

---

Computational  
Neuroscience  
Laboratory

# Machine Learning for Neuroimaging

Autumn 2023

10/18/2023

Hands on Statistics & ML (Python)

# Intro into Jupyter Notebook

- Jupyter Notebook is an open-source web application
- Create and share documents that contain live code, equations, visualizations, and text
- The name, **Jupyter**, comes from the core supported programming languages: Julia, Python, and R



# Getting Started with Jupyter

- Install Jupyter using pip (or Anaconda)

Shell

```
$ pip install jupyter
```

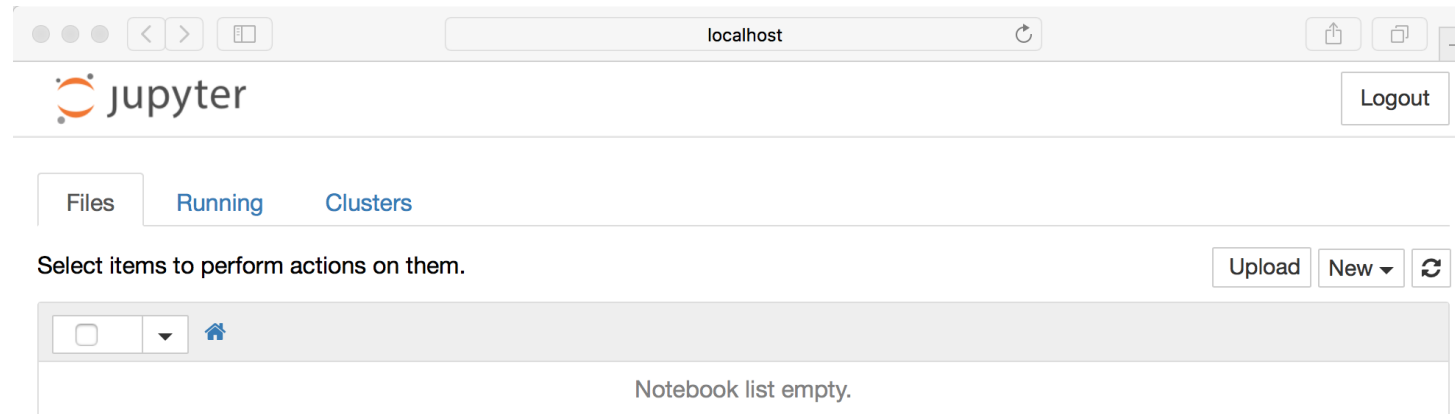
- Run Jupyter

Shell

```
$ jupyter notebook
```

# Create New Notebook

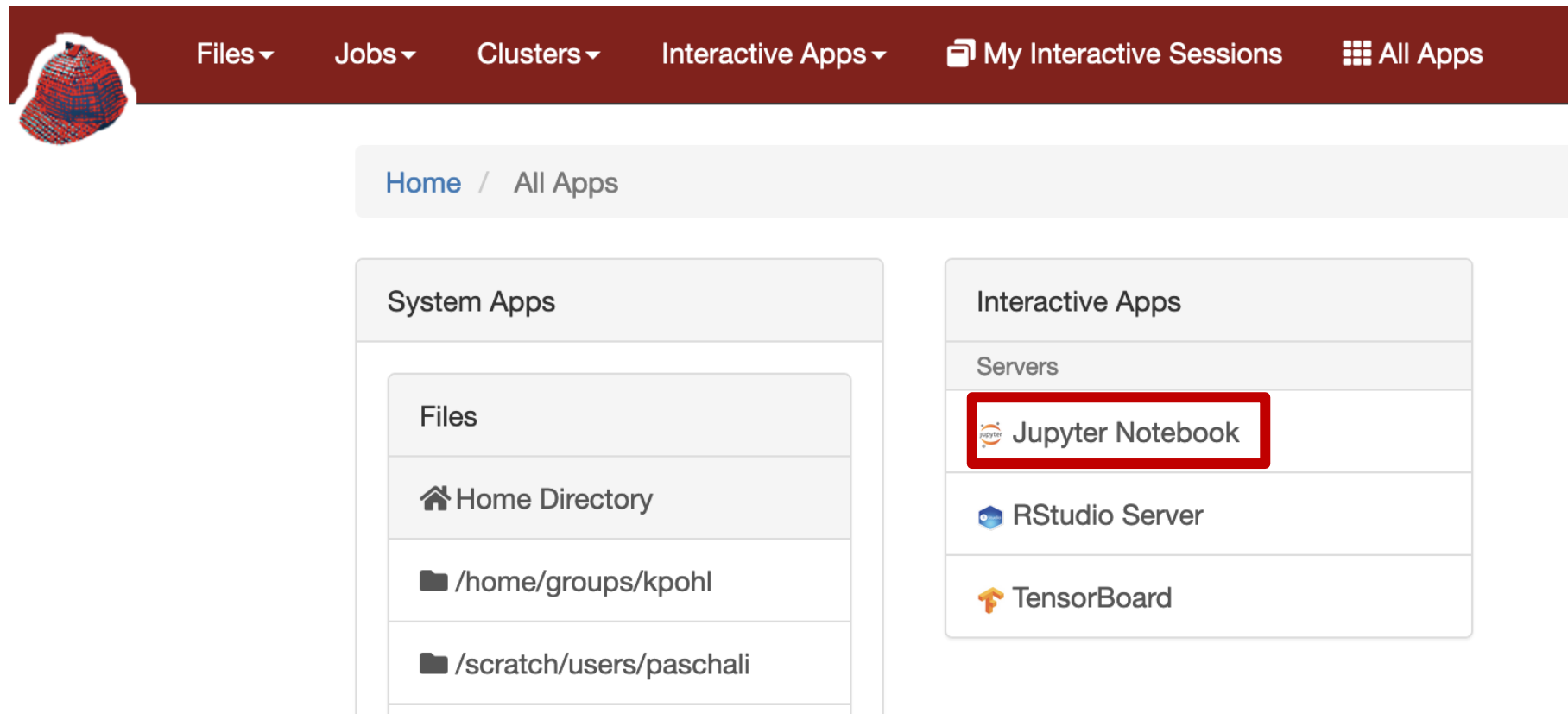
- Jupyter will run on your default browser (or open a new tab) to the following URL: <http://localhost:8888/tree>



- Click 'New' to create a new Jupyter Notebook

# Jupyter on Sherlock

<https://login.sherlock.stanford.edu/pun/sys/dashboard>



The screenshot displays the Sherlock dashboard interface. At the top, a dark red navigation bar contains a Sherlock logo (a red and blue wireframe helmet) and several menu items: 'Files', 'Jobs', 'Clusters', 'Interactive Apps', 'My Interactive Sessions', and 'All Apps'. Below this bar, a light gray breadcrumb trail shows 'Home / All Apps'. The main content area is divided into two columns. The left column, titled 'System Apps', contains a 'Files' section with links to 'Home Directory', '/home/groups/kpohl', and '/scratch/users/paschali'. The right column, titled 'Interactive Apps', contains a 'Servers' section with three options: 'Jupyter Notebook' (which is highlighted with a red rectangular border), 'RStudio Server', and 'TensorBoard'.

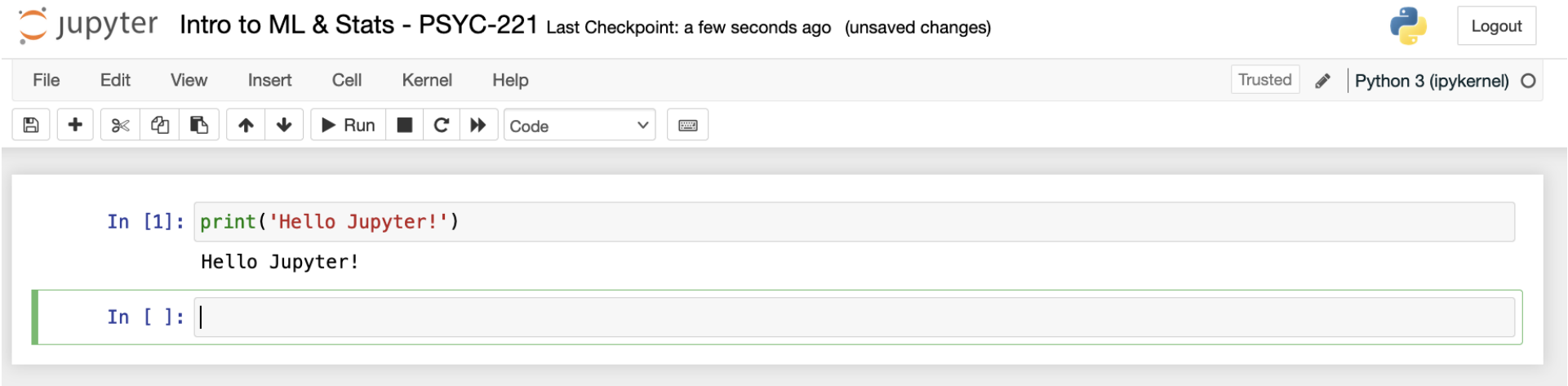
# Running your first cell

- Add Python code to the cell and try running its contents

Jupyter Python Cell

```
print('Hello Jupyter!')
```

⇧ Shift + Enter ↵

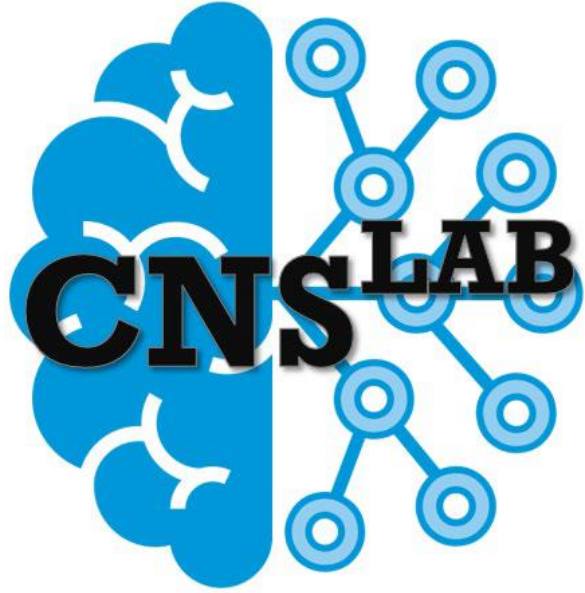


The screenshot shows the JupyterLab web interface. At the top, the title bar reads "jupyter Intro to ML & Stats - PSYC-221" with a status indicator "Last Checkpoint: a few seconds ago (unsaved changes)" and a "Logout" button. Below the title bar is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". To the right of the menu bar are "Trusted" and "Python 3 (ipykernel)" indicators. Below the menu bar is a toolbar with icons for saving, adding, deleting, and running cells. The main area contains a code cell with the prompt "In [1]:" followed by the code `print('Hello Jupyter!')`. The output of the cell is "Hello Jupyter!". Below the first cell is a new, empty code cell with the prompt "In [ ]:" and a cursor.

# Requirements

Install in a virtual environment or load modules in Sherlock

- `numpy==1.24.4`
- `pandas==2.0.2`
- `scikit-image==0.22.0`
- `scikit-learn==1.2.2`
- `scipy==1.11.0`
- `seaborn==0.12.2`
- `statsmodels==0.14.0`
- `nibabel==5.1.0`
- `nilearn==0.10.1`
- `torch==2.0.1` (PyTorch)
- Executed (after installing [Anaconda](#))
  - `conda create --name tut3 python=3.11`
  - `conda activate tut3`
  - `pip install -r requirements.txt`
  - `pip install torch --index-url https://download.pytorch.org/whl/cu118` (taken from [here](#))
- I also include a [.yaml file](#)!
- *Make sure to activate the environment (`conda activate tut3`) before calling `jupyter notebook`!*



---

Computational  
Neuroscience  
Laboratory

# Part 1: Data Preparation and Hypothesis Testing





# Outline

1. Load the data
2. Handle missing values and remove outliers
3. Test hypothesis on preconceptions we may have
4. Residualization: dealing with co-founder effects
5. Save processed data frame

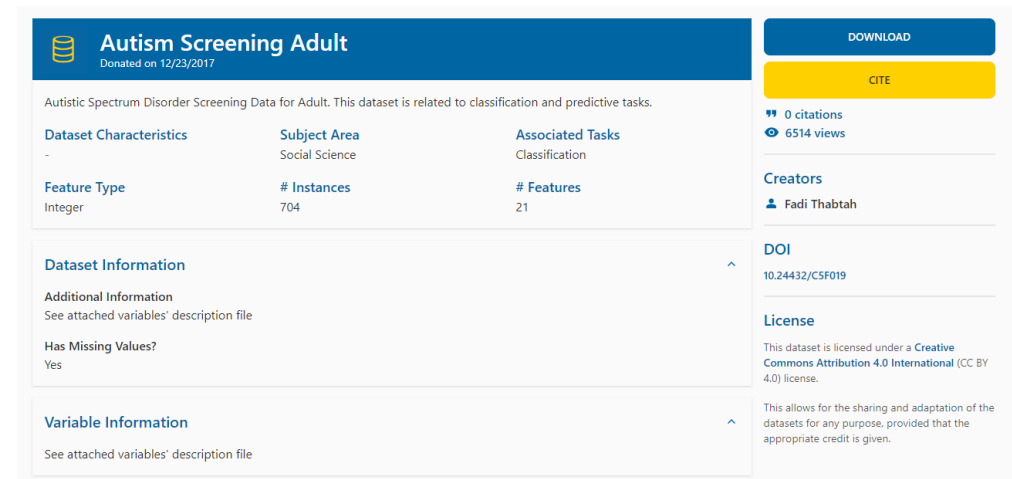
# Autism Screening on Adults

- Autism spectrum disorder (ASD) is a developmental disability
- Diagnosing ASD can be difficult because there is no test, e.g. blood test, to diagnose the disorder
- Many adults can and do have ASD. But it's only recently that doctors have begun diagnosing them with it

<https://www.cdc.gov/ncbddd/autism/screening.html>

# Autism Screening on Adults - Dataset

- Dataset contains survey results for more than 700 subjects
- Results from a quick referral guide for adults with suspected autism
- Download dataset from [here](#)



The screenshot shows the dataset page for 'Autism Screening Adult'. The header is blue with the dataset name and a 'DONATED' status. Below the header, there's a description of the dataset. A table provides key characteristics: Subject Area (Social Science), Associated Tasks (Classification), Feature Type (Integer), # Instances (704), and # Features (21). The right sidebar contains buttons for 'DOWNLOAD' and 'CITE', along with statistics like '0 citations' and '6514 views'. It also lists the creator 'Fadi Thabtah', the DOI '10.24432/CSF019', and the license 'Creative Commons Attribution 4.0 International (CC BY 4.0)'.

Autism Screening Adult		
Donated on 12/23/2017		
Autistic Spectrum Disorder Screening Data for Adult. This dataset is related to classification and predictive tasks.		
<b>Dataset Characteristics</b>	<b>Subject Area</b>	<b>Associated Tasks</b>
-	Social Science	Classification
<b>Feature Type</b>	<b># Instances</b>	<b># Features</b>
Integer	704	21

**Dataset Information**

**Additional Information**  
See attached variables' description file

**Has Missing Values?**  
Yes

**Variable Information**  
See attached variables' description file

**Download**

**Cite**

0 citations  
6514 views

**Creators**  
Fadi Thabtah

**DOI**  
10.24432/CSF019

**License**  
This dataset is licensed under a [Creative Commons Attribution 4.0 International \(CC BY 4.0\)](#) license.  
This allows for the sharing and adaptation of the datasets for any purpose, provided that the appropriate credit is given.

- 1) Thabtah, F. (2017). Autism Spectrum Disorder Screening: Machine Learning Adaptation and DSM-5 Fulfillment. Proceedings of the 1st International Conference on Medical and Health Informatics 2017, pp.1-6. Taichung City, Taiwan, ACM.
- 2) Thabtah, F. (2017). ASDTests. A mobile app for ASD screening. [www.asdtests.com](http://www.asdtests.com) [accessed December 20th, 2017].
- 3) Thabtah, F. (2017). Machine Learning in Autistic Spectrum Disorder Behavioural Research: A Review. To Appear in Informatics for Health and Social Care Journal. December, 2017 (in press)

# Data Loading

- Load csv file
- Visualize the top five rows (*head*)
- Notice data types, column values, empty cells

	A1_Score	A2_Score	A3_Score	A4_Score	A5_Score	A6_Score	A7_Score	A8_Score	A9_Score	A10_Score	...	gender	ethnicity	jundice
0	1	1	1	1	0	0	1	1	0	0	...	f	White-European	no
1	1	1	0	1	0	0	0	1	0	1	...	m	Latino	no
2	1	1	0	1	1	0	1	1	1	1	...	m	Latino	yes
3	1	1	0	1	0	0	1	1	0	1	...	f	White-European	no
4	1	0	0	0	0	0	0	1	0	0	...	f	?	no

# Data Cleaning

- Remove row with outlier age
- Look for missing values
  - Empty cells
  - Cells with '?'
- Count amount of missing values
  - 2 missing age values
  - 95 missing ethnicities
  - 95 missing relations

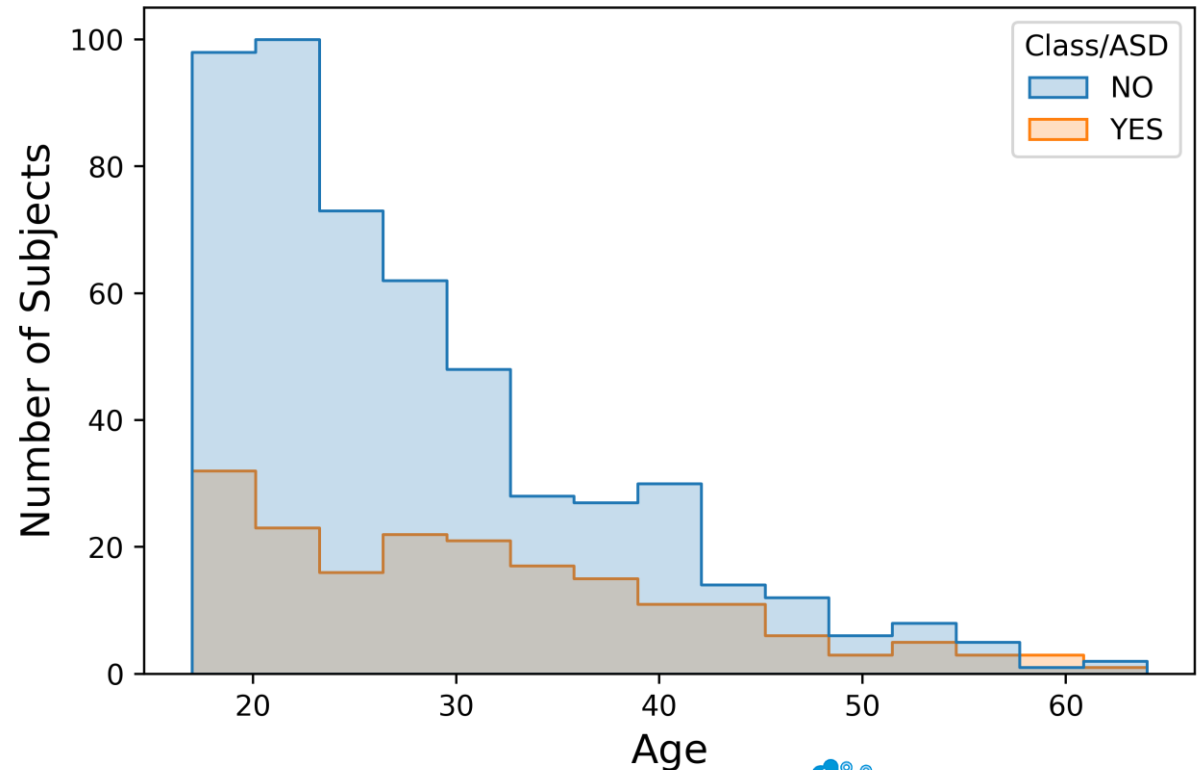
# Data Imputation

- Fill out missing age values with column mean
  - **Important!** This step comes after we have removed the outliers
- Replace missing ethnicities and relations with 'Other'
  - We do not know the actual values

# Hypothesis Testing

- Dataset is split into two cohorts: controls and positive for autism (ASD)
- Null hypothesis: the **age** distribution of the two cohorts is equal

- Histograms of ages of two cohorts
- Hard to visually compare



# Compare age of cohorts

- Mann-Whitney U-Test
- Tests whether the distributions of two independent samples are equal
- Nonparametric test
- $p < 0.05 \rightarrow$  We reject the null hypothesis

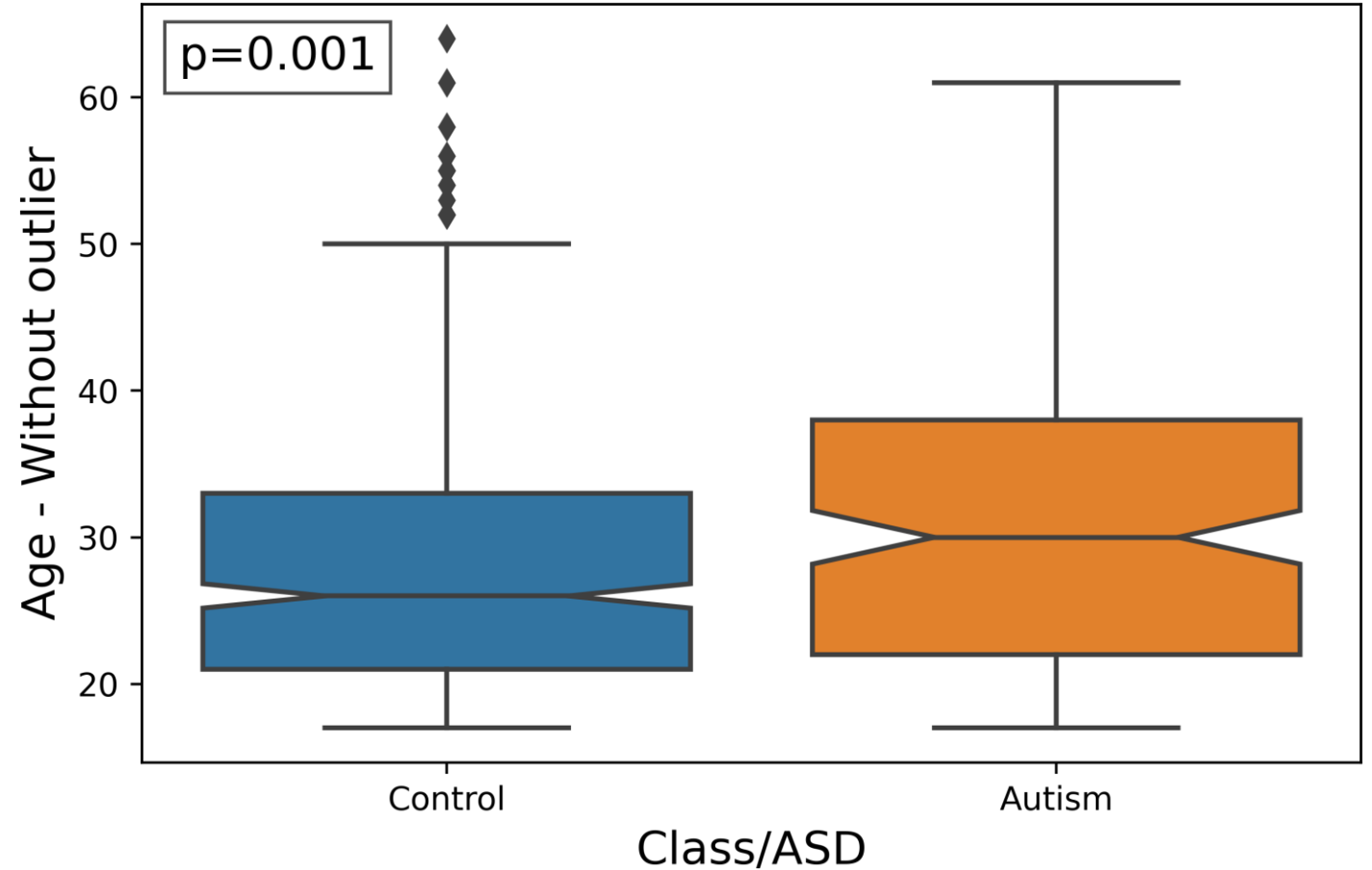
Jupyter Python Cell

```
U1, p = mannwhitneyu(controls["age"], autism["age"])
```



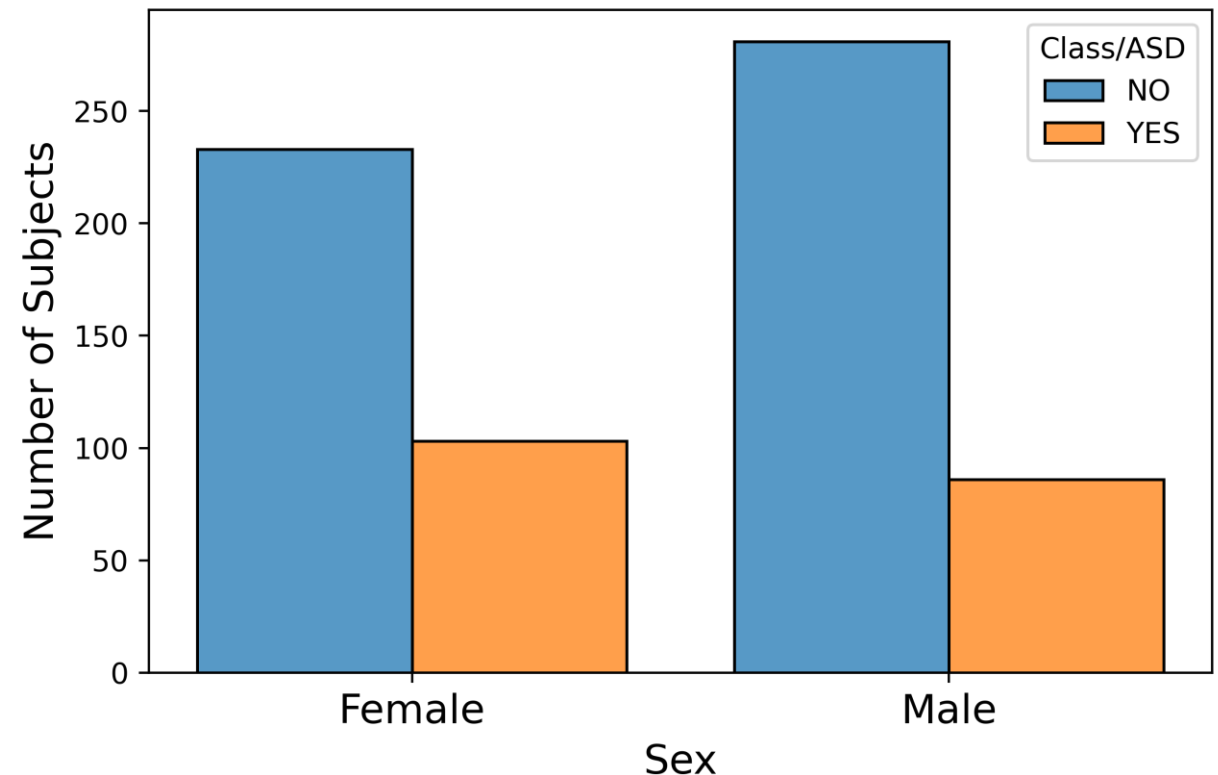
# Compare age of cohorts

- The notches in the box plot do not overlap
- We can conclude, with 95% confidence that the true medians differ



# Hypothesis Testing

- Dataset is split into two cohorts: controls and positive for autism (ASD)
- Null hypothesis: the frequency of autism is the same across sexes
- Histograms of subjects of the two cohorts
- Is there a difference between Females and Males?



# Compare autism across sexes

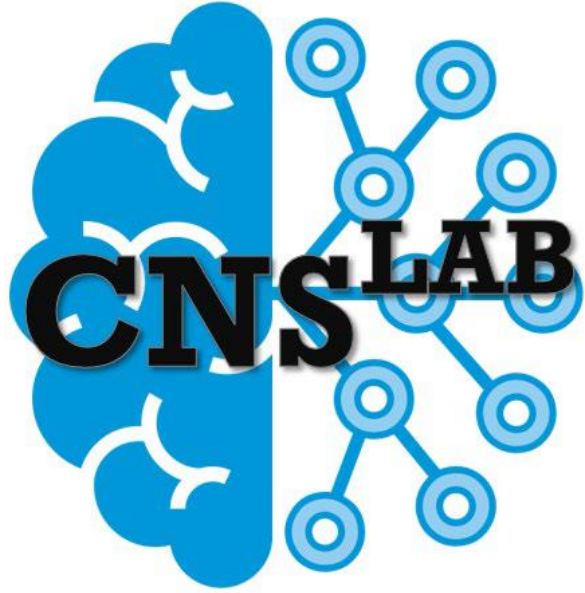
- Null hypothesis: the frequency of autism is the same across sexes
- Chi-squared test
- $p < 0.05 \rightarrow$  We reject the null hypothesis

Contingency Table

	<b>Control</b>	<b>ASD</b>	<b>Total</b>
Male	280	86	366
Female	232	103	335
Total	512	189	701

# Removing the effect of age and sex

- Remove the effect of age and sex
  - Subtract  $\beta_{age}age$  and  $\beta_{sex}sex$  from raw measurements
  - Adjust 'reference point' to  $\overline{age}$  and  $\overline{sex}$



---

Computational  
Neuroscience  
Laboratory

## Part 2: Classifier Training and Evaluation with PyTorch



# Outline

1. Adapt data to serve as PyTorch input
2. Design a PyTorch architecture
3. Train and evaluate models
4. Save and restore models
5. Compare two models

# Autism Classification

- Train a model  $f$  that given an input  $x$  predicts a classification label  $y$
- In this problem  $x$  denotes the input features of the questionnaire
- $y$  is the subject classification to control and ASD based on their responses
- We will compare two ML classifiers

# Data Preparation

- Use the questionnaire answers A1 to A10 **after residualization** as input  $x$  to our classifier
- Transform the 'YES/NO' labels into integers

Country	Category #
Control	NO
Autism	YES

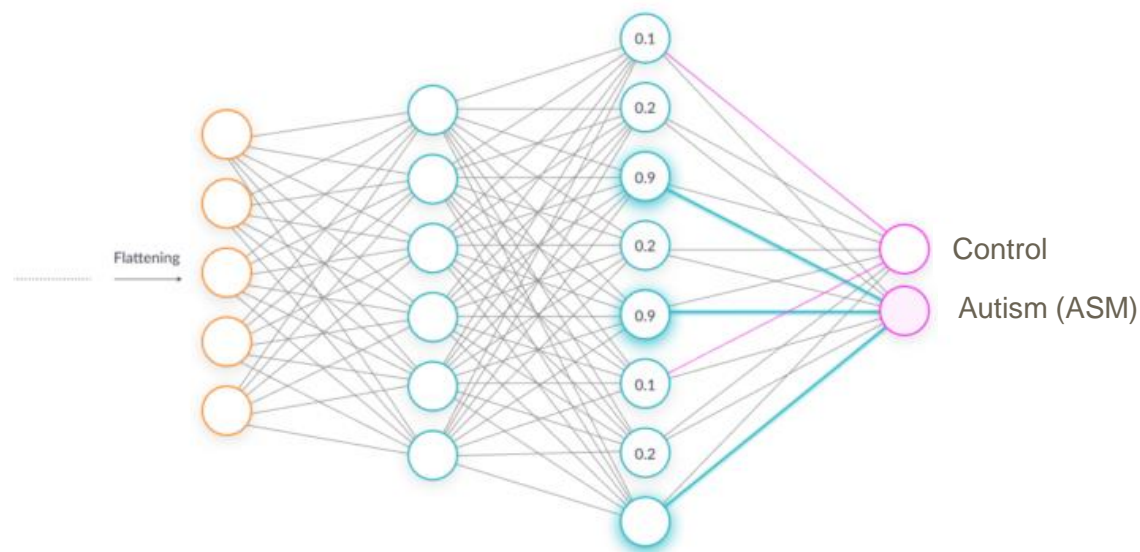


Class
0
1



# Fully-connected Deep Neural Network

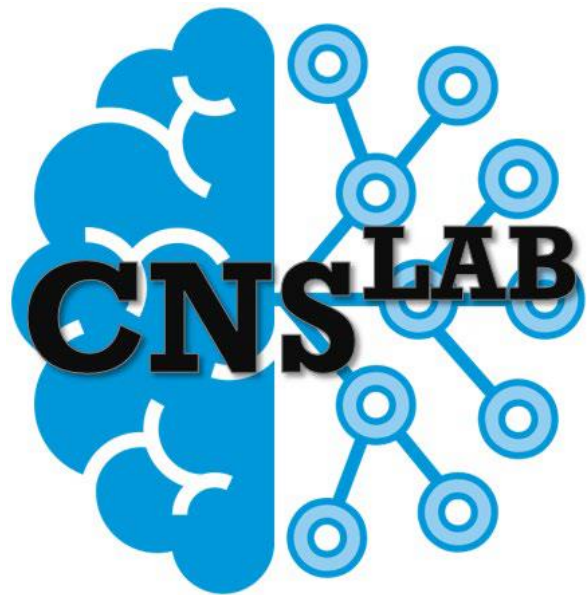
- Fully connected feedforward artificial neural network
- Consists of an input layer, hidden layers and an output layer



<https://towardsdatascience.com/a-simple-guide-to-convolutional-neural-networks-751789e7bd88>

# Classifier Comparison

- Null hypothesis: the two models are equally accurate on the same cohort
- McNemar's Test
- $p < 0.05 \rightarrow$  We reject the null hypothesis



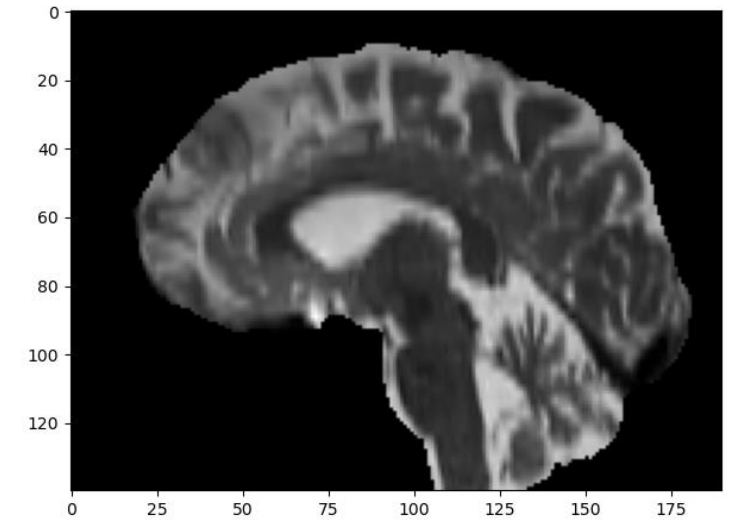
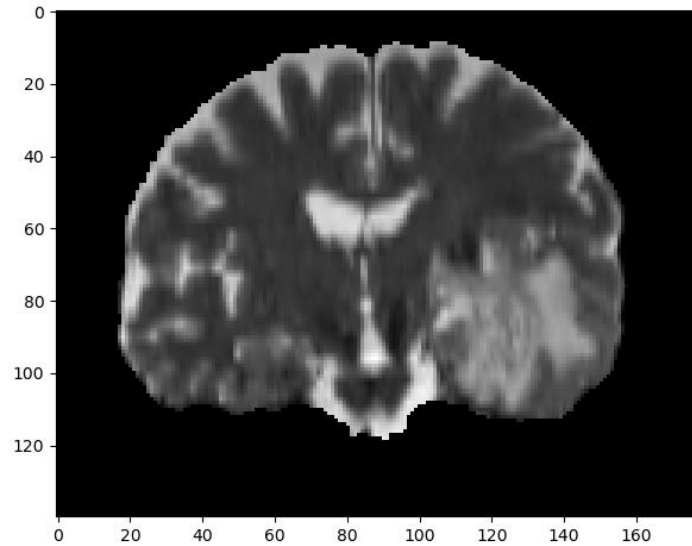
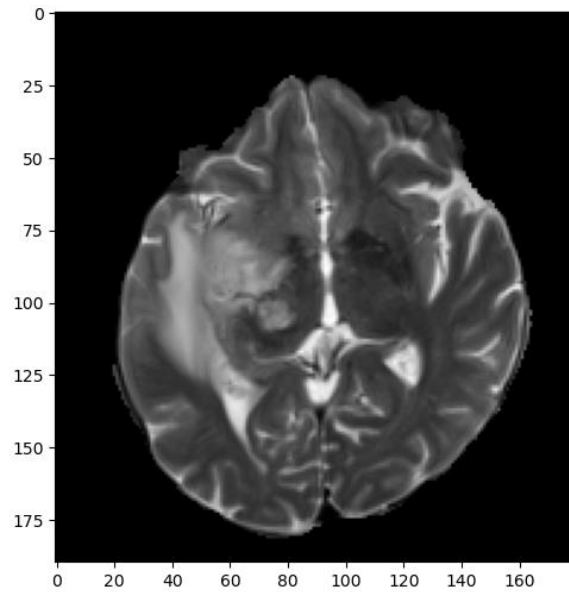
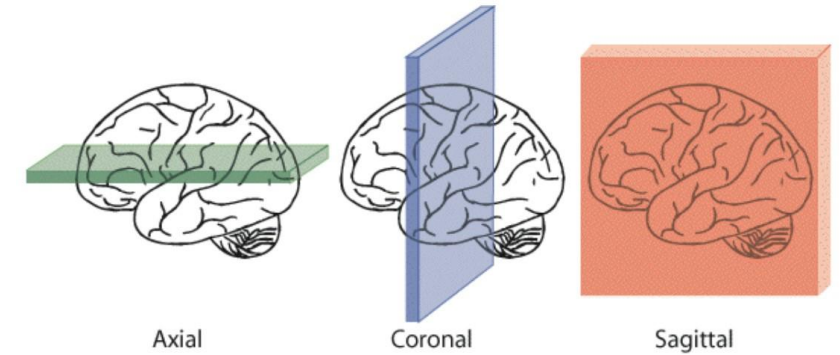
---

Computational  
Neuroscience  
Laboratory

## Part 3: Visualization of MRI Data



# Visualization of an MRI



Goschke, T.: VL Kognitionspsychologie: Denken, Problemlösen, Sprache - Methoden der Kognitiven Neurowissenschaft: Kurze Einführung in die funktionelle Bildgebung (Summer 2013). Technical University Dresden, Dresden (2013)

# Libraries for Neuroimaging Files

## Loading, Processing and Visualization



NiBabel

Access a cacophony of neuro-imaging file formats

[Community](#) | [NiBabel Home](#) | [Mailing list](#) | [License](#) | [Neuroimaging in Python](#)

### NiBabel

Read and write access to common neuroimaging file formats, including: [ANALYZE](#) (plain, SPM99, SPM2 and later), [GIFTI](#), [NIFTI1](#), [NIFTI2](#), [CIFTI-2](#), [MINC1](#), [MINC2](#), [AFNI BRIK/HEAD](#), [ECAT](#) and Philips PAR/REC. In addition, NiBabel also supports [FreeSurfer's MGH](#), geometry, annotation and morphometry files, and provides some limited support for [DICOM](#).

NiBabel's API gives full or selective access to header information (metadata), and image data is made available via Numpy arrays. For more information, see NiBabel's [documentation site](#) and [API reference](#).

### Installation

To install NiBabel's [current release](#) with `pip`, run:

```
pip install nibabel
```

To install the latest development version, run:

```
pip install git+https://github.com/nipy/nibabel
```

When working on NiBabel itself, it may be useful to install in "editable" mode:

```
git clone https://github.com/nipy/nibabel.git
pip install -e ./nibabel
```

For more information on previous releases, see the [release archive](#) or [development changelog](#).

### Mailing List

Please send any questions or suggestions to the [neuroimaging mailing list](#).

### License


NiBabel is licensed under the terms of the [MIT license](#). Some code included with NiBabel is licensed under the [BSD license](#). For more information, please see the [COPYING](#) file.

## NiBabel

- Read and write access to common neuroimaging file formats; access header and image data

# Libraries for Neuroimaging Files

## Loading, Processing and Visualization



Nilearn

Search

- Quickstart
- Examples
- User guide
- API References
- Glossary

DEVELOPMENT

- Contributing
- Maintenance
- What's new
- Team
- GitHub Repository

### Nilearn

Nilearn enables **approachable and versatile analyses of brain volumes**. It provides statistical and machine-learning tools, with **instructive documentation & open community**.

It supports general linear model (GLM) based analysis and leverages the [scikit-learn](#) Python toolbox for multivariate statistics with applications such as predictive modelling, classification, decoding, or connectivity analysis.

#### Quickstart

Get started with Nilearn

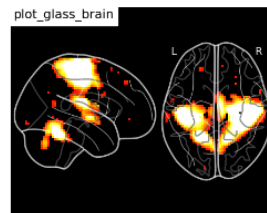
#### Examples

Discover functionalities by reading examples

#### User guide

Learn about neuroimaging analysis

### Featured examples



#### Glass brain plotting

Explore how to retrieve data and plot whole brain cuts in glass mode.



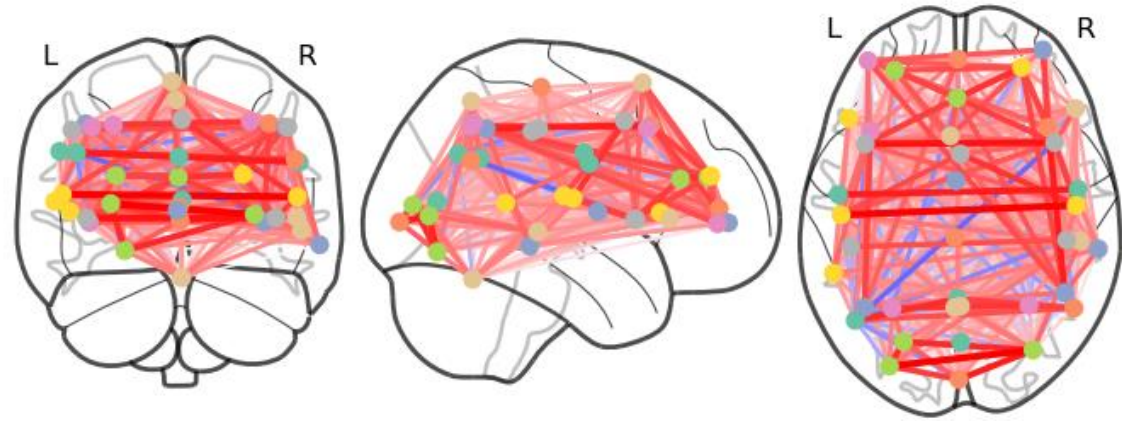
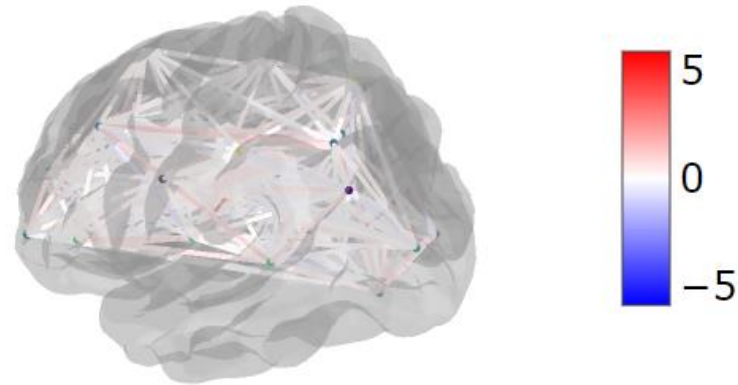
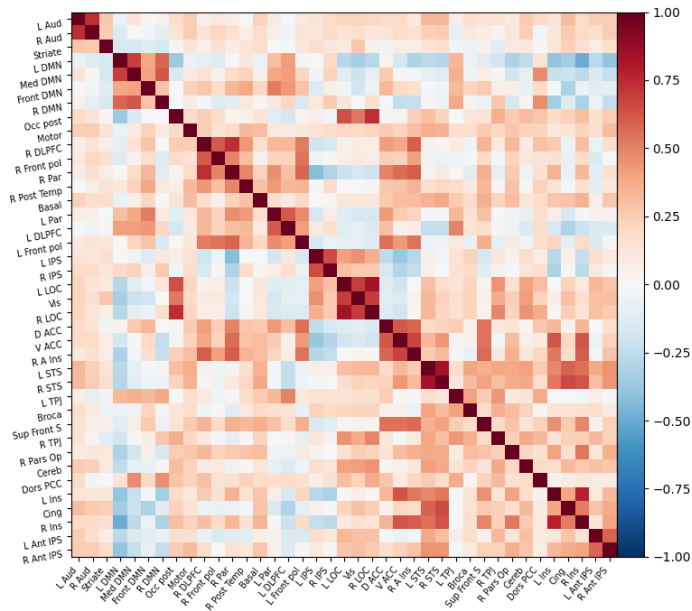
#### Computing a connectome with sparse inverse covariance

Construct a functional connectome using the sparse inverse covariance, and display the corresponding graph and matrix.

## Nilearn

- Glass and surface plotting
- Very rich gallery of examples!

# Visualizations of an fMRI





# Libraries for Neuroimaging Files

## Loading, Processing and Visualization



**Nipype:**  
**Neuroimaging in Python**  
**Pipelines and Interfaces**

[Home](#) · [Quickstart](#) · [User Guide and Examples](#) · [Interfaces Index](#) · [Developers](#) · [About](#) · [Nipy](#)

### User Guide and Examples

Michael Notter's User Guide

Be sure to read [Michael's excellent tutorials](#).

### Examples

The following examples are [literate programming](#) documents which solve specific problems using Nipype. In some cases, the same problem is solved with multiple underlying tools.

- [dMRI: Camino, DTI](#)
- [dMRI: Connectivity – Camino, CMTK, FreeSurfer](#)
- [dMRI: Connectivity – MRtrix, CMTK, FreeSurfer](#)
- [dMRI: DTI – Diffusion Toolkit, FSL](#)
- [dMRI: HARDI – Diffusion Toolkit, FSL](#)
- [dMRI: DTI, FSL](#)
- [dMRI: Group connectivity – Camino, FSL, FreeSurfer](#)
- [dMRI: Group connectivity – MRtrix, FSL, FreeSurfer](#)
- [dMRI: DTI – MRtrix, FSL](#)
- [dMRI: Preprocessing](#)
- [dMRI: TBSS on NKI RS data](#)

## Nipype

- Load, preprocess and analyze diffusion, structural and functional MRIs