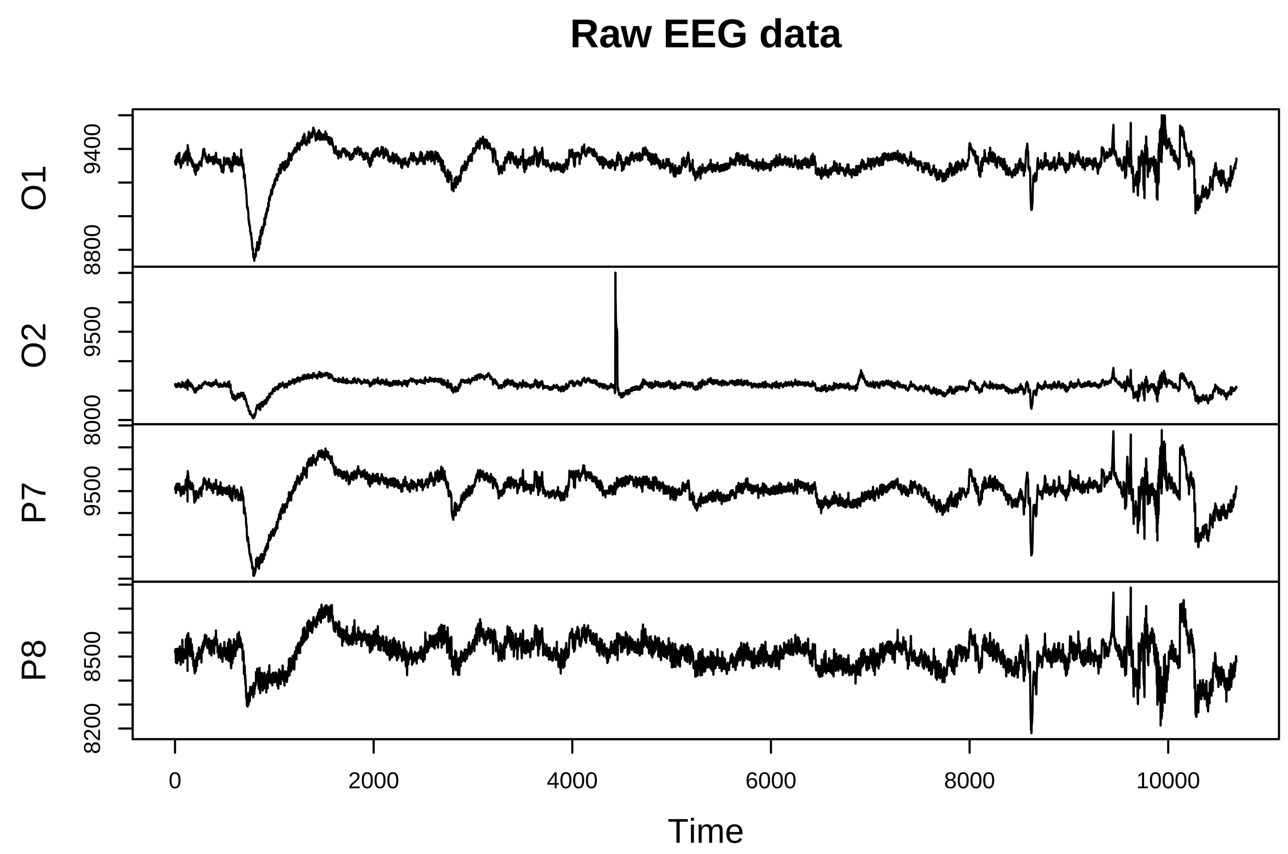
Improving Classification Algorithm of Brain-Computer Interface

Author: Anti Ingel[1]
https://github.com/kahvel/MAProject

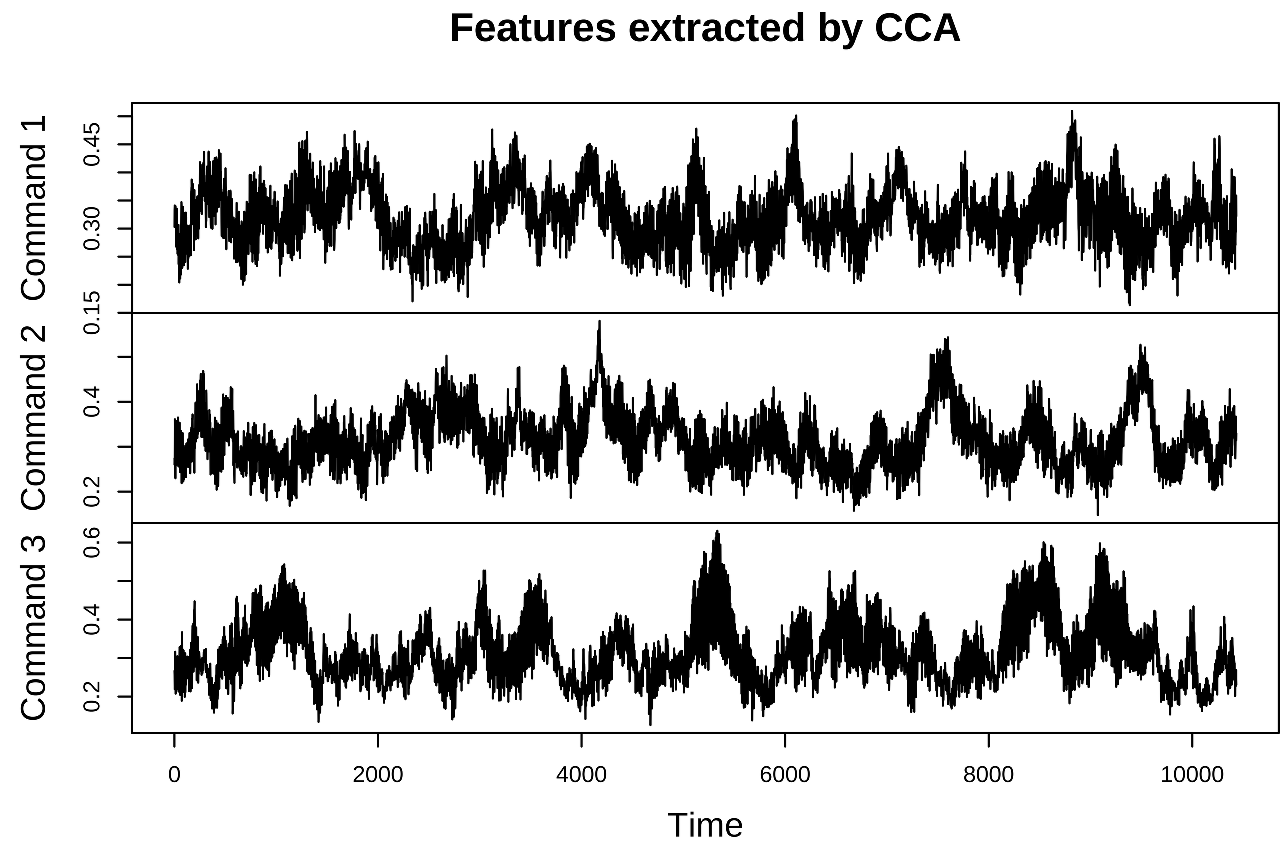
[1]Computer Science, 1st year of MSc,
University of Tartu faculty of Science and Technology,
Institute of Computer Science

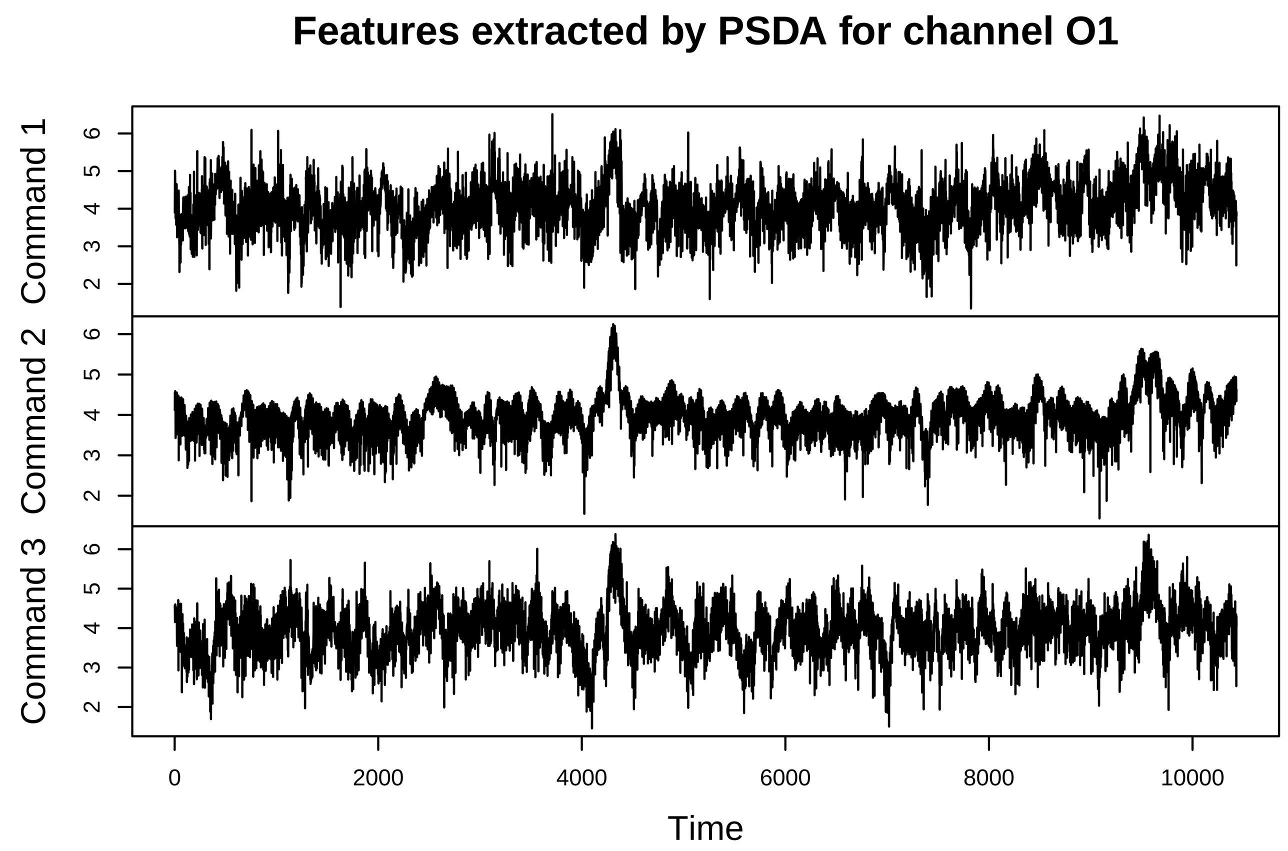
INTRODUCTION

The aim of this project is to improve the classification algorithm of a brain-computer interface (BCI) by using machine learning techniques. The BCI under consideration is author's previous work and therefore all the steps from data collection to classification were done either as a part ot this project or by using author's previous work. The BCI works by measuring users brain activity using electroencephalography (EEG) device Emotiv EPOC and then tries to find certain patterns from the EEG signal. In this case, the patterns we are interested in are changes in the amounts of frequencies present in the signal. But since brain signals are inherently very noisy and the EEG device used to collect data was a consumer-grade device, finding patterns in the signal turned out to be very challenging.



To extract frequency information from the raw signal, canonical correlation analysis (CCA) and power spectral density analysis (PSDA) methods were used. These methods can be used to estimate how much certain frequency is present in the signal over some time window. During the data collection, users could send three different commands to the BCI and each command corresponds to some frequency change. CCA method extracts three different features from the data, one for each command. PSDA method, however, is not multidimensional and it extracts three features for each EEG channel (P7, O1, O2, P8).





FACTOR STRUCTURE

	Factor 1	Factor 2	Factor 3
PSDA_1	**0.72**	0.48	0.27
PSDA_2	0.32	**0.75**	0.31
PSDA_3	0.42	0.41	**0.72**
CCA_1	**0.67**	-0.33	-0.14
CCA_2	-0.15	**0.68**	-0.17
CCA_3	-0.29	-0.40	**0.70**

Before training classifiers, the data was analysed using exploratory factor analysis. The results showed that indeed there are similarities between the features that correspond to the same command. Features corresponding to same command were grouped into one factor. Detailed report is in repository.

MACHINE LEARNING ALGORITHMS

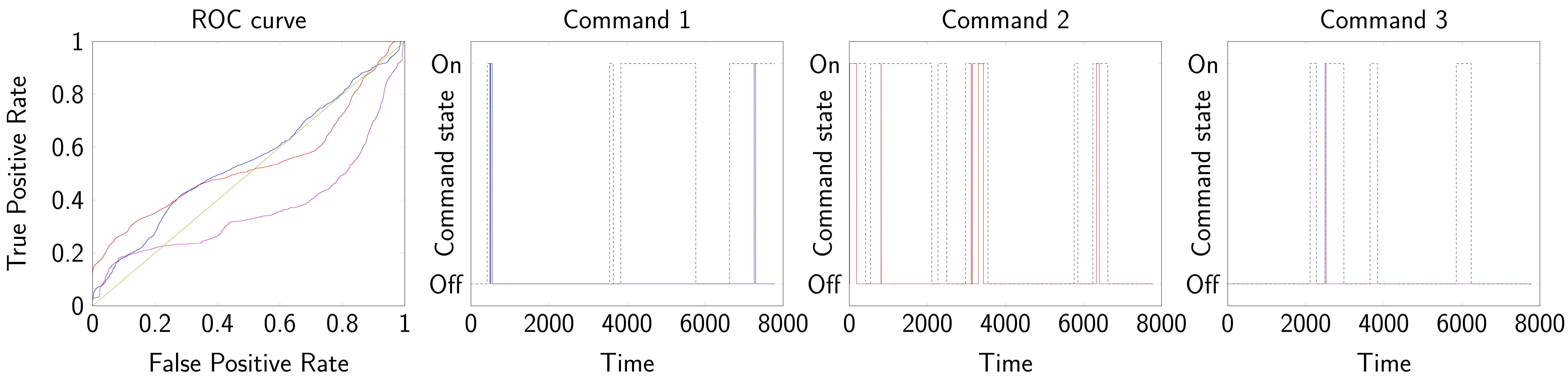
The multiclass classification task with three classes (commands) was divided into three binary classification tasks. Due to the noisiness of the data, stable learning algorithms were preferred. Many different learning algorithms were tested, including logistic regression, linear discriminant analysis, support vector machines (SVM) with different kernels, random forests and finally boosting, bagging and voting of different classifiers. The best results were achieved using soft voting of SVM and boosting of decision tree stumps. Final decision was made using the sum of the probabilities of the classes for given datapoint—if the sum of probabilities was larger than a given threshold, then the class was predicted.

RESULTS AND CONCLUSION

In the figure below, the black dashed line denotes the expected state and the coloured lines denote the state predicted by the classifier on the test set. The thresholds were chosen so that there are no false positives. As can be seen the classifier is quite good at classifying command 2, but not so good at classifying command 3, which had the least training examples. Having high precision was preferred to filter out as many false positives as possible

The results are good starting point for further study. In this project, the classification algorithms only minimally took into account that we are predicting on time series, but the fact that observations are sequential in time contains very useful information and using it more should improve the performace of the classifiers.

Command 2	On	Off
Predicted on	**514**	**45**
Predicted off	2751	4479
Command 1	On	Off
Predicted on	**202**	**43**
Predicted off	3106	4438



Thesis on the BCI under consideration: A. Ingel. Control a Robot via VEP Using Emotiv EPOC. Bachelor's thesis, University of Tartu, 2015.
BCI code repository: https://github.com/kahvel/VEP-BCI
The code repository of this project (https://github.com/kahvel/MAProject)
The code repository also contains detailed report of the factor analysis of the current data (report in Estonian).

UNIVERSITY OF TARTU
Institute of Computer Science
1632

Study IT in .ee
sponsored by Skype™