fMRI, the Star/Plus experiment and our toolbox

Francisco Pereira

February 13, 2002

1 fMRI

1.1 Overview

Functional Magnetic Resonance Imaging (fMRI) is a technique for obtaining three-dimensional images of activity in the brain throughout time as the owner of the brain performs a given task. More correctly, it measures the level of oxygenation in the blood with respect to a control baseline, which can be used as an indicator that neural activity took place.

1.2 Neural Activity/Vascular-Metabolic Changes/fMRI signal

Neural activity implies the consumption of oxygen. When this happens, there is an increase in the flow of oxygenated blood. The amount of oxygen brought in surpasses that which was consumed, and there is a temporary increase in the ratio of oxygenated to deoxygenated haemoglobin, which affects the fMRI signal [see mini-review paper for details about this].

[add stuff about fMRI != activity, how it is triggered by activity and how it changes as the activity pattern giving rise to it changes].

1.3 Experiments

1.3.1 Data

A fMRI scanner displays the values of the fMRI signal at all the points in a three dimensional grid (a.k.a. *image*) covering part of the brain. The resolution of this grid depends on the type of scanner used, the amount of brain to be imaged and also the time that the scanning process should take, among other things. The latter sampling rate (called the *TR* [what does TR mean? Relaxation time?] comes into play because the scanning is repeated time after time as the subject performs the task, to provide a dynamic picture of the signal.

The cells in the grid are called *voxels*, and typically have a volume of a few cubic millimetres. A voxel will contain at least hundreds of thousands of neurons. Apart from the main scanning process, a static image with much higher resolution, the *structural scan* is collected in order to obtain a detailed structure of the brain.

Each of the regions of interest (ROI) in an experiment, such as Broca's area or visual cortex, comprises a few hundred voxels, and usually the analyses range over these.

1.3.2 Experimental Design/Stimuli

1.3.3 Segmentation/Active Voxels/Averaging

1.4 What does it look like?

Describe simple visual stimulation experiment - flash checkerboard (8hz) - and what fMRI shows.

2 The Star/Plus experiment

The experiment was designed to engage several different cortical areas, in order to look at their interaction. The areas plausibly involved are known from several other studies.

In this experiment, the subject sees a sentence for 4 seconds, such as

"The plus sign is above the dollar sign"

then a blank screen for 4 seconds, and finally a picture such as

```
 +
---
 \$
```

for another 4 seconds, during which the subject must press a button for yes or no, depending on whether the sentence matches the picture seen or not. The subject is instructed to rehearse the sentence in his/her head until the picture is presented. rather than try to visualize the sentence immediately. The second variant, for which we also have data, switches the presentations of sentences and pictures, and the instruction is to keep the picture in mind until the presentation of the sentence.

This task engages several brain areas, such as visual cortex for reading and seeing the sentence, Broca's area for language processing, the Intra Parietal Sulcus for spatial visualization, motor cortex for pressing the button, etc. The exact way in which the areas coordinate varies across subjects, based on their mental and verbal abilities and also on the strategy followed (e.g. mentally rehearse the sentence until you see the picture vs making up a picture as you read the sentence and then match it with the picture being displayed). The subjects were instructed to try to rehearse the sentence and delay making a mental picture of its semantic content until the picture was presented. Similarly, in the picture first variant, the subjects should try not to express the picture as a sentence while they wait for sentence presentation.

The subject does this task 10 times (condition 2), a similar task where the sentence contains a negative 10 times (condition 3), and a control task where he or she looks at a fixed point in the screen (not surprisingly, this is called *fixation*). The order in which these repetitions are performed is randomized, and each repetition is called a *trial*.

3 Data, Tools, etc

3.1 Location

Data for all the subjects in the two experiments (sentence-picture and picture-sentence) will be kept in a machine called `multivac.cald.cs.cmu.edu`, in the directory `/usr1/fpereira/fmri_data` and the subdirectories `data-starplus-sp` or `data-starplus-ps`. It can be reached from other machines in the cald lab through `/MULTIVAC/usr1/fpereira/fmri_data`.

Inside a given experiment directory there are directories for individual subjects, e.g. `04772_20_sp3terC`, inside which are subdirectories for a number of purposes. For our purposes, there are two important ones, `data` and `data_distilled`, in which are kept raw data and distilled representations of data, respectively. The following sections list the contents of these directories in more detail, and all the directory names will assume we are within a subject's directory unless otherwise stated.

Code is in `/MULTIVAC/usr1/fpereira/fmri_code`, and you can use the matlab command `addpath` to let matlab know about it. More details about what has been implemented and made usable is in one of the sections below.

3.2 Raw Data

The scanner records the fMRI signal over a grid with 64*64*8 voxels throughout the performance of all these task repetitions. A typical trial lasts around 27 seconds (4+4+4+15 rest), and an image of the grid is obtained every 1/2 second. This gives us a 54 point time course per trial per voxel, which looks like this (example taken from visual cortex):

[insert image here]

As the scanner is recording throughout the sequence of trials, we have a 1400 point time course for each of 32768=64*64*8 voxels. After scanning, this undergoes processing by a software package named FIASCO (made by the bold folks at the Statistics Department) which, among other things, removes much of the noise from several sources, performs motion correction, etc. At the end, we have a binary file for each subject, `data/detrend.dat`, containing the said 32768 time courses.

Many of these correspond to voxels outside the brain as, thankfully, the heads of the subjects are not cubical. Many other correspond to brain areas we are not interested in. This leaves us with a handful of ROIs, each of which comprises a few hundred voxels. The dataset has already been partitioned so that the time courses of the voxels of each area are kept in a separate matlab file.

3.3 Matlab Data

We have a small set of matlab functions that take the information in the binary plus other files, and produces a matlab file containing three arrays:

- data - contains the actual data for the voxels in the ROI, normalized to percentage above average signal value during control baseline (fixation trials)
- info - contains information about the trial sequence - trial types, lengths and other parameters
- meta - contains information that is not trial dependent (right now, mappings between voxel coordinates and column # in the data array

The matlab file for a given ROI is `data/`detrend.¡ROI¿.mat+.

3.3.1 Data

The `data` cell array contains the 1400 point time course for each voxel segmented trial wise across several arrays. The following is an example for a ROI with n voxels and t 54 point trials:

$$trial(1) = \begin{bmatrix} v_1(1) & v_2(1) & \dots & v_n(1) \\ v_1(2) & v_2(2) & \dots & v_n(2) \\ \dots & & & \\ v_1(54) & v_2(54) & \dots & v_n(54) \end{bmatrix}$$

...

$$trial(t) = \begin{bmatrix} v_1(1) & v_2(1) & \dots & v_n(1) \\ v_1(2) & v_2(2) & \dots & v_n(2) \\ \dots & & & \\ v_1(54) & v_2(54) & \dots & v_n(54) \end{bmatrix}$$

Each cell contains one trial and the order reflects the trial order in the experiment.

3.3.2 Info

The `info` structure array contains additional information about each trial, where `info(t)` pertains to trial t. Here, `info(t).len` is the number of images, `info(t).cond` is the experimental condition (1-fixation,2-affirmative,3-negative).

We also have information about reaction times after presentation of the picture/sentence and until the button press, as well as error rates in answering the question. That information is not yet in the info array, but will be shortly.

3.3.3 Meta

The `meta` structure contains the remaining information that is not trial dependent. This is a recent addition, so right now it contains the mappings between the column # in the data matrix) to (x,y,z) cordinates in the MRI scan (this is matlab, origin is 1,1,1). Therefore, `meta.colToCoord(col,:)`, maps column # to coordinates and `meta.coordToCol(x,y,z)` does the reverse.

It also has `meta.ntrials` and `meta.nvoxels`, not that useful given that that information can be derived straight from the data cell array.

3.4 Matlab distilled data

The `data` cell array in the previous section contained the time courses for all the voxels in a given ROI. Typically, people use not these but a single or a few time courses to represent the signal in an entire ROI. An example of this is the spatial averaging of the active voxels.

I wrote some code that distills representative time courses from each ROI file into a single representative "megavoxel". Right now, it extracts and produces averages of active voxels (where *active* can be a number of different criteria), but it can be extended to do it for whatever else we come up with.

These files are stored in the `data_distilled` directory, and the name of the file for a given ROI and method of reducing dimensionality is `distilled.<method>.<ROI>.mat`.

The methods available are

- `ava` - active by false discovery rate (a few subjects don't have any, a few have tens)
- `tava` - top 5 active by my t-value calculation (this way everyone has a few active - simpler)
- `ctava` - top 5 active by CCBI t-value calculation (almost matches mine, but not quite)

The matlab file contains three cell arrays: `ddata`, `dinfo` and `dmeta`. Each cell array has one cell per condition. This corresponds to the possibility of having each method produce a different result by looking at different conditions. For instance, the active voxels if you contrast condition 2 with baseline may not be the same as those you'd find by contrasting condition 3.

As an example, in order to retrieve the top 5 voxel average of active voxels in CALC during condition 2, you'd load `distilled.tava.CALC.mat` into matlab and pull out `data=ddata{2}`.

An additional information is in the `dn` array, which contains a positive number if the method produced a proper result or 0 if something went astray (sometimes there are no active voxels, or the ROI is too small, or ...).

In addition, the file contains cell arrays `pvalues` and `pvalues_ccbi`, also with one cell per condition. `pvalues{3}{1}` contains the list of pvalues for the t-test on the difference of the mean signal between condition 3 and fixation.

pvalues{3}{2} contains the corresponding column numbers in the data array. This is for the t-tests that our code does, and there is an equivalent cell array name pvalues_ccbi for the t-tests done in the CCBI.

3.5 Behavioural data

Besides the actual data, we have:

- information about the experiment (the exact sequence of trial types - affirmative, negative, fixation, # images taken, etc)) and coordinates of voxels belonging to a given ROI, etc. This information is kept in the info and meta arrays.
- information about reaction time from the presentation of picture or sentence to deciding and answering, as well as whether the decision was correct.
- list of stimuli used (sentences or pictures, with the latter to be available soon).
- scores of the subjects in verbal and visuospatial (?) ability tests.

3.6 Typical processing and tools

3.6.1 Code

The code that is available and commented to the degree that it may be usable is:

- mri_averageInterVoxel.m - average within a voxel, across trials for the same condition
- mri_averageIntraVoxel.m - spatial average between voxels, keeps all trials, allows weighted averages
- mri_smoothLowPassFilter.m - very ad hoc filter to remove frequencies up from 1/4 sampling frequency (2Hz) (this is being tested, but give it a spin).
- - soon to come, a template for code that iterates over all subjects and ROIs, loading data and doing something.

The code has comments about inputs and outputs.