

Author: Jingyi Wang, Ph.D (wangjingyimayer@gmail.com)

This pipeline calculates the rate of regional-whole brain resting state functional connectivity (rsFC) pattern changes across time. To obtain the regional-whole brain rsFC patterns for each session, we first averaged the time series across all voxels within each region of interest. To measure the connectivity pattern in each session, Pearson's correlation values (indexing resting state functional connectivity with each seed) were then obtained using AFNI for all gray-matter voxels, which were Fisher-z-transformed. Next, we calculated Pearson's correlation coefficients (i.e., indexing similarity) for resting state functional connectivity patterns between every pair of sessions (across all gray matter voxels), which were Fisher-z-transformed. Between-session similarity values that exceeded 3 standard deviations relative to the mean across all pairs of sessions for each seed region were considered outliers and excluded from subsequent analyses. Finally, to capture whether connectivity patterns reliably tracked elapsed time, a temporal drift score was calculated for each seed ROI. To do so, we correlated the similarity of connectivity patterns (Z-transformed correlation coefficients obtained for every session pair) with the delta time interval between session pairs.

Steps for calculating the temporal drift scores and time-related temporal drift for each participant and each ROI:

Step 0: To run the scripts below, ensure that you have structural scans (i.e., T1) for a single session or multiple sessions, resting-state functional scans for each session, and a spreadsheet that indicates run-wise parameters (e.g., scan time, emotion state, hormone level, etc.).

Step 1: **Preparing the T1 scans:** The `AverageDenoise_AnatScans.slurm` script is written in Bash and is designed to run on a Slurm-based Linux system. This script uses FSL to average multiple T1 scans and then denoise the resulting file. If you have only one T1 scan per participant, you may skip this script and proceed directly to Step 2.

Step 2: **Skull Strip T1 scan:** The `AntSkullstrip.slurm` script is written in Bash and is designed to run on a Slurm-based Linux system. This script uses ANTs to remove the skull from the T1 scans (single-session T1 or averaged T1 from Step 1) for each participant.

Step 3: **Create template .fsf file.** You need to modify the place folders in the FSL feat example file: RS_preproc.fsf. Below is the list of place holders:

- YOURPROJECTDIR
- SUBID
- SESSION
- RUN
- /sw/fsl/data/standard/MNI152_T1_2mm_brain (Change to the correct MNI standard brain file provided by FSL).

Step 4: **Preprocessing:** The `NKIpprocess.sh` script is written in Bash and requires both FSL and AFNI. This script creates a functional mask for the resting-state functional scan, calculates six motion parameters, and extracts the average signal from FSL-derived cerebrospinal fluid and white matter masks. These signals are then entered as nuisance regressors using AFNI's 3dDeconvolve function. Additionally, the script registers the preprocessed resting-state functional data to structural space and applies a band-pass filter ($0.01 \text{ Hz} < f < 0.1 \text{ Hz}$) using AFNI's 3dBandpass function.

Step 5: **Create Masks:** All scripts in this step are written in Bash. The `MNIMasks_registration.slurm` script requires FSL and registers standard-space masks to T1 space. The `Freesurfer.slurm` script requires FreeSurfer and generates cortical masks in T1 space. The `CreateFinalMasks.slurm` script also requires FSL and ensures that the final masks include only voxels with valid functional signal, based on the functional mask created in Step 4.

Step 6: **Compute Regional–Whole Brain Functional Connectivity:** The `amyConnect_NKI_HemiMerge_all.sh` script is written in Bash. It requires AFNI and uses the masks created in Step 5 to compute regional–whole brain functional connectivity for each session.

Step 7: **Compute Regional–Whole Brain Functional Connectivity Matrix Correlation Between Sessions:** The `VoxelCorrel_HemiMerge_z_sep.py` script is written in Python. It requires the Nilearn package and uses the regional–whole brain functional connectivity files generated in Step 6 to compute the pairwise functional connectivity pattern correlation matrix between each session pair.

Step 8: **Getting the Session Pair Calculations:** The `VoxelCorrel_postanat_cluster.py` script is written in Python and requires the Pandas package. It adds pairwise information to the functional connectivity pattern correlation matrix, including temporal intervals, emotional differences, and hormonal differences.

Step 9: **Compute Temporal Drifting Score:** The `RegionalWholeBrain_Basic.Rmd` script is written in R. The required packages are listed in lines 8–16. It calculates the temporal drifting score for each mask and each participant, and performs control analyses to determine whether the neural pattern in a specific ROI is more "drifty" than that in control ROIs.

The folder structure is illustrated below. The scripts expect raw data to be organized in a specific folder structure (illustrated on the right side of the diagram) for each participant and session. When run, the scripts will automatically process the raw data and generate output files, organized into the folder structure shown on the left side of the diagram.

`/YOURPROJECTDIR/YOURSUBJECTDIR/results/ses-${session}` contains the preprocessed files from FSL FEAT and AFNI.

`/YOURPROJECTDIR/YOURSUBJECTDIR/results/FS` contains the Freesurfer outputs.

`/YOURPROJECTDIR/YOURSUBJECTDIR/results/finalmasks` contains the regional masks.

`/YOURPROJECTDIR/YOURSUBJECTDIR/results/anat` contains the processed anatomical files.

/YOURPROJECTDIR/YOURSUBJECTDIR/results/1D_fc contains the regional-wholebrain functional connectivity pattern file for each participant, each session

