

# Reproducibility in BCI

## Benchmarking offline pipelines with MOABB

Marie-Constance Corsi  
Inria NERV team, Paris-Brain Institute, France  
[marie-constance.corsi@inria.fr](mailto:marie-constance.corsi@inria.fr)

Neurogonomics 2024 Workshop  
Designing Brain-Computer Interfaces with Open-Source Tools  
8 July 2024

**Reproducible research in BCI** built on a rich Python ecosystem  
to design FAIR benchmarks with the help of a community

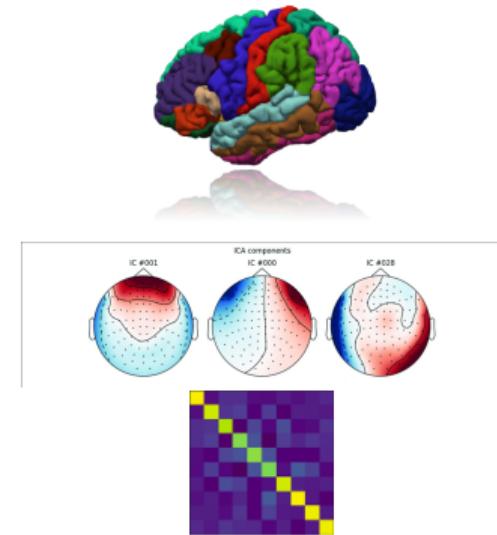
# Why open source matters

## Reproducibility issues

**Freesurfer** Popular software for extracting features from MRI  
→ Software variation lead to different conclusions

**ICA** Popular matrix factorization problem  
→ Different results with different machines

**eigs/eigsh** Popular solver for eigenvalues decomposition  
→ Solvers can lead to different outcome



Neurophysiological analysis is complex, require advanced processing  
⇒ **Need for collective efforts to build open science**

# Why do we need MOABB?

Reproducible research in BCI has a long way to go...

- Unavailable code
- Exotic data format/language/toolboxes
- Preprocessed data (including errors)

**No comprehensive benchmark of BCI algorithms**

**Huge waste of time for everyone**

⇒ **MOABB aims to be the standard benchmark for any new paper**

- Comprehensive benchmark of popular BCI algorithms
- Extensive list of freely available EEG datasets
- Ranking algorithms with fair evaluations

Reproducible research in BCI **built on a rich Python ecosystem**  
to design FAIR benchmarks with the help of a community

# MNE

<https://github.com/mne-tools/mne-python>

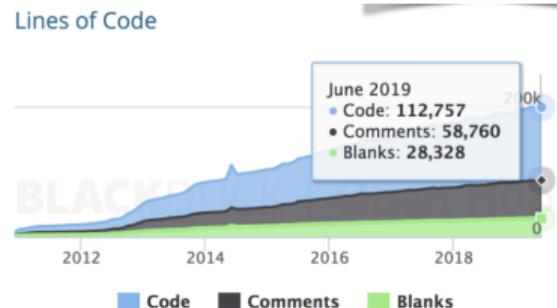
## History

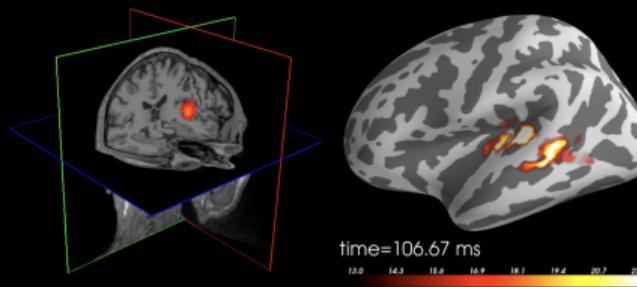
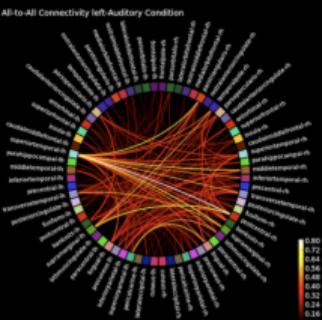
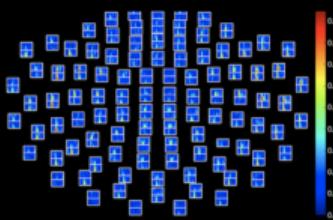
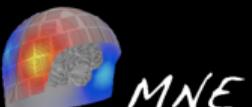
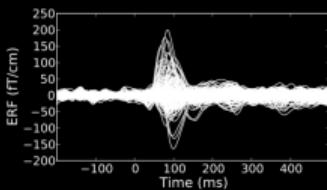
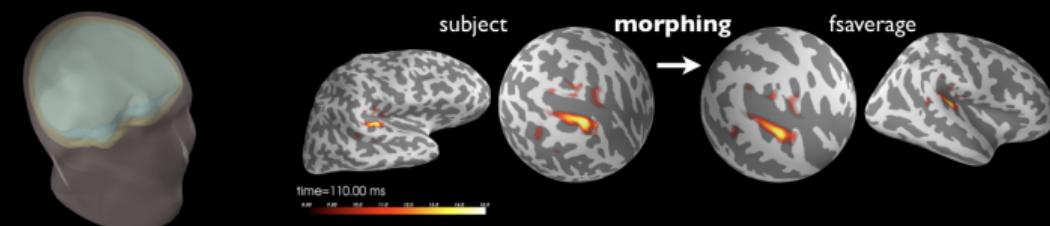
- based on C code developed for 18 years by Matti Hämäläinen
- Python started in 2010 at MGH, Boston

## In a nutshell

- 236 contributors, 100k LOC
- mature codebase, large dev team
- ~ 29 years of efforts (COCOMO)

⇒ BSD licensed (commercial use ok)  
⇒ Mac / Linux / Windows





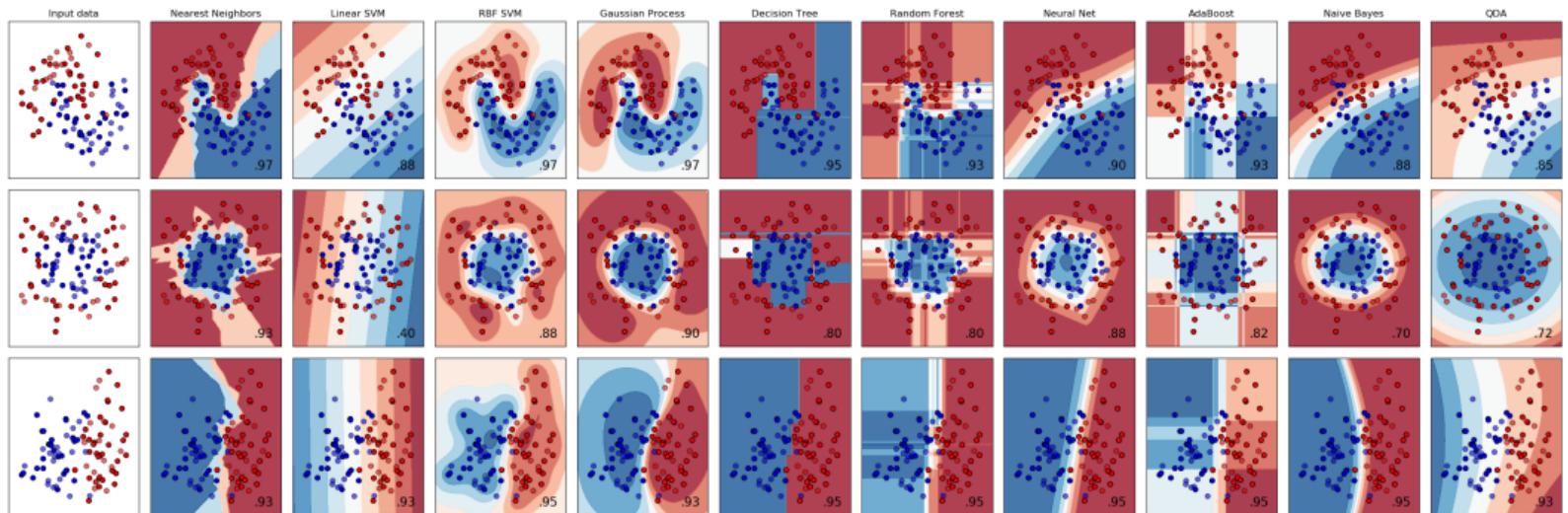
# Scikit-learn – accessible machine learning

<http://scikit-learn.org>

- **Machine learning for all**  
⇒ No specific application domain  
⇒ No requirements in machine learning
- **High-quality Pythonic software library**  
⇒ Interfaces designed for users
- **Community-driven development**  
⇒ BSD licensed, very diverse contributors

Easy as py:

```
from sklearn import svm
classifier = svm.SVC()
classifier.fit(X_train, Y_train)
Y_test = classifier.predict(X_test)
```



# PyRiemann – Riemannian ML for All !

<https://pyriemann.readthedocs.io>

- **Scikit-learn compatible**

⇒ High-level interface

⇒ Wide machine learning models

- **Multivariate time series**

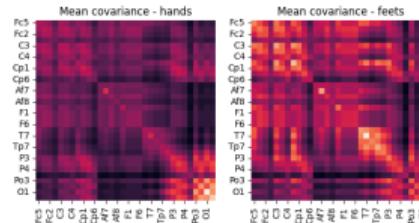
⇒ Biosignals: MEG, EEG, EMG

⇒ Radar, sensor networks, ...

- **Batteries included**

⇒ Preprocessing, transfer learning

⇒ Documentation, examples

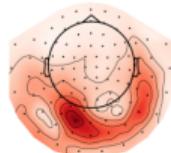
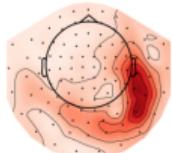
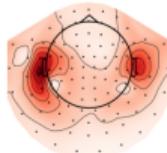
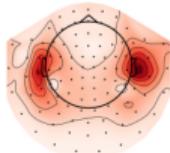


Pattern 0

Pattern 3

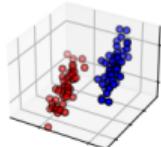
Pattern 6

Pattern 9

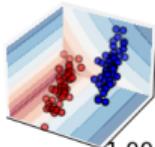


Compare classifiers with metric='riemann'

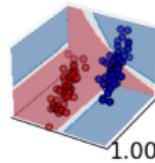
Input data



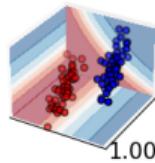
MDM



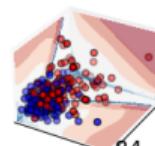
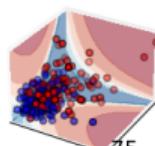
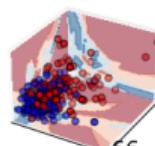
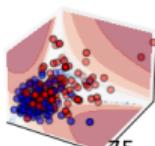
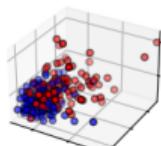
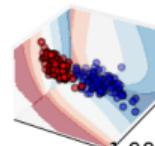
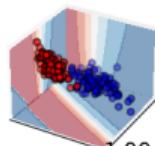
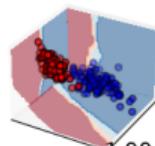
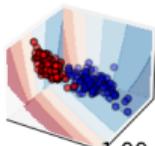
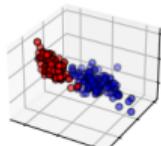
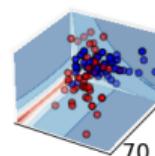
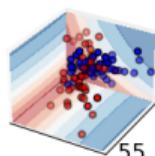
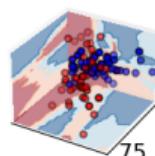
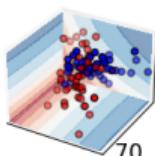
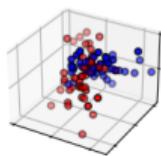
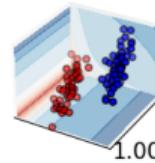
k-NN



SVC

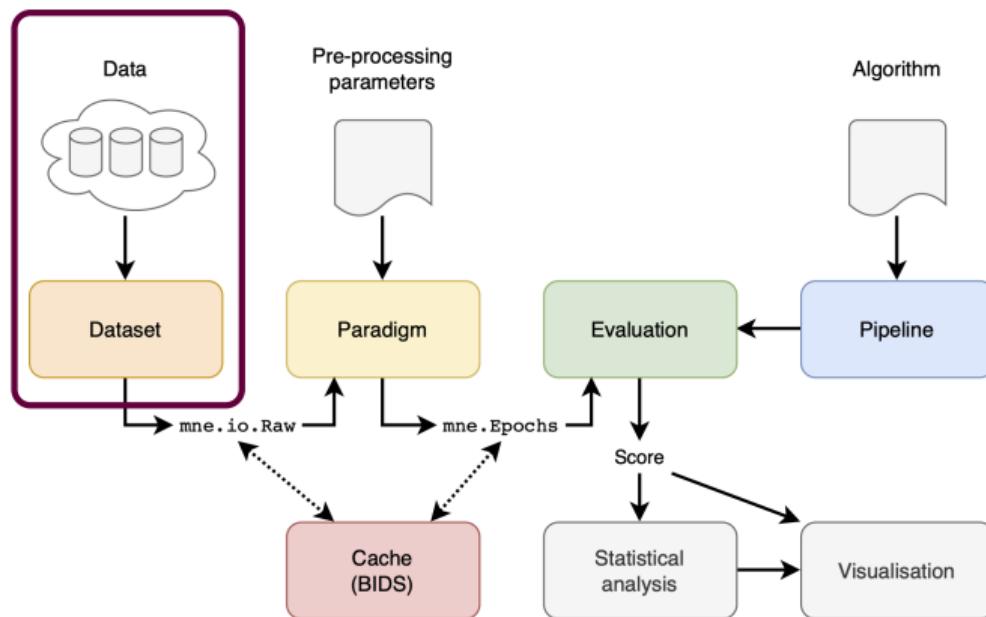


MeanField



Reproducible research in BCI built on a rich Python ecosystem  
**to design FAIR benchmarks** with the help of a community

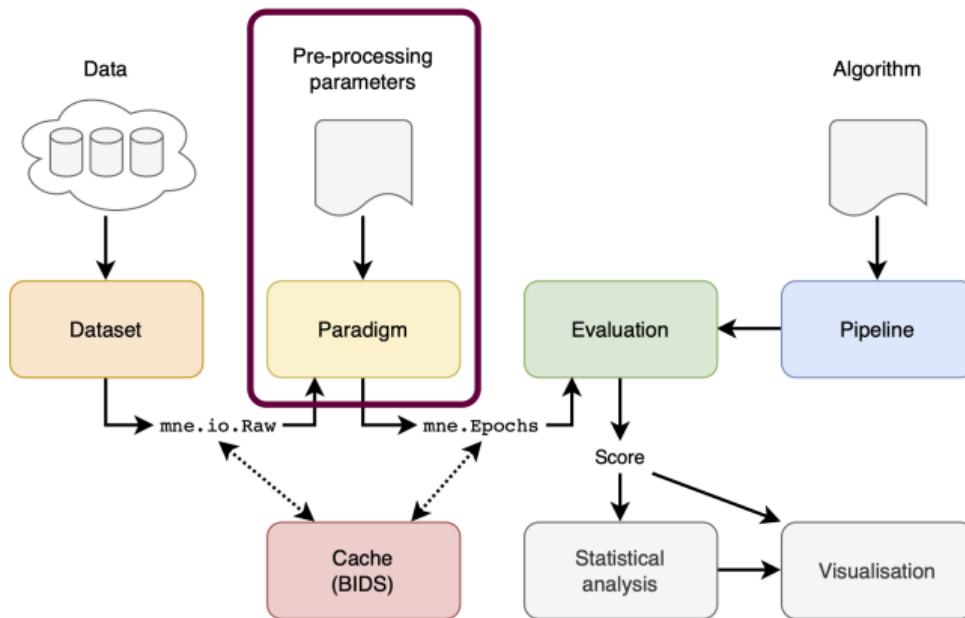
# MOABB Architecture: Datasets



## Dataset

- Stored locally, converted in MNE format
- Use BIDS cache if needed

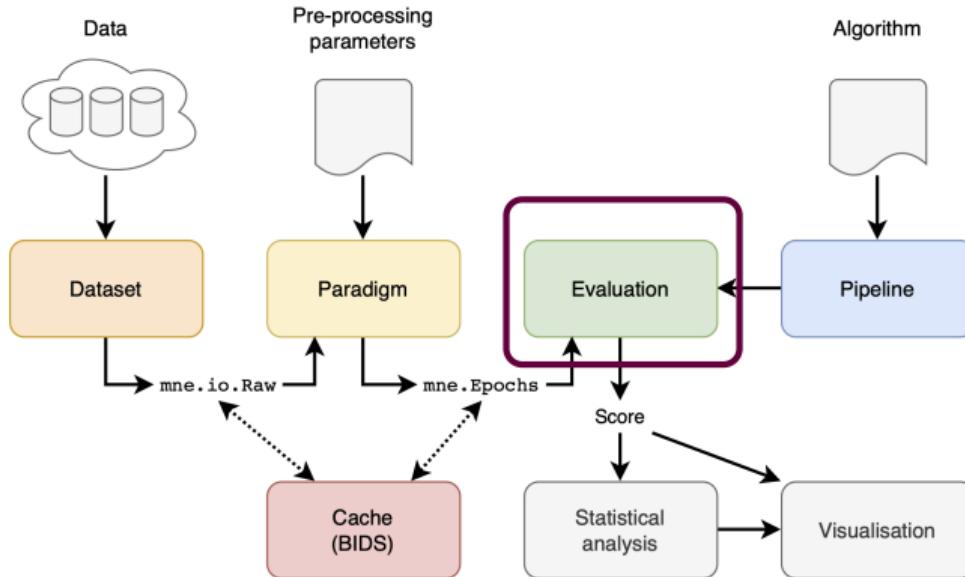
# MOABB Architecture: Paradigm



## Paradigm

- Motor Imagery, P300, SSVEP, cVEP, resting state
- Common preprocessing, output MNE Epochs or Numpy array

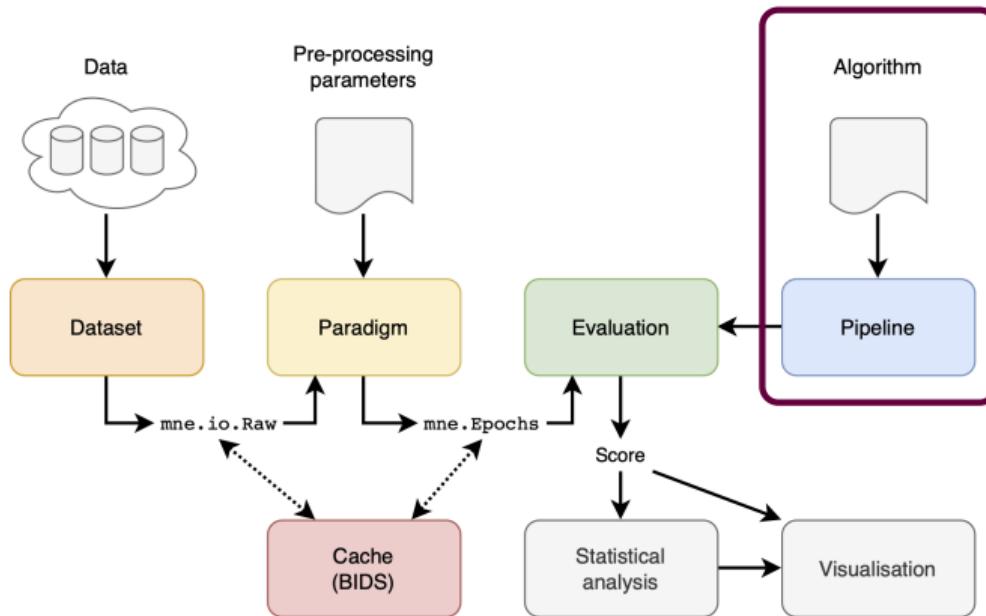
# MOABB Architecture: Evaluations



## Evaluations

- Defines a scoring method (AUC, accuracy, ...)
- within or across session, across-subject, ...

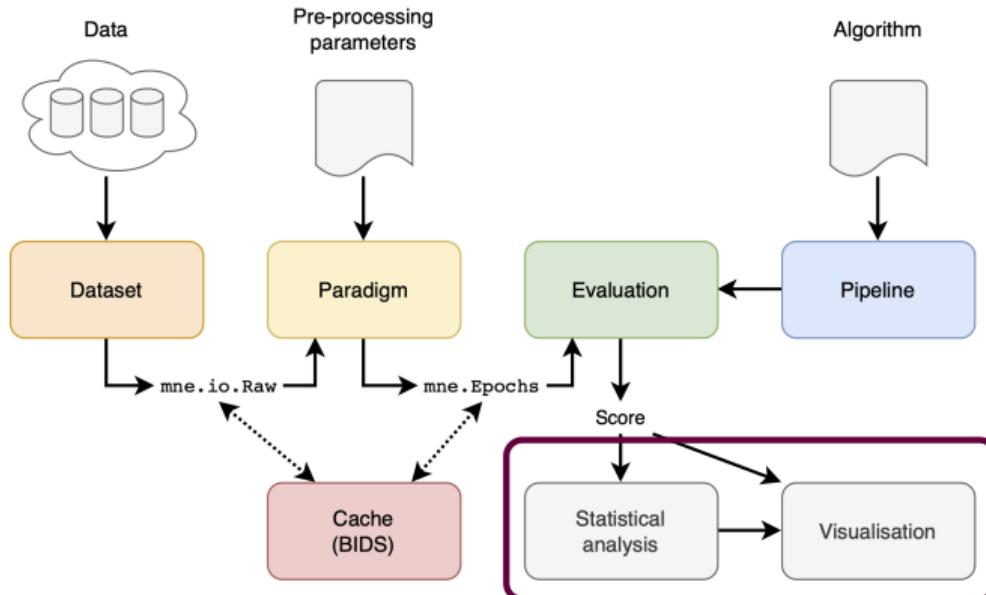
# MOABB Architecture: Pipelines



## Pipelines

- All steps required to obtain a prediction
- Scikit-learn API, deep learning

# MOABB Architecture: Results

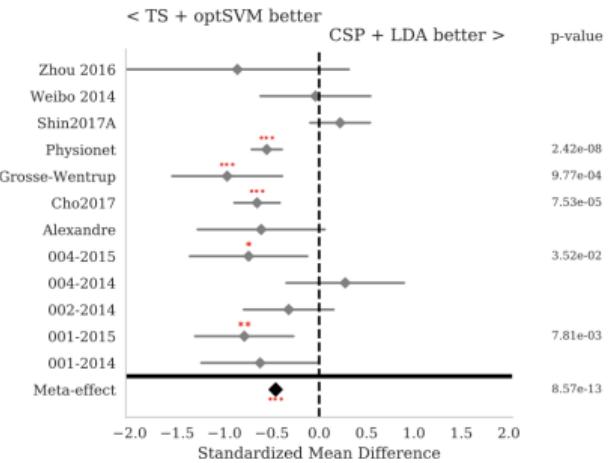


## Results

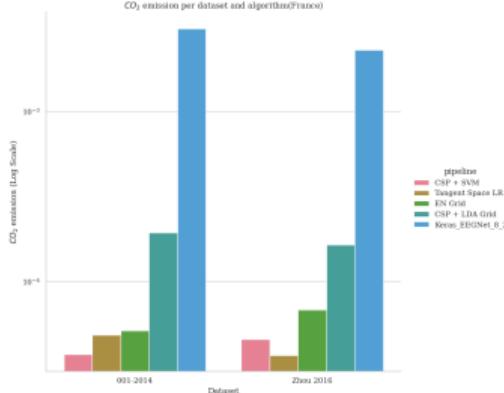
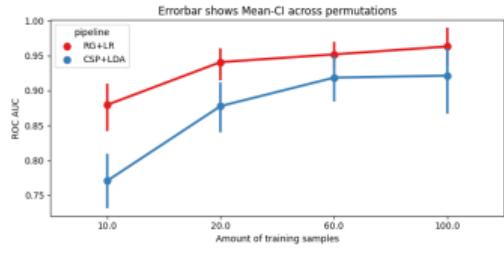
- Statistics & visualization
- Results are stored in a DataFrame

# Fair and Reproducible Benchmarks

- ➊ Load multiple datasets
- ➋ Apply pipelines
- ➌ Run meta-analysis and plot



# What's new?



New version 1.1.0 is out !

- Support latest python, MNE and sklearn version
- New documentation pages
- New datasets: 7 SSVEP, 15 ERP, 14 MI (> 1000 subjects total)
- Deep learning pipelines (Pytorch/Tensorflow)
- Carbon footprint evaluation
- Cache in BIDS format,

⇒ MOABB paper: <https://arxiv.org/abs/2404.15319>

Reproducible research in BCI built on a rich Python ecosystem  
to design FAIR benchmarks **with the help of a community**

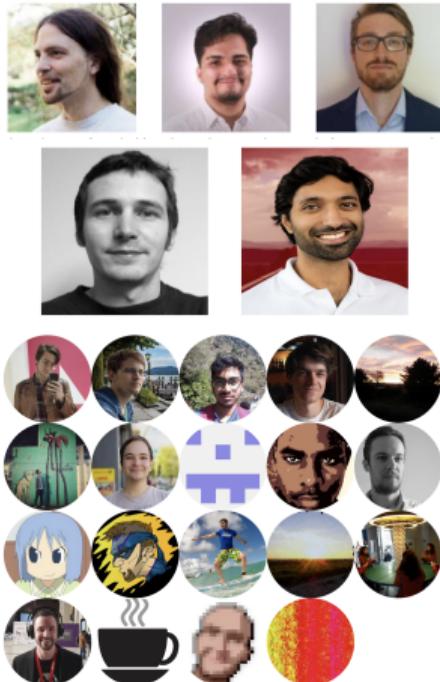
# A Community Project

**Maintainers:** Sylvain Chevallier, Bruno Aristimunha, Igor Carrara

**Founders:** Alexandre Barachant, Vinay Jayaram

**Contributors:**

- Pedro Rodrigues
- Jan Sosulski
- Erik Bjäreholt
- Divyesh Narayanan
- Pierre Guetschel
- Quentin Barthelemy
- Vladislav Goncharenko
- Ali Abdul Hussain
- Ramiro Gatti
- Lucas Custódio
- Robin Schirrmeyer
- Mohammad Mostafa Farzan
- Yannick Roy, Morgan Hough & a vibrant community



# How to contribute

Check the github and the documentation

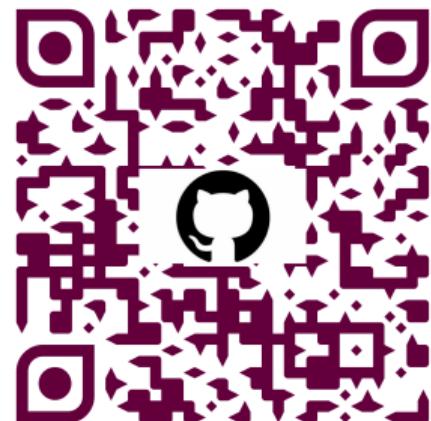
- <https://github.com/NeuroTechX/moabb>
- <https://neurotechx.github.io/moabb/>

Discuss during Office Hours or on Gitter

- <https://github.com/NeuroTechX/moabb/issues/191>
- [https://gitter.im/moabb\\_dev/community](https://gitter.im/moabb_dev/community)

Possible contributions:

- Migrate to BIDS format
- Evaluation closer to online setup
- Add new datasets and pipelines



Thank you !