

Construction of Functional Brain Connectivity Networks

Ricardo Magalhães^{1,2,3}, Paulo Marques^{1,2,3}, Telma Veloso⁴, José Miguel Soares^{1,2,3},
Nuno Sousa^{1,2,3}, and Victor Alves⁴

¹ Life and Health Sciences Research Institute (ICVS), School of Health Sciences,
University of Minho, Campus Gualtar, 4710-057, Portugal

² ICVS/3B's - PT Government Associate Laboratory, Braga/Guimarães, Portugal

³ Clinical Academic Center – Braga, Braga, Portugal

⁴ Department of Informatics, University of Minho, Portugal

{ricardomagalhaes,paulo.c.g.marques,josesoares,
njcsousa}@eceaude.uminho.pt,
valves@di.uminho.pt

Abstract. Graph theory and the study of complex networks have, over the last decade, received increasing attention from the neuroscience research community. It allows for the description of the brain as a full network of connections, a connectome, as well as for the quantitative characterization of its topological properties. Still, there is a clear lack of standard procedures for building these networks. In this work we describe a specifically designed full workflow for the pre-processing of resting state functional Magnetic Resonance Imaging (rs-fMRI) data and connectome. The proposed workflow focuses on the removal of confound data, the minimization of resampling effects and increasing subject specificity. It is implemented using open source software and libraries through shell and python scripting, allowing its easy integration into other systems such as BrainCAT. With this work we provide the neuroscience research community with a standardized framework for the construction of functional connectomes, simplifying the interpretation and comparison of different studies.

Keywords: Rs-fMRI, Graph Theory, Connectome, Python, Shell scripting.

1 Introduction

The world of neuroscience research is constantly marked by the development of new methodologies that seek to deliver new tools to allow researchers to better study the structure and function of the brain. While these developments are key to the success of the field, their translation into practice is neither necessarily direct nor simple in terms of application and interpretation. There exists a gap between the highly subject focused and specialized research in charge of developing these methodologies and the more broadly competent research who needs to apply them in the laboratories. To complete this process it is then necessary the creation of tools and frameworks that will allow the application of these new methodologies to every day research situations.

One such development that has received a great deal of attention over the last decade is the application of graph theory and the principle of complex networks to the

study of the brain through neuroimaging data [1]. A graph is a mathematical representation of a network, constituted by nodes, that can represent different brain areas, and edges that represent some connection or relationships between them, and allows the study of how the brain behaves as a network [2]. It is commonly applied using resting state functional Magnetic Resonance Imaging (rs-fMRI) data [3].

Despite the strong potential of this application, nowadays there is still a clear lack of free guidelines and standardized tools to build these graphs [4]. Difficulties range from the very large amounts of data to be processed to the lack of a standardized pre-processing and graph building workflows.

In this work we present a complete framework for the pre-processing of rs-fMRI data and the construction of functional connectivity graphs, using open source tools, through python and shell scripting, capable of parallel processing and ready for implementation on any research center.

2 The Complete Framework

The necessary data processing for the construction of a connectivity graph can be divided into two main phases (Figure 1): the pre-processing of MRI data, meant to improve the signal to noise ratio, remove confound data and define the brain areas that will represent the graph nodes (Figure 1 a)); the conversion and extraction of the rs-fMRI data, with the definition of the graph edges and construction of connectivity graphs (Figure 1 b)).

As much as possible the software chosen for the implementation of each block should be compatible and easily integrated with each other, improving the ease of implementation and workflow design. Furthermore, for the choice of software to be implemented in a data processing framework in research centers, there are several requisites that need to be taken into consideration:

- Robustness and validity proved by the specialized scientific community;
- A modular structure that allows for the easy re-adaptation of data processing workflows;
- Compatibility with the main operative systems used in the neuroimaging community;

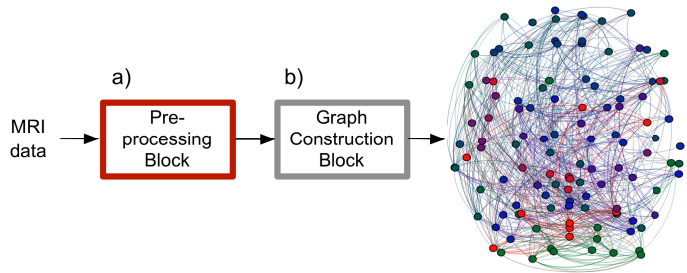


Fig. 1. Block design of the framework. Divided in two main components meant to properly prepare the data (a), and build the connectivity graphs (b).

- Possibility of integration in computational clusters and performance optimization through parallel processing;
- Open source and freely available software that can be implemented at no extra costs;

An overview of the software commonly reported in neuroimaging articles easily reveals three main software packages used in the pre-processing and analysis of rs-fMRI data: Statistical Parametric Mapping (SPM, <http://www.fil.ion.ucl.ac.uk/spm/>, [5]), Analysis of Functional NeuroImage (AFNI, <http://afni.nimh.nih.gov/>, [6,7]) and FMRIB’s Software Library (FSL, <http://fsl.fmrib.ox.ac.uk/fsl/>, [8]). While SPM does present several free and well-tested tools for the pre-processing of rs-fMRI and the construction of functional connectivity graphs, such as REST [9], it is presented as a Matlab (<http://www.mathworks.com>) toolbox, with its use conditioned to the acquisition of a paid license AFNI is developed in C, compatible with Mac OS, Linux, Solaris and SGI, with open source code. Finally, FSL is freely available for Mac OS and Linux in C++ and shell scripts. Both AFNI and FSL present the necessary tools for the pre-processing and extraction of rs-fMRI data, while FSL presents more tools for the analysis of this kind of data, as well as better compatibility with other software. Furthermore, we have previously developed and tested workflows for the general processing of MRI data [10,11] using FSL, making it an ideal choice for the current implementation. Another very useful set of tools in the pre-processing of MRI data is FreeSurfer [12] developed by the Martinos Center for Biomedical Imaging. It is a unique tool in its ability to segment individual brains using different atlas. It is developed in C while also using shell scripts and is compatible with Mac OS and Linux.

For the construction of the connectivity graphs the options are far scarcer. While there are several options for the analysis and visualization of such graphs, only a few present the necessary tools to build them. Two of the most common options, REST and Conn [13], are once again developed in MATLAB, making its implementation in the desired framework impossible. An alternative strategy is the use of freely available libraries for a language compatible with the desired framework. Nipy [14] is a library developed in Python for the processing and analysis of neuroimaging data, meant to give an integrative platform for the easy implementation of new methods, with compatibility with FSL and SPM output data. In its nitime package it presents

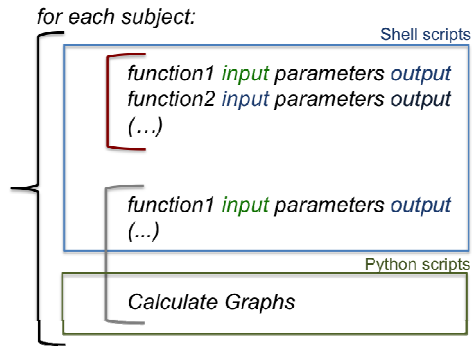


Fig. 2. Detailed structure of the framework implementation. Data pre-processing on the red bracket, graph construction block on gray bracket. Blue block represents the shell scripts and green block represents python scripts.

the necessary tools for the construction of connectivity graphs from rs-fMRI time series, and as such was chosen for the implementation in the present framework.

The present framework will then consist of a series of shell and python scripts, capable of carrying state of the art pre-processing of rs-fMRI data and constructing connectivity graphs using FSL and nipy. It can be easily integrated in the data processing workflows of any research center, or more user-friendly software such as BrainCAT. The use of shell scripts allows for an easy automatization of the all process and the processing of large amounts of data with minimal interaction from the user. The overall structure of the scripts can be seen in Figure 2. The common syntax for the use of FSL functions requires the indication of the input data, the parameters and of the output name. The use of nitime follows the common python language structure.

3 Pre-processing Tools

We have previously described pre-processing workflows for the analysis of rs-fMRI data [10], and the construction of connectivity graphs [11]. In the current framework we propose the full application of the referred workflows using FSL commands implemented in shell scripts. Using as inputs a list of subject reference codes, the pre-processing shell script then calls each of the necessary FSL functions, properly naming the output files.

The proposed pre-processing workflow is constituted by the steps of format conversion, initial volumes removal, slice timing correction, motion correction and brain tissue extraction (Figure 3 a)), and normalization to the functional native space and temporal filtering (Figure 3 c)).

Furthermore we propose the addition of a step of removal of confound data: removal of white matter (wm) and cerebrospinal fluid (csf) through an aCompCor strategy [15], the removal of motion confounds [16] from the motion correction step and of the global signal (Figure 3 b)).

Additionally, we include in this block the processing of anatomical data (aMRI), used in the removal of confound data in the aCompCor strategy. The anatomical data is later segmented using the Freesurfer workflow (Figure 3 d)) and the result of the segmentation will be used in the connectivity graph construction block.

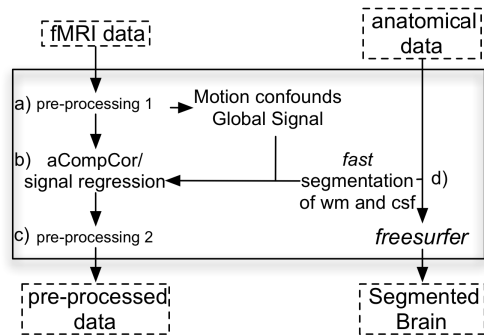


Fig. 3. Data pre-processing workflow. Input and output data in dashed lines. fMRI pre-processing described in three phases: a), b) and c); Anatomical segmentation in two: fast segmentation and freesurfer processing.

4 Implementation of Graph Construction Tools

As previously described, a graph is a mathematical representation of a network constituted by nodes, that represent the units under study, and edges that represent some connection or relation between them. When constructing brain connectivity graphs, the most natural interpretation of nodes is as the anatomical units of the brain, such as the ones obtained from the FreeSurfer segmentation. For the edges, when dealing with functional data, the Pearson correlation between the different anatomical areas is the most common option. As such, we suggest the use of FSL's `fslmeants` (Figure 3 a)) function for the extraction of the time series, using the output from FreeSurfer as the label option. This function extracts, for each different label in the brain segmentation, the average time-series across all voxels of a brain region and saves it in a text file. To do this it is necessary to convert the output from FreeSurfer into the FSL compatible nifti format, which can be done using the FreeSurfer command line tool `mri_convert`.

The final step necessary to build the connectivity graphs is to calculate the correlations between each pair of time series, creating an adjacency matrix. This is done using the python libraries `nipy` and `nitime`. Using the `TimeSeries` function the timeseries for each region is imported into an object. Finally, the function `CorrelationAnalyzer` (Figure 4 b)) can be used to calculate all the Pearson correlations between the time series and create a correlation matrix object, which can then be saved as a text-file, or a correlation matrix image.

The entire workflow was implemented using shell and python scripts. For each subject code the script sequentially calls all the FreeSurfer and FSL functions as well as the python scripts necessary to calculate the adjacency matrices (Figure 3).

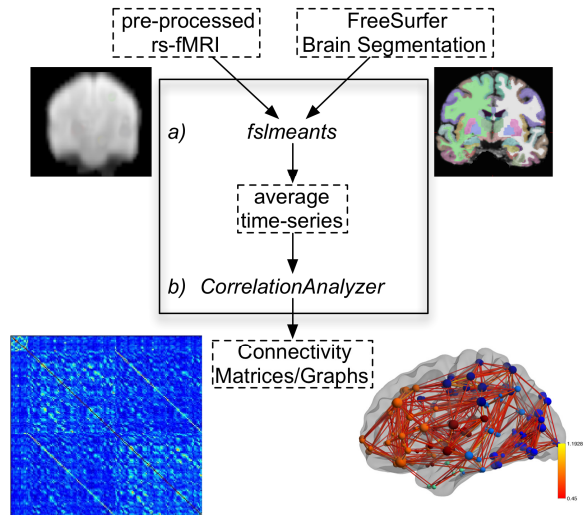


Fig. 4. Construction of the connectivity matrices. Inputs and outputs surrounded by dashed lines. This step is comprised by two stages: a) extraction of the time series using the segmented brains and the fMRI data; b) calculation of the connectivity matrices;

5 Application of the Framework to Real Data

To test the developed framework, a dataset of 54 healthy subjects from the SWITCHBOX project were used in this study (30 males and 24 females aged between 51 and 82 years old and with a mean age of 64.85 – 8.82 years). The goals and tests of the study were explained to all participants, who provided informed written signed consent. For each subject two different acquisitions were used in the framework: a T1 Magnetization-Prepared Rapid Gradient Echo (MPRAGE) with repetition time (TR)=2730 msec, echo time (TE)=3.5 msec, field of view (FoV)=256·256mm, flip angle=7°, in-plane resolution of 1·1mm, and 1mm slice thickness, used as a structural acquisition; a T2* Echo-Planar Imaging (EPI) acquisition with 180 volumes, TR=2000 msec, TE=30 msec, FoV=224·224mm, flip angle=90°, in-plane resolution of 3.5·3.5mm, and 4.5mm slice thickness, as a resting state acquisition. All acquisitions were performed on a clinically approved Siemens Magnetom Avanto 1.5 T (Siemens Medical Solutions, Erlangen, Germany) scanner, in Hospital de Braga, using a 12-channel receive-only Siemens head coil. During the resting-state acquisition, all subjects were instructed to remain with their eyes closed and to not think of anything in particular. The described framework was applied to all the subjects and the corresponding adjacency matrices were obtained. Using Matlab (v.2009b, www.mathworks.com), the fisher r-to-z transform was applied to all subjects, to assure normality of distribution. To allow the evaluation of the networks created, a one-sample t-test was done to calculate the connections significantly different from zero, correcting for multiple comparisons with the Bonferroni procedure, which controls for the Family Wise Error Rate (FWER). The resulting networks were averaged across all subjects and then further thresholded to keep values of z greater than 0,45. The resulting adjacency matrix can be visualized in Figure 5. It is possible to observe that sub-cortical regions are more interconnected with each other than with cortical regions. Furthermore it is also possible to observe a stronger tendency for regions to be connected with regions in the same hemisphere. Finally, it can be seen a strong tendency for cortical regions to have a stronger connection to their counterpart in the other hemisphere, as observed by the diagonal crossing the Cort L/Cort R quadrant (Figure 5).

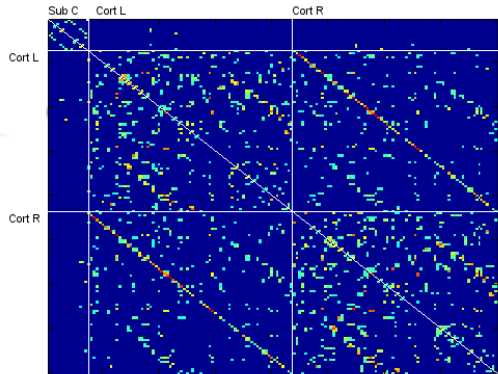


Fig. 5. Average z -connectivity matrix of the significant connections, FWE corrected and thresholded at 0.45. Sub C: Sub-Cortical areas; Cort L: Areas of the left cortex; Cort R: Areas of the right cortex.

Using the Brain Connectivity Toolbox [17] the key network properties were calculated. When compared to equivalent random networks, the network built was found to have a much higher tendency to form clusters (ratio of 5.11) and a slightly lower global efficiency (ratio of 0.7), meaning that they are small-world networks, as was expected. Finally, a modularity algorithm was applied to determine communities within the network. BrainNet Viewer [18] was used to build a 3D visualization of the graph (Figure 6). It is possible to observe a tendency for communities to be composed of spatially proximal areas.

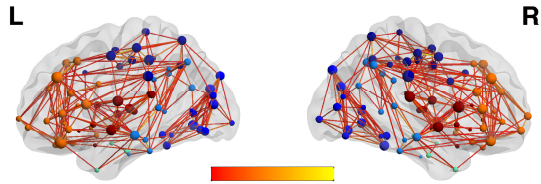


Fig. 6. Graph visualized using BrainNet Viewer; Connection color represents its z value. The node size codes the degree and the color the community of the node.

6 Conclusion

In the present work we have described the implementation of a complete framework for the construction of connectivity graphs from functional MRI data, as well as the necessary pre-processing steps. The framework uses open source software, capable of running on any Unix system. It uses shell and python scripts, in such a way that it can be easily integrated into other frameworks.

The implemented fMRI data pre-processing workflow was specifically designed for the construction of connectivity graphs and tested in a dataset of real rs-fMRI data.

Our team in the ICVS is currently applying the complete framework for the analysis of larges amount of data in projects related to the study of aging, stress, education, neurological and psychiatric disorders such as obsessive compulsive disorders.

Acknowledgements. This work has been supported by FCT - Fundação para a Ciência e Tecnologia within the Project Scope UID/CEC/00319/2013. RM is supported by a fellowship of the project FCT-ANR/NEU-OSD/0258/2012 founded by FCT/MEC (www.fct.pt) and by Fundo Europeu de Desenvolvimento Regional (FEDER). PM is supported by a fellowship of the project SwitchBox-FP7-HEALTH-2010-grant 259772-2 (www.switchbox-online.eu).

References

1. Sporns, O.: From simple graphs to the connectome: networks in neuroimaging. *NeuroImage* 62(2), 881–886 (2012), doi:10.1016/j.neuroimage.2011.08.085
2. Gross, J.L., Yellen, J.: *Graph Theory and Its Applications*. Chapman & Hall/CRC (2006)
3. Bullmore, E., Sporns, O.: Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience* 10(3), 186–198 (2009), doi:10.1038/nrn2575

4. Bullmore, E.T., Bassett, D.S.: Brain graphs: graphical models of the human brain connectome. *Annual Review of Clinical Psychology* 7, 113–140 (2011), doi:10.1146/annurev-clinpsy-040510-143934
5. Penny, W.D., Friston, K.J., Ashburner, J.T., Kiebel, S.J., Nichols, T.E.: *Statistical parametric mapping: the analysis of functional brain images: the analysis of functional brain images*. Academic Press (2011)
6. Cox, R.W.: AFNI: software for analysis and visualization of functional magnetic resonance neuroimages. *Computers and Biomedical Research, An International Journal* 29(3), 162–173 (1996)
7. Cox, R.W.: AFNI: what a long strange trip it's been. *NeuroImage* 62(2), 743–747 (2012), doi:10.1016/j.neuroimage.2011.08.056
8. Smith, S.M., Jenkinson, M., Woolrich, M.W., Beckmann, C.F., Behrens, T.E.J., Johansen-Berg, H., Bannister, P.R., De Luca, M., Drobnjak, I., Flitney, D.E.: Advances in functional and structural MR image analysis and implementation as FSL. *NeuroImage* 23, S208–S219 (2004)
9. Song, X.W., Dong, Z.Y., Long, X.Y., Li, S.F., Zuo, X.N., Zhu, C.Z., He, Y., Yan, C.G., Zang, Y.F.: REST: a toolkit for resting-state functional magnetic resonance imaging data processing. *PloS One* 6(9), e25031 (2011), doi:10.1371/journal.pone.0025031
10. Marques, P., Soares, J.M., Alves, V., Sousa, N.: BrainCAT - a tool for automated and combined functional magnetic resonance imaging and diffusion tensor imaging brain connectivity analysis. *Frontiers in Human Neuroscience* 7, 794 (2013), doi:10.3389/fnhum.2013.00794
11. Magalhaes, R., Marques, P., Soares, J., Alves, V., Sousa, N.: The Impact of Normalization and Segmentation on Resting-State Brain Networks. *Brain Connectivity* (2014), doi:10.1089/brain.2014.0292
12. Fischl, B.: FreeSurfer. *NeuroImage* 62(2), 774–781 (2012), doi:10.1016/j.neuroimage.2012.01.021
13. Whitfield-Gabrieli, S., Nieto-Castanon, A.: Conn: a functional connectivity toolbox for correlated and anticorrelated brain networks. *Brain Connectivity* 2(3), 125–141 (2012), doi:10.1089/brain.2012.0073
14. Millman, K.J., Brett, M.: Analysis of functional magnetic resonance imaging in Python. *Computing in Science & Engineering* 9(3), 52–55 (2007)
15. Behzadi, Y., Restom, K., Liu, J., Liu, T.T.: A component based noise correction method (CompCor) for BOLD and perfusion based fMRI. *NeuroImage* 37(1), 90–101 (2007), doi:10.1016/j.neuroimage.2007.04.042
16. Power, J.D., Barnes, K.A., Snyder, A.Z., Schlaggar, B.L., Petersen, S.E.: Spurious but systematic correlations in functional connectivity MRI networks arise from subject motion. *NeuroImage* 59(3), 2142–2154 (2012), doi:10.1016/j.neuroimage.2011.10.018
17. Rubinov, M., Sporns, O.: Complex network measures of brain connectivity: uses and interpretations. *NeuroImage* 52(3), 1059–1069 (2010), doi:10.1016/j.neuroimage.2009.10.003
18. Xia, M., Wang, J., He, Y.: BrainNet Viewer: a network visualization tool for human brain connectomics. *PloS One* 8(7), e68910 (2013), doi:10.1371/journal.pone.0068910