Event-related fMRI

Klaas Enno Stephan

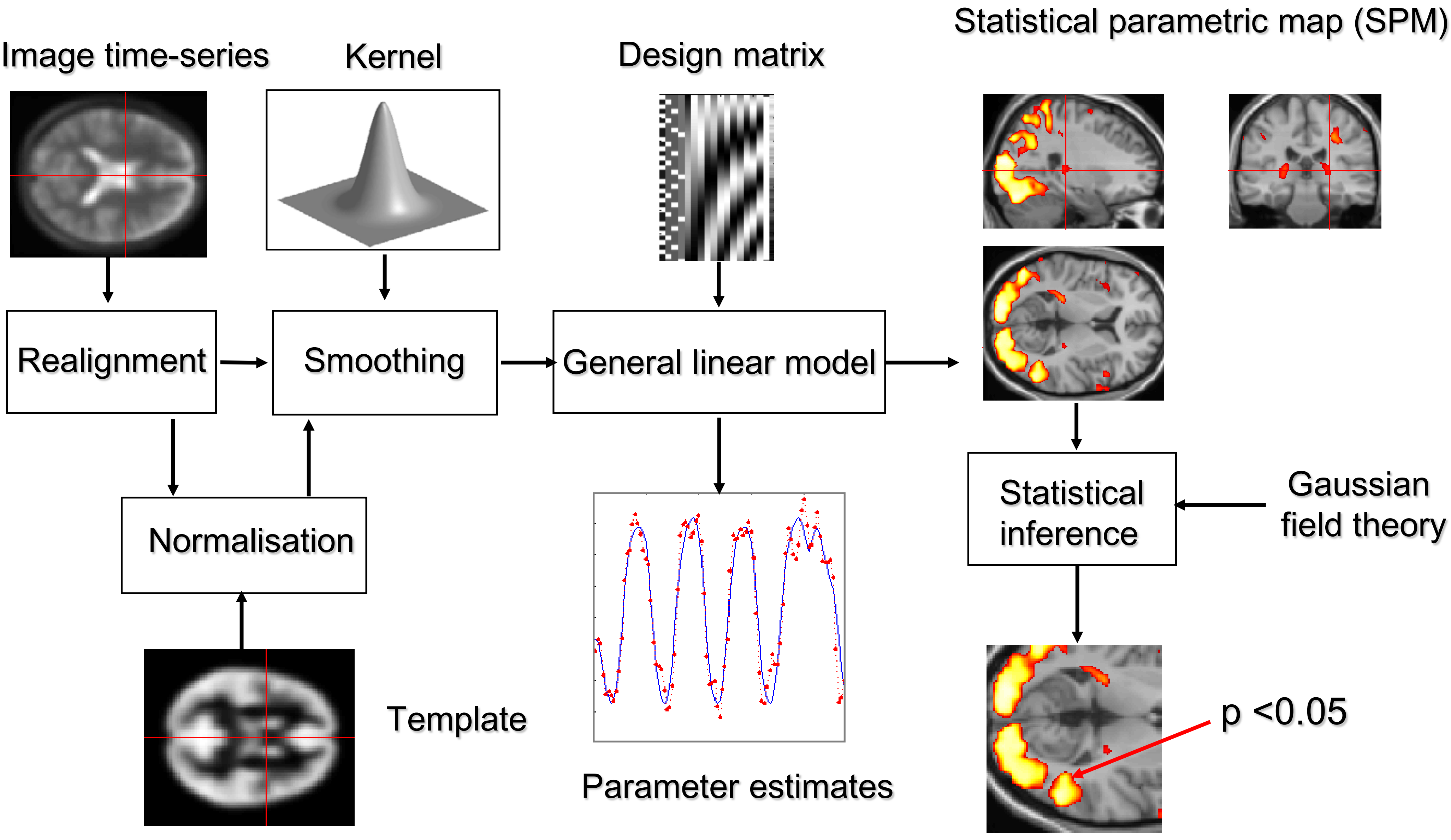
Translational Neuromodeling Unit (TNU)
Institute for Biomedical Engineering, University of Zurich & ETH Zurich

With many thanks for slides & images to:

FIL Methods group, Rik Henson and Christian Ruff

Methods & models for fMRI data analysis
23 October 2016

Overview of SPM



Overview

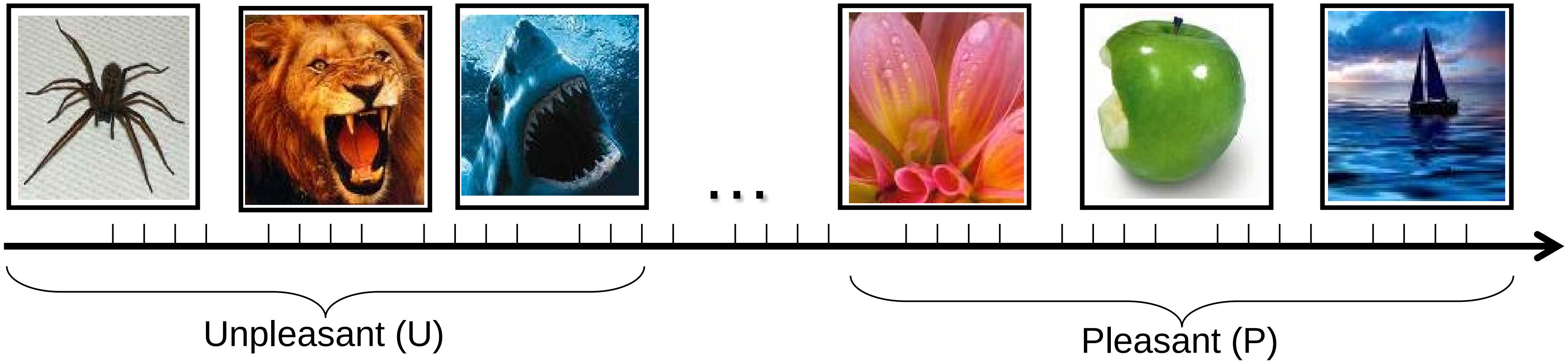
1. Advantages of er-fMRI
2. BOLD impulse response
3. General Linear Model
4. Temporal basis functions
5. Timing issues
6. Design optimisation

Advantages of er-fMRI

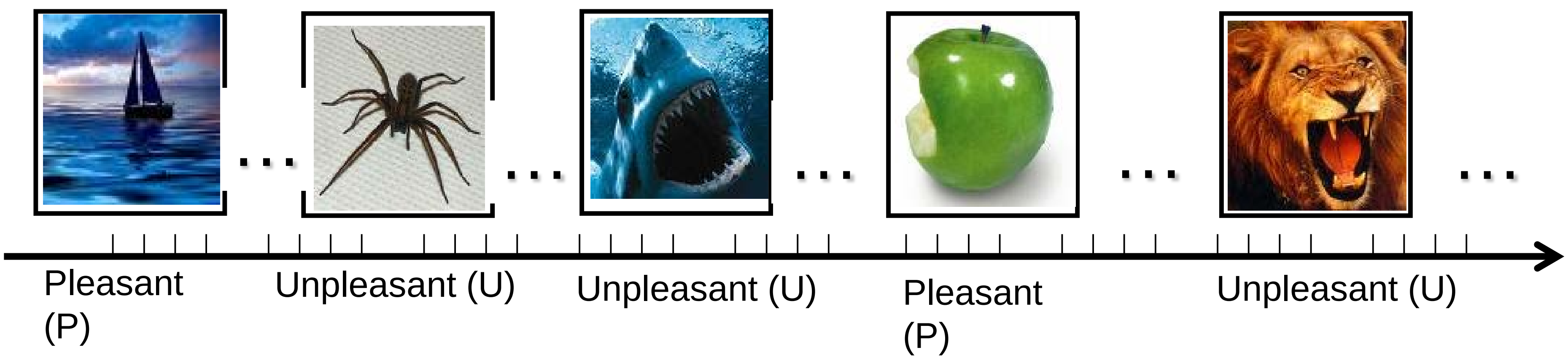
1. Randomised trial order
   cf. confounds of blocked designs

er-fMRI: Stimulus randomisation

Blocked designs may trigger expectations and cognitive sets



Intermixed designs can minimise this by stimulus randomisation

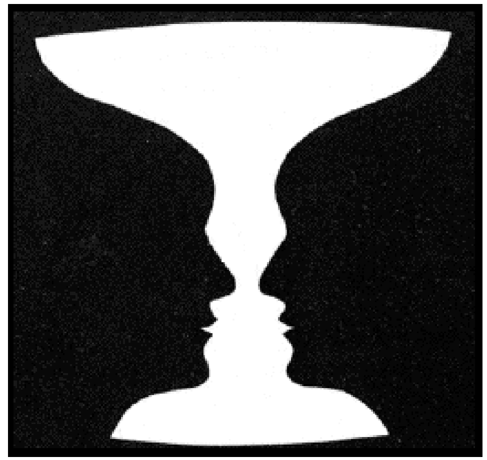


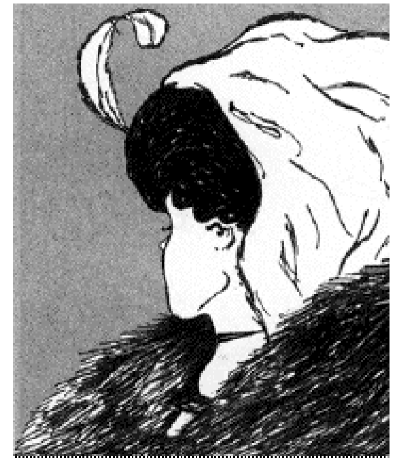
Advantages of er-fMRI

1. Randomised trial order
   cf. confounds of blocked designs
2. **Post hoc classification of trials:**
   according to performance, or because some events can only be indicated by the subject (e.g. spontaneous perceptual changes)

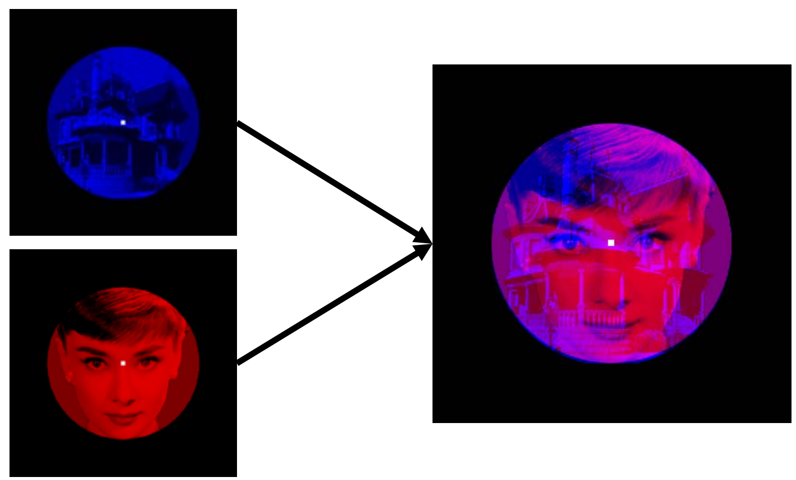
er-fMRI: “on-line” event-definition

Bistable percepts





Binocular rivalry



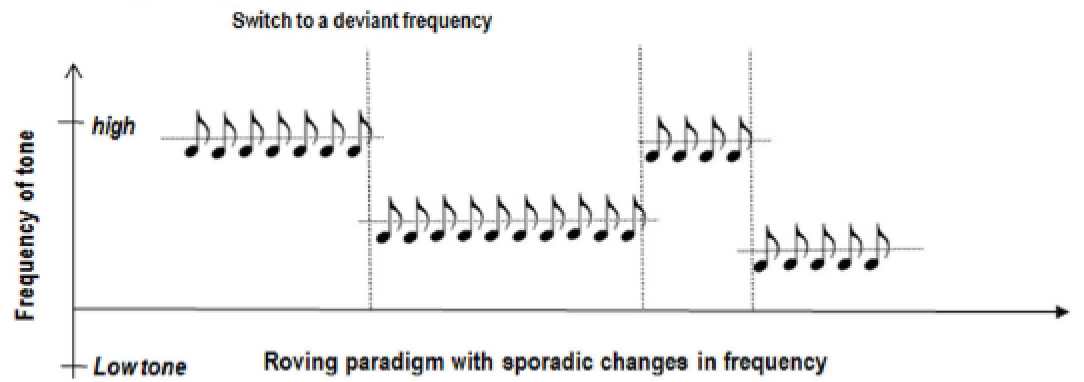
Advantages of er-fMRI

1. Randomised trial order
   cf. confounds of blocked designs

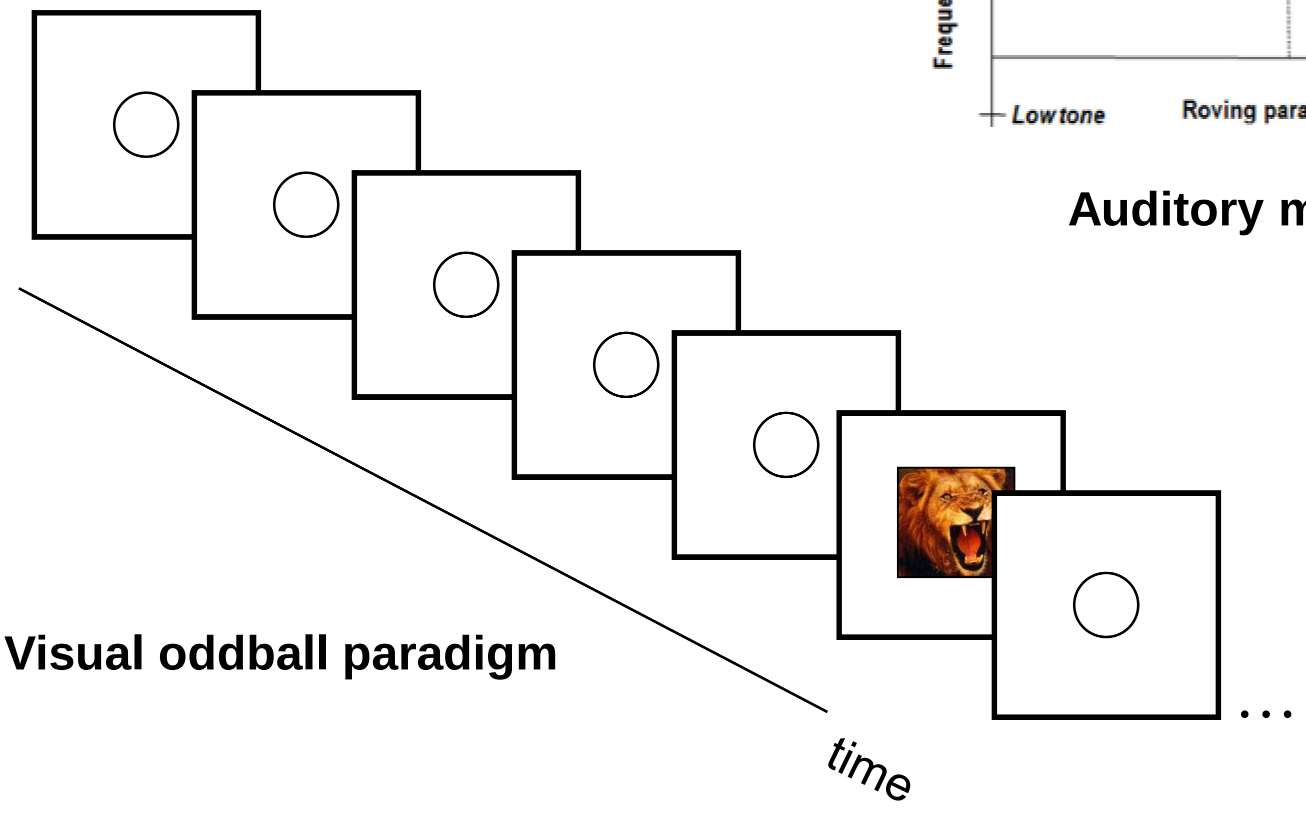
2. Post hoc classification of trials:
   according to performance, or because some events can only be indicated by the subject (e.g. spontaneous perceptual changes)

3. **Some trials cannot be blocked**
   e.g. “oddball” designs

er-fMRI: "oddball" designs



Auditory mismatch negativity (MMN)



Visual oddball paradigm

Advantages of er-fMRI

1. Randomised trial order
   cf. confounds of blocked designs

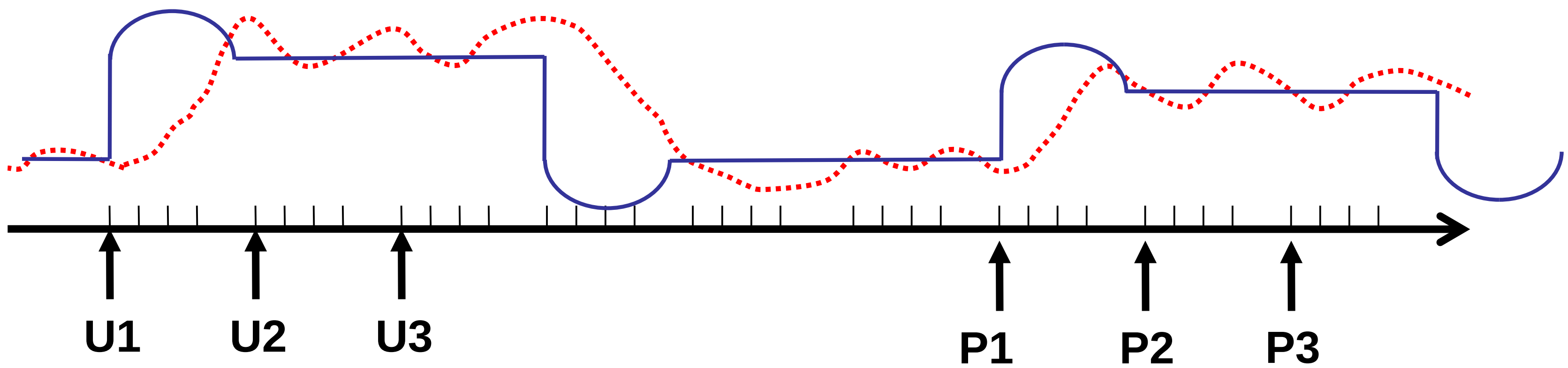
2. Post hoc classification of trials:
   according to performance, or because some events can only be indicated by the subject (e.g. spontaneous perceptual changes)

3. Some trials cannot be blocked
   e.g. “oddball” designs

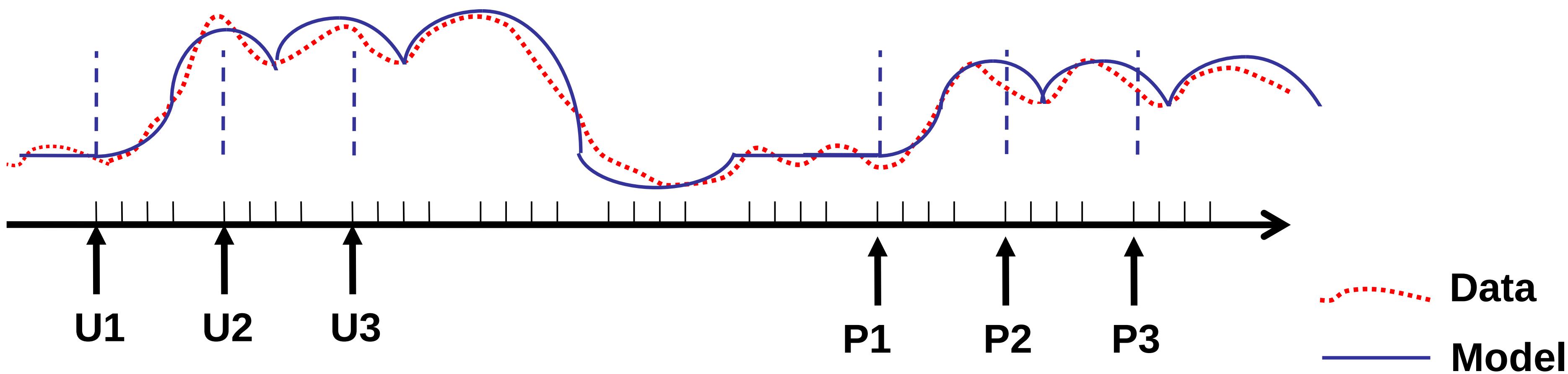
4. **More accurate models even for blocked designs?**

er-fMRI: "event-based" model of block-designs

"Epoch" model assumes constant neural processes throughout block



"Event" model may capture response better

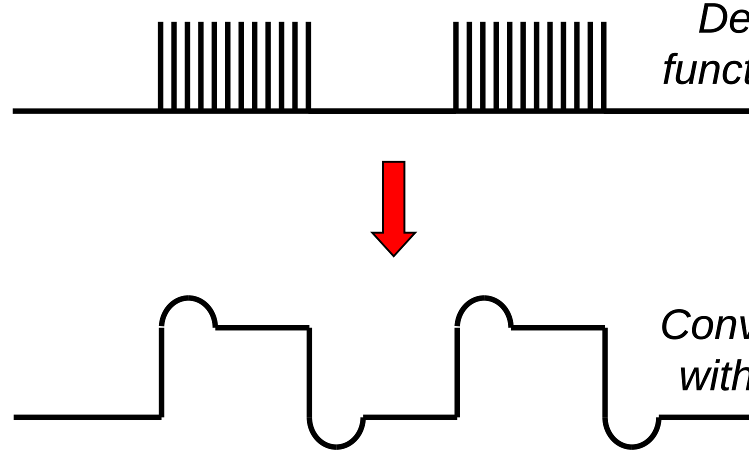


Modeling block designs: epochs vs events

- *Models for ER designs are based on events* (delta functions)...
- ... but models for blocked designs can be epoch- or event-related
- Near-identical regressors can be created by 1) sustained epochs, 2) rapid series of events (SOAs<~3s)
- In SPM, all conditions are specified in terms of their 1) onsets and 2) durations
  - epochs: variable or constant duration, unit amplitude
  - events: zero duration, amplitude: 1/dt

Sustained epoch

Series of events



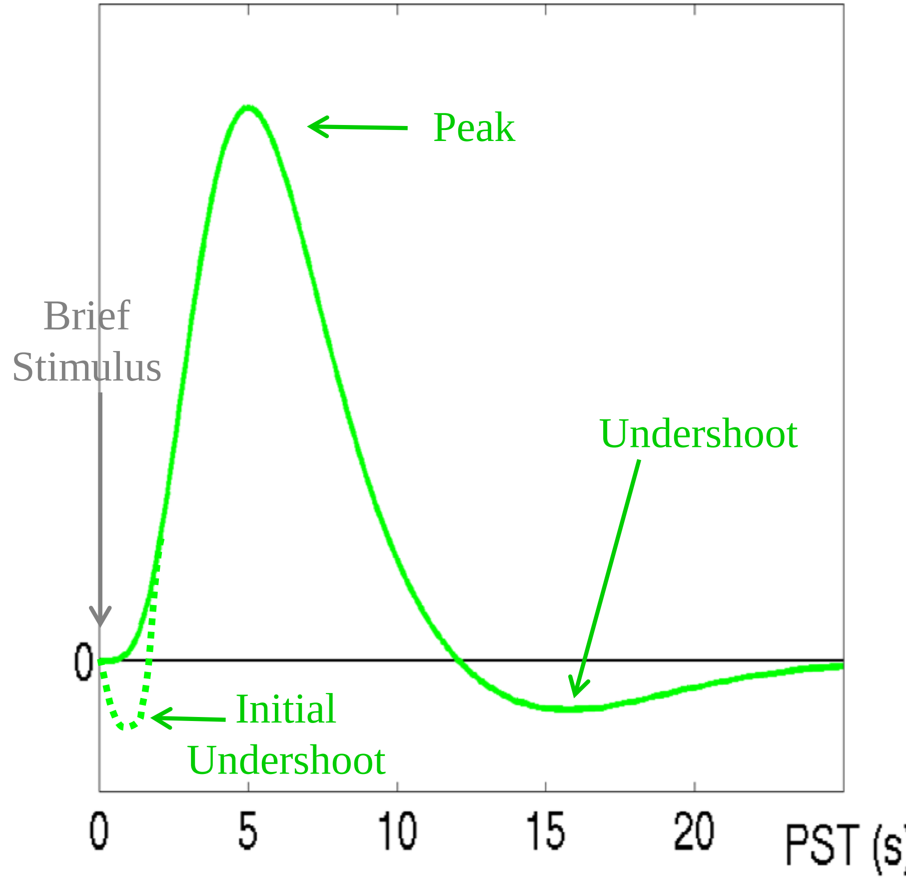
Disadvantages of er-fMRI

1. Less efficient for detecting effects than blocked designs (discussed in detail later).

2. Some psychological processes may be better blocked (e.g. task-switching, attentional instructions).

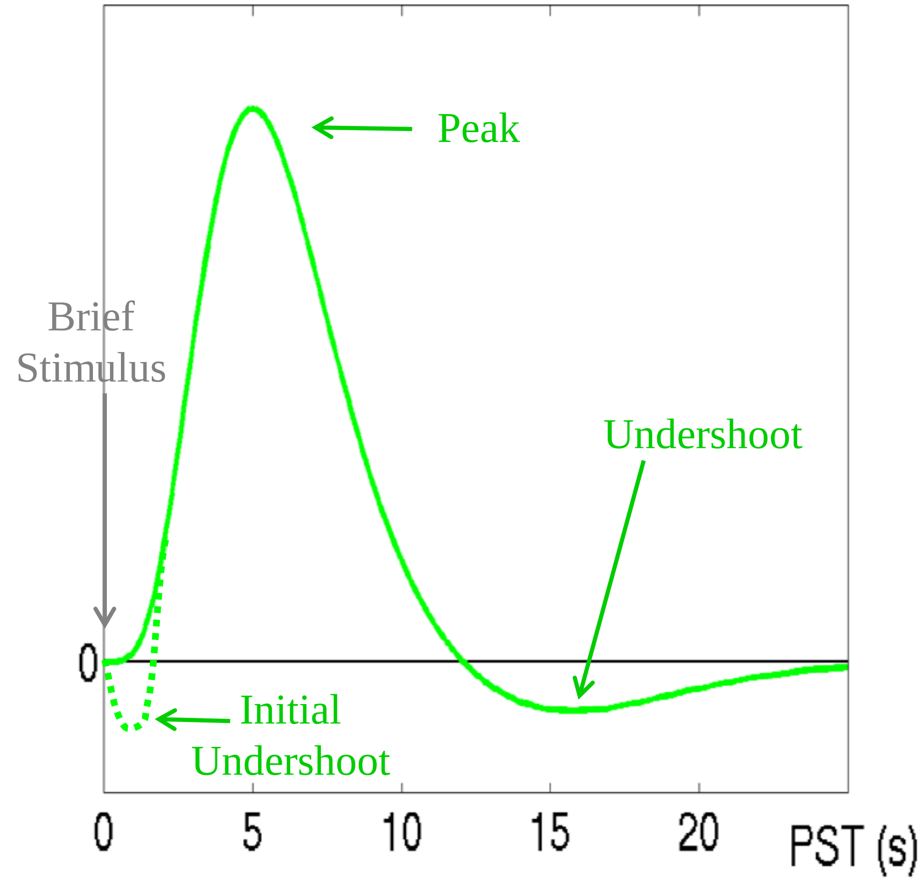
BOLD impulse response

- Function of blood volume and deoxyhemoglobin content (Buxton et al. 1998)
- Peak (max. oxygenation) 4-6s post-stimulus; return to baseline after 20-30s
- initial undershoot sometimes observed (Malonek & Grinvald, 1996)
- Similar across V1, A1, S1…
- … but differences across other regions (Schacter et al. 1997) and individuals (Aguirre et al. 1998)

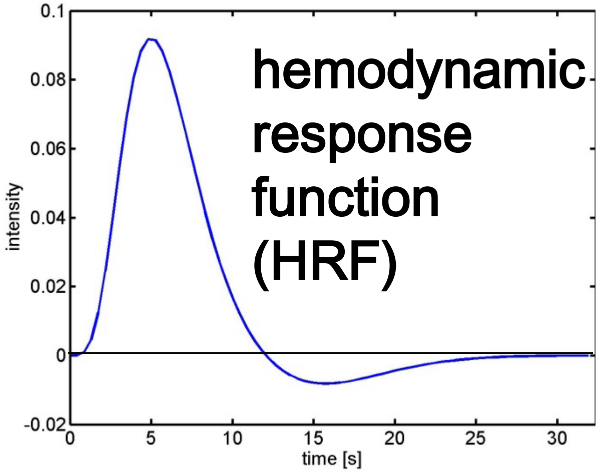


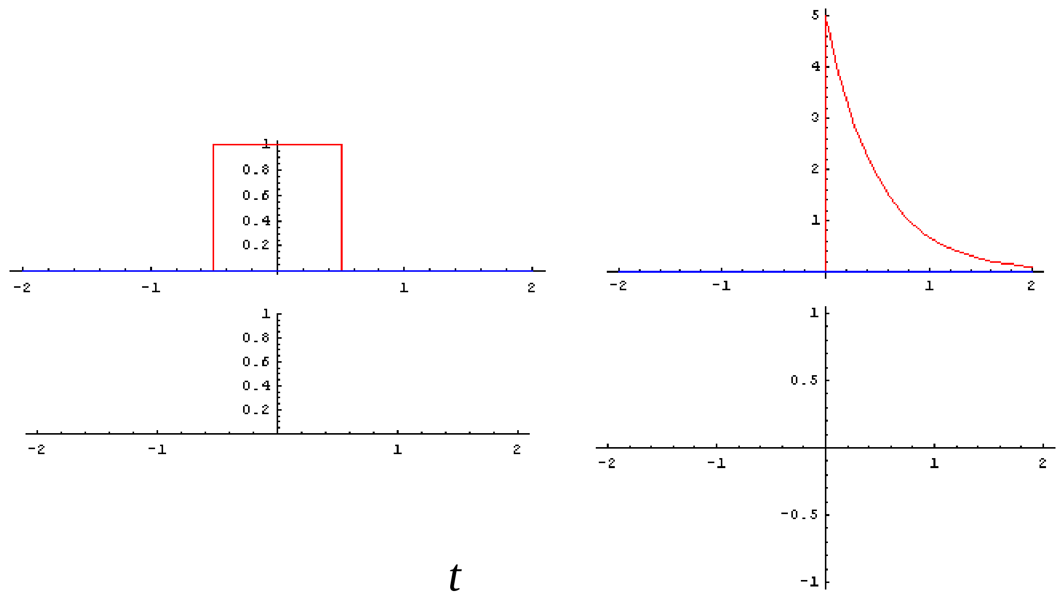
BOLD impulse response

- Early er-fMRI studies used a long Stimulus Onset Asynchrony (SOA) to allow BOLD response to return to baseline.
- However, if the BOLD response is explicitly modelled, overlap between successive responses at short SOAs can be accommodated…
- … particularly if responses are assumed to superpose linearly.
- Short SOAs can give a more efficient design (see below).



Reminder: BOLD response as output from LTI





$$f \otimes g(t) = \int_0^t f(\tau) g(t-\tau) d\tau$$

The response of a linear time-invariant (LTI) system is the convolution of the input with the system's response to an impulse (delta function).

expected BOLD response

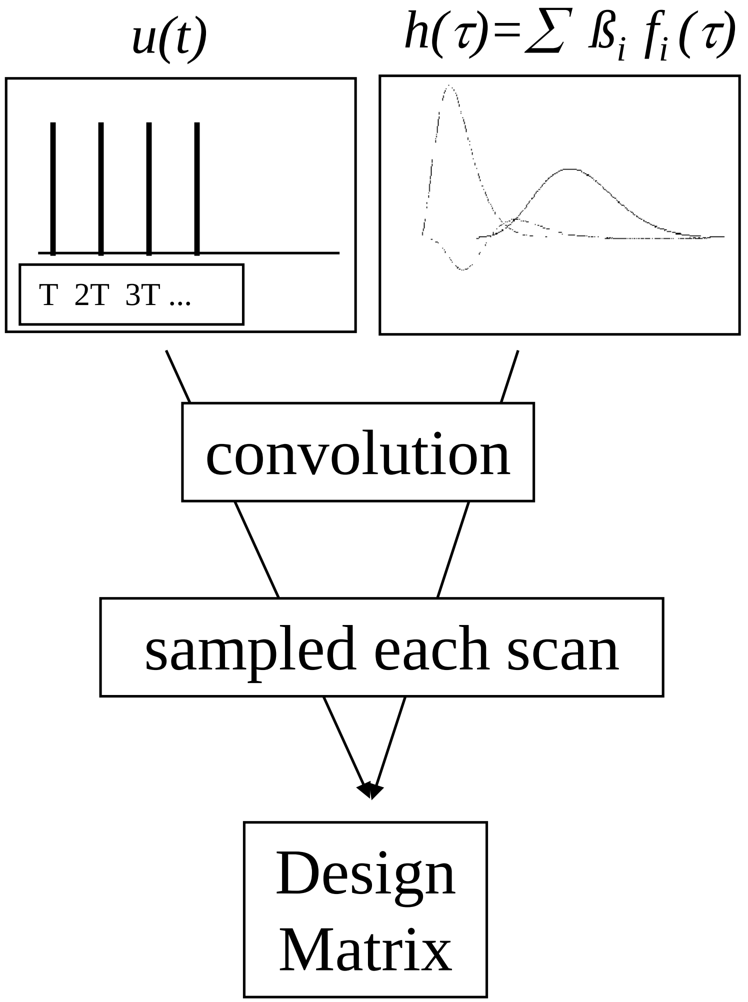
= input function ⊗ impulse response function (HRF)

General Linear (Convolution) Model

For block designs, the exact shape of the convolution kernel (i.e. HRF) does not matter much.

For event-related designs this becomes much more important.

Usually, we use more than a single basis function to model the HRF.



GLM for a single voxel:

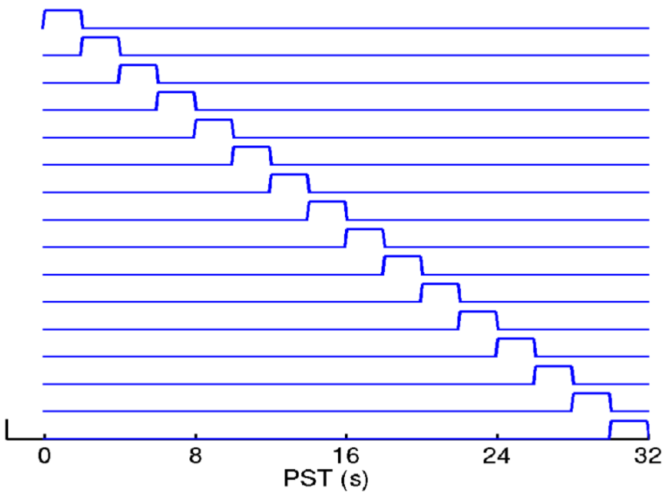
$$\mathbf{y(t)} = [\mathbf{u(t)} \otimes \mathbf{h(\tau)}]\boldsymbol{\beta} + \mathbf{e(t)}$$

Omitting time index:

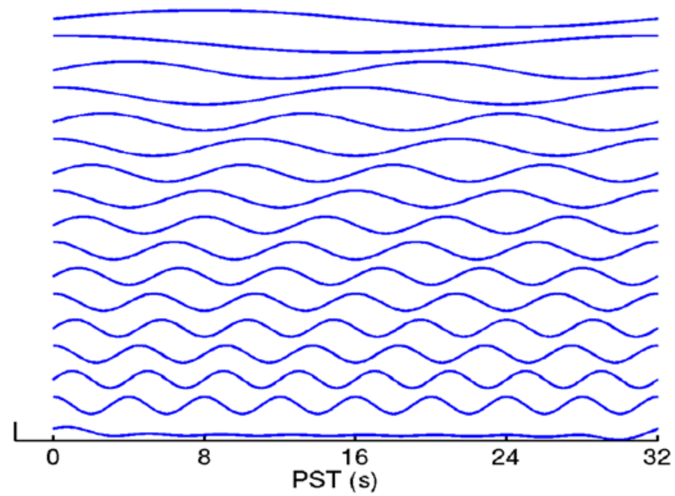
$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{e}$$

Temporal basis functions

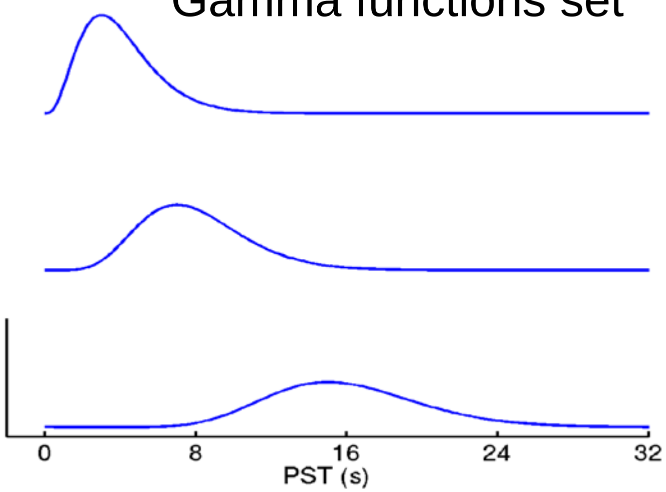
Finite Impulse Response (FIR) model



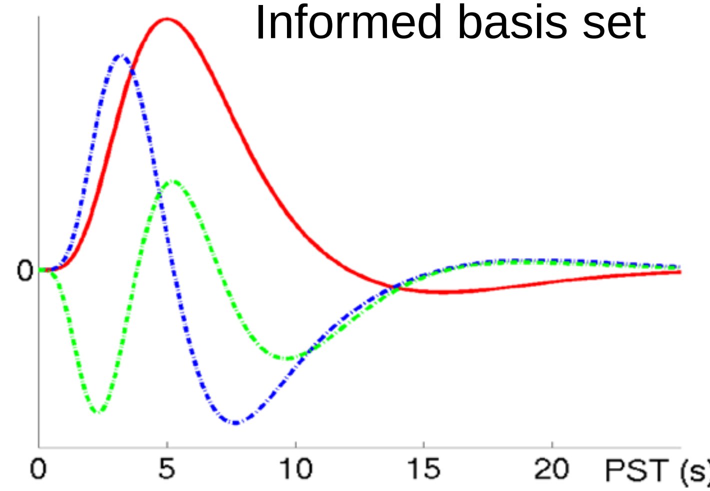
Fourier basis set



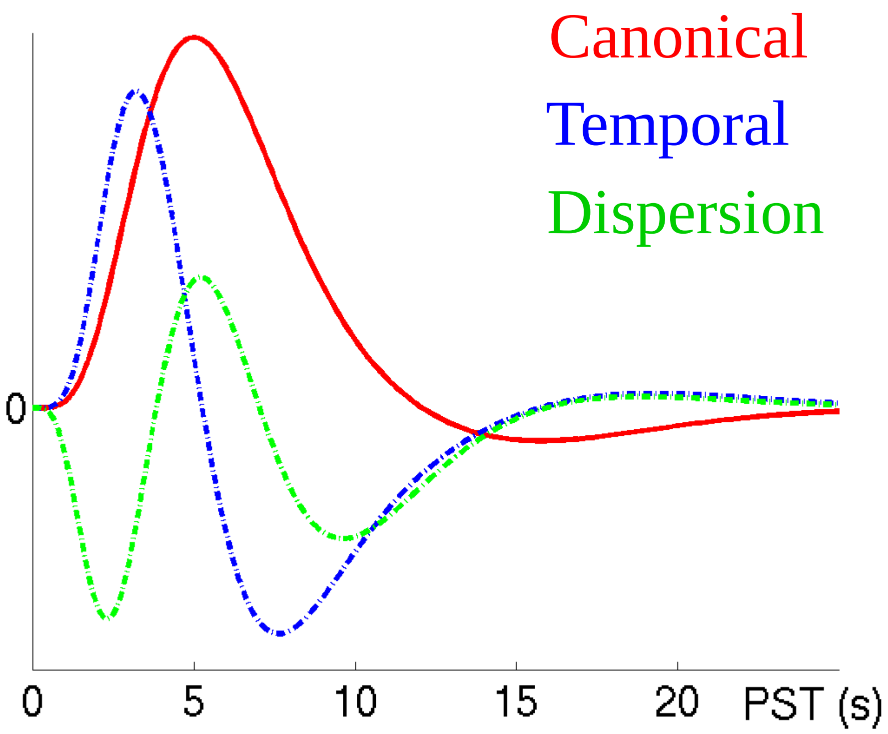
Gamma functions set



Informed basis set



Informed basis set

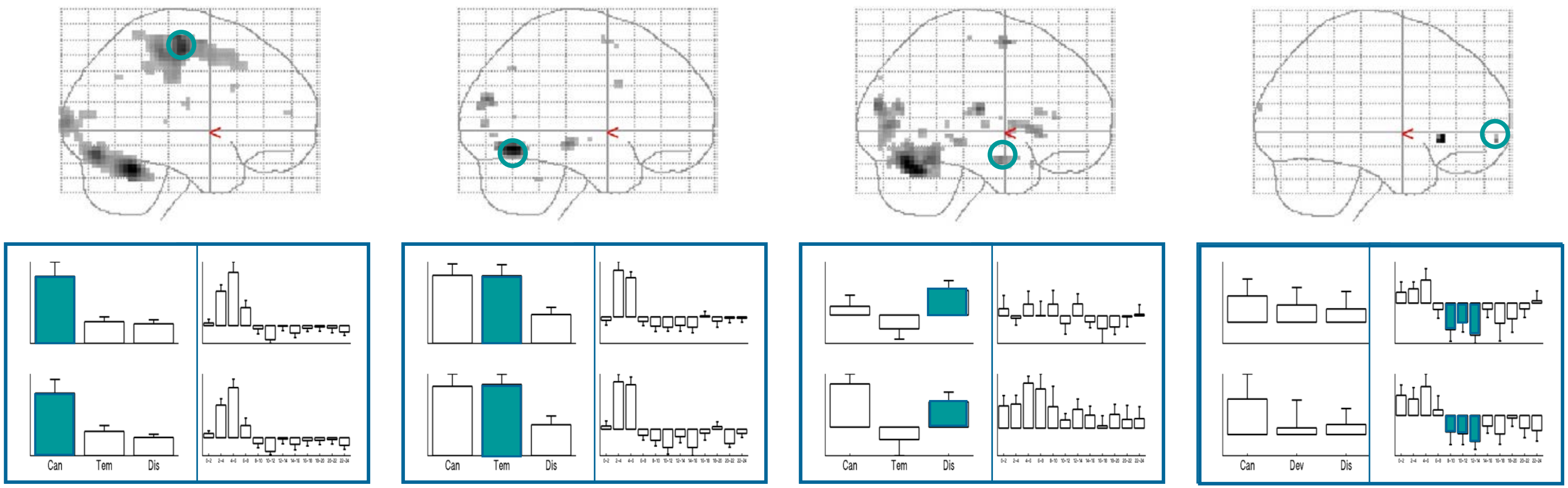


Friston et al. 1998, *NeuroImage*

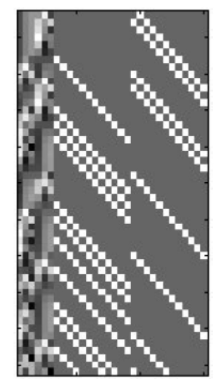
- Canonical HRF:
  - linear combination of 2 gamma functions
  - 7 parameters, see spm_hrf
- *plus* multivariate Taylor expansion in:
  - time (*Temporal Derivative*)
  - width (*Dispersion Derivative;* partial derivative of canonical HRF wrt. parameter controlling the width)
- F-tests: testing for responses of any shape.
- T-tests on canonical HRF alone (at 1st level) can be improved by derivatives reducing residual error, and can be interpreted as "amplitude" differences, assuming canonical HRF is a reasonable fit.

Temporal basis sets: Which one?

In this example (rapid motor response to faces, *Henson et al, 2001*)…



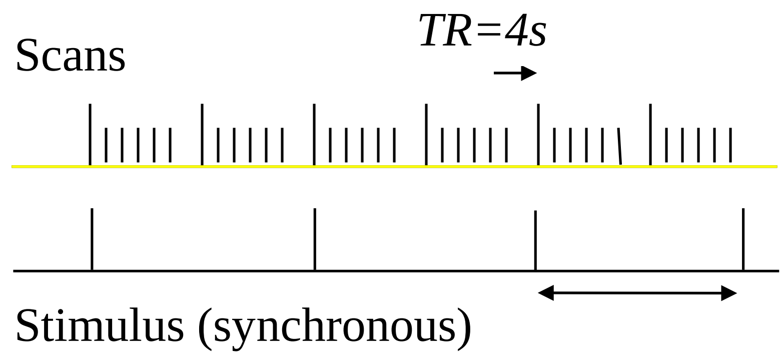
Canonical **+ Temporal** **+ Dispersion** **+ FIR**

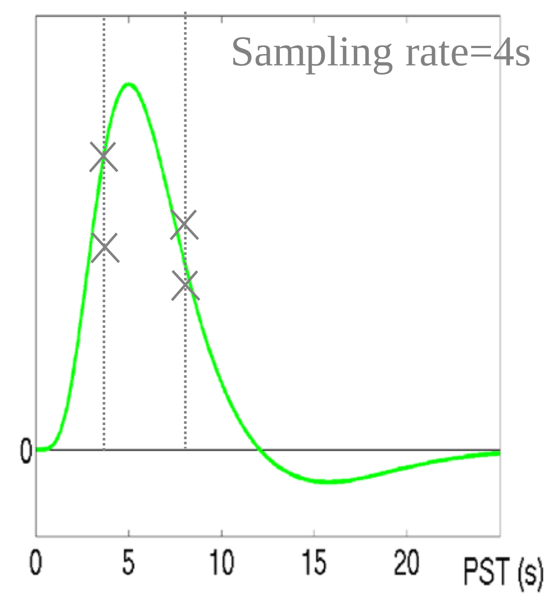


- canonical + temporal + dispersion derivatives appear sufficient
- may not be for more complex trials (e.g. stimulus-delay-response)
- but then such trials better modelled with separate neural components (i.e. activity no longer delta function) (Zarahn, 1999)

Timing Issues : Practical

- Assume TR is 4s
- Sampling at [0,4,8,12…] post- stimulus may miss peak signal

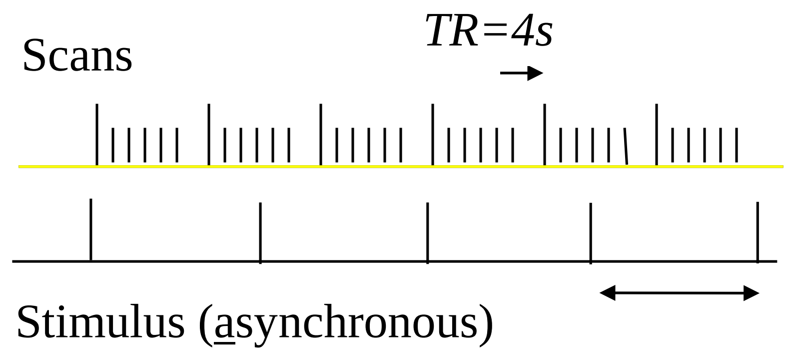


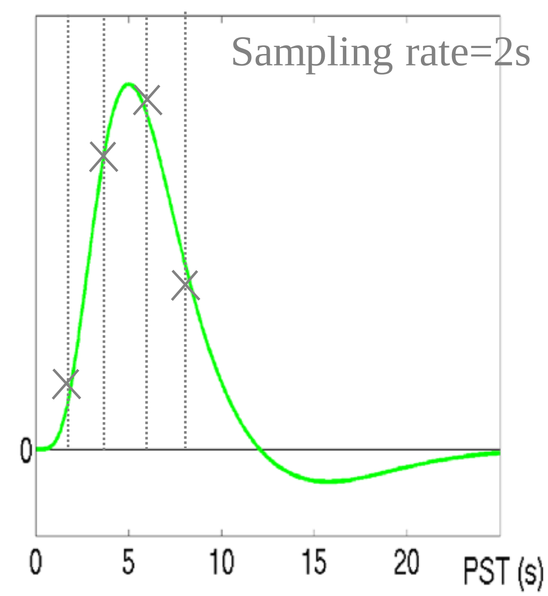


SOA = Stimulus onset asynchrony
(= time between onsets of two subsequent stimuli)

Timing Issues : Practical

- Assume TR is 4s
- Sampling at [0,4,8,12…] post- stimulus may miss peak signal
- Higher effective sampling by:
  - 1. Asynchrony, *e.g. SOA* = 1.5×TR

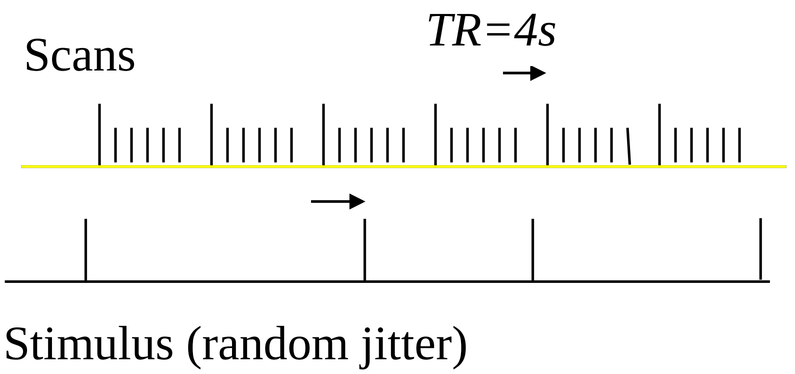


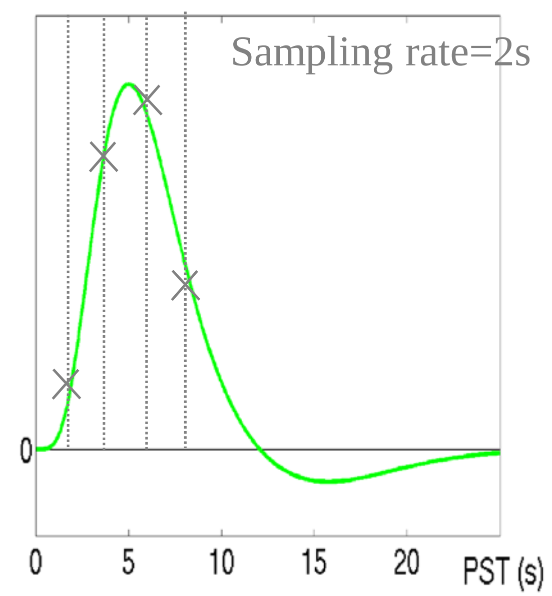


SOA = Stimulus onset asynchrony
(= time between onsets of two subsequent stimuli)

Timing Issues : Practical

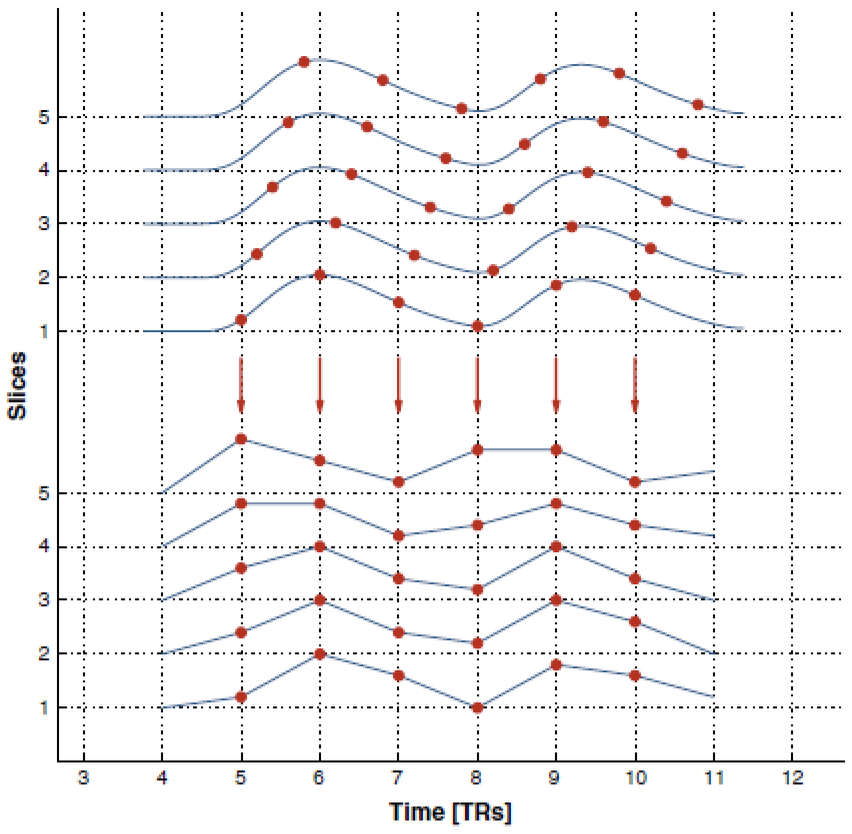
- Assume TR is 4s
- Sampling at [0,4,8,12…] post- stimulus may miss peak signal
- Higher effective sampling by:
  - 1. Asynchrony, *e.g. SOA* = 1.5×TR
  - 2. Random jitter, e.g. SOA = (2 ± 0.5)×TR
- Better response characterisation (Miezin et al, 2000)





SOA = Stimulus onset asynchrony
(= time between onsets of two subsequent stimuli)

Slice-timing



Sladky et al. 2011, NeuroImage

Slice-timing

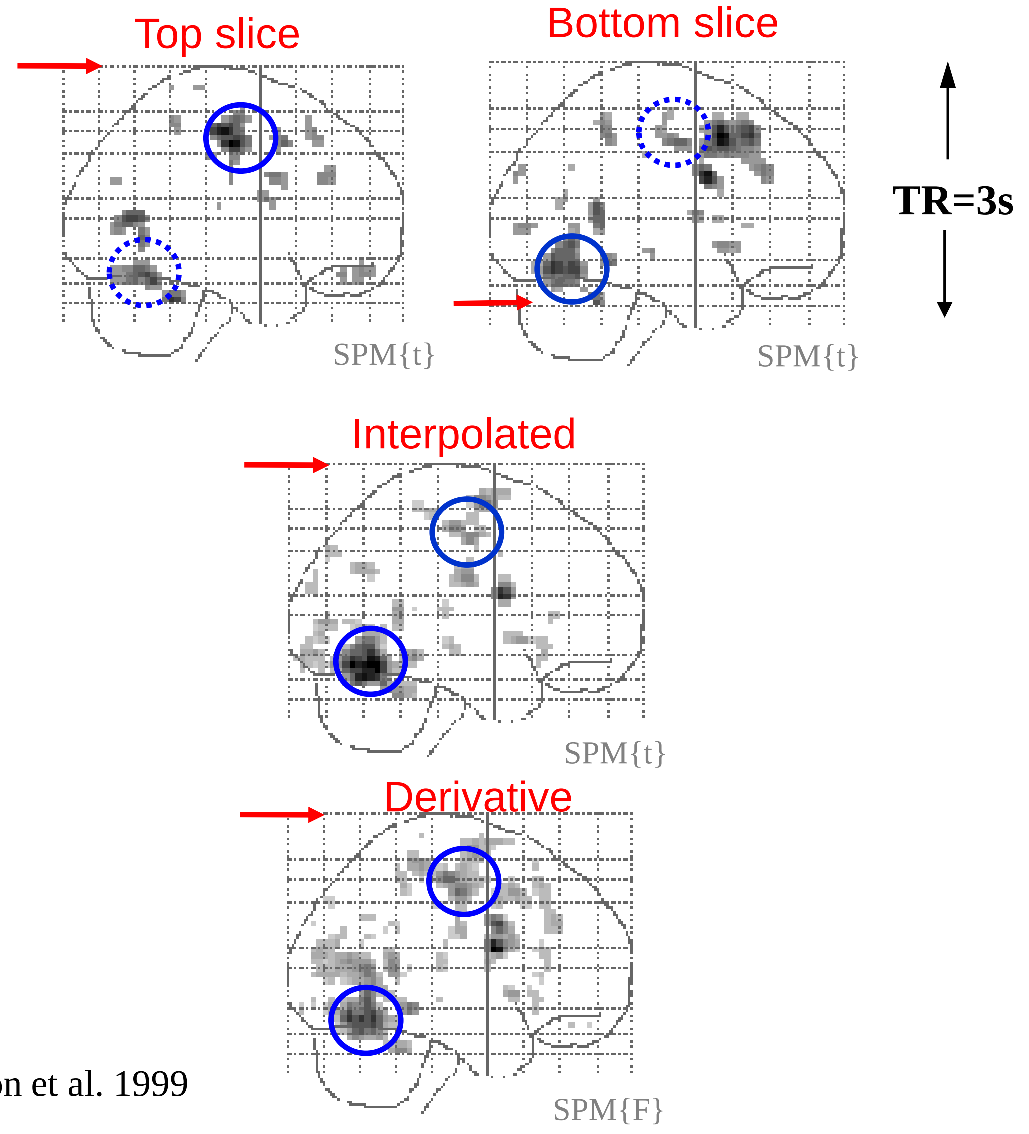
- Slices acquired at different times, yet model is the same for all slices

  => *different results (using canonical HRF) for different reference slices*

- Solutions:

1. Temporal interpolation of data
   … but may be problematic for longer TRs

2. More general basis set (e.g. with temporal derivatives)
   … but more complicated design matrix



Henson et al. 1999

Design efficiency

- The aim is to minimize the standard error of a *t*-contrast (i.e. the denominator of a t-statistic).

$$T = \frac{c^T \hat{\beta}}{\sqrt{\mathrm{var}(c^T \hat{\beta})}}$$

$$\mathrm{var}(c^T \hat{\beta}) = \hat{\sigma}^2 c^T (X^T X)^{-1} c$$

- This is equivalent to maximizing the efficiency ε:

$$\varepsilon(\hat{\sigma}^2, c, X) = (\hat{\sigma}^2 c^T (X^T X)^{-1} c)^{-1}$$

Noise variance ($\hat{\sigma}^2$)

Design variance ($X^T X$)

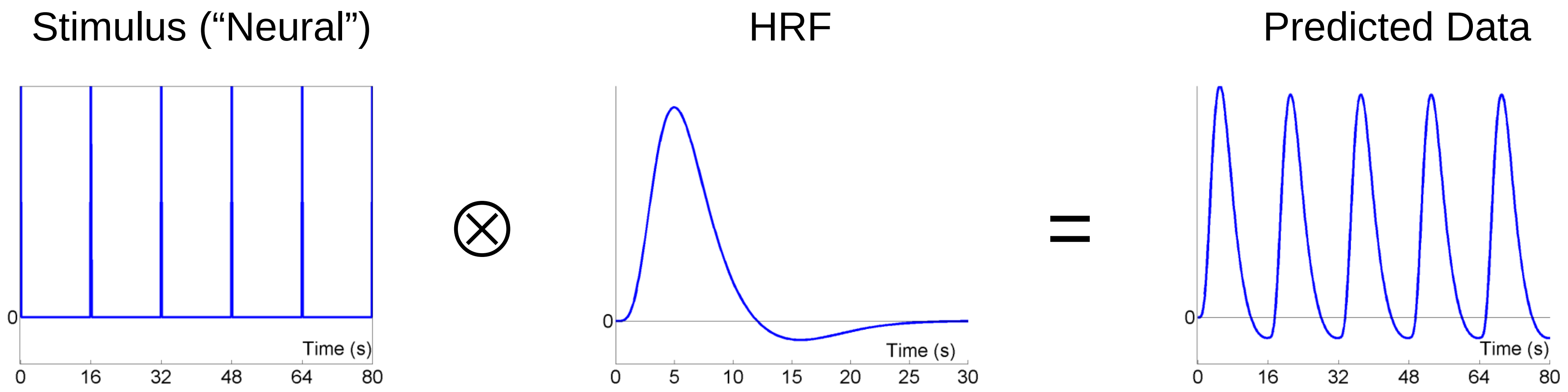
- If we assume that the noise variance is independent of the specific design:

$$\varepsilon(c, X) = (c^T (X^T X)^{-1} c)^{-1}$$

NB: efficiency depends on design matrix and the chosen contrast !

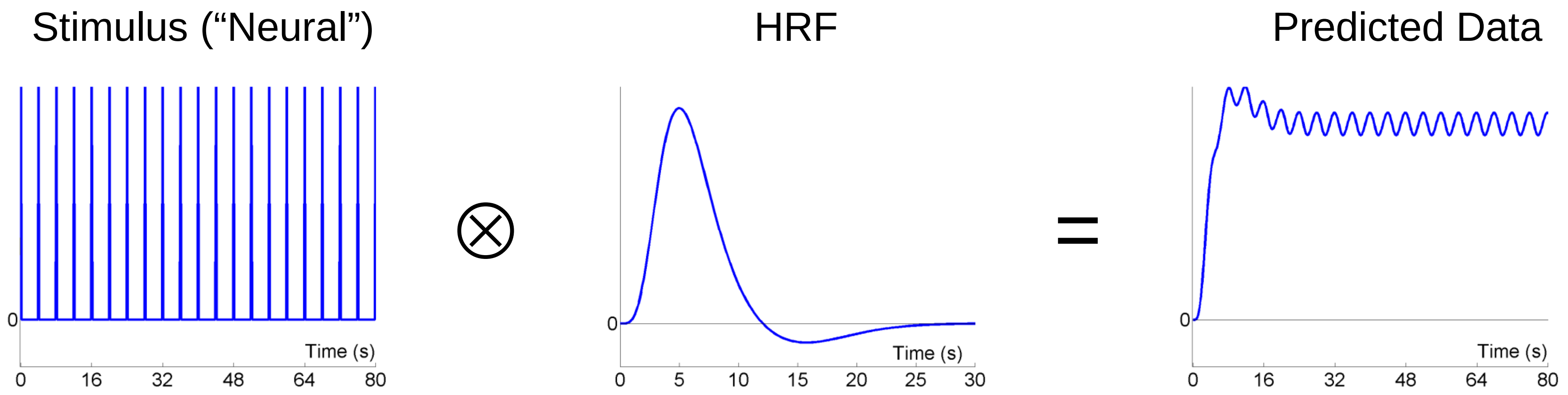
- This is a relative measure: all we can say is that one design is more efficient than another (for a given contrast).

Fixed SOA = 16s



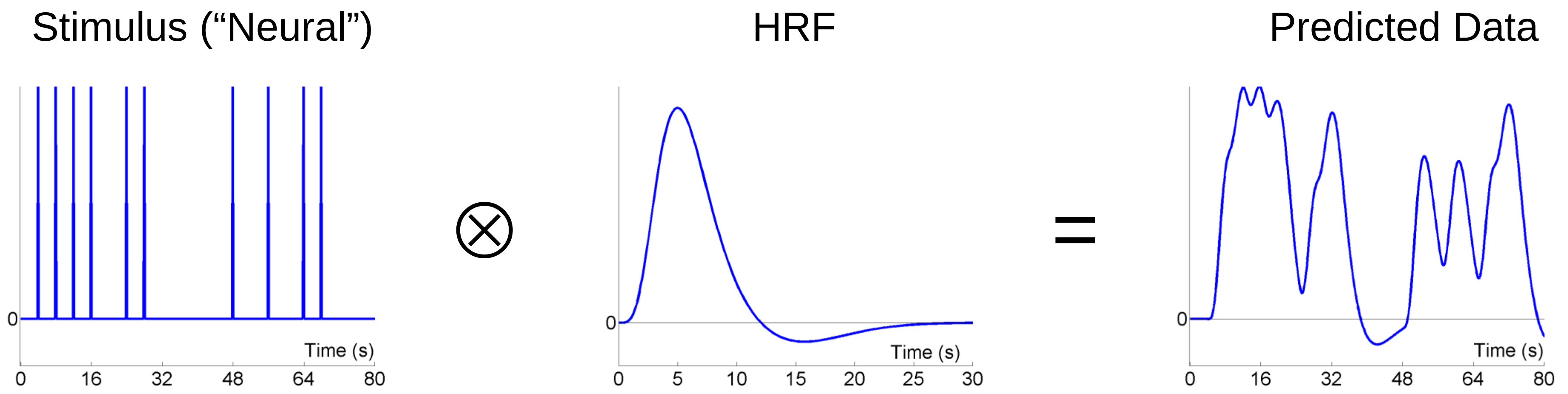
Not particularly efficient…

Fixed SOA = 4s



Very inefficient…

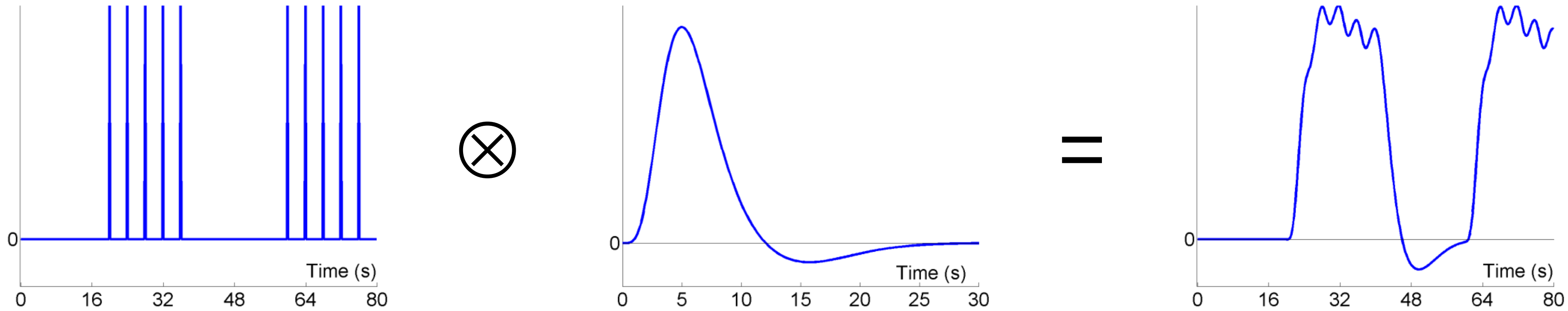
Randomised, SOA_{min} = 4s



More efficient …

Blocked, SOA_{min}= 4s

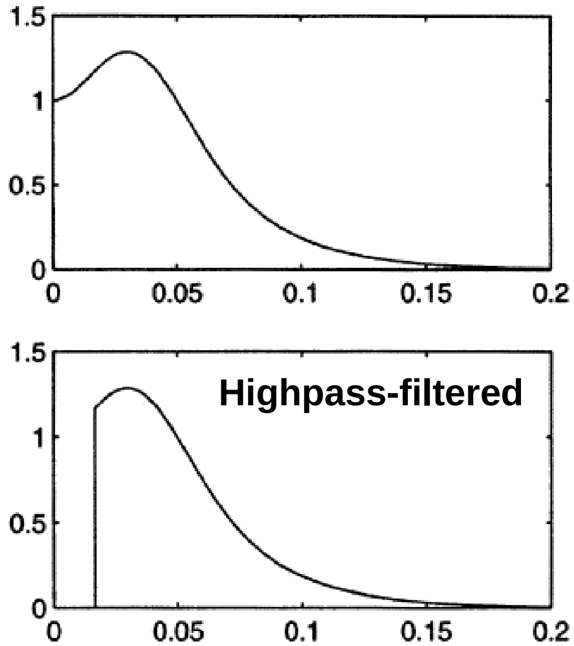
Stimulus (“Neural”) HRF Predicted Data

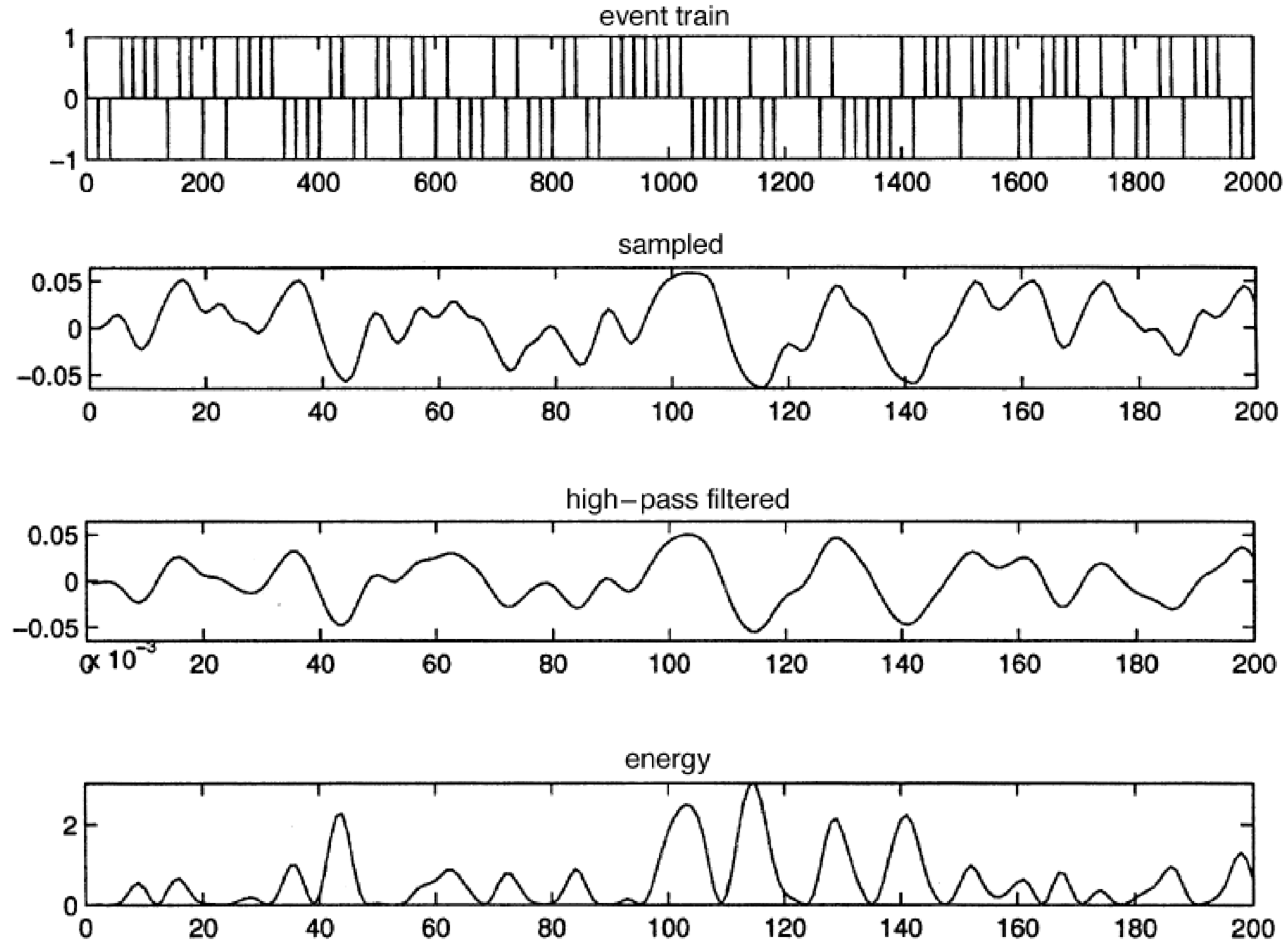


Even more efficient…

Another perspective on efficiency

Hemodynamic transfer function
(based on canonical HRF)**:**
neural activity (Hz) → BOLD





efficiency = bandpassed signal energy

Josephs & Henson 1999, *Phil Trans B*

Fourier series

Sine wave

$$y(t) = A\sin(2\pi ft + \varphi) = A\sin(\omega t + \varphi)$$

where:

- A = the *amplitude*, the peak deviation of the function from zero.
- f = the *ordinary frequency*, the *number* of oscillations (cycles) that occur each second of time.
- $\omega = 2\pi f$, the *angular frequency*, the rate of change of the function argument in units of radians per second
- φ = the *phase*, specifies (in radians) where in its cycle the oscillation is at $t = 0$.

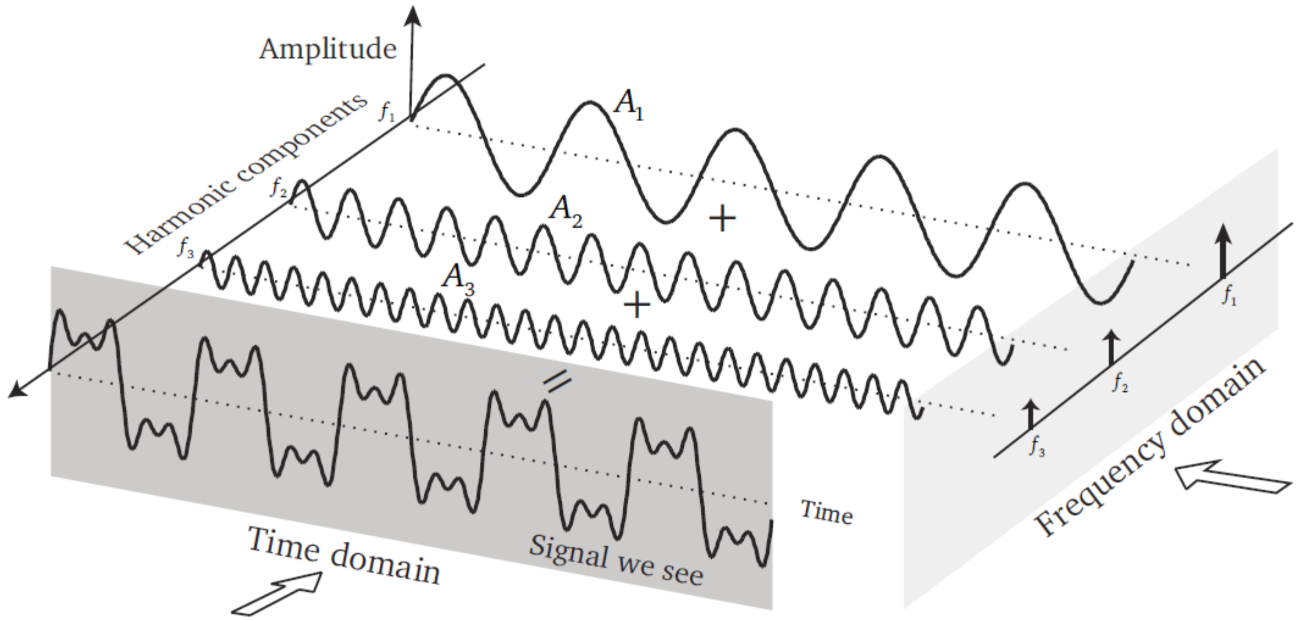
Power = squared amplitude (often represented in logs)

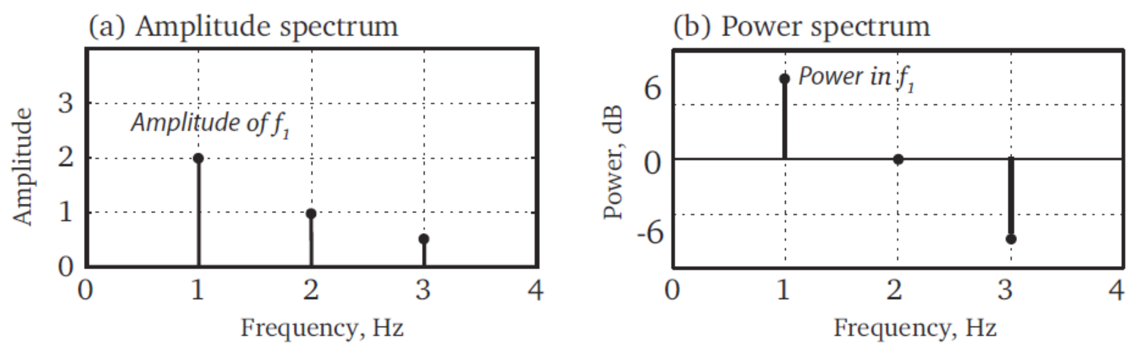
Signal energy = integral of power over time

Fourier series
= infinite sum of sines and cosines of different frequencies

$$f(t) = a_0 + \sum_{k=1}^{\infty} a_k \cos(2\pi f_k t) + \sum_{k=1}^{\infty} b_k \sin(2\pi f_k t)$$

Fourier series





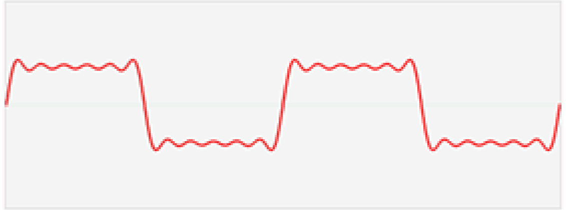
Langton & Levin (2016) Intuitive Guide to Fourier Analysis

Fourier transform

- simply speaking, the Fourier transform F provides the Fourier series coefficients for a signal, i.e., it decomposes a function of time (a signal) into the frequencies it consists of

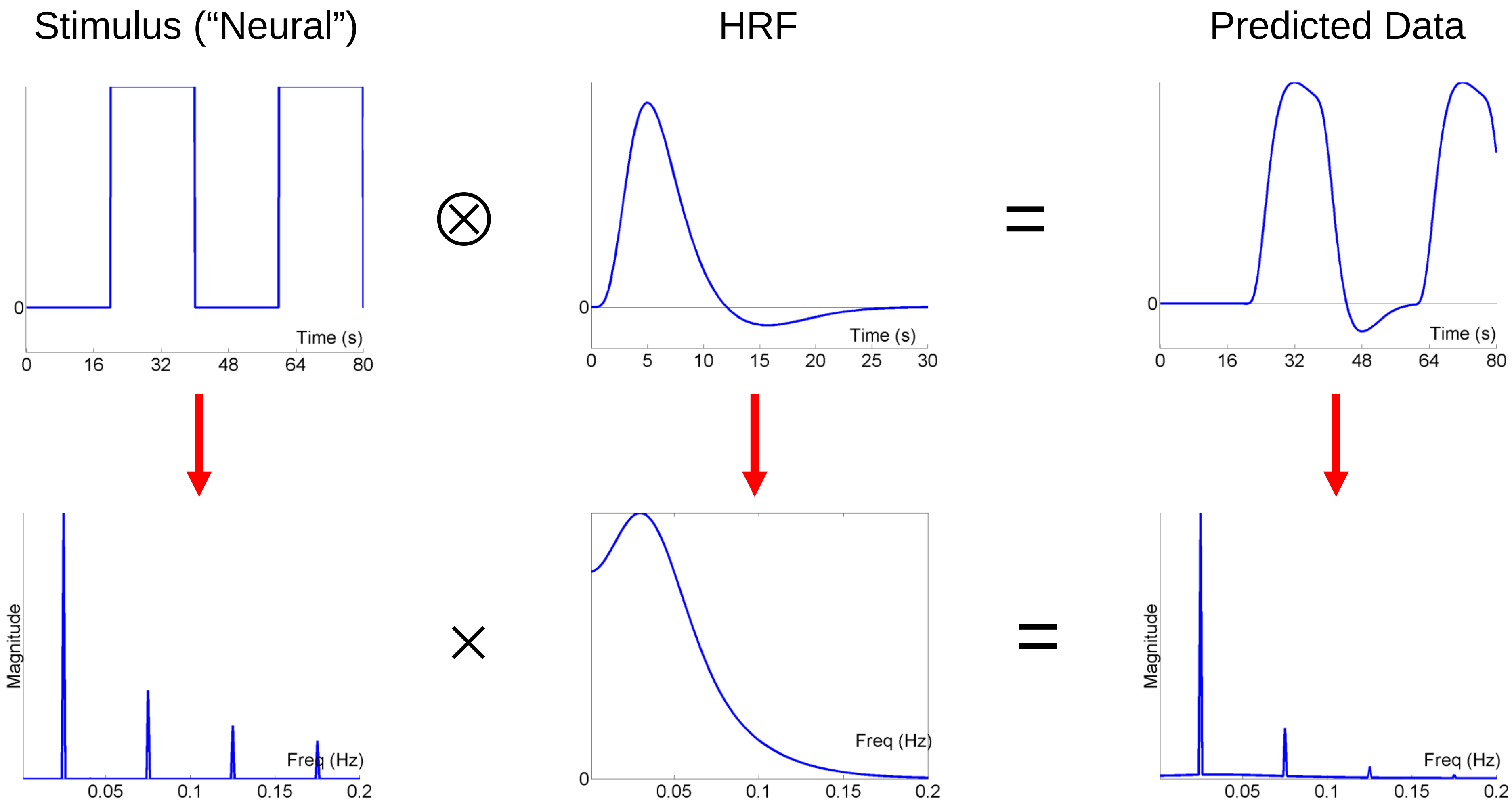
- linear operator

- convolution in time domain = multiplication in frequency domain: F(f*g) = F(f)F(g)



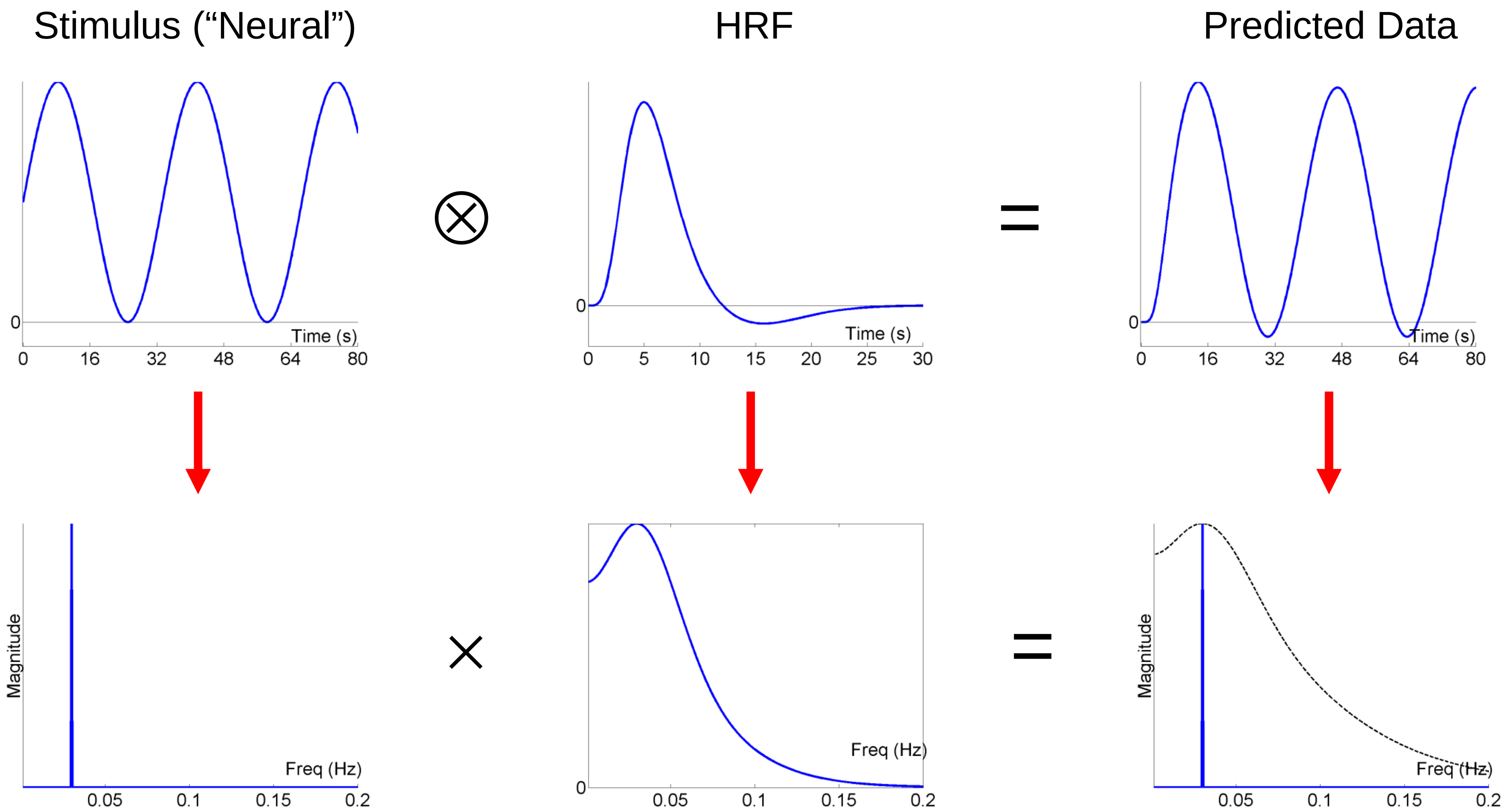
Animation: https://en.wikipedia.org/wiki/Fourier_transform

Blocked, epoch = 20s



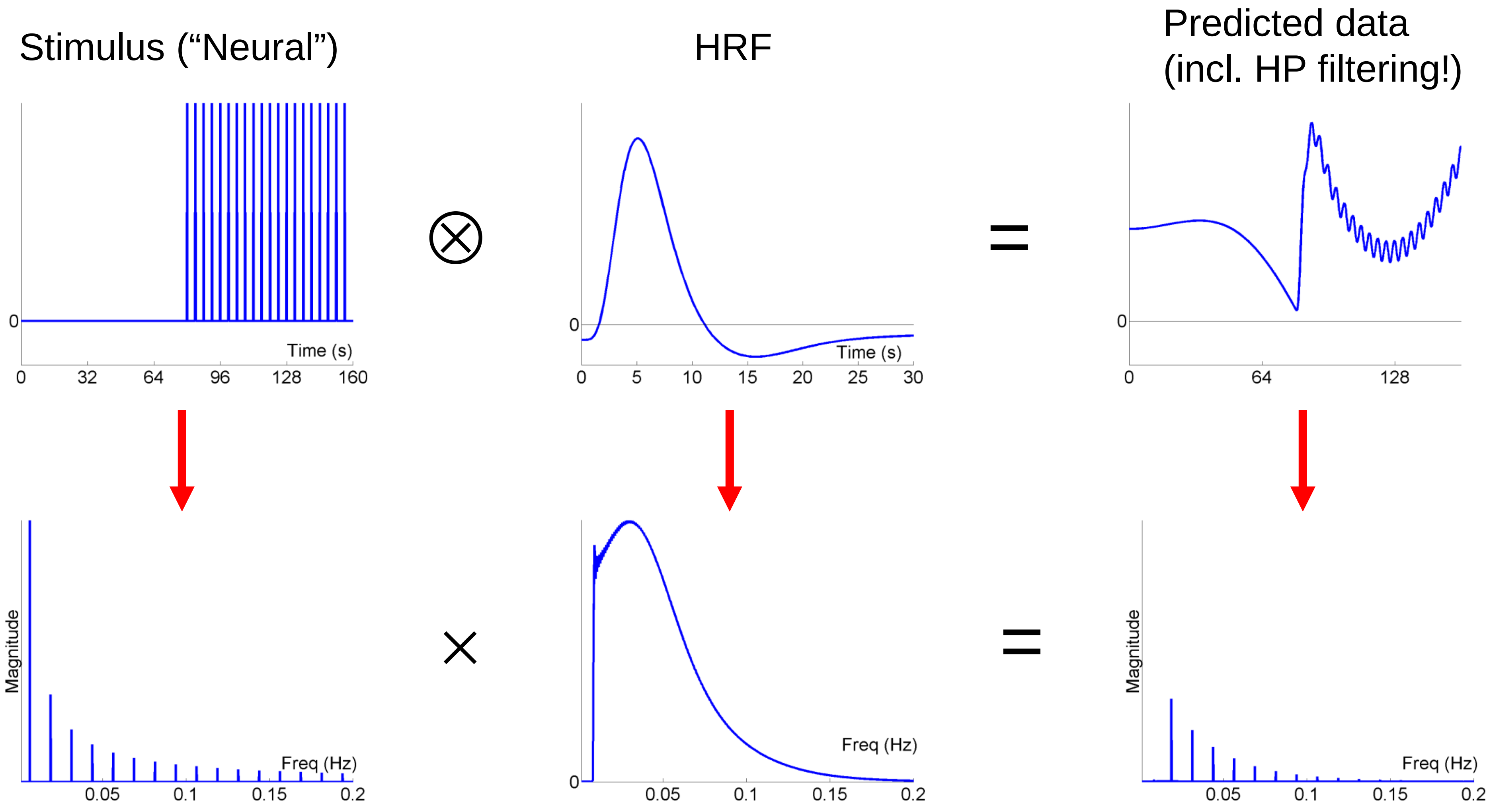
Blocked-epoch (with short SOA)

Sinusoidal modulation, f = 1/33s



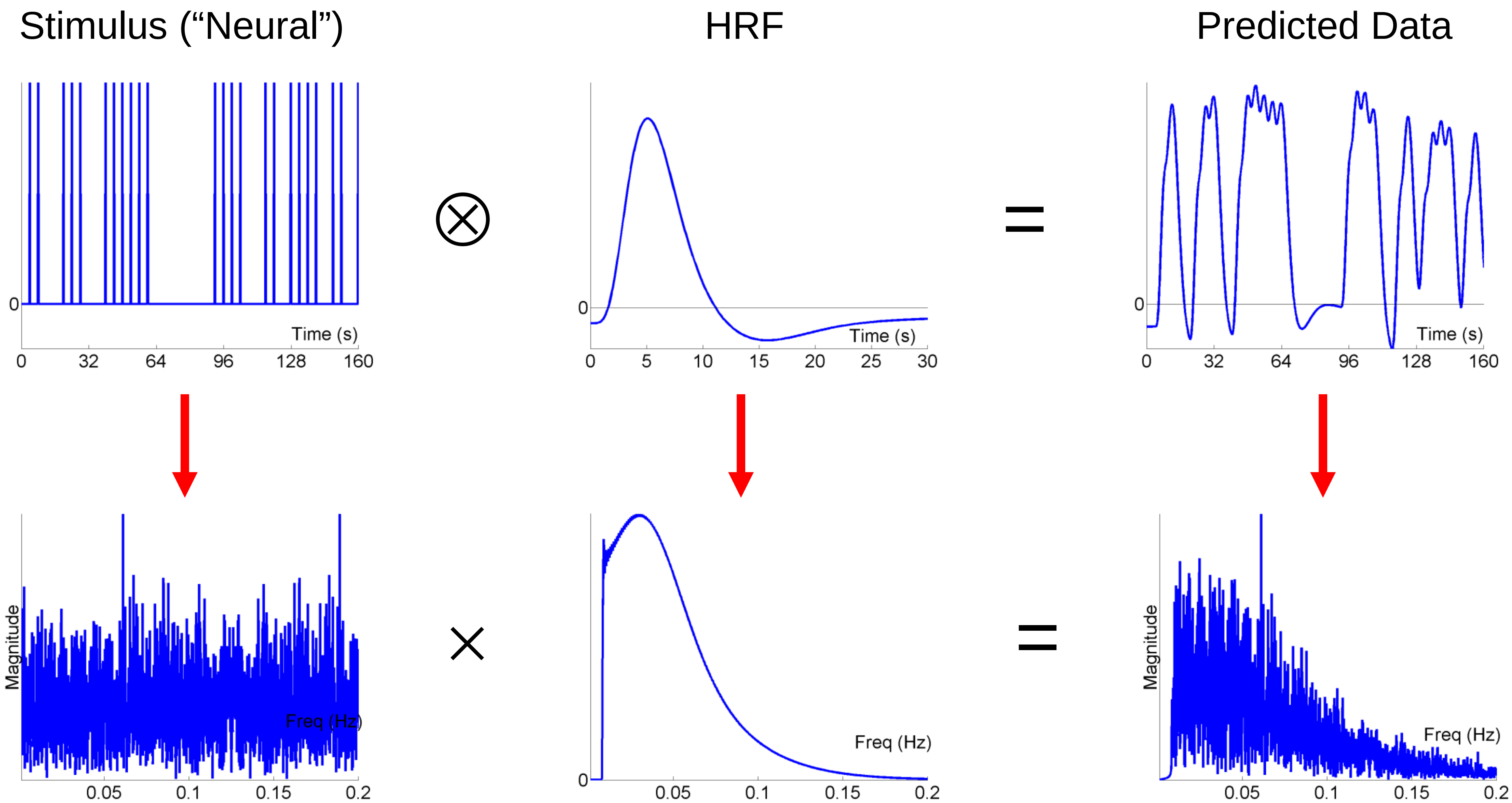
The most efficient design of all!

Blocked (80s), SOA_{min}=4s, highpass filter = 1/120s



Don't use long (>60s) blocks!

Randomised, SOA_{min}=4s, highpass filter = 1/120s



Randomised design spreads power over frequencies.

Efficiency: Multiple event types

- Design parametrised by:

  SOA$_{min}$ Minimum SOA

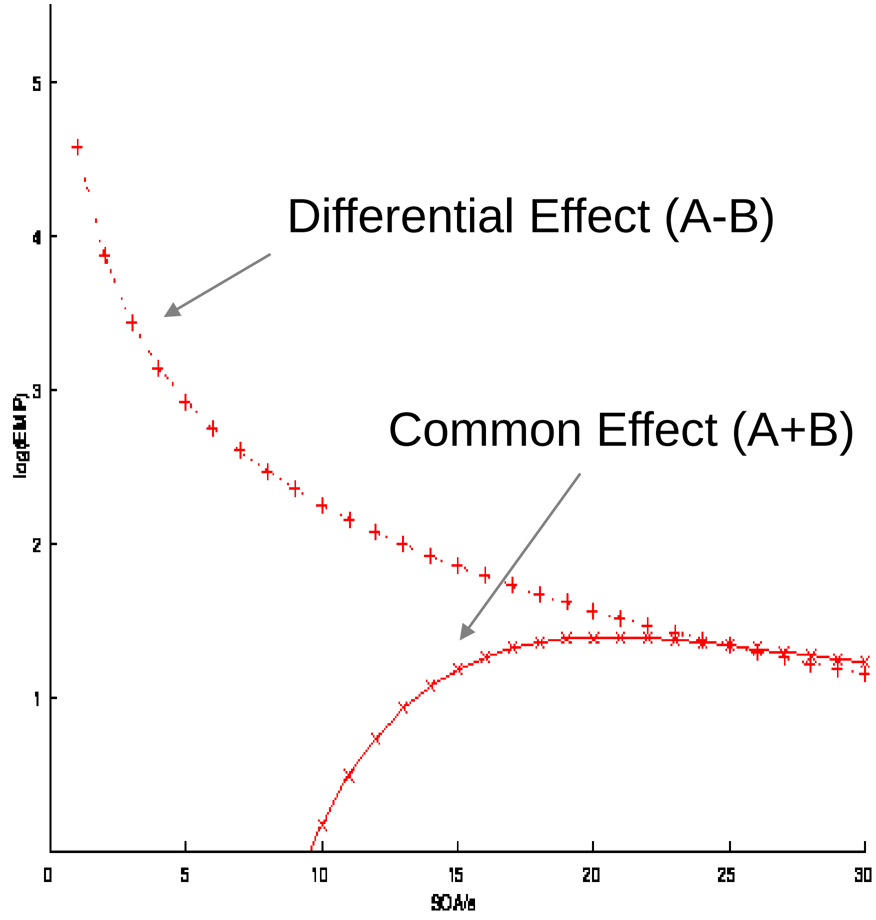
  $p_i(\mathbf{h})$ Probability of event-type *i* given history ***h*** of last *m* events

- With *n* event-types $p_i(\mathbf{h})$ is a $n^m \times n$ *Transition Matrix*

- Example: Randomised AB

	A	**B**
A	0.5	0.5
B	0.5	0.5

=> **ABBBABAABABAAA...**



4s smoothing; 1/60s highpass filtering

Efficiency: Multiple event types

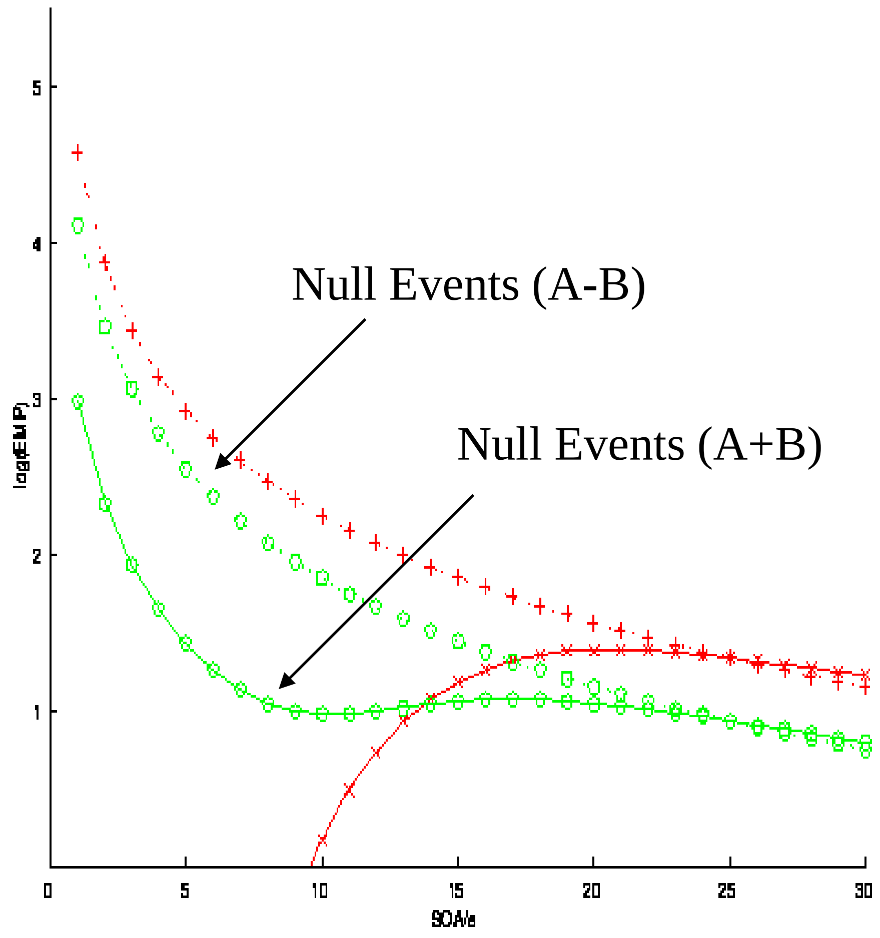
- Example: Null events

	A	B
A	0.33	0.33
B	0.33	0.33

=> **AB-BAA--B---ABB...**

- Efficient for differential ***and*** main effects at short SOA

- Equivalent to stochastic SOA (null event corresponds to a third unmodelled event-type)



4s smoothing; 1/60s highpass filtering

Efficiency – main conclusions

- Optimal design for one contrast may not be optimal for another.
- Generally, blocked designs with short SOAs are the most efficient design.
- With randomised designs, optimal SOA for differential effect (A-B) is minimal SOA (assuming no saturation), whereas optimal SOA for common effect (A+B) is 16-20s.
- Inclusion of null events gives good efficiency for both common and differential effects at short SOAs.

Thank you