Today, we’re gonna look at functional connectivity in an fMRI dataset.

We’ll need the following modules

*scipy, numpy, pandas, networkx, pyplot*

Set the default graphics style with:

pyplot.style.use(‘default’)

Also, import the following sub-modules from nilearn:

*connectome, maskers, plotting, datasets*

1. Start by downloading the ADHD fMRI dataset with

data = datasets.fetch\_adhd(40)

for all 40 subjects.

The data is about 1.75 GB, so it will take a moment to download the first time. After that, this command should load the data from your computer.

Locate the folder with dataset on your computer.

Then load the subject metadata from

“ADHD200\_40subs\_motion\_parameters\_and\_phenotypics.csv”, for example into a pandas dataframe. From the column “adhd”, get the indices for patients and controls.

Load the MSDL atlas (39 parcels) into a variable using:

atlas = datasets.fetch\_atlas\_msdl()

Get the atlas’ parcel labels with:

labels = atlas.labels[1:]

and the parcels’ coordinates with:

coords = atlas.region\_coords

In nilearn, you can map the fMRI voxels to an atlas with “masks”:

masker = NiftiMapsMasker( maps\_img=atlas.maps,

standardize="zscore\_sample",standardize\_confounds="zscore\_sample")

For the first subject, get time series in the atlas with

masker.transform(data.func[0])

1. Now let’s get the connectivity matrix for this subject, where we use (Pearson) correlation as connectivity metric. You can use the ConnectivityMeasure() function from nilearn.connectome, or functions from numpy or scipy.

Plot the resulting matrix using an appropriate function from either nilearn or pyplot. All label names should be given as ticks on the x and y axes. Make sure that the z-axis (or v-axis) is centered on zero.

(10 points)

1. Next, let’s look at the strongest connections. Threshold the matrix so that only the strongest 5% edges/connections remain and plot the resulting sparse matrix with plotting.plot\_connectome()

Repeat with strongest 10%, and with strongest 30% connections.

(10 points)

1. Now, let’s identify some hubs. Threshold the matrix at 30% and compute the degree for each node/parcel.

Identify the parcels with largest degree.

Now, for each of these parcels, repeat plot\_connectome() but so that no other connections are shown other than the 8 strongest connections of this parcel.

(10 points)

1. Next, let’s move on to group-level analysis and investigate which hubs differ between patients and controls.

Get the time series and the connectivity matrices for all subjects. Note that the subjects’ recordings have differing numbers of samples.

Compute and plot the group mean connectivity matrix for all subjects, for patients, and for controls. Then, plot the difference between patients and controls.

(10 points)

1. For the patient mean, and the control mean, threshold at 30% and compute the parcels’ degrees. Compute the difference in degree between controls and patients for each parcel. Identify the 7 parcels with the largest differences in degree between both groups. Repeat with betweenness centrality (for which you can use functions from *networkx*).

(10 points)