

3dDeconvolve

Advanced Features

**Just in case you weren't
confused enough already**

Other Features of 3dDeconvolve - 1

- **-input1D** = used to process a single time series, rather than a dataset full of time series
 - ★ e.g., test out a stimulus timing sequence on sample data
 - ★ **-nodata** option can be used to check for collinearity
- **-censor** = used to turn off processing for some time points
 - ★ for time points that are “bad” (e.g., too much movement; scanner hiccup)
 - ★ **-CENSORTR 2:37** = newer way to specify omissions (e.g., run #2, index #37)
- **-sresp** = output standard deviation of HRF (β) estimates
 - ★ can then plot error bands around HRF in AFNI graph viewer
- **-errts** = output residuals (difference between fitted model and data)
 - ★ for statistical analysis of time series noise
- **-TR_times dt** = calculate **-iresp** and **-sresp** HRF results with time step **dt** (instead of input dataset TR)
 - ★ Can be used to make HRF graphs look better
- **-jobs N** = run with independent threads — **N** of them
 - ★ extra speed, if you have a dual-CPU system (or more)!

Other Features - 2

<http://afni.nimh.nih.gov/pub/dist/doc/misc/Decon/DeconSummer2004.html>

<http://afni.nimh.nih.gov/pub/dist/doc/misc/Decon/DeconSpring2007.html>

- Equation solver: Program computes **condition number** for **X** matrix (measures of how sensitive regression results are to changes in **X**)
 - ★ If the condition number is “bad” (too big), then the program will not actually proceed to compute the results
 - ★ You can use the **-GOFORIT** option on the command line to force the program to run despite **X** matrix warnings
 - But you should strive to understand why you are getting these warnings!!
- Other matrix checks:
 - ★ Duplicate stimulus filenames, duplicate regression matrix columns, all zero matrix columns
- ★ Check the screen output for **WARNINGS** and **ERRORS** ★
 - ★ Such messages also saved into file **3dDeconvolve.err**

Other Features - 3

- All-zero regressors *are* allowed (via `-allzero_OK` or `-GOFORIT`)
 - ★ Will get zero weight in the solution
 - ★ Example: task where subject makes a choice for each stimulus (e.g., male or female face?)
 - You want to analyze correct and incorrect trials as separate cases
 - What if some subject makes no mistakes? Hmmmm...
 - ➔ Can keep the all-zero regressor (e.g., all `-stim_times = *`)
 - ➔ Input files and output datasets for error-making and perfect-performing subjects will be organized the same way

- **3dDeconvolve_f** program can be used to compute linear regression results in single precision (7 decimal places) rather than double precision (16 places)
 - ★ For better speed, but with lower numerical accuracy
 - ★ Best to do at least one run **both** ways to check if results differ significantly (Equation solver *should* be safe, but ...)

Other Features - 4

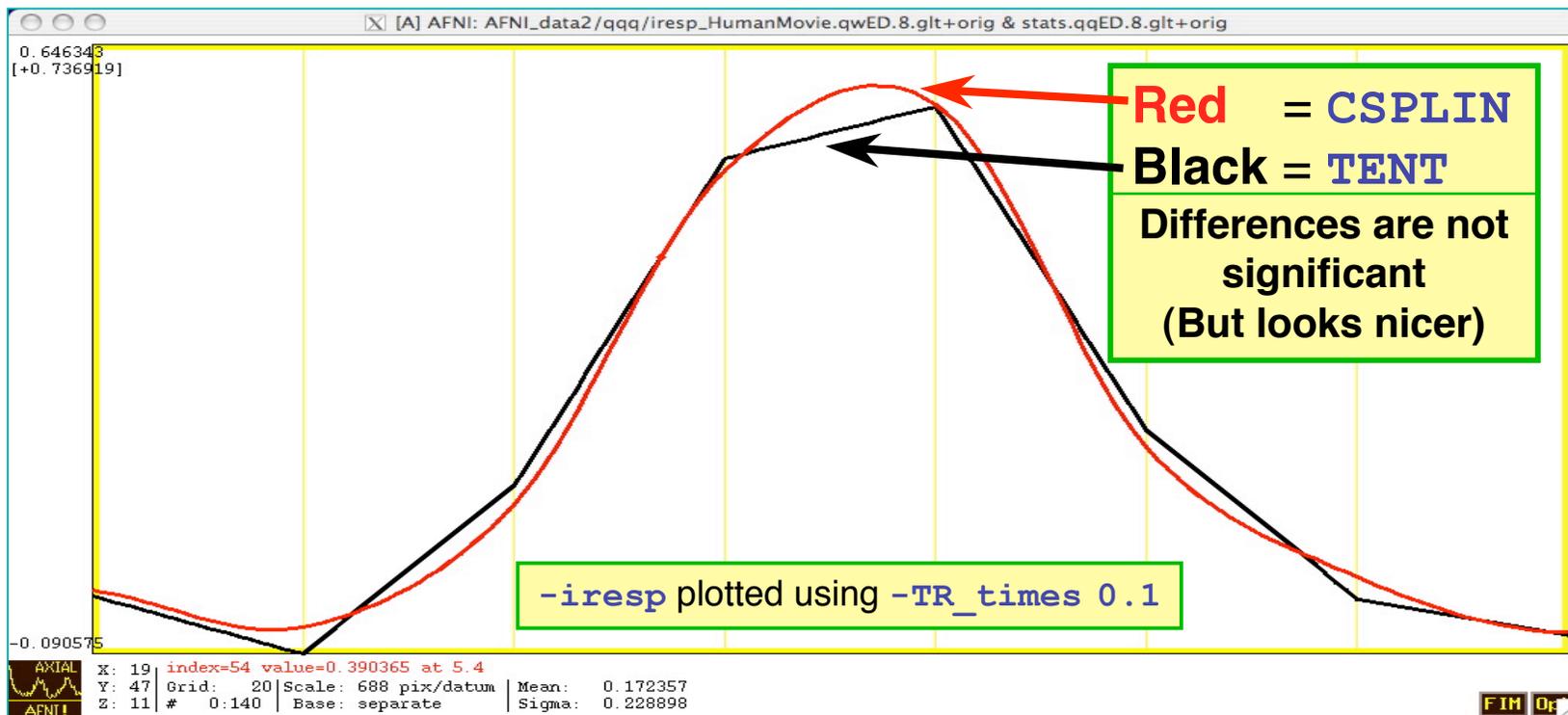
- Default output format is 16-bit short integers, with a scaling factor for each sub-brick to convert it to floating point values
 - ★ `-float` option can be used to get 32-bit floating point format output — more precision, and more disk space

- `3dDeconvolve` recommends a `-polort` value, and prints that out as well as the value you chose (or defaulted to)
 - ★ `-polort A` can be used to let the program set the detrending (AKA “high pass filtering”, since detrending removes low frequency content from data) level automatically

- `-stim_file` is used to input a column directly into **X** matrix
 - ★ Motion parameters (as in previous examples)
 - ★ If you create a stimulus+response model outside `3dDeconvolve` (e.g., using program `waver`)

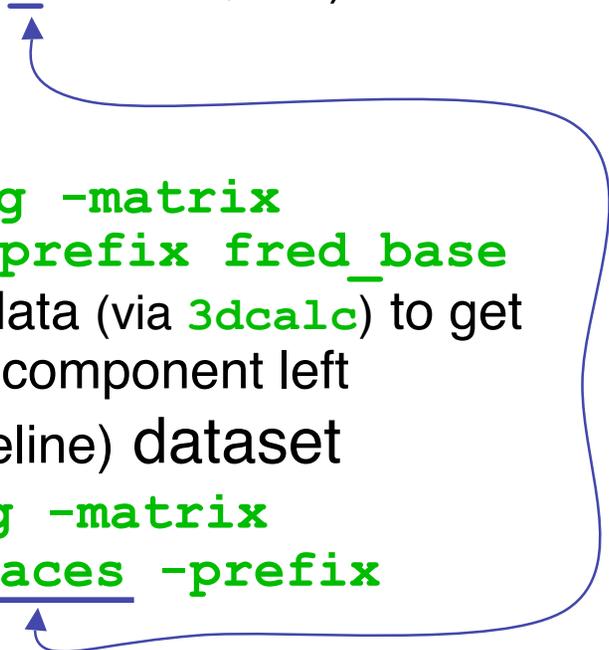
Other Features - 5

- `-stim_times` has some other basis function options for the HRF model besides **BLOCK** and **TENT**
 - ★ **CSPLIN** = cubic spline instead of **TENT** = linear spline
 - Same parameters: (`start, stop, number of regressors`)
 - Can be used as a “drop in” replacement for **TENT**



Other Features - 6

- **-fitts** option is used to create a synthetic dataset
 - ★ each voxel time series is full (signal+baseline) model as fitted to the data time series in the corresponding voxel location

 - **3dSynthesize** program can be used to create synthetic datasets from *subsets* of the full model
 - ★ Uses **-x1D** and **-cbucket** outputs from **3dDeconvolve**
 - **-cbucket** stores β coefficients for each **X** matrix column into dataset
 - **-x1D** stores the matrix columns (and **-stim_labels**, etc.)
 - ★ Potential uses:
 - Baseline only dataset
 - ↳ **3dSynthesize -cbucket fred+orig -matrix fred.xmat.1D -select baseline -prefix fred_base**
 - ↳ Could subtract this dataset from original data (via **3dcalc**) to get signal+noise dataset that has no baseline component left
 - Just one stimulus class model (+ baseline) dataset
 - ↳ **3dSynthesize -cbucket fred+orig -matrix fred.xmat.1D -select baseline Faces -prefix fred_Faces**
- 

Other Recent Small Changes

- Defaults are changed:
 - ★ **-nobout** & **-full_first** & **-bucket** & **-x1D** are always implied
 - ★ Names of statistics sub-bricks are slightly altered (to be more consistent)

- Checks if **-stim_times** inputs are out of range (AKA: the PSFB syndrome)
 - ★ Prints **WARNING** message, but continues analysis

- When using **-nodata** with **-stim_times**, it is important to give the number of time points and the TR, as in **-nodata 250 2.3**
 - ★ With **-input1D**, use **-TR_1D 2.3** to specify TR

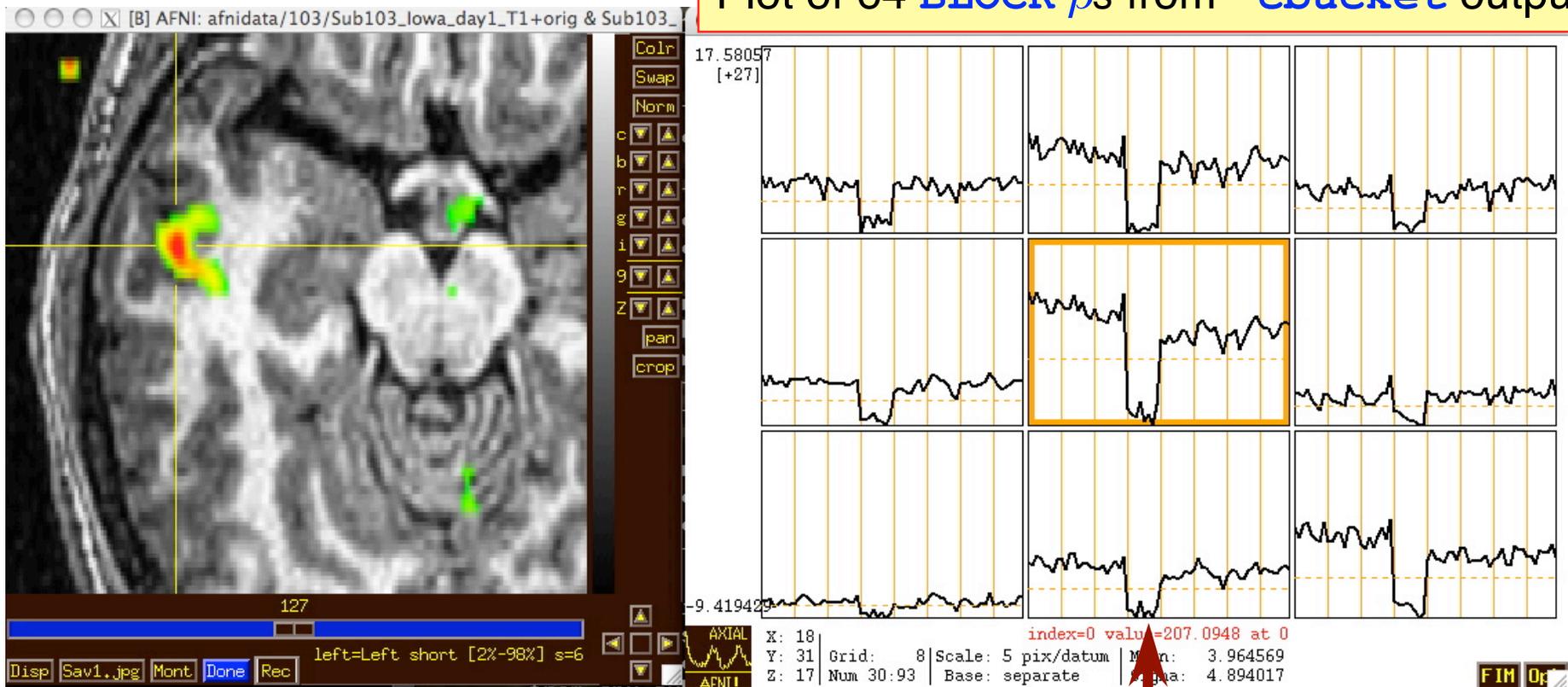
IM Regression - 1

- **IM** = Individual **M**odulation
 - ★ Compute *separate* amplitude of response for each stimulus
 - Instead of computing average amplitude of responses to multiple stimuli in the same class
 - ★ Response amplitudes (β s) for each individual block/event will be highly noisy
 - Can't use individual activation map for much
 - Must pool the computed β s in some further statistical analysis (*t*-test via **3dttest**? inter-voxel correlations in the β s? correlate β s with something else?)
 - ★ Usage: **-stim_times_IM k tname model**
 - Like **-stim_times**, but creates a separate regression matrix column for each time given

IM Regression - 2

- Only application of IM thus far has been in checking some data we received from another institution
- Experiment: 64 blocks of sensorimotor task (8 runs each with 8 blocks)

Plot of 64 **BLOCK** β s from **-cbucket** output



N.B.: sign reversal in run #4 = stimulus timing error!

IM Regression - 3

- IM works naturally with fixed-shape IRF analysis, which only give 1 amplitude parameter per stimulus
- With variable-shape (deconvolution) analysis, have multiple amplitude parameters per stimulus
 - ★ Difficulty: each event in same class won't get the same shaped HRF this way
 - ★ Desideratum: allow response shape to vary (that's deconvolution), but only allow amplitude to vary between responses in the same stimulus class
 - ★ Problem: get unknowns that multiply each other (shape parameters \times amplitude parameters) — and we step outside the realm of *linear* analysis
 - ★ Possible solution: **semi-linear** regression (nonlinear in global shape parameters, linear in local amplitude params)

AM Regression - 1

- **AM** = **A**mplitude **M**odulated (or **M**odulation)
 - ★ Have some extra data measured about each response to a stimulus, and *maybe* the BOLD response amplitude is modulated by this
 - ★ Reaction time; Galvanic skin response; Pain level perception; Emotional valence (happy or sad or angry face?)
 - Want to see if some brain activations vary proportionally to this **ABI** (**A**uxiliary **B**ehaviorial **I**nformation)
-

- Discrete levels (2 or maybe 3) of ABI:
 - ★ Separate the stimuli into sub-classes that are determined by the ABI (“on” and “off”, maybe?)
 - ★ Use a GLT to test if there is a difference between the fMRI responses in the sub-classes
- ```
3dDeconvolve ... \
-stim_times 1 regressor_on.1D 'BLOCK(2,1)' -stim_label 1 'On' \
-stim_times 2 regressor_off.1D 'BLOCK(2,1)' -stim_label 2 'Off' \
-gltsym 'SYM: +On | +Off' -glt_label 1 'On+Off' \
-gltsym 'SYM: +On -Off' -glt_label 2 'On-Off' ...
```
- “**On+Off**” tests for any activation in *either* the “on” or “off” conditions
  - “**On-Off**” tests for differences in activation *between* “on” and “off” conditions
  - Can use **3dcalc** to threshold on *both* statistics at once to find a **conjunction**

## AM Regression - 2

- Continuous (or several finely graded) ABI levels
  - ★ Want to find active voxels whose activation level also depends on ABI
  - ★ **3dDeconvolve** is a linear program, so must make the assumption that the change in fMRI signal as ABI changes is linearly proportional to the changes in the ABI values
- Need to make 2 separate regressors
  - ★ One to find the mean fMRI response (the usual `-stim_times` analysis)
  - ★ One to find the variations in the fMRI response as the ABI data varies
- The second regressor should have the form

$$r_{AM2}(t) = \sum_{k=1}^K h(t - \tau_k) \cdot (a_k - \bar{a})$$

- ★ Where  $a_k$  = value of  $k^{\text{th}}$  ABI value, and  $\bar{a}$  is the average ABI value
- Response ( $\beta$ ) for first regressor is standard activation map
- Statistics and  $\beta$  for second regressor make activation map of places whose BOLD response changes with changes in ABI
  - ★ Using 2 regressors allows separation of voxels that are active but are *not* detectably modulated by the ABI from voxels that *are* ABI-sensitive

# AM Regression - 3

- New feature of **3dDeconvolve**: `-stim_times_AM2`
- Use is very similar to standard `-stim_times`
  - ★ `-stim_times_AM2 1 times_ABI.1D 'BLOCK(2,1)'`
  - ★ The `times_ABI.1D` file has time entries that are “married” to ABI values:

|      |      |      |      |
|------|------|------|------|
| 10*5 | 23*4 | 27*2 | 39*5 |
| 17*2 | 32*5 |      |      |
| *    |      |      |      |
| 16*2 | 24*3 | 37*5 | 41*4 |
  - ★ Such files can be created from 2 standard ASCII .1D files using the new **1dMarry** program
    - The `-divorce` option can be used to split them up
- **3dDeconvolve** automatically creates the two regressors (unmodulated and amplitude modulated)
  - ★ Use `-fout` option to get statistics for activation of the pair of regressors (i.e., testing null hypothesis that *both*  $\beta$  weights are zero: that there is no ABI-independent *or* ABI-proportional signal change)
  - ★ Use `-tout` option to test each  $\beta$  weight separately
  - ★ Can **1dplot**  $X$  matrix columns to see each regressor

# AM Regression - 4

- The **AM** feature is new, and so needs more practical user experiences before it can be considered “standard practice”
  - ★ In particular: don’t know how much data or how many events are needed to get good ABI-dependent statistics
- If you want, `-stim_times_AM1` is also available
  - ★ It only builds the regressor proportional to ABI data directly, with no mean removed:
$$r_{AM1}(t) = \sum_{k=1}^K h(t - \tau_k) \cdot a_k$$
  - ★ Can’t imagine what value this option has, but you never know ... (if you can think of a good use, let me know)
- This one goes up to 11:
  - ★ You can have up to 11 amplitudes married to each stimulus time (insert obligatory polygamy/polyandry joke here)
    - How many ABI types at once is too many? I don’t know.

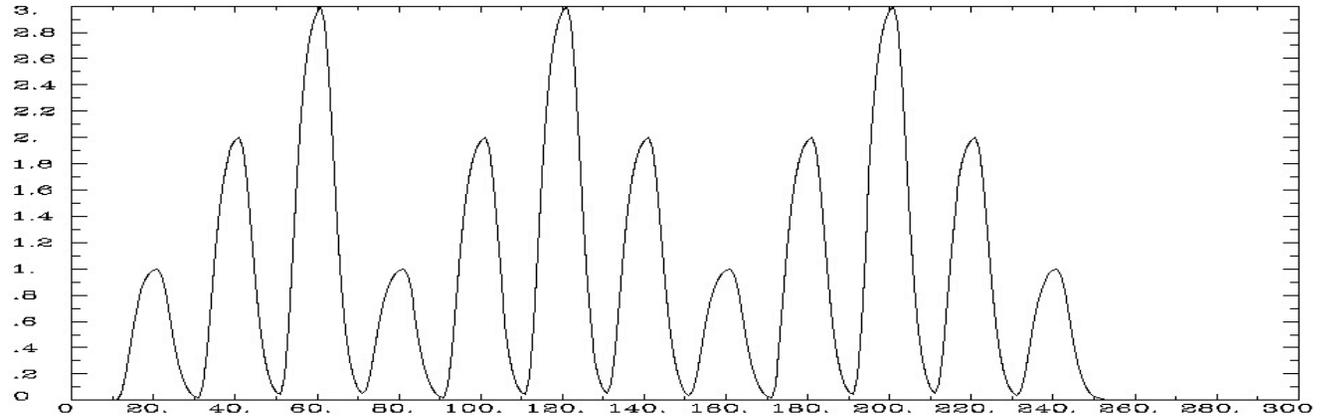
# AM Regression - 5

Timing: AM.1D = 10\*1 30\*2 50\*3 70\*1 90\*2 110\*3 130\*2 150\*1 170\*2 190\*3 210\*2 230\*1

- `3dDeconvolve -nodata 300 1.0 -num_stimts 1 \`  
`-stim_times_AM1 1 AM.1D 'BLOCK(10,1)' -x1D AM1.x1D`

- `1dplot AM1.x1D' [2]'`

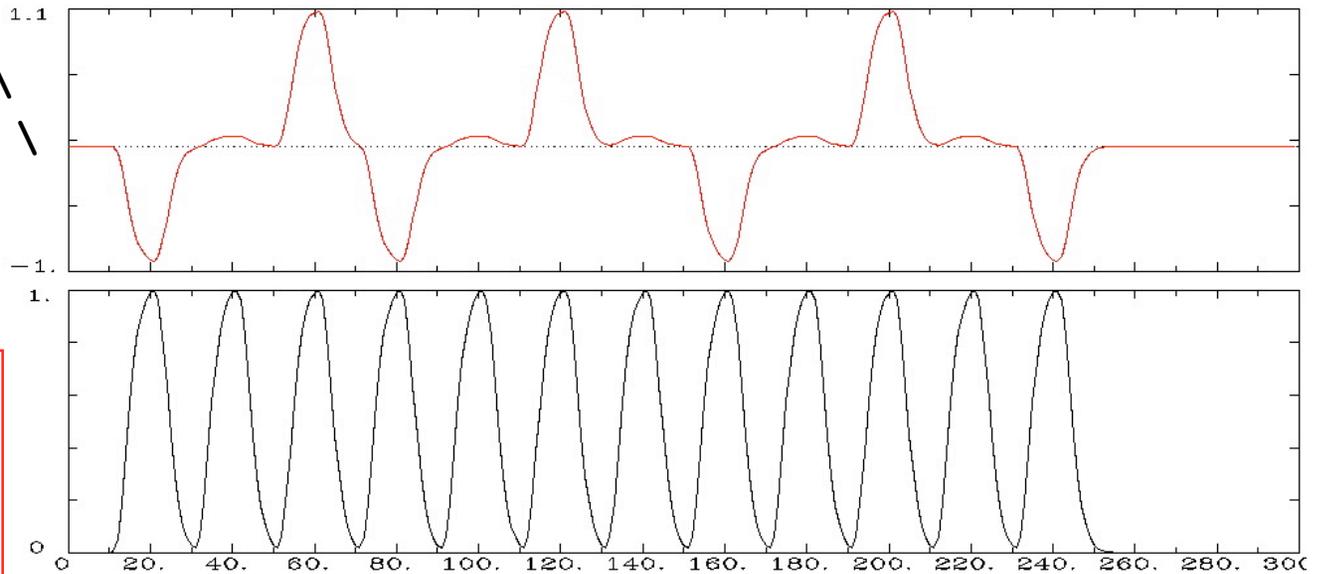
**AM1** model of signal  
(modulation = ABI)



- `3dDeconvolve -nodata 300 1.0 \`  
`-num_stimts 1 \`  
`-stim_times_AM2 1 \`  
`AM.1D 'BLOCK(10,1)' \`  
`-x1D AM2.x1D`

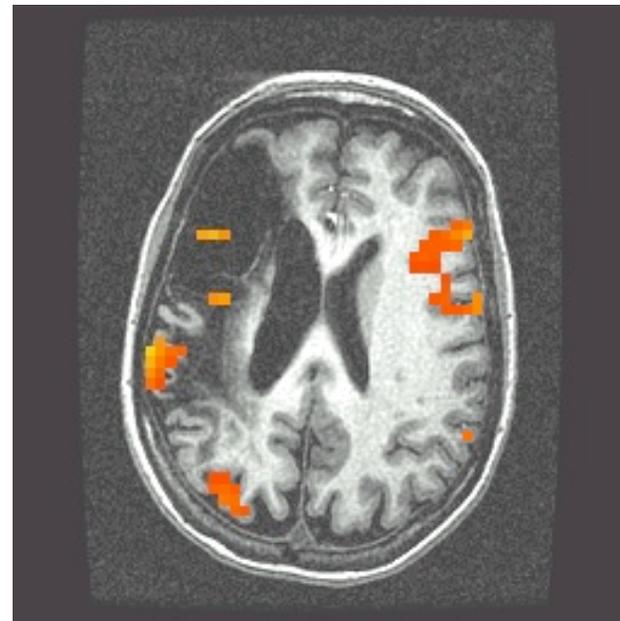
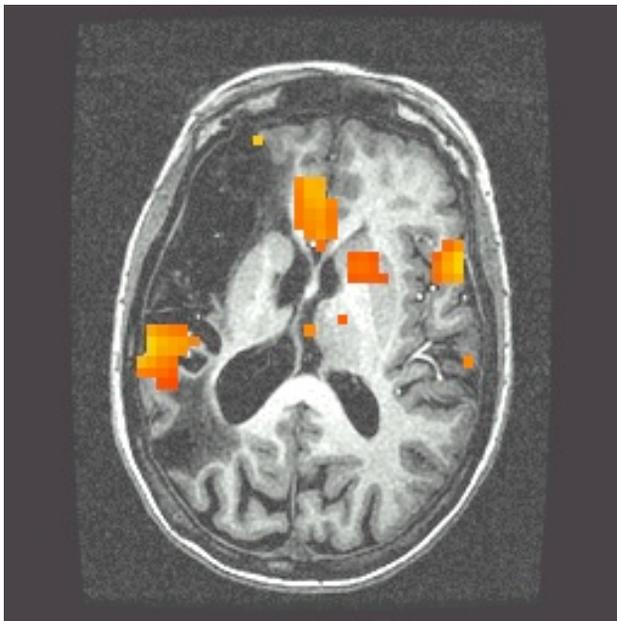
- `1dplot -sepscl \`  
`AM2.x1D' [2,3]'`

**AM2** model of signal:  
is 2D sub-space  
spanned by these 2  
time series



## AM Regression - 6

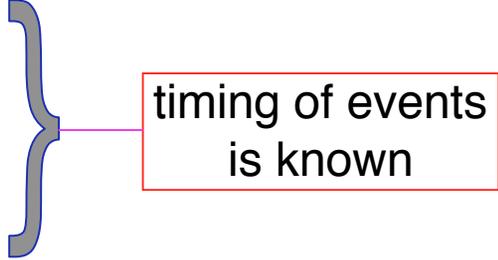
- First actual user: Whitney Postman (formerly NIDCD; PI=AI Braun)
- Picture naming task in aphasic stroke patient
- ABI data = number of alternative names for each image (e.g., “balcony” & “porch” & “veranda”, vs. “strawberry”), from 1 to 18
  - 8 imaging runs, 144 stimulus events
- 2 slices showing activation map for BOLD responses proportional to ABI ( $\beta_{AM2}$ )
  - What does this mean? Don't ask me!



## AM Regression - 7

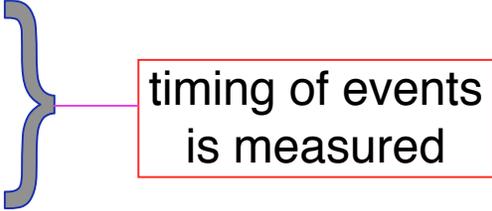
- Alternative: use **IM** to get individual  $\beta$ s for each block/event and then do external regression statistics on those values
- Could do nonlinear fitting via **3dNLfim**, or inter-class contrasts via **3dtttest**, **3dLME**, **3dANOVA**, etc.
- What is better: **AM** or **IM**+something more ?
  - We don't know – experience with these options is limited thus far – you can always try both!
  - If **AM** doesn't fit your models/ideas, then **IM** is clearly the way to go
  - Probably need to consult with SSCC to get some hints/advice

# Other Advanced Topics in Regression

- Can have activations with multiple phases that are not always in the same time relationship to each other; e.g.:
  - a) subject gets cue #1
  - b) variable waiting time (“hold”)
  - c) subject gets cue #2, emits response
    - ↳ which depends on both cue #1 and #2

timing of events is known
- ★ Cannot treat this as one event with one HRF, since the different waiting times will result in different overlaps in separate responses from cue #1 and cue #2
- ★ Solution is multiple HRFs: separate HRF (fixed shape or deconvolution) for cue #1 times and for cue #2 times
  - Must have significant variability in inter-cue waiting times, or will get a nearly-collinear model
    - ↳ impossible to tell tail end of HRF #1 from the start of HRF #2, if always locked together in same temporal relationship
  - How much variability is “significant”? Good question.

# Even More Complicated Case

- Solving a visually presented puzzle:
  - a) subject sees puzzle
  - b) subject cogitates a while
  - c) subject responds with solution

timing of events  
is measured
- The problem is that we expect some voxels to be significant in phase (b) as well as phases (a) and/or (c)
- Variable length of phase (b) means that shape for its response varies between trials
  - ★ Which is contrary to the whole idea of averaging trials together to get decent statistics (which is basically what linear regression for the  $\beta$  weights does, in an elaborate sort of way)
- Could assume response **amplitude** in phase (b) is constant across trials, and response **duration** varies directly with time between phases (a) and (c)
  - ★ Need three HRFs
  - ★ Can generate (b) HRF with **3dDeconvolve**'s **dmBLOCK**

# Noise Issues

- “Noise” in FMRI is caused by several factors, not completely characterized
  - ★ MR thermal noise (well understood, unremovable)
  - ★ Cardiac and respiratory cycles (partly understood)
    - In principle, could measure these sources of noise separately and then try to regress them out
      - ➔ RETROICOR program now available ([RetroTS.m](#))
  - ★ Scanner fluctuations (e.g., thermal drift of hardware)
  - ★ Small subject head movements (10-100  $\mu\text{m}$ )
  - ★ Very low frequency fluctuations (periods longer than 100 s)
- Data analysis should try to remove what can be removed and should allow for the statistical effects of what can't be removed
  - ★ “Serial correlation” in the noise time series affects the  $t$ - and  $F$ -statistics calculated by [3dDeconvolve](#)
  - ★ Next slides: new AFNI program for dealing with this issue

# Allowing for Serial Correlation

- $t$ - and  $F$ -statistics denominators: estimates of noise variance
    - ★ White noise estimate of variance:
      - $N$  = number of time points
      - $m$  = number of fit parameters
      - $N-m$  = degrees of freedom = how many equal-variance independent random values are left after the time series is fit with  $m$  regressors
- $$\hat{\sigma}^2 = \frac{1}{N-m} \sum_{i=0}^{N-1} [\text{data}_i - \text{fit}_i]^2$$
- **Problem:** if noise values at successive time points are correlated, this estimate of variance is biased to be too small, since there aren't really  $N-m$  independent random values left
    - ★ Denominator too small implies  $t$ - and  $F$ -statistics are too large!
    - ★ And number of degrees of freedom is also too large.
    - ★ So significance ( $p$ -value) of activations in individuals is overstated.
  - **Solution #1:** estimate correlation structure of noise and then adjust statistics (downwards) appropriately
  - **Solution #2:** estimate correlation structure of noise **and** also estimate  $\beta$  fit parameters using more efficient “generalized least squares”, using this correlation, all at once (REML method)

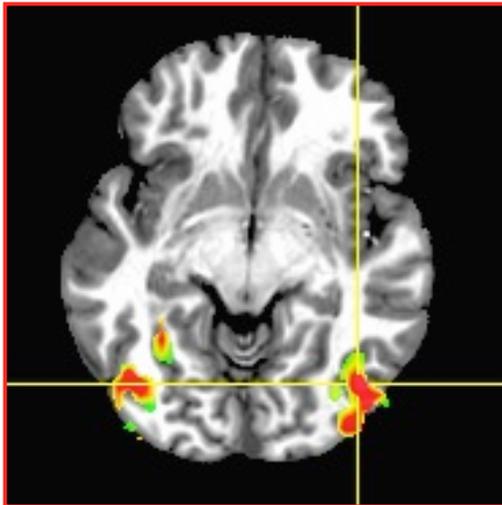
## New Program: 3dREMLfit

- Implements Solution #2
  - ★ REML is a method for simultaneously estimating variance + correlation parameters **and** estimating regression fit parameters ( $\beta$ s)
  - ★ Correlation structure of noise is ARMA(1,1)
    - 2 parameters **a** (AR) and **b** (MA) in each voxel
      - ➔ **a** describes how fast the noise de-correlates over time
      - ➔ **b** describes the short-range correlation in time (1 lag)
    - Unlike SPM and FSL, *each voxel* gets a separate estimate of its own correlation parameters
- Inputs to 3dREMLfit
  - ★ run 3dDeconvolve first to setup .xmat.1D matrix file and GLTs (don't have to let 3dDeconvolve finish analysis: `-x1D_stop`)
    - 3dDeconvolve also outputs a command line to run 3dREMLfit
  - ★ then, input matrix file and 3D+time dataset to 3dREMLfit
- Output datasets are similar to those in 3dDeconvolve

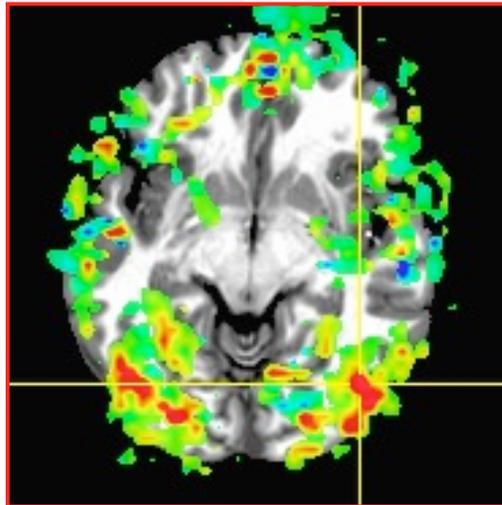
# Sample Outputs

- Compare with [AFNI\\_data3/afni/rall\\_regress](#) results
- `3dREMLfit -matrix rall_xmat.x1D -input rall_vr+orig -fout -tout \`  
`-Rvar rall_varR -Rbuck rall_funcR -Rfitts rall_fittsR \`  
`-Obuck rall_funcO -Ofitts rall_fittsO`

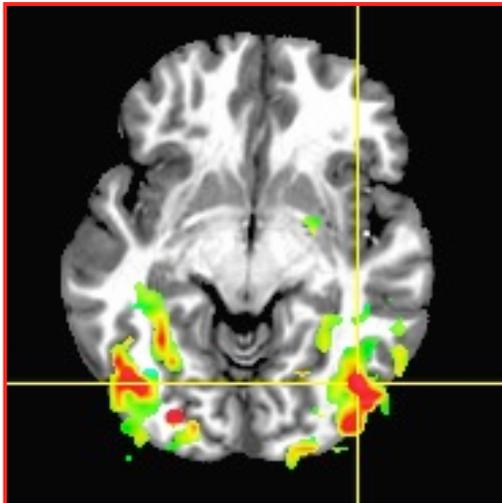
**REML**  
 $F=3.15$   
 $p=0.001$



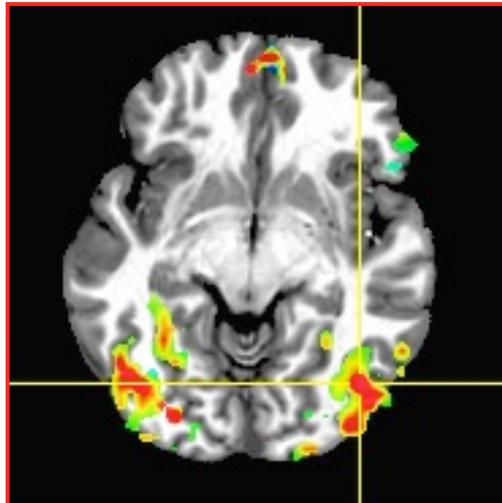
**OLSQ**  
 $F=3.15$   
 $p=0.001$



**REML**  
 $F=1.825$   
 $p=0.061$   
▪  $F$  = No activity outside brain!



**OLSQ**  
 $F=5.358$   
 $p=5e-7$   
▪  $F$  = No activity outside brain!



**O  
h  
M  
y  
G  
O  
D  
!?!**

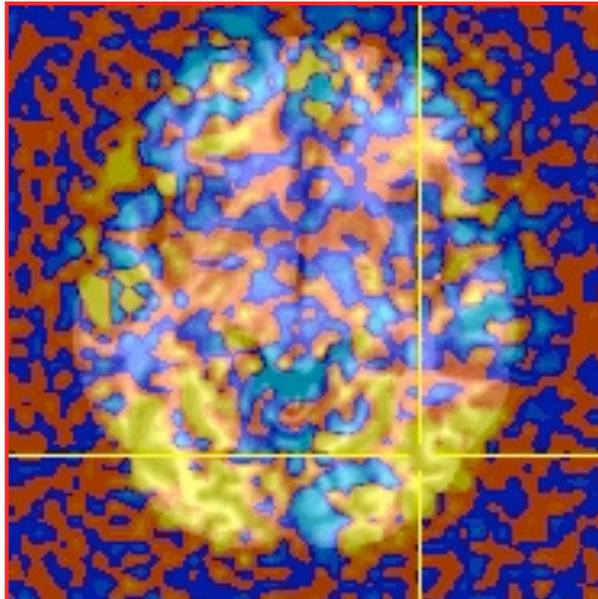
# It's Not So Bad: $\beta$ !

- For individual activation maps, **3dREMLfit**-ized  $t$ - and  $F$ -statistics are significantly different, and more accurate
- But ... There are at present very few applications for such individual FMRI activation maps
  - ★ pre-surgical planning
- For standard group analysis, inputs are only  $\beta$  fit parameters
  - ★ Which don't change so much between REML and OLSQ

**Color Overlay =  $\beta$  weight from analysis on previous slide, no threshold**

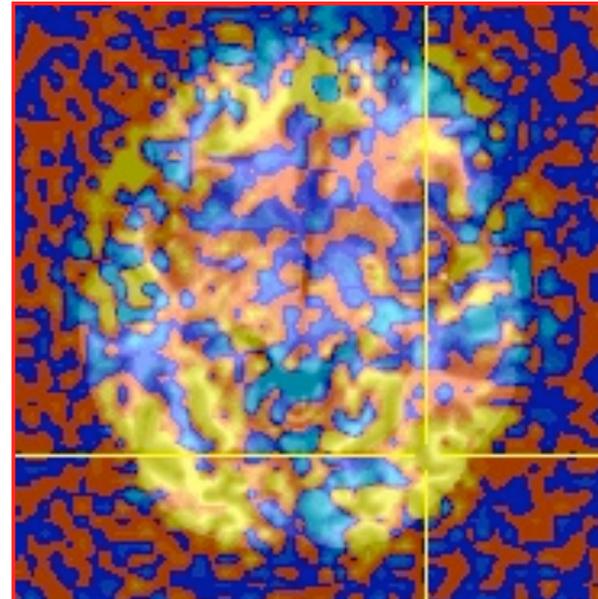
**REML**

**CPU  
500 s**



**OLSQ**

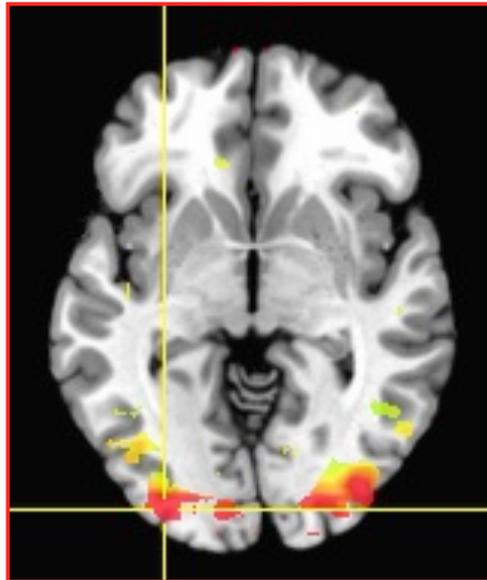
**CPU  
156 s**



# It's Not So Bad At All: Group!

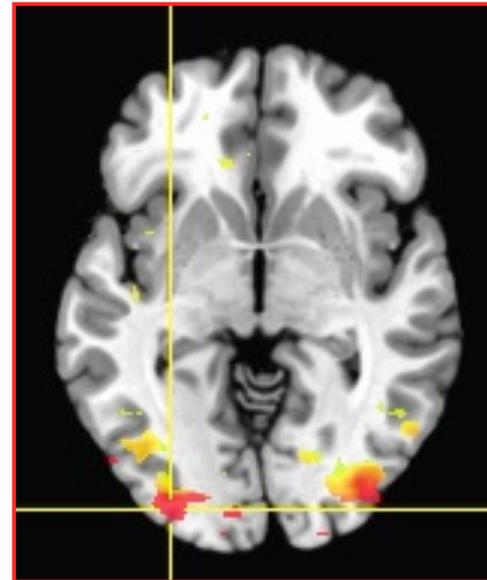
- Group analysis activation maps (**3dANOVA3**) from 16 subjects

**REML**



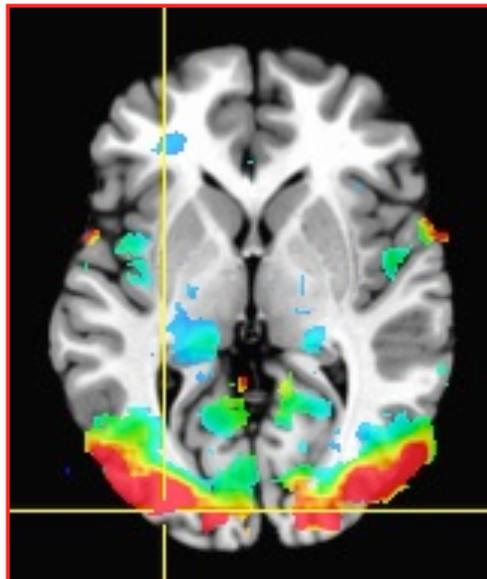
**F-test for  
Affect  
condition**

**OLSQ**

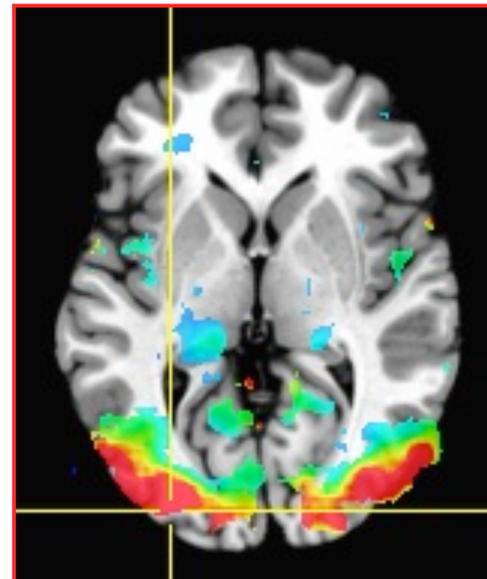


**F-test for  
Affect  
condition**

**F-test for  
Category  
condition**



**F-test for  
Category  
condition**

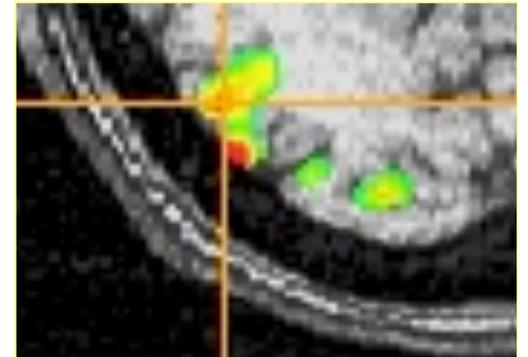


# Nonlinear Regression

- Linear models aren't the only possibility
  - ★ e.g., could try to fit HRF of the form  $h(t) = a \cdot t^b \cdot e^{-t/c}$
  - ★ Unknowns  $b$  and  $c$  appear nonlinearly in this formula
- Program **3dNLFit** can do nonlinear regression (including nonlinear deconvolution)
  - ★ User must provide a C function that computes the model time series, given a set of parameters (e.g.,  $a$ ,  $b$ ,  $c$ )
    - We could help you develop this C model function
    - Several sample model functions in the AFNI source code distribution
  - ★ Program then drives this C function repeatedly, searching for the set of parameters that best fit each voxel
  - ★ Has been used to fit pharmacological wash-in/wash-out models (difference of two exponentials) to fMRI data acquired during pharmacological challenges
    - e.g., injection of nicotine, cocaine, ethanol, etc.
    - these are difficult experiments to do **and** to analyze

## 3dBlurToFWHM

- New program to smooth FMRI time series datasets to a specified smoothness (as estimated by FWHM of noise spatial correlation function)
  - ★ Don't just add smoothness (à la **3dmerge**) but control it (locally and globally)
  - ★ Goal: use datasets from diverse scanners
- Why blur FMRI time series?
  - ★ Averaging neighbors will reduce noise
  - ★ Activations are (usually) blob-ish (several voxels across) 
  - ★ Diminishes the multiple comparisons problem
- **3dBlurToFWHM** blurs only inside a mask
  - ★ To avoid mixing air (noise-only) and brain voxels
  - ★ Partial Differential Equation (PDE) based blurring method
    - 2D (intra-slice) or 3D blurring



# Remaining New-ish AFNI Features: a quick once-over

**Princeton, Fall 09**



# Outline

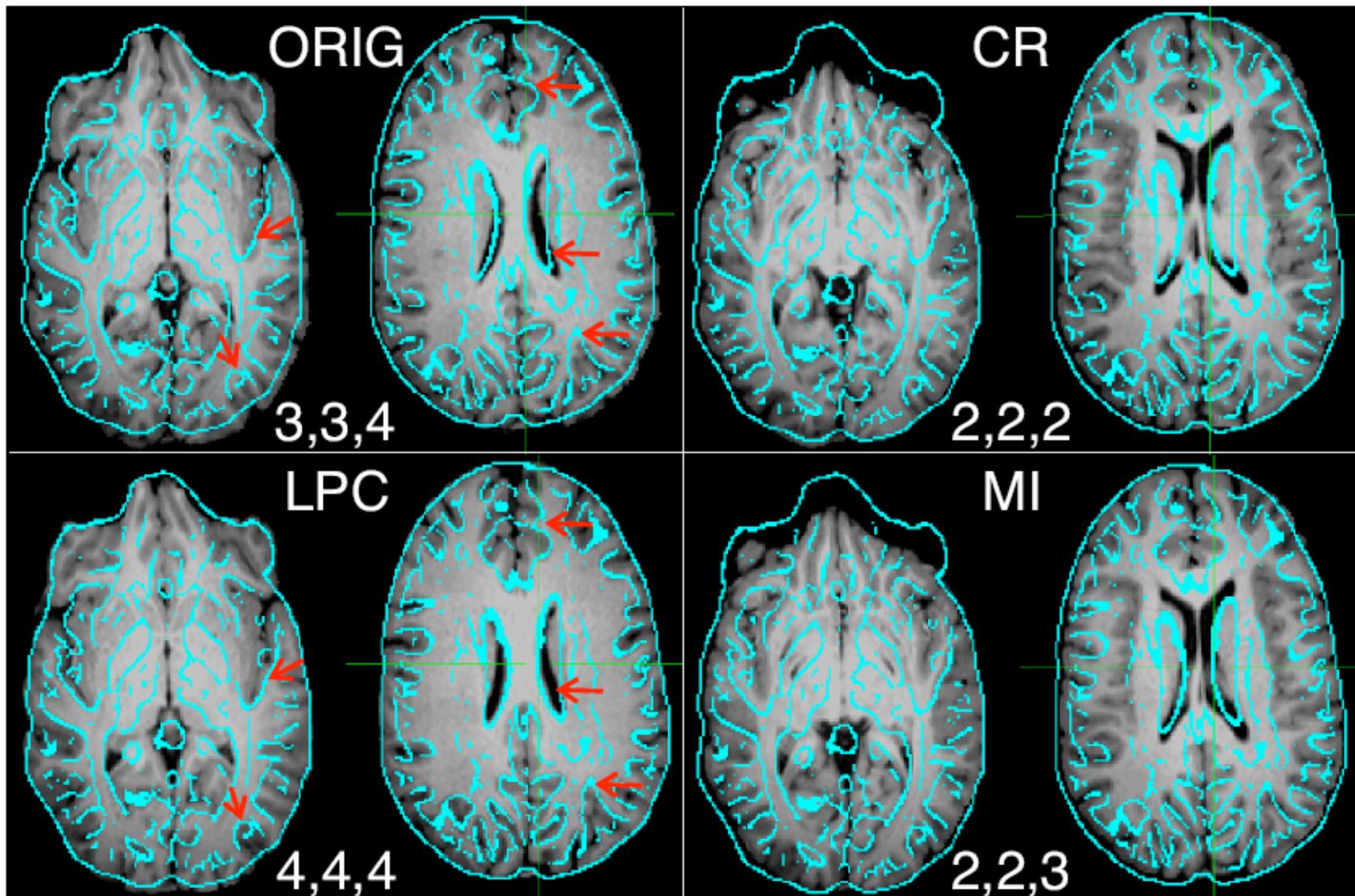
- 3dREMLfit = analysis allowing for serial correlation
- 3dLME = generalized ANOVA
- 1dGC = Granger Causality analysis
- align\_epi\_anat.py = align EPI and structural ( $T_1$ ) datasets
- Miscellany
  - ★ Manganese MRI = tracing anatomical connectivity
  - ★ DCEMRI = Dynamic Contrast Enhanced MRI
  - ★ Realtime AFNI = feedback to the subject
  - ★ ICA = Spatial Indep. Component Analysis
  - ★ ICC = Intra Class Correlation
  - ★ DTI = new plugin from UCSD
  - ★ ExamineXmat.R = analyze X matrix for potential problems
  - ★ RetroTS.m = create regressors for removing heart, respiration, and respiration-volume-per-time artifacts

# **align\_epi\_anat**

**Aligning EPI and T1-weighted structural volumes**

## Alignment of EPI and Anatomical Datasets

- New LPC method gives consistently better alignment—based on visual inspection—over other cost functionals, including MI and CR



## `align_epi_anat.py`

- aligns EPI and structural datasets using LPC method in `3dAllineate`
- `align_epi_anat.py` script prepares data, then does the work:
  - deobliquing
  - skull stripping
  - slice timing correction
  - motion correction
  - weighting, resampling
  - Talaraich transformation
- Applies concatenated matrices (oblique, volume registration, tlrc)
- Aligns EPI→Anat or Anat→EPI

### Basic Example:

```
align anatomical dataset to epi dataset at sub-brick 5
align_epi_anat.py -anat anat+orig -epi epi+orig \
 -epi_base 5
```

## align\_epi\_anat.py

### More advanced example:

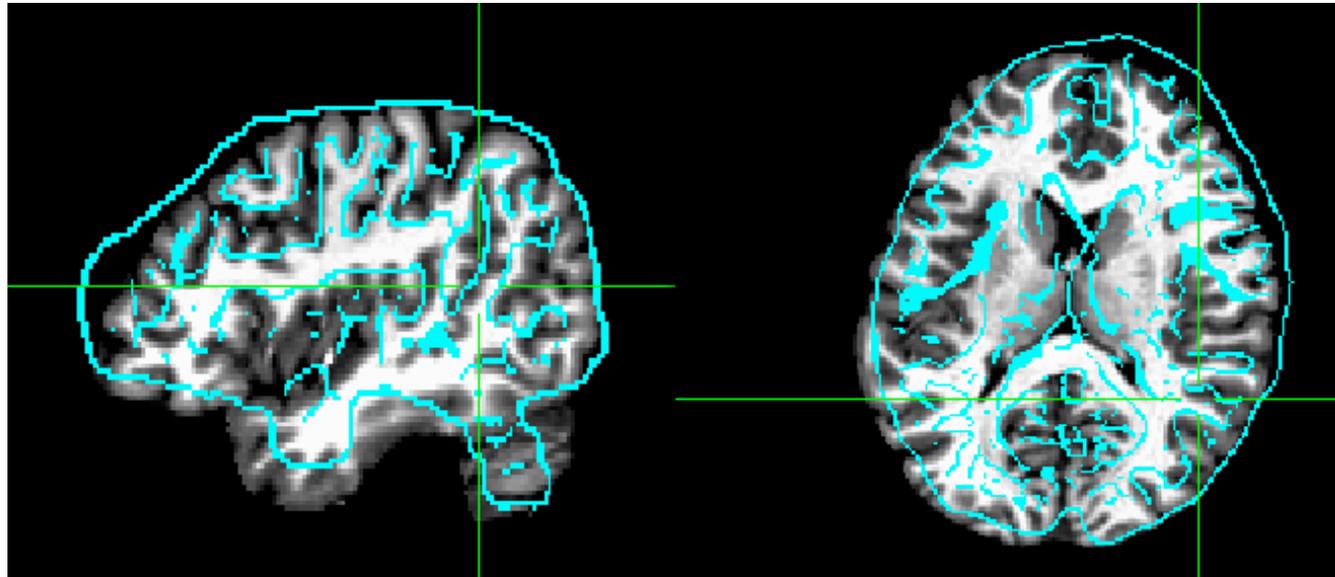
- # Transform EPI dataset to match Anat
- # Register “child EPI” datasets to “parent” EPI and align with Anat
- # Warp EPI and child EPI datasets to +tlrc space based on existing
- # Anatomical +tlrc datase
- # Also, create composite edge images

```
@auto_tlrc -base ~/abin/TT_N27+tlrc \
 -input sb23_mpra+orig
align_epi_anat.py -anat sb23_mpra+orig \
 -epi epi_r03+orig \
 -epi_base 6 -child_epi epi_r??+orig.HEAD \
 -epi2anat -suffix _al2anat \
 -tlrc_apar sb23_mpra_at+tlrc -AddEdge
```

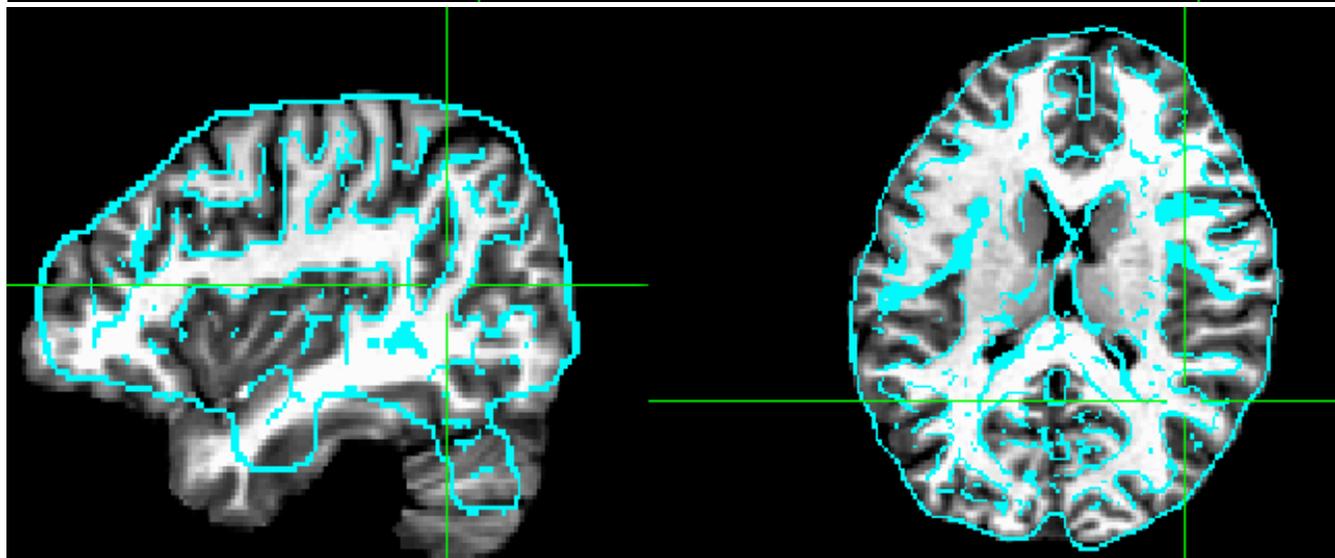
Flexibility in options for cost functionals and processing steps allow alternate uses. Already used also for  $T_1$ -to- $T_1$  (SPGR, FLAIR, 3T, 7T), EPI-to-EPI, rat and monkey data, and partial coverage data.

# align\_epi\_anat.py example output

Pre-alignment



Post-alignment



@AddEdge -single-edge display shows before and after with edges from transformed EPI dataset as overlay

**3dICA.R Spatial  
Independent Component  
Analysis Using  
fastICA**

## **3dICC.R Intra-Class Correlation**

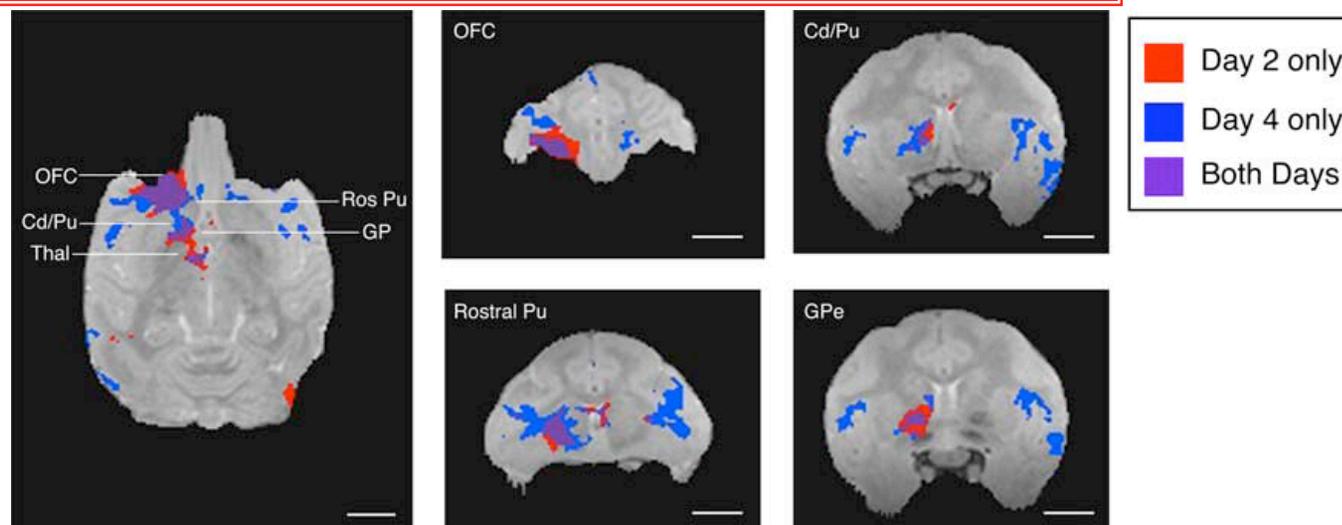
- For estimating the variance contribution from each random factor.
  - ★ For 2- or 3-way ANOVA designs with all factors random
  - ★ Outputs the voxelwise fraction of the variance from each of the factors

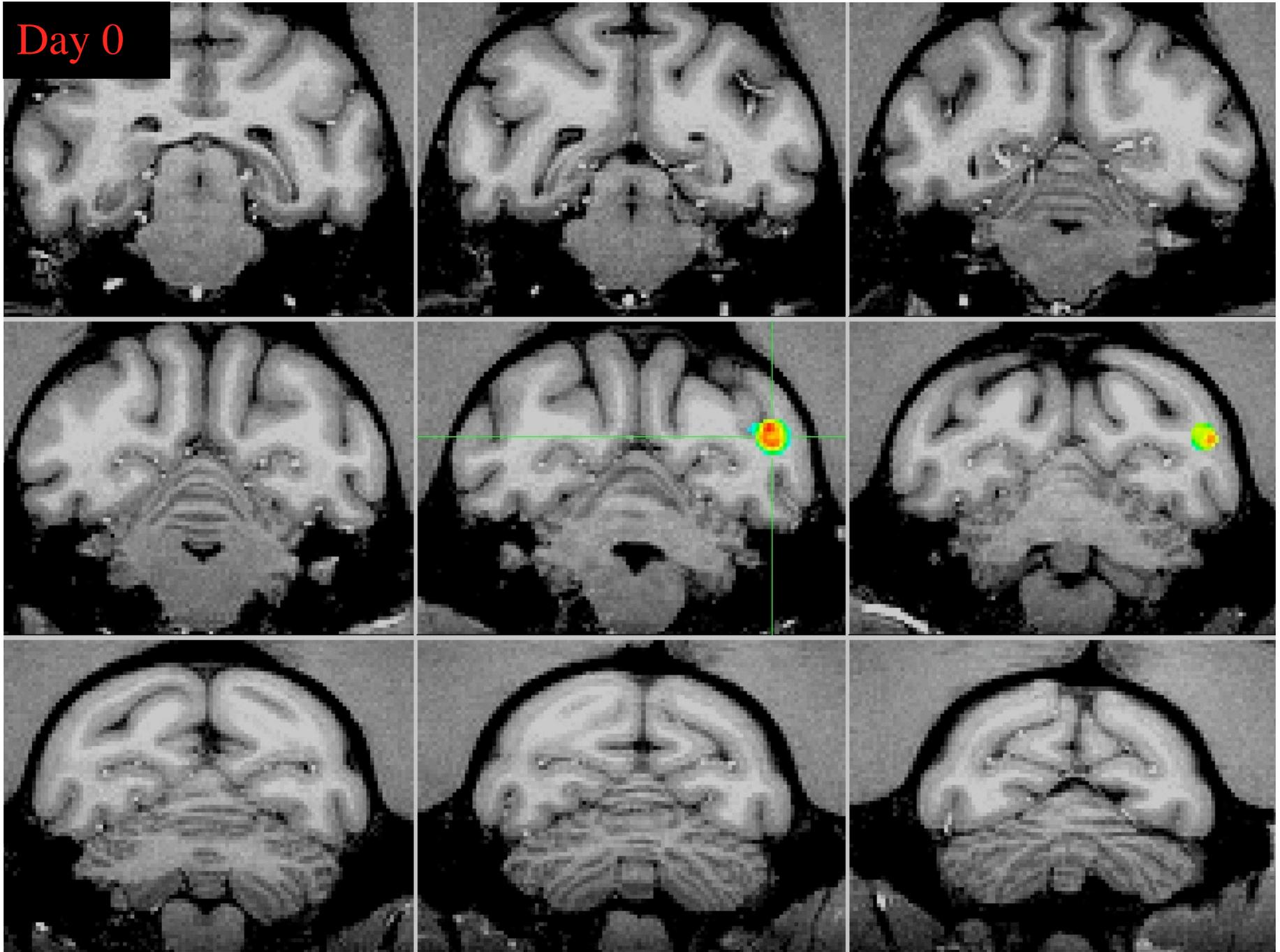
# **Manganese Enhanced MRI**

## Manganese Enhanced MRI

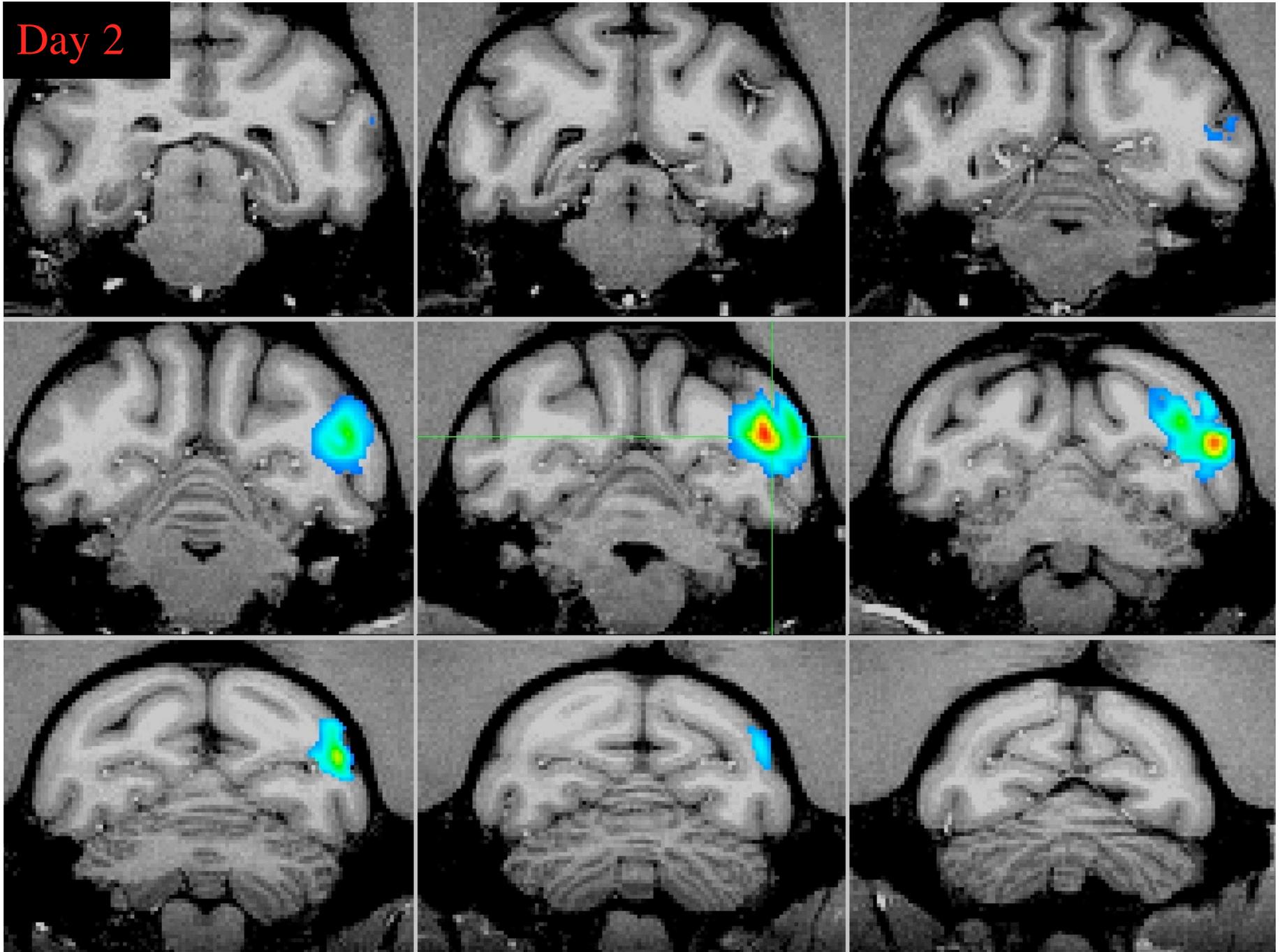
- We have a pipeline for *voxelwise* detection of Manganese induced signal enhancement
  - ★ Robust skull removal and intra-subject longitudinal alignment
  - ★ Parametric and non-parametric signal detection approaches with multiple comparison correction
  - ★ Output of summary results from each stage for quality checking
  - ★ Morals from our experiences thus far:
    - Get as many scans as possible (10+) in pre-injection phase
    - Get several post-injection scans at each time point of interest (2+)
    - Examine your images immediately for bad artifacts and correct!!!

First generation results: Fig. 7, Simmons et al. J. Neuroscience 08

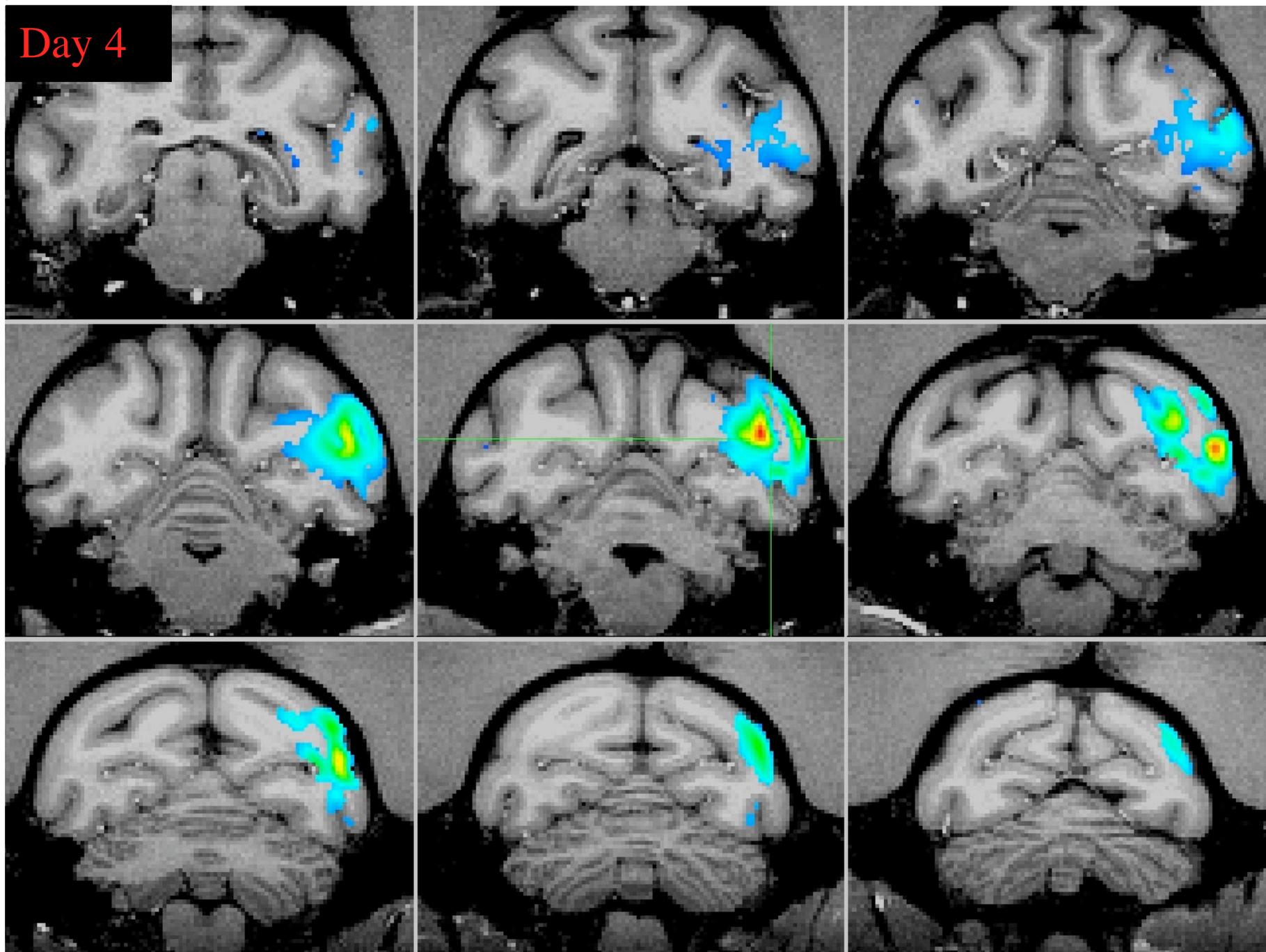




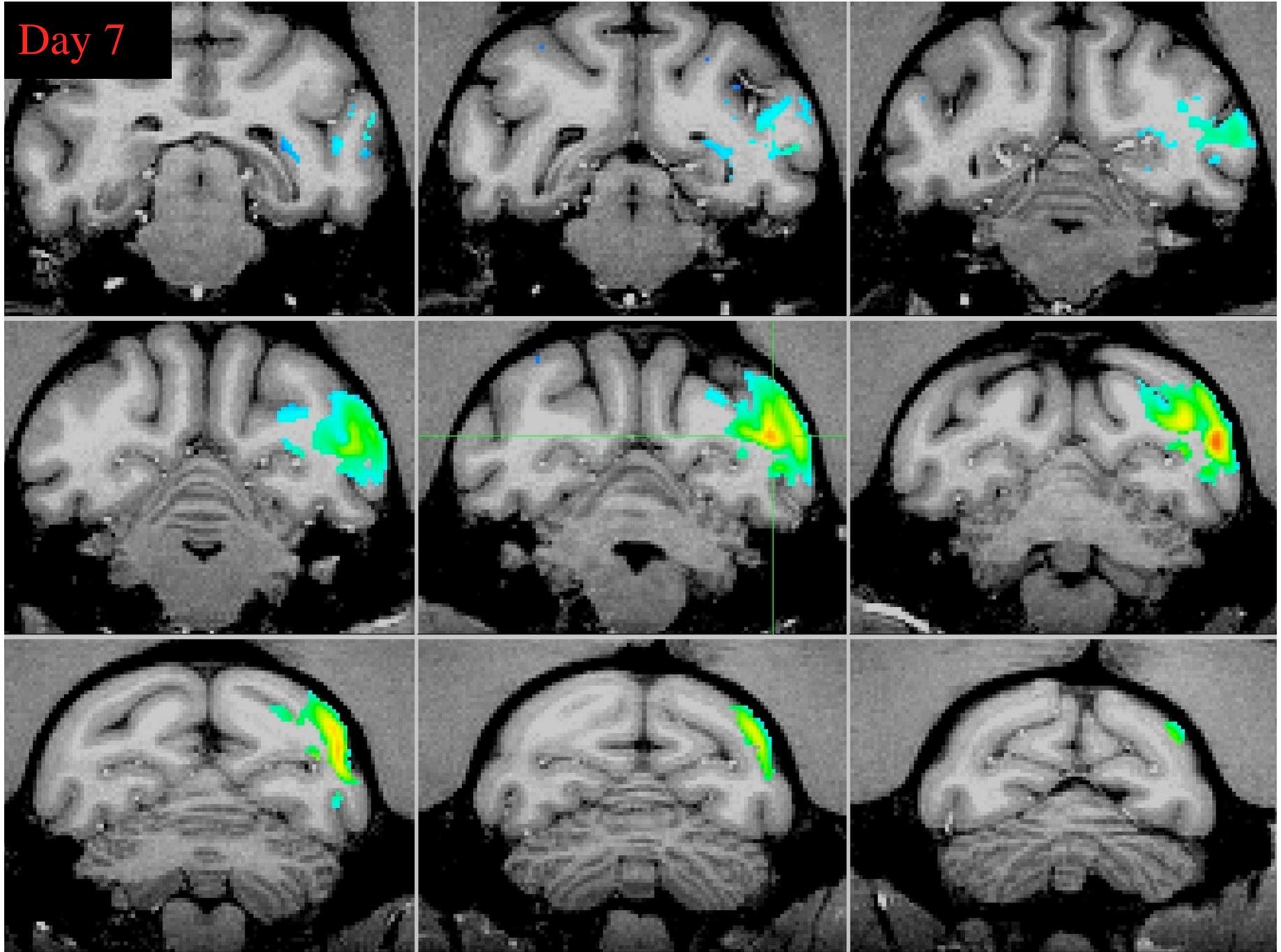
Day 2



Day 4



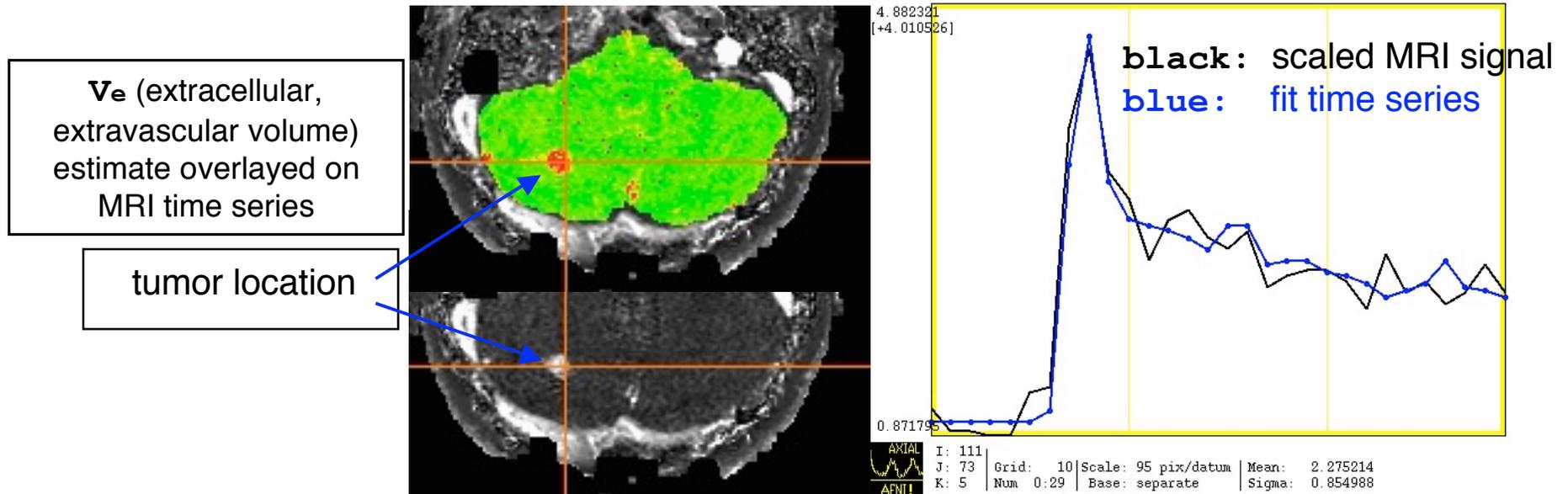
Day 7



# **DCEMRI**

**Dynamic Contrast Enhanced MRI:  
Analysis with 3dNLFim**

## DEMRI: Dynamic (contrast) Enhanced MRI



- Collaboration with John Butman, Hemant Sarin in Clinical Center, on Dynamic Contrast Enhanced MRI (DCEMRI or DEMRI)
- Gd-DTPA injection – large, relatively inert molecule that doesn't pass intact blood-brain barrier injected after short baseline, but brightens T1-weighted images
- Non-linear model in `3dNLFit` framework to compute kinetic parameters ( $K_{trans}$ ,  $k_{ep}$ ,  $V_e$ ,  $fpv$ ) of brain tissue in a two compartment model to model breakdown of blood-brain barrier
- This implementation in AFNI is the only freely available DEMRI software for volumetric analysis (at this time)

**Realtime AFNI at  
NIH Scanners**

## Realtime fMRI-Feedback at NIH Scanners

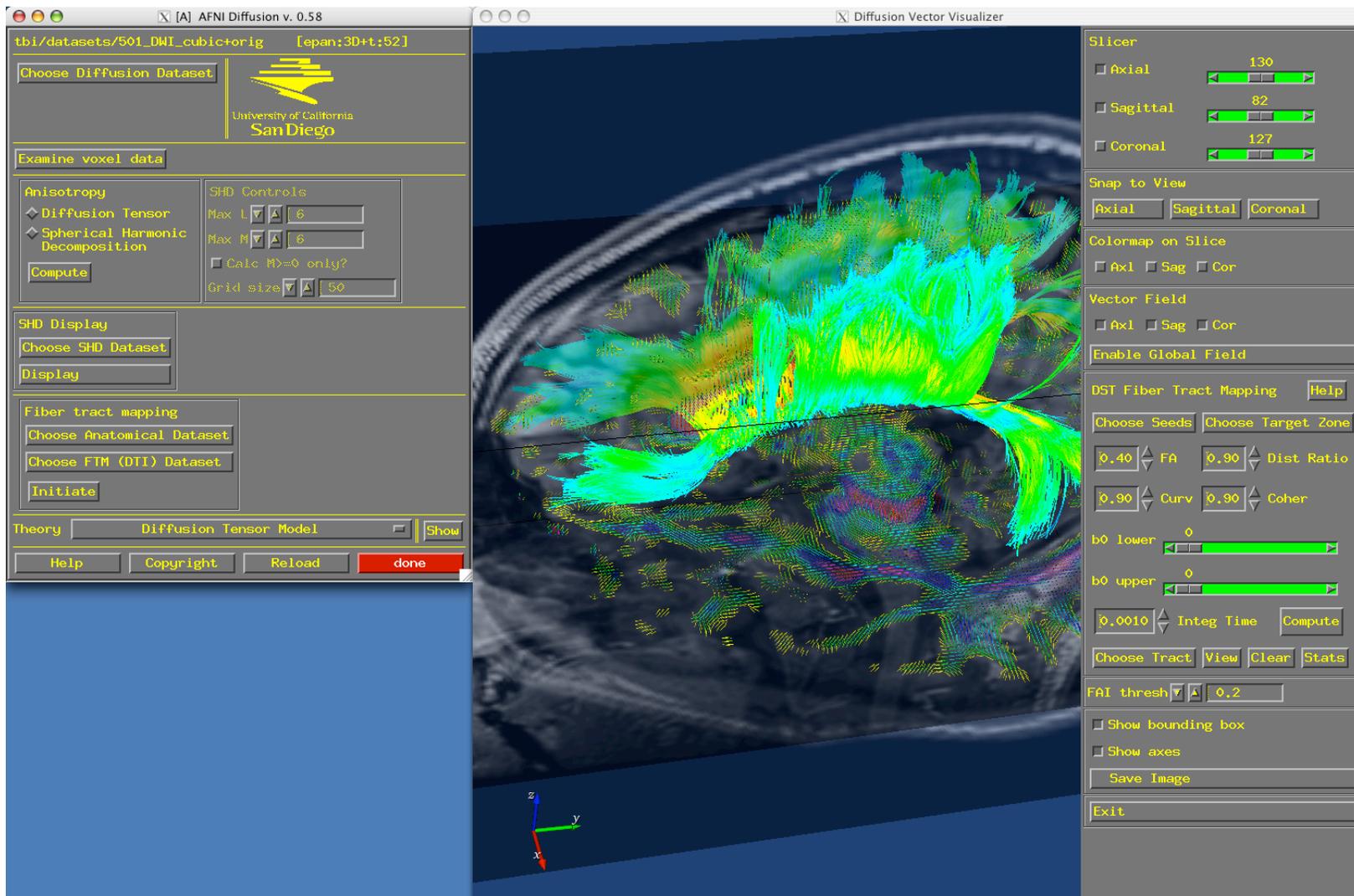
- Enhanced version of the NIH realtime MRI scanner software
  - ★ installed on all GE MRI scanners (written by Jerzy Bodurka)
  - ★ can be used with AFNI to conduct realtime fMRI with feedback to the subject
  - ★ a sample real-time plotting tool is installed on all FMRIF MRI scanners
    - based on **serial\_helper**, with updates written by Javier and Jerzy
    - uses **Grace**: a 2D plotting tool for the X Window System
- MRI data is captured each TR and used to drive the realtime subject feedback display
  - ★ motion parameters: to show the subject when they move “too much”
  - ★ ROI averages: to show real-time “activation” at one or more ROIs
  - ★ raw (registered) voxel data: for other nefarious purposes
- AFNI’s realtime updates:
  - ★ **Dimon** → **afni** is more responsive, to improve subject feedback
  - ★ has enhanced stability and environmental controls
  - ★ **afni** can send ROI averages or raw voxel data to **serial\_helper**, each TR

# **Diffusion Tensor Imaging**

**New Plugin from UCSD**

# Diffusion Plug-in

- From UCSD group led by Larry Frank with Greg Balls, Ning Kang
- seed-based “diffusion model” tractography allows for fiber crossing
- Pretty 3D primary eigenvector and FA-encoded tractography display

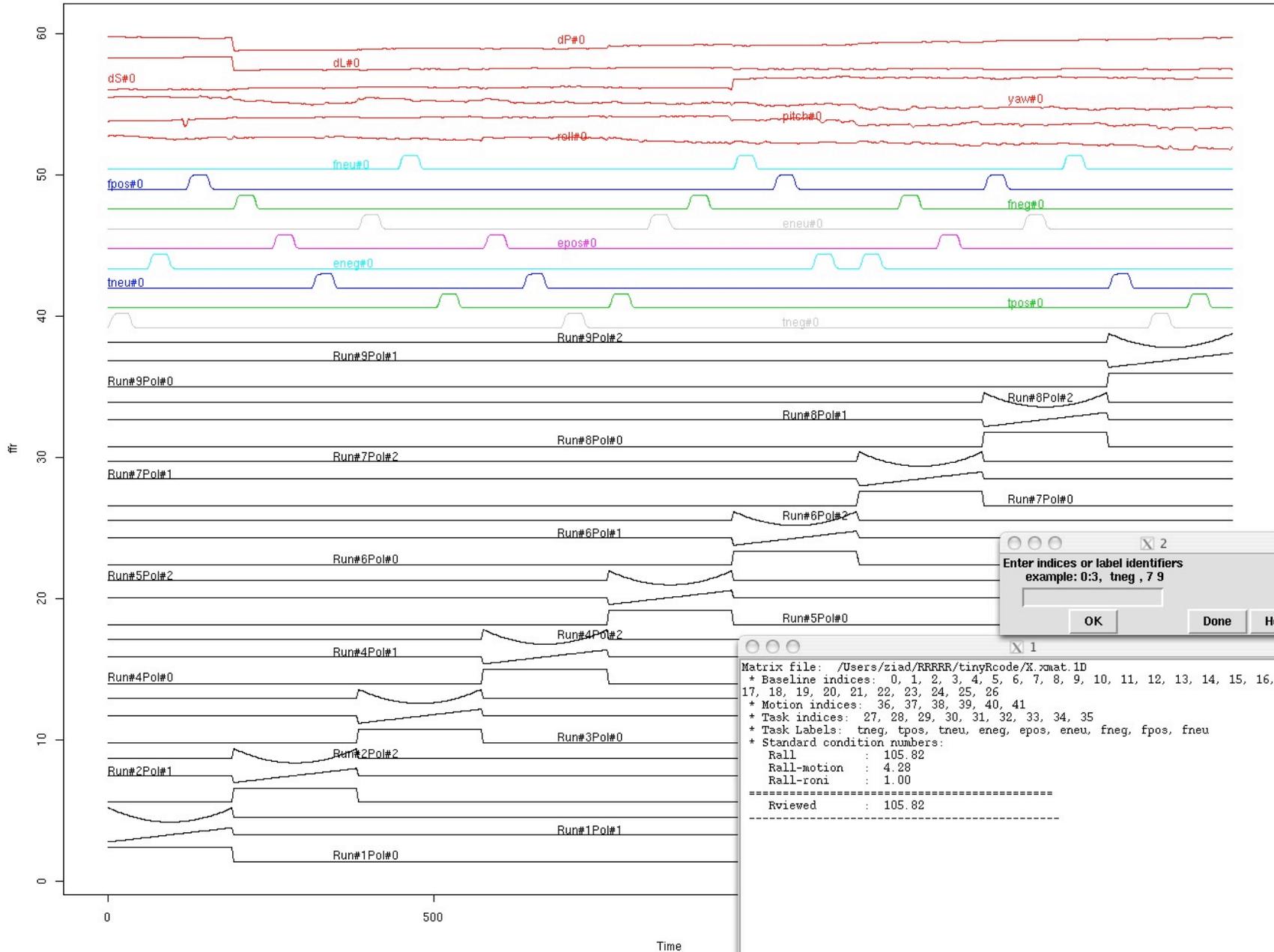


# ExamineXmat

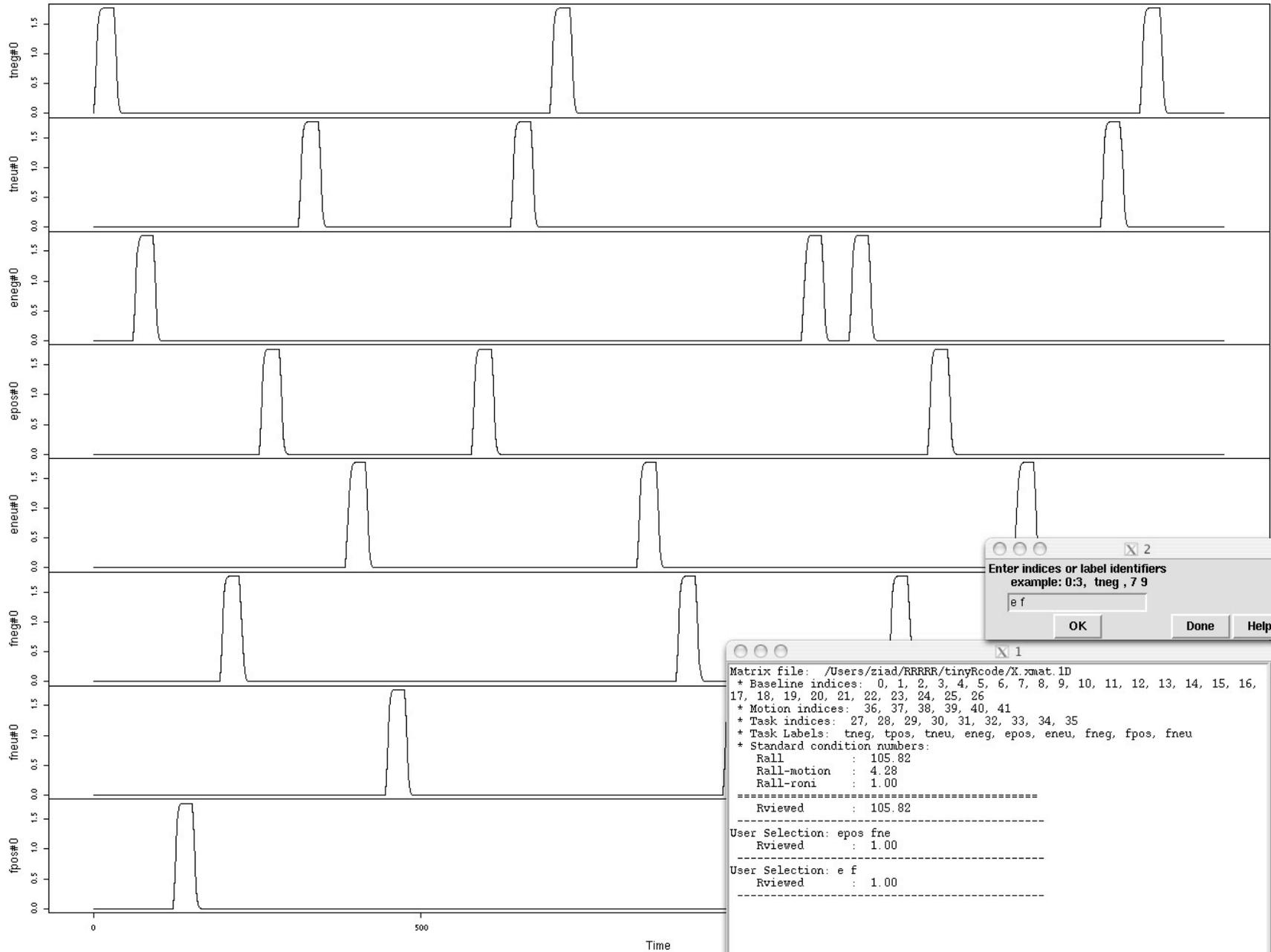
**Putting your time series regression matrix up on the rack and checking it for problems**

- A tool to examine design matrices
  - Visualize matrix and selected subsets of it
  - Condition numbers for various subsets of matrix and selected regressors
- More experiment design tools:
  - [xmat\\_tool.py](#), [make\\_random\\_timing.py](#), and [timing\\_tool.py](#)

# ExamineXmat.R



# ExamineXmat.R



Enter indices or label identifiers  
example: 0:3, tneg , 7 9  
  
OK Done Help

Matrix file: /Users/ziad/RRRRR/tinyRcode/X.mat.ID  
\* Baseline indices: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26  
\* Motion indices: 36, 37, 38, 39, 40, 41  
\* Task indices: 27, 28, 29, 30, 31, 32, 33, 34, 35  
\* Task Labels: tneg, tpos, tneu, eneg, epos, eneu, fneg, fpos, fneu  
\* Standard condition numbers:  
Rall : 105.82  
Rall-motion : 4.28  
Rall-roni : 1.00  
-----  
Rviewed : 105.82  
-----  
User Selection: epos fne  
Rviewed : 1.00  
-----  
User Selection: e f  
Rviewed : 1.00  
-----

# Retrots.m

Removing physiological noise with  
**RETROICOR** (Glover et al. 2000) and **RVT** (Birn et  
al. 2006)

- A matlab tool to create slicewise regressors for reducing physiological noise caused by respiration and heart rate.
- Regressors (~13 for each slice) are added to **3dREMLfit**'s design matrix with -slibase option.

**"That's  
all  
folks!"**

