Version

16

CNoS

Cognitive Neuropsychiatry of Schizophrenia Lab

CPCA MANUAL

Contents

APPENDIX

I.I. Definition of fMRI-CPCA

Constrained Principal Component Analysis (CPCA) combines regression analysis and principal component analysis into a unified framework. fMRI-CPCA uses CPCA to analyze fMRI data, and derives images of functional functional networks from singular-value decomposition of BOLD signal time series with the analyzed variance constrained to that which is predictable from stimulus timing and other constraints. Constraining the analyzed BOLD signal variance allows derivation of functional networks that change activation in peristimulus time, using non-peristimulus time scans as a baseline. fMRI-CPCA can also be thought of as deconvolution PCA.

CPCA allows:

1. Determination of multiple functional networks involved in a task.
2. Estimation of the pattern of BOLD changes associated with each functional network over peristimulus time points.
3. Quantification of the degree of interaction between these multiple functional networks.
4. A statistical test of the reliability of components, and the degree to which experimental manipulations affect each functional network.

- fMRI-CPCA provides all results in *matlab.mat* file format, and writes images in analyze format for all components, rotated and unrotated.

  Note: fMRI-CPCA does not assume ANALYZE images need to be left-right flipped and does not perform any left-right flipping at all. Therefore, it is up to the user to understand whether or not the images submitted to CPCA are in radiological or neurological orientation, as this orientation will be maintained throughout the CPCA analyses. When we refer to "flipping" in CPCA below we are referring to reversing the sign of a component.

This manual includes a tutorial that will guide you through your first CPCA analysis on example data set. A similar tutorial is available in a condensed quick and easy version in a separate document downloadable from the fMRI-CPCA website: "*Getting Started: An fMRI-CPCA Tutorial*".

I.II. Used Terms Glossary

Blood oxygen level dependant (BOLD) signal	The "activation" measure imaged with fMRI. Changes in BOLD signal are correlated with increases and decreases in blood flow relative to increases and decreases in oxygen consumption.
Component	In fMRI data, the extracted component(s) represent a network (or networks) of functionally interconnected voxels. The networks are imaged by superimposing component loadings on a brain template.
Component loadings	Can be interpreted as correlation coefficients between the component scores and the BOLD signal that is predictable from the imposed constraints (e.g., stimulus timing). The Component loadings are overlaid on the brain templates, and provide the image of the brain regions on the network represented by the component of interest.
Component scores	Value indicating how important the component is for each combination of subject and scan.
fMRI scan	A whole brain image (TR, or repetition time) of BOLD signal values for one point in time (usually 2-3 seconds per TR). When taken sequentially over time, these images can provide information about which brain regions show increases and decreases in activation in response to a particular perceptual or cognitive task.
G matrix	Design matrix. G contains a model of the predicted BOLD signal changes (columns) caused by stimulus presentation timing over all fMRI scans (rows). In traditional univariate analysis a hemodynamic response model (HRF) is used, but with fMRI-CPCA a finite impulse (FIR) response model is typically used to allow deconvolution.
Hemodynamic response (HDR)	The change in brain blood flow that is assumed to be associated with neural activity.
Mask	Image file specifying the regions of each fMRI scan that are brain, and not other materials like bone or air. The brain voxels to be included in the analysis are determined by the mask.
Predictor weights	Indicate the contribution of each aspect of the G matrix model to how each component changes over the series of fMRI scans. Can also be interpreted (somewhat loosely) as the correlations between the component scores and the columns of G.
Run	One run is a complete sequence of fMRI scans associated with one completed run of an experiment. A full experiment usually consists of anywhere from 1 to 8 runs, and each run usually consists of approximately 200 to 300 full brain scans.
Time windows (time bins)	Segment of time of interest, for instance, peristimulus time. When using a FIR model it is usually of interest to attempt to map the entire hemodynamic response (± 20 seconds). Therefore time window is usually between 16 and 24 seconds.

TR	Repetition time. Time to collect one full-brain fMRI scan (2-3 seconds).
Z matrix	Z contains the time series of fMRI scans for all subjects (rows) for all voxels of interest in the brain (columns). For fMRI data, Z can be referred to as the Activation matrix.

I.III. CPCA Overview

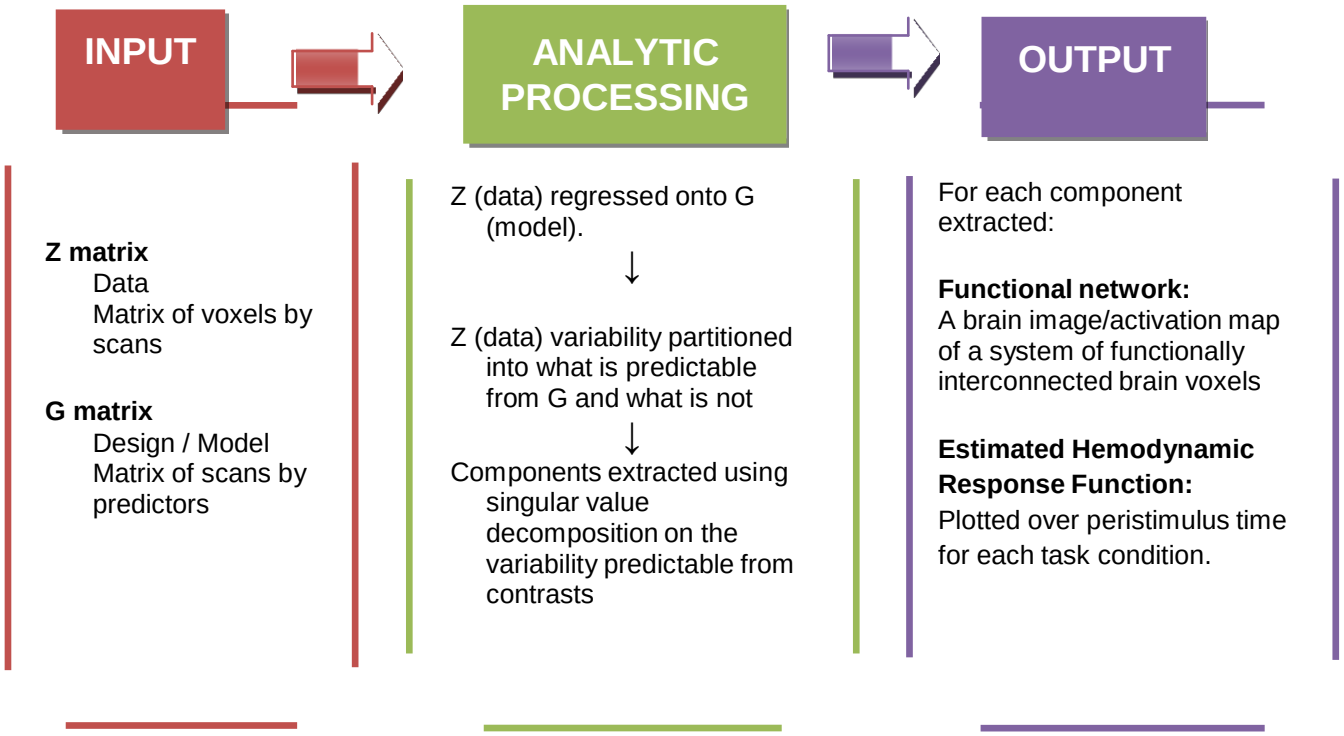


Figure 1: CPCA Overview

II. Installation Instructions

II.I. Software requirements

You will need to have the following software installed on your computer:

- MATLAB (ideally version 2008b or newer) http://www.mathworks.com/
- MRIcroN http://www.mccauslandcenter.sc.edu/mricro/mricron/index.html
- Window, Mac or Linux

II.II. Hardware requirements

The memory requirements to run CPCA are minimal. CPCA has been successfully tested on a machine with 2 GB memory, though greater efficiency is possible with greater memory capacity. The most significant requirements are:

1. Storage capacity

   The processing space required is indicated by the application, with the requirements being shown in yellow, orange and red as the needed storage space approaches or exceeds the existing storage space.

2. Memory capacity

   It is necessary to run the singular value decomposition on an entire array. The array is based on the width of the G matrix, both in depth and width. As the data set increases in size, the width of the G matrix increases, requiring more memory to fill the C matrix.

II.III. CPCA installation

- Go to the fmri-cpca website: www.nitrc.org/projects/fmricpca

There are two options for downloading files from this web page.

Downloading Option 1: The '**Download Now'** box

- Ensure that desired file, in this case "**fMRI_CPCA GUI: cpca_1.1.0.05_June_05_2013.zip",** has been selected in the '**Download Now'** box.
  - Note: There are frequent updates to the cpca gui. When downloading, always select the most current version.
- Click on '**Download Now'**. (This is a large file, so download may take a few minutes).
- Extract the files into a folder of your choosing. The extraction will create a cpca folder in the selected directory, of the format cpca_{version}_{month}{year}, such as '**cpca_1.1.0.05_June_05_2013**.'

Downloading Option 2: The '**See All Files >>**' button

- Press the '**See All Files >>**' button (located on main page to the right of the Download Now box). This will bring you to a table listing all downloadable files.
- Clicking on a given file name will initiate downloading.
- Extract the files into a folder of your choosing. The extraction will create a cpca folder in the selected directory, of the format cpca_{version}_{month}{year}, such as '**cpca_1.1.0.05_June_05_2013**.'

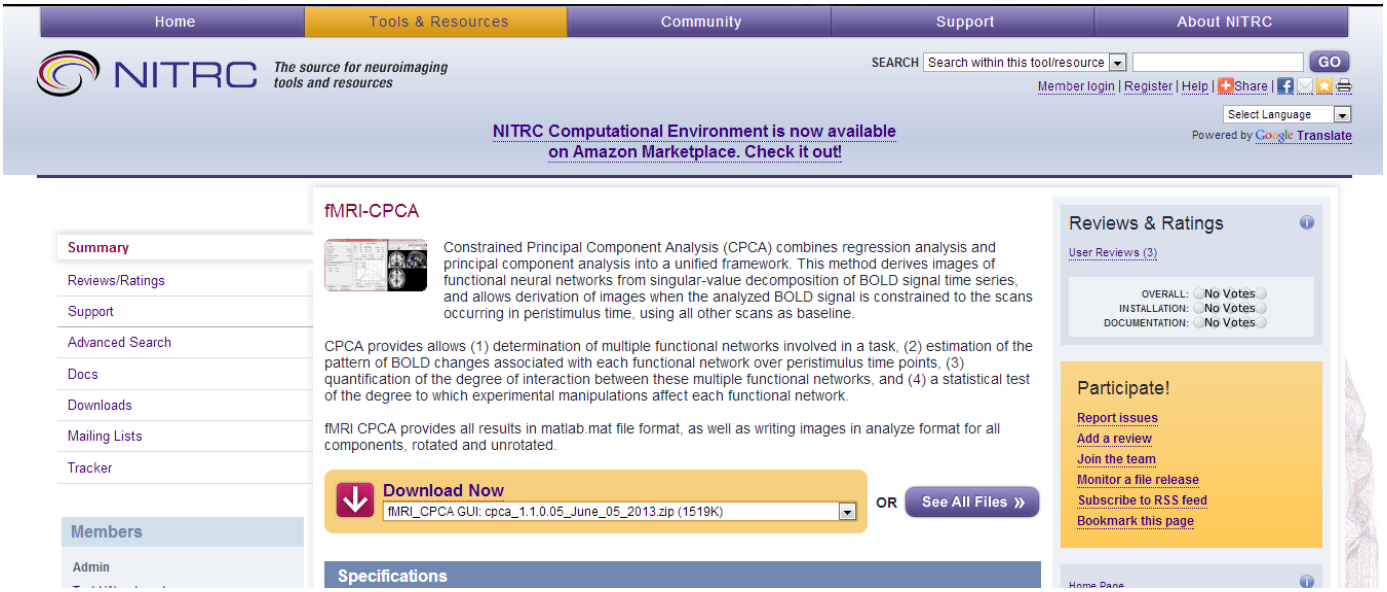


Figure 2 : NITRIC fMRI-CPCA web page

Release	Date				
	Filename	Size	D/L	Arch	Type
fMRI_CPCA GUI	2013-06-05 11:00				
	cpca_1.1.0.05_June_05_2013.zip	1.48 MB	2	Any	.zip
fMRI-CPCA Installation & Use	2013-06-05 11:00				
	fMRI_CPCA_Manual_V14_August_2011.pdf	7.36 MB	539	Any	pdf
	fMRI_CPCA_manual_V15_March_2012_draft.pdf	3.82 MB	264	Any	pdf
	fmri_CPCA_Tutorial_V11_June_2013.pdf	1.68 MB	4	Any	pdf
fMRI-CPCA Example Data	2011-12-02 11:17				
	example_data_Multiple_Groups_Subjects_Runs.zip	189.19 MB	138	Any	.zip
	example_data_Multiple_Runs.zip	189.03 MB	17	Any	.zip
	example_data_Multiple_Subjects.zip	93.94 MB	32	Any	.zip
	example_data_Single_Subject.zip	23.53 MB	25	Any	.zip
	Tutorial_Data_Set.zip	46.61 MB	49	Any	.zip
fMRI-CPCA Literature	2011-07-07 09:01				
	Decreased-efficiency-of-task-positive-and-task-negative-networks-during-working-memory-in-schizophrenia.pdf	613 KB	349	Any	pdf
	Metzak_et_al_wm_hbm_2011.pdf	678 KB	281	Any	pdf
	woodward_func_connec_wm.pdf	1.72 MB	714	Any	pdf

Figure 3: "See All Files" download page

After you have downloaded the CPCA graphical user interface (GUI):

- Start MATLAB
- In the *File* dropdown menu, select *Set Path,* then choose the *Add with subfolders* option and select the CPCA directory you created.

- Save your new path (it doesn't matter where CPCA appears in your path).
- To start the CPCA GUI, type '**cpca**' at the MATLAB prompt.

II.IV. Installing the example data

There are several example data files provided on the fmri-cpca website (http://www.nitrc.org/frs/?group_id=203). Downloading these files, via the methods described in the previous section, will allow you to work through the examples provided throughout this manual.

- Extract the selected zip file to a folder of your choosing. The extraction will create a folder in this directory, such as 'C:\cpca_fMRI\Tutorial_Data_Set.'

Table 1: Example data to download	
Example data file	Purpose
Tutorial_Data_Set	Get Started using the CPCA GUI on an existing Z matrix. View and Interpret the results of your analysis.
example_data_Single_Subject	Create Z & G matrices for a single subject with 1 run
example_data_Multiple_Subjects	Create Z & G matrices for multiple subjects with 1 run
example_data_Multiple_Subject_Runs	Create Z & G matrices for multiple subjects with multiple runs
example_data_Multiple_Groups_Subjects_Runs	Create Z & G matrices for two groups of subjects. Each group contains two subjects with 2 runs each.

III. Tutorial: Getting Started

III.I. Purpose of this tutorial

The purpose of this tutorial is to guide you through your first full CPCA analysis on a given set of data. It is intended to demonstrate all of CPCA's features and provide an example of how to interpret the results of a CPCA analysis.

This tutorial is an expanded version of the *condensed* tutorial available for download from the fMRI-CPCA web site: Getting Started: An fMRI-CPCA Tutorial.pdf. The procedures carried out are similar but more extensive explanations are offered here.

III.II. Prerequisites

This tutorial assumes that you have the following:

1. The required software installed your computer (*see* #Software requirements).
2. CPCA installed on your computer (*see* II.III. CPCA installation).
3. **'example_data_Multiple_Groups_Subjects_Runs'** downloaded and extracted (*see* II.IV. Installing the example data).
4. Experience using Windows or Linux (depending on which platform you are using).

III.III. Tutorial Data Set

III.III.1. The Files

The **'example_data_Multiple_Groups_Subjects_Runs.zip'** file contains three matrices needed to carry out any CPCA analysis: a Z matrix, a G matrix and a mask image.

1. Z matrix (file Z.mat):

   Contains the series of fMRI scans for all subjects, stacked vertically. For this tutorial example Z, the scans have undergone typical SPM pre-processing steps (realigned, spatially normalized, and smoothed).
2. Mask (files mask.img, image.hdr):

   One mask.img that fits all subjects simultaneously must be prepared. This requires all subjects' data to be normalized to a common template. The brain voxels to be included in the analysis are determined by this mask.

3. G matrix (file G.mat):

Contains the model of the BOLD signal changes associated with the cognitive task timing over all scans. For this example, a finite impulse (FIR) response model was used.

The '**example_data_Multiple_Groups_Subjects_Runs.zip**' file also contains '**shapes.mat**' and **'timing_onsets.txt'** files which are required at more advanced stages of analysis (i.e. rotation methods, including hrfmax, and G matrix creation).

III.III.2. The Data

The example Z.mat contains scans for four subjects who completed two runs of a verbal working memory task. Each run lasted 214 scans (TR=3) for a total of 1712 scans.

The 4 X 4 X 4 voxel mask for these subjects contains 29254 in-brain voxels. Thus the dimensions of Z are 29254 x 1712.

This verbal working memory task consisted of encoding, maintenance, and retrieval phases for four different memory load conditions. During a single trial, subjects were presented with a string of two, four, six or eight different letters which they were instructed to remember over a short delay. Subjects were required to indicate if a letter presented after the delay was the same as one of the remembered letters or different from all of the remembered letters.

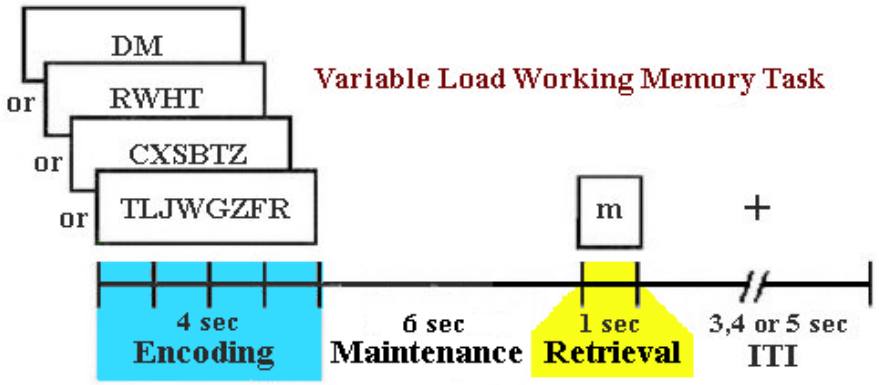


Figure 4: Tutorial data set, experimental task

Completing CPCA on this data set will allow neural systems underlying working memory to be identified. For each system identified, a functional network of the functionally connected regions will be created along with an associated plot of the estimated HDR for each memory load condition.

A more extensive explanation of the experimental procedures used in collecting this data can be found in the paper by Cairo, Woodward, & Ngan (2006). Two papers that describe the

method and results of an fMRI-cpca analysis on a similar data set are available for download on the cpca website (Woodward, Cairo, Ruff, Takane, Hunter, & Ngan, 2006; Metzak, Feredoes, Takane, Wang, Weinstein, Cairo, Ngan & Woodward, 2010).

III.III.3. The Model

CPCA requires information about both the data (Z) and the model (G).

In fMRI, the data includes images of actual BOLD signal changes over time, and the model provides a prediction of how the BOLD signal changes over time relative to the experimental task. For this example, a finite impulse response was modeled for each scan of the working memory task. The dimensions of the example **G** are 1712 x 128.

Because it provides a model of BOLD signal change, the rows of G matrix contain same number of rows as the time series (1712 scans in this example). For a FIR model, the columns of a G matrix will be equal to the number of experimental conditions (n) times the count of time windows (b), times the number of subjects per group (s), times the number of groups (g).

cols = (n * b) * s * g

In the example, four conditions were modeled, one for each memory load: 2 letters, 4 letters, 6 letters and 8 letters. A time window of 8 was selected to allow the derived predictor weights to be plotted over 24 seconds (TR of 3 sec * 8 windows) which covers the approximate duration of a hemodynamic response for this experiment (± 20 seconds). Therefore, in the example G matrix, the **n** = 4, **b** =8, **s** =2, **g** = 2, thus:

cols = (4 * 8) * 2 * 2

cols = 128

Running a CPCA Analysis

I. Create the Z matrix

The Z matrix contains preprocessed BOLD signal from 1712 fMRI scans (2 subjects x 2 runs x 214 scans per subject) collected during a working memory task. The Z matrix must is typically normalized to ensure that all voxels for all subjects have standardized variance. In order to create and normalize a new Z matrix, a File List containing information on the location of scan data must be created first.

1. Create the 'Subject Scans File List'

The **'Subject Scan Files List'** directs the application to the fMRI scans to be analyzed. It is created in a structure that allows the scans associated with each group, subject and run to be differentiated. The following steps are involved in the creation of this scan list which will later be used to create the Z matrix.

a. Ensure the directory structure is correct.

Before creating the **'Subject Scans File List'** you must ensure that your directory structure that contains the fMRI scans is organized appropriately. When analyzing multiple groups, CPCA is designed to have the directory format as follows:

```
{base directory}
        {group}
          {participant}
{run}  {run}
          {participant}
            {run}  {run}
            ...
          {participant}
            {run}  {run}

{group}
          {participant}
            {run}  {run}
          {participant}
            {run}  {run}
            ...
          {participant}
            {run}  {run}
```

b. Click the **'Create List'** button, located on the *System Information* panel to open the **'Scan File List Creation'** window.

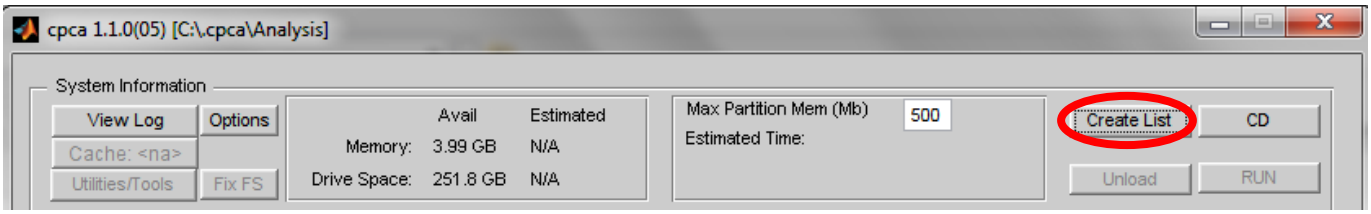


Figure 5: *Create List* button

c. Press the directory **'Select'** button, located at the top right of the **'Scan File List Creation'** window to navigate to and select the root directory that contains the fMRI scans: **example_data_Multiple_Groups_Subjects_Runs.'** Press **'OK.'**

*** The gui will automatically attempt to categorize the data into groups, subjects, and runs based on the number of subdirectories. Ensure that the boxes at the bottom of the three columns correctly match the data (groups, subjects, runs). If you are working with multiple groups, subjects, and runs (as we are in this example), the gui will not allow you to change how these categories are selected. Therefore, if you see an error, ensure you have organized your directories correctly (see section……above)

d. Choose and edit a wildcard

In this example, a common root for image names is 'fsn*.img.' Since this is not one of the wildcards listed in the **'Sample File Specification'** window, we will have to edit one of the file names.

The **'Sample File Specification'** window will list the file names (wildcards) within a given base directory tree and will show the number of files (in brackets) that match the specific wildcard specification.

The **'Use File Specification'** window displays the wildcard name that will be used to generate the scan file list.

i. Double click on 'fsnwmla_F*.img(214)' in the Sample File Specification window. This will copy the wildcard name to the 'Use File Specification' window.

ii. In the Use File Specification box, edit the wildcard name so that 'fsn*.img' is displayed.

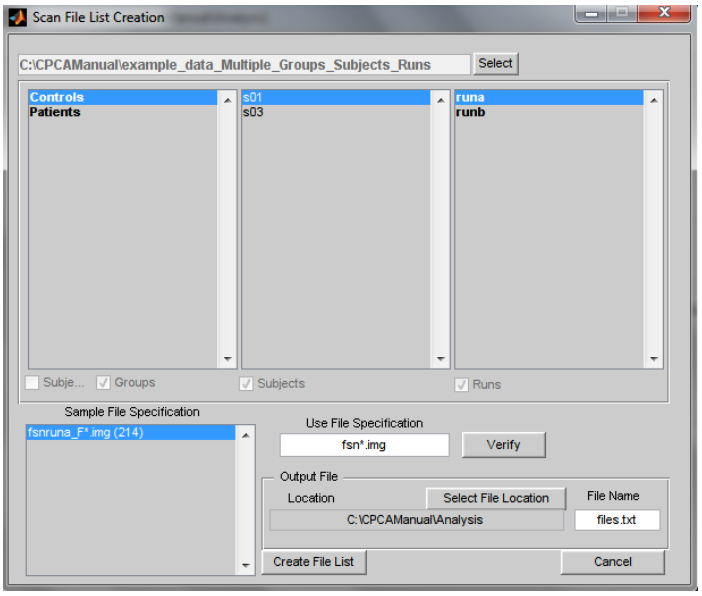


Figure 6: *Scan File List Creation* window

e. Verify the Scan List

Click the '**Verify**' button to the right of the '**Use File Specification**' text box to display **'File Specification Results'** ('**fsp_results**'). All the scan files contained in a directory that match the File Specification will be displayed. An asterisk (*) beside a file name indicates a possible error, such as the file not in its expected position. When you are finished viewing the results, close the window.

Note: It is likely that that Sample File Specification names may differ when different folders within the base directories are selected. The '**File Specification'** window will update based on the initial directory scanned so may not apply to all files. Thus the '**File Specification'** must be renamed to a common root.

Remember, CPCA has difficulty reading directory names that contain empty spaces.

*** **Example Showing Errors:**

In this example, file names have been deliberately altered to demonstrate mistakes. S01 had the underscores (_) in two files changed to hyphens
(-) which can be a common typo.

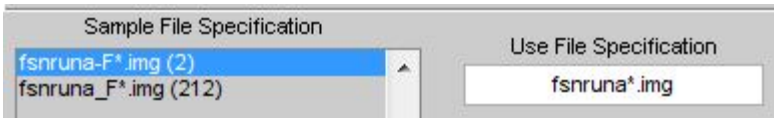


Figure 7: Common mistake in file names

It is apparent from the '**Sample File Specification'** that two files are different from the others. In the event this is overlooked, when '**Verify**' is clicked to display **fsp_results**, any inconsistencies in the data will be marked with asterisks (*). Here, due to the two files with hyphens (-) vs. underscores (_), the overall order of scans was affected. By checking the **fsp_results**, one can make sure the data is free of naming errors prior to continuing.

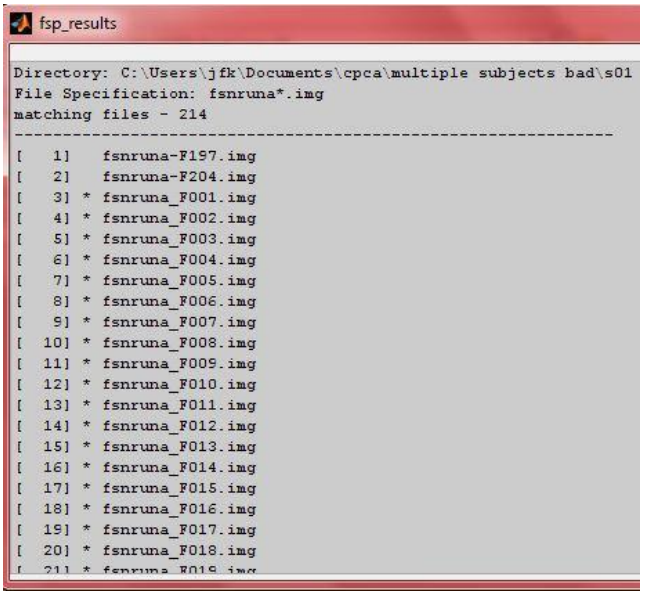


Figure 8: fsp_results of files with naming errors

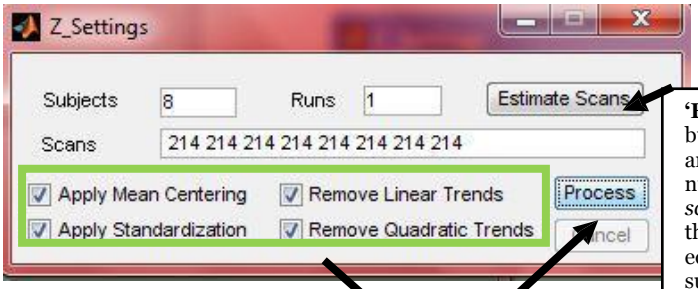
f. Create the file list

Click the **'Create File List'** button to save the file list. Output will be written as **files.txt** (this can be edited in the File Name box). If you want to save the file list to a directory other than the working directory, you can change the output location by using the **'Select File Location'** button. A pop-up window will appear showing what information is contained in the newly created file. Close the window after viewing.

2. Select Z or file list

Z contains the time series of fMRI scans for all subjects (rows) for all voxels of interest in the brain (columns). For fMRI data, Z can be referred to as the Activation matrix. Prior to creating Z, the scans for each subject must be preprocessed: realigned, filtered, normalized and smoothed.

The Z settings window will open automatically when an already-created Z file is loaded. The number of **Subjects**, **Runs** and **Scans** must be edited to match with your data. For this tutorial however, we will be creating a new Z matrix.



'Estimate Scans' button: Pressing this button after the correct number of **Subjects** and **Runs** have been entered will estimate the number of scans per subject (*total number of scans / [subjects x runs]*) and automatically fill the **Scans** box. This estimate may have to be edited if the scans are not equal between subjects and runs.

Figure 9: Z Settings

Normalization Options:
The checked procedures will be applied when the **'Process'** button is activated.

a. In the **'Subjects'** panel, click **'Select'** and navigate to the correct directory to select either your files.txt if creating a new Z or ZInfo.mat to use a previously created Z. For this example, select the **'files.txt'** file.

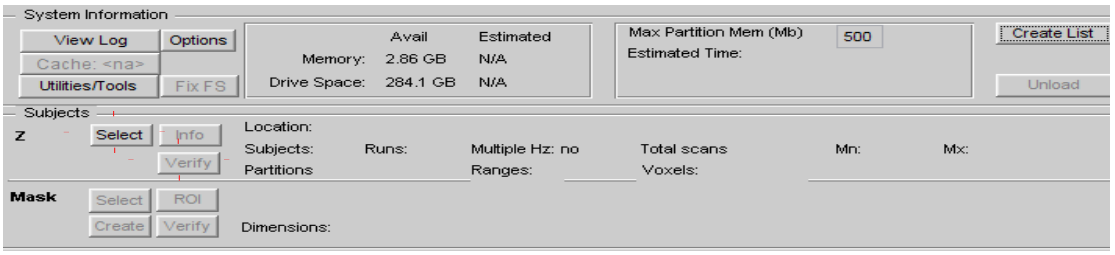


Figure 10: *Subjects* panel, *Select* button

Once the **'Subject Scan Files List'** is loaded the ***Subjects*** panel will be updated with the number of subjects, runs, and scans in the subjects data and the location of the root directory. '**Exists'** is listed on the right hand side panel, indicating that the fMRI scan images listed in files.txt actually exist.

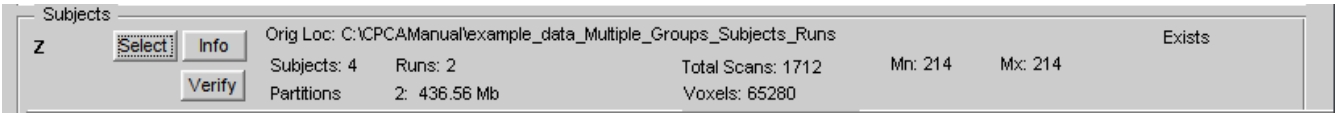


Figure 11: *Subjects* panel after File list was loaded

b. Verify the scans

In the ***Subjects*** panel, click **'Verify.'** The **'scan_verification'** window will be displayed. Click **'Verify'** in this window to scan Z for errors. When the verification is complete, the **'scan_verification'** window give a count of the number of good files and the number of files containing errors (you can select if you would like to view only good or error containing files by selecting the appropriate button in the window).

Click **'Done'** when finished viewing.

*** Z verify cannot tell if the data is good, just if the files are readable (i.e. no errors on disk). This is particularly useful for files that have been transferred or copied from another source or location.

3. Create or select a mask

*** If you are using an ROI (region of interest) mask instead of a full mask for all brain areas, use the **'ROI'** button to navigate to and select your ROI file. If you have selected multiple files and want to combine multiple ROI's, click on the '**Combine'** button.

The mask must be loaded to indicate how the computed voxel information should be written to a brain image. This panel will become active after a Z matrix is loaded.

To use a previously created mask image, click **'Select'** and navigate to your **'mask.img'** file.

a. To create a new mask image, click **'Create.'** Ensure all information in the resulting pop-up window is correct and that the boxes next to **'All Subjects'** and **'All Runs'** are checked. If desired, you can edit the file name of the mask. For this example change the file name to **'mask2.img'** to avoid overwriting the mask that was downloaded with the sample data.

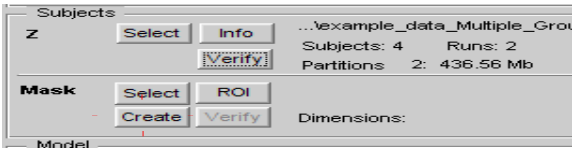


Figure 12: *Create* mask button

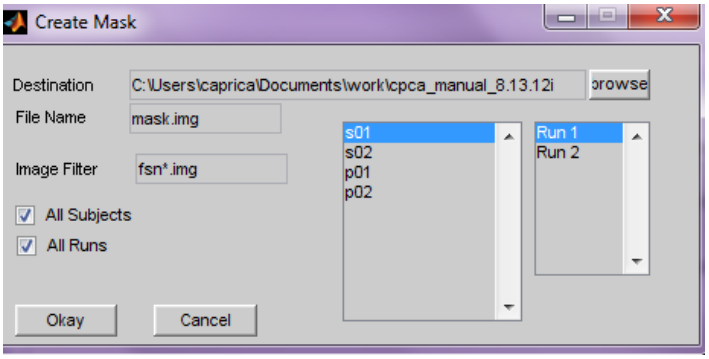


Figure 13: *Create Mask* window

b. Press **'OK'**. A window will appear showing the progress of the mask creation.

Once your mask is created/selected, the ***Subjects*** information panel will update with the location of the mask image and the image slice dimensions (x, y and z), with the voxel size in brackets.

4. Verify the mask

Clicking on the **'Verify'** button in the ***Mask*** panel will apply the mask to all subject scans, verifying that the brain regions are consistent across all subjects. Once clicked, the **'mask_verification'** window will appear.

Figure 14: Mask *Verify* button

a. If you want to verify the mask for all subjects, ensure the **'All'** box at the bottom of the window is checked. Else, highlight the subjects you wish to verify.

b. Click the **'Verify'** button in the **'mask_verification'** window to begin the verification process. Any errors will be indicated in this window. If there is an error, a **'Create New Mask'** button will appear. In *Figure 14* below, it is apparent that s01 is error-free (zero: 0), whereas s02 contains errors (zero: 2).

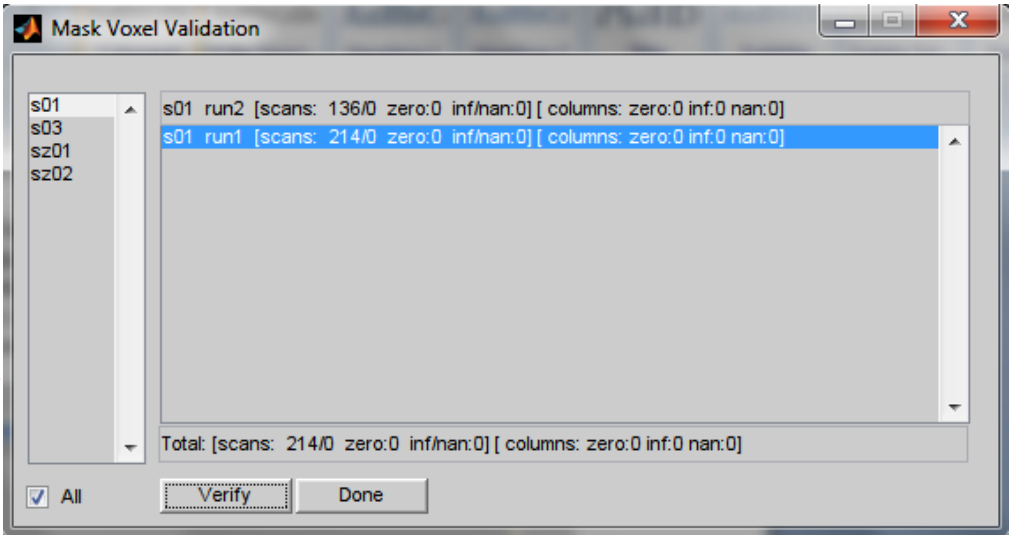


Figure 15: *Mask Verification* window, *Create New Mask* button

c. If there are errors in the mask, Click '**Create New Mask.**' This will collect the voxel columns that contain errors and pull them out. Mask verification can remove voxels showing no variance before application. The default filename will be '**new_mask.img'**. This can be adjusted in the '**Filename**' window to any name you choose.

d. Click '**Done**' to return to the main processing panel and select the newly created mask.

Note: The most recent verified data will be saved so you will not need to work through all subjects again.

❖ At this point, you can either continue on to the next step (Normalize Z) returning after normalization has been completed to create and apply the G matrix. For this example, we will continue on to normalize the Z matrix.

5. Standardize Z

The **'Z'** button will be activated, indicated by the button turning green and the default normalization options being selected. (The **'RUN'** button will also have become active).

a. Select normalization options (see #Appendix_section_III for more information on normalization options).

   a. Ensure that **'Create Z'**, **'Regression Covariates'** (**'Linear'** and **'Quadratic'**) are all selected.

   b. The **'Head Movement'** box will activate if there is an rp{…}.txt file for each subject or you can select **'User Defined Covariates'** if you have your own files for head movement. For this example we will not be regressing out any head movement data.

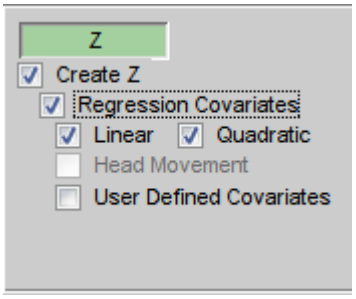


Figure 16: Z normalization options

b. Click the **'RUN'** button located in the top, right-hand corner of the 'System Information' panel. A **'CPCA Process'** window will open to inform you of the status of the processing. **Do not close** the progress window yourself. Once progress has gone to completion, the window will close by itself.

When the Z matrix/matrices have been created, the run and completion information will appear on the MATLAB console screen. The Subject panel will be updated to indicate that Z has been normalized (mean centered and standardized). The following Z matrix files will have been written to your working directory: ZInfo.mat, Z1.mat, Z2.mat, Z3.mat, Z4.mat (segmenting Z in this manner allows you to use as large of a data set as you wish).

Now that the Z matrix has been created, when running future analysis with the same data set you can select the **'ZInfo.mat'** file when loading the Z matrix (see #1.2_Select_Z_or_file_list).

II. Create or select a Fir G Matrix

(To create an HRF G matrix, see #Appendix_section_VII)

The G matrix is the Design matrix. In fMRI data, G contains a model of the predicted BOLD signal changes (columns) caused by stimulus presentation timing over all fMRI scans (rows). In traditional univariate analysis a hemodynamic response model (HRF) is used, but with fMRI-CPCA a finite impulse response (FIR) model is typically used to allow deconvolution. This article provides a clear overview of both types:

Lindquist, M. & Wager, T.D. (2007). Validity and power in hemodynamic response modeling: a comparison study and a new approach. *Human Brain Mapping, 28(8),* 764-784.

The G matrix contains the predictor model that will be applied to the Z matrix (subject data) to constrain the variance analyzed in Z to that related to stimulus presentation.

The following instructions will guide you through creating a FIR G matrix using a previously created timing onsets file. When analyzing your own data, you must first create a timing onsets file and save it as a .txt document.

The timing onsets link the timing of experimental procedures to the corresponding fMRI time series. Thus, for each subject the timing onsets associate each experimental condition with the correct scan. e *Timing Onsets File* should have a **.txt** or **.m** extension (created using SPM GUI). The *format* of the Timing Onsets file is irrelevant (e.g. a spreadsheet containing the onsets can be easily exported to a text file with no problems). However, the *order* in which the onsets are listed is critical.

To use a previously made G matrix, navigate to the directory where the matrix is saved and select.

1. In the Model panel, click the **'Create'** button located beside the 'G' indicator to create a G Matrix. This will open the G matrix creation window.

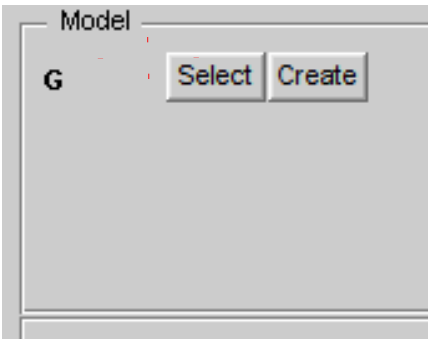


Figure 17: *Create* G button

2. Define Runs and Conditions

   a. To define the conditions that are applied to subject runs, click the **'Runs and Conditions'** button.

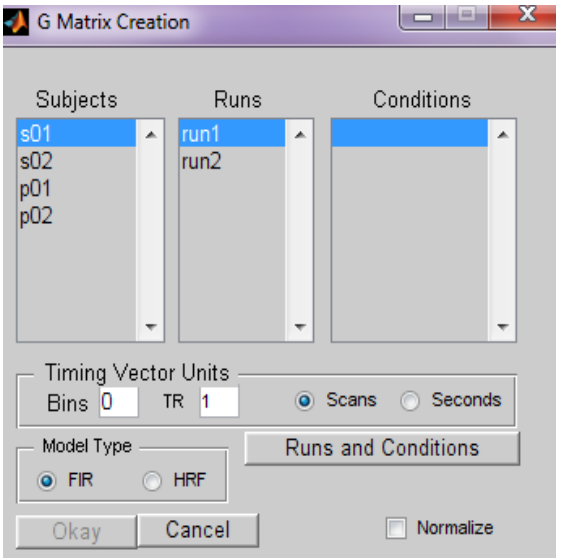


Figure 18: *Runs and Conditions* button

   b. In the box located at the top right add type in the condition names (2_letters, 4_letters, 6_letters, 8_letters) and press 'Enter' or click **'Add'** after each entry.

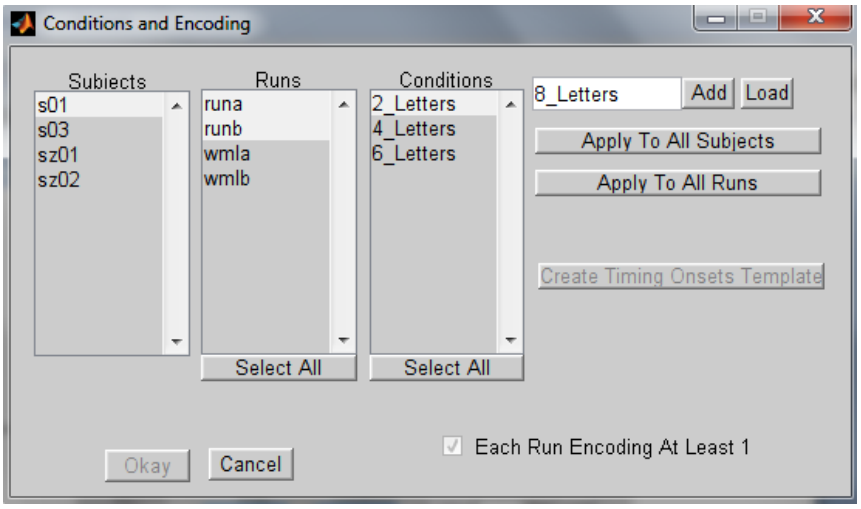


Figure 19: *Runs and Conditions* window

If you make a typing mistake, you can remove the mistyped condition from the **'Conditions'** box by highlighting it and pressing the delete key on your keyboard.

c. Highlight all **'Runs'** and **'Conditions'** by clicking the **'Select All'** buttons located at the bottom of each box.

d. Click the **'Apply to All Subjects'** and **'Apply to All Runs'** buttons. Ensure the **'Each Run Encoding at Least 1'** box is checked.

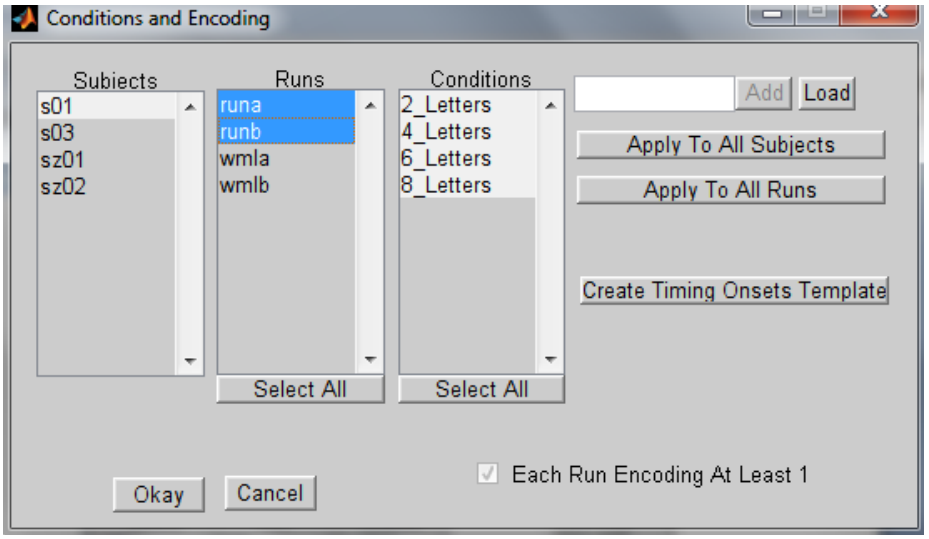


Figure 20: *Runs and Conditions* window, selection buttons

*** If there is a subject that does not encode every condition in every run, you can edit which subjects/runs/conditions to be applied by highlighting only those events you want to be included. The figure below shows only 2 conditions (4_letters and 8_letters) being selected for s01, runb.

When at least 1 condition for each run (over all subjects) has been selected, the **'Each Run Encoding At Least 1'** box will be automatically checked.

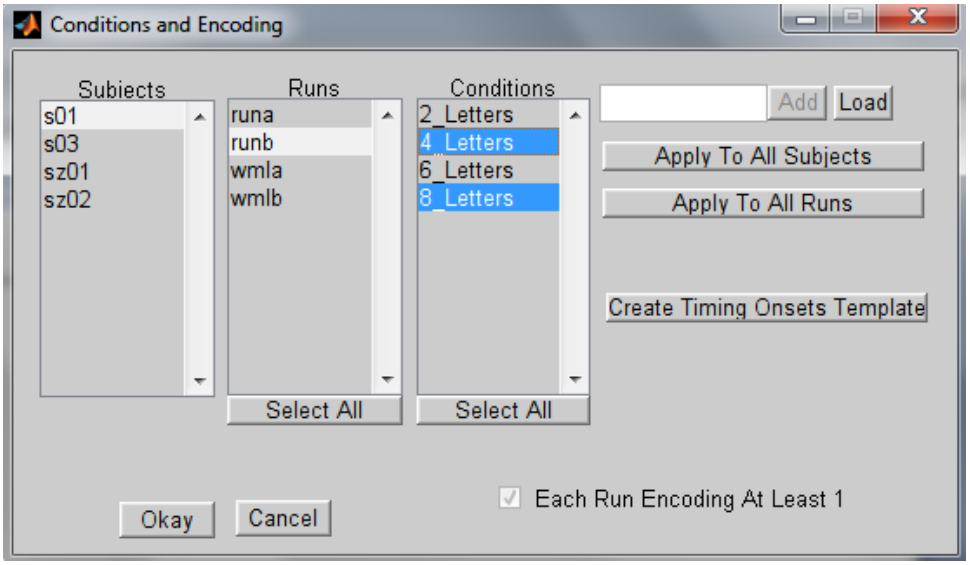


Figure 21: ***Runs and Conditions*** **window, selecting only some conditions**

e. Click the **'Create Timing Onsets'** button. An editor will appear displaying the timing onsets template text file.

   ***** The sequence of timing onset definitions is critical.** All timing onsets must be prepared in the order displayed in this file. Timing onsets may be inserted directly into this file or imported from a separate text file (see step 8 below for instructions on how to import onsets). All timing onsets imported from a separate text file must be prepared in the order displayed in the onsets template. Any onset condition names in an imported text file WILL BE IGNORED so you can label these any way you choose (or not label them at all).

   Close the editor when you have verified the template is correct.

f. In the **'Runs and Conditions'** window, click **'OK'** to return to the **'G Matrix Creation'** window.

3. Edit Bins, TR, and timing units

   a. Change the number of bins by entering **'8'** in the **'Bins'** box.
   b. Change the timing rate by entering **'3'** in the **'TR'** box.
   c. Ensure the timing units selected are **'Scans'** (if your timing onsets are in seconds then select **'Seconds'**).
   d. Ensure the **'Model Type'** selected is **'FIR'**.

      The number of bins you wish to use will be based on the TR. Generally, we expect an hrf response of approximately 20 seconds. Therefore, in this example, we have selected 8 bins to give a total time of 24 seconds (8 bins * 3sec/bin -> TR = 24sec).

4. Import timing onsets

    a. Click on **'Import Onsets List'**, navigate to the directory where the text file is saved (for this example, a previously created **'timing_onsets.txt'** file is included in the example data you downloaded), and open the file. An editor will appear showing the previously created template with the imported data inserted into it.

    b. Ensure the information is correct and then close the editor.

    Now there are *two* new files in your working directory. The first is the *imported* Timing Onsets text file and the second is the *template* Timing Onsets text file.

5. Check the box next to **'Normalize'** so that the G matrix is normalized during the creation process.

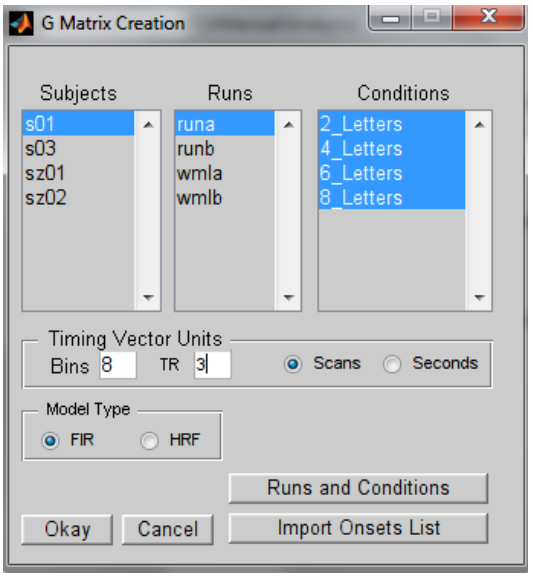


Figure 22: *G Matrix Creation* window, *Import Onsets List* and *Normalize* buttons

6. Click **'OK'** on the **'G Matrix Creation'** window (this button will activate after the timing onsets have been imported). The window will automatically close when the G matrix has been created.

    - Now that the G matrix has been created, the ***G Matrix*** panel should be updated with a small check mark beside the **'Select'** button and the size of the G matrix.

    - The G matrix must contain the same number of rows as the total number of scans.

    - A 'Gsegs' directory will be created in your working directory. The 'Gsegs' directory will contain one file for every subject.

- In the working directory, a file called G.mat will be created, containing the information on where to find the subject G segments.

III. Apply the G matrix to the Z matrix

Once the Z matrix (data) has been created, normalized and loaded and the G model has been created and loaded, analytic processing can be begun. The first step is to apply the model to the data (apply G to Z) to partition out the variability in the data that can be explained by the model, which will create GZ. Once GZ has been created, solutions of various types can be applied to GZ and components extracted.

1. Press the **'G'** button, which will become green, to activate its options.
2. Check the **'Regress G'** box.

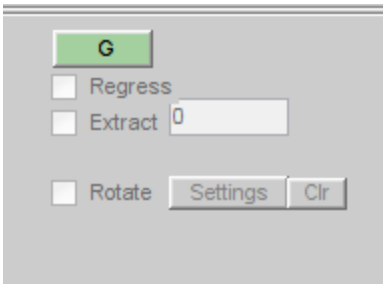


Figure 23: *Regress* G box

3. Press the **'Run'** button again (located in the upper right hand corner of the gui).

   A **'CPCA Process'** window will open to inform you of the status of the processing. When the application has completed its calculations, the completion information will appear on the MATLAB console screen. As well, the text **"Completed"** will appear next to the **'Apply G'** box.

IV. Extract Components

1. Choose the number of components to extract by examining the scree plot.

   *** For this tutorial, you will only be extracting **two** components, in order to keep it succinct. However, the method used to determine the number of components to extract will be introduced and briefly explored.

   After the G matrix has been applied to Z, the **'Scree'** button in the ***G Model*** panel will become active. Press the **'Scree'** button to open a new window with the plot of the singular values, shown in the following figure. The singular values show the variation between components, which is useful in determining how many components are statistically significant.

Each point on the scree plot represents a component that can be extracted. The greater the value on the y-axis, the more variance that component accounts for. When points correspond to a low y-value, they should not be extracted as they could represent noise etc., and potentially compromise the data. To interpret a scree plot, work your way from right (error) to left (increasingly strong signal), and when a subjectively important "jump" in the curve occurs, you may consider keeping the component that causes that jump and all those above it. In the scree plot shown above, a three-component solution appears appropriate, due to the subjectively important jump (where the curve becomes more vertical) from the 4th to the 3rd component. Following this, significance testing is necessary to confirm the appropriate number of components were extracted, but this will be covered below. Interpretation of scree plots is more an art than a science, but has a long history in the area of component analysis (starting with Cattell, 1966).

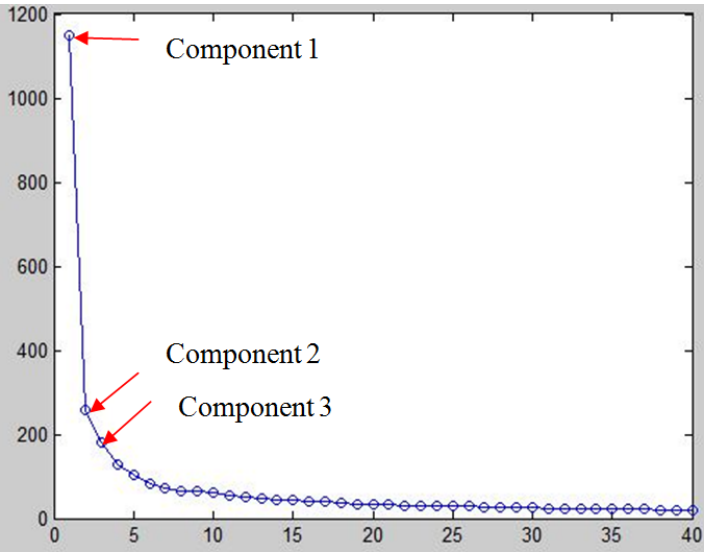


Figure 24: Scree Plot showing important components to extract

2. Return to the ***Processing*** panel and press the **'G'** button to activate the boxes below it.

3. Check the box labeled **'Extract'** and enter **'2'** in the box to the right of it.

   In a full analysis, you would determine the number of components to enter based on your decision after viewing the scree plot i.e. For a full analysis, you would enter **'3'** in the box to the right of the **'Extract'** box. You also have the option of extracting a multiple number of components in a single run by entering all the numbers you wish to extract in the box to the right of the **'Extract'** box, for example you could enter **'3 4 5'**. As aforementioned, for simplicity reasons, we will only extract two components in the tutorial.

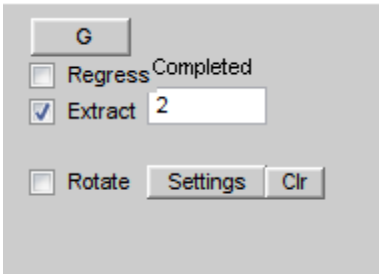


Figure 25: *Extract* components button

4. Press the **'Run'** button again. A **'CPCA Process'** window will open to inform you of the status of the processing. When the application has completed its calculations, the text "**CPCA Processing Completed**" will appear on the MATLAB console screen. At this point, the **'Stats'** button in the ***G Model*** panel will become active (see #Appendix_section_V for instructions on interpreting results).

V. Rotate Components

Rotating components allows for a different view of the components that may make interpretation easier. As the optimal methods for rotation are currently under empirical investigation, the type of rotation to use is up to the discretion of the experimenter. We recommend using a rotation method that results in clear hemodynamic response shapes in the predictor weights.

As well as providing an unrotated solution, the GUI can also perform the following rotations:

- Varimax (orthogonal)
- HRFmax* (orthogonal or oblique)-Currently not available as it still being tested

***** The HRFmax methodology is described in our published work (Metzak et al. 2011), and requires creation of a set of target hemodynamic response shapes, which is possible in the GUI (as outlined in the steps below).**

A previously created shapes file can be loaded or a new shape can be created for Varimax rotation. In this example, we will be creating a new shapes file to be used in an hrfmax rotation.

It is possible to extract multiple rotation solutions in one analysis. In the following example, we will be performing only one rotation.

1. If the **'Extract'** box is still checked, uncheck it.
2. Click on the **'Settings'** button to open the **'G Rotation Settings'** window.

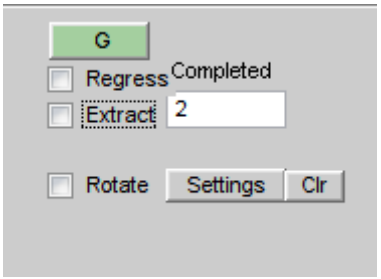


Figure 26: Rotation *Settings* button

1. Select a rotation method

   The **'Active Rotation List'** displays the rotations that will be applied when the **'RUN'** button is selected. In this example, we will be running a varmiax rotation following the steps below.

   *** If you are running a varimax rotation, you do not need to create a shapes.mat file.

   Click the **'Method'** dropdown menu and select **'varimax'**.

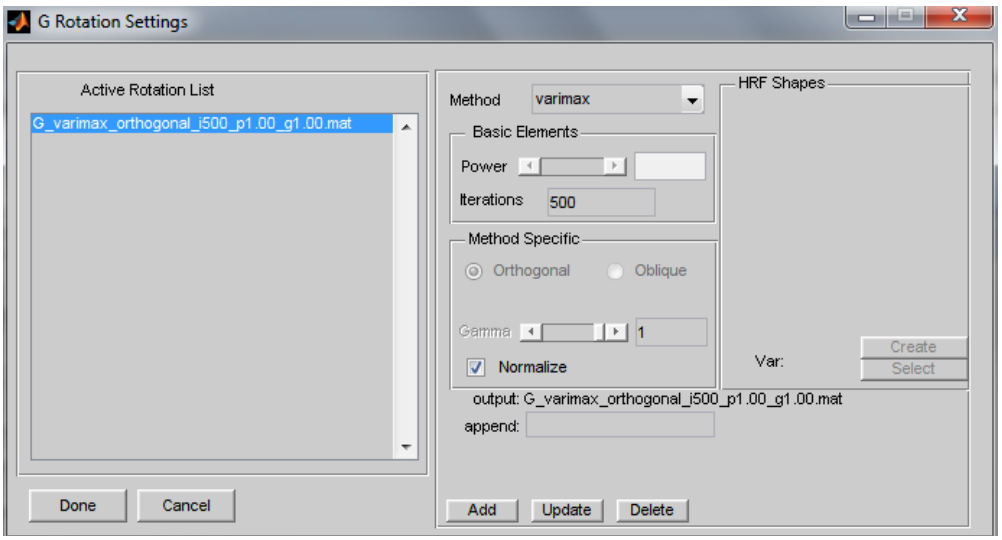


Figure 27: *G Rotation Settings* window, select rotation *Method*

Click **'Done'** to return to the gui mainscreen.

2. Ensure the green **'G'** button is pressed and select the checkbox labeled **'Rotate.'**

   Note: Clicking the **'Clr'** button will unload any settings you have made.

3. Click **'Run'**.

   A timer will be displayed for the duration of the rotation so you can ensure the program is still working and that progress is being made. Updates will also appear in the matlab window.

   NOTE: The **'Extract'** and **'Rotate'** functions can be run at same time:

1. Checkmark the **'Extract'** box and enter the number of components to extract.
2. Checkmark the **'Rotate'** box and set up your rotation settings.
3. Click **'Run'**.

VI. View Results

A previously processed Z matrix file (ZInfo.mat) can be loaded and the **'Stats'** button selected without previously completed analyses having to be rerun. This button appears on the ***Model*** panel once processing is complete.

The Statistics window allows the user to view the general statistical results for the solution selected in the top left hand corner selection box. Component specific statistical results can be viewed on the right hand side of the window. Via this window the user can also plot components, flip components, set plot options, plot singular values and complete beta checks.

1. In the ***Model*** panel, click the **'Stats'** button (the statistics button will only show if you have extracted or rotated data in the current directory). This will open the multivariate statistics window which displays the general statistics of the selected process (e.g. sum of squares, percentages, correlation coefficients, etc, based on the predictors).

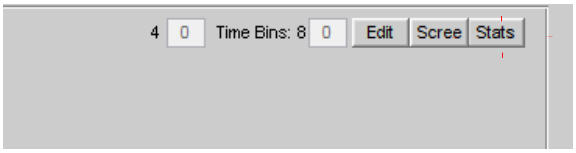


Figure 28: *Model* panel, *Stats* button

2. Select a rotation method (for this example select **'varimax orthogonal'**) from top left drop down menu. (The total number of components that were extracted in the selected analysis are shown in brackets).

3. Beta Check and Component Flipping

   To ensure that a given component's HDR plot is oriented in the correct y-direction and its activation map is displayed correctly as either activation or deactivation, it is necessary to first review the Beta time course associated with that component. This has to be done because the sign (positive or negative) associated with the predictor weights and component loadings during SVD analytic processing is arbitrary. If necessary, the sign of the predictor weights and component loadings can be switched or flipped.

   a. Select a component from the drop down menu on the left below the '**Varimax orthogonal**' (for this example, select component **'1'**) to view the statistics for that component of the solution.

      You will see red text warning: **'This component may need to be flipped.'**

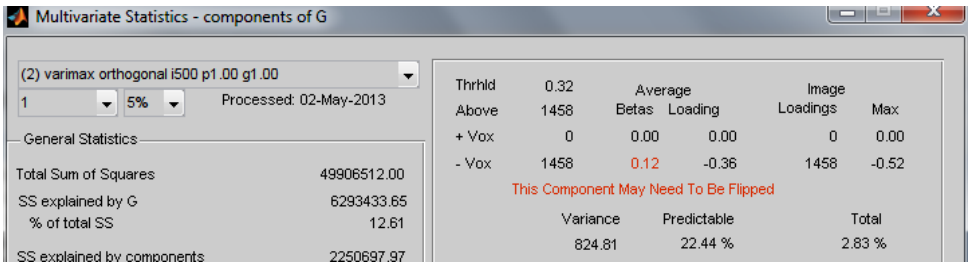


Figure 29: *This Component May Need To Be Flipped* warning text

   b. Beta Check

      Press the **'Beta Check'** button to examine the hdr plots for the selected component.

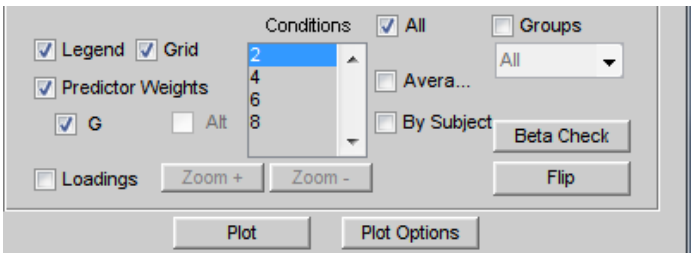


Figure 30: *Beta Check* button

Examples of How to Interpret Beta Check Results:

Example 1:

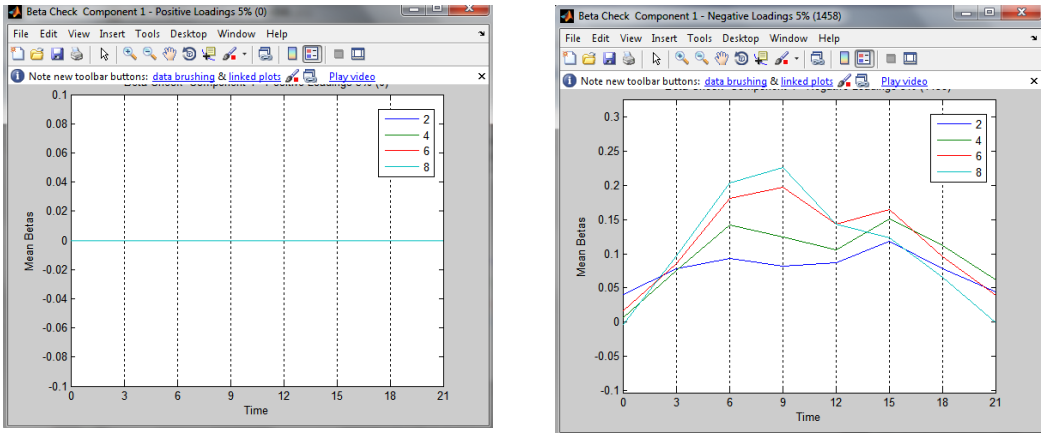


Figure 31: Beta check component 1, positive and negative loadings

The positive loadings can be disregarded because there are none. The sign of the majority of the negative loadings should be going in the negative direction and that is not the case here. In this instance the peak positive value for the Y (vertical) axis is 0.2, and the peak negative value is 0. Therefore, Component 1 **does need to be flipped**.

When component 1 is flipped the sign of the y values on the plot will reverse. In addition the positive and negative functional network images will be rewritten to the image output directory with the previously negative loadings becoming positive and vice versa.

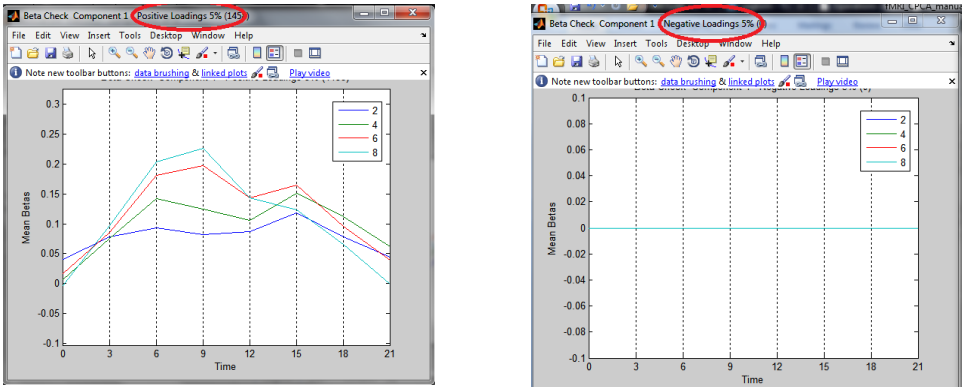


Figure 32. Beta check after Component 1 is flipped

Example 2:

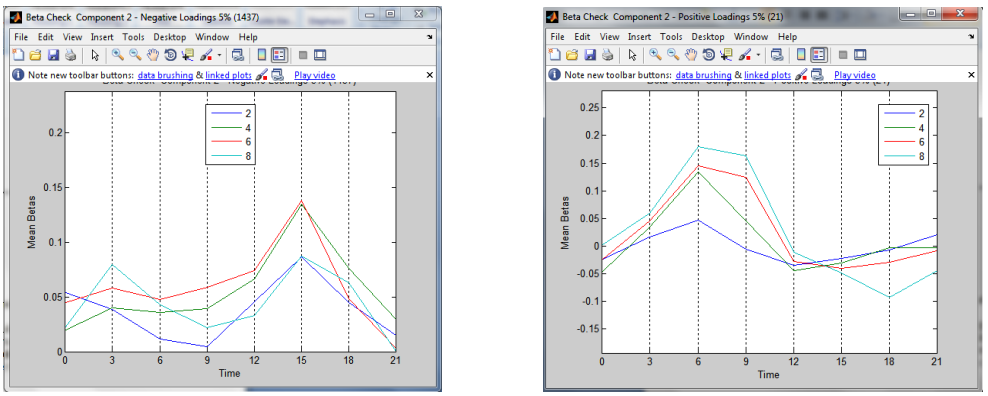


Figure 33: Beta check component 2, negative and positive loadings

The sign of the majority of the positive loadings should be going in the positive direction and the majority of the negative loadings should be going in the negative direction. Since this is the case here, Component 2 **does not need to be flipped** (reminder: click the **'Plot Component'** button to see the Estimated HDR Component plot)**.**

c. Flip Components

Flip component 1 by pressing the **'Flip Component(s)'** button. The **'flipping data'** window will appear throughout the process.

When finished, the graph in the Multivariate Statistics window will be updated and the red warning text will be gone. As well, any associated output (images, threshold values) will be updated and saved automatically (older data will be overwritten).

Note: On some platforms, the graph may not update properly, so to refresh the flipped predictor weights select the '1' from component drop-down menu again.

Note: On some platforms, clicking the '**Flip**' button will cause Matlab to crash. To flip data in this case, open up Matlab command window and type in '**edit MvsfMRI.m'.** Run the '**Flip**' using debug mode by searching for the word 'flip' and clicking on these corresponding lines: 1271, 1286, and, 1289.

Note: The application will only flip one component at a time if any component is selected. To flip all components, deselect the current component by choosing '<select>' in the component drop-down menu.

4. Additional options for viewing results in the Multivariate Statistics window

1. Saving plots

For each component extracted, predictor weights are computed and saved. These mean predictor weights can then be plotted (using any package such as Excel, SPSS, or via the fMRI-CPCA GUI interface).

To create and view a plot of the estimated Varimax for the two extracted components using the GUI:

a. Ensure you have the **'Multivariate Statistics'** window open (by clicking the **'Stats'** button on the gui mainscreen).

b. Ensure you have selected a rotation method and component to view (using the drop down menus at the top of the **'Multivariate Statistics'** window).

c. You can select or deselect **'Legend,'** **'Grid,'** and **'Predictor Weights'** to appear or not appear on the plot.

d. Selecting the **'Loadings'** box will show the distribution of the Loadings. These are sorted, and displayed with markings for the top 5, 10, and 20% thresholds.

e. You can select specific components to plot/view by selecting a component from the **'Component'** drop down menu.

f. You can select whether to plot/view predictor weights averaged over groups, one group specifically, or all groups separately from the drop down menu next to the **'Component'** drop down menu.

g. The **'Condition'** menu allows you to select which conditions to view. **'All'** plots all the conditions as separate lines on one graph. **'Average'** plots the average of all conditions as one line.

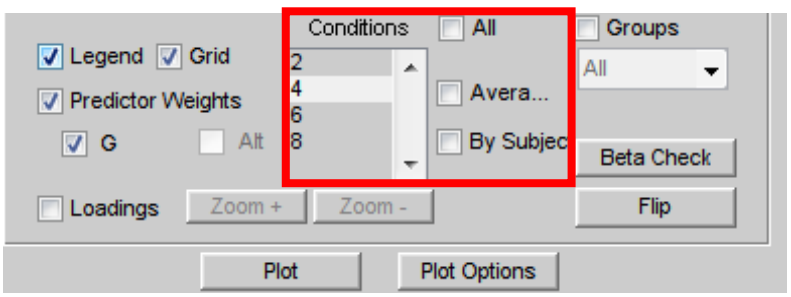


Figure 34: ***Multivariate Statistics*** **window, select plot conditions**

h. The next menu allows you to select which groups (control or patients) you would like to view (this is helpful when comparing two different groups). **'Groups'** plots values for each group separately on the same plot.

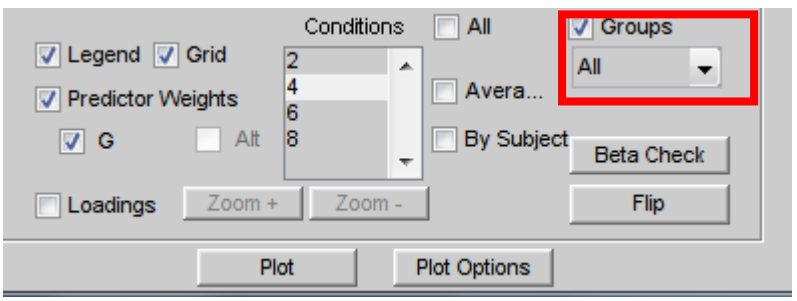


Figure 35: ***Multivariate Statistics*** **window, select plot groups**

i. Selecting the **'By Subject'** box will display predictor weight values for each subject.

j. When you are satisfied with your changes, press the **'Plot Component(s)'** button. The plot image will appear in a new window.

k. To save an image of your plot from this window, In the **'File'** dropdown menu, select ***Save As...***

2. Plot Options

There are a number of plots that can be shown for any of the models applied. Using this panel will allow the user to specify which plots they would like to see. The user can process multiple models and then use this panel to select plots without having to reapply the models.

a. Click the **'Plot Options'** button to open the '**plotSettings'** window. You will see three areas: *Global Options, Extended Options* and *Betas*.

i. In the **'Global Options'** area, you can make changes to the layout of the plot including labels (axes and plot titles), lines (style and size), and markers (size, colour, and edge).

ii. In the **'Betas'** area you can select to perform the Beta check on the top 1%, top 5% or top 10% of loadings in order to ensure the Varimax plot is oriented correctly.

iii. In **'Extended Options'** experimental conditions should be listed in the left hand box. The items listed under **'Conditions'** will be plotted.

To plot only some of the conditions:

a. Activate the **'Selected'** box and then double click on a condition name in the **'Conditions'** box. You can remove a condition from the **'Select'** box by double clicking the condition name in the **'Select'** box.

b. Any components you have selected to include in your plot can be modified (line colour, markers, etc.) individually by checking the box next to **'Custom Settings'** and selecting any desired changes from the menus below.

c. You can edit a condition name by highlighting it in the **'Conditions'** box then clicking **'Edit'**.

d. The **'Add'** button allows you to add a component back onto the

plot if it has been accidently deleted.

iv. The **'Preview'** button gives a preview of what the finished plot will look like. From this plot preview window, you can save an image of your modified plot.

v. When you are finished making changes, click the **'Okay'** button to return to the **'Statistics'** window.

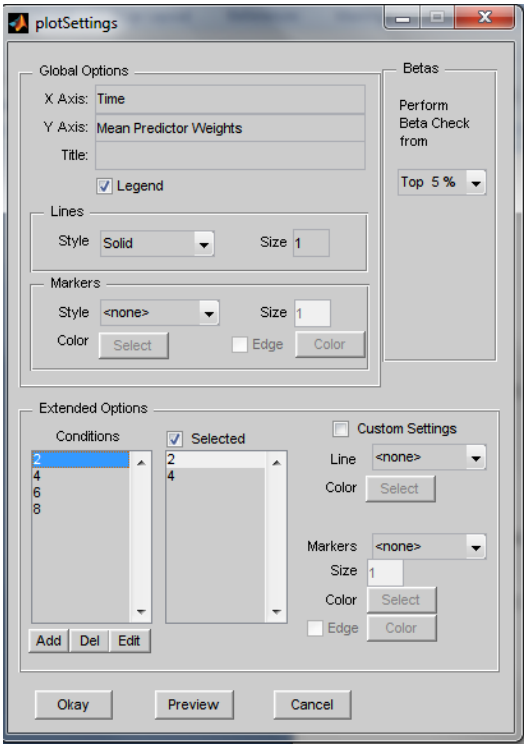


Figure 36: *Plot Options* window

3. Pressing the **'Cluster Data'** button will open the **'Cluster Information'** window which has information on clusters of active voxels including size, location (MNI coordinates) and positivity/negativity of clusters. This data is also written to **G_MNI_*.txt** (where * represents a rotation method name) in the **'G'** results folder. For this example, the text file is located in **\G\2_components\varimax\G_MNI_varimax_orthogonal_i500_p1.00_g 1.00.txt**

4. The number of points displayed on the scree plot can be edited in the box to right of **'Scree Plot'** button (default is 40).

5. Close the **'Multivariate Statistics'** when finished viewing/saving plots.

Appendix

I. GUI Mainscreen Panels

The main screen is separated into four panels:

- **System Information**

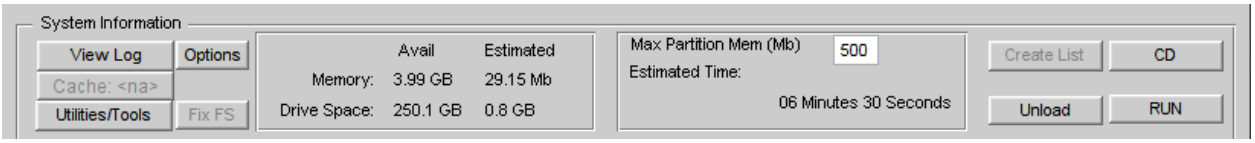


Figure 37: *System Information* panel

- **Subjects**

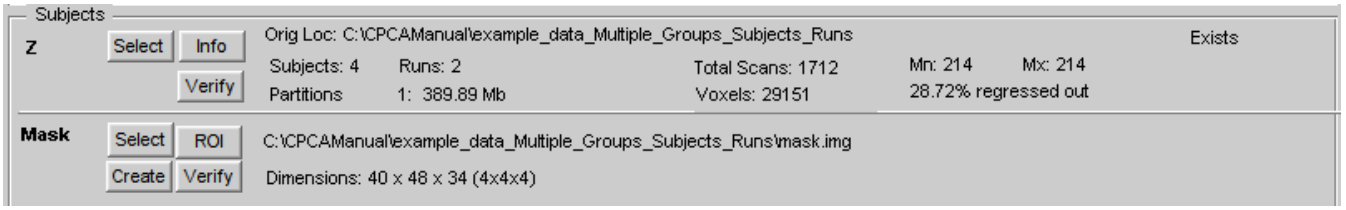


Figure 38: *Subjects* panel with Z and mask loaded

- **Model**

Figure 39: *Model* panel

- **Processing**

Figure 40: *Processing* panel

II. Additional CPCA GUI Features

- **Z Info**

  The Z **'Info'** button opens the Z Editor (see *Figures 46* and *47* below). The Z Editor allows the user to recreate a broken ZInfo file or to update the location of Z files and that have been copied or moved to a new location. It also allows subject groupings to be reviewed and set. (For details on how to use the Z editor, see Appendix section V: #Z_Editor).

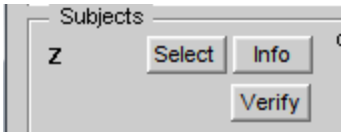


Figure 41: *Subjects* panel, Z *Info* button

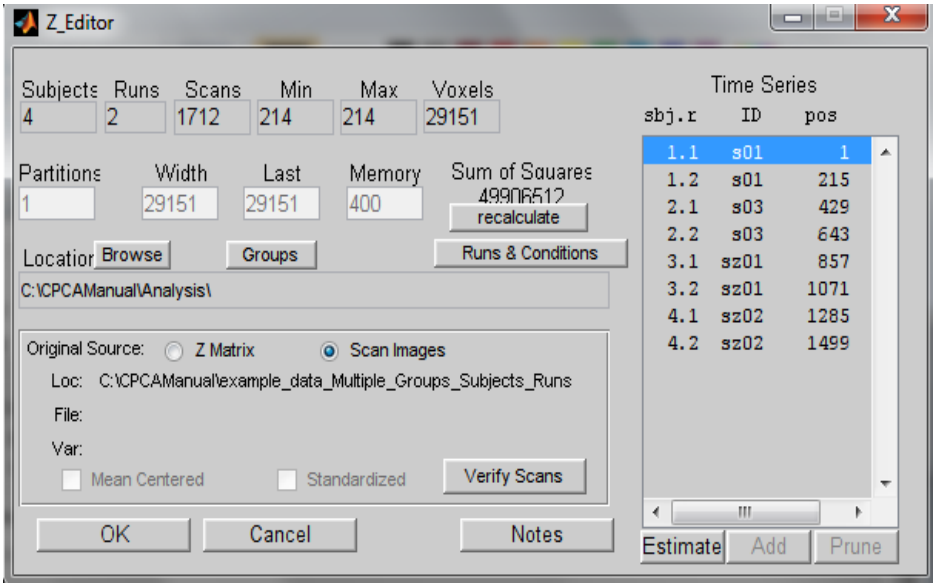


Figure 42: *Z Editor* window

- **Options**

  Allows you to edit global settings such as how often to clear cache, create larger variables, set default loadings percentage to use…(Options available will vary with cpca release versions. If you have any questions, please feel free to contact our team).

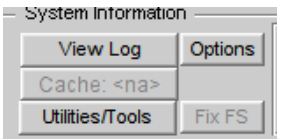


Figure 43: *System Information* panel, *Options* button

- **View Log**

  Opens a dated log of processes run from the current working directory.

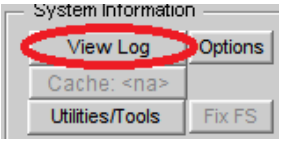


Figure 44: *System Information* panel, *View Log* button

- **Cache**

  The cache button will be enabled on Linux systems and shows how much cache space is being consumed. Pressing this button will prompt you, on the MATLAB command window, to enter your Linux password for sudo operation. If you have sudo permissions to clear the cache, it will be cleared.

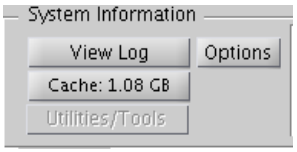


Figure 45: *System Information* panel, *Cache* button

- **Utilities/Tools**

  Clicking the '**Utilities/Tools**' button will bring up the following:

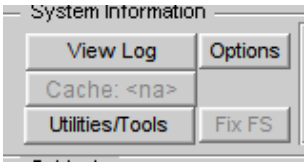


Figure 46: *System Information* panel, *Utilities/Tools* button

  - **Citation**
    - Displays a citation that researchers can include in any publications they submit.

  - **Debug Statistics**
    - Shows basic information for tracking down errors or issues—information about MATLAB, user memory, java, cpca versions, basic matrix elements etc.
    - Discovered errors are listed at the bottom of the text.
    - Creates a file 'Debug_information.txt' in your present working directory which can be emailed to the CNoS Lab for assistance with troubleshooting.

  - **Create Beta Images**
    - Produces images based on the mean betas for all subjects/all conditions, all subject/each condition, and each subject/each condition.
    - Allows user to recreate images for analyses run in previous versions of the GUI.

  - **Analysis Memory Estimates**
    - Based on entered information, provides estimates the memory size of the various CPCA matrices.
    - Indicates the recommended minimal computer installed memory necessary to run the analysis (most will fall within the 2-16GB range).

Note: Debug Statistics and **Create Beta Images** are only available once Z is loaded.

- **Unload**

  Pressing the 'Unload' button resets the GUI by unloading all loaded files: matrices (Z, G, A, H, etc.) and masks.

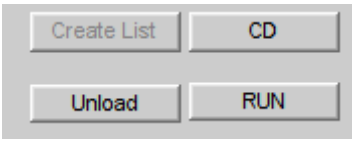


Figure 47: *System Information* Panel, *Unload* all data button

- **MEMORY AND DRIVE SPACE**

  - The **Avail** column shows how much memory and drive space is currently available.

  - The **Estimated** column shows how much memory and drive space the data load will need.

  - **MAX PARTITION MEM (MB)** value can be adjusted to minimize processing time.

  - **ESTIMATED TIME** indicates the estimated time for running any selected analytic processes.

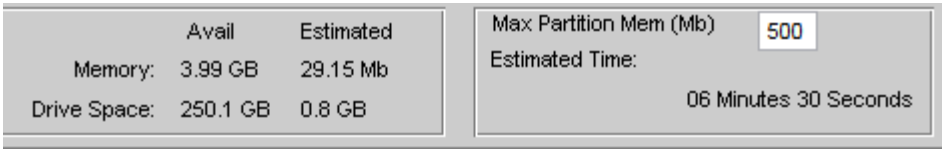


Figure 48: *System Information* Panel, memory and drive space

- **G Editor**

  The **'Edit'** button in the ***Model*** panel is activated after loading a G.mat file (G matrix). Selecting it opens the G editor. The main purpose of the G editor is to avoid having to reapply G once it has been applied to a large data set. The editor allows any information associated with the G model to be recovered (including the number of conditions or time bins, vector units, or condition names. It also allows a G segment and an image to be selected and viewed.

  *** **Note:** If the '**Edit**' button is red, this indicates that there is an error with G (a malformed matrix) so rank is out of order. Click on the red '**Edit**' button or look at a file called **G_ranking.txt** in the **G_segs** directory to see information about the problem(s).

- Click the **'Edit'** button to open the **'G Editor'** window.
- **'Source'** Informs you as to where the G was created from and which timing vector file or pre-created G matrix was specified.
- Clicking the **'select sou…'** button allows you to select the timing onsets for your G matrix.
- Clicking the **'View'** button opens a new window displaying a plot of the G matrix. You can select to view the G matrix for individual subjects by selecting the subject from the drop-down menu and checking the 'subject only' box.
- **'Stored:'** Shows where the G data is stored, in segmented format, to apply to Z1-Zn data. The usual location will be in a folder named 'Gzsegs' in the directory where you ran the application from. If you are unsure of where this was, you can search the usage logs to assist you, as this information is saved to the logs. You can change where the G matrix is stored by clicking the 'Browse' button next to the 'Stored:' text box.
- **'Applied'** gives the date when the G was applied to a data set.
- **'to'** shows where the Z1-Zn data set the G was applied to is located. You can update this field by using the Browse button associated with this entry, and selecting the directory the Z1-Zn data resides in.
- **'GZ Information' Panel**
  - Pressing the **'Recalulate'** button will redo the math for determining the GC sum diagnol.
  - **'Browse'** allows you to change where GZ information is stored (such as if you have moved to a new working directory and want to recalculate and save the results in the new location).
- Clicking the **'Load'** button allows you to load a G header file.
- The **'Notes'** button opens a new window where you can enter text to make notes.
- The **'Subject Rank'** button gives the rank of the matrix which should equal the columns of G. This will help you determine if your G is malformed.

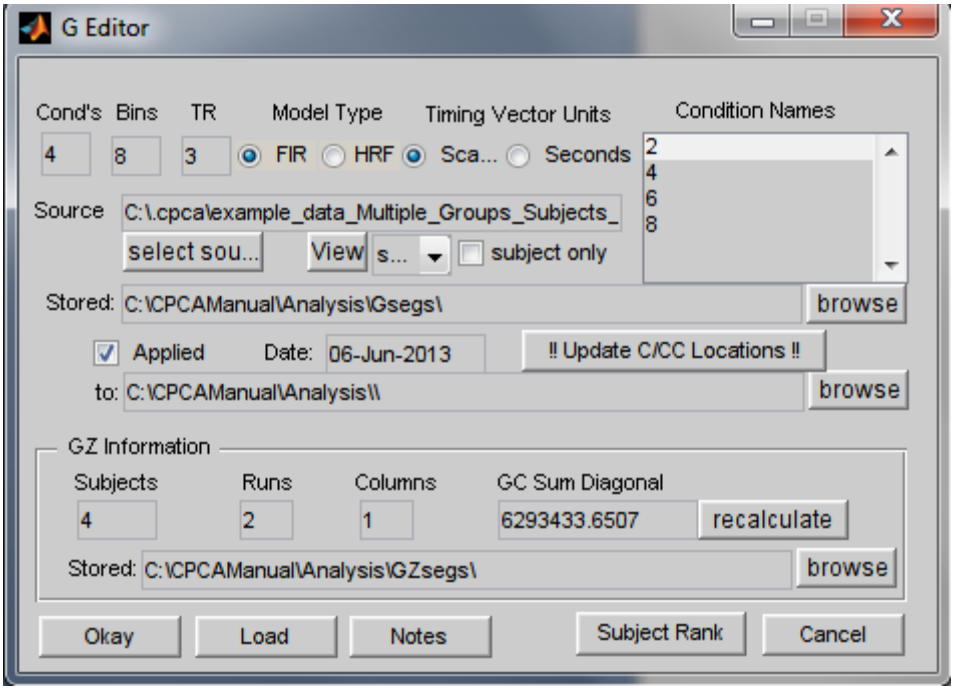


Figure 49: *G Editor* window

III. Normalization options

It is important to normalize Z (the fMRI scans) in order to standardize the variance that exists across the brain voxels. If normalization is omitted some voxels will incorrectly dominate the components that are extracted. The normalization options are outlined below.

For Z matrices that are created with the GUI via a subject scan file list (as outlined in the previous section), normalization parameters are specified on the processing panel. The default options are shown here.

NORMALIZATION	
✓ **Create Z**	Normalize the subject scan data using the selected options:
✓ **Regression Covariates**	Refers to any user-defined differences
✓ **Linear Regress**	Removes the linear trend on BOLD signal over all scans.
✓ **Quadratic Regress**	Removes the quadratic trend on BOLD signal over all scans.
✓ **Head Movement***	This option allows selection of a file that contains head movement data to be regressed from the scans.
✓ **User Defined Covariates**	Can be used in lieu of the defaults if you have information of your own you want to use. Note: anything you prepare must be precise and in the proper subject order.
✓ **Mean Centering* Not an option**	Takes the mean of the data, per subject, and reduces the raw data amount by that value, centering the data on zero.
✓ **Standardize*Not an option**	Calculates the Standard Deviation of the data, and divides that into the data values: (val / (sqrt(sum_of_squares / num_scans))

***Head Movement**: CPCA will automatically try to find movement parameter files (in the format rp_{…}.txt). These files are found in each individual subject's directory after you do the realignment. If the specific files are found (there needs to be one in every run directory), then the Head Movement option will activate and it can then be selected.

*** For the rare occasion when a Z matrix is created outside of the GUI, the method to normalize is different. In this case, the unnormalized Z is loaded using the Z 'Select' button located on the Subjects panel. The application will automatically check to see if this loaded Z has previously been normalized and open the Z_Settings window allowing the normalization options to be set and processed.

Note: All groups must be included in one Z and normalized together. Should normalization be interrupted for any reason (usually lack of disk space), it can be resumed from the point where it was stopped.

IV. Z Editor

After a Z matrix is loaded the Z editor can be opened by pressing the **'Info'** button located on the Subjects panel. (See Figures 61 and 62 below).

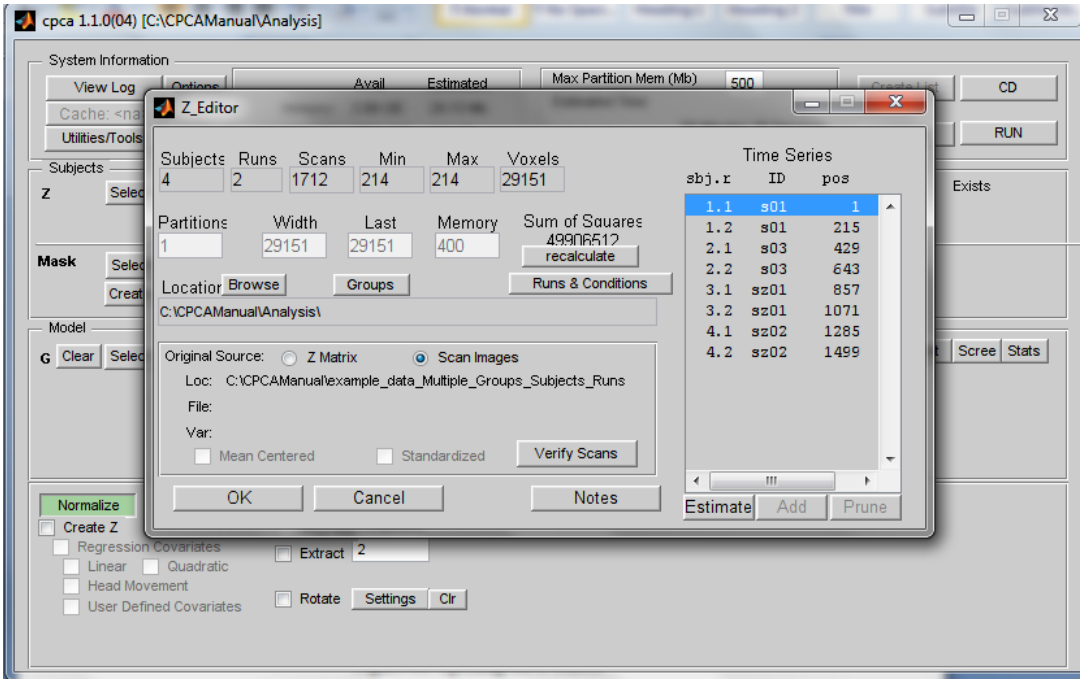


Figure 50: Opening the Z Editor

The Z editor allows Z to be repaired or changed in numerous ways. The Editor:

1. Avoids the need to renormalize a large data set by allowing the user to:

    - Update the location of Z files and that have been copied or moved to a new location or computer.

    - Recreate a broken Z Info file. For example, it would be necessary to repair Z file headers that are corrupted due to an upgrade to a newer version of fMRI-CPCA with a different data header model.

2. Allows subject groupings to be reviewed and set.

3. Allows subject ID's to be altered should there be duplication under group folders.
    - All subjects are assigned an ID. Normally, this is simply the Subject folder name. This ID is used to ensure a proper alignment between subject data and G creation by onsets.

Using the Z Editor the following numbers can be directly edited:

- Subjects
- Run
- Voxels in mask
- Total Scans

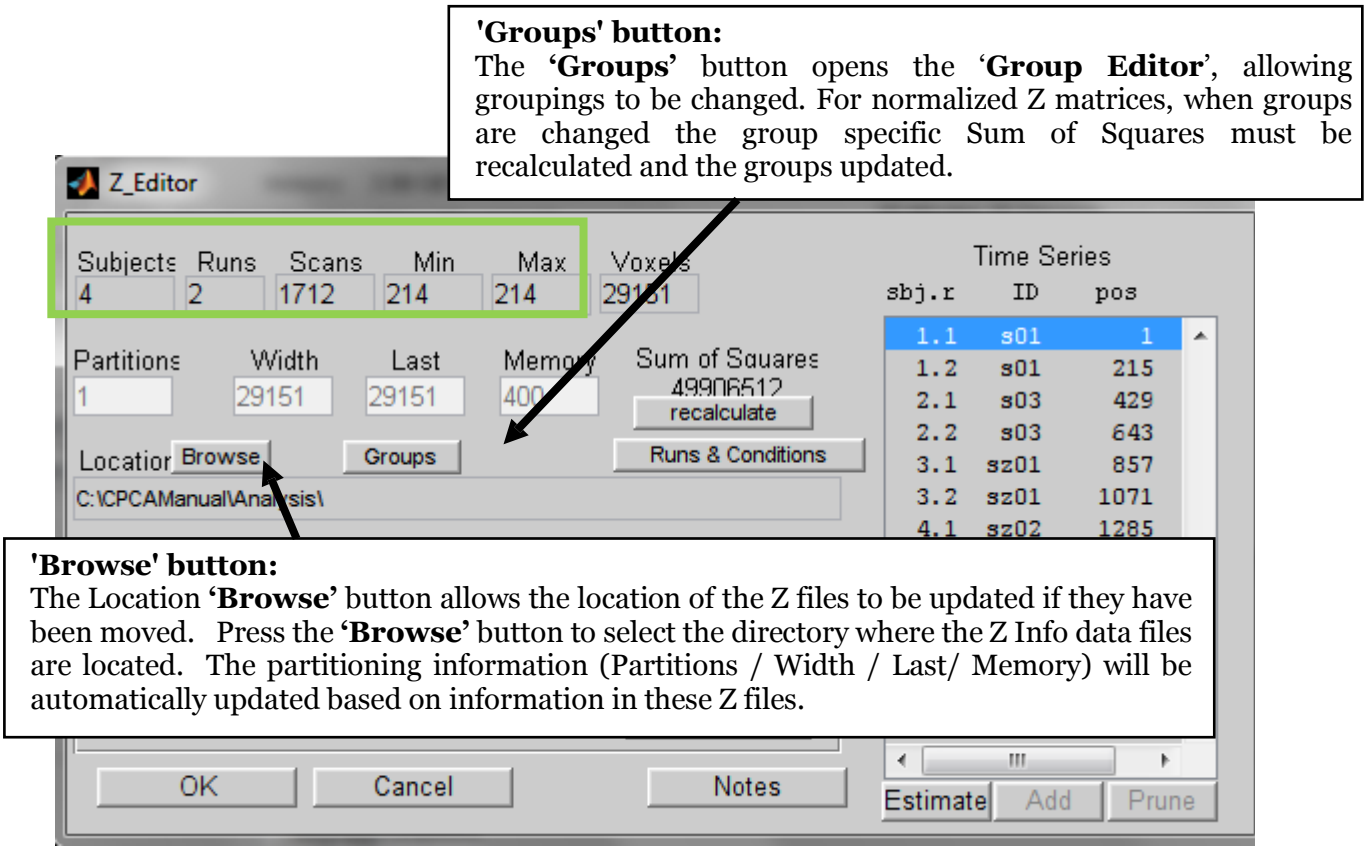


Figure 51: The Z Editor

4. Double clicking on any subject (listed under **sbj.r**) will open the **'Edit Subject'** window, allowing the user to edit the number of scans (depth) and the subject ID (name or code of users choice e.g. s01) associated with that subject. Press '**Okay**' in the '**Edit Subject'** window and '**Okay**' in the '**Z Editor**' window to save the changes and return to the main GUI.

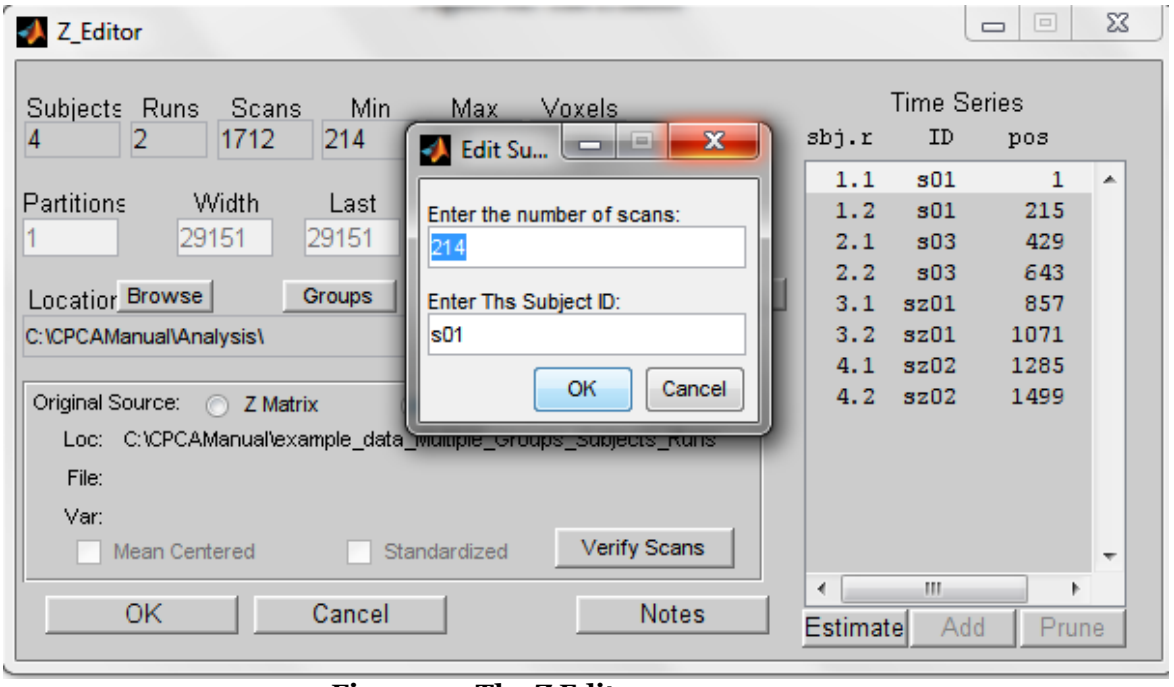


Figure 52: The Z Editor

V. Viewing the functional networks

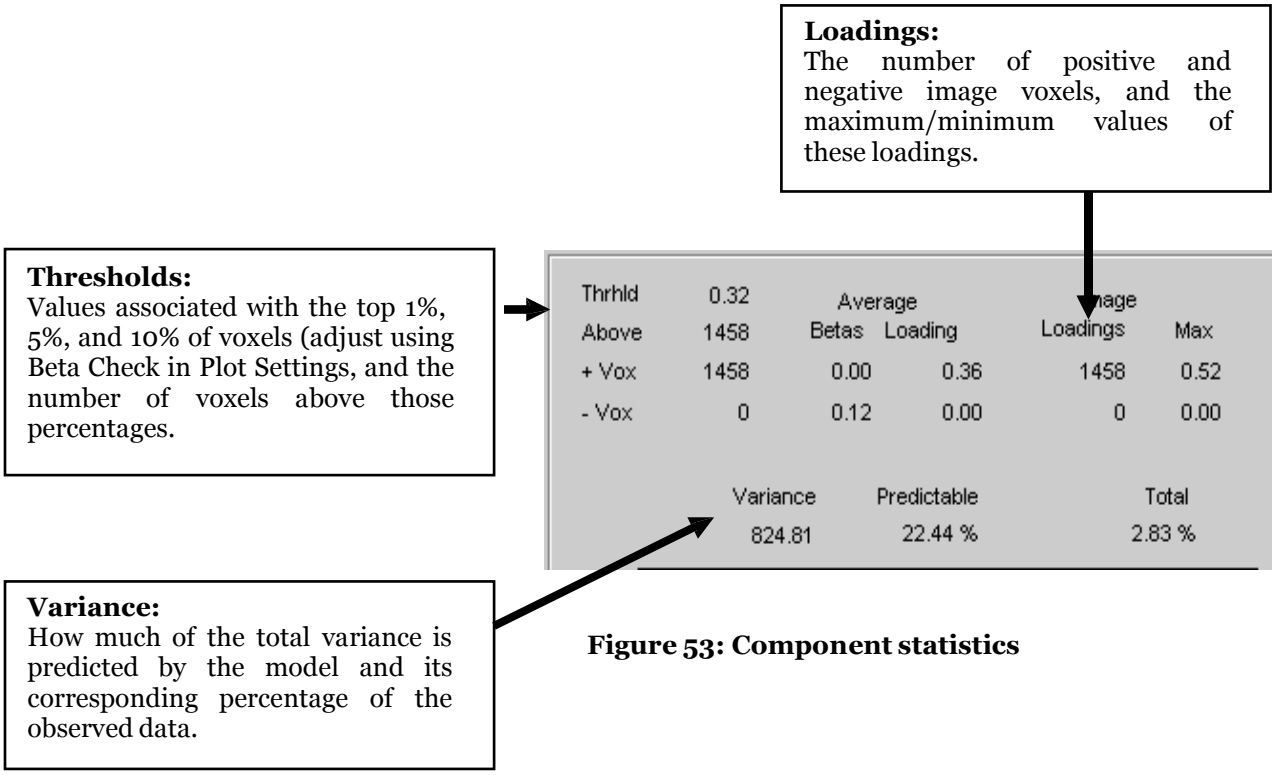


Figure 53: Component statistics

The minimum files that need to be transported to use CPCA to view results of an analysis are as follows:

- **G directory**
- **G header**
- **Z Info**

Note: When the Z Info file is selected, there will be a pop-up saying that the path needs to be corrected. Click '**OK**' and the results should then appear. With these minimum files, the results can be viewed but not edited or altered in any way.

The functional networks are whole brain images representing systems of functionally interconnected brain voxels. To view a functional network, the loadings contained in the image file are overlaid onto a brain template. A given functional network can contain only positive loadings (activations), only negative loadings (deactivations), or both.

The networks images will have been written to a folder in your output directory labeled **'Images'**. Each network image is composed of a disc image file and an .hdr file. In this example only the unrotated solution was extracted, but it possible to extract multiple rotations in one analysis (see the comprehensive manual). For each solution, a separate functional network is created for each component. In this one solution example, one functional network was created for each of the two components extracted:

unrotated_Component_1.hdr
unrotated_Component_2.hdr

V.1. How to view the functional networks using MRIcron

Step 1: *Determine the display values*

In the Statistics window:

* Ensure that **(2) unrotated** is selected in the top left hand corner.

* Select **'1'** in the component selection pull down menu. The statistics for the selected component will be displayed in the upper right hand section of the statistics window.

  From the statistics displayed for component 1 (see Figure 65 below), it is evident only positive loadings passed the display threshold (positive loadings as listed in the +Vox row) and the display threshold values (threshold and maximum) can be determined.

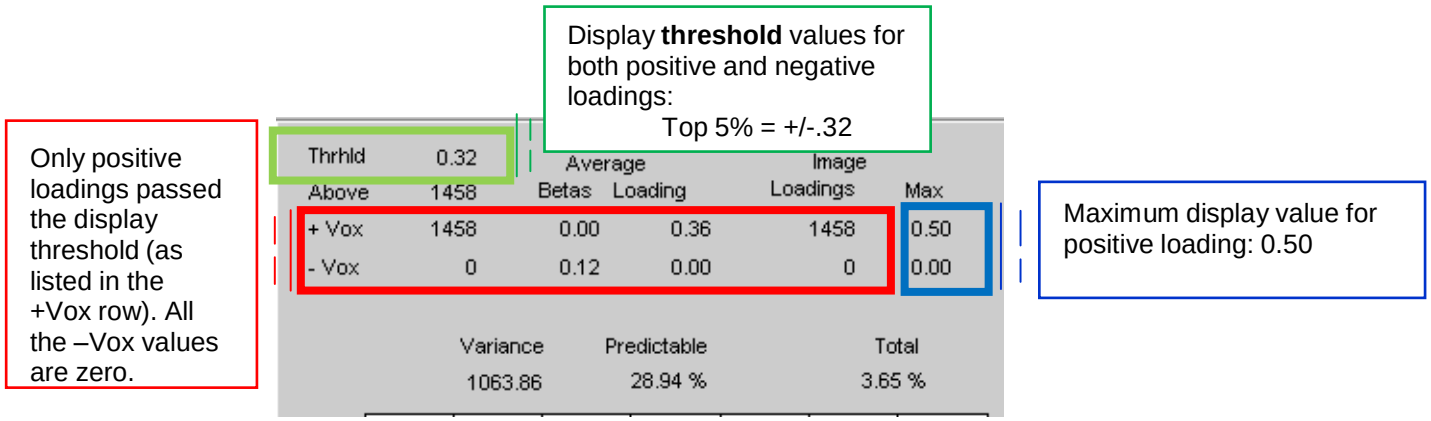


Figure 54: Display values for component 1

Note that the threshold values are the same for both the positive and negative voxels, but the maximum (most extreme positive and negative) values differ (in this case the maximum value for positive and negative voxels are 0.50 and 0.00, respectively). The thresholds for display must be the same for both the positive and negative loading voxels. The maximum values can optionally be adjusted to control activation brightness.

Step 2: *Start MRIcroN*

- Start the application ***MRIcroN.exe.*** The Statistics window and the MRIcroN window can be viewed side by side to facilitate viewing of the network.
- In the MRIcroN *File* dropdown menu, select *Open Templates.*
- Select the template named **'ch2.nii.gz.'**

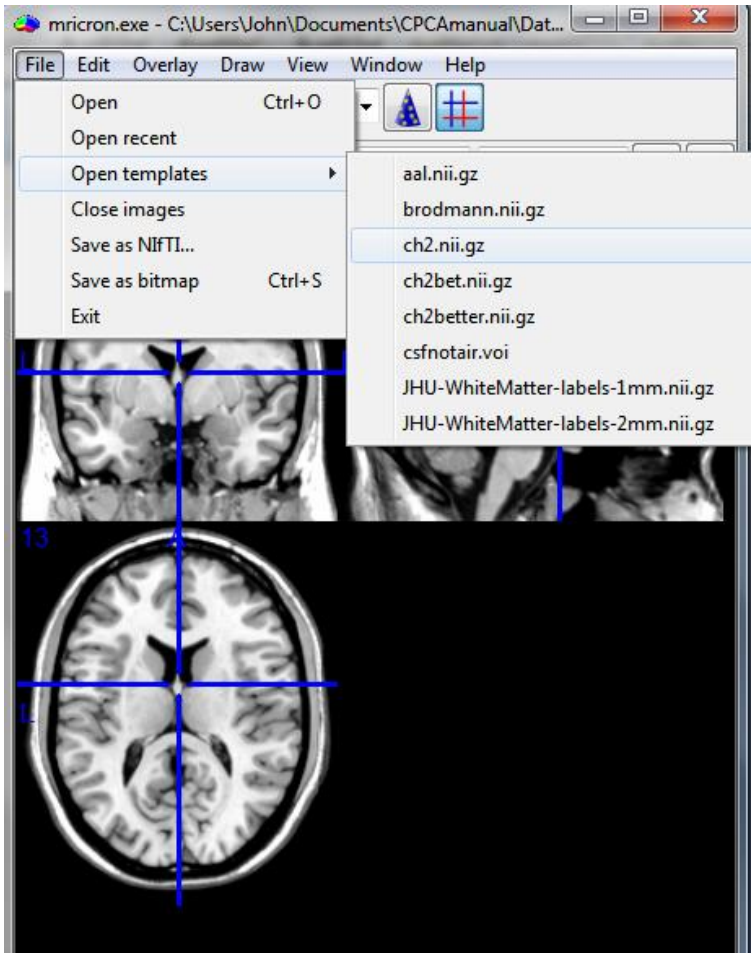


Figure 55: Opening Templates in MRIcroN

Step 3: *Load the functional network images*

- In the Overlay dropdown menu, select *Add*. Navigate to and select the .hdr image for component 1.

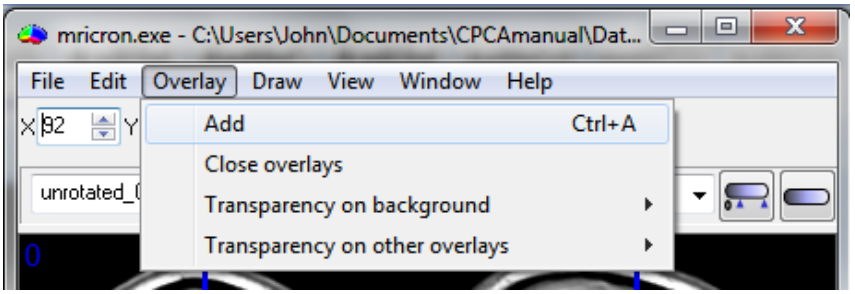


Figure 56: Add an overlay in MRIcroN

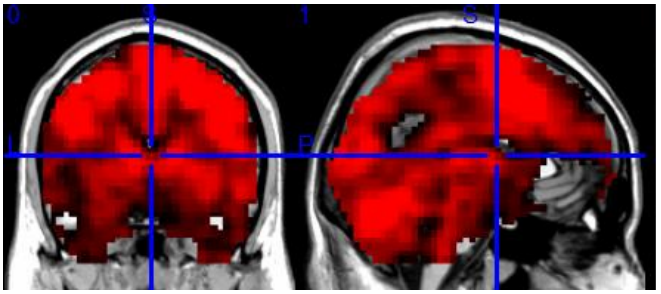


Figure 57: Unthresholded positive loadings

- The loadings will appear as a red coloured activation map. (Red is the default colour for the first overlay added, and blue is the default for the second).

Step 4: *Set the display values*

The MRIcroN menu contains two boxes where the threshold and maximum display values must be entered. The relation between the first number (left-side box) and the second number (right-side box) is always from the lowest to the highest value. Mostly, to display activations or deactivations, the threshold value is entered into the left-side box and the maximum value into the right-side box. When displaying negative deactivations both the thresholds and maximum values should include a minus sign. This may cause their values (and therefore the relation between the numbers) to change. You will need to adjust what you enter into the boxes accordingly with the lower value entered in the left-side box and the higher value entered in the right-side box.

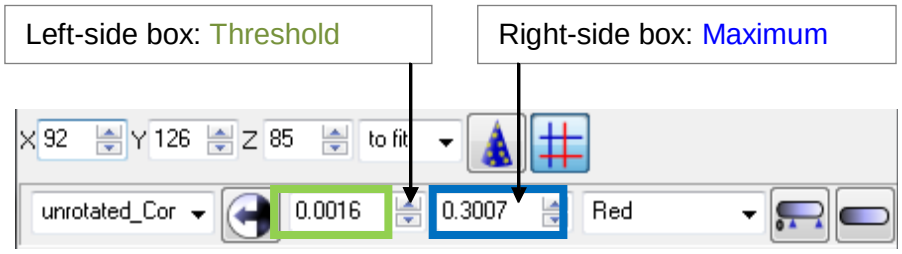


Figure 58: MRIcroN display value boxes

- Match the threshold and the maximum values to those in the statistics window:
- Set the display values for the positive loadings:
  - Enter 0.32 in the threshold / left-side box (top 5% of loadings in this case).
  - Enter 0.50 in the maximum / right-side box.

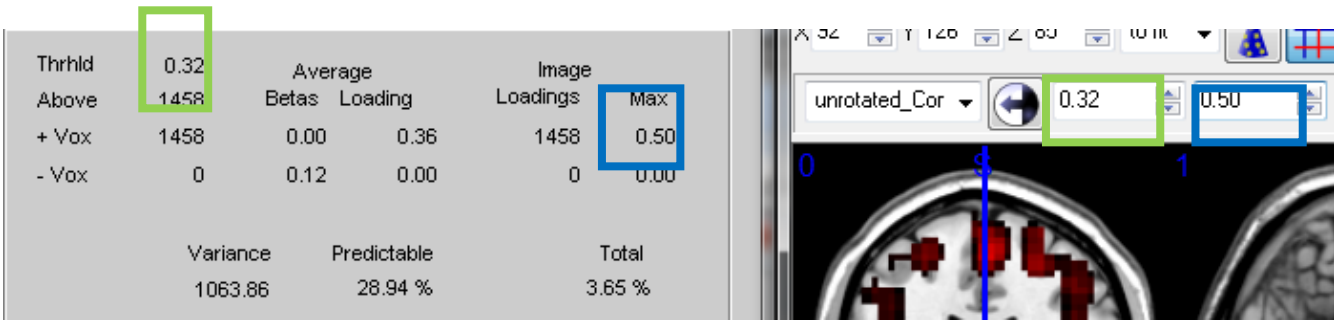


Figure 59: Setting component 1 positive loadings

Step 5: Set the viewing options

The functional networks loaded in MRIcroN can be viewed in several different orientations and slice levels. Any of the selected views can be saved as bitmap images. One viewing option is the multislice option.

- In the *Window* dropdown menu, select *Multislice,* which will open the MultiSlice window.

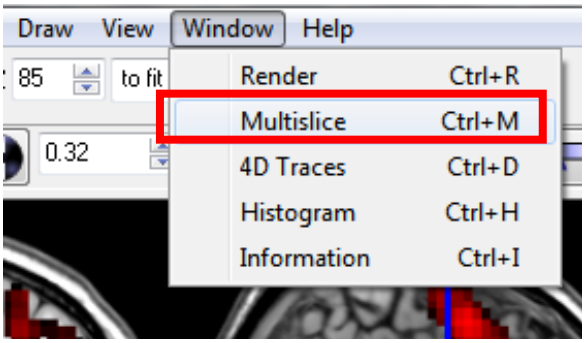


Figure 60: Open Multislice view

- In the MultiSlice *View* dropdown menu:
  - Check the *Orthogonal View* and the *Show Slice Label* option.
  - Select *Orient,* and choose *Coronal.*

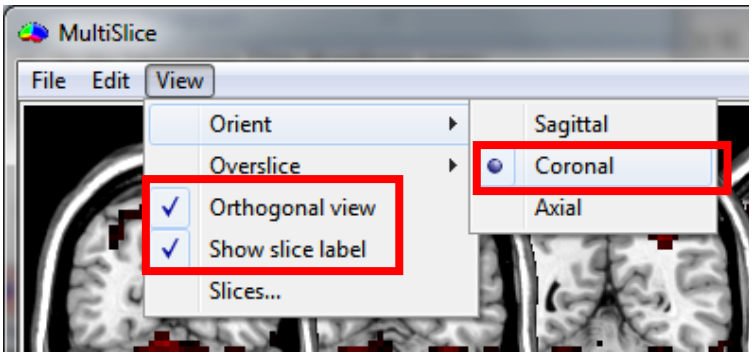


Figure 61: Set Multislice view to coronal

- Select *Overslice*, and select *0%*.

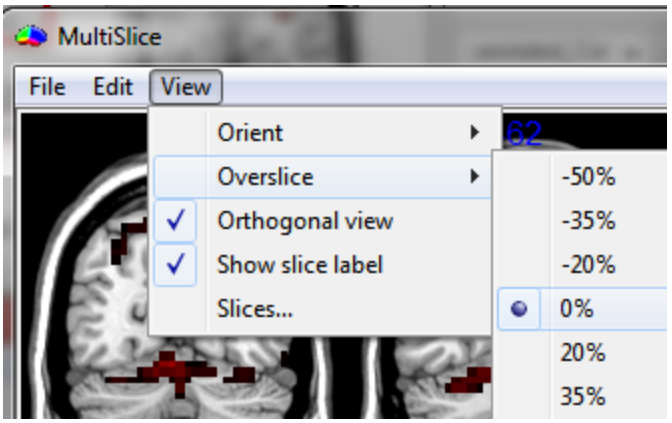


Figure 62: Select Overslice

- In the MultiSlice *View* dropdown menu:

  - Select *Slices…, and e*nter Slice Numbers **68,87,123,136,162** (y-coordinate values) into the Select multislices window. Press '**OK'** to view an image showing multiple brain slices**.**

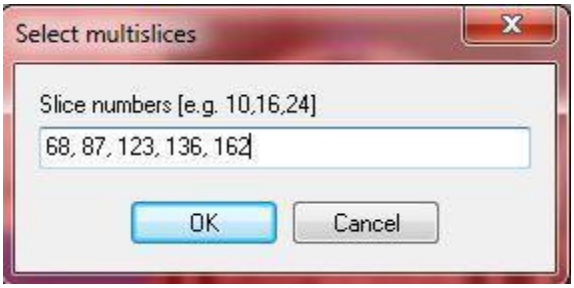


Figure 63: Select multislices window

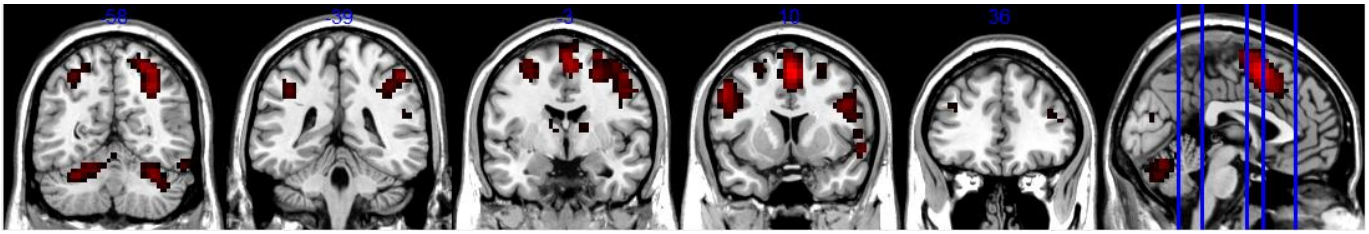


Figure 64: Functional network for component 1

Step 6: *Save an image of the functional network*

- In the MulitSlice *File* dropdown menu, select *Save as bitmap*…

- Repeat Step 1 – 6 for Component 2 using the threshold, max (0.40), and min (-0.31) values associated with the component.

  Note: The relation between the first number (left-side box) and the second number (right-side box) is always from the lowest to the highest value. Therefore, when dealing with negative loadings, we will have to look at the values of the negative maximum and the value of the threshold (remember to make this negative) to determine which number should be entered into which box.

VI. Interpret the results

The tutorial data are from a working memory task that required subjects to remember 2, 4, 6 or 8 Letters across a delay. When fMRI-CPCA was used to extract two components, aspects of the task-positive and task-negative (i.e. default) networks emerged.

VI.1. Interpretation of component 1

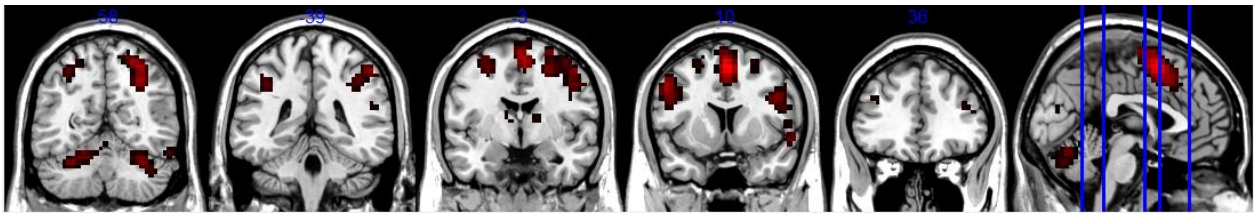


Figure 65: **Interpretation of component 1**

The network for component 1 reveals the system of functionally connected brain areas that are active during the working memory task. Included in this system are the dorsal anterior cingulate gyrus, dorsolateral prefrontal cortex, bilateral middle and inferior frontal gyrus, bilateral supramarginal gyrus, and insular cortex.

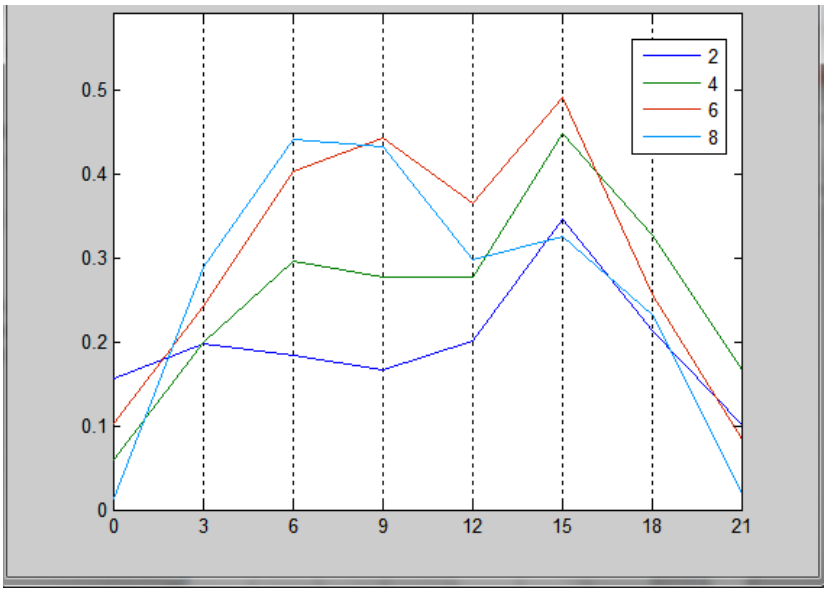


Figure 66: Estimated HDR of Component 1

The plot of the estimated HDR for component 1, reveals that the activation of this functional network is load dependent, with the greatest activation present for the highest memory load condition (8 and 6 letters) and the lowest level of activation present for the lowest memory load condition (2 letters)

VI.2. Interpretation of component 2

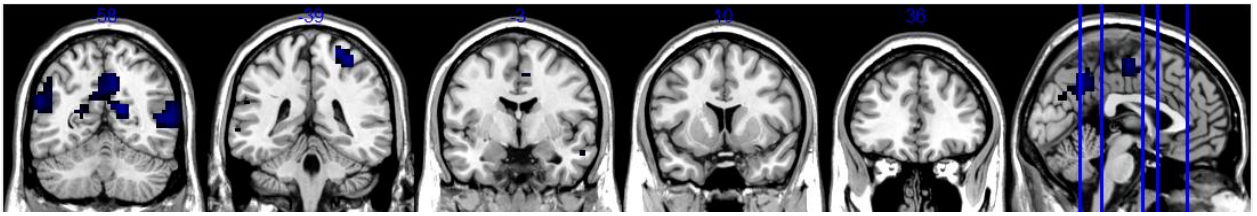


Figure 67: Interpretation of component 2

The Functional network for component 2 contains elements of the task positive network found in component one, but it also contains elements of the task negative (i.e., default) network. The task negative network consists of a system of functionally connected brain areas that are suppressed during the working memory task. Included in this system are the ventral anterior cingulate gyrus, posterior cingulate/precuneus, bilateral inferior parietal lobes, primary auditory cortex, planum temporale and posterior superior temporal gyrus.

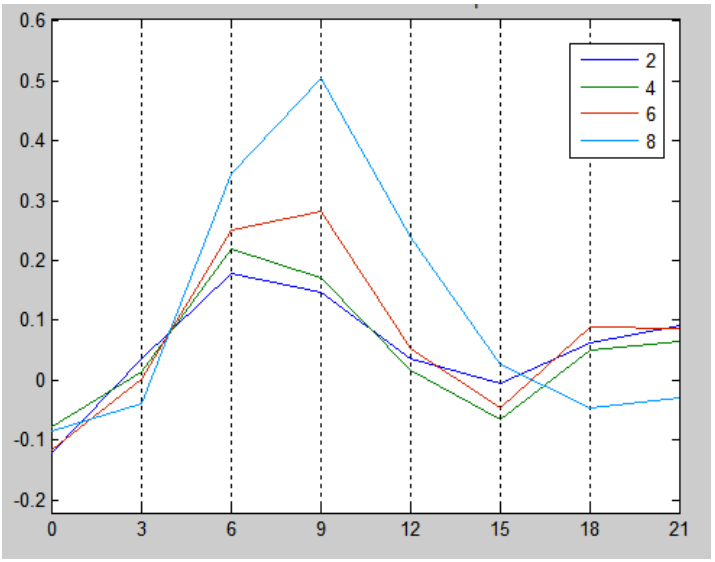


Figure 68: Estimated HDR of component 2

The plot of the estimated HDR for component 2 reveals that the intensity of the activation increases (positive red loadings) and decreases (negative blue loadings) is clearly load dependent, with the greatest intensities present for the highest memory load condition (8 letters) and the least for the lowest memory load condition (2 letters).

VII. Creating an HRF G Model

- Click on the G '**Create**' button to bring up the '**G Matrix Creation'** window.

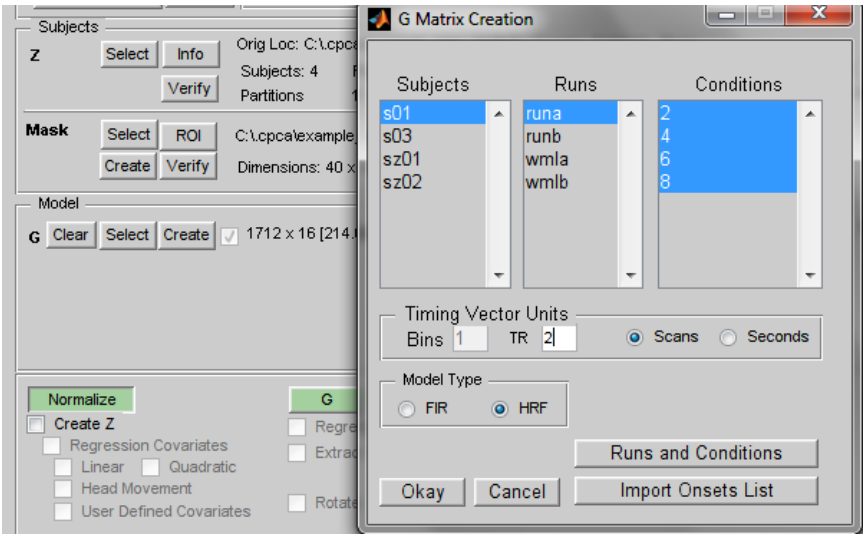


Figure 69: HRF G Model

- Click on '**HRF**'. This will deactivate the Bins and set it to '1' (as with the HRF model, each condition is a single run).

- For this example, the TR was set to 2.

- Click on '**Runs and Conditions**' and follow the same steps as with a FIR model (see section #Create_a_FIR_G)

There is a difference between the FIR Model and the HRF Model with regards to the Timing Onset File. With HRF, each model will have a duration entry after it. There is no default for this. That is, a single [**0**] must be entered to be default in each file imported.

```
%       ------------------------------------------------------
%       ---     timing  onsets  for     subject s01
%       ------------------------------------------------------
s01_runa_2_letters      =       [55.269 79.07   89.12   118.57  123.27  138.01  163.45  179.19  189.25  ];
s01_runa_2_letters_dur  =       [0]
s01_runa_4_letters      =       [       5.03    10.38   20.77   69.68   74.38   108.52  132.99  158.43  19
s01_runa_4_letters_dur  =       [0]
s01_runa_6_letters      =       [       15.74   25.79   40.54   143.37  148.73  168.81  173.83  183.89  19
s01_runa_6_letters_dur  =       [0]
s01_runa_8_letters      =       [       30.49   35.84   45.24   59.96   64.66   84.1    94.48   113.21  12
s01_runa_8_letters_dur  =       [0]

s01_runb_2_letters      =       [55.269 79.07   89.12   118.57  123.27  138.01  163.45  179.19  189.25  ];
s01_runb_2_letters_dur  =       [0]
s01_runb_4_letters      =       [       5.03    10.38   20.77   69.68   74.38   108.52  132.99  158.43  19
s01_runb_4_letters_dur  =       [0]
s01_runb_6_letters      =       [       15.74   25.79   40.54   143.37  148.73  168.81  173.83  183.89  19
s01_runb_6_letters_dur  =       [0]
s01_runb_8_letters      =       [       30.49   35.84   45.24   59.96   64.66   84.1    94.48   113.21  12
s01_runb_8_letters_dur  =       [0]
```

Figure 70. Timing Onset file for HRF: The duration entry is added after each condition with the number set to 0.

The single value entry for HRF duration will be used as a default for all condition onsets. Example: [**2**] will apply to all [**1 2 3 4 5**] unless otherwise specified [**2 2 3 2 2**].

FIR

```
% ---------------------------------------
% --- timing onsets for subject s01
% ---------------------------------------
s01_runa_2 = [ ];
s01_runa_4 = [ ];
s01_runa_6 = [ ];
s01_runa_8 = [ ];

s01_runb_2 = [ ];
s01_runb_4 = [ ];
s01_runb_6 = [ ];
s01_runb_8 = [ ];
```

HDR

```
% ---------- ----------
% --- timing onsets for subject s01
% ----------------------------------
s01_runa_2 = [ ];
s01_runa_2_dur = [ ];
s01_runa_4 = [ ];
s01_runa_4_dur = [ ];
s01_runa_6 = [ ];
s01_runa_6_dur = [ ];
s01_runa_8 = [ ];
s01_runa_8_dur = [ ];

s01_runb_2 = [ ];
s01_runb_2_dur = [ ];
s01_runb_4 = [ ];
s01_runb_4_dur = [ ];
s01_runb_6 = [ ];
s01_runb_6_dur = [ ];
s01_runb_8 = [ ];
s01_runb_8_dur = [ ];
```

Figure 71: FIR vs. HRF G timing onset template

```
% ------------------------------------------------------
% --- timing onsets for subject s01
% ------------------------------------------------------
s01_runa_2 = [55.269 79.07 89.12 118.57 123.27 138.01 163.45 179.19 189.25 ];
s01_runa_2_dur = [0];
s01_runa_4 = [ 5.03 10.38 20.77 69.68 74.38 108.52 132.99 158.43 199.29 ];
s01_runa_4_dur = [0];
s01_runa_6 = [ 15.74 25.79 40.54 143.37 148.73 168.81 173.83 183.89 194.269];
s01_runa_6_dur = [0];
s01_runa_8 = [ 30.49 35.84 45.24 59.96 64.66 84.1 94.48 113.21 128.29 ];
s01_runa_8_dur = [0];

s01_runb_2 = [55.269 79.07 89.12 118.57 123.27 138.01 163.45 179.19 189.25 ];
s01_runb_2_dur = [0];
s01_runb_4 = [ 5.03 10.38 20.77 69.68 74.38 108.52 132.99 158.43 199.29 ];
s01_runb_4_dur = [0];
s01_runb_6 = [ 15.74 25.79 40.54 143.37 148.73 168.81 173.83 183.89 194.269];
s01_runb_6_dur = [0];
s01_runb_8 = [ 30.49 35.84 45.24 59.96 64.66 84.1 94.48 113.21 128.29 ];
s01_runb_8_dur = [0];
```

Figure 72: HRF model imported timing onset file

After the '**Timing Onsets File'** has been imported, click on G '**Edit**'. To view the hrf curve for one subject at a time, check off the '**Subjects Only**' box then select the subject to view in the dropdown menu. If '**Subjects Only**' is not checked off, then the hrf curves for all subjects will be displayed in one chart. Click on '**View**' to view the hrf curves.

Rotation is still possible with the HRF G model.

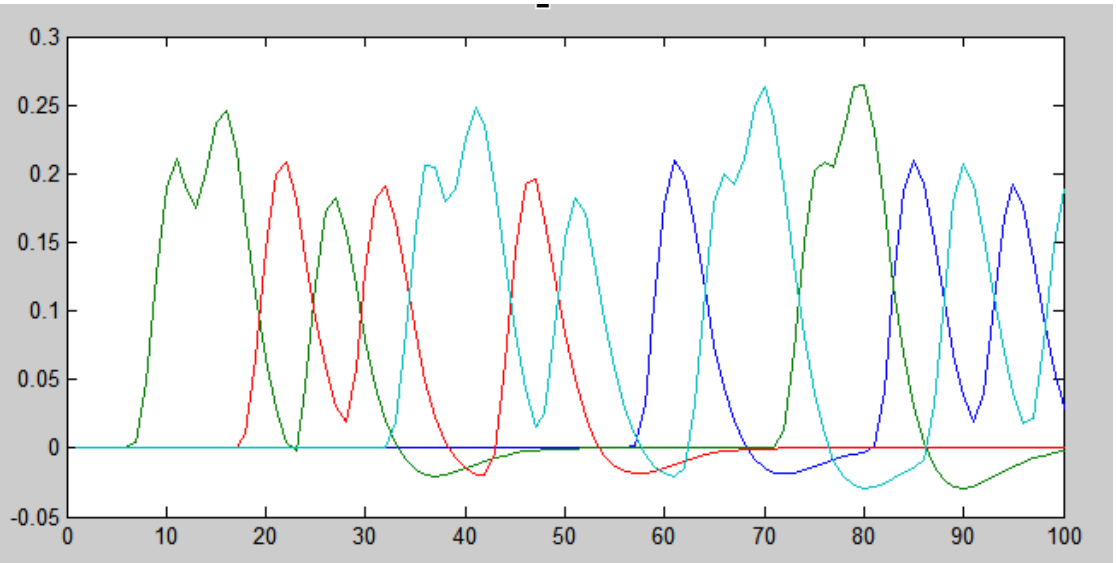


Figure 73: HRF curves plot

VIII. Output Files

VIII.1. CPCA Output

CPCA analysis output will be written to the output directory you selected at the beginning of the analysis.

The CPCA GUI creates an **'hrfmax_shapes_used.mat'** file which is the most recently used shapes file. It also creates a similar file, **'mask_used.mat'**, for the most recently used mask.

- **Gsegs folder:** The original G matrix will be partitioned into separate G matrix files for each subject and run and written to this folder.

- **GZsegs folder:** Contains matrices for each individual subject for G applied to C and Z.

- **Z folder:** To reduce processing demands the original Z.mat is partitioned into individual subject specific .mat files. Z Info.mat, holds the header information for all subjects.

- **G folder:**

  - **'#_components'** folder (where # is the number of components you extracted in your analysis. For this tutorial, you will have a folder titled **'2_components.'**

    - **Rotation** folders. Within the components directory you will find folders with names corresponding to the rotation methods you performed in your analysis. For this tutorial, you will have **'varimax'** and **'unrotated'** directories.

      - **'Images'** folder. Within each rotation method folder there is an **'Images'** folder which contains header and image files for each component of the cpca analysis. This folder also contains an **'image_loadings.mat'** file.

  - 'Beta' folder (in the G folder, created using #Utilities/Tools. See Appendix, section II, Additional CPCA Features). This directory contains image and header files for overall beta values as well as subject specific beta values.

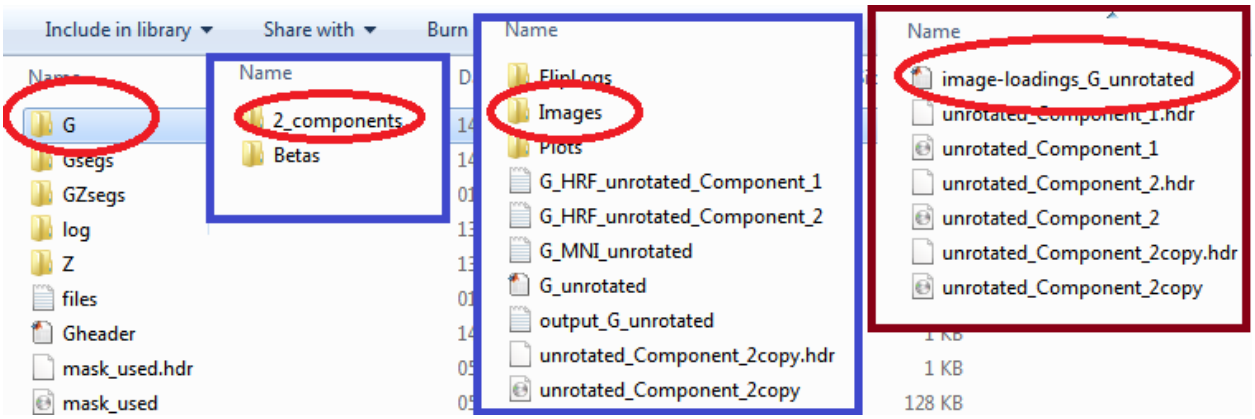


Figure 74: CPCA output folders

VIII.2. MATLAB Output

You should expect output in the MATLAB window during the analytic processing similar to that shown here:

```
Subjects:   4  [   1712 x  29151 ]  Runs:  2
Estimated Completion Time: 06 Minutes 30 Seconds
 - Subject s01  1 of 4 Run:  1 [=============] [00:17]
 - Subject s03  2 of 4 Run:  1 [=============] [00:15]
 - Subject sz01  3 of 4 Run:  1 [=============] [00:14]
 - Subject sz02  4 of 4 Run:  1 [=============] [00:14]
  Process           start        end          Duration
-------------------------------------------------------------------------------
   Normalization    12:52:14     12:53:17     1 Minute 2 Seconds
-------------------------------------------------------------------------------
         Elapsed    12:52:14     12:53:17     1 Minute 2 Seconds
Subjects:   4  [   1712 x  29151 ]  Runs:  2
       G:      [   1712 x    128 ]  C:\CPCAManual\Analysis\Gheader.mat
      GZ:      [    128 x  29151 ]

   -Subject:   4  Run:  2
   - Creating C*C
         Apply G    12:57:36     12:57:52     0 Minute 16 Seconds
********************< Computation of SS GC >*********************

SS_Z               = 49906512.00
SS_GC              = 6293433.65
SS percent of Z    = 12.61
  Process           start        end          Duration
-------------------------------------------------------------------------------
         Apply G    12:57:36     12:57:52     0 Minute 16 Seconds
-------------------------------------------------------------------------------
         Elapsed    12:57:36     12:57:54     0 Minute 17 Seconds
Subjects:   4  [   1712 x  29151 ]  Runs:  2
       G:      [   1712 x    128 ]  C:\CPCAManual\Analysis\Gheader.mat
      GZ:      [    128 x  29151 ]

------------------------------------------------------------
Extracting 2 components from GZ
------------------------------------------------------------


filename: output_G_unrotated.txt
original location: C:\CPCAManual\Analysis\G\2_components\unrotated\
date created: 06-Jun-2013
created version: cpca 1.1.0(05)
 edited version:
 displaying top 10%

----------------------------------------
General Information
----------------------------------------
  Subjects:   4
  Runs:       2
```

```
  Total Scans:       1712
  Voxel Width:       29151

  Model Type:        FIR
  Onsets measured in scans
  Conditions:        4
  Bins:       8
  TR:         3.00

-----------------------------------------
% of variance in original BOLD signal regressed out in CPCA preprocessing
-----------------------------------------
         Total Removed:     28.72% (   14332795.40)
          Linear Trend:     23.77% (   11863422.67)
       Quadratic Trend:      4.95% (    2469372.73)
         Head Movement:      0.00% (          0.00)
          User Defined:      0.00% (          0.00)

Extreme Positive negative loading for unrotated components:
-------------------------------------------------------------------------------
Component 1         minimum:  -0.50     maximum:  0.19

percentage of loadings            5%        10%        20%
abs threshold                   0.32       0.29       0.25
voxels above                    1458       2915       5830
positive voxels                    0          0          0
negative voxels                 1458       2915       5830

SS explained by Component 1:   1821320.36
predictable:                  28.94%
total:                         3.65%

-------------------------------------------------------------------------------
Component 2         minimum:  -0.31     maximum:  0.40

percentage of loadings            5%        10%        20%
abs threshold                   0.19       0.16       0.11
voxels above                    1458       2915       5830
positive voxels                  683       1245       2205
negative voxels                  775       1670       3625

SS explained by Component 2:   429377.61
predictable:                   6.82%
total:                         0.86%




-----------------------------------------------------------
Creating non rotated Images:
-----------------------------------------------------------
   image: unrotated_Component_1.img
location: G\2_components\unrotated\Images\


minimum: -0.50      maximum: 0.19
  abs threshold: 0.29
   voxels above: 2915
positive voxels: 0
negative voxels: 2915

           Avg   Loadings        Max
    +vox:  0.05       1263       0.19
    -vox: -0.17      27888      -0.50
variance: 1063.86   28.94%    3.65%

Positive:   total           5%         10%         20%
Loadings:    1263           0           0           0
Minimum :    0.00        0.00        0.00        0.00
Maximum :    0.19        0.00        0.00        0.00
Mean    :    0.05        0.00        0.00        0.00

Negative:   total           5%         10%         20%
Loadings:   27888        1458        2915        5830
Minimum :   -0.00       -0.32       -0.29       -0.25
```

```
Maximum :   -0.50       -0.50       -0.50       -0.50
Mean    :   -0.17       -0.36       -0.33       -0.30

MNI coordinates for cluster peaks at 10% threshold for component 1

Local maximum for positive part..
No positive loadings above threshold

Local maximum for negative part..

        Volume  Peak MNI Coordinate
  Voxels   (mm)    x    y    z  peak loading value
-------------------------------------------------------
  1328    84992        4    8   56         -0.50463
   640    40960      -28  -56  -28         -0.42898
 -- Complete MNI list contained in G\2_components\unrotated\G_MNI_unrotated.txt

-------------------------------------------------------
   image: unrotated_Component_2.img
location: G\2_components\unrotated\Images\


minimum: -0.31      maximum: 0.40
  abs threshold: 0.16
   voxels above: 2915
positive voxels: 1245
negative voxels: 1670

           Avg   Loadings        Max
    +vox:  0.07      11969       0.40
    -vox: -0.07      17182      -0.31
variance:  250.80    6.82%    0.86%

Positive:   total           5%         10%         20%
Loadings:   11969          683        1245        2205
Minimum :    0.00         0.19        0.16        0.11
Maximum :    0.40         0.40        0.40        0.40
Mean    :    0.07         0.25        0.21        0.18

Negative:   total           5%         10%         20%
Loadings:   17182          775        1670        3625
Minimum :   -0.00        -0.19       -0.16       -0.11
Maximum :   -0.31        -0.31       -0.31       -0.31
Mean    :   -0.07        -0.22       -0.20       -0.16

MNI coordinates for cluster peaks at 10% threshold for component 2

Local maximum for positive part..

        Volume  Peak MNI Coordinate
  Voxels   (mm)    x    y    z  peak loading value
-------------------------------------------------------
  1012    64768      -24  -92   -8         0.39838
    61     3904        4    0   64         0.23887
 -- Complete MNI list contained in G\2_components\unrotated\G_MNI_unrotated.txt


Local maximum for negative part..

        Volume  Peak MNI Coordinate
  Voxels   (mm)    x    y    z  peak loading value
-------------------------------------------------------
   455    29120       40  -20   52         -0.31326
   400    25600       16  -60   16         -0.26978
 -- Complete MNI list contained in G\2_components\unrotated\G_MNI_unrotated.txt

       Extract G    13:03:30     13:03:54     0 Minute 24 Seconds
-------------------------------------------------------------------------
         Elapsed    13:03:30     13:03:55     0 Minute 24 Seconds
Subjects:   4  [   1712 x  29151 ]  Runs:  2
       G:      [   1712 x    128 ]  C:\CPCAManual\Analysis\Gheader.mat
      GZ:      [    128 x  29151 ]

-------------------------------------------------------------
Rotating 2 components from GZ
-------------------------------------------------------------
```

```
------------------------------------------------------------
--- varimax orthogonal iter: 500 power: 1.00 gamma: 1.00 ---
--- G_varimax_orthogonal_i500_p1.00_g1.00.mat ---
------------------------------------------------------------
filename: output_G_varimax_orthogonal_i500_p1.00_g1.00.txt
original location: C:\CPCAManual\Analysis\G\2_components\varimax\
date created: 06-Jun-2013
created version: cpca 1.1.0(05)
 edited version:
 displaying top 10%

----------------------------------------
General Information
----------------------------------------
  Subjects:   4
  Runs:       2
  Total Scans:       1712
  Voxel Width:       29151

  Model Type:        FIR
  Onsets measured in scans
  Conditions:        4
  Bins:       8
  TR:         3.00

----------------------------------------
% of variance in original BOLD signal regressed out in CPCA preprocessing
----------------------------------------
         Total Removed:     28.72% (   14332795.40)
          Linear Trend:     23.77% (   11863422.67)
       Quadratic Trend:      4.95% (    2469372.73)
         Head Movement:      0.00% (          0.00)
          User Defined:      0.00% (          0.00)

Extreme Positive negative loading for unrotated components:
------------------------------------------------------------------------------
Component 1         minimum:  -0.52     maximum:  0.22

percentage of loadings            5%        10%        20%
abs threshold                   0.32       0.27       0.22
voxels above                    1458       2915       5830
positive voxels                    0          0          1
negative voxels                 1458       2915       5829

SS explained by Component 1:   1412076.76
predictable:                  22.44%
total:                         2.83%

------------------------------------------------------------------------------
Component 2         minimum:  -0.41     maximum:  0.26

percentage of loadings            5%        10%        20%
abs threshold                   0.23       0.20       0.17
voxels above                    1458       2915       5830
positive voxels                   21         48        125
negative voxels                 1437       2867       5705

SS explained by Component 2:   838621.21
predictable:                  13.33%
total:                         1.68%




------------------------------------------------------------
Creating rotated Images:
------------------------------------------------------------
   image: varimax_Component_1_orthogonal_i500_p1.00_g1.00.img
location: G\2_components\varimax\Images\


minimum: -0.52      maximum: 0.22
  abs threshold: 0.27
   voxels above: 2915
```

```
positive voxels: 0
negative voxels: 2915

          Avg   Loadings        Max
    +vox:  0.05       2831       0.22
    -vox: -0.15      26320      -0.52
variance:  824.81   22.44%    2.83%

Positive:   total           5%         10%         20%
Loadings:    2831           0           0           1
Minimum :    0.00        0.00        0.00        0.22
Maximum :    0.22        0.00        0.00        0.22
Mean    :    0.05        0.00        0.00        0.22

Negative:   total           5%         10%         20%
Loadings:   26320        1458        2915        5829
Minimum :   -0.00       -0.32       -0.27       -0.22
Maximum :   -0.52       -0.52       -0.52       -0.52
Mean    :   -0.15       -0.36       -0.33       -0.28

MNI coordinates for cluster peaks at 10% threshold for component 1

Local maximum for positive part..
No positive loadings above threshold

Local maximum for negative part..

        Volume  Peak MNI Coordinate
  Voxels   (mm)    x    y    z  peak loading value
-----------------------------------------------------------
  1306    83584      -32  -60  -28         -0.43178
   745    47680        4    8   56         -0.52077
 -- Complete MNI list contained in G_MNI_varimax_orthogonal_i500_p1.00_g1.00.txt

-----------------------------------------------------------
   image: varimax_Component_2_orthogonal_i500_p1.00_g1.00.img
location: G\2_components\varimax\Images\


minimum: -0.41      maximum: 0.26
  abs threshold: 0.20
   voxels above: 2915
positive voxels: 48
negative voxels: 2867

          Avg   Loadings        Max
    +vox:  0.05       2527       0.26
    -vox: -0.12      26624      -0.41
variance:  489.85   13.33%    1.68%

Positive:   total           5%         10%         20%
Loadings:    2527          21          48         125
Minimum :    0.00        0.23        0.20        0.17
Maximum :    0.26        0.26        0.26        0.26
Mean    :    0.05        0.24        0.23        0.20

Negative:   total           5%         10%         20%
Loadings:   26624        1437        2867        5705
Minimum :   -0.00       -0.23       -0.20       -0.17
Maximum :   -0.41       -0.41       -0.41       -0.41
Mean    :   -0.12       -0.27       -0.24       -0.21

MNI coordinates for cluster peaks at 10% threshold for component 2

Local maximum for positive part..

        Volume  Peak MNI Coordinate
  Voxels   (mm)    x    y    z  peak loading value
-----------------------------------------------------------
    31     1984       36  -96   -4         0.25877
    14      896      -20  -92   -8         0.25760
 -- Complete MNI list contained in G_MNI_varimax_orthogonal_i500_p1.00_g1.00.txt


Local maximum for negative part..
```

```
        Volume  Peak MNI Coordinate
  Voxels   (mm)    x    y    z  peak loading value
-------------------------------------------------------
  1975   126400       40  -20   52         -0.41129
   311    19904       56  -64    4         -0.32277
 -- Complete MNI list contained in G_MNI_varimax_orthogonal_i500_p1.00_g1.00.txt

       Extract G    13:04:07     13:04:23     0 Minute 16 Seconds
-------------------------------------------------------------------------
         Elapsed    13:04:07     13:04:24     0 Minute 17 Seconds
>>
```

IX. File and memory management

IX.1. Location and portability of output files

The display of statistics for a given analysis is dependent on the .mat files existing in the current working directory. If these files are not in the current directory, the statistics display will not be enabled. In addition, to be able to view the statistics for a given functional network, that functional network (.img/.hdr) must be in the same Component_Images folder, within the working directory, as its associated image-loading*.mat file. The **G_unrotated.mat** that is written directly to the working directory can be moved, although the associated files in the Component_Images folder must be moved as well.

Now, when you select a Z Info file and change to a different location, the Z Info data from the original location is copied over, and any updates applied to the local version of the Z Info file. This will allow you to apply a new G model, process a different group, etc. without overwriting the original processed data. This local Z Info file will still reference the actual normalized data (Z1 - Zn).

X. Memory Allocation

X.1. Memory considerations

Due to sizes of some of the CPCA matrices, this table can be used as a rough estimate of the installed RAM required based on sample sizes. These estimates are based on 4 conditions and 8 estimated time bins.

# of Subjects	Suggested Installed RAM
1-50	2 GB
50-200	4 GB
200-300	8 GB
300-500	16 GB
500+	24+ GB

Figure 75: Memory usage estimates

X.2. Memory allocation in Linux OS

As Linux allocates memory to an application, it consumes Kernel cache space for that memory, and retains it to improve overall speed. As the MATLAB application does not take this into account, the application is forced to look only at unused memory, and not the available cache. This can result in MATLAB hanging or giving the **'Out of Memory'** message.

For a listing of cache and memory stats type '**free –m'** at the linux prompt :

	total	used	free	shared	buffers	cached
Mem:	3973	2420	1553	0	719	1077
-/+ buffers/cache:		622	3350			

total: Total system memory (Gb)
used: Amount of memory (Gb) being used or cached
free: Current amount of memory (Gb)available
cached: Amount of memory (Gb) being used to store information applications have
used: It is still available memory, but not all applications are aware of this.

There are three methods Linux users can use when they encounter this memory problem:

1. Without root access the only option is to reboot the machine. This forces the cache to write what needs writing to disk, and resets it.

2. Use the '**cache**' button in the '*System Information'* panel to clear the cache. You may have to enter a su password. (See #Appendix_II_Additional_CPCA_GUI_Features).

3. If you have root access you can enter the following at the linux prompt :

   su

   [enter pwd]

   sync

   echo 3 | tee /proc/sys/vm/drop_caches

   ******* Typing ***'sync'*** forces any data in cache to be written to disk if it needs to be. It is important to never forget this step. Typing ***'echo 3 | tee /proc/sys/vm/drop_caches'*** tells the kernel to release all memory currently in cache.

X.3. Common memory issues for Linux and Mac users

The main issue Linux and Mac users may encounter are memory issues (noted by an 'Out of Memory' message) when processing large numbers of Subjects, or very wide data, such as 2x2x2. The issue is due to the method of memory management used by the Linux and Mac kernels, and the manner in which MATLAB allocates memory for its' own use.

The Linux and Mac kernels both implement a 'used memory caching' method. What this means is whenever an application says it is done with the particular data it is using in memory, the area of memory is not erased, but retains the data in case the application requires it again. By keeping the data in memory, it reduces the number of times data is retrieved from disk, which can significantly **increase** the speed of applications.

This memory is still available to other applications should they require more memory.

The MATLAB application requires that memory to store any matrix be in a contiguous block. Unfortunately, the MATLAB application cannot determine if memory being returned by the kernel cache is contiguous, and thus will ignore it completely. If there is 500MB of free and available memory, but the largest free contiguous block is 220Mb, then the application will give the 'OUT OF MEMORY' error when trying to load in a 250MB matrix, even though the computer says it has 500MB available.

X.4. Setting the maximum partition value

To avoid the memory problems that can be encountered when working with large matrices of data the CPCA software often divides such matrices into multiple partitions, smaller blocks of data that are processed individually and then recombined back into one matrix at the end of processing. The **'Max partition Mem (Mb)'** value indicates the maximum amount of memory that will be allocated to process a given partition of data. When this value is changed by the user the number of partitions is adjusted accordingly. Thus the **'Max partition Mem (Mb)'** value determines if the data is analyzed as one block of data (one partition) or as multiple blocks of data (two or more partitions).

When setting the **'Max partition Mem (Mb)'** value one must consider balancing the time it takes to *load* data with the time it takes to *process* data. A smaller max partition memory value results in a greater number of partitions of smaller size. And vice versa, a greater max partition memory value results in a smaller number of partitions of larger size. Which is better? This depends on (1) the size of your data and (2) the amount of available memory.

With smaller data sets a *greater* **'Max partition Mem (Mb)'** value is favored to limit data loading time. Loading data (Input/Output operations, disk IO) is usually more time consuming than the memory operations involved in processing/manipulating data. This is why computers use caches, copies of data kept in memory, to avoid having to get the same data from disk again and again.

Nevertheless, with a very wide partition, memory problems can occur because all data is referenced from the beginning of the block of memory. If the data you want is at the end, scanning will start at the beginning of the block of data and move sequentially until that data is reached at the end of the block. With small blocks of data, this is not an issue, but when partitions approach gigabytes in size, the accessing of cells further into the block take progressively longer to manipulate. Thus with very large data sets a smaller max partition memory value is favored to limit manipulation demands.

The processing/memory demands are dependent on the width of a partition. There is a CPCA utility called **array_sizes** that will tell you how much memory a particular array will consume. The researcher must keep in mind that some operations may take up to three times

that memory value to perform. For example an operation such as A = A'*A, though working on itself, requires a copy of the original matrix, the transposition of itself, and the final resulting data. Thus, on a computer with 4GB of memory, the **'Max partition Mem (Mb)'** value should be kept to less than 1GB of memory. Even with 8GB or more of memory it is a good idea to keep the **'Max partition Mem (Mb)'** value between 500 and 800 MB to optimize the speed of data manipulation and disk reading.

Example: A data matrix of 50,000 scans by 25,000 voxels per scan will consume about 10 GB memory. Setting the **'Max partition Mem (Mb)'** to 9700 Mb would result in all the data being kept in a single partition. This would involve manipulating a block of data 25,000 values wide: This would take forever. Setting the **'Max partition Mem (Mb)'** to 195 Mb would result in the data being divided into 50 partitions, manipulating each partition would be fast but the 50 disk reads would be time consuming. In this case, a good compromise would be to set **'Max partition Mem (Mb)**' to 780 Mb, the resulting 25 partitions would be slow to manipulate, but not excessively so, and only 25 disk reads would be required.

Example: Data set of 50,000 scans with 25,000 voxels per scan			
Max partition Mem	9,700 Mb (10 GB)	780 Mb	195 Mb
Partition size	50 000 x 25 000	50 000 x 2000	50 000 x 500
Partition number	1	25	50
Loading time / Disk reads	Very Fast	Slow *but* Happy Medium	Very Slow
Processing time / Data manipulation	Very Very Slow	Slow *but* Happy Medium	Very Fast

Figure 76: Partition size and processing speeds

XI. Error Messages

XI.1. Z selection errors

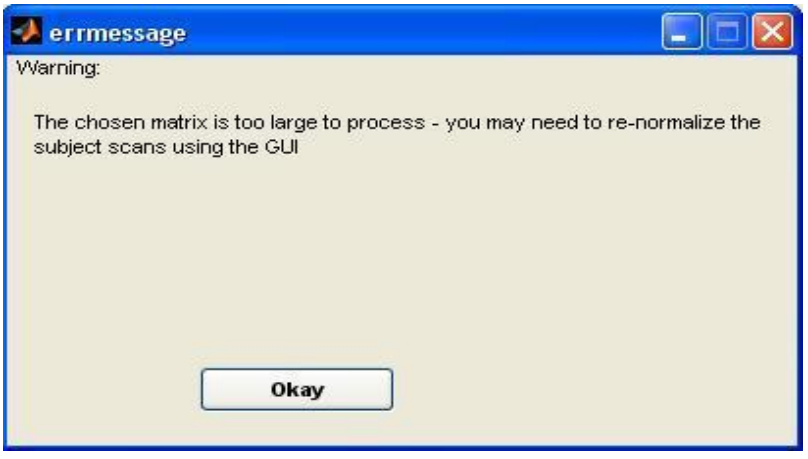


Figure 77: Z selection error A

Meaning: Your computer doesn't have enough memory to l**oad and process** the matrix.

Solution: Change computer.

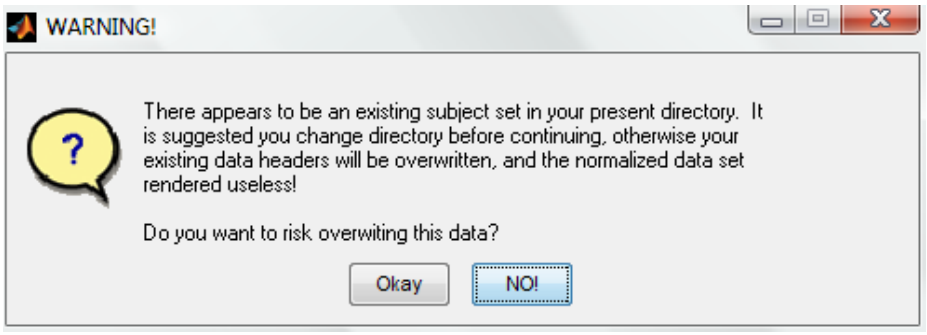


Figure 78: Z selection error B

Meaning: You are trying to create a new data set in a directory that already has a normalized data set.

Solution: Change the directory (folder).

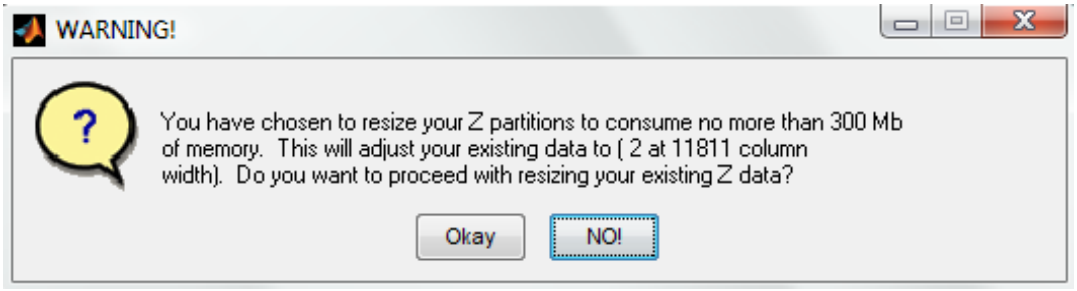


Figure 79: Z selection error C

Meaning: This is confirming you want to resize your column partitions.

Solution: Click **'Okay'** if you do, **'No'** if you don't.

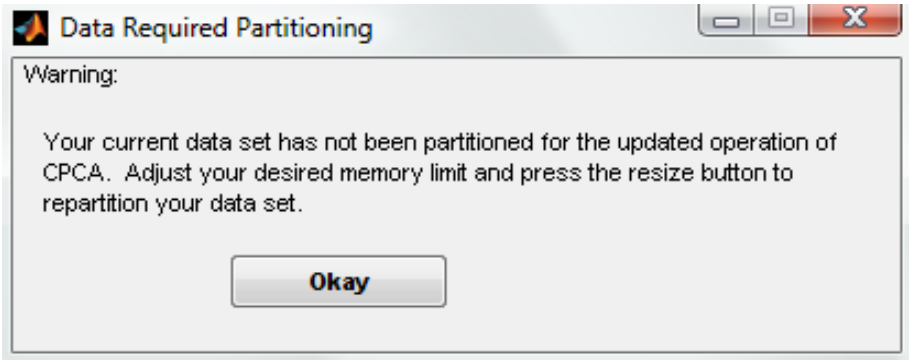


Figure 80: Z selection error D

Meaning: There is no partition information in this data set.

Solution: Press **'Okay'** to attempt to fix it, or use the Z Editor to repair it manually.

XI.2. Mask error message

Figure 81: Mask error A

Meaning: The interface can't find the mask.hdr image associated with the mask.img.

Solution: Make sure both files are in the same directory, if so, exit and open the GUI again or reboot your computer.

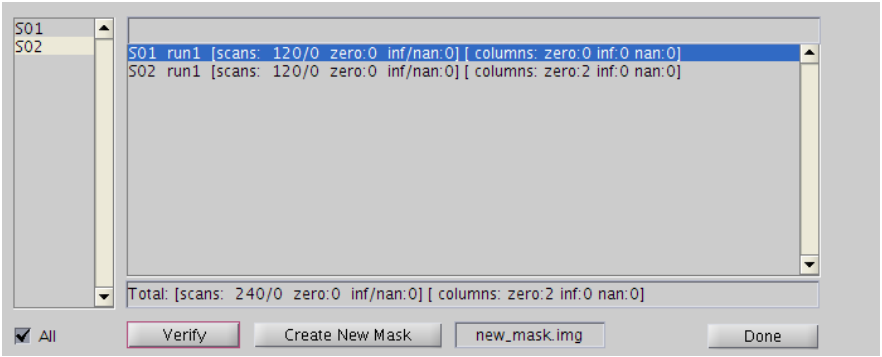


Figure 82: Mask error B

Meaning: The mask you are applying to the imagery contains invalid data.

Solution: Click on '**Create New Mask'** to have a new mask created with the voxels showing no variance removed before application.

XI.3. G selection error

Figure 83: G selection error A

Meaning: A red '**Edit**' button indicates an error with G (a malformed matrix) so rank is out of order.
Solution: Click on the red '**Edit**' button or look at a file called G_ranking.txt in the G_segs directory to see information about the problem(s).

References

Cairo, T. A., Woodward, T. S. & Ngan, E.T.C. (2006). Decreased encoding efficiency in schizophrenia. Biological Psychiatry, 59(8), 740-746.

Metzak, P.D., Riley, J., Wang, L., Whitman, J.C., Ngan, E.T.C. & Woodward, T.S. (2011). Decreased efficiency of task-positive and task-negative networks during working memory in schizophrenia. Schizophrenia Bulletin, DOI: 10-1093/schbul/sbq154.

Woodward, T. S., Cairo, T. A., Ruff, C. C., Takane, Y., Hunter, M.A. & Ngan, E.T.C. (2006). Functional connectivity reveals load dependent neural systems underlying encoding and maintenance in verbal working memory. *Neuroscience, 139*(1), 317-325.