QC_Pipeline Documentation

Task DataFrame

The Task DataFrame is a Python class that generates motion metrics and percent coverage metrics for fMRI preprocessed data. The class takes in the name of the task and the BIDS directory where the preprocessed data is stored, generates a DataFrame containing different metrics and then flags values that exceed certain thresholds as potentially problematic.

Usage

To use the class, create an instance of the class and pass in the task name and fMRIprep directory:

```
td = Task_DataFrame(task_name, fmriprep_dir)
```

The class will then generate several DataFrames of motion metrics and percent coverage tables:

- `task_df`: the main DataFrame of motion metrics for the task
- `mean_highlighted`: a DataFrame of motion metrics with values exceeding a mean threshold of 0.7 flagged as

problematic

- `max_highlighted`: a DataFrame of motion metrics with values exceeding a max threshold of 5 flagged as problematic
- `mean_table`: a DataFrame of motion metrics with mean values exceeding the thresholds of 5 for translations and 0.5 for framewise displacement
- `max_table`: a DataFrame of motion metrics with max values exceeding the thresholds of 5 for translations and 0.5 for framewise displacement
- `mean_table_highlighted`: a DataFrame of motion metrics with mean values exceeding the thresholds of 5 for translations and 0.5 for framewise displacement, and with values exceeding the mean threshold of 0.7 flagged as problematic
- `max_table_highlighted`: a DataFrame of motion metrics with max values exceeding the thresholds of 5 for translations and 0.5 for framewise displacement, and with values exceeding the max threshold of 5 flagged as problematic
- `percent_coverage_table`: a DataFrame containing the percent coverage found for each ROI present in the specified atlas

- `percent_coverage_highlighted`: Same table as *percent_coverage_table* but with ROIs where subject has less than 20% coverage highlighted

Methods

The Task DataFrame class has several methods for processing the motion metrics:

- `make_directory`: creates a directory if it does not already exist
- `get_file_info`: takes in a file path and returns the subject name, session name, and task name
- `pull_confounds`: pulls specified confounds from a confounds file and outputs a motion file in tsv format
- `generate_metrics`: generates metrics for the motion parameters in the confounds file

Dependencies

The Task DataFrame class requires the following libraries:

- `os`
- `shutil`
- `sys`

- `pandas`
- `glob`

Building the Conda Environment from `QC_env.yaml`

Here are the instructions to build a conda environment using the `QC_env.yaml` file:

1. Ensure that you have the latest version of conda installed on your system. You can check your version by running the following command in your terminal:
   `conda --version`
2. Create a new conda environment using the `QC_env.yaml` file by running the following command:
   `conda env create -f QC_env.yaml`
3. Activate the new environment by running the following command:
   `conda activate QC_env`
4. Verify that the environment was created successfully by checking the list of installed packages:
   `conda list`

5. **To deactivate the environmnet, run the following:**

   ```
   conda deactivate
   ```

6. **To delete the environment**

   ```
   conda env remove -n QC_env
   ```