1 Introduction

This is a short step-by-step description of how to process the data in bert-functional using the FreeSurfer Functional Analysis STream (FS-FAST). Anatomical and functional data have been collected from a subject code-named "bert". This is not meant to be an exhaustive tutorial on all the functionality in FS-FAST but rather a means to get users started on how to use FS-FAST, including interfacing with FreeSurfer features such as registration, rendering functional data in the volume using tkmedit as well as on the cortical surface using tksurfer. For more information on using FS-FAST, consult $FREESURFER_HOME/fsfast/docs, especially MGH-NMR-StdProc-Handbook.doc.

2 Unpacking

This data can be found in the FreeSurfer distribution in a file called bert.func.tar.gz. To begin processing, cd to the directory where you want the functional data to be stored. This directory will be called the Session Parent. Run the following command from the session parent directory:

```
tar xvfz <fsdist>/bert.func.tar.gz
```

This will create a directory called bert-functional. This will have 4 subdirectories:

- **3danat** - 3d anatomicals collected with this session. These are needed to automatically register the functionals with the reconstructed anatomicals found in $SUBJECTS_DIR/bert
- **bold** -location of the functional runs. This will have three subdirectories: 007, 008, and 009. Each of these will have a single volume with stem "f" stored in bshort format. bold will also have a file called seq.info in which information about the acquisition sequence is held.
- **t1epi** - location of a t1-weighted EPI volume whose slice prescription is identical to that of the functionals. The actual volume is stored in a subdirectory called 004 in bshort format with a stem called "f".
- **parfiles** - this is where the paradigm files for each run are stored. The paradigm file gives information about which stimulus was presented when (ie, the stimulus schedule). These files were created by an FS-FAST program called optseq.

3 Experimental Design

The experiment involves an associative memory task using an event-related design. The paradigm is described in detail in [**?**]. This tutorial data was not part of that study, and that study was not processed with FS-FAST. Russ Poldrack generously ran his experiment using a subject who had agreed to provide data for this tutorial.

The subject is shown one of 4 stimulus types. Each presentation lasted 3 seconds. Each stimulus has 5 words in the following pattern:

```
      targetword
testword1     testword3
testword2     testword4
```

The subject is asked to choose which test word is most related to the target word. By design, the target word is either highly related or loosely related to one of the test words. Sometimes there are four test words and sometimes there are two test words and two "words" with just X's in them.

This makes 4 conditions:

1. H2 - highly related, two words/two X's
2. H4 - highly related, four words
3. L2 - loosely related, two words/two X's
4. L4 - loosely related, four words

This number coding is used in the paradigm files to identify which stimulus was presented when. The contrast (L2+L4) - (H2-H4) should show activity in the left frontal region.

4 Processing Steps

4.1 Create a Study Directory

The study directory is the "home base" for running processing commands. The location of the study directory can be completely independent from the directories where the functional data are stored. It's best if you have only one study directory for a given project regarless of how many sessions belong to the project. It also avoids confusion if the study directory is NOT one of the session directories. All the programs are run from the study directory regardless of which session is being processed.

4.2 Create Session Id and Session Parent Files

The Session Parent file is also known as the Session Directory file. The Session Parent is the directory underwhich one or more sessions are found. The root of the session is called the Session Id. In this case, the Session Parent is the directory from which you run the tar command to unpack the data; this is the full path (ie, starting with /). The Session Id is "bert-functional". Create an ascii file called "sesspar" (the name can actually be anything) and set its contents to the Session Parent. Create another file called "sessid" (its name can also be anything) and set its contents to the Session Id. The processing programs will search all the directories listed in the Session Parent file for all the sessions listed in the Session Id file.

4.3 Copy the paradigm files into the functional run directories

cd into bert-functional. There will be three paradigm files in a subdirectory called "parfiles": sem_assoc-s001-r001.dat, sem_assoc-s001-r002.dat, sem_assoc-s001-r003.dat, one for each run. The paradigm file reprensents which stimulus was presented when during the run. The paradigm file for a run must be copied into the directory with the data for that run. The paradigm file can be called anything, but the name must be the same across runs. Here, we will call it sem_assoc.par:

```
cp parfiles/sem_assoc-s001-r001.dat bold/007/sem_assoc.par
cp parfiles/sem_assoc-s001-r002.dat bold/008/sem_assoc.par
cp parfiles/sem_assoc-s001-r003.dat bold/009/sem_assoc.par
```

4.4 Motion Correction

cd to the Study Directory and type:

```
mc-sess -sf sessid -df sesspar
```

This uses AFNI to motion correct the functional data to the first volume of the first run. After completion, there will be a new volume called fmc in each run directory. There will also be an ascii file called fmc.mcdat with the motion correction parameters for each time point. There are 10 columns in this file corresponding to: (1) time point number, (2) rotation (deg), (3) rotation (deg), (4) rotation (deg), (5) translation (mm), (6) translation (mm), (7) translation (mm), (8) RMS error before correction, (9) RMS error after correction, and (10) total (vector) translation (mm). Those who are using this program should cite [**?**].

4.5 Intensity Normalization

This stage is slightly misnamed because intensity normalization does not actually occur here. Rather, it segments the tissue from the air and computes the global mean of the fMRI signal inside the tissue. This number is later used to rescale the data so that, when intersubject averaging is done, all subjects have the same global mean. To intensity normalize the motion corrected data, cd to the Study Directory and run:

```
inorm-sess -sf sessid -df sesspar -motioncor
```

After processing, you will find a file called fmc.meanval in each run directory. This is an ascii file in which the global mean is stored. You will also find a file called fmc.report in which many statistics relating to the ”cleanliness” of your data are stored.

4.6 Create Analysis

An ”analysis” is a collection of parameters used to analyze each session. It encompasses the signal model, the stimulus schedule, and any preprocessing options such as smoothing and motion correction. The key here is to realize that you give a name to all these parameters and then refer to that name in later processing stages. Creating the analysis does not actually cause the data to be analyzed. The analysis is created ONCE and then applied to any and all sessions. If you want to change your analysis parameters, you should either create a new analysis (with a new name) or delete the original and analysis and create another one from scratch with the same name. For this tutorial, create an analysis called ”sem_assoc” using the following command:

```
mkanalysis-sess.new \
  -analysis sem_assoc-3b \
  -TR 2 \
  -paradigm sem_assoc.par \
  -designtype event-related \
  -funcstem fmc \
  -inorm \
  -fwhm 4 \
  -nconditions 4 \
  -timewindow 26 \
  -tprestim 4
```

The name of the analysis is passed with the -analysis flag. The TR is 2 seconds. The name of the paradigm file is called sem_assoc.par. This is an event-related design as indicated by -designtype. The -funcstem is the stem of the volume to be processed. In this case this is ”fmc” which is the volume created by the motion correction stage above. -inorm instructs the processor to use the intensity normalization file fmc.meanval created above to rescale the functional data to a global mean of 1000. In-plane spatial smoothing is indicated with the -fwhm flag (fwhm = full-width/half-max); in this case a fwhm of 4 mm is used. The number of conditions in the paradigm is given by -nconditions; note that this does not include the null or fixation condition (code 0 in the paradigm files). The argument of the -timewindow flag has been set to 26 seconds indicating the time range in an FIR signal model. The -tprestim 4 option instructs the processor to begin the FIR time window 4 seconds before the stimulus onset. This step will create a subdirectory in the Study Directory with the given analysis name. In that directory will be two files: analysis.info and analysis.cfg with the parameters indicated on the command line.

4.7 Average the Data for the Session

Step **??** created the analysis by collecting all the parameters needed to analyse the data. The actual analysis is done by the command:

```
selxavg-sess -sf sessid -df sesspar -analysis sem_assoc
```

This will separately analyze the data in each session indicated in the sessid file. After completion, there will be a new subdirectory under bold with the same name as the analysis. This directory will have two volumes: h-offset and h. h-offset is a map of the mean value at each voxel. h contains the estimated hemodynamic response at each voxel as well as the standard devation of the residual error. There is also an ascii file called h.dat with information about the parameters using during the analysis.

4.8 Create Omnibus Contrast

A *contrast* is an instantiation of a hypothesis. The omnibus contrast is a test for any task-related activity against the baseline. Here, the ”baseline” means the variance left unexplained after fitting the time course at each voxel for the mean, linear trend, and task-related activity. Mathematically, the mean of the baseline is forced to zero. The omnibus is usually tested first to assure that the data have been processed properly and that the subject is responding. If there is no omnibus activity, then there is no use in continuing the analysis. The omnibus is created using the command:

```
mkcontrast-sess \
   -analysis sem_assoc \
   -contrast omnibus \
   -a 1 -a 2 -a 3 -a 4 -c 0 \
   -nosumconds
```

As with mkanalysis-sess, making the contrasts is not the same as computing the contrast at each voxel. Making the contrast just collects all the information needed to compute the contrast. A contrast only needs to be made once. If you change the analysis (ie, the time window, the prestimulus window, or the number of conditions), the contrast will need to be remade. After making the contrast, you will see a file in the analysis directory called omnibus.mat.

4.9 Compute the Omnibus Contrast

The actual contrasts are computed with stxgrinder-sess:

```
stxgrinder-sess -contrast omnibus \
    -analysis sem_assoc \
    -sf sessid -df sesspar
```

This will create a directory called bert-functional/bold/sem_assoc/omnibus in which several volumes will be stored. Each volume corresponds to a different type of statistical test and so represents a different statistical map. These volumes are: f (value of the F ratio), fsig (significance of the F-test), t (value of the t ratio), sig (significance of the t-test, possibly mulitple maps for multiple post-simulus delays), minsig (the single best signifiance of the t-test (bonferroni corrected) at each voxel), and iminsig (the index of the best t-test signifiance at each voxel). These will be elaborated below.

4.10 View the Contrast on the Slices

```
sliceview-sess -contrast omnibus \
    -analysis sem_assoc \
    -sf sessid -df sesspar \
    -map fsig -slice mos -nohdr
```

This displays all the slices as a mosaic. The base image is the average functional image. The overlay is the signifiance of the F-test. Color indicates signifiances in the range of 2 (ie, $10^{-}2$) to 7 (ie, $10^{-}7$). The red/yellow scale indicates activations above baseline; the blue scale indicates activations above below baseline. Other maps (ie, f, sig, minsig, iminsig) can be viewed by changing the -map option (see below). For the omnibus test, only the fsig is relevant. If one wants to view only a single slice, use -slice SLICENO, where SLICENO is the zero-based slice number. The -nohdr option tells sliceview not to load the hemodyanmic response (see below).

4.11 Viewing the Hemodynamic Responses

The hemodynamic responses could have been viewed with the omnibus contrast, but it is more instructive to view them using an all-versus-baseline overlay. First, create the contrast:

```
mkcontrast-sess \
   -analysis sem_assoc \
   -contrast allvbase \
   -a 1 -a 2 -a 3 -a 4 -c 0
```

Note that this is similar to the omnibus created in Step **??** except that it does not include the -nosumconds flag. This means that the contrast will sum the conditions together whereas the omnibus tests each condition separately. The omnibus and all-v-baseline generally paint the same picture, but all-v-baseline allows for some more sophisticated display options. Compute the contrast maps:

```
stxgrinder-sess -contrast allvbase \
    -analysis sem_assoc \
    -sf sessid -df sesspar
```

View the contrast maps:

```
sliceview-sess -contrast allvbase \
    -analysis sem_assoc \
    -sf sessid -df sesspar \
    -map sig -slice mos
```

Note that the -nohdr flag has been removed which forces the hemodynamic averages to be loaded (this can take a while). Also the map is now sig instead of fsig, meaning that you are about to view the per-poststimulus-delay t-significance maps. When the slices are displayed, you will not see much activation. This is good because you are looking at the map of the activation 4 seconds *before* stimulus onset. Click in the window and hit “g”. This will bring up another window with four plots; each plot represents the unbiased hemodynamic response to each of the four conditions at the voxel selected in the image window. As you click in the image window, you will change the time courses in the plot window. Click in the plot window and hit “e”; this will display the standard error bars for each condition at each time point. Click in the image window and hit “+”. This will advance the map from that of 4 seconds before stimulus onset to 2 seconds before stimulus onset. You will also see a vertical line in the plot window appear at -2 sec. This bar indicates which post-stimlus delay you are currently viewing. Hitting “+” again will advance to the next frame thus allowing you to view the activation like a movie. Note, to get help with the keypress commands, click in either window and hit “h”.

4.12 A Contrast-of-Interest

Upto this point, you have only been looking at task-vs-nothing contrasts, which, while useful, are not particularly interesting. This contrast will compare the response to two tasks, loosely related with 4 words (L4) versus highly related with 2 words (H2):

```
mkcontrast-sess \
   -analysis sem_assoc \
   -contrast L4vH2 \
   -a 4 -c 1 -sumdelays
```

Using "-a 4 -c 1" means that positive values indicate that L4 is greater than H2. Conditions H4 and L2 are ignored. Normally, a separate t-test is done at each poststimulus delay. However, the -sumdelays flag indicates that the statistical test should be done after summing the hemodynamic responses across poststimulus delays. This can help bring out activation when the responses to two conditions are similar but one is slightly and consistently larger than the other over time. Compute and view the contrast:

```
stxgrinder-sess -contrast L4vH2 \
    -analysis sem_assoc \
    -sf sessid -df sesspar
sliceview-sess -contrast L4vH2 \
    -analysis sem_assoc \
    -sf sessid -df sesspar \
    -map sig -slice mos
```

In the 9^{th} slice, you should see a small patch of activation in the *posterior* portion of the Left Inferior Prefrontal Cortex (LIPC). In the 13^{th} slice, you should see a small patch of activation in the *anterior* portion of the LIPC. Note that the images are in radiological convention, so the activity will appear on the right side of the image. Also note that it is not possible to scroll through the different post-stimulus delays because they have all been collapsed to one image.

5 Interfacing with FreeSurfer

Linking into FreeSurfer allows the user to view the functional results on high-resolution 3D anatomical images as well as on the cortical surface. It also allows the user to resample into talairach, spherical, or region-of-interest spaces in order to perform intersubject averaging. Of course, to link to FreeSurfer, the subject must have been processed through the FreeSurfer anatomical processing stream (see $FREESURFER_HOME/docs/FreeS

5.1 The Subjectname File

One of the first steps in the anatomical processing stream is to create a subject directory using the command *mksubjdirs subjectname* where "subjectname" is a unique identifier for the subject that will be analyzed. This is also known as the *Subject Identifier String* . The subject's anatomical directory must be visible from $SUBJECTS_DIR (ie, you should see something when running the unix command *$SUBJECTS_DIR/subjectname*). To interface FS-FAST to FreeSurfer, you must create a file called subjectname in the functional session directory the contents of which should be the name of the subject. Note that it is the contents that hold the subject identifier string. The name of the file should actually be "subjectname". For this tutorial, go to bert-functional and create a file called subjectname with conents "bert" (no quotes). Note, run *ls $SUBJECTS_DIR/bert* to assure that bert exists on your system.

5.2 Functional/Anatomical Registration

The functional data and the anatomical data are collected using different resolutions, fields-of-view, and contrast weightings. Often, they are collected on different scanning sessions, sometimes years apart, and on different scanners. Yet, we still need to assign voxels in the anatomical space to voxels in the functional space in order to view the functional data on the anatomical. This is done in three stages, two of which are automatic.

5.2.1 Automatic Registation

The automatic registration can be used only if a high-resolution T1-weighted anatomical was acquired during the same scanning session. These may have been collected in order to process the subject with FreeSurfer, in which case there will probably be 2 or 3 very high-quality scans. Or, if a subject has already been reconstructed, there may only be one low-quality anatomical. The latter is the case for bert. One run of low-quality anatomicals can be found in bert-functional/3danat. These are referred to as the "same-session anatomicals" where as the those in $SUBJECTS_DIR/bert are referred to as the "database anatomicals". As mentioned above, the automatic registration is performed in two stages. First, the transform between the same-session anatomicals and the functional is computed based on the information about the field-of-view of each scan. Second, the transform between the same-session anatomicals and the database anatomicals is computed by performing a voxel-by-voxel comparison between the two and adjusting the positions so that the two volumes match best. Concatenating these two transforms yields the tranform between the functionals and the database anatomicals. Those using this procedure should cite [**?**]. This is accomplished using the command:

```
autoreg-sess  -sf sessid -df sesspar
```

This will create a file called register.dat in bert-functional/bold. This file will have 8 lines. The first four lines are: subjectname, in-plane resolution, between-plane resolution, and intesity value. The between-plane resolution is the slice thickness if there is no skip. The intensity value does not actually have anything to do with the registration itself; it is used when displaying the functional volume during the manual registration phase. The last four lines of register.dat have the values of the registration matrix that converts a location in anatomical space to a location in functional space.

5.2.2 Manual Registation

Manual registration is used to check and fine-tune the automatic registration or to perform the entire registation in the case that there are no same-session anatomicals or if the automatic registration fails. Even if the automatic registration succeeds, *always check the registation manually* using the following command:

```
tkregister-sess -sf sessid -df sesspar
```

This will bring up two windows, one with a brain, the other a control window. Hitting the "Compare" button in the control window will cause the image window to alternate between the functional and the anatomical volumes. Change the view by hitting the "Coronal" or "Sagital" or "Horizonal" buttons. The functional brain can be translated left or right by adjusting the horizontal scroll bar under "Translate Brain". It can be translated up and down using the vertical scroll bar. The brain can be rotated about the red cross using the "Rotate Brain" scroll bar. After adjusting the registration, hit the "SAVE REG" button (answer yes to overwrite).

5.2.3 Registration Used in Tutorial

Here are the contents of register.dat used in this tutorial:

```
bert
3.124990
6.000000
0.200000
 1.003765e+00 -6.999285e-02  8.023400e-02  3.269092e+00
 9.321700e-02  2.028884e-01 -9.838270e-01  1.762333e+01
 5.287400e-02  9.710374e-01  2.113440e-01  1.254106e+01
 0.000000e+00  0.000000e+00  0.000000e+00  1.000000e+00
```

6 Viewing the Functional Results on the Anatomical

In section **??**, we showed how to view the functional results overlayed on the original functional slices. In this section we show how to view the functional results on the high-resolution T1 anatomical volume.

```
tkmedit-sess -sf sessid -df sesspar -a sem_assoc -c omnibus -map fsig
```

You should see activation in the posterior LIPC at approximately talairach coordinates $(-54, 16, +28)$. You should see activation in the anterior LIPC at approximately talairach coordinates $(-58, 12, +2)$. There should also be activation in visual and motor cortices.

7 Viewing the Functional Results on the Surface

```
paint-sess -sf sessid -df sesspar -a sem_assoc -c omnibus -map fsig
surf-sess -sf sessid -df sesspar -a sem_assoc -c omnibus  -map fsig
```

References

[Collins, et al, 1994] D. L. Collins, P. Neelin, T. M. Peters and A. C. Evans, Automatic 3D Inter-Subject Registration of MR Volumetric Data in Standardized Talairach Space, Journal of Computer Assisted Tomography, 18(2) p192-205, 1994.

[Cox, 1996] RW Cox; AFNI: Software for analysis and visualization of functional magnetic resonance neuroimages; Computers and Biomedical Research, 29: 162-173, 1996.

[Cox and Jesmanowicz, 1996] RW Cox and A Jesmanowicz. Real-time 3D image registration for functional MRI. Magnetic Resonance in Medicine, 42: 1014-1018, 1999.

[Wagner, et al, 2001] Wagner, A.D., Pare-Blagoev, J., Clark, J. and Poldrack, R.A. Recovering Meaning: Left Prefrontal Cortex Guides Controlled Semantic Retrieval. Neuron 31:329-338. August 2, 2001.