in each fold and each time.

A comparison of these feature selection methods is shown in Fig. 6, which plots the average classification accuracies, Kappa coefficients and average F_1-scores versus different numbers of features selected by different methods. Figures 6(a) to 6(f) show that, on the Dataset I and Dataset II, the three measurements obtained using PS1 + KISS, PS1 + XQDA, PS2 + KISS and

TABLE III
COMPUTATIONAL TIMES CORRESPONDING TO 10 FEATURES AND 20 FEATURES IN FIG. 6

No.of selected features	Data sets	FS	CS1	UDFS	l_1-LSMI	S-FS	ReliefF	mRMR	ILFS	RFS	KISS+ PS1	XQDA+ PS1	KISS+ PS2	XQDA+ PS2
10 features	Dataset I	0.11s	0.07s	30.62s	72.26s	4.50s	1.00s	0.84s	0.62s	0.79s	8.02s	6.81s	5.84s	5.79s
	Dataset II	0.06s	0.03s	29.87s	74.99s	3.88s	0.90s	0.79s	0.24s	0.78s	7.96s	7.02s	5.76s	5.76s
20 features	Dataset I	0.05s	0.03s	30.61s	102.85s	5.17s	1.20s	0.88s	0.52s	0.98s	7.26s	5.64s	6.61s	6.33s
	Dataset II	0.06s	0.02s	29.34s	93.86s	4.46s	0.87s	0.92s	0.42s	1.23s	8.38s	6.97s	5.90s	5.88s

TABLE IV
CLASSIFICATION ACCURACIES (%) OF SIX SLEEP STAGES BASED ON DATASET I. THE NUMBER OF ITERATIONS OF THE PROPOSED ALGORITHM IS 200, AND THE NUMBER OF LABELED DATA OR CONSTRAINTS IS 10000

Methods	10 features							20 features						
	AWA	S1	S2	S3	S4	REM	Average	AWA	S1	S2	S3	S4	REM	Average
FS	95.70	97.56	94.91	97.35	98.76	95.60	96.65	96.81	97.70	95.63	97.82	99.04	96.25	97.21
CS1	95.60	97.63	95.27	97.71	98.93	95.97	96.85	96.66	97.69	95.87	98.00	99.10	96.53	97.31
UDFS	85.37	97.62	85.69	97.28	98.68	94.55	93.20	94.30	97.75	94.72	97.72	98.80	96.23	96.59
l_1-LSMI	96.03	97.63	95.42	97.76	98.96	95.90	96.95	96.77	97.70	95.85	**98.06**	**99.17**	96.51	97.34
S-FS	**96.51**	**97.71**	95.07	97.32	98.75	95.95	96.89	96.86	**97.78**	95.86	97.99	**99.17**	96.21	97.31
ReliefF	95.86	97.58	94.94	97.43	98.79	95.68	96.71	96.74	97.70	95.67	97.84	99.10	96.49	97.26
mRMR	95.40	97.53	94.91	97.45	98.62	95.43	96.56	96.55	97.61	95.37	97.60	98.84	96.33	97.05
ILFS	93.64	97.46	93.65	**97.80**	**99.02**	95.43	96.17	94.99	97.55	94.05	97.90	99.07	95.30	96.48
RFS	94.75	97.59	95.09	97.70	98.90	95.74	96.63	96.01	97.64	95.65	97.94	99.11	96.36	97.12
KISS + PS1	96.14	97.66	95.48	97.76	98.98	**95.98**	**97.00**	**97.13**	97.72	**96.01**	98.00	99.11	**96.68**	**97.44**
XQDA + PS1	96.07	97.66	95.47	97.76	98.95	95.91	96.97	97.03	97.70	95.89	97.94	99.09	96.59	97.37
KISS + PS2	96.04	97.64	**95.49**	97.78	98.97	95.89	96.97	**97.13**	97.71	95.94	97.98	99.11	96.67	97.42
XQDA + PS2	95.97	97.65	95.43	97.75	98.93	95.83	96.93	97.03	97.68	95.83	97.92	99.09	96.55	97.35

TABLE V
CLASSIFICATION ACCURACIES (%) OF SIX SLEEP STAGES BASED ON DATASET II. THE NUMBER OF ITERATIONS OF THE PROPOSED ALGORITHM IS 200, AND THE NUMBER OF LABELED DATA OR CONSTRAINTS IS 10000

Methods	10 features							20 features						
	AWA	S1	S2	S3	S4	REM	Average	AWA	S1	S2	S3	S4	REM	Average
FS	**96.06**	92.46	**87.58**	**93.36**	96.94	89.65	**92.68**	96.27	**92.86**	87.94	93.54	97.15	90.75	93.08
CS1	95.29	92.23	85.44	92.6	96.33	89.59	91.91	95.96	92.40	87.84	93.50	97.15	90.81	92.94
UDFS	93.85	**92.55**	72.95	91.62	95.13	82.01	88.02	95.06	92.25	85.21	92.74	96.59	89.24	91.85
l_1-LSMI	95.67	92.26	86.51	93.35	**97.09**	89.40	92.36	96.12	92.45	87.91	**93.72**	97.26	91.00	93.08
S-FS	95.99	92.21	87.44	93.22	97.02	89.25	92.54	96.31	92.72	87.96	93.54	97.10	90.85	93.08
ReliefF	95.31	92.11	86.42	93.03	96.58	89.16	92.10	96.15	92.43	87.79	93.63	97.24	90.81	93.01
mRMR	95.19	92.51	85.84	92.43	96.05	89.61	91.94	**96.47**	92.65	87.52	93.44	96.90	90.96	92.99
ILFS	95.76	92.47	79.26	92.67	96.67	84.09	90.15	96.28	92.53	85.08	93.32	96.94	89.08	92.21
RFS	94.20	92.32	82.52	92.52	96.19	88.07	90.97	95.53	92.39	86.76	93.30	96.94	90.57	92.58
KISS + PS1	95.71	92.35	86.63	92.93	96.62	90.25	92.42	96.28	92.54	**88.09**	93.64	97.21	**91.13**	**93.15**
XQDA + PS1	95.33	92.23	86.20	92.76	96.43	89.83	92.13	96.00	92.29	87.73	93.55	97.16	90.74	92.91
KISS + PS2	95.83	92.31	87.17	93.12	96.71	**90.45**	92.60	96.21	92.43	88.02	93.66	**97.27**	91.03	93.10
XQDA + PS2	95.55	92.17	86.67	92.83	96.53	90.13	92.31	95.93	92.33	87.77	93.54	97.19	90.73	92.92

PS2 + XQDA have similar trends that obviously increase as the number of selected features increases from 2 to 10, and slightly increase as the number of selected features increases from 10 to 20. Once the number of selected features exceeds 20, further increasing it contributes little to the classification performance.

As shown in Figs. 6(a) to 6(f), PS1 + KISS and PS2 + KISS can perform best in most cases on the two data sets. For instance, Fig. 6(a) shows that, on Dataset I, PS1 + KISS and PS2 + KISS show the two best performances as the number of selected features exceeds 6. Figure 6(d) shows that, on Dataset II, PS2 + KISS performs best when 4, 6 or 8 features is selected. Using FS can achieve the highest accuracy as the number of selected features is between 10 and 18, but obviously lower than the proposed method as the number of selected features is 4 and 6.

In Table III, the computational time of each method is listed corresponding to the results with 10 and 20 selected features shown in Fig. 6. It can be seen that, despite the good overall performance of l_1-LSMI on the two data sets, its computational time is much longer than those of other methods and is significantly affected by the number of features. The computational times of the proposed PS1 + KISS, PS1 + XQDA, PS2 + KISS and PS2 + XQDA are comparable with each other, which are less than those of l_1-LSMI and UDFS but more than those of other methods.

To further analyze the performance of each method, we list the classification accuracies obtained using different methods on the six sleep stages, respectively, as shown in Tables IV and V. In Tables IV and V, the highest values are shown in bold. As shown in Table IV, on the Dataset I, the average accuracy obtained using KISS + PS1 are the highest when 10 or 20 features are selected. In addition, KISS + PS1 performs best on some sleep stages, and performs second best on most of the rest. Table V shows that,

TABLE VI
P-VALUE BETWEEN THE ACCURACIES OBTAINED BY USING KISS + PS1 AND THE ACCURACIES OBTAINED BY USING NINE EXISTING METHODS, RESPECTIVELY. SIGNIFICANT RESULTS ARE INDICATED BY: $*$ P $<$ 0.05

Methods	10 features		20 features	
	Dataset I	Dataset II	Dataset I	Dataset II
FS	5.71E-06*	5.70E-02	6.15E-11*	3.53E-01
CS1	2.28E-01	1.36E-02*	5.29E-05*	1.24E-02*
UDFS	1.29E-38*	8.58E-35*	6.82E-36*	1.45E-25*
l_1-LSMI	5.24E-01	7.96E-01	2.61E-03*	3.47E-01
S-FS	8.89E-02	3.44E-01	7.85E-06*	3.00E-01
ReliefF	4.02E-03*	8.30E-02	2.37E-08*	2.86E-02*
mRMR	1.34E-07*	6.92E-04*	1.36E-19*	2.40E-02*
ILFS	4.33E-09*	4.38E-15*	1.01E-31*	2.31E-17*
RFS	2.24E-04*	2.70E-08*	6.75E-13*	4.94E-08*

on the Dataset II, FS and KISS + PS1 perform best when 10 and 20 features are selected, respectively. On the two data sets, the highest accuracy on REM in most cases can be obtained by using KISS + PS1.

To compare the proposed method that performs best with other existing methods from another angle, we use an unpaired Student's t-test to calculate the P-value associated with the average classification accuracy. Table VI shows the p values between KISS + PS1 and other nine existing methods. On the Dataset I, the p values corresponding to FS, UDFS, ReliefF, mRMR, ILFS and RFS with 10 selected features and all the methods with 20 selected features are smaller than 0.01. It indicates that the accuracies obtained by using KISS + PS1 in Table IV are very significantly higher than those obtained by using the corresponding methods. On the Dataset II, the p values corresponding to FS, l_1-LSMI, S-FS, ReliefF with 10 selected features and FS, l_1-LSMI, S-FS with 20 selected features exceed 0.05, indicating that the average accuracies obtained using KISS + PS1 in Table V are comparable with these methods, and significantly or very significantly higher than those obtained by using other methods.

As well as classification accuracy, the overall accuracy, Kappa coefficient and average F_1-score with 10 and 20 features based on Dataset I and Dataset II are listed, as shown in Tables VII and VIII, respectively. The overall accuracy here is defined as the ratio of the number of correctly classified instances to the number of total instances, which is different from that in Eq. 18. Tables VII and VIII show that, on the two data sets, using KISS + PS1 can achieve the highest overall accuracies, Kappa coefficients and average F_1-scores in most cases, except for the case while 10 features are selected based on Dataset II.

Figure 7 shows the classification accuracies obtained using different feature selection methods versus different numbers of constraints or labeled data. Overall, the accuracies obtained by using the PS1 + KISS, PS1 + XQDA, PS2 + KISS and PS2 + XQDA present rapid grow at first and then slow grow as the number of constraints increases. Figures 7(a) and 7(b) show that, on the Dataset I, PS1 + KISS and PS2 + KISS can achieve the two highest accuracies among all the compared methods as the number of constraints exceeds 6000 while using 10 and 20 features, respectively. Figures 7(c) and 7(d) show that, on the

TABLE VII
OVERALL ACCURACIES (OA)(%), KAPPA COEFFICIENTS (κ)(%) AND AVERAGE F_1-SCORE (AF_1)(%) OF SIX SLEEP STAGES BASED ON DATASET I. THE NUMBER OF ITERATIONS OF THE PROPOSED ALGORITHM IS 200, AND THE NUMBER OF LABELED DATA OR CONSTRAINTS IS 10000

Methods	10 features			20 features		
	κ	OA	$\mathbf{AF_1}$	κ	OA	$\mathbf{AF_1}$
FS	78.27	89.94	63.17	82.20	91.62	70.50
CS1	79.75	90.55	67.46	82.78	91.92	71.56
UDFS	54.23	79.59	53.56	77.91	89.76	68.05
l_1-LSMI	80.39	90.85	67.92	83.05	92.03	72.24
S-FS	80.04	90.66	66.12	82.91	91.94	72.04
ReliefF	78.81	90.14	65.02	82.45	91.77	71.03
mRMR	77.78	89.67	63.52	81.15	91.15	67.98
ILFS	74.56	88.50	63.38	76.96	89.43	65.00
RFS	77.99	89.89	65.31	81.38	91.35	69.83
KISS + PS1	**80.80**	**91.00**	**68.59**	**83.72**	**92.33**	**72.41**
XQDA + PS1	80.61	90.92	68.22	83.28	92.12	71.73
KISS + PS2	80.59	90.91	68.54	83.60	92.27	72.17
XQDA + PS2	80.31	90.78	67.93	83.14	92.05	71.43

TABLE VIII
OVERALL ACCURACIES (OA)(%), KAPPA COEFFICIENTS (κ)(%) AND AVERAGE F_1-SCORE (AF_1)(%) OF SIX SLEEP STAGES BASED ON DATASET II. THE NUMBER OF ITERATIONS OF THE PROPOSED ALGORITHM IS 200, AND THE NUMBER OF LABELED DATA OR CONSTRAINTS IS 10000

Methods	10 features			20 features		
	κ	OA	$\mathbf{AF_1}$	κ	OA	$\mathbf{AF_1}$
FS	**68.36**	**78.03**	**68.05**	70.01	79.25	69.33
CS1	64.58	75.74	63.95	69.37	78.83	68.45
UDFS	45.01	64.06	50.98	64.48	75.54	64.56
l_1-LSMI	66.64	77.08	66.40	69.96	79.23	69.25
S-FS	67.72	77.63	67.05	69.98	79.24	69.30
ReliefF	65.52	76.30	64.41	69.62	79.02	68.75
mRMR	64.65	75.82	62.46	69.42	78.97	68.49
ILFS	56.01	70.46	60.25	65.82	76.62	66.26
RFS	60.14	72.91	60.55	67.73	77.74	67.05
KISS + PS1	66.87	77.25	66.12	**70.21**	**79.44**	**69.43**
XQDA + PS1	65.64	76.39	64.84	69.18	78.73	68.40
KISS + PS2	67.76	77.79	66.95	69.98	79.31	69.10
XQDA + PS2	66.51	76.94	65.56	69.20	78.75	68.44

Dataset II, the accuracies obtained using PS1 + KISS or PS2 + KISS are comparable with l_1-LSMI and S-FS, but slightly lower than those obtained using FS while the number of constraints exceeds 8000. These results indicate that, to obtain a good performance while using the proposed feature selection method and the proposed constraint-created method, adequate number of constraints should be used.

Figure 8 shows the accuracies obtained using the proposed method and different classifiers including KNN, support vector machine (SVM), random forest (RF), back propagation neural network (BPNN) and decision tree (DT). Overall, on both the datasets, the SVM and RF are more competitive than the other three classifiers. On Dataset I, the highest accuracy 97.66% is obtained using PS1 + KISS and SVM based on 20 features, and on Dataset II, the highest accuracy 93.57% is obtained using PS1 + KISS and RF based on 20 features. When the number of selected features is reduced to 10, the decline of the highest accuracies is not obvious, only 0.39% and 0.53% on Dataset I and on Dataset II, respectively.

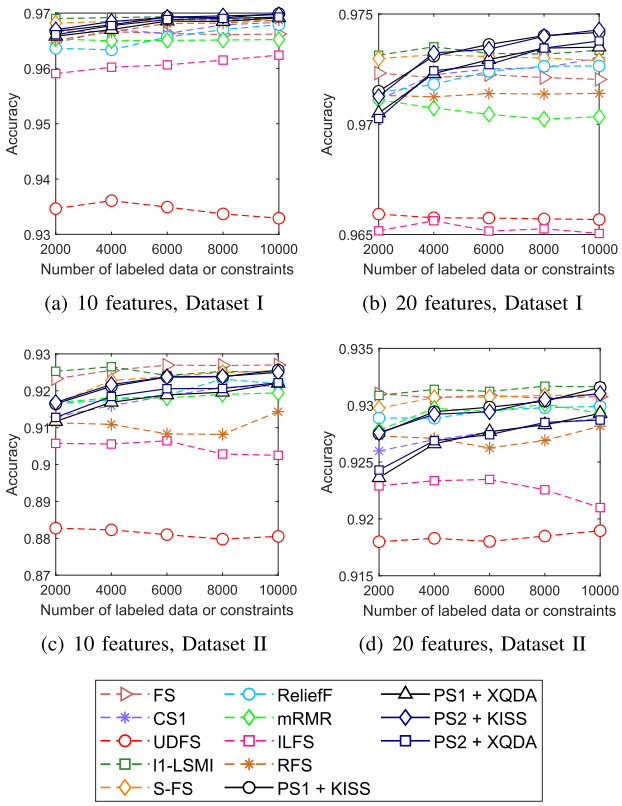


Fig. 7. Classification accuracy versus number of labeled data or constraints with different number of selected features on Dataset I and Dataset II, respectively. In these cases, the number of iterations of the proposed algorithm is 200 and number of labeled data or constraints are 10000. The constraint-created methods of 100-NN and 50-NN are used for Dataset I and Dataset II, respectively.

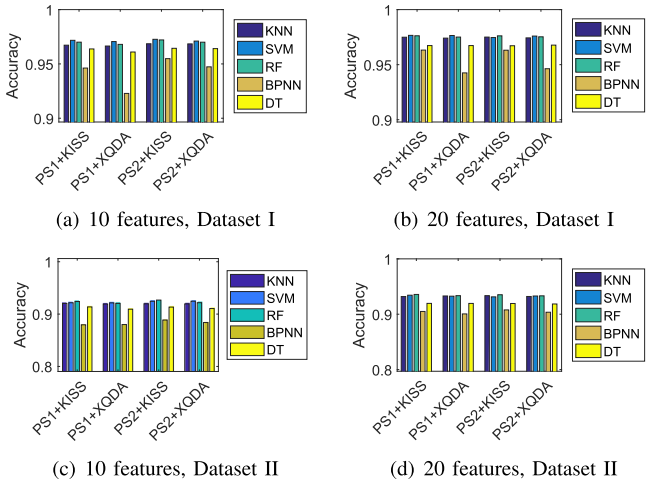


Fig. 8. Comparison of accuracies obtained using the proposed method and different classifiers on Dataset I and Dataset II, respectively. In these cases, the number of iterations of the proposed algorithm is 200 and number of constraints are 10000. The constraint-created methods of 100-NN and 50-NN are used for Dataset I and Dataset II, respectively.

IV. Discussion

This work proposed a novel method for feature selection with pairwise constraints to classify sleep stages. According to Fig. 6, Table IV and Table V, the proposed PS1 + KISS and PS2 + KISS are competitive while comparing them with some widely used methods and some state-of-the-art methods. Although the proposed method is not faster than some other methods, the proposed PS1 + KISS and PS2 + KISS can achieve the best performance on the two data sets in many cases.

Figure 6 shows that the performance differences of various methods are reduced as more features are selected. The reason may be that all these methods tend to select more similar feature sets when the number of selected features increases. As the feature selection methods using pairwise constraints, the proposed PS1 and PS2 combining with the metric learning performs better than CS1, especially in the cases with less features. the difference may result from whether the feature interaction are considered and the distance metric learning is applied. In addition, we notice that the classification performance using KISS metric learning is usually better than that using XQDA metric learning. The reason would be that using XQDA metric learning is often achieved with dimension reduction of feature space. However, it can show the advantage when compared to the feature spaces of the same dimension obtained using other methods [16].

As an important sleep stage for the diagnosis of sleep disorders, REM stage generally presents the EEG signals similar to those of wake and S1 stages. Therefore, it is difficult to detect REM stage only by means of EEG signals. Instead, the combinations of EEG, EMG and/or EOG are often used to better detect REM stage [23]. However, using only EEG channels to detect sleep stages would be more convenient than the combinations in practical applications. Tables IV and V show that our proposed method using a single EEG channel can achieve the highest accuracies on REM stage, which might be benefit to the convenient diagnosis of sleep disorders.

We found that the influence of number of iterations on the accuracies is more obvious as less features are selected (See Fig. 4), implying that searching for the features with profitable interaction based on KISS metric or XQDA metric is more critical in the case. However, this does not mean that one need to add additional iterations to achieve significant performance improvements, and thus the number of iterations can be fixed for the cases with different number of selected features. For example, by fixing the number of iterations as 200, our method can obtain higher accuracies than those obtained by CS1 that neglects the feature interaction, and this gap is more visible as less features are selected (See Fig. 6).

The method used for creating constraints significantly impacts the classification performance (See Fig. 5). On the whole, the constraint-created methods of 50-NN and 100-NN perform well and robustly in most cases. The reason may be that they can create adequate informative constraints for feature selection. In comparison, the 1-NN method would tend to create less informative constraints and the random created method would be susceptible to noise because of its unstable performance. Therefore, how to stably identify and search the informative constraints is critical for improving the classification performance. In fact, some work has been done on this issue, e.g., the work in [54] and [55] that aims to select the informative constraints for

TABLE IX
COMPARISON OF CLASSIFICATION ACCURACIES REPORTED IN OTHER WORK WITH THE PROPOSED METHOD

Method	Channel	Accuracy(%)	No. of epochs
Diykh et al.,2016 [43]	EEG	95.93	14963
Hassan et al.,2017 [21]	EEG	92.43	15188
Hassan et al.,2017 [57]	EEG	88.07	15188
Abdull et al.,2019 [56]	EEG	93.1	23806
Sharma et al.,2017 [58]	EEG	90.02	15136
Proposed Method	EEG	97.66	41079
Proposed Method	EEG	93.57	14621

semi-supervised clustering. On the basis of this kind of work, it is still possible to further improve the performance of sleep staging by utilizing more helpful constraints for feature selection and metric learning.

It is difficult to accurately compare the results of different work, since different data sets were used. In particular, the degree of imbalance in the used data sets would significantly impact the results. Here we compare our best results with those in some previous work using a single EEG channel for sleep staging. For the sake of fairness and convenience, the results on the data sets whose instances belonging to the same class account for more than 50% and the rest are respectively compared. Table IX shows the accuracies reported in previous work and the proposed method on Dataset I (41079 epochs) and Dataset II (14621 epochs). Except for [56] and Dataset II, the instances belonging to one class (AWA stage) in [43], [21], [57], [58] and Dataset I account for more than 50%. Compared with the results from the five methods, the accuracy obtained using our proposed method is the highest one on maximum amount of data (41079 epochs). In addition, compared with the result on Dataset II and that in [56], our proposed method can also achieve the highest accuracy. However, to achieve the best results, 20 features have been selected and used in the proposed method. How to use less features to achieve a good performance of sleep staging, especially on more balanced data, is still a challenging task.

Single features and their combinations for sleep staging have been deeply investigated in previous work. As an example, here we report the 15 features that are selected most frequently by using PS1 on Dataset I and Dataset II, as shown in Table X. One-way ANOVA is used to show whether different levels of a factor have a significant effect on observed variables. The null hypothesis for one-way ANOVA is that all population means are identical. The null hypothesis is rejected when at least one mean is significantly different from others. However, it can not indicate which mean(s) is/are different. Thus, here a post-hoc Bonferroni test is used to address this issue. For convenience, Table X only shows the p-values by the post-hoc test on some sleep stage pairs. On Dataset I, most features are significantly different between AWA and REM, and least features are significantly different between REM and S1. On Dataset II, all features are significantly different between S2 and S3, and least features are significantly different between REM and S1. This results indicate that it is easy to confuse REM with S1. Nevertheless, without regard to the feature interaction and the influence of metric learning, many features are significantly different in the five cases, such as the permutation entropy (74) on both data sets.

TABLE X
15 TOP SELECTED FEATURES WITH THEIR P VALUES USING PS1 ON DATASET I AND DATASET II. A POST-HOC BONFERRONI TEST IS USED TO OBTAIN THE SIGNIFICANCE RESULTS, WHICH IS SHOWN ONLY BETWEEN TWO SLEEP STAGES IN SOME SLEEP STAGE PAIRS. THE OTHER CONDITIONS ARE THE SAME AS THOSE IN FIG. 6. SIGNIFICANT RESULTS ARE INDICATED BY: $*$ P $<$ 0.05

Dataset I					
No.	AWA-REM	REM-S1	S1-S2	S2-S3	S3-S4
4	1.000	0.000*	0.000*	0.000*	0.000*
5	0.000*	0.000*	0.000*	0.000*	1.000
6	0.000*	0.000*	0.000*	0.013*	1.000
19	0.000*	0.076	0.000*	0.000*	0.000*
21	0.000*	0.000*	0.000*	0.000*	0.245
22	0.000*	0.031*	1.000	1.000	1.000
26	0.000*	1.000	1.000	1.000	1.000
28	0.000*	1.000	0.000*	0.000*	0.002*
30	0.000*	0.003*	1.000	1.000	1.000
32	0.000*	0.016*	0.000*	0.000*	0.000*
52	0.000*	0.199	0.000*	0.024*	1.000
61	0.000*	1.000	0.000*	0.000*	0.000*
62	0.000*	1.000	0.000*	0.000*	0.000*
74	0.000*	0.000*	0.000*	0.000*	0.000*
75	0.000*	1.000	1.000	0.000*	0.000*
Dataset II					
No.	AWA-REM	REM-S1	S1-S2	S2-S3	S3-S4
4	0.726	0.000*	0.000*	0.000*	0.002*
13	0.000*	0.000*	0.000*	0.000*	0.008*
15	0.000*	0.000*	0.000*	0.000*	0.000*
18	0.000*	0.000*	0.000*	0.000*	0.000*
19	0.000*	0.007*	0.000*	0.000*	1.000
20	0.000*	0.020*	0.000*	0.000*	0.185
25	0.000*	0.048*	0.000*	0.000*	0.000*
29	0.000*	0.280	0.377	0.000*	0.000*
35	0.000*	0.000*	0.000*	0.000*	0.000*
46	0.000*	0.425	0.000*	0.000*	1.000
57	0.000*	1.000	0.000*	0.000*	0.000*
59	0.000*	1.000	0.000*	0.000*	0.000*
61	0.000*	0.000*	0.000*	0.000*	0.000*
62	0.000*	0.368	0.295	0.000*	0.000*
74	0.000*	0.000*	0.000*	0.000*	0.000*

V. CONCLUSION

In this work, we proposed a novel feature selection method based on KISS metric learning and XQDA metric learning called Plausibility Score. In this method, we designed two feature selection criteria by making simple changes to the KL divergence, and an effective sequential search strategy based on the proposed criteria to find out the optimal feature subset. Subsequently, the combinations of the proposed method and KISS metric learning or XQDA metric learning have been used in the experiments of sleep staging. The experimental results show that, compared with nine representative feature selection methods, the proposed method can perform best in many cases by using pairwise constraints instead of more expensive class labels. In the following work, we plan to investigate the effectiveness of Plausibility Score in more real-world applications.

REFERENCES

[1] R. Duda, P. Hart, and D. Stork, "*Pattern Classification*," New York: Wiley, 2001.

[2] J. Li *et al.*, "Feature selection: A data perspective," *ACM Comput. Surveys*, vol. 50, no. 6, pp. 1–45, 2016.

[3] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 4, pp. 491–502, 2005.

[4] F. Nie, S. Xiang, Y. Jia, C. Zhang, and S. Yan. "Trace ratio criterion for feature selection," in *Proc. 23rd AAAI Conf. Artif. Intell.*, AAAI 2008, pp. 13–17.

[5] M. Robnik-Sikonja and I. Kononenko, "Theoretical and empirical analysis of reliefF and RReliefF," *Mach. Learn.*, vol. 53, no. 1-2, pp. 23–69, 2003.

[6] D. Zhang, S. Chen, and Z. H. Zhou, "Constraint score: A new filter method for feature selection with pairwise constraints," *Pattern Recognit.*, vol. 41, no. 5, pp. 1440–1451, 2008.

[7] Q. Gu, Z. Li, and J. Han, "Generalized fisher score for feature selection," *UAI 2011, Proc. 27th Conf. Uncertainty Artifi. Intell.*, Barcelona, Spain, Jul. 2011, pp. 14–17.

[8] J. C. Lu, F. L. Liu, and X. Y. Luo, "Selection of image features for steganalysis based on the Fisher criterion," *Digit. Investigation*, vol. 11, no. 1, pp. 57–66, 2014.

[9] M. M. Kabir, M. M. Islam, and K. Murase, "A new wrapper feature selection approach using neural network," *Neurocomputing*, vol. 73, no. 16-18, pp. 3273–3283, 2010.

[10] M. Kretowski and M. Grzes, "Evolutionary learning of linear trees with embedded feature selection," *Int. Conf. Artif. Intell. Soft Comput.*, Springer-Verlag, 2006.

[11] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell, "Distance Metric Learning with Application to Clustering with Side-Information," *Int. Conf. Neural, Inf. Process. Syst.*. MIT Press, 2002.

[12] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov. "Neighbourhood components analysis," *Int. Conf. Neural, Inf. Process. Syst.*, MIT Press, 2004.

[13] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification,"*J. Mach. Learn. Res.*, vol. 10, pp. 207–244, 2009.

[14] J. Davis, B. Kulis, S. Sra, and I. Dhillon, "Information-theoretic metric learning," *Icml 07: Int. Conf. Mach. Learn.*, 2007.

[15] M. Kostinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," *IEEE Proc. Conf. Comput. Vision, Pattern, Recognit. IEEE*, 2012.

[16] S. Liao, Y. Hu, X. Zhu, and S. Z. Li, "Person re-identification by local maximal occurrence representation and metric learning," *Conf. Comput. Vision Pattern Recognit. IEEE*, 2015, pp. 2197$ndash;2206.

[17] Z. Prekopcsak and D. Lemire, "Time series classification by class-specific Mahalanobis distance measures,"*Adv. Data Anal. Classification*, vol. 6, no. 3, pp. 185–200, 2012.

[18] M. Matton, D. V. Compernolle and R. Cools, "Minimum classification error training in example based speech and pattern recognition using sparse weight matrices," *J. Comput. Appl. Math.*, vol. 234, no. 4, pp. 1303–1311, 2010.

[19] R. Paredes, and E. Vidal, "Learning prototypes and distances: A prototype reduction technique based on nearest neighbor error minimization," *Pattern Recognit.*, vol. 39, no. 2, pp. 180–188, 2006.

[20] A. Rechtschaffen and A. Kales, "A manual of standardized terminology, techniques and scoring system for sleep stages of human subjects," US Government Printing Office, Washington, 1969.

[21] A. R. Hassan and A. Subasi, "A decision support system for automated identification of sleep stages from single-channel EEG signals," *Knowl.-Based Syst.*, vol. 128, no. 15, pp. 115–124, 2017.

[22] M. E. Tagluk, N. Sezgin, and M. Akin, "Estimation of sleep stages by an artificial neural network employing EEG, EMG and EOG," *J. Med. Syst.*, vol. 34, no. 4, pp. 717–725, 2010.

[23] S. A. Imtiaz and E. Rodriguez-Villegas, "A low computational cost algorithm for REM sleep detection using single channel EEG," *Ann. Biomed. Eng.*, vol. 42, no. 11, pp. 2344–2359, 2014.

[24] M. Langkvist, L. Karlsson, and A. Loutfi, "Sleep stage classification using unsupervised feature learning," *Adv. Artif. Neural Syst.*, 2012.

[25] L. Zoubek, S. Charbonnier, S. Lesecq, A. Buguet, and F. Chapotot, "Feature selection for sleep/wake stages classification using data driven methods," *Biomed. Signal Process. Control*, vol. 2, no. 3, pp. 171–179, 2007.

[26] S. Khalighi, T. Sousa, D. Oliveira, G. Pires, and U. Nunes, "Efficient feature selection for sleep staging based on maximal overlap discrete wavelet transform and SVM," *2011 Annu. Int. Conf. Eng. Medicine Biol. Soc.*, 2011, pp. 3306–3309.

[27] E. Alickovic, and A. Subasi, "Ensemble SVM method for automatic sleep stage classification," *IEEE Trans. on Instrum. Meas.*, Vol. 67, no. 6, pp. 1258–1265, 2018.

[28] Y. Ji, X. Bu, J. Sun, and Z. Liu, "An improved simulated annealing genetic algorithm of EEG feature selection in sleep stage," *Signal Info. Process. Assoc. Summit. Conf. IEEE*, 2017.

[29] S. Ozsen, "Classification of sleep stages using class-dependent sequential feature selection and artificial neural network," *Neural Comput. Appl.*, vol. 23, no. 5, pp. 1239–1250, 2013.

[30] T. Tuncer, S. Dogan, and A. Subasi, "Surface EMG signal classification using ternary pattern and discrete wavelet transform based feature extraction for Hand Movement Recognition," *Biomed. Signal Process. Control*, Vol. 58, p. 101872, 2020.

[31] H. Phan, Q. Do, T.-L. Do, and D.-L. Vu, "Metric learning for automatic sleep stage classification," in *Proc. 35th Annu. Int. Conf. IEEE Eng. Med. Bio. Soc.*, Jul. 2013, pp. 5025–5028.

[32] C. L. Huang and C. J. Wang, "A GA-based feature selection and parameters optimizationfor support vector machines," *Expert Syst. Appl.*, vol. 31, no. 2, pp. 231–240, 2006.

[33] M. Draminski, A. Rada-Iglesias, S. Enroth, C. Wadelius, J. Koronacki, and J. Komorowski, "Monte Carlo feature selection for supervised classification," *Bioinformatics*, vol. 24, no. 1, pp. 110–117, 2008.

[34] K. H. Quah and C. Quek, "MCES: A novel Monte Carlo evaluative selection approach for objective feature selections," *IEEE Trans. Neural Networks*, vol. 18, no. 2, pp. 431–448, 2007.

[35] J. L. Rodriguez-Sotelo, A. Osorio-Forero, A. Jimenez-Rodriguez, D. Cuesta-Frau, E. Cirugeda-Roldan, and D. Peluffo, "Automatic sleep stages classification using EEG entropy features and unsupervised pattern analysis techniques," *Entropy*, vol. 16, no. 12, pp. 6573–6589, 2014.

[36] G. Cornuejols, "Valid inequalities for mixed integer linear programs," *Math. Program. B*, vol. 112, no. 1, pp. 3–44, 2008.

[37] G. L. Nemhauser and L. A. Wolsey, *Integer and Combinatorial Optimization*, Wiley-Interscience, New York, 1999.

[38] E. Danna, E. Rothberg, and C. L. Pape, "Exploring relaxation induced neighborhoods to improve MIP solutions," *Math. Program.*, vol. 102, no. 1, pp. 71–90, 2005.

[39] M. Grotschel, *Primal Heuristics for Mixed Integer Programs*, Technischen Universitat Berlin, September 2006.

[40] Kemp, "The Sleep-EDF Database," World Wide Web. [Online] http://www.physionet.org/physiobank/database/sleep-edf/, Accessed Aug. 2009.

[41] B. Kemp, A. H. Zwinderman, B. Tuk, H. A. C. Kamphuisen, and J. J. L. Oberye. "Analysis of a sleep-dependent neuronal feedback loop: The slow-wave microcontinuity of the EEG," *IEEE Trans. Biomed. Eng.* vol. 47, no. 9, pp. 1185–1194, 2000.

[42] A. L. Goldberger *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals," *Circulation* 101, e215-e220, 2000.

[43] M. Diykh, Y. Li, and P. Wen, "EEG sleep stages classification based on time domain features and structural graph similarity," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 11, pp. 1159–1168, 2016, pp. 113–117.

[44] M. I. Alkadi, M. B. I. Reaz, and M. A. M. Ali, "Compatibility of mother wavelet functions with the electroencephalographic signal," *IEEE-EMBS Conf. Bio. Eng. Sci.*, 2012.

[45] E. Estrada1, H. Nazeran, G. Sierra, F. Ebrahimi, and M. Mikaeili, "Wavelet-based EEG denoising for automatic sleep stage classification," *21st IEEE Int. Conf. Electric. Commun. Comput.*, 2011, pp.295–298.

[46] A. Krakovska and K. Mezeiova, "Automatic sleep scoring: A search for an optimal combination of measures," *Artif. Intell. Medicine*, vol. 53, no. 1, pp. 25–33, 2011.

[47] B. Sen, M. Peker, A. Cavusoglu, and F. V. Celebi, "A comparative study on classification of sleep stage based on EEG signals using feature selection and classification algorithms," *J. Med. Syst.*, vol. 38, no. 18, pp. 1–21, 2014.

[48] V. Bajaj and R. B. Pachori, "Automatic classification of sleep stages based on the time-frequency image of EEG signals," *Comput. Methods Prog. Biomed.*, vol. 112, no. 3, pp. 320–328, 2013.

[49] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, 2005.

[50] G. Roffo, S. Melzi, U. Castellani, and A. Vinciarelli, "Infinite latent feature selection: A probabilistic latent graph-based ranking approach," *IEEE Int. Conf. Comput. Vision*, 2017, pp. 1398–1406.

[51] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, and X. Zhou, "L_{21}-norm regularized discriminative feature selection for unsupervised learning," in *Proc. 22nd Int. Joint Conf. Artif. Intell.*, 2011, pp. 1589–1594.

[52] F. Nie, H. Huang, X. Cai, and C. Ding, "Efficient and robust feature selection via joint $l_{2,1}$-norms minimization," *Adv. Neural. Inform. Process. System*, 2010, pp. 1813–1821.

[53] W. Jitkrittum, H. Hachiya, and M. Sugiyama. "Feature selection via L1-penalized squared-loss mutual information," *Ieice Trans. Inf. & Syst.*, 2012, vol. E96-D, no. 7, pp. 1513–1524.

[54] D. Greene, and P. Cunningham, "Constraint selection by committee: An ensemble approach to identifying informative constraints for semi-supervised clustering," *Eur. Conf. Mach. Learn.*, 2007, pp. 140–151.

[55] S. Xiong, J. Azimi, and X. Z. Fern, "Active learning of constraints for semi-supervised clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 1, pp. 43–54, 2014.

[56] S. Abdull, M. Diykh, R. L. Laft, K. Saleh, and R. C. Deo, "Sleep EEG signal analysis based on correlation graph similarity coupled with an ensemble extreme machine learning algorithm,"*Expert Syst. Appl.*, vol. 138, pp. 1–15, 2019.

[57] A. R. Hassan and M. I. H. Bhuiyan, "Automated identification of sleep states from EEG signals by means of ensemble empirical mode decomposition and random under sampling boosting," *Comput. Methods. Program. Biomed.*, vol. 140, pp. 201–210, 2017.

[58] R. Sharma, R. B. Pachori, and A. Upadhyay, "Automatic sleep stages classification based on iterative filtering of electroencephalogram signals," *Neural Comput. Appl.*, vol. 28, pp. 2959–2978, 2017.