

# A Novel Approach for Lie Detection Based on F-Score and Extreme Learning Machine

Junfeng Gao<sup>1,4</sup>, Zhao Wang<sup>2</sup>, Yong Yang<sup>3</sup>, Wenjia Zhang<sup>1</sup>, Chunyi Tao<sup>1</sup>, Jinan Guan<sup>1\*</sup>, Nini Rao<sup>4\*</sup>

**1** College of Biomedical Engineering, South-Central University for Nationalities, Wuhan, People's Republic of China, **2** Department of Biomedical Engineering, Case Western Reserve University, Cleveland, Ohio, United States of America, **3** School of Information Technology, Jiangxi University of Finance and Economics, Nanchang, People's Republic of China, **4** School of Life Science and Technology, University of Electronic Science and Technology of China, Chengdu, People's Republic of China

## Abstract

A new machine learning method referred to as F-score\_ELM was proposed to classify the lying and truth-telling using the electroencephalogram (EEG) signals from 28 guilty and innocent subjects. Thirty-one features were extracted from the probe responses from these subjects. Then, a recently-developed classifier called extreme learning machine (ELM) was combined with F-score, a simple but effective feature selection method, to jointly optimize the number of the hidden nodes of ELM and the feature subset by a grid-searching training procedure. The method was compared to two classification models combining principal component analysis with back-propagation network and support vector machine classifiers. We thoroughly assessed the performance of these classification models including the training and testing time, sensitivity and specificity from the training and testing sets, as well as network size. The experimental results showed that the number of the hidden nodes can be effectively optimized by the proposed method. Also, F-score\_ELM obtained the best classification accuracy and required the shortest training and testing time.

**Citation:** Gao J, Wang Z, Yang Y, Zhang W, Tao C, et al. (2013) A Novel Approach for Lie Detection Based on F-Score and Extreme Learning Machine. *PLoS ONE* 8(6): e64704. doi:10.1371/journal.pone.0064704

**Editor:** Luis M. Martinez, CSIC-Univ Miguel Hernandez, Spain

**Received:** November 25, 2012; **Accepted:** April 17, 2013; **Published:** June 3, 2013

**Copyright:** © 2013 Gao et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Funding:** The work is supported by National Nature Science Foundation of China (No. 81271659, 61262034, 91120017 and 81171411), by the Science and Technology Research Project of the Education Department of Jiangxi Province (No. GJJ13302) and by the Special Fund for Basic Scientific Research of Central Colleges, South-Central University for Nationalities (No. CZY12011). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

**Competing Interests:** The authors have declared that no competing interests exist.

\* E-mail: raonini@uestc.edu.cn (NR); guanja@tom.com (JG)

## Introduction

Deception is an important social and legal behavior. The traditional method for detecting deception is based on polygraph measurements. In recent years, significant progress in neuroscience has inspired investigations on lie detection. A number of studies have used neurophysiological signals, such as Functional Magnetic Resonance Imaging (fMRI) and Event Related Potential (ERP) [1–3], to investigate lie detection [4]. An endogenous ERP component, P300 (P3), has been extensively investigated and successfully used in the detection of deception and malingering [5–7].

The widely used P3-based lie-detection methods can be roughly divided into three categories: bootstrapped amplitude difference (BAD), bootstrapped correlation difference (BCD) [8] and pattern recognition (PR) methods [4,9,10]. Compared with BAD and BCD, PR-based lie detection is a promising approach for two main reasons: 1) more physiological features can be extracted from raw P300 and 2) a variety of PR classifiers can be utilized to improve the accuracy of the lie detection. However, the adoption of PR classifiers for lie detection has not yet been widely reported. Davatzikos et al. [4] proposed a support vector machine (SVM)-based method to classify the patterns of brain activity (fMRI data) obtained during lying and truth-telling. Abootalebi et al. [9] used linear discrimination analysis (LDA) to identify P3 responses and obtained a higher detection rate (86%) than that obtained using BAD- and BCD-

based methods. SVM was used for the first time in the investigation of P3-based lie detection by Gao et al. [10]. Compared to fisher discrimination analysis (FDA) and back-propagation neural networks (BPNN), SVM classifier obtained the highest average classification accuracy (91.8%) between P3 responses from the guilty and non-P3 responses from the innocent.

In the current studies of EEG classification, there is a general trend to test various classifiers to ultimately obtain the highest classification accuracy possible [11,12]. Rooted in statistical learning theory, SVM classifier implements structural risk minimization and margin hyperplane maximization [13]. More importantly, the SVM can map the nonlinear separable data onto a high-dimension space, and hence classify the data linearly by using a technique of kernel function mapping. In the past 30 years, SVM classifier has demonstrated great advantages over most other classifiers in terms of classification accuracy and generalization power [4,11,14]. However, it should be noted that the time required to train the classification models should be considered, especially when the training data is substantial and the training procedure is complex. Taking this into account, SVM and gradient descent-based artificial neural network (ANN, e.g., BPNN) may be unsuitable and unsatisfactory due to their high computational cost [15,16]. Extreme learning machine (ELM), a single-layer feedforward network (SLFN)-based method, was proposed by Huang et al. [15] to overcome some inherent drawbacks of SVM and BPNN (complex and long parameter

training procedure). ELM randomly specifies the input weights and biases and then analytically calculates the output weights with the smallest norm. Hence, ELM tends to provide good generalization power at an extremely fast training speed [17,18]. During the past several years, ELM has drawn considerable attentions in many fields related to PR [19,20].

Some researchers have studied the performance of ELM in the classification of ERP. Liang et al. applied ELM for the first time to the classification of mental tasks using EEG signals [21]. Their results showed that ELM obtained similar classification accuracy with a training time that was 1–2 orders of magnitude shorter, compared with SVM and BPNN. ELM was successfully adopted by Shi et al. for EEG-based vigilance estimation [22]. Several ELM-based investigations on epileptic seizure detection have also demonstrated the promising performance of ELM in the classification of different EEG tasks [16,23,24]. To date, ELM has not been used to detect lying and to classify guilty and innocent subjects.

Feature selection plays an important role in the construction of a classification model. Chen demonstrated for the first time that feature selection strategies for SVM classification should be included [25]. Polat et al. classified medical datasets using a hybrid system of feature selection and several classifiers and obtained better performance compared with the methods that did not utilize feature selection [26]. Akay proposed a breast cancer diagnosis method which integrated SVM and F-score feature selection [27]. The experimental results showed that the hybrid method attained the higher classification accuracy compared with all other models without feature selection. To date, few researchers have conducted studies combining ELM with feature selection. Han et al. combined principal component analysis (PCA) and ELM to predict the postoperative survival time of patients who suffered from non-small cell lung cancer [24]. Their results showed that the CPU time with their proposed method was significantly less than that obtained with other classification models, such as BPNN and BPNN combined with PCA. In the area of EEG classification, no reported investigation has combined ELM with feature selection.

The number of hidden nodes ( $NHN$ ) in ELM is an important parameter that may affect the classification performance (the other one important parameter the activation function).  $NHN$  is usually randomly assigned in the basic ELM algorithm. Huang et al. found that for some special datasets, the generalization performance of ELM was very stable over a wide range of  $NHN$  [15]. However, Cao et al. indicated that the classification boundary may not be optimal when this number remains unchanged during the training procedure [28]. In addition, too many or too few  $NHN$  might lead to over-fitting or under-fitting [22]. Because this is one of the hottest issues related to the ELM research, a few methods were recently proposed to investigate this problem [29–33]. However, these improved ELM algorithms are relatively complicated for real application in classification system. Moreover, similar to SVM algorithm, ELM cannot directly obtain the feature importance. Finally, there exists a close relationship between the  $NHN$  and the dimensions of the feature space, which, however, was not stressed in these improved algorithms.

In this study, we combined ELM with feature selection to classify truth-telling and lying signals. In addition, we simultaneously optimized the feature subspace and the  $NHN$  in ELM. We hypothesize that this joint optimization strategy could not only further enhance the classification accuracy of lie detection, but also significantly decrease the training and testing times.

## Materials

### 1. Ethics Statement

The experiment was approved by Psychology Research Ethical Committee (PREC) of the College of Biomedical Engineering in South-Central University for Nationalities. Thirty-three healthy subjects (15 females, mean age of 22) were recruited from the university. The participants provided their written informed consent according to a human research protocol in this study.

### 2. Subjects and Experimental Protocol

The guilty knowledge test (GKT) [9] and three-stimulus protocol [10] were used in this study. The probe (P) stimuli consisted of some images or sound related to criminal acts, such as the weapon in the scene of the crime. The guilty is certainly familiar with these stimuli, whereas this is not the case for the innocent. The target (T) stimuli are known by all the subjects, but these are not related to the criminal acts. The irrelevant (I) stimuli are not known by all the subjects and are not related to criminal acts. All of the participants were randomly divided into a guilty group and an innocent group. Six different jewels were prepared, and their pictures served as the stimuli during the detection procedure. A safe containing one (for the innocent) or two (for the guilty) jewels was given to each subjects, who were told that only one examiner knew the contents in the safe. The subjects were instructed to open the safe and memorize the details of the object. All of the subjects were asked to write down the information of the objects in the safe, such as styles and colors. As the subjects stole the jewels, all of the researchers were asked to stay out.

We instructed the guilty steal one jewel and pocket the object, which served as a P stimulus, whereas the other one in the safe served as T stimulus and the remaining four pictures were I stimuli. The guilty group was instructed to press the “Yes” and “No” buttons when facing with T and I stimuli, respectively. With a P stimulus, they were asked to press the “No” button in an attempt to hide the stealing act. We told the guilty that they would earn 100 RMB if successfully concealed the identity of the probe stimuli during the experimental session. For the innocent, the object in the safe was a T stimulus, whereas the object stolen by guilty subjects served as a P stimulus in order to add comparability (although the object is indeed random in the remaining five pictures); the other four images were I stimuli. In contrast, the innocent group responded honestly to all of the stimuli.

### 3. EEG Data Acquisition

All of the subjects were seated in a chair, facing a video screen 1 m from their eyes. The stimuli were presented in a random order on the screen for a duration of 0.5 s at a random interval of 1.4–1.6 s. Each session lasted approximately 5 minutes with a 3-minute resting time. Each subject was instructed to participate in 3 sessions. The EEG was recorded on the following nine silver electrodes: C3, Cz, C4, P3, Pz, P4, O1, O2, and Oz from an International 10–20 system. The vertical EOG (VEOG) signals and the horizontal EOG (HEOG) signals were recorded. The EEG and EOG signals were passed through a Neuroscan Synamps Amplifier with a bandpass filter of 0.1–30 Hz, and digitized at 500 Hz. All of the electrodes were referenced to the right earlobe, and the electrode impedances were less than 2 k $\Omega$ . The artifact removal criterion was  $\pm 75 \mu\text{V}$ . The EEG data obtained from 5 of the subjects were excluded due to significant eye blinking and eye movement artifacts. Although none did, any subjects with a clicking error rate of more than 5% would be

excluded. Finally, EEG signals from 14 subjects in each group were further preprocessed.

#### 4. Preprocessing

During the experiment, if the subject did not provide a response or failed to give the right response by pressing the corresponding button, the corresponding EEG responses were first rejected by visual inspection. The resulting EEG data were then segmented into epoch datasets from 0.2 s before to 0.8 s after the stimulus onset. All of the trials were baseline-corrected based on the pre-stimulus interval.

All of the epoch datasets were further processed as follows. Within each subject, 30 single-trials from each type of stimulus and each site were pooled into one wave. Figure 1 shows the results from a randomly selected guilty subject and an innocent subject. The averaged waves at the Pz electrode from the guilty and innocent subjects are shown in Figure 1A and Figure 1B, respectively. Comparing the two subfigures, we can observe that there is significant P300 in the P responses (i.e., the response waves from the P stimuli) from the guilty subject, but no P300 in the P responses from the innocent subject. Furthermore, the brain topographies at the latency of 348 ms (see Figure 1A) and at the peak point of 316 ms (see Figure 1B) are shown in Figure 1C and Figure 1D, respectively. By comparing the two figures, we can observe that there exists a significant difference between the P responses on the Pz site from the two subjects. Similar to some early reports [34,35], all of the P responses on the Pz site were finally selected, and each of the 5 P responses for each subject were pooled into one average to enhance the signal-to-noise ratio (SNR) of the P300 [10,36]. Hence, there were approximately 300 P responses for each group of subjects. The P responses from the guilty subjects represent P3, whereas the responses from the innocent subjects represent non-P3.

### Methods

#### 1. Feature Extraction

Three groups of features based on time-domain, frequency-domain, and time-frequency domain features were extracted from each P response with the time varying from 0.2 to 1 s. Burg's method was used for spectrum estimation [14]. There were nine time- and frequency-domain features as follows: maximum amplitude  $V_{\max}$ , latency  $t_{\max}$ , latency/amplitude ratio  $R_{L/\Lambda}$ , minimum amplitude  $V_{\min}$ , peak-to-peak amplitude  $V_{\text{pp}}$ , positive area  $A_p$ , maximum frequency  $f_{\max}$ , mean frequency  $f_{\text{mean}}$ , and the power of the frequency band containing the P3  $A_{\text{P3}}$ . In this study, we used discrete wavelet transform (DWT) [11,37] to decompose each P response into seven sets of wavelet coefficients. The coefficient set corresponding to the first frequency band (0.1 to 3.9 Hz) was selected as the 22 wavelet features, which were denoted by  $W$  where  $i = 1, 2, \dots, 22$ . Please refer to our previous report for more details on the extracted features [36]. After the feature extraction, two feature sample sets (represent P3 and non-P3) were obtained with the class label 1 and  $-1$ , respectively. Each sample consisted of 31 feature values. Before the classification, all of the feature values were normalized to  $[-1, 1]$ .

#### 2. Feature Selection

The feature selection can help the original classification system achieve a better predictive performance and a lower computational cost by removing any redundant features. The F-score is a simple but effective technique for evaluating the discriminative power of each feature in the feature set. Chen proposed and combined this method with SVM to participate NIPS 2003

Feature Selection Challenge and was ranked third [38]. Recently, many researchers have successfully applied the combination of F-score with an SVM classifier to various classification tasks [25,26,27,39].

Given the  $i$ th feature vector  $\{x_{i1}, x_{i2}, \dots, x_{in_+}, \dots, x_{iN}\}$  with the number of positive instances  $n_+$  and the number of all the instances  $N$ , the  $F$ -score value of the  $i$ th feature is defined by

$$F(i) = \frac{(\bar{x}_i^{(+)} - \bar{x}_i)^2 + (\bar{x}_i^{(-)} - \bar{x}_i)^2}{\frac{1}{n_+ - 1} \sum_{k=1}^{n_+} (x_{ik} - \bar{x}_i^{(+)})^2 + \frac{1}{N - n_+ - 1} \sum_{k=n_+ + 1}^N (x_{ik} - \bar{x}_i^{(-)})^2}, \quad (1)$$

where  $\bar{x}_i^{(+)}$ ,  $\bar{x}_i^{(-)}$ , and  $\bar{x}_i$  are the average of the positive, negative, and whole samples, respectively, and  $x_{ik}$  is the  $k$ th feature value in the  $i$ th feature vector. The numerator indicates the discrimination between the positive and the negative sets, and the denominator is the sum of the deviation within each feature set. A larger the  $F$ -score value indicates that the feature has more discriminative power. We adopted the F-score method in this study due to its simplicity of its use in a lie detection system with real applications.

There are two main methods that are used to select the appropriate feature subset: the filter method [40] and the wrapper method [41,42]. Although there is higher computation cost associated with the wrapped method, many experimental results are in favor of the wrapper method for feature selection due to its good performance. Hence, we also used this method in this study.

For comparison, we adopted another popular method, principal component analysis (PCA), to select the features [43]. PCA extracts dominant features from the original input samples. The dominant features retain most of the information, both in the sense of maximum variance of the features and in the sense of minimum reconstruction error [44]. In this study, similarly to the F-score, PCA was combined with classifiers to identify the optimal feature set.

Given a set of  $N$  input samples  $\mathbf{x} = (x_1, x_2, \dots, x_t, \dots, x_N)$ , each of which has  $m$  dimensions  $x_t = (x_{t1}, x_{t2}, \dots, x_{tm})^T$ , PCA first solves an eigenvalue problem, i.e.,

$$\lambda_i u_i = S u_i \quad (i = 1, 2, \dots, m), \quad (2)$$

where  $S = \frac{1}{N} \sum_{t=1}^N x_t x_t^T$  is the sample covariance matrix, and  $\lambda_i$  is the corresponding eigenvalue of the eigenvector  $u_i = (u_{i1}, u_{i2}, \dots, u_{im})^T$ . After all of the  $\lambda_i$  are sorted in descending order, PCA uses the first  $d$  eigenvalues and their corresponding eigenvectors to project the original input samples  $\mathbf{x}$  into a  $d$ -dimensional space using the following linearly transform:

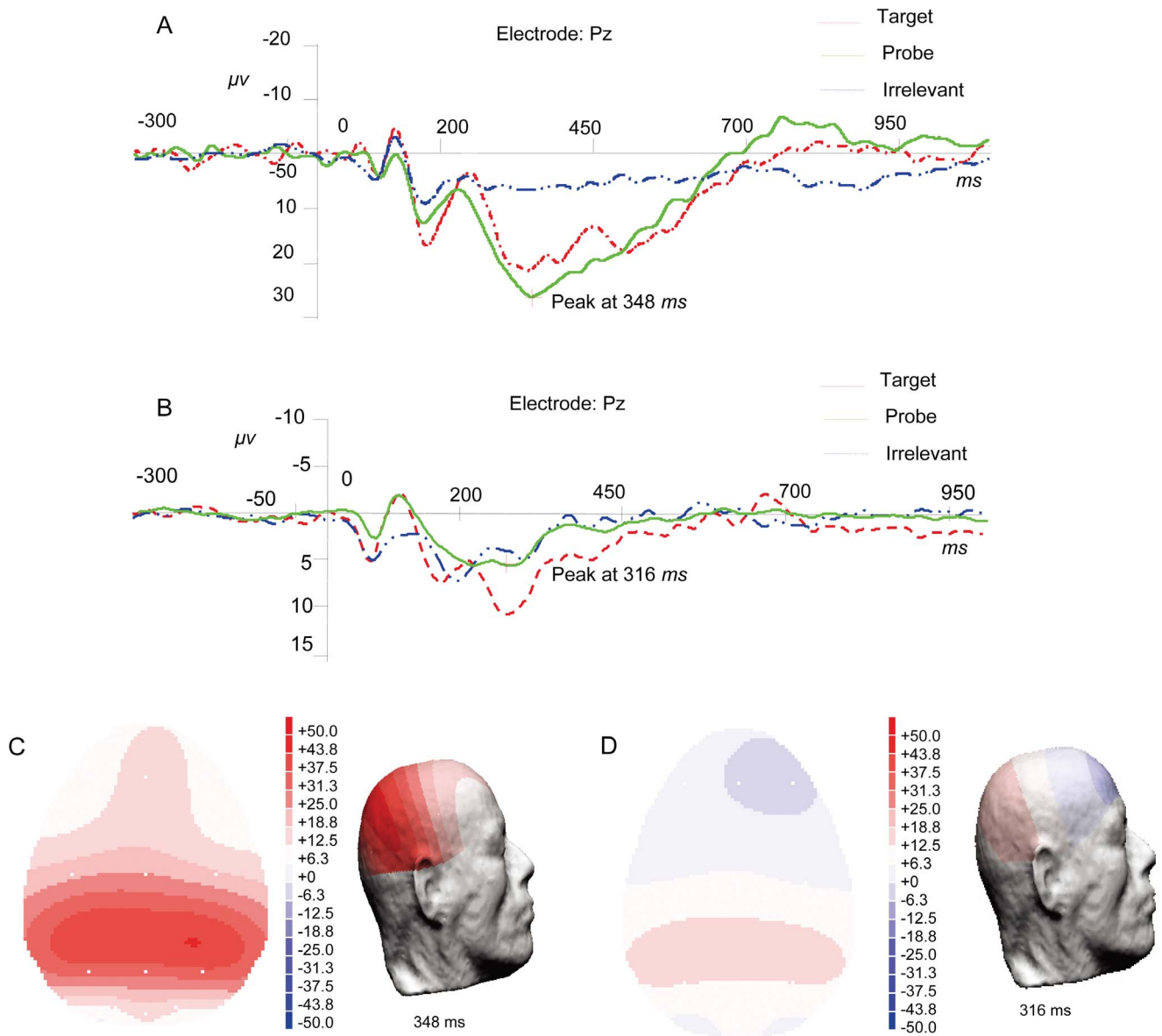
$$\mathbf{Y} = U_d^T \mathbf{X} \quad (d < m), \quad (3)$$

where  $U_d^T$  is a  $d \times m$  matrix, the  $i$ th row of which is the eigenvector  $u_i$ . Each feature vector of the new projection samples  $\mathbf{Y}$  is referred to as a principal component.

#### 3. Extreme Learning Machine

For comparison purposes, ELM, SVM, and BPNN were selected as the three types of representative machine learning-based classifiers to classify the P300 data for lie detection.

Given  $N$  different training instances  $(x_i, t_i)$ , where  $x_i = [x_{i1}, x_{i2}, \dots, x_{im}]^T \in \mathbb{R}^m$  and  $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in \mathbb{R}^m$ , we train a



**Figure 1. The preprocessing results of a guilty subject and an innocent subject.** 1A: Three averaged waves over the three kinds of stimuli respectively at Pz electrode from the guilty subject. 1B: The averaged waves over the three kinds of stimuli respectively at Pz electrode from the innocent subject. 1C: The brain topographies at the latency of 348 ms of the averaged P responses (the solid line in Figure 1A). 1D: The brain topographies at the peak point of 316 ms of the averaged P responses (the solid line in Figure 1B). doi:10.1371/journal.pone.0064704.g001

SLFN with  $K$  hidden nodes and an activation function  $g(x)$ , as shown in Figure 2. This network can be mathematically modeled as

$$\sum_{i=1}^K \beta_i g_i(x_j) = \sum_{i=1}^K \beta_i g(\mathbf{a}_i \cdot \mathbf{x}_j + b_i) = \mathbf{t}_j, \quad j = 1, \dots, N, \quad (4)$$

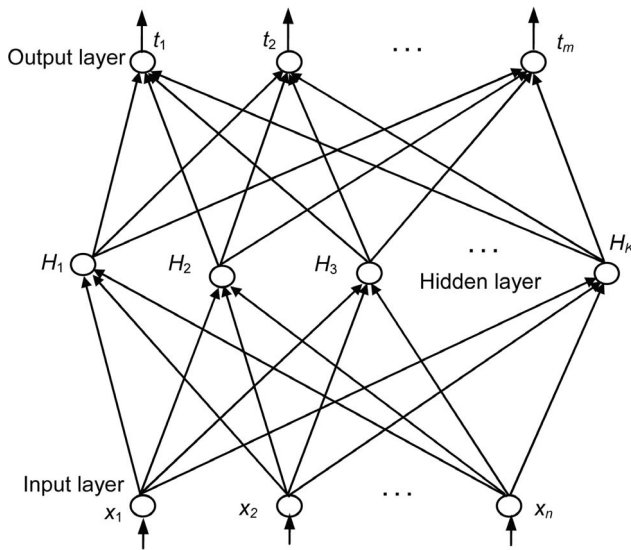
where  $\mathbf{a}_i = [a_{i1}, a_{i2}, \dots, a_{im}]^T$  denotes the weight vector connecting the  $i$ th hidden node and the  $n$  input nodes,  $b_i$  is the bias of the  $i$ th hidden node,  $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$  denotes the weight vector

connecting the  $i$ th hidden node and the  $m$  output nodes, and  $\mathbf{a}_i \cdot \mathbf{x}_j$  denotes the inner product of  $\mathbf{a}_i$  and  $\mathbf{x}_j$ .

The above  $N$  equations can be rewritten in a matrix form as

$$\mathbf{H}\beta = \mathbf{T}, \quad (5)$$

where



**Figure 2. SLFN with  $K$  hidden,  $n$  input and  $m$  output nodes.**  
doi:10.1371/journal.pone.0064704.g002

$$\mathbf{H} = \begin{bmatrix} g(\mathbf{a}_1 \cdot \mathbf{x}_1 + b_1) & \cdots & g(\mathbf{a}_K \cdot \mathbf{x}_1 + b_K) \\ \vdots & \cdots & \vdots \\ g(\mathbf{a}_1 \cdot \mathbf{x}_N + b_1) & \cdots & g(\mathbf{a}_K \cdot \mathbf{x}_N + b_K) \end{bmatrix}_{N \times K}, \quad (6)$$

$$\boldsymbol{\beta} = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_K^T \end{bmatrix}_{K \times m} \quad \text{and} \quad \mathbf{T} = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times m}.$$

$\mathbf{H}$  is called the hidden-layer output matrix, where the  $i$ th column is the output of the  $i$ th hidden node [26]. To learn the  $N$  instances for a SLFN, the conventional method is to find the solution set  $\mathbf{W}$ , including  $\mathbf{a}_i$ ,  $\boldsymbol{\beta}_i$  and  $b_i$ , by minimizing the following cost function:

$$E(\mathbf{W}) = \sum_{j=1}^N \left( \sum_{i=1}^K \beta_i g(\mathbf{a}_i \cdot \mathbf{x}_j + b_i) - t_j \right)^2. \quad (7)$$

Given an arbitrarily small value  $\varepsilon > 0$ , Huang et al. proved that if the input weights and the biases of the hidden nodes are randomly assigned and the activation function in the SLFN is infinitely differentiable, the SLFN can approximate the  $N$  training data with  $\varepsilon$  error, i.e.,  $\|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\| \leq \varepsilon$  [15]. In this case, the matrix  $\mathbf{H}$  has been randomly fixed. Hence, the training procedure of SLFN is equivalent to the identification of a least-squares (LS) solution of the linear system:

$$\|\mathbf{H}\hat{\boldsymbol{\beta}} - \mathbf{T}\| = \min_{\boldsymbol{\beta}} \|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\|, \quad (8)$$

where  $\hat{\boldsymbol{\beta}} = \mathbf{H}^\dagger \mathbf{T}$  is the LS solution of the above problem with the smallest norm, and  $\mathbf{H}^\dagger$  is the Moore-Penrose generalized inverse of  $\mathbf{H}$ . Bartlett [17] and Huang et al. [15] indicated that SLFNs with smaller output weights have a better generalization ability.

#### 4. The Proposed Method: F-score\_ELM

In this study, we combined the ELM methodology with a feature selection method for lie detection. There are two important problems for the proposed method: the choice of the optimal feature subset for F-score and the determination of the value of  $NHN$  for ELM.

Taking into account a lie detection system with real applications, the wrapper method mentioned previously should be more suitable for solving the first problem than the filter method because the feature subset was relatively fixed after the training procedure. With respect to the optimal  $NHN$ , we did not randomly assign but integrated the optimization of  $NHN$  into the selection of feature subset. The proposed method is referred to as F-score\_ELM.

Figure 3 presents the block diagram of F-score\_ELM using a grid-search technique [45] to jointly optimize the feature subset and the  $NHN$  in ELM. Let  $D$  denote the number of the originally extracted features, which equals 31 in this paper. The F-score\_ELM method consists of the following steps:

*Step 1:* Calculate the  $F$ -score values of the  $D$  feature vectors. Then, rearrange the feature vector set such that the first feature vector has the highest  $F$ -score, and the second vector has the second highest  $F$ -score, and so on. Let  $\mathbf{F}$  denote the new feature vector set, and initialize a feature subset, denoted by  $\mathbf{FS}$ , to be empty.

*Step 2:* Pick one feature vector with the highest  $F$ -score value from  $\mathbf{F}$ . Add the selected vector to the subset  $\mathbf{FS}$ . Set  $S$  to be the number of features in subset  $\mathbf{FS}$ .

*Step 3:* Denote the  $NHN$  of the ELM by  $K$  and initialize  $K = S$ .

*Step 4:* Feed subset  $\mathbf{FS}$  into the ELM classifier with  $K$  hidden nodes to train and search for the optimal combination  $(S, K)$ . Considering the specific requirements for lie detection and to avoid over-fitting problem, a Subject-Wise cross validation (SWCV) [46] was adopted, resulting in 14 pairs of training sets and testing sets. Furthermore, a 10-fold CV was performed on each pair of training set. Hence, the averaged accuracy, denoted by  $BA_{train}$ , is calculated by averaging the values of  $TR_{sen}$  (the mean of 14 sensitivities [47]) and  $TR_{spe}$  (the mean of 14 specificities).

*Step 5:* Update  $K$  to  $K+1$ . Repeat *Step 3* through *5* until  $K = S+20$  (based on prior knowledge and the computational limitation of the lie detection system), as shown by the inner loop in Figure 3.

*Step 6:* Update  $S$  by  $S+1$ . Repeat *Step 2* through *6* until  $\mathbf{F}$  is empty, as shown by the outer loop in Figure 3.

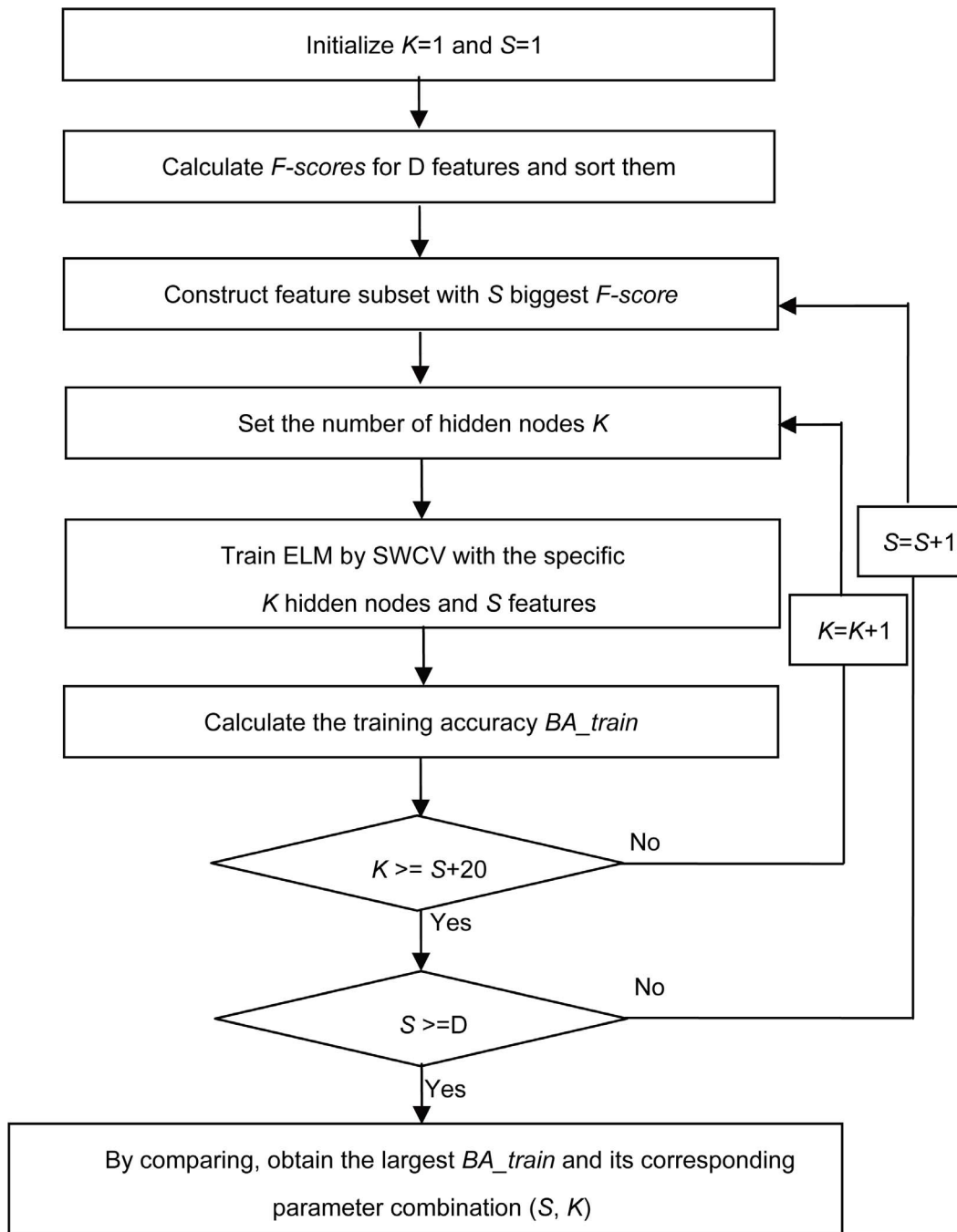
*Step 7:* Comparing all of the  $BA_{train}$  values obtained in *step 4*, the optimal parameter combination  $(S, K)$  is finally obtained when the  $BA_{train}$  reaches its highest value. Accordingly, the solution  $\hat{\boldsymbol{\beta}}$  and its corresponding value of the hidden node  $(\mathbf{a}, \mathbf{b})$  in the ELM are also obtained, which are fixed and then used in testing phase.

*Step 8:* Calculate the testing accuracy on the 14 pairs of testing sets with the optimal feature subset and the trained ELM. Hence,  $TE_{sen}$  (the mean of the 14 sensitivities) and  $TE_{spe}$  (the mean of the 14 specificities) can be obtained.

In the *Step 4* and *Step 8*, the sensitivity and the specificity refer to the percentage of the correctly classified feature samples with the class label 1 (P3 class) and  $-1$  (*non*-P3 class), respectively.

To objectively evaluate the performance of the proposed method, the following combined classification models were also performed: PCA\_ELM, PCA\_BPNN, PCA\_SVM, F-score\_BPNN and F-score\_SVM. Three individual classification models (i.e., the models without integrating feature selection) were also conducted: ELM, BPNN and SVM. Each individual model was trained only to obtain the optimal classifier parameters when the training accuracy  $BA_{train}$  reached its highest value.

For the models that utilized PCA, the eigenvalues were first calculated and sorted in a descending order. Then, the



**Figure 3. The block diagram of the proposed method F-score\_ELM.**  
doi:10.1371/journal.pone.0064704.g003

transformed new feature set was constructed using the  $d$  largest eigenvalues. The new feature set was then fed into the classifiers. Similar to F-score\_ELM, we used grid-search technique to jointly optimize the optimal value of  $d$  (see Section Feature Selection) and the classifier parameters.

In this study, a sigmoid activation function  $g(x) = 1/(1 + \exp(-x))$  was used in all of the classification models to fairly and objectively compare these models. The learning rate  $\eta$  and the control precision  $\varepsilon$  of the models that integrated BPNN were set to be 0.025 and 0.002, respectively; The Levenberg-Marquardt algorithm was used for the training of these

models, and the  $NHN$  of BPNN was also optimized by the grid-searching. The training and testing strategies mentioned above were also used for the models that utilized the SVM, and, based on our previous experience, the penalty parameter  $C$  and the radial width  $\sigma$  for radial basis function (RBF) [48] (kernel function  $K(x,y) = e^{-1/2 * (\|x-y\|/\sigma)^2}$ ) were tuned with the following grid:  $C = [2^5, \dots, 2^8]$ ,  $\sigma = [2^3, \dots, 2^6]$  (step size =  $2^1$ ). To decrease the huge training time, 10-fold SWCV and then normal 5-fold CV, which consists of a three-dimensional grid-search procedure, were used in the training stage for BPNN and SVM.

Using the optimization procedure described above, the following measures were used to evaluate the performance of the total nine classification models:

- (i) The training accuracy. This measure consists of the sensitivity, the specificity, and their respective standard deviations (SDs). They correspond to  $TR_{sen} \pm SD$  and  $TR_{spe} \pm SD$  when the corresponding  $BA_{train}$  reaches its highest value.
- (ii) The test accuracy. Similar to the above measures, This measure refers to  $TE_{sen} \pm SD$  and  $TE_{spe} \pm SD$ . Additionally, all of the  $TE_{sen}$  and  $TE_{spe}$  are averaged to obtain a balanced testing accuracy, which is denoted by  $BA_{test}$ .
- (iii) The optimal number of features in the feature subset when the classification model reaches the highest value of  $BA_{train}$ . This optimal number is denoted by  $NFS$ .
- (iv) The optimal classifier parameters when the classification model reaches the highest value of  $BA_{train}$ . For the models that integrate BPNN and ELM, this parameter is the optimal value of  $NHN$ , whereas for the models that integrate SVM, the number of the support vectors ( $NSV$ ) is used to compare the models with ELM and BPNN.
- (v) The training time of the classification models, which refers to the time spent on the *Step 1* through and is denoted by  $TTR$ .
- (vi) The testing time of the classification models, which refers to the time spent on testing the 14 pairs of unseen testing sets. For individual models, which is denoted by  $TTE$ , refers to the time required to test the 14 pairs of unseen datasets in the original feature space.

## Results

Table 1 shows the  $F$ -score values of the 31 original features. Those features with relatively larger  $F$ -score values were selected to construct the feature subset. The detailed results of the above mentioned six measures are summarized in Table 2. Furthermore, we listed the values of  $BA_{train}$  and  $BA_{test}$  in Table 3 for each classification model. Finally, the averaged value of each pair of the sensitivity and the specificity in each model was calculated for training and testing, respectively, which yielded 14 balanced accuracies for the training (and the testing) for each model. Hence, the paired  $t$ -test was performed between F-score\_ELM and each of the other models to obtain the corresponding significance level ( $p$  value) of the difference of the balanced accuracy. These significance levels are also provided in Table 3.

### 1. General Classification Performance

First, the comparison of the accuracy results in the first three rows in Table 2 shows that ELM, which exhibited training sensitivity of 98.72% and training specificity of 98.16%, performs significantly better than SVM (paired  $t$ -test,  $p < 0.001$ ) and BPNN (paired  $t$ -test,  $p < 0.001$ ). The results of the comparison of the generalization performance are the same as the training results (paired  $t$ -test,  $p < 0.001$ ).

As shown in Table 3, the comparison of the accuracy between the hybrid and its corresponding individual model revealed that each hybrid model achieves significantly higher accuracy than the corresponding individual model, with the exception of F-score\_BPNN. For example, the BPNN obtained a  $BA_{test}$  value of 90.72%, whereas PCA\_BPNN achieved 98.27% (paired  $t$ -test,  $p < 0.0001$ ). Both the  $BA_{test}$  and the  $BA_{train}$  of F-score\_ELM are significantly higher than the corresponding values obtained with

**Table 1.** The results of feature valuation on original 31 features using F-score method.

Features	$F$ -score values
$V_{max}$	<b>0.966</b>
$t_{max}$	0.581
$R_{L/A}$	0.162
$V_{min}$	0.019
$V_{ptp}$	<b>0.890</b>
$A_p$	0.124
$f_{max}$	0.001
$f_{mean}$	0.305
$A_{if}$	<b>0.886</b>
$W_1-W_5$	0.075, 0.001, 0.317, 0.002, 0.075,
$W_6-W_{10}$	0.003, 0.154, 0.097, 0.069, 0.364,
$W_{11}-W_{16}$	<b>0.977</b> , 0.554, 0.213, <b>0.886</b> , 0.874, <b>0.987</b>
$W_{17}-W_{22}$	<b>0.893</b> , <b>0.953</b> , <b>0.892</b> , <b>0.987</b> , 0.874, <b>0.881</b>

doi:10.1371/journal.pone.0064704.t001

the ELM (see Table 3,  $p = 0.044 < 0.05$  and  $p = 0.008 < 0.01$ , respectively).

Second, as shown in the last column of Table 2, it is obvious that the feature selection reduces the value of the  $NFS$  for the hybrid models from the number of the original features ( $NFS = 31$ ). For example, F-score\_ELM selected 11 features ( $NFS = 11$ ), which are most informative to the classification and thus highlighted in Table 1, to construct the feature subset.

Third, the feature selection effectively helped the hybrid models obtain lower values of the  $NHN$  than the corresponding individual models, as can be observed from the table. For example,  $NHN$  is equal to 29 and 51 for F-score\_ELM and ELM, respectively. Additionally,  $NHN$  values of 20 and 45 were obtained for PCA\_BPNN and BPNN, respectively.

The smaller  $NFS$  and  $NHN$  and the higher accuracy obtained with the hybrid models confirms the hypothesis that the combination of the feature selection method and these classifiers improves the classification performance. The above results are analyzed further in the next section.

Comparing the PCA and F-score methods, we can observe that each classifier combined with F-score achieves an accuracy that is similar to that obtained by the same classifier combined with PCA. As shown in Table 3, there is no significant difference of accuracy between F-score\_ELM and PCA\_ELM ( $p = 0.233 > 0.01$  for training and  $p = 0.173 > 0.01$  for testing). The results appear to indicate that both F-score and PCA can be successfully combined with these three classifiers to jointly optimize the feature space and the classification parameters for lie detection. However, each classifier combined with F-score requires significantly less computation time than the same classifier combined with PCA. For instance, the  $TTR$  of F-score\_BPNN equals 58.58 hours, whereas the  $TTR$  of PCA\_BPNN equals 106.64 hours. We will compare these two methods further in Discussion.

Moreover, as shown in Table 2, the difference between the  $TTR$  of F-score\_ELM and that of F-score\_SVM is almost 1108-fold. In addition, the  $TTE$  of F-score\_ELM is approximately 0.003 s, which is the shortest among all nine models tested. This short time for F-score\_ELM should be attributed to the optimized results: the smallest  $NHN$  ( $NHN = 29$ ) and smallest feature number ( $NFS = 11$ ).

**Table 2.** Performance of the classification models with the optimal *NFS* and *NHN* (or *NSV*).

Classification models	Times		Accuracy (%)				<i>NHN/NSV</i>	<i>NFS</i>
	<i>TTR(h)</i>	<i>TTE(s)</i>	Training		Testing			
			<i>TR<sub>sen</sub>±SD</i>	<i>TR<sub>spe</sub>±SD</i>	<i>TE<sub>sen</sub>±SD</i>	<i>TE<sub>spe</sub>±SD</i>		
<b>BPNN</b>	1.25	0.51	95.31±1.38	88.61±4.44	95.05±3.25	86.38±3.15	45	31
<b>SVM</b>	20.22	22.06	97.88±0.41	97.68±0.41	96.33±1.87	95.15±2.29	58.28	31
<b>ELM</b>	0.03	0.004	98.72±0.35	98.16±0.51	98.26±0.32	98.14±0.44	51	31
<b>PCA_BPNN</b>	106.64	1.59	95.10±1.82	95.75±1.97	98.20±1.82	98.34±1.93	20	14
<b>PCA_SVM</b>	675.58	25.23	99.32±0.35	99.28±0.33	99.65±0.02	98.91±0.02	57.76	14
<b>PCA_ELM</b>	0.64	0.014	98.82±0.36	99.31±0.36	98.99±0.48	98.79±0.51	30	13
<b>F-score_BPNN</b>	58.58	0.184	95.26±0.23	88.05±1.36	95.80±3.92	90.05±5.27	34	25
<b>F-score_SVM</b>	633.77	21.62	98.06±0.48	98.21±0.44	97.68±0.02	97.95±0.02	60.25	25
<b>F-score_ELM</b>	0.61	0.003	99.42±0.38	98.52±0.72	99.27±0.24	98.17±0.25	29	11

doi:10.1371/journal.pone.0064704.t002

## 2. Impact of Network Size and Feature Selection on the Classification Performance

Based on the above results, we next investigated the effect of the feature selection on the optimal network size and the classification accuracy. In this study, the network size refers to the *NHN* for the ELM classifier and the *NSV* for the SVM classifier. The results of analysis are shown in Figure 4. Each curve in Figure 4A illustrates the highest sensitivities  $TR_{sen} \pm SD$  that the indicated classification model can achieve with *NFS* varying from 1 to 31. The specificities  $TR_{spe} \pm SD$  are similarly plotted in Figure 4B. For each model, Figure 4C demonstrates the relationship between the *NFS* and the *NHN* for which  $BA_{train}$  achieves its highest value. Similarly, the relationship between the *NFS* and the *NSV* is shown in Figure 4D.

As shown in Figure 4A and 4B, in general, there is no obvious monotonically increasing tendency as the *NFS* increases from 1 to 31. For example, the results of F-score\_ELM show a  $TR_{sen}$  of 99.42% and a  $TR_{spe}$  of 98.52% when *NFS*=11; these values are slightly less than the highest corresponding values (99.47% and

98.89%, respectively), which are obtained for *NFS* equal to 25. As shown in Figure 4, this phenomenon is also exhibited by F-score\_BPNN and F-score\_SVM models. Hence, the accuracy that approximates the highest value with a significantly smaller *NHN* and *NFS* is regarded as *the best accuracy* (i.e., the highest value was not always the optimal for our training purpose). These best accuracies and the corresponding *NHN* and *NSV* for the three models, which correspond to the results in the last three rows in Table 2, are labeled in Figure 4A through 4D.

We also investigated the individual influence of the *NHN* on the classification accuracy of the ELM classifier. We set the *NFS* to be 11 (the optimal values mentioned previously) and then trained F-score\_ELM with a grid search of the *NHN*, which varied from 1 to 200. Figure 5 shows the training accuracy as a function of the *NHN*. Figures 5A and 5B show the sensitivities  $TR_{sen}$  and the specificities  $TR_{spe}$ , respectively. There is a large fluctuation in the classification accuracy as the *NHN* increases gradually. For example, the sensitivity and specificity only equal 88.8% and 71.7%, respectively when *NHN* is set to be 2, whereas these are 96.03% and 91.02%, respectively, when *NHN* is set to be 12. The sensitivity and specificity reach almost the highest value (99.43% and 98.76%, respectively, when the *NHN* is set to be 29). Both of these measures decrease when *NHN* is varied from 80 to 200.

**Table 3.** Balanced accuracy of each model and statistical analysis results between F-score\_ELM and the other models.

Classification models	Balanced accuracy and statistical results (%)			
	Training		Testing	
	<i>BA<sub>train</sub></i>	<i>p value (2-t)</i>	<i>BA<sub>test</sub></i>	<i>p value (2-t)</i>
<b>BPNN</b>	91.96	0.000*	90.72	0.000*
<b>SVM</b>	97.78	0.000*	95.74	0.000*
<b>ELM</b>	98.44	0.044 <sup>▲</sup>	98.20	0.008*
<b>PCA_BPNN</b>	95.43	0.000*	98.27	0.009*
<b>PCA_SVM</b>	99.30	0.006*	99.30	0.004*
<b>PCA_ELM</b>	99.06	0.233	98.89	0.173
<b>F-score_BPNN</b>	91.66	0.000*	92.93	0.000*
<b>F-score_SVM</b>	98.14.	0.0008*	97.82	0.006*
<b>F-score_ELM</b>	98.97	□	98.72	□

"\*" and "▲" denotes *p value*<0.01 and *p value*<0.05, respectively for the comparison of the F-score and the indicated model using paired *t*-test.  
doi:10.1371/journal.pone.0064704.t003

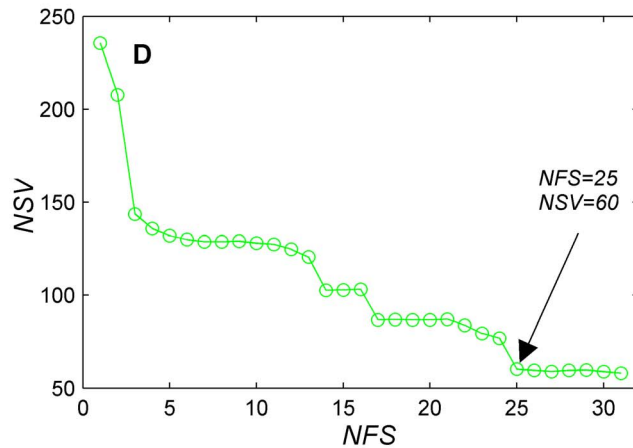
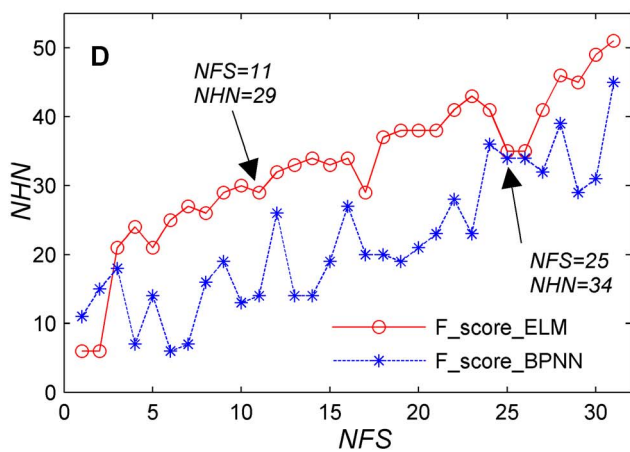
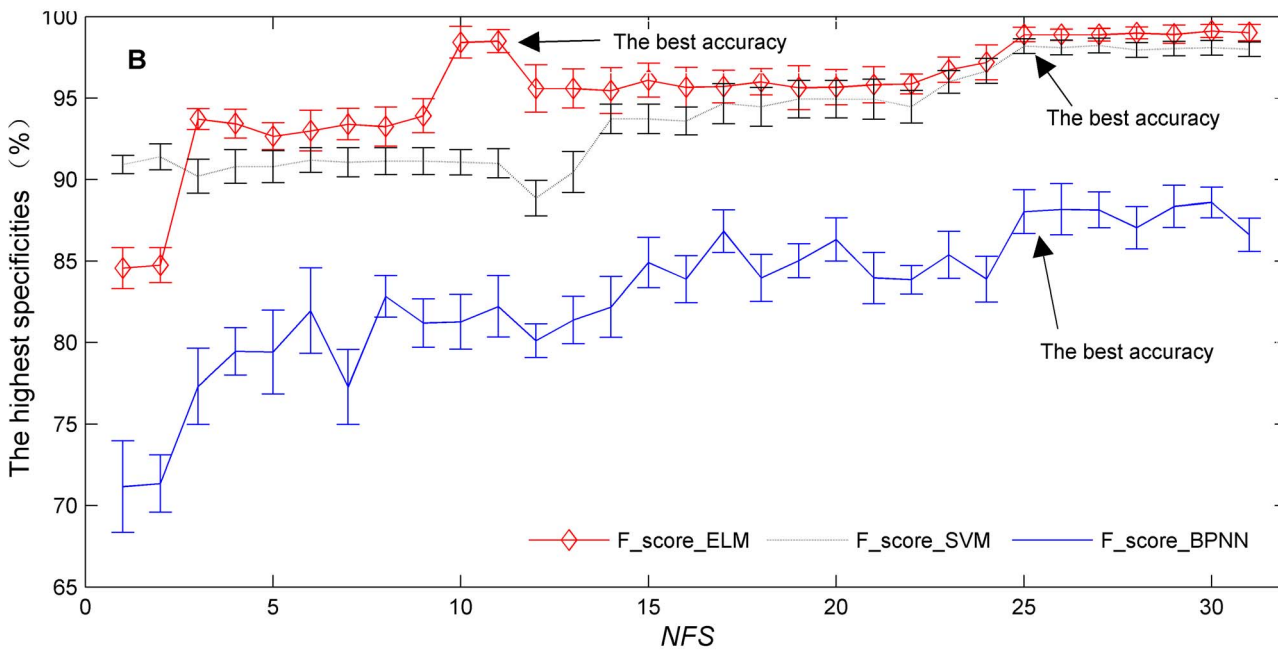
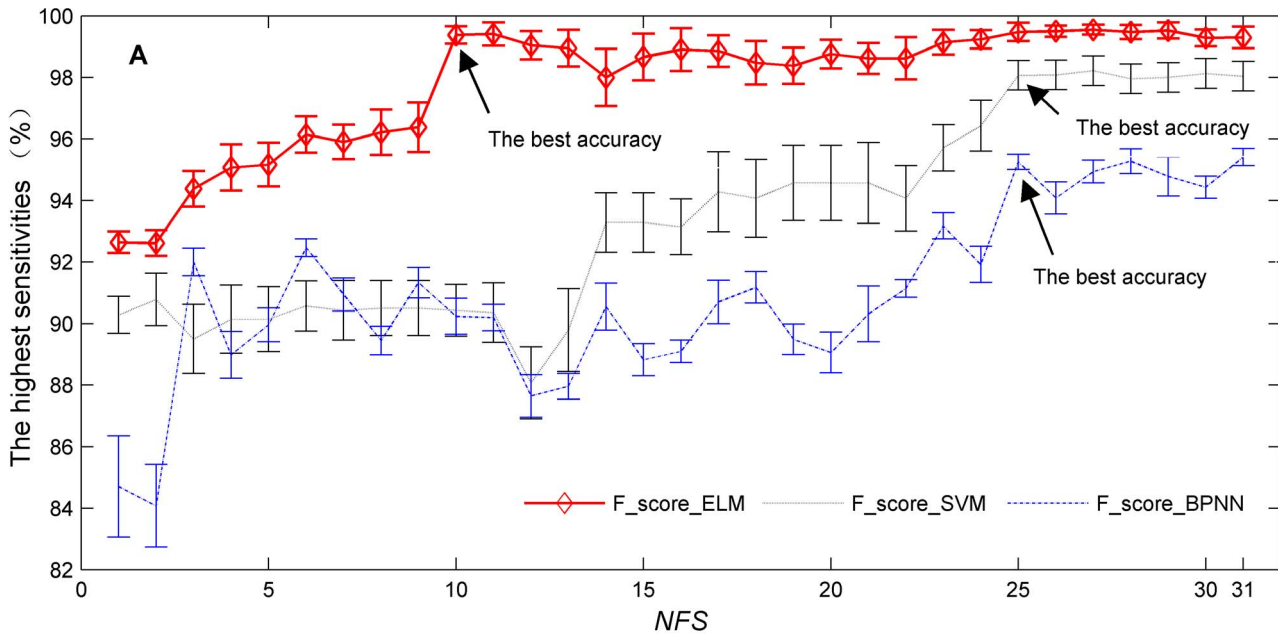
## 3. Individual Diagnostic Rate

The final aim of a lie detection system is to correctly separate the guilty subjects from the innocent subjects. The individual diagnostic rate is the most important evaluation measurement for a lie detection system. As shown in Table 2, the testing accuracies  $TE_{sen}$  and  $TE_{spe}$  of our proposed method are 99.27% and 98.17%, respectively. Therefore, the averaged testing accuracy is 98.72%, which is the threshold for individual diagnosis (a subject will be classified as a liar if either the sensitivity or the specificity is higher than 98.72%). This number is higher than most results that have been reported in the literature [4,7,9] and is also acceptable for practical applications.

## Discussion and Conclusions

In this work, ELM method was first introduced for the purpose of lie detection, and the optimization of the *NHN* of ELM was combined with the F-score feature selection method. As a popular feature selection method, PCA was also combined with ELM,





**Figure 4. Training accuracy and  $NHN/NSV$  as a function of  $NFS$  achieved by the three classification models.** Each point in each curve corresponds to the highest classification performance of the indicated model with the optimal  $NHN/NSV$ . 4A: Highest sensitivity with the optimal  $NHN$  or  $NSV$  vs  $NFS$ . 4B: Highest specificity with the optimal  $NHN$  or  $NSV$  vs  $NFS$ . 4C:  $NHN$  vs  $NFS$  for which  $BA_{train}$  achieves its highest value. 4D:  $NSV$  vs  $NFS$  for which  $BA_{train}$  achieves its highest value.  
doi:10.1371/journal.pone.0064704.g004

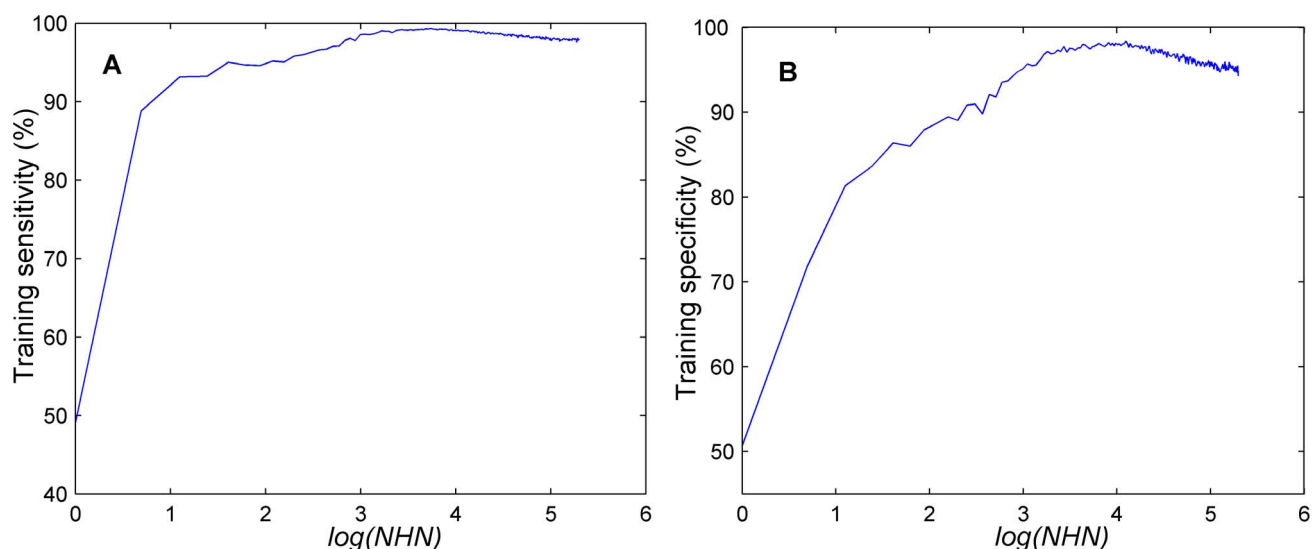
BPNN, and SVM to construct various classification models. The training and testing times, classification accuracy, and network size were used to completely assess the classification performance of these models. Compared with the other classification models tested, the experimental results showed that the proposed method (F-score\_ELM) achieves nearly the highest training and testing accuracies for the identification of lying and truth-telling using the most compact network and the shortest training and testing times. Additionally, the proposed method obtains a very high individual diagnostic rate.

The ELM method has many advantages over most other classifiers, such as BPNN and SVM. However, only a few very effective methods have been developed to decide the optimal  $NHN$ . Most of the investigations focused on the improvement on the ELM algorithm itself. If the feature space is changed, the training procedure needs to be rebooted. Hence, the  $NHN$  in the ELM should be changed accordingly. We therefore opted to not use the improved ELM algorithms that have been proposed in the past few years and combined ELM with the feature selection method to automatically select the optimal  $NHN$ . As shown in Table 2, the trained F-score\_ELM obtains a smaller  $NHN$  ( $NHN=29$ ) with a higher accuracy than the trained ELM model ( $NHN=51$ ). As a result, an advantage of our proposed method is the integration of the optimization of  $NHN$  and of the feature subset into one optimized procedure. The proposed method provides a new concept for the determination of the optimal  $NHN$  in ELM algorithms.

With respect to feature selection, the experimental results show that F-score\_ELM enhanced the classification performance with the most informative features ( $NFS$  equals 11) selected by F-score method, compared with the individual ELM model, as shown in Table 3. This experimental result indicates the importance of the feature selection for the ELM classifier.

For the F-score methodology, most researchers always remove redundant features by a threshold calculated based on the  $F$ -score values. However, this thresholding strategy exhibits some limitations. First, it appears too harsh to directly remove those features with the  $F$ -score values that are less than the threshold. Second, there should be a close relationship between the  $NFS$  and the  $NHN$  of the classifiers. Hence, we abandoned the commonly used technique mentioned above and proposed a combined optimized strategy. It is worth mentioning that the proposed optimization strategy is especially suitable to the ELM classifier because only one classifier parameter in ELM needs to be tuned.

Although we find that  $BA_{test}$  of F-score\_ELM is slightly lower than that of PCA\_ELM ( $p=0.173>0.05$ ) and significantly lower than that of PCA\_SVM ( $p=0.006<0.01$ ), it was stressed in this study that we evaluated the classification model by its comprehensive performance. Hence the optimal parameter values were not decided only when the training or testing accuracies were maximal. In fact, when  $NFS$  equals 25, the correspondingly testing accuracy reached its highest value with  $BA_{test}$  equal to 99.19%, which is significantly higher (paired  $t$ -test,  $p<0.01$ ) than that of PCA\_ELM ( $BA_{test}=98.37\%$ ). Furthermore, it is noted that there are three inherent limitations in the models that utilize PCA. First, if a new training set is fed into the classification system, the optimal transformed feature set needs to be calculated again because the eigenvector matrix  $\mathbf{U}$ , as mentioned in Feature Selection section, is calculated solely on the input samples. In contrast, the optimal feature subset does not need to be calculated again for the models that utilize F-score methodology. The optimal feature subset with the optimal  $NFS$  can be directly selected from the new input samples. Second, it is very obvious that F-score has significantly shorter calculation time than PCA. Third, the implication of each feature transformed by PCA is not clear, whereas the features selected by F-score are clear, which could help evaluate the



**Figure 5. Training accuracy (constant  $NFS=11$ ) as a function of  $NHN$  achieved by the F-score\_ELM.** 5A: Highest sensitivity  $TR_{sen}$  vs  $\log(NHN)$ . 5B: Highest specificity  $TR_{spe}$  vs  $\log(NHN)$ .  
doi:10.1371/journal.pone.0064704.g005

importance level of each original feature. The above discussion appears to indicate that F-score is superior to PCA for the identification of the optimal feature subset.

A good lie detection system with real applications should achieve a high classification accuracy with a lower computational burden. Through the proposed joint optimization, the testing time can be decreased significantly because the smaller *NFS* and *NHN* will be used in testing phase. In addition, in the future, we may need to train on specific subjects to avoid the problem of individual difference. The time required for future training for the proposed method would be also significantly shorter than that for individual ELM and that spent on the originally training procedure because after the original learning procedure, the new training samples with a smaller *NFS* can be fed into the original trained ELM with a corresponding smaller *NHN*. Furthermore, if we found a few important new features, we only need to add these to the originally selected optimal features and then re-perform the above training procedure.

The grid-search strategy was used in many literatures to tune the optimal classifier parameters [40,46]. In this study, we did not use a more sophisticated algorithm, but employed the grid-search strategy with a SWCV procedure on the feature samples to find the optimal combination of *NHN* and *NFS*. We assumed that this strategy could ransack each combination of *NHN* and *NFS* and hence avoid the local minimum. In addition, because each parameter combination is independent, one can parallelize this searching procedure and thus significantly reduce the learning time of the proposed method.

For the sake of simplicity, we averaged 5 responses to remove the noise in the original ERP signals. We recognized that the averaging method adopted in this study is not very effective for removing noise of P300 because some noise is also time-locked to the stimuli [49]. If more appropriate methods of ERP reconstruction [36,50] were utilized, our method should be more robust. In addition, other feature selection methods, such as mutual information (MI) and correlation-based method, can be attempted to combine with the ELM to enhance the robustness and the efficiency of the ELM classifier in lie detection or other EEG-based classification problems. This will be the focus of our future works.

Due to space constraints, only two comparable improved ELM algorithms, I-ELM [51] and Pruning ELM (P-ELM) [52], were selected to compare with the proposed method. After training and testing, the following results were obtained: 1) I-ELM: *BA\_train* = 98.61%, *BA\_test* = 98.22%, *NHN* = 51, *NFS* = 31, *TTR* = 0.04 hours; 2) P-ELM: *BA\_train* = 98.63%, *BA\_test* = 98.18%, *NHN* = 50, *NFS* = 31, *TTR* = 0.21 hours. Based on the principle of

these algorithms, it is not surprising that the trained values of *NHN* and *NFS* are nearly the same as the training results of ELM (see Table 2). Using paired *t*-test, we found no significant difference of training and testing accuracies between these two improved algorithms and ELM model ( $p = 0.228 > 0.01$  for I-ELM and  $p = 0.24 > 0.01$  for P-ELM on testing accuracy). As for other improved algorithms such as Kernel based ELM [53] and OS-ELM [54], it may be unsuitable to apply them in this detecting system due to their more complex calculation and training procedure.

The advantage of the proposed method over BAD and BCD is the trials-by-trial analysis, which is particularly beneficial for lie detections. First, we could research the wealth of dynamic variation information between the single trials. Second, based on the testing results on a few trials, the examiner could decide whether to continue to display, terminate or adjust the stimuli content and/or sequence, which would help the examiner make his/her final decision. In contrast, BAD and BCD must make a decision after all the trials are presented. From this perspective, the proposed testing system is based on a small number of stimuli. In fact, the use of many repeated stimuli with little information would induce two problems: fatigue in the subjects and an increase of the countermeasures [7,9]. This is particular true for real lie detection, when the real criminals are familiar with the stimuli contents, which would result in the criminals resisting the detection procedure on purpose [36]. In our opinion, the development of a lie-detection method with a small number of stimuli is crucial to extend the laboratory study to practical application. In comparison with previously reported PR methods, the presented method reaches a very high classification accuracy with the shortest training and testing times. Therefore, although the classification accuracy in the presented method may be lower than some early reported results, the proposed method achieves a good tradeoff between the various evaluation measures introduced earlier.

Although this study is focused on lie detection, the method reported is not limited to this application. The proposed method has the potential to be used in a variety of classification tasks of brain states, such as identification of ERP or fMRI signals.

## Author Contributions

Conceived and designed the experiments: JFG. Performed the experiments: WJZ YY CYT. Analyzed the data: WJZ CYT JAG NNR. Contributed reagents/materials/analysis tools: WJZ. Wrote the paper: JFG ZW.

## References

- Langenblein DD, Loughhead JW, Bilker WB, Ruparel K, Childress AR, et al. (2005) Telling truth from lie in individual subjects with fast event-related fMRI. *Hum Brain Mapp* 26(4): 262–272.
- Phan KL, Magalhaes A, Ziemlewicz TJ, Fitzgerald DA, Green C, et al. (2005) Neural correlates of telling lies: a functional magnetic resonance imaging study at 4Tesla. *Acad Radiol* 12(2): 164–172.
- Rosenfeld JP (2002) Event-related potentials in the detection of deception. *Handbook of Polygraph Testing*. Academic Press, New York. 265–286.
- Davatzikos C, Ruparel K, Fan Y, Shen DG, Acharyya M, et al. (2005) Classifying spatial patterns of brain activity with machine learning methods: Application to lie detection. *NeuroImage* 28(3): 663–668.
- Polich J, Herbst KL (2000) P300 as a clinical assay: rational, evaluation, and findings. *Int J Psychophys* 38(1): 3–19.
- Meijer EH, Smulders FTY, Merckelbach HLGJ, Wolf AG (2007) The P300 is sensitive to concealed face recognition. *Int J Psychophys* 66(3): 231–237.
- Rosenfeld JP, Labkovsky E, Winograd M, Lui MA, Vandenberg C, et al. (2008) The Complex Trial Protocol (CTP): A new, countermeasure-resistant, accurate, P300-based method for detection of concealed information. *Psychophysiology* 45(6): 906–919.
- Farwell LA, Donchin E (1991) The truth will out: interrogative polygraphy (“lie detection”) with event-related potentials. *Psychophysiology* 28(5): 531–547.
- Abootalebi V, Moradi MH, Khalilzadeh MA (2009) A new approach for EEG feature extraction in P300-based lie detection. *Comput Method Program Biomed* 94: 48–57.
- Gao JF, Yan XG, Sun JC, Zheng CX (2011) Denoised P300 and machine learning-based concealed information test method. *Comput Method Program Biomed* 104(3): 410–417.
- Hsu WY (2011). Continuous EEG signal analysis for asynchronous BCI application. *Int J Neural Syst* 21 (4): 335–350.
- Gordo MAL, Pelayo F, Prieto A, Fernandez E (2012) An auditory Brain-Computer interface with accuracy prediction. *Int J Neural Syst* 22(3): 1–14.
- Vapnik V (1998) *Statistical Learning Theory*. Wiley, New York.
- Faust O, Acharya UR, Aust LM, Sputh BHC (2010) Automatic identification of epileptic and background EEG signals using frequency domain parameters. *Int J Neural Syst* 20(2): 159–176.
- Huang GB, Zhu QY, Siew CK (2006) Extreme learning machine: theory and applications. *Neurocomputing* 70: 489–501.
- Yuan Q, Zhou WD, Li SF, Cai DM (2011) Epileptic EEG classification based on extreme learning machine and nonlinear features. *Epilepsy Res* 96: 29–38.

17. Bartlett PL (1998) The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network. *IEEE Trans Inform Theory* 44(2): 525–536.
18. Huang GB, Wang DH, Lan Y (2011) Extreme learning machines: a survey. *Int J Mach Learn & Cyber* 2: 107–122.
19. Choi K, Toh KA, Byun H (2011) Realtime training on mobile devices for face recognition applications. *Pattern Recogn* 44(2): 386–400.
20. Chen FL, Ou TY (2011) Sales forecasting system based on gray extreme learning machine with Taguchi method in retail industry. *Expert Syst Appl* 38(3): 1336–1345.
21. Liang NY, Saratchandran P, Huang GB, Sundararajan N (2006) Classification of mental tasks from EEG signals using extreme learning machine. *Int J Neural Syst* 16: 29–38.
22. Shi LC, Lu BL (2013) EEG-based vigilance estimation using extreme learning machines. *Neurocomputing*. 102: 135–143.
23. Song Y, Liò P (2010) A new approach for epileptic seizure detection: sample entropy based feature extraction and extreme learning machine. *J Biomed Sci Eng* 3: 556–567.
24. Han F, Huang DS (2006) The forecast of the postoperative survival time of patients suffered from non-small cell lung cancer based on PCA and extreme learning machine. *Int J Neural Syst* 16(1): 39–46.
25. Chen FL, Li FC (2010) Combination of feature selection approaches with SVM in credit scoring. *Expert Syst Appl* 37: 4902–4909.
26. Polat K, Güneş S (2009) A new feature selection method on classification of medical datasets: Kernel F-score feature selection. *Expert Syst Appl* 36 (7): 10367–10373.
27. Akay MF (2009) Support vector machines combined with feature selection for breast cancer diagnosis. *Expert Syst Appl* 36(2): 3240–3247.
28. Cao JW, Lin ZP, Huang GB, Liu N (2012) Voting based extreme learning machine. *Information Sciences* 285(1): 66–77.
29. Huang GB, Chen L, Siew CK (2006) Universal approximation using incremental constructive feedforward networks with random hidden nodes. *IEEE Trans Neural Networks* 17(4): 879–892.
30. Huang GB, Chen L (2008) Enhanced random search based incremental extreme learning machine. *Neurocomputing* 71: 3460–3468.
31. Feng GR, Huang GB, Lin QP, Gay R (2009) Error minimized extreme learning machine with growth of hidden nodes and incremental learning. *IEEE Trans Neural Networks* 20: 1352–1357.
32. Miche Y, Heeswijk MV, Bas P, Simula O, Lendasse A (2011) TROP-ELM: a double-regularized ELM using LARS and Tikhonov regularization. *Neurocomputing* 74(16): 2413–2421.
33. Huynh HT, Won Y, Kim J (2008) An improvement of extreme learning machine for compact single-hidden-layer feedforward neural networks. *Int J Neural Syst* 18(5): 433–441.
34. Rosenfeld JP, Ellwanger JW, Nolana K, Wua S, Bermanna RG, et al. (1999) P300 Scalp amplitude distribution as an index of deception in a simulated cognitive deficit model. *Int J Psychophys* 33(1): 3–19.
35. Polich J (2007) Updating P300: An integrative theory of P3a and P3b. *Clin Neurophysiol* 118(10): 2128–2148.
36. Gao JF, Lu L, Yang Y, Yu G, Na LT, et al. (2012) A Novel Concealed Information Test Method Based on Independent Component Analysis and Support Vector Machine. *Clin EEG Neurosci* 43(1): 54–63.
37. Hsu WY (2012) Application of competitive Hopfield neural network to brain-computer interface systems. *Int J Neural Syst* 22(1): 51–62.
38. Chen YW, Lin CJ (2003) Combining SVMs with various feature selection strategies. *NIPS 2003 feature selection challenge*. pp.1–10.
39. Lee MC (2009) Using support vector machine with a hybrid feature selection method to the stock trend prediction. *Expert Syst Appl* 36: 10896–10904.
40. Dash M (2002) Feature selection for clustering—a filter solution. in *Proceedings of the second international conference on data mining*. pp.115–122.
41. Kohavi R, John GH (1997) Wrappers for feature subset selection. *Artificial Intelligence* 97(1–2): 273–324.
42. Huang CJ, Dian X, Chuang YT (2007) Application of wrapper approach and composite classifier to the stock trend prediction. *Expert Syst Appl* 34 (4): 2870–2878.
43. Malhi A, Gao RX (2004) PCA-Based Feature Selection Scheme for Machine Defect Classification. *IEEE Trans Instrum Meas* 53(6): 1517–1525.
44. Chawla MPS (2009) A comparative analysis of principal component and independent component techniques for electrocardiograms. *Neural Comput Appl* 18(6): 539–556.
45. Hsu CW, Lin CJ (2002) A comparison of methods for multiclass support vector machines. *IEEE Trans Neural Networks* 13: 415–425.
46. Shao SY, Shen KQ, On CJ, Wilder-Smith EPV, Li XP (2009) Automatic EEG artifact removal: A weighted support vector machine approach with error correction. *IEEE Trans Biomed Eng* 56(2): 336–344.
47. Acharya UR, Sree SV, Chattopadhyay SJ, Suri S (2012) Automatic diagnosis of normal and alcoholic EEG signals. *Int J Neural Syst* 22(3): 1–11.
48. Burges C (1998) A tutorial on support vector machines for pattern recognition. *Data Min Knowl Discov* 2: 121–157.
49. Jung TP, Makeig S, Humphries C, Lee TW, McKeown MJ, et al. (2000) Removing electroencephalographic artifacts by blind source separation. *Psychophysiology* 37(2): 163–178.
50. Ihrke M, Schrobsdorff H, Herrmann JM (2011) Recurrence-based estimation of time-distortion functions for ERP waveform reconstruction. *Int J Neural Syst* 21(1): 65–78.
51. Huang GB, Chen L (2008) Enhanced random search based incremental extreme learning machine. *Neurocomputing* 71: 3460–3468.
52. Rong HJ, Ong YS, Tan AH, Zhu Z (2008) A fast pruned extreme learning machine for classification problem *Neurocomputing* 72: 359–366.
53. Huang GB, Zhou H, Ding X, Zhang R (2010) Extreme learning machine for regression and multiclass classification. *IEEE Trans Syst Man Cybernetics* 42(2): 513–529.
54. Liang NY, Huang GB, Saratchandran P, Sundararajan N (2006) A fast and accurate on-line sequential learning algorithm for feedforward networks *IEEE Trans Neural Networks* 17: 1411–1423.