Inserting an index of motion degradation into the analysis of MRI data (QUIQI)

Analysis code

Authors: Antoine Lutti & Giulia Di Domenicantonio, 2021

Laboratory for Neuroimaging Research

Lausanne University Hospital & Lausanne University, Lausanne, Switzerland

INTRODUCTION

This package includes supporting material for the scientific article by Lutti et al. entitled 'Inserting an image quality index into the analysis of MRI data' (REF) , which introduces a method is to account for the degradation of data quality due to motion in the analysis of MRI data. The purpose of this package is to allow the scientific community to cross-examine the analysis script written to implement the method and to replicate the main results obtained. This package also constitutes a template to help interested users to implement this method for their own neuroimaging studies. Note that this code has been integrated into a customized version of the hMRI toolbox () to facilitate its use. Integration into the main version of hMRI toolbox (https://hmri-group.github.io/hMRI-toolbox/) is planned for a later date.

The complete support package for the QUIQI method includes:

1. A copy of the original analysis code used to compile the results presented in the original scientific publication. This analysis script is described in further details here.
2. A subset of the data used in the original publication for computation of the results, available here: . This data also includes a set of analysis results obtained by running the code described in 1. on the provided data. From the provided code and data, users should be able to replicate the computation of these results.

This method requires the values of a Motion Degradation Index for each of the datasets to be analysed. The proposed method can accommodate all types of MDI, which need to be computed separately prior to the analysis of the data. In its current form, the analysis script expects the MDI values as a separate field in a matlab structure that contains relevant metadata (Subject_Details.mat). This can be modified according to users' preferences.

CONTENT

The provided material contains the matlab scripts required for data analysis. The main function is RunQUIQI.m. In the SPM folder, spm_spm.m and spm_est_non_sphericity.m are SPM12 files (version 7771) that were edited to provide the Free Energy estimates from the ReML computation as an output. These files are not a requirement for the analyses but may be used when change in Free Energy need to be considered. Also, note that the provided spm_spm.m script also contains an optional call to spm_reml_sc to enforce positive hyper-parameter estimates.

REQUIREMENTS

A running version of Matlab (https://ch.mathworks.com/products/matlab.html) and SPM (https://www.fil.ion.ucl.ac.uk/spm/) are pre-requisites for running these analyses. This package may be run on the publicly available data () or on data provided by the user.

INSTALLATION

- Download the provided package and set the directory locations in GetParams.m according to your installation

- In matlab, set the paths to your SPM installation and to the folder that contains the provided package.

MAIN FEATURES

Input variables in the main 'RunQUIQI' function

a. 'Subregions': brain regions of interest for the analysis. This variable can take value 'p1' or 'p2' for grey or white matter tissue class, or alternatively the value of the regional labels defined in the neuromorphometrics atlas (see getROIpairs function for example). Note that multiple regions can be entered into 'Subregions'. In this case, analyses will be conducted in succession for all entered regions.

b. 'lambda': desired powers of the motion degradation index used for the computation of the ReML basis functions. Setting lambda to 0 amounts to running an ordinary least-square analysis. If 'lambda' is an array of values, separate basis functions will be computed from the MDI, one for each value of the array. If 'lambda' contains multiple entries/arrays, analyses will be conducted in succession for each entry.

c. 'DataType': type of data to be analysed. In the context of this package, this variable may be set to 'MTw_R2s', 'T1w_R2s' or 'PDw_R2s' in order to analyse R2* maps computed from the MTw, T1w or PDw raw image contrasts respectively.

d. 'AnalType': type of analysis to be conducted. This variable can take the following values:

- 'Full': analysis of all the datasets available, identified from the Subject_details.mat object. Statistical analysis: one-sample t-test of the age-dependence of the data on age (a 2nd-order model of ageing is assumed).

- 'Residuals': identical to 'Full' analysis but residual images are saved for subsequent analysis vs the MDI values ('MDIvsResAnalysis' function). Note that saving individual residual images increases computation time and requires extensive disk space.

- 'Exclusion': Conducts analysis after exclusion of the fraction of the cohort with the highest value of the MDI. The main purpose of this analysis is to replicate the common use of an MDI in the community and only really makes sense by setting lambda to 0 (Ordinary Least-Square analyses). The value of the fraction of the cohort excluded from analysis is given by AnalParams.Exclusion, in the GetAnalParams function

- 'Specificity': As for a 'Full' analysis, the statistical analysis involves one-sample t-tests of the age-dependence of the data on age. However, the age assigned to each dataset is scrambled randomly across the datasets to allow for the monitoring of false positives. The number of repetitions of this analysis is set by AnalParams.NRepeats in the GetAnalParams function. Note that in order to limit the

size of the provided data, the number of repetitions was set to 10 to compute the provided results (1000 in the scientific article).

Another two types of analysis are enabled in the provided script but **were not run to generate analysis results** as they require datasets within a narrow age-range, of which not enough are available here:

- 'GroupComparison': as 'Specificity', this analysis aims to monitor the occurrence of false positives in the analyses. Here, this is done in the context of a group comparison using two-sample t-statistics. The number of datasets in the 1st group is set by AnalParams.Ngroup1 in the GetAnalParams function. As for 'Specificity' analyses, the number of repetitions of this analysis is set by AnalParams.NRepeats in the GetAnalParams function.

- 'MotionBias': this analysis examines the dependence of the R2* data on the MDI (bias). This is achieved using one-sample t-test, inserting the MDI values into the design matrix. The powers of the MDI values inserted into the design matrix is set by Params.MotionRegPowers field in the GetParams function.

Note that the variable settings underlying each type of analysis are determined in the function GetAnalParams

Main QUIQI functions

Besides the top-level function RunQUIQI.m, the main functions used in this package are:

1. **PrepAnalysis.m**

   Reads-in the details of the data to be analysed contained in Subject_Details.mat and completes all preparatory steps required for analysis (e.g. computation of the basis functions for ReML estimation, computation of the explicit masks for image analysis, editing of the analysis cohort if required,...).

   Main output: a QUIQI structure which, for each analysis to be conducted, contains information such as the folder location of the data to be analysed, the output folder for the analysis, the type of data to be analysed, the basis functions for the ReML computation, etc...

2. **RunAnalysis.m**

   Runs the analyses. Two inputs: QUIQI structure and analysis type.

FOLDER STRUCTURE

Following analysis, the folders that contain the analysis results are organized as follows:

1. Top-level folder – in the home directory defined in GetParams. The name of this folder is given by the Params.AnalysisFolder field of GetParams (default: 'Analysis'), followed by the analysis type (see above).
2. Sub-folder: repetition number (repetitions are relevant for e.g. the 'GroupComparison' and 'Specificity' analyses).
3. Sub-sub-folder: type of input data (MTw, T1w or PDw)
4. Sub-sub-sub-folder: region of interest of the analysis and powers of the MDI used

ADDITIONAL NOTES

- Regional grey matter analyses can be conducted using the provided script, for example using getROIpairs.m in the definition of the variable SubRegions in RunQUIQI.m (see

commented example in this script). However, we do not provide the results of such analyses due to the large size of the data (>100GB).

- The FalsePositiveCount.m function that monitors the rate of false positives in statistical maps relies on the cp_cluster_Pthresh.m function, included in this package, that was written by Christophe Phillips, Cyclotron Research Centre, University of Liege, Belgium.