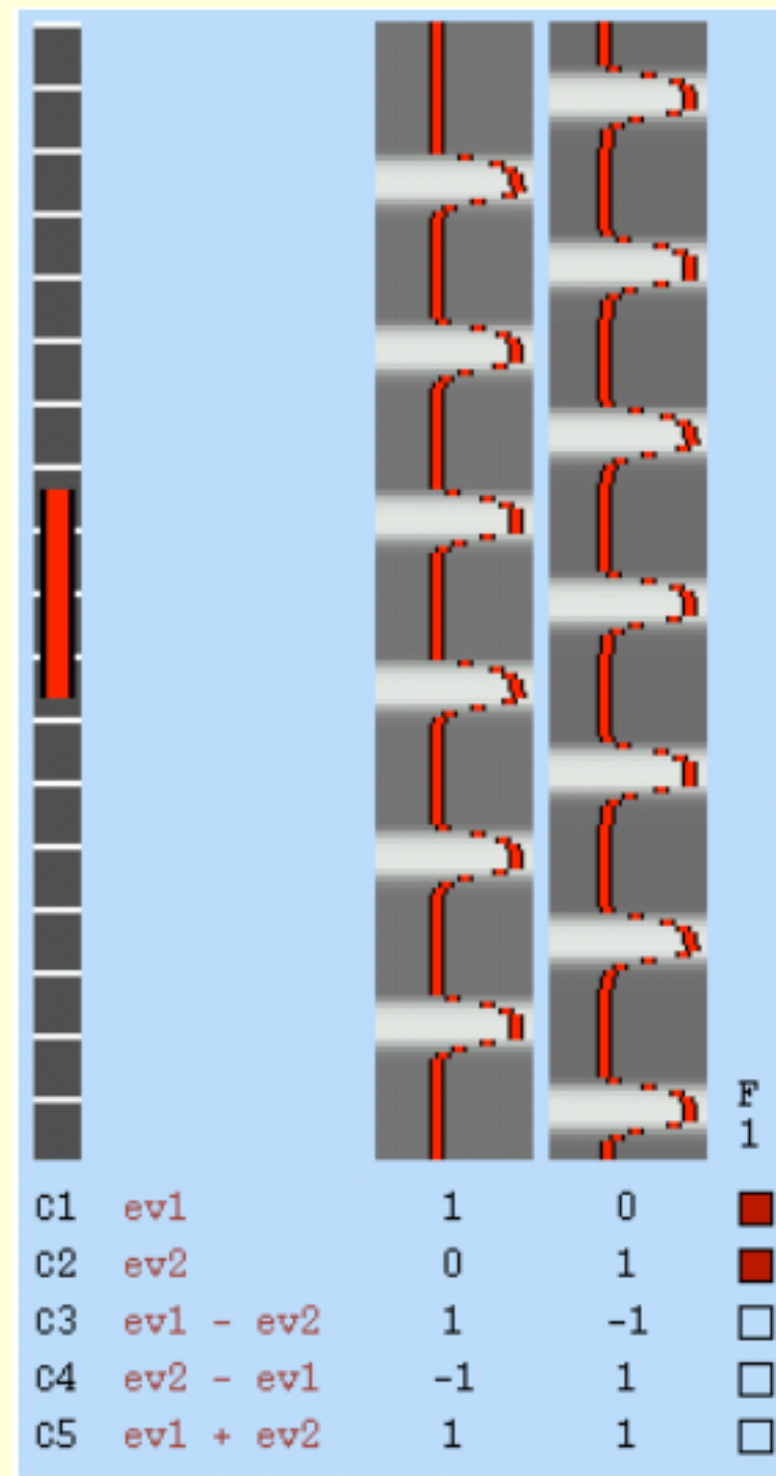


Single session analysis using FEAT

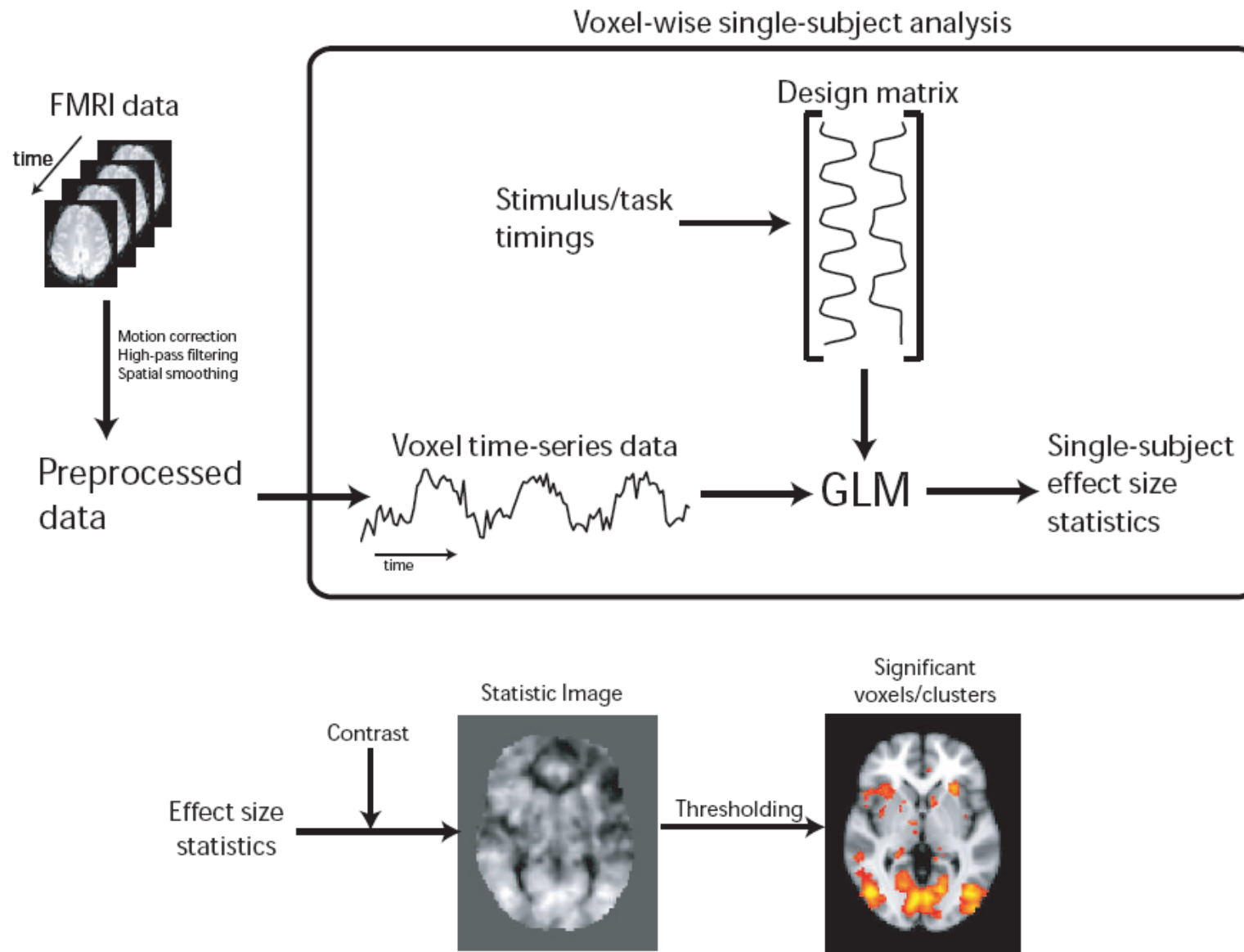


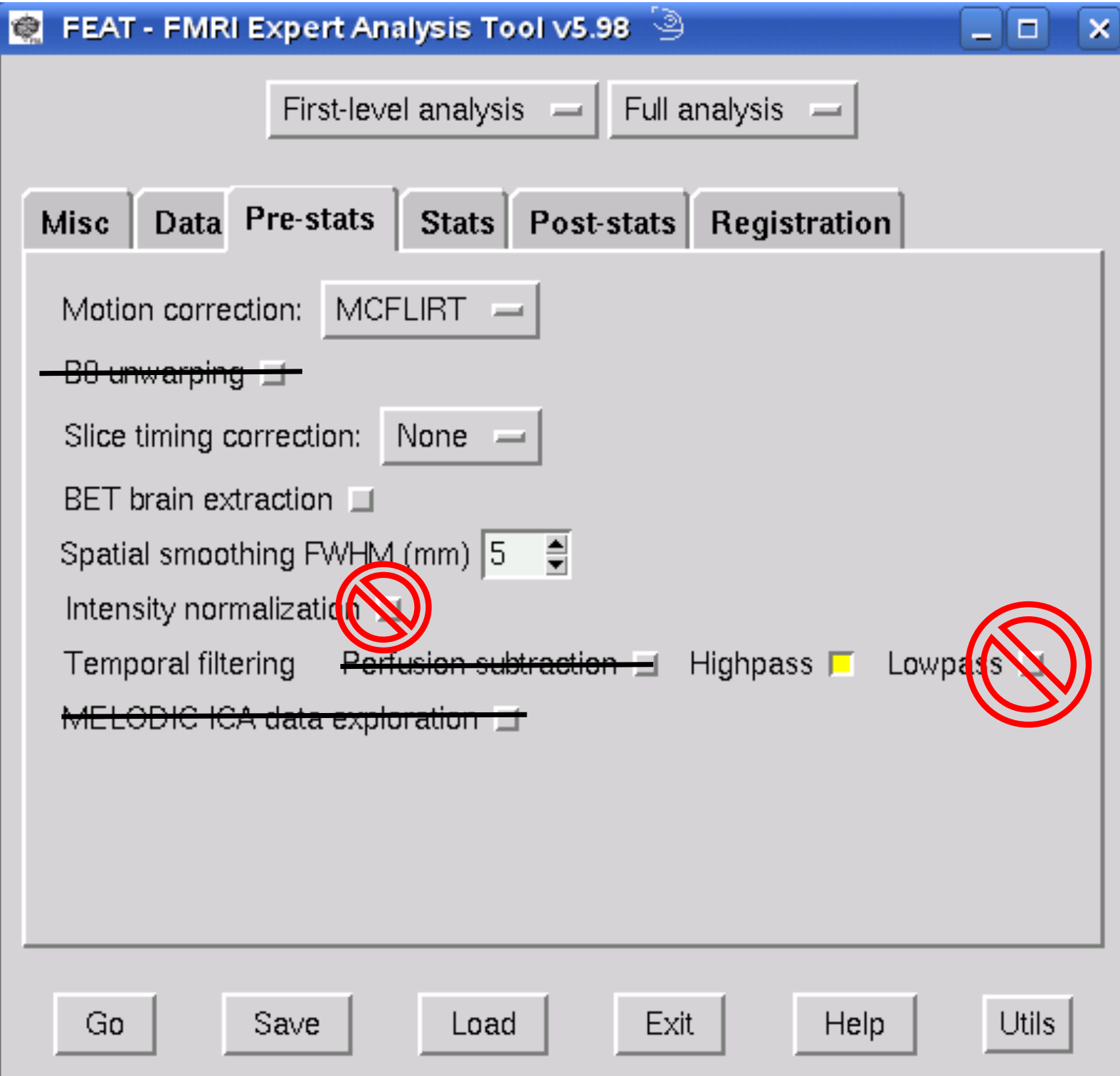
Overview of single session fMRI

- The data is a set of 4D functional time series
 - Many thousands of spatial locations (voxels)
 - Each voxel has a time course defined by a single intensity value per TR (= per volume acquired)
- The task is to model the changes in image intensity over time *separately* for each voxel
 - “mass univariate approach”
 - begin with a set of regressors (“design matrix” / model)
 - regressors usually reflect the time course of experimental conditions
 - find the best linear combination of regressors to explain each voxel time course (basically, multiple regression)
- Before modelling the 4D time series a number of preprocessing steps are applied to the data
 - remove unwanted sources of variation from the time series
 - increase the signal to noise ratio

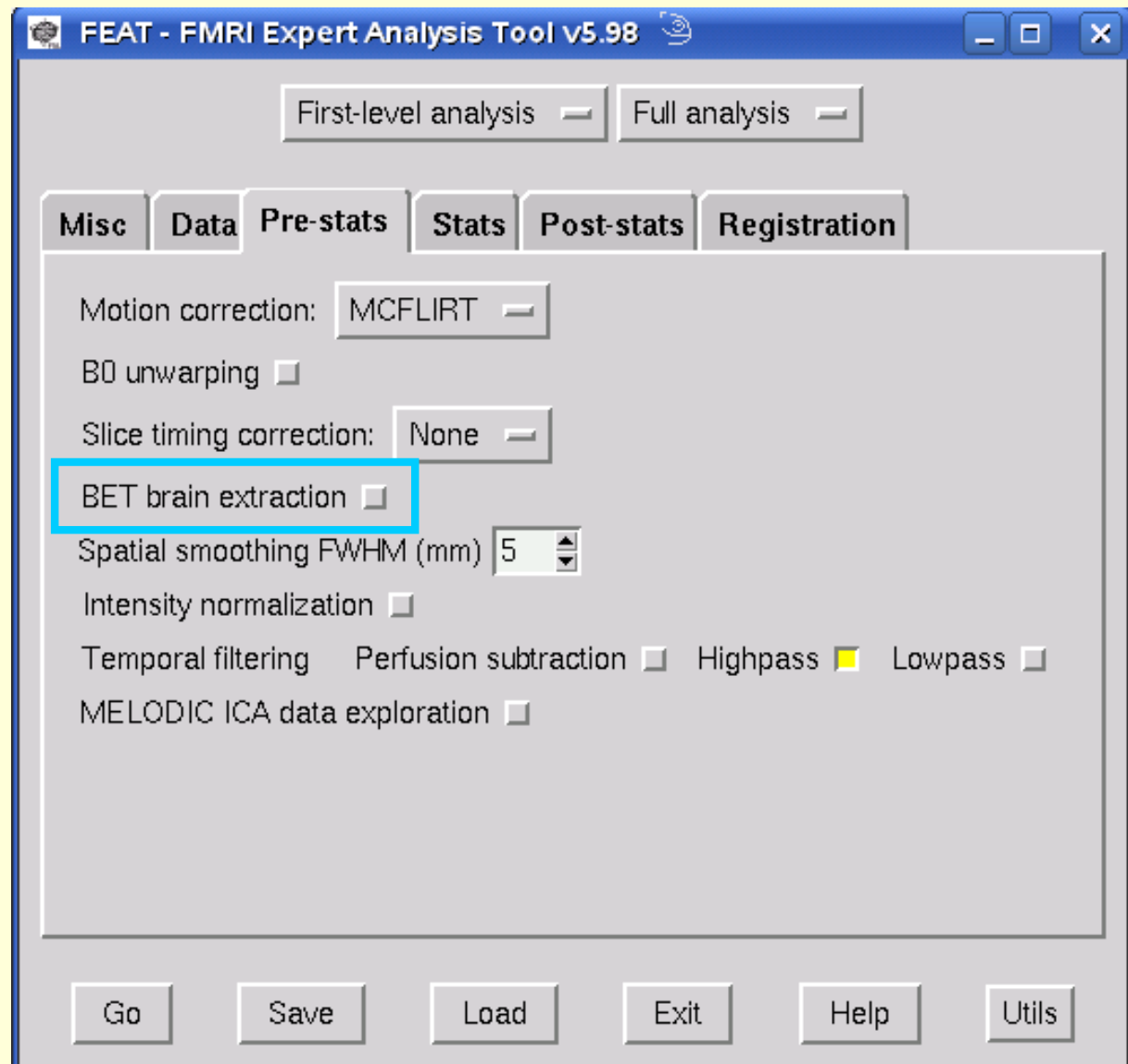


Single-Session Analysis

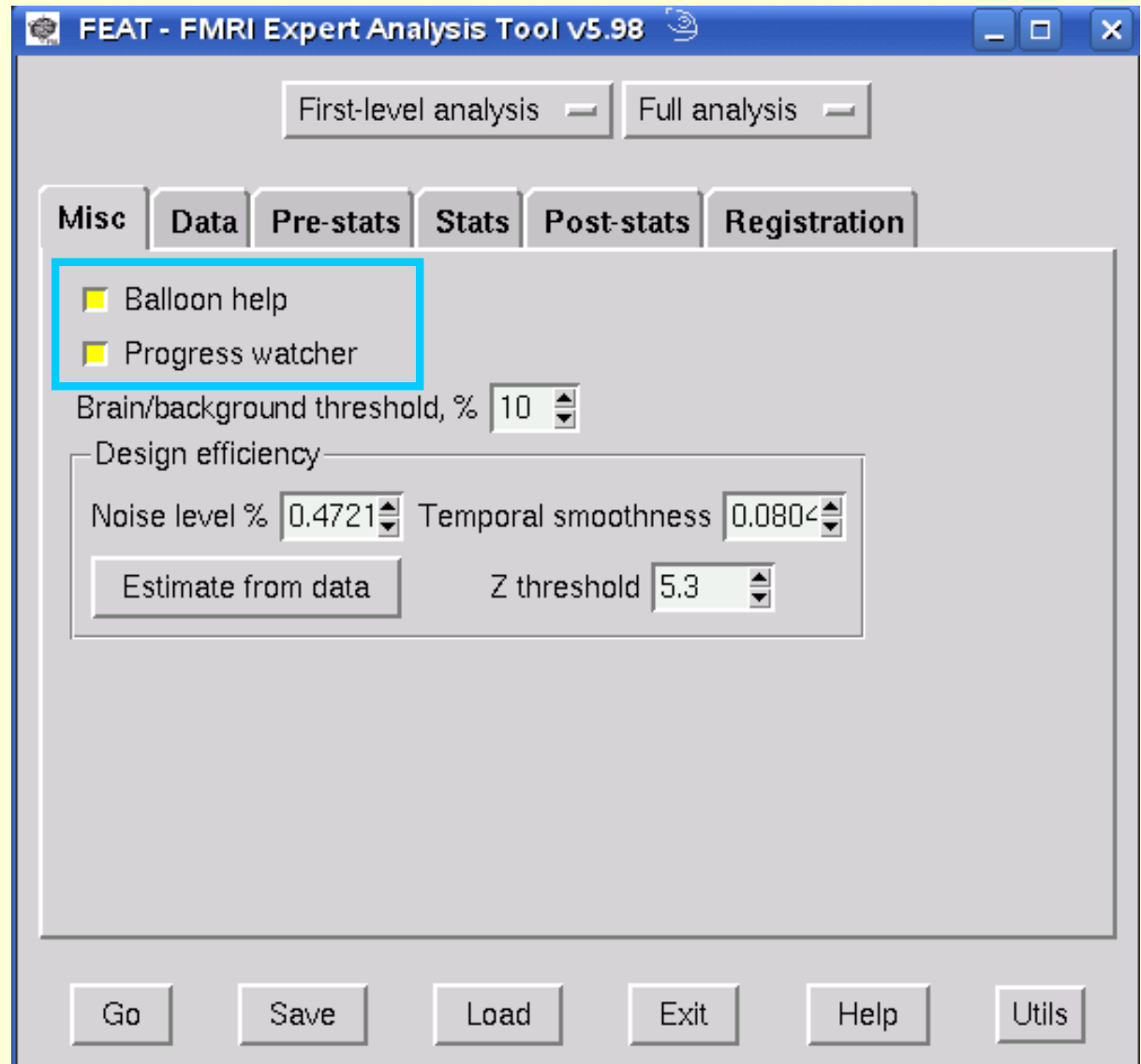




- The BET brain extraction option refers to the 4D functional series
- This will not run BET on the structural
- Normally turn this on
- Brain extraction for the structural image that will be used as the target for registration has to be performed by you, *before* you use FEAT to set up the processing pipeline



- On the Misc tab you can toggle balloon help
- Balloon help will tell you what most of the options mean if you hover the mouse over a button or tickbox
- But it gets annoying after a while
- If progress watcher is selected then FEAT opens a web browser that shows regular updates of the stage your analysis has reached

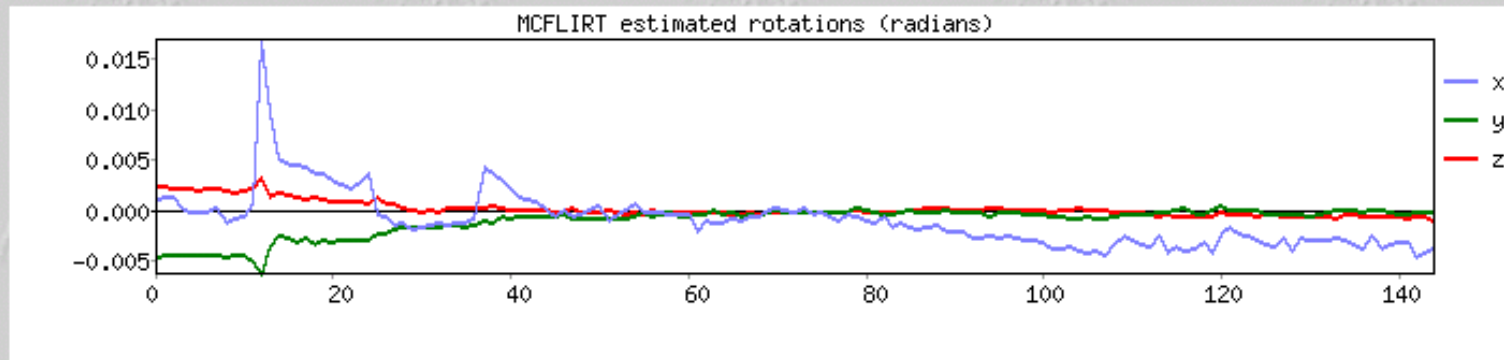


Motion Correction: MCFLIRT

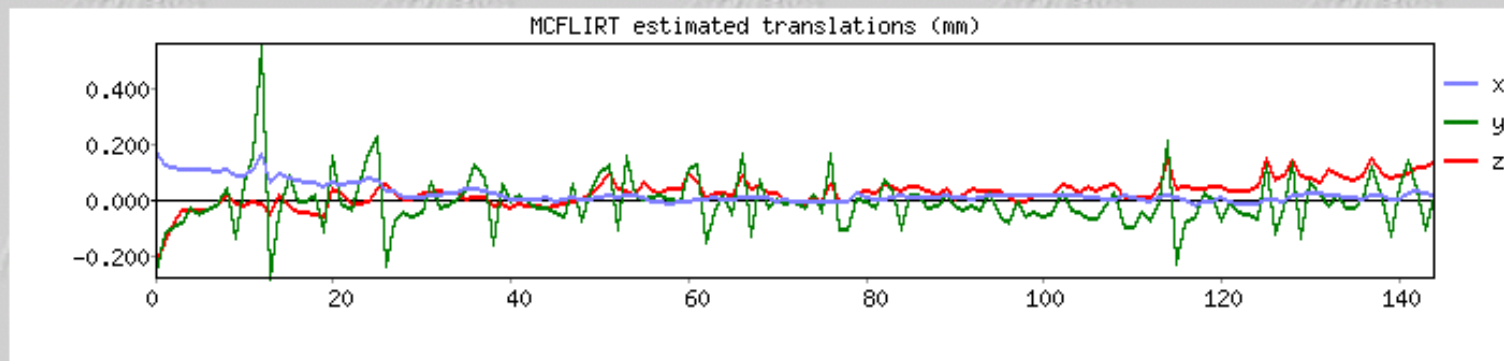
- Aims to make sure that there is a consistent mapping between voxel X,Y,Z position and actual anatomical locations
- Each image in the 4D series is registered using a 6DOF spatial transform to the reference image (by default, the series mid point)

MCFLIRT output

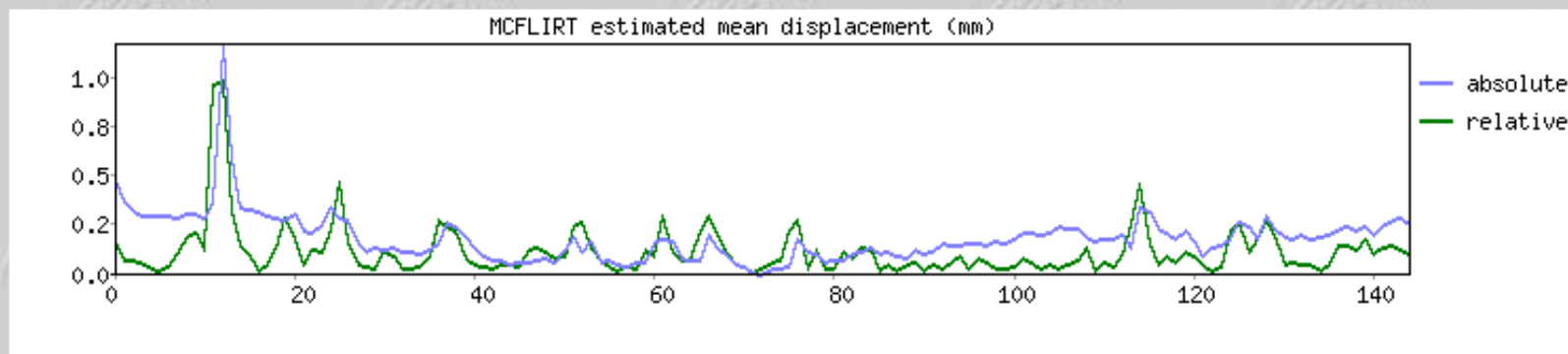
Mean displacements: absolute=0.18mm, relative=0.11mm



Head
rotation

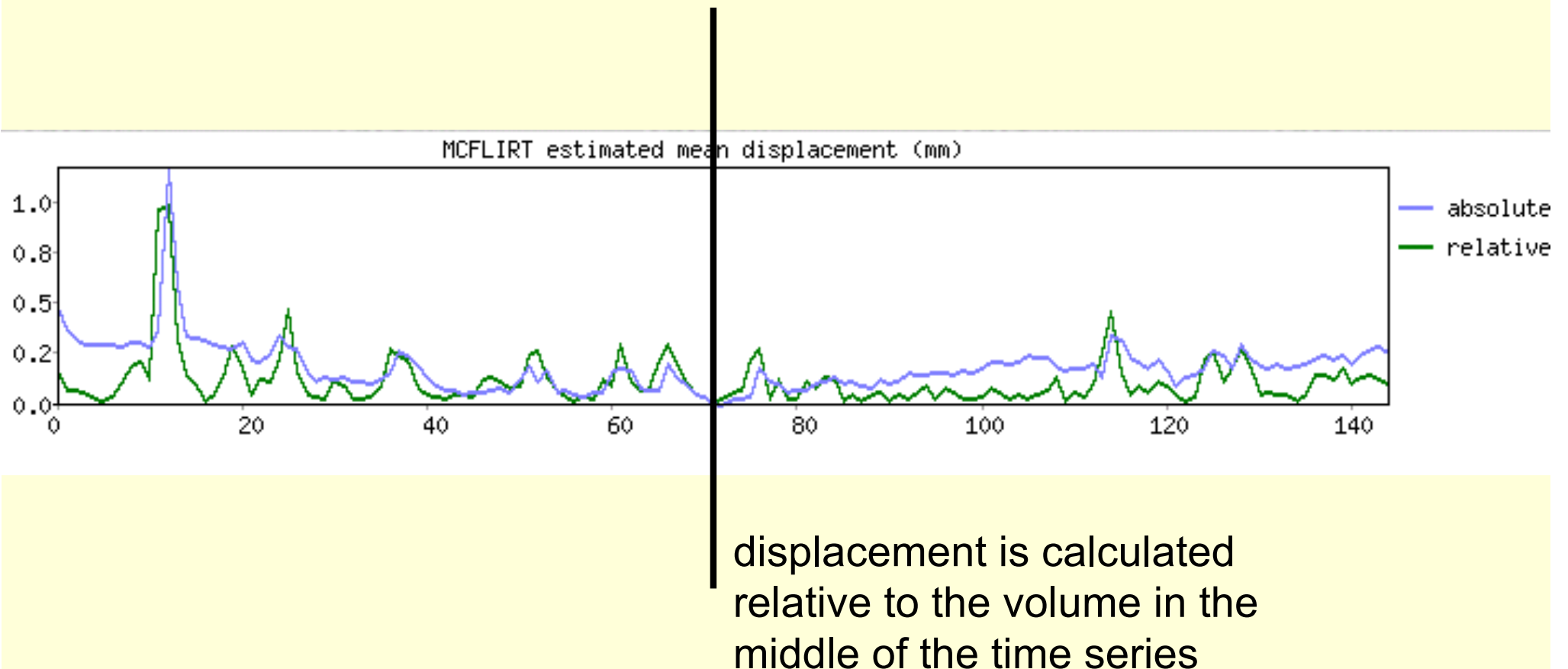


Head
translation



Total
displacement

The displacement plot



Relative displacement is head position at each time point relative to the previous time point. Absolute displacement is relative to the reference image.

Slice timing correction

- Each functional volume that forms part of the 4D time series is made up of slices
- Each slice is acquired at a different point in time relative to the start of the TR
 - e.g., slice 1 at 100 msec, slice 2 at 200 msec, etc
- For each slice, it's the same time point relative to the start of the TR in every volume
- So, the interval between successive acquisitions is constant for every voxel
- But the actual time of acquisition is different for every slice
- *The model of the time course assumes that within each volume, every slice was acquired simultaneously at the mid point of the TR*
 - *so, the model is likely to fit better for one slice than all the others (bad)*
- To use slice timing correction, you will need to tell FSL the order your slices were acquired in
 - interleaved is the most common, but ask your scanner technician!
 - Adjustment is to the middle of the TR period

Slice timing correction

- For each voxel, slice-timing correction examines the time course and shifts it by a small amount
- This requires interpolating between the time points you *actually* sampled to infer a more detailed version of the time course
- The more detailed time course can have small shifts applied to it that are slightly different for each voxel, depending on the actual order the slices were acquired in
- *This allows you to make the assumption in your modelling that every voxel in each volume was acquired simultaneously*

Slice timing correction

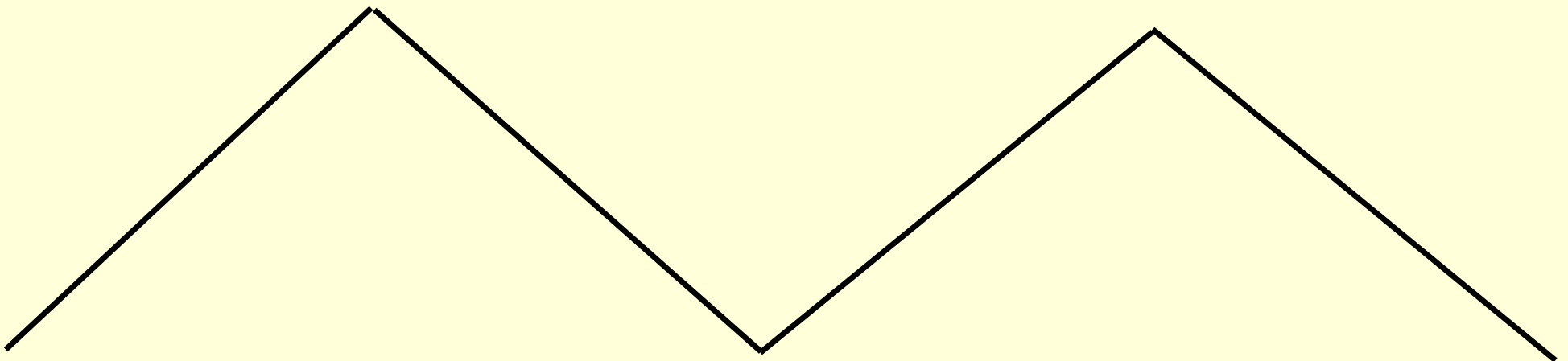
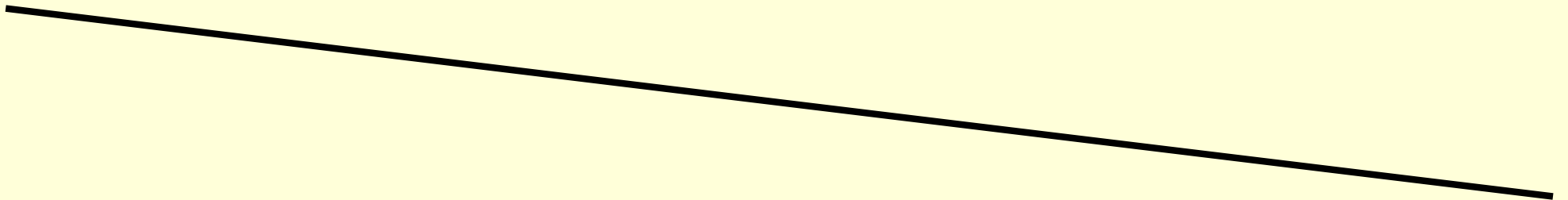
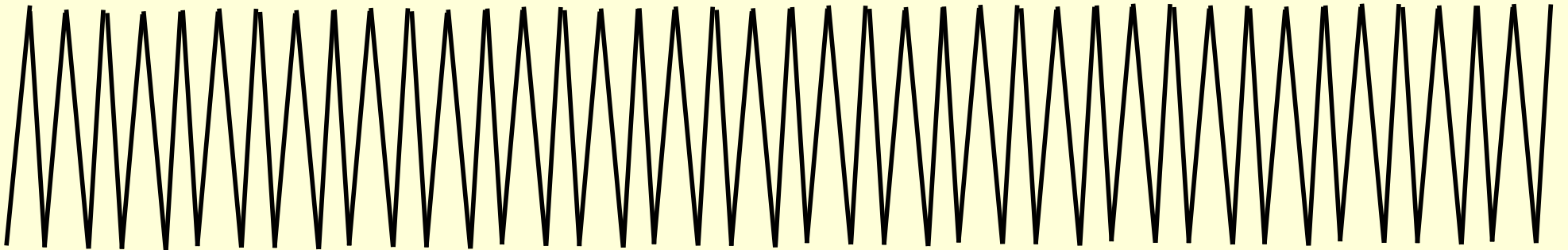
- The problem this tries to solve is more severe if you have a longer TR (e.g. 4 seconds)
 - two adjacent slices in an interleaved sequence could be sampled almost 2 seconds apart
- But temporal interpolation also becomes dodgy with longer TR's 😞
- For block designs (stimuli that are long relative to the TR, e.g. TR = 2 sec, stimulus lasts 16 sec) slice timing errors are not a significant factor influencing the fitting of a model to the data
- For event related designs (brief stimuli separated by variable pauses), slice timing correction is important
- People argue about whether to do slice timing correction before or after motion correction
 - FSL does motion correction first
 - some people advise against any slice timing correction

Temporal filtering

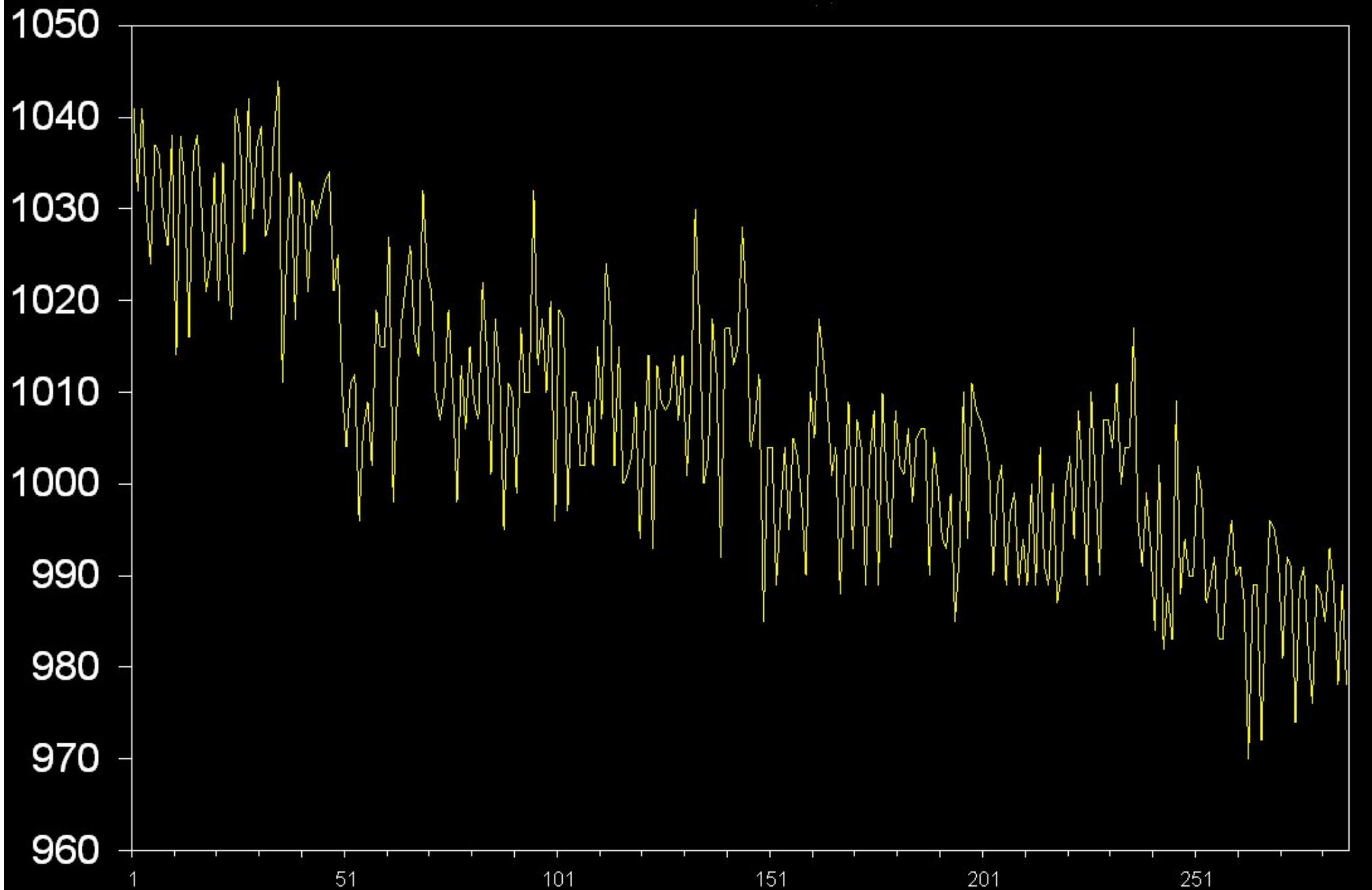
- Filtering in time and/or space is a long-established method in any signal detection process to help "clean up" your signal
- The idea is if your signal and noise are present at separable frequencies in the data, you can attenuate the noise frequencies and thus increase your signal to noise ratio

AAAAAAAAAVVVVVVRRRRRRRAAAAAAAAAAVVVVVVRRRRRRRR

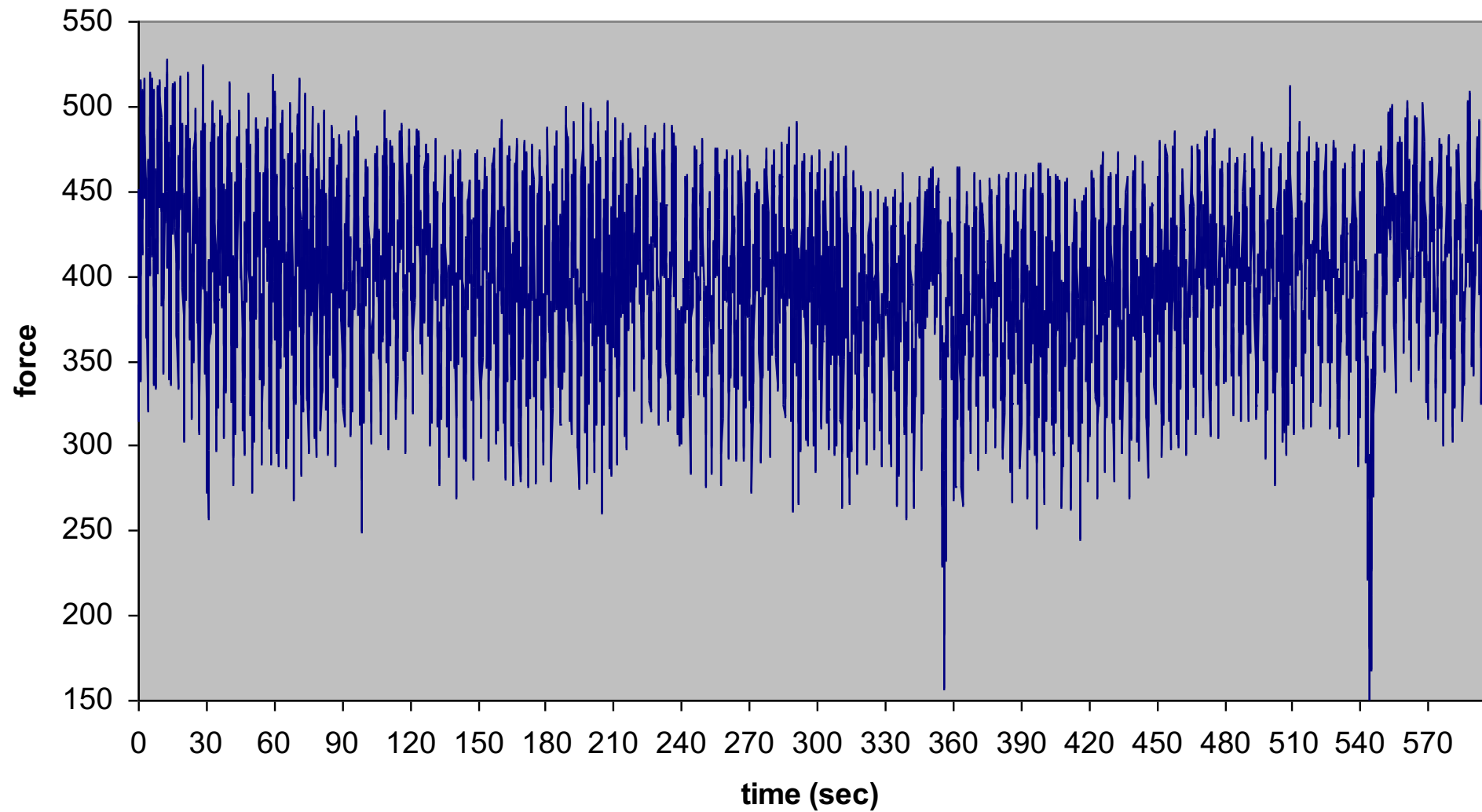
Time



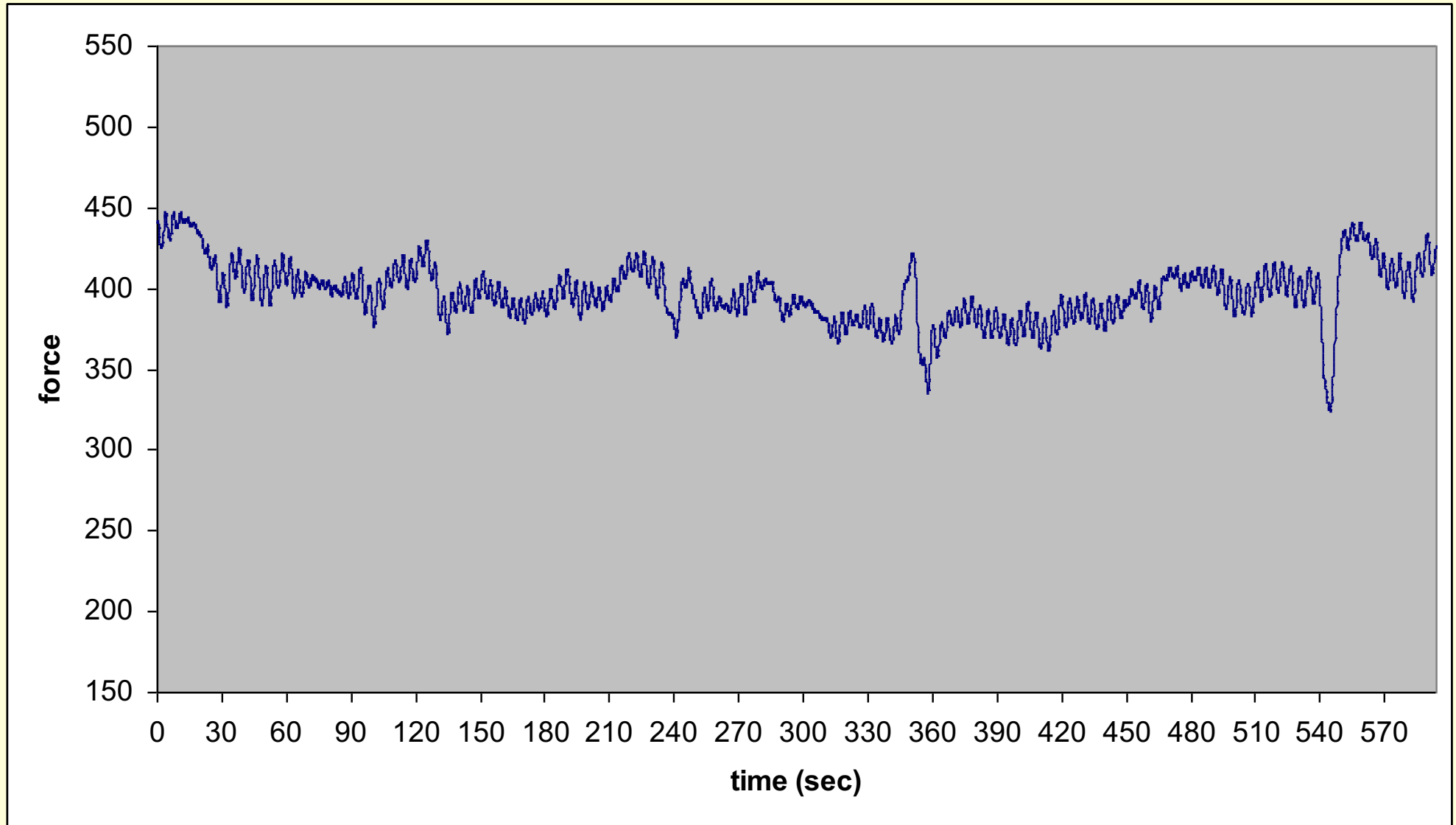
AAAAAAAAVVVVVRRRRRRRAAAAAAAAAVVVVVRRRRRRR



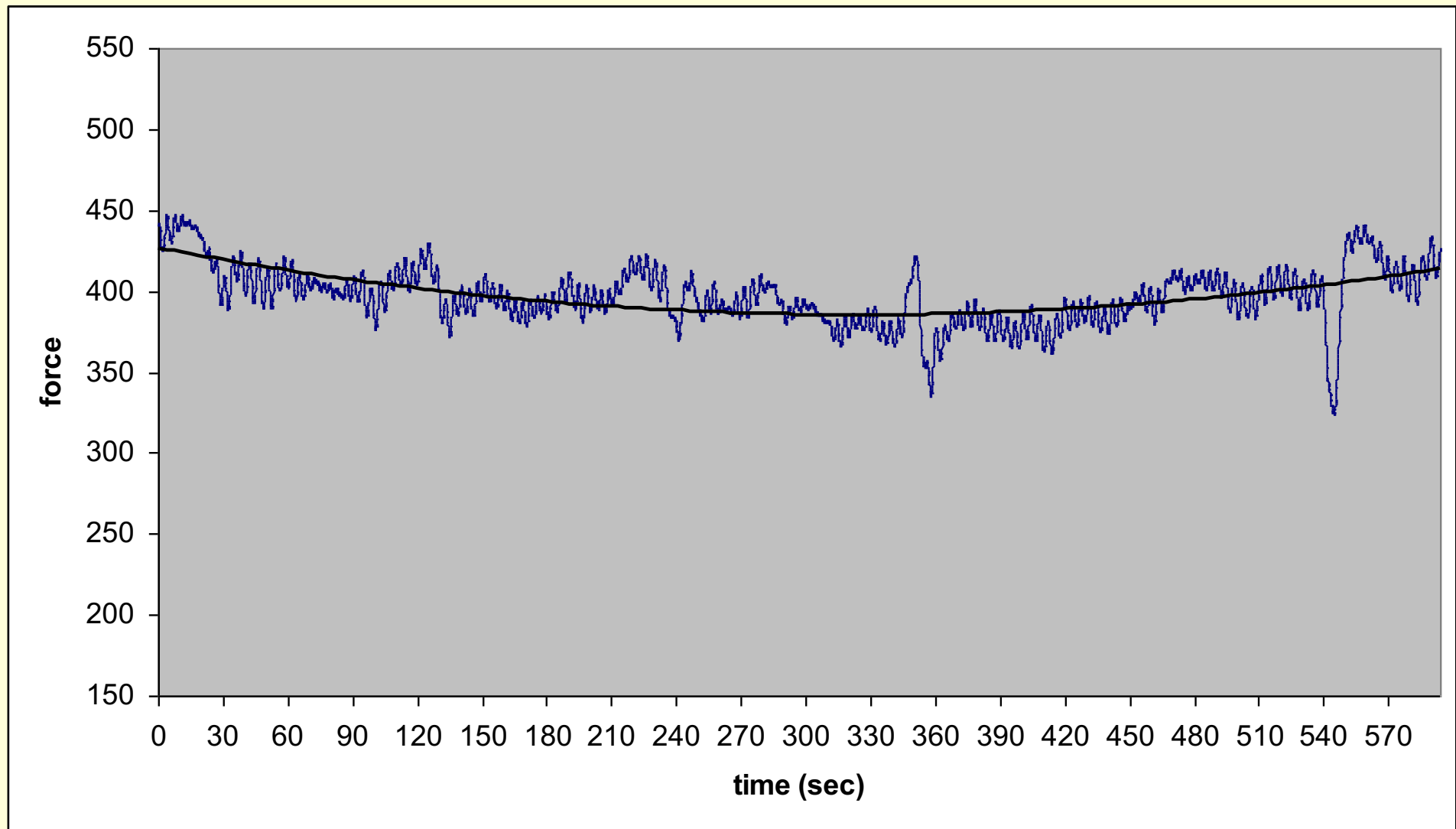
Raw data



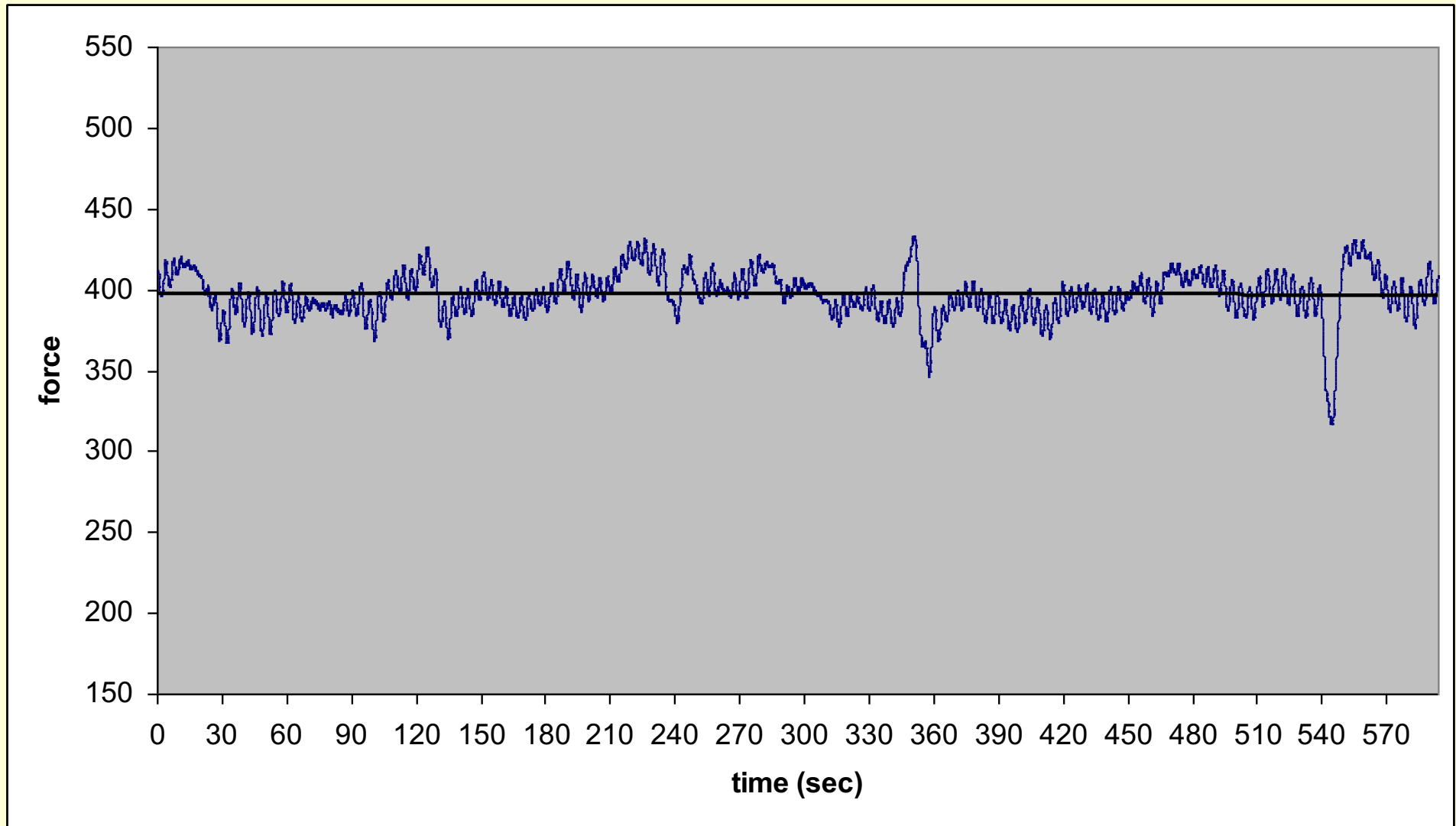
After low pass filter



Very low frequency component, suggests that high pass filter also needed



Low and high frequencies removed



Setting the high pass filter for FMRI

- The rule recommended by FSL is the lowest setting for the high pass filter should be equal to the duration of a single cycle of the design
 - In an ABCABCABC block design, it will be equal to ABC
- If you set a shorter duration than this you will remove signal that is associated with the experiment
- In complicated event related designs without a strongly periodic structure there is a subjective element to the setting of the high pass filter
- Do not use experimental designs with many conditions where the duration of a single experimental cycle is very long
 - e.g. ABCDEABCDE, where ABCDE = 300 seconds
 - setting the high pass to 300 sec will allow a lot of the low frequency FMRI noise to pass through to the modelling stage
 - Furthermore, that noise can easily become correlated with the experimental time course because you are using an experimental time course that has a similar frequency to noise in the data
 - In any signal detection experiment, not just FMRI, you need to make sure that signal of interest and noise are present at different frequencies

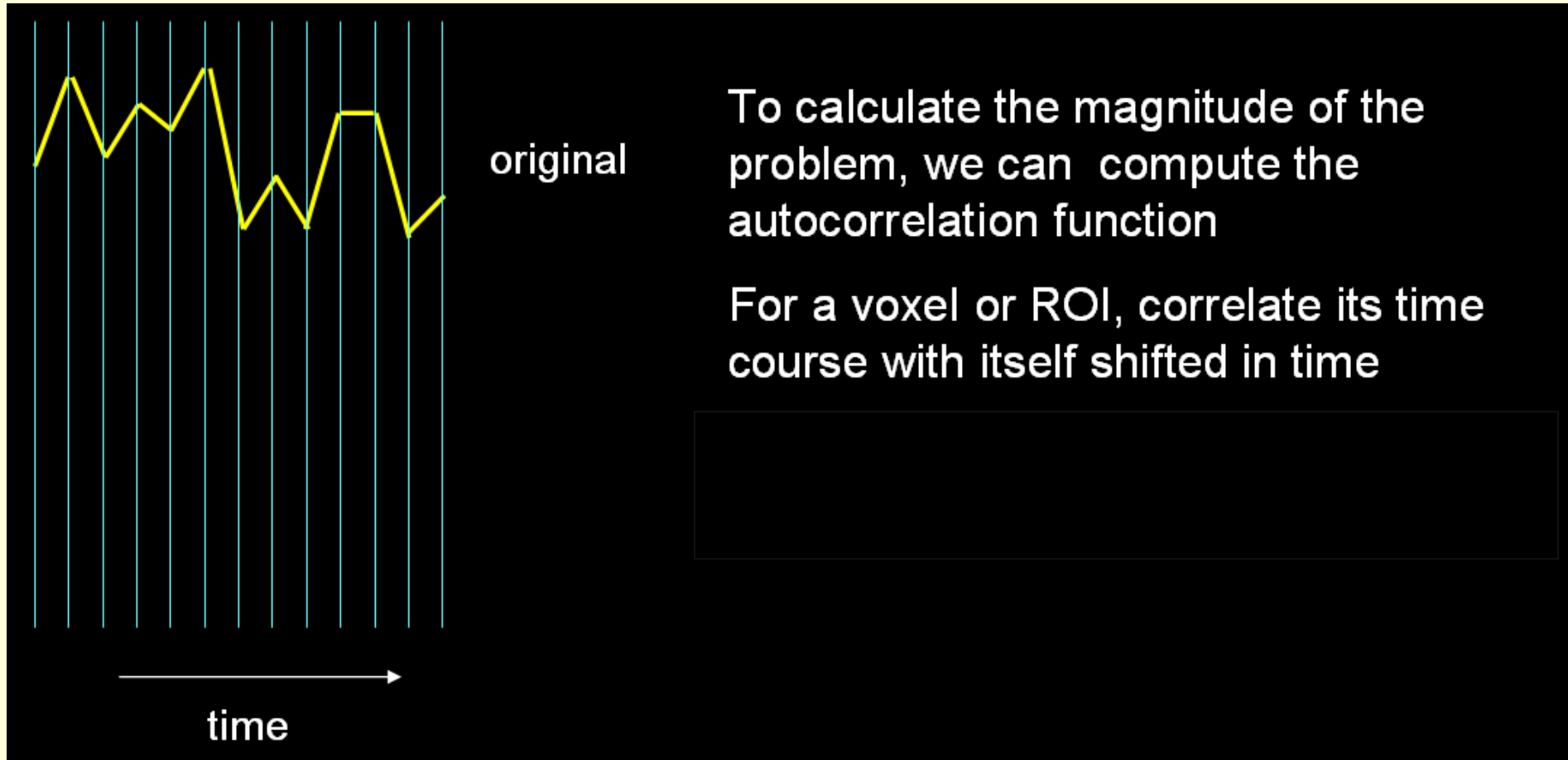
Low pass filter?

- As well as removing oscillations with a longer cycle time than the experiment, you can also elect to remove oscillations with a higher cycle time than the experiment
- In theory this should enhance signal and reduce noise, and it was practiced in the early days of FMRI
- However, it has now been demonstrated that because FMRI noise has temporal structure (i.e. it is not white noise), the low pass filter can actually enhance noise relative to signal
- The temporal structure in the noise is called “temporal autocorrelation” and is dealt with in FSL using FILM prewhitening instead of low pass

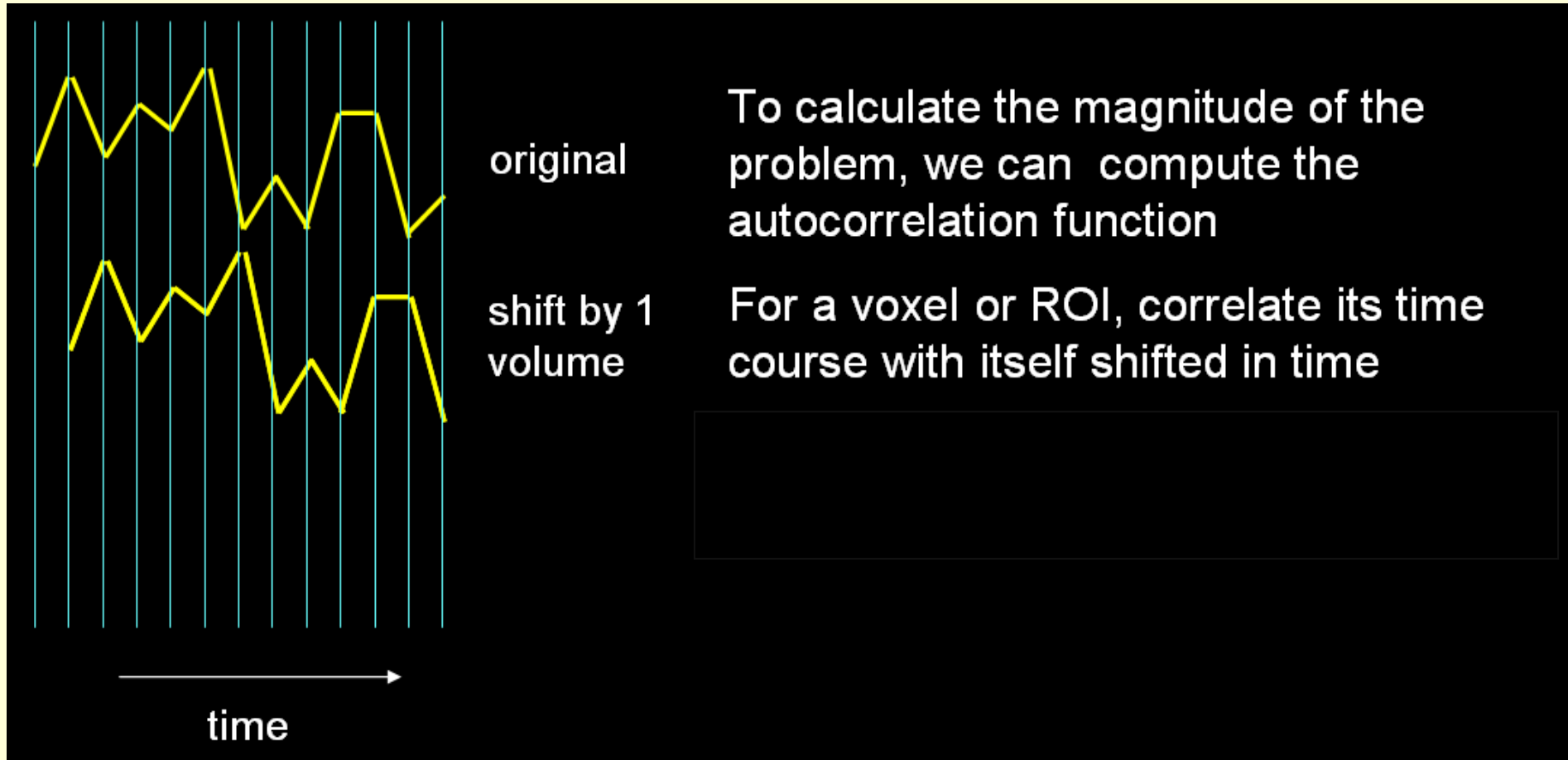
Temporal autocorrelation

- The image intensity value in a voxel at time t is partially predictable from previous time points
 - why is this a problem?
- Because the final stage of single session analysis consists of voxelwise statistical comparisons, using t or F contrasts, between experimental conditions
 - assessing the p value of t or F requires the degrees of freedom (DOF) to be calculated
 - DOF is the number of *independent* observations
 - for a voxel timecourse, if you know the mean intensity, then naively the DOF is the number of time points -1
- *But the presence of autocorrelation means that the time points are not independent , so DOF is less than $n-1$*
 - *using $n-1$ would inflate statistical significance*

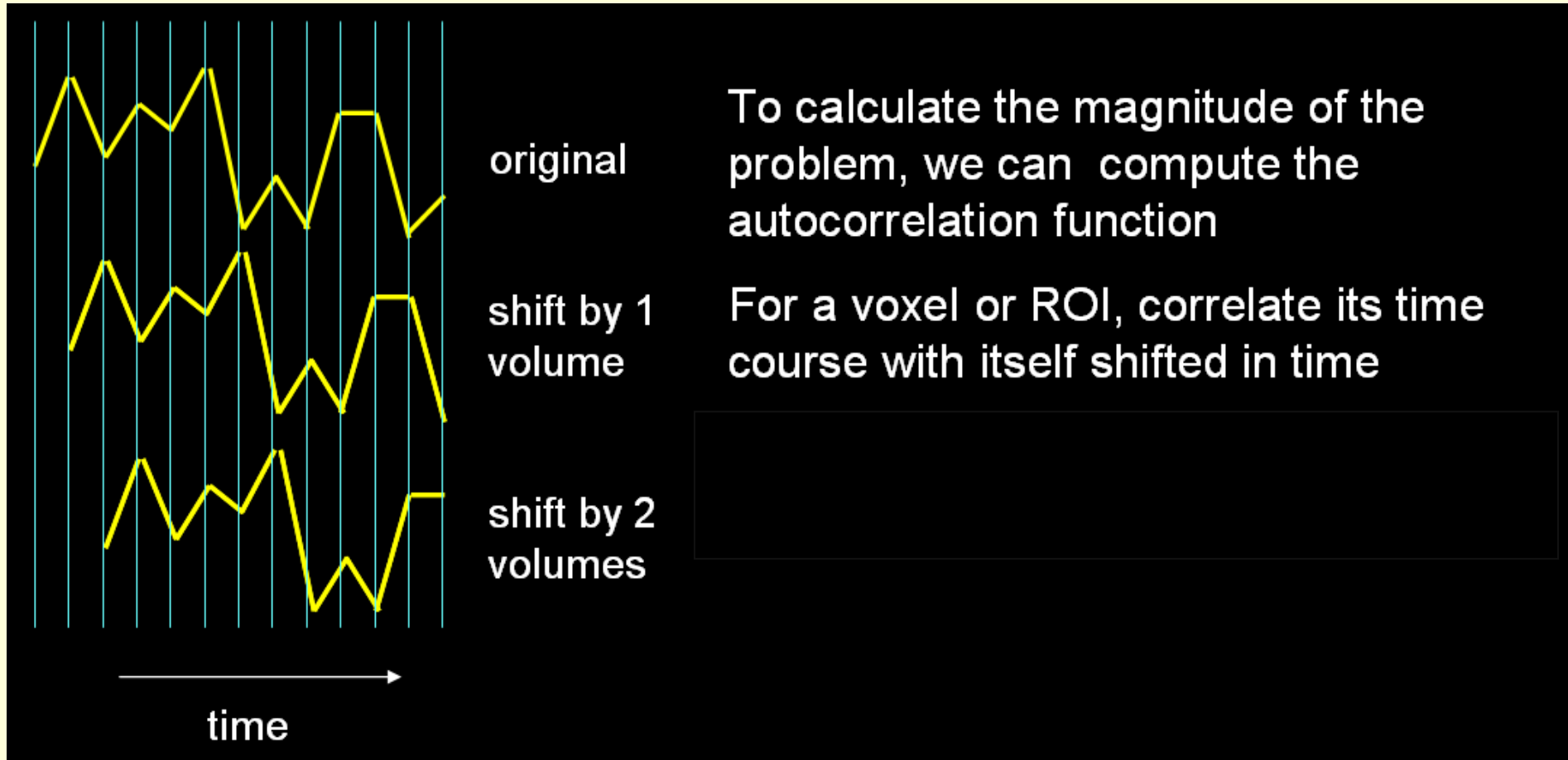
Measuring autocorrelation



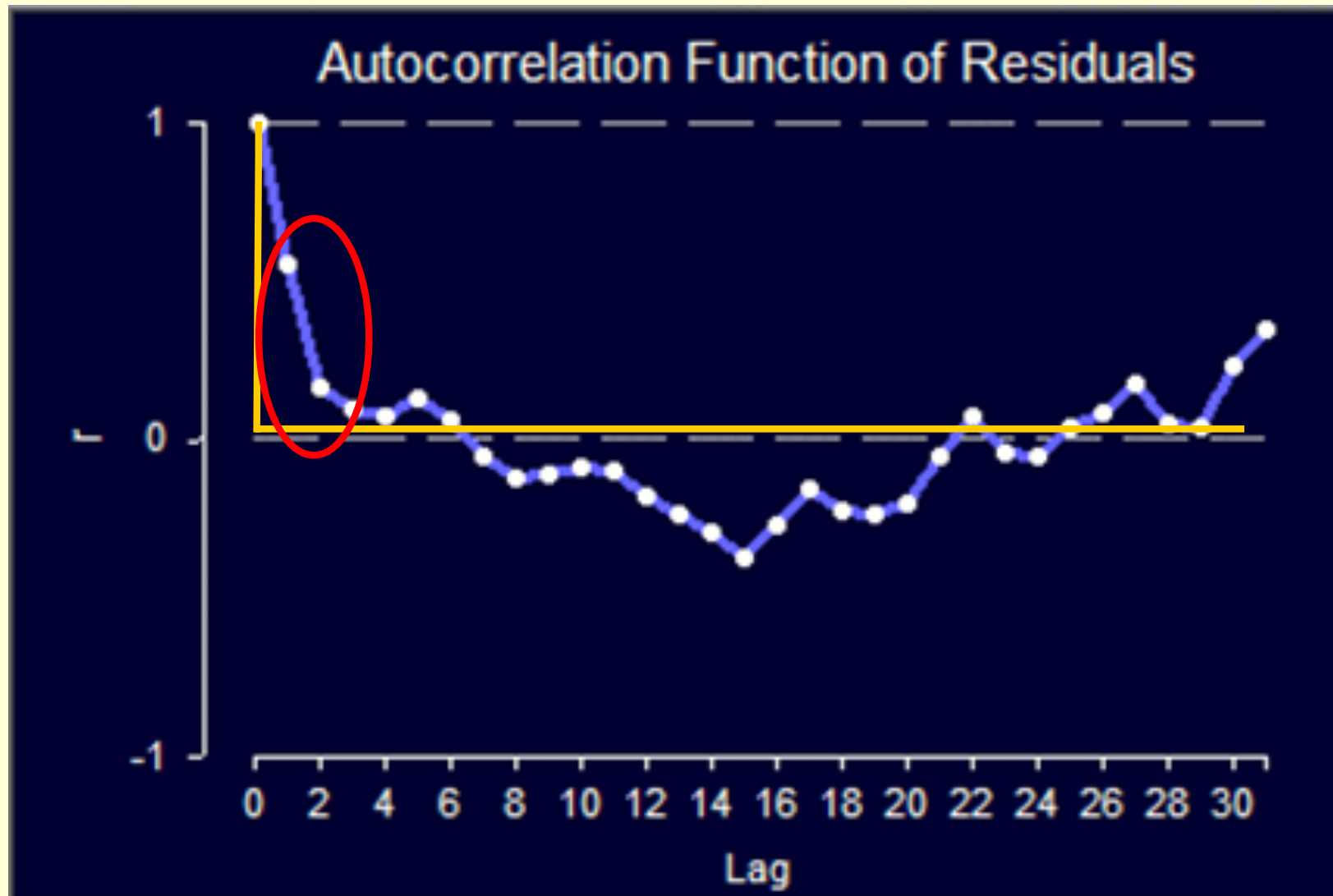
Measuring autocorrelation



Measuring autocorrelation



Plot the correlation against the degree of shift



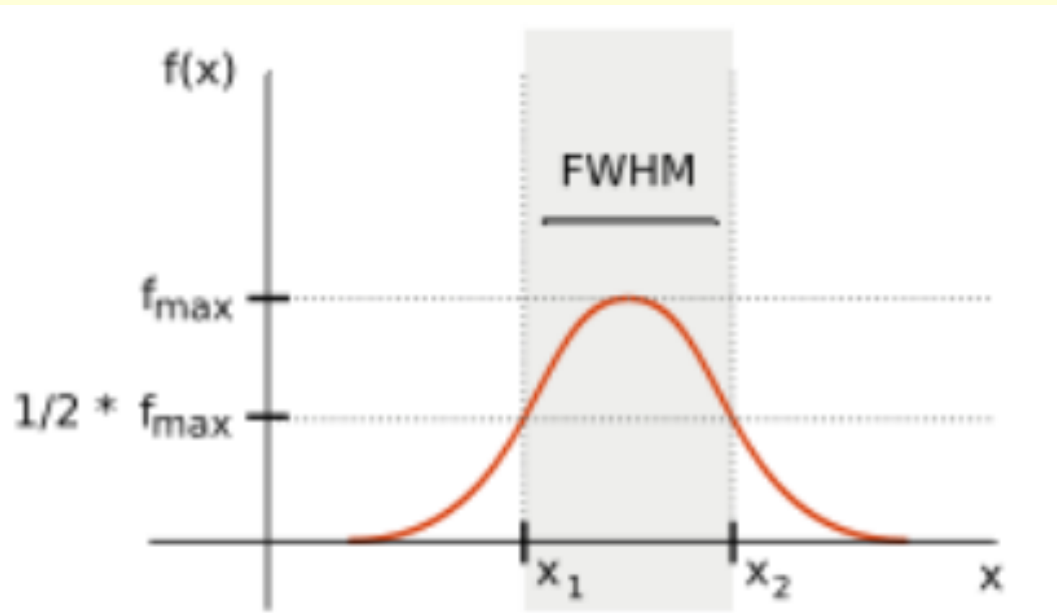
Temporal autotocorrelation

- The value of a voxel at time t is partially predictable from nearby time points about 3-6 seconds in the past
 - this correlation arises from temporal structure in FMRI noise
 - but we need to assume that the noise is independent at each time point so that $\text{DOF} = n-1$
 - this problem is addressed during the modelling stage (see later)
- Voxel values are also predictable from more distant time points due to low frequency noise
 - but the high pass filter removes most of this problem

Spatial smoothing

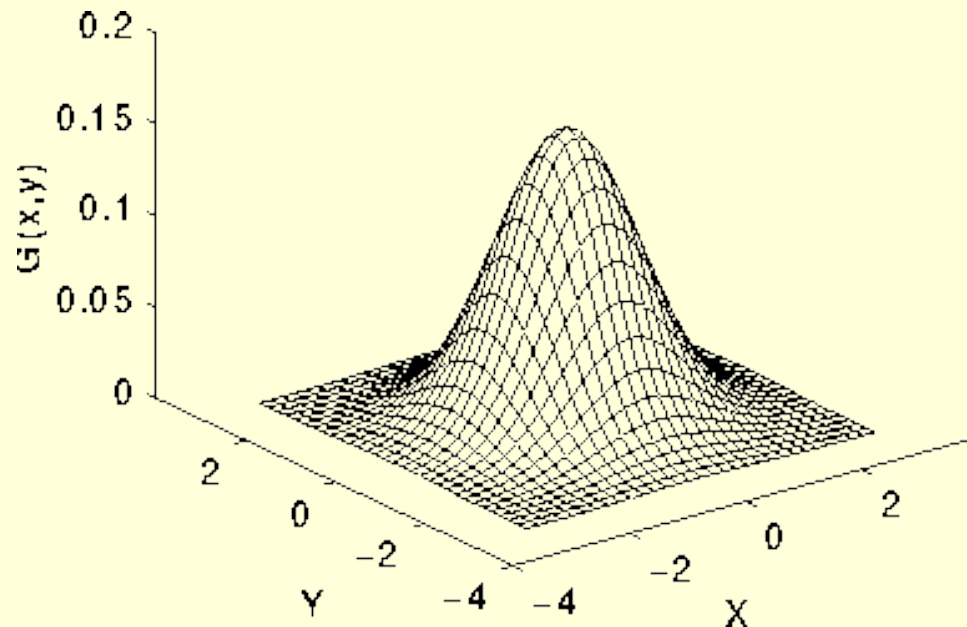
- FMRI noise varies across space as well as time
 - spatial smoothing is a way of reducing spatial noise and thereby increasing the ratio of signal to noise (SNR) in the data
- Unlike FMRI temporal noise, FMRI spatial noise is more like white noise, making it easier to deal with
 - it is essentially random, essentially independent from voxel to voxel, and has a mean of about zero
 - therefore if you average image intensity across several voxels, noise tends to average towards zero, whereas signal that is common to the voxels you are averaging across will remain unchanged, dramatically improving the SNR

Spatial smoothing: FWHM

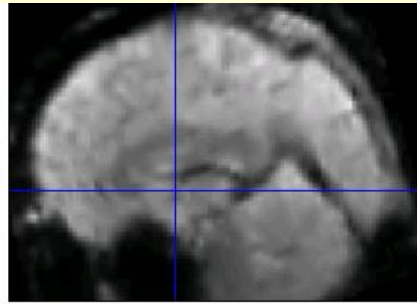
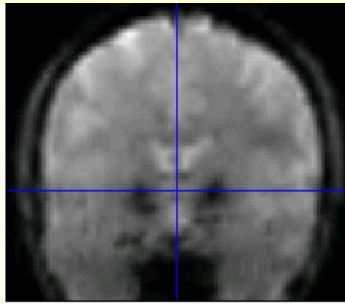


- FSL asks you to specify a Gaussian smoothing kernel defined by its Full Width at Half Maximum (FWHM)
- Find the point on the y axis where the function attains half its maximum value
- Then read off the corresponding x axis values

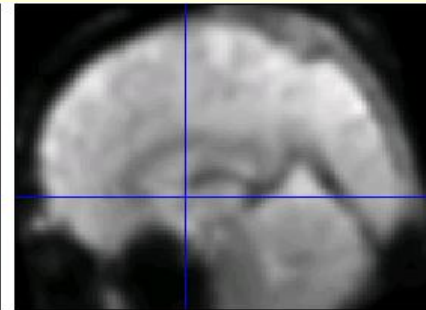
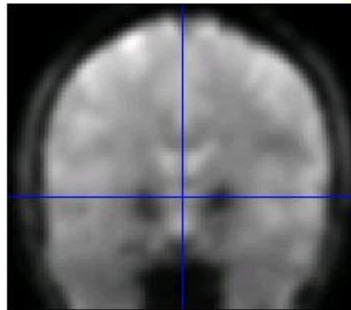
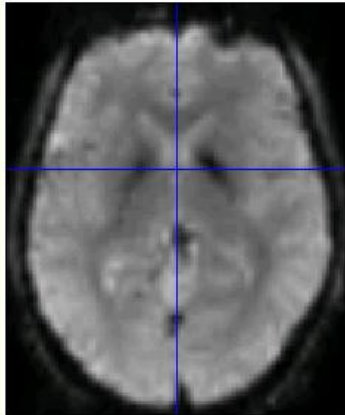
Spatial smoothing: FWHM



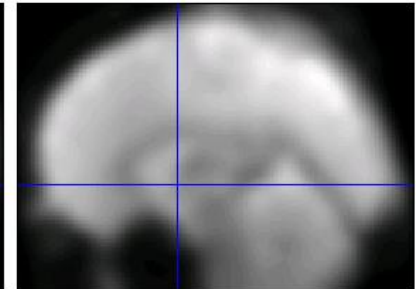
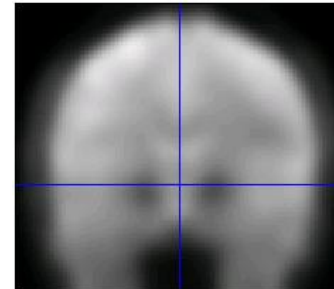
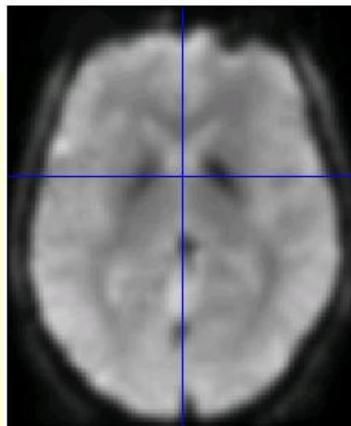
- The Gaussian is centred on a voxel, and the value of the voxel is averaged with that of adjacent voxels that fall under the Gaussian
- The averaging is weighted by the y axis value of the Gaussian at the appropriate distance



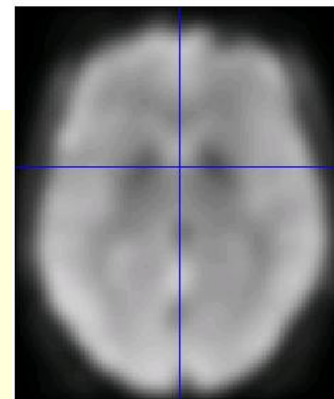
No smoothing



4 mm

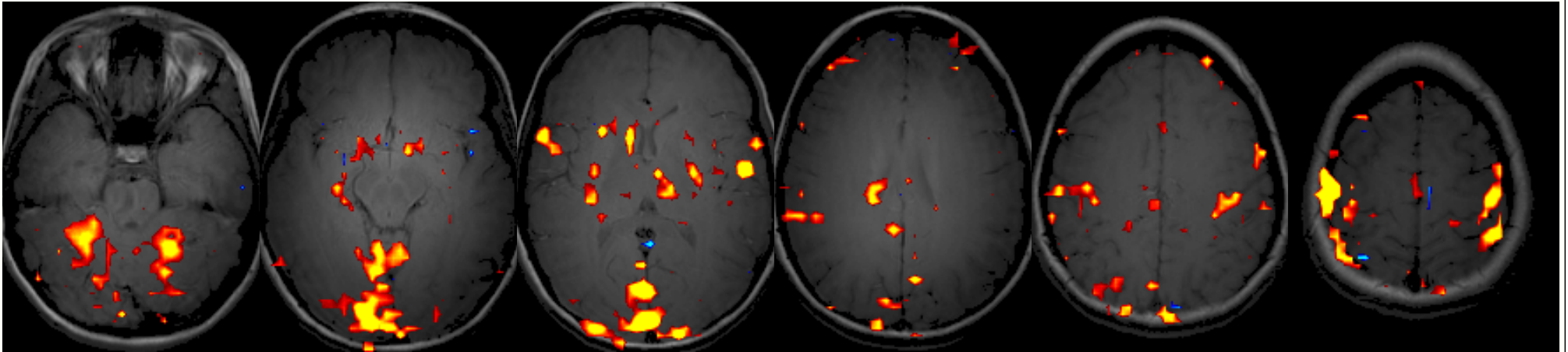


9 mm

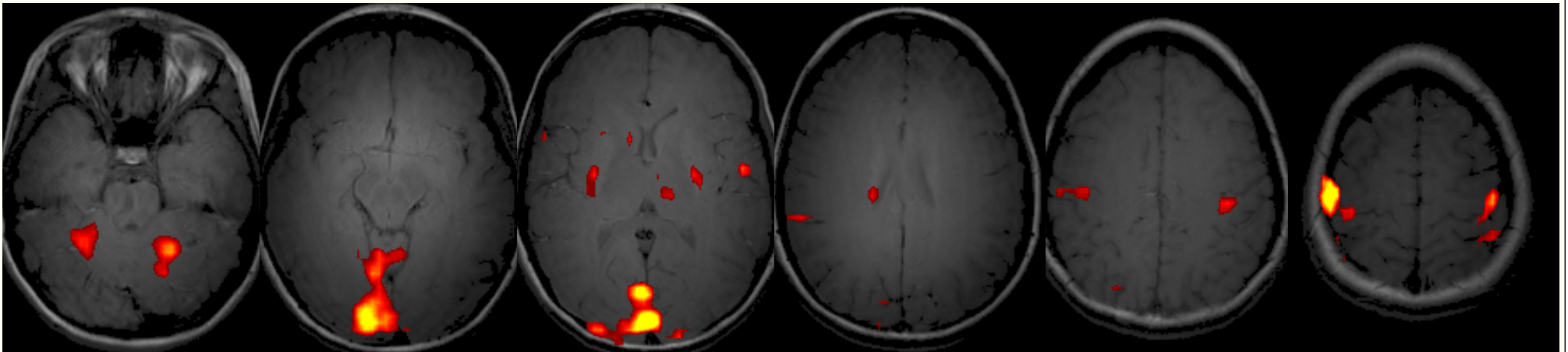


Effects of Smoothing on activations

Unsmoothed Data



Smoothed Data (kernel width 5 voxels)



When should you smooth? When should you not?

- Smoothing is a good idea if
 - You're not particularly concerned with voxel-by-voxel resolution.
 - You're not particularly concerned with finding small (less than a handful of voxels) clusters
 - You want (or need) to improve your signal-to-noise ratio
 - You're averaging results over a group, in a brain region where functional anatomy and organization isn't precisely known
 - You want to use p-values corrected for multiple comparisons with Gaussian field theory (as opposed to False Discovery Rate)
 - this is the “Voxel” option in FSL and the “FWE” option in SPM
- Smoothing is not a good idea if
 - You need voxel-by-voxel resolution
 - You believe your activations of interest will only be a few voxels large
 - You're confident your task will generate large amounts of signal relative to noise
 - You're working primarily with single-subject results
 - You're mainly interested in getting region-of-interest data from very specific structures that you've drawn with high resolution on single subjects

How do you determine the size of the kernel?

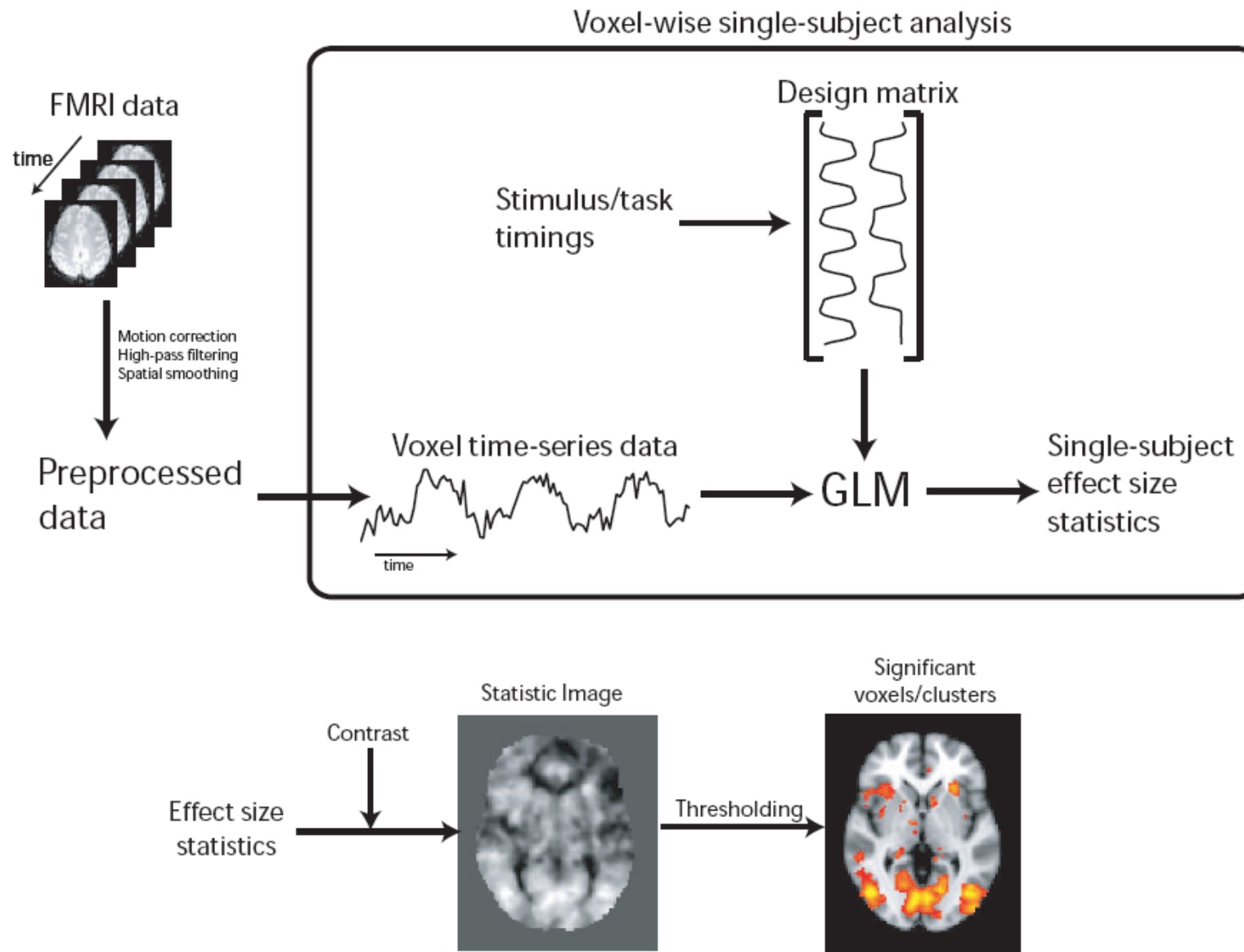
- Based on functional voxel size? Or brain structure size?
 - A little of both, it seems.
- The matched filter theorem, from the signal processing field, tells us that if we're trying to recover a signal (like an activation) in noisy data (like fMRI), we can best do it by smoothing our data with a kernel that's about the same size as our activation.
- Trouble is, though, most of us don't know how big our activations are going to be before we run our experiment
- Even if you have a particular structure of interest (say, the hippocampus), you may not get activation over the whole region - only a part
- A lot of people set FWHM to functional voxel size * 2

Voxel-wise single session modelling

- After the data has been optimised by preprocessing you search for voxels where the time course of image intensity changes is correlated with the experimental time course
 - activation
- This is achieved using the General Linear Model (GLM)
 - similar to multiple regression
- The input to the GLM is the data, plus a set of explanatory variables called the “Design Matrix”
 - sometimes EV’s are included to model sources of variation that are of no interest to the experimenter
 - this is to reduce the residual (error) variance
- The GLM is fitted independently for each voxel timecourse
 - ignores the spatial structure in the brain

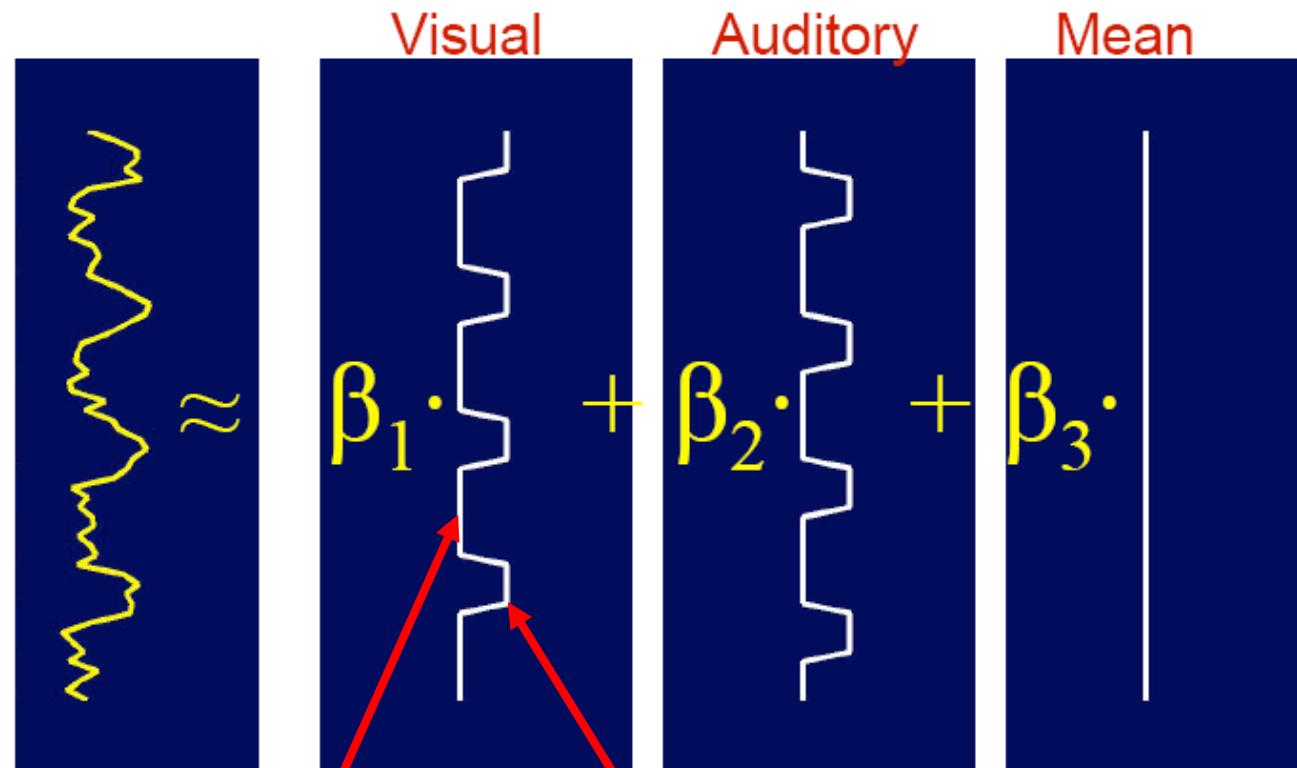


Single-Session Analysis





Linear Combinations of Predicted Responses

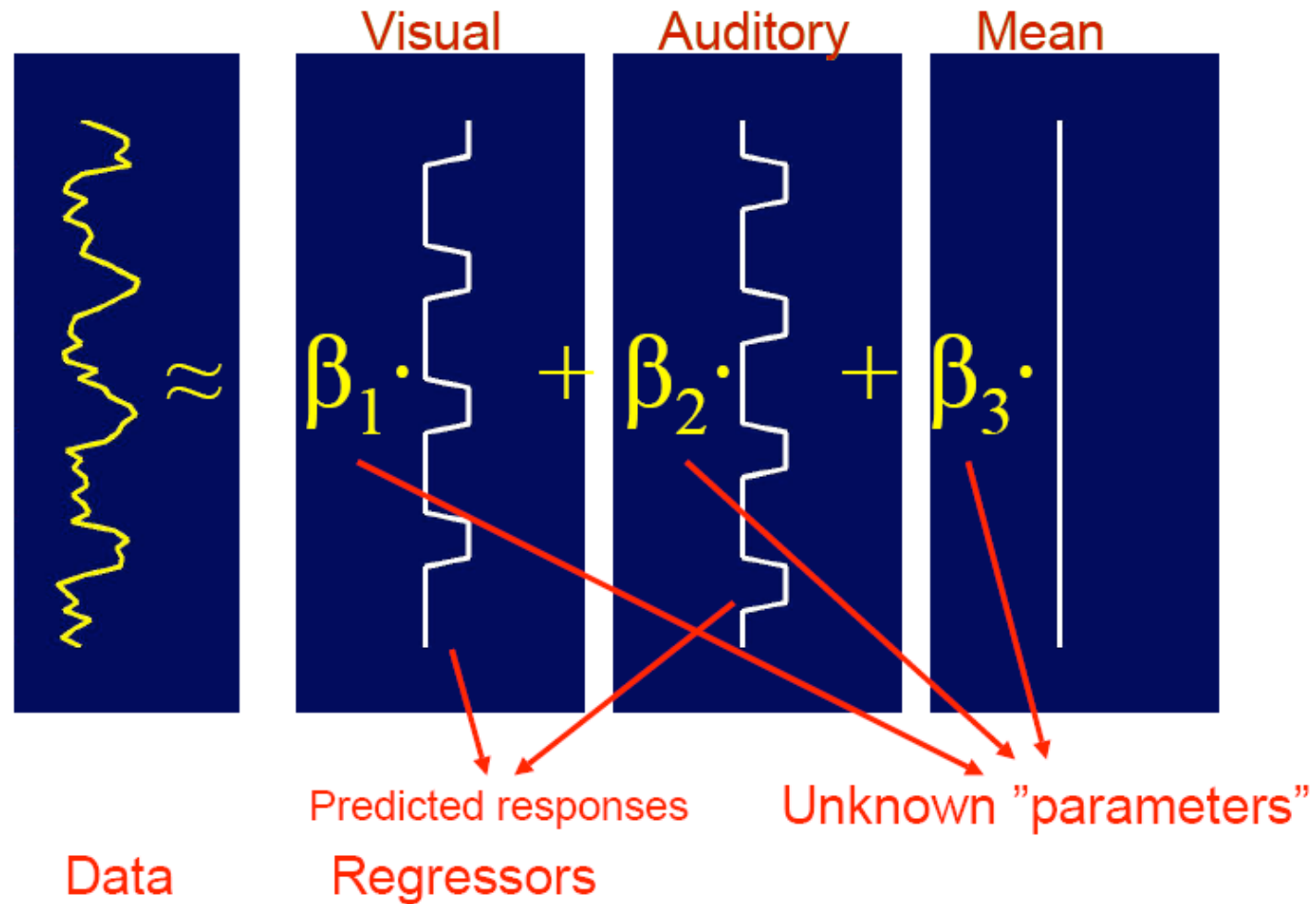


Regressor = 0
(stimulus on)

Regressor = 1
(stimulus off)



Linear Combinations of Predicted Responses





Estimation:

Finding the "best" parameter values

- The estimation entails finding the parameter values such that the linear combination "best" fits the data.

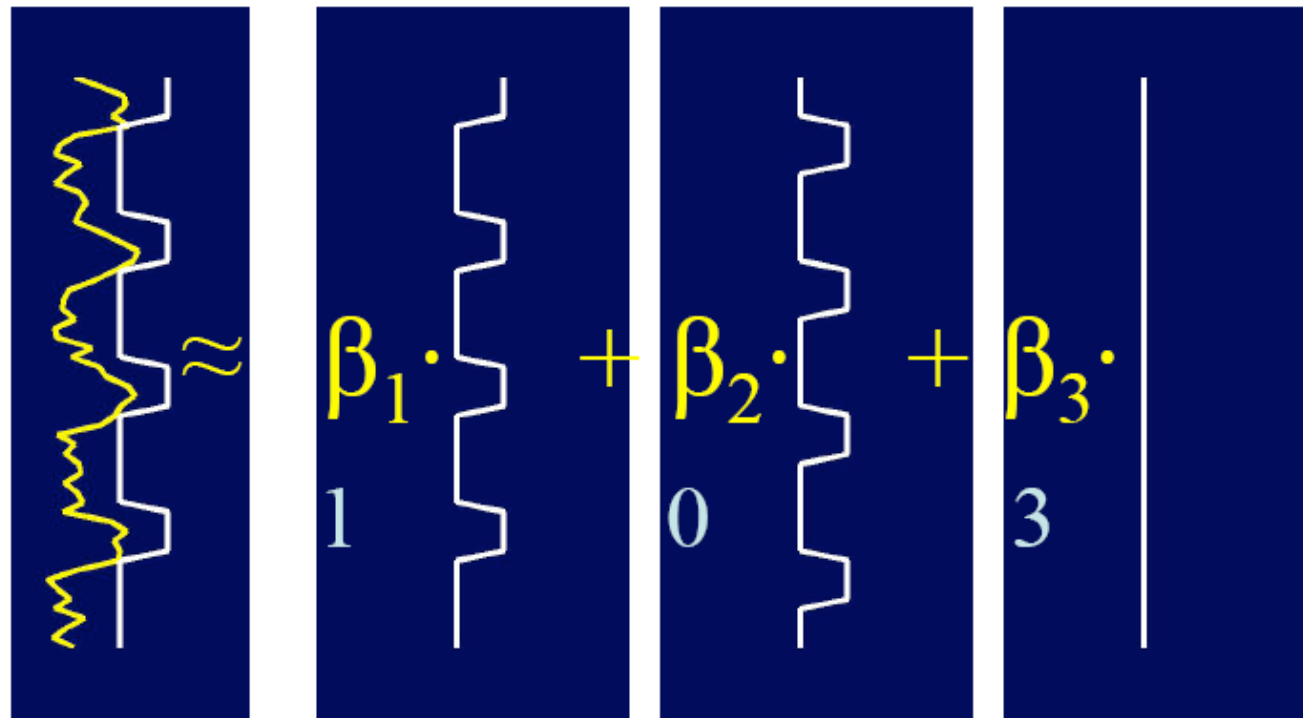
$$\text{Noisy signal} \approx \beta_1 \cdot \text{Basis}_1 + \beta_2 \cdot \text{Basis}_2 + \beta_3 \cdot \text{Basis}_3$$



Estimation:

Finding the "best" parameter values

- The estimation entails finding the parameter values such that the linear combination "best" fits the data.



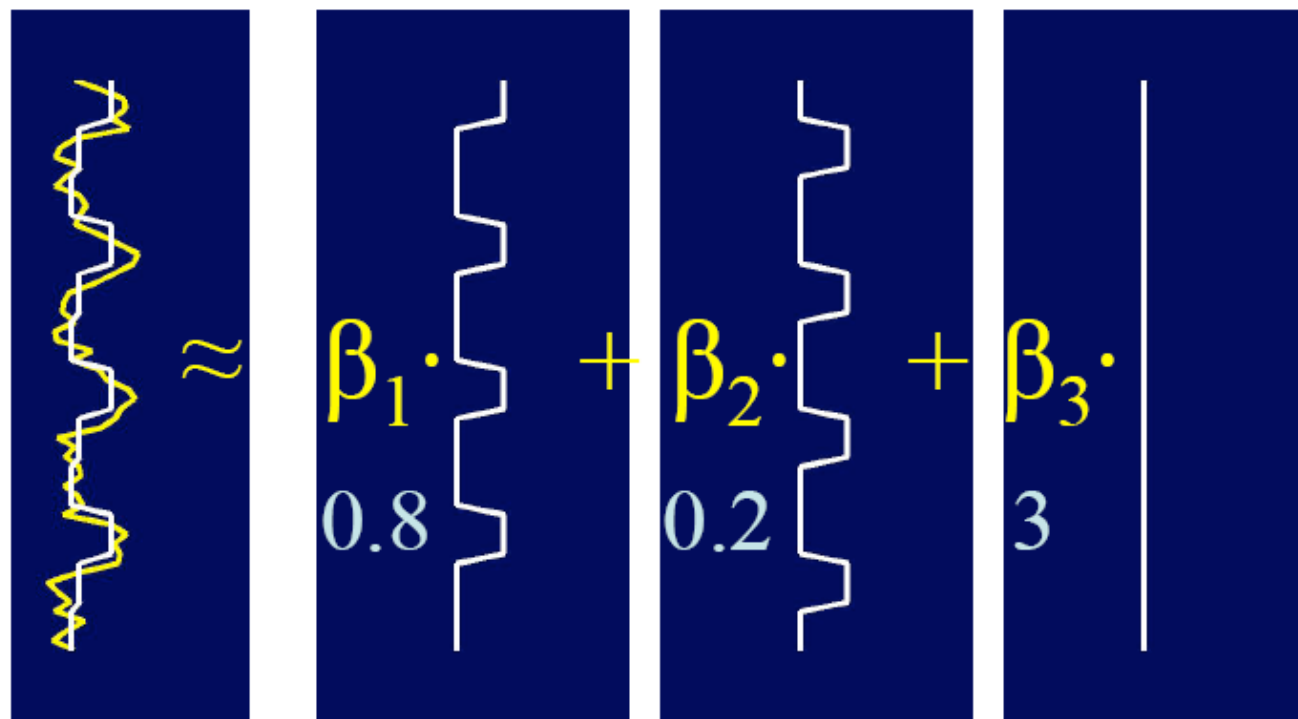
Bad fit



The Estimation:

Finding the "best" parameter values

- The estimation entails finding the parameter values such that the linear combination "best" fits the data.



Good fit



GLM - Estimation

- Estimates of the regression parameters are obtained using Least Squares

$$Y = X\beta + \epsilon$$



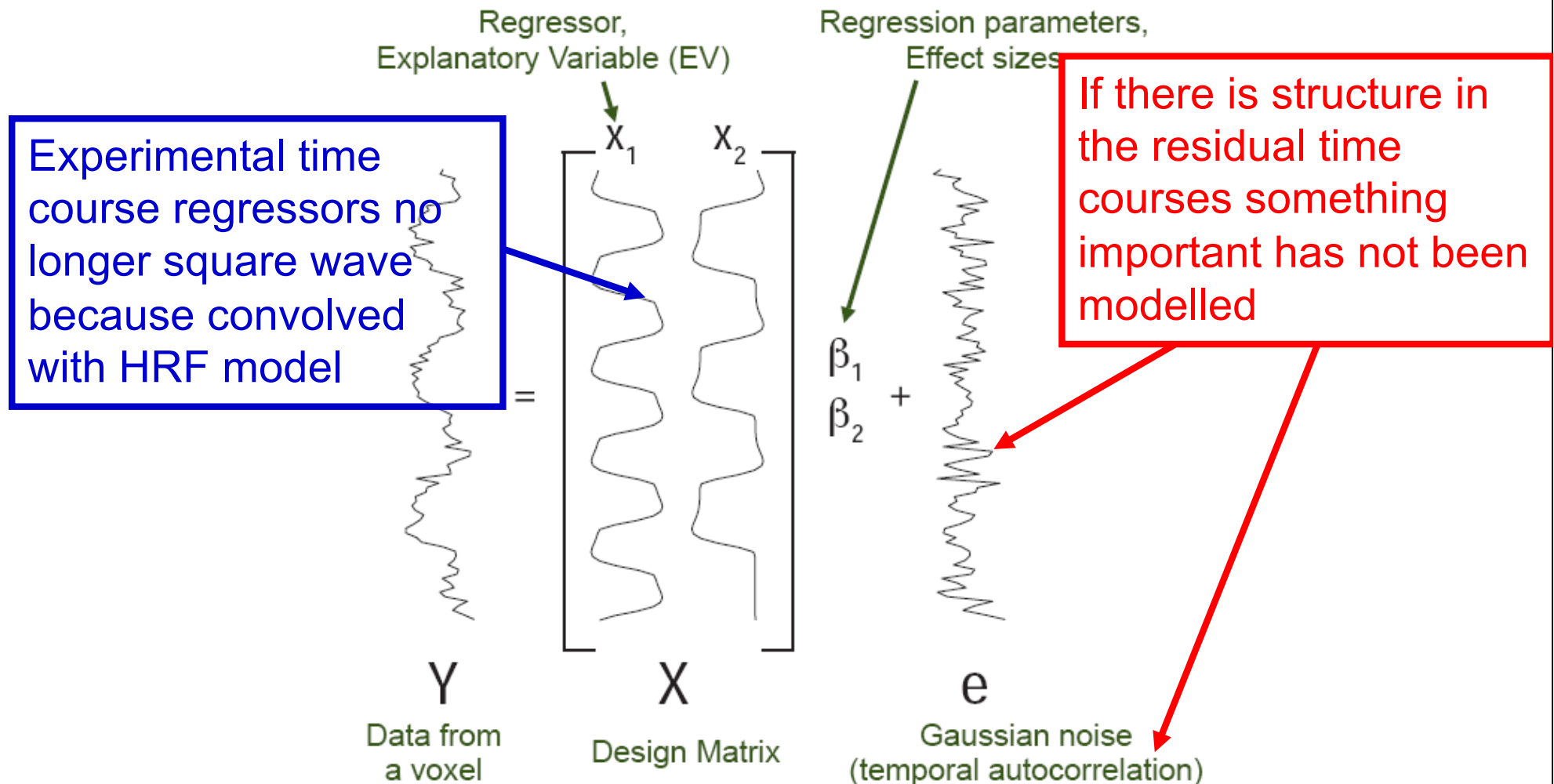
Parameter estimates (PEs) reflect how much of the data is explained by each regressor

The voxel time courses are standardised so that beta weights are comparable between voxels



General Linear Model (GLM)

- This is placed into the General Linear Model (GLM) framework





Autocorrelation: FILM prewhitening

- First, fit the GLM
- Estimate the temporal autocorrelation structure in the residuals
- The estimated structure can be inverted and used as a temporal filter to undo the autocorrelation structure in the data
 - the filter is also applied to the design matrix
- Refit the GLM
 - DOF $n-1$ will now correctly reflect what is really free to vary in the timecourse
- Prewhitening is selected on the stats tab in FEAT
 - it is computationally intensive, but with a modern PC it is manageable and there are almost no circumstances where you would turn this option off



Modelling the Mean

Data

EVs

$$\text{Data} = \begin{bmatrix} \text{EV}_1 & \text{EV}_2 & \vdots \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \mu \end{bmatrix} + \text{Noise}$$

The diagram illustrates a linear model for fMRI data. On the left, a noisy signal is labeled 'Data'. This is followed by an equals sign. To the right of the equals sign is a large square bracket containing two columns of smooth, periodic waveforms labeled 'EVs' (Explanatory Variables). To the right of these waveforms is a vertical line, and to the right of that is a column vector containing the parameters β_1 , β_2 , and μ . This is followed by a plus sign and another noisy signal, representing the noise term in the model.



Removing the Mean

Demeaned Data

Demeaned EVs

$$\text{Demeaned Data} = \begin{bmatrix} \text{Demeaned EV}_1 & \text{Demeaned EV}_2 \end{bmatrix} \begin{matrix} \beta_1 \\ \beta_2 \end{matrix} + \text{Noise}$$

In FEAT the data and EVs are demeaned before the GLM is fit
(at first level, but not for higher levels)



Inferring Activity. Is β Non-zero?

- If β is non-zero then voxel is "active"
- We have estimated β 's from the best fit to the data



Inferring Activity. Is β Non-zero?

- If β is non-zero then voxel is "active"
- We have estimated β 's from the best fit to the data
- BUT the data is noisy - so can we trust the estimated β 's??



Inferring Activity. Is β Non-zero?

- If β is non-zero then voxel is "active"
- We have estimated β 's from the best fit to the data
- BUT the data is noisy - so can we trust the estimated β 's??

t-statistic $t = \frac{\beta}{std(\beta)}$

- Higher β = higher t
Higher $std(\beta)$ = lower t

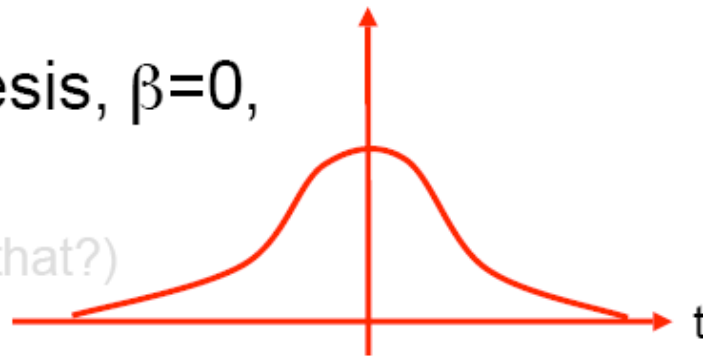


Null Hypothesis Testing

t-statistic: $t = \frac{\beta}{std(\beta)}$

Under null hypothesis, $\beta=0$,
t is t-distributed

(what are the chances of that?)

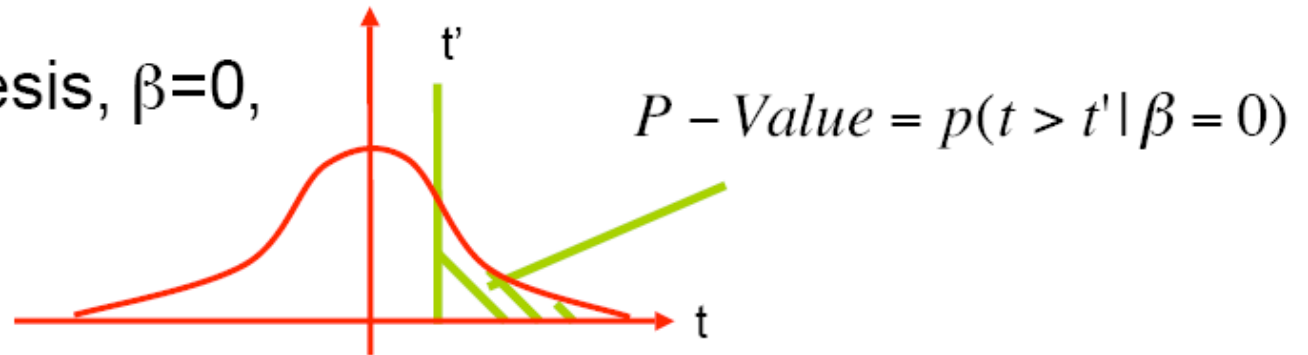




Null Hypothesis Testing

t-statistic: $t = \frac{\beta}{std(\beta)}$

Under null hypothesis, $\beta=0$,
t is t-distributed

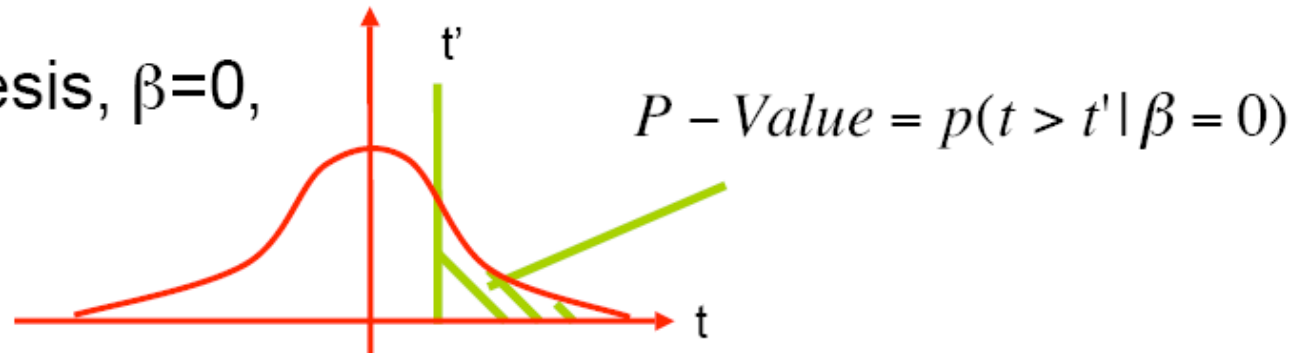




Null Hypothesis Testing

t-statistic: $t = \frac{\beta}{std(\beta)}$

Under null hypothesis, $\beta=0$,
 t is t-distributed



Small P-Value = null hypothesis unlikely
If P-Value < P-threshold then voxel is “active”
P-threshold corresponds to False Positive Rate (FPR)



t-contrasts

- A contrast of parameter estimates (COPE) is a linear combination of PEs:

$$[1 \ 0 \ 0]: \text{ COPE} = 1 \times \hat{\beta}_1 + 0 \times \hat{\beta}_2 + 0 \times \hat{\beta}_3$$

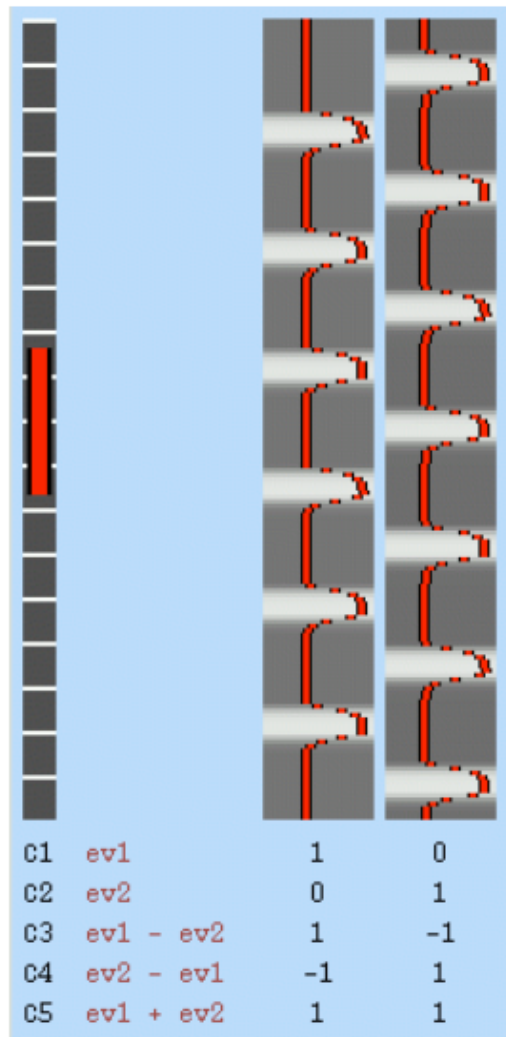
$$[1 \ -1 \ 0]: \text{ COPE} = 1 \times \hat{\beta}_1 + -1 \times \hat{\beta}_2 + 0 \times \hat{\beta}_3$$

- Test null hypothesis that COPE=0

t-statistic: $t = \frac{\text{COPE}}{\text{std}(\text{COPE})}$



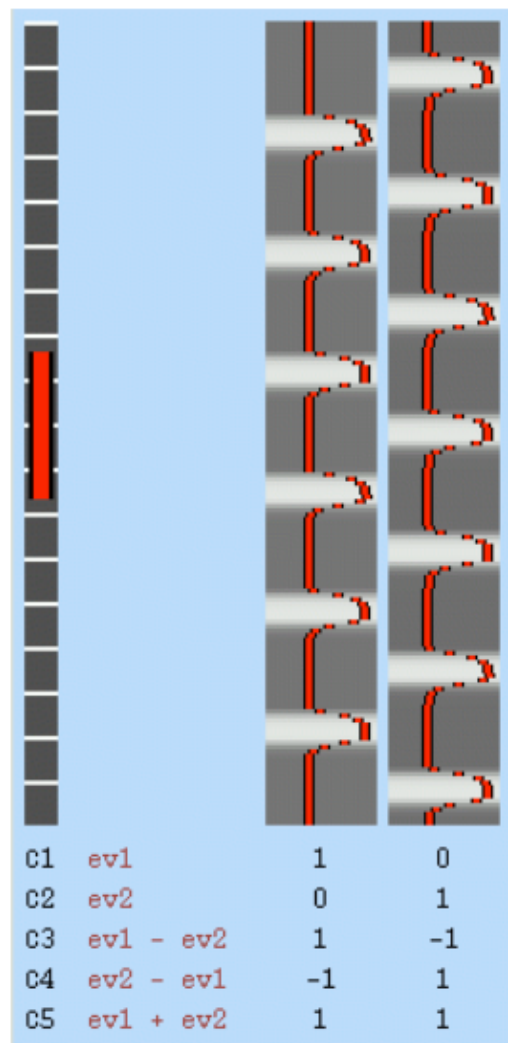
t-contrasts



- $[1 \ 0]$: EV1 only (i.e. stimulus 1 - rest)
- $[0 \ 1]$: EV2 only



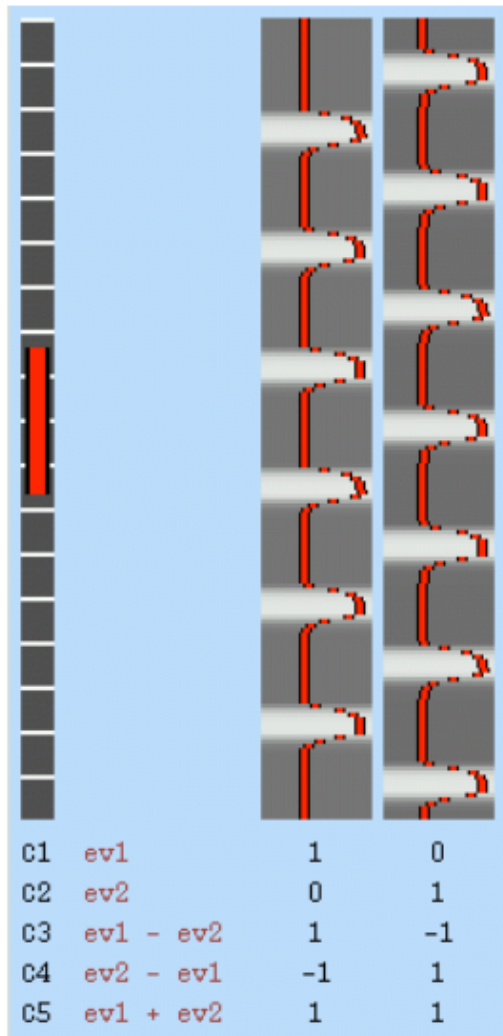
t-contrasts



- $[1 \ 0]$: EV1 only (i.e. stimulus 1 - rest)
- $[0 \ 1]$: EV2 only
- $[1 \ -1]$: EV1-EV2 ("where does EV1 fit better than EV2?")
- $[-1 \ 1]$: EV2-EV1 (tests are directional)



t-contrasts



- $[1 \ 0]$: EV1 only (i.e. stimulus 1 - rest)
- $[0 \ 1]$: EV2 only
- $[1 \ -1]$: EV1-EV2 ("where does EV1 fit better than EV2?")
- $[-1 \ 1]$: EV2-EV1 (tests are directional)
- $[1 \ 1]$: EV1+EV2 (mean activation)



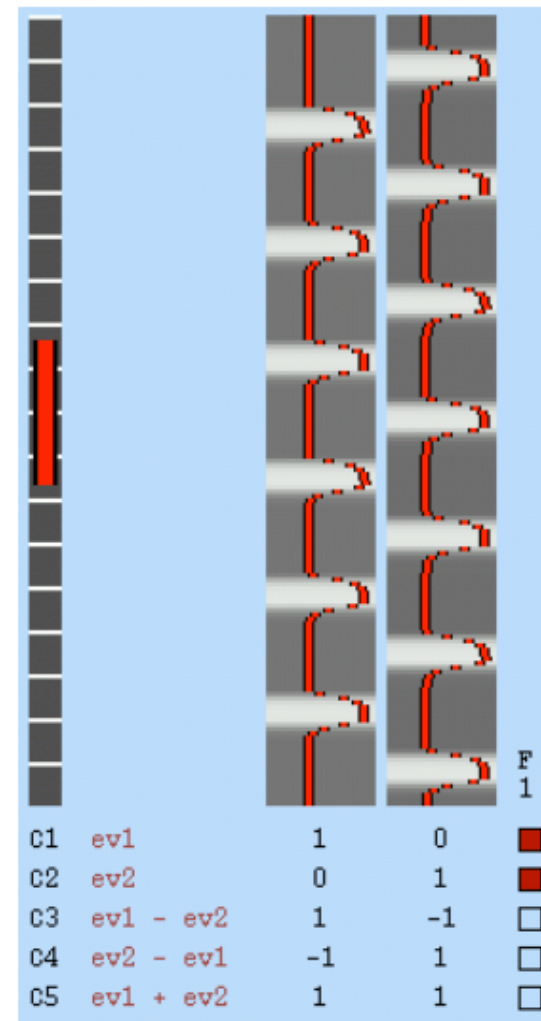
f-contrasts

Two conditions: A,B

- Is any condition significant?
- Set of COPEs form an *f*-contrast
- Tests null hypothesis "do ALL of the COPEs in the *f*-contrast equal zero?"

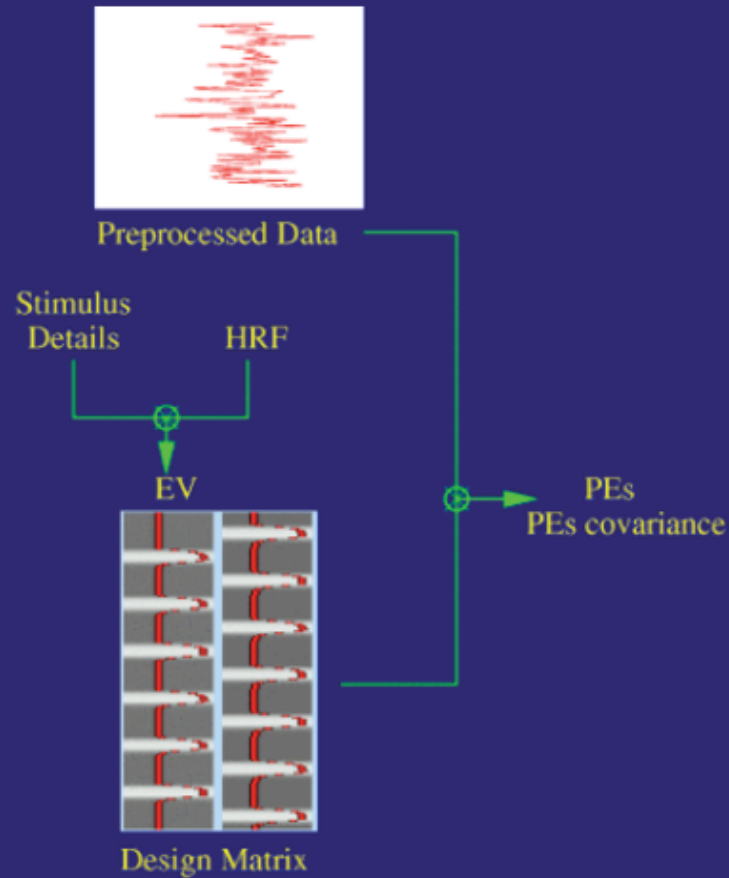
$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

f-contrast



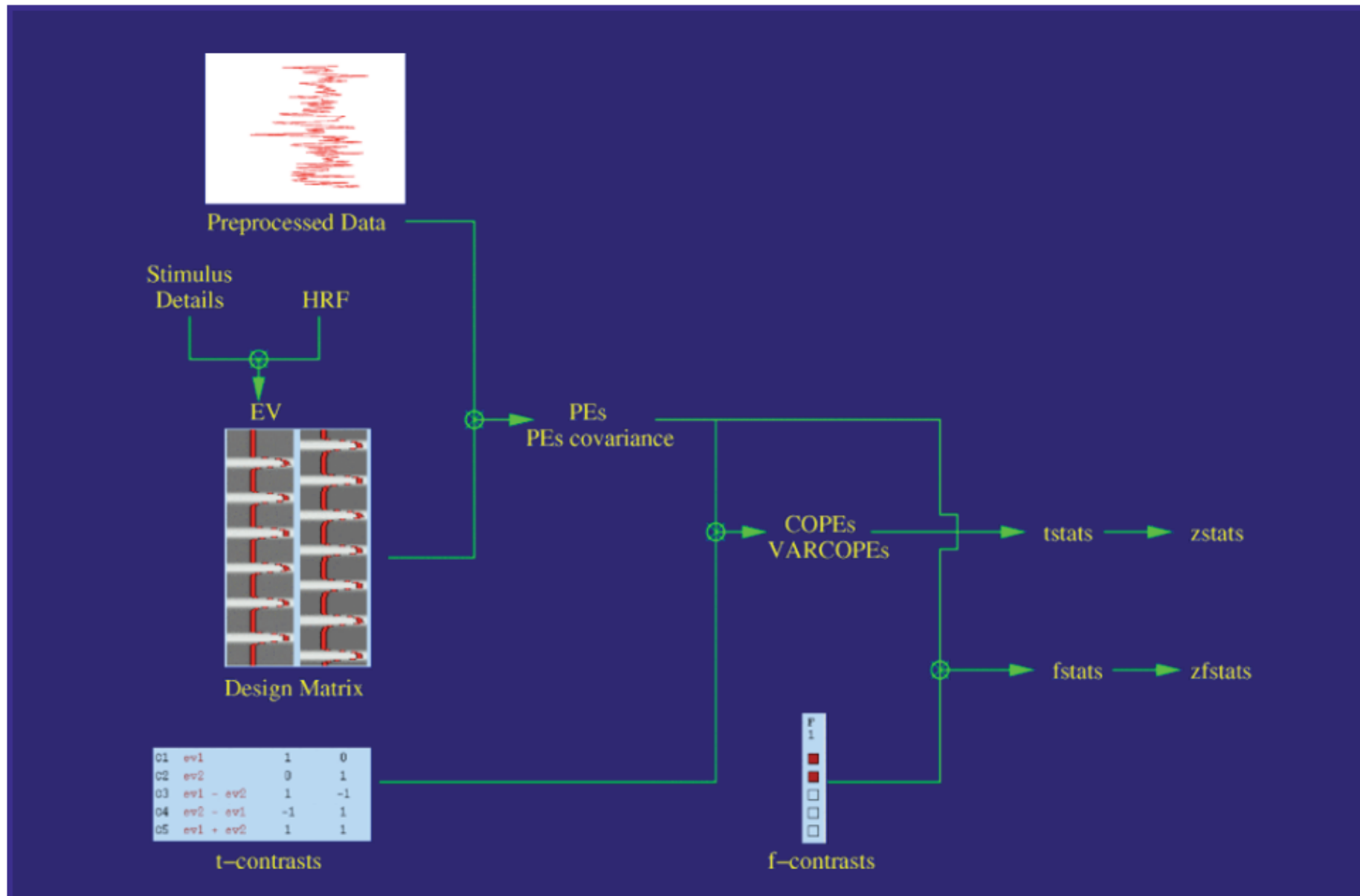


FEAT Schematic



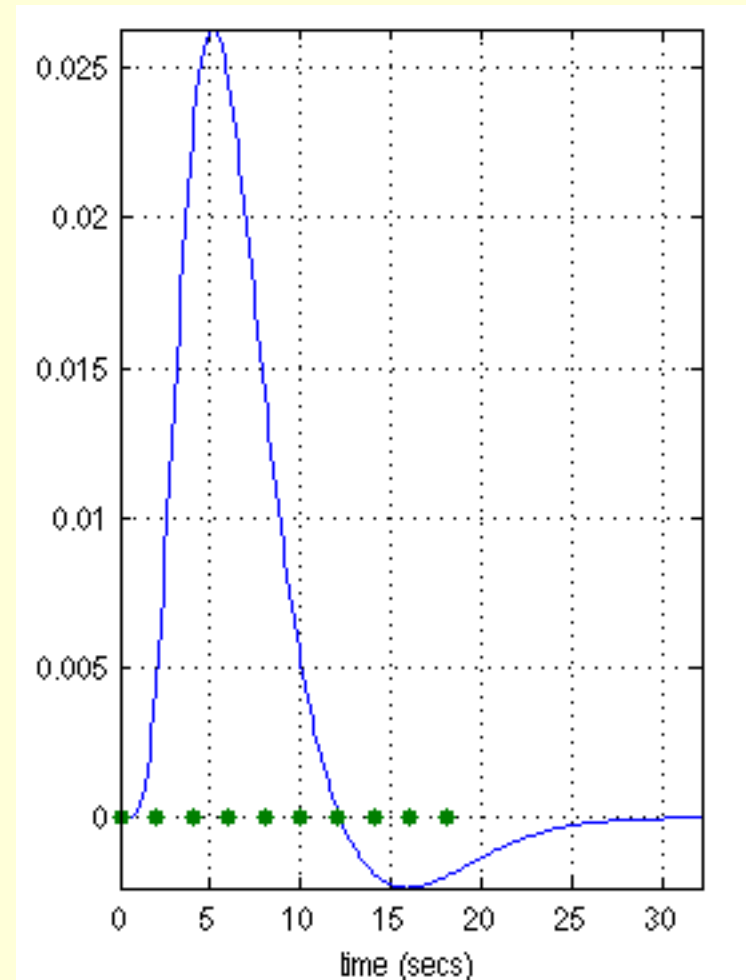


FEAT Schematic



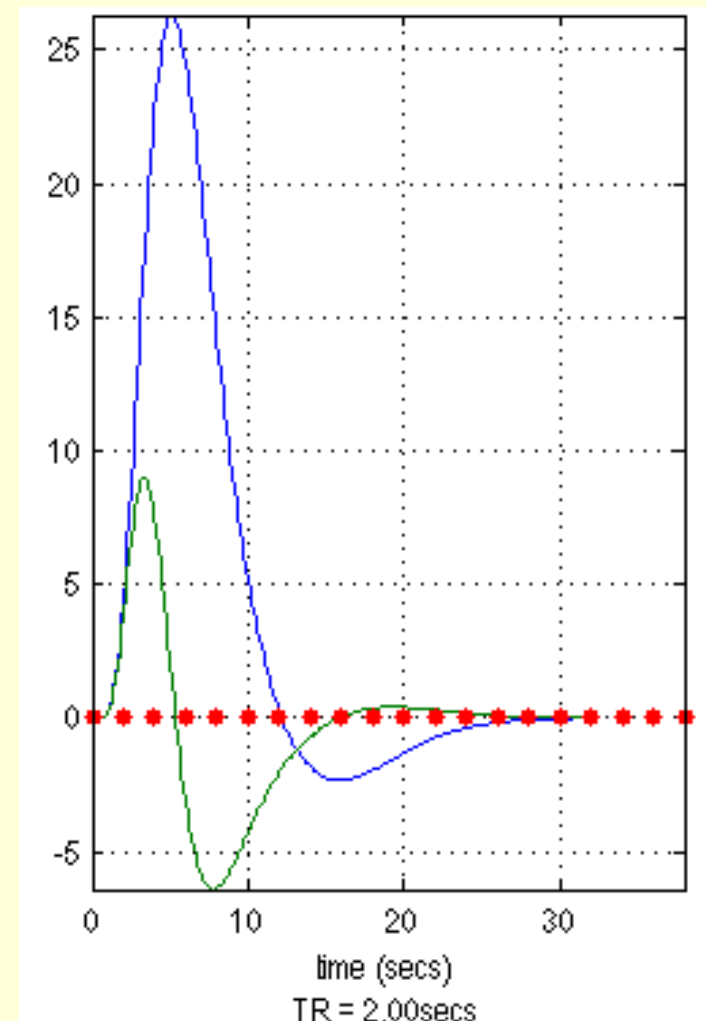
Temporal derivatives

- In the FEAT practical you will add temporal derivatives of the HRF convolved experimental time courses to the design matrix
 - what is the purpose of this?
- Each experimental time course is convolved with a model of the HRF
 - this is to build the delay and blurring of the blood flow response relative to the neural response into the model
 - but the delay varies between brain areas and between people



Temporal Derivatives

- The green line is the first temporal derivative of the blue line
 - it's rate of change
 - the positive max of the derivative is earlier than the normal HRF peak
 - the negative max of the derivative is later than the normal HRF peak
- If fitting the model results in a positive beta weight on a derivative this implies that the HRF peak is earlier in that voxel
- A negative beta weight for the derivative implies a later peak than “typical”



Temporal derivatives

- The basic HRF shape (blue on the previous slide) has some physiological underpinning (in visual cortex...)
- But the use of the derivative to model faster / slower responses is just a mathematical convenience
- The second temporal derivative (dispersion in time) can be used to model haemodynamic responses that are “thinner” or “fatter” in time than the basic shape
- The three functions together are sometimes called the “informed basis set” by SPM users
 - the blue line is referred to as “canonical”, but in fact it is only canonical for primary visual cortex
- The informed basis set can only model slight departures from the canonical response shape
- If you are interested in the prefrontal cortex of the elderly you’ll need to use a more flexible basis set to model the temporal dynamics of the response
 - or use a block design where timing issues are less severe



Multiple Comparison Problem

- We could carry out a null hypothesis test for each voxel using a chosen false positive rate (FPR)
- However, if we used $FPR=0.05$ with 20,000 brain voxels then we would get 1000 FPs
- **Not good** - we would really like the FPR to correspond to the probability of getting one FP in the entire brain



Bonferroni Correction

- Bonferroni: divide uncorrected p-threshold by number of voxels before thresholding

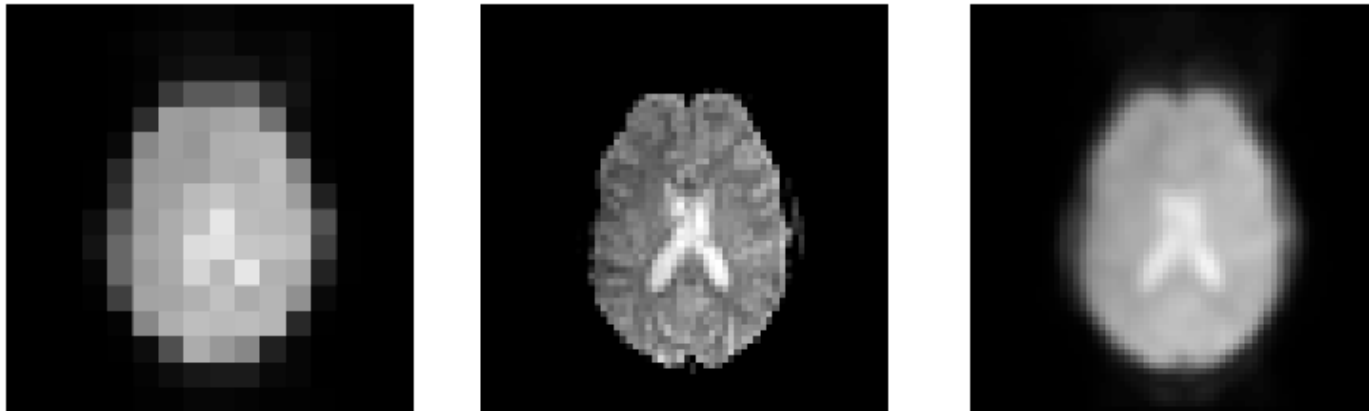
- E.g. 20,000 brain voxels:

an uncorrected p-threshold of 0.05 becomes

$$0.05 / 20,000 = 0.0000025$$



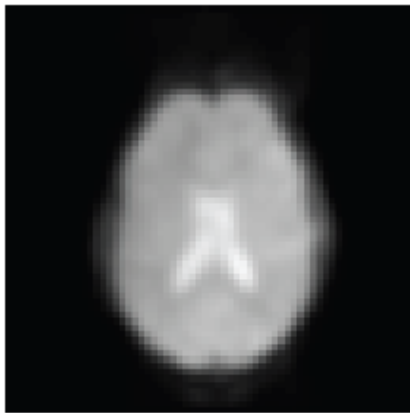
Random Field Theory (RFT)



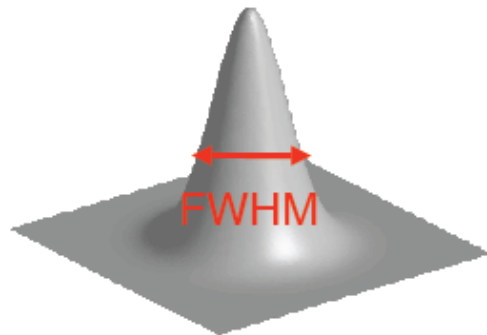
- But: voxels are not independent, due to inherent spatial smoothness
- Spatial filtering helps to condition smoothness
- Random Field Theory can then be used to do (in general) less conservative null hypothesis tests via RESEL based correction



What is a RESEL?



- RESEL stands for **RES**olution **E**lement
- It is a bunch of voxels equal to the same size as the **FWHM** of the smoothness of the image



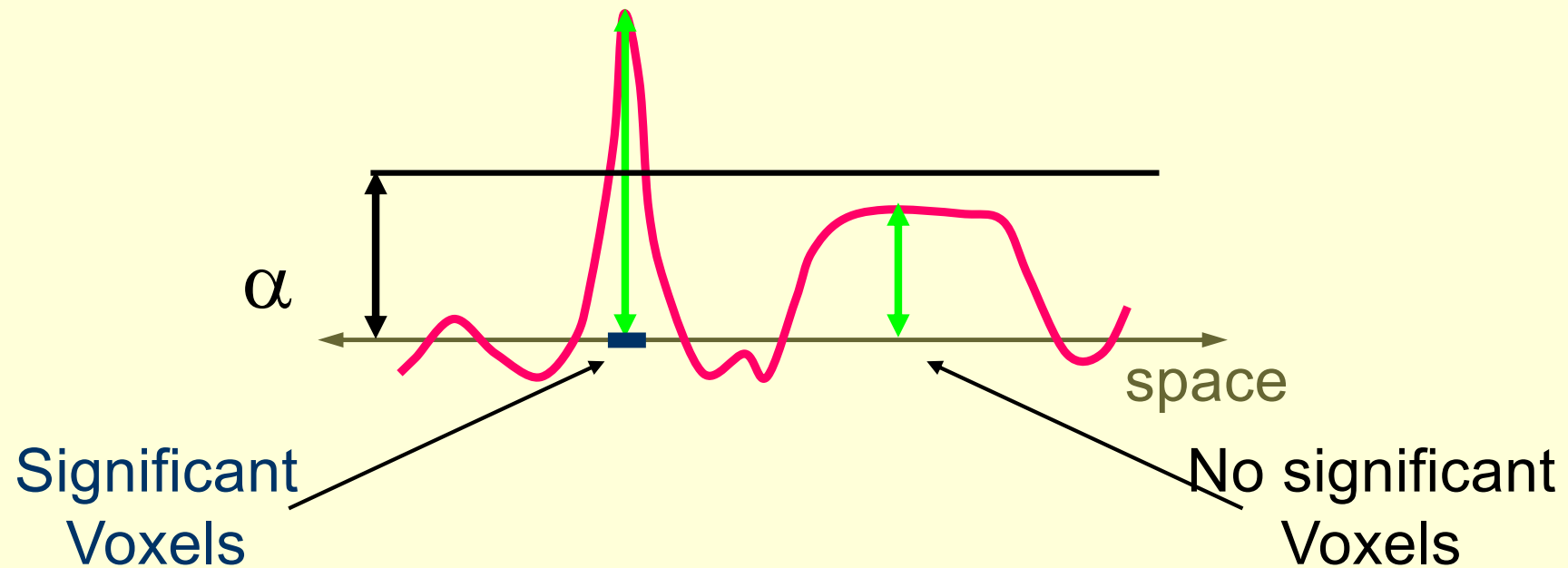
- Number of **RESELS** is similar to, but NOT equal to, the number of independent observations in an image

See <http://www.mrc-cbu.cam.ac.uk/Imaging/Common/randomfields.shtml> for a nice tutorial

Cluster size based thresholding

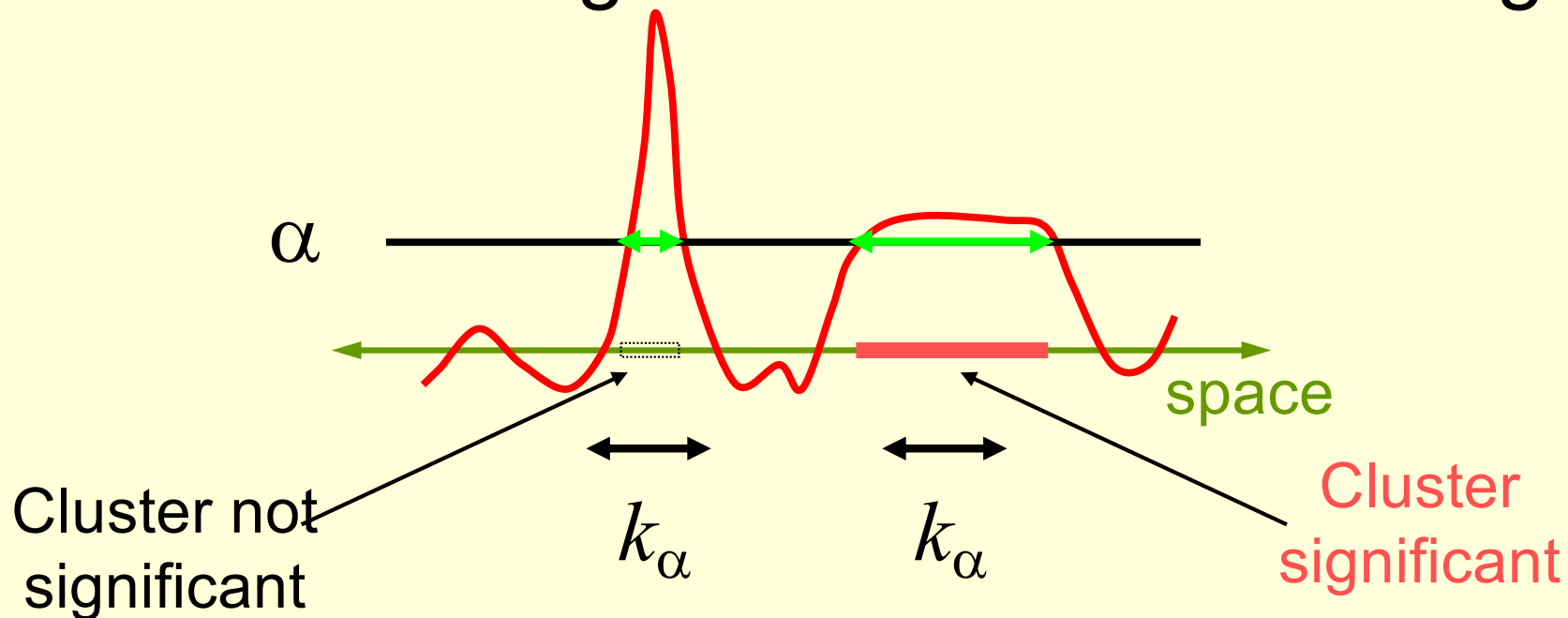
- Intuitively, if a voxel with a Z statistic of 1.96 for a particular COPE is surrounded by other voxels with very low Z values this looks suspicious
 - unless you are looking for a very small brain area
- Consider a voxel with a Z statistic of 1.96 is surrounded by many other voxels with similar Z values, forming a large blob
- Intuitively, for such a voxel the Z of 1.96 ($p = 0.05$) is an overestimate of the probability of the model fit to this voxel being a result of random, stimulus unrelated, fluctuation in the time course
- The p value we want to calculate is the probability of obtaining one or more clusters of this size or larger under a suitable null hypothesis
 - “one or more” gives us control over the multiple comparisons problem by setting the family wise error rate
 - p value will be low for big clusters
 - p value will be high for small clusters

Comparison of voxel (“height based”) thresholding and cluster thresholding



α is the height threshold, e.g. 0.001
applied voxelwise (will be $Z = \text{about } 3$)

Comparison of voxel (“height based”) thresholding and cluster thresholding

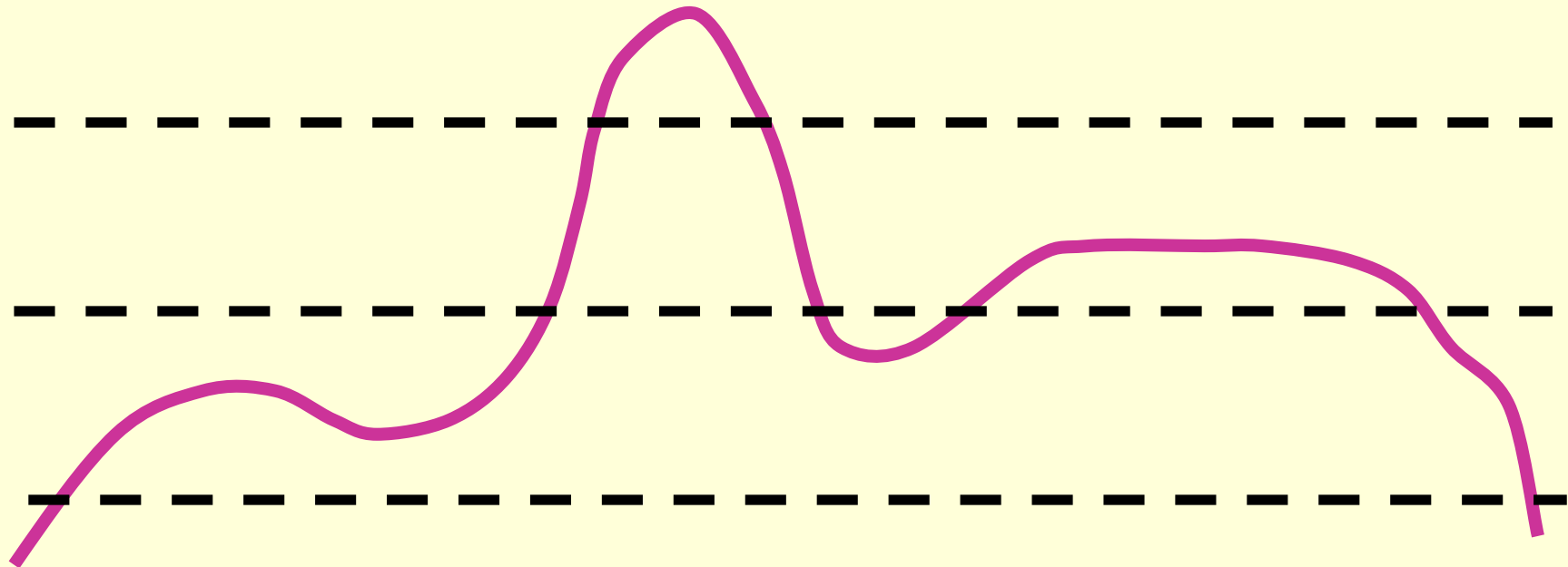


K_α is the probability of the image containing 1 or more blobs with k or more voxels (and you can control is at 0.05)

The cluster size, in voxels, that corresponds to a particular value of K_α depends upon the initial value of height threshold α used to define the number of clusters in the image and their size

It is usual to set height α quite low when using cluster level thresholding, but this arbitrary choice will influence the outcome

Dependency of number of clusters on choice of height threshold



The number and size of clusters also depends upon the amount of smoothing that took place in preprocessing



contract

3

1

1

1

- Nyquist frequency is important to know about
 - Half the sampling rate (e.g. TR 2 sec is 0.5 Hz, so Nyquist is 0.25 hz, or 4 seconds)
 - No signal higher frequency than Nyquist can be present in the data (*important for experimental design*)
 - But such signal could appear as an aliasing artefact at a lower frequency

