Biomedical Signal Processing and Control xxx (2017) xxx–xxx

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

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

Classification algorithms analysis for brain–computer interface in drug craving therapy

Mirko Mazzoleni [a,*], Fabio Previdi [a], Natale Salvatore Bonfiglio [b]

[a] Department of Management, Information and Production Engineering, University of Bergamo, Via Galvani 2, 24044 Dalmine, BG, Italy
[b] Department of Psychology, University of Pavia, Italy

ARTICLE INFO

Article history:
Received 9 December 2015
Received in revised form
16 December 2016
Accepted 24 January 2017
Available online xxx

Keywords:
Machine learning
Pattern recognition
Brain–computer interface
Signal processing

ABSTRACT

This paper presents a novel therapy to recover patients from drug craving diseases, with the use of brain–computer interfaces (BCIs). The clinical protocol consists of trying to mentally repel drug-related images, and a Stroop test is used to evaluate the blue therapy effect. The method requires a BCI hardware package and a software program which communicates with the device. In order to improve the BCI detection rates, data were collected from five different healthy subjects during the training. These measurements are then used to design a better classification algorithm with respect to the default BCI classifier. The investigated algorithms are logistic regression, support vector machines, decision trees, k-nearest neighbors and Naive Bayes. Although the low number of participants is not enough to guarantee statistically significant results, the designed algorithms perform better than the default one, in terms of accuracy, F1-score and area under the curve (AUC). The Naive Bayes method has been chosen as the best classifier between the tested ones, giving a +12.21% performance boost as concerns the F1-score metric. The presented methodology can be extended to other types of craving problems, such as food, pornography and alcohol. Results relative to the effectiveness of the proposed approach are reported on a set of patients with drug craving problems.

1. Introduction

The term *craving* refers to the impulsive desire for a psychoactive substance, food or any other rewarding thing [1–4]: this supports the "additive" behaviour and the compulsion aimed to avail oneself of the object of desire. Initially used by the dependent subjects to describe a strong and irrepressible opiates urge during abstinence periods [1,2,5], it has assumed subsequently the meaning of a longing for whichever psychotropic substance, in any situation [2,5]. Some authors have highlighted strong differences between the meaning that patients give to the term, and the interpretation given by the clinical staff [1]. In this work, craving is considered as a strong desire or an intense longing, as suggested by [2]. Two models have been proposed to explain the mechanism with which the craving would contribute to cause a relapse. The first one suggests that this need shares common characteristics with the obsessive-compulsive disorder (OCD) [3]. The second model tries to explain the craving as being induced by conditioning phenomena with positive and negative reinforcement mechanisms [4]. In the drug or alcohol abuse therapy, it is very important to detect craving, in order to intervene as soon as possible [5]. However, usually the patient must face alone the craving condition and it is of paramount importance that he/she is able to take the correct actions. In this view, reinforcement of will could be an important tool.

Brain–computer interface (BCI) systems can be used to train the patient's will and enhance his/her capability of overcoming a craving condition. In this work, a BCI system is used as a tool in drug addiction therapy. This aims to increase the ability of autonomously dealing with craving situations. The use of BCIs can be beneficial for both clinical purposes, as in case of people suffering from amyotrophic lateral sclerosis (ALS) [6], and for playful ones [7]. Brain–computer interfaces are composed by sensors (the headset) and algorithms that perform pattern identification and classification. According to the brain activities that have to be detected, two main approaches exist to control a BCI [8]. In the first approach, subjects are exposed to a sequence of stimuli, while focusing on a particular one of them. When this target is recognized, an event related potential (ERP), which represents an electrophysiological response to a specific stimulus, can be detected. Appearing 300 ms after a surprising or task-relevant event, and observed in EEG signals, the P300 is an example of such brain patterns [9]. Other

* Corresponding author.
E-mail addresses: mirko.mazzoleni@unibg.it (M. Mazzoleni), fabio.previdi@unibg.it (F. Previdi), salvo.bonfiglio@unipv.it (N.S. Bonfiglio).

http://dx.doi.org/10.1016/j.bspc.2017.01.011
1746-8094/

examples of ERPs consist of steady-state visual evoked potentials (SSVEPs) [10] and motor-related potentials (MRPs) [11]. The BCI system therefore has to identify when the target stimulus has happened. The second approach consists of imagine a specific mental task, for example a hand movement or other actions. These activities are reflected by oscillatory brain waves in EEG data, such as the delta [0.5–4 Hz], theta [4–8 Hz], alpha and mu [8–13 Hz], beta [13–30 Hz], and gamma [31–40 Hz] rhythms. The BCI system must correlate the mental tasks with the features of the brain waves. Other types of signals, resulting from the focus on mental task, are slow cortical potentials (SCPs) [6] and neuronal ensemble activity [12]. Although scalp-recorded EEG represent the majority of adopted signals for BCI systems, other types of recordings can be used, such epidural, subdural or intracortical. For a survey on the topic, see [13].

Machine learning algorithms are strongly used during the signal processing stage in BCI applications, mainly for feature extraction, pattern recognition and classification. The term "machine learning" refers to the scientific discipline of "learning from data using machines", which traces its origins from computer science, statistics, engineering and artificial intelligence [14]. In BCI applications, machine learning techniques are often used [8,11,13,15]. In fact, it can be assumed that different brain activities lead to different actions. So, there are patterns, i.e. there is a relationship between signals and actions. However, the analytical function which maps this relationship is hard to be described analytically. But there are data available, i.e. the BCI signal measurements, so the mission is to tame this numbers and retrieve something useful from them to solve a specific problem. In the view of a BCI system, a machine learning algorithm is expected to distinguish between different mental statuses. This is accomplished by training a classification algorithm with data measured from the BCI electrodes. Since the subject knows what he/she was thinking, this information can be used to drive the algorithm training. This is known as supervised learning; otherwise the tuning is done in an unsupervised way.

As a *first contribution* of this paper, a therapy for drug craving dependency using BCIs is introduced and its protocol developed. The therapy is intended to be personalized for each patient, feasible at their homes and outside the clinic. This can accustom patients to enacting the behavioural intervention even outside the place in which they are assisted, making the BCI an enabling technology as discussed in [5]. As a *second contribution*, various types of classification algorithms have been investigated, comparing their performance between multiple subjects and with the default algorithm of the headset device, in order to find the best one for therapy in consideration. Related work on the selection of features and classification methods on BCI systems is presented in [15]. Algorithms such as Logistic Regression [16], support vector machines [17], decision trees [18], K-nearest neighbors [19] and Naive Bayes [20] have been compared. These classifiers have been chosen as representatives of different categories: linear, non-linear, tree models, instance-based and probabilistic. A preliminary version of the work in this paper appeared in [21] by the same authors. However, this paper goes beyond that work by analyzing the developed algorithms for more than one subject, and comparing their performance with that of the BCI default software.

The remainder of the paper is organized as follows. In Section 2, the use of BCIs for the therapy of drug addiction diseases is explained. In Section 3, the steps involving the data acquisition and experimental setup are described, while Section 4 highlights the preprocessing and feature extraction methodology. Section 5 shows a comparison between different classifiers for each subject, with indications about the algorithms implementation details. Section 6 analyzes the various algorithms performance by looking at common metrics, and presents an assessment of the therapy effectiveness on patients suffering from drug craving. Section 7 discusses the experimental results and limitations of the approach. Finally, Section 8 is devoted to concluding remarks and future developments.

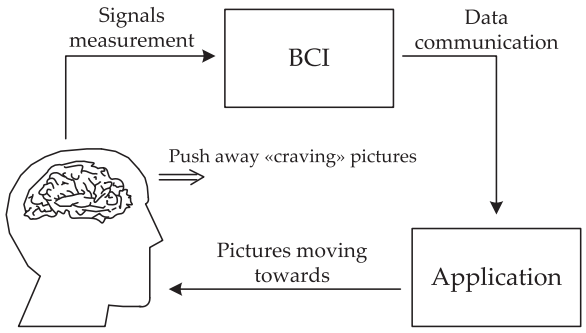

Fig. 1. Use of BCI for craving treatment.

2. Drug craving therapy

In this section, the therapy protocol and rationale are outlined, describing how and on which tasks the device has been trained. This work proposes the use of the BCI as an instrument to recover patients from drug craving diseases. The therapy consists of two consecutive moments: during the first activity, depicted in Fig. 1, the subject performs first a training of the device, and then uses the BCI in an active way, by trying to push away pictures which have a link with his disease. The second moment of the therapy is related to the treatment assessment. This is done via the emotional Stroop test (see [22] for a review of the topic).

The training operation is done at the beginning of every treatment session, in order to make the algorithm more reactive to the patient current mental state. It was experimentally observed, on healthy subjects, that the time of response and accuracy (for a given task) was higher on a freshly trained BCI, with respect to an algorithm trained even a few days before the test. This condition implies that the training session cannot be bore for too much time. Therefore, the experiments done in Section 3 consisted in trying to replicate the exact procedure to which patients will be exposed. The training is done on the basis of two mental tasks: a *neutral task* and a *push task*. In the neutral task, the subject does not have to focus on a particular activity, while in the push task he is required to think about pushing further away an object. The choice of focusing on this two tasks is due to the fact that most BCI systems require the user to train a neutral action first. Furthermore, concentrating on only one of the two tasks makes the training phase lighter and faster, which can be beneficial for patients. Then, other actions could be trained. The ability to train correctly the BCI for two classes is not time consuming. Relying on the experiments carried out with the BCI equipped software, few minutes are required to perform a full training. Just by only adding a third action, the training time grows in the range of hours. This is not acceptable in terms of a treatment of people who could be mentally unstable. By considering the neutral activity as a proper mental task, this problem has been overcome. During the treatment, via the developed software which interacts with the BCI (Fig. 2), the patient undergoes to a series of different images, loaded through it. These pictures could represent images related with his personal craving disease, or, instead, pictures of relaxing landscapes and not harmful scenery. If a picture of the former type appears, the subject has to actively try to repel it, thinking the push task: the picture then will be visually pushed away from the screen. If the patients succeeds to dwindle enough the image, with respect to a set threshold level, a new picture will be prompted. In case of pictures which represent "good" scenes (these are, as the craving scenes, subject dependent) the subject

Please cite this article in press as: M. Mazzoleni, et al., Classification algorithms analysis for brain–computer interface in drug craving therapy, Biomed. Signal Process. Control (2017), http://dx.doi.org/10.1016/j.bspc.2017.01.011

Fig. 2. Subject performing the trials with the developed software.

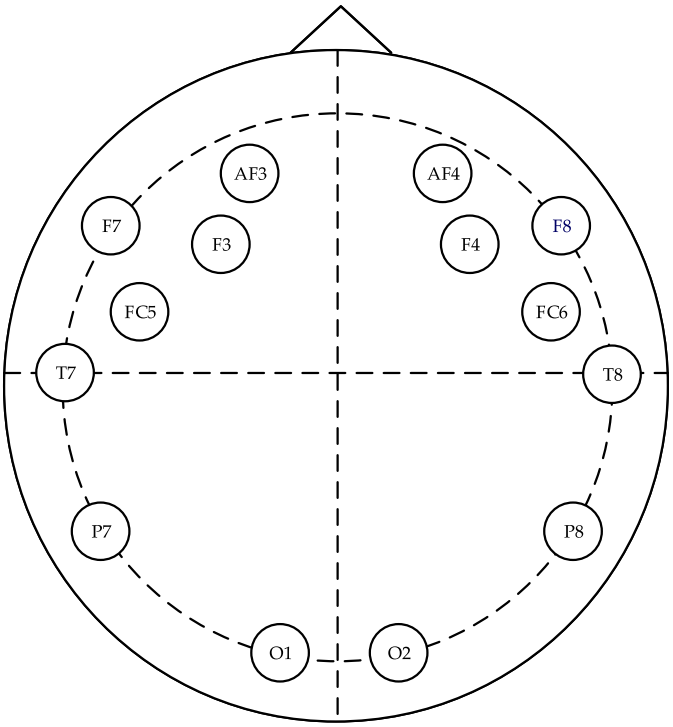

Fig. 3. BCI headset sensor displacement and naming.

does not have to think the push task, but let the system detect the neutral activity. Whenever the push thought is not detected by the BCI, the software makes the pictures bigger and bigger, giving the impression of their approaching. When the image reach a maximum dimension, it is switched to an another one. The developed software permits to set the maximum time of exposure to a picture, and the maximum time in which a picture can reach its biggest dimension before it will be replaced by a new one. During the experiments, the former time was set to 6 s, guided by psychologists indications. The maximum time for which a picture is displayed on screen was set to 60 s. The emotional Stroop test consists of displaying font-colored words, for a defined time, to the user. Two different colors were chosen. The user has to select the color of the words prompted. The time elapsed between the words appearance on screen and the response given is stored. When a word which meaning refers to something related to the craving disease, the "elaboration time" to give a response is affected by the word meaning, because it could be something which is of interest for the subject. This phenomenon is known as "attentional bias". By measuring these reaction times before and after the therapy, a degree of "drop of interest" for the craving-related words could be assessed. A second software program has been developed in order to execute the test, and the user interacts with it via the directional arrows of the keyboard, choosing the one which reflects the word font color.

3. Experimental setup

The experiments involved 5 healthy subjects, chosen at random to reduce the risk of sampling bias. The choice of a odd number of subjects is due to the fact of having the possibility to draw conclusions based on a majority vote. Data were collected from 14 EEG channels located on an Emotiv Epoc BCI headset, sampled at 128 Hz. The device has an output band of [0.2–45 Hz]. Measurements were already filtered via a notch filter at 50 Hz and 60 Hz to further reduce interferences. Fig. 2 depicts the software interface and the test setup, while Fig. 3 reports the headset channel displacement and configuration.

The training phase consisted, in total, of less than 5 min for every subject, with training sessions of 8 s. This requirement is mandatory to avoid an excessive stress for the patients, since the training has to be done for each training session. In this phase, each person had to train the device to recognize a neutral and a push thought. During the first mental task, the thoughts were of not particular attention, while in the second one they were related to pushing away a wooden box with both hands. Participants were asked to train the push thought with a level of training at least of 50% (reported directly by the headset own software algorithm). For the neutral task, the equipped BCI software does not present a percentage: neutral training sessions were then done in order to reach the same number of push sessions. Test data were collected from a simulation of the real therapy: the subjects were prompted 4 images, 2 drug related and 2 peaceful ones. This therapy is then repeated several times, collecting both the true picture label and the decision taken by the BCI default algorithm. This will serve to compare its performance with those of the developed machine learning classifiers.

A summary of the training results for each subject is reported in Table 1. Differences in the neutral and push recording times within a specific subject are due to software computations, which can delay the moment in which the data saving routine is stopped. Subject 4 did an extra neutral recording with respect to the push one. Intra-subject differences are due to the different number of trials in order to reach a training percentage of more than 50%.

4. Feature extraction

After the acquisition phase, data were filtered with a 2-order Butterworth high-pass filter, centered at 0.6 Hz, in order to completely remove the DC component. The next step is the feature extraction phase. For a detailed list of methods used in feature extraction for the processing of data measured from BCIs, see [23]. Features have been extracted by means of sliding windows, with window length of 1 s and window overlapping time of 0.5 s. Therefore, each window selects a batch of data samples, on which various features are computed. Since there are multiple channels, the sliding window is applied for every data source. These times have been chosen by thinking about a compromise between the number of data available for each window (the higher the better), and the response time of the classification algorithm (the shorter the better). Then, for every data window of each channel, the spectral power in the frequency bands [0.5–3.9 Hz], [4–7.9 Hz], [8–12.9 Hz], [13–30.9 Hz] and [31–43 Hz] has been computed. These bands correspond to the brain rhythms cited in Section 1. By extracting

Please cite this article in press as: M. Mazzoleni, et al., Classification algorithms analysis for brain–computer interface in drug craving therapy, Biomed. Signal Process. Control (2017), http://dx.doi.org/10.1016/j.bspc.2017.01.011

Table 1
Training and therapy results.

Subject	Neutral recording [s]	Push recording [s]	Therapy recording [s]	Training level
1	34	34	124	63%
2	36	36	124	58%
3	56	58	100	75%
4	88	71	124	65%
5	44	45	124	52%

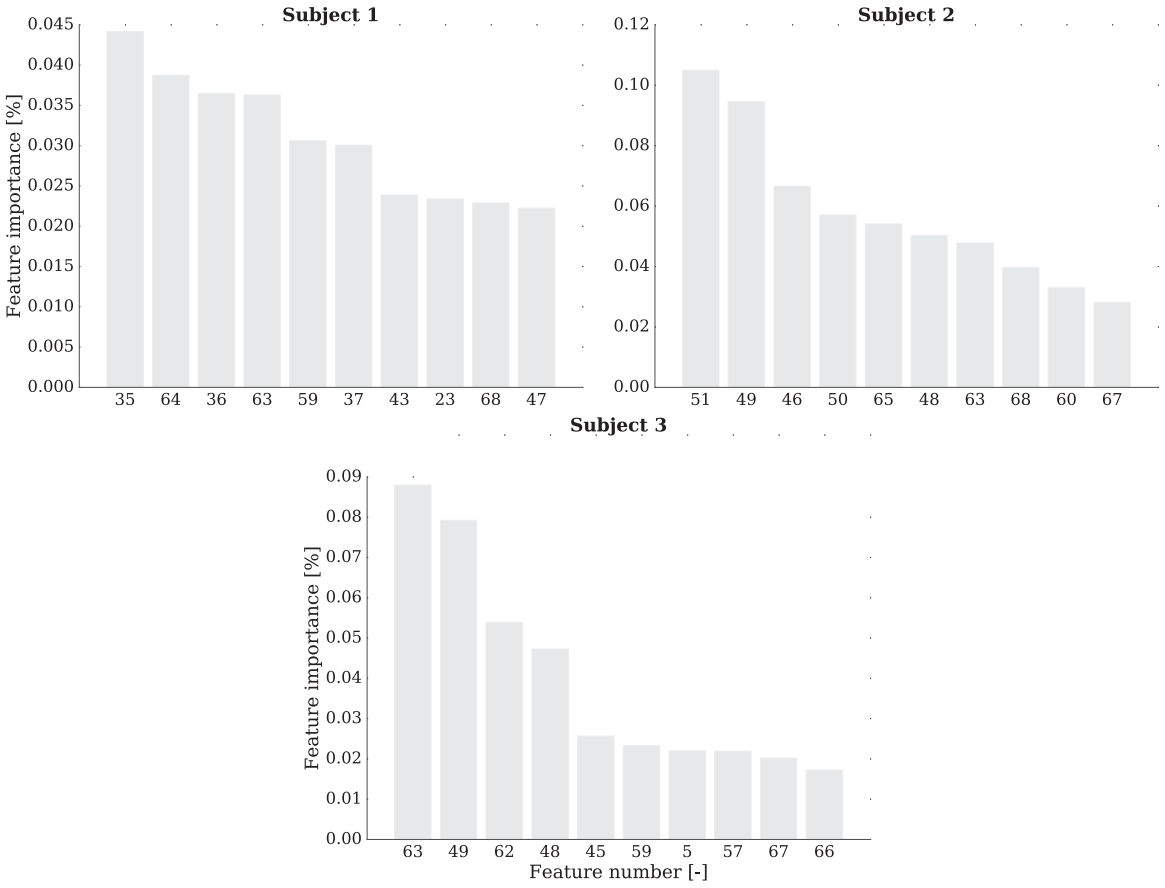

Fig. 4. Most important features for subject 1, subject 2 and subject 3.

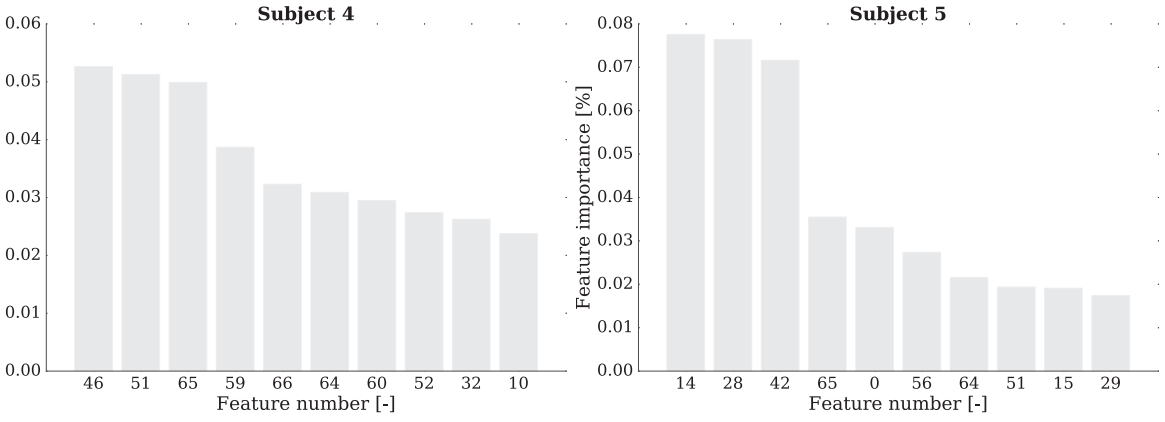

Fig. 5. Most important features for subject 4 and subject 5.

5 frequency features for 14 channels for each batch of data, a 70-dimensional vector is created for each data window, while the number of observations varies slightly with the subject (see Table 2).

After the construction of the feature space, the feature selection step is performed by first fitting an Extreme Randomized Tree (ERT) classifier [24], then by taking only the 10 most important features, ranked by the tree. The choice of retaining only 10 features concerns

Please cite this article in press as: M. Mazzoleni, et al., Classification algorithms analysis for brain–computer interface in drug craving therapy, Biomed. Signal Process. Control (2017), http://dx.doi.org/10.1016/j.bspc.2017.01.011

Table 2
Observations extracted for each subject.

Subject	Training observations	Test observations
1	136	227
2	143	223
3	228	186
4	318	227
5	179	232

the generalization ability of the classifiers that will be designed in the next stages. The VC-dimension (d_{VC}) [25] is a measure of a classifier degrees of freedom, and it is defined as the cardinality of the largest set of points that the algorithm can shatter. An higher d_{VC} shows that a statistical learning algorithm is able to approximate more complex functions, with respect to a classifier with lower VC-dimension. Thus, the d_{VC} is directly related to the well known bias-variance tradeoff in machine learning research. Given this measure, a probabilistic upper bound can be given on a classification model test error. Considering the rule of thumb of having a number of training data at least $10 \cdot d_{VC}$ in order to ensure good generalization [26], this requirement is met for all subjects, each one of them having at least one hundred observations. The choice of the ERT algorithm is due to the fact that it reduces the variance of the feature importance estimate, by averaging the estimation over several randomized trees. This procedure helps to identify the *specific* features for each single subject, which can differ from another one. A summary of extracted features for every subject is depicted in Figs. 4 and 5.

The selected features have been standardized using a robust scaler methodology, which removes the median and scales the data according to the interquantile range (IQR), that is, the range between the 1st quartile (25th quantile) and the 3rd quartile (75th quantile). This procedure is robust against outliers, which can deceive the standard scaling to zero mean and unit variance. The median and the IQR were computed *only* on training data, and then the transformation was applied to the training data *and* the test data. The whole process is depicted in Fig. 6 to better clarify the methodological steps. By considering that a feature number corresponds to a specific frequency band power detected by a specific sensor, and representing it as *Frequency Band Power*$_{Channel\ Number}$, the most important features can be lead back to a specific wave in a specific region of the brain, and opens for further investigations. The sensors, depicted in Fig. 3, have been saved in the following order: [*AF*3, *F*7, *F*3, *FC*5, *T*7, *P*7, *O*1, *O*2, *P*8, *T*8, *FC*6, *F*4, *F*8, *AF*4]. Therefore, the first feature, numbered "0" and indicated as α_0, represents the power of the α-wave detected by the *AF*3 sensor and so on.

5. Classifiers

In this section, for every subject, several classifiers for the BCI system have been designed. The algorithms have to discern between two mental tasks: a neutral one, and a thought of pushing away an object. Along with the parameters of each trained algorithm, their training time is reported, observed on a personal computer with an i7–2.30 GHz quad-core CPU, 16 GB RAM, in a Python environment.

5.1. Classifiers evaluation

The evaluation of a classifier is performed by analyzing the confusion matrix reported in Table 3. For a binary classifier the precision *P* is defined as the ratio between the number of true positive observations and the number of observation predicted as positive by the algorithm (Eq. (1)):

$$P = \frac{True\ Positive}{Predicted\ Positive} = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (1)$$

The recall *R* is defined as the ratio between the number of true positive observations and the number of observation which are actually positive (Eq. (2)):

$$R = \frac{True\ Positive}{Actual\ Positive} = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (2)$$

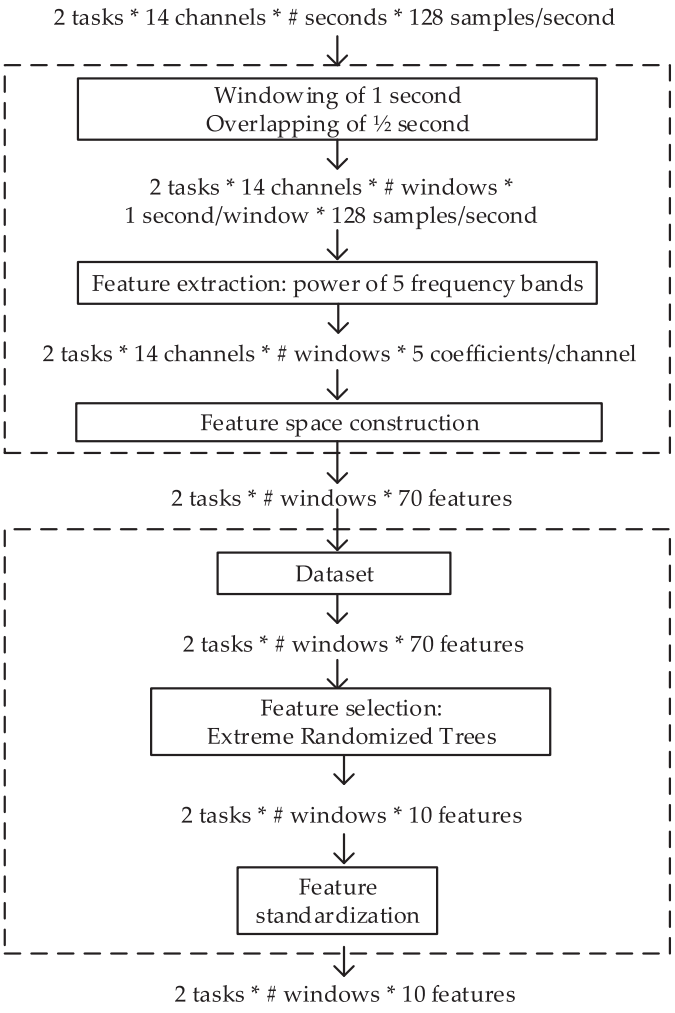

Fig. 6. Sequence diagram for the BCI classification algorithm design.

Table 3
Confusion matrix.

		Predicted value: 1	Predicted value: 0
Actual value	1	True Positive	False Negative
	0	False Positive	True Negative

The F1-score is defined as a combination of precision and recall (Eq. (3)):

$$\mathrm{F1} = 2 \cdot \frac{PR}{P+R} \tag{3}$$

5.2. Logistic regression

Logistic regression is one of the most known methods in machine learning and represents, despite its name, a binary linear classification algorithm. Consider the typical form of a linear model:

$$s = \sum_{i=1}^{d} w_i x_i = \boldsymbol{w}^T \boldsymbol{x} \tag{4}$$

with $s \in \Re$ the model output and $\boldsymbol{w},\boldsymbol{x} \in \Re^d$ representing the model coefficients and regressors in the d-dimensional space. The hypothesis $h(\boldsymbol{x}) = \theta(s)$ is learned, where $\theta(\cdot)$ represents the logistic function, such that:

$$\theta(s) = \frac{e^s}{1+e^s} \tag{5}$$

The solution $h(\boldsymbol{x})$ is obtained by minimizing the cross-entropy error cost function:

$$J(\boldsymbol{w}) = \frac{1}{N} \sum_{n=1}^{N} \ln(1 + e^{-y_n \boldsymbol{w}^T \boldsymbol{x}_n}) \tag{6}$$

with N the number of training observations and $\boldsymbol{x}_n$ the nth training point. The output of a logistic regression is therefore bounded between [0–1], and represents a genuine probability that a point $\boldsymbol{x}$ will belong to one of the two classes. If this probability is greater than 0.5 (the decision threshold), the point will be classified as belonging to class 1, otherwise to class 0. The algorithm configuration chosen is a logistic regression with a L1-penalty term in order to further reduce the d_{VC} of the classifier. This type of regularization is in fact known to produce sparse output with respect to a L2-regularization, which tends to produce small weights [16]. The degree of regularization is controlled by the parameter C. This coefficient has been chosen via a 3-fold cross-validation: the choice is motivated accounting for the balance between the number of training and validation data which will be used by the routine; furthermore, this type of operation makes no assumptions about the data. The cost function used during the cross-validation was the F1-score. This, as opposed to accuracy, better represents the case of unbalanced dataset and considers both precision and recall of the classifier. The training of the classifiers took a mean time of 0.128 s, with a standard deviation of 0.007 s.

5.3. Support vector machines

Support vector machines (SVMs) belongs to the class of binary maximum margin classifiers. The aim is to find a linear classifier which separates the data, but not just any: the chosen boundary is the one that leaves more "space" between the nearest data points of the two classes. This has an implication on the VC-dimension, because the effect is that of shrinking the hypothesis space [26]. SVMs have therefore the nice property of "not paying so much" for the data dimensionality in terms of the VC-dimension. The optimal hyperplane is then found through an optimization problem of the form:

$$\begin{aligned} &\text{Minimize} \quad \frac{1}{2}\boldsymbol{w}\boldsymbol{w}^T + C\sum_{n=1}^{N}\xi_n \\ &\text{subject to} \quad y_n(\boldsymbol{w}\boldsymbol{x}_n + b) \geq 1 - \xi_n \quad n = 1, \ldots, N \\ &\qquad\qquad \xi_n \geq 0 \quad n = 1, \ldots, N \end{aligned} \tag{7}$$

with $\boldsymbol{w} \in \Re^d$ and $b \in \Re$ the hyperplane are the weight vector and bias, $\boldsymbol{x}_n$ is the nth training point and $\boldsymbol{\xi} \in \Re^N$ represents the slack variables introduced to cope with non-linear separable datasets, given N the number of training points. The coefficient $C \in \Re$ has a role similar to the regularization parameter of the logistic regression. But it is only when equipped with a powerful mathematical tool called *kernel* that SVM can unleash all their power: it is indeed possible to exploit the expressiveness of non-linear transformation, paying for such a big dimensionality only in terms of the number of *support vectors*, that is, training points for which the corresponding Lagrange multiplier is not null, and "support" the decision boundary. The trained classifier adopted has been a radial basis function (RBF) kernel, with RBF parameters γ and C, chosen via 3-fold cross-validation. Having to solve a quadratic optimization problem, the SVM algorithm can take longer times than logistic regression, especially as the number of observation grows. The training of the classifiers took a mean time of 0.128 s, with a standard deviation of 0.005 s.

5.4. Decision tree

Decision trees divide the feature space into regions via a greedy approach, known as recursive binary splitting, where each region is a leaf of the tree. For classification trees, the predicted class for each observation is given by the most commonly occurring class of training observations that belong to the same terminal node. In order to define the binary split, usually the Gini index is used:

$$G = \sum_{m=1}^{M} \hat{p}_{rm}(1 - \hat{p}_{rm}) \tag{8}$$

where $\hat{p}_{rm}$ represents the proportion of training observations in the rth region that are from the mth class, with M the number of classes. A small value of G indicates that a node contains predominantly observations from a single class. If a node contains only elements of one single class, it is pure. At each step, the feature which leads to the purest nodes generation is chosen for the splitting. Decision trees are highly interpretable: indeed they can be easily visualized and explained. However, they tend to overfit: to overcome this, the tree has been trained by choosing the maximum tree depth, the minimum samples per leaf and the number of samples required for splitting a node via 3-fold cross-validation. As noticed in the preprocessing stage, a decision tree can be used to perform a feature selection step. All classifiers took 0.125 s to train.

5.5. k-Nearest neighbour

The k-nearest neighbour (k-NN) is a non-parametric instance-based algorithm which classifies a point by first considering the k nearest vectors to that point. The data space is partitioned based on the training dataset, and the test set is evaluated into these partitions. An advantage of this rule is that it is simple and intuitive, easy to implement and there is no "training phase". The main disadvantage is the computational overhead. It can be shown that the VC-dimension of the classifier is infinite [14], so there are no guarantees of generalization by minimizing the training error. However,

Please cite this article in press as: M. Mazzoleni, et al., Classification algorithms analysis for brain–computer interface in drug craving therapy, Biomed. Signal Process. Control (2017), http://dx.doi.org/10.1016/j.bspc.2017.01.011

Table 4
Classifiers accuracy report.

Subject	Default algorithm	Logistic regression	SVM	Tree	k–NN	Naive Bayes
1	0.61	***0.63***	***0.63***	0.53	0.58	***0.62***
2	0.70	***0.80***	***0.80***	***0.72***	***0.79***	***0.84***
3	0.44	***0.52***	***0.57***	***0.60***	***0.53***	***0.52***
4	0.56	0.50	***0.58***	0.50	0.53	***0.58***
5	0.46	***0.55***	0.45	0.46	0.43	***0.47***

Table 5
Classifiers F1-score report.

Subject	Default algorithm	Logistic regression	SVM	Tree	k–NN	Naive Bayes
1	0.61	***0.63***	***0.63***	0.52	0.58	***0.63***
2	0.68	***0.80***	***0.81***	***0.72***	***0.80***	***0.84***
3	0.33	***0.45***	***0.49***	***0.60***	***0.48***	***0.43***
4	0.56	0.50	0.55	0.49	0.53	***0.58***
5	0.44	***0.55***	0.43	0.37	0.42	***0.46***

Table 6
Classifiers AUC score report.

Subject	Logistic regression	SVM	Tree	k–NN	Naive Bayes
1	***0.68***	0.67	0.53	0.62	***0.68***
2	0.90	0.90	0.74	0.86	***0.91***
3	0.51	0.56	***0.60***	0.50	0.55
4	0.53	0.54	0.53	0.54	***0.60***
5	***0.54***	0.45	0.47	0.42	0.44

Table 7
ANOVA results on the F1-score metric.

Classifier	Δ Mean effect	Δ Std. deviation	Percentage variation
k-NN	0.038	0.092	+7.25%
Logistic regression	0.062	0.092	+11.83%
Naive Bayes	0.064	0.092	+12.21%
SVM	0.058	0.092	+11.06%
Decision tree	0.016	0.092	+3.05%

the error committed by the 1-NN classifier is (asymptotically) at most twice that of the optimal Bayesian one [14].

Since there are 2 classes, it is convenient to choose an odd number of neighbors to avoid indecisions on which class to classify the points. Therefore, via a 3-fold cross validation, the k parameter is chosen for every subject. The computation of distances, with the Euclidean distance as metric, has been done with the "Ball Tree" algorithm which has a computational time of $O(d \log N)$. The training of the classifiers took a mean time of 0.123 s, with a standard deviation of 0.0005 s.

5.6. Naive Bayes

The Naive Bayes algorithm is a probabilistic classifier which derives from the optimal Bayesian one. Let ω_1 and ω_2 be the two classes in which the patterns belong, and $P(\omega_1)$ and $P(\omega_2)$ the a-priori class probabilities. Being $p(\boldsymbol{x}|\omega_m)$, $m = 1, 2$, the likelihood of ω_m with respect to $\boldsymbol{x}$ (being $\boldsymbol{x}$ a data point) it is possible to use the Bayes rule to obtain:

$$p(\omega_m|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\omega_m)p(\omega_m)}{p(\boldsymbol{x})} \quad m = 1, 2 \tag{9}$$

The classification rule will be then:

$$\begin{aligned} &\text{if} \quad p(\omega_1|\boldsymbol{x}) > p(\omega_2|\boldsymbol{x}) \quad \boldsymbol{x} \in \omega_1 \\ &\text{if} \quad p(\omega_1|\boldsymbol{x}) < p(\omega_2|\boldsymbol{x}) \quad \boldsymbol{x} \in \omega_2 \end{aligned} \tag{10}$$

where $p(\boldsymbol{x})$ can be neglected because it is the same in each class. If the likelihoods are not known, they have to be estimated. This can be a problem in very high data dimensionality like BCI datasets. To simplify this computation, the Naive Bayes classifier assumes that the features are independent, so that:

$$p(\boldsymbol{x}|\omega_m) = \prod_{i=1}^{d} p(x_i|\omega_m)p(\omega_m) \quad m = 1, 2 \tag{11}$$

The chosen model is a Gaussian-likelihood Naive Bayes, because its ease of training and Gaussian distribution has been thought to better represent the BCI features, with respect to Bernoulli or Multinomial likelihoods [27]. The advantage of this classifier is that it does not need parameters to be tuned, but the underline distribution assumptions may not hold always. All classifiers took 0.125 s to train.

6. Experimental results

This section reports a summary of the considered classifiers results, comparing their accuracy performance also with the default headset algorithm (Table 4).

6.1. Classifiers performance

The accuracy performance is 1 when the algorithm is able to correctly classify all the patterns. Furthermore, for each of the 5 subjects, the classifiers are compared via the F1-score (Table 5), receiver operating characteristic (ROC) curve, and the area under the curve (AUC) value (Figs. 7 and 8). The ROC curve plots the false positive rate against the true positive rate of the classifier: a curve which pass for the point [0, 1] would represent therefore a perfect classifier, while the plot diagonal line represents a pure random one. The AUC score is defined as the area under the ROC curve. A perfect model will score an AUC of 1, while random guessing will score an AUC of 0.5. The use of ROC curves is motivated when the proportion of the two classes in the population (the base rate of the class) is not known [28]. In Tables 4 and 5, values of metrics which are better than the default BCI algorithm are depicted in bold font, while in Table 6, the bold font is used to highlight the best performing algorithm, in terms of AUC, for the specific subject.

6.2. Classifier statistical analysis

In order to assess if the tested classifiers are statistically better than the default one, an ANalysis Of VAriance (ANOVA) [29] was performed on results from accuracy, F1-score and AUC. The considered groups are six: the five designed classifiers and the default one. Each group has five observations. The ANOVA procedure tests the null hypothesis that the groups have the same mean value. This hypothesis is rejected if at least one group has a mean which is different from the others. The groups are independent since one classifier is designed separately from the others. The observations inside a group are independent since a healthy subject performance is independent from the results of another subject. The assumption of normality is confirmed via a qq-plot (visually) and through a Shapiro–Wilk test for gaussianity [30]. The assumption of homoscedasticity is confirmed via a Levene test [31]. The ANOVA does not reject the null hypothesis of equal group means. However, the mean value of the designed classifiers is always higher than the mean of the default one. The designed classifiers achieve better performance, although not statistically significant. Table 7 reports the ANOVA results on the F1-score metric. The column "Δ Mean Effect" indicates the difference of the group mean effect with respect to the mean effect of the default classifier, which has an average F1-score of 0.524. Thus, the average F1-score of the Logis-

Please cite this article in press as: M. Mazzoleni, et al., Classification algorithms analysis for brain–computer interface in drug craving therapy, Biomed. Signal Process. Control (2017), http://dx.doi.org/10.1016/j.bspc.2017.01.011

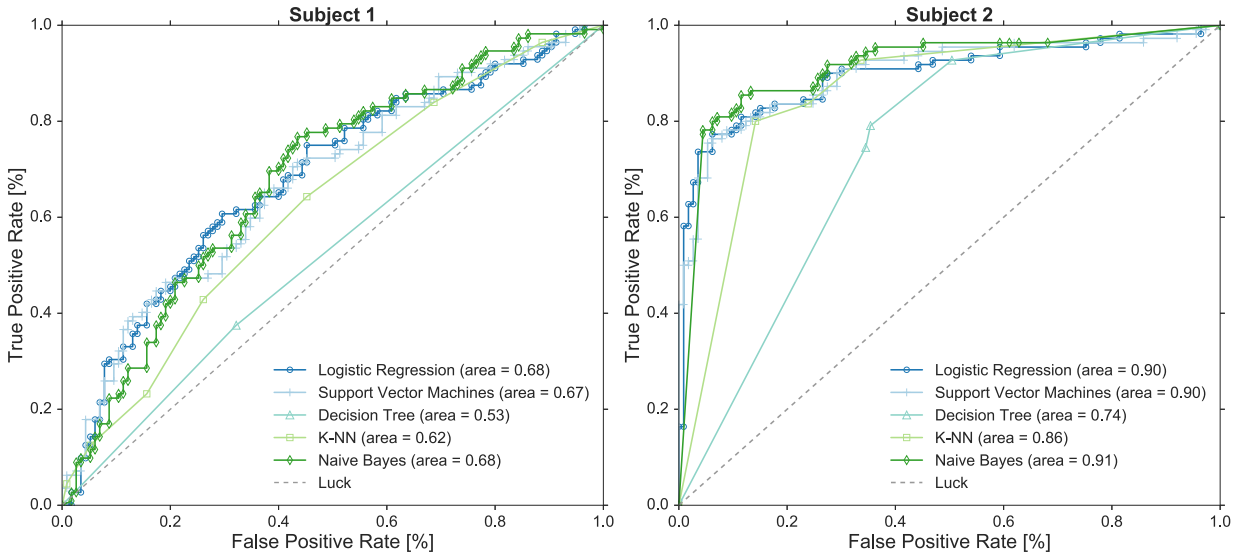

Fig. 7. ROC curves subject 1 and subject 2.

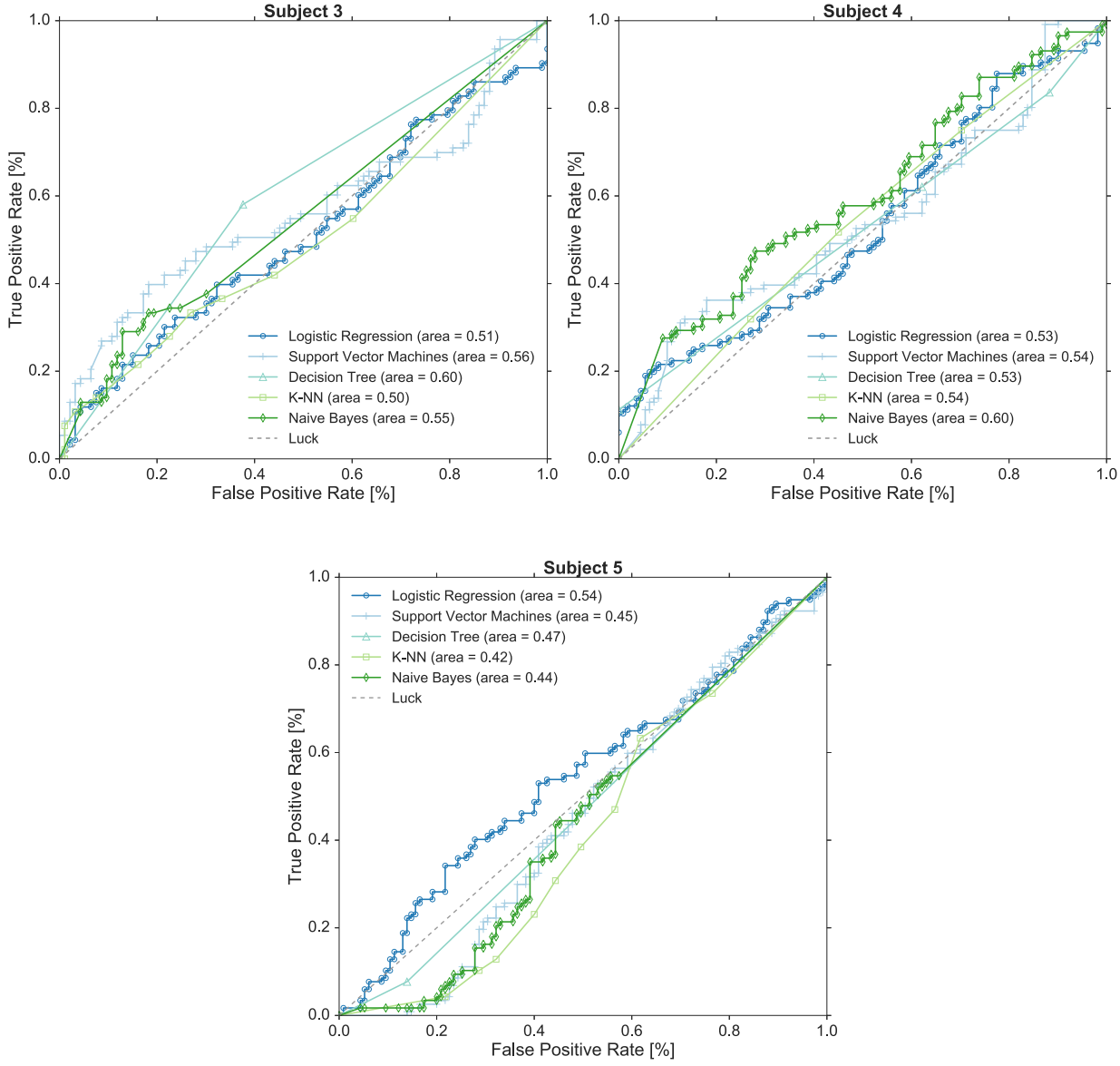

Fig. 8. ROC curves subject 3, subject 4 and subject 5.

Please cite this article in press as: M. Mazzoleni, et al., Classification algorithms analysis for brain–computer interface in drug craving therapy, Biomed. Signal Process. Control (2017), http://dx.doi.org/10.1016/j.bspc.2017.01.011

Table 8
Stroop test results. All timings are expressed in milliseconds.

Patient	$E[t_1^+]$	$E[t_1^-]$	$E[t_4^+]$	$E[t_4^-]$	$\Delta E[t_1]$	$\Delta E[t_4]$	$\Delta E[t_4] - \Delta E[t_1]$
1	344.1	341.2	492.2	479.7	2.9	12.5	***+9.6***
2	514.5	556.4	515.5	455.9	-41.9	59.6	***+101.5***
3	449.9	402.9	443.7	415.0	47.0	28.7	−18.3

tic Regression group is 0.524 + 0.062 = 0.586. The column "Δ Std. Deviation" represents the standard deviation of the group difference. This quantity is always equal because the variables in the ANOVA model are categorical and the number of observations for each group is the same (five observations).

6.3. Therapy assessment

In order to test the effectiveness of the proposed method, the therapy was applied on 3 real patients with craving diseases. All subjects did not present eyesight problems. The visually impaired subjects wore eyeglasses in order to correct their sight, and none of them is colorblind. The therapy is assessed via an emotional Stroop test. This modified Stroop test is an attentional bias procedure [32]. The attention bias index is computed by subtracting the mean reaction time of the neutral words (defined as $E[t^+]$) with the mean reaction time of the stimulus words (defined as $E[t^-]$). Outlier and incorrect times have been eliminated as shown in previous studies [33–38]. The experimental protocol used for performing and testing the therapy consists of four phases:

1 Baseline assessment: Prior to any treatment, reaction times and EEG rhythms are recorded to develop a baseline for the next steps of the therapy.
2 Cue elicitation: In this phase, stimuli related to the substance are presented to the patient. The reaction times and EEG rhythms are recorded.
3 Training: After the first two moments, the patient will actively move the stimuli related to the substance.
4 Recovery: In this phase, the patient will see relaxing images and he will listen to music, in order to lower his physiological and psychological state. After this, the reaction times and EEG rhythms are recorded.

For each subject were created three words lists. Each list is made of 20 words (10 neutral and 10 that elicits craving). The lists were tailored to the subject, because he reports the triggers words associated with his abused substance. The Stroop tests have been therefore personalized. Each word was presented on screen for 300 ms. The aim of the subject is to press the keyboard directional arrow associated to the word font color. The counter starts as soon as the previous word disappeared from the screen. After the key is pressed, the next word is presented, with a latency of 500 ms. Each word was randomly depicted in red or blue. The words were showed on a 12 in. screen, at a distance of 30 cm.

Table 8 reports the results of the Stroop test on three subjects. The notation $E[t_i^+]$ indicates the mean response time for neutral words, recorded during phase i, $i = \{1, 4\}$ of the treatment protocol. The notation $E[t_i^-]$ indicates the mean response time for stimulus words in phase i. The column $\Delta E[t_i]$ represents the difference between neutral and craving words mean times during phase i, that is, $\Delta E[t_i] = E[t_i^+] - E[t_i^-]$. The difference $\Delta E[t_4] - \Delta E[t_1]$ is the expected to be greater than zero if the therapy succeeded, since $\Delta E[t_4]$ will be greater, due to the fact that craving words had a lower interference power. It can be seen how two out of three subjects had a potential benefit in response times. However, the therapy was not effective on subject 3. This preliminary results are encouraging, and more effort is necessary to better assess the therapy effectiveness.

7. Discussion

The following section presents a discussion of the main results and analysis performed, considering also the limitations of the considered approach. Results reported in Tables 4 and 6 show that:

- The analysed classification methodology and algorithms bring an improvement with respect to the default BCI equipped classification software, for all the considered subjects.
- The decision tree and k-NN algorithm have poor overall accuracy and lack of constant performance across subjects.
- The algorithms provide results which are better than the standard BCI algorithm. This is true even for subjects whose performances are slightly better than random guessing, as subject 3, subject 4 and subject 5.
- Considering Subject 2 the analysed classifiers strongly overperform the standard BCI algorithm.
- Support vector machines, despite their higher computational complexity, do not add significant gain in accuracy with respect to logistic regression and Naive Bayes.

The Naive Bayes algorithm could be chosen as the best classifier, because it offers the following benefits:

- It does not need any subject-specific parameter tuning before its use.
- Its accuracy is constantly better than the default algorithm for all the subjects.
- Its F1-score is constantly better than the default algorithm for all the subjects.
- Its AUC score is the best for 3 out of 5 subjects.

However, Table 7 shows that there is not a clearly better classifier, even if all of them perform better than the default one. The limitation can be due to the low number of tested subjects, which makes the statistical power of the test not high enough.

The compared classifiers have been designed by collecting data from healthy subjects. The BCI performance is expected to be worse in case of real patients. It is however expected that the proposed algorithm will be better even for patients, as it is for healthy subjects.

The training data are independent with respect to the default classifier, thus they can be used to train the designed algorithms. The validation of the tested classifiers has been performed offline, as opposite to the accuracy evaluation of the default BCI algorithm. The comparison between the designed algorithms is valid since they are all evaluated in the same manner: their performances are then compared with those of the default classifier. This comparison remains reasonable since the default classifier performance is a worst case scenario in this type of applications.

The computational complexity of the designed algorithms is not a critical point, since the amount of data is very low. The reported computational time for the classifier training is comparable amongst all methods.

Results on real patients have shown that the proposed method can be an effective aid for dealing with craving disease problems. The discussed therapy must not be seen as a stand-alone application, but otherwise it has to be considered as an additional instrument and possibility to recover patients from their diseases.

8. Conclusions

In this paper, the use of brain–computer interface has been introduced in the field of rehabilitation from drug craving diseases. By actively using the instrument to repel illness-related pictures,

Please cite this article in press as: M. Mazzoleni, et al., Classification algorithms analysis for brain–computer interface in drug craving therapy, Biomed. Signal Process. Control (2017), http://dx.doi.org/10.1016/j.bspc.2017.01.011

the patient can push away even its craving desire. In order to recognize when the subject is trying to do this, a machine learning algorithm is mandatory. By using the feature selection capability of the Extra Randomized Tree classifier, the feature selection stage can be done automatically. Several types of classifiers, each one with a different "way of thinking", have been compared offline, and the chosen classifier has been the Naive Bayes. Results have shown that the procedure is sound for more than one subject. The developed method is an overall improvement with respect to the standard BCI equipped classification algorithm, even if results are not statistically significant. The therapy effectiveness has been assessed via an emotional Stroop test on three real patients. Results have shown that the method can be a valid aid to recover patients from craving problems. Further developments consist of replicate the analysis and assessing the efficacy of the therapy with more subjects, in order to confirm the hypothesis traced in this work, and deploying the system in an online environment.

Acknowledgements

This work was partially funded by the University of Bergamo, within the Italy project Bergamo, for health in ageing and globalized society. The ethical board refers to the Department of Brain and Behavioral Sciences, University of Pavia, Italy.

References

[1] T. Sitharthan, D. McGrath, G. Sitharthan, J.B. Saunders, Meaning of craving in research on addiction, Psychol. Rep. 71 (3) (1992) 823–826.
[2] L.T. Kozlowski, D.A. Wilkinson, Use and misuse of the concept of craving by alcohol, tobacco, and drug researchers, Br. J. Addict. 82 (1) (1987) 31–36.
[3] J.G. Modell, F.B. Glaser, L. Cyr, J.M. Mountz, Obsessive and compulsive characteristics of craving for alcohol in alcohol abuse and dependence, Alcohol. Clin. Exp. Res. 16 (2) (1992) 272–274.
[4] J. Littleton, Acamprosate in alcohol dependence: how does it work? Addiction 90 (9) (1995) 1179–1188.
[5] E.W. Boyer, R. Fletcher, R.J. Fay, D. Smelson, D. Ziedonis, R.W. Picard, Preliminary efforts directed toward the detection of craving of illicit substances: the iHeal project, J. Med. Toxicol. 8 (1) (2012) 5–9.
[6] N. Birbaumer, N. Ghanayim, T. Hinterberger, I. Iversen, B. Kotchoubey, A. Kübler, J. Perelmouter, E. Taub, H. Flor, A spelling device for the paralysed, Nature 398 (6725) (1999) 297–298.
[7] E.C. Lalor, S.P. Kelly, C. Finucane, R. Burke, R. Smith, R.B. Reilly, G. Mcdarby, Steady-state VEP-based brain–computer interface control in an immersive 3D gaming environment, EURASIP J. Appl. Signal Process. 2005 (2005) 3156–3164.
[8] U. Hoffmann, J.-M. Vesin, T. Ebrahimi, Recent advances in brain–computer interfaces, in: IEEE International Workshop on Multimedia Signal Processing (MMSP07), no. LTS-CONF-2007-063, 2007.
[9] S. Sutton, M. Braren, J. Zubin, E. John, Evoked-potential correlates of stimulus uncertainty, Science 150 (3700) (1965) 1187–1188.
[10] S.P. Kelly, E.C. Lalor, R.B. Reilly, J.J. Foxe, Visual spatial attention tracking using high-density SSVEP data for independent brain–computer communication, IEEE Trans. Neural Syst. Rehabil. Eng. 13 (2) (2005) 172–178.
[11] J. Donoghue, B. Blankertz, G. Curio, K. Muller, Boosting bit rates in non-invasive EEG single trial classification by feature combination and multi class paradigm, IEEE Trans. Biomed. Eng. 51 (6) (2004) 993–1002.
[12] L.R. Hochberg, M.D. Serruya, G.M. Friehs, J.A. Mukand, M. Saleh, A.H. Caplan, A. Branner, D. Chen, R.D. Penn, J.P. Donoghue, Neuronal ensemble control of prosthetic devices by a human with tetraplegia, Nature 442 (7099) (2006) 164–171.
[13] A. Bashashati, M. Fatourechi, R.K. Ward, G.E. Birch, A survey of signal processing algorithms in brain–computer interfaces based on electrical brain signals, J. Neural Eng. 4 (2) (2007) R32.
[14] R.O. Duda, P.E. Hart, et al., Pattern Classification and Scene Analysis, vol. 3, Wiley, New York, 1973.
[15] H. Bashashati, R.K. Ward, G.E. Birch, A. Bashashati, Comparing different classifiers in sensory motor brain computer interfaces, PLOS ONE 10 (6) (2015) e0129435.
[16] T. Hastie, R. Tibshirani, J. Friedman, T. Hastie, J. Friedman, R. Tibshirani, The Elements of Statistical Learning, vol. 2, Springer, 2009.
[17] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods, Cambridge University Press, 2000.
[18] L. Breiman, J. Friedman, C.J. Stone, R.A. Olshen, Classification and Regression Trees, CRC Press, 1984.
[19] T.M. Cover, P.E. Hart, Nearest neighbor pattern classification, IEEE Trans. Inf. Theory 13 (1) (1967) 21–27.
[20] H. Zhang, The Optimality of Naive Bayes, vol. 1 (no. 2, 2004, pp. 3.
[21] M. Mazzoleni, F. Previdi, A comparison of classification algorithms for brain computer interface in drug craving treatment, IFAC-PapersOnLine 48 (20) (2015) 487–492, http://dx.doi.org/10.1016/j.ifacol.2015.10.188.
[22] A.R. Jensen, W.D. Rohwer, The Stroop color-word test: a review, Acta Psychol. 25 (1966) 36–93.
[23] D.J. McFarland, C.W. Anderson, K. Muller, A. Schlogl, D.J. Krusienski, BCI meeting 2005 – workshop on BCI signal processing: feature extraction and translation, IEEE Trans. Neural Syst. Rehabil. Eng. 14 (2) (2006) 135.
[24] P. Geurts, D. Ernst, L. Wehenkel, Extremely randomized trees, Mach. Learn. 63 (1) (2006) 3–42.
[25] V.N. Vapnik, V. Vapnik, Statistical Learning Theory, vol. 1, Wiley, New York, 1998.
[26] Y.S. Abu-Mostafa, M. Magdon-Ismail, H.-T. Lin, Learning From Data, AMLBook, 2012.
[27] B. Blankertz, G. Dornhege, C. Schäfer, R. Krepki, J. Kohlmorgen, K.-R. Müller, V. Kunzmann, F. Losch, G. Curio, Boosting bit rates and error detection for the classification of fast-paced motor commands based on single-trial EEG analysis, IEEE Trans. Neural Syst. Rehabil. Eng. 11 (2) (2003) 127–131.
[28] F. Provost, T. Fawcett, Data Science for Business: What You Need to Know About Data Mining and Data-Analytic Thinking, O'Reilly Media, Inc., 2013.
[29] D. Freedman, R. Pisani, R. Purves, Statistics, 2007, 1978.
[30] S.S. Shapiro, M.B. Wilk, An analysis of variance test for normality (complete samples), Biometrika 52 (3/4) (1965) 591–611.
[31] H. Levene, Robust tests for equality of variances Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling, vol. 2, 1960, pp. 278–292.
[32] S. Liu, S.D. Lane, J.M. Schmitz, A.J. Waters, K.A. Cunningham, F.G. Moeller, Relationship between attentional bias to cocaine-related stimuli and impulsivity in cocaine-dependent subjects, Am. J. Drug Alcohol Abuse 37 (2) (2011) 117–122.
[33] A.S. Attwood, H. O'Sullivan, U. Leonards, B. Mackintosh, M.R. Munafó, Attentional bias training and cue reactivity in cigarette smokers, Addiction 103 (11) (2008) 1875–1882.
[34] M. Field, B. Eastwood, Experimental manipulation of attentional bias increases the motivation to drink alcohol, Psychopharmacology 183 (3) (2005) 350–357.
[35] M. Field, T. Duka, B. Eastwood, R. Child, M. Santarcangelo, M. Gayton, Experimental manipulation of attentional biases in heavy drinkers: do the effects generalise? Psychopharmacology 192 (4) (2007) 593–608.
[36] E. Smeets, A. Roefs, A. Jansen, Experimentally induced chocolate craving leads to an attentional bias in increased distraction but not in speeded detection, Appetite 53 (3) (2009) 370–375.
[37] R.K. McHugh, H.W. Murray, B.A. Hearon, A.W. Calkins, M.W. Otto, Attentional bias and craving in smokers: the impact of a single attentional training session, Nicotine Tobacco Res. 12 (12) (2010) 1261–1264.
[38] T. Schoenmakers, R.W. Wiers, B.T. Jones, G. Bruce, A. Jansen, Attentional re-training decreases attentional bias in heavy drinkers without generalization, Addiction 102 (3) (2007) 399–405.

Please cite this article in press as: M. Mazzoleni, et al., Classification algorithms analysis for brain–computer interface in drug craving therapy, Biomed. Signal Process. Control (2017), http://dx.doi.org/10.1016/j.bspc.2017.01.011