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
    - o 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 **WARNING**s and **ERROR**s
  - ★ 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?)
    - o You want to analyze correct and incorrect trials as separate cases
    - o What if some subject makes no mistakes? Hmmm…
      - ➥ 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

---

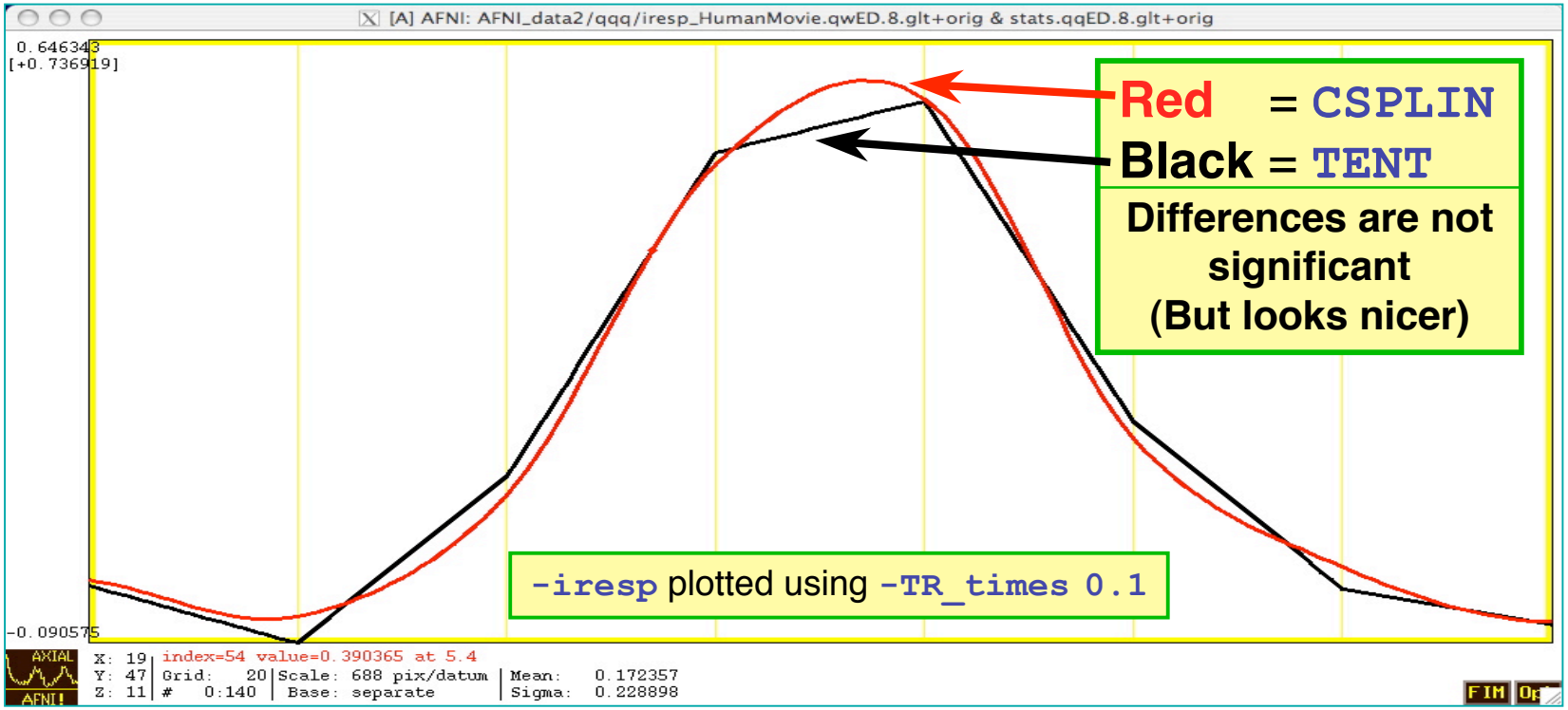
- `3dDeconvolve` recommends a `-polort` value, and prints that out as well as the value you chose (or defaulted to)
  - ★ `-polort A` can be used to let the program set the detrending (AKA "high pass filtering", since detrending removes low frequency content from data) level automatically

---

- `-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
    - o Same parameters: **`(start,stop,number of regressors)`**
    - o Can be used as a “drop in” replacement for **`TENT`**

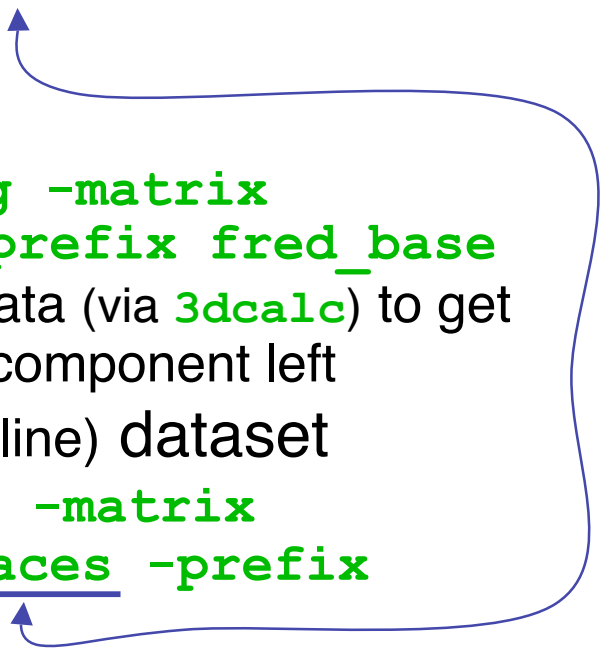


Other Features - 6

- **-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**
    - o **-cbucket** stores β coefficients for each $\mathbf{X}$ matrix column into dataset
    - o **-x1D** stores the matrix columns (and **-stim_label**s, etc.)
  - ★ Potential uses:
    - o 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
    - o 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

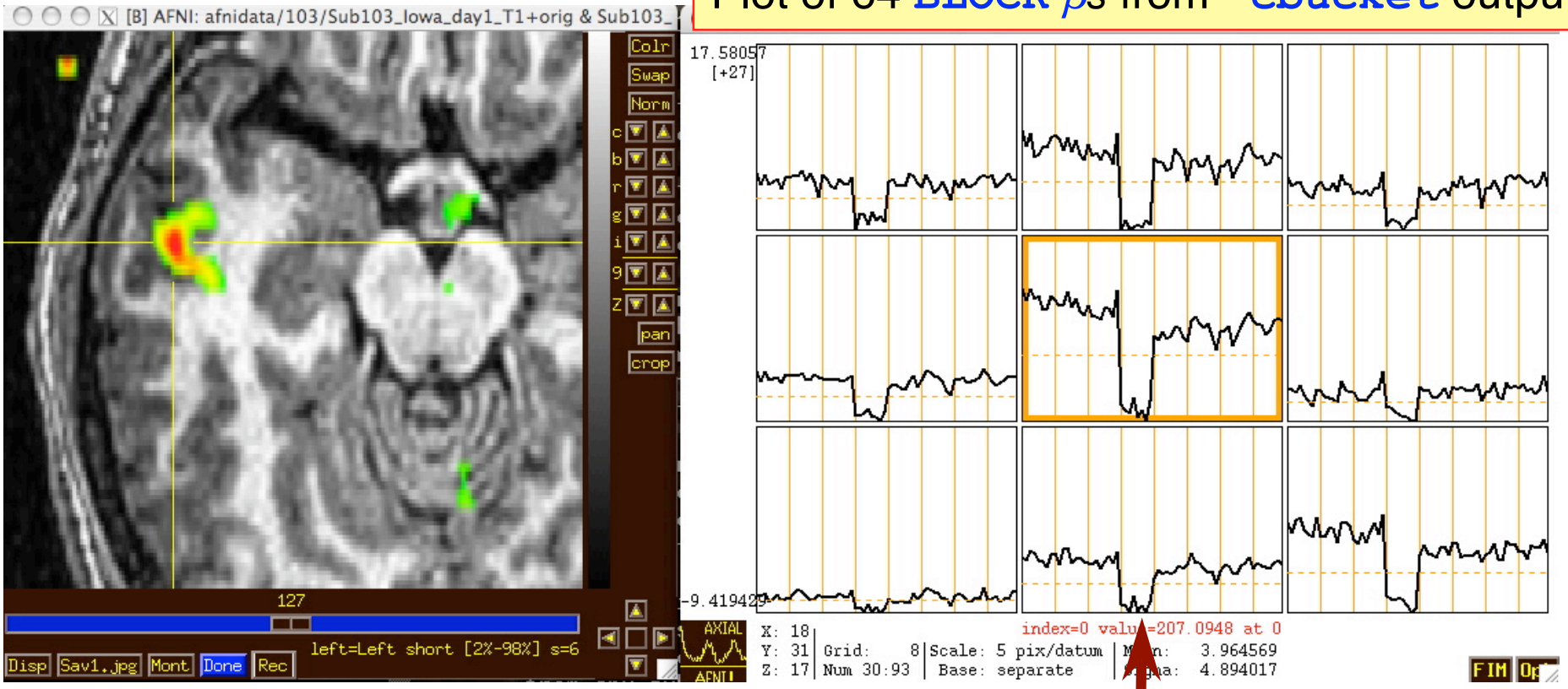
IM Regression - 1

- **IM** = **I**ndividual **M**odulation
  - ★ Compute *separate* amplitude of response for each stimulus
    - o Instead of computing average amplitude of responses to multiple stimuli in the same class
  - ★ Response amplitudes (βs) for each individual block/event will be highly noisy
    - o Can't use individual activation map for much
    - o Must pool the computed βs in some further statistical analysis (*t*-test via `3dttest`? inter-voxel correlations in the βs? correlate βs with something else?)
  - ★ Usage: **`-stim_times_IM k tname model`**
    - o 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** βs from **-cbucket** output



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

IM Regression - 3

- IM works naturally with blocks, which only have 1 amplitude parameter per stimulus
- With event-related experiment and deconvolution, 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 × 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 Modulation)
  - ★ 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

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

  - ★ Where a_k = value of k^{th} ABI value, and $\bar{a}$ is the average ABI value
- Response (β) for first regressor is standard activation map
- Statistics and β 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
    - o 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* β weights are zero: that there is no ABI-independent *or* ABI-proportional signal change)
  - ★ Use **-tout** option to test each β weight separately
  - ★ Can **1dplot** $\mathbf{X}$ 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:

$$r_{\mathrm{AM1}}(t) = \sum_{k=1}^{K} h(t-\tau_k)\cdot a_k$$

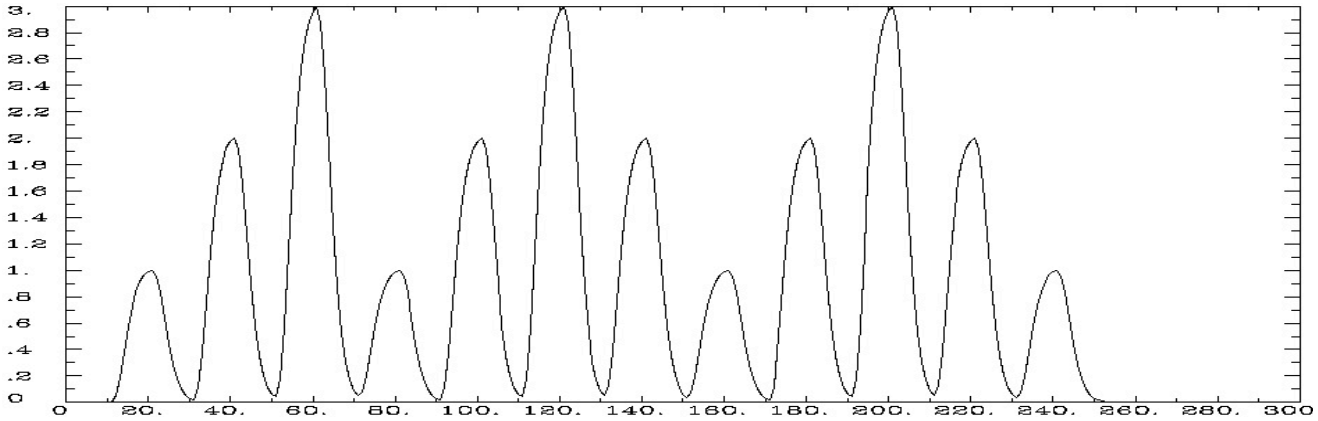
  - ★ Can't imagine what value this option has, but you never know … (if you can think of a good use, let me know)
- Future directions:
  - ★ Allow more than one amplitude to be married to each stimulus time (insert obligatory polygamy/polyandry joke here)
    - o How many ABI types at once is too many? I don't know.
  - ★ How to deal with unknown nonlinearities in the BOLD response to ABI values? I don't know. (Regress each event separately, then compute MI?)
  - ★ Deconvolution with amplitude modulation? Requires more thought.

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

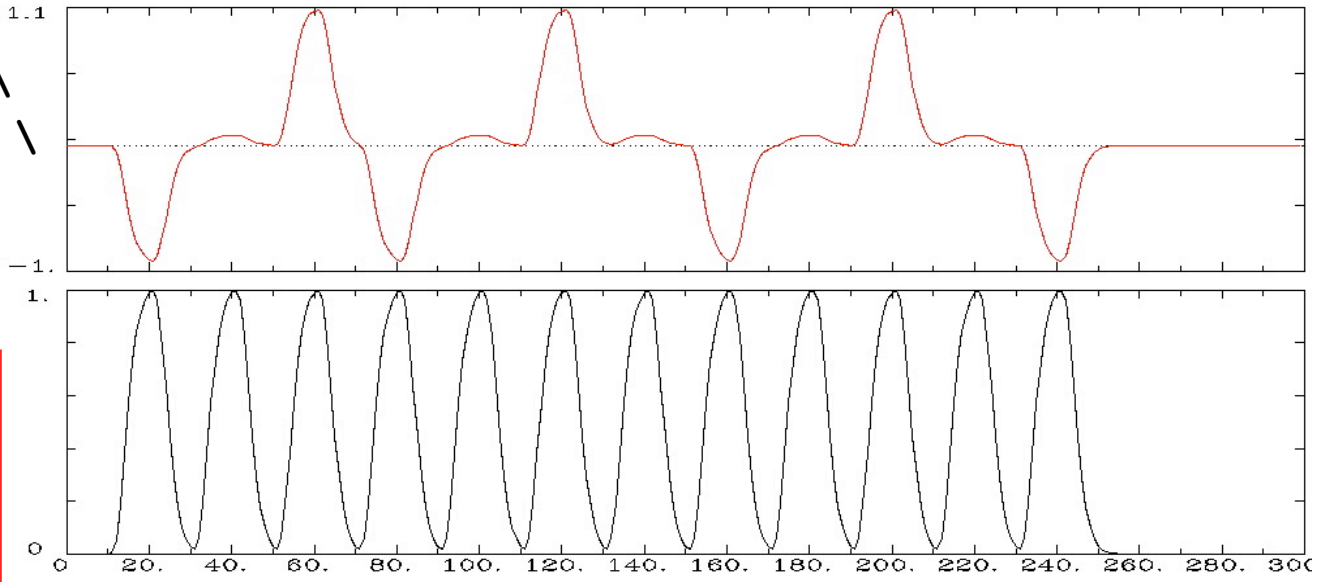
- **3dDeconvolve -nodata 300 1.0 -num_stimts 1 **
  -stim_times_AM1 1 AM.1D 'BLOCK(10,1)' -x1D AM1.x1D
- **1dplot AM1.x1D'[2]'**

AM1 model of signal (modulation = ABI)



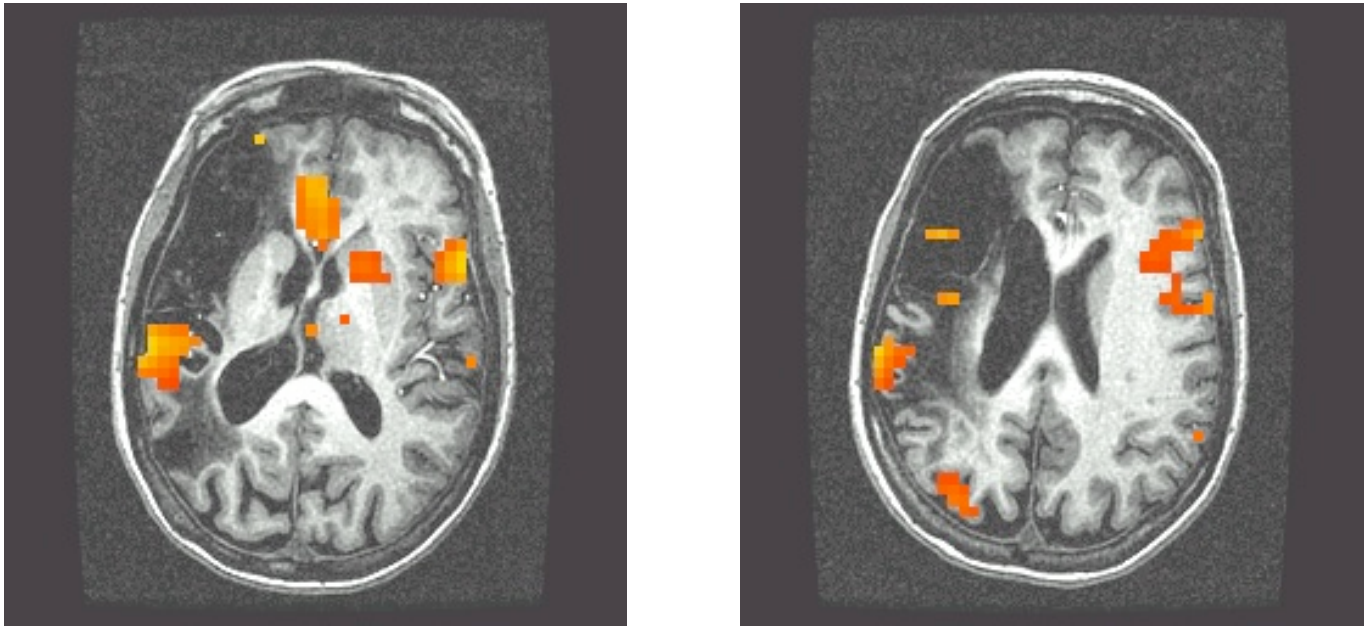
- **3dDeconvolve -nodata 300 1.0 **
  **-num_stimts 1 **
  **-stim_times_AM2 1 **
  **AM.1D 'BLOCK(10,1)' **
  -x1D AM2.x1D
- **1dplot -sepscl **
  AM2.x1D'[2,3]'

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=Al 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 (β_{AM2})
  - What does this mean? Don't ask me!

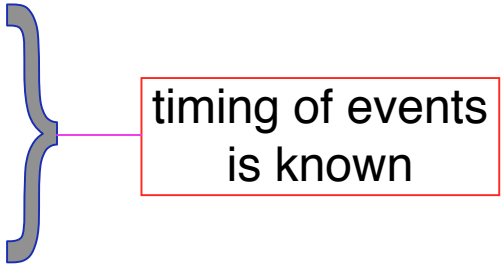


AM Regression - 7

- Alternative: use **IM** to get individual βs for each block/event and then do external regression statistics on those values
- Could do nonlinear fitting via `3dNLfim`, or inter-class contrasts via `3dttest`, `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



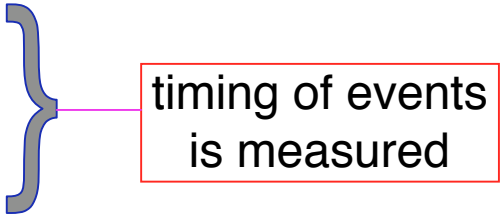
★ Cannot treat this as one event with one HRF, since the different waiting times will result in different overlaps in separate responses from cue #1 and cue #2

★ 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



- 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 β 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't generate (b) HRF in `3dDeconvolve`

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)
    - o In principle, could measure these sources of noise separately and then try to regress them out
      - ➥ RETROICOR program underway (Rasmus Birn of FIM/NIMH)
  - ★ Scanner fluctuations (e.g., thermal drift of hardware)
  - ★ Small subject head movements (10-100 mm)
  - ★ 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:
    - o N = number of time points
    - o m = number of fit parameters
    - o $N-m$ = 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 β fit parameters using more efficient "generalized least squares", using this correlation, all at once (REML method)

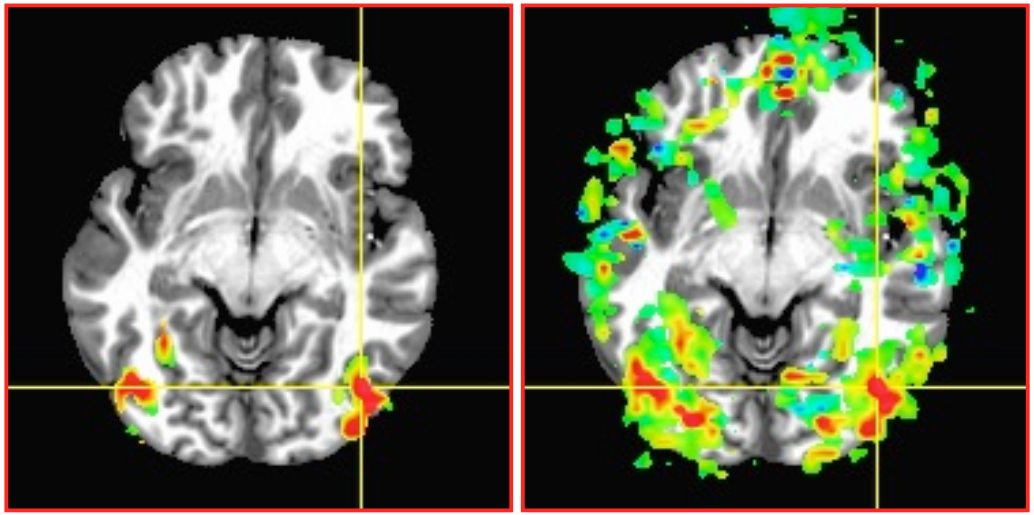
New Program: 3dREMLfit

- Implements Solution #2
  - ★ REML is a method for simultaneously estimating variance + correlation parameters ***and*** estimating regression fit parameters (βs)
  - ★ Correlation structure of noise is ARMA(1,1)
    - o 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)
    - o 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)
    - o 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

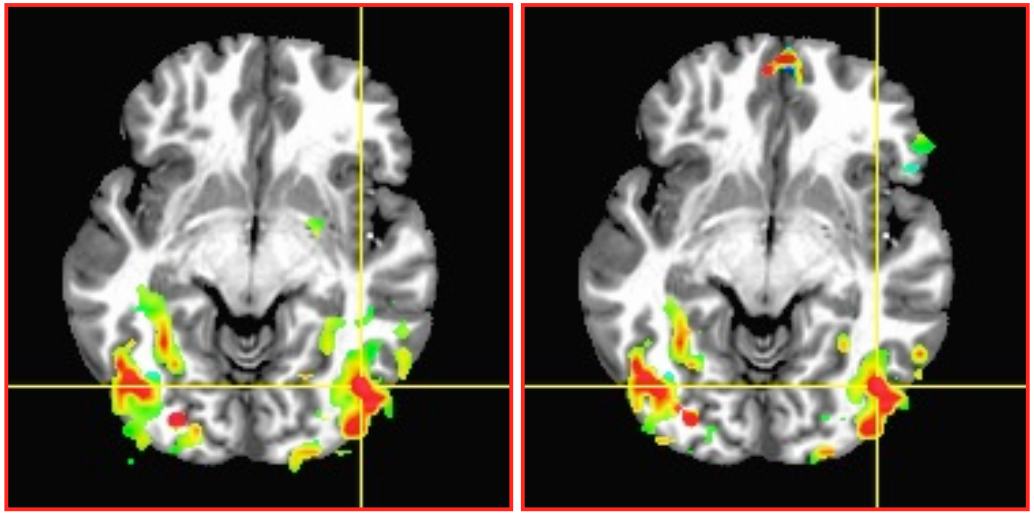
- Compare with **AFNI_data3/afni/rall_regress results**
- **3dREMLfit -matrix rall_xmat.x1D -input rall_vr+orig -fout -tout **
  **-Rvar rall_varR -Rbuck rall_funcR -Rfitts rall_fittsR **
  -Obuck rall_funcO -Ofitts rall_fittsO

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 *!?!*

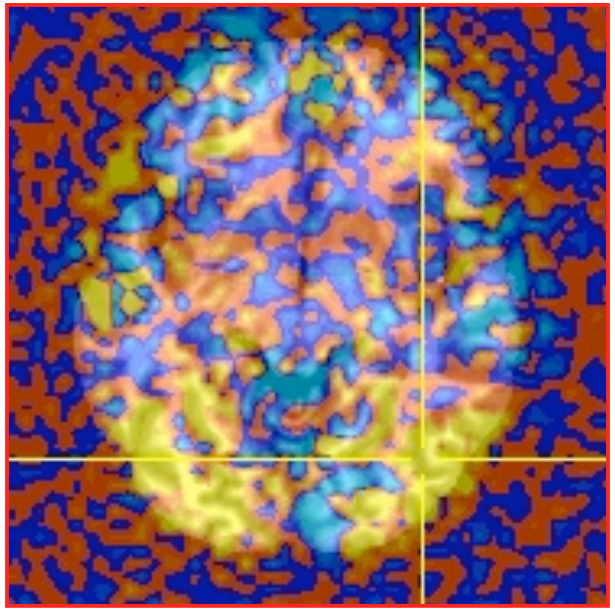
It's Not So Bad: β !

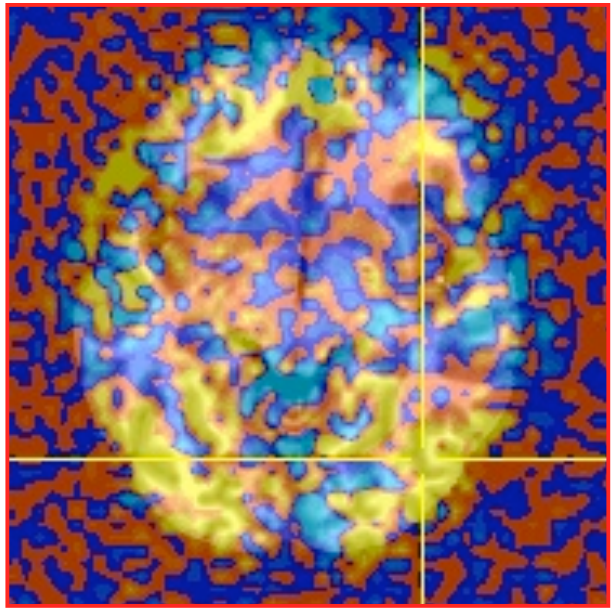
- For individual activation maps, `3dREMLfit`-ized *t*- and *F*-statistics are significantly different, and more accurate
- But … There are at present very few applications for such individual FMRI activation maps
  - ★ pre-surgical planning
- For standard group analysis, inputs are only β fit parameters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

Color Overlay = β weight from analysis on previous slide, no threshold

REML

CPU 500 s





OLSQ

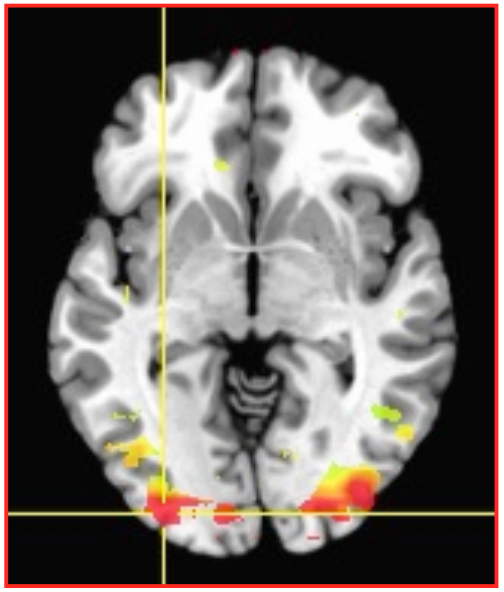
CPU 156 s

It's Not So Bad At All: Group!

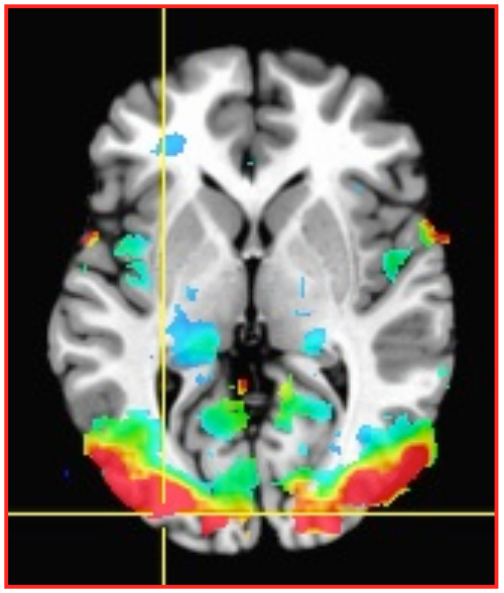
- Group analysis activation maps (`3dANOVA3`) from 16 subjects

REML

***F*-test for Affect condition**

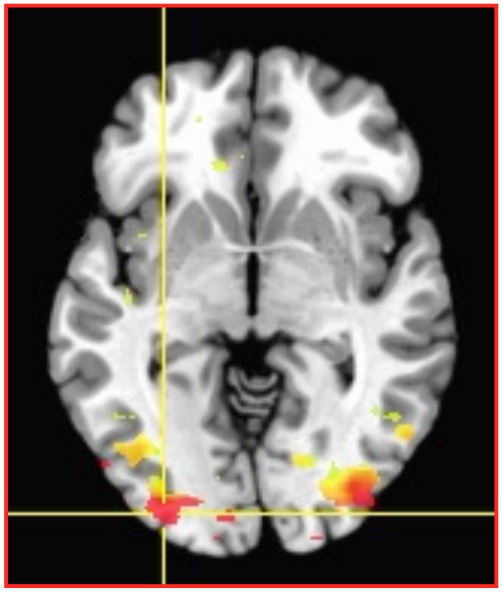


***F*-test for Category condition**

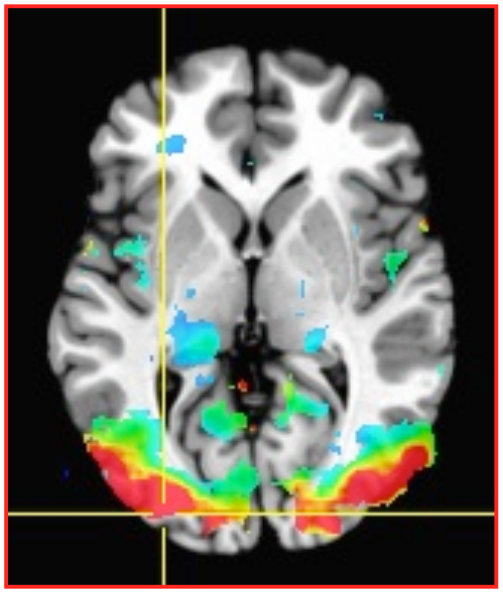


OLSQ

***F*-test for Affect 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 **3dNLfim** 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)
    - o We could help you develop this C model function
    - o 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
    - o e.g., injection of nicotine, cocaine, ethanol, etc.
    - o these are difficult experiments to do ***and*** to analyze

Spatial Models of Activation

- Smooth data in space before analysis

- Average data across anatomically-selected regions of interest ROI (before or after analysis)
  - Labor intensive (*i.e.*, hire more students)

- Reject isolated small clusters of above-threshold voxels after analysis

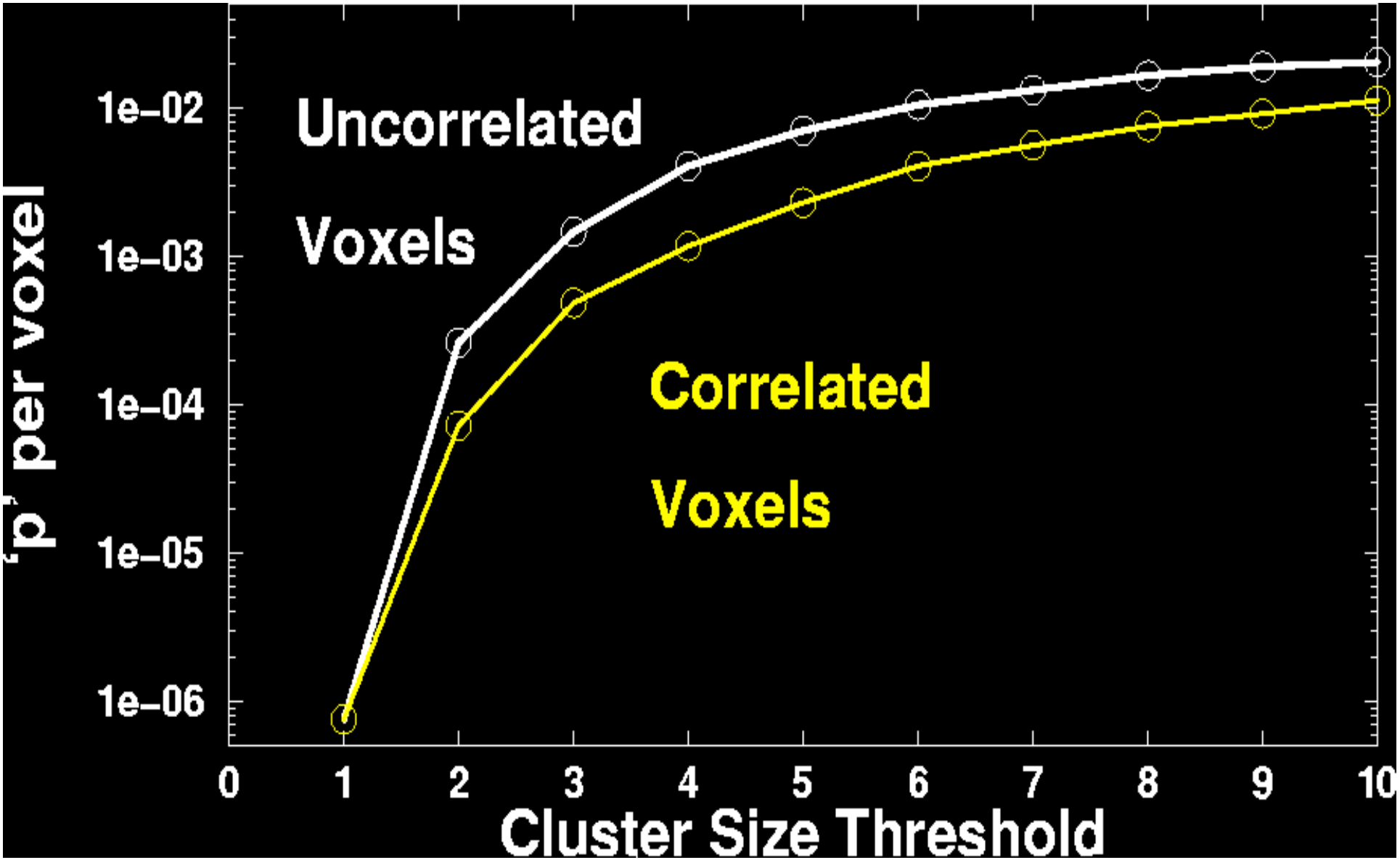
Spatial Smoothing of Data

- Reduces number of comparisons
- Reduces noise (by averaging)
- Reduces spatial resolution
  - Blur it enough: Can make FMRI results look like low resolution PET data
- Smart smoothing: average ***only*** over nearby brain or gray matter voxels
  - Uses resolution of FMRI cleverly
    - New AFNI program: **`3dBlurToFWHM`**
  - Or: average over selected ROIs
  - Or: cortical surface based smoothing

Spatial Clustering

- Analyze data, create statistical map (*e.g.*, ***t*** statistic in each voxel)
- Threshold map at a low ***t*** value, in each voxel separately
  - Will have many false positives
- Threshold map by rejecting clusters of voxels below a given size
- Can control false-positive rate by adjusting ***t*** threshold and cluster-size thresholds together

Cluster-Based Detection



What the World Needs Now

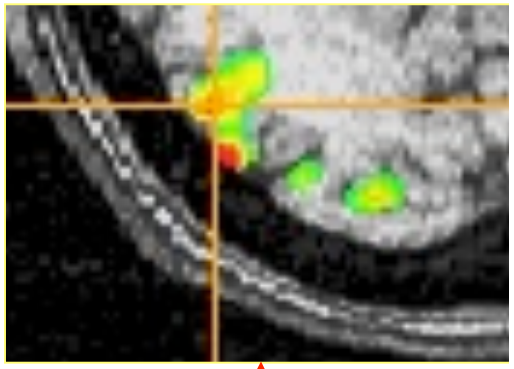
- Unified HRF/Deconvolution⊕Blob analysis
  - Time⊕Space patterns computed all at once, instead of arbitrary spatial smoothing
    - Increase statistical power by bringing data from multiple voxels together cleverly
  - Instead of time analysis followed by spatial analysis (described earlier)
  - Instead of component-style analyses (*e.g.*, ICA) that do not use stimulus timing

---

- Difficulty: models for spatial blobs
  - Little information *à priori* ⇒ must be adaptive

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
    - o 2D (intra-slice) or 3D blurring

In the Pondering Stages

- "Area under curve" addition to `-gltsym` to allow testing of pieces of HRF models from `-stim_times`
- Slice- and/or voxel-dependent regressors
  - ★ For physiological noise cancellation, etc.
  - ★ To save memory? (Could process each slice separately)
    - o One slice-at-a-time regression can be done in a Unix script, using 3dZcutup and 3dZcat programs
- Extend AM regression to allow for more than 1 piece of auxiliary information at each stimulus time
- Interactive tool to examine `-x1D` matrix for problems
  - ★ and `3dDeconvolve` testing of GLT submatrices
- Semi-linear deconvolution program