The MNE package for M/EEG data processing

MGH/HST Athinoula A. Martinos Center for Biomedical Imaging

A. Gramfort[1], M. Luessi[2], E. Larson[3], D.A. Engemann[4], D. Strohmeier[5], C. Brodbeck[6], M. Hamalainen[2]

[1]Telecom ParisTech, CNRS LTCI - CEA/Neurospin [2]MGH, Harvard Med. School, USA [3]U. of Washington, USA
[4]INM-3, Juelich Research Center, Germany [5]Ilmenau U., Germany [6]New York U., USA

Features

Preprocessing

- Review raw data, filter, correct ECG / EOG with SSPs, ICA

Forward & inverse modeling

- Automatic BEM with FreeSurfer reconstruction
- MNE - dSPM - sLORETA - LCMV - (TF-)MxNE

Statistics (sensor & source space)

- Time-Frequency (Phase-Lock, Induced Power)
- Parametric and non-parametric stats, with clustering

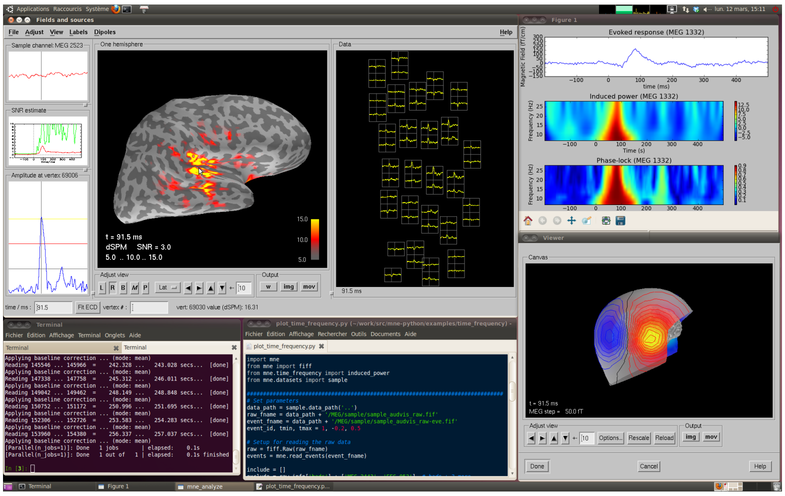
Connectivity (sensor & source space)

- Functional & effective connectivity measures

Project vision & Goals

- State of the art, many examples, documented & tested
- Open development, collaboration between many labs
- Sharing best practices, making reproducible research

C / Unix & Matlab & Python


Applications Raccourcis Système
lun. 12 mars, 15:11
Fields and sources
File Adjust View Labels Dipoles Help
Sample channel: MEG 2523
SNR estimate
Amplitude at vertex 69006
One hemisphere
t = 91.5 ms
dSPM SNR = 3.0
5.0 .. 10.0 .. 15.0
15.0
10.0
5.0
Adjust view
L R B M P Lat
10
Output
w img mov
Data
91.5 ms
time / ms : 91.5
Fit ECD
vertex # :
vert: 69030 value (dSPM): 16.31
Figure 1
Evoked response (MEG 1332)
Magnetic Field (fT/cm)
time (ms)
Induced power (MEG 1332)
Frequency (Hz)
Time (s)
Phase-lock (MEG 1332)
Viewer
Canvas
t = 91.5 ms
MEG step = 50.0 fT
Options... Rescale Reload
img mov
Done Cancel Help
Terminal
Fichier Édition Affichage Terminal Onglets Aide
plot_time_frequency.py (~/work/src/mne-python/examples/time_frequency) -
Fichier Édition Affichage Rechercher Outils Documents Aide
Terminal Figure 1 mne_analyze plot_time_frequency.p...

```
Applying baseline correction ... (mode: mean)
Reading 145546 ... 145966  =    242.328 ...   243.028 secs...  [done]
Applying baseline correction ... (mode: mean)
Reading 147338 ... 147758  =    245.312 ...   246.011 secs...  [done]
Applying baseline correction ... (mode: mean)
Reading 149042 ... 149462  =    248.149 ...   248.848 secs...  [done]
Applying baseline correction ... (mode: mean)
Reading 150752 ... 151172  =    250.996 ...   251.695 secs...  [done]
Applying baseline correction ... (mode: mean)
Reading 152306 ... 152726  =    253.583 ...   254.283 secs...  [done]
Applying baseline correction ... (mode: mean)
Reading 153960 ... 154380  =    256.337 ...   257.037 secs...  [done]
Applying baseline correction ... (mode: mean)
[Parallel(n_jobs=1)]: Done   1 jobs       | elapsed:    0.1s
[Parallel(n_jobs=1)]: Done   1 out of   1 | elapsed:    0.1s finished

In [3]:
```

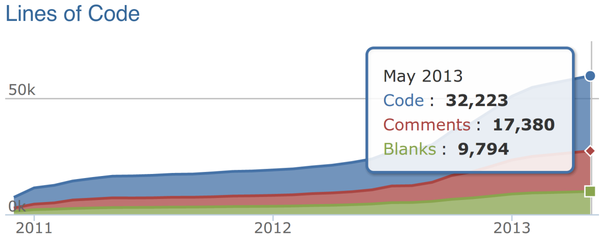
```
import mne
from mne import fiff
from mne.time_frequency import induced_power
from mne.datasets import sample

###############################################################################
# Set parameters
data_path = sample.data_path('..')
raw_fname = data_path + '/MEG/sample/sample_audvis_raw.fif'
event_fname = data_path + '/MEG/sample/sample_audvis_raw-eve.fif'
event_id, tmin, tmax = 1, -0.2, 0.5

# Setup for reading the raw data
raw = fiff.Raw(raw_fname)
events = mne.read_events(event_fname)

include = []
```

http://martinos.org/mne   http://github.com/mne-tools

A. Gramfort, M. Luessi, E. Larson, D. Engemann, D. Strohmeier, C. Brodbeck, L. Parkkonen, M. Hämäläinen
MNE software for processing MEG and EEG data, To appear.

MNE-Python

- Python: general-purpose, high-level language
- Free (can run on a cluster without license problems)
- Permissive BSD license (allows use in commercial products)
- Many third-party packages easily integrated (e.g., ML)
- Open, 29 contributors so far (≈ 8 person-years of effort)

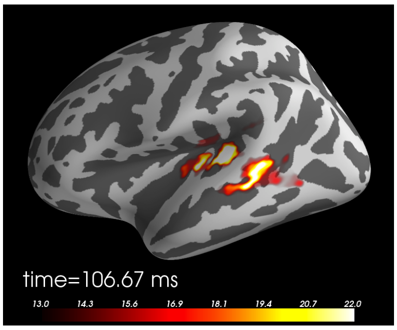


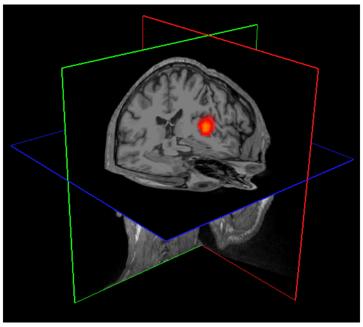
Learn more

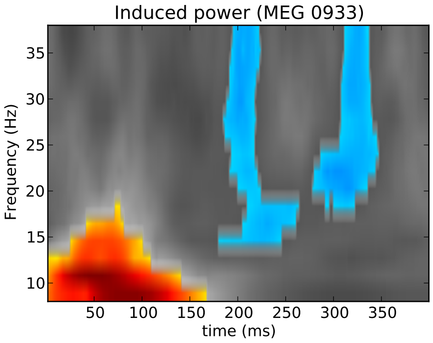
- Mailing list: mne_analysis@nmr.mgh.harvard.edu
- http://martinos.org/mne/ (general doc)
- http://martinos.org/mne/python_tutorial.html
- http://martinos.org/mne/auto_examples/ (> 70 demos)
- http://mne-tools.github.com/mne-python-intro-slides

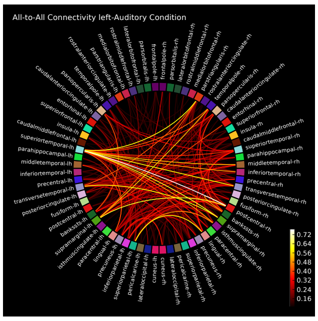
From raw to dSPM in < 30 lines of code

```
import mne

# load data
fname = 'raw.fif'
raw = mne.fiff.Raw(fname)
raw.info['bads'] = ['MEG 2443', 'EEG 053']   # mark bad channels

# band-pass filter data in beta band, and save it
raw.filter(13.0, 30.0, filter_length=4096, n_jobs='cuda')
raw.save(fname[:-4] + '_beta.fif')

# extract epochs
picks = mne.fiff.pick_types(raw.info, meg=True, eeg=True, eog=True,
                            exclude='bads')
events = mne.find_events(raw)
epochs = mne.Epochs(raw, events, event_id=1, tmin=-0.2, tmax=0.5, proj=True,
                    picks=picks, baseline=(None, 0), preload=True,
                    reject=dict(grad=4000e-13, mag=4e-12, eog=150e-6))

# compute evoked response and noise covariance, and plot evoked
evoked = epochs.average()
cov = mne.compute_covariance(epochs, tmax=0)
evoked.plot()

# compute inverse operator
fwd_fname = 'sample_audvis-meg-eeg-oct-6-fwd.fif'
fwd = mne.read_forward_solution(fwd_fname, surf_ori=True)
inv = mne.minimum_norm.make_inverse_operator(raw.info, fwd, cov, loose=0.2)

# compute inverse solution
stc = mne.minimum_norm.apply_inverse(evoked, inv, lambda2=1 / 3.0 ** 2,
                                     method='dSPM')

# morph it to average brain for group study
stc_avg = mne.morph_data('sample', 'fsaverage', stc, 5, smooth=5)
stc_avg.plot()
```









Funded by: NIH P41RR14075, R01EB009048, F32DC012456, NSF 0958669, 1042134