rsHRF-Toolbox (GUI)

— Version 0.0.1 (23.07.2020**)**

Amogh Johri

https://github.com/BIDS-Apps/rsHRF/tree/amogh/rsHRF-GUI

1. Contents

2. System Overview

This aims to provide a graphical-user interface for the functionalities of the rsHRF-toolbox (https://pypi.org/project/rsHRF/1.0.8/). The aim is to facilitate retrieval of resting-state hemodynamic response function, and BOLD deconvolution from resting-state fMRI, while having provisions for intuitively managing multiple files (from multiple subjects) and visualizing the various time-series artifacts produced in the process.

3. Requirements

1. Python >= 3.5
2. Run the following command to install the required packages:

   ```
   pip3 install numpy nibabel matplotlib mpld3 scipy pybids==0.6.5 pandas patsy duecredit joblib
   ```

3. Getting Started

This section explains how to install and use rsHRF-GUI Toolbox. It also presents the various functionalities provided in the toolbox.

3.1 Installation

The current version does not have an explicit installation procedure. Clone the repository present here, and execute the *run.py* file using python (>= 3.5). The tool is operating-system independent.

3.2 GUI Application

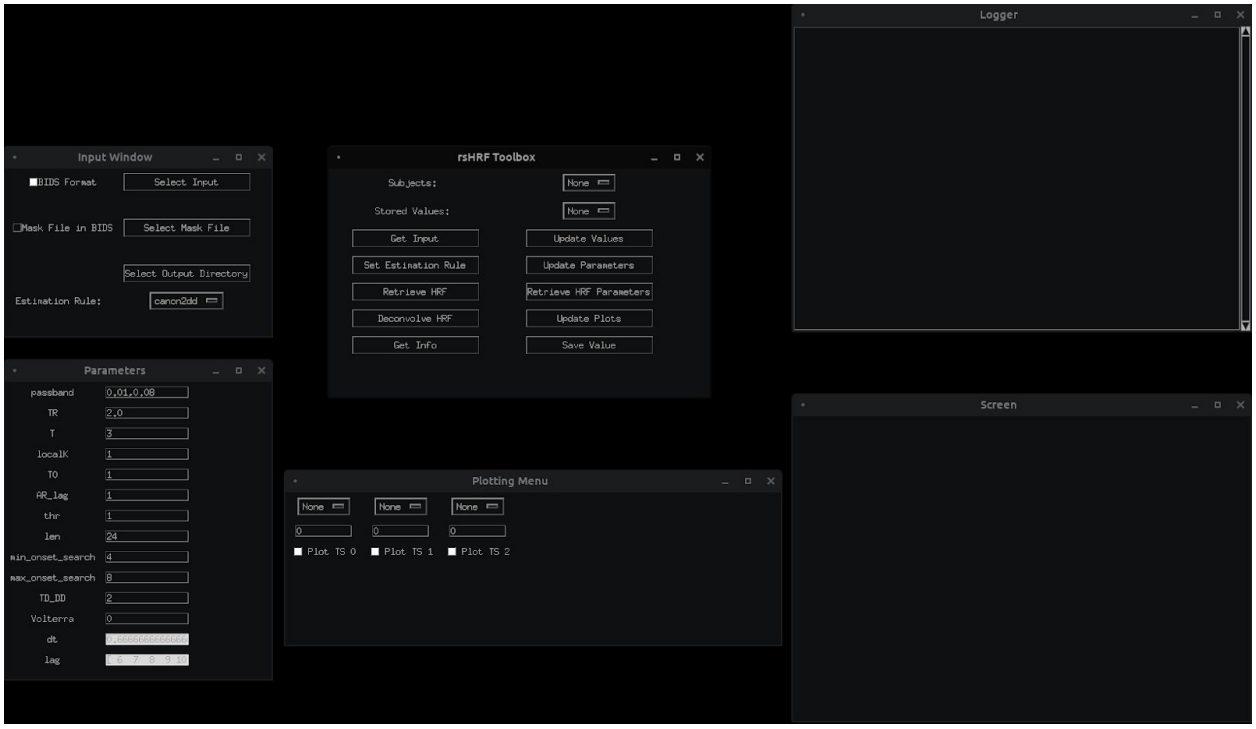

Figure 1. Overall appearance of the application.

rsHRF-GUI is a multi-window application, which consists of 6 windows (*Input window, Parameters, rsHRF Toolbox, Plotting Menu, Logger and Screen*) as shown in Figure 1. Each of these windows deals with a different-set of tasks which are discussed later in this section.

3.2.1 rsHRF Toolbox

This serves as the main window which connects all the other windows together. It can be thought of as a point-of-control for the application environment. The various widgets are described below:

1. **Subjects:** A drop-down menu which contains all the subject labels that have been included in the present analysis. Each subject is identified by their label, where the label is defined as in Brain Imaging Data Structure (BIDS).

   For example: when dealing with an input-file = "sub-031274_task-rest_bold.nii", the label is *031274.* Hence, if such an input file is included in the analysis, 031274 shall appear as one of the options in the drop-down menu.

   The figure below (Figure 2) shows how it appears in the toolbox.

Figure 2. Selecting the subject.

2. **Stored Values:** A drop-down menu which contains all the stored-values (time-series artifacts) corresponding to a particular subject. It shows all the different time-series corresponding to the current selected subject. For now, there are four types of time-series data:
    a. Raw BOLD data
    b. Preprocessed BOLD data
    c. Retrieved Hemodynamic Response Function
    d. Deconvolved BOLD data

    Each subject can have multiple time-series of each of these types.

    For example: if the input file, say *x.nii*, is used with two different mask-files: *y.nii* and *z.nii*, then they form two separate logical entries of 'Raw BOLD data'. Similarly, the other three-files involve *parameter* based processing in their formation. Hence, two files of the same-type pertaining to different sets of parameters form different logical entries.

    The figure below (Figure 3) shows how it appears in the toolbox.

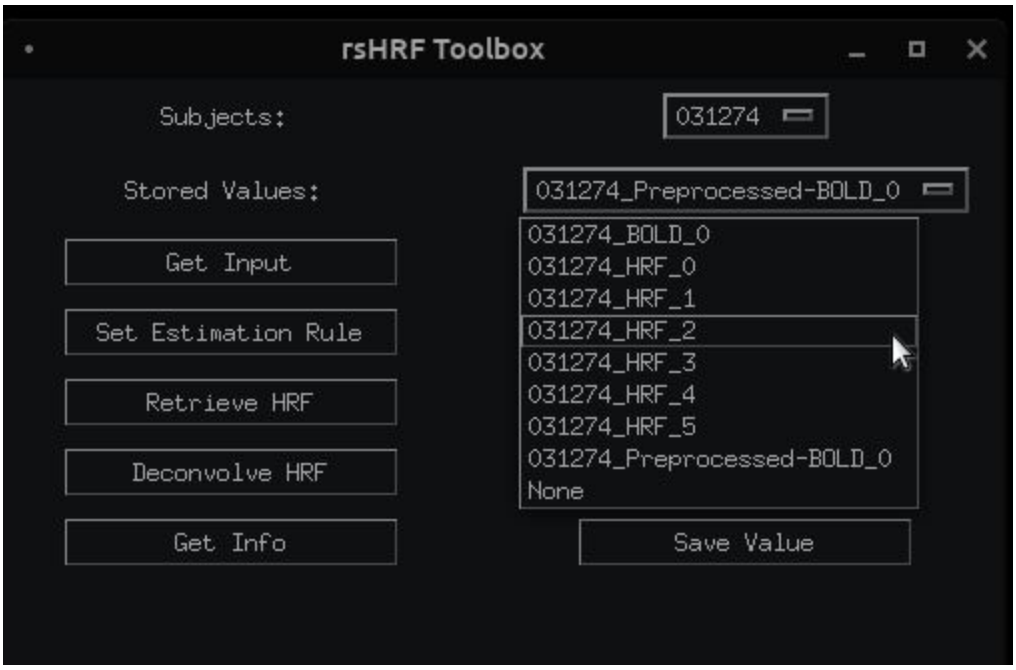

Figure 3. Selecting the time-series.

The multiple different files of the same type are simply indexed (starting at 0). This is because there can be multiple reasons which lead to two similar-type files to be different. For Figure 3., all the different HRF (Hemodynamic Response Function), were retrieved from the same *Preprocessed BOLD Data* using all six-estimation rules.

3. **Get Input:** It loads the input (corresponding to the current selections in the input-window), sets the estimation-rule, preprocesses the Raw BOLD data according (according to the current values in the parameter window) and sets the output-path (for all consequent *Save Value* operations. If the same set of values are provided in the *input window*, (one which has already occurred in the present analysis), then none of the above actions are taken.
4. **Update Values:** Once an action which produces a stored-value (time-series) artifact has been performed, to have it appear in the *Stored Value* drop-down menu, the user needs to click on this. This updates the stored-value drop-down menu to contain all the artifacts corresponding to the current subject selection. For example: Once the HRF has been retrieved for a subject (using one of the *Preprocessed BOLD Data*, the HRF does not appear directly in the *Stored Values* drop down. The user needs to update the same using this button.
5. **Set Estimation Rule:** Once the appropriate estimation rule has been selected from the *Input Window*, using this button the estimation rule gets updated, along with the *parameter window* which now displays the parameters corresponding to the new estimation rule.

6. **Update Parameters:** Once the required values are entered for each parameter in the *parameter window*, this button can be used to update the parameters to their new values. In case of an illegal value being specified, the parameters revert back to their previous legal values.
7. **Retrieve HRF:** This is used to retrieve the HRF corresponding to a *Preprocessed BOLD Input*, using the current set of estimation rules and its corresponding parameters. Having done the following, the *Update Value* button can be used to update the *Stored-Values* corresponding to the particular subject.

   Note: The HRF can only be retrieved for the *Preprocessed BOLD Input*, and hence, a suitable selection needs to be made corresponding to *Stored-Values* in order for this to function.

8. **Retrieve HRF Parameters:** This is used to retrieve the HRF parameters (Full Width at Half-Max, Time to Peak, and Height) corresponding to a retrieved HRF. This is only valid for the *Stored-Value* of type *HRF* and hence, a suitable selection needs to be made.
9. **Deconvolve HRF:** This is used to obtain the *Deconvolved BOLD Time-series* corresponding to a *HRF* time-series. The HRF time-series carries along with it, the details of the *Preprocessed BOLD Input* from which it was obtained, and the deconvolved BOLD is corresponding to the same preprocessed BOLD time-series.
10. **Update Plots:** The toolbox allows to plot the time-series of types: Raw BOLD Time-series, Preprocessed BOLD Time-series, HRF and Deconvolved BOLD Time-series. To update the options in the *Plotting Menu Window*, the *Update Plots* button needs to be clicked. It updates the *Plotting Menu Window* with all the additional plots corresponding to the current subject's *Stored-Values.*
11. **Get Info:** Since the toolbox deals with multiple-time-series corresponding to multiple-time-series-types which in turn correspond to multiple-subjects, it can get overwhelming to keep track of all the time-series artifacts even for a few (or even a single subject). On using the *Get Info* button, the *Logger Window* displays the information regarding the currently selected *Stored-Value*.

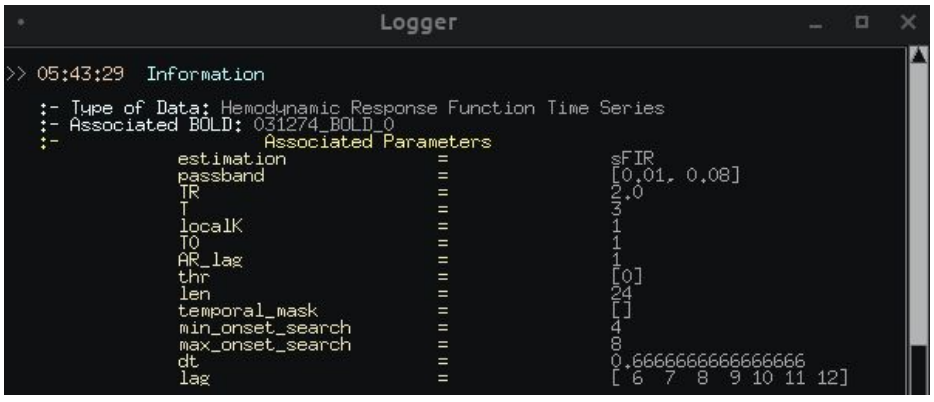

Figure 4. Get time-series Information

An example has been provided in Figure 4. for the information regarding a typical HRF time-series.

12. **Save Value:** It saves all the artifacts corresponding to the current-selected subject, in the specified *Output Directory*, as a *.mat* file.

3.2.2 Input Window

The *Input Window* deals with 4 elements: Input File, Mask File, Output Directory, Estimation Rule.

There can be three types of input files:

1. .nii or .nii.gz
2. .gii or .gii.gz
3. BIDS directory

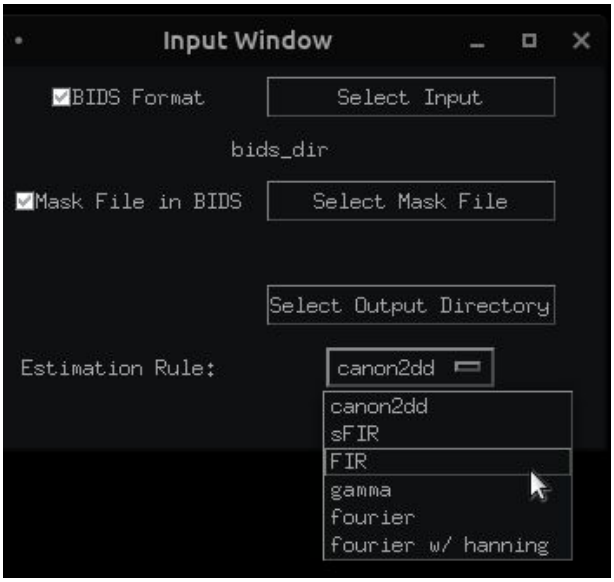

Figure 5. Input Window

If the input file is of category 'c', the checkbox *BIDS Format* needs to be selected.

The Select Input button opens a file-selection dialog, from which the appropriate file/directory can be selected. Similarly, the mask file can also be selected, however if the mask file is present for all the subjects in the BIDS directory, then the checkbox *Mask File in BIDS* can be selected and the application shall pick the corresponding mask files on its own (corresponding to the *--brainmask* option of the rsHRF-command-line-toolbox).

Similarly, the output directory can be selected and all the saved *.mat* files get stored in this directory.

The *Estimation Rule* can be selected from the drop-down menu as shown.

Having performed the above steps, *Get Input* should be selected from the *rsHRF Toolbox* window for the corresponding processing and updates.

3.2.3 Parameter Window

Displays the various parameters corresponding to the *estimation rule* that has been selected. The user can modify the value of the parameters from here. After modifying the values, the user needs to use the *update parameters* button. In case of a valid/invalid modification, a suitable message is given through the *logger window*. For more information regarding the parameters, look at the documentation of rsHRF-toolbox.

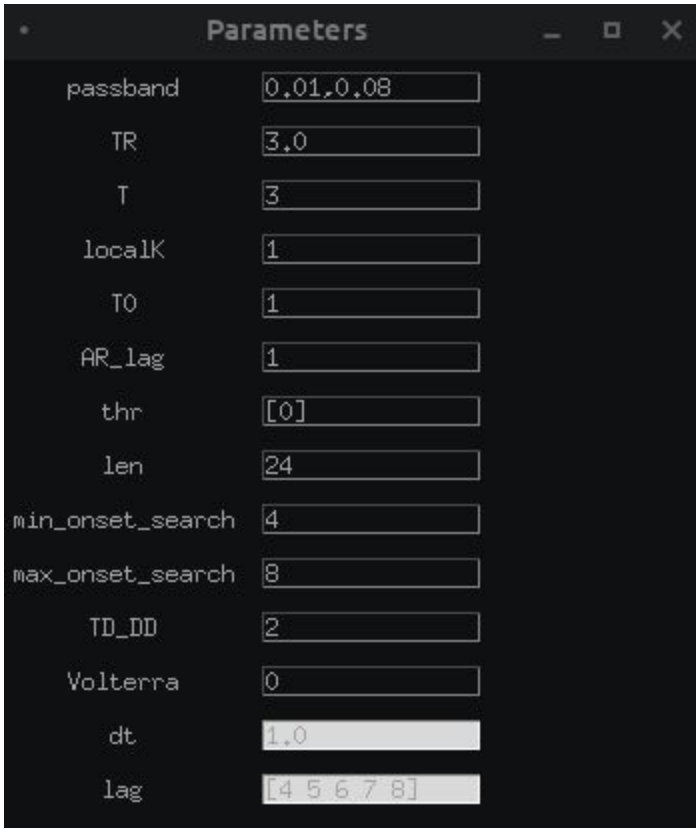

Figure 6. Parameter Window

Figure 6 shows the *parameter window* corresponding to *canon2dd* estimation rule. The parameters that are *grey* represent the ones which aren't explicitly specified, but computed on the basis of other parameters.

3.2.4 Plotting Menu

The *plotting menu* contains three identical sets of three-widgets. The first widget is a drop-down menu which allows the user to select the time-series to be plotted. The second widget is an input-box which takes a single integer as input. Every time-series has two dimensions: (x, t) where x specifies the location and t specifies time. The single-integer input is to select the suitable *x*. The third widget is a checkbox to toggle between displaying the plot (on the *screen*).

A maximum of three time-series can be plotted at a time. Figure 7 shows the usage of all the widgets of the *plotting menu*.

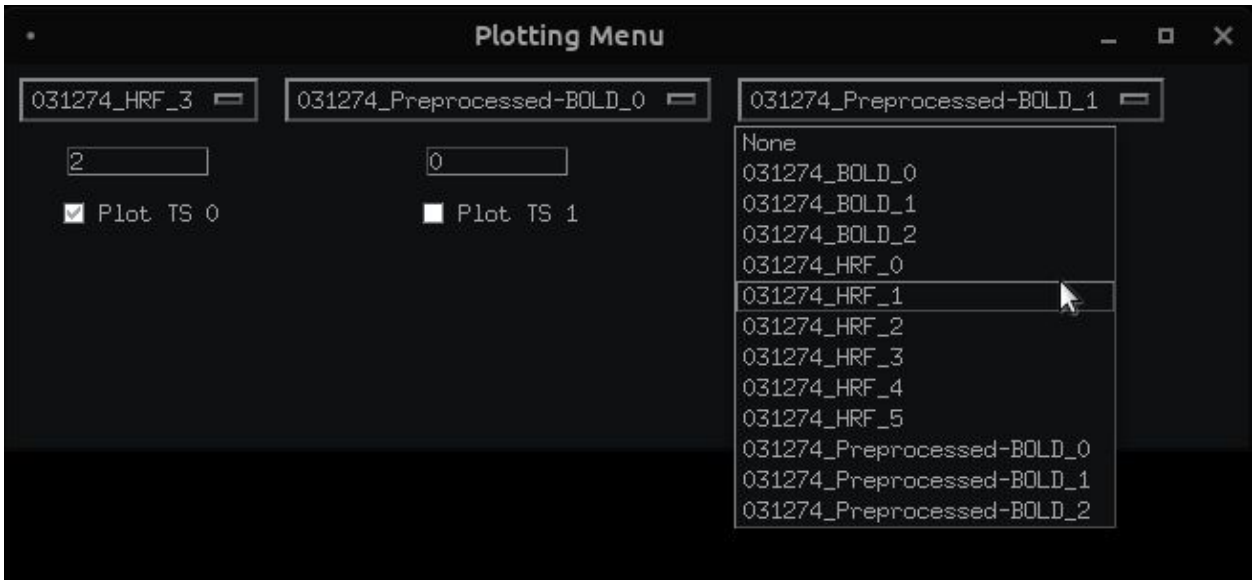

Figure 7. Plotting Menu

3.2.5 Plotting Screen

Depending upon the configuration of the *plotting menu*, the *plotting screen* displays the time-series plots. Figure 8 shows three different HRF plotted simultaneously. The three plots correspond to three different *estimation rules.*

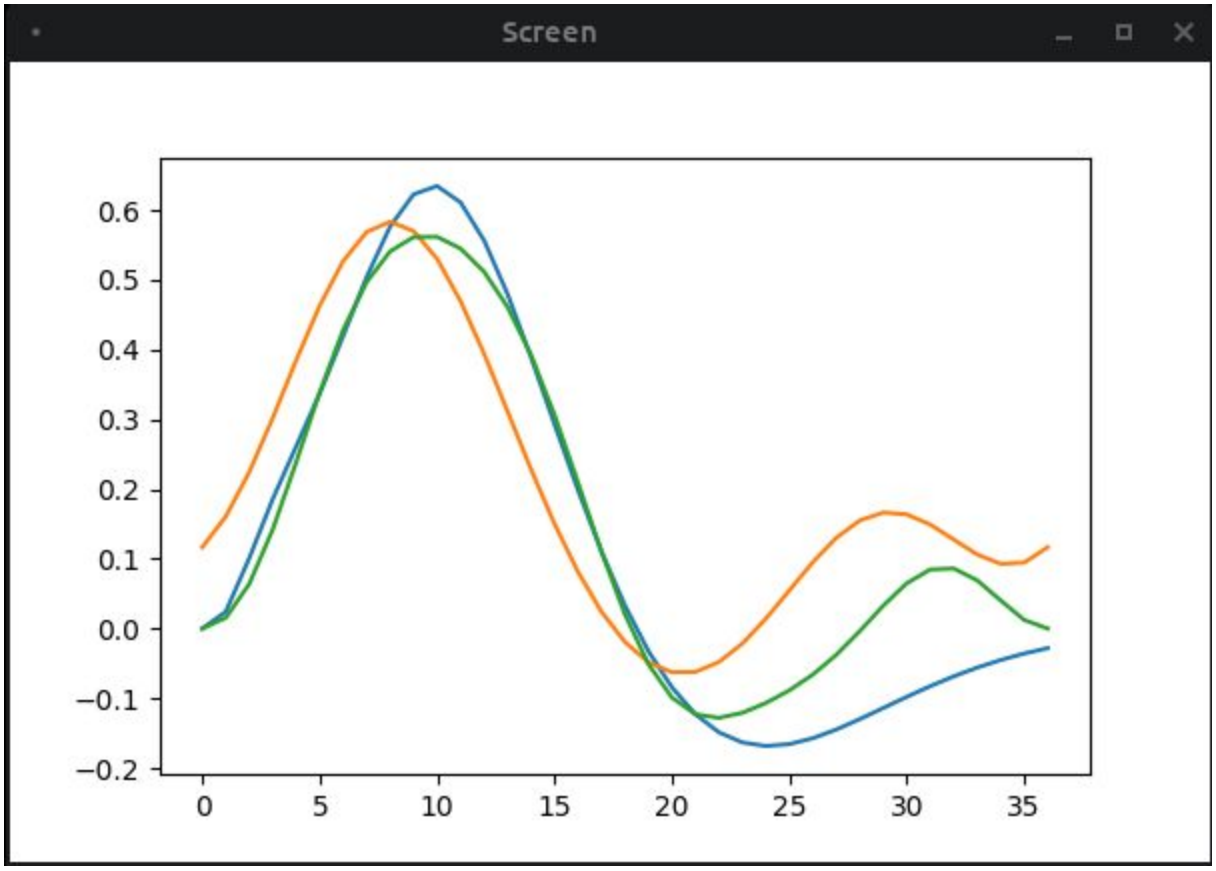

Figure 8. Plotting screen.

3.2.5 Logger

The aim for this window is to make the user-experience more interactive. It provides immediate feedback to the user regarding the state-changes, errors, etc which occur in the application environment. It also provides information, suggestions for error-corrections, etc to make the user-experience more comfortable. Figure 9 shows a typical view of how the *logger* shall appear throughout the application use.

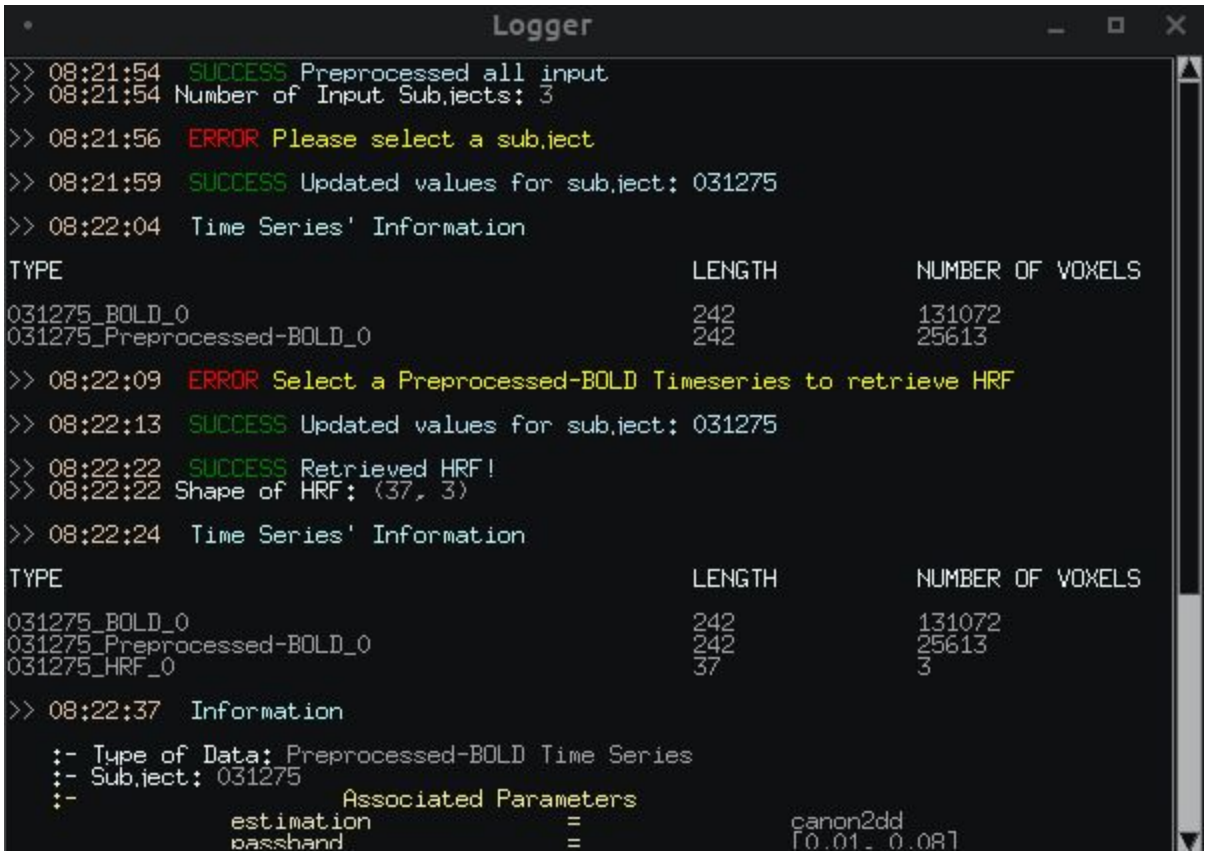

Figure 9. Logger Window

3.3 Exiting the Application

To exit the application, the user can close the *rsHRF Toolbox* window, which will in turn close all the other application windows.

Miscellaneous

For more information regarding the technical aspects of the application, visit https://github.com/BIDS-Apps/rsHRF.

In case of queries, please contact:

Amogh Johri

International Institute of Information Technology Bangalore

amogh.johri@iiitb.org

Fin.