Expert Systems with Applications 42 (2015) 7825–7830

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Fast and accurate PLS-based classification of EEG sleep using single channel data

Temel Kayikcioglu, Masoud Maleki *, Kubra Eroglu

Karadeniz Technical University, Faculty of Engineering, Department of Electrical and Electronics Engineering, 61080 Trabzon, Turkey

ARTICLE INFO

Article history:
Available online 10 June 2015

Keywords:
Partial least squares regression
Auto-regressive model
Electroencephalograph
k-Nearest neighborhood
Bayes
Linear discriminant classifier

ABSTRACT

Since speed of classification is important to real-time applications, this study proposed fast classification of sleep and wake stages using a single electroencephalograph (EEG) channel. Changes in the sleep and wake stages are accompanied by changes in the frequency spectrum of the EEG signals; so, the features extracted from the 5-s epoch of the EEG using auto-regressive (AR) coefficients were used to represent EEG signals of different sleep and wake stages. The proposed fast classification method was based on partial least squares regression (PLS), which was used to classify these features by finding an optimum beta using K-fold cross validation. The Physionet database was used to confirm accuracy and speed of the proposed classification system. This system could be used in real-time implementations because of its high classification rate, speed and capability to be implemented on hardware owing to be very comfortable. Finally, results of the PLS were compared with those of other classifiers such as *k*-nearest neighborhood (*k*-NN), linear discriminant classifier (LDC) and Bayes. We achieved 91% classification accuracy by selecting PLS as the classifier. These comparisons revealed that the proposed algorithm could recognize an emergency situation in less than 1 s with high accuracy.

1. Introduction

1.1. Background

Visual sleep scoring is a difficult process because of requiring a great deal of time and being a subjective procedure. In response to these challenges, automatic sleep-staging methods based on multichannel signals, including EEG, EMG and EOG (Kuwahara et al., 1988; Park, Park, & Jeong, 2000; Schaltenbrand et al., 1996; Smith, Negin, & Nevis, 1969; Smith and Karacan, 1971), have been developed. Two important items of sleep scoring are feature extraction, which helps researchers to analyze recording epoch, and classification, which helps researchers to recognize sleep stage of the epoch. A few features that adhere to the Rechtschaffen and Kales (R&K) standard have been proposed for sleep staging, which include alpha ratio (Agarwal and Gotman, 2001), spindle ratio (Duman, Erdamar, Erogul, Telatar, & Yetkin, 2009) and SWS ratio (Berthomier, Prado, & Benoit, 1999). Spectral power, power ratio and spectral frequency (Schaltenbrand et al., 1996) have been also used in previous studies. In addition, many methods have been proposed for classification, among which linear discriminant analysis (LDA) (Šušmáková & Krakovská, 2008), artificial neural network (Schaltenbrand et al., 1996), fuzzy system (Berthomier et al., 2007) and decision tree (Anderer et al., 2005) can be mentioned. Success of these methods has been in the range of 80–85%. One recent study (Sheng-Fu, Kuo Hu, Pan, & Wanga, 2012) proposed an automatic sleep-scoring method that combined multi scale entropy (MSE) and autoregressive models for a single-channel EEG. This work also recommended comparatively assessing performance of the method with the manual scoring based on full polysomnograms. Indeed, EEG data have been used for sleep scoring; but, using a system that is fast, accurate and comfortable with an implemented algorithm would be even more beneficial. Most of the previously proposed approaches are not suitable for implementation in real-time systems and many of them are not comfortable for the subjects. Some of the reasons for these shortcomings are low accuracy, infeasibility for hardware implementation, computational complexities and lack of generalization.

The objective of the present analysis was to test and compare performance of the PLS algorithm for sleep scoring with a single-source EEG (a single electrode) to test its feasibility in future works. This article is organized as follows. The following section presents sleep and sleep frequencies. Section 2 introduces the methods. Data acquisition, feature extraction and classifications

* Corresponding author. Mobile: +90 538 5787035.
E-mail addresses: tkayikci@ktu.edu.tr (T. Kayikcioglu), masoud.maleki1361@yahoo.com (M. Maleki), eroglu.kubra@gmail.com (K. Eroglu).

http://dx.doi.org/10.1016/j.eswa.2015.06.010
0957-4174/

are also described in this section. Finally, in Sections 3 and 4, the results and conclusions are respectively presented.

1.2. Sleep

The standard (R&K) for sleep stage classification defines two groups of stages for determining sleep depth. The first stage is non-rapid eye movement (NREM) stage, which is then sub-divided to four stages. Rapid eye movement (REM) stage is the second stage, which is characterized by high ocular activity in the EOG recordings. Four stages of NREM sleep are called Stage 1, Stage 2, Stage 3 and Stage 4. Recently, Stages 3 and 4 have been combined to form new slow-wave sleep stage (SWS) because they exhibit many similar characteristics.

1.3. Sleep frequencies

EEG rhythms are closely related to sleep and wake stages. Characteristics and patterns of the EEG recordings associated with the wake stage and various sleep stages are: alpha-band (8–12 Hz with 20_60 micro volt _ Amplitude) in wake, stages 1 and REM, beta-band (13–49 Hz with 2_20 micro volt _ Amplitude) in wake and theta-band (4–7 Hz with 50_75 micro volt _ Amplitude) in Stages 1, 2, 3 and 4 and REM and delta-band (0–4 Hz with 75 micro volt _ Amplitude) in Stages 3 and 4.

2. Material and methods

2.1. Dataset acquisition

The database used in this study was provided by Physionet (http://www.physionet.org/cgi-bin/atm/ATM). EEG signals were obtained from seven subjects ranging from healthy to abnormal. These subjects included Caucasian males and females (21–35 years old) who were neither on prescription drugs nor on recreational drugs at the time of the study. Sleep EEG for 80 h was extracted from the recordings and sampled at 100 Hz. Format of the dataset was an EDF (European Define Format) so that it could be downloaded. The 10–20 standard electrode placement system was used for the EEG recordings, which contained horizontal EOG, Fpz-Cz EEG and Pz-Oz EEG, each one sampled at 100 Hz. The process of sleep scoring involves identifying EEG signal epochs according to the sleep stage using a graphical plot called a hypnogram, which shows the sleep profile. Hypnograms were manually scored according to the R&K scale based on the Fpz-Cz and Pz-Oz EEGs and then classified into the following stages: waking, NREM 1, NREM 2, NREM 3, NREM 4, REM and movement time (M). One of the hypnograms is shown in Fig. 1. The epochs were pre-processed by filtering all of the data. This filtering step eliminated unwanted artefacts from the EEG data and enhanced their accuracy (http://www.physionet.org/cgi-bin/atm/ATM).

2.2. System design to classification of EEG sleep

Fig. 2 shows the flowchart of the proposed classification of EEG sleep method that includes three parts: (1) pre-processing; (2) feature extraction; and (3) classification. Each of these parts will be explained in the following subsections.

2.3. Pre-processing

The alternations in the EEG signals across the sleep stages are very delicate and therefore require advanced signal processing techniques to extract the features. EEG has different specific frequency components, some of which contain this discriminative information, which includes energy of the delta, theta, alpha and beta bands. This energy is important for classifying different brain states. In this study, the alpha, theta and beta bands were used to classify the sleep and wake stages.

First, the epochs were normalized between $[-1\,1]$ so that they all possessed similar conditions. In the next step, the epochs were filtered by three Butterworth band pass filters in the order of ten for the alpha-band (8–12 Hz), in the order of eleven for the beta-band (13–49 Hz) and in the order of eight for the theta-band (4–8 Hz). After the filtration, all three types of data were ready for any epoch and they were placed as input to the AR model in this format.

2.4. Feature extraction

2.4.1. Auto-regressive coefficients

AR model is a powerful and useful tool for signal modeling. In this model, each sample of a given signal is considered a prediction of the previous weighted samples of that signal. The number of coefficients determines the model order. In this paper, autoregressive coefficients were estimated with Burg method (Stoica and Moses, 1997). The Burg method fits the p'th order AR model to the input signal, x, by minimizing (least squares) the forward and backward prediction errors while constraining AR coefficient, a_i, to satisfy the Levinson–Durbin recursion. Eq. (1) shows the AR model.

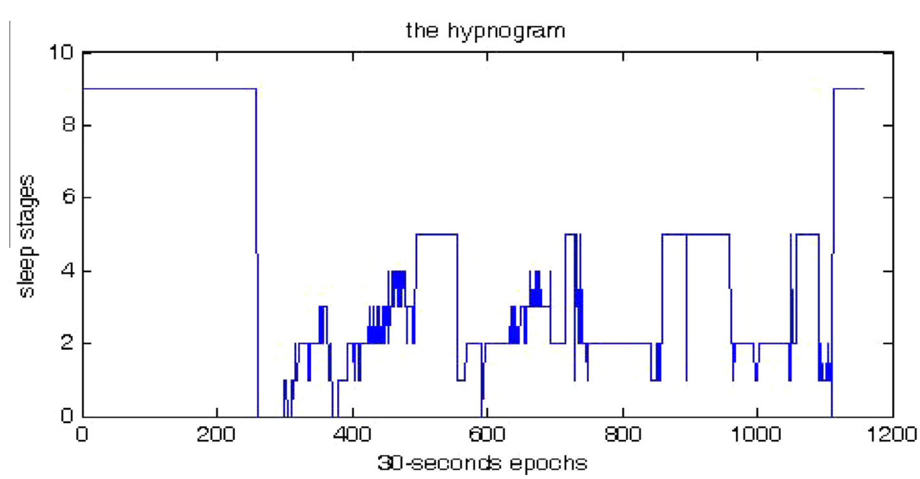

Fig. 1. Hypnogram of a subject; stages 0, 1, 2, 3, 4, and 5 are waking, NREM 1, NREM 2, NREM 3, NREM 4, and REM, respectively.

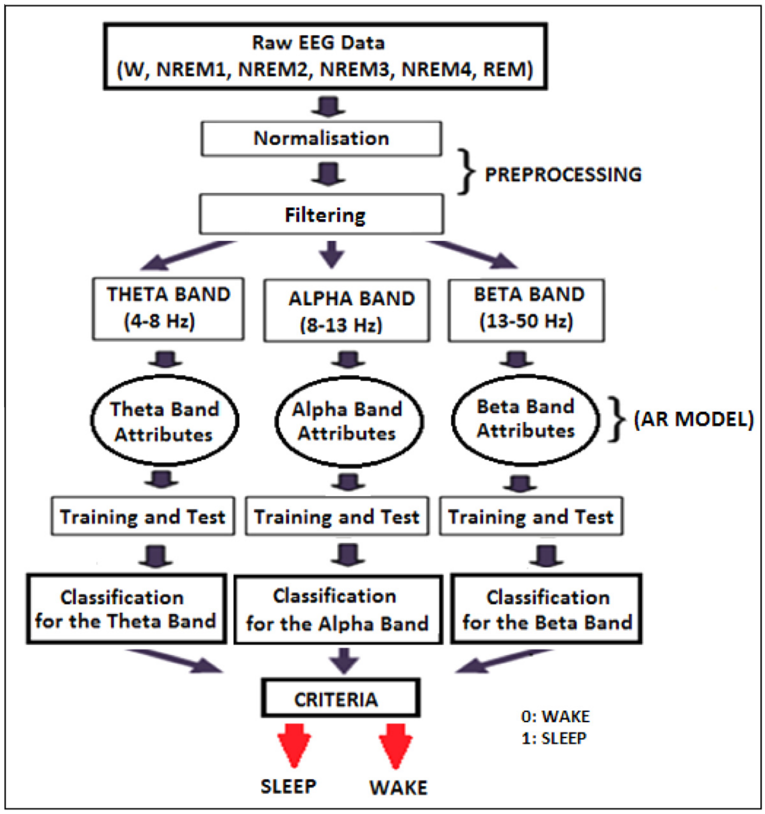

Fig. 2. Flow chart of design system.

$$x(t) = -\sum_{i=1}^{p} a_i x(t-i) + e(t) \tag{1}$$

In this paper, order of the AR model was 22 and inputs of the AR model were the theta-band, alpha-band and beta-band signals that were separately extracted by eighth-order, tenth-order and eleventh-order Butterworth band-pass filter during pre-processing. 22 AR coefficients were the whole of features in this method, which meant that the present feature vector had 22 dimensions for any epoch in any band.

2.5. Classification

2.5.1. PLS algorithm

Partial least squares (PLS) is an extended class of methods for modeling relations between sets of observed variables by means of latent variables. By encoding the class membership in a suitable indicator matrix, PLS can also be applied to classification problems. To build a typical model, the number of latent variables should be carefully selected. A metric that is frequently used by chemometricians for determining the number of latent variables is that of Wold's R criterion while a number of statisticians have advocated use of Akaike information criterion (AIC) more recently (Li, Morris, & Martin, 2002).

Data matrix of the process variables is X_{N*M} and data matrix of the quality variables is Y_{N*K}. Data matrixes are recorded for N time points. A number of latent variables is make by linear PLS, say t_j and u_j ($j = 1,\ldots,$A) where A is the number of latent variables and then develop a linear regression model between t_j and u_j:

$$u_j = b_j t_j + e_j \quad (j = 1, \ldots, \mathrm{A}) \tag{2}$$

where e_j is a vector of errors and b_j is an unknown parameter estimated by

$$b_j^{\wedge} = \left(t_j^T t_j\right)^{-1} t_j^T u_j \tag{3}$$

The latent variables are computed by $u_j = Y_j q_j$, where both w_j and q_j have a unit length and are determined by maximizing the covariance between t_j and u_j. Then, $X_{j+1} = X_j - t_j p_j^T$, where $X_1 = X$ and $p_j = X_j^T t_j / (t_j^T t_j)$, and $Y_{j+1} = Y_j - b_j^{\wedge} t_j q_j^T$, where $Y_1 = Y$.

If $u_j^{\wedge} = b_j^{\wedge} t_j$ be is prediction of u_j, matrices X and Y can be separated into simpler compounds as sum of the following outer products:

$$X = \sum_{j=1}^{A} t_j p_j^T + E \quad \text{and} \quad Y = \sum_{j=1}^{A} u_j^{\wedge} q_j^T + F \tag{4}$$

where after extracting the first A pairs of latent variables, E and F are remainders of X and Y (Li et al., 2002).

2.5.2. k-NN algorithm

k-NN is one of the oldest, fastest and easiest algorithms for implementation among the existing classification algorithms. In this algorithm, k (i.e. the nearest neighbor to the sample) is first determined. Then, the label that is in maximum between these neighbors is diagnosed. Thus, the sample is labeled with its maximum label. In binary classification problems, it is beneficial to use odd numbers for k because they do not cause any problems for researchers while deciding upon a label. Moreover, the query instance compares all training samples; so, k-NN encounters a high response time. It is worth mentioning that Euclidean distance method is commonly used to calculate the nearest neighbors to the sample.

2.5.3. Bayes algorithm

Bayes' theorem is named after Thomas Bayes, an eighteenth-century British mathematician and minister. He conducted some of the earliest works in probability and decision theory. Bayes classification is a statistical classification method which can predict class membership probabilities such as the probability by which a given tuple belongs to a particular class.

X is a data tuple which is described by measurements that are based on a set of *n* attributes $x \equiv [x(1), x(2), \ldots, x(l)]$. It also comprises the corresponding feature vector. The number of possible classes is equal to *C*, that is, $[w_1, w_c]$. The aim is to determine the probability $P(w_i|x)$ by which tuple *x* belongs to class *C* based on the attribute description of *x*. $P(w_i|x)$ is defined as the probability that class w_i holds given the data tuple *x*. $P(w_i|x)$ is the posterior probability that w_i is conditioned on *x* and $P(w_i)$ is prior probability of w_i. Bayes' Theorem provides a way for calculating posterior probability, $P(w_i|x)$, from $P(w_i)$, $P(x|w_i)$ and $P(x)$. $P(x)$ is probability density function (PDF) of *x*. Bayes theory states that:

$$P(w_i|x) = P(x|w_i)P(w_i)/P(x) \tag{5}$$

According to the Bayes decision theory, *x* is assigned to class w_i if:

$$P(w_i|x) > P(w_j|x), \quad \forall j \neq 1 \tag{6}$$

One of the most useful and commonly used is the use of discriminant function for expressing the pattern classifiers. The *ln* of both sides of expressed in the Eq. (5) is taken to determine the discriminant function.

$$g_i(x) = ln\ p(\underline{x}|w_i) + ln\ P(w_i) \tag{7}$$

$$g_i(x) > g_j(x), \quad j \neq i \tag{8}$$

According to the Eq. (8) discriminant function which belongs to the class is large, the data will be included in that class.

The above description is for two class. Therefore, the singular discriminant function will be in this form;

$$g(x) \equiv g_1(x) - g_2(x) \tag{9}$$

If this expression obtained as $g(x) > 0$ for *x* data, *w*1 class, if $g(x) < 0$, *w*2 class is decided.

2.5.4. LDC algorithm

LDC operates on two classes based on the hypothesis that both classes are under normal distribution with equal covariance matrices. The separating hyper-plane is obtained by finding projection of the labeled training data that maximizes the distance between means of the two classes and minimizes the interclass variance. The main task is to solve the following problem:

$$y = w^T x + w_0 \tag{10}$$

where *x* is feature vector. The vectors *w* and *w*0 are determined by maximizing the interclass means and minimizing the interclass variance, respectively. LDC classifier is more robust than the *k*-NN and SVM algorithms because it has only limited flexibility (less free parameters to tune) and is less prone to over-fitting (Muller, Anderson, & Birch, 2003).

3. Results

In this study, EEG data analysis, signal pre-processing and classification were implemented by the scripts running in MATLAB (R2012b). The dataset contained seven objects. For any object, the following algorithm was used.

After pre-processing, the three kinds of AR coefficients (for the three bands) were calculated for any epoch.

Classification of 30-s epochs, 15-s epochs, 10-s epochs and 5-s epochs was separately done in three bands. Selection of these epochs is described in Table 1. For example, 120 and 60 epochs of waking and 12 epochs for each stage of sleep were randomly selected from NREM 1–4 to REM for 30 s. The selected epochs were

Table 1
Description of the selection of the data set.

30-s epochs	120 epochs	60 for waking, 12 for any stage of sleep
15-s epochs	200 epochs	100 for waking, 20 for any stage of sleep
10-s epochs	240 epochs	120 for waking, 24 for any stage of sleep
5-s epochs	600 epochs	300 for waking, 50 for any stage of sleep

Table 2
Estimated label based on criteria.

	Theta band	Alpha band	Beta band	Estimated label based on criteria
Epoch label	0	0	0	***0***
	0	0	1	***0***
	0	1	0	***0***
	0	1	1	***1***
	1	0	0	***0***
	1	0	1	***1***
	1	1	0	***1***
	1	1	1	***1***

Table 3
Results of PLS classifier for Pz-Cz and Fpz-Cz channels.

PLS algorithm	30 s 60 train-60 test	15 s 100 train-100 test	10 s 120 train-120 test	5 s 300 train-300 test
	Mean ± std	Mean ± std	Mean ± std	Mean ± std
Pz-Cz				
Object.1	92.75 ± 4.0	93.15 ± 3.06	93.16 ± 3.0	91.38 ± 1.6
Object.2	94.91 ± 3.1	95.3 ± 2.31	94.58 ± 3.1	94.15 ± 1.7
Object.3	96.58 ± 2.7	96.05 ± 1.7	96.04 ± 2.3	96.28 ± 0.94
Object.4	89.58 ± 3.81	86.70 ± 2.9	86.04 ± 3.4	84.31 ± 2.07
Object.5	90.91 ± 3.9	92.75 ± 2.5	92.12 ± 2.3	91.45 ± 1.4
Object.6	88.66 ± 3.3	90.85 ± 2.8	92.62 ± 2.1	91.86 ± 1.5
Object.7	92.25 ± 2.71	92.5 ± 1.9	91.41 ± 2.4	91 ± 1.44
Fpz-Cz				
Object.1	95.41 ± 2.6	95.90 ± 1,7	95.66 ± 1.78	95.23 ± 1.1
Object.2	93.66 ± 2.2	90.90 ± 2.71	88.5 ± 2.06	87.43 ± 2.08
Object.3	91 ± 3.6	92.90 ± 1.9	92.66 ± 2.2	93.18 ± 1.31
Object.4	88.88 ± 4.05	85.10 ± 2.5	83.7 ± 2.04	81.46 ± 1.32
Object.5	89.41 ± 4.4	90.20 ± 2.85	88.20 ± 2.4	87.36 ± 1.4
Object.6	89.16 ± 3.6	91.15 ± 2.32	89.91 ± 1.9	89.45 ± 1.51
Object.7	85.66 ± 3.83	88.55 ± 3.2	89.12 ± 3.6	89.60 ± 1.7

Table 4
Results of *k*-NN classifier for Pz-Cz and Fpz-Cz channels.

k-NN algorithm	30 s 60 train-60 test	15 s 100 train-100 test	10 s 120 train-120 test	5 s 300 train-300 test
	Mean ± std	Mean ± std	Mean ± std	Mean ± std
Pz-Cz				
Object.1	84.1 ± 3.4	86.4 ± 2.2	86 ± 2.9	85.3 ± 2.3
Object.2	96 ± 2.8	95.4 ± 2.3	94.8 ± 1.6	93.2 ± 1.3
Object.3	94.8 ± 2.7	93.5 ± 2.6	89.8 ± 3.2	91.8 ± 1.09
Object.4	94.8 ± 3.2	94.9 ± 1.7	95.3 ± 2.3	96.4 ± 1.04
Object.5	84.6 ± 3.9	84.6 ± 3.3	83.8 ± 3.7	94.4 ± 1.9
Object.6	91.1 ± 3.4	90.6 ± 2.09	88.8 ± 2.1	86.9 ± 1.6
Object.7	84.1 ± 3.4	85.2 ± 2.8	80.6 ± 2.6	79.6 ± 2.3
Fpz-Cz				
Object.1	92.83 ± 2.94	95.30 ± 2.05	94.5 ± 2.55	84.9 ± 2.6
Object.2	77.5 ± 5.45	78.40 ± 4.92	77.80 ± 4.32	92.7 ± 1.85
Object.3	89.33 ± 3.35	92 ± 2.58	91.5 ± 2.68	91.3 ± 1.9
Object.4	83.66 ± 3.4	82.70 ± 2.66	77.83 ± 4.76	95.1 ± 1.5
Object.5	90 ± 3.2	85.7 ± 2.31	85.91 ± 3.5	93.0 ± 1.4
Object.6	88.5 ± 3.08	87.6 ± 4.1	85.5 ± 3.9	85.5 ± 1.6
Object.7	81.66 ± 3.33	84.5 ± 2.63	85.5 ± 2.0	79.1 ± 2.5

Table 5
Results of Bayes classifier for Pz-Cz and Fpz-Cz channels.

Bayes algorithm	30 s 60 train-60 test	15 s 100 train-100 test	10 s 120 train-120 test	5 s 300 train-300 test
	Mean ± std	Mean ± std	Mean ± std	Mean ± std
Pz-Cz				
Object.1	80.5 ± 2.9	83.8 ± 3	83.8 ± 4.5	84.7 ± 3.1
Object.2	81.3 ± 11.3	86.8 ± 9.2	86.5 ± 8.7	89.8 ± 2.3
Object.3	72.1 ± 19.3	88.3 ± 6	84.5 ± 9.3	88.9 ± 6.6
Object.4	64.7 ± 14.9	86.2 ± 14.4	84.4 ± 13.9	90.5 ± 10.1
Object.5	78.5 ± 7.9	85.8 ± 3.6	85.8 ± 3.6	85.8 ± 5.3
Object.6	81.7 ± 4.2	84.4 ± 5.6	83.4 ± 4.5	80.1 ± 3.16
Object.7	80.5 ± 2.9	82.9 ± 3.9	79.8 ± 5.2	81.7 ± 3
Fpz-Cz				
Object.1	81.6 ± 2.6	84.8 ± 3.5	81.8 ± 5	82.7 ± 3.5
Object.2	80.7 ± 11.5	85.9 ± 9.2	85.5 ± 4.5	88.8 ± 4.3
Object.3	73.9 ± 19.8	86.3 ± 6.6	84.5 ± 5.3	88.9 ± 5.6
Object.4	65.7 ± 14.2	85.7 ± 14.4	83.4 ± 10.6	91.5 ± 1.1
Object.5	78.8 ± 7.9	84.7 ± 2.6	86.8 ± 6.6	82.8 ± 6.3
Object.6	80.7 ± 4.0	82.8 ± 4.6	82.4 ± 6.5	81.1 ± 3.9.6
Object.7	81.9 ± 2.0	82.5 ± 4.9	80.3 ± 5.5	80.7 ± 2.3

randomly divided into the training and testing data. Thus, 50% of the samples was considered the training set and the other 50% was considered the testing data.

Using the K-fold cross validation beta, *k*, and the discriminant function for PLS, *k*-NN and Bayes algorithms could be calculated, respectively. K-fold cross-validation (K-FCV) is one of the most frequently adopted criteria for assessing performance of a model and selecting a hypothesis within a class. The advantage of this method over the simple training–testing data splitting is its repeated use of all available data for both building and testing a learning machine. Therefore, K-FCV reduces risk of (un)lucky splitting. In K-FCV, the dataset is randomly split into K subsets of equal size and this step is repeated K times. Each time, one of the K subsets is used as a validation set and other $K-1$ subsets are put together to form pre-training. A common problem is determining the number of folds, to which the training set should be divided. In this paper, K was selected to be 10.

At the end of the classification, the result was evident for each band of any epoch according to a decision mechanism. This decision mechanism is summarized in Table 2. In this table, the final decision is called "estimated label based on criteria". It takes the value 0 if estimated epoch label for two band (or three band) is 0 and is considered to be "wake". Also it takes the value 1 if estimated epoch label for two band (or three band) is 1 and is considered to be "sleep". The method is applied for any epoch.

To confirm the results, this algorithm was performed 20 times on different epochs and each result was separately averaged for each subject. In this review, EEG signal of the Fpz-Cz and Pz-Cz channels was separately analyzed. Tables 3–6 show results of each channel for the four classifiers. These results indicated that the Pz-Cz channel had better accuracy than the Fpz-Cz channel. In addition, PLS algorithm increased accuracy of the Pz-Cz channel.

The training and testing times of the applied classifiers were calculated. All the runtime experiments were checked on a PC with an Intel Pentium® 5 processor with 2.67 GHz and 4 GB RAM. Table 7 shows speed of these classifiers for 60 epochs. Each epoch lasted for 30 s. The average time taken to train PLS algorithm was about 2.3 CPU sec for the entire training set. For the 60 epochs in the testing set, the average test time of PLS and LDC algorithms was approximately 0.4 and 0.5 s, respectively. Therefore, PLS has better training and testing time.

Table 6
Results of LDC classifier for Pz-Cz and Fpz-Cz channels.

LDC algorithm	30 s 60 train-60 test	15 s 100 train-100 test	10 s 120 train-120 test	5 s 300 train-300 test
	Mean ± std	Mean ± std	Mean ± std	Mean ± std
Pz-Cz				
Object.1	92.5 ± 3.7	90.75 ± 3.5	91.54 ± 2.29	89.16 ± 1.9
Object.2	96.16 ± 3.11	96.15 ± 2.8	95.58 ± 1.81	94.53 ± 1.2
Object.3	97.16 ± 2.23	97.35 ± 1.9	96.70 ± 1.4	96.68 ± 0.96
Object.4	87.91 ± 5.40	85.85 ± 3.39	83.83 ± 3.9	81.56 ± 2.04
Object.5	91 ± 4.2	92.35 ± 2.25	90.95 ± 2.7	89.15 ± 1.73
Object.6	88.33 ± 3.6	87.60 ± 2.9	87.12 ± 2.23	86.15 ± 2.3
Object.7	91.83 ± 2.75	91.85 ± 2.9	89.70 ± 1.9	88.83 ± 1.9
Fpz-Cz				
Object.1	96.5 ± 2.8	96.45 ± 1.8	95.37 ± 1.80	95.68 ± 1.18
Object.2	92.08 ± 3.2	92.55 ± 2.7	91.20 ± 1.8	88.05 ± 2.38
Object.3	92.08 ± 3.32	95.60 ± 2.68	93.83 ± 2.31	93.65 ± 0.95
Object.4	88.58 ± 5.3	85.25 ± 4.15	84.66 ± 2.43	82.20 ± 2.25
Object.5	89.5 ± 3.2	90.35 ± 1.98	90.54 ± 2.9	87.51 ± 1.7
Object.6	89.91 ± 4.4	91.95 ± 3.2	90.66 ± 2.2	89.46 ± 1.2
Object.7	88.75 ± 3.7	89.35 ± 3.9	89.5 ± 3.6	90.10 ± 1.98

Table 7
Speed of four classifiers for 60 epochs.

Data/algorithms	PLS	*k*-NN	Bayes	LDC
TRAIN	~2.3 s	~220 s	~3.5 s	~2.4 s
TEST	~0.4 s	~6.75 s	~1.2 s	~0.5 s

4. Conclusion

This study developed a novel PLS-based approach for automated sleep scoring by analyzing a single EEG channel. The dataset was provided by Physionet. The alternative indicators were extracted from the three bands of the EEG signal. Gamma-band of the EEG signal was not used in the present study. AR coefficient-based feature extraction was applied to classify the sleep and wake stages of the EEG signal using PLS as a high-speed classifier. The PLS classifier performed well while classifying stages of sleep and wakefulness at a high speed. Moreover, speed and accuracy of PLS were compared with those of *k*-NN, Bayes and LDC classifiers. The proposed algorithm could be easily implemented on the hardware for real-time applications. Using only one channel of EEG was also very important because of being very comfortable for the subjects.

The low-cost electroencephalograph with minimum electrodes could be also used to evaluate driver fatigue with high accuracy and speed. Drivers, especially those who usually drive for long periods of time, will feel more comfortable with the minimum electrodes placed on their heads. Achieving this aim without putting pressure on the psychological factors that are associated with fatigue (e.g. anxiety, mood, temperament and personalities) would greatly benefit field of sleep study.

References

Agarwal, R., & Gotman, J. (2001). Computer-assisted sleep staging. *IEEE Transactions on Biomedical Engineering, 48*(12), 1412–1423.

Anderer, P., Gruber, G., Parapatics, S., Woertz, M., Miazhynskaia, T., Klosch, G., et al. (2005). An E-health solution for automatic sleep classification according to Rechtschaffen and Kales: Validation study of the Somnolyzer 24 × 7 utilizing the Siesta database. *Neuropsychobiology, 51*(3), 115–133.

Berthomier, C., Drouot, X., Herman-Stoïca, M., Berthomier, P., Prado, J., Bokar-Thire, D., et al. (2007). Automatic analysis of single-channel sleep EEG: Validation in healthy individuals. *Sleep, 30*(11), 1587–1595.

Berthomier, C., Prado, J., & Benoit, O. (1999). Automatic sleep EEG analysis using filter banks. *Biomedical Sciences Instrumentation, 35*, 241–246.

Duman, F., Erdamar, A., Erogul, O., Telatar, Z., & Yetkin, S. (2009). Efficient sleep spindle detection algorithm with decision tree. *Expert Systems with Applications, 36*(6), 9980–9985.

Kuwahara, H., Higashi, H., Mizuki, Y., Matsunari, S., Tanaka, M., & Inanaga, K. (1988). Automatic real-time analysis of human sleep stages by an interval histogram method. *Electroencephalography and Clinical Neurophysiology, 70*(3), 220–229.

Li, B., Morris, J., & Martin, E. B. (2002). Model selection for partial least squares regression. *Chemometrics and Intelligent Laboratory Systems, 64*, 79–89.

Muller, K.-R., Anderson, C. W., & Birch, G. E. (2003). Linear and nonlinear methods for brain-computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering, 11*(2), 165–169.

Park, H. J., Park, K. S., & Jeong, D.U., 2000. Hybrid neural-network and rulebased expert system for automatic sleep stage scoring. In *Proceedings of 22nd annual EMBS international conference* (pp. 1316–1319).

Schaltenbrand, N., Lengelle, R., Toussaint, M., Luthringer, R., Carelli, G., Jacqmin, A., et al. (1996). Sleep stage scoring using the neural network model: Comparison between visual and automatic analysis in normal subjects and patients. *Sleep, 19*(1), 26–35.

Sheng-Fu, L., Kuo Hu, Y., Pan, Y., & Wanga, Y. (2012). Automatic stage scoring of single-channel sleep EEG by using multiscale entropy and autoregressive models. *IEEE Transactions on Instrumentation and Measurement, 61*(6).

Smith, J. D., & Karacan, I. (1971). EEG sleep stage scoring by an automatic hybrid system. *Electroencephalography and Clinical Neurophysiology, 31*(3), 231–237.

Smith, J. R., Negin, M., & Nevis, A. H. (1969). Automatic analysis of sleep electroencephalograms by hybrid computation. *IEEE Transactions on Systems Science Cybernetics, SSC-5*(4), 278–284.

Stoica, P., & Moses, R. L. (1997). *Introduction to spectral analysis*. Prentice Hall.

Šušmáková, K., & Krakovská, A. (2008). Discrimination ability of individual measures used in sleep stages classification. *Artificial Intelligence in Medicine, 44*(3), 261–277.