Introduction

Marie-Constance Corsi
Inria NERV team, Paris-Brain Institute, France
marie-constance.corsi@inria.fr

Graz 2024 Workshop
Designing Brain-Computer Interfaces, from theory to real-life scenarios
9 September 2024

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

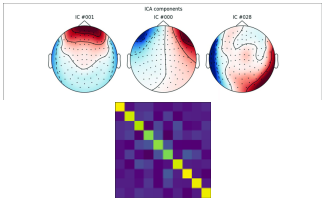
Why open source matters

Reproductibility 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**

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

MNE

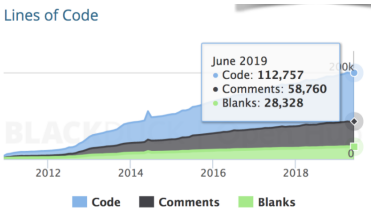
https://github.com/mne-tools/mne-python

History

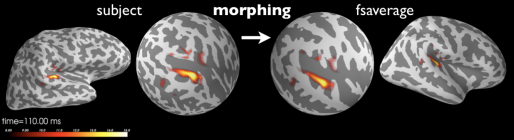
- based on C code developed for 18 years by Matti Hämäläinen
- Python started in 2010 at MGH, Boston

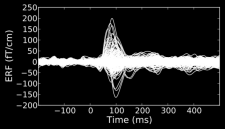
In a nutshell

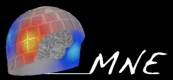
- 236 contributors, 100k LOC
- mature codebase, large dev team
- $\sim$ 29 years of efforts (COCOMO)

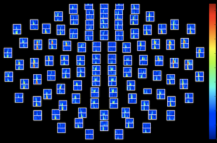


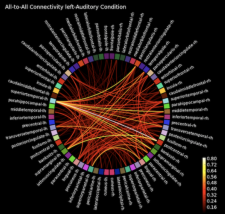
⇒BSD licensed (commercial use ok)
⇒Mac / Linux / Windows

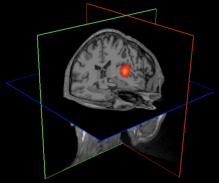

subject
morphing
fsaverage
time=110.00 ms

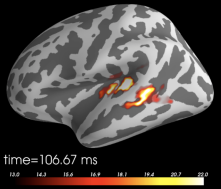

ERF (fT/cm)
Time (ms)


MNE




All-to-All Connectivity left-Auditory Condition




time=106.67 ms

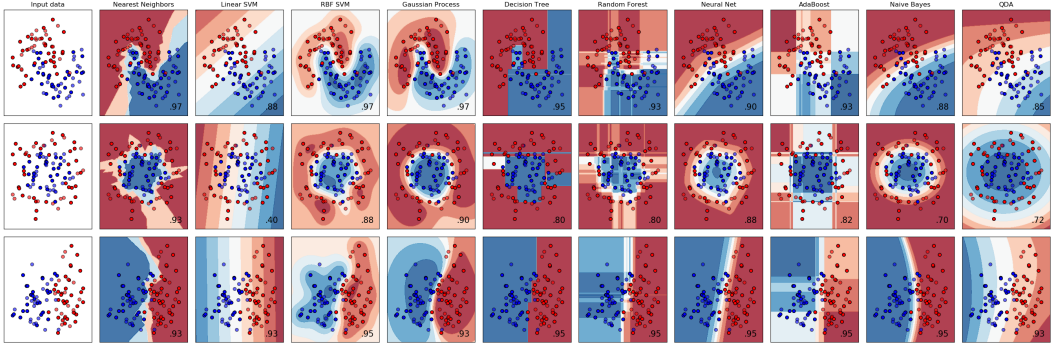
Scikit-learn – accessible machine learning

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:

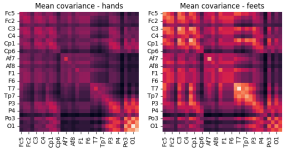
```
from sklearn import svm
classifier = svm.SVC()
classifier.fit(X train, Y_train)
Y_test = classifier.predict(X_test)
```

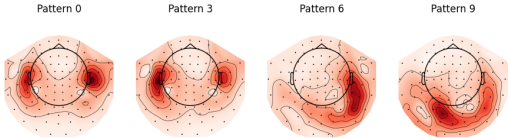

Input data
Nearest Neighbors
Linear SVM
RBF SVM
Gaussian Process
Decision Tree
Random Forest
Neural Net
AdaBoost
Naive Bayes
QDA
.97
.88
.97
.97
.95
.93
.90
.93
.88
.85
.93
.40
.88
.90
.80
.80
.88
.82
.70
.72
.93
.93
.95
.93
.95
.95
.95
.95
.95
.93

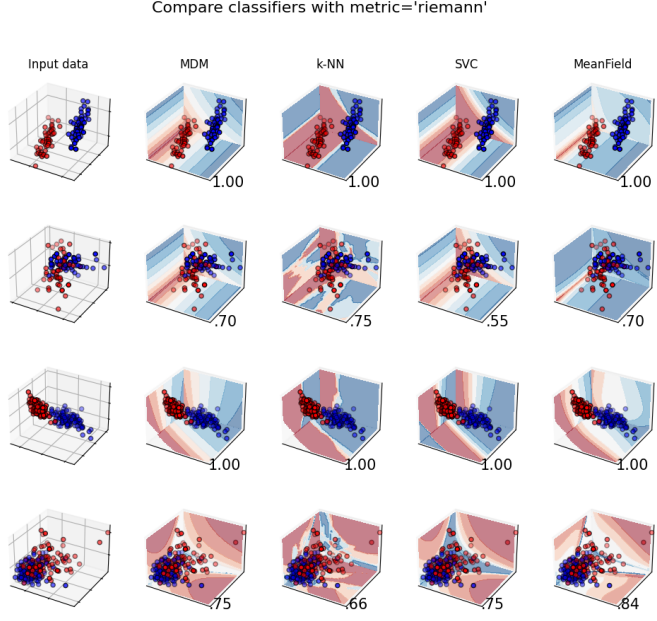
PyRiemann – Riemannian ML for All !

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






Compare classifiers with metric='riemann'
Input data
MDM
k-NN
SVC
MeanField
1.00
1.00
1.00
1.00
.70
.75
.55
.70
1.00
1.00
1.00
1.00
.75
.66
.75
.84

Outline

Welcome and introduction

Part 1 - Tools to build your pipelines

- Benchmarking EEG pipelines in BCI with **MOABB**, by *Pierre Guetschel*
- **Braindecode**, Deep Learning for EEG Decoding, by *Bruno Aristimunha*
- Classification features extraction and selection using **HappyFeat**, by *Arthur Desbois*

Break

Part 2 - Tools to setup your experiment

- **Timeflux** presentation, by *Pierre Clisson*
- c-VEP: an introduction and live demo *P. Clisson*

Discussion panel and concluding remarks

Useful links

Access to the materials, resources

- https://github.com/Inria-NERV/Graz24-DesigningBCITools-Workshop

Links to the tools presented during the workshop

- MOABB: https://neurotechx.github.io/moabb/
- Braindecode: https://braindecode.org/stable/index.html
- HappyFeat: https://happyfeat.readthedocs.io/en/latest/
- Timeflux: https://timeflux.io/