TWO-SENSOR EEG-BASED STRESS DETECTION SYSTEM

Guillermo Ramos-Auñón, Inma Mohino-Herranz, Héctor A. Sánchez-Hevia, Cosme Llerena-Aguilar, David Ayllón
Department of Signal Theory and Communications
Universidad de Alcalá
Alcalá de Henares, Madrid, Spain
guillermo.ramos.aunon@gmail.com

ABSTRACT
In this paper, we propose a computationally-efficient EEG-based stress detection that uses only two non-invasive electrodes. The system is designed to classify between two situations: high stress level or low stress level. A linear classifier is trained using supervised learning using a subset of features that has been selected among a larger proposed set of features, using a tailored feature selection algorithm. The proposed algorithm has been evaluated with subjects playing skill games, obtaining errors of 19.2% in the train set and 29.2% in the test set.

KEY WORDS
Stress detection, EEG, biomedical signal processing.

1 Introduction
In the last years, the study of communications between brain and computer, known as Brain-Computer Interface (BCI), has drawn the attention of the biomedical research community. One of the current research lines in this area involves the study of stress parameters in the human EEG signals [1, 2, 3, 4]. The analysis of existing patterns in the EEG signals allows providing the machine with the ability of understanding the different states of stress of its human partner. Stress is a reaction or response of the subject when faces daily mental, emotional or physical challenges, and it can be classified in different levels.

In order to classify stress levels, several features and machine learning algorithms have been proposed in the literature, for instance [1, 2, 4]. The features with greater classification capability are related to the fractal dimension, according to [1, 5, 6]. Other works are focused on selecting a suitable but reduced set of features, in order to improve the generalization capability of the proposed classifiers, for instance [2], in which a huge classification method is used to classify several number of parameters. Additionally, the scalp position of EEG electrodes can provide different information. For instance, in [3] it is demonstrate that an increase in the $\theta$ rhythm in Fz electrode and a reduction in the $\alpha$ rhythm in Pz electrode in the 10-20 system (see Figure 1) involve a growing mental activity. And, in citeWalker, it is demonstrated that the activity recorded by electrode Pz is associated with perception and the activity recorded in electrode Fz is associated with motion intention.

In some works, the analysis of EEG signals has been studied while the subject is playing video-games. In [8], an acquisition hardware with 24 electrodes is placed in the scalps of a subject while playing “Grand Turismo 5”. In [9], the association between violent video-games exposure and EEG signals is studied, using an acquisition system composed of 28 electrodes.

The goal of this paper is the design of a stress detection system based on EEG signals and supervised machine learning, using a non-invasive and simple EEG acquisition system, as well as algorithms with low computational cost that allow a real-time implementation of the system. The system is designed to classify between high/low stress levels, considering the information from the $\alpha$, $\beta$ and $\theta$ rhythms. The main novelty resides in the EEG signal acquisition system, which consists of a headset with only 2 electrodes. The system is validated with subjects playing skill games, where high stress is associated with full concentration.

2 Proposed stress detection system
As previously described, the objective in this work is to distinguish between two levels of stress (low and high), using supervised machine learning techniques. An overview of the proposed system is shown in Figure 2. It is divided in two stages, the training stage and the detection stage. In
the training stage, a computationally-efficient classifier is designed using supervised machine learning, and this classifier is used during the detection stage. The first step involves the design of an EEG signal acquisition system based on a headset containing only two sensors. Figure 1 represents the common 10-20 montage and the positions of the 2 sensors used in this work is shadows (Fp1 and A1 positions). A specific software has been developed to obtain the raw data provided by such system, which is denominated \( x(t) \) and represents the voltage between the two electrodes, i.e. \( x_1(t) - x_2(t) \). The next step involves some pre-processing operations of the raw EEG signals, in order to make easier the feature extraction step. A large set of features \( Q' \) is generated but only a subset \( Q \in Q' \) is used by the final system. The best subset of features is selected using a feature selection algorithm based on evolutionary computation. The selection process is made in terms of the mean square error (MSE) obtained by a least-squares linear classifier (LSLC), which is the classification schema chosen in this work. Once the best subset of features has been selected, the classifier is trained to distinguish between high stress level and low stress. Medium stress level is not considered in this paper due to the low number of available files in the database. In the detection stage, the same pre-processing operations should be carried out, but only the selected features are calculated and linearly combined, with the weights of the LSLC calculated in the training stage, to obtain the stress level.

The different system steps are described in detail in the remaining of this section.

### 2.1 EEG acquisition system

We measured the EEG signal with the single montage (Fp1 - A1 positions) surface electrodes using data acquisition hardware at a sampling rate of 512Hz. The hardware used was *Mindwave Mobile*™ Bluetooth Wireless Brainwave

*Headset.* In order to get the acquired data, we developed an Android application using the ThinkGearBase library that allow establishing a bluetooth connection between the EEG acquisition system and a smartphone. The data acquired by the headset is sent to the smartphone in real time, and stored in its internal memory for its posterior processing.

### 2.2 Pre-processing

After analyzing the data recorded from the EEG acquisition system, we found necessary the next pre-processing operations:

- Analyzing the acquired data we find that it has a continuous offset of 50 \( \mu \)V, so the first step was to remove this offset from the input data.

- EEG signals are usually interfered by artifacts from other biomedical sources such as muscle movements produced by the heart or eyes movements, but also are interfered by electrical equipments. In order to remove these artifacts, the EEG signals are filtered with a FIR low pass filter with a cut-off frequency of 40 Hz. Thus, we remove electrical interferences (typically at 50 Hz) and most of the biomedical artifacts, at the same time that the main information of the EEG signal is preserved (frequencies lower than 40 Hz).

- Due to the low voltage values (in the order of micro volts) of the EEG signals and that one of the surface electrodes is in the Fp1 position (above the eye), eye blinks suppose a notable interference and consequently necessary to remove. Eye blinks were removed detecting the positions where the amplitude of the whole EEG signal is higher or lower than three times the standard deviation of that amplitude in the time domain. An example of an eye blink in a EEG signal and its removal can be observed in Figure 3.
Linear classifiers are characterized by the use of linear decision boundaries, which implies that they cannot discriminate classes separated with complex shapes. Let us consider vector $x_k = [x_{k1}, x_{k2}, ..., x_{KL}]^T$, containing $L$ features of a pattern $k$ that can be assigned to one of the two possible classes defined in this work, high or low stress levels. The pattern matrix $P$ of size $L \times K$ is defined as a matrix that contains the patterns of a set of $K$ data samples, $P = [x_1, x_2, ..., x_K]$. The matrix $Q$ is then defined as

$$Q = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_{11} & x_{12} & \cdots & x_{K1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1L} & x_{2L} & \cdots & x_{KL} \end{bmatrix}. \quad (1)$$

The output of a linear classifier is obtained as a weighted linear combination of the input features, according to

$$y = v^T \cdot Q,$$ \quad (2)

where the vector $y = [y_0, y_1, ..., y_K]$ contains the output of the classifier for the $K$ input patterns and $v = [v_0, v_1, ..., v_L]^T$ contains the weights and the bias applied to each of the $L$ input features.

The design of the classifier consist in finding the best values of $v$ that minimize the classification error. In the case of supervised learning, the desired output values are available and used for training. The target vector containing the labels of each pattern is defined as $t = [t_0, t_1, ..., t_K]$, with values of 1 in the case of high stress level and 0 in the case of low stress level. The estimation error is defined as the difference between the outputs of the linear classifier and the true values, according to

$$e = y - t = v^T Q - t,$$ \quad (3)

and the mean square error (MSE) is computed according to

$$MSE = \frac{1}{K} \|y - t\|^2 = \frac{1}{K} \|v^T Q - t\|^2, \quad (4)$$

In the least squares approach, the weights are adjusted in order to minimize the MSE. Deriving expression (4) with respect to every weight of the linear combination [11], giving raise to the following expression:

$$v = tQ^T (QQ^T)^{-1}. \quad (5)$$

Finally, since we are in a two-classes problem, the decision rule compares the output of the linear combination to a threshold:

$$d_k = \begin{cases} 1 & y_k > y_0 \\ 0 & \text{otherwise} \end{cases}, \quad (6)$$

where $y_0$ is the threshold value, which is set to 0.5, and $d_k$ is the decision of the system, i.e. high/low stress.

### 2.4 Feature extraction

The main information of EEG signals is described in terms of rhythmic activity, which is divided into frequency bands. The frequency intervals of interest in EEG are the $\theta$ (4-8 Hz), $\alpha$ (8-13 Hz), low $\beta$ (13-15 Hz), middle $\beta$ (15-20 Hz) and high $\beta$ (20-40 Hz) [12]. The proposed features are extracted from these rhythms, which are obtained by filtering the EEG signal with 30-order band-pass IIR filters designed for each frequency interval.

Among the amply variety of EEG features that exists in the literature, we selected as candidate features the next: energy, standard deviation, variance, maximum level, median, 25-th and 75-th percentiles, geometric mean, harmonic mean, mode, range, mean absolute deviation (MAD), skewness, entropy, the mean of the logarithm, the fractal dimension obtained by Higuchi’s algorithm [5, 6], and an epileptic EEG feature like the Hurst exponent [10]. All these features are computed from the different EEG rhythms and from the signal unfiltered, all in the time domain. According to this, the whole features set contains 102 features, 17 from each EEG rhythm and 17 from the unfiltered EEG signal.

The computational cost of the algorithm can be reduced by decreasing the number of features used for classification. In order to select the best subset of features, among the set previously described, a tailored feature selection algorithm has been implemented. The number of features available to classify the stress level leads to a huge number of possible combinations. Consequently, to perform an exhaustive search is not affordable, and a feature selection algorithm based on evolutionary computation is proposed. Evolutionary algorithms exhibit a great potential to solve certain problems that otherwise would be intractable. This type of algorithms are inspired in natural evolution laws such as selection, mutation and crossover, to iteratively search for the optimum solution from the solutions obtained in previous iterations [14].

Figure 3. Raw EEG signal with eye blink (-) and EEG signal with eye blink removed.
This genetic algorithm is started with an initial population of 100 candidate solutions. Each candidate solution represents a subset of features $L'$ selected from $L$ total features, so that only $L'$ rows of $P$ are selected. The number of features selected is MaxFT. The initial population is randomly generated, and the successive population are generated by random crossover and mutations from the best candidates of the previous iteration. The best candidates are selected using the MSE as the fitness function. The complete steps of the genetic algorithm are the next:

1. An initial population of 100 candidate solutions is generated. Each solution is composed of MaxFT random features.

2. In the case that two or more candidates have the same subset of features, they are randomly mutated until the fulfill with the requirement.

3. The fitness function is then evaluated for each candidate solution. It consists in training a LSLC with each subset of features and calculating the associated MSE.

4. Once each candidate solution of the population has been evaluated, a selection process is then applied. A subpopulation of 10% candidate individuals that best fit the fitness function is selected to survive to the next generation. The remaining solutions are removed.

5. The new generation is bred by mixing the parents using a crossover operator. In this way, the 90% new candidates for the next generation are generated by random crossover of the 10% best candidates of the previous generation. The crossover operator is a uniform crossover operator with a crossover probability of 90%, using the half of the elements from the first parent and the second half from the second parent. These elements are randomly selected.

6. Mutations are applied to all new population except the best solution. The mutation operator consists of changing an 1% of the selected features at random.

7. The process is repeated from 2 to 6 until 120 generations are evaluated. The best candidate solution of the final iteration is considered the best solution since the best candidate solution of each iteration is not modified.

A summary in Table 1 complete this description.

### 3 Experimental work

#### 3.1 Database generation

The suitable database design plays a vital role in any kind of problem based on supervised machine learning. In order to validate the algorithms proposed in this work, a database with EEG signals recorded in 4 mentally healthy subjects under high and low stress levels has been generated. The subject were playing the Tetris video-game while their EEG signal were being recorded. We chose this video-game because it allows the generation of a progressive stress level as the test subjects play through progressive difficult levels (10). Test subjects started the experiment at the 3rd level in the game. This level is considered easy, and low stress signals were considered between the 3rd and the 5th levels. High stress was considered as the level when the test subject lost the game (always higher than the 5th). The database resultant is composed of 124 files containing EEG signals relative to high stress level and 157 files containing EEG signals relative to low stress level. The length of each file is 1024 samples, equivalent to 2 seconds of time (sampling rate of 512 Hz).

For properly designing and testing the classification system, the database should be split into two different subsets, one for training and another for testing. However, the generated database only contains 281 files, which is a reduced number to halve the database without running the risk of over-fitting. Nevertheless, the database can be extended with bootstrapping techniques, using the ‘K-fold cross-validation’ technique. According to this paradigm, the database is randomly partitioned into $K$ equal size subsets. This technique is used in several EEG studies like [4], where the used of a 5-fold cross-validation is proposed. In this paper, a 10-fold cross-validation is used. By this way, the test set is composed of one of the $K = 10$ subsets, designing the system with the remaining samples. The design and evaluation processes are repeated 10 times, changing the test subject in every iteration. Using this technique, the database size limitation is overcome. The classification error is obtained as the average error for all the iterations.

### 3.2 Results

The feature selection algorithm has been executed 12 runs for each iteration of the 10-fold cross validation, with different number of maximum selected features (MaxFT). For each value of MaxFT, the classifier that obtains lower MSE over the design set has been selected. Table 3 shows the classification error, in the train and test sets, for the different values of MaxFT. We can see how the error in the training set decreases monotonically with the increment in the number of features. However, in the case of the test error, this reduction only happens until MaxFT=15. After

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial population</td>
<td>100</td>
</tr>
<tr>
<td>Crossover probability</td>
<td>0.9</td>
</tr>
<tr>
<td>Mutation probability</td>
<td>0.1</td>
</tr>
<tr>
<td>Max number of generations</td>
<td>120</td>
</tr>
</tbody>
</table>
Table 2. Genetic algorithm probability selection of each feature

<table>
<thead>
<tr>
<th>Feature Selection Algorithm</th>
<th>Energy</th>
<th>Standard deviation</th>
<th>Variance</th>
<th>Max level</th>
<th>Median</th>
<th>25th percentile</th>
<th>75th percentile</th>
<th>Geometric mean</th>
<th>Harmonic mean</th>
<th>Mode</th>
<th>Range</th>
<th>MAD</th>
<th>Skewness</th>
<th>Higuchi’s algorithm</th>
<th>Mean of the logarithm</th>
<th>Hurst exponent</th>
<th>Entropy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raw EEG signal</td>
<td>50%</td>
<td>10%</td>
<td>16.67%</td>
<td>0%</td>
<td>53.33%</td>
<td>6.67%</td>
<td>3.33%</td>
<td>66.67%</td>
<td>0%</td>
<td>66.67%</td>
<td>3.33%</td>
<td>10%</td>
<td>60%</td>
<td>90%</td>
<td>10%</td>
<td>73.33%</td>
<td>23.33%</td>
</tr>
<tr>
<td>α low</td>
<td>53.33%</td>
<td>46.67%</td>
<td>56.67%</td>
<td>56.67%</td>
<td>3.33%</td>
<td>3.33%</td>
<td>3.33%</td>
<td>6.67%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>low β</td>
<td>0%</td>
<td>10%</td>
<td>10%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>middle β</td>
<td>10%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>10%</td>
<td>10%</td>
<td>10%</td>
<td>20%</td>
<td>10%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>high β</td>
<td>0%</td>
<td>10%</td>
<td>20%</td>
<td>20%</td>
<td>3.33%</td>
<td>3.33%</td>
<td>3.33%</td>
<td>20%</td>
<td>20%</td>
<td>76.67%</td>
<td>20%</td>
<td>56.67%</td>
<td>56.67%</td>
<td>36.67%</td>
<td>36.67%</td>
<td>36.67%</td>
<td>36.67%</td>
</tr>
<tr>
<td>θ</td>
<td>0%</td>
<td>10%</td>
<td>20%</td>
<td>20%</td>
<td>3.33%</td>
<td>3.33%</td>
<td>3.33%</td>
<td>20%</td>
<td>20%</td>
<td>76.67%</td>
<td>20%</td>
<td>56.67%</td>
<td>56.67%</td>
<td>36.67%</td>
<td>36.67%</td>
<td>36.67%</td>
<td>36.67%</td>
</tr>
</tbody>
</table>

In Figure 4, the relation between false positive ($P_{FA}$) and true positives ($P_D$) of each MaxFT is shown (ROC curves). We can see that the best ROC curves are obtained in the range from MaxFT=15 to MaxFT=25, highlighting the ROC curve generated with MaxFT = 25 for low values of $P_{FA}$. Values of MaxFT larger than 25 and smaller than 15 produce worse ROC curves.

### 4 Conclusions

The best classification performance has been found to be about 19.18% error in the training set and 29.18% in the test set, by using only 15 features. The results obtained have demonstrated that the features that show best performance for the classification problem at hand are the fractal dimension and range calculated in the unfiltered EEG signal, the Hurst exponent and the standard deviation of the α rhythm, the energy of the θ rhythm and the entropy and geometric mean of the high β rhythms.
mean of the high $\beta$ rhythm. The best number of maximum selected features is in the range from 15 to 25, generating a better true positive rate against false positive rate in the case of 25 selected features. In this paper, we demonstrate that it is possible to make a computationally-efficient stress detection system based on only 2-sensor EEG signals, using supervised machine learning. We have designed different classifiers, with different number of features, in order to differentiate between two stress levels (high/low).

More complex classifiers, such as SVMs or neural networks have not been implemented in this paper due to the limited number of files in the database, which implies a risk of over-fitting in the case of complex classifiers. In order to test the suitability of these kind of classifiers, more data is required, which will be studied in future works.

Acknowledgments

This work has been funded by the University of Alcala under project CCG2013/EXP-074.

References


