



MODIFICATIONS OF UNSUPERVISED NEURAL NETWORKS FOR SINGLE TRIAL P300 DETECTION

L. Vařeka, P. Mautner**

Abstract: P300 brain-computer interfaces (BCIs) have been gaining attention in recent years. To achieve good performance and accuracy, it is necessary to optimize both feature extraction and classification algorithms. This article aims at verifying whether supervised learning models based on self-organizing maps (SOM) or adaptive resonance theory (ART) can be useful for this task. For feature extraction, the state-of-the-art Windowed means paradigm was used. For classification, proposed classifiers were compared with state-of-the-art classifiers used in BCI research, such as Bayesian Linear Discriminant Analysis, or shrinkage LDA. Publicly available datasets from fifteen healthy subjects were used for the experiments. The results indicated that SOM-based models yield better results than ART-based models. The best performance was achieved by the LASSO model that was comparable to state-of-the-art BCI classifiers. Further possibilities for improvements are discussed.

Key words: *brain-computer interfaces, event-related potentials, P300, adaptive resonance theory, self-organizing maps, Fuzzy ARTMAP, Bayesian Linear discriminant analysis*

Received: May 10, 2015

DOI: 10.14311/NNW.2018.28.001

Revised and accepted: February 16, 2018

1. Introduction

Brain-computer interfaces (BCIs) have been gaining popularity in recent years. The main aim of BCIs is to recognize the intention of its user directly, typically by processing electroencephalographic (EEG) signal. This has potential benefits for both healthy and paralyzed people. For patients with locked-in syndrome that cannot speak or make any intentional movement, BCIs may be the only option for communication with the outside world. Many different BCI paradigms have been described in the literature [26]. One of the most established BCI systems are P300-based BCIs.

*Lukáš Vařeka – Corresponding author; Pavel Mautner; Department of Computer Science and Engineering, University of West Bohemia, E-mail: lvareka@kiv.zcu.cz, mautner@kiv.zcu.cz

The P300 is an event-related (ERP) component in EEG signal. It appears in the signal during an odd-ball experimental paradigm. This paradigm is based on a sequence of visual or auditory stimuli that are presented to the measured participant. The stimuli can be categorized into rare (target) and common (non-target). The participant is asked to silently count occurrences of target stimuli that randomly appear among a series of non-target stimuli. Typically, the P300 waveform is measurable across the parieto-central area of the head [25]. Its latency is usually between 300 ms and 450 ms, and its amplitude is much larger following target stimuli. An example of the P300 waveform is depicted in Fig. 1 [25].

Since the P300 component depends on attention and stimulus category, it can be used for BCI design. In 1988, Farwell and Donchin [9] introduced the P300 speller that displays letters arranged in rows and columns. The user chooses one letter he or she wants to type. Subsequently, rows and columns randomly flash. The P300 component should appear in the signal following the row and column associated with that letter [9].

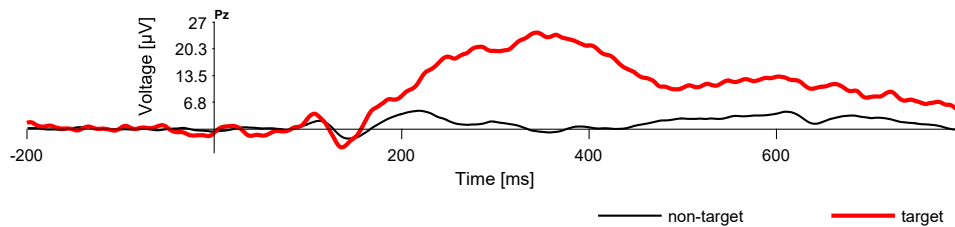


Fig. 1 Comparison of averaged EEG responses to common (*non-target*) stimuli and rare (*target*) stimuli. There is a clear P300 component following the target stimuli.

Since P300-based BCIs rely on the P300 detection, it is worth optimizing its performance. With a reliable P300 classifier, different BCIs based on this principle can be implemented easily. Unfortunately, the P300 waveform has a much lower amplitude in comparison with background EEG signal. Consequently, accurate detection of the P300 component remains challenging.

1.1 State of the Art

P300 BCI algorithms aim at the detection of the P300 component in ERP data. To improve signal-to-noise ratio (SNR) of the P300 data, pre-processing is often used as the first step. Because ERPs occur mostly in certain frequency bands (typically between 0.1 and 30 Hz [25]), band-pass filtering can preserve the P300 and suppress the noise. Moreover, since the P300 occurs regularly following the same types of stimuli while background EEG is randomly distributed, it is possible to amplify the P300 using averaging stimulus-locked trials [25]. However, averaging slows down information transfer rate. As the next step, features are extracted from ERP trials. Different considerations including size of the dataset, focus on time or frequency domain should influence feature extraction choice. The P300 is mostly pronounced in time domain, so discrete wavelet transform, window averaging and other related methods are commonly applied. The final step is the decision about the presence

of the P300 component based on classification. The first proposed classifier was a step-wise discriminant analysis (SWDA) followed by peak picking and covariance evaluation [9]. Linear classifiers have been commonly applied to the P300 detection including support vector machine (SVM) [32], and linear discriminant analysis (LDA) [13]. In [21], it was demonstrated that SWDA and Fisher's linear discriminant analysis provided the best performance and implementation characteristics for practical classification. Furthermore, Bayesian LDA (BLDA) was also successfully applied to the P300 detection problem and is currently widely used in this area [14]. However, relative simplicity of linear classifiers can be seen both as advantage and disadvantage. When classes in the feature space are not linearly separable (which is probably the case for P300 data), they are likely to perform worse than nonlinear classifiers. Moreover, they commonly suffer from the curse of dimensionality that limits their benefits for high-dimensional feature vectors [10].

The search for a suitable classifier has inspired many researchers to propose or even design alternative classification algorithms that may yield better performance when compared to traditional algorithms. This paper focuses on self-organizing maps and Adaptive Resonance Theory that have so far not been extensively used for P300 BCIs. These neural networks learn representation of different features that occur in the dataset. After unsupervised training, a labeling process can follow that associates those representatives with the most likely class labels. Using this procedure, these methods can be able to spot differences between target and non-target even before supervised labeling and thus provide interesting results.

There have been only a few studies regarding self-organizing maps (SOMs) [20] or Adaptive Resonance Theory (ART) [5] for the P300 detection. Self-organizing maps were successfully applied to recognition of topographic patterns of EEG spectra in [17]. Six classes were used, for continuous alpha activity, flat EEG, theta activity, eye movements, muscle activity and bad electrodes contact. The authors concluded that SOMs were able to recognize similar topographic patterns, even in EEGs not used for the training of the map. According to [24], Learning Vector Quantization (LVQ) – a supervised SOM-based classification model – was applied to EEG classification tasks. In [23], supervised LVQ1 has successfully been applied to the P300 data. This further supports the hypothesis that similar models may be beneficial for P300 BCIs. Moreover, in [19], the authors proposed SOM-based clustering of P300 feature vectors obtained by wavelet transform. However, the authors did neither implement nor test their proposed system. In [28], another SOM-based P300 detection system was proposed and evaluated. Its goal was to detect start and end times of the children P300 components that were previously estimated by experts.

In our previous work, it was proven that the ART 2 networks [36] and SOM networks [37] were able to separate some ERP components from background noise.

1.2 Aims of this paper

The objective of this paper is to propose rare or novel P300 classification techniques based on SOM or ART, and to compare them with traditional classification approaches. The paper is organized as follows: Section 2 introduces methods based on SOM and ART. Subsequently, the proposed experiment is described: in Sec-

tion 3, details about the obtained data and experimental conditions are given. Section 4 explains feature extraction and informs about parameters that were used for classification. Results are given in Section 5 and discussed in Section 6.

2. Methods

This chapter provides description of the SOM and ART-based methods used – both already existing in the literature (the LASSO model, both original and simplified fuzzy ARTMAP) and authors’ SOM modifications that will be referred to as SOM1 and SOM2 methods.

2.1 SOM1

The SOM1 method was a simple procedure we have designed to turn an unsupervised SOM into a supervised classifier. It was based on assigning the most likely class label for each SOM neuron (codebook vector). The motivation to test this method was an assumption that once the SOM training using self-organization is finished, each SOM unit represents a distinct ERP pattern that may correspond to both target and non-target labels. Such distinct patterns could maintain some stability and thus provide reasonable estimation for testing data as well.

As the first step of the SOM1 method, the SOM network was trained using self-organization in the standard way. The second process was performed to label the SOM cluster units with the most likely outputs. A memory unit containing two variables (t for number of targets, n for number of non-targets) was associated with each SOM unit. One iteration of the labeling process was performed as follows (\mathbf{x} was the input vector, ω_{ij} was the weight between i -th input and j -th neuron):

1. In the same way like it would be performed for traditional SOM learning: for each SOM unit j , compute distance $D(j)$ from the input vector \mathbf{x} :

$$D(j) = \sum_i (\omega_{ij} - x_i)^2. \quad (1)$$

2. Find a cluster unit j such that $D(j)$ is a minimum, and denote it J . (J is the closest cluster unit.)
3. If the expected class of feature vector \mathbf{x} is target, set $J_T \leftarrow J_T + 1$. Otherwise, set $J_N \leftarrow J_N + 1$.

Consequently, for each SOM unit, numbers of associated target and non-target features were obtained.

Finally, to compensate for possible differences in numbers of targets and non-targets, for each SOM unit J in the map, the following computation was performed: $J_T \leftarrow J_T / \text{number_of_targets}$ and $J_N \leftarrow J_N / \text{number_of_nontargets}$.

In the testing phase, the algorithm was as follows:

1. The closest cluster unit J was found for an input \mathbf{x} .

2. The class label was assigned to the input \mathbf{x} as a result of comparison between the number of targets and non-targets previously associated with that cluster unit during the training process. If J_T was higher than J_N , the input \mathbf{x} was classified as the target. Otherwise, it was classified as the non-target.

2.2 SOM2

The previous method was modified to the SOM2 method by adding k-means clustering. The SOM network was trained using self-organization in the same way as for the SOM1 method. Subsequently, k-means clustering was performed to create a selected number of clusters from the SOM unit vectors. Instead of assigning statistics to each SOM cluster unit, statistics were merged for each cluster. In other words, for each labeling iteration:

1. The winner SOM unit J was found.
2. The cluster C was assigned if $J \in C$.
3. Based on the training feature vector label, C_T or C_N was incremented.

Finally, any possible differences between numbers of targets and non-targets were compensated as for the SOM1 method.

This method allowed each SOM cluster to be associated with one prevailing classification class label. In the testing phase, the cluster containing the winning neuron made the decision about the classification class of that input pattern.

2.3 LASSO model

The LASSO model (Learning ASsociations by Self-Organization) [27] was designed for supervised learning based on unsupervised SOM model. The main idea behind this model is that output patterns can be presented to the SOM map for its organization simultaneously with input patterns. The principle of the LASSO model is illustrated in Fig. 2. It consists of three layers: an input layer (n_I dimensional), a Kohonen (SOM) layer (represented by traditional SOM units), and an output layer (n_O dimensional). Each SOM cluster unit is connected to all the units of the two other layers. The weights of connections from input units to SOM units are called ω^{IK} . Unlike the traditional SOM learning model, connections from SOM units to output units are bidirectional: the weights from the SOM layer to output are called ω^{KO} while ω^{OK} is used for the weights from the output to the SOM layer.

In the learning phase, the weights ω^{IK} and ω^{OK} are learned simultaneously using self-organization. Weights are gradually adjusted using input and desired output presented together at each learning step. At the end of the learning phase, the weights ω^{OK} are used for computing associated outputs: the weights ω^{KO} are set to ω^{OK} .

After the learning phase, input and output layer are separated again. The SOM map is then used to associate missing data (output pattern) with partial information (input pattern).

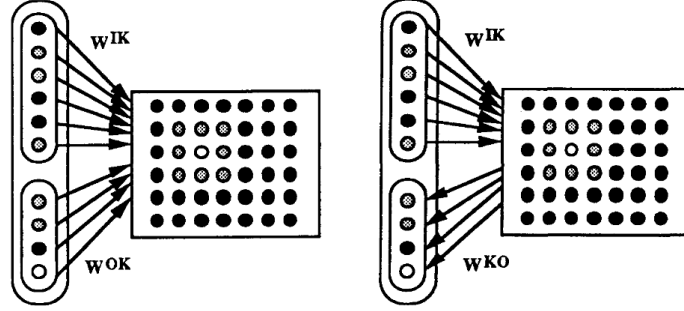


Fig. 2 The principle of the LASSO model. On the left, the training phase is illustrated. Both original feature vector ω^{IK} and known class label vector ω^{OK} are connected into one big feature vector that is used for the SOM training. In the testing phase (on the right), only feature vector is used and the expected class label vector ω^{KO} is filled from the winning neuron [27].

An output pattern is associated with one input pattern in the following way:

1. An input pattern \mathbf{x} is presented onto the input layer (n_I dimensional)
2. Minimizing the Euclidian distance in the n_I dimensional input space, the n_O other dimensions are ignored during the winning SOM cluster unit J assignment. The achieved Euclidian distance in the n_I dimensional input space is called d_I .
3. Selection of the activation group referring to all SOM-units that contribute to setting up the system output. Unit j is selected if $d_D(j, J) < s_D$ and $\frac{d_1^2(\omega_j^{\text{IK}}, \mathbf{x}) - d_1^2(\omega_J^{\text{IK}}, \mathbf{x})}{d_1^2(\omega_j^{\text{IK}}, \mathbf{x})} < p_I$ where d_D is Euclidian distance in the SOM unit space (typically 2-dimensional for the grid layout of SOM units), s_D is the threshold and p_I is the relative distance.

Activations of non-selected units are set to 0 while activations of selected units are determined in proportion to their representativeness of the input pattern:

$$Y_j = \frac{d_1^2(\omega_j^{\text{IK}}, \mathbf{x})}{d_1^2(\omega_J^{\text{IK}}, \mathbf{x})}. \quad (2)$$

4. The output pattern is calculated as a sum of output weight vectors in the activation group (AG), weighted by the activation level given to the SOM units:

$$\mathbf{y} = \frac{1}{\sum_{j \in \text{AG}} Y_j} \sum_{j \in \text{AG}} Y_j \omega_j^{\text{KO}}. \quad (3)$$

2.4 Fuzzy ARTMAP

The Fuzzy ARTMAP model [4] is a model that was proposed by the authors of the ART model. It is a generalization of an ARTMAP system [6] that was proposed for supervised learning of binary feature vectors. In contrast, the Fuzzy ARTMAP system is more general as it learns to classify real-valued inputs, more specifically patterns of fuzzy membership values between 0 and 1 indicating the extent to which each feature is present. Each fuzzy ARTMAP system contains a pair of fuzzy adaptive resonance theory modules (ART_a and ART_b) that create stable recognition categories in response to sequences of input patterns [4]. The first ART module receives input feature vectors \mathbf{a} while the second ART module typically receives the associated class labels \mathbf{b} . Both modules are connected by an associative learning network and an internal controller that manages related system operation in real time.

In comparison with SOM-based models, the advantage of the Fuzzy ARTMAP is that the neurons representing distinct patterns in the inputs are dynamically created during run time and their number can be controlled using the vigilance parameter.

2.5 Simplified Fuzzy ARTMAP

The Simplified Fuzzy ARTMAP (SFAM) [33] deals with a significant drawback of the original fuzzy ARTMAP model – its authors introduced complicated architectures for their networks instead of presenting them as simple algorithms. Moreover, the original model could generally associate any pairs of feature vectors which creates too much complexity for classification tasks if the output vectors are simple class labels. As a solution, redundancies are removed. Thus this model more resembles multi-layer perceptron. [33]

The main idea of the SFAM model is as follows [33]:

1. Find the nearest subclass prototype J that resonates with the input pattern \mathbf{x} (winner). To resonate, the following condition had to be fulfilled: $\frac{|\mathbf{x} \wedge \omega_J|}{|\mathbf{x}|} \geq \rho$ (see Fig. 3 and its description for context).
2. If the labels of that subclass prototype and the input pattern match, move the prototype closer to the input pattern.
3. If not, reset the winner, temporarily increase the resonance threshold (ρ) and try the next winner.
4. If the winner is uncommitted (unlabeled and with its weights set to 1), assign \mathbf{x} to be its prototype and label it using the known input class label.

The algorithm is illustrated in Fig. 3.

3. Conducted experiments

To compare different neural network-based classifiers, our publicly available P300 data were used. These datasets are described in more detail in [34]. Supporting

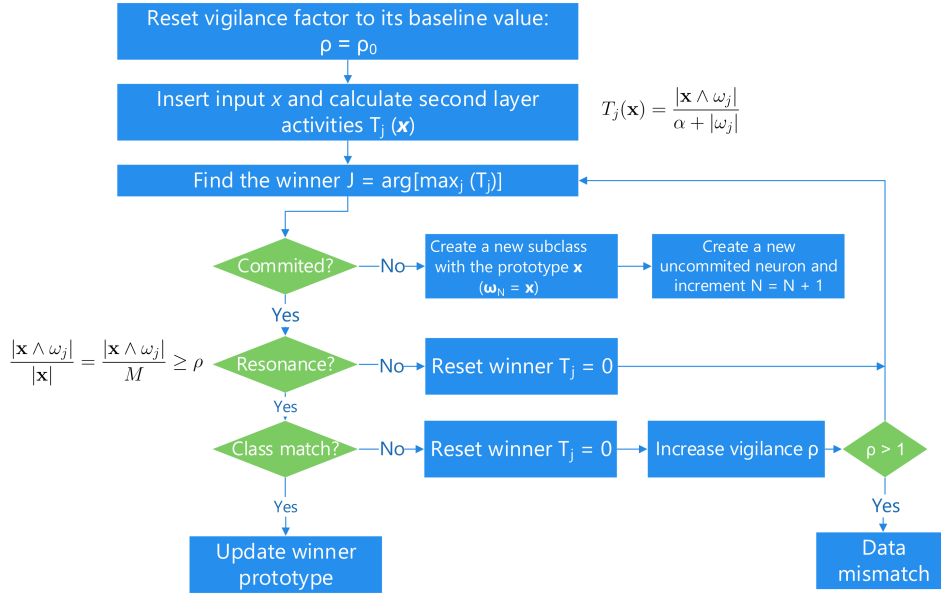


Fig. 3 Simplified Fuzzy ARTMAP training algorithm for one feature pattern. T_j , calculated as $T_j(\mathbf{x}) = \frac{|\mathbf{x} \wedge \omega_j|}{\alpha + |\omega_j|}$ is activity of the j -th neuron in the second layer, \mathbf{x} is the input vector, ω_j is the weight vector of the j -th neuron in the second layer, \wedge is the fuzzy AND operator, α represents a choice parameter, the norm $|\cdot|$ is L1 norm [33].

datasets and related metadata for this paper can be downloaded from the GigaScience database, GigaDB [35]. 15 healthy university students, aged between 20 and 26, participated in the experiments.

3.1 Recording of the data

The EEG/ERP laboratory at the University of West Bohemia, Czech Republic was used to perform the experiments. The data were recorded in an electrically and acoustically shielded cabin with a BrainVision amplifier. Standard EEG caps with 19 electrodes were placed according to a 10-20 system. The BrainVision Recorder 1.2 was used [3] for recording the EEG/ERP data. The data were recorded with 1 kHz sampling rate and recording low-pass filter with the cut-off frequency of 250 Hz.

3.2 Stimulation protocol

Three high-power Light-Emitting Diodes (LEDs): red, green and yellow were used for stimulation. This configuration is suitable for modified odd-ball paradigm experiments (also referred to as three stimulus paradigm [8]). In addition to target and non-target stimuli, the third diode flash can be used as the distractor stimulus. The distractor can elicit the subcomponent of the P300 – P3a [31].

The following stimulation protocol was followed: each diode flashed once a second and duration of each flash was 500 ms. The probabilities of the red, green and yellow diodes flashing were 83 %, 13.5 % and 3.5 %, respectively. Therefore, the green diode was the target stimulus and the red diode the non-target stimulus. The yellow diode was the distractor stimulus, and was ignored in the subsequent processing. The subjects were sitting 1 m from the stimulation device for 20 minutes. They were instructed to pay attention to the stimulation and not to perform any other task-relevant cognitive or behavioral activity.

4. Development of the P300 detection algorithms

4.1 Preprocessing and Feature Extraction

For feature extraction, the Windowed means paradigm [2] was used. The method is based on averaging epoch time windows that contain the components of interest (e.g. the P300 component). The following steps were taken:

1. Each dataset was split into epochs (trials) using stimuli markers of target events – the green diodes flashing (S 2) and non-target events – the red diodes flashing (S 4). Each trial started 500 ms before the stimulus to perform baseline correction (i.e. subtracting the average of this part of the signal from the whole trial), and ended 1000 ms after the stimulus.
2. The following time windows relative to the stimuli onsets were averaged for each epoch:
 - 200 ms – 250 ms
 - 250 ms – 300 ms
 - 300 ms – 350 ms
 - 350 ms – 375 ms
 - 375 ms – 400 ms
 - 400 ms – 425 ms
 - 425 ms – 450 ms
 - 450 ms – 500 ms
 - 500 ms – 550 ms

The selection of exact time intervals was based on [2] and further adjusted after empirical tuning. The amplitude of the P300 component is largest over central and parietal brain areas [31]. Consequently, these nine time windows averages were extracted from three EEG channels (Fz, Cz, and Pz). The process of selecting features from a single EEG channel is depicted in Fig. 4.

3. Averages from all selected channels were concatenated. As a result, each feature vector had dimensionality of 27.

The process of optimizing P300 feature extraction process using the Windowed means paradigm is described in detail in [38].

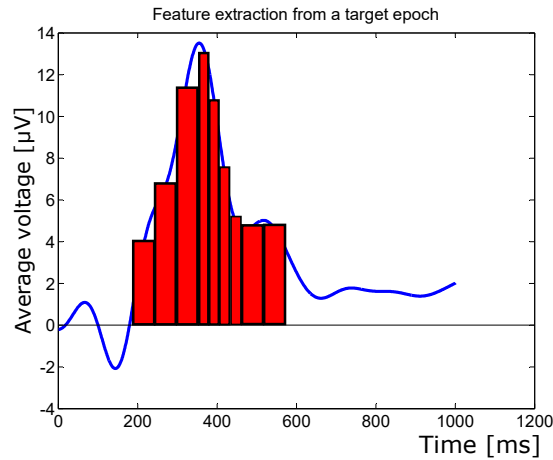


Fig. 4 Feature extraction from the averaged Pz channel. The related part of the feature vector is depicted in bars.

4.2 Classification

The classification models described above were compared based on supervised classification. The SOM toolbox [19] was used for the implementation of all SOM-based methods. In summary, the following methods described in Section 2 were compared:

- SOM1 method.
- SOM2 method.
- LASSO model [27].
- Fuzzy ARTMAP model [4]. Matlab library [11] was used for implementation. [4,6]
- SFAM model [33]. Matlab implementation [1] was used.
- Traditionally used algorithms for comparison. To evaluate possible benefits of using unsupervised neural networks, two traditionally used classifiers for the P300 detection were also implemented: Linear Discriminant Analysis (LDA) with shrinkage regularization [2] and BLDA. The BCILAB [7] implementation of LDA was used. For BLDA, implementation from [14] was used.

5. Results

The training set contained 732 feature vectors that were concatenated from the data from four subjects (with experimental IDs 88, 90, 100, and 104, respectively). Randomly, 25% of the training set was split for validation. The validation set was then used to find optimal settings of parameters for each method.

In the testing phase, the data from each experiment were evaluated separately.

The features vectors were extracted as described in Section 4.1. For each classification model, training and testing phase were repeated twice and the classification results were averaged.

The parameters of different classification models were empirically optimized to find the parameters that yielded the best classification results on validation set. For all SOM-based models, training used batch algorithm with automatically optimized learning parameters. The following parameters (in most cases selected empirically) were used:

- For the SOM1 method, the size of the SOM network was set to 9×9 .
- For the SOM2 method, the size of the SOM network was set to 12×12 . The number of clusters was set to 4.
- For the LASSO model, the size of the SOM network was set to 20×20 . s_D was set to 8 and p_I was set to 0.4.
- The Fuzzy ARTMAP model was trained in 200 epochs. The maximum number of categories was set to 30, alpha was set to 0.001 as proposed in [33], epsilon to 0.001 [33], beta to 0.5 [33] and vigilance to 0.1.
- The SFAM model was trained in 100 epochs. The maximum number of categories was also set to 35, alpha was set to 1, epsilon to 0.001 [33], beta to 0.5 [33], and vigilance to 0.1.

For evaluation, accuracy, precision and recall were computed. Let us denote t_p – number of targets classified as targets, t_n – number of non-targets classified as non-targets, f_p – number of non-targets classified as targets, and f_n – number of targets classified as non-targets. The following metrics were calculated:

$$\text{ACCURACY} = \frac{t_p + t_n}{t_p + t_n + f_p + f_n}, \quad (4)$$

$$\text{PRECISION} = \frac{t_p}{t_p + f_p}, \quad (5)$$

$$\text{RECALL} = \frac{t_p}{t_p + f_n}. \quad (6)$$

First, the results were evaluated for each testing dataset and each classification model. Fig. 5, 6 and Fig. 7 depict accuracies, precisions, and recalls, respectively. Tab. I depicts accuracy averages for each training model across all subjects.

Furthermore, for each training model, training and testing times were evaluated (Intel Core i7, 64 GB RAM, SSD hard drive). The results are shown in Tab. II.

6. Discussion

The aim of the paper was to investigate unsupervised neural networks in the P300 component detection. The results seem to indicate that SOM-based models perform better than ART-based models. LASSO models yields the best accuracy (approximately 64.09 % on average). This algorithm and the SOM1 algorithm were the

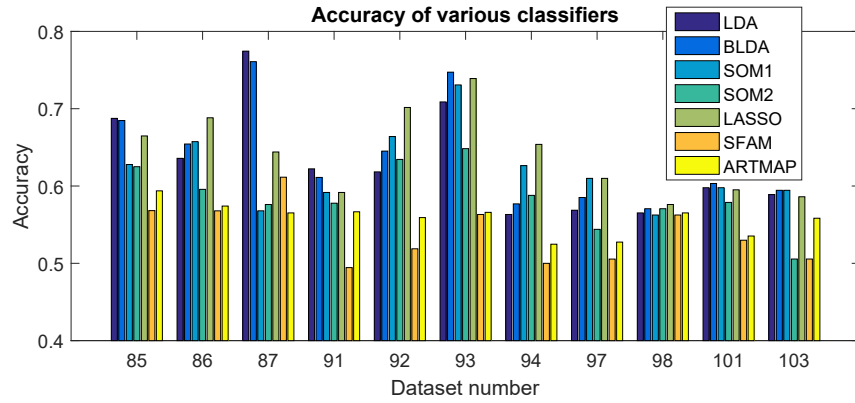


Fig. 5 Accuracy of different classifiers for all datasets.

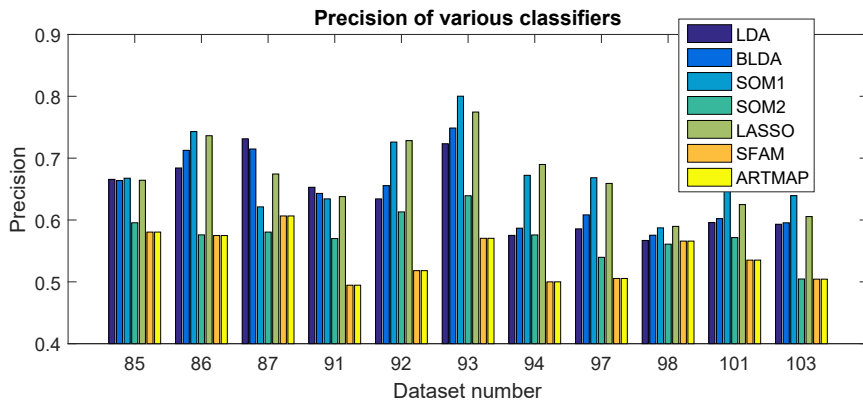


Fig. 6 Precision of different classifiers for all datasets.

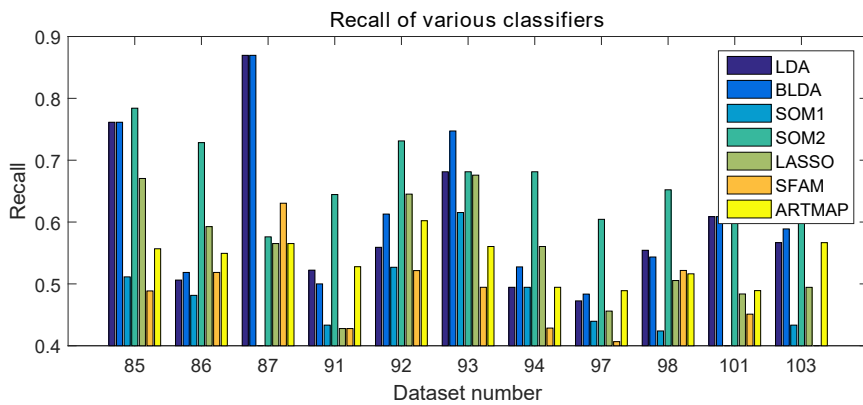


Fig. 7 Recall of different classifiers for all datasets.

Classifier	Average accuracy
LDA	63.01
BLDA	63.94
SOM1	62.10
SOM2	58.58
LASSO	64.09
SFAM	53.88
ARTMAP	55.78

Tab. I Average accuracies for different classifiers.

Classifier	Training calculation time (ms)	Testing calculation time (ms)
LDA	40	18
BLDA	4	1
SOM1	61	1
SOM2	588	1
LASSO	48	250
SFAM	6732	38
ARTMAP	412	19

Tab. II Average calculation speed for different classifiers.

only ones that matched the state-of-the-art BLDA. The accuracies in general may appear fairly low but it should be taken into account that the P300 component was detected in single trials while typically, BCI systems average many subsequent trials to limit the background noise.

Despite the importance of classification accuracy, only BCIs capable of running on-line with sufficient information transfer rate are comfortable to use for disabled users. Therefore, all used methods were compared based on the average time needed to evaluate all feature vectors from one subject in the testing phase (average time was measured from the acceptance of all feature vectors from one subject until the decision about their classification class). Fortunately, once the BCI system is trained, classifying a single feature vector is usually not very time consuming. According to our experience, to be comfortable to use, inter-stimulus intervals should be at least 200 ms. Consequently, any system that evaluates a single ERP trial faster may be used. As Tab. II clearly shows, all algorithms can be used in an on-line BCI system.

Training times are usually less crucial for the usability of a BCI systems. When the BCI system is trained on various subjects to be universal, training has to be performed only once. If a more personalized system is requested, the training process can be repeated for each new user. Tab. II depicts that all training algorithms except for SFAM were trained fast enough. The SFAM model did not yield high accuracy anyway.

There is an ongoing discussion whether linear or non-linear classification methods are preferable for BCIs. In [12], it is speculated that high-dimensional and

noisy nature of EEG may limit the advantage of non-linear classification methods over linear ones. However, tests revealed that non-linear classifiers perform slightly better. In [29], the authors claim that in some applications, non-linear classifiers can provide better results, particularly with complex and/or other very large data sets. This paper reveals that both approaches yield comparable results, although only BLDA and not LDA was able to match the accuracy of the LASSO classifier.

For the future work, the benefits of using pre-selected training data could be verified. Furthermore, the algorithms described will be applied to implement a real on-line BCI for both healthy and paralyzed subjects. Specifically, the LASSO algorithm yielding the best results will be further explored. Perhaps its modifications are possible that will boost its classification accuracy. Furthermore, since the aim of this paper was to compare classification algorithms, feature extraction can also be improved in the future. This includes using different intervals or different EEG channels.

Although classification algorithms are important for the performance of P300-based BCIs, there are also other ways to improve these systems. For example, in [16], mismatch negativity paradigm was used instead of classic oddball paradigm. The authors claim that this modification increases amplitudes of the ERP components (such as P300 and N200) and boost the accuracy and bit-rate of P300 BCIs. Furthermore, in [18] and in [15], face flashing paradigm based on flashing characters with superimposed pictures of well-known faces significantly outperformed the commonly used character flashing paradigm. Alternatively, in [22], multimodal BCIs were proposed that significantly outperform classic P300 BCIs. Moreover, as reported in [30], the reliability of P300-based BCIs depends on the environment during the experiment: the noisy everyday environment slightly lowers the accuracy of P300 BCI. Health state of the participant affected the accuracy even more. Participants with neuromuscular disorders who need BCI the most, achieved the lowest accuracy. This problem, however, can be partially solved by tuning the BCI parameters, such as using longer inter-stimulus intervals [30]. In the future work, the state-of-the-art approaches for BCIs may be integrated to further improve bit-rate and accuracy, especially when applied to paralyzed subjects.

Acknowledgement

This publication was supported by the UWB grant SGS-2016-018 Data and Software Engineering for Advanced Applications and by the project LO1506 of the Czech Ministry of Education, Youth and Sports under the program NPU I.

References

- [1] AKBAS E. Simplified Fuzzy ARTMAP neural network [software]. 2007-08-10 [accessed 2018-02-15]. Available from: <http://www.mathworks.com/matlabcentral/fileexchange/11721-simplified-fuzzy-artmap-neural-network>
- [2] BLANKERTZ B., LEMM S., TREDER M.S., HAUFE S., MÜLLER K.-R. Single-trial analysis and classification of ERP components – a tutorial. *NeuroImage*. 2011, 56(2), pp. 814–825, doi: [10.1016/j.neuroimage.2010.06.048](https://doi.org/10.1016/j.neuroimage.2010.06.048).
- [3] BRAINPRODUCTS BrainVision Recorder [software]., 2012-07-01 [accessed 2018-02-21]. Available from: <http://www.brainproducts.com/downloads.php?kid=2>

- [4] CARPENTER G., GROSSBERG S., MARKUZON N., REYNOLDS J., ROSEN D. Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps. *Neural Networks, IEEE Transactions on* 3. 1992, 5, pp. 698–713, doi: [10.1109/72.159059](https://doi.org/10.1109/72.159059).
- [5] CARPENTER G.A., GROSSBERG S. *The handbook of brain theory and neural networks*. Cambridge, MA: MIT Press, 1998.
- [6] CARPENTER G.A., GROSSBERG S., REYNOLDS J.H. ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Networks*. 1991, 4, pp. 565–588, doi: [10.1016/0893-6080\(91\)90012-T](https://doi.org/10.1016/0893-6080(91)90012-T).
- [7] DELORME A., MULLEN T., KOTHE C., ACAR Z.A., BIGDELY-SHAMLO N., VANKOV A., MAKEIG S. EEGLAB, SIFT, NFT, BCILAB, and ERICA: New tools for advanced EEG processing. *Intell. Neuroscience*. 2011, pp. 10:10–10:10, doi: [10.1155/2011/130714](https://doi.org/10.1155/2011/130714).
- [8] DUDÁČEK K., MAUTNER P., MOUČEK R., NOVOTNÝ J. Odd-ball protocol stimulator for neuroinformatics research. In: *Applied Electronics (AE), 2011 International Conference on*, Plzeň, Czech Republic. Piscataway: IEEE, 2011, pp. 1–4.
- [9] FARWELL L.A., DONCHIN E. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and Clinical Neurophysiology*. 1988, 70(6), pp. 510–523, doi: [10.1016/0013-4694\(88\)90149-6](https://doi.org/10.1016/0013-4694(88)90149-6).
- [10] FAZEL-REZAI R., ALLISON B.Z., GUGER C., SELLERS E.W., KLEIH S.C., KÜBLER A. P300 brain computer interface: current challenges and emerging trends. *Frontiers in neuroengineering*. 2012, 14(5), doi: [10.3389/fneng.2012.00014](https://doi.org/10.3389/fneng.2012.00014).
- [11] GARRETT A. Fuzzy ART and Fuzzy ARTMAP neural networks [software]. 2003-12-01 [accessed 2018-01-10]. Available from: <https://www.mathworks.com/matlabcentral/fileexchange/4306-fuzzy-art-and-fuzzy-artmap-neural-networks>
- [12] GARRETT D., PETERSON D.A., ANDERSON C.W., THAUT M.H. Comparison of linear, nonlinear, and feature selection methods for EEG signal classification. *IEEE Trans Neural Syst Rehabil Eng*. 2003, 11(2), pp. 141–144, doi: [10.1109/TNSRE.2003.814441](https://doi.org/10.1109/TNSRE.2003.814441).
- [13] GUGER C., DABAN S., SELLERS E., HOLZNER C., KRAUSZ G., CARABALONA R., GRAMATICA F., EDLINGER G. How many people are able to control a P300-based brain-computer interface (BCI)? *Neurosci. Lett.* 2009, 462(1), pp. 94–98, doi: [10.1016/j.neulet.2009.06.045](https://doi.org/10.1016/j.neulet.2009.06.045).
- [14] HOFFMANN U., VESIN J.-M., EBRAHIMI T., DISERENS K. An efficient P300-based brain-computer interface for disabled subjects. *Journal of Neuroscience Methods*. 2008, 167(1), pp. 115–125, doi: [10.1016/j.jneumeth.2007.03.005](https://doi.org/10.1016/j.jneumeth.2007.03.005).
- [15] JIN J., ALLISON B.Z., ZHANG Y., WANG X., CICHOCKI A. An ERP-based BCI using an oddball paradigm with different faces and reduced errors in critical functions. *Int J Neural Syst*. 2014, 24(8), doi: [10.1142/S0129065714500270](https://doi.org/10.1142/S0129065714500270).
- [16] JIN J., SELLERS E.W., ZHOU S., ZHANG Y., WANG X., CICHOCKI A. A P300 brain-computer interface based on a modification of the mismatch negativity paradigm. *Int. J. Neural Syst*. 2015, 25(3), doi: [10.1142/S0129065715500112](https://doi.org/10.1142/S0129065715500112).
- [17] JOUTSINIEMI S.L., KASKI S., LARSEN A.T. Self-organizing map in recognition of topographic patterns of EEG spectra. *IEEE Transactions on Biomedical Engineering*. 1995, 42, pp. 1062–1068, doi: [10.1109/10.469372](https://doi.org/10.1109/10.469372).
- [18] KAUFMANN T., SCHULZ S.M., KOBLITZ A., RENNER G., WESSIG C., KUBLER A. Face stimuli effectively prevent brain-computer interface inefficiency in patients with neurodegenerative disease. *Clin Neurophysiol*. 2013, 124(5), pp. 893–900, doi: [10.1016/j.clinph.2012.11.006](https://doi.org/10.1016/j.clinph.2012.11.006).
- [19] KAUR M., AHMED P., SONI A., RAFIQ M.Q. Wavelet transform use for P300 signal clustering by self-organizing map. *International Journal of Scientific and Engineering Research*. 2013, 11(4), pp. 1631–1638, doi: [10.1109/ISCCSP.2008.4537317](https://doi.org/10.1109/ISCCSP.2008.4537317).
- [20] KOHONEN T. *Self-organization and associative memory: 3rd edition*. New York, NY: Springer-Verlag, Inc., 1989.

- [21] KRUSIENSKI D.J., SELLERS E.W., CABESTAING F., BAYOUDH S., MCFARLAND D.J., VAUGHAN T.M., WOLPAW J.R. A comparison of classification techniques for the P300 Speller. *Journal of Neural Engineering*. 2006, 3(4), pp. 299–305, doi: [10.1088/1741-2560/3/4/007](https://doi.org/10.1088/1741-2560/3/4/007).
- [22] LI Y., PAN J., LONG J., YU T., WANG F., YU Z., WU W. Multimodal BCIs: Target detection, multidimensional control, and awareness evaluation in patients with disorder of consciousness. *Proceedings of the IEEE*. 2016, 104(2), pp. 332–352, doi: [10.1109/JPROC.2015.2469106](https://doi.org/10.1109/JPROC.2015.2469106).
- [23] LIANG N., BOUGRAIN L. Non-identity Learning Vector Quantization applied to evoked potential detection. In: *Deuxième conférence française de Neurosciences Computationnelles, "Neurocomp08"*, Marseille, France. 2008.
- [24] LOTTE F., CONGEDO M., LÉCUYER A., LAMARCHE F., ARNALDI B. A review of classification algorithms for EEG-based brain-computer interfaces. *Journal of neural engineering*. 2007, 4(2), doi: [10.1088/1741-2560/4/2/R01](https://doi.org/10.1088/1741-2560/4/2/R01).
- [25] LUCK S. *An introduction to the event-related potential technique*. Cambridge, MA: Cognitive neuroscience, MIT Press, 2005.
- [26] MCFARLAND D.J., WOLPAW J.R. Brain-computer interfaces for communication and control. *Commun. ACM*. 2011, 54(5), pp. 60–66, doi: [10.1016/S1388-2457\(02\)00057-3](https://doi.org/10.1016/S1388-2457(02)00057-3).
- [27] MIDENET S., GRUMBACH A. Learning ASociations by self-organization: The LASSO model. *Neurocomputing*. 1994, 6(3), pp. 343 – 361, doi: [10.1016/0925-2312\(94\)90069-8](https://doi.org/10.1016/0925-2312(94)90069-8).
- [28] MORALES C., HELD C.M., ESTEVEZ P.A., PEREZ C.A., REYES S., PEIRANO P., ALGARIN C. Single trial P300 detection in children using expert knowledge and SOM. In: *Conf Proc IEEE Eng Med Biol Soc 2014*, Chicago, USA. Piscataway: IEEE, 2014, pp. 3801–3804.
- [29] MULLER K.R., ANDERSON C.W., BIRCH G.E. Linear and nonlinear methods for brain-computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*. 2003, 11(2), pp. 165–169, doi: [10.1109/TNSRE.2003.814484](https://doi.org/10.1109/TNSRE.2003.814484).
- [30] ORTNER R., PRUECKL R., PUTZ V., SCHARINGER J., BRUCKNER M., SCHNUEERER A., GUGER C. Accuracy of a P300 speller for different conditions: A comparison. In: *Proceedings of the 5th International Brain-Computer Interface Conference*, Graz, Austria. New York City: ACM, 2011.
- [31] POLICH J. Updating P300: an integrative theory of P3a and P3b. *Clinical neurophysiology*. 2007, 118(10), pp. 2128–2148, doi: [10.1016/j.clinph.2007.04.019](https://doi.org/10.1016/j.clinph.2007.04.019).
- [32] THULASIDAS M., GUAN C., WU J. Robust classification of EEG signal for brain-computer interface. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*. 2006, 14(1), pp. 24–29, doi: [10.1109/TNSRE.2005.862695](https://doi.org/10.1109/TNSRE.2005.862695).
- [33] VAKIL-BAGHMISHEH M.-T., PAVEŠIĆ N. A fast simplified Fuzzy ARTMAP network. *Neural Process. Lett.* . 2003, 17(3), pp. 273–316, doi: [10.1023/A:1026004816362](https://doi.org/10.1023/A:1026004816362).
- [34] VAŘEKA L., BRŮHA P., MOUČEK R. Event-related potential datasets based on a three-stimulus paradigm. *GigaScience*. 2014, 3(1), doi: [10.1186/2047-217X-3-35](https://doi.org/10.1186/2047-217X-3-35).
- [35] VAŘEKA L., BRŮHA P., MOUČEK R. Supporting material for: "Event-related potential datasets based on three-stimulus paradigm". *GigaDB*. 2014, doi: [10.5524/100111](https://doi.org/10.5524/100111).
- [36] VAŘEKA L., MAUTNER P. The event-related potential data processing using ART 2 network. In: *5th International Conference on Biomedical Engineering and Informatics (BMEI)*, Chongqing, China. Piscataway: IEEE, 2012, pp. 605–609.
- [37] VAŘEKA L., MAUTNER P. Self-organizing maps for event-related potential data analysis. In: *HEALTHINF 2014 – Proceedings of the International Conference on Health Informatics*, Angers, France. Setúbal: Scitepress, 2014, pp. 387–392.
- [38] VAŘEKA L., MAUTNER P. Using the windowed means paradigm for single trial P300 detection. In: *38th International Conference on Telecommunications and Signal Processing (TSP)*, Prague, Czech Republic. Piscataway: IEEE, 2014.