



# Improving the performance of P300 BCI system using different methods

Islam A. Fouad<sup>2</sup> · Fatma El-Zahraa M. Labib<sup>1</sup> · Mai S. Mabrouk<sup>2</sup> · Amr A. Sharawy<sup>1</sup> · Ahmed Y. Sayed<sup>3</sup>

Received: 16 May 2020 / Revised: 21 August 2020 / Accepted: 26 August 2020 / Published online: 21 September 2020  
© Springer-Verlag GmbH Austria, part of Springer Nature 2020

## Abstract

A brain–computer interface (BCI) can be used for people with severe physical disabilities such as ALS, or amyotrophic lateral sclerosis. BCI can allow these individuals to communicate again by creating a new communication channel directly from the brain to an output device. BCI technology can allow paralyzed people to share their intent with others, and thereby demonstrate that direct communication from the brain to the external world is possible, and that it might serve useful functions. In this paper, we propose a system to exploit the P300 signal in the brain, a positive deflection in event-related potentials. The P300 signal can be incorporated into a spelling device. BCI systems include machine learning algorithms (MLA). Their performance depends on the feature extraction and classification techniques employed. This work discusses the performance of different machine learning algorithms. First, a preprocessing step is introduced to the subjects to extract the important features before applying the machine learning algorithms. The presented algorithms are linear discriminant analysis (LDA I and LDA II), support vector machine (SVM I, SVM II, SVM III, and SVM IV), linear regression (LREG), and Bayesian linear discriminant analysis (BLDA). It is found that BLDA and SVMIV classifiers yield the highest performance for both subjects considered in our study.

**Keywords** Brain Computer Interface (BCI) · P300 signal · Machine learning algorithms (MLA) · Linear discriminant analysis (LDA) · Support vector machine (SVM) · Linear regression (LREG) · Bayesian linear discriminant analysis (BLDA)

## 1 Introduction

Amyotrophic lateral sclerosis or ALS is a progressive neurodegenerative disease that affects nerve cells in the brain and the spinal cord, which often leads to complete paralysis ([www.alsa.org/news/media/quick-facts.html](http://www.alsa.org/news/media/quick-facts.html), 2012).

ALS usually strikes people between the ages of 40 and 70. The incidence of ALS is two per 100,000 people, and it is estimated that about 30,000 people in the United States are living with ALS ([www.alsa.org/news/media/quick-facts.html](http://www.alsa.org/news/media/quick-facts.html), 2012). As the disease progresses many assistive communication devices that have been once a necessity, may become ineffective (McCane et al. 2015).

Brain Computer Interface (BCI) is one of the best research fields that have been developed in the last decades to get a new way of communication for those people (Fouad and Hadidi 2014).

BCI research is based mainly on acquiring signals from the brain and obtaining discriminative information from

---

✉ Islam A. Fouad  
islam\_azeem@yahoo.com

Fatma El-Zahraa M. Labib  
fatma\_zahraa\_mlabib@yahoo.com

Mai S. Mabrouk  
msm\_eng@yahoo.com

Amr A. Sharawy  
amrarsha@gmail.com

<sup>1</sup> Biomedical Engineering and Systems Department, Cairo University, Giza, Egypt

<sup>2</sup> Biomedical Engineering Department, MUST University, Giza, Egypt

<sup>3</sup> Mathematics and Physics Engineering Department, El-Matara, Helwan University, Cairo, Egypt

them, which could help in classifying these signals to various applications (Azar et al. 2014).

This interface creates a direct communication between the brain and the object which is to be controlled by the brain. In case of the P300 speller, the basic purpose of the BCI system is to map the P300 signals into the right character to spell. Therefore, it detects P300 signals and converts these neurophysiologic signals into basic actions (Wolpaw et al. 2003). These actions are then displayed on a computer screen.

An event-related potential (ERP) is a signal observed on the scalp of the subject that occurs when an event such as a visual stimulus happens in a short period (Wu 2014). The ERP for each trial is hypothesized to be synchronous with the event and similar between trials (Wu 2014).

The P300-based speller was originally introduced by Farwell and Donchin in 1988 (Lu et al. 2013). Subjects using the P300 speller select characters from a matrix presented on a computer screen. Then, the system proceeds to analyze and classify the resulting EEG signals (Lu et al. 2013). The flashing of a character being focused on elicits an event-related potential (ERP) that distinguishes between target and non-target characters (Lu et al. 2013). A “target” refers to the character being focused on.

The P300 response is a positive deflection in the EEG over the parietal cortex that appears around 300 ms after a stimulus as shown in Fig. 1 (Lu et al. 2013). Groups of characters are successively and repeatedly flashed, but only the group that contains the target character will elicit a response (Mattout et al. 2014).

In the implementation of a P300 BCI, characters are grouped for flashing as rows and columns (Townsend et al. 2010). This orientation is referred to the row–column paradigm (RCP) (Townsend et al. 2010). The computer can identify the target character as the intersection of the row and column that induced the largest P300 (Townsend et al. 2010).

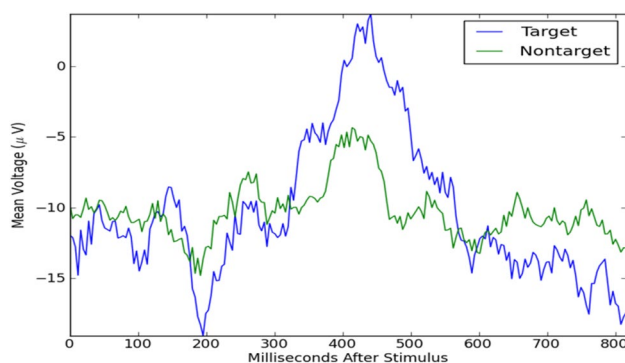


Fig. 1 P300 signal (Cecotti and Graser 2011)

Reviews of BCI systems in general have been given in Nicolas Alonso and Gomez-Gil (2012), Wolpaw et al. (2002) among others. Waldert et al. concentrated on extracting directional information only (Waldert et al. 2009). McFarland et al. provide an overview over the feature extraction and translation methods for classification-based systems (McFarland et al. 2006). Teplan concentrates on measurement in EEG-based systems (Teplan 2002) and Lotte et al. on classification (Lotte et al. 2007). After that, Nicolas Alonso and Gomez-Gil (2012) handled many kinds of BCI applications range over adapted web browsers, word processors, brain control of a wheelchair or neuro-prostheses, games, and more. Furthermore, Moore (2003) identified and classified the following four main challenges of BCI systems in real world use:

1. The information transfer rate of such systems is far too low.
2. The error rate during input is very high due to highly variable brain signals.
3. Autonomy is not really given since sensors usually need to be applied by someone else.
4. The cognitive load, under distractions of the brain generates different and noisier signals. BCI systems need to work even under such a distracting environment.

Ferrez and Millan (2008) used error-related potentials within the EEG recording shortly after an action to apply reinforcement learning. That was an attempt towards a more adaptive BCI. Vidaurre et al. also strove to solve the so-called “BCI Illiteracy” problem, that there is a significant portion of people (about 15–20%), who are unable to control a BCI system. Therefore, they also tried to make BCIs more adaptive such that they do not solely rely on off-line calibration when using supervised learning (Vidaurre et al. 2011).

Regarding filtering methods, Blankertz et al. relied on common spatial patterns (CSP) as proposed by Ramoser et al. (2000). They extended the common spatial patterns approach to consider typical variations of a subject’s EEG signals between trials (Blankertz et al. 2007).

Carlson and Millán also used BCI to control a wheelchair with “left” and “right” commands and additionally equipped the wheelchair with obstacle avoidance to ensure safe use (Carlson and Millán 2013). They employed motor imagery as input using Laplacian filtering and power spectral density. Also, there is a thesis from Rebsamen (2009) that covers a wheelchair control through BCI using P300-based destination selection. Mandel et al. created a wheelchair control depending on EEG. They stressed that due to slight inaccuracy of the control method it is important to have the wheelchair equipped with a safety mechanism that avoids collisions. Also, they mentioned that they had one subject

who was not able to use their control, so he fell under the so-called BCI illiteracy (Mandel et al. 2009).

However, in the current era, the achievements are very impressive (Tsui et al. 2011; Geng et al. 2007), but there is still much research to be conducted and many studies to be performed in the whole world of brain computer interface (Fouad and Lalib 2017).

## 2 Materials

The data that support the findings of this study are openly available in BCI competition III challenge 2004 was provided by the Wadsworth BCI dataset (P300 evoked potential) at (<https://bbci.de/competition/iii/>) (Wolpaw et al. 2004). The data were acquired using BCI2000s P3 speller paradigm described by Donchin et al. (2000) and originally by Farwell and Donchin (1988).

In this work, Matlab version 9.2.0.556344 (R2017a) and its Signal Processing Toolbox which supports an extensive range of signal processing operations is used for data analysis and technical computing, as it is a high-performance and powerful language. This work is implemented using a personal computer with a processor: Intel (R) Core (TM) i7-2.6 GHz.

### 2.1 Paradigm

The user was presented with a 6 by 6 matrix of characters as shown in Fig. 2. The user's task was to focus attention on characters in a word that was prescribed by the investigator (i.e., one character at a time). All rows and columns of this matrix were successively and randomly intensified. Two out of 12 intensifications of rows or columns contained the desired character. The responses evoked by these infrequent stimuli (i.e., the 2 out of 12 stimuli that did contain the desired character) are different from those evoked by the stimuli that did not contain the desired character and they are similar to the P300 responses. This P300-based paradigm

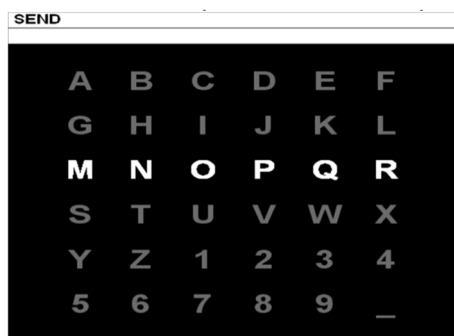


Fig. 2 P300 Speller Paradigm (Donchin et al. 2000)

can be considered as a "Virtual Keyboard on a BCI system's computer screen".

## 2.2 Data collection

The collected signals have the following specifications:

- The signals are bandpass filtered in the 0.1–60 Hz range and digitized at 240 Hz from two subjects: subject A and subject B in five sessions each.
- The training set for both subjects consists of 85 characters.
- The test set for both subjects consists of 100 characters.
- For each character, sets of 12 intensifications as shown in Fig. 3 (Donchin et al. 2000) were repeated 15 times and thus there were  $12 \times 15$  (180) total intensifications.
- The EEG signals have been acquired using 64-channels.
- A more detailed description of the dataset can be found in the BCI competition online web site (Donchin et al. 2000).

## 3 Methods

### 3.1 Data preprocessing

Preprocessing was used to remove the noise and enhance the EEG signals. It was known that the acquired signals were bandpass filtered in the range [0.1, 60] Hz and digitized at 240 Hz.

#### 3.1.1 Trials and filtration

As noticed, before, there are 12 post-intensification segments which are repeated 15 times ( $12 \times 15$ ) as one signal for each character over 64 channels. The target is to extract each post-intensification segment from the provided signals in both training and test sets. It was known that P300 ERP

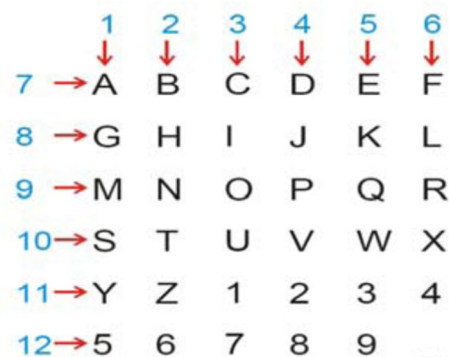


Fig. 3 Matrix rows and columns (Donchin et al. 2000)

appears after 300–500 ms from the onset of the stimulus, so the data samples are extracted 650 ms from the beginning of each intensification. This is long enough to obtain the required time features.

Filtering is an important step in noise reduction. Extra filtering using an eighth-order bandpass filter was applied to the training ( $180 \times 85$  characters) and test ( $180 \times 100$ ) post-intensification segments at different cutoff frequencies, which were selected, because the cognitive activity very rarely occurs outside of the range 300–500 ms (Allison 2003).

### 3.1.2 Decimation

Decimation reduces the original sampling rate of a sequence to a lower rate. It lowpass filters the input to guard against aliasing and down-samples the result (Matlab 2017). The filtered segments are decimated according to the high cut-off frequencies. As an example,

- The sampling frequency is 240 Hz.
- The selected window is 650 ms.
- Then, the post-intensification segments' length is  $240 \times 0.65 = 156$  samples.
- If the high cut-off frequency is 30 Hz, then, the decimated factor will be  $240/30 = 8$ .
- Therefore, the post-intensification segments' length is decimated to  $156/8 \approx 20$  samples.
- Then, the filtered training response is  $((180 \times 85) \times 20 \times 64)$ , and the Filtered Test Response is  $((180 \times 100) \times 20 \times 64)$ .

### 3.2 Feature vector

The Feature vector is constructed from the concatenation of the measurements of 64 channels of all post-intensification segments. As was mentioned before, the training set of both subjects 'A' and 'B' consists of 85 characters; total of  $180 \times 85$  (15,300) post-intensification segments, while the test set of both subjects consists of 100 characters;  $180 \times 100$  (18,000) post-intensification segments.

### 3.3 Normalization

"Z-Score Normalization" is commonly used in machine learning where the data include multiple dimensions can be handled. Feature standardization makes the values of each feature in the data have zero-mean (when subtracting the mean in the numerator) and unit-variance. The general method of calculation is to determine the distribution mean and standard deviation for each feature. Next, the mean is subtracted from each feature. Then, the values

(mean is already subtracted) of each feature are divided by its standard deviation.

$$x' = \frac{x - \bar{x}}{\sigma},$$

where  $x'$  is the original feature vector,  $\bar{x}$  average ( $x$ ) is the mean of that feature vector, and  $\sigma$  is its standard deviation.

The training feature vector is normalized to zero mean and unit variance. Then, the test feature vector is normalized depending on the normalization parameters acquired from the normalization of the training feature vector.

## 3.4 Classification

### 3.4.1 Linear discriminant analysis

Linear classifiers are discriminant algorithms that use linear functions to separate between different classes. They are likely the most known algorithms for BCI applications.

The aim of LDA is to use hyper-planes to separate the data representing the different classes (Duda et al. 2001; Fukunaga 1990). For a two-class problem, the class of a feature vector depends on which side of the hyper-plane the vector is.

LDA assumes a normal distribution of the data, with equal covariance matrices for both classes. The separating hyper-plane is obtained by seeking the projection that maximizes the distance between the two class means and minimizes the interclass variance (Fukunaga 1990).

This technique has a very low computational requirement which makes it suitable for BCI systems. Moreover, this classifier is simple to use and generally provides good results.

Two procedures of applying an LDA classifier were applied.

**3.4.1.1 LDA method I (LDA I)** In the first procedure which is referred to as "LDA method I", a test label of '1' which was assigned to a target or '-1' to a non-target from each row or column were averaged over sequences as shown in Fig. 4. It is considered that the most probable row and column is the one that maximizes the score.

**3.4.1.2 LDA method II (LDA II)** In the second procedure which is referred to as "LDA method II" test scores from each row or column were averaged over sequences as shown in Fig. 5. It is considered that the most probable row and column is the one that maximizes the score.

### 3.4.2 Support vector machine

A support vector machine uses a discriminant hyperplane to distinguish classes (Burges 1998; Bennett and Campbell

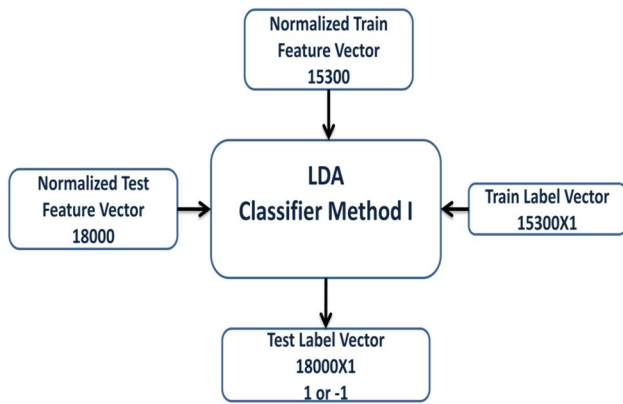


Fig. 4 LDA method I classifier

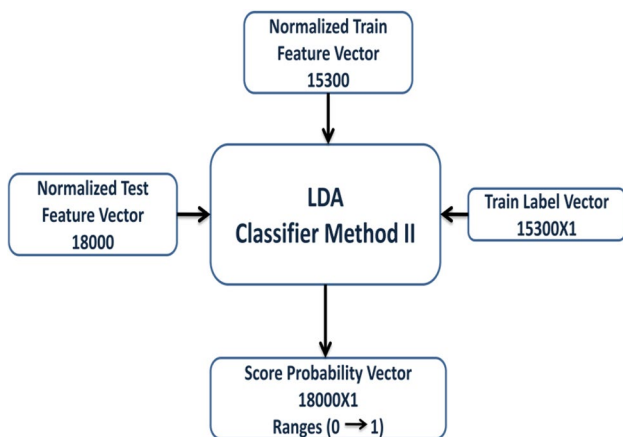


Fig. 5 LDA method II classifier

2000). However, concerning SVM, the selected hyperplane is the one that maximizes the margins, i.e., the distance from the nearest training points. Maximizing the margins is known to increase the generalization capabilities (Borges 1998; Bennett and Campbell 2000). This classifier uses a regularization parameter  $C$  that enables accommodation to outliers and allows errors on the training set.

Support vector machines enable classification using linear decision boundaries and are known as linear SVMs. These

classifiers have been applied, always with success, to a relatively large number of synchronous BCI problems (Blankertz et al. 2002; Garrett et al. 2003; Rakotomamonjy et al. 2005; Rakotomamonjy and Guigue 2008).

SVMs have several advantages. Actually, thanks to the margin maximization and the regularization term, SVM are known to have good generalization properties (Bennett and Campbell 2000; Jain et al. 2000) to be insensitive to over-training (Jain et al. 2000) and to the curse of dimensionality (Borges 1998; Bennett and Campbell 2000).

SVMs have been used in BCI research since it is a powerful procedure for pattern recognition, especially for high-dimensional problems (Rakotomamonjy and Guigue 2008). Two approaches for the application of support vector machine classifier are presented.

**3.4.2.1 SVM method I and II** In the first approach, two procedures are introduced and mentioned as "method I" and "method II". After preprocessing the post-intensification segments of the training feature vector which belong to the same row or column are averaged, which indicates that for each character there are only 12 post-intensification segments instead of 180 segments. Therefore, the training feature vector and the training label vector will be of size  $(12 \times 85)$  instead of  $(180 \times 85)$  and will be fed to the support vector machine classifier as shown in Fig. 6.

In the procedure named "method I" test labels (1 or -1) from each row or column were averaged over sequences. While in the other procedure named "method II" test scores from each row or column were averaged over sequences. For the two approaches, it is considered that the most probable row and column is the one that maximizes the score.

**3.4.2.2 SVM method III and IV** In the second approach which is a multiple-classifier combination approach, two procedures are presented and mentioned as "SVM method III" and "SVM method IV" (Fig. 7). After preprocessing, the training feature vector  $(180 \times 85)$  and the training label vector are gathered into 17 equal partitions. Each partition consists of five consecutive characters  $(180 \times 5)$  as shown in Fig. 7 and is fed to a linear support vector machine classifier.

Fig. 6 SVM classifier. a SVM method I. b SVM method II

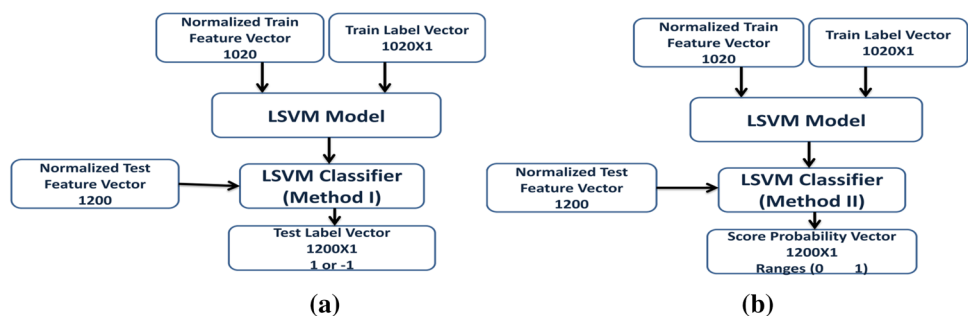
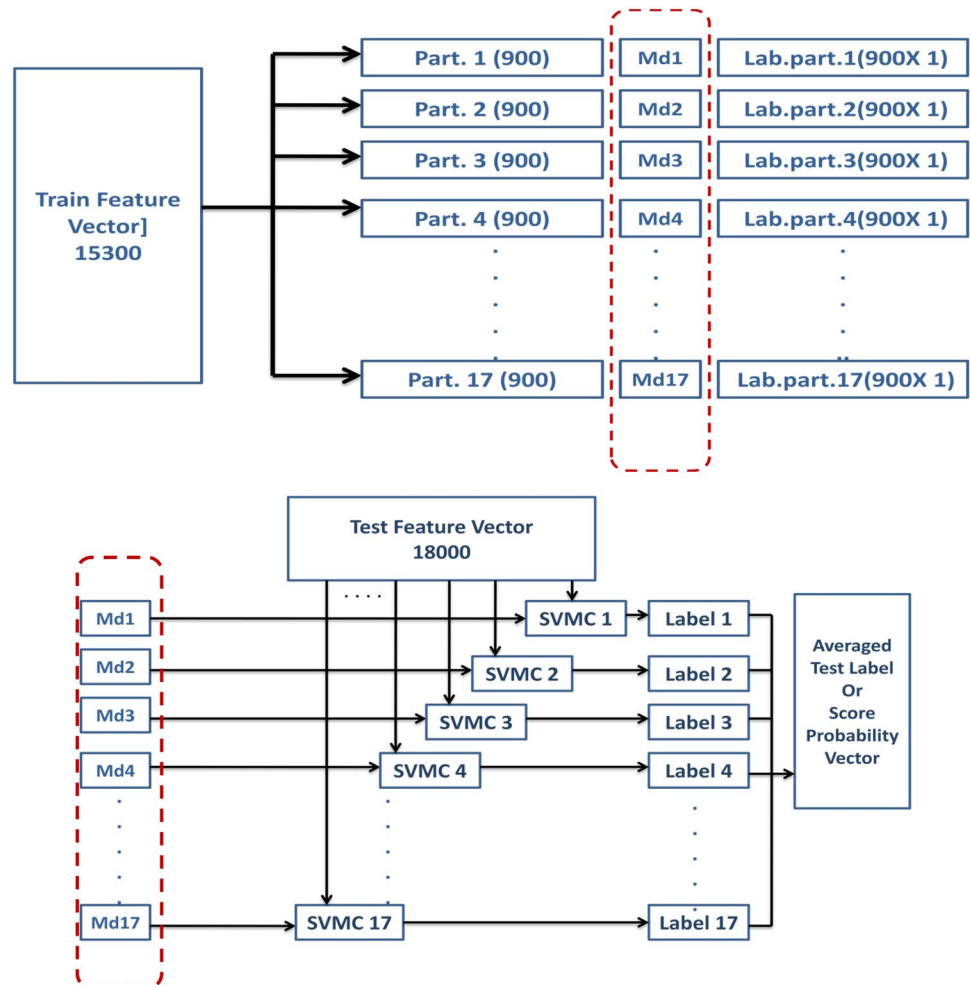


Fig. 7 SVM method III and IV



Therefore, a multiple classifier system was designed from the combination of the 17 SVM classifiers outputs.

In the procedure named "method III" test labels (1 or -1) from each row or column were averaged over sequences. While in the other procedure named "method IV" test scores from each row or column were averaged over sequences. For the two approaches, it is considered that the most probable row and column is the one that maximizes the score.

In the previous mentioned approaches, the selected value of the hyper-parameters (regularization parameters) through cross-validation of a subset of the training dataset as a validation set are shown in Table 1.

### 3.4.3 Linear regression

Linear regression algorithms (LREG) mostly differ depending on the number of independent variables and the type of relationship between the independent and dependent variables.

Linear regression is a linear model, e.g. a model that assumes a linear relationship between the input variables ( $\mathbf{x}$ )

and the single output variable ( $y$ ). More specifically,  $y$  can be calculated from a linear combination of the input variables ( $\mathbf{x}$ ) (<https://towardsdatascience.com/introduction-to-machine-learning-algorithms-linearregression-14c4e325882a>).

In the present work, a linear regression algorithm is implemented as a classifier that can be implemented in a BCI system as shown in Fig. 8.

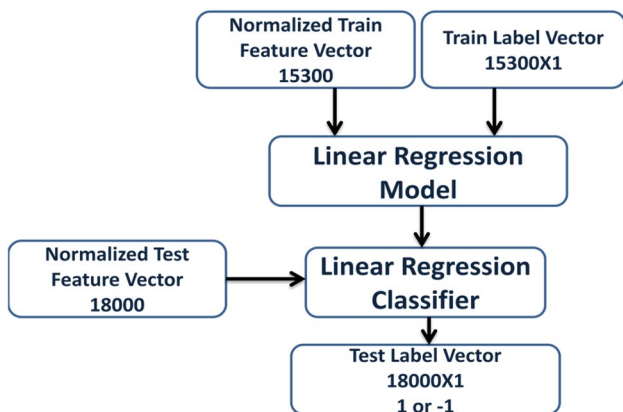
### 3.4.4 Bayesian classifier

Bayesian linear discriminant analysis (BLDA) is a simple and efficient method for classification. This technique was presented by Hoffmann et al. (2004) as follows:

- Based on the training labels, the regression targets are calculated.
- An iterative estimation of the parameters alpha and beta was calculated.
- To predict each character (belongs to a certain row or column) related to which class, the mean of the predic-

**Table 1** SVM hyper-parameters values

Standardize	Kernel scale	Kernel function	Number of sequences	Subject	SVM Method
True	27.1	Linear	15th sequences	Subject A	Method I
True	20	Linear	5th sequences		
True	16.4	Linear	15th sequences	Subject B	
True	2.1	Linear	5th sequences		
True	27.1	Linear	15th sequences	Subject A	Method II
True	20	Linear	5th sequences		
True	16.4	Linear	15th sequences	Subject B	
True	2.1	Linear	5th sequences		
True	(7) for all 17 models	Linear	15th sequences	Subject A	Method III
True	4, 4, 6, 7.2, 10, 4, 1, 7.7, 5.6, 15, 8, 4, 1, 7.4, 7.2, 16, 4	Linear	5th sequences		
True	1.1, 2, 1.9, 2, 3.1, 2.1, 1.1, 1.7, 1.6, 1, 1.6, 1.9, 2.6, 1.8, 1.1, 2, 1.7	Linear	15th sequences	Subject B	
True	1, 3.1, 1.7, 2, 7.2, 3, 2.2, 2, 1, 5.4, 2.3, 2, 1, 3.1, 1, 1, 4	Linear	5th sequences		
True	(7) for all 17 models	Linear	15th sequences	Subject A	Method IV
True	4, 4, 6, 7.2, 10, 4, 1, 7.7, 5.6, 15, 8, 4, 1, 7.4, 7.2, 16, 4	Linear	5th sequences		
True	1.1, 2, 1.9, 2, 3.1, 2.1, 1.1, 1.7, 1.6, 1, 1.6, 1.9, 2.6, 1.8, 1.1, 2, 1.7	Linear	15th sequences	Subject B	
True	4, 4, 6, 7.2, 10, 4, 1, 7.7, 5.6, 15, 8, 4, 1, 7.4, 7.2, 16, 4	Linear	5th sequences		



**Fig. 8** LREG classifiers

tive distribution was calculated for each character which was considered as the score for the row or column.

Based on our proposed Bayesian classifier, the number of iterations and the estimated parameters are shown in Table 2.

**Table 2** Number of iterations and estimated parameters of Bayesian classifier

	Subject A		Subject B	
	15th sequences	5th sequences	15th sequences	5th sequences
No. of iterations	16	35	10	15
Alpha <sub>Final Iteration</sub>	0.244547	0.386812	0.027999	0.053704
Beta <sub>Final Iteration</sub>	0.167973	0.160048	0.184965	0.180069
Log evidence	-36,058.316913	-12,188.393533	-35,198.809550	-11,953.269824

### 3.5 Character prediction

The scores predicted from the previously applied classifiers are averaged over the 15 sequences, yielding 12 scores for each character. The target character is determined by finding the row and column with the highest score.

## 4 Results and discussion

Depending on the percentage of the correctly predicted characters in the test sets, the performance is evaluated. Based on the evaluation criteria, Tables 3, and 5 show subject ‘A’ and ‘B’ performances among different frequencies which yield the best results for each classifier during 5th and 15th sequences. The time taken to predict a character in subject ‘A’ does not exceed 24.0786 s as shown in Table 4, while the time taken to predict a character in subjects ‘B’ does not exceed 55.2476938 s as shown in Table 6. The run times need to be improved to be more suitable for online applications.

**Table 3** Subject A—performances of the presented classifiers

Classifiers	Subject A		
	High cut-off frequency	Accuracy	
		15th sequences (%)	5th sequences (%)
LDA I	30 Hz	83	35
LDA II		94	49
SVM I	40 Hz	81	32
SVM II		88	45
SVM III	30 Hz	94	39
SVM IV		99	49
LREG	20 Hz	95	55
BLDA	40 Hz	98	55

Figure 9a and b shows that the highest performances obtained when applying the proposed classifiers during the 15th sequence for subjects ‘A’ and ‘B’.

Table 7 presents the performance results achieved by the test sets for subjects ‘A’ and ‘B’ during 15th and 5th sequences. LDA II method outperformed LDA I, as it achieved 95.5% and 61% for 15th and 5th sequences, respectively. SVM IV method outperformed the other support vector machine methods; SVM I, SVM II, and SVM III as it achieved 98% and 54.5% for 15th and 5th sequences, respectively.

By comparing the performance results of the mentioned classifiers, it is obvious that SVM IV and BLDA methods give the highest results for 15th sequence, as they achieved 98%. While the LREG and BLDA methods gave the highest results for 5th sequence as they achieved 66.5% and 66%, respectively. Therefore, the BLDA method yields the highest performance for both sequences and outperformed all the classifiers.

As previously discussed, to predict a target character, it is important to determine the row and column this character

belongs to. Thus, it is difficult to determine the four possible observations, where positive means target and negative means non-target:

- True positives (TP): observation is positive and is predicted to be positive.
- True negatives (TN): observation is negative and is predicted to be negative.
- False positives (FP): observation is negative but is predicted positive.
- False negatives (FN): observation is positive but is predicted negative.

Hence, it is impossible to compute more evaluation measurements: accuracy, sensitivity, specificity, and precision. Instead of predicting a specific character, its corresponding row and column were distinguished.

The actual and predicted testing data were averaged over the 15 sequences, resulting in an averaged data of 1200 observations (12 R/C × 100 character) for each.

**Table 5** Subject B—performances of the presented classifiers

Classifiers	Subject B		
	High cut-off frequency	Accuracy	
		15th sequences (%)	5th sequences (%)
LDA I	10 Hz	96	61
LDA II		97	73
SVM I		85	47
SVM II		92	58
SVM III		96	56
SVM IV		97	60
LREG		96	78
BLDA		98	77

**Table 4** Subject A—timing of the presented classifiers

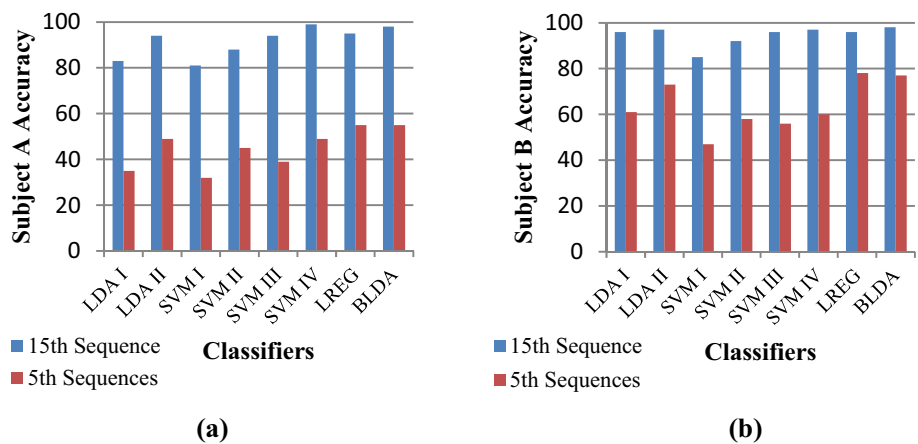
Classifiers	Subject A			
	15th Sequences		5th Sequences	
	Training Time (s)/character	Testing Time (s)/character	Training Time (s)/character	Testing Time (s)/character
LDA I	10.783565	11.670506	10.519284	10.630982
LDA II				
SVM I	10.825558	12.550250	10.692489	11.755377
SVM II				
SVM III	10.785005	13.293617	10.476025	11.239532
SVM IV				
LREG	10.78356	11.147101	10.519284	10.862071
BLDA	10.783565	11.081267		11.689230



**Table 6** Subject B-timing of the presented classifiers

Subject B				
Classifiers	15th Sequences		5th Sequences	
	Training Time (s)/character	Testing Time (s)/character	Training Time (s)/character	Testing Time (s)/character
LDA I	25.333588	28.000378	23.525394	28.588103
LDA II				
SVM I	25.333588	26.024595	26.705974	28.541719
SVM II				
SVM III	24.656749	24.319893	24.768453	25.346064
SVM IV				
LREG	25.333588	28.981693	23.525394	25.414990
BLDA	25.333588	28.121856		24.377657

**Fig. 9** Performances of the presented classifiers



**Table 7** Performances of the presented classifiers

Classifiers	15th sequences		Average (%)	5th sequences		Average (%)
	Subject A (%)	Subject B (%)		Subject A (%)	Subject B (%)	
LDA I	83	96	89.5	35	61	48
LDA II	94	97	95.5	49	73	61
SVM I	81	85	83	32	47	39.5
SVM II	88	92	90	45	58	51.5
SVM III	94	96	95	39	56	47.5
SVM IV	99	97	98	49	60	54.5
LREG	95	96	95.5	55	78	66.5
BLDA	98	98	98	55	77	66

The confusion matrices for the eight presented classifiers are shown in Tables 8 and 9 for both subjects A and B, respectively.

Tables 10 and 11 show the performance measurements for the eight classifiers applied on subjects A and B, respectively.

Receiver operating characteristic (ROC) curve is a major visualization technique for presenting the performance of a classification model. It summarizes the trade-off between the true positive rate (TPR) and false positive rate (FPR) for a

predictive model using different probability thresholds. ROC curves of the eight presented classifiers for both subjects A and B are clearly shown in Fig. 10.

It is clearly obvious from Tables 10 and 11 that all the performance measurements yield high results for all suggested classifiers. By analyzing the ROC curves of both subjects shown in Fig. 10, it is found that the areas under the ROC curves (AUC) indicate high performances (more than 0.934) of the presented classifiers. This means that the suggested classifiers are promising models.

**Table 8** Confusion matrices for all classifiers concerning subject A

	Actual value	
	Positive (target)	Negative (non-target)
<b>LDA I</b>		
Predicted value		
Positive (target)	TP = 181	FP = 19
Negative (non-target)	FN = 19	TN = 981
<b>LDA II</b>		
Predicted value		
Positive (target)	TP = 193	FP = 7
Negative (non-target)	FN = 7	TN = 993
<b>SVM I</b>		
Predicted value		
Positive (target)	TP = 178	FP = 22
Negative (non-target)	FN = 22	TN = 978
<b>SVM II</b>		
Predicted value		
Positive (target)	TP = 187	FP = 13
Negative (non-target)	FN = 13	TN = 987
<b>SVM III</b>		
Predicted value		
Positive (target)	TP = 194	FP = 6
Negative (non-target)	FN = 6	TN = 994
<b>SVM IV</b>		
Predicted value		
Positive (target)	TP = 199	FP = 1
Negative (non-target)	FN = 1	TN = 999
<b>LREG</b>		
Predicted value		
Positive (target)	TP = 195	FP = 5
Negative (non-target)	FN = 5	TN = 995
<b>BLDA</b>		
Predicted value		
Positive (target)	TP = 198	FP = 2
Negative (non-target)	FN = 2	TN = 998

**Table 9** Confusion matrices for all classifiers concerning subject B

	Actual value	
	Positive (target)	Negative (non-target)
<b>LDA I</b>		
Predicted value		
Positive (target)	TP = 195	FP = 5
Negative (non-target)	FN = 5	TN = 995
<b>LDA II</b>		
Predicted value		
Positive (target)	TP = 197	FP = 3
Negative (non-target)	FN = 3	TN = 997
<b>SVM I</b>		
Predicted value		
Positive (target)	TP = 184	FP = 16
Negative (non-target)	FN = 16	TN = 984
<b>SVM II</b>		
Predicted value		
Positive (target)	TP = 190	FP = 10
Negative (non-target)	FN = 10	TN = 990
<b>SVM III</b>		
Predicted value		
Positive (target)	TP = 196	FP = 4
Negative (non-target)	FN = 4	TN = 996
<b>SVM IV</b>		
Predicted value		
Positive (target)	TP = 197	FP = 3
Negative (non-target)	FN = 3	TN = 997
<b>LREG</b>		
Predicted value		
Positive (target)	TP = 195	FP = 5
Negative (non-target)	FN = 5	TN = 995
<b>BLDA</b>		
Predicted value		
Positive (target)	TP = 198	FP = 2
Negative (non-target)	FN = 2	TN = 998

## 5 Comparative study

Different classification techniques were investigated for the same dataset to achieve high performances. This is obvious in Table 12.

By comparing some researchers' classification performances, it is clearly shown that the suggested BLDA and SVM IV methods gives better performances after 15th sequences than other researchers' methods.

**Table 10** Performance measurements for all classifiers concerning subject A

Classifiers	Subject A			
	Accuracy (%)	Precision (%)	Sensitivity (%)	Specificity (%)
LDA I	96.834	90.5	90.5	90.5
LDA II	98.834	96.5	96.5	96.5
SVM I	96.334	89	89	89
SVM II	97.834	93.5	93.5	93.5
SVM III	99	97	97	97
SVM IV	99.834	99.5	99.5	99.5
LREG	99.167	97.5	97.5	97.5
BLDA	99.667	99	99	99

**Table 11** Performance measurements for all classifiers concerning subject B

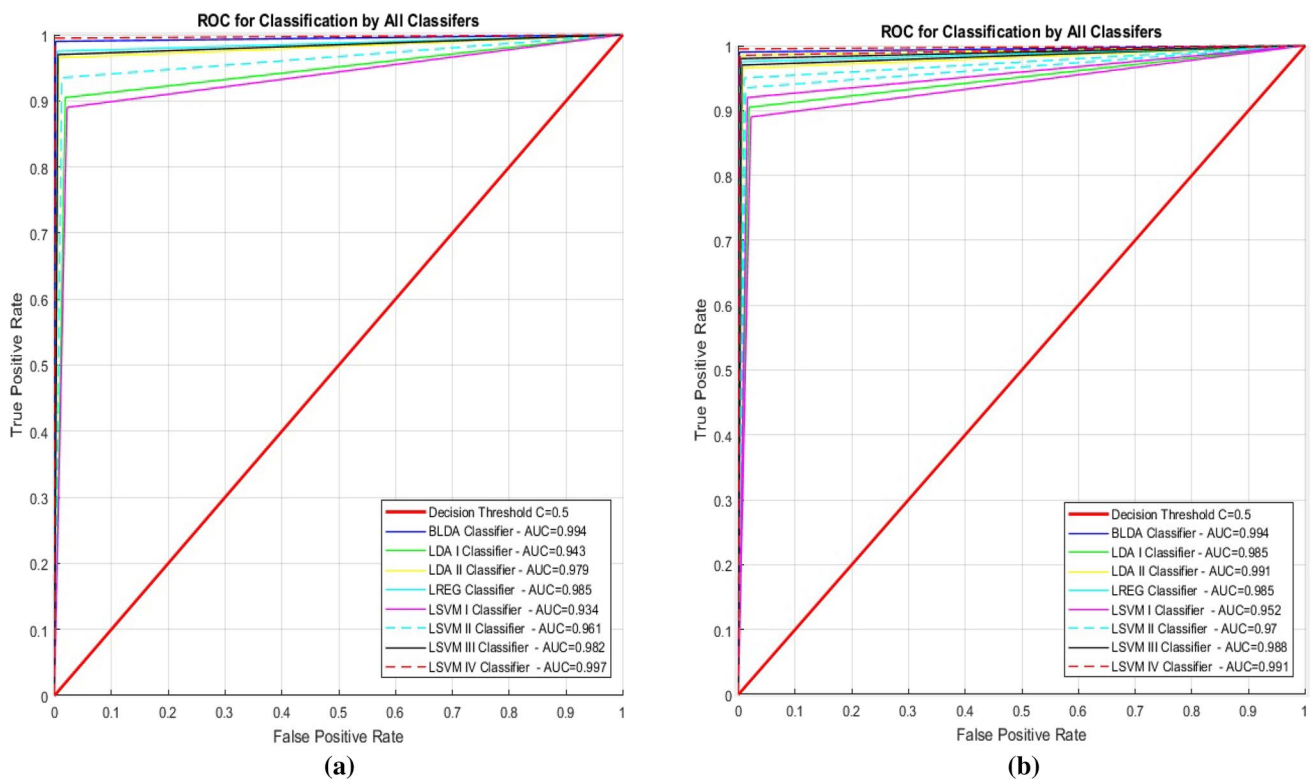
Classifiers	Subject B			
	Accuracy (%)	Precision (%)	Sensitivity (%)	Specificity (%)
LDA I	99.167	97.5	97.5	97.5
LDA II	99.5	98.5	98.5	98.5
SVM I	97.334	92	92	92
SVM II	98.334	95	95	95
SVM III	99.334	98	98	98
SVM IV	99.5	98.5	98.5	98.5
LREG	99.167	97.5	97.5	97.5
BLDA	99.667	99	99	99

### 6 Conclusion

A complete BCI system was presented. For subjects 'A' and 'B' of competition III, the training and test sets were filtered by an eighth-order bandpass filter with different cut-off frequencies (0.1–10), (0.1–20), (0.1–30), and (0.1–40). The two sets were decimated according to the filter high cut-off frequency. The feature vector of both sets was reconstructed and normalized.

Concerning the classification process, the performance obtained when applying all classifiers to the test sets during the 15th sequence is better than the 5th sequence. It was observed that the time taken to train or classifying a test character is about 2 s. Eight classification procedures were presented. By comparing the performances of the mentioned classifiers, it was observed that LDA II method performs better than LDA I. SVM IV method yields better results than the other proposed methods. Finally, BLDA achieved the highest results during the 15th sequence and the 5th sequence.

In the future, several classification algorithms could be tried to achieve higher accuracy with fewer number of sequences, taking into consideration decreasing the number of used electrodes and thus decreasing the taken time.



**Fig. 10** ROC Curves of the presented classifiers. **a** Subject A. **b** Subject B

**Table 12** Performances presented from different researchers

#	Researcher	Accuracy		Research lab	Differences
		15th sequences (%)	5th sequences (%)		
1	Alain Rakotomamonjy	96.5	73.5	PSI CNRS FRE-2645, INSA de Rouen, France	Classify with SVM III
2	Li Yandong	90.5	55	Department of Automation Department of Biomedical Engineering, Tsinghua University, China	Classify with SVM
3	Zhou Zongtan	90	59.5	Department of Automatic Control, National University of Defense Technology, China	Different Feature Extraction Methods
4	Ulrich Hoffmann	89.5	53	Signal Processing Institute, Ecole Polytechnique Federale de Lausanne (EPFL), Switzerland	Using a Gradient Boosting Classification Method

## References

- Allison B (2003) P3 or not P3: toward a better P300 BCI. Ph.D. dissertation. University of California, San Diego
- ALS Association (2012) Quick Facts about ALS & The ALS Association. [www.alsa.org/news/media/quick-facts.html](http://www.alsa.org/news/media/quick-facts.html)
- Azar AT, Balas VE, Olariu T (2014) Classification of EEG-based brain computer interfaces: advanced intelligent computational technologies and decision support systems studies in computational intelligence, vol 97. Springer, Berlin, p 106
- Bennett KP, Campbell C (2000) Support vector machines: type. *Explor Newslett* 2:1
- Blankertz B, Curio G, Müller KR (2002) Classifying single trial EEG: towards brain computer interfacing. In: *Advances in neural information processing systems (NIPS 01)*, vol 14, p 157
- Blankertz B, Kawanabe M, Tomioka R, Hohlefeld F, Müller K-R, Nikulin VV (2007) Invariant common spatial patterns: alleviating nonstationarities in brain-computer interfacing. In: *Advances in neural information processing systems*, pp 113–120
- Burges CJC (1998) A tutorial on support vector machines for pattern recognition. *Knowl Discov Data Min* 2:121
- Carlson T, Millán JR (2013) Brain-controlled wheelchairs: a robotic architecture. *IEEE Robot Autom Mag* 20:65
- Cecotti H, Graser A (2011) Convolutional neural networks for P300 detection with application to brain-computer interfaces. *IEEE Trans Commun* 33:433
- Donchin E, Spencer KM, Wijensinghe R (2000) The mental prosthesis: assessing the speed of a P300-based brain computer interface. *IEEE Trans* 8:174
- Duda RO, Hart PE, Stork DG (2001) *Pattern recognition*, 2nd edn. Wiley-Interscience, New York
- Farwell LA, Donchin E (1988) Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalogr Clin Neurophysiol* 70:510
- Ferrez PW, Millan JR (2008) Error-related EEG potentials generated during simulated brain computer interaction. *IEEE Trans Biomed Eng* 55:923
- Fouad IA, Hadidi T (2014) Classifying brain-computer interface features based on statistics and density of power spectrum. *Int J Biomed Eng Technol* 18:1–13
- Fouad IA, Labib FE-ZM (2017) Attempts towards the first brain computer interface (BCI) system in INAYA medical college. *Int J Comput Appl Technol Inderscience* 55:2
- Fukunaga K (1990) *Statistical pattern recognition*, 2nd edn. Academic Press Inc, New York
- Garrett D, Peterson DA, Anderson CW, Thaut MH (2003) Comparison of linear, nonlinear, and feature selection methods for EEG signal classification. *IEEE Trans Neural Syst Rehabil Eng* 11:141
- Geng T, Dyson M, Tsui CS, Gan JQ (2007) A 3-class asynchronous BCI for controlling mobile robots. In: *MAIA BCI workshop—BCI meets robotics: challenging issues in brain computer interaction and shared control*, Leuven
- Hohffmann U, Garcia G, Vesin J-M, Ebrahimi T (2004) Application of the evidence framework to brain computer interfaces. *IEEE Eng Med Biol Conf* 1:446
- Jain AK, Duin RPW, Mao J (2000) Statistical pattern recognition: a review. *IEEE Trans Pattern Anal Mach Intell* 22:4
- Lotte F, Congedo M, Lécuyer A, Lamarche F, Arnaldi B (2007) A review of classification algorithms for EEG-based brain computer interfaces. *J Neural Eng* 4:R1
- Lu J, Speier W, Hu X, Pouratian N (2013) The effects of stimulus timing features on P300 speller performance. *Clin Neurophysiol* 124:306
- Mandel C, Luth T, Laue T, Rofer T, Graser A, Krieg-Bruckner B (2009) Navigating a smart wheelchair with a brain computer interface interpreting steady-state visual evoked potentials. In: *Intelligent robots and systems: IEEE/RSJ international conference on IEEE*, pp 1118–1125
- Matlab (2017) Matlab version 9.2.0.556344 (R2017a) and its signal processing toolbox
- Mattout J, Perrin M, Bertrand O, Maby E (2014) Update article: improving BCI performance through co-adaptation: applications to the P300-speller. *Ann Phys Rehabil Med* 58:23–28
- McCane L, Heckman S, McFarland D, Townsend G, Mak J, Sellers E, Zeitlin D, Tenteromano L, Wolpaw J, Vaughan T (2015) P300-based brain-computer interface (BCI) event-related potentials (ERPs): people with amyotrophic lateral sclerosis (ALS) vs. age matched controls. *Clin Neurophysiol* 126:2124–2131
- McFarland DJ, Anderson CW, Muller K, Schlögl A, Krusienski DJ (2006) Bci meeting 2005 workshop on BCI signal processing: feature extraction and translation. *IEEE Trans Neural Syst Rehabil Eng* 14:135
- Moore MM (2003) Real-world applications for brain-computer interface technology. *IEEE Trans Neural Syst Rehabil Eng* 11:162
- Nicolas Alonso LF, Gomez-Gil J (2012) Brain computer interfaces, a review. *Sensors* 12:1211
- Rakotomamonjy A, Guigue V (2008) BCI competition III: dataset II—ensemble of SVMs for BCI P300 speller. *IEEE Trans. Biomed Eng* 55:1147
- Rakotomamonjy A, Guigue V, Mallet G, Alvarado V (2005) Ensemble of SVMs for improving brain computer interface P300 speller

- performances. In: International conference on artificial neural networks
- Ramoser H, Muller-Gerking J, Pfurtscheller G (2000) Optimal spatial filtering of single trial EEG during imagined hand movement. *IEEE Trans Rehabil Eng* 8:441
- Rebsamen B (2009) A brain controlled wheelchair to navigate in familiar environments. Ph.D. dissertation. National University of Singapore
- Teplan M (2002) Fundamentals of EEG measurement. *Meas Sci Rev* 2:1
- Townsend G, LaPallo B, Boulay C, Krusienski D, Frye G, Hauser C, Schwartz N, Vaughan T, Wolpaw J, Sellers E (2010) A novel P300-based brain-computer interface stimulus presentation paradigm: moving beyond rows and columns. *Clin Neurophysiol* 121:1109
- Tsui CSL, Gan JQ, Hu H (2011) A self-paced motor imagery based brain computer interface for robotic wheelchair control. *J Clin EEG Neurosci* 42:225
- Vidaurre C, Sannelli C, Müller K-R, Blankertz B (2011) Machine-learning-based co-adaptive calibration for brain computer interfaces. *Neural Comput* 23:791
- Waldert S, Pistohl T, Braun C, Ball T, Aertsen A, Mehring C (2009) A review on directional information in neural signals for brain-machine interfaces. *J Physiol Paris* 103:244
- Wolpaw JR, Birbaumer N, McFarland DJ, Pfurtscheller G, Vaughan TM (2002) Brain computer interfaces for communication and control. *Clin Neurophysiol* 113:767
- Wolpaw JR, McFarland DJ, Vaughan TM, Schalk G (2003) The wadsworth center brain computer interface (BCI) research and development program. *IEEE Trans Neural Syst Rehabil Eng* 11:1
- Wolpaw JR, Schalk G, Krusienski D (2004) Wadsworth Center, NYS Department of Health. <https://bbci.de/competition/iii/>
- Wu Z (2014) Studying modulation on simultaneously activated SSVEP neural networks by a cognitive task. *J Biol Phys* 1:55

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.