# BCPy, an open-source python platform for offline EEG signals decoding and analysis

Aurélien Appriou, Léa Pillette, Andrzej Cichocki, Fabien Lotte

# BCPy, an open-source python platform for offline EEG signals decoding and analysis

Aurélien Appriou
Inria, LaBRI (Univ. Bordeaux, CNRS, Bordeaux-INP), France / RIKEN BSI, Wakoshi, Japan

Léa Pillette
Inria, LaBRI (Univ. Bordeaux, CNRS, Bordeaux-INP), France / RIKEN BSI, Wakoshi, Japan

Andrzej Cichocki
RIKEN BSI, Wakoshi, Japan / SKOLTECH, Moscow, Russia

Fabien Lotte
Inria, LaBRI (Univ. Bordeaux, CNRS, Bordeaux-INP), France / RIKEN BSI, Wakoshi, Japan

## Introduction

Although promising, BCIs are still barely used outside laboratories due to their poor robustness. Moreover, they are sensitive to noise, outliers and the non-stationarity of electroencephalographic (EEG) signals. Many algorithms have been developed for EEG signals processing and classification. The current major platforms (BCI2000, BBCI, OpenVIBE, BCILAB, BCI++, BF++) provide modules going along the BCI process: data acquisition, signal processing, classification, statistical modelling and visualization [3]. Each platform has specific strengths, e.g., numerous data acquisition methods for BCI2000, extensive Machine Learning (ML) algorithms for BCILAB, or real time graphical-user interface (GUI) for BCI++ and OpenViBE. However, all comprise classification tools, more or less developed. For example, BCI2000, BBCI or OpenVIBE propose the Common Spatial Pattern (CSP) filter and the Linear Discriminant Analysis (LDA) classifier, which are very popular BCI ML tools [5]. Most of these platforms are open source and implemented in C++ or Matlab. We propose BCPy, an open-source, easy-to-use python BCI platform for offline EEG signal analysis. Using Python makes it simple and extendable by non-computer scientists. Python is free, whereas Matlab is not, and contains good scalable libraries for scientific computing, such as numpy (https://www.numpy.org) or scipy (https://www.scipy.org). Moreover, Python is the major language used to implement recent advances in ML and Deep Learning ([6], https://www.tensorflow.org/, http://scikit-learn.org/), thus making them easily available for BCI research.

## Material, Methods and Results

BCPy comprises four main modules: 1) reading different EEG data format, e.g. ".gdf", ".mat" and ".pkl"; 2) filtering and representing EEG signals, e.g., CSP or Filter Bank CSP (FBCSP), mutual information feature selection; 3) classifying EEG signals, e.g. LDA, Riemannian geometry or Convolutional Neural Network (CNN); 4) visualizing statistics on the analysis results. All algorithm examples mentioned above are already implemented. Each module can be used independently. Moreover, BCPy has a jupyter notebook GUI (see Figure 1), allowing users to test and compare algorithms with various parameters on their data, without any programming. BCPy has already been used to analyze 3 types of BCI data. First, we used algorithms for classification of Motor-Imagery EEG signals and compared results to the literature. For example, the FBCSP results were compared to [1]. The second study compared algorithms performances for EEG classification of workload levels (high vs low) [2]. This revealed that CNN obtained significantly better classification accuracy (user-specific mean = 72.7%) than CSP+LDA (67.0%), with both user-specific and user-independent ML calibration. Moreover, CNN also outperformed FBCSP and Riemannian geometry. Each of these algorithms has proved efficient either in recent active BCI classification competition [1,7] or in other independent studies [6]. The third study classifies four types of attention (Alertness, Sustained, Selective and Divided attention) [9], two by two, using CSP+LDA in the alpha band (8-12 Hz). Results indicate that each of the four types of attention is distinguishable from the others, with accuracies ranging from 74% to 83%. Next, we will look at the other frequency pass-bands by using FBCSP.

## Discussion

BCPy accelerated EEG signals analysis for three studies. More tools will be implemented rapidly, e.g. statistical analysis and data visualization, to make BCPy more versatile. New ML tools will be added, notably Recurrent Neural Networks, which showed promising results in many areas [8]. Finally, we aim at making BCPy available online, using jupyter notebook.

**Significance**
We propose BCPy, a free, open-source EEG analysis platform based on Python, usable by anyone without programming knowledge, hopefully bridging gaps between engineering and neuroscience/psychology researchers and accelerating BCI research.

We acknowledge support from the Japanese Society for the Promotion of Science and the European Research Council (grant ERC-2016-STG-714567).

**References**
*1. Ang et al, Frontiers in Neuroscience, 2012*
*2. Appriou et al, ACM CHI, submitted*
*3. Brunner et al, Towards Practical BCI, 2013*
*5. Lotte et al, JNE, 2007*
*6. Schirrmeister et al, HBM, 2017*
*7. Yger et al, IEEE TNSRE, 2016*
*8. Yang et al, http://arxiv.org/abs/1707.01786*
*9. Pillette et al, BCI meeting, submitted*