MNE

Open-source Python software for exploring, visualizing, and analyzing human neurophysiological data: MEG, EEG, sEEG, ECoG, and more.

https://martinos.org/mne

```
import numpy as np

pip install mne

from mne.datasets import eegbci
from mne.io import concatenate_raws, read_raw_edf

subject = 1
runs = [6, 10, 14]  # motor imagery: hands vs feet

raw_fnames = eegbci.load_data(subject, runs)

raw = concatenate_raws([read_raw_edf(f, preload=True) for f in raw_fnames])
```

Plot the data using

```
raw.plot(start=..., duration=..., n_channels=..., scalings='auto')
```

```python
# Apply band-pass filter
raw.filter(7., 30., fir_design='firwin', skip_by_annotation='edge')
```

Divide into epochs

```
from mne import Epochs, pick_types, events_from_annotations

events, _ = events_from_annotations(raw, event_id=dict(T1=2, T2=3))
picks = pick_types(raw.info, meg=False, eeg=True, stim=False, eog=False,
                   exclude='bads')
```

have a look to `events` and `picks`

```
event_id = dict(hands=2, feet=3)
tmin, tmax=-1,4

epochs = Epochs(raw, events, event_id, tmin, tmax, proj=True, picks=picks,
                baseline=None, preload=True)
```

Consider only 1 second for each epoch

```
epochs_design = epochs.copy().crop(tmin=1., tmax=2.)
```

Create a new variable `y` (**label**) from `events` (or from `epochs_design.events`)

`y`:

- 0: event T1
- 1: event T2

```
#y =...
```

Get **data** from `epochs_design`, using the method `get_data()`

Have a look to the data, using `shape`

```
#X=...
```

```
X.shape
```

SCIKIT-LEARN

Machine learning in python

https://scikit-learn.org

Split data and labels into random train and test subsets using

`train_test_split` from `sklearn.model_selection`.

Have a look to the data.

```python
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33)

X_test.shape
```

Feature extraction:

Common Spatial Pattern (CSP)

- Zoltan J. Koles. The quantitative extraction and topographic mapping of the abnormal components in the clinical EEG. Electroencephalography and Clinical Neurophysiology, 79(6): 440–447, December 1991.

- https://en.wikipedia.org/wiki/Common_spatial_pattern

```
from mne.decoding import CSP
csp = CSP(n_components=4, reg=None, log=True, norm_trace=False)
```

Use of **CSP**

- 'train' the decoder using the `fit()` method.
- transform the data using the `tranform()` method

have a look to the data

```
# csp.fit(...)
# X_train_csp=...
# X_test_csp=...
```

Create a linear discriminant classifier

```
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
lda = LinearDiscriminantAnalysis()
```

- Train the classifier using the `fit()` method
- Classify the test set using the `predict()`method
- Estimate accuracy

Repeat the process using the knn classifier

```
from sklearn.neighbors import KNeighborsClassifier
knn=KNeighborsClassifier(k)
```