**From:** Laurence Hunt <[lhunt@fmrib.ox.ac.uk](mailto:lhunt@fmrib.ox.ac.uk)>

**Date:** 1 October 2012 16:06:49 GMT+01:00

**To:** Peter Smittenaar <[petersmittenaar@gmail.com](mailto:petersmittenaar@gmail.com)>

**Subject: Re:**

Hey Peter,

So I've just looked back at it (ps\_corr.m) and it should be fine to tease apart, but it's still a bit more customised for my epxt. than I would like.

Basically you need to load in each subject's timeseries (one datapoint per TR), load in the relevant times at which events happened (in seconds, relative to the beginning of the first volume), use this to create an upsampled time-locked data matrix (dimensions trials \* timepoints). To sanity check this, in a region like the ventral striatum then in most subjects you can see a clear evoked response if you just run imagesc(dat{i}).

You then create a design matrix for each subject (dimensions trials \* regressors), run regression, and then do a second-level regression to test for significant effects across subjects.

Hopefully it should be pretty clear which parts of my script do what, as it is commented. The only things that are really customised for each experiment are:

(a) loading in the times - this will vary from experiment to experiment, depending on how many different events you have per trial

and (b) creating the (single-subject) design matrix - I do this in a separate function

I've attached the script and hopefully all the dependent functions you'll need:

ols.m is a simple script Tim wrote that implements the GLM, runs a bit faster than glmfit.

reject\_artifacts.m is a simple 'artifact detection' script I wrote to try and throw out bad trials (e.g. as you'd do in MEG) via high variance/peak-to-peak height - but I ended up finding that it made very little difference to the results...

create\_design\_matrix.m is the function I use to generate the design matrix for each subject

It might be a bit of a bore trying to figure out how it works without loading in some example data. If you're on dropbox I can share the entire directory structure with you, containing timeseries, explanatory variables etc. and then you can just run it in matlab and go through it step-by-step. Just let me know.

Cheers

Laurence

**From:** Laurence Hunt <[lhunt@fmrib.ox.ac.uk](mailto:lhunt@fmrib.ox.ac.uk)>

**Date:** 4 October 2012 10:16:49 GMT+01:00

**To:** Peter Smittenaar <[petersmittenaar@gmail.com](mailto:petersmittenaar@gmail.com)>

**Subject: Re:**

Hi Peter,

On 4 Oct 2012, at 01:24, Peter Smittenaar wrote:

hey Laurence, do you normally extract timeseries as the mean of all voxels? Or do you take the first Eigenvariate? What I've found so far in your papers it seems you take the mean of the voxels...

I take the mean, but I imagine the first eigenvariate would work just as well (or better...)

The script is a little hard to run as it seems you have quite a few customized functions (findc, get\_homedir, and some others). I do the same (i.e. use my own little customized scripts), so I can sympathize :)

Apologies :)

get\_homedir is just a little things that I have because I keep the same directory structure on different computers, but in a different location. The simplest way to get around it is:

(i) put the stim\_act\_comp folder in a directory called 'projects', in a place of your choosing - e.g. C:\Folder1\Folder2\projects\stim\_act\_comp

(ii) replace the line [hd,sd] = get\_homedir; with  hd = 'C:\Folder1\Folder2'; wherever necessary (should only be in a couple of places)

I've attached the other functions that I think you'll need, just let me know when you find other ones...:)

Laurence

**From:** Laurence Hunt <[lhunt@fmrib.ox.ac.uk](mailto:lhunt@fmrib.ox.ac.uk)>

**Date:** 5 October 2012 10:41:20 GMT+01:00

**To:** Peter Smittenaar <[petersmittenaar@gmail.com](mailto:petersmittenaar@gmail.com)>

**Subject: Re:**

On 4 Oct 2012, at 21:52, Peter Smittenaar wrote:

Got it to work! Incredibly exciting to watch the heavily processed 'raw' data :)

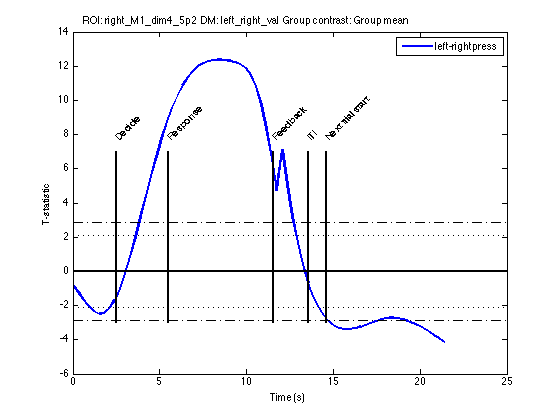
Not significant, I know, but still cool to see. STN more active when you fail to stop a response compared to when you successfully stop a response. This might partly explain why I have been utterly unable to find any STN activation...

Just wanted to share the joy. Thanks for all the help!

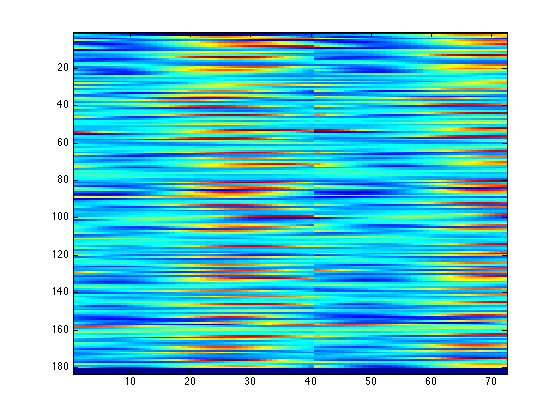
<image.png>

Hooray! Looks good. Am a bit surprised by the length of the HRF tho - even if it's triggered by the response, a 7s-to-peak HRF sounds quite long. A couple of sanity checks would be worthwhile (if you haven't already):

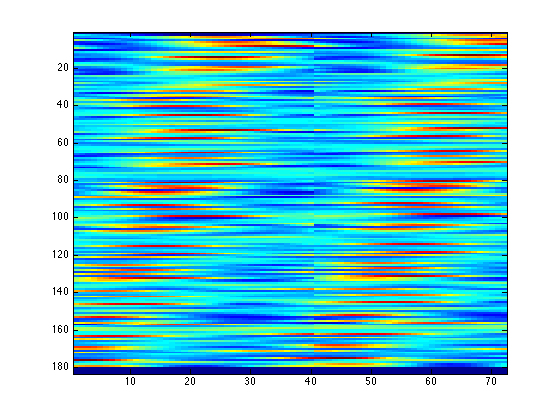
1. Pick a contrast that is absolutely massive, and make sure it has a sensible looking response. For example, in my experiment I can look in right primary motor cortex on trials where they picked left vs. trials where they picked right:



2. Check your TR (and other timings) are right. In an area which is quite task-responsive your data matrix should look like something like this:



But a 0.01s error in specifying the TR causes something like this (and messes everything up with the regression):



Laurence