AFNI

Didactics and Demonstrations

FMRI Task-Based Data Analysis at the Individual Level

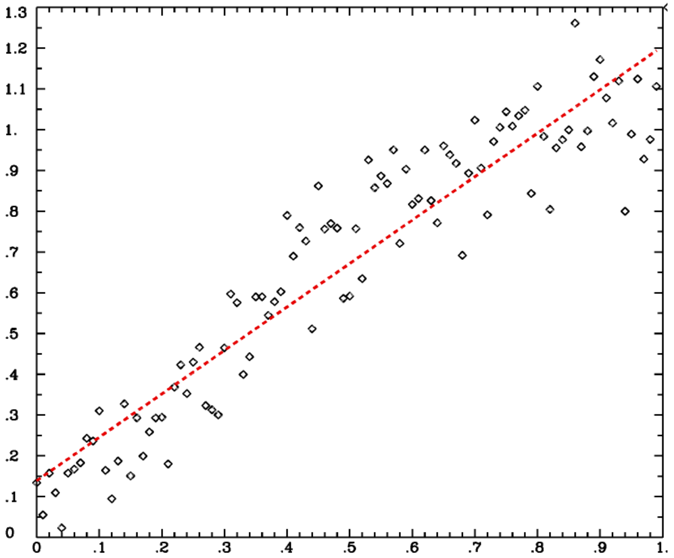
Concepts 1 – Linear Models

Basics of Linear Modeling

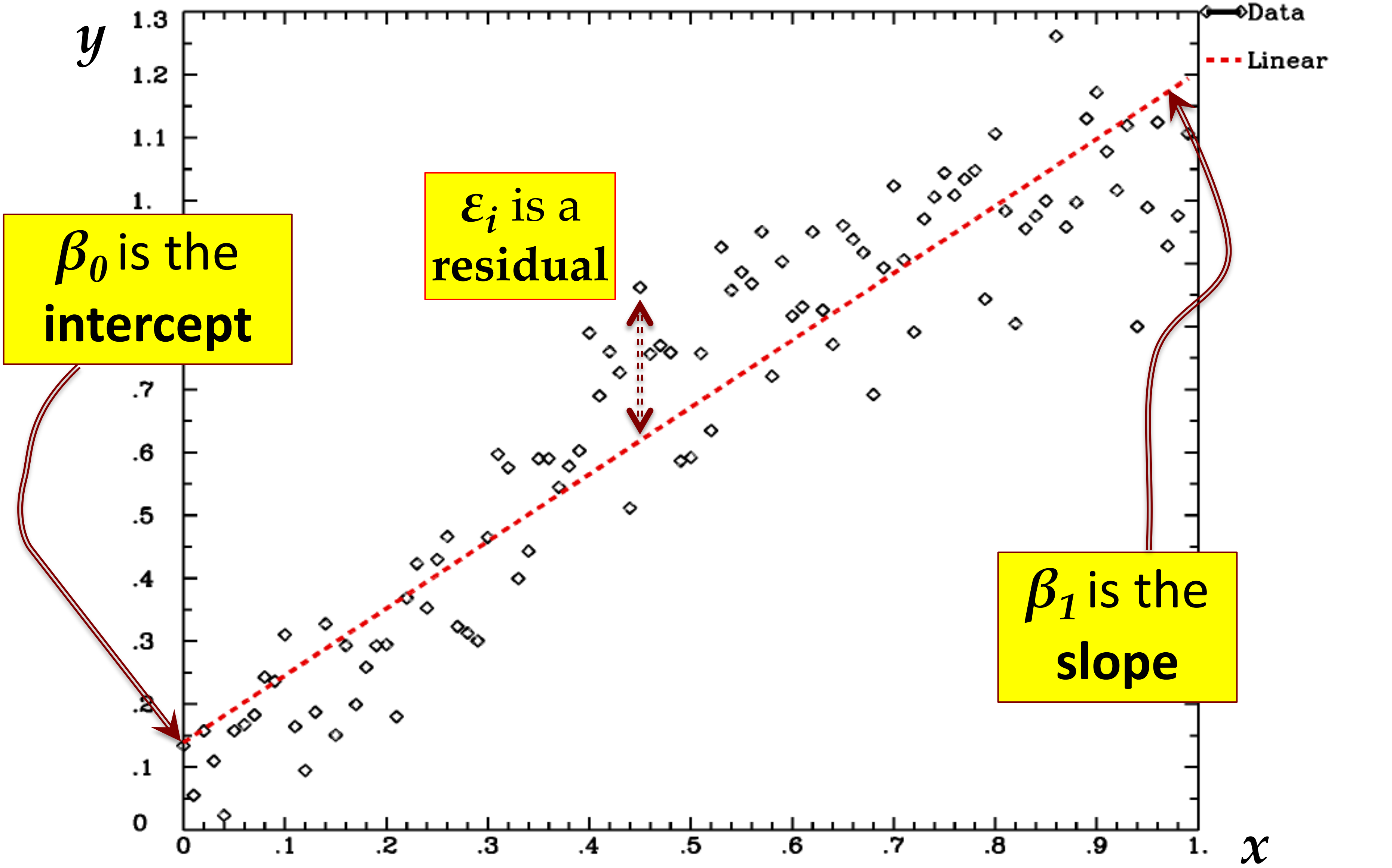
- **Regression**:
  - o Finding a mathematical relationship between
    - ▪ a measured response/outcome (dependent) variable
    - ▪ and one or more explanatory (independent) variables (**regressors**)
  - o Also called **linear modeling** or **linear regression**
- **Linear = Additive** = model for data is sums of regressors
  - o **Goal**: find out how much each regressor is needed

Basics of Linear Modeling

- Simple Sample Equations
  - $\boldsymbol{i}$ = index of data = 0, 1, 2 … N-1 (total of N data points)
  - $\boldsymbol{x_i}$ = explanatory model (known) for data point number $\boldsymbol{i}$
  - $\boldsymbol{y_i}$ = data value for data point number $\boldsymbol{i}$
  - $\boldsymbol{y_i = \beta_0 + \beta_1 x_i + \varepsilon_i}$ **or** $\boldsymbol{y_i \approx \beta_0 + \beta_1 x_i}$
  - $\boldsymbol{\beta_0}$ and $\boldsymbol{\beta_1}$ are **model fit parameters**
    - to be calculated from the $\boldsymbol{x_i}$ and $\boldsymbol{y_i}$
  - $\boldsymbol{\varepsilon_i}$ are the **residuals**
    - what are left after regression
    - assumed to be **random noise**



Linear Fit: $y_i \approx \beta_0 + \beta_1 x_i$

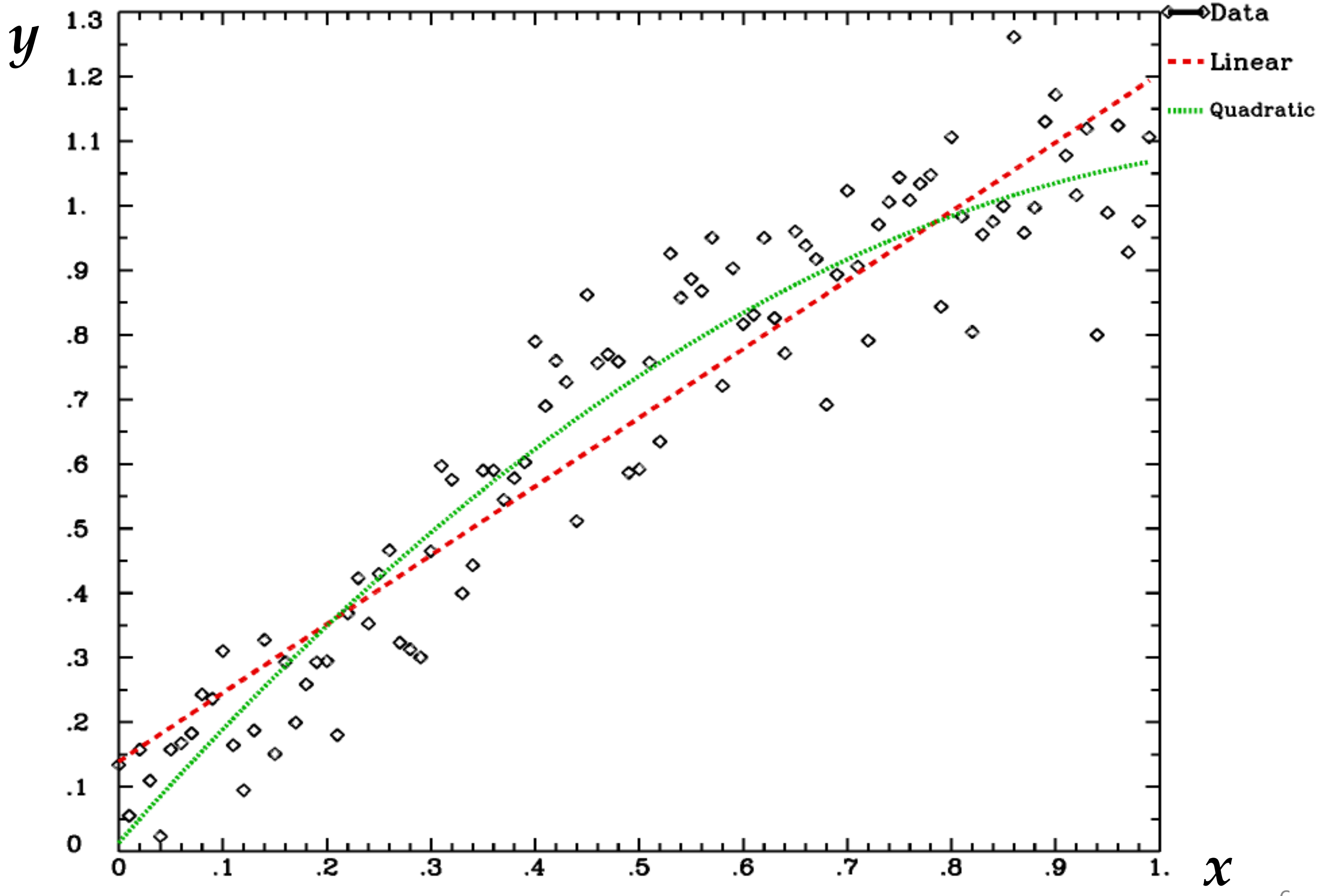


β_0 is the **intercept**

ε_i is a **residual**

β_1 is the **slope**

Quadratic Fit: $y_i \approx \beta_0 + \beta_1 x_i + \beta_2 x_i^2$



AFNI Script

```
1deval -num 100 -dt 0.01  \
       -expr "abs(sin(1.7*t)+gran(0,0.1))" > s1.temp.data.1D

3dTfitter -RHS s1a.temp.data.1D   \
          -polort 1 -prefix NULL -fitts s1a.temp.fitts.1.1D
3dTfitter -RHS s1a.temp.data.1D   \
          -polort 2 -prefix NULL -fitts s1a.temp.fitts.2.1D

1dplot -one -dt 0.01 -xaxis 0:1:10:5   \
       -dashed 0:2:3 -png s1a.fit1     \
       -ynames Data Linear -           \
       s1a.temp.data.1D s1a.temp.fitts.1.1D

1dplot -one -dt 0.01 -xaxis 0:1:10:5   \
       -dashed 0:2:3 -png s1a.fit2     \
       -ynames Data Linear Quadratic - \
       s1a.temp.data.1D s1a.temp.fitts.1.1D s1a.temp.fitts.2.1D
```

Script to produce
plots on previous slides

s1a.TimeSeriesAnalysis.LinearRegression.csh

Modeling with Vectors and Matrices

- Write the model $y_i \approx \beta_0 + \beta_1 x_i$ out in columns (**vectors**)

y vector

$$\underbrace{\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \cdots \end{bmatrix}}_{\text{data } \mathbf{vector}} \approx \begin{bmatrix} 1 \\ 1 \\ 1 \\ \cdots \end{bmatrix} \beta_0 + \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \cdots \end{bmatrix} \beta_1 = \underbrace{\begin{bmatrix} 1 & x_0 \\ 1 & x_1 \\ 1 & x_2 \\ \cdots & \cdots \end{bmatrix}}_{N\times 2\ \mathbf{matrix}} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}$$

β vector

X matrix

- In **vector-matrix** form (**bold** letters for vectors/matrices)
  - $\mathbf{y} \approx \mathbf{X}\,\boldsymbol{\beta}$ *or,* with residual vector $\mathbf{y} = \mathbf{X}\,\boldsymbol{\beta} + \boldsymbol{\varepsilon}$
- Writing it out this way, equations become more compact, easier to look at, easier to understand at a single glance (with practice)

Modeling with Vectors and Matrices

- Write the model $y_i \approx \beta_0 + \beta_1 x_i$ out in columns (**vectors**)

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \dots \end{bmatrix} \approx \begin{bmatrix} 1 \\ 1 \\ 1 \\ \dots \end{bmatrix} \beta_0 + \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \dots \end{bmatrix} \beta_1 = \begin{bmatrix} 1 & x_0 \\ 1 & x_1 \\ 1 & x_2 \\ \dots & \dots \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}$$

- Each column of **X** matrix is a **regressor** (or **model component**)
- We assume the columns of **X** are known ("the model"), and that data vector **y** is known (measured)

Modeling with Vectors and Matrices

- Write the model $y_i \approx \beta_0 + \beta_1 x_i$ out in columns (**vectors**)

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \cdots \end{bmatrix} \approx \begin{bmatrix} 1 \\ 1 \\ 1 \\ \cdots \end{bmatrix} \beta_0 + \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \cdots \end{bmatrix} \beta_1 = \begin{bmatrix} 1 & x_0 \\ 1 & x_1 \\ 1 & x_2 \\ \cdots & \cdots \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}$$

- Goal is to compute **parameter vector β** (and statistics about **β**)
- <u>Much of sections 3, 4, and 6 that follow</u>:
  - o Where do we get **X** for FMRI task analysis?

Solving a Linear Model

Vector **y** is sum of matrix **X** times vector β plus residuals ε

- Solution for linear regression $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$
  - "Project" data **y** onto "space" of explanatory variables (**X**)
  - **OLSQ** formula for solution: $\hat{\boldsymbol{\beta}} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}$
  - Columns of **X** are the **model** for data vector **y**
- Meaning of coefficients $\boldsymbol{\beta}$:
  - β_k value is slope, or marginal effect, or effect size associated with **regressor** number *k* [column *k* in **X**]
- β_k value says how much of regressor number *k* is needed to fit the data "best" – in the **O**rdinary **L**east **SQ**uares sense
  - The sum of squares of ε_i is made as small as possible by adjusting all entries in $\boldsymbol{\beta}$ to make it so

Solving a Linear Model

- Solution for linear regression $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$
  - o "Project" data **y** onto "space" of explanatory variables (**X**)
  - o **OLSQ** formula for solution: $\hat{\boldsymbol{\beta}} = (\mathbf{X}^\mathbf{T}\mathbf{X})^{-1}\mathbf{X}^\mathbf{T}\mathbf{y}$
  - o Columns of **X** are the **model** for data vector **y**

- If we don't care about regressor number *k,* then we don't care about the value of β_k – or any statistics about it
  - o But we included regressor number *k* in the model because it was needed to fit some part of the data
- **Regressors of no interest** make up the global **Null Hypothesis** in the model – in **AFNI**, we call these regressors the **baseline model**

Statistics in a Linear Model

- Various statistical tests can be carried out after solving for **β** vector
- Some examples, with particular null hypotheses H_0
  - Student *t*-test for each β_i of interest

    H_0: $\beta_3 = 0$ [task has no response?]
  - Student *t*-test for linear combination of some β_i values = general linear test (GLT)

    H_0: $\beta_3 - \beta_5 = 0$

    [two tasks have equal response?]

    H_0: $0.5^*(\beta_3 + \beta_4) - \beta_5 = 0$

    [average response of two tasks = third task response?]

Statistics in a Linear Model

- Various statistical tests can be carried out after solving for **β** vector
- Some examples, with particular null hypotheses H_0
  - o *F*-test for composite null hypothesis

$$H_0\text{:}\ \ \beta_3 = \beta_4 = \beta_5$$

[all 3 tasks have identical responses?]

$$H_0\text{:}\ \ \beta_3 = \beta_4 = \ \beta_5 = 0$$

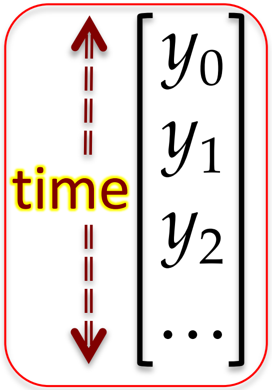
[all 3 tasks have no response at all?]

  - o Omnibus or Full *F*-test for the entire model

H_0: all β_i values of interest are 0

Linear Model with FMRI

- Time series regression: data vector **y** is time series = all values from ***one*** voxel throughout multiple image acquisitions (TRs)


time

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \ldots \end{bmatrix}$$

- Regressors: idealized BOLD response curves
  - o We can only find what we're looking for
  - o Regression will miss something if we do not look for it
    - So we must include regressors of no interest, so we can model things like baseline drifting up or down

Linear Model with FMRI

- Regressor construction requires decisions
  - How to model response(s) we look for?
  - What kind of regressors of no interest to include, and how many of them?
  - Don’t want to **over**-fit or **under**-fit data
- **Usually**: Same model matrix **X** for all voxels in brain
  - Simultaneously solve all the models (1 for each voxel)
  - Voxel-wise analysis = “massively univariate” method

AFNI

Didactics and Demonstrations

FMRI Task-Based Data Analysis at the Individual Level

Concepts 2 – FMRI Data

FMRI Experiment Terminology

- Experiment setup
    - Number of subjects
    - Number of conditions
        - Tasks, stimulus (trial, event) types
        - Factorial design?
    - Sample size (repetitions) per condition
    - Block, event-related, or mixed?
    - Inter-stimulus interval (ISI) – regular, random?

FMRI Experiment Terminology

- Scanning parameters:
  - TR = time between repetitions (3D volumes)
  - echo time (TE) voxel size; number of 3D volumes; slice sequence (interleaved, multi-slice); slice thickness; removing first few TRs
- Scanning terms
  - Run: continuous scanning; brief break between runs
  - Session: subjects return after long period of time
  - Experiment or study

Types of FMRI Experiments

- Two classical types of experiment design

- **Block** (boxcar) design
  - Each stimulus block lasts for more time than BOLD response takes to rise (e.g., 6+ sec)
  - Each block is under one task condition (e.g., watch a video clip), or a series of multiple short exchangeable trials (e.g., 10 consecutive face images)
    - BOLD responses from close-in-time trials overlap and are not distinguishable in the data
  - BOLD response is often visible in time series
  - SNR: noise size about same as BOLD response

Types of FMRI Experiments

- Two classical types of experiment design

- **Event-related** design
  - Each event or trial is distinguishable from others
    - Spaced apart in time enough for BOLD responses to be separately identified
  - Events often randomly spaced in time
  - BOLD response to stimulus tends to be weaker, since fewer nearby-in-time activations have overlapping signal changes
  - Data looks more like noise (to the pitiful human visual system)

Types of FMRI Experiments

- Other types of experiment design

- **Mixed designs**
  - Containing both events and blocks
  - *e.g.*, cue on what to pay attention to + face images
    - "Is face angry or happy?" vs "Is face female or male?"
    - Block = cue ; Event = individual image inside block
- **Naturalistic stimulation** (*e.g.*, movie watching)
  - Not directly covered here
  - Like resting state analysis in the first stages
    - no task response model but with regressors of no interest – to reduce unwanted effects (*e.g.*, head motion)

FMRI Data

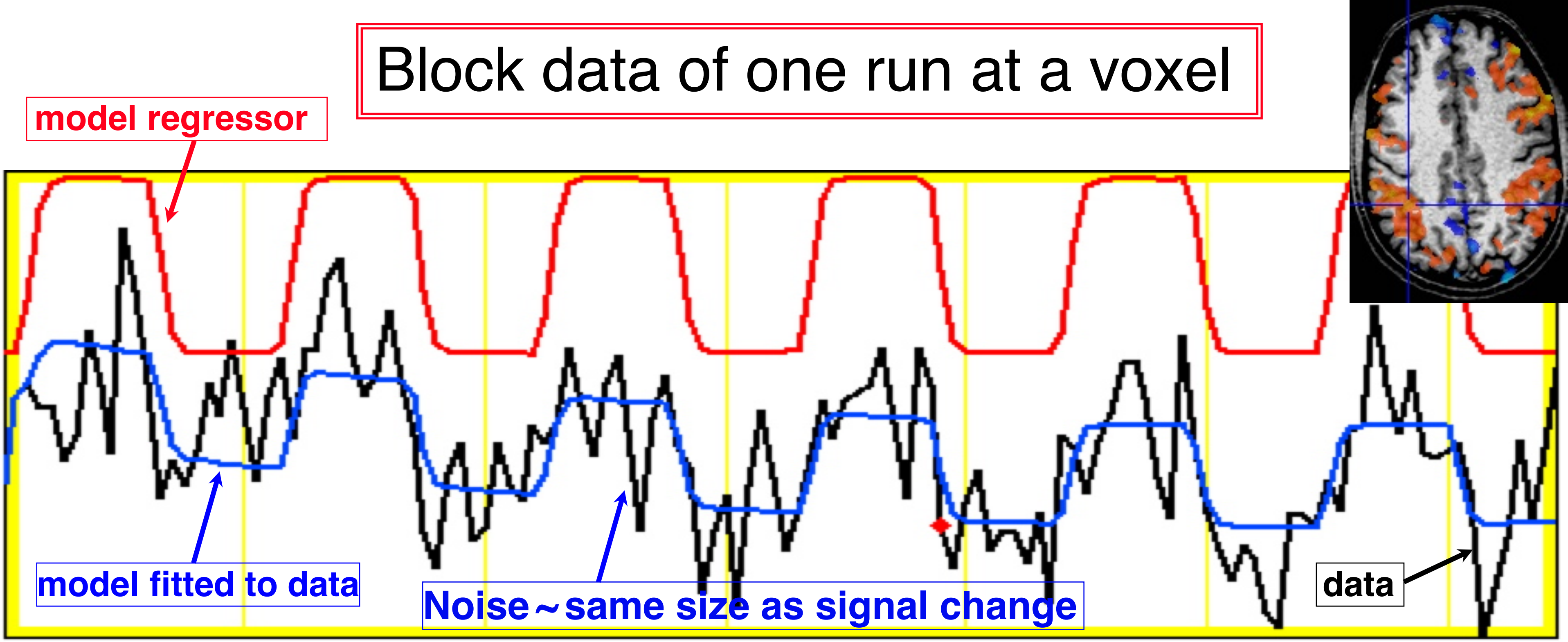
- Data partition: **Data** = **Signal** + **Noise**

  - **Data** = from scanner (voxel-wise time series)
  - **Signal** = BOLD response to stimulus = effects of interest + effects of no interest
    - **We don't actually know real signal shape to look for*!!!***
    - Look for idealized task responses by assuming a **fixed shape** for BOLD effect (FMRI response) for each task trial
    - ***Or*** search for signal shape via **basis functions**
    - Of interest: effect size (response size) for tasks=**betas**
    - Of no interest: baseline, slow drifts, head motion effects, respiration …

FMRI Data

- Data partition: **Data** = **Signal** + **Noise**
  - ○ **Noise** = components in data that interfere with signal detection
    - Practically: the part of the data we can't explain with the model
    - Must make some assumptions about its probability distribution – to be able to carry out the statistical tests
- **Data = baseline + slow drift + other effects of no interest** + **response$_1$ + … + response$_k$** + **noise**
- How to construct the regressors of interest (responses)? And the regressors of no interest?

Block data of one run at a voxel



model regressor

model fitted to data

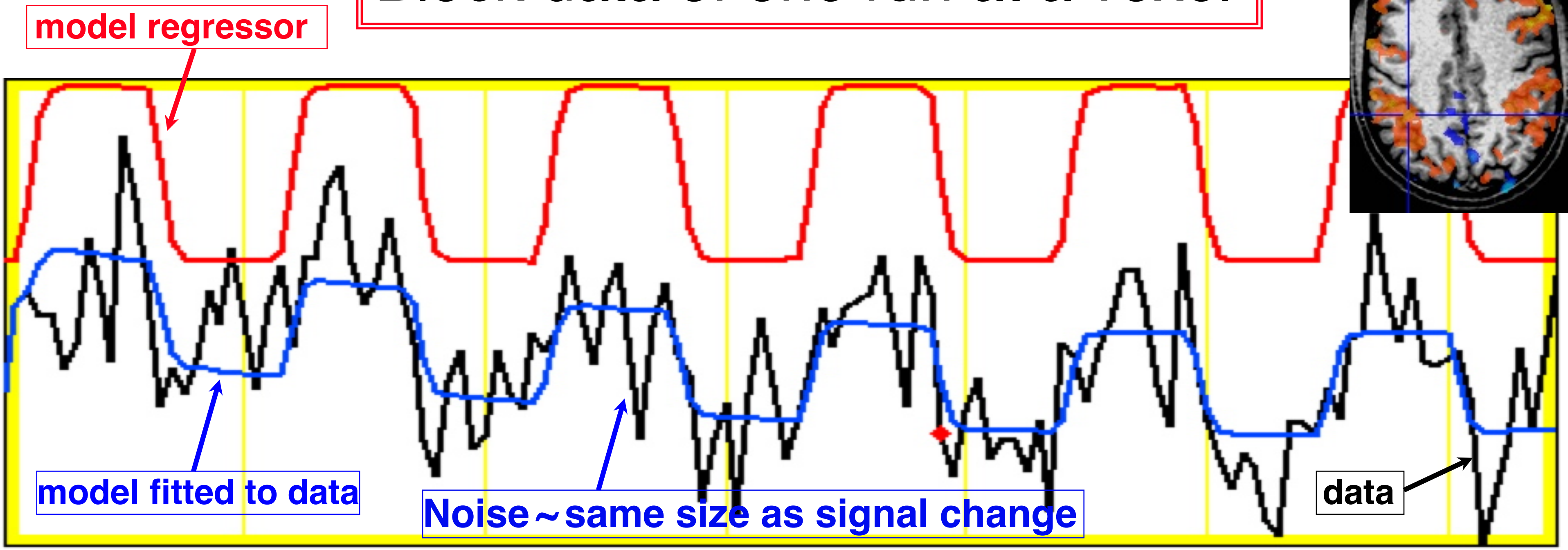
Noise~same size as signal change

data

Block: 27 s “on” / 27 s “off”; TR=2.5 s; 130 time points

- This is “best” voxel; most voxels are not fitted as well as this
- Noise size about same as block activation size

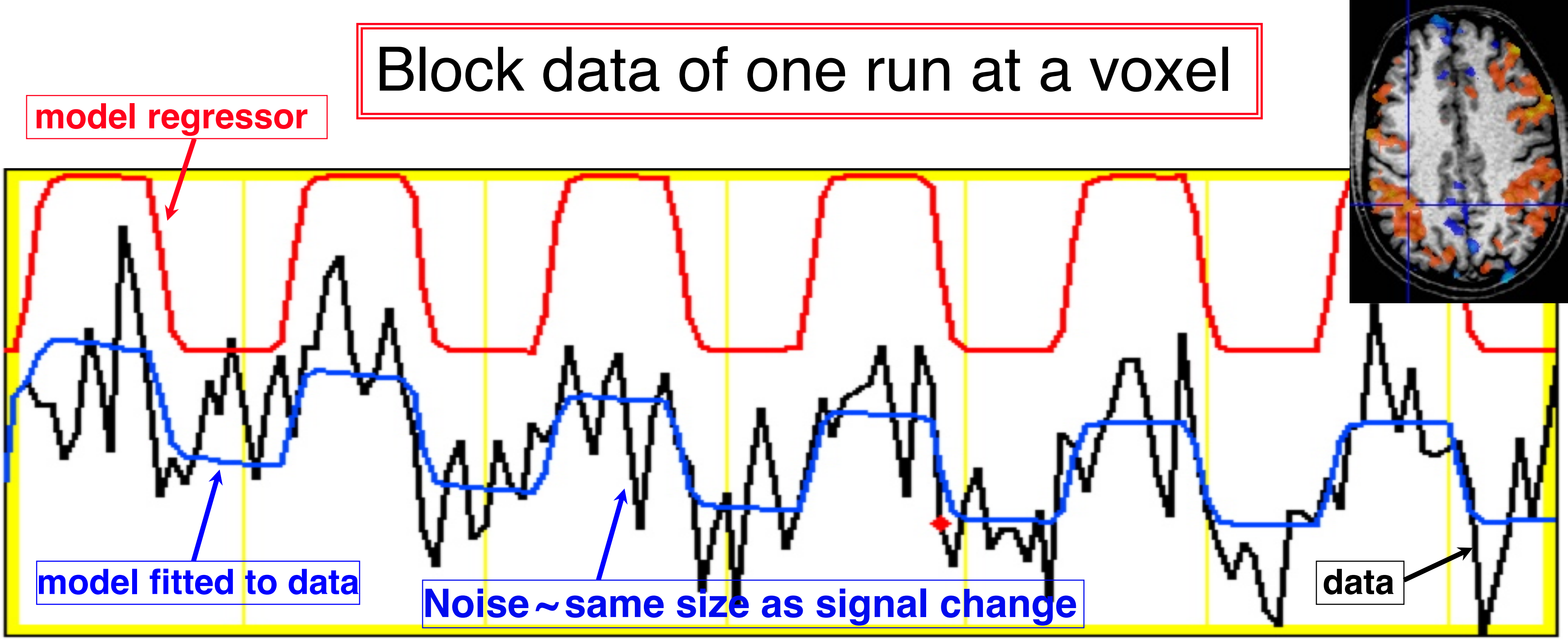
Block data of one run at a voxel



Block: 27 s “on” / 27 s “off”; TR=2.5 s; 130 time points

- Data drifts downwards slowly – this effect is captured in the model fit by baseline drift regressors
- If we did *not* model for drift, our fit would not be as good

Block data of one run at a voxel



model regressor

model fitted to data

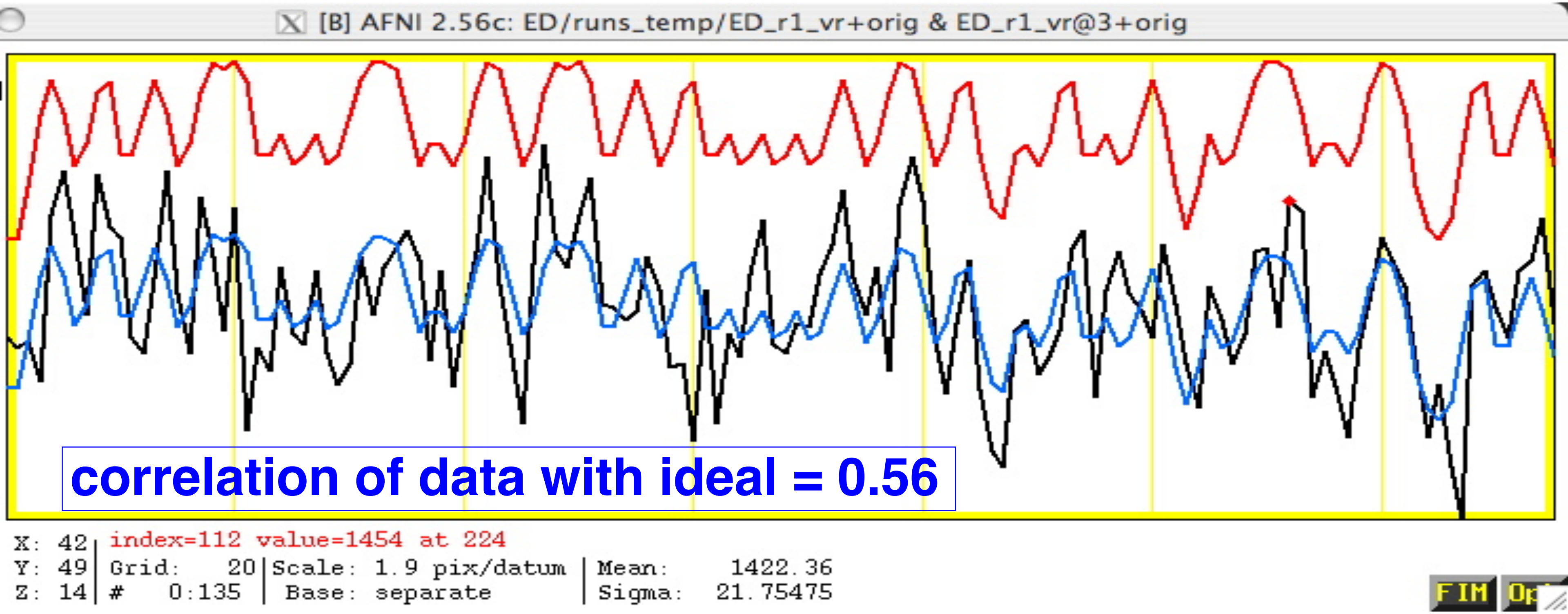
Noise ~ same size as signal change

data

Block: 27 s “on” / 27 s “off”; TR=2.5 s; 130 time points

- Activation amplitude and shape vary across blocks
  - Reasons why? We can only guess 😢
  - Habituation? Attention? Noise? Respiration?

Event related design data of one run at a voxel



[B] AFNI 2.56c: ED/runs_temp/ED_r1_vr+orig & ED_r1_vr@3+orig

correlation of data with ideal = 0.56

```
X: 42| index=112 value=1454 at 224
Y: 49| Grid:    20|Scale: 1.9 pix/datum |Mean:     1422.36
Z: 14| #    0:135 | Base: separate      |Sigma:   21.75475
```

Lesson: ER-FMRI activation is not obvious via casual inspection

AFNI

Didactics and Demonstrations

FMRI Task-Based Data Analysis at the Individual Level

Concepts 3 – FMRI Fixed-Shape Models

BOLD Response

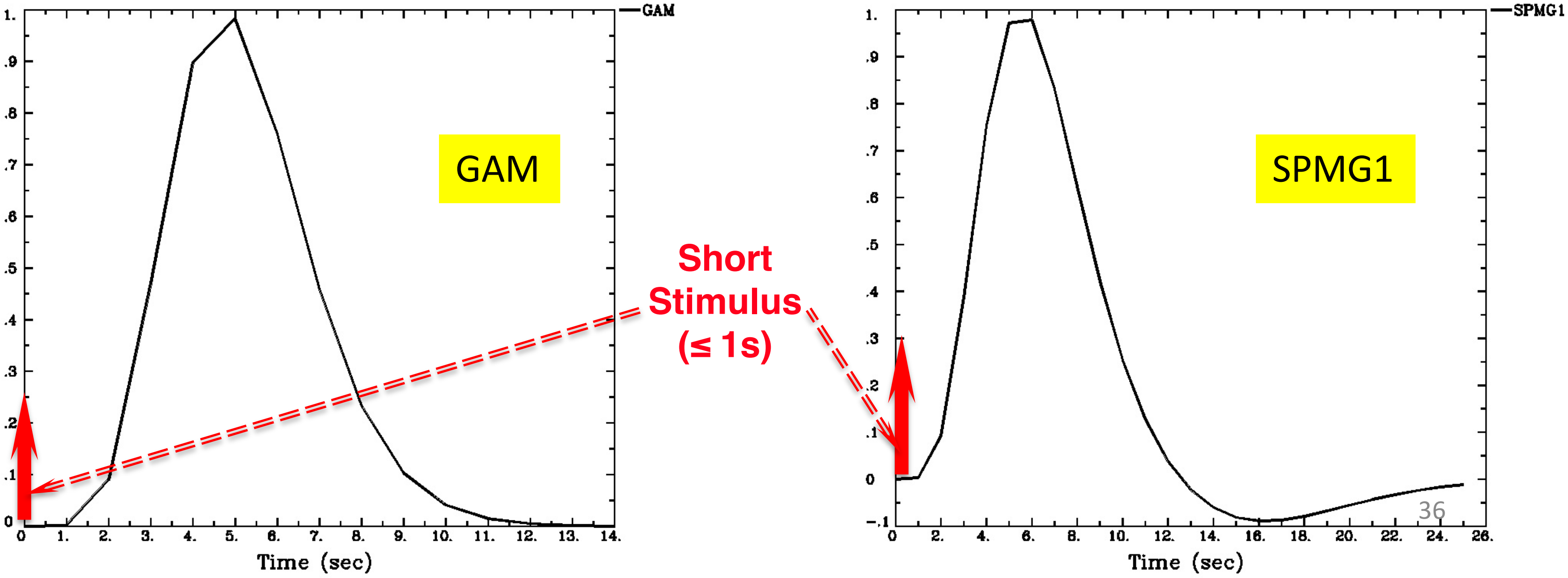
- Hemodynamic response (**HDR**)
  - Brain+FMRI response to stimulus/task/condition
  - Indirect measure of neural response: brain activation ➔ changes in blood oxygen ➔ changes in FMRI signal
- Hemodynamic response function (**HRF**)
  - Mathematical formulation/idealization of HDR for ***one*** full stimulus interval
  - HRF *bridges* between neural response (what we like) and BOLD signal (what we measure)
  - Multiple copies of HRF are needed to model responses to multiple stimuli

BOLD Response

- How to build the HRF bridge?
  - **Most simple**: Assume a fixed-shape (idealized) HRF – one β output per task (per voxel)
    - This is the most common approach in FMRI
  - **Most complex**: No assumption about HDR shape
    - Basis function expansion of HRF shape and size
    - Multiple functional shapes added up to give an adjustable shape
    - Multiple $\beta's$ instead of a single β
  - **In the middle**: 1 major fixed shape + a little space for shape adjustment

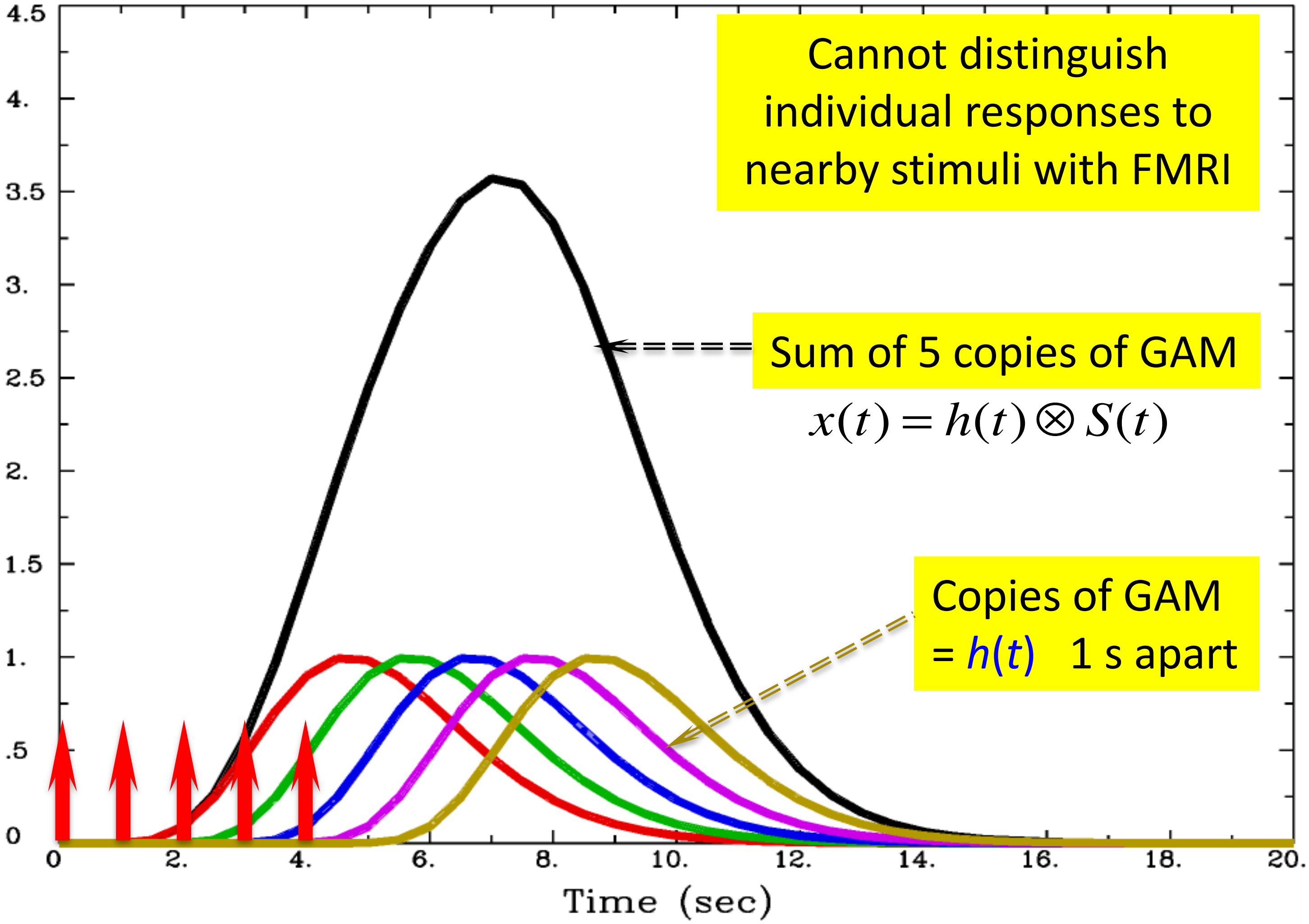
Fixed-Shape HRF – ≤ 1 s Stimulus

- Assume a fixed shape *h*(*t*) for HRF to a very short stimulus: **impulse response function** (**IRF**)
  - ○ **GAM**(*p*,*q*): $h(t) = t^{8.6} \exp(-t/0.547)$ [MS Cohen, 1997]
    - ▪ A variation: **SPMG1** (undershoot is added in)



Fixed-Shape HRF – 5 s Stimulus

- Combine IRF *h*(*t*) with stimulus duration:



Cannot distinguish individual responses to nearby stimuli with FMRI

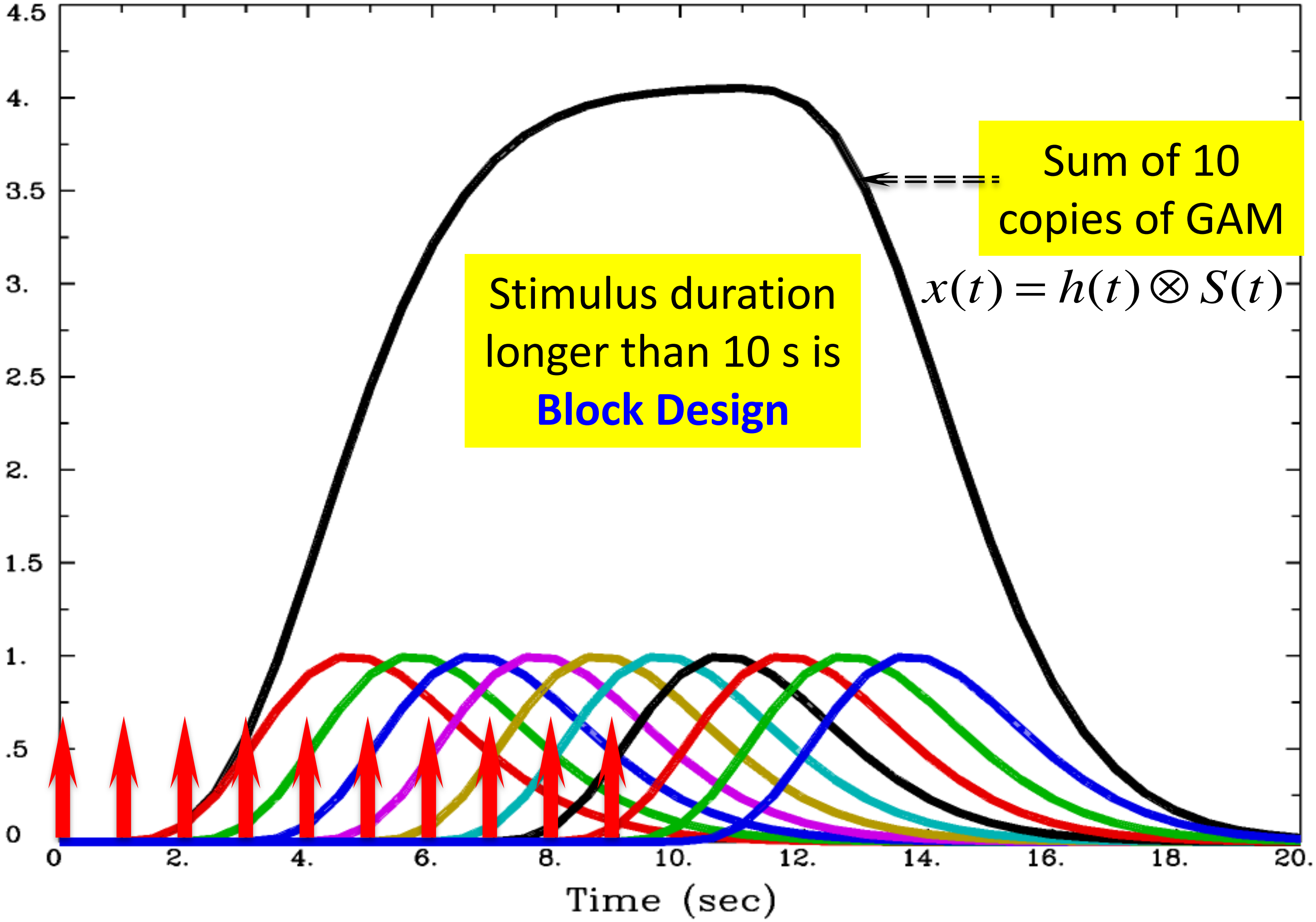
Sum of 5 copies of GAM

$$x(t) = h(t) \otimes S(t)$$

Copies of GAM = *h*(*t*) 1 s apart

Fixed-Shape HRF – 10 s Stimulus

- Combine IRF *h*(*t*) with stimulus duration:



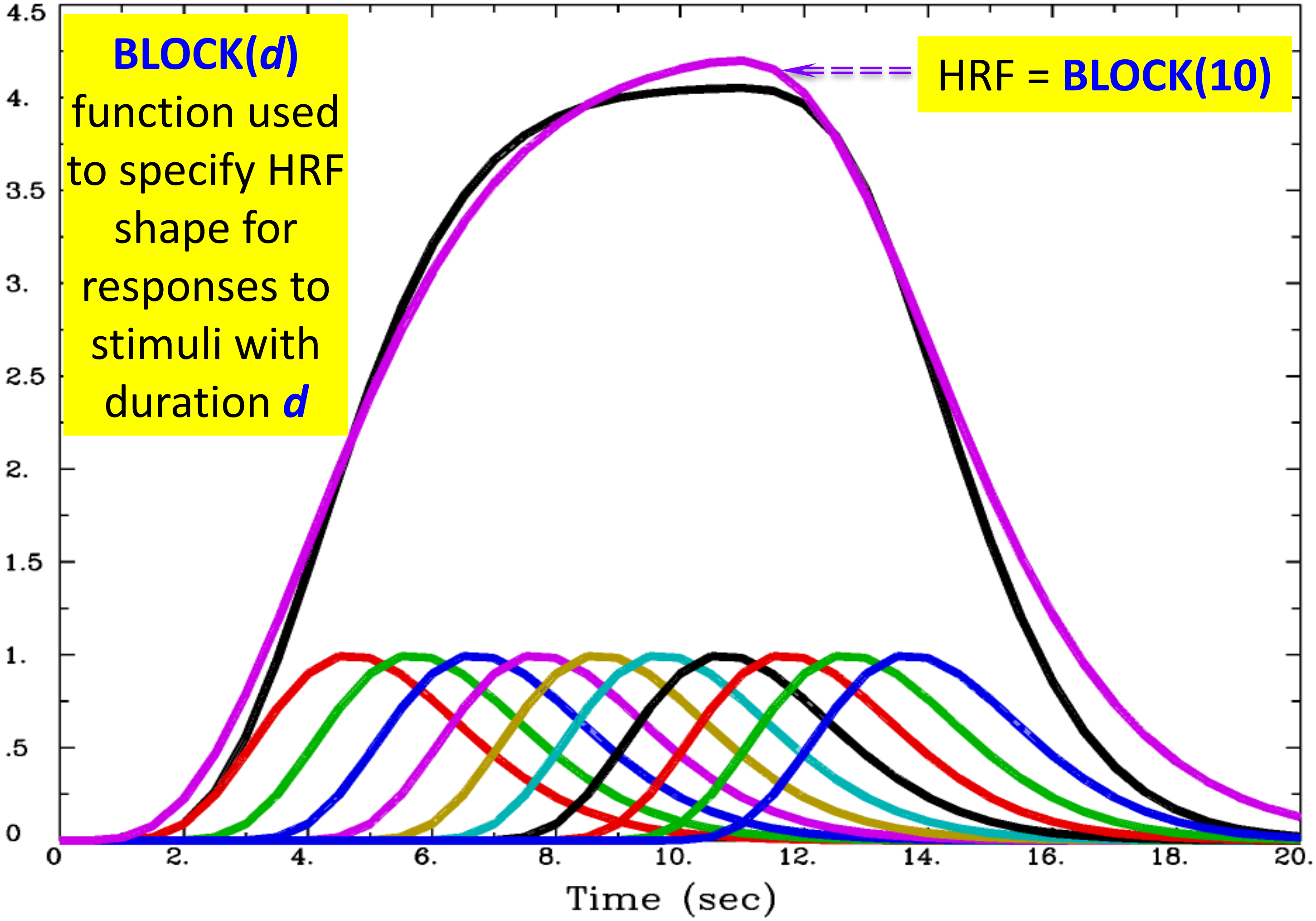
Sum of 10 copies of GAM

$$x(t) = h(t) \otimes S(t)$$

Stimulus duration longer than 10 s is **Block Design**

Fixed-Shape HRF – 10 s Stimulus

- With the '**BLOCK(10)**' function in **AFNI**

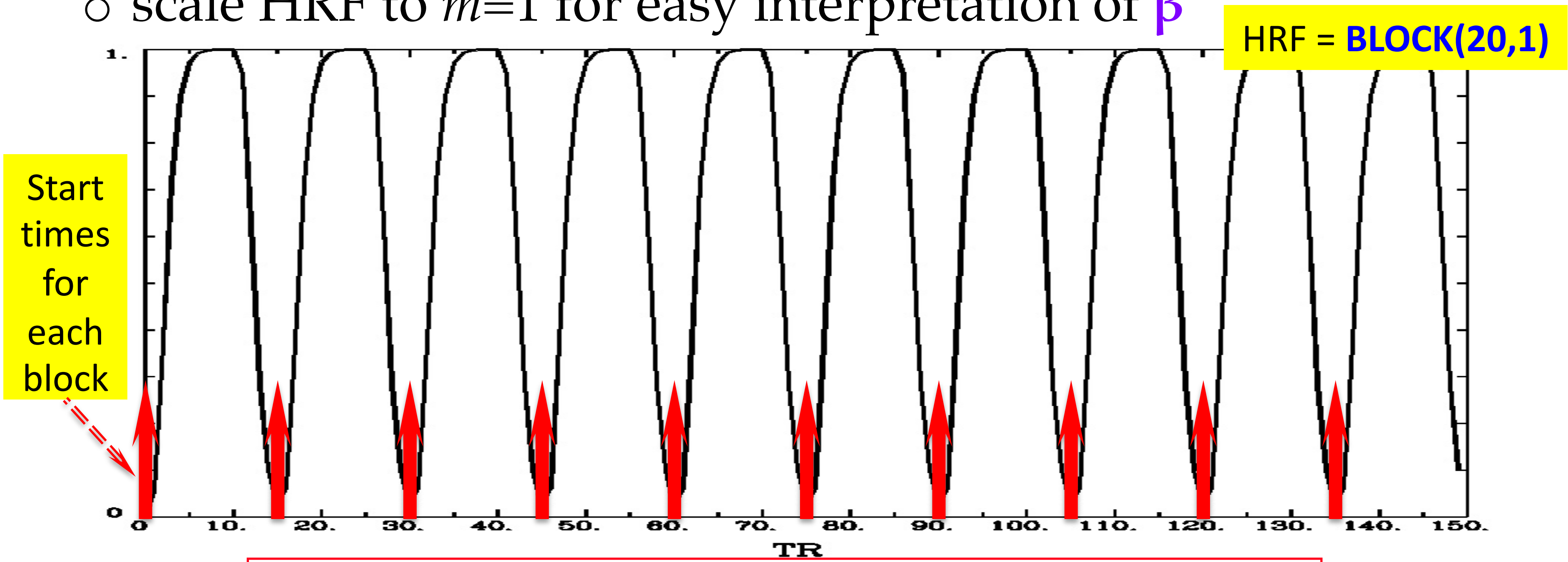


BLOCK(*d*) function used to specify HRF shape for responses to stimuli with duration ***d***

HRF = **BLOCK(10)**

Fixed-Shape HRF for Block Design

- *For each block,* IRF *h*(*t*) is "**convolved**" with **stimulus start time** *and* **duration** (*d*) to get regressor
  - HRF = **BLOCK(*d*,*m*)**
  - **Equivalent to adding up sequence of consecutive events**
  - scale HRF to *m*=1 for easy interpretation of β



HRF = **BLOCK(20,1)**

Start times for each block

Blocks: 20 s on and 10 s off; TR=2 s; 150 time points

AFNI Script

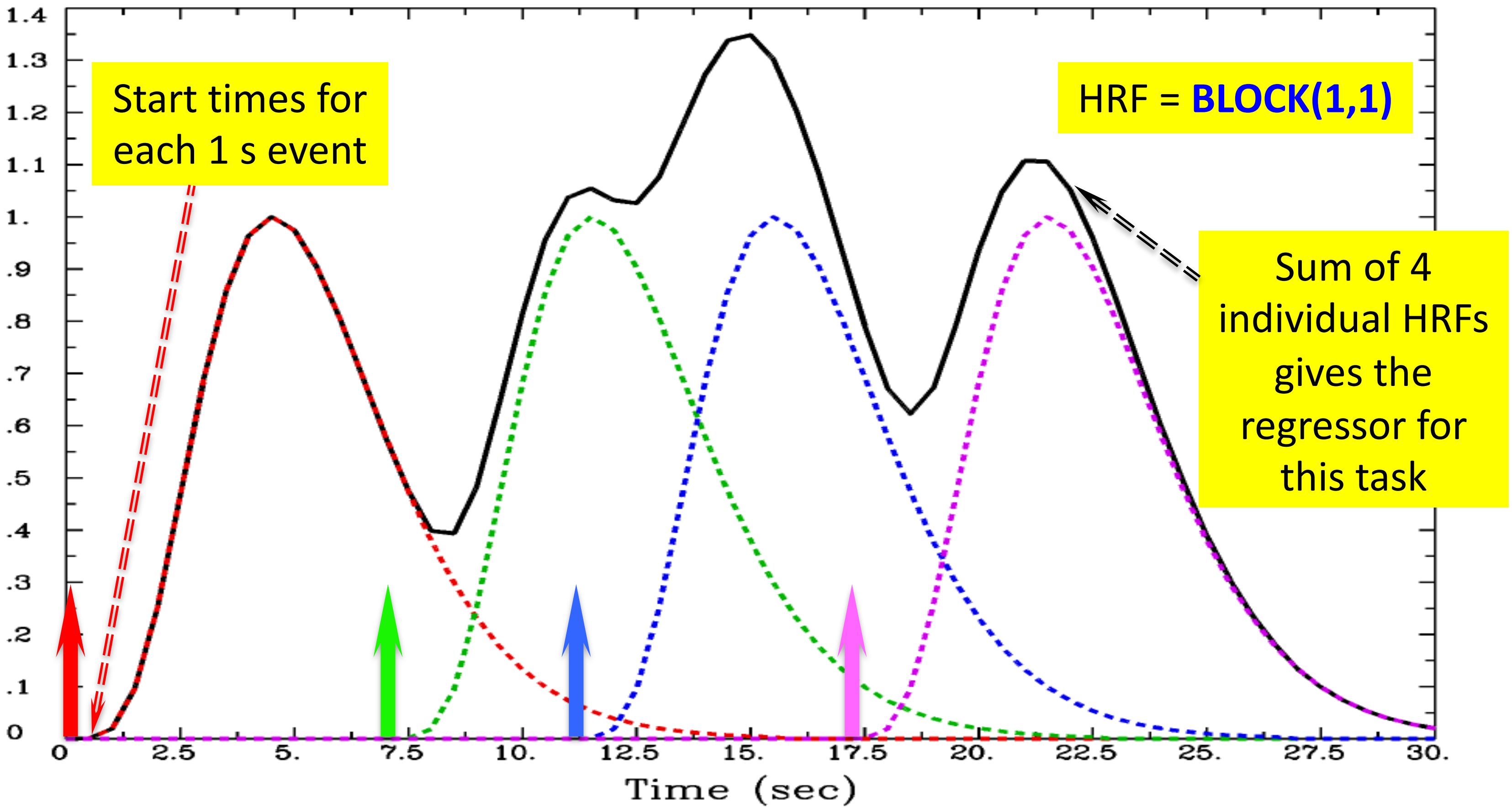
```
3dDeconvolve -nodata 150 2.0  \
             -polort -1       \
             -x1D s3a.xmat.1D \
             -num_stimts 1    \
             -stim_times 1    \
              '1D:  0 30 60 90 120 150 180 210 240 270' \
              'BLOCK(20,1)'

1dplot -xaxis 0:150:15:2 -xlabel TR -png s3a.png s3a.xmat.1D
```

Script to produce plot on previous slide

s3a.TimeSeriesAnalysis.BlockModel.csh

Fixed-Shape HRF for Event-Related Design

- **BLOCK** HRF shape also useful with event-related experiments
- Just use a short duration, such as 1 second
- Real experiments have more than 4 task repetitions!



Start times for each 1 s event

HRF = **BLOCK(1,1)**

Sum of 4 individual HRFs gives the regressor for this task

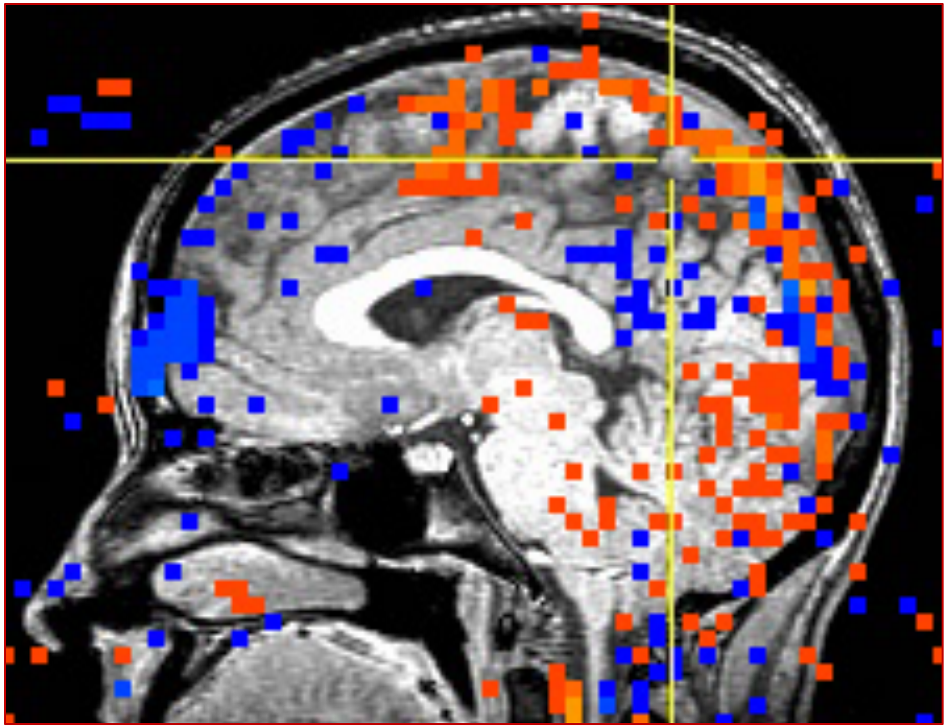
Linear Model with Fixed-Shape HRF

- FMRI data = **baseline**+**response**$_1$+…+**response**$_k$+**noise**
- "**baseline**" = **baseline constant** + **drift up or down** + **other effects of no interest** (e.g., motion)
  - **drift**: caused by physiological effects, tiny motions, scanner fluctuations, …
  - "**baseline**" is treated in **AFNI** as the **null hypothesis model**, an additive effect, not an effect of interest
  - "**baseline**" also needs parameters in the model fit
    - For the constant, the drift shape, and other effects
    - These parameters are not "of interest" and are not included in the **Full F** statistic of response model fit

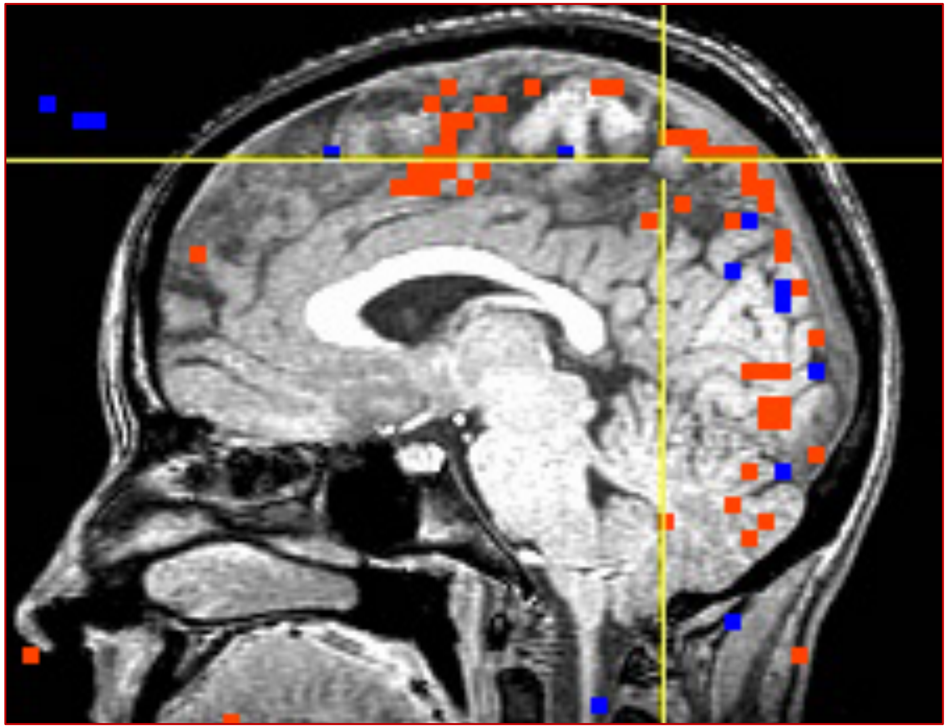
Linear Model with Fixed-Shape HRF

- $y_i = \alpha_0 + \alpha_1 t_i + \alpha_2 t_i^2 + \beta_1 x_{1i} + \ldots + \beta_k x_{ki} + \ldots + \varepsilon_i$ [i = time]
- $\mathbf{y} = \mathbf{X}\beta + \varepsilon$, $\mathbf{X} = [1, t, t^2, x_1, x_2, \ldots, x_k, \ldots]$ [vector format]
- In **AFNI**, **baseline + slow drift** is modeled with polynomials: $\alpha_0 + \alpha_1 t_i + \alpha_2 t_i^2$ (polynomial order=2)
  - Longer run needs a higher order of polynomials
    - One order per 150 sec is the default in **AFNI**
      - Actually uses Legendre polynomials for accuracy
  - With $m > 1$ runs, m sets of polynomials needed to allow for temporal discontinuities across runs
  - $m(p+1)$ columns just for **baseline+slow drift** (order=p)
- Another effect of no interest: head movement ➜➜➜

Stimulus Correlated Motion = Bad



Activation map with image registration but *without* using movement estimates as regressors



Activation map when also using 6 movement estimates as regressors

Lesson: movement regressors (of no interest) are necessary!

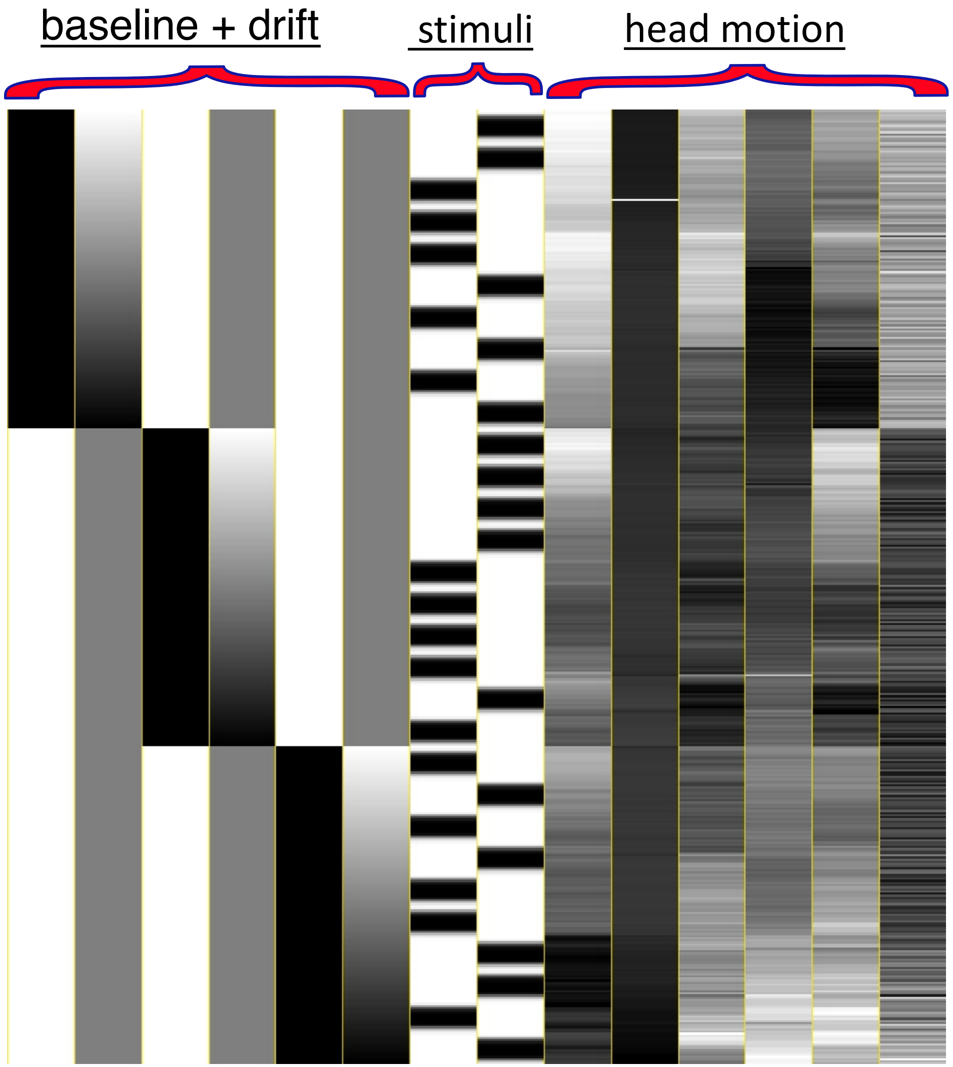
Design Matrix **X** with Fixed-Shape HRF

- Voxel-wise (massively univariate) linear model

$$y = \mathbf{X}\beta + \varepsilon$$

  - **X**: explanatory variables (regressors; ″the model″)
    - **same** across voxels (in most analyses)
  - y: data (time series) at a voxel (from scanner)
    - **different** across voxels
  - β: regression coefficients (effect magnitudes)
    - **different** across voxels
  - ε: anything we can't account for ("noise")
    - **different** across voxels

Design Matrix **X** with Fixed-Shape HRF

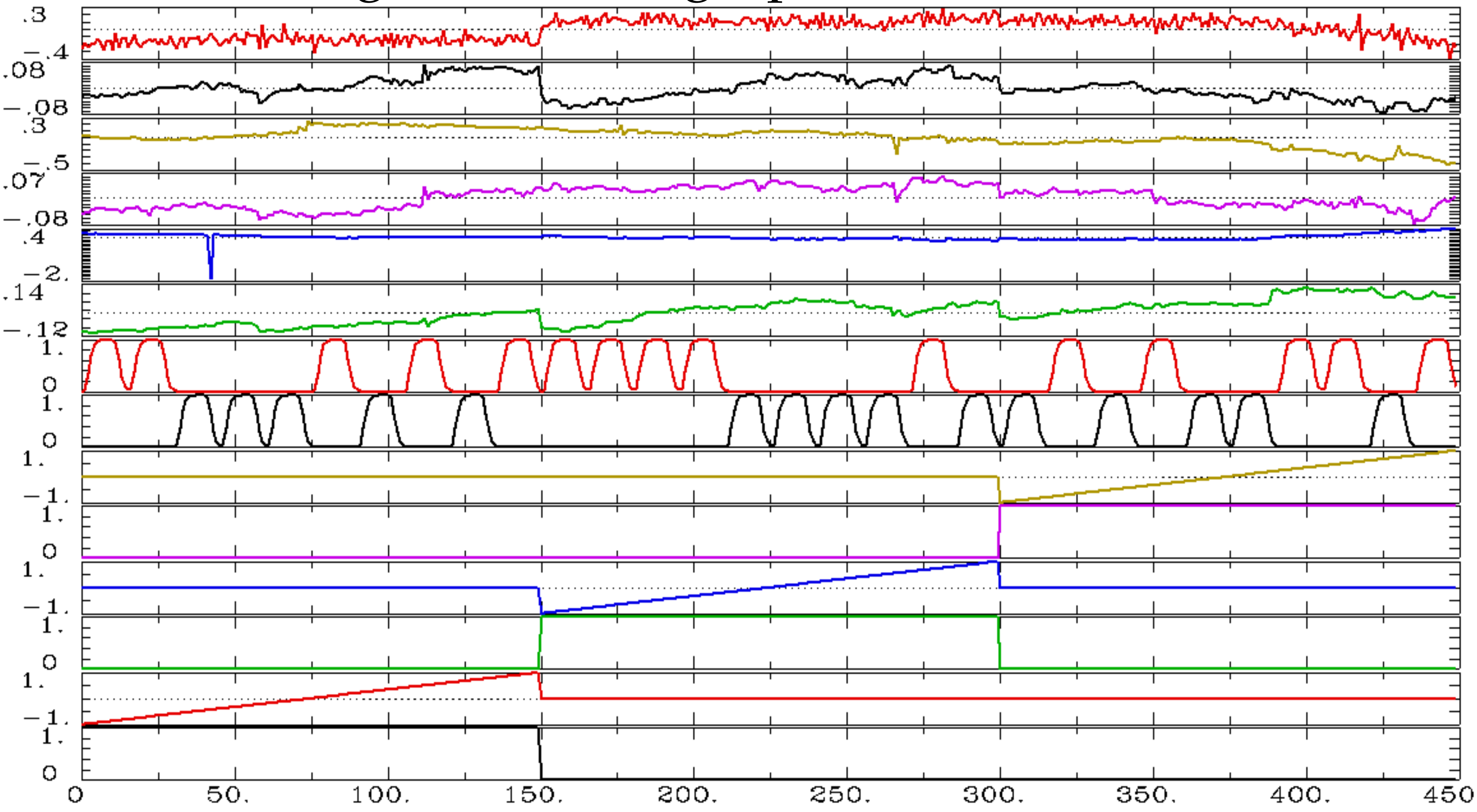


- 6 drift effect regressors
  - linear (p=1) baseline model
  - 3 runs x 2 parameters/run
- 2 regressors **of interest**
  - *i.e.*, relevant to brain activity
  - from 2 distinct tasks
- 6 head motion regressors
  - 3 rotations + 3 shifts

Black = bigger numbers
White = smaller numbers
Each column of **X** scaled separately
Image produced by afni_proc.py

Design Matrix X with Fixed-Shape HRF

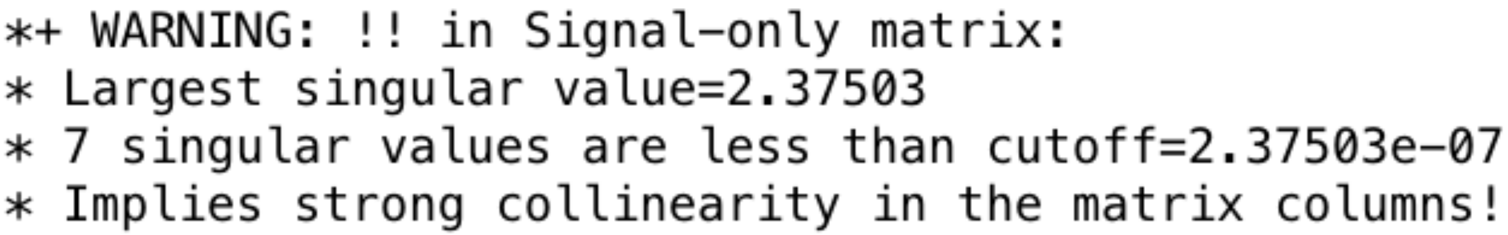
- Same design matrix in graphs



Model Quality Check

- First thing to do!
  - Most users in FMRI simply jump to specific effects of interest, their contrasts, and their significance. They simply don’t pay attention to overall model performance

- Approaches to judge your model
  - Design matrix report from **3dDeconvolve**



```
*+ WARNING: !! in Signal-only matrix:
* Largest singular value=2.37503
* 7 singular values are less than cutoff=2.37503e-07
* Implies strong collinearity in the matrix columns!
```

This message is usually due to setup mistakes

Model Quality Check

- First thing to do!
  - Most users in FMRI simply jump to specific effects of interest, their contrasts, and their significance. They simply don’t pay attention to overall model performance
- Approaches to judge your model
  - Full *F*-statistic (automatically provided in **AFNI**)
  - Testing compares two possibilities (voxel-wise)
    - Data = ‘**baseline**‘ + ***all* effects of interest** + **noise**

      versus
    - Data = ‘**baseline**‘ + **noise**

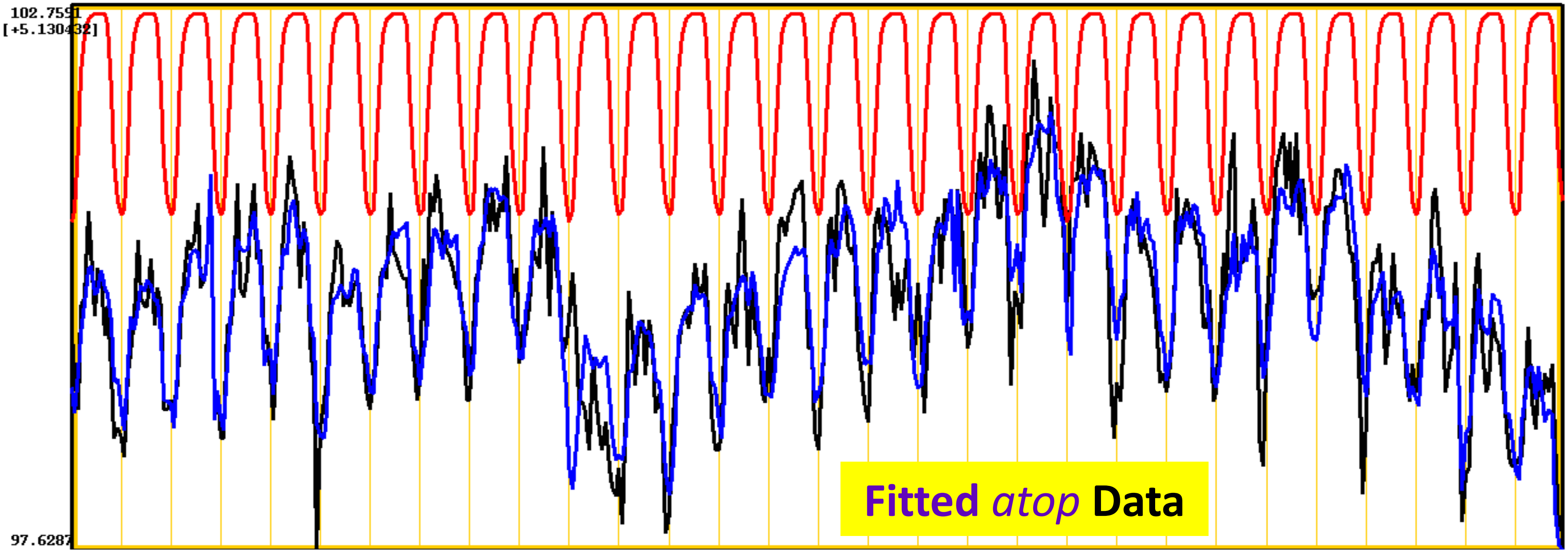
Model Quality Check

- First thing to do!
  - Most users in FMRI simply jump to specific effects of interest, their contrasts, and their significance. They simply don’t pay attention to overall model performance

- Approaches to judge your model
  - Modeled *vs* not modeled: **–fitts** and **–errts** outputs
    - Fitted curve = ‘baseline‘ + effects of interest
    - Residuals = noise = error = components we have no idea about (not included in model)

Model Quality Check

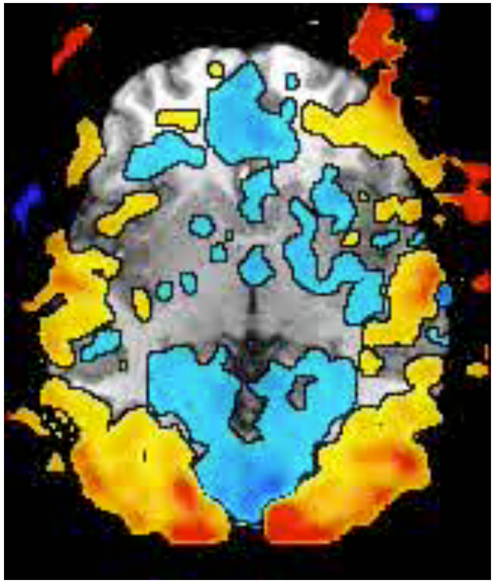
- First thing to do!
  - Most users in FMRI simply jump to specific effects of interest, their contrasts, and their significance. They simply don’t pay attention to overall model performance



Fitted *atop* Data

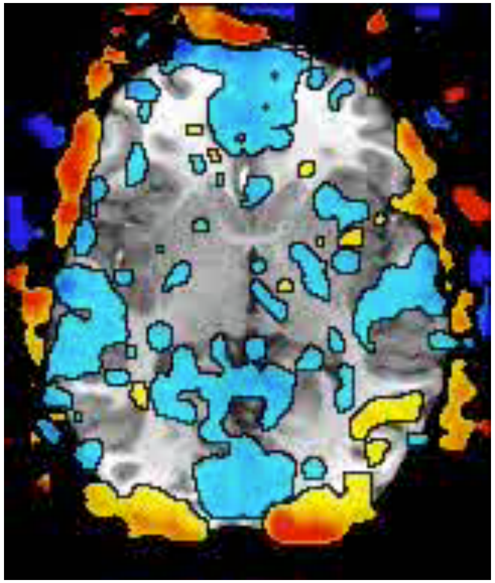
Statistical Testing

- Everything is about contrast (changes)!
- Effects (regression coefficients) of interest
  - β = effect relative to baseline condition
  - β_A = how much of regressor A had to be added to baseline model to fit data the best
  - *t*-statistic: statistical significance of a single β (visual stimulus)
    - Video: as *t* rises from 2 to 5
    - Colorized from β_{vis}, *not* from *t*



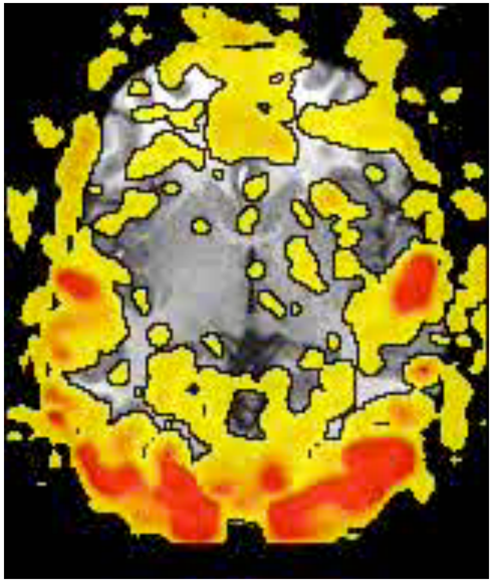
Statistical Testing

- Everything is about contrast (changes)!
- Effects (regression coefficients) of interest
- Pairwise comparisons (contrasts)
  - Conditions $\beta_{\text{vis}}-\beta_{\text{aud}}$ (*e.g.*, visual *vs* auditory)
    - How much of visual regressor was needed *minus* how much of auditory regressor
    - Positive=yellow/red ($\beta_{\text{aud}}<\beta_{\text{vis}}$)
    - Negative=blue ($\beta_{\text{aud}}>\beta_{\text{vis}}$)
  - *t*-statistic: statistical significance of this difference *vs* 0



Statistical Testing

- Everything is about contrast (changes)*!*
- Effects (regression coefficients) of interest
- Composite tests
  - F-statistic for composite (multiple part) null hypotheses
  - $\beta_{\text{vis}} \neq 0$ **and/or** $\beta_{\text{aud}} \neq 0$
  - Did *any* of the stimuli, or *any combination* of the stimuli, evoke a measurable response?
    - Video: as F rises from 4 to 34
    - Colorized from F (which is always > 0)



Assessing Fixed-Shape HRF Approach

- Used 99% of time: Why is it popular? 🤔
  - Assume brain responds with **same shape** across four levels: **subjects**, activated **regions**, stimulus **conditions/tasks**, **trials**
    - Difference in **magnitude** β in different conditions or different subjects (and its significance) is what we focus on
    - **But**: Strong assumption about **four** levels of shapes of BOLD response**?**
  - **Easy to handle and think about**
    - Just one value per effect/task 😀

Assessing Fixed-Shape HRF Approach

- Works relatively well, despite the caveats
  - Block design: shape usually not important due to accumulating effects of consecutive events
    - Really flat plateau? Same magnitude across blocks?
  - Event-related experiment: OK most of time
    - Linearity when responses overlap? Same effect across events?
- ***Not*** what you want if you
  - Care/worry about shape difference across subjects, across regions, across conditions, and across trials
  - More complex modeling can allow for such effects

AFNI

Didactics and Demonstrations

FMRI Task-Based Data Analysis at the Individual Level

Concepts 4 – FMRI Variable-Shape Models

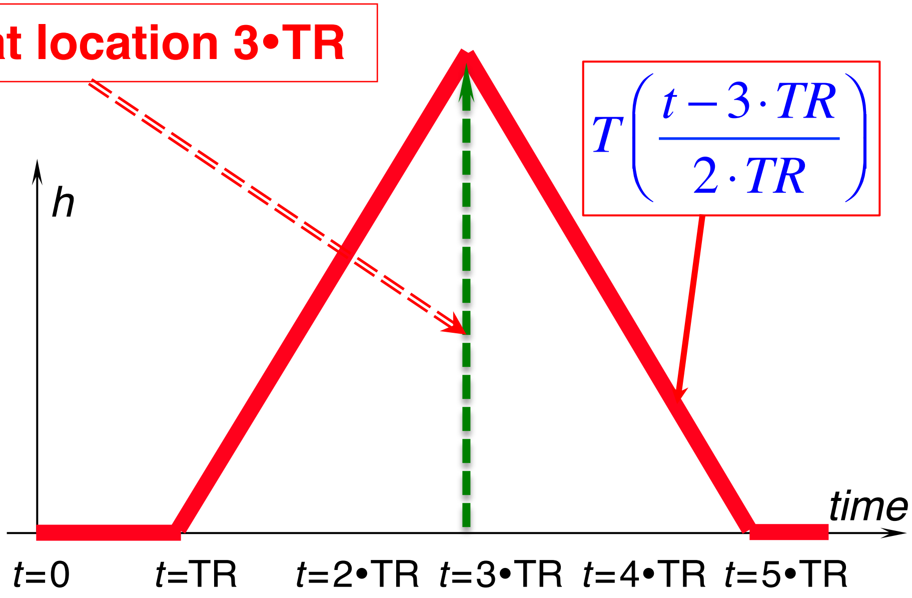
Alternative: No Constraint on HRF Shape

- TENT expansion of HRF (shape and magnitude)
  - Set multiple tents at various equally-spaced locations to cover the potential BOLD response period
    - Each TENT is a **basis function**
    - HRF is a sum of multiple basis functions, each with its own β
  - BOLD response measured by TENT heights (βs) at all locations
    - TENTs are also known as 'piecewise linear splines'

$$T(x) = \begin{cases} 1 - |x| & \text{for } -1 < x < 1 \\ 0 & \text{for } |x| > 1 \end{cases}$$

Formula for standardized TENT centered at *x*=0, width=±1

Cubic splines (CSPLIN) also in **AFNI**



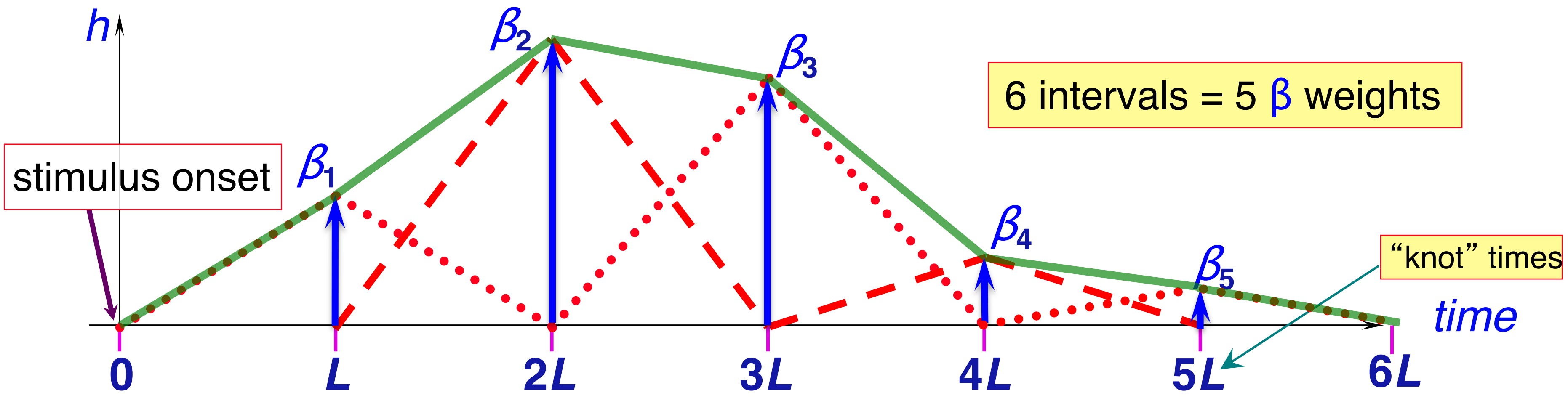
TENT at location 3•TR

$$T\left(\frac{t-3\cdot TR}{2\cdot TR}\right)$$

Sum of Tent Functions = Linear Interpolation

- 5 equally-spaced TENT functions = linear interpolation between "knots" using response model **TENTzero(*b*,*c*,*n*)** = **TENTzero(0,12,7)**

$$h(t)=\beta_1\cdot T\left(\frac{t-L}{L}\right)+\beta_2\cdot T\left(\frac{t-2\cdot L}{L}\right)+\cdots+\beta_5\cdot T\left(\frac{t-5\cdot L}{L}\right)$$



Sum of Tent Functions = Linear Interpolation

- TENT output parameters are easily interpreted as function values
  - o β_2 = response at time $t = 2L$ after stimulus onset
- Relationship of TENT spacing L and TR ($L \geq$ TR):
  - o *e.g.*, with TR=2s, usually choose L=2, or 4
- In **afni_proc.py** or **3dDeconvolve** using

  TENTzero(0, *D*, *n*)
  - o specify duration (D) of HRF and number (n) of knots
  - o $L = D/(n\text{-}1)$ with (n-2) full tents
  - o Each TENT overlaps ½ tent with two neighbors
  - o Example, D=12s, pick L=2s ➜ n=7 ➜ **TENTzero(0,12,7)**

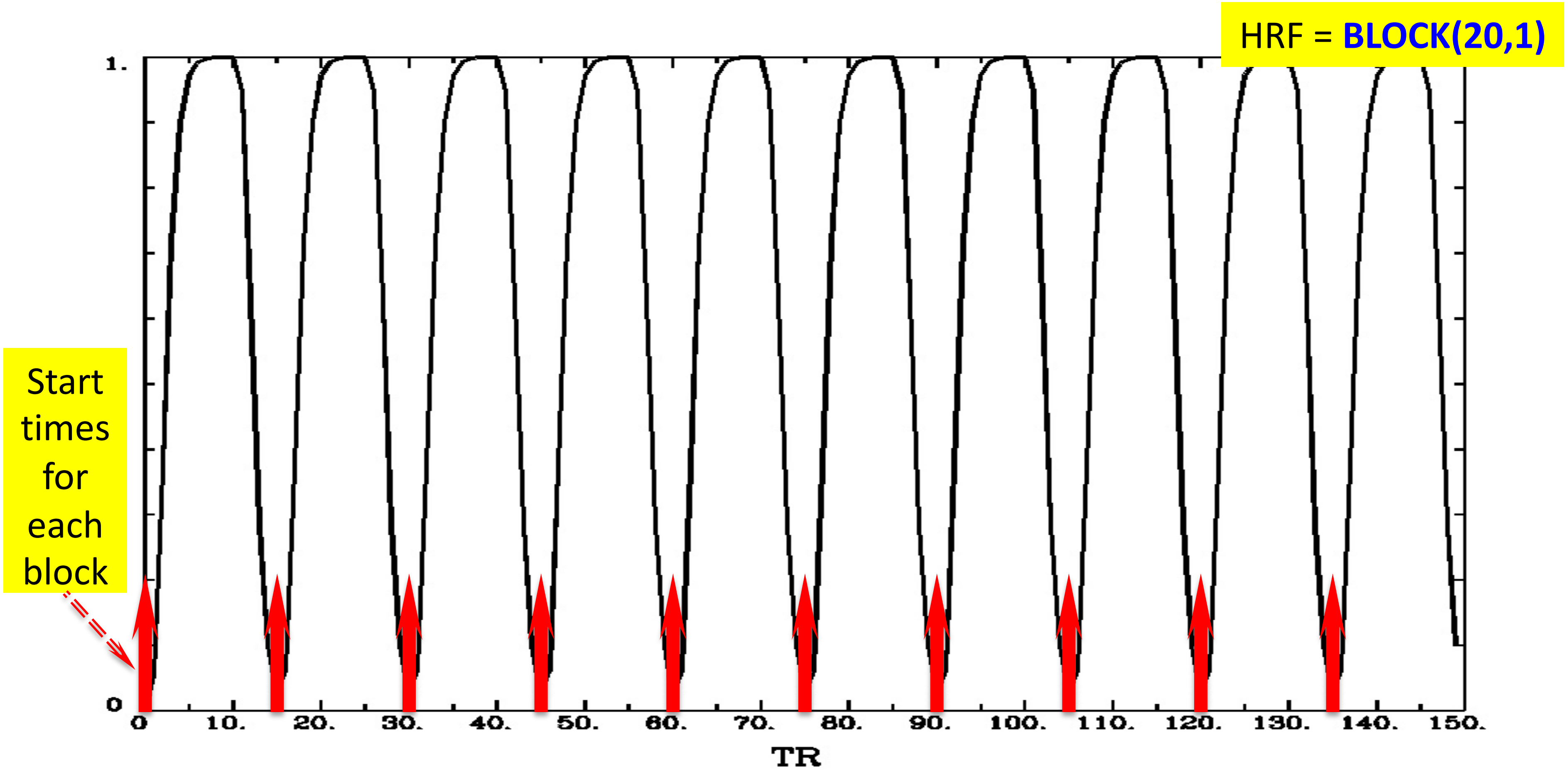
Basis Functions Create the HRF

- The HRF is repeated for all stimuli of the same type
- In the example, the HRF has 5 parameters (βs) to be estimated
- The βs determine the amplitude (percent signal change) ***and*** the shape of the HRF
- Each voxel in each subject gets a separate HRF shape now, not just a separate amplitude
  - If there are multiple types of tasks, each task gets a separate shape
- Stimulus times don't have to be on TR grid

Why **TENTzero**(b,c,n)?

- "**zero**" means that the HRF goes to 0 at the beginning and end of the time interval $b < t < c$
    - No response just after start or just before end
- "$\mathbf{b}$" means start of the response is "b" seconds after stimulus time – usually b=0
    - $b < 0$ is OK, to allow for pre-stimulus anticipation
- "$\mathbf{c}$" means end of the response is "c" seconds after stimulus time – must have $c > b$
- "$\mathbf{n}$" is the number of knots in the spline
    - $n-2$ is the number of βs (interior knots)

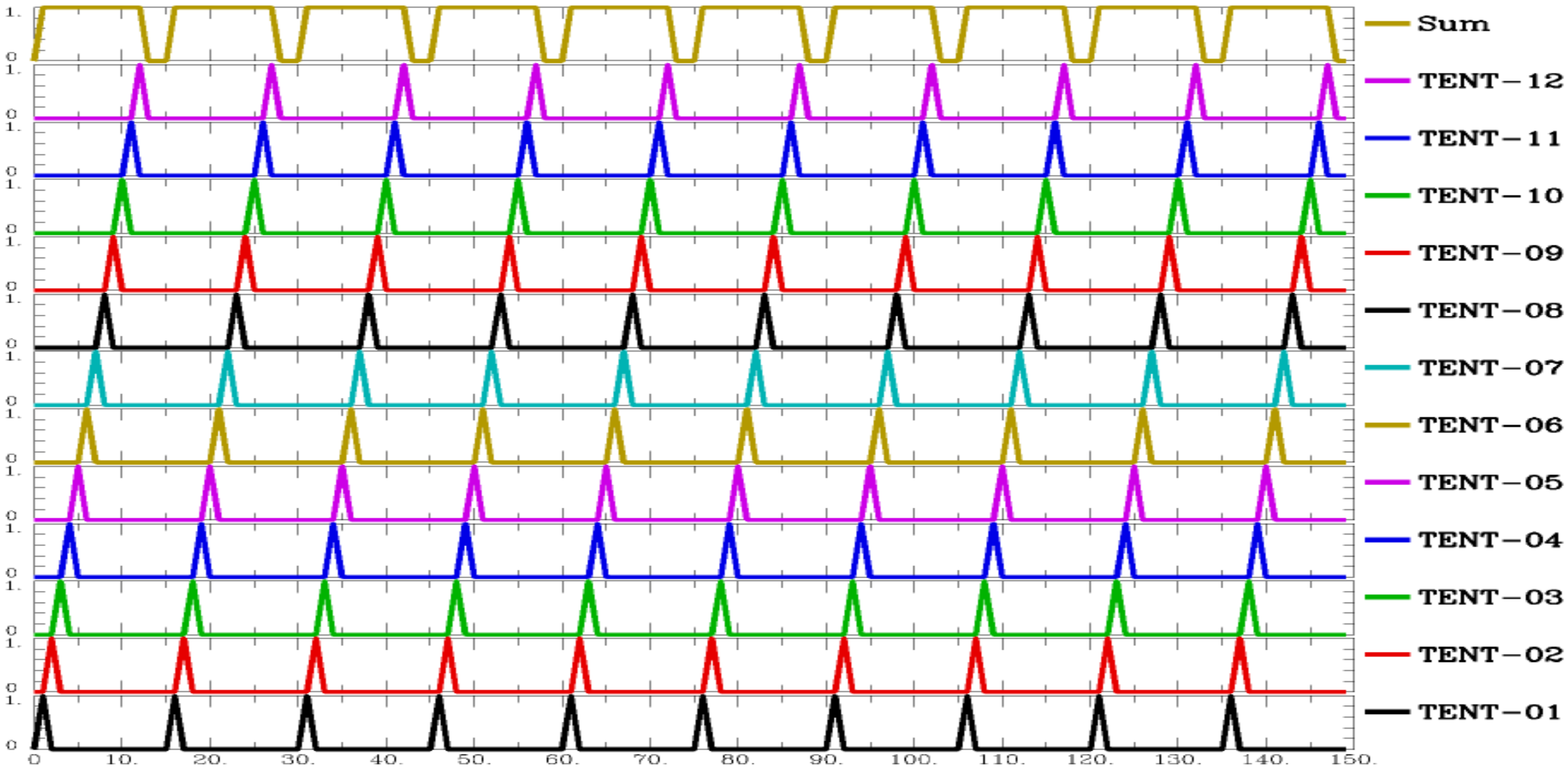
Fixed-Shape HRF for 20s Block Design



Blocks: 20 s on and 10 s off; TR=2 s; 150 time points

- From Talk 3 in this series

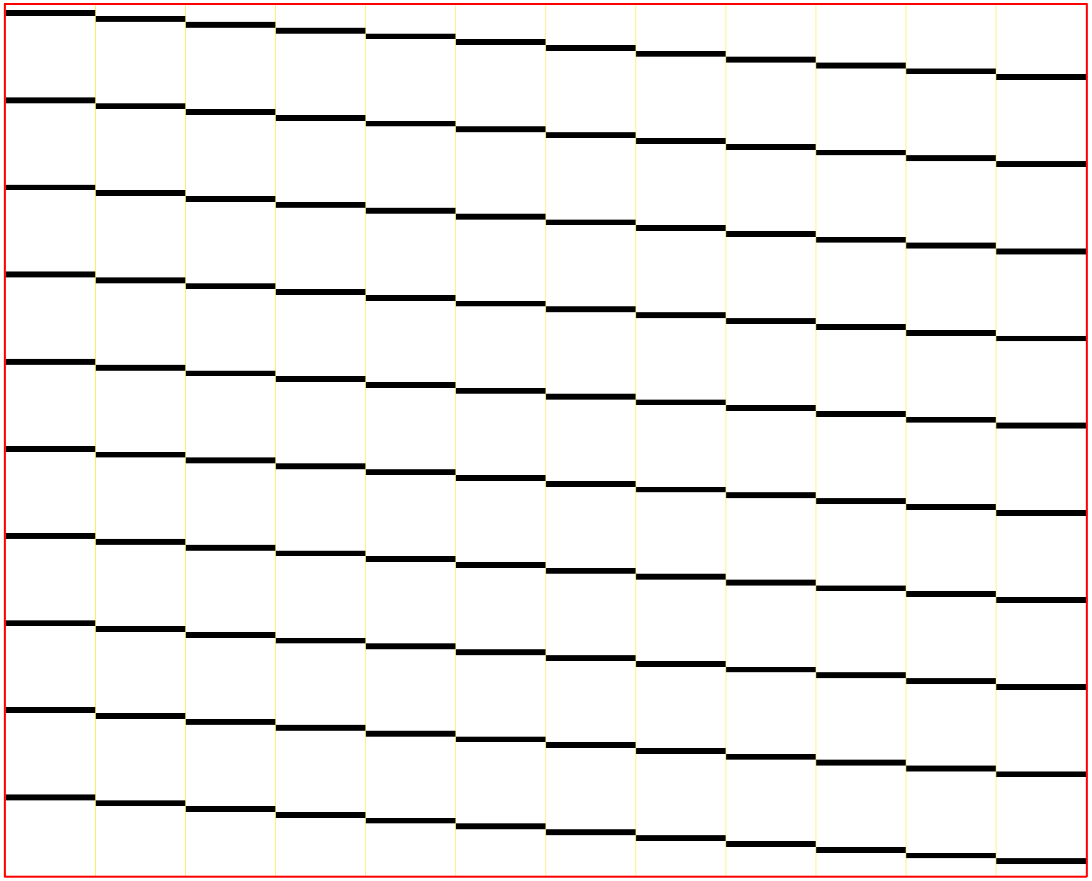
TENTzero HRF for 20s Block Design



HRF = **TENTzero(0,26,14)**

- 12 Basis Functions instead of 1
- Top sub-graph = sum of all TENTS
  - for comparison with BLOCK(20,1)

TENTzero HRF for 20s Block Design



HRF = **TENTzero(0,26,14)**

- 12 Basis Functions instead of 1

AFNI Script

```
3dDeconvolve -nodata 150 2.0     \
             -polort -1          \
             -x1D s4a.xmat.1D    \
             -xjpeg s4a.xmat.png \
             -num_stimts 1       \
             -stim_times 1       \
              '1D: 0 30 60 90 120 150 180 210 240 270' \
              'TENTzero(0,26,14)'
3dTstat -sum -prefix stdout: s4a.xmat.1D > s4a.sum.1D
1dplot -xaxis 0:150:15:2 -xlabel TR -png s4a.png      \
       -ynames TENT-01 TENT-02 TENT-03 TENT-04 TENT-05 \
               TENT-06 TENT-07 TENT-08 TENT-09 TENT-10 \
               TENT-11 TENT-12 Sum -                   \
       s4a.xmat.1D s4a.sum.1D
```

Script to produce plots on previous slides

s4a.TimeSeriesAnalysis.TentModel.csh

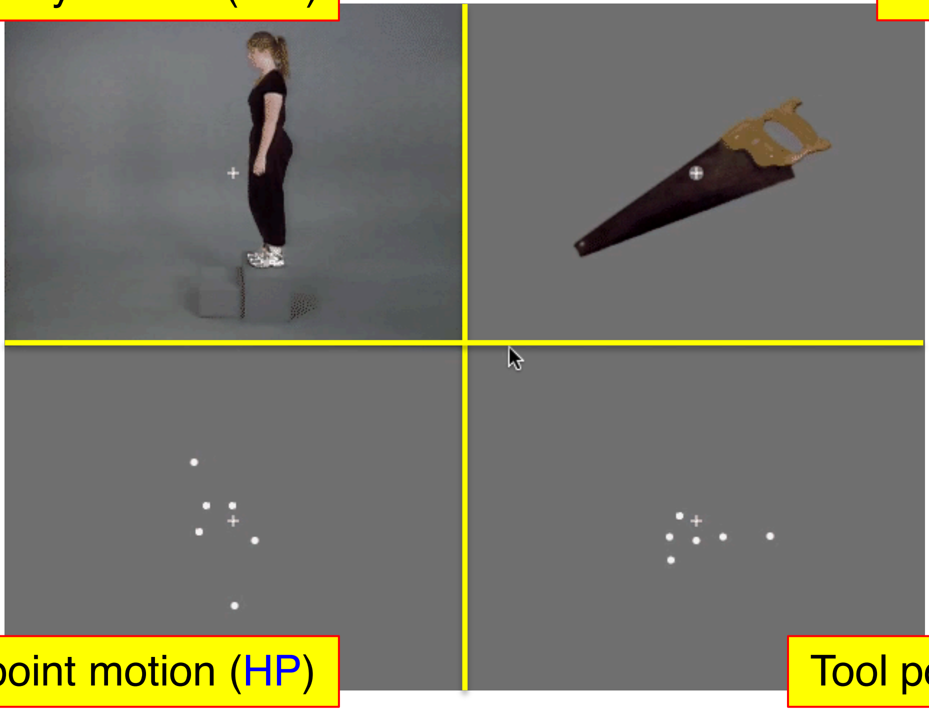
Modeling with TENTs – Real Example

- Event-related study (Beauchamp *et al.*, J Cogn Neurosci **15**:991-1001)
  - 10 runs, 136 time points per run, TR=2 s
  - Two factors
    - Object type: **human** *vs* **tool**
    - Object form in videos: **real image** *vs* **points**
  - 4 types (2x2 design) of stimuli (2s videos)
    - Tools moving (*e.g.*, hammer pounding) - **ToolMovie**
    - People moving (*e.g.*, walking, sitting) - **HumanMovie**
    - Points outlining tools moving - **ToolPoint**
    - Points outlining people moving - **HumanPoint**
  - Goal: find brain area that distinguishes natural motions (**HumanMovie** and **HumanPoint**) from simpler rigid motions (**ToolMovie** and **ToolPoint**)

- **Experiment: 2 x 2 design**

Human body motion (HM)

Tool motion (TM)



From Figure 1 *Beauchamp et al. 2003*

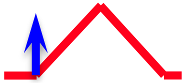
Actual videos do not loop

Human point motion (HP)

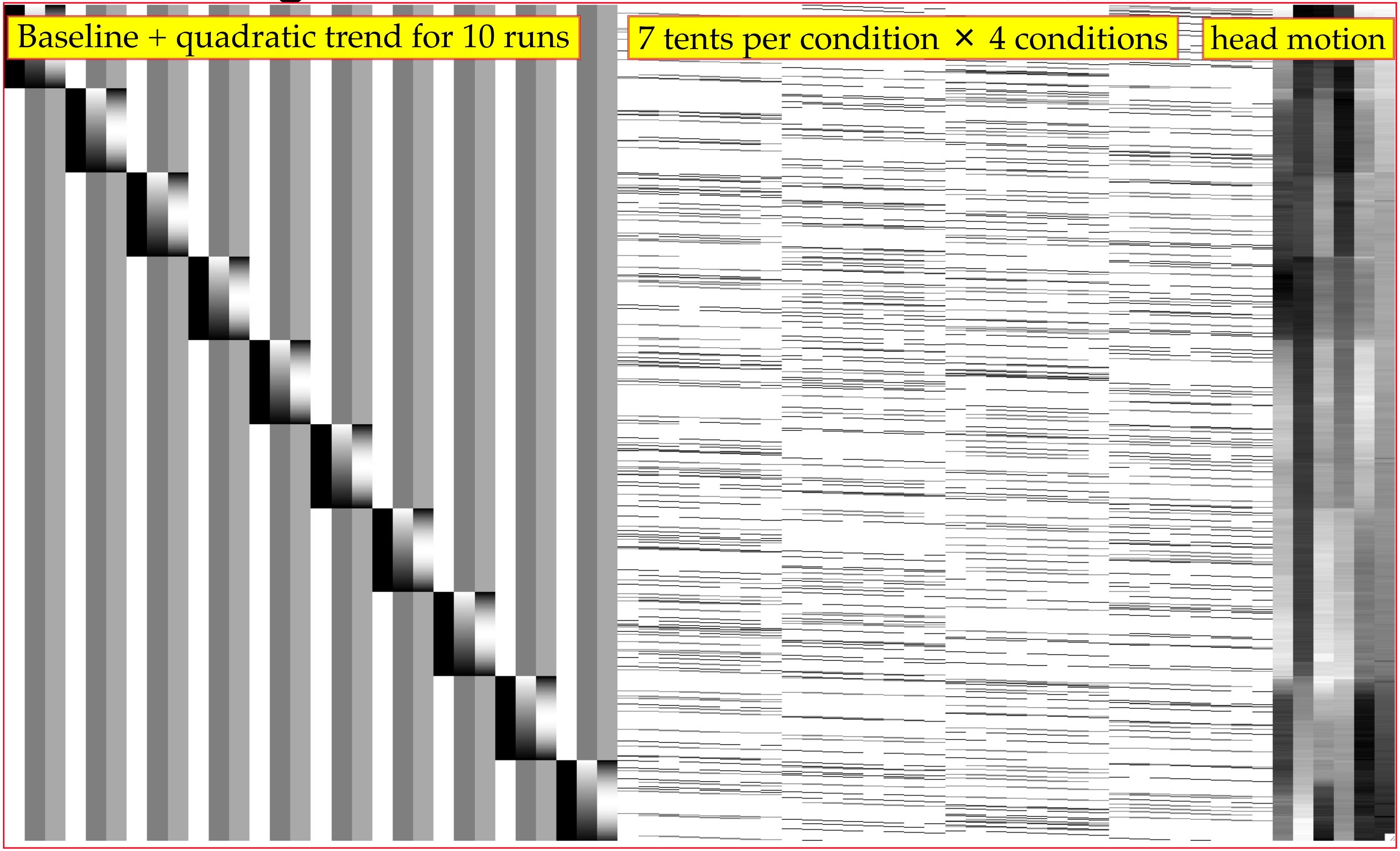
Tool point motion (TP)

- Which areas differentially activated by any of stimuli (main effect)?
  - o **Point** motion *vs* **natural** motion? (image type: top 2 vs bottom 2)
  - o **Human**-like *vs* **tool**-like motion? (motion type: left 2 vs right 2)
- Interaction effects?
  - o **Point**: **human**-like *vs* **tool**-like? **Natural**: **human**-like *vs* **tool**-like?
  - o **Human**: **point** *vs* **natural**? **Tool**: **point** *vs* **natural**?

No Constraint on HRF Shape = **Deconvolution**

- Shape estimation: Deconvolution via regression
  - Known: **stimulus timing**
  - Unknown: **HRF** = BOLD shape/size
  - **HRF** in each voxel estimated as linear combination (sum) of multiple basis functions: TENTs (or CSPLINs) – *rather than just one function*
  - Each TENT ➜ one regressor column
    - Copy of TENT shape  starting at stimulus times plus its assigned "knot" offset in time
  - Deconvolution ➜ **HRF** = set of β via regression

Design Matrix with `TENTzero(0,16,9)`

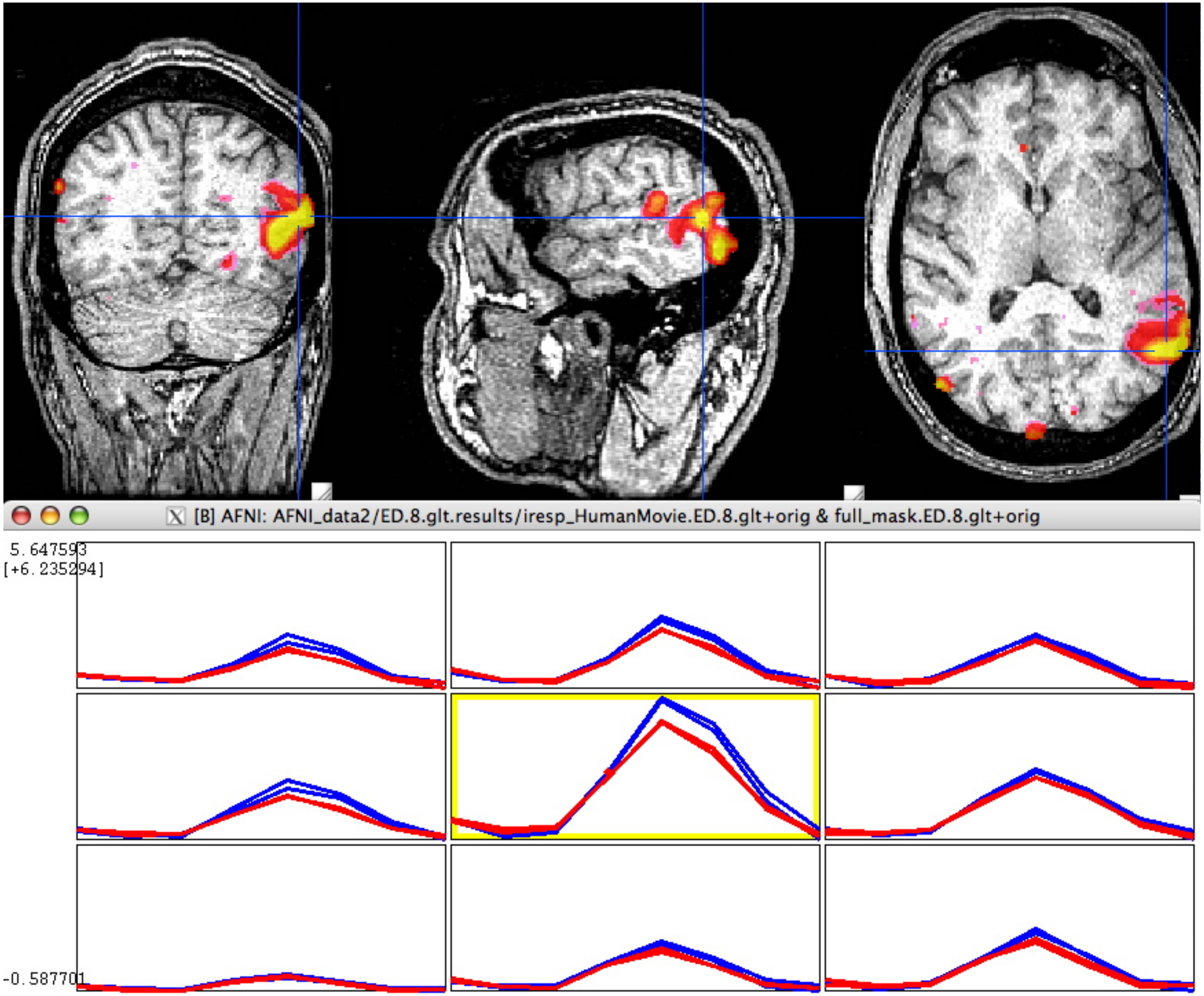


Baseline + quadratic trend for 10 runs

7 tents per condition × 4 conditions

head motion

Results: Humans vs. Tools


[B] AFNI: AFNI_data2/ED.8.glt.results/iresp_HumanMovie.ED.8.glt+orig & full_mask.ED.8.glt+orig
5.647593
[+6.235294]
-0.587701

- Color overlay: **Human vs Tool ($\beta_{HM}+\beta_{HP}-\beta_{TM}-\beta_{TP}$)**

- **Blue HRF** (upper) : Human

- **Red HRF** (lower) : Tool

No Constraint on HRF Shape: **Pros** + Cons

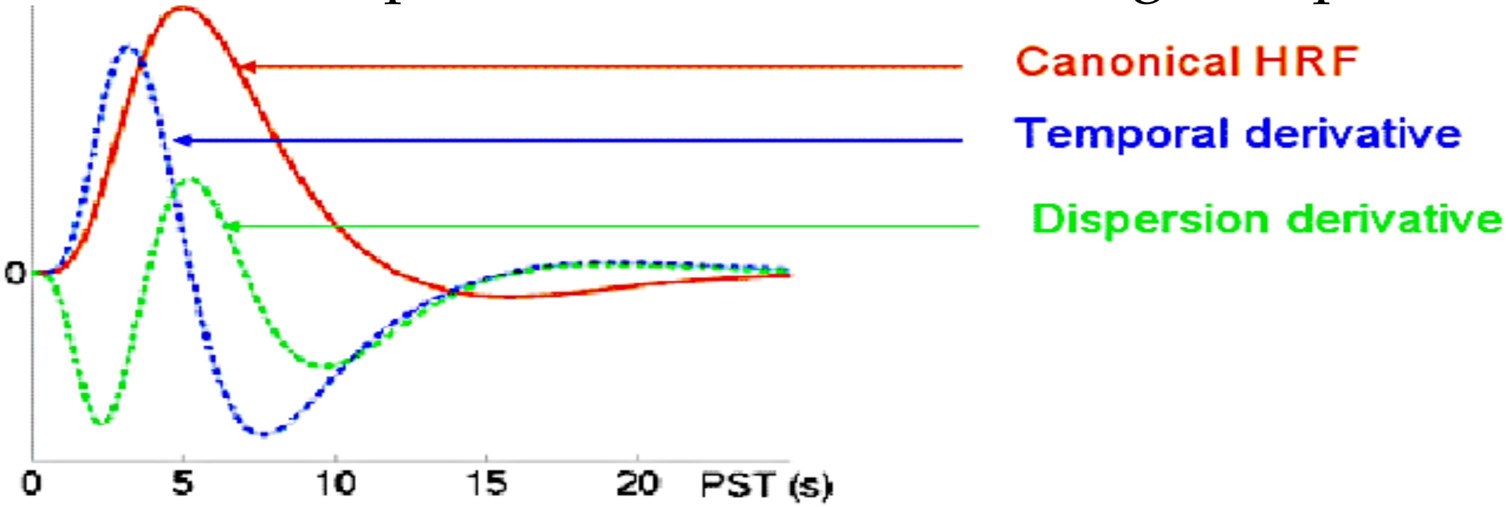
- What is the approach good at?
  - Usually: event-related designs; can be used for BLOCK
    - Multiple basis functions for blocks: can find within-block attenuation with time
  - Likely to have more accurate estimate on HDR shape across
    - Subject (*e.g.*, young *vs* old)
    - conditions/tasks
    - brain regions
  - Usually get better model fit (goal in sample experiment)
  - Usually statistically more powerful on test significance
    - Unless you overfit the data, with too many βs

No Constraint on HRF Shape: Pros + **Cons**

- Why is the approach not popular?
  - Difficult to combine individual results at group level
    - Multiple parameters (βs) per task condition, instead of just one β per subect
    - **But**: see the **AFNI** program **3dMVM**
  - More regressors than simpler alternatives
    - Degrees of Freedom per subject (data points–regressors)
  - Risk of highly correlated regressors: Multicollinearity
    - May need to reduce the number of basis functions
    - Probably need to randomize stimulus timing
  - Over-fitting: picking up something (head motion) unrelated to HDR

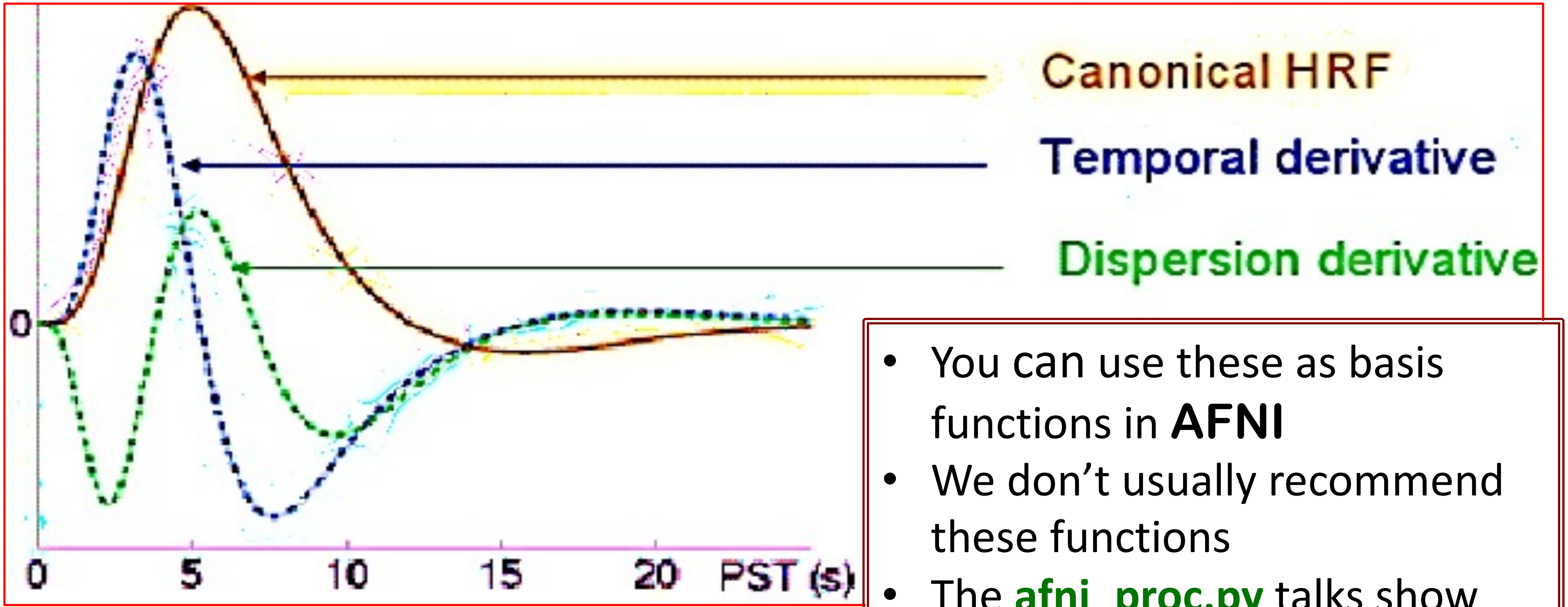
Intermediate Approach: SPMG1/2/3

- Use just a few (2-3) basis functions
  - Constrain HRF shape with a principal basis function
    - SPMG1 (similar to GAM in **AFNI**, with undershoot added)
  - 2 or 3 basis functions: parsimonious, economical
    - $\beta_1 \cdot \text{SPMG1} + \beta_2 \cdot \text{SPMG2} + \beta_3 \cdot \text{SPMG3}$
    - SPMG2: time derivative ➜ changes in peak delay
    - SPMG3: dispersion derivative ➜ changes in peak width



SPMG1/2/3

[Ready for their closeup, Mr. DeMille]



- You can use these as basis functions in **AFNI**
- We don't usually recommend these functions
- The **afni_proc.py** talks show the details of how to select basis functions for the HRF model

Group Analysis with TENTS

- Use multiple βs from each subject in a group analysis?
  - What to do depends on your goal in the study
- **Goal**: find activation magnitude differences
  - Add up TENT βs in each voxel to get "area under the response curve"
  - Carry that sum as a single scalar to the group level as usual (*e.g.*, **3dttest++** or **3dLME**)
- **Goal**: be sensitive to shape differences
  - Use **3dMVM** program (**M**ulti**V**ariate **M**odeling), which allows for multiple βs in each condition
- More on this subject in the Group Analysis Talks

AFNI

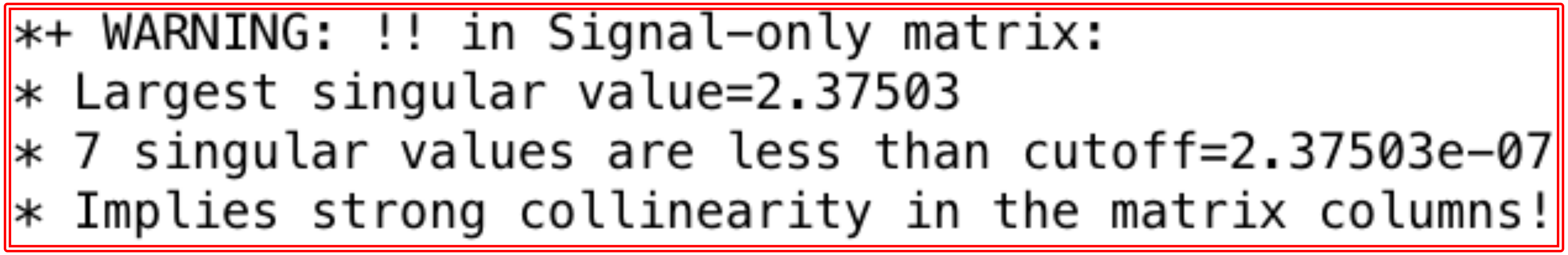
Didactics and Demonstrations

FMRI Task-Based Data Analysis at the Individual Level

Concepts 5 – Modeling Issues

Multicollinearity

- Voxel-wise regression model: $y = \mathbf{X}\beta+\varepsilon$
  - ○ Regressors in matrix $\mathbf{X} = [1, t, t^2, x_1, x_2, \ldots, x_k, \ldots]$
- Multicollinearity problem
  - ○ Two (or more) regressors highly correlated
  - ○ Difficult or impossible to distinguish effects among these regressors (*i.e.*, get reliable β estimates)
    - ▪ Sample message from **3dDeconvolve** – indicates that regressors of interest (Signal) are in trouble



```
*+ WARNING: !! in Signal-only matrix:
* Largest singular value=2.37503
* 7 singular values are less than cutoff=2.37503e-07
* Implies strong collinearity in the matrix columns!
```

Multicollinearity

- Multicollearity scenarios
  - Exact collinearity: $\boldsymbol{x_i} = \mathbf{c} \cdot \boldsymbol{x_j}$ = **model specification error**
    - *e.g.*, 2 identical regressors (mistake in stimulus timing)
  - Exact multicollinearity: linear dependence among multiple regressors = **faulty design** (rare)
  - High degree of correlation (+ or -) among regressors = **design problem** (*e.g.*, cue + short video watching)
  - Too many basis functions in response model
- Matrix diagnostic tools:
  - **ExamineXmat.R**, **timing_tool.py**, **xmat_tool.py**
  - Better to prototype analysis and find problems *before* acquiring hard-to-analyze datasets***!***

Serial Correlation in Residuals

- Temporal correlations – in residuals/noise – not "white"
  - Physiological effects (breathing, heartbeat, motion)
  - βs from OLSQ regression are unbiased
  - But statistics (t, F) tend to be inflated – assuming no correlations in time – this is a modeling error about *noise*
  - Little impact on group analysis – ***if*** only using βs (BOLD signal magnitude estimates) from subjects
  - Will affect group analysis *if* also using β's reliability, as in **AFNI**'s **3dMEMA** program (where βs and ts are both used)
- **AFNI** approach – program **3dREMLfit** – see section 7
  - Voxel-wise correction for inter-TR correlation, using GLSQ (Generalized Least SQuares) regression – more on this later

Dealing with Multiple Runs per Subject

1. Analyze each run separately: **AFNI**, FSL
   - Have to have enough task repetitions in each run
   - Can test cross-run difference (trend, habituation) at group level
   - Usually need to summarize multiple β's (one from each run for each task/condition type) before group analysis
   - Unless using **AFNI**'s **3dMVM** program
     - Which allows multiple values per subject per task

Dealing with Multiple Runs per Subject

2. Concatenate runs but analyze with separate regressors across runs for each condition type: **AFNI**, SPM
   - Can then test cross-run difference (trend, habituation, etc.) at both individual and group levels
   - Usually still need to summarize multiple β's before group analysis

Dealing with Multiple Runs per Subject

3. Concatenate runs but analyze with a single regressor (for each condition type) across runs: default in **AFNI**
   - Assumes no response attenuation across runs
   - That is, a task event in run #1 is treated identically to a task event in run #7
   - Allowing for cross-block (or cross-event) attenuation
     - Method: **IM** or **AM** regression models
     - Described in later talk/slides

Percent Signal Change

- Why convert/scale to make β = % signal change?
  - Comparing across subjects – uniform measurements
  - MRI and BOLD data values don't have any useful physical/physiological meanings or units
  - Baseline is different across subjects
    - And possibly scaling of raw data values (from scanner hardware/software)
  - It is relative changes that can be compared across subjects
  - BOLD effect is multiplicative on overall voxel signal

Percent Signal Change

- **AFNI** approach
  - Pre-processing: data scaled so **voxel-wise** mean = 100
    - β = % signal change relative to **mean**, not to **base**line
    - Difference is tiny: less than 5% (BOLD effect small)
- Alternatives:
  - Global mean scaling for whole brain drift
    - Scale so mean of **each** EPI volume is the same
  - Grand mean scaling for cross-subject comparison: not %
    - Scale each subject so mean over **all** volumes is a constant
  - These can be performed in **AFNI** if *truly* desired
    - Not our recommendation

Lackluster Performance in Modeling

- **All models are wrong, but some are useful** (GEP Box)
- Regressors: we use an idealized response model
  - We find what we're looking for
  - We may miss something when we do not look for it
- Lots of variability across trials (responses and noise)
  - **A**mplitude **M**odulation if behavioral data are available
  - Model each trial separately (**I**ndividual **M**odulation)
- Linearity assumptions
  - Data = baseline+drift+response1+response2+…+noise
  - When a trial is repeated, response is assumed same
  - Response for a block = linearity (no attenuation)

AFNI

Didactics and Demonstrations

FMRI Task-Based Data Analysis at the Individual Level

Concepts 6 – Advanced Regression

More Complicated Regression Models

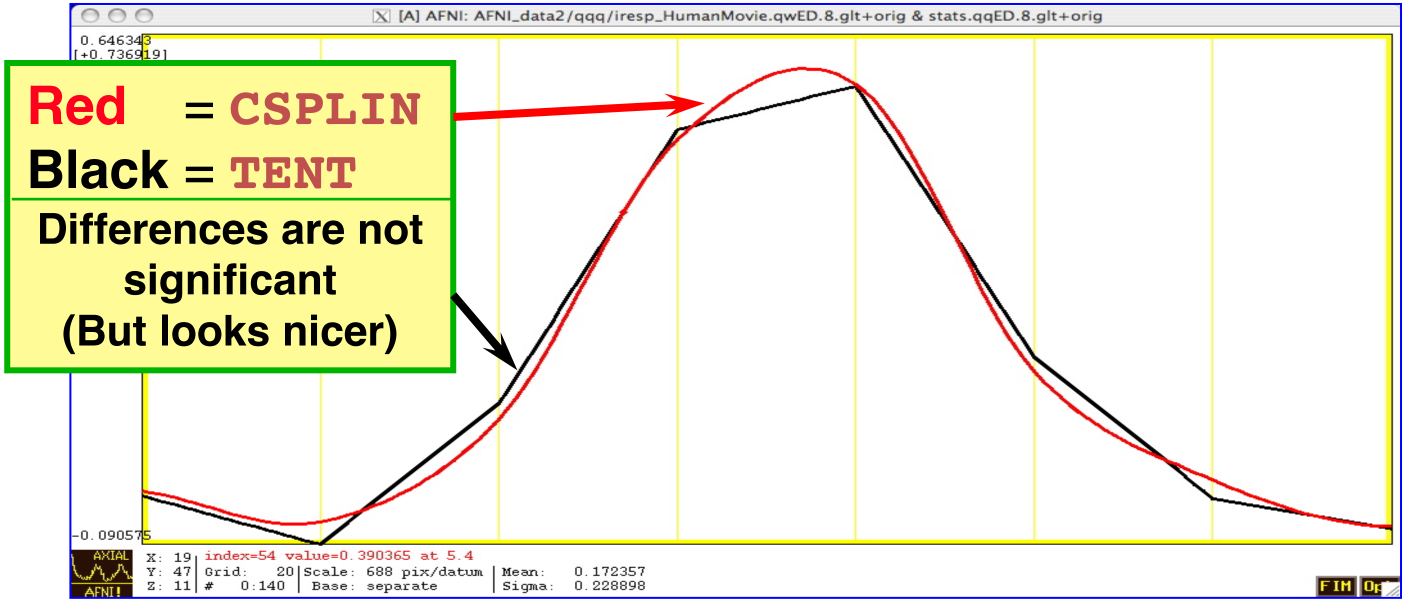
- Regression models in **AFNI** are (usually) set up via **afni_proc.py**
  - Which in turn uses program **3dDeconvolve** to construct the regression matrix **X**
  - And solves the linear system in **3dDeconvolve** or **3dREMLfit** (task) or **3dTproject** (*e.g.*, resting state)
- This set of slides covers more complicated linear model setups
  - Used for special situations
    - But aren't *all* research situations special?

All Zero Regressors

- All-zero time series regressors *are* allowed
  - Via **3dDeconvolve** option **-allzero_OK**
  - Will get zero *β* weight and zero *t* in the solution
  - **Example**: task where subject makes a forced choice for each stimulus (e.g., male or female face?)
    - Analyze correct and incorrect trials as separate cases
    - What if some subject makes no mistakes? Hmmm …
      - Can keep all-zero regressor (**-stim_times** = *)
      - Input files and output datasets for error-making and perfect subjects will be organized same way
      - Makes it simpler to setup group analyses when all subject-level results are consistent

Other Basis Functions

- **3dDeconvolve -stim_times** has other basis function options for HRF models besides **BLOCK** and **TENT**
  - ○ **CSPLIN** = **cubic** spline, instead of **TENT** = **linear** spline
    - ▪ Same parameters: **(start,stop,number of regressors)**
    - ▪ A "drop in" replacement for **TENT**



```
[A] AFNI: AFNI_data2/qqq/iresp_HumanMovie.qwED.8.glt+orig & stats.qqED.8.glt+orig
```

```
X: 19| index=54 value=0.390365 at 5.4
Y: 47| Grid:    20|Scale: 688 pix/datum | Mean:   0.172357
Z: 11| #   0:140  | Base: separate      | Sigma:  0.228898
```

Red = **CSPLIN**

Black = **TENT**

Differences are not significant

(But looks nicer)

- ○ **TENTzero** & **CSPLINzero** = force start & end of HRF = 0

All Basis Functions – Single β

- **BLOCK** = discussed previously
- **GAM** = for short responses (discussed previously)
- **TWOGAM** = for compatibility with BrainVoyager
- **SPMG1** = discussed previously
- **WAV** = very old **AFNI** waveform [don't use]
- **MION** = for use with MION contrast agent

- **BLOCK** and **MION** always have duration parameter

- **GAM**, **TWOGAM**, **SPMGx**, and **WAV** have *optional* duration parameter

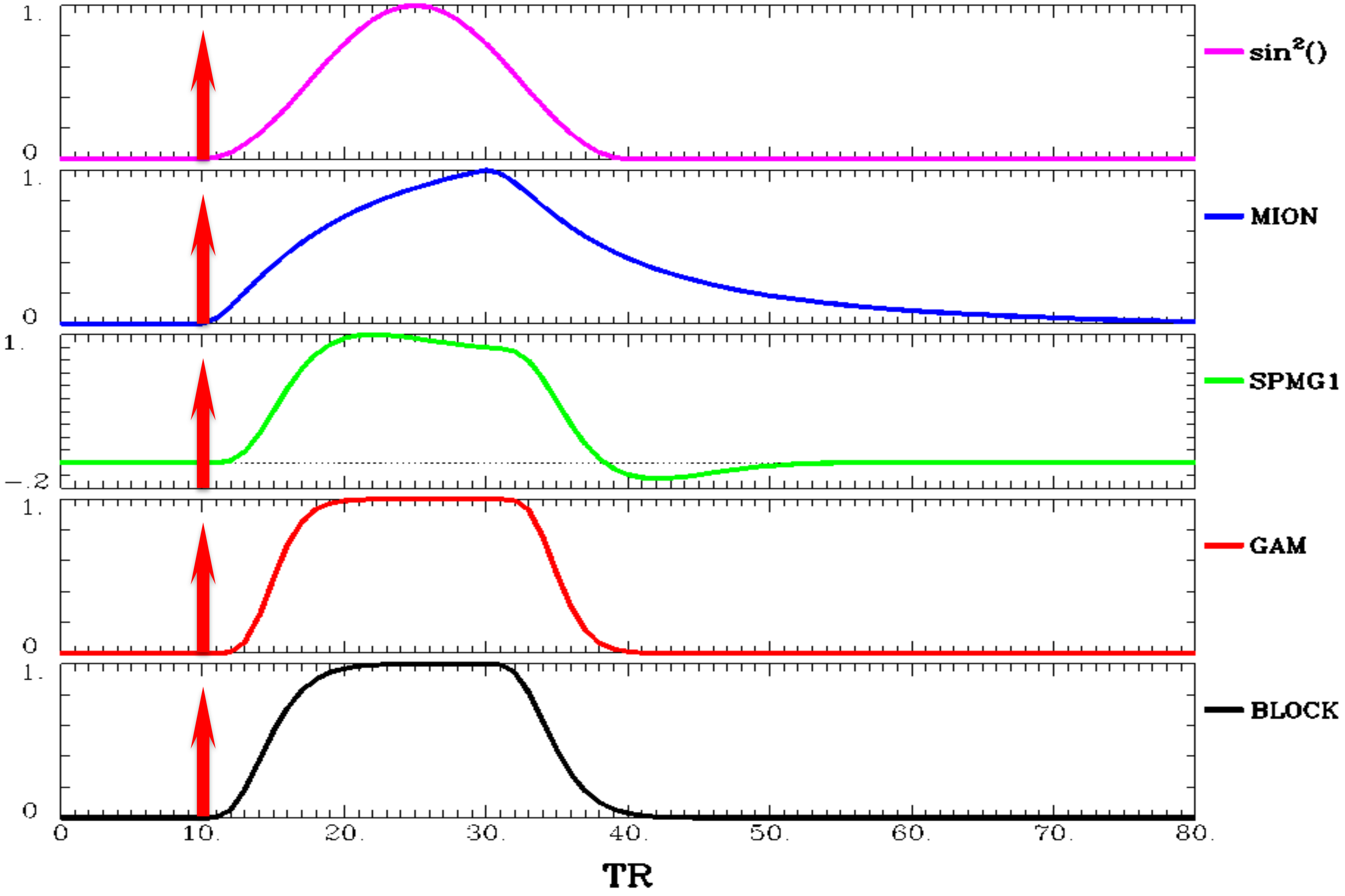
- For details, see output of **3dDeconvolve -help**

All Basis Functions – Multiple β

- **TENT** = discussed previously
  - and **CSPLIN** and **TENTzero** and **CSPLINzero**
- **SPMG2** = discussed previously (and **SPMG3)**
  - Unlike other multiple β functions, **SPMGx** can take an optional duration parameter – to convolve its basis functions with a "square wave" in time
  - All other multiple β functions just use a duration over which the basis functions are defined
- **POLY** = Legendre polynomial expansion
- **SIN** = sine series expansion
- **EXPR** = arbitrary set of formulas

- For details, see output of **3dDeconvolve -help**

Some Basis Functions – Single β



AFNI Script

```
3dDeconvolve -nodata 81 1.0   \
             -polort -1       \
             -x1D s6a.xmat.1D \
             -x1D_stop        \
             -num_stimts 5    \
             -stim_times 1 '1D: 10' 'BLOCK(20,1)'      \
             -stim_times 2 '1D: 10' 'GAM(8.6,.547,20)' \
             -stim_times 3 '1D: 10' 'SPMG1(20)'        \
             -stim_times 4 '1D: 10' 'MION(20)'         \
             -stim_times 5 '1D: 10’                    \
                           'EXPR(0,30) sin(PI*t/30)^2'
1dplot -sepscl -xaxis 0:80:8:10 -xlabel TR            \
       -ynames 'BLOCK' 'GAM' 'SPMG1' 'MION' 'sin^2()' \
       -png s6a.png s6a.xmat.1D
```

Script to produce plot on previous slide

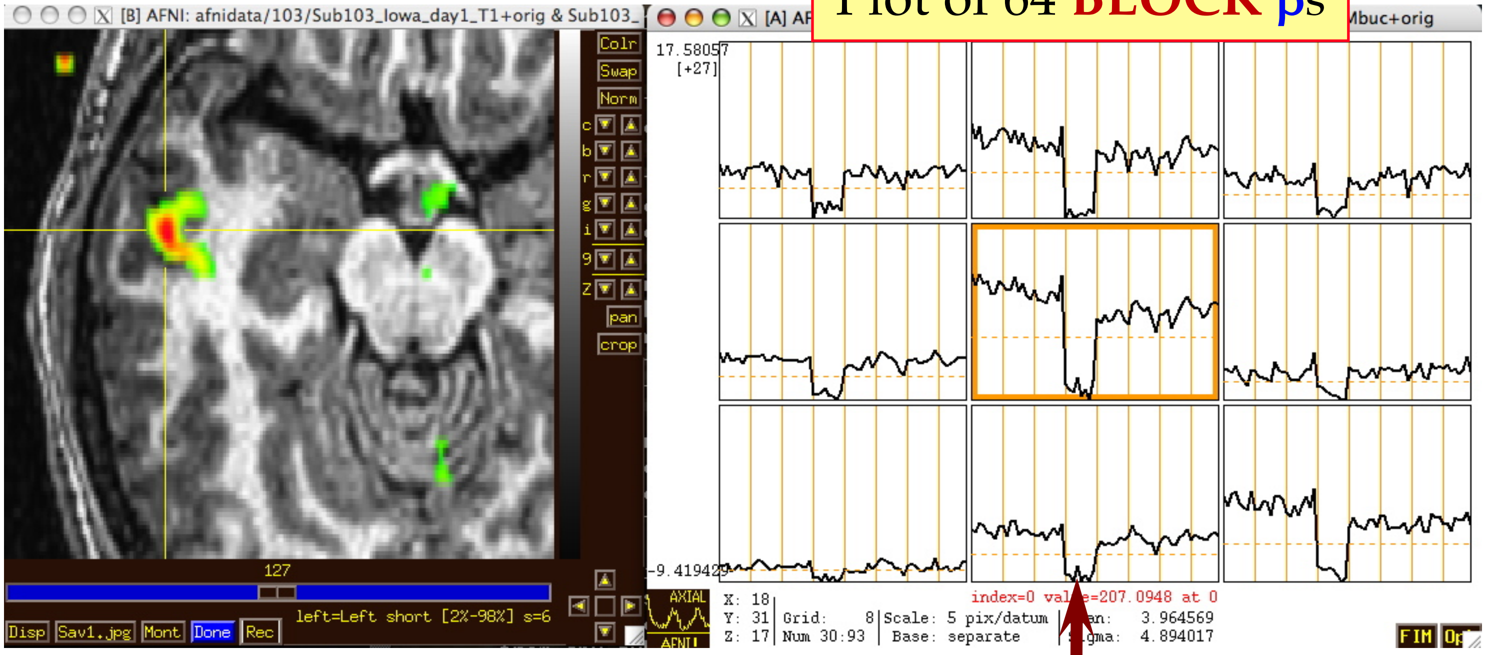
s6a.TimeSeriesAnalysis.MultiModels.csh

IM Regression - 1

- **IM** = **I**ndividual **M**odulation
  - Compute *separate* amplitude of response (β) for each stimulus block/event in each stimulus class
    - Instead of computing average amplitude of responses to multiple stimuli in the same class
    - *Separate* regression column for each stimulus time
  - βs for each separate block/event will be very noisy
    - Can't use individual activation maps for much
    - Must pool computed βs in some further statistical analysis (individual and/or group)
      - t-test via **3dttest++**? inter-voxel correlations in the βs? Correlate βs with something else?

IM Regression - 2

- First application of IM was checking some data we received from another institution
- Experiment: 64 blocks of sensorimotor task (8 runs each with 8 blocks)



[B] AFNI: afnidata/103/Sub103_Iowa_day1_T1+orig & Sub103_

[A] AF

buc+orig

17.58057 [+27]

-9.419429

index=0 val e=207.0948 at 0

```
X: 18 |
Y: 31 | Grid:      8 | Scale: 5 pix/datum |   an:    3.964569
Z: 17 | Num 30:93    |  Base: separate    |  gma:    4.894017
```

left=Left short [2%-98%] s=6

Plot of 64 **BLOCK** βs

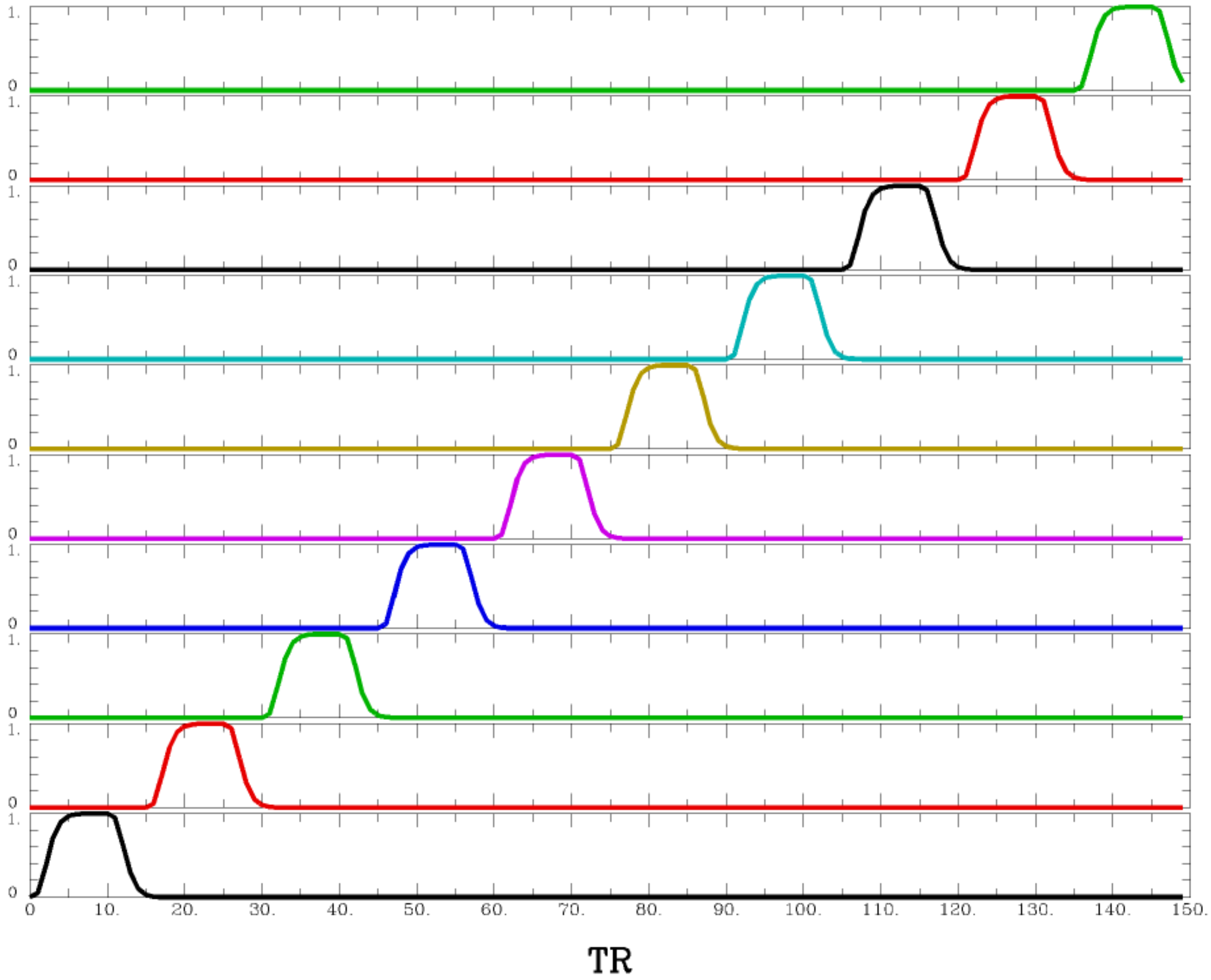
sign reversal in run #4 = stimulus timing error!

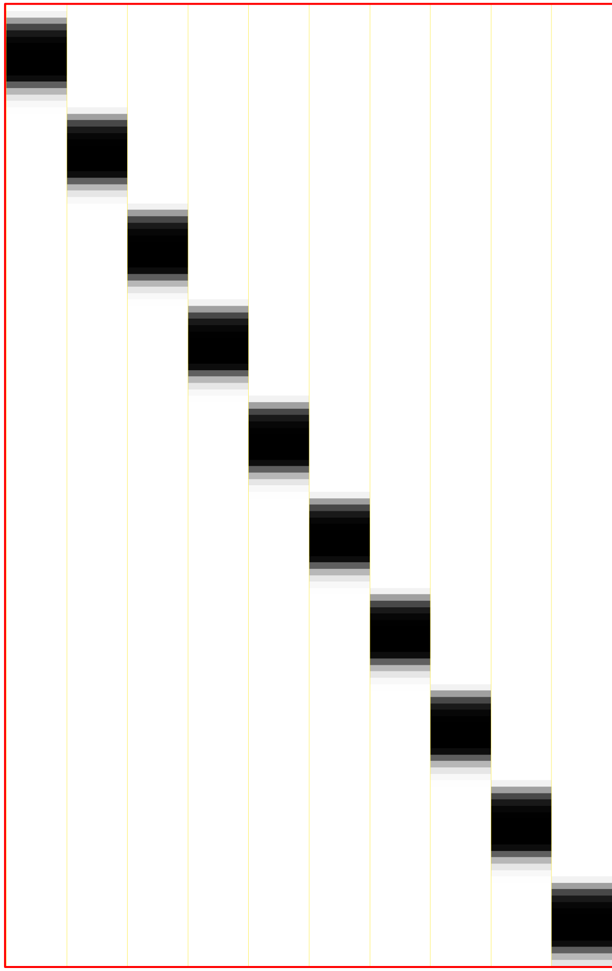
IM Regression - 3

- **IM** works naturally with **BLOCK**s, which only have 1 amplitude parameter β per given stimulus start time
  - o More difficult conceptually to use with multiple β basis functions, as each event gets not just different amplitude but different *shape*
- Work in progress now (spring 2020)
  - – Combine multiple βs using a linear mixed effects (LME) model to include mean and variance of subject-level response within a single task class (*e.g.*, "faces")
  - – Randomness from measurement fluctuations ("noise")
  - – *and* Randomness from subject response fluctuations

IM Regression - 4

Regressors for IM analysis





AFNI Script

```
3dDeconvolve -nodata 150 2.0       \
             -polort -1            \
             -x1D s6b.xmat.1D      \
             -x1D_stop             \
             -xjpeg s6b.xmat.png   \
             -num_stimts 1         \
             -stim_times_IM 1      \
              '1D: 0 30 60 90 120 150 180 210 240 270' \
              'BLOCK(20,1)'
1dplot -xaxis 0:150:15:2 -xlabel TR -png s6b.png s6b.xmat.1D
```

Script to produce plot on previous slide

s6b.TimeSeriesAnalysis.IMModel.csh

AM Regression - 1

- **AM** = **A**mplitude **M**odulated (or Modulation)
  - Have some extra data measured about response to each individual stimulus, and *maybe* BOLD response is modulated by this
  - Reaction time; Galvanic skin response; Pain level perception; Emotional valence
- Want to see if some brain regions vary proportionally to this **ABI** (**A**uxiliary **B**ehavioral **I**nformation – my personal acronym, not a standard!)

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 assume that the change in FMRI/BOLD signal as the ABI changes is *linearly proportional* to the changes in the ABI values
    - If needed, transform ABI values (e.g., logarithm)
- Need to make 2 separate regressors
  - One to find mean FMRI response (usual analysis)
  - One to find the variations in the FMRI response as the ABI data varies

AM Regression - 3

- The second regressor is

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

  - Where a_k =value of k th ABI value, and $\bar{a}$ is mean ABI value
    - Set UNIX environment **AFNI_3Deconvolve_rawAM2** to **YES** so mean of $\{a_k\}$ is not removed – for advanced users
- β for first regressor is standard activation map
- Statistics and β for second regressor make activation map of voxels 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 which *are* ABI-sensitive

AM Regression - 4

- AM2 regression: **-regress_stim_types AM2**
- Use is very similar to standard **times**
  - But the timing file has 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
    ```

    - One line per imaging run in the analysis
    - Each stimulus time entry is of form **TIME*ABI**
    - If a run has *no* stimuli of this type, put in a single *****
  - Such files can be created from 2 standard ASCII (text) .1D files using the **1dMarry** program
    - The **-divorce** option can be used to split them up

AM Regression - 5

- **3dDeconvolve** (the matrix creator) automatically creates the two regressors:
  - unmodulated and amplitude modulated
  - Use **-fout** option to get statistics for activation of 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 X** matrix columns to see each regressor

AM Regression - 6

- If you want, **AM1** regression is also available:
  - It only builds the regressor proportional to ABI data directly, with no ABI parameter mean removed:

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

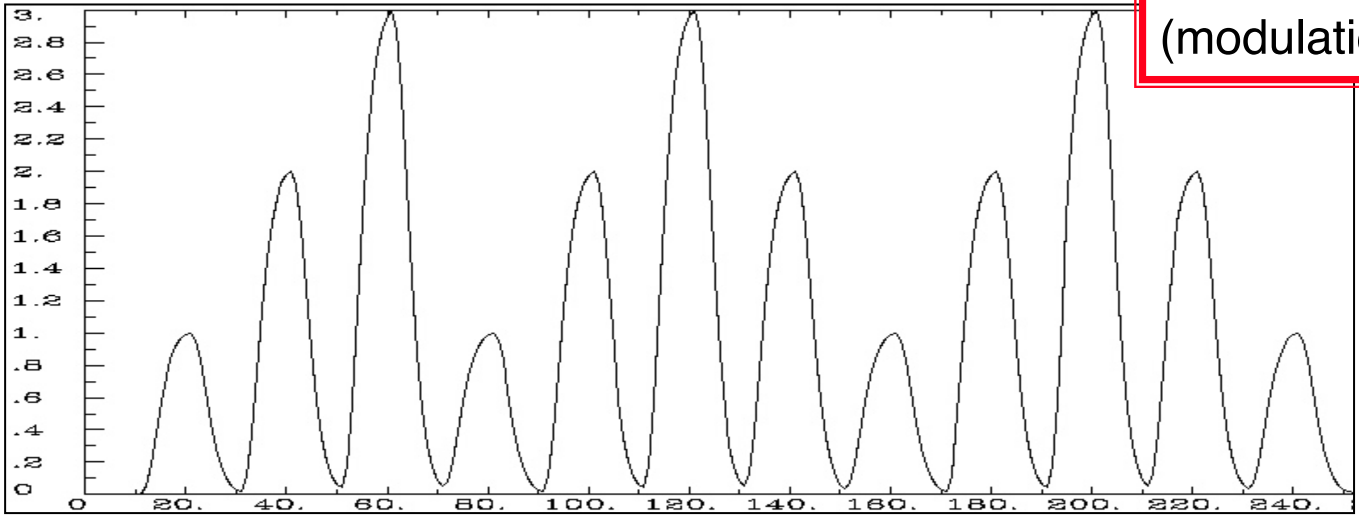
  - **AM1** is useful for duration modulated analysis (**dmBLOCK**) – to be described *real* soon
- Can have multiple amplitudes married to stimulus times
  - *e.g.*, To fit response model with cubic polynomial (nonlinear in ABI value ***a***), by giving 3 ABI values from a Legendre expansion in $\boldsymbol{a_k}$
  - Try not to go crazy with parameters***!***

AM Regression – 7a

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        {300 time points TR=1}
             -polort -1             {no polynomial baseline}
             -num_stimts 1          {one stimulus file}
             -stim_times_AM1 1 AM.1D 'BLOCK(10,1)'
             -x1D AM1.x1D           {save matrix to file}
1dplot AM1.x1D
```

AM1 model of signal (modulation = ABI)

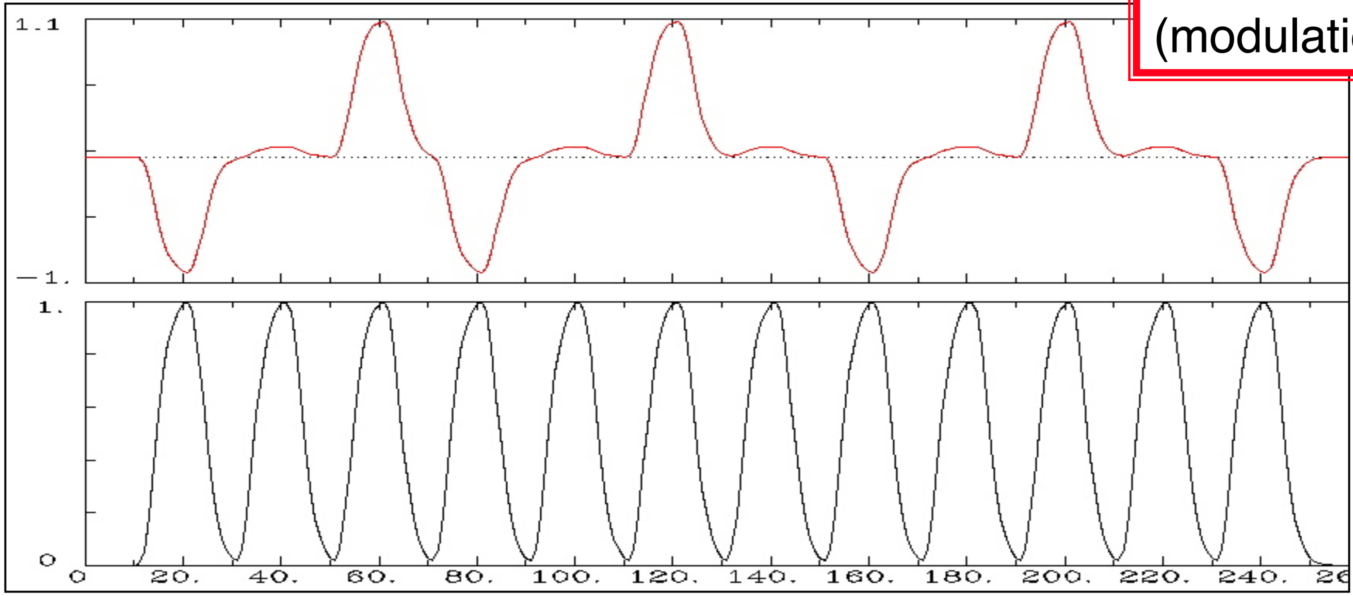


`s6c.TimeSeriesAnalysis.AMModel.csh`

AM Regression – 7b

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

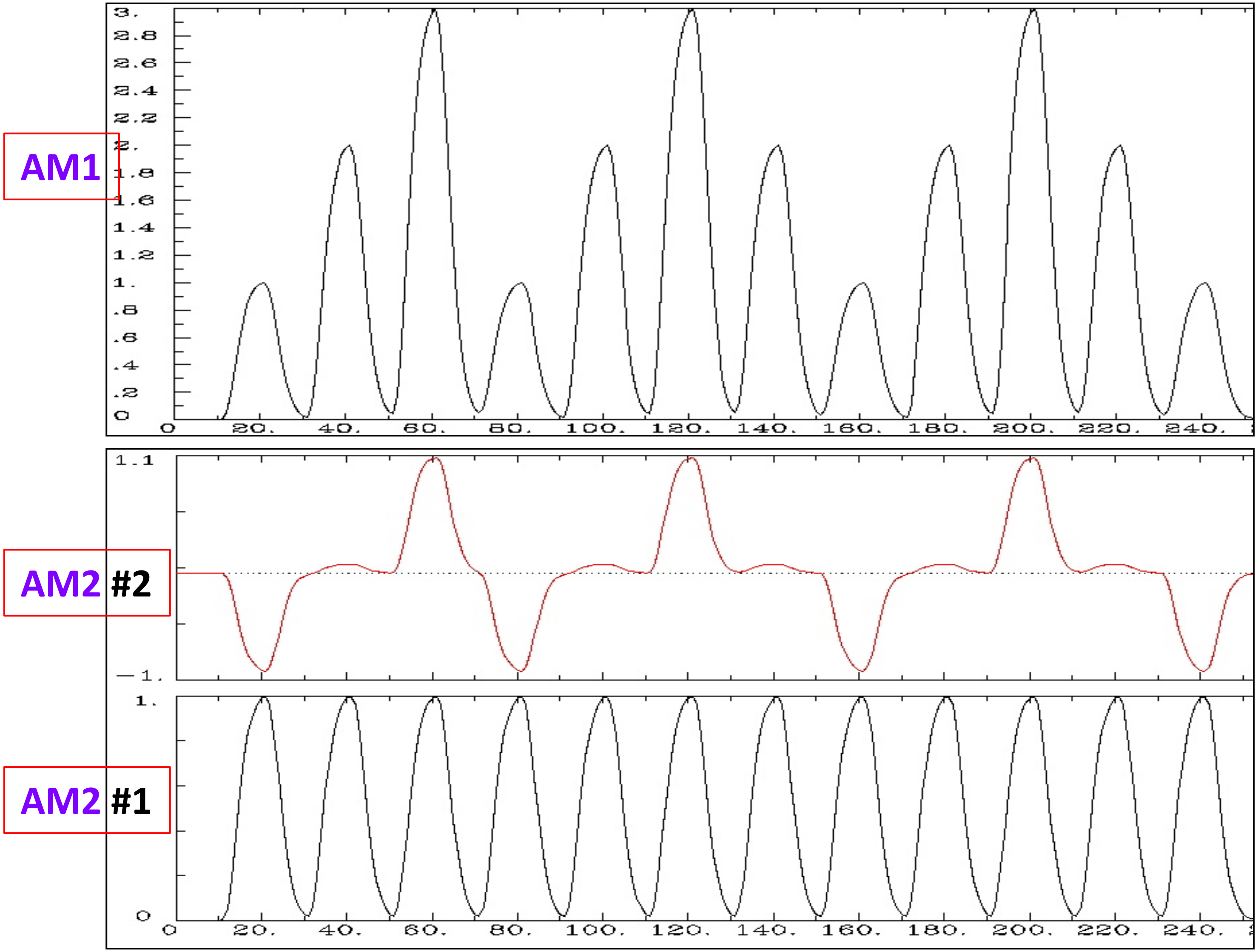
```
3dDeconvolve -nodata 300 1.0          {300 time points TR=1}
             -polort -1               {no polynomial baseline}
             -num_stimts 1            {one stimulus file}
             -stim_times_AM2 1 AM.1D 'BLOCK(10,1)'
             -x1D AM2.x1D             {save matrix to file}
1dplot AM2.x1D
```



AM2 model of signal (modulation = ABI)

2D sub-space spanned by these two time series

AM Regression – 7c

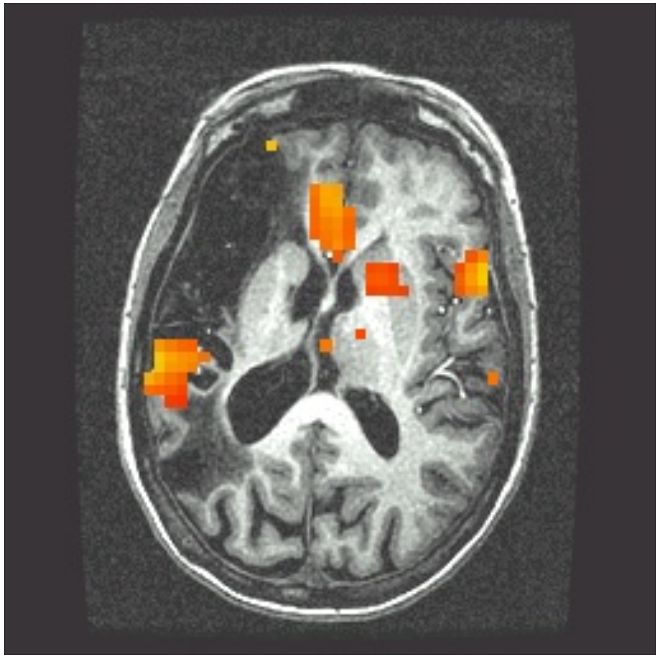


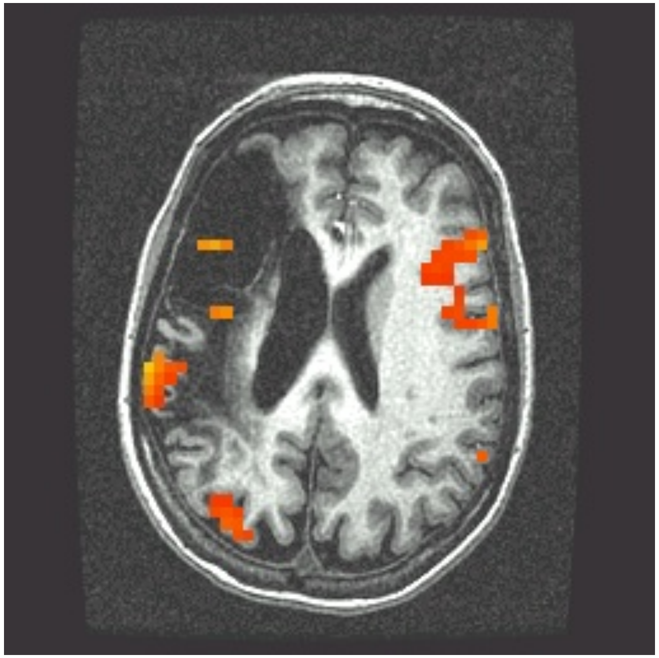
AM1 model is highly correlated with first (bottom) regressor from **AM2** model

If **AM1** model was used, but activation was really like **AM2** regressor #1, β would be positive

AM Regression – 8

- First actual user: Whitney Postman (formerly NIDCD)
- Picture naming task in aphasic stroke patients
- 2 slices showing activation map for BOLD responses proportional to ABI (β_{AM2})
  - What does this mean? Don’t ask me*!*





AM Regression – 9

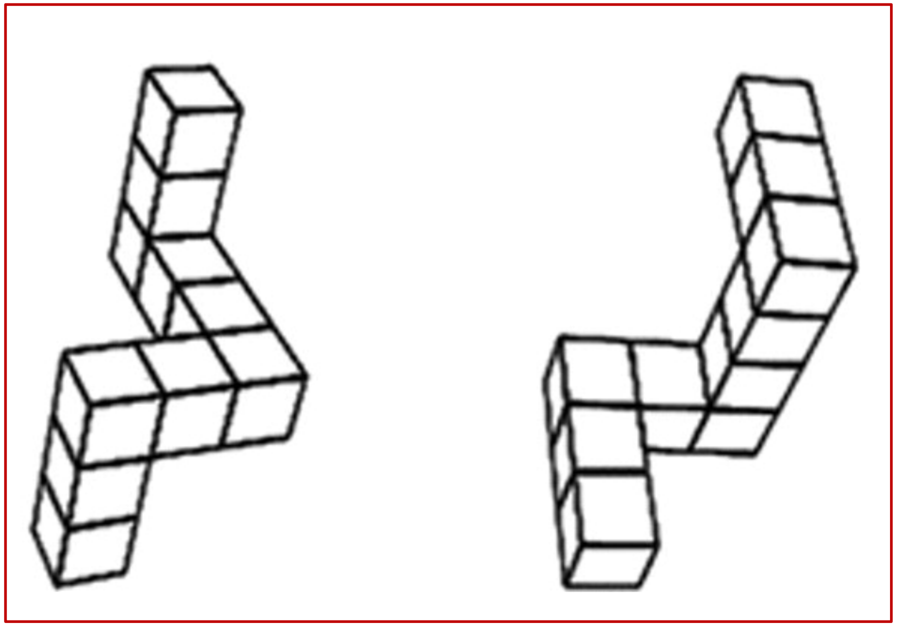
- Alternative: use **IM** to get individual βs for each block/event, then another regression on those values
- Could do nonlinear fitting (to these βs) via **3dNLfim**, or inter-class contrasts via **3dttest**, **3dLME**, **3dANOVA**, or intra-class correlations via **3dICC**, etc.
- What is better: **AM** or **IM**+*something more* ?
  - If you want linear fit of amplitude to ABI, then direct use of **AM** seems better than using 2 regression steps
  - If **AM** doesn't fit your models/ideas, then **IM**+ is clearly the way to go
  - Maybe consult with **AFNI** group to get hints/advice

DM Regression - 1

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

timing of events is measured

Sample Puzzle:
Are these 2 block figures just rotated in 3D from each other,
OR
Are they different block arrangements in 3D space?



DM Regression - 2

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

timing of events is measured

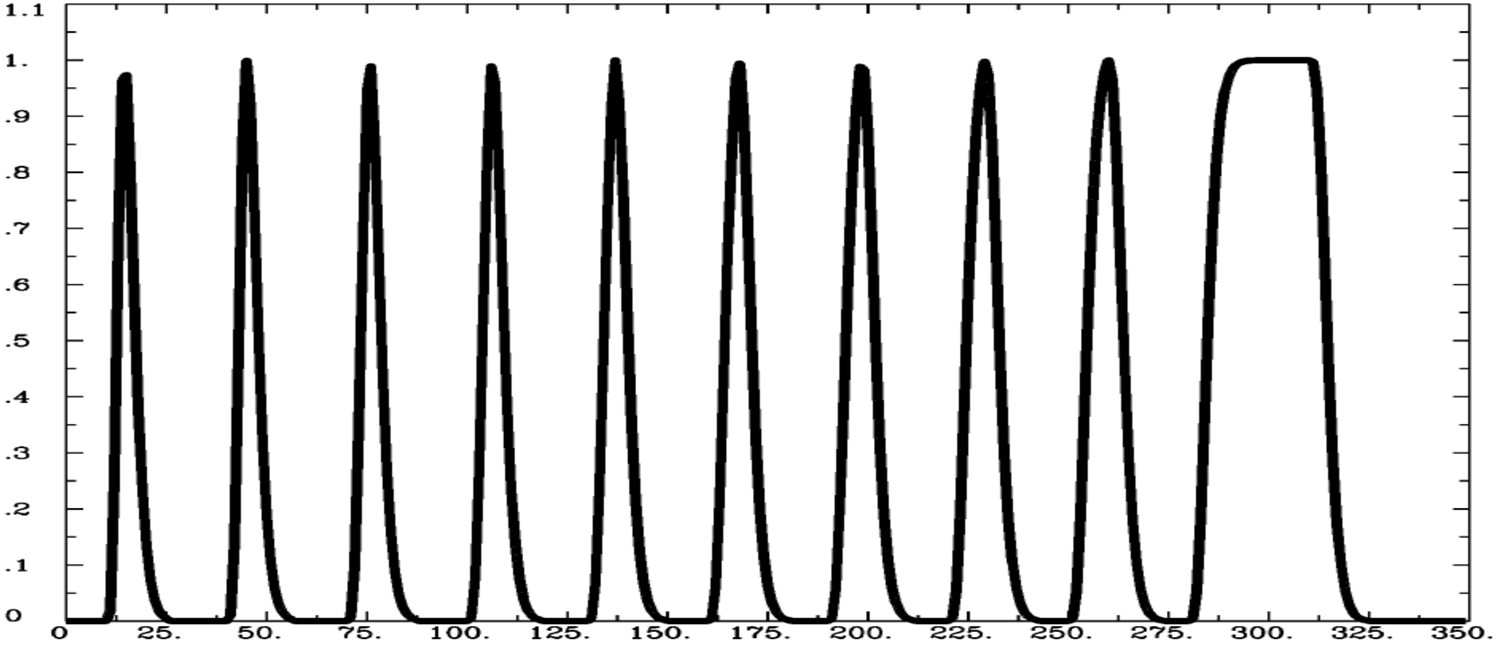
- Variable duration of phase (**b**) means that shape for task response varies between trials
    - If the variability of duration is more than ±1 TR
    - Which is contrary to the whole idea of averaging trials together to get decent statistics
        - Which averaging is basically what linear regression for the β weights does, in an elaborate way

DM Regression - 3

- **D**uration **M**odulated regression
- When different stimuli in the same class have different (and known) durations
- Controlled by using **dmBLOCK** response model
- Usually with **-stim_times_AM1** to indicate that an extra parameter is *married* to each stimulus time
  - But parameter is *duration*, not *amplitude modulation*
- You can also use **-stim_times_AM2** , by adding extra amplitude modulation parameter(s)
  - The duration parameter for **dmBLOCK** is always the ***last*** parameter in a *marriage*
  - Try not to go crazy with parameters***!***

DM Regression - 4

-
  ```
  3dDeconvolve -nodata 350 1 -polort -1 \
    -num_stimts 1                       \
    -stim_times_AM1 1 q.1D 'dmBLOCK(1)' \
    -x1D stdout: | 1dplot -stdin -thick -thick
  ```
- q.1D = 10:1 40:2 70:3 100:4 130:5 160:6 190:7 220:8 250:9 280:30



s6d.TimeSeriesAnalysis.DMModel.csh

Other Linear Regression Software in **AFNI**

- Program **3dTfitter**: solves linear regression models for special purposes
  - Voxel-dependent regressors
  - L2, L2 LASSO, and L1 solution methods
  - Constraints on fit parameter (β) signs
  - No statistics, just fits
- Program **3dTproject**: just calculates the residuals
  - No statistics, no βs, …
  - When there are only nuisance regressors, and the idea is to "clean up" or "regress out" these nuisances
  - Use cases: Resting state and Naturalistic FMRI
  - Much faster than other **3d*** regression programs

Nonlinear Regression in **AFNI**

- 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*
- Program **3dNLfim** can do nonlinear regression
  - User provides **C** function that computes a model time series, given a set of parameters (*e.g.*, ***a***, ***b***, ***c***)
    - Many sample model functions in the **AFNI** source code distribution – **`https://github.com/afni/afni`**
  - **3dNLfim** drives this **C** function repeatedly, finding set of parameters to best fit each voxel time series
  - Has been used to fit pharmacological models for DSC-MRI and other IV drugs (*e.g.*, cocaine)

AFNI

Didactics and Demonstrations

FMRI Task-Based Data Analysis at the Individual Level

Concepts 7 – Noise Modeling

“Noise” in FMRI

- MR thermal noise
- Cardiac and respiratory cycles
  - o In principle, could measure these sources of noise separately and then try to regress them out
  - o Scanner fluctuations (e.g., thermal drift of hardware, pulse sequence timing errors)
- Small subject head movements (10-100 μm)
- Very low frequency fluctuations (periods ≥ 100 s)
- “Serial correlation” in the noise time series affects the *t*- and *F*-statistics calculated by **3dDeconvolve**
- **Next slides**: **AFNI** program for this latter problem

Allowing for Serial Correlation - 1

- *t*- and *F*-statistics denominators: estimates of noise variance
  - White noise estimate of variance

$$\hat{\sigma}^2 = \frac{1}{N - m} \sum_{i=0}^{N-1} [\text{data}_i - \text{fit}_i]^2$$

    - N = number of time points
    - m = number of fit parameters
    - $N-m$ = degrees of freedom (DOF)
      - how many equal-variance ***independent*** random values are left after a pure noise time series is fit with m regressors

Allowing for Serial Correlation - 2

- **Problem**: if noise values at successive time points are correlated, this estimate of variance $\hat{\sigma}^2$ is biased to be too small
  - There aren't really $N-m$ ***independent*** random values left
  - Denominator $N-m$ too small implies t- and F-statistics are too large ☹
  - And number of degrees of freedom $N-m$ also too large
  - So significance (p-value) of activations ***in individuals*** is overstated (since t/F are too big)
  - **Subtler problem**: actual variance of $\boldsymbol{\beta}$ estimate is larger than one thinks ➜ $\boldsymbol{\beta}$ isn't as accurate as it could be

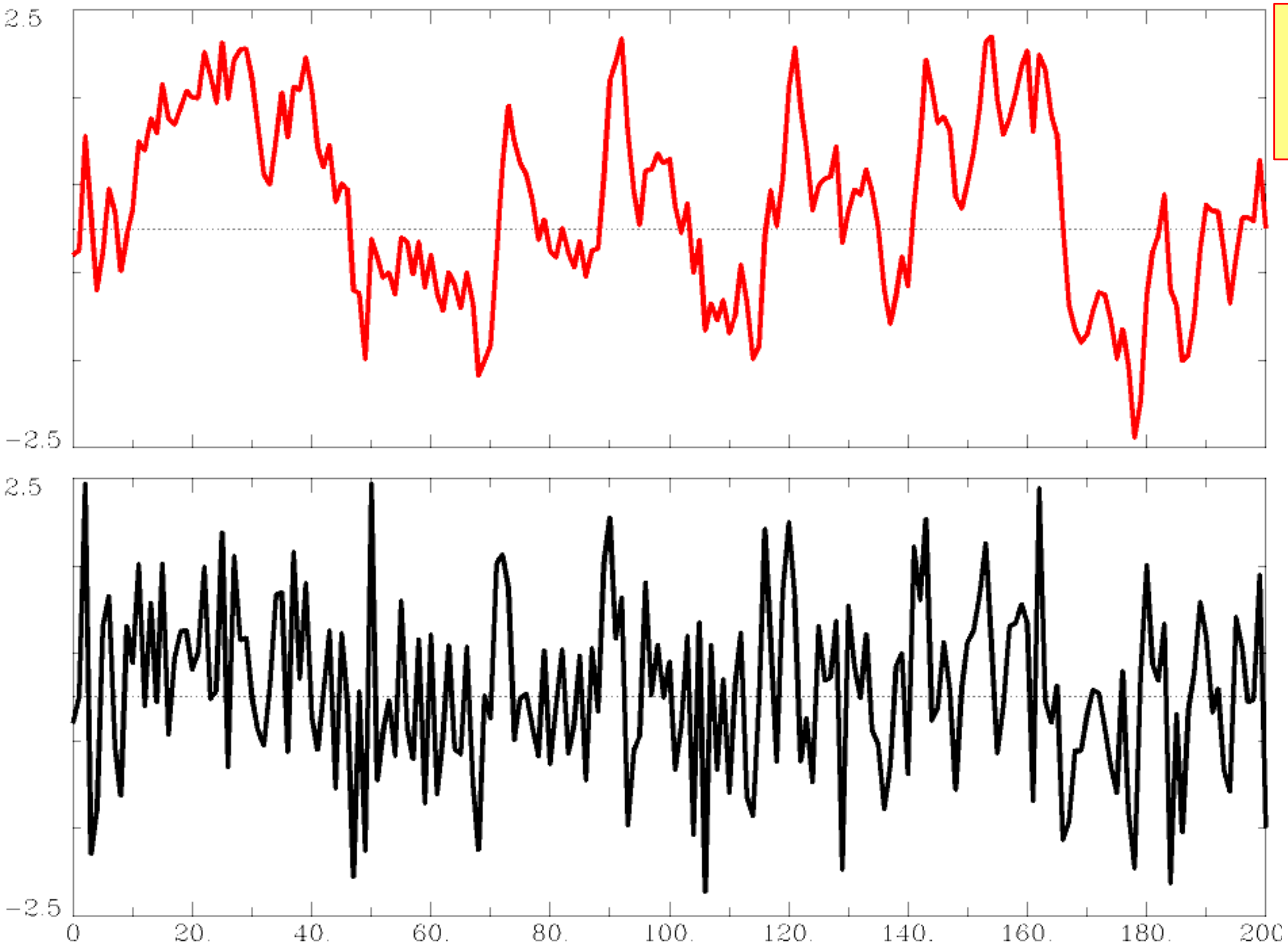
Allowing for Serial Correlation - 3

- **Possible ways to patch these problems:**
- **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 **g**eneralized **l**east **sq**uares (**G**LSQ instead of **O**LSQ), using this correlation, in one model (**REML** method)
    - Better estimates for $\hat{\sigma}^2$, for β, & keeps DOF $= N - m$
  - This is the technique that **AFNI** uses

Allowing for Serial Correlation - 4

- **REML** is a method for estimating variance+correlation parameters ***and*** estimating fit parameters (***β***s)
- Noise correlation structure is modeled as ARMA(1,1)
  - 2 parameters in ***each voxel***: ***a*** (AR) and ***b*** (MA)
    - ***a*** models how fast noise de-correlates over time
    - ***b*** models short-range correlation in time (1 TR)
    - Unlike SPM and FSL, *each voxel* gets a separate estimate of its own temporal correlation parameters
      - W Olszowy et al. Accurate autocorrelation modeling substantially improves fMRI reliability. *Nature Commun.* https://doi.org/10.1038/s41467-019-09230-w (2019)

Allowing for Serial Correlation - 5



ARMA(1,1)
$a = 0.8$ $b = 0.2$

White Noise
$a = 0$ $b = 0$

Correlation between top and bottom time series = **0.5225**

AFNI Script

```
1dgenARMA11 -seed 666 -num 201 -a 0.8 -b 0.2 > s7a.ARMA.1D
1dgenARMA11 -seed 666 -num 201 -a 0.0 -b 0.0 > s7a.WHIT.1D

1ddot -terse s7a*.1D

1dplot -png s7a.png s7a.WHIT.1D s7a.ARMA.1D
```

Script to produce plot on previous slide

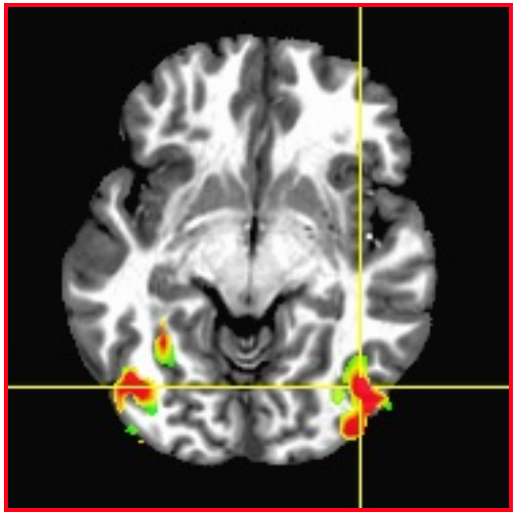
s7a.TimeSeriesAnalysis.ARMA11.csh

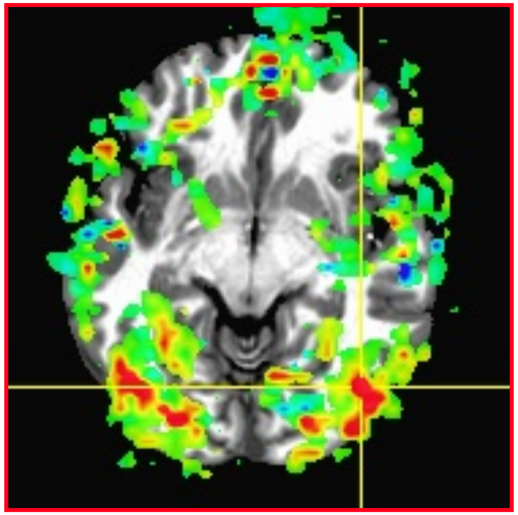
Allowing for Serial Correlation - 6

- Inputs to **3dREMLfit**
  - Regression matrix file (plain text)
  - Usually pre-computed by **3dDeconvolve** using **afni_proc.py**
  - **afni_proc.py** then runs **3dREMLfit**
    - Inputs are matrix file and 3D+time dataset
- Output datasets are structured as from **3dDeconvolve**
  - But statistics and βs are improved (we hope)

Allowing for Serial Correlation - 7

REML
F=3.15
p=0.001

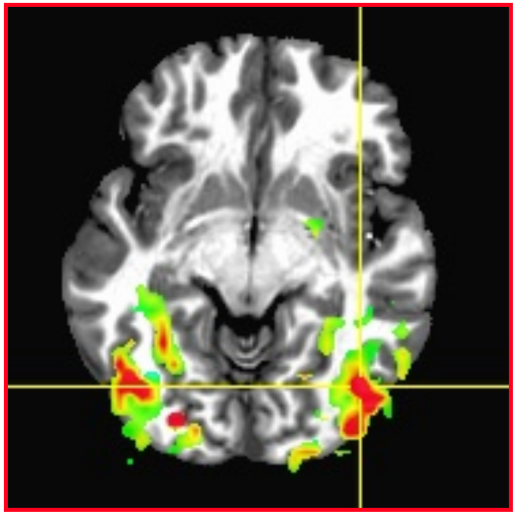


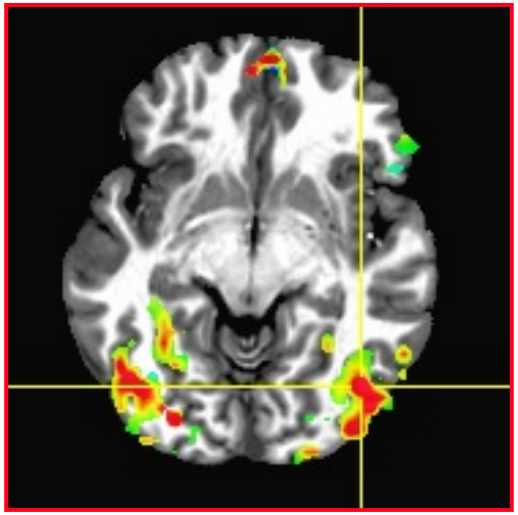


OLSQ
F=3.15
p=0.001

REML
F=1.825
p=0.061

- *F* = No activity outside brain!





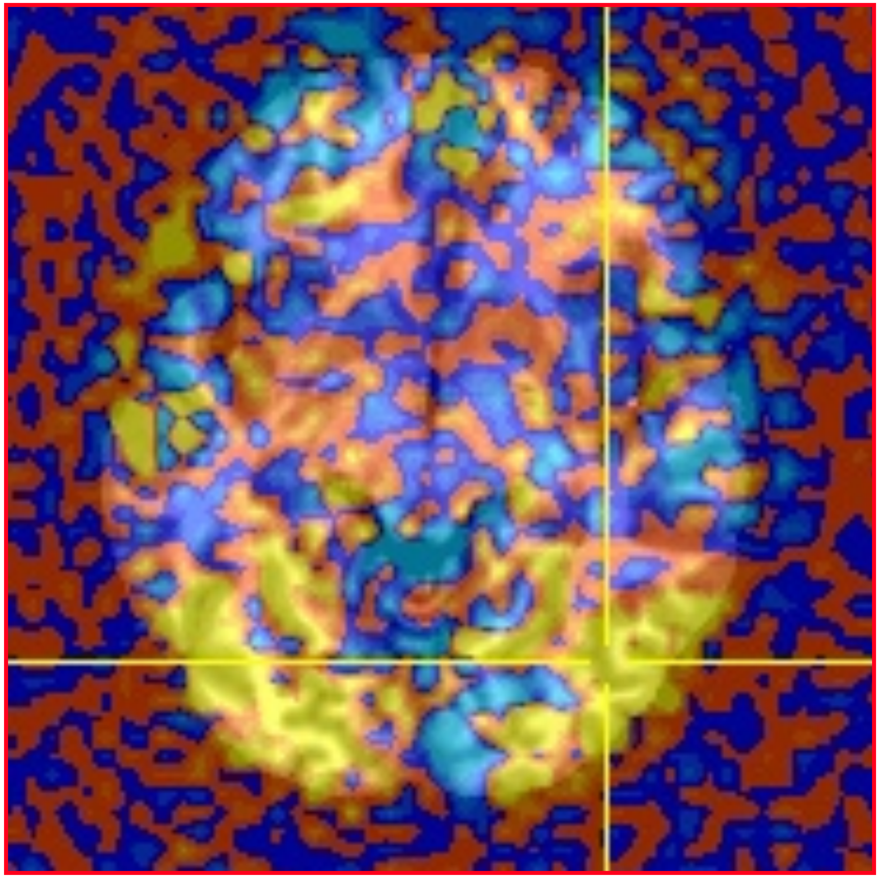
OLSQ
F=5.358
p=5e-7

- *F* = No activity outside brain!

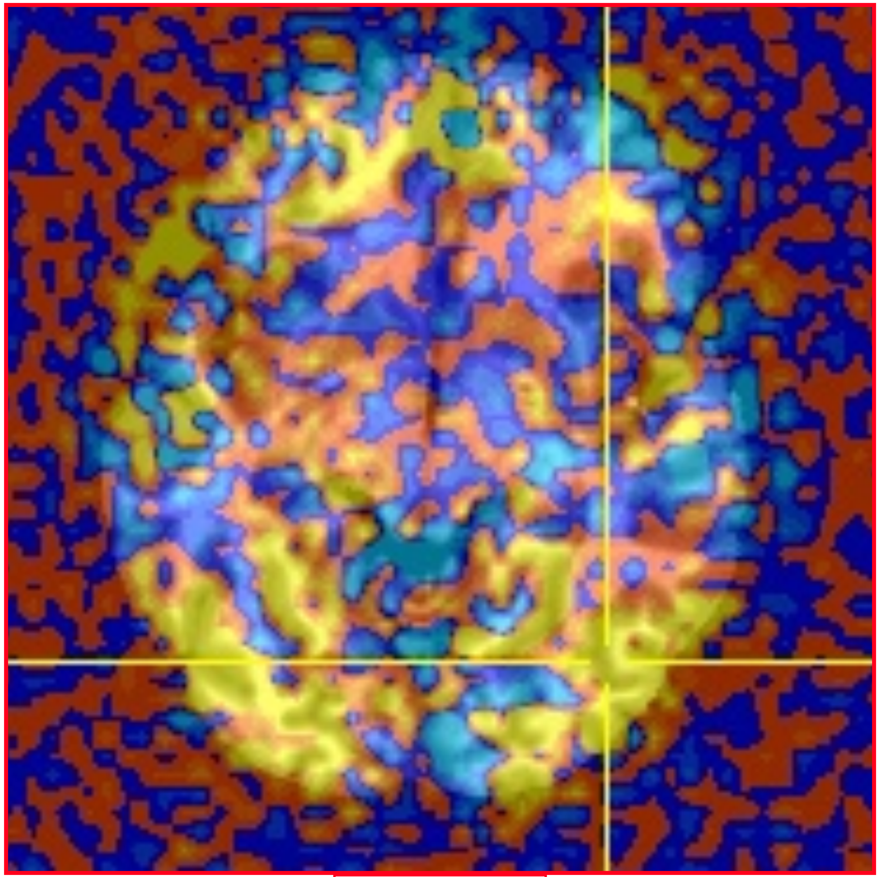
Oh My GOD ***!?!***

Allowing for Serial Correlation - 8

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



REML/GLSQ



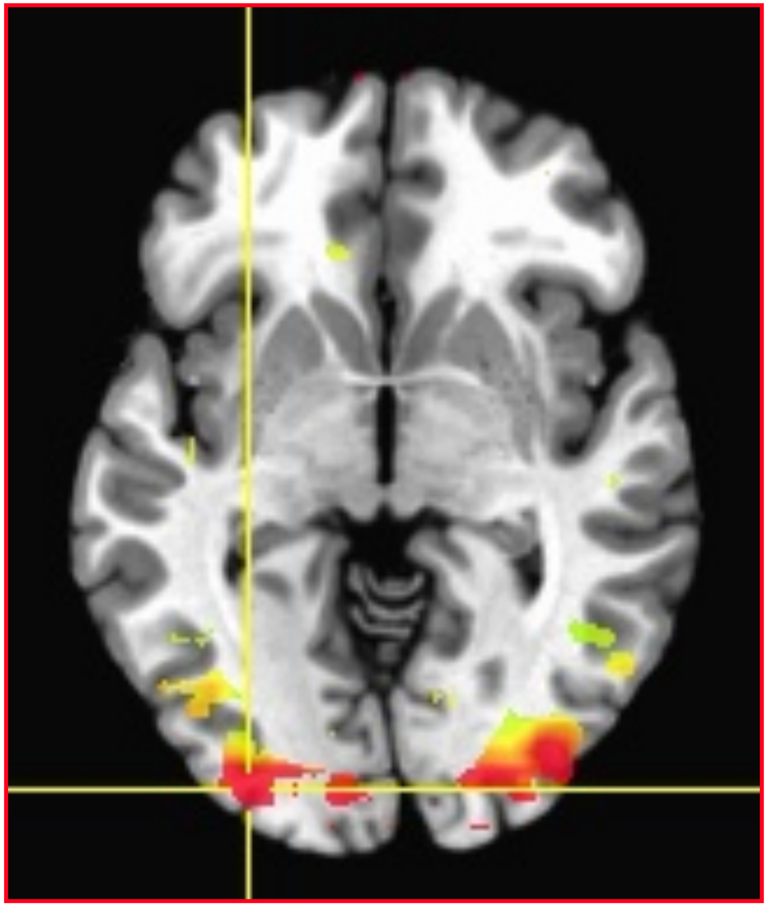
OLSQ

Allowing for Serial Correlation - 9

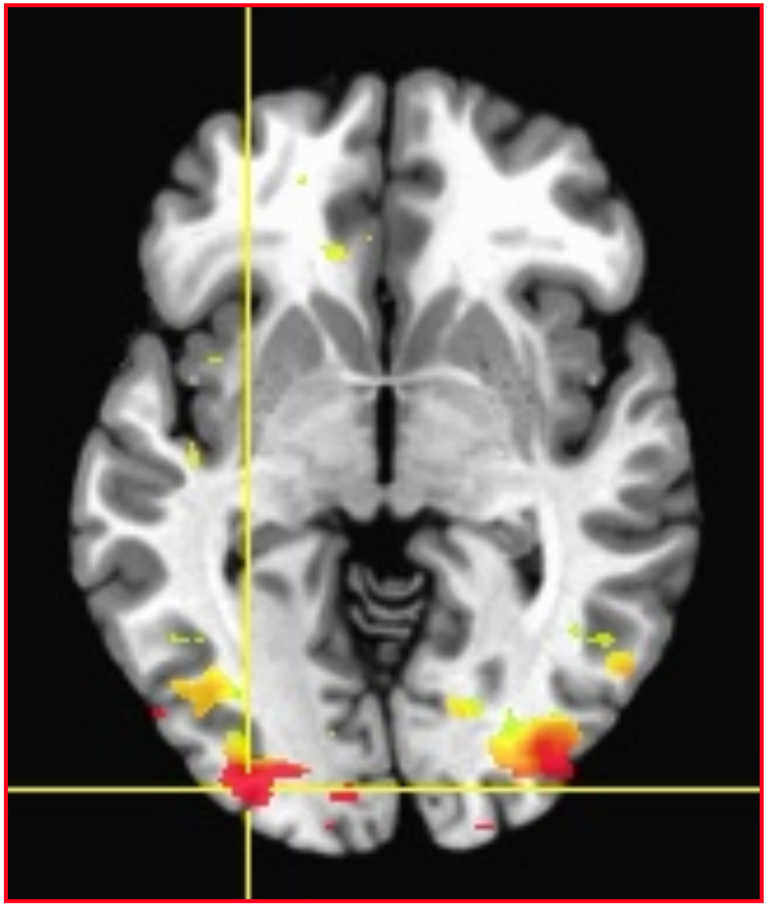
- For individual activation maps, **3dREMLfit**-ized *t*- and *F*-statistics are significantly different, and more accurate
- But … There are at present very few applications for such individual FMRI activation maps
  - pre-surgical planning; longitudinal studies
- For standard group analysis, inputs are only *β* fit parameters
  - Which don't change so much between REML/GLSQ and OLSQ
  - In other words – older OLSQ-based group analyses are *not* invalidated

Allowing for Serial Correlation - 10

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



REML/GLSQ



OLSQ

Allowing for Serial Correlation - 11

- Current plans (spring 2020)
- Extend the temporal correlation model to higher order ARMA
- **Motivation**: faster TR data (1 s or less) shows respiration and cardiac ″noise″
- Instead of noise correlation decaying away monotonically as the "lag" between 2 time points increases, it both decays and oscillates
- ARMA(3,1) or ARMA(5,1) can pick up these effects and compensate for them, with extra parameters
  - Needed = efficient algorithm for voxel-wise estimation

AFNI

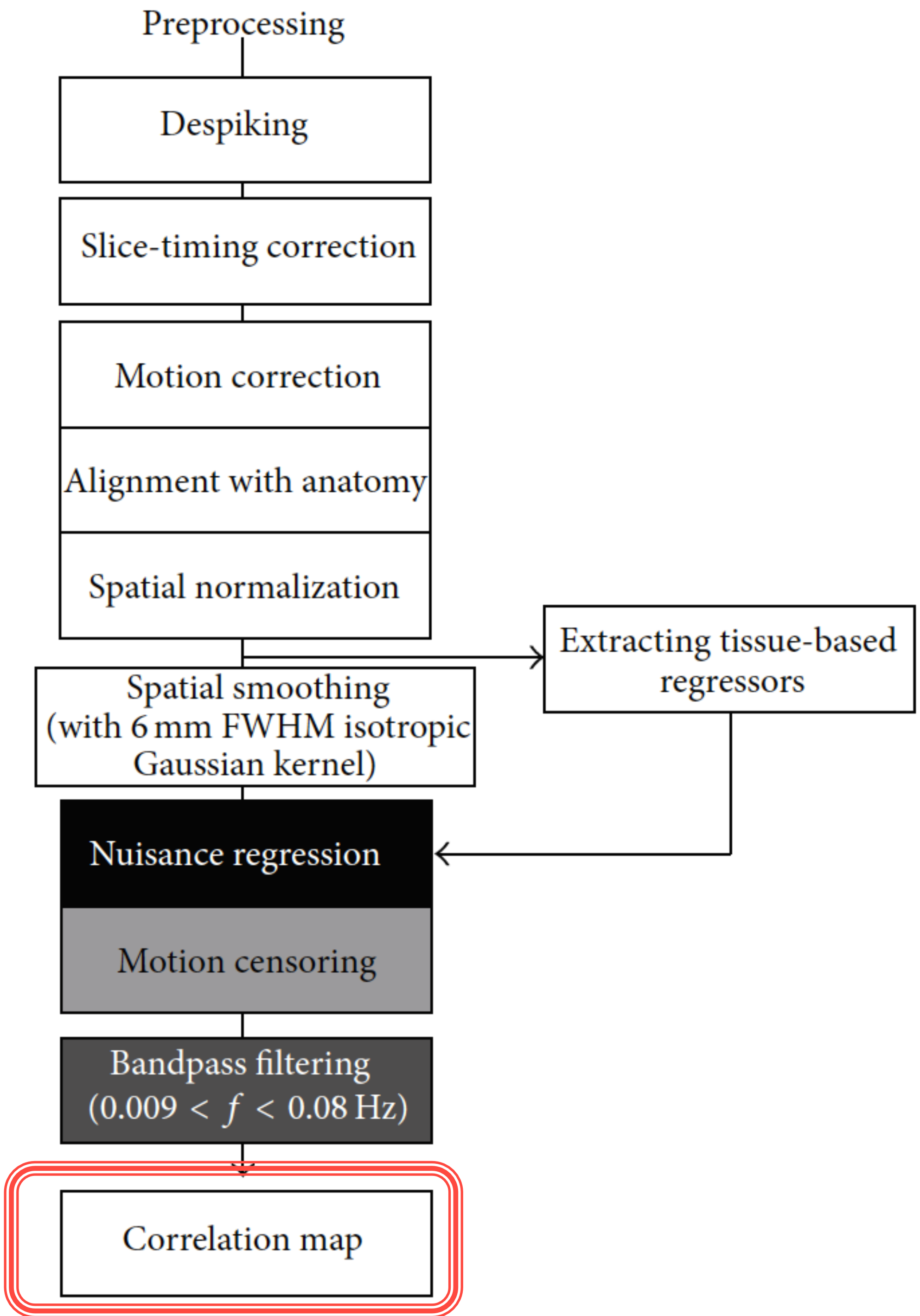
Didactics and Demonstrations

FMRI Task-Based Data Analysis at the Individual Level

Concepts 8 – Pre-Processing

Pre-Processing Steps

- Regression is *final* step in time series analysis
- Before regression, 3D+time dataset(s) are processed in several ways to "clean up" the data
- Pre-processing for task based, resting state, and naturalistic FMRI are much the same
  - The main difference is the attitude towards the outputs of the regression step
  - **Task**: the $\boldsymbol{\beta}$s are output of interest
  - **RS**, **Naturalistic**: no task regressors – ***residuals*** are output of interest
    - What is left after regressors of no interest are removed

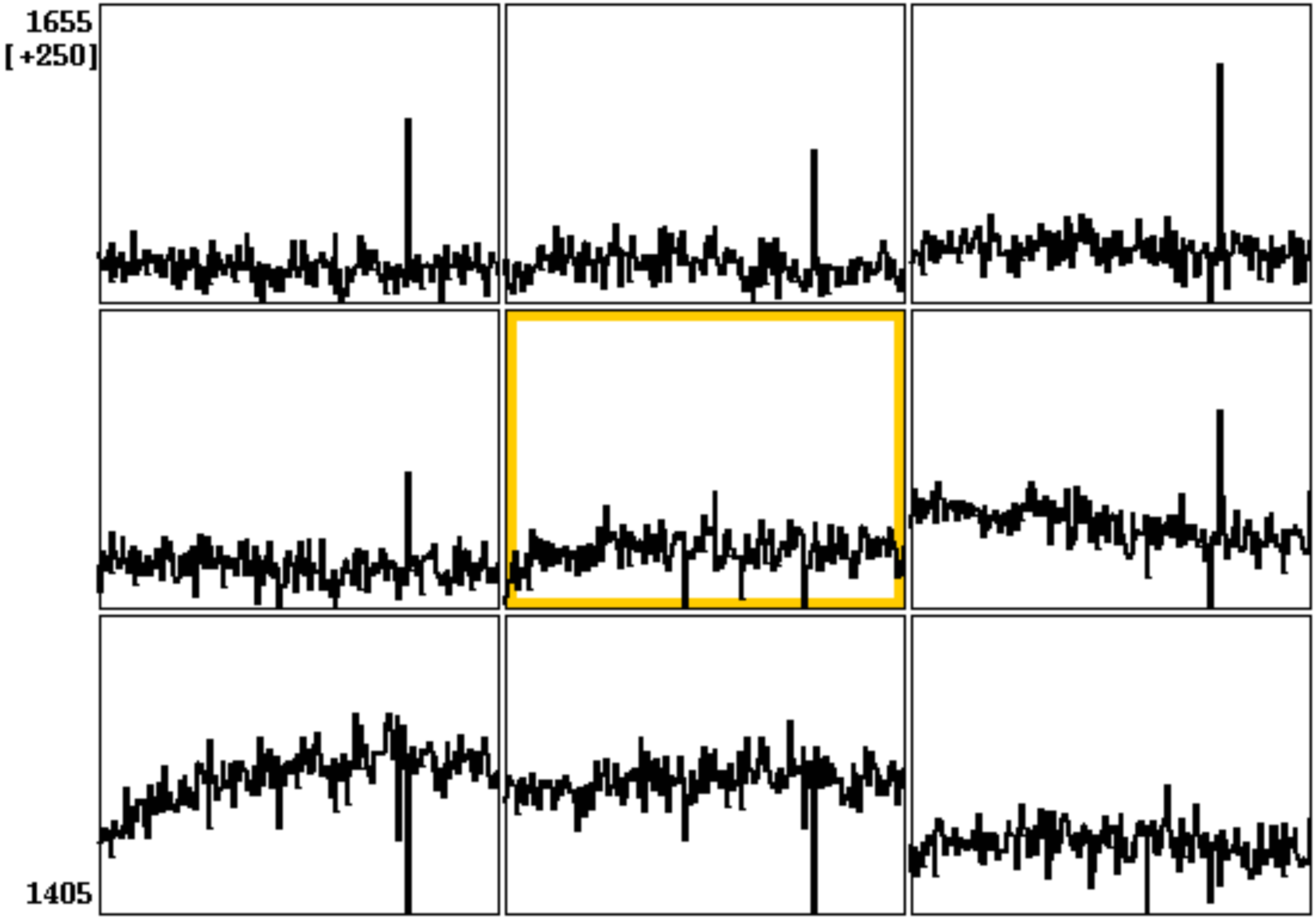


AFNI's recommended RS-FMRI pre-processing steps

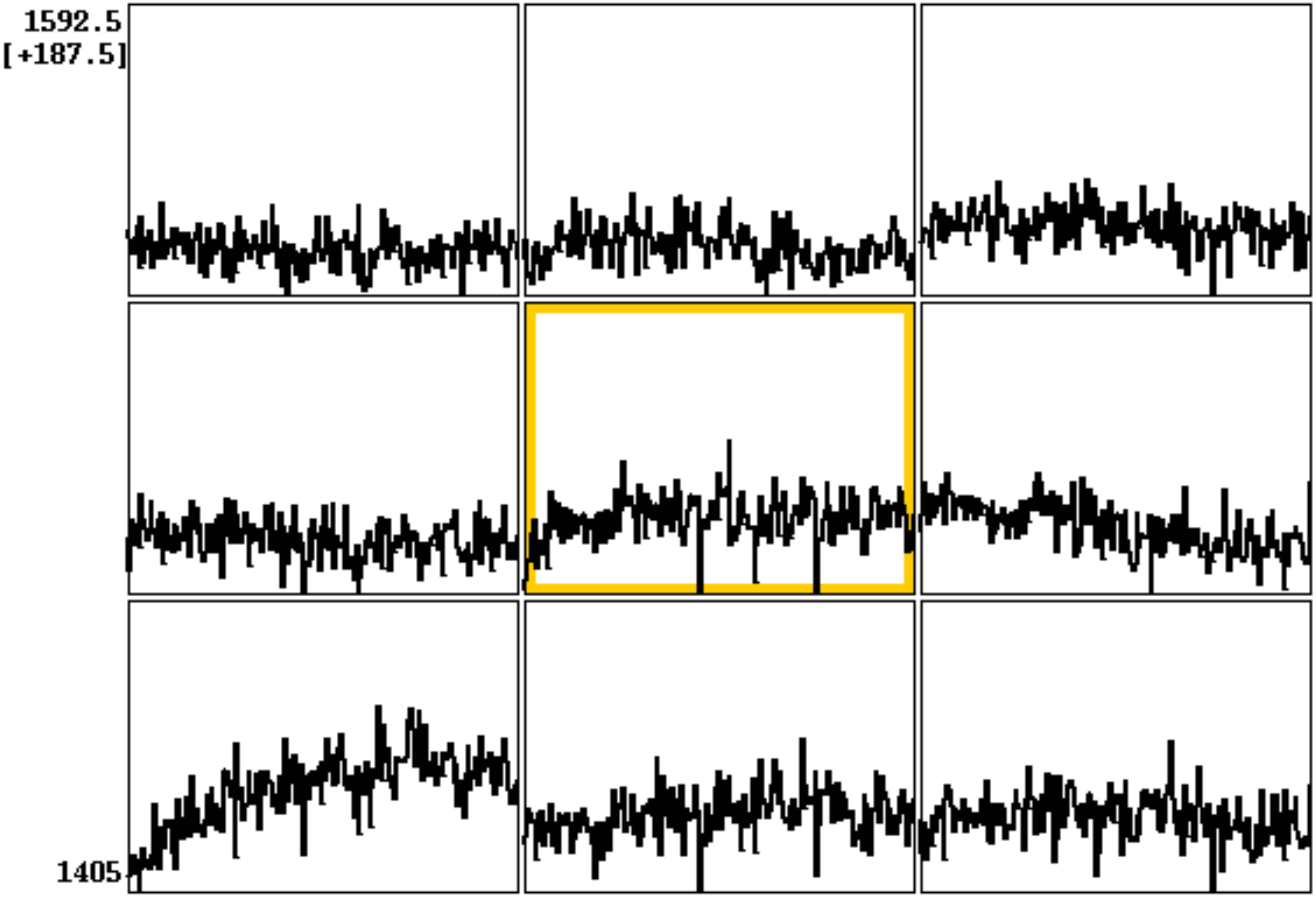
HJ Jo *et al*, 2010 and 2013

Carried out using **afni_proc.py**

Step 1 = Despiking (before)



Step 1 = Despiking (after)



Step 2 = Slice Timing Correction

- 2D Slices are acquired at different times within one 3D "volume" TR
- Even the same physiological BOLD effect in 2 different slices will show up (slightly) differently due to being measured at different times
- And so will be less correlated/less identical than they "should be"
- Solution: interpolate in time to some common reference point before calculating regression
    - Not perfect, because are also interpolating noise

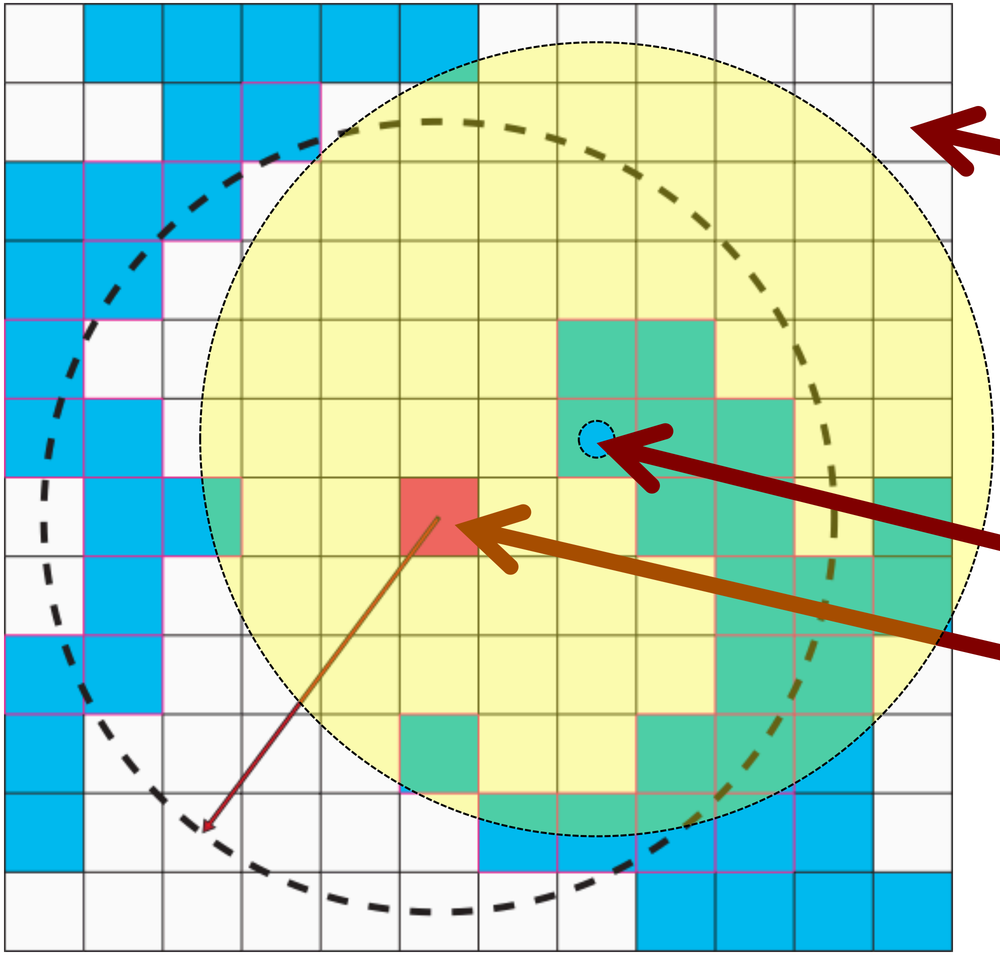
Step 3 = Motion Correction
Step 4 = EPI Alignment with Anatomy
Step 5 = Spatial Normalization to Template

- **Step 3**: Even more important for RS/naturalistic FMRI, since the BOLD effect is smaller and more spatially diffused than in task FMRI, so compensating for subject head motion is crucial
- **Step 4**: Needed for step 5, and for assigning RS-FMRI results to brain regions
- **Step 5**: Needed for group studies or using atlases
  - Spatial transformations to bring 3D datasets into alignment computed separately in Steps 3-5
  - Combined to transform datasets in one final operation

Step 6 = Extract Tissue Based Regressors

- Purpose of tissue based regressors is to extract time series fluctuations that are ***not*** BOLD signal
- So we can regress them out of the data at **Step 8**
- Common choices include:
  - Average of all white matter (WM) signal time series
  - Several principal components of all WM time series (CompCor method)
  - Average global brain signal time series (GS) ☹
  - Average signal from CSF in ventricles
- Less common (only in **AFNI**): **ANATicor** …

ANATicor – Tissue Based *per voxel*



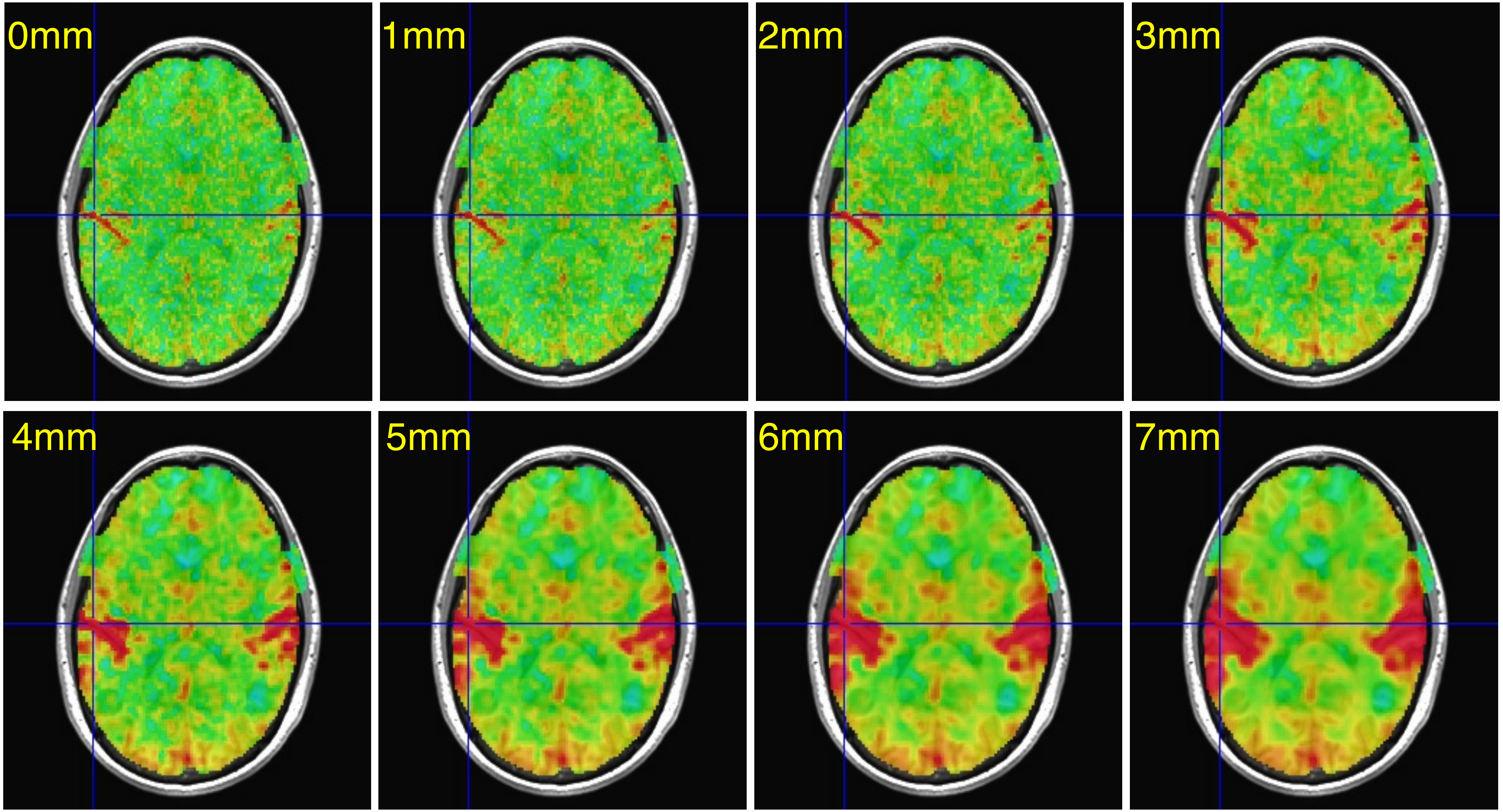
Eroded WM mask (WMe) [voxels in white]

Average over WMe voxels inside 25mm radius

Step 7 = Spatial Blurring

- Important for RS/naturalistic FMRI since the BOLD signal fluctuations are small
  - Also important in group studies so that errors in inter-subject alignment can be compensated for
- Averaging locally will tend to cancel noise and add up coherent (similar looking) signals
- **Important**: blur ***after*** tissue based signal extraction
- *Otherwise,* will get unintended signals in WM and CSF that were blurred in from nearby GM (gray matter)

Effects of Blurring on RS-Correlation



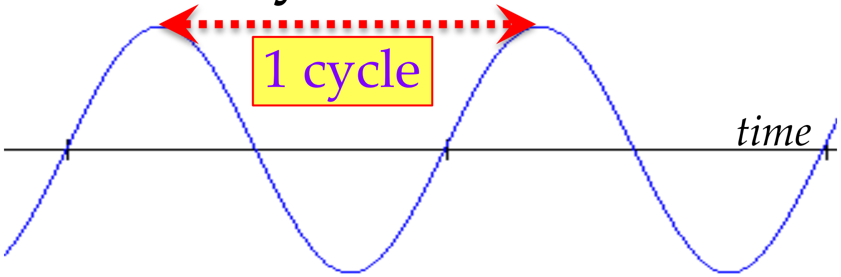
Little blurring means little long-range RS correlation!

Step 8 = Nuisance Regression - 1

- In task-FMRI, regression is to find signal amplitudes (*β*s) of task model components while also removing the nuisance model components
  - Nuisances: motion parameters, *motion parameter time derivatives*, WM signals, *measured respiration signal*, *etc*
- In RS/naturalistic-FMRI, there are no task model components to estimate
- All we want from the pre-processing and regression is to remove the nuisance components (as much as practicable) and compute the residuals
  - These residuals are the "purified" output, ready for further analysis (*e.g.*, correlations)

Step 8 = Nuisance Regression - 2

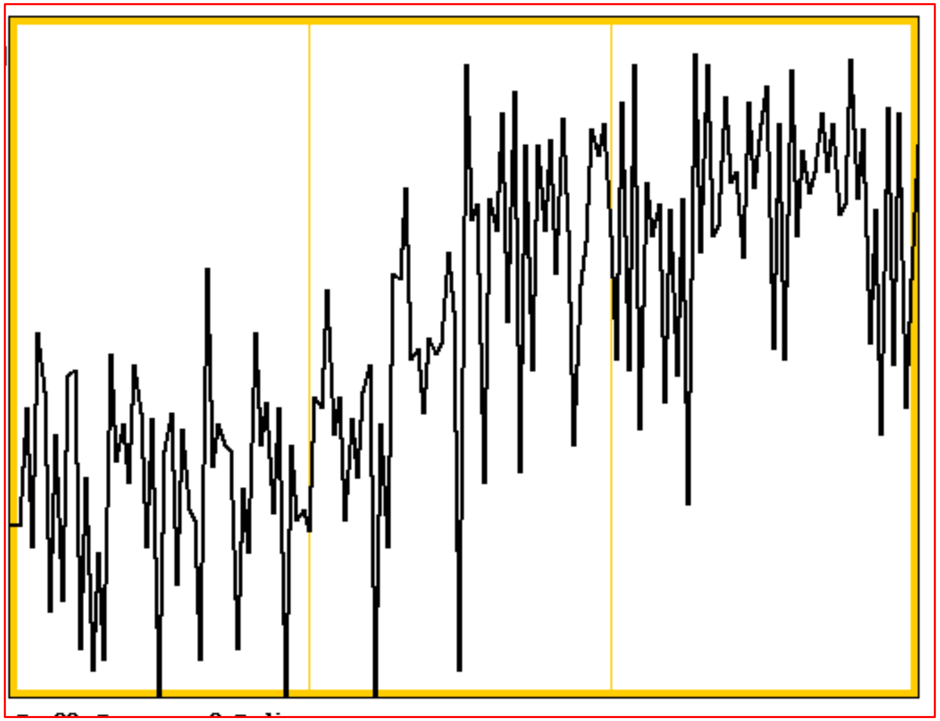
- Another operation usually (but not always) used in RS-FMRI is **bandpassing**
  - It involves removing all frequency components from the data *except* those in a specific band (or range)
- Frequency: units are Hertz (Hz) = cycles per second
  - 1 Hz = 1 cycle per second
  - 0.01 Hz = 0.01 cycle per second = 1 cycle in 100 sec
  - 100 Hz = 100 cycles per second = 1 cycle in 0.01 sec
  - ″cycle″ = full sine wave ➜
  - Larger frequency = faster
  - Lower frequency = slower



Step 8 = Nuisance Regression - 3

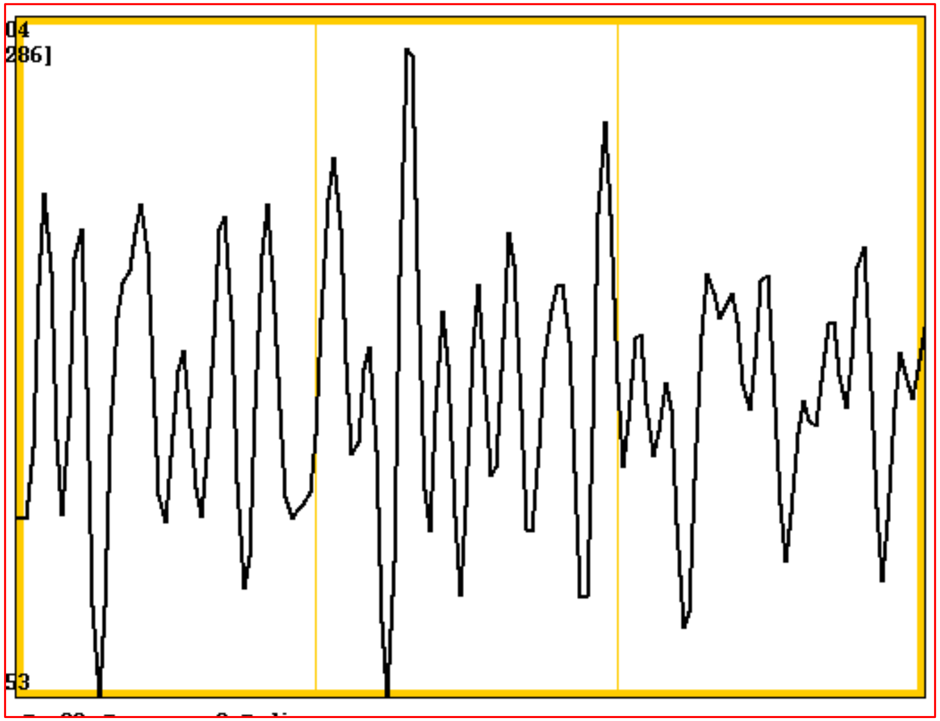
- In RS-FMRI, it is common to bandpass out all frequencies **higher than** 0.10 Hz and **smaller than** 0.01 Hz
  - Keep only data fluctuations that occur in the 10-to-100 second range
  - Faster or slower = **OUT**
- The idea is these don't contain much BOLD effect, so should be removed before analysis of residuals
- This idea is controversial
  - There is evidence that neurally relevant fluctuations occur up to 0.20 Hz (5 s time scale)

Step 8 = Nuisance Regression – 3a



Data voxel
No bandpass

→→→→→→→→→→→



Data voxel
Bandpass 0.01-0.10

BP: removes slow drift and reduces rapid oscillations

`s8a.TimeSeriesAnalysis.Bandpass.csh`

Step 8 = Nuisance Regression - 4

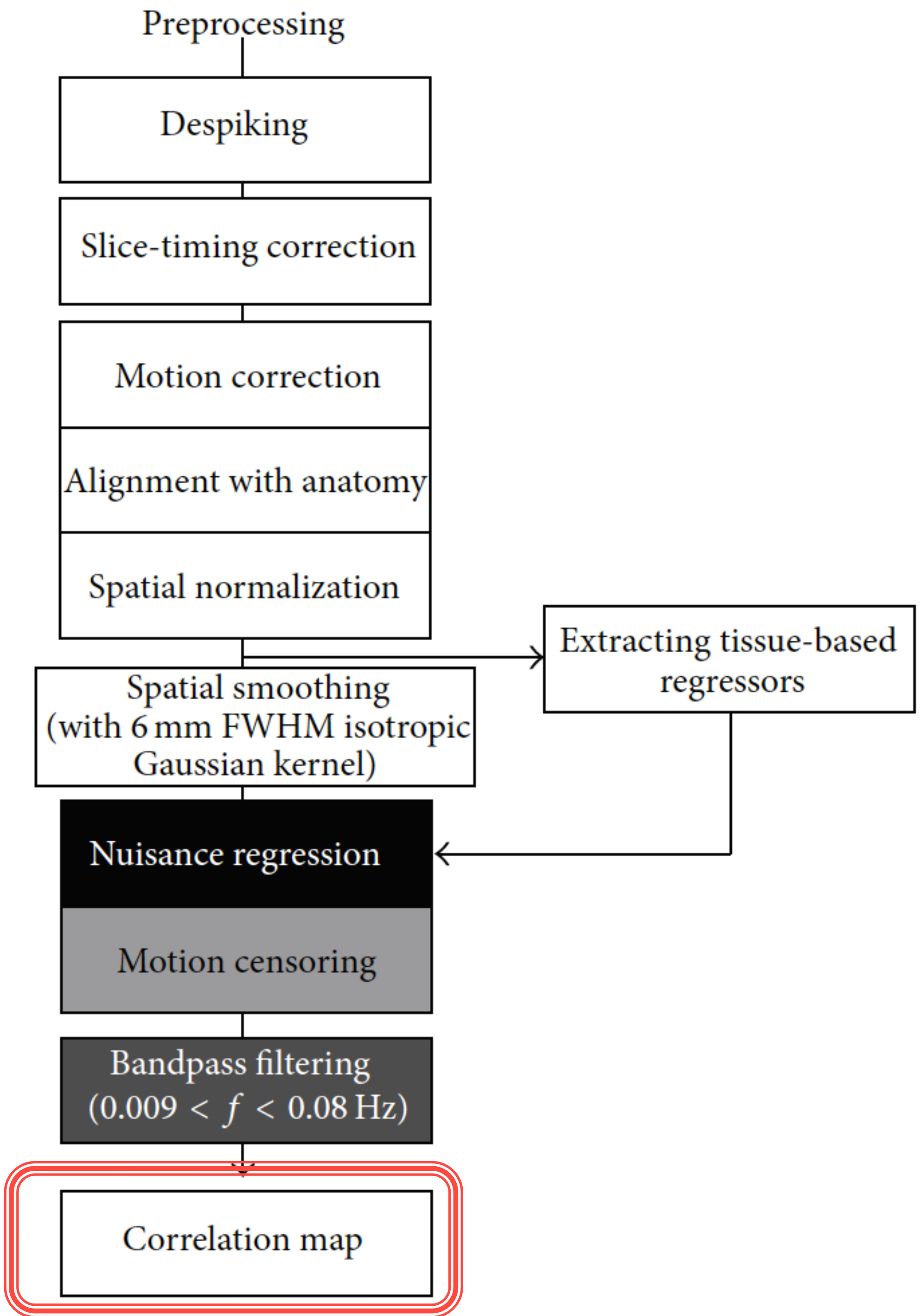
- It is common to censor out “**bad**” time points, so they aren’t used in the analysis (task or RS)
  - o “Bad” = too much motion, or that volume has too many “outlier” data points, or …
- It is important to censor bad time points ***before/during*** the nuisance regression, not afterwards
  - o Otherwise, they will affect regression results and contaminate residuals even at un-censored times
  - o In **AFNI**, censoring is done by removing the offending data time point from the analysis (matrix and data)
  - o Alternative: include an extra regressor which=0, except=1 at time point to be killed (SPM, FSL)

Step 8 = Nuisance Regression - 5

- In **AFNI**, nuisance regression, bandpassing, and censoring for RS-FMRI are all done in the same program: **3dTproject** (residual computing only)
  - Which allows for voxel-specific regressors (ANATicor)
  - **3dTproject** is much faster than **3dREMLfit**, since it does not have to compute βs or statistics
- For task-FMRI, regression is done with program **3dREMLfit** (also allows for voxel-specific regressors)
- How does **afni_proc.py** know which program to use?
  - If no task timing files are given, then it uses **3dTproject**, otherwise **3dREMLfit**

Step 8 = Nuisance Regression - 6

- Naïve people have done these 2 steps in sequence:
  - Bandpass the data
  - Regress *other* nuisance vectors from bandpassed data
  - Doing these operations in 2 steps (instead of one) is not just ***bad***, it is ***WRONG***
- Since nuisance regressors will contain some rejected frequency components, these unwanted components will "leak" back into the data at second regression
  - *Unless* nuisance regressors were bandpassed also
  - The same warning applies to bandpassing and censoring – they should be done together
- These reasons (plus speed) are why **3dTproject** was written



Preprocessing

Despiking

Slice-timing correction

Motion correction

Alignment with anatomy

Spatial normalization

Extracting tissue-based regressors

Spatial smoothing (with 6 mm FWHM isotropic Gaussian kernel)

Nuisance regression

Motion censoring

Bandpass filtering ($0.009 < f < 0.08$ Hz)

Correlation map

AFNI's recommended RS-FMRI pre-processing steps

HJ Jo *et al*, 2010 and 2013

Carried out using **afni_proc.py**

Preprocess via afni_proc.py

```
## Adapted from Example 9b in afni_proc.py -help
afni_proc.py -subj_id s620                        \
 -dsets s620_rest_r1+orig.HEAD                    \
 -blocks despike tshift align tlrc volreg         \
         blur mask regress                        \
 -tcat_remove_first_trs 2                         \
 -volreg_align_e2a                                \
 -blur_size 6                                     \
 -regress_anaticor_fast                           \
 -regress_censor_motion 0.2                       \
 -regress_censor_outliers 0.1                     \
 -regress_bandpass 0.01 0.2                       \
 -regress_apply_mot_types demean deriv            \
 -regress_run_clustsim no  -regress_est_blur_errts
```