

FMRI Task-Based Data Analysis at the Individual Level

SSCC/NIMH/NIH/DHHS/USA/Earth

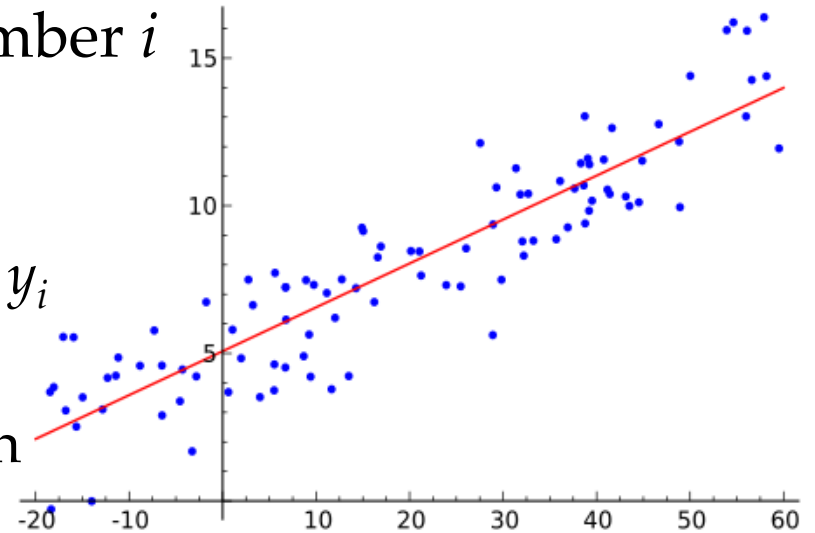


Overview

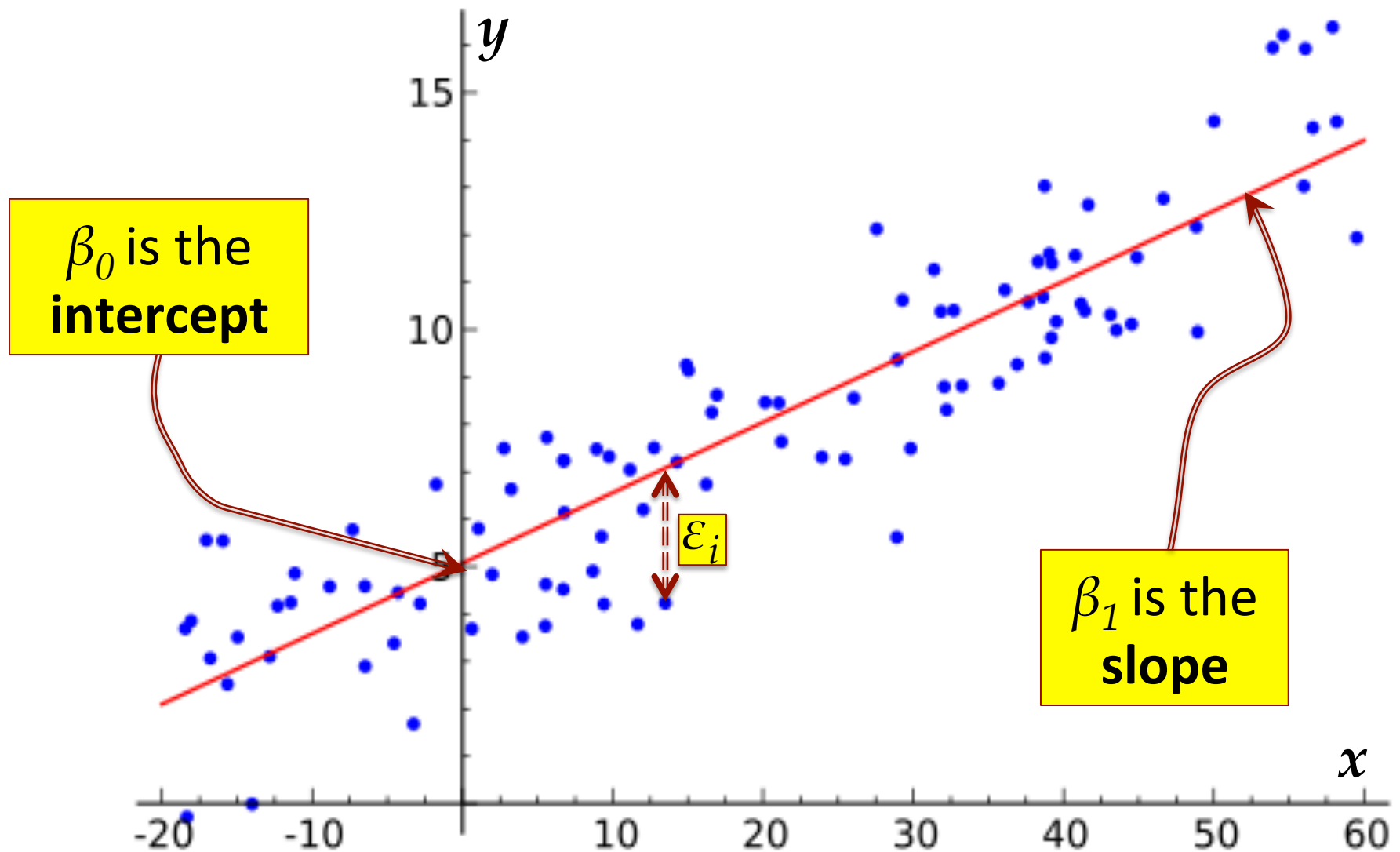
- Basics of linear models for data analysis
- FMRI data decomposition: three components
 - **Baseline + slow drift + effects of no interest; Effects of interest; Noise**
 - Effects of interest – understanding BOLD vs. stimulus
 - IRF and HRF and HDR
- Three modeling strategies
 - Fixed-shape HRF
 - Variable HRF shape
 - Fixed major HRF shape plus a little shape adjustment
- Other issues
 - Multicollinearity
 - Run catenation
 - Percent signal change

Basics of Linear Modeling

- **Regression**: finding a relationship between a response / outcome (dependent) variable and one or more explanatory (independent) variables (**regressors**)
 - Also called **linear model** or **linear regression**
- Equations
 - i =index of data = 0, 1, 2 ... N-1 (total of N data points)
 - x_i =explanatory model (known value) for data point number i
 - y_i =data value for data point number i
 - $y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$ **or** $y_i \approx \beta_0 + \beta_1 x_i$
 - β_0 and β_1 are **model fit parameters**
 - to be calculated from the x_i and y_i
 - ε_i are the **residuals**
 - what are left after the regression
 - assumed to be **random noise**



$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \quad \text{or} \quad y_i \approx \beta_0 + \beta_1 x_i$$



Modeling with Vectors and Matrices

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

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

- In **vector-matrix** form (**bold** letters for vectors and matrices)
 - $\mathbf{y} \approx \mathbf{X} \boldsymbol{\beta}$ or with residual vector $\mathbf{y} = \mathbf{X} \boldsymbol{\beta} + \boldsymbol{\varepsilon}$
- By writing it out this way, the equations become more compact and easier to look at and easier to understand
- Each column of \mathbf{X} matrix is a **regressor** or **model component**
- We assume the columns of \mathbf{X} are known, and that data vector \mathbf{y} is known (measured)
- Goal is to compute **parameter vector** $\boldsymbol{\beta}$ (and statistics about $\boldsymbol{\beta}$)
- Most of this talk: where do we get \mathbf{X} for FMRI task analysis? ⁵

Solving a Linear Model

Vector \mathbf{y} is sum of matrix \mathbf{X} times vector β plus residuals ϵ

- Solution for linear regression $\mathbf{y} = \mathbf{X}\beta + \epsilon$
 - “Project” data \mathbf{y} onto the space of explanatory variables (\mathbf{X})
 - **OLS** formula for solution: $\hat{\beta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$
 - Columns of \mathbf{X} are the **model** for data vector \mathbf{y}
- Meaning of coefficient: β_i value is slope, marginal effect, or effect size associated with **regressor** number i [column i in \mathbf{X}]
- β_i value says how much of regressor number i is needed to fit the data “best” – in the **Ordinary Least Squares** sense
 - That is, the sum of the squares of ϵ_i is made as small as possible
- If we don’t care about regressor number i , then we don’t care about the value of β_i
 - But we included regressor number i 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 carried out after solving for β vector
- Some examples, with particular null hypotheses H_0

- Student t -test for each β_i of interest

$$H_0: \beta_3 = 0$$

- Student t -test for linear combination of some β_i values = general linear test (GLT)

$$H_0: \beta_3 - \beta_5 = 0$$

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

- F -test for composite null hypothesis

$$H_0: \beta_3 = \beta_4 = \beta_5$$

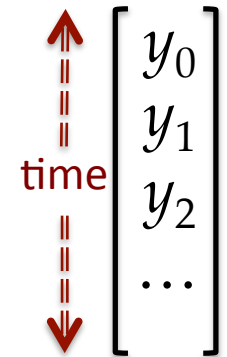
$$H_0: \beta_3 = \beta_4 = \beta_5 = 0$$

- Omnibus or Full F -test for the entire model

$$H_0: \text{all } \beta_i \text{ values } \underline{\text{of interest}} \text{ are } 0$$

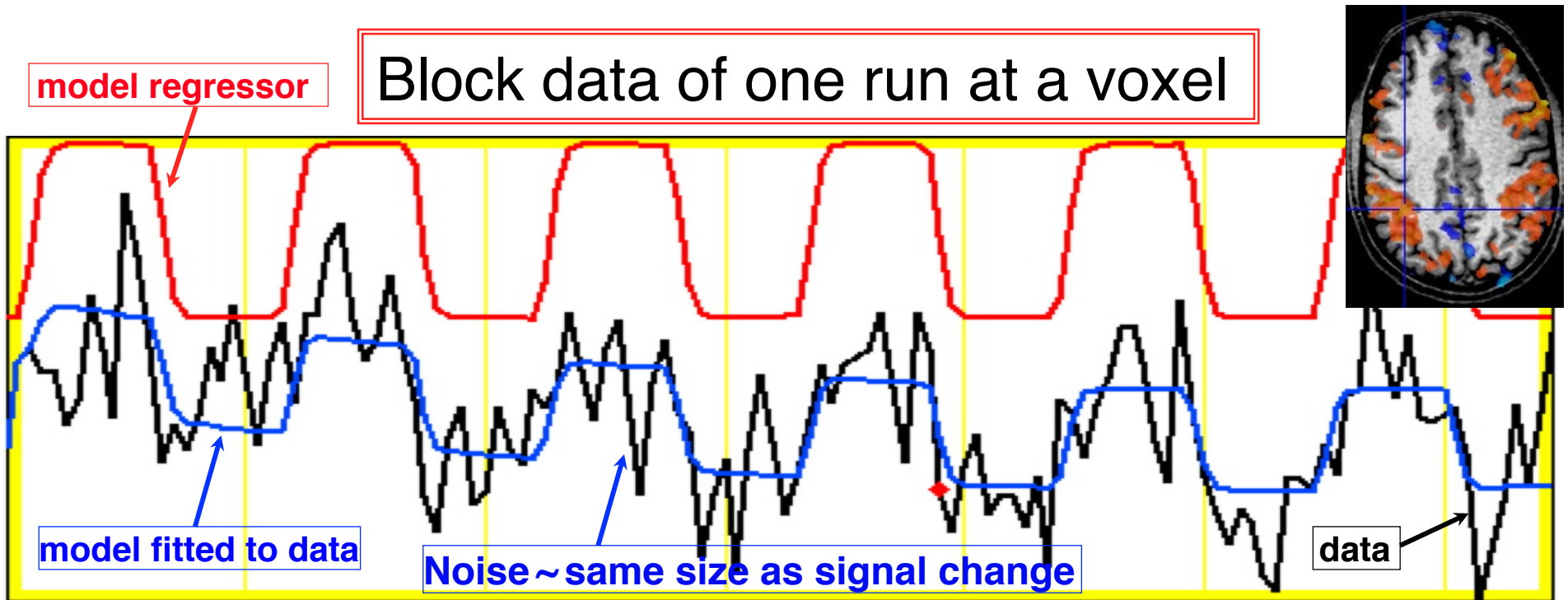
Linear Model with fMRI

- Time series regression: data vector \mathbf{y} is time series = all values from *one* voxel throughout multiple image acquisitions (TRs)
- Regressors: idealized BOLD response curves
 - We can only find what we're looking for
 - 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
 - Regressor construction requires decisions
 - Don't want to **over-fit** or **under-fit** data
- Same model matrix \mathbf{X} for all voxels in the brain
 - Simultaneously solve all the models (1 for each voxel)
 - Voxel-wise analysis = massively univariate method



FMRI Data

- Data partition: **Data = Signal + Noise**
 - **Data** = acquisition from scanner (voxel-wise time series)
 - **Signal** = BOLD response to stimulus; effects of interest + no interest
 - **We don't actually know the 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 repeated trials and **basis functions**
 - Of interest: effect size (response amplitude) for each task: **beta**
 - Of no interest: baseline, slow drifts, head motion effects, ...
 - **Noise** = components in data that interfere with signal
 - Practically: the part of the data we can't explain with the model
 - Will have to 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 + response₁ + ... + response_k + noise**
- How to construct the regressors of interest (responses)?



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

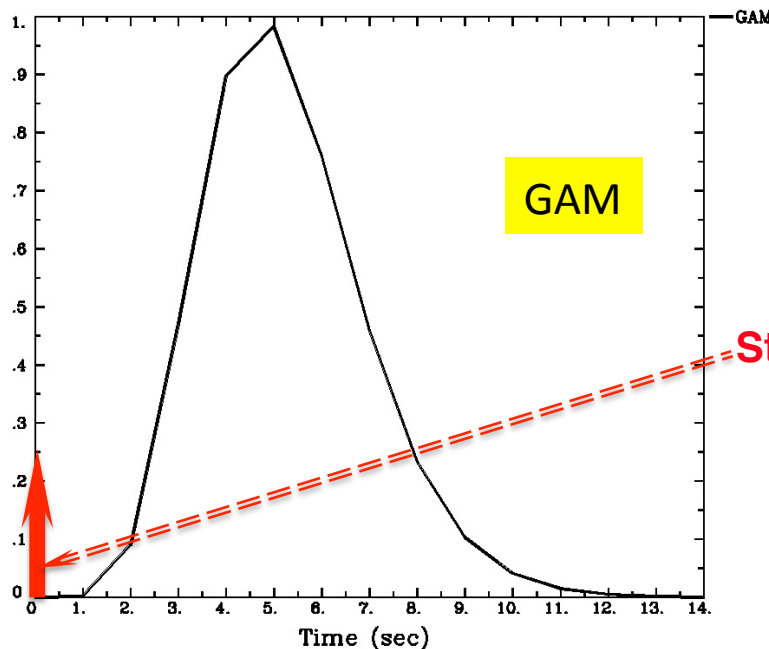
- This is “best” voxel; most voxels are not fitted as well as this
- Data drifts downwards – this effect is captured in the model fit by baseline drift regressors
 - If we did *not* model for drift, our fit would not be as good
- Activation amplitude and shape vary across blocks
 - Reasons why? We can only guess
 - Habituation? Attention? Noise?

BOLD Response

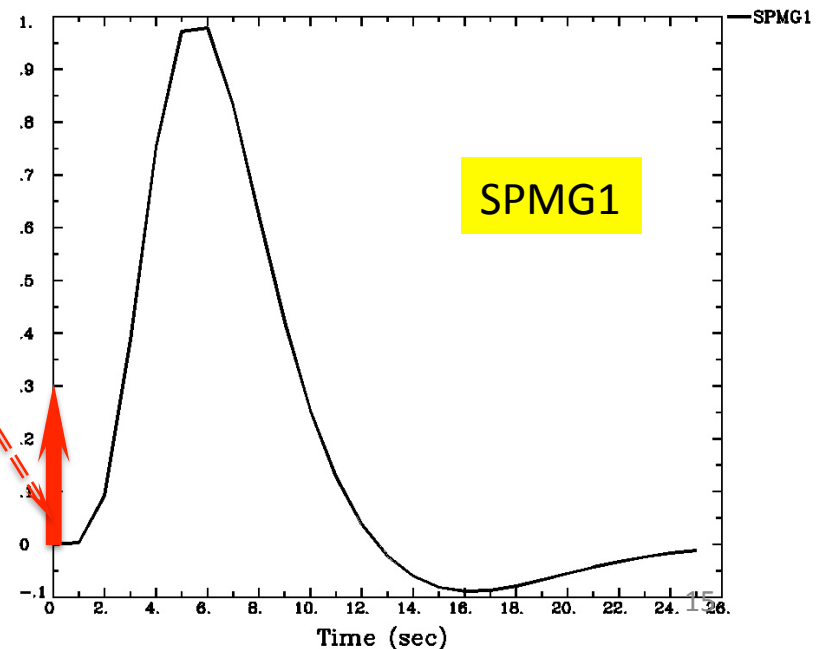
- Hemodynamic response (**HDR**)
 - Brain+FMRI response to stimulus / task / condition
 - Indirect measure of neural response: brain activation → changes in blood oxygen → changes in FMRI signal
- Hemodynamic response function (**HRF**)
 - Mathematical formulation / idealization of HDR for *one* full stimulus interval
 - HRF bridges between neural response (what we like) and BOLD signal (what we measure)
- How to build the bridge?
 - Most simple: Assume a fixed-shape (idealized) HRF
 - Most complex: No assumption about HDR shape
 - Basis function expansion of HRF shape and size
 - 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 an **instantaneous** (very short) stimulus: impulse response function (**IRF**)
 - GAM(p, q): $h(t) = t^p \exp(-t/q)$ for power p and time q
 - Sample IRF: $h(t) = t^{8.6} \exp(-t/0.547)$ [MS Cohen, 1997]
 - A variation: SPMG1 (undershoot is added in)
 - Build HRF based on presumed IRF through **convolution**
 - Combine IRF $h(t)$ with stimulus timing $S(t)$: $x(t) = h(t) \otimes S(t)$

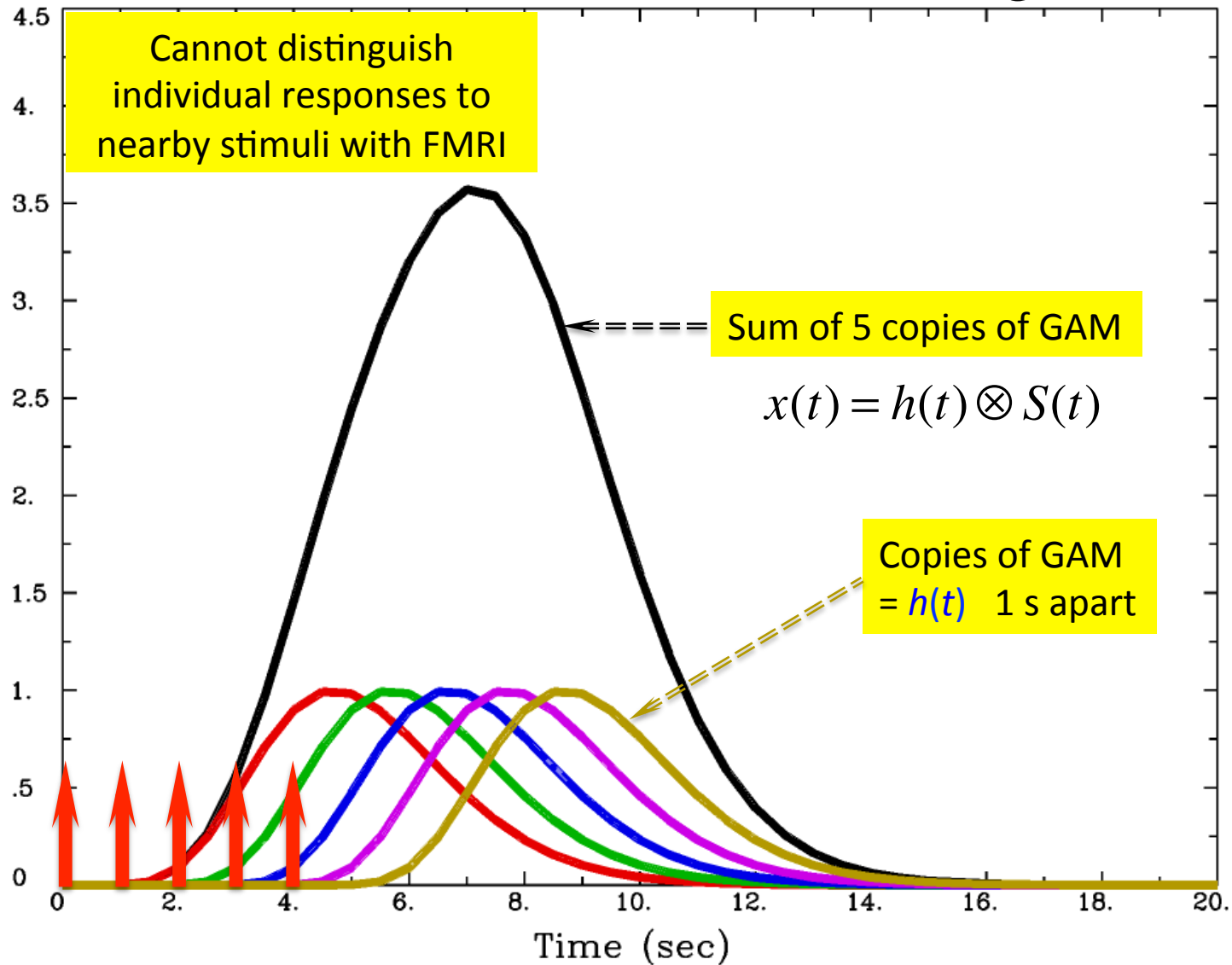


Short
Stimulus
(≤ 1 s)



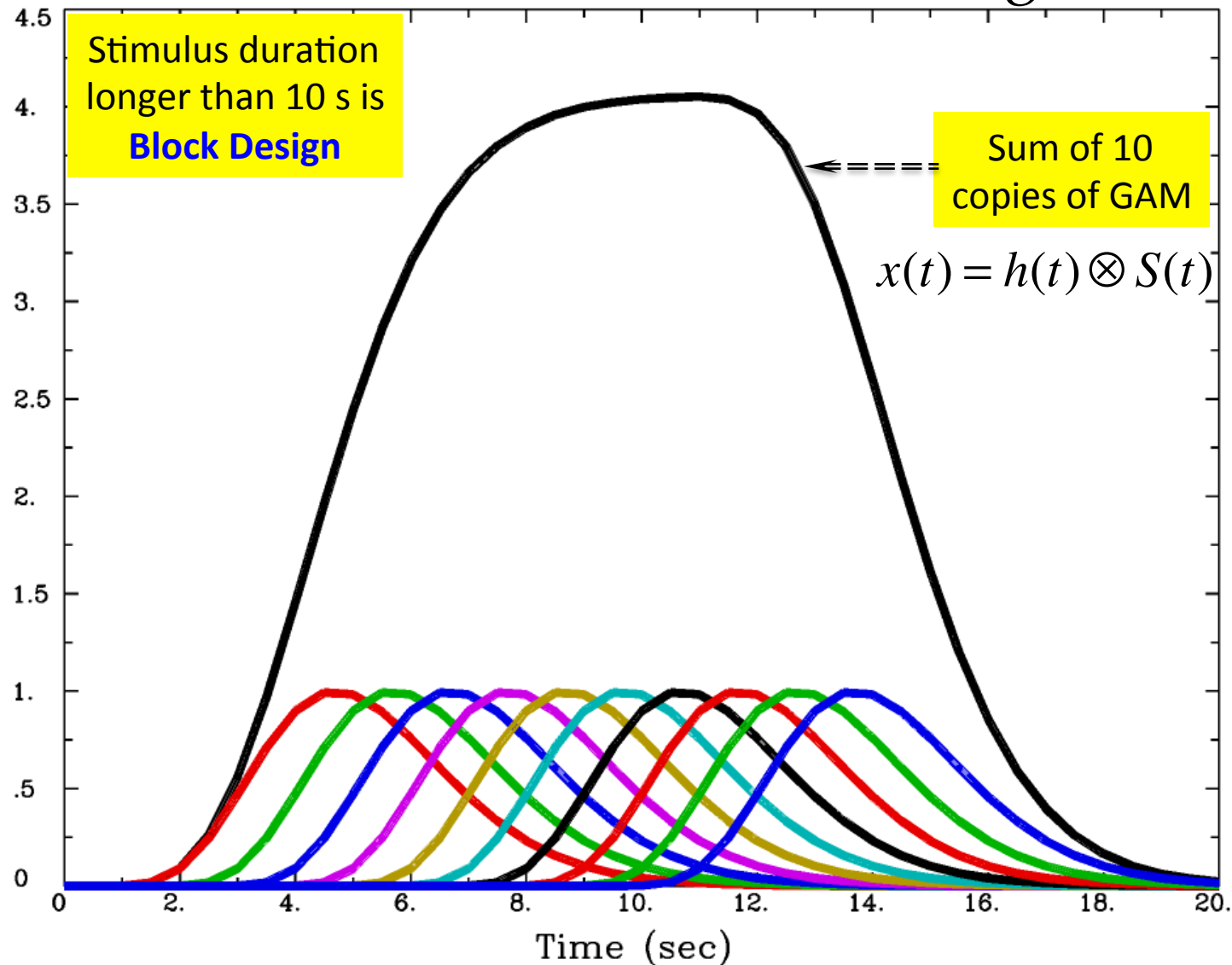
Fixed-Shape HRF – 5 s Stimulus

- Combine IRF $h(t)$ with stimulus timing $S(t)$:



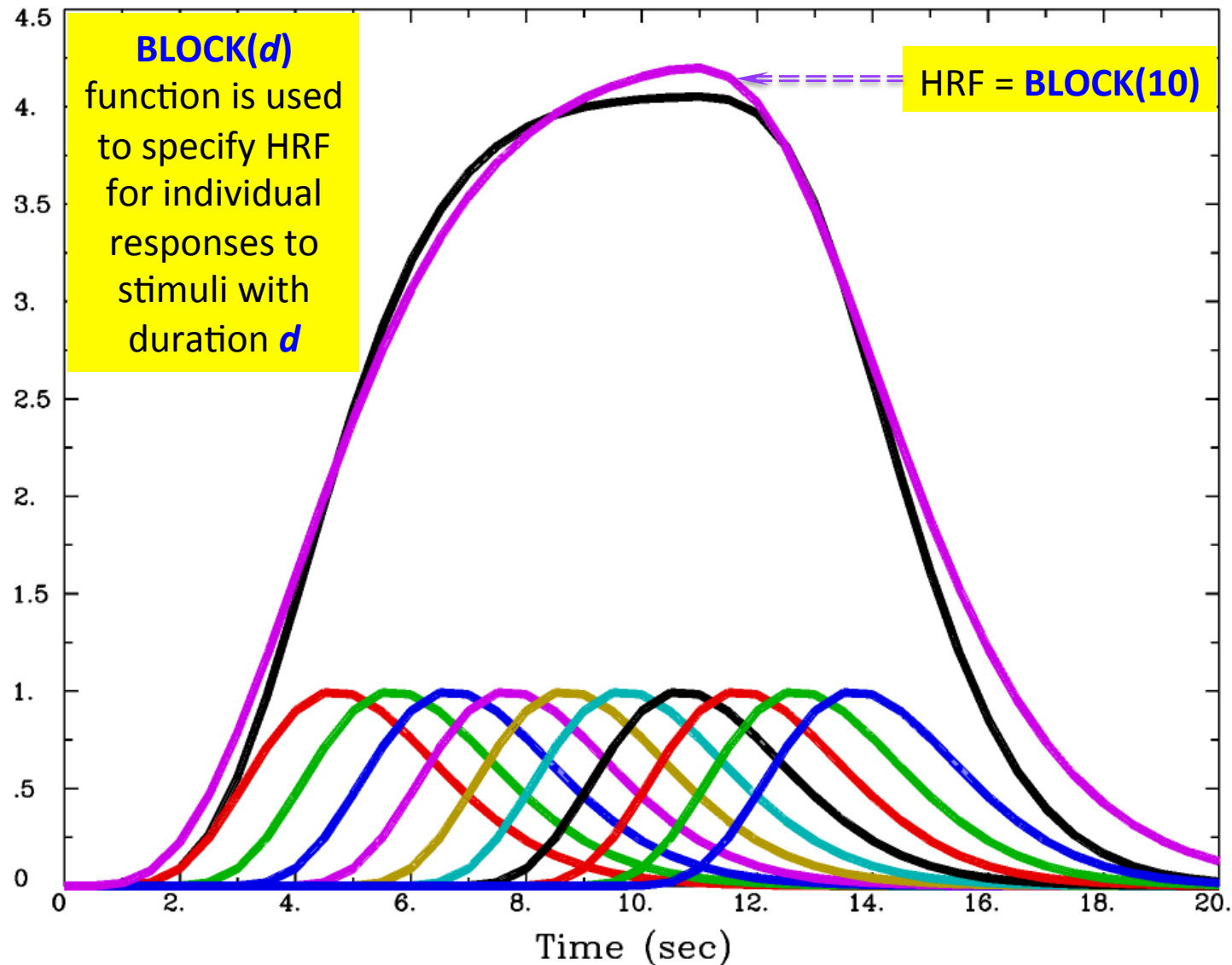
Fixed-Shape HRF – 10 s Stimulus

- Combine IRF $h(t)$ with stimulus timing $S(t)$:



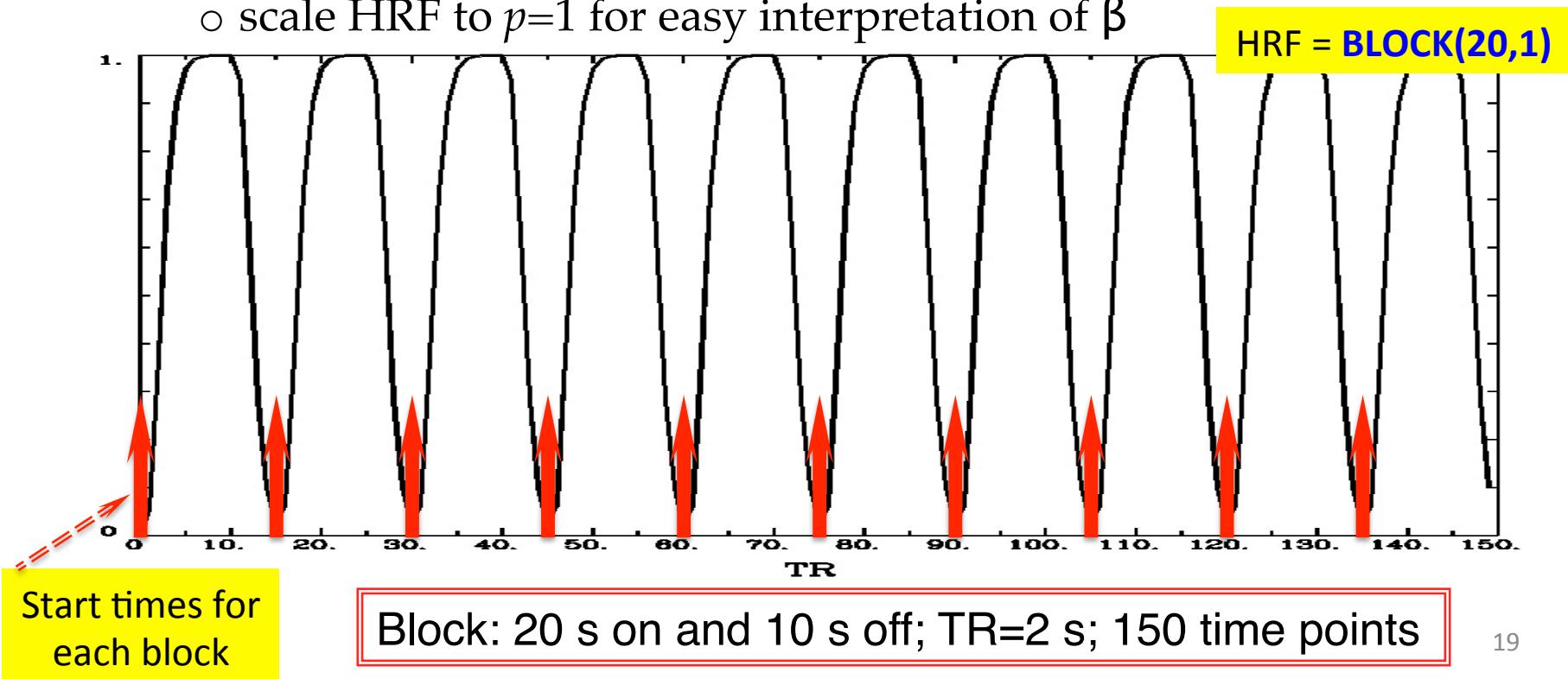
Fixed-Shape HRF – 10 s Stimulus

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



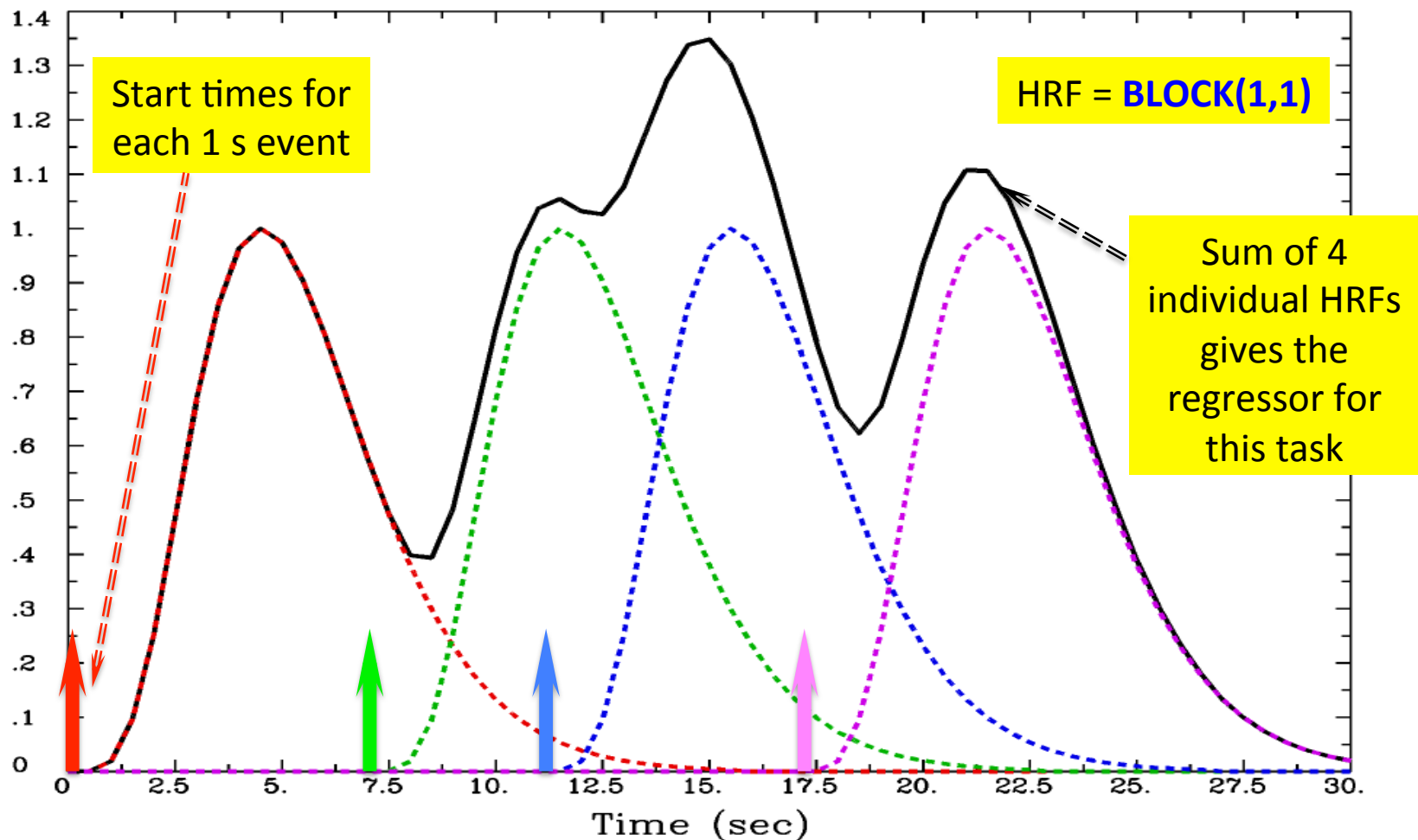
Fixed-Shape HRF for Block Design

- Assuming a fixed shape $h(t)$ for IRF to an **instantaneous** (very short) stimulus
 - For each block, $h(t)$ is convolved with **stimulus timing** and **duration** (d) to get idealized response (temporal pattern) as an explanatory variable (regressor): $\text{HRF} = \text{BLOCK}(d,p)$
 - **Equivalent to adding up a series of consecutive events**
 - scale HRF to $p=1$ for easy interpretation of β



Fixed-Shape HRF for Event-Related Design

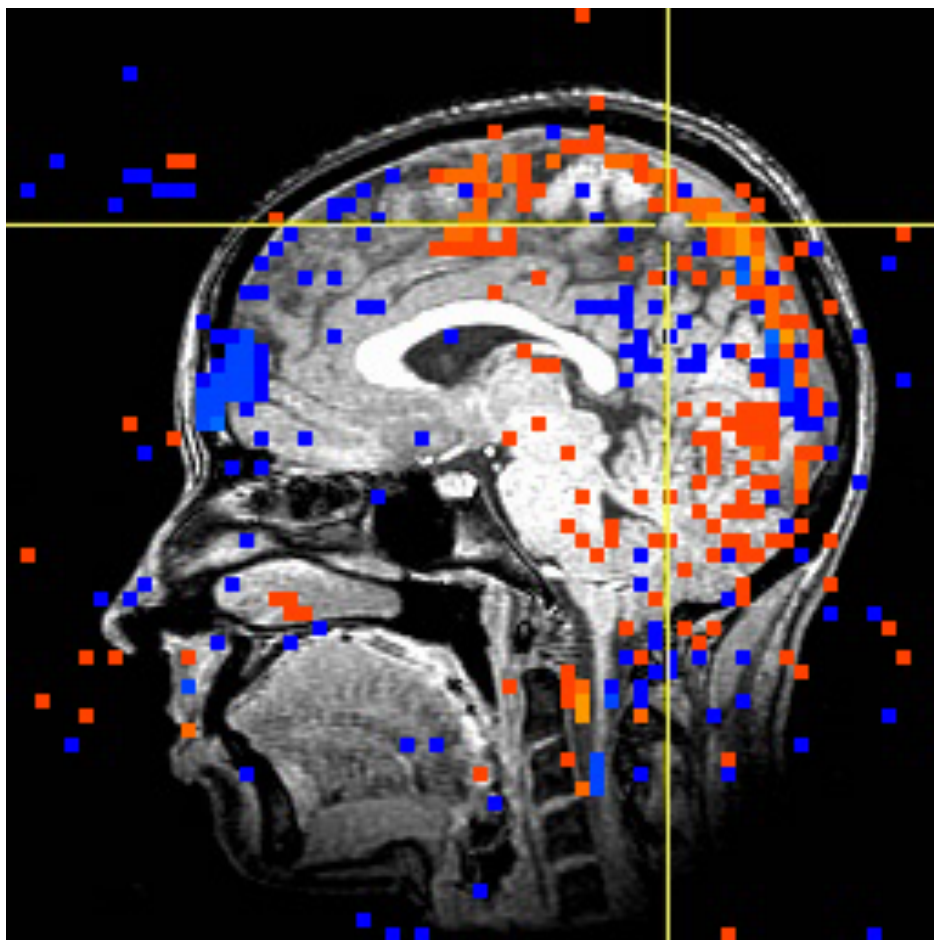
- The **BLOCK** HRF shape is useful with event-related experiment designs
- Just use a short duration, such as 1 second
- Real experiments have more than 4 task repetitions!



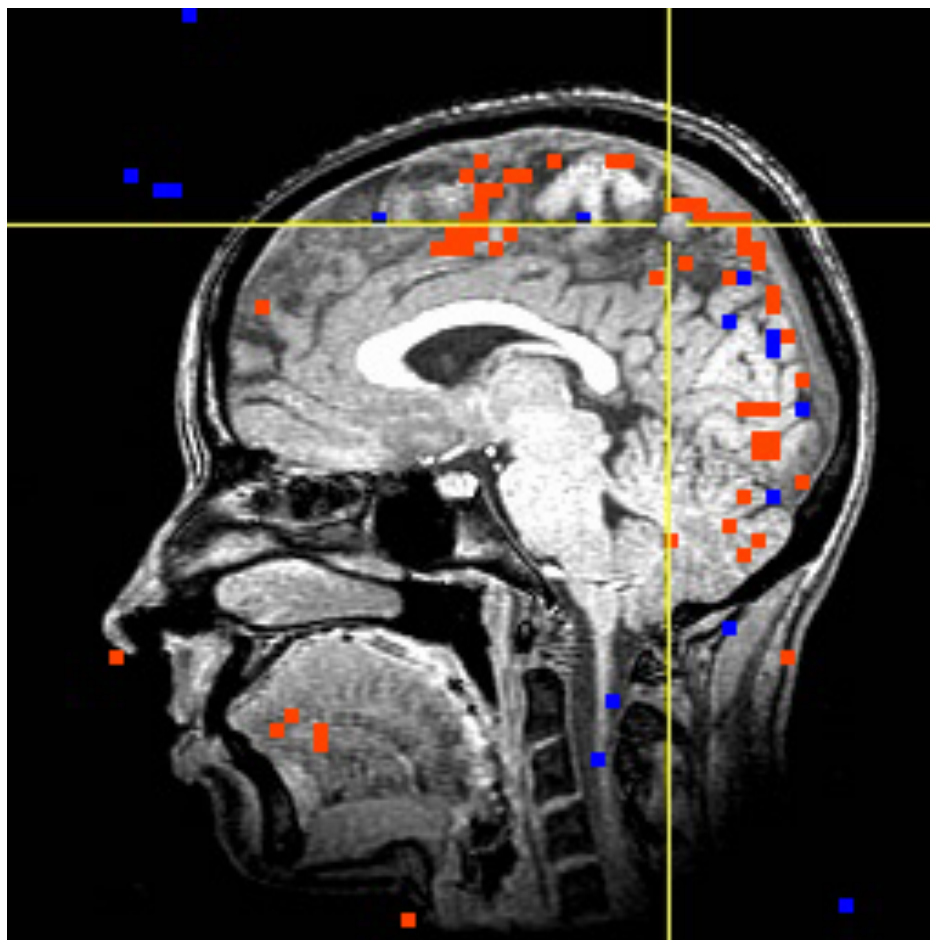
Linear Model with Fixed-Shape HRF

- FMRI data = **baseline** + **drift** + **other effects of no interest** + **response₁** + ... + **response_k** + noise
- 'baseline' = **baseline** + **drift** + **other effects of no interest**
 - Drift: physiological effect, tiny motions, scanner fluctuations
 - Data = 'baseline' + **effects of interest** + noise
 - **Baseline condition (and drift)** is treated in AFNI as **baseline model**, an additive effect, not an effect of interest (*cf.* **SPM/FSL**)!
 - **Baseline+drift+...** also need parameters in the model fit
- $y_i = \alpha_0 + \alpha_1 t_i + \alpha_2 t_i^2 + \beta_1 x_{1i} + \dots + \beta_k x_{ki} + \dots + \varepsilon_i$ [$i = \text{time}$]
- $y = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, $\mathbf{X} = [1, t, t^2, x_1, x_2, \dots, x_k, \dots]$ [vector format]
- In AFNI **baseline + slow drift** is modeled with polynomials
 - A longer run needs a higher order of polynomials
 - One polynomial order per 150 sec is the default in AFNI
 - With m runs, m sets of polynomials needed to allow for temporal discontinuities across runs
 - $m(p+1)$ columns for **baseline+slow drift** with p -order polynomials
- Other effects of no interest: head movement estimates

Stimulus Correlated Motion = Bad



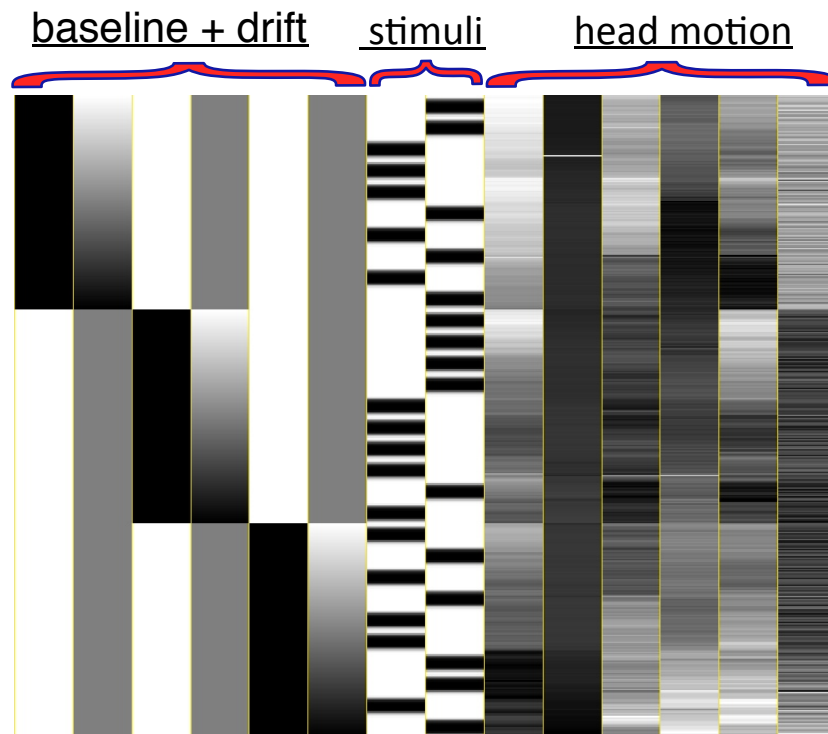
Activation map without
using movement
estimates as regressors



Activation map when
using movement
estimates as regressors

Design Matrix with Fixed-Shape HRF

- Voxel-wise (massively univariate) linear model: $y = X\beta + \epsilon$
 - X : explanatory variables (regressors) – **same** across voxels
 - y : data (time series) at a voxel – **different** across voxels
 - β : regression coefficients (effects) – **different** across voxels
 - ϵ : anything we can't account for – **different** across voxels
- Visualizing design matrix $X = [1, t, x_1, x_2, \dots, x_k, \dots]$ in grayscale image

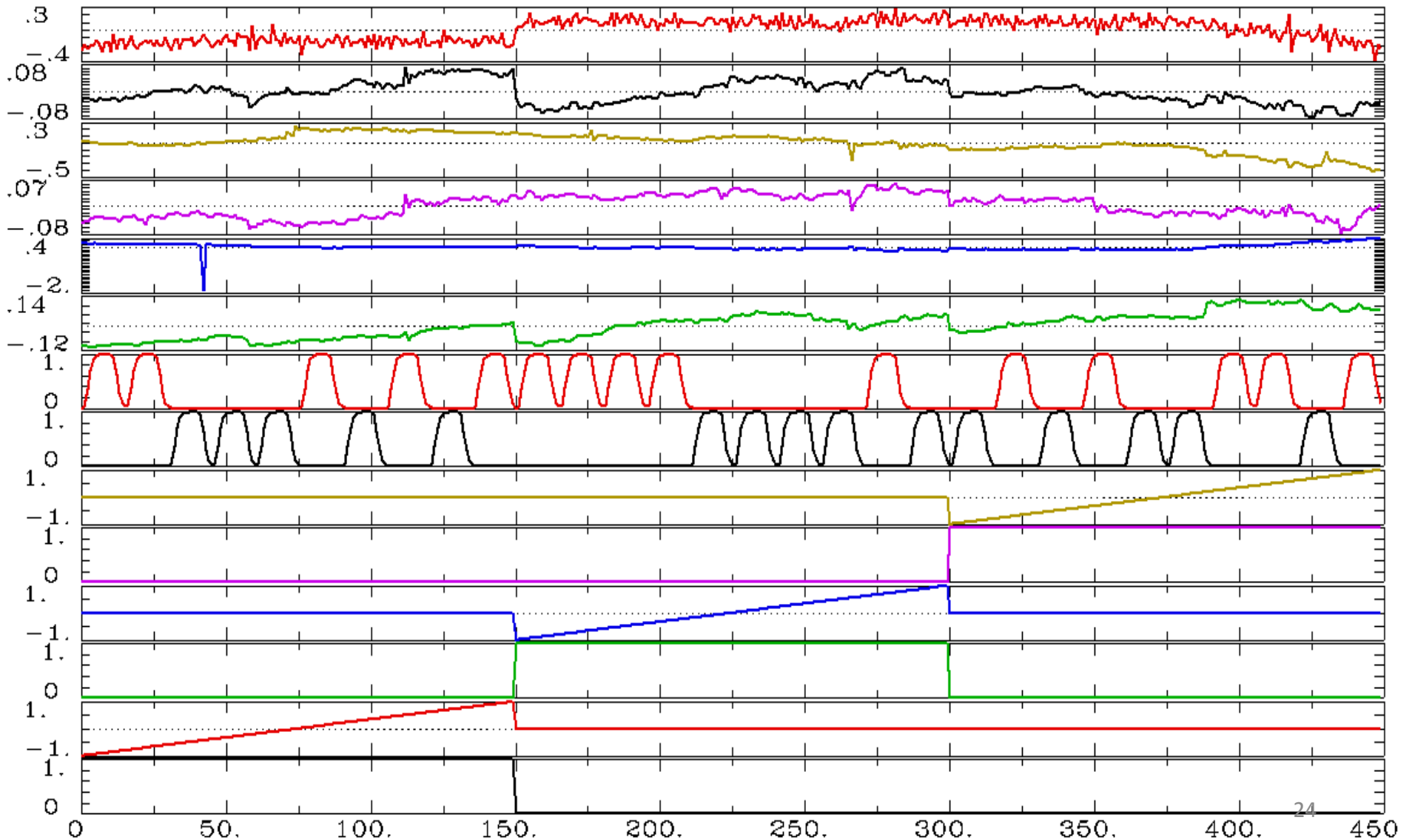


- 6 drift effect regressors
 - linear baseline
 - 3 runs x 2 parameters/run
- 2 regressors of interest
 - that is, relevant to brain activity
- 6 head motion regressors
 - 3 rotations + 3 shifts

Black = bigger numbers
White = smaller numbers
Each column of X scaled separately

Design Matrix with Fixed-Shape HRF

- Visualizing same design matrix $X = [1, t, x_1, x_2, \dots, x_k, \dots]$ in graphs



Assessing Fixed-Shape HRF Approach

- Used 99% of time: Why is it popular?
 - Assume brain responds with **same shape** across 4 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
 - Strong assumption about **four** levels of shape information?
 - **Easy to handle and think about**: one value per effect / task
 - Works relatively well
 - Block design: shape usually not important due to accumulating effects (modeled via convolution) of consecutive events
 - Really 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
 - Improved modeling

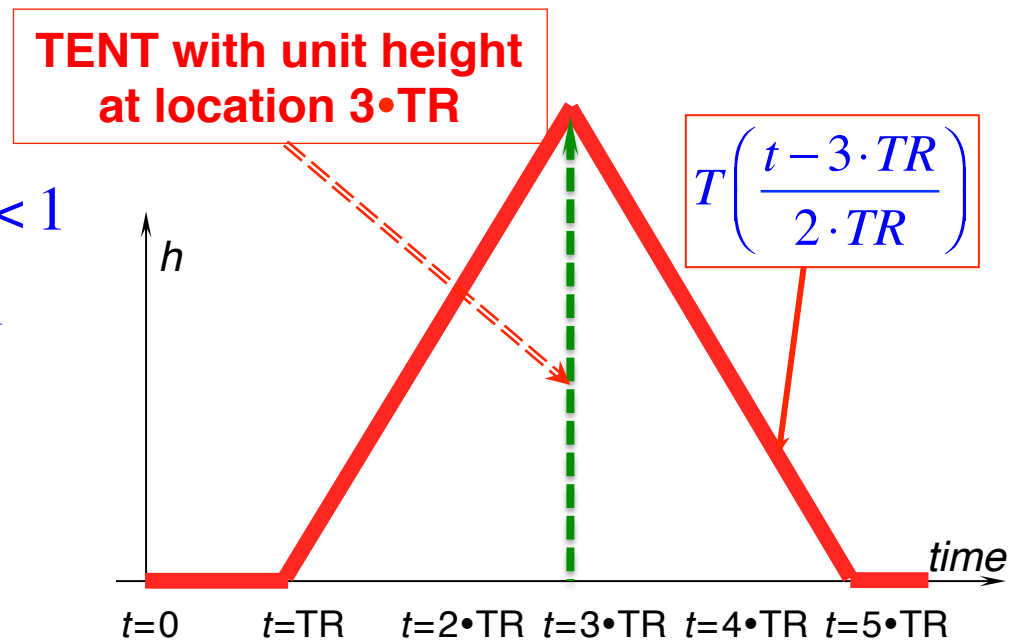
Alternative: No Constraint on HRF Shape

- TENT expansion of HRF
 - 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 (β_s) 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

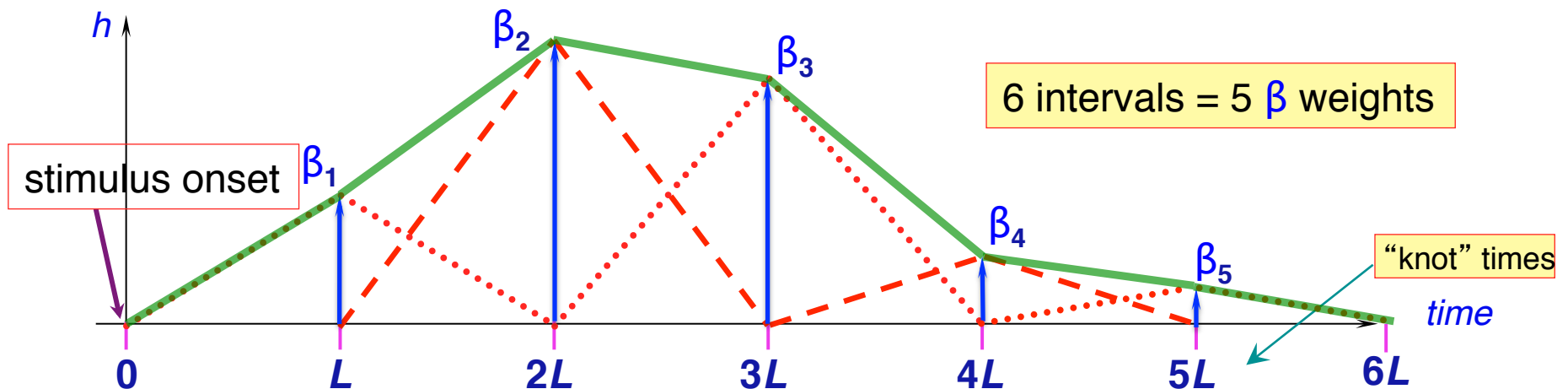
Cubic splines (CSPLIN)
are also available in AFNI



Σ Tent Functions = Linear Interpolation

- 5 equally-spaced TENT functions = linear interpolation between “knots” with $\text{TENTzero}(b,c,n) = \text{TENTzero}(0,12,7)$

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



- TENT parameters are easily interpreted as function values (e.g., L : TENT radius; β_2 = response (TENT height) at time $t = 2L$ after stimulus onset)
- Relationship of TENT spacing L and TR ($L \geq \text{TR}$), e.g., with TR=2s, $L=2, 4$ s
- In `uber_subject.py` or `3dDeconvolve` with $\text{TENTzero}(0, D, n)$, specify duration (D) of HRF and number (n): radius $L = D / (n-1)$ with $(n-2)$ full tents, each TENT overlaps half tent with two neighboring ones.
 - In above example, $D=12$ s, then $L=2$ s $n=7$; covering 12s; $\text{TENTzero}(0,12,7)$

Tent Functions Create the HRF

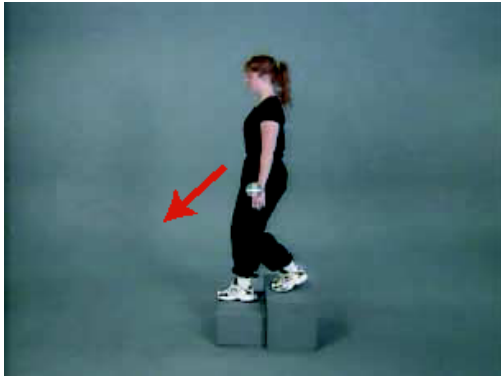
- And then the HRF is repeated for all stimuli of the same type
- In the example on the last slide, the HRF has 5 parameters (β s) to be estimated
- The β s determine the amplitude (percent signal change) *and* the shape of the HRF
- Each voxel in each subject gets a separate HRF shape now, not just a separate amplitude
 - And if there are multiple types of tasks, each task gets a separate shape
- Stimulus times do *not* have to be exactly on the TR grid

Modeling with TENTs - Example

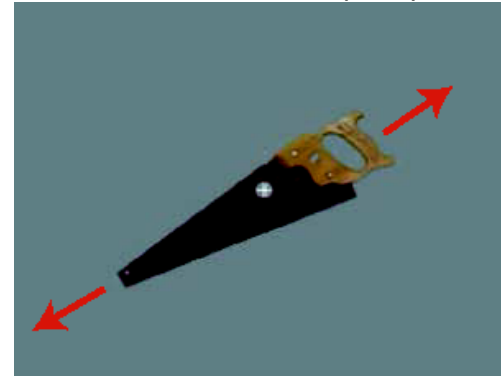
- Event-related study (Beauchamp et al., J Cogn Neurosci 15:991-1001)
 - 10 runs, 136 time points per run, TR=2 s
 - Two factors
 - Object type: **human** vs. **tool**
 - Object form in videos: **real image** vs. **points**
 - 4 types (2x2 design) of stimuli (short videos)
 - Tools moving (e.g., a hammer pounding) - [ToolMovie](#)
 - People moving (e.g., jumping jacks) - [HumanMovie](#)
 - Points outlining tools moving (no objects, just points) - [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 whole-body motion (HM)



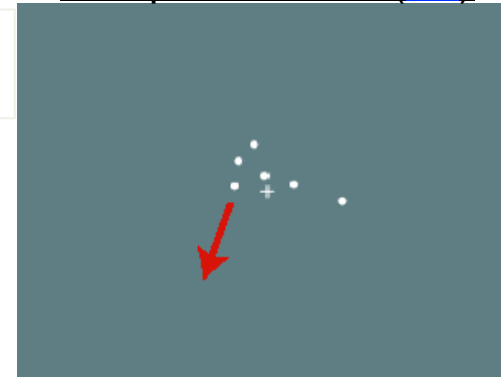
Tool motion (TM)



Human point motion (HP)



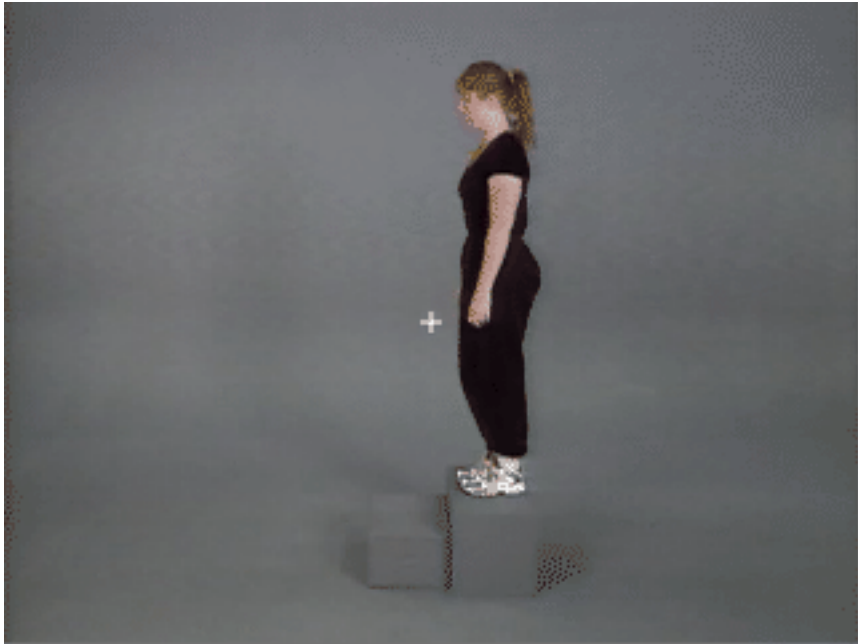
Tool point motion (TP)



From Figure 1
Beauchamp et al. 2003

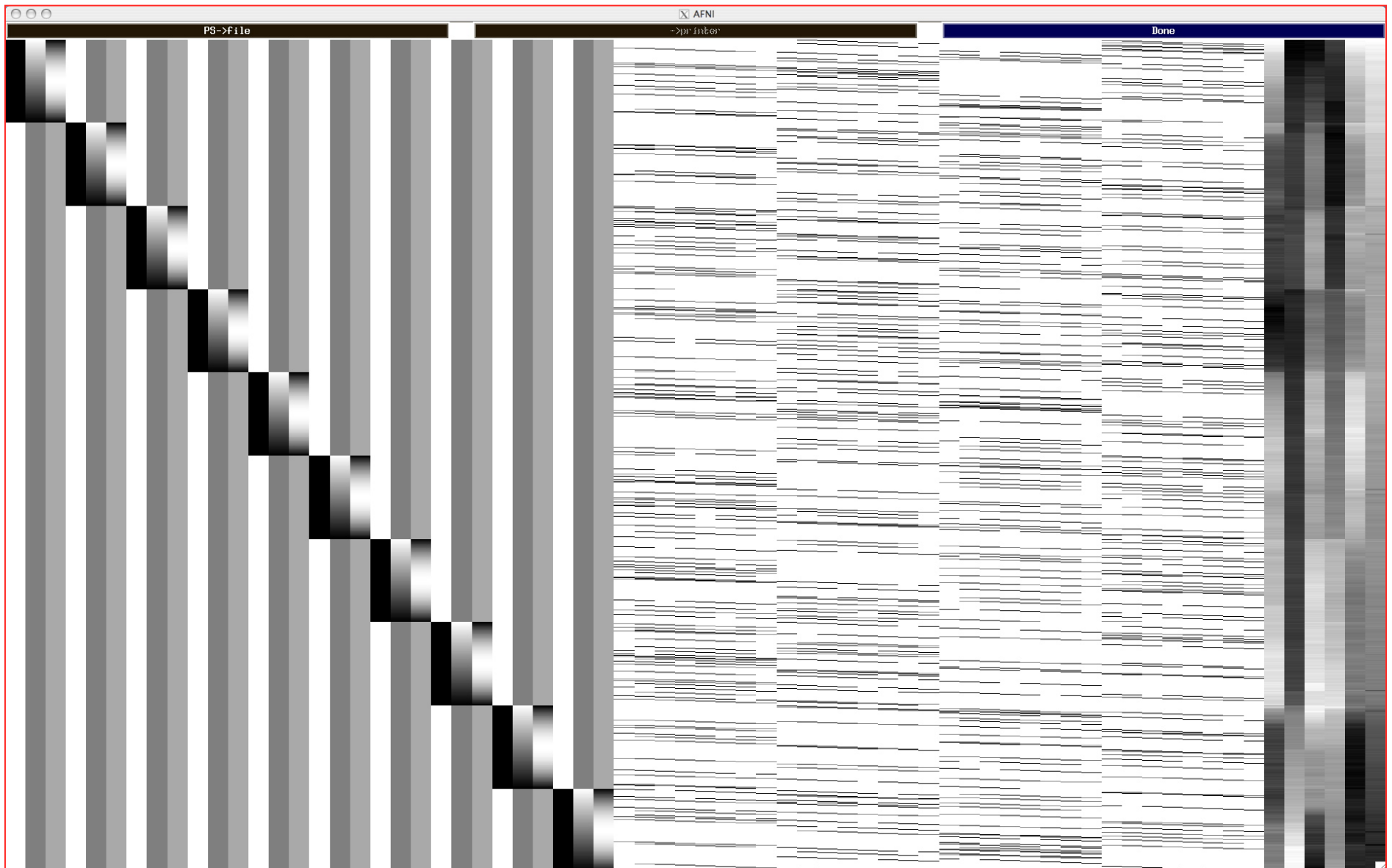
Hypotheses to test:

- Which areas are differentially activated by any of these stimuli (main effect)?
 - point motion versus natural motion? (type of image)
 - human-like versus tool-like motion? (type of motion)
- Interaction effects?
 - Point: human-like versus tool-like? Natural: human-like versus tool-like?
 - Human: point versus natural? Tool: point versus natural?



Each video is only shown once (2 seconds)

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



Baseline + quadratic trend for 10 runs

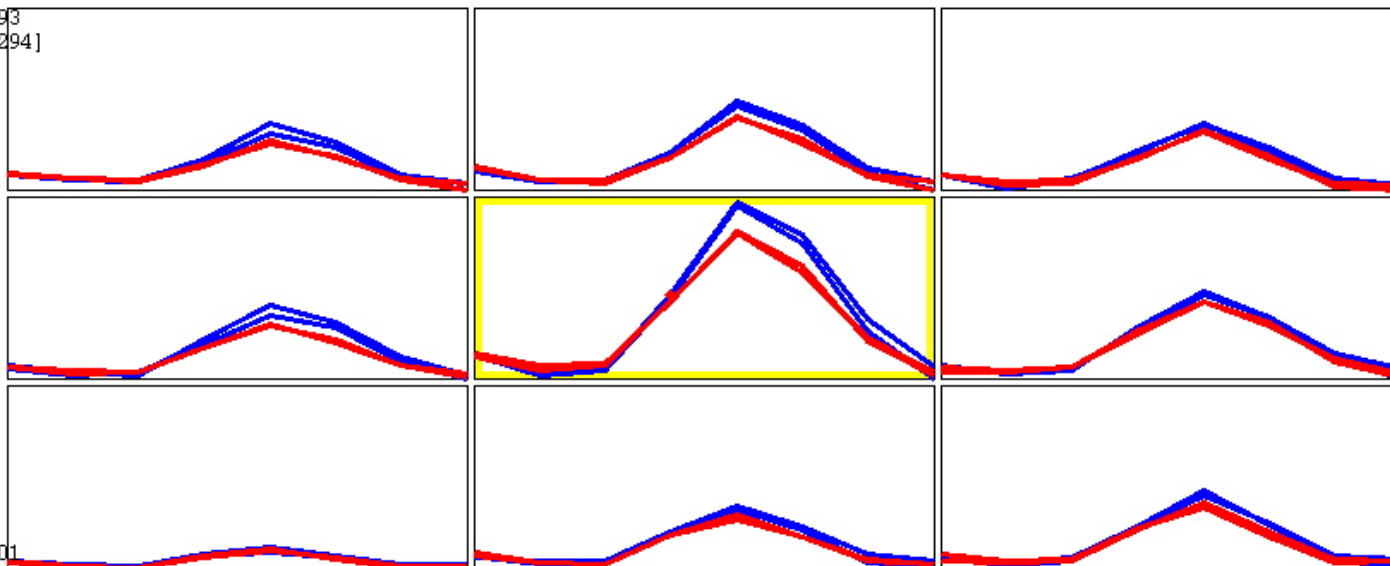
7 tents per condition \times 4 conditions

head motion³⁴

Results: Humans vs. Tools



5. 647593
[+6. 235294]



-0.587701

- Color overlay:
Human vs Tool

$$(\beta_{HM} + \beta_{HP} - \beta_{TM} - \beta_{TP})$$

- **Blue**
(upper) :
Human

- **Red**
(lower) :
Tool

No Constraint on HRF Shape = **Deconvolution**

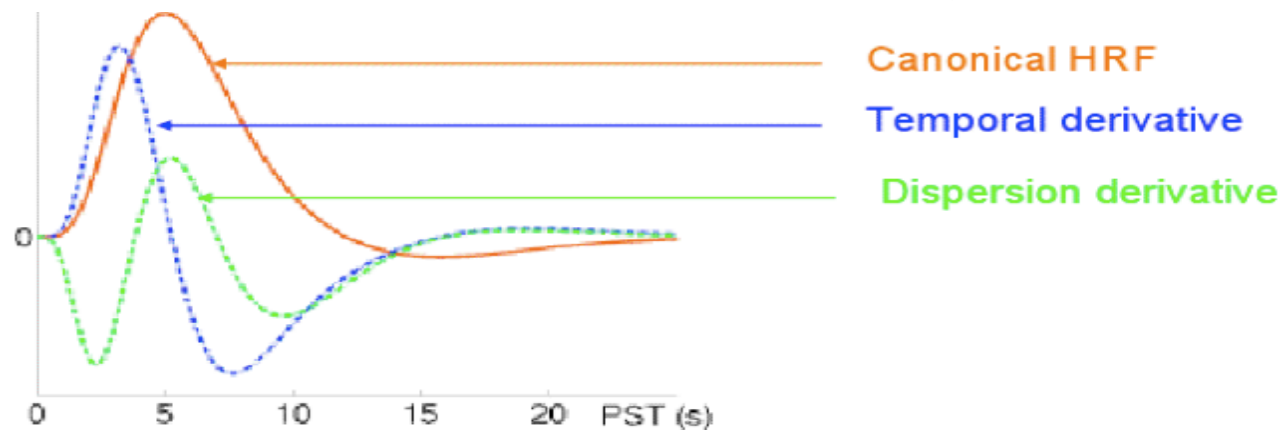
- Deconvolution perspectives: inverse process of convolution
 - **HRF** \otimes **stimulus** = **unit BOLD response**
 - Like multiplication, we have to know two and estimate the 3rd
 - Fixed-shape approach: Convolution + regression
 - Known: **HRF shape**, **stimulus**
 - Use convolution to create **regressors** (hidden from user inside 3dDeconvolve program)
 - Response strength (β) estimated via linear model with programs 3dDeconvolve or 3dREMLfit
 - Shape estimation: Deconvolution + regression
 - Known: **stimulus** + **BOLD response**; unknown: **impulse response**
 - **HRF** \otimes **stimulus** = **BOLD response** (note: HRF, not IRF)
 - **HDR** estimated as a linear combination of multiple basis functions: TENTS
 - Each TENT \otimes stimulus = one regressor column
 - Deconvolution: **HRF** = a set of β s estimated via regression

No Constraint on HRF Shape: Pros + Cons

- What is the approach good at?
 - Usually for event-related experiments, but can be used for BLOCK
 - Multiple basis functions for blocks: within-block attenuation with time
 - Likely to have more accurate estimate on HDR shape across
 - subject
 - conditions / tasks
 - brain regions
 - Likely to have better model fit (the goal in the sample experiment)
 - Likely to be statistically more powerful on test significance
 - For block design, may detect within-block attenuation
 - Cross-block attenuation?
- Why is the approach not popular?
 - Difficult to summarize at group level [see the program 3dMVM]
 - Multiple parameters (β_s) per task condition, instead of just one
 - More regressors than alternatives: DoF's per subject
 - Risk of highly correlated regressors: Multicollinearity
 - May need to reduce the number of basis functions
 - Over-fitting: picking up something (head motion) unrelated to HDR₃₇

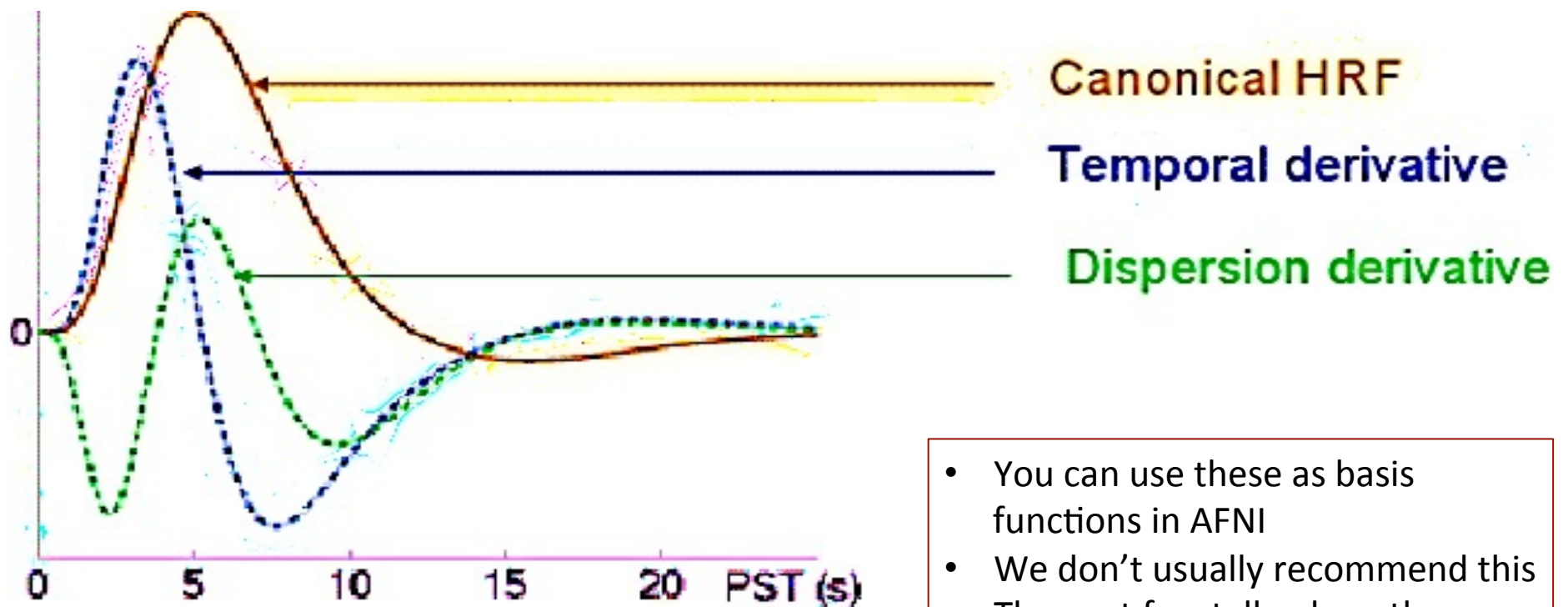
Intermediate Approach: SPMG1 / 2 / 3

- Use just a few basis functions
 - Constrain the HDR shape with a principal basis function
 - SPMG1 (similar to GAM in AFNI): $e^{-t}(a_1 t^{p1} - a_2 t^{p2})$ where
 $a_1 = 0.0083333333$ $p1 = 5$ (main positive lobe)
 $a_2 = 1.274527e-13$ $p2 = 15$ (undershoot part)
 - 2 or 3 basis functions: parsimonious, economical
 - SPMG1+SPMG2+SPMG3
 - SPMG2: temporal derivative capturing differences in peak latency
 - SPMG3: dispersion derivative capturing differences 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 this
- The next few talks show the details of how to choose the basis functions for the HRF

Multicollinearity

- Voxel-wise regression model: $y = X\beta + \varepsilon$
 - Regressors in design matrix $X = [1, t, t^2, x_1, x_2, \dots, x_k, \dots]$
- Multicollinearity problem
 - Two or more regressors highly correlated
 - Difficult or impossible to distinguish the effects among these regressors (*i.e.*, get reliable β estimates)
- Multicollinearity scenarios
 - Collinearity - $x_i = \lambda x_j =$ **model specification error**; *e.g.*, 2 identical regressors (mistake in stimulus timing specifications)
 - Exact multicollinearity: linear dependence among multiple regressors = **faulty design** (rare)
 - High degree of correlation (+ or -) among regressors = **design problem** (*e.g.*, cue + movie watching)
 - Too many basis functions in response model
- Diagnosis tools: `ExamineXmat.R`, `timing_tool.py`, `xmat_tool.py`

Serial Correlation in Residuals

- Why temporal correlation?
 - In the residuals / noise (not the time series data)
 - Short-term physiological effects (breathing, heartbeat)
 - Other unknown reasons (scanner issues?)
- What is the impact of temporal correlation?
 - With white noise assumption, β s are unbiased, but the statistics tend to be inflated
 - Little impact on group analysis – if only using β s from subjects
 - May affect group analysis if considering effect reliability, as in AFNI's 3dMEMA program (where β s and t s are used)
- Approach in AFNI
 - ARMA(1,1) noise model for residual time series correlation
 - Slightly different from other packages
 - Described in the Advanced Regression talk: 3dREMLfit

Dealing with Multiple Runs per Subject

- Possible approaches
 - Analyze each run separately: AFNI, FSL
 - Have to have enough task repetitions per run
 - Can test **cross-run** difference (trend, habituation) at **group** level
 - Usually need to summarize multiple β 's before group analysis
 - Concatenate but analyze with separate regressors across runs for each condition type: AFNI, SPM
 - Can test **cross-run** difference (trend, habituation, etc.) at **both individual and group** level
 - Still need to summarize multiple β 's before group analysis
 - Concatenate but analyze with same regressor across runs for each condition type: default in AFNI
 - Assume no attenuation across runs
- Cross-block (or cross-event) attenuation
 - Method: IM or AM regression models
 - *cf.* Advanced Regression talk

Percent Signal Change

- Why conversion / scaling for %? Comparable across subjects
 - MRI and BOLD data values don't have any useful physical / physiological meaning
 - Baseline is different across subjects (and possibly scaling)
 - It's the relative changes that can be compared across subjects
- AFNI approach
 - Pre-processing: data scaled by **voxel-wise** mean
 - % signal change relative to mean, not exactly to **baseline**
 - Difference is tiny: less than 5% (since BOLD effect is small)
 - Tied with modeling baseline as additive effects in AFNI
 - Sometimes baseline explicitly modeled: in SPM and FSL
 - Global mean scaling (multiplicative) for whole brain drift
 - Grand mean scaling for cross-subject comparison: not %
 - Global and grand mean scaling, although not usually practiced, can be performed in AFNI if desired

Lackluster Performance in Modeling

- **Essentially, all models are wrong, but some are useful**
(G.E.P. Box)
- Noisy data: too easy excuse!
- Regressors: idealized response model
 - We find what we're looking for
 - We may miss something when we fail to look for it
- Lots of variability across trials
 - **A**mplitude **M**odulation if behavioral data are available
 - Model each trial separately (**I**ndividual **M**odulation)
- Linearity assumptions
 - $\text{Data} = \text{baseline} + \text{drift} + \text{response1} + \text{response2} + \dots + \text{noise}$
 - When a trial is repeated, response is assumed same
 - Response for a block = linearity (no attenuation)
- Poor understanding of BOLD mechanism

Summary

- Basics of linear model
- FMRI data decomposition: three components
 - Baseline + slow drift; Effects of interest; Unknown
 - Effects of interest - understanding BOLD vs. stimulus: IRF
- Modeling with fixed-shape IRF: $GAM(p,q)$, $BLOCK(d,p)$
- Modeling with no assumption about IRF shape
 - $TENT(b,c,n)$ or $CSPLIN(b,c,n)$
- Modeling with one major IRF plus shape adjustment
 - SPMG1/2/3
- Other issues
 - Multicollinearity
 - Catenation
 - Percent signal change