

## Analysis of fMRI data within Brainvisa Example with the Saccades database

*Note : All the sentences in italic correspond to informations relative to the specific dataset under study*  
TP participants are asked to perform all the instructions starting with an arrow

---

### Outline

1. fMRI data hierarchy
  2. Importation of fMRI data within Brainvisa
  3. Pre-processing of fMRI data (use of FSL within Brainvisa)
  4. General linear model
  5. fMRI group analysis
  6. Using the commands as script
  7. Advanced use (cortical surface mapping)
- 

### 1. fMRI data hierarchy

First we can take a look at an existing database. In order to do this, we need to add it to BrainVISA:

→ *BrainVISA/Preferences/Databases/* → *add INPUTDATA=/tsi/medikit/tp-data/brainvisa/fmri/Saccades*

Then, we can access to the example of Saccades data:

*bv/data management/database browser* -> *select INPUTDATA=/tsi/medikit/tp-data/brainvisa/fmri/Saccades*

This database already contains two protocols: 'Pilot\_Saccades' and 'Saccades'

- 'Pilot\_Saccades' will be used to make the full analysis (FSL pre-processing+GLM) on two subjects.

- 'Saccades' (18 subjects already pre-processed) will be used for second-level analysis.

Now we create a new database: *DBPATH= /scratch/[login]/TPIRMf*

→ *create directory /scratch/[login]/TPIRMf*

→ *BrainVISA/Preferences/Databases/* → *add /scratch/[login]/TPIRMf*

### 2. Importation of fMRI data within Brainvisa

The importation process takes an image somewhere on your disk and put it within an existing database. It is assumed that:

- i. the fMRI data has been converted from DICOM to NIFTI format (4D image using the SPM converter or the MRICro one). Each fMRI session corresponds to a 4D image.
- ii. the EPI distortion have been corrected using suitable solutions (to discuss with the physicists responsible for fMRI acquisitions).

Trick: To convert a series of 3D NIFTI images to a 4D NIFTI image, use:

*bv/fMRI/Import/fMRIimages/ImportfMRI3Dto4D.py*

You can thus import images that have been pre-processed using SPM or FSL (outside BrainVisa).

Tip: To convert a series of 3D NIFTI images to a 4D NIFTI image, use:

*bv/fMRI/analysis pipeline/SPM\_preprocessings/step 7/ transformation of fMRI series.*

*Example of Saccades data:*

*bv/fMRI/Import/fMRIImages/Generic Import of fMRI dataset (4D Volume) +  
INPUTDATA/Pilot\_Saccades/[ad080125/cg080108]/fMRI/SaccadesLents/Session0[1-2]/Session0[1-2].nii.gz*  
(4 images overall).

**Fill the following entries:**

Database: DBPATH  
Data type: fMRI NIFTI-1 image  
File format: NIFTI-1 image  
protocol: Pilot\_Saccades  
subject: [ad080125]  
acquisition: SaccadesLentes  
session: Session01  
preprocessing: It must be "<any>"  
distortion correction: yes

→ run it for each session, and provide a name for each session

**Importing the experimental paradigm :**

Inputs:

- paradigm file
- Sessions
- Experimental conditions

Output:

- Experimental paradigm
- Information file

**Fill the following entries :**

Input : paradigm file

*INPUTDATA/Pilot\_Saccades/[ad080125/cg080108]/fMRI/SaccadesLents/Minf/paradigm.csv*

Ouput:

Database : DBPATH  
Data type : Paradigm  
File format : CSV File  
protocol : pilot\_Saccades  
subject : ad080125 / cg080108  
acquisition : SaccadesLentes  
Meta Informations: 'Ffx fMRI informations'  
type: 'csv file'

sessions : 'Session01' 'Session02' (separated by spaces not commas)

experimental conditions : 'Left' 'Right' (separated by spaces not commas)

→ run it for each subject [ad010128, cg080108]

**Importation of T1 images**

You should be able to do it !

Input data for the two subjects can be found in

*INPUTDATA/Pilot\_Saccades/[ad080125/cg080108]/t1mri/SaccadesLents/[ad080125/cg080108].nii[.gz]*

### **3. Pre-processing of fMRI data**

For fMRI data pre-processing, Brainvisa relies either on SPM or FSL. In this session, we will use only FSL

pre-processings. The different pre-processing steps have to be performed in the following order:

### 3.1 Slice timing correction

Slice timing correction (STC) consists in coregistering temporally the data; this is important for quick event-related paradigms. Otherwise STC can safely be bypassed.

This requires for each session:

- the fMRI data
- the repetition time, i.e. the time interval between two volumes. (Saccades: 2.4s)
- the order of the slices: ascending/descending, interleaved or sequential (Saccades: ascending interleaved )

The output of the data is the temporally coregistered 4D data. This process can be iterated across sessions and subjects.

*Example of Saccades data:*

*[bv/fMRI/analysis\\_pipeline/FSL\\_preprocessings/step 1\[...\]](#)*

*protocol Saccades, Subjects [ad080125/cg080108], all sessions*

*Options :*

*TR : 2.4*

*Slice\_order : Ascending*

*interleaved : Yes*

- This can be iterated through several sessions and subjects

➔ run it for two subjects

### Create a reference image

Simply take one 3D image from one session as a template to realign the other images.

Inputs:

- the first session series
- the volume selected as a reference (first, middle, last...)

Output:

- the reference image

- This can be iterated across subjects

➔ run it using one session

### 3.2 Motion estimation and correction (realignment)

This preprocessing step consists in detecting and possibly correcting the motion of the subject during the scanning session. The motion consists of 6 rigid body parameters.

Inputs:

- a 4D images
- a reference image

Outputs:

- the coregistered data
- the realignment parameters (6 rigid body parameters per image)

*Example of Saccades data:*

*[bv/fMRI/analysis\\_pipeline/FSL\\_preprocessings/step 2\[...\]](#)*

*protocol Saccades, Subjects [ad080125/cg080108], all sessions*

- This can be iterated through several sessions and subjects

→ run it for each subject

### 3.3 Normalization of the anatomical image

This preprocessing step realigns the anatomical image to a T1 template. Note that this is just to prepare fMRI data normalization.

#### Inputs:

- anatomical image of the subject
- anatomical template: [/tsi/medikit/tp-data/brainvisa/fsl/data/standard/MNI152\\_T1\\_2mm.nii.gz](#)
- alignment needed: rigid, translation, small displacement

#### Output:

- transformation matrix
- the normalized anatomical image (resolution: 2\*2\*2 mm)

*Example of Saccades data:*

[bv/fMRI/analysis\\_pipeline/FSL\\_preprocessings/step 3\[...\]](#)  
*protocol Saccades, Subjects [ad080125/cg080108]*

*anatomical template : /tsi/medikit/tp-data/brainvisa/fsl/data/standard/MNI152\_T1\_2mm.nii.gz*

*Alignment : Incorrectly Oriented*

- This can be iterated across subjects

→ run it for one subject

### 3.4 Coregistration of functional and anatomical data and normalization of the fMRI data

This basically warps the fMRI data to the template, using the transformation estimated previously.

#### Inputs:

- fMRI dataset (one session)
- Anatomical image (raw)
- Anatomical image (normalized)
- previously computed affine displacement matrix
- alignment needed from fMRI to anatomical MRI: rigid, translation, small displacement

#### Output:

- normalized fMRI session

*Example of Saccades data:*

[bv/fMRI/analysis\\_pipeline/FSL\\_preprocessings/step 4\[...\]](#)  
*protocol Saccades, Subjects [ad080125/cg080108], all sessions*

*Alignment : Incorrectly Oriented*

- This can be iterated through several sessions and subjects

→ run it for two subjects

### 3.5 Smoothing

Smoothing of the fMRI data.

#### Inputs:

- fMRI series
- kernel width: a vector of smoothing values in x,y,z (fwhm in mm)

#### Output:

- fMRI smoothed image

*Example of Saccades data:*

*bv/fMRI/analysis\_pipeline/FSL\_preprocessings/step 5[...]*

*protocol Saccades, Subjects [ad080125/cg080108], all sessions*

*Smoothing Kernel : [5.0 5.0 5.0]*

- This can be iterated through several sessions and subjects
- ➔ run it for two subjects

## **4. General Linear Model**

Applying the GLM on fMRI data entails the following steps:

### **4.1 Computation of the Brain mask**

Computation of a brain mask for each subject

Inputs:

- fMRI series of all sessions for a given subjects
- threshold infThreshold in [0,1]
- threshold supThreshold in [infThreshold,1]

Output:

- a 3D mask image

NB: the mask can be wrong: play with parameter infThreshold to improve it (increase infThreshold to decrease the mask volume).

*Example of Saccades data:*

*bv/fMRI/analysis\_pipeline/first\_level\_glm/Compute Mask*

*or*

*bv/fMRI/analysis\_pipeline/first\_level\_glm/Compute One Mask*

*protocol Saccades, Subjects [ad080125/cg080108], all sessions*

- This can be iterated across subjects or done for all the subjects (use the appropriate process)
- ➔ run it for two subjects

### **4.2 Read/Import the paradigm**

This enters the paradigm using an Eprime stimulation file and converts it. Here we use a .csv file (produced by excel) and then create an information file that contains the experimental condition's IDs and the session IDs. This is done only once for all sessions in one subject.

Inputs:

- experimental paradigm in csv format
- the sessions names
- the experimental conditions names

Outputs:

- the name of the experimental paradigm in the database
- an information file

*Example of Saccades data:*

*bv/fMRI/Import/fMRIImages/Importation of experimental paradigm*

*protocol Saccades, Subjects [ad080125/cg080108]*

- This can be iterated across subjects
- ➔ This has already been done

### 4.3 Defining the design matrix

Given the paradigm description, we have to create the design matrix that will be used to obtain task-related activation maps. This is done for each session independently.

Inputs:

- Session directory
- Experimental paradigm
- Number of Frames in the Dataset
- The Information file
- The type of HRF : Canonical, Canonical with derivative, FIR model
- The type of drift and the corresponding options

Output:

- The design matrix

*Example of Saccades data:*

*[bv/fMRI/Analysis Pipeline/first\\_level\\_glm/Step 2\[...\]](#)*

*protocol Saccades, Subjects [ad080125/cg080108], number of frames:174, Session01/02, Canonical HRF with derivative, Cosine drift with 128s Frequency Cut option*

- This can be iterated across subjects
- ➔ run it for two subjects (2 sessions each)
- ➔ Look at the result

### 4.4 Applying the GLM (Single Session GLM)

Inputs:

- The fMRI dataset to analyse
- The Design Matrix
- The mask image
- The fitting algorithm (Kalman AR1, Kalman, Ordinary Least Square)
- the analysis session: a string describing the model

Output:

- The GLM object
- The GLM informations
- The residual variance matrix

*Example of Saccades data:*

*[bv/fMRI/Analysis Pipeline/first\\_level\\_glm/Step 4\[...\]](#)*

*protocol Saccades, Subjects [ad080125/cg080108], session Session01/02, Kalman AR1, model=default*

- This can be iterated across sessions and subjects
- ➔ run it for one subject (2 sessions)

### 4.5 Defining functional contrasts to perform tests

To create the contrasts, use the script in :

*[INPUTDATA/Saccades/scripts/createContrasts.py](#)*

*[\(INPUTDATA=/tsi/medikit/tp-data/brainvisa/fmri/Saccades/Pilot\\_Saccades\)](#)*

*It currently creates two contrasts: 'Left-Right' and 'Right-Left'*

*Adapt it to as to write the results in the*

*[DBPATH/Pilot\\_Saccades/\[ag080125/cg080108\]/fMRI/SaccadesLentes/glm/default/contrast.con file](#)*

(DBPATH= */scratch/[login]/TPIRMf*)

→ do it for two subjects

#### 4.6 Computing the statistical maps (contrasts)

Inputs:

- The contrast file
- The information file
- The type of file saving : Using contrast names or number

*Example of Saccades data:*

*bv/fMRI/Analysis Pipeline/first\_level\_glm/Step 5[...]*

*protocol Saccades, Subjects [ad080125/cg080108], All sessions, Contrast Names*

- This can be iterated across subjects
- run it for one subject.
- Look at the resulting html page in a web browser

### 5. fMRI Group analysis

#### 5.1 Using a new database

We need more than two subjects to perform group analysis !

- Copy the existing database to work on it  
cp -R INPUTPATH/Saccades to DBPATH/Saccades
- Then perform *Data Management/Update Databases/DBPATH*

#### 5.2 Defining a group of subjects

Inputs:

- The list of subjects to analyse
- The acquisition

Output:

- The group file

*Example of Saccades data:*

*bv/fMRI/Analysis Pipeline/group\_analysis/Build Group*

*Select all subjects in the database, acquisition*

- no need to iterate
- run it for one group

#### 5.3 Performing a One sample group analysis

Inputs:

- The group file
- The contrast to use
- The decision statistic
- The voxel level threshold
- The correction for comparisons

Outputs:

- The analysis results
- The contrast image

*Example of Saccades data:*

*bv/fMRI/Analysis Pipeline/group\_analysis/One-sample group analysis*

*Use contrast named : Left-Right*

- Can be iterated across contrasts

➔ run it for all contrasts

## 5.4 Checking whether the distribution of the signal is Gaussian

Inputs:

- The group file
- The contrast to use
- test on the parameters ('no normalization') or the t-values ('normalization by the noise variance')

Outputs:

- The normality test image

*Example of Saccades data:*

*bv/fMRI/Analysis Pipeline/group\_analysis/One-sample group analysis*

*Use any contrast named*

➔ Run it on any contrast, with and without 'normalization'

## 5.5 Detecting brain functional landmarks at the group level

This is similar to a one-sample group analysis: it detects the regions that are found active in a certain proportion of the subjects.

But the function requires additional parameters.

Inputs:

- The group file
- The contrast to use
- 4 specific parameters
  - a first level threshold of the z-map: only supra-threshold structures will be considered. We advise a value of 2 to 3 (this is in z/t scale), but value should be fine
  - a size threshold (in voxels) that discards all supra-threshold structures that are too small
  - reproducibility threshold: the number of subjects in which one is sure that there is an active spot (typically a small fraction of the total the group size)
  - a pvalue that controls the probability that the region prevalence is greater than the specified reproducibility threshold

*Example of Saccades data:*

*bv/fMRI/Analysis Pipeline/parcellation/Detection of Functional Landmarks*

*Use contrast named : Left-Right*

- ➔ run it on one contrast
- ➔ look at the 'landmark image': it provides the group model of the detected ROIs and a labelling of the individual instance
- ➔ look at the html page to get the location of the detected regions at the group level

## 5.6 Creating a parcellation of the brain volume

This will divide the brain volume into the pre-specified number of parcels.

This division can be purely geometric ("spatial model only") or take into account functional information

("functional parcellation"). We propose to use the purely geometric model only.

Inputs:

- The group file
- The contrast to use
- The number of parcels

Output:

- the brain volume parcellation

*Example of Saccades data:*

*bv/fMRI/Analysis Pipeline/parcellation/Definition of Parcels – spatial model only*

*Use any contrast*

## **5.7 Performing parcel-level group analysis**

This is similar to a one-sample group analysis, but performed on parcel-based signal averages instead of voxel signals .

Inputs:

- The group file
- The contrast to use

Outputs:

- the resulting parcel-based random effects map

*Example of Saccades data:*

*bv/fMRI/Analysis Pipeline/parcellation/Parcel-based random effects analysis*

*Use any t contrast*

- ➔ *Do it for one contrast*
- ➔ *Compare the result with the corresponding one-sample group analysis*

## **6. Use in script mode**

The scripts are

*INPUTDATA/Pilot\_Saccades/scripts/script\_first\_level\_saccades.py*