Nilearn Tutorial

Alexandre Abraham and João Loula

Brainhack 2016 Vienna

http://nilearn.github.io

What is Nilearn?

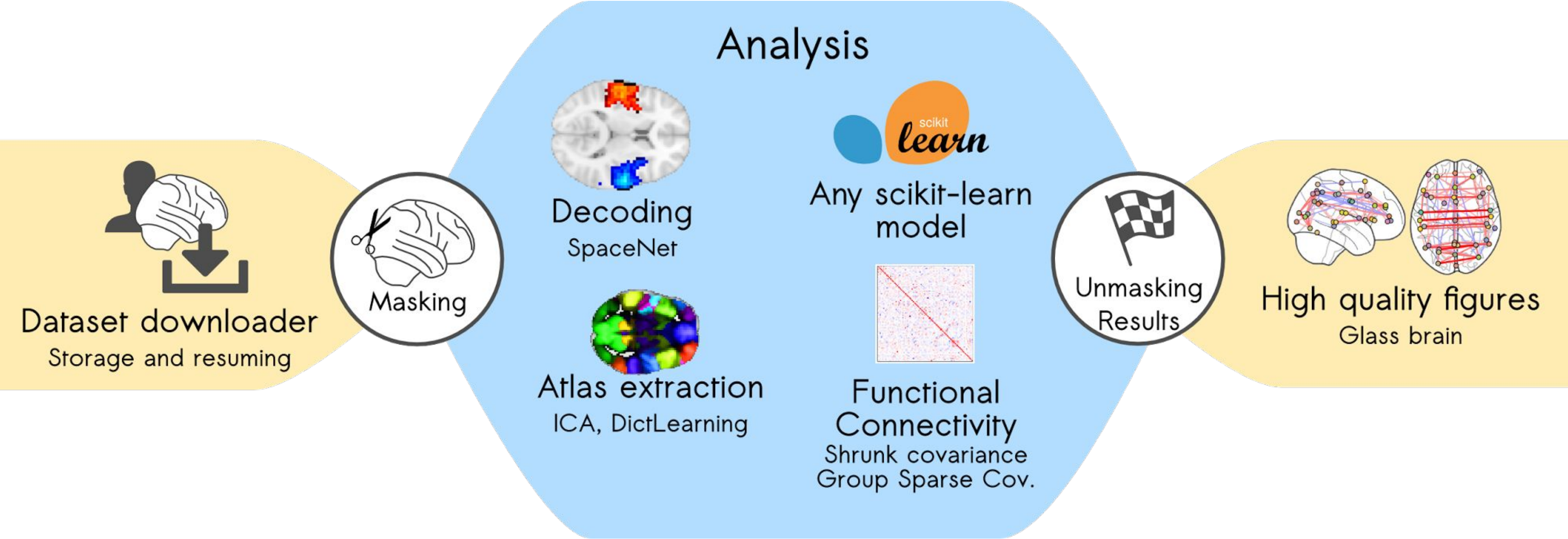
Nilearn is a Python module for **statistical** and **machine learning** analysis on brain data.

It leverages Python's **simplicity** and Scikit-learn's **versatility** into an easy-to-use integrated pipeline.

Better reproducibility: run a whole analysis in a single script!

Free and open source
41 contributors and growing: peer reviewed codebase
Ease collaboration between computer scientists and neuroscientists

A typical Nilearn analysis



Dataset downloader

Storage and resuming

Masking

Analysis

Decoding

SpaceNet

Any scikit-learn model

Atlas extraction

ICA, DictLearning

Functional Connectivity

Shrunk covariance

Group Sparse Cov.

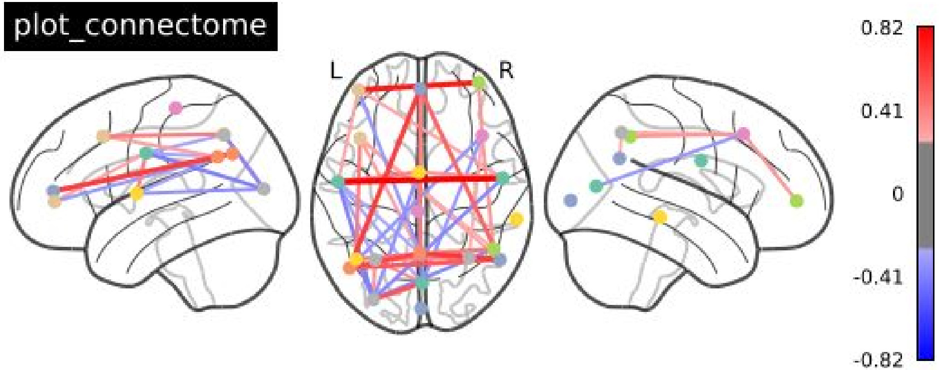
Unmasking Results

High quality figures

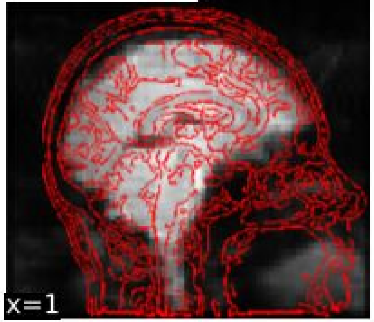
Glass brain

A wide range of plotting capabilities

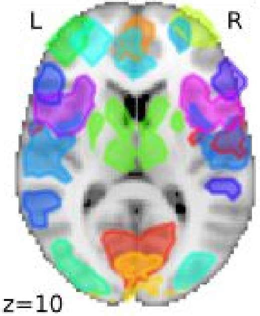
plot_connectome



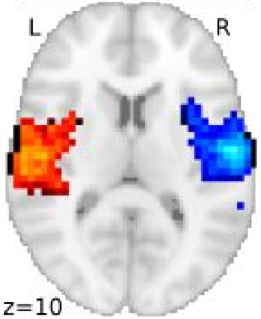
add_edges

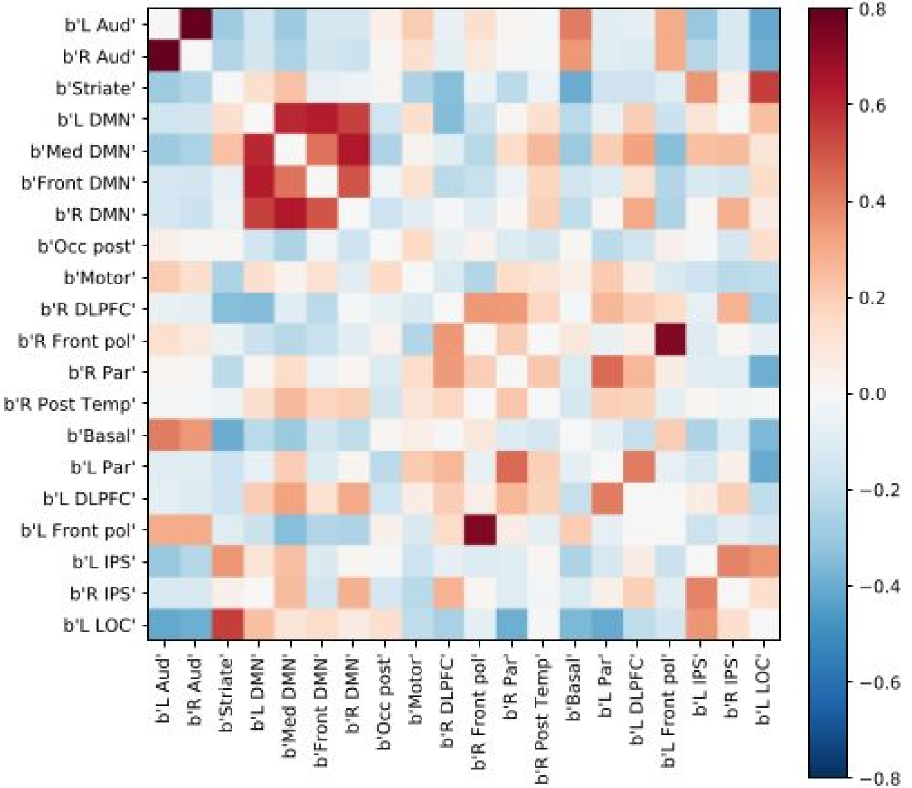


plot_prob_atlas

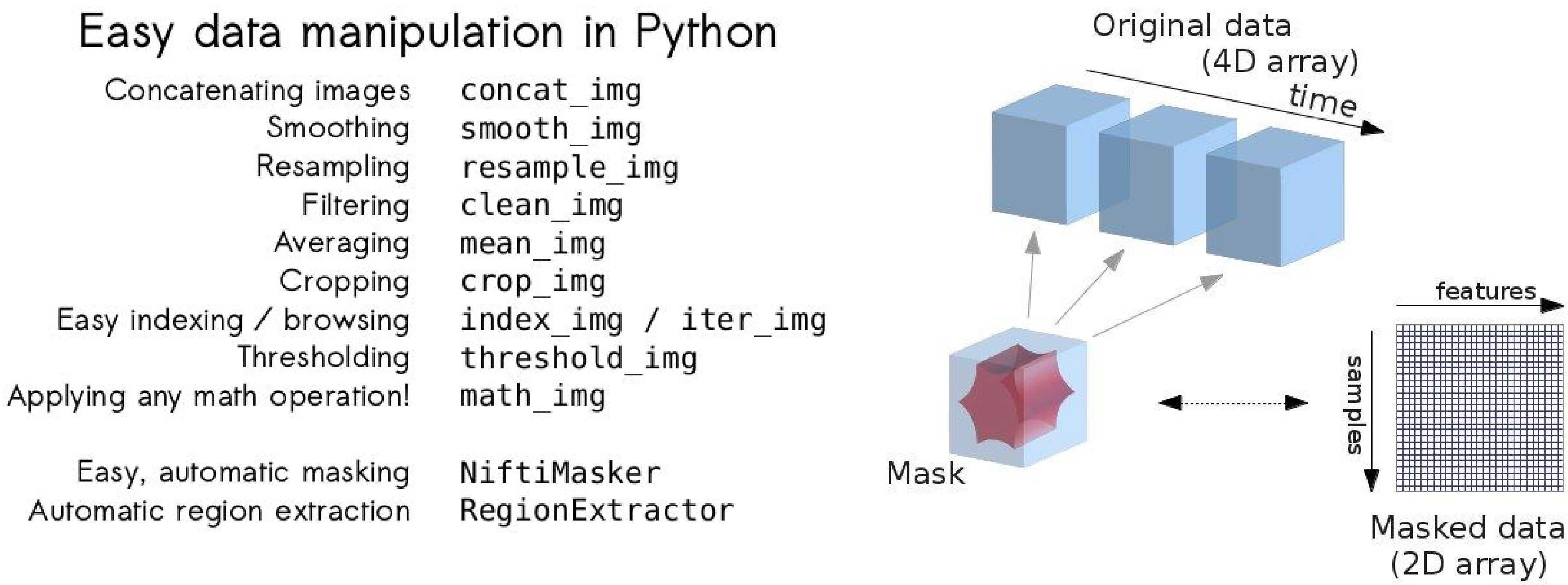


plot_stat_map





Data pre-treatment



Easy data manipulation in Python

Concatenating images	concat_img
Smoothing	smooth_img
Resampling	resample_img
Filtering	clean_img
Averaging	mean_img
Cropping	crop_img
Easy indexing / browsing	index_img / iter_img
Thresholding	threshold_img
Applying any math operation!	math_img
Easy, automatic masking	NiftiMasker
Automatic region extraction	RegionExtractor

Why make the switch?

Scalable and efficient

- Online implementations
- Out-of-core computing
- Parallel computing and efficient caching (joblib powered)

Learn and work by example

- 58 examples covering a wide spectrum of use cases
- Tutorials as interactive Jupyter notebooks
- Your results are just 1 copy/paste away!

High quality code and doc

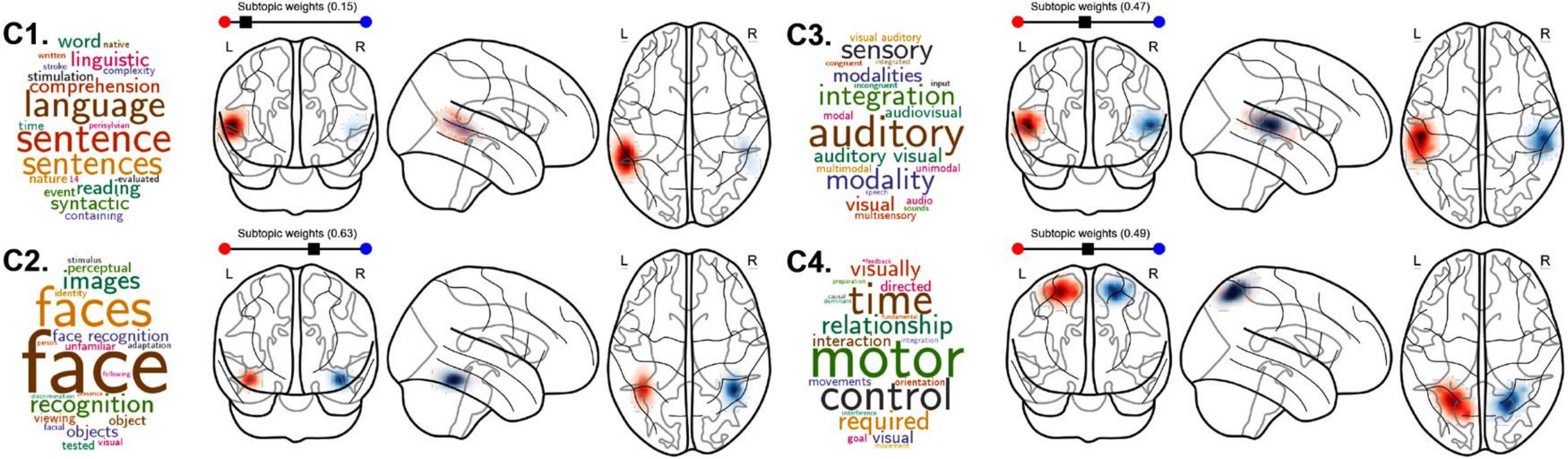
- 90% code coverage
- Commented and readable code
- Algorithms contributed by domain experts
- Extensive documentation

Nilearn is spreading fast

- Part of nipy ecosystem
- Among KDnuggets' top 20 Python Machine Learning Projects

All the cool kids are using it

Ex.:


C1.
word native written linguistic stroke complexity stimulation comprehension language time perisylvian sentence sentences nature 14 evaluated event reading syntactic containing
Subtopic weights (0.15)
L
R
C2.
stimulus perceptual images identity faces face recognition person unfamiliar adaptation face following discrimination presence recognition viewing object facial objects tested visual
Subtopic weights (0.63)
C3.
visual auditory sensory congruent integrated modalities incongruent input integration modal audiovisual auditory auditory visual multimodal unimodal modality speech visual audio sounds multisensory
Subtopic weights (0.47)
C4.
feedback visually preparation directed causal dominant time fundamental relationship interaction integration motor movements orientation control interference required goal visual movement
Subtopic weights (0.49)

[Rubin et al. 2016]

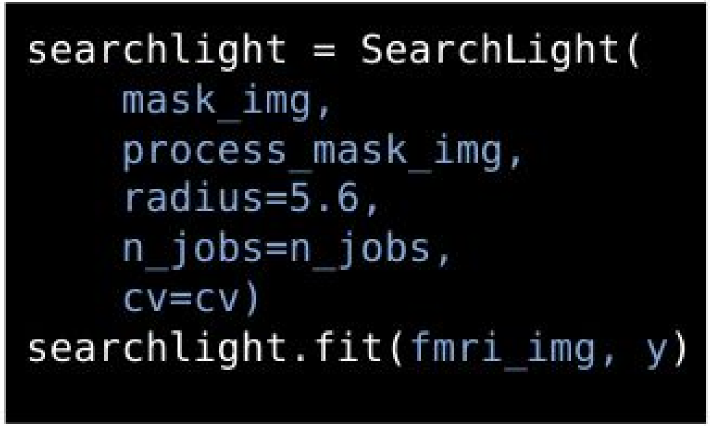
Some use cases

" I'LL SHOW YOU HOW TO TURN THIS SWISS ARMY KNIFE INTO A NUCLEAR BOMB! "

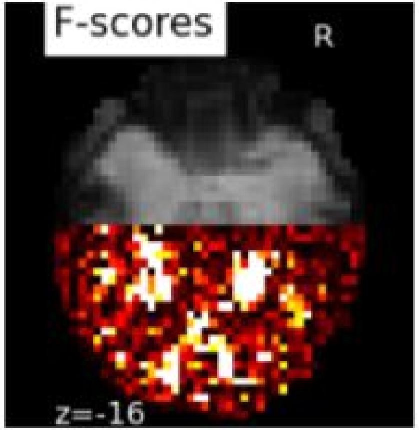
MACGYVER

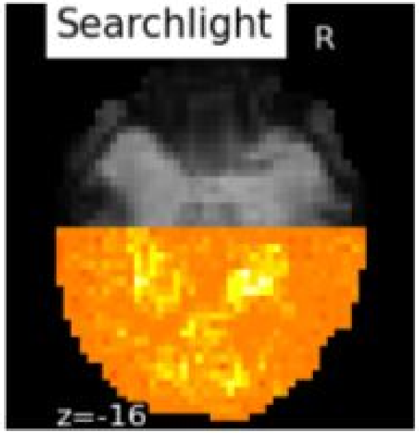
SearchLight:
Multi-Variate Pattern Analysis

Efficient implementation Customizable Only one line of code



```
searchlight = SearchLight(
    mask_img,
    process_mask_img,
    radius=5.6,
    n_jobs=n_jobs,
    cv=cv)
searchlight.fit(fmri_img, y)
```



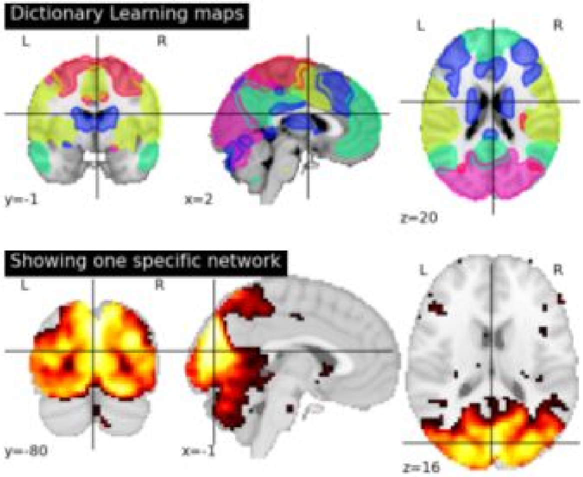


Application: Brain connectivity estimation

ADHD dataset, 20 subjects

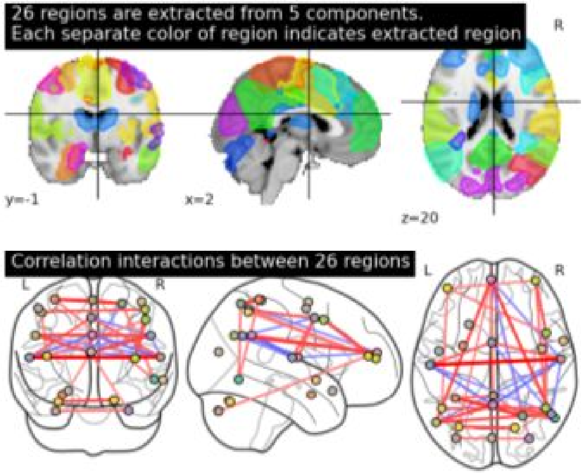
Dictionary Learning

Brain atlas extraction



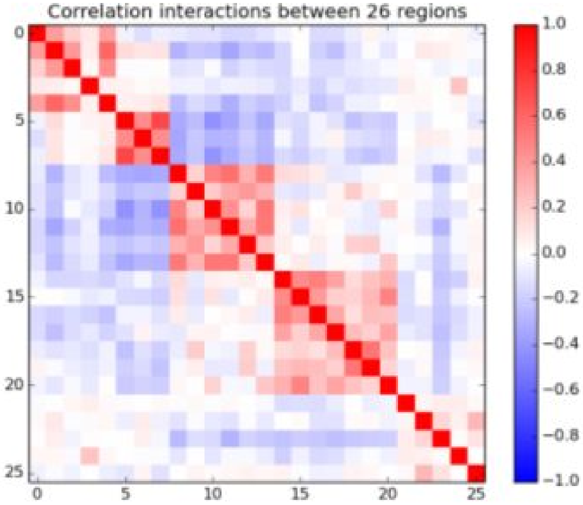
Region extraction

Break down atlases into ROIs


26 regions are extracted from 5 components.
Each separate color of region indicates extracted region
L
R
y=-1
x=2
z=20
Correlation interactions between 26 regions

Connectome estimation

Correlation estimation



SpaceNet decoder, multi-variate priors: Total-Variation + L1 and GraphNet

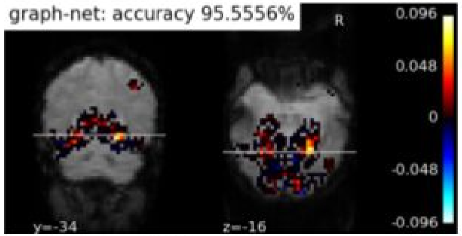
Classification: Brain stimuli when showing face vs. houses

Haxby dataset
Download 1 subject

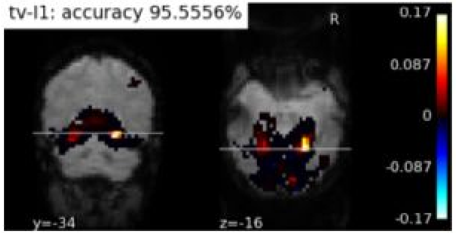
Sample selection
Face vs. house

SpaceNet Classifier
Automatic masking
Multiple penalties

Result extraction
Prediction accuracy
Model coefficients



108 seconds



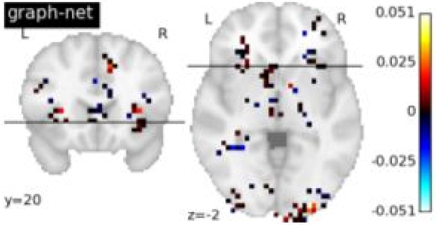
237 seconds

Classification: Brain stimuli when showing face vs. houses

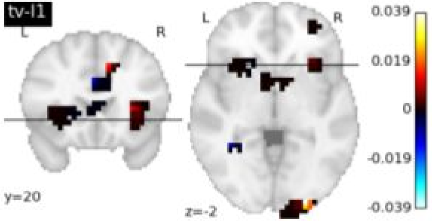
Poldrack mixed-gamble
Download 16 z-maps and
brain mask

SpaceNet Regressor
Automatic masking
Multiple penalties

Result extraction
Model coefficients



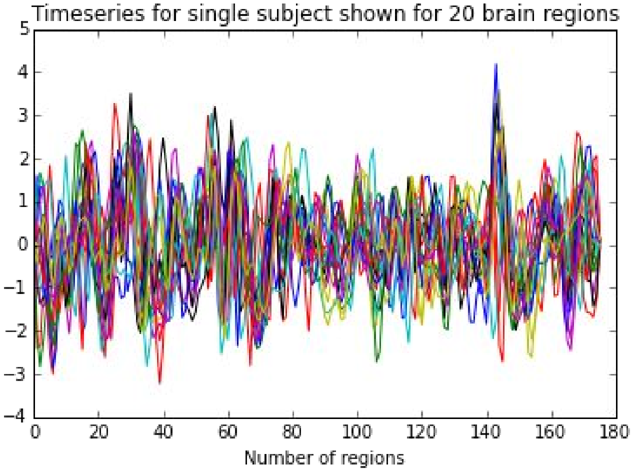
157 seconds

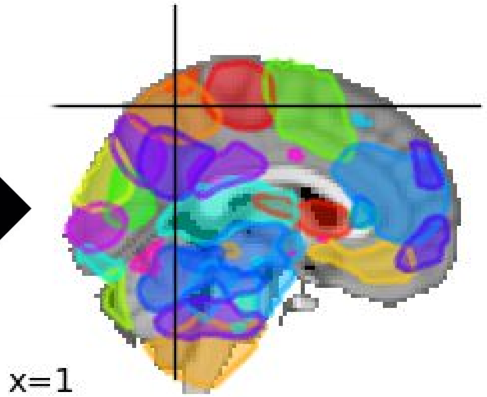


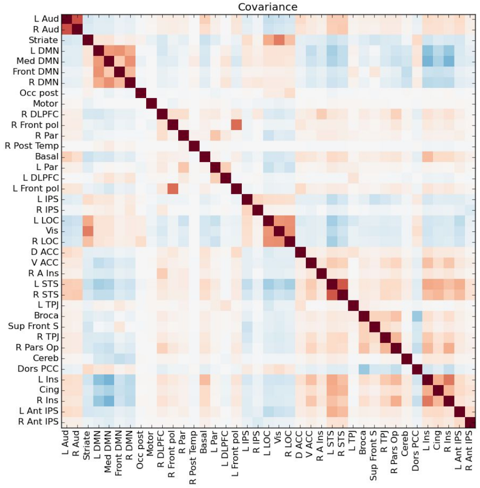
426 seconds

Interactive tutorial

Creating an atlas using the ABIDE resting-state time-series and then building a functional connectome







Thanks!

For installing or seeing more tutorials, go to nilearn.github.io