

Using Randomise

Goals: These instructions will walk you through running your own randomise, permutation-based thresholding of a model. The instructions are for a very simple 1-sample t-test, but we're also supplying some instructions for if you had to run this for a model with additional regressors. For example, if you wanted to look at the linear relationship between age and your BOLD contrast, you can do that with randomise, but it takes a couple of more steps.

1. Start by creating a **randomise** directory to hold your output files. It will create a handful, so it is best to store these all in a directory. It is up to you where this directory will go.
2. Type "randomise" at the linux prompt and have a look at the help. The first line is the most useful, since it gives an example of the usage. Specifically, it says
`randomise -i <input> -o <output> -d <design.mat> -t <design.con> [options]`
3. **What is the input file?** You need to replace `<input>` with a 4D data set. Specifically this data set will contain the BOLD contrast estimates for all of your subjects, such that the 4th dimension is subjects (not time). Where will you get this file?
 - (a) **When you've already run the group Feat analysis.** Do you recall Jeanette's favorite file...ever? Her second stop if there's a problem in the analysis and one of the files we should inspect to QC our analysis? The dependent variable no matter what level of analysis you're running? That's right, the `filtered_func_data.nii.gz` file in the feat directory of interest. So, if you've already run a group analysis using Feat, you'll have all the files you need for randomise! Pretty convenient. Locate the `filtered_func_data.nii.gz` in your `.gfeat/cope#.feat/` directory of interest.
 - (b) **What if you didn't run a group Feat analysis and you don't want to?** If you don't have the group Feat analysis you'll need to create the 4D data set on your own. First, find each subject's `cope.nii.gz` image of interest. Importantly, make sure you find cope image that are already in standard space! If you're getting it from a within-subject, level 2 analysis stats directory (Level 2 that averages runs within-subject) that's already in standard space. If you only have 1 run per subject and are grabbing the cope images from a level 1 analysis, first check to see if it contains `reg_standard` directory. If it does, tunnel into that directory and you'll find your copes in standard space! If not, then you haven't run any higher level analyses and you can create the `reg_standard` directory by running `featregapply` on the first level feat directory using `featregapply featdir.feat`. This will create a `reg_standard` directory within your feat directory! Then you need to use `fslmerge` to concatenate the data over subjects, across the fourth dimension. For the precooked level1 analyses you've downloaded from openfmri, the `reg_standard` directories already exist. If you `cd` to the directory that contains the subject directories, the following should work. Note, cutting and pasting from this pdf likely won't work well, so type it out on your own don't enter a return before each file path, it should be all in a single line of code

```
fslmerge -t cope1_data_all
sub-10159/bart.feat/stats/cope1.nii.gz \
sub-10171/bart.feat/stats/cope1.nii.gz \
sub-10189/bart.feat/stats/cope1.nii.gz \
sub-10193/bart.feat/stats/cope1.nii.gz \
sub-10206/bart.feat/stats/cope1.nii.gz \
sub-10217/bart.feat/stats/cope1.nii.gz \
sub-10225/bart.feat/stats/cope1.nii.gz \
sub-10227/bart.feat/stats/cope1.nii.gz \
sub-10228/bart.feat/stats/cope1.nii.gz \
sub-10235/bart.feat/stats/cope1.nii.gz \
sub-10249/bart.feat/stats/cope1.nii.gz \
sub-10269/bart.feat/stats/cope1.nii.gz \
sub-10271/bart.feat/stats/cope1.nii.gz \
sub-10273/bart.feat/stats/cope1.nii.gz \
sub-10274/bart.feat/stats/cope1.nii.gz \
sub-10280/bart.feat/stats/cope1.nii.gz \
sub-10290/bart.feat/stats/cope1.nii.gz \
sub-10292/bart.feat/stats/cope1.nii.gz \
```

Go ahead and run it and then open your new file in the viewer of choice (fslview or fsleyes). You can also check that it has the correct number of volumes from the command line using `fslnvol cope1_data_all`. You should have 18. Play the movie in the viewer to run a quick QC on your data.

4. What is the output file? This is basically an output file root name. A lot of files will be generated, so this is where you'll use the randomise directory you created. You'd set it to something like, `/path/to/randomise/cope1_output`, where "randomise" is the directory you created yourself.
5. What is the design.mat? Again, if you've already run the group analysis, this file exists! `cd` into your group gfeat directory and have a look at the design.mat in your favorite text editor (not word :)) You'll see there's some header information and after that you'll see your design matrix! If you don't have a group Feat analysis, no worries. Use the `Glm` gui to make one. In linux you open it with `Glm` and on a mac you use `Glm_gui`. Change the "timepoint" field to how many subjects you have and then change the "timeseries" design to "Higher level" and the GUI will transform to what you saw in Feat previously. Set up your design and save it and you'll have your design.mat and your design.con files (coming up next).
Common question: **Why the heck doesn't the paste window work for me???** I've only ever gotten this to work with middle mouse button clicking for pasting. On a mac you can set your trackpad to do this, but you may need to install something like the MagicPrefs app. Ask me for help, if your'e interested.
6. What is the design.con? You should now have this if you've created the design.mat file. Open the file in a viewer and you'll again see some header info followed by your contrasts.

7. What other options? What thresholding strategy do we use? I highly recommend using the -T option, which is the TFCE. Hopefully I talked about this during lecture! The -c and -C flags run cluster size and cluster mass statistics. For reference, the -c 3.1 option will mirror what you do in the Feat GUI with random field theory most closely, as it is a cluster size statistic where the cluster forming threshold is set at 3.1. That said, TFCE is recommended, so use -T!
8. What other options? Number of permutations I'd set the -n flag to control the number of permutations. 5000 is publication ready and 1000 is a good place to start. May as well run 5000 for the lab. Note that it will run 5000 permutations for each contrast specified in the .con file. So, if you run 5 contrasts it will take 5 times longer than 1 contrast. Due to this, when I'm using my cluster I typically generate a separate .con file for each contrast so I can run the 5 in parallel. For today, just run a single contrast.
9. **What does the design.grp file do?** This has a completely different function in randomise compared to Feat! It is used to specify groups over which you want the permutations to be done. This is more advanced and you're not likely going to use it, so you won't need to include it. An example of when you might use it is if you had data grouped by family. In that case you can only permute data within family (and not between family) and the group file is used to group the subjects, accordingly.
10. **Shortcut to a 1-sample t-test!** Just replace the -d and -c flags with -1 (that's the number 1, not the letter l!). Voila! May as well use that for this lab.
11. **Set up your command and run it!** For today just use the 1-sample t-test shortcut.
`randomise -i <inputfile> -o <output root> -1 -n 5000 -T`
12. **Making sense of the output** Randomise creates 1-p-value maps. Actually, there are a lot of files! There's a guide here:
<http://fsl.fmrib.ox.ac.uk/fsl/fslwiki/Randomise/UserGuide>.
 Scroll down and look for the table and focus on the TFCE column. The files ending in `_tfce_corrptstat` are what you want, the 1-pvalue maps. Why not just the p-values? Because in our viewer we can only apply lower bounds, so to see p-values less than 0.05, we apply a lower bound threshold of 0.95 to the 1-p-value images.
Common mistake...you forget to manually threshold in the viewer! I've done this so many times! It looks like I have nice blobs and I get excited and then I realize the lower bound is 0, so I'm seeing a bunch of insignificant clusters. Make sure you set the lower bound to 0.95 and upper bound to 1.