Brain Neural Data Analysis Using Machine Learning Feature Selection and Classification Methods

Lachezar Bozhkov[1], Petia Georgieva[2], and Roumen Trifonov[1],

[1] Computer Systems Department, Technical University of Sofia, 8 St.Kliment Ohridski Boulevard,Sofia 1756, Bulgaria
[2] DETI/IEETA, University of Aveiro, 3810-193 Aveiro, Portugal
lachezar.bozhkov@gmail.com, petia@ua.pt, r_trifonov@tu-sofia.bg

Abstract. The Electroencephalogram (EEG) is a powerful instrument to collect vast quantities of data about human brain activity. A typical EEG experiment can produce a two-dimensional data matrix related to the human neuronal activity every millisecond, projected on the head surface at a spatial resolution of a few centimeters. As in other modern empirical sciences, the EEG instrumentation has led to a flood of data and a corresponding need for new data analysis methods. This paper summarizes the results of applying supervised machine learning (ML) methods to the problem of classifying emotional states of human subjects based on EEG. In particular, we compare six ML algorithms to distinguish event-related potentials, associated with the processing of different emotional valences, collected while subjects were viewing high arousal images with positive or negative emotional content. 98% inter-subject classification accuracy based on the majority of votes between all classifiers is the main achievement of this paper, which outperforms previous published results.

Keywords: emotion valence recognition, feature selection, Event Related Potentials (ERPs).

1 Introduction

The quantification and automatic detection of human emotions is the focus of the interdisciplinary research field of Affective Computing (AC). In [1] a broad overview of the current AC systems is provided. Major modalities for affect detection are facial expressions, voice, text, body language and posture. Affective neuroscience is a new modality that attempt to find the neural correlates of emotional processes [2]. Literature on learning to decode human emotions from Event Related Potentials (ERPs) was reviewed by [3], building automatic recognition systems from EEG was proposed by [4] and [5]. Despite the first promising results of the affective neuroscience modality to decode basic human emotional states, a confident neural model of emotions is still not defined. The recent overview of EEG-based emotion recognition studies, provided in [6], show that the recognition rate ranges between 65-90 %. Therefore, the primary motivation of the present paper is to determine a

V. Mladenov et al. (Eds.): EANN 2014, CCIS 459, pp. 123–132, 2014.

framework to improve the recognition of human affective states based on brain data and more particularly on ERPs. ERPs are transient components in the EEG generated in response to a stimulus (a visual or auditory stimulus, for example). We studied six supervised machine learning (ML) algorithms, namely Artificial Neural Networks (ANN), Logistic Regression (LogReg), Linear Discriminant Analysis (LDA), k-Nearest Neighbors (kNN), Naïve Bayes (NB), Support Vector Machines (SVM), Decision Trees (DT) and Decision Tree Bootstrap Aggregation (Tbagger) to distinguish affective valences encoded into the ERPs collected while subjects were viewing high arousal images with positive or negative emotional content. Our work is also inspired by advances in experimental psychology [7], [8] that show a clear relation between ERPs and visual stimuli with underlined negative content (images with fearful and disgusted faces). A crucial step preceding the classification process is to discover which spatial-temporal patterns (features) in the ERPs indicate that a subject is exposed to stimuli that induce emotions. We applied successfully the Sequential Feature Selection (SFS) technique to minimize significantly the number of the relevant spatial temporal patterns.

The paper is organized as follows. In section 2 we briefly describe the data set. The ML feature selection and classification methods used in this study are summarized in section 3. The results of learning to discriminate emotional states with positive or negative valences across multiple subjects (inter-subject setting) are presented in section 4. Finally, in section 5 our conclusions are drawn.

2 Data Set

A total of 26 female volunteers participated in the study, 21 channels of EEG, positioned according to the 10-20 system and 2 EOG channels (vertical and horizontal) were sampled at 1000Hz and stored. The signals were recorded while the volunteers were viewing pictures selected from the International Affective Picture System. A total of 24 of high arousal (> 6) images with positive valence (7.29 +/- 0.65) and negative valence (1.47 +/- 0.24) were selected. Each image was presented 3 times in a pseudo-random order and each trial lasted 3500ms: during the first 750ms, a fixation cross was presented, then one of the images during 500ms and at last a black screen during the 2250ms.

The signals were pre-processed (filtered, eye-movement corrected, baseline compensation and epoched using NeuroScan. The single-trial signal length is 950ms with 150ms before the stimulus onset. The ensemble average for each condition was also computed and filtered using a zero-phase filtering scheme. The maximum and minimum values of the ensemble average signals were detected. Then starting by the localization of the first minimum the features are defined as the latency and amplitude of the consecutive minimums and the consecutive maximums: minimums (Amin1, Amin2, Amin3), the first three maximums (Amax1, Amax2, Amax3), and their associated latencies (Lmin1, Lmin2, Lmin3, Lmax1, Lmax2, Lmax3). The ensemble average for each condition (positive/negative valence) was also computed and filtered using a Butterworth filter of 4th order with passband [0.5 - 15]Hz. The number of

features stored per channel is 12 corresponding to the latency (time of occurrence) and amplitude of either n = 3 maximums and minimums, the features correspond to the time and amplitude characteristics of the first three minimums occurring after T = 0s and the corresponding maximums in between. The total number of features per trail is 252. The data is saved in file with the following structure: 252 columns: 12 features for 21 channels, 52 lines: 26 people x 2 classes – 0 (negative) and 1 (positive).

3 Classification Methodology

Predictor data is normalized to maximally ease the learning algorithms.. In order to maximize the training examples, leave-one-out cross-validation technique is used. The following supervised machine learning models are studied: Artificial Neural Networks (ANN), Logistic Regression (LogReg), Linear Discriminant Analysis (LDA), k-Nearest Neighbors (kNN), Naïve Bayes (NB), Support Vector Machines (SVM), Decision Trees (DT) and Decision Tree Bootstrap Aggregation (Tbagger).

3.1 Features Normalization

Many of the models require normalized version of the data. The rest of the models can highly benefit from it. Therefore this is often a good preprocessing practice.

Feature normalization is a standard preprocessing step, that may improve the classification, particularly when the range of the features is dispersed. There are a number of normalization techniques, in this work we use the following expression:

$$X_{norm} = (X - X_{mean}) / std(X) , \quad (\mathbf{1})$$

The normalized data (X_{norm}) is obtained by subtracting the mean value of each feature from the original data set X and divided by the standard deviation std(X). Hence, the normalized data has zero mean and standard deviation equal to 1.

3.2 Leave-One-Out Cross-Validation (LOOCV)

Leave-one-out is the degenerate case of K-Fold Cross Validation, where K is chosen as the total number of examples. For a dataset with N examples, perform N experiments. For each experiment use N-1 examples for training and the remaining 1 example for testing [9]. In our case N = 26 (pairs of classes per person). We will train the models with 25 people x 2 classes (50 examples) and test on the left-out 2 classes. We are more interested in the total prediction accuracy for each model, therefore the predictions are accumulated in confusion matrices for each model from each training experiment in the LOOCV.

3.3 Artificial Neural Network (ANN)

The ANNs origin from algorithms that try to mimic the brain neuronal structure. ANNs are widely used ML technique as classifiers and repressors in countless applications. In the present work, prediction is performed by a feedforward neural network (FFNN) with 1 hidden layer with 12 neurons with sigmoid activation function and training is performed by backpropagation algorithm to compute the gradient [10].

3.4 Logistic Regression (LogReg)

In statistics, LogReg is a type of probabilistic statistical classification model [11]. It is also used to predict a binary response from a binary predictor, used for predicting the outcome of a categorical dependent variable (i.e., a class label) based on one or more predictor variables (features).

3.5 Linear Discriminant Analysis (LDA)

Discriminant analysis is a classification method. It assumes that different classes generate data based on different Gaussian distributions. To train (create) a classifier, the fitting function estimates the parameters of a Gaussian distribution for each class. To predict the classes of new data, the trained classifier finds the class with the smallest misclassification cost. LDA is also known as the Fisher discriminant, named for its inventor, Sir R. A. Fisher [12].

3.6 k-nearest Neighbor (kNN)

Given a set X of n points and a distance function, kNN searches for the k closest points in X to a query point or set of points Y [13]. The kNN search technique and kNN-based algorithms are widely used as benchmark learning rules. The relative simplicity of the kNN search technique makes it easy to compare the results from other classification techniques to kNN results. The distance measure is Euclidean.

3.7 Naive Bayes (NB)

The NB classifier is designed for use when features are independent of one another within each class, but it appears to work well in practice even when that independence assumption is not valid. It classifies data in two steps:

Training step: Using the training samples, the method estimates the parameters of a probability distribution, assuming features are conditionally independent given the class.

Prediction step: For any unseen test sample, the method computes the posterior probability of that sample belonging to each class. The method then classifies the test sample according the largest posterior probability.

The class-conditional independence assumption greatly simplifies the training step since you can estimate the one-dimensional class-conditional density for each feature individually. While the class-conditional independence between features is not true in general, research shows that this optimistic assumption works well in practice. This assumption of class independence allows the NB classifier to better estimate the parameters required for accurate classification while using less training data than many other classifiers. This makes it particularly effective for datasets containing many predictors or features [13].

3.8 Support Vector Machines (SVM)

An SVM classifies data by finding the best hyperplane that separates all data points of one class from those of the other class. The best hyperplane for an SVM means the one with the largest margin between the two classes. Margin means the maximal width of the slab parallel to the hyperplane that has no interior data points. We use radial basis function for kernel function [13].

3.9 Decision Tree (DT)

Classification trees and regression trees are the two main DT techniques to predict responses to data. To predict a response, follow the decisions in the tree from the root (beginning) node down to a leaf node. The leaf node contains the response. Classification trees give responses that are nominal, such as 'true' or 'false' [13].

3.10 Decision Tree Bootstrap Aggregation (Tbagger)

Bagging, which stands for "bootstrap aggregation," is a type of ensemble learning. To bag a weak learner such as a decision tree on a dataset, generate many bootstrap replicas of this dataset and grow decision trees on these replicas. Obtain each bootstrap replica by randomly selecting N observations out of N with replacement, where N is the dataset size. To find the predicted response of a trained ensemble, take an average over predictions from individual trees [13].

4 Features Selection

The feature space consists of 252 features (21 channels x12 features) and the trial examples are 52 (2 classes x 26 people), therefore feature reduction techniques are required. First classification tests are made on all predictor data features (252 features) and the accuracy results from ML methods are set as base line to improve and compare. Next we try feature reduction using Principal Component Analysis (PCA) [14] and dimensions reduction with 99%, 95%, 75% and 50% data variation retained. After that we implement exhaustive feature selection and compare the results. Finally we construct voting ensemble bucket of models to take the prediction among all the models which resulted in very promising final data discrimination (98%).

4.1 Principal Component Analysis (SFS)

Principal component analysis is a quantitatively rigorous method for achieving this simplification. The method generates a new set of variables, called principal components. Each principal component is a linear combination of the original variables. All the principal components are orthogonal to each other, so there is no redundant information. The principal components as a whole form an orthogonal basis for the space of the data [13].

4.2 Sequential Feature Selection (SFS)

Sequential feature selection selects a subset of features from the data matrix X that best predict the data in y by sequentially selecting features until there is no improvement in prediction. Starting from an empty feature set, SFS creates candidate feature subsets by sequentially adding each of the features not yet selected. For each candidate feature subset, SFS performs leave-one-out cross-validation by repeatedly calling fun with different training subsets X_{TRAIN} and y_{train}, and test subsets X_{TEST} and y_{test}. Each time it is called, fun must return a scalar value criterion. After computing the mean criterion values for each candidate feature subset, SFS chooses the candidate feature subset that minimizes the mean criterion value. This process continues until adding more features does not decrease the criterion or to predefined number of selected feature. In our case the criterion function is based on the accuracy of the model: criterion = 1 – Accuracy. Accuracy can be either 1 if it accurately predict the one left training example or 0 if doesn't. Therefore the minimization cost function will have 1/52 = 0.0192 step. Because SFS is computationally heavy operation, not all models are suitable for this technique, especially TBagger and ANN.

4.3 Voting from Ensemble Bucket of Models

After selecting suitable features for each model, we ensemble a model consisting of the five models. When we predict we would train all 5 models with the training data and predict with all of them using the test data. We get the consensus from at least 3 of the models to select the result.

5 Results for Inter-Subject Classification

5.1 Classification Using All Features

In Table 1 are given the prediction accuracy results using all features for test and train data. Comparison of the prediction accuracy using all features and the selected features is shown on Fig. 2.

Table 1. Prediction accuracy results from classification models using all features

Model	ANN	LogReg	LDA	kNN	NB	SVM	DT	Tbagger
Accuracy X_{TEST}	71,2	67,31	71,2	59,6	69,2	50	69,2	75
Accuracy X_{TRAIN}	75,6	100	100	100	93	100	96,2	100

5.2 PCA Feature Reduction and Classification

After calculating eigenvectors we estimate the numbers of vectors used to project the data with 99%, 95%, 75% and 50% data variance retained corresponding number of features is 43, 34, 16, and 7. Results from the prediction accuracies can be seen in Table 2. It is seen that we cannot improve significantly prediction accuracy using PCA and data projection in lower dimensionality.

Table 2. Results from models using reduced (projected) by PCA features set

Model	ANN	LogReg	LDA	kNN	NB	SVM	DT	Tbagger
43 Features (99%)	53,9	67,31	65,4	59,6	61,5	57,7	48,1	57,69
34 Features (95%)	61,5	69,23	65,4	57,7	67,3	57,7	53,9	63,46
16 Features (75%)	57,7	71,15	67,3	55,8	65,4	63,5	63,5	69,23
7 Features (50%)	55,8	59,62	61,5	69,2	65,4	71,2	63,5	61,54

5.3 Exhaustive Sequential Feature Selection (SFS) and Classification

Exhaustive SFS is computationally very intensive operation, therefore the SFS was performed on a smaller set of ML models. The resulting cost function (1-accuracy) based on the number of selected features is depicted on Fig. 1. Note that the number of features that minimizes the cost function is different for each model, typically between 5 and 10.

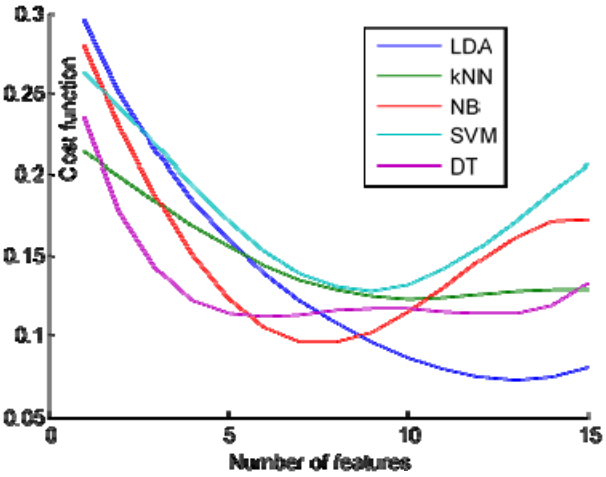

Fig. 1. Features selection: Cost function on numbers of features selected

Table 3. Features selected by SFS for each model

Features	LDA		kNN		NB		SVM		DT	
Number	Ch.	Feature	Ch.	Feature	Ch.	Feature	Ch.	Feature	Ch.	Feature
1	1	amp5	4	amp1	1	amp4	1	amp6	1	amp4
2	3	amp1	5	amp1	2	latency4	3	latency1	12	amp3
3	5	latency2	8	latency3	3	amp1	5	latency4	14	amp2
4	6	latency3	10	amp1	4	amp6	5	latency5	20	latency4
5	6	latency4	10	amp6	9	amp2	11	latency6		
6	11	amp3	13	amp3	20	latency4	13	amp2		
7	13	amp1	14	latency4						
8	13	amp6	20	amp2						
9	17	latency6								
10	20	latency4								

Table 4. Prediction accuracy on test and train data for models trained using the selected features from Table 2

Model	LDA	kNN	NB	SVM	DT
Accuracy $X_{\mathbf{TEST}}$	92,3	90,38	86,5	88,5	88,5
Accuracy $X_{\mathbf{TRAIN}}$	94,2	100	91,2	100	97,4

5.4 Voting from Ensemble Bucket of Models

The combination of SFS and the five ML classifiers in the previous section brought already results very close or even slightly better than the best classification rates published in previous related researches. However, we made an intuitive step ahead to build an ensemble classifier based on the majority vote among the five trained models. Thus, the prediction rates achieved by the individual classifiers in the range of [87 – 92] %. were significantly improved and achieved 98%, see Table 4.

4). Finally we can observe and compare the prediction accuracy on all features and selected features and ensemble bucket models vote in fig. 2.

Table 5. Accuracy and confusion matrix on test data using voting from models trained using the selected features from Table 2

Accuracy $X_{\mathbf{TEST}}$	True 1	False 1	False 0	True 0
98,08	26	0	1	25

Discussion of the Results

We used supervised ML methods to predict two human emotions based on 252 features collected from 21 channels EEG. The achieved prediction accuracy based on

all features is in the range of 60-75% (see Table 1). These results are similar to other related studies, [6] and they can be explained by the limited examples in the data set (2 examples per subject, 26 subjects, that corresponds to 52 examples in total) and the very high dimensional feature space (252). It was expected that predictions based on reduced number of features will perform better. While the PCA feature reduction did not bring any improvement (see Table 2), the Sequential Feature Selection (SFS) reduced the feature set to 4-10 features (see Fig. 1 and Table 2) and significantly improve the prediction accuracy of all studied ML models in the range of 88-92 % (Table 3). Finally, our empirical approach of combining the five previous classifiers in an ensemble bucket of models and use the majority vote as the final attributed class further improve substantially the prediction accuracy to 98% (Table 4). This is the main contribution of this paper, because such inter-subject classification accuracy was never before reported. The influence of the SFS is visualized on Fig. 2. We may also argue that our models can be used in real time, because after finding off-line the right features and training, the feature generation from monitored EEG signals is less than 1000ms and prediction is instantaneous.

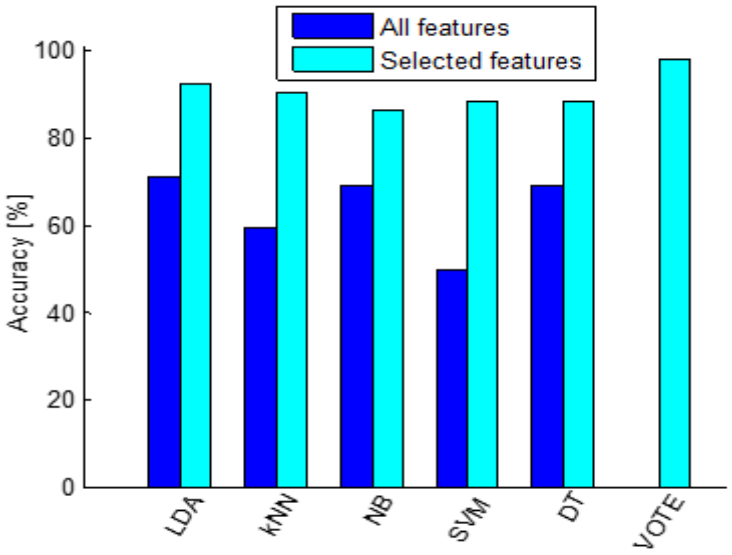

Fig. 2. Classification accuracy on test data. 5 classifiers (LDA, kNN, NB, SVM, DT) and their majority vote combination (VOTE).

6 Conclusion

In this paper, we have presented results demonstrating the feasibility of ML classification techniques to distinguish the processing of stimuli with positive and negative emotion valence based on ERPs observations. This problem is interesting both because of its relevance to studying human emotions, and as a case study of supervised machine learning (ML) in high dimensional data settings. The focus of our work was to explore

the feasibility of training cross-subject classifiers to make predictions across multiple human subjects. Feature selection is an important aspect in the design of the recognition systems, particularly in the inter-subject framework. The combination of adequate features and channel selection has the potential to reduce the inter-subject variability and improve the learning of representative models valid across multiple subjects.

It can be concluded that ML is a powerful technique to reveal the brain activity and to interpret human emotions. There are many additional opportunities for ML research in the context of affective neuroscience, such as discrimination of more than two emotional states related not only with the emotional valence but also with the emotional arousal. Discrimination of high versus low neurotic type of personality is also a challenging problem that ML can deal.

Acknowledgements. We would like to express thanks to the PsyLab from Departamento de Educação da UA, and particularly to Dr. Isabel Santos, for providing the data sets.

References

1. Calvo, R.A., D'Mello, S.K.: Affect Detection: An Interdisciplinary Review of Models, Methods, and their Applications. IEEE Transactions on Affective Computing 1(1), 18–37 (2010)
2. Dalgleish, T., Dunn, B., Mobbs, D.: Affective Neuroscience: Past, Present, and Future. Emotion Rev. 1, 355–368 (2009)
3. Olofsson, J.K., Nordin, S., Sequeira, H., Polich, J.: Affective Picture Processing: An Integrative Review of ERP Findings. Biological Psychology 77, 247–265 (2008)
4. AlZoubi, O., Calvo, R.A., Stevens, R.H.: Classification of EEG for Emotion Recognition: An Adaptive Approach. In: Proc. 22nd Australasian Joint Conf. Artificial Intelligence, pp. 52–61 (2009)
5. Petrantonakis, P.C., Hadjileontiadis, L.J.: Emotion Recognition from EEC Using Higher Order Crossings. IEEE Trans. Information Technology in Biomedicine 14(2), 186–194 (2010)
6. Jatupaiboon, N., Panngum, S., Israsena, P.: Real-Time EEG-Based Happiness Detection System. The ScientificWorld Journal, Article ID 618649, 12 pages (2013)
7. Santos, I.M., Iglesias, J., Olivares, E.I., Young, A.W.: Differential effects of object-based attention on evoked potentials to fearful and disgusted faces. Neuropsychologia 46, 1468–1479 (2008)
8. Pourtois, G., Grandjean, D., Sander, D., Vuilleumier, P.: Electrophysiological correlates of rapid spatial orienting towards fearful faces. Cerebral Cortex 14(6), 619–633 (2004)
9. Lecture 13: Validation, `http://research.cs.tamu.edu/prism/lectures/iss/iss_l13.pdf`
10. CS229 Machine Learning, Andrew Ng, `http://cs229.stanford.edu/`
11. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer (2006)
12. Fisher, R.A.: The Use of Multiple Measurements in Taxonomic Problems. Annals of Eugenics 7, 179–188 (1936)
13. Matlab documentation, `http://www.mathworks.com/help/matlab/`
14. Palaniappana, R., Ravi, K.V.R.: Improving visual evoked potential feature classification for person recognition using PCA and normalization (2005)