EVENT ABSTRACT

Back to Event

Towards Automated Analysis of Connectomes: The Configurable Pipeline for the Analysis of Connectomes (C-PAC)

Cameron Craddock[1, 2*], Sharad Sikka[2, 3], Brian Cheung[1], Ranjeet Khanuja[1], Satrajit S. Ghosh[4], Chaogan Yan[2], Qingyang Li[1], Daniel Lurie[1], Joshua Vogelstein[1, 5], Randal Burns[6], Stanley Colcombe[2], Maarten Mennes[7], Clare Kelly[3], Adriana Di Martino[3], Francisco X. Castellanos[3] and Michael Milham[1, 2*]

[1]   Child Mind Institute, Center for the Developing Brain, United States
[2]   Nathan Kline Institute for Psychiatric Research, United States
[3]   New York University Child Study Center, Phyllis Green and Randolph Cowen Institute for Pediatric Neuroscience, United States
[4]   Massachusetts Institute of Technology, McGovern Institute for Brain Research, United States
[5]   Duke University, Departments of Statistical Sciences, United States
[6]   John Hopkins University, Department of Computer Science, United States
[7]   Radboud University Nijmegen, Donders Institute for Brain, Cognition and Behavior, Netherlands

Introduction

To successfully examine the brain's functional architecture (connectome) and its behavioral associations, researchers need tools that facilitate reliable, replicable connectivity analyses. Here we introduce C-PAC, a configurable, open-source, automated processing pipeline for functional MRI data that builds upon a robust set of existing software packages. Users can rapidly orchestrate automated large-scale pre-processing and data analyses, and can easily explore the impact of processing decisions on their findings by specifying multiple analysis pipelines to be run simultaneously. C-PAC can reliably process hundreds or thousands of subjects through a variety of preprocessing strategies in a single run. Thus C-PAC has been optimized for use on large data sets such as those made public by the International Neuroimaging Data-sharing Initiative (INDI, http://fcon_1000.projects.nitrc.org/).
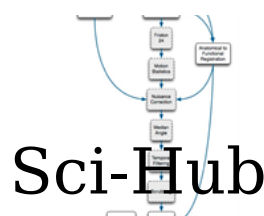
Methods

    C-PAC has been implemented in Python using the Nipype pipelining library. Nipype provides C-PAC with mechanisms to automatically detect and exploit parallelism present in a pipeline, iterate over several parameter settings, and to restart a pipeline without having to recompute previously completed processing steps. C-PAC extends Nipype functionality by providing workflows specific to connectivity analyses, functional connectivity derivatives and analyses not present in other neuroimaging packages, and a simplified interface for specifying and running pipelines. The CPAC workflows are built from AFNI and FSL tools, as well as algorithms coded in Python using Scipy, Numpy and scikit-learn.
The C-PAC processing and analysis pipeline (fig. 1) is configured through a simple configuration file, which permits the inclusion and exclusion of different steps, and setting of a variety of parameters. A variety of input data organization schemes and subject specific acquisition parameters (slice acquisition, slice timing information, time point censoring) are easily configured through a subject configuration file. Available preprocessing options include: motion correction, anatomical/functional coregistration, spatial normalization, spatial and temporal filtering, tissue segmentation, slice-timing correction, several variations of nuisance signal removal and volume censoring (motion "scrubbing"). C-PAC also includes a number of advanced analysis methods that facilitate detailed exploration of connectivity patterns, network structure, and brain-behavior relationships. Individual-level measures include: Seed-based Correlation Analysis, Amplitude of Low Frequency Fluctuations (ALFF) and Fractional ALFF, Regional Homogeneity, Voxel-Mirrored Homotopic Connectivity, and Network Centrality (Degree and Eigenvector). At the group level, C-PAC features Connectome-Wide Association Studies, Bootstrap Analysis of Stable Clusters, and integrated group statistics using FSL/FEAT. Additionally, users can easily extract preprocessed time-series data and connectivity matrices for analysis with other packages. C-PAC can seamlessly interact with shared memory (multi-core) and cluster-based (sun grid engine, OpenPBS) high performance computing environments to minimize computation time.

Results and Conclusions

    Currently in its alpha release, C-PAC is available as an open source project through github along with user and developer documentation (http://fcp_indi.github.com). Alpha testing is currently being performed in five partner labs and it has already been employed to process and analyze several large (~1000 subject) datasets available through the 1000 Functional Connectomes Project and INDI (e.g., ABIDE, ADHD-200, NKI-Rockland). The beta release is scheduled for early spring and will include feature enhancements including a graphical user interface, a quality assessment interface, and several new functional connectivity derivatives. Future enhancements will include integration with Freesurfer to enable surface based analyses, and the ability to process and analyze diffusion tensor imaging data.

Figure 1

* Correspondence:

Dr. Cameron Craddock, Child Mind Institute, Center for the Developing Brain, New York, NY, 10022, United States, cameron.craddock@austin.utexas.edu

Dr. Michael Milham, Child Mind Institute, Center for the Developing Brain, New York, NY, 10022, United States, Michael.Milham@childmind.org

Sci-Hub

▶ **Помочь проекту** ⧉

URL статьи или журнала, или DOI, или строка для поиска