fMRI Basics: Single subject analysis using the general linear model

With acknowledgements to Matthew Brett, Rik Henson, and the authors of Human Brain Function (2nd ed)

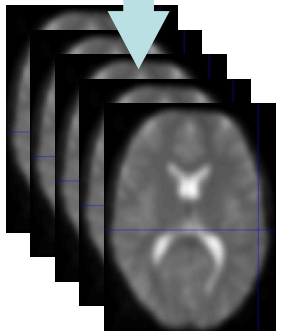
Overview

Synchronisation pulses

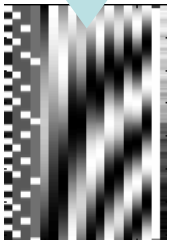
Stimulus presentation

Pre-processing

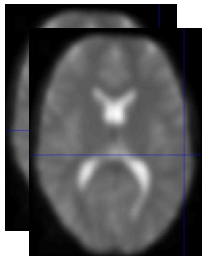
Event timings etc



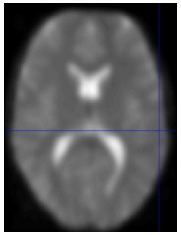
=



x



+



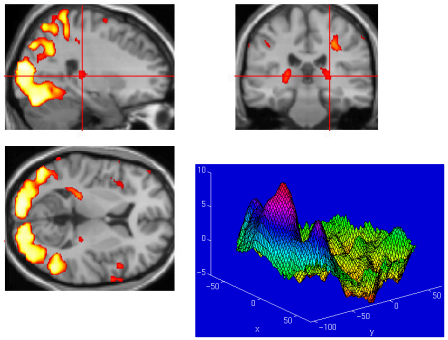
Observed Data

Design Matrix

Parameter Estimates

Error

General Linear Model



Statistical Inference

General Linear Model

GLM models observed data (dependent variable) as a linear combination of predictor variables (independent variables, covariates etc)

For i observations modeled using j predictor variables:

$$y_i = \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3} + \ldots x_{ij} \beta_j + \varepsilon_i$$

y_i	Observation i
X_{ij}	Value i for predictor variable j
β_j	Parameter estimate for predictor variable j
ε_i	Error for observation i

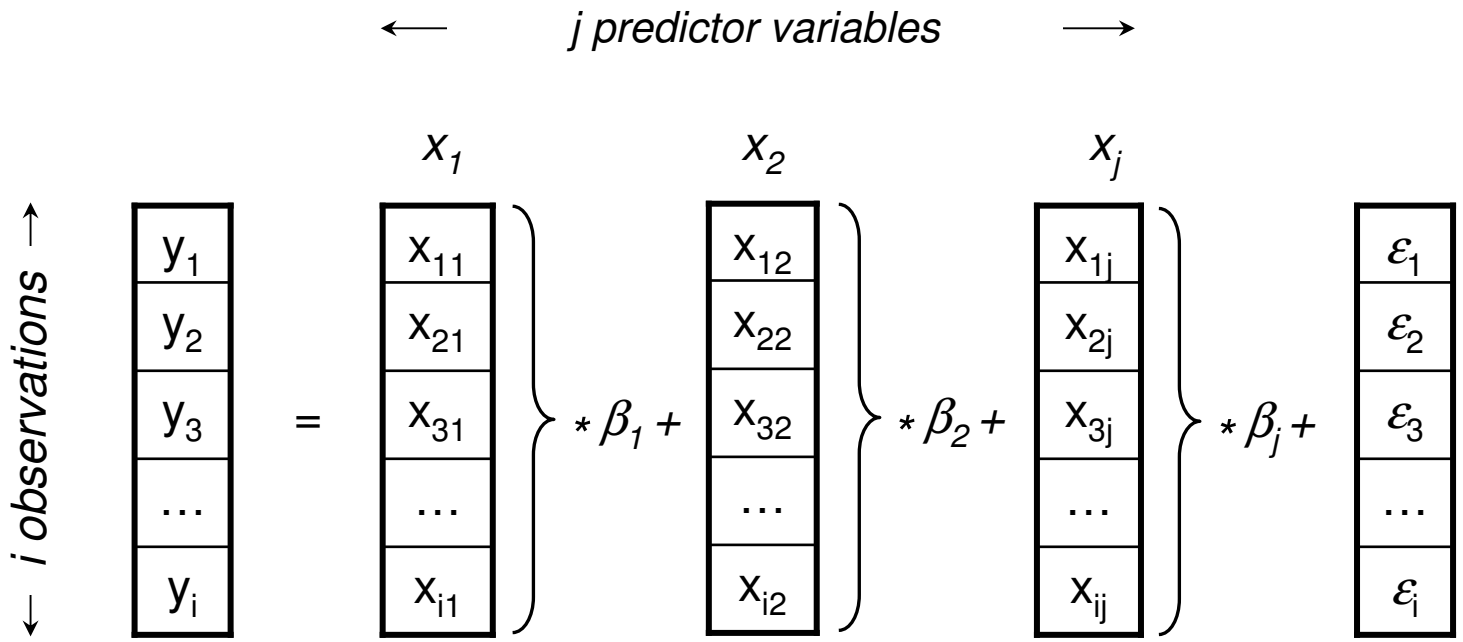
- Linear refers to the additive relationship between the different weighted predictor variables. These predictor variables themselves can be anything, including non-linear functions of more basic predictors.

General Linear Model

- Many statistical techniques are special cases of the GLM – e.g. ANOVA, ANCOVA, t-test, multiple regression (and also MANOVA, LDA and canonical correlation in multivariate analyses)
    - ANOVA asks whether different experimental conditions (X_1, X_2, etc) are associated with different scores
    - Multiple regression asks whether scores are related to predictor variables (X_1, X_2, etc)
- Both are essentially doing the same thing – asking whether there is a relationship between a dependent variable and one or more independent variables
- Parameter estimates also called Beta values (= treatment effects in ANOVA, or the slope of the regression line)
- Measure of the strength of relationship between a predictor and the observed data (taking into account all the other predictors in the model)

General Linear Model

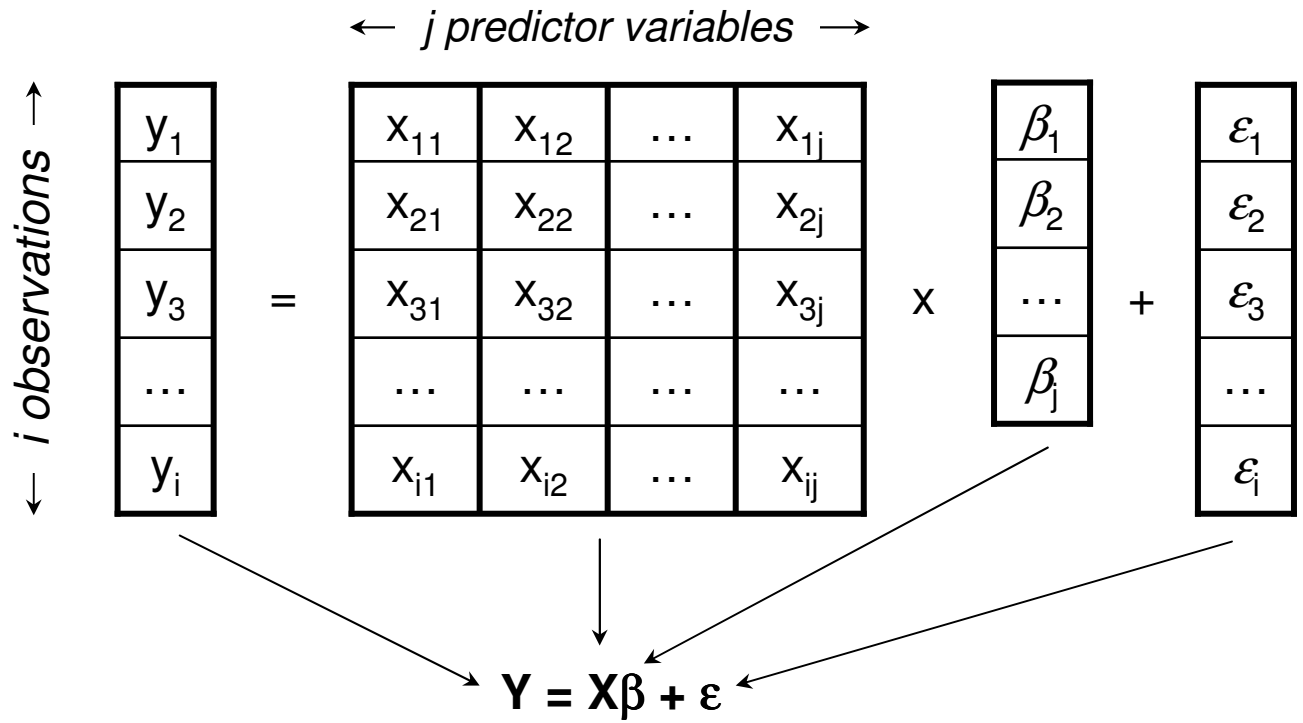
A full experiment:



This can be interpreted as a system of simultaneous equations that we wish to solve for the unknowns $\beta_{1...j}$

General Linear Model

In matrix form:



X = “Design Matrix” = the statistical model of the data

β = vector of j parameter estimates

Design Matrix

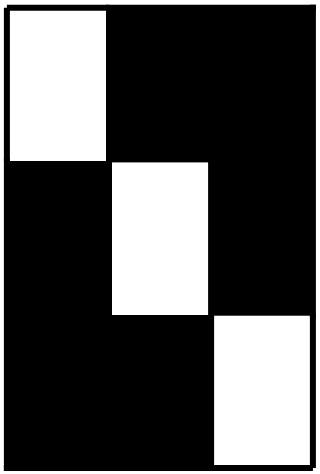
- A numerical description of known sources of variance in the experiment - information about stimulus conditions, covariates etc.
- Can also be represented graphically.
- E.g. 1 way ANOVA with 3 levels:

Factor Level
1
1
2
2
3
3

Group membership coded using dummy variables

$(x_1$	x_2	$x_3)$
1	0	0
1	0	0
0	1	0
0	1	0
0	0	1
0	0	1

Graphical representation



Design Matrix

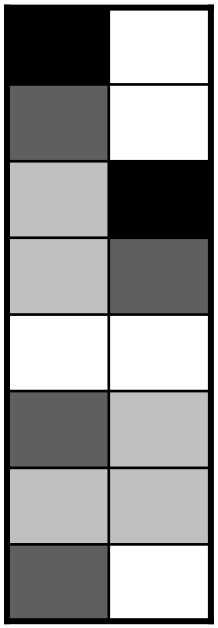
- E.g. multiple regression with 2 continuous predictor variables:

Predictor variables

$(x_1$	$x_2)$
1	4
2	4
3	1
3	2
4	4
2	3
3	3
2	4

Graphical representation

$(x_1$ $x_2)$

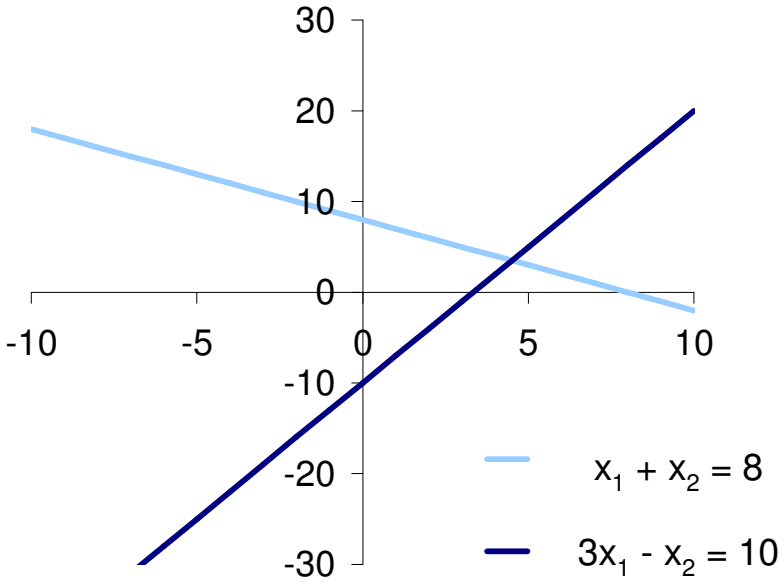


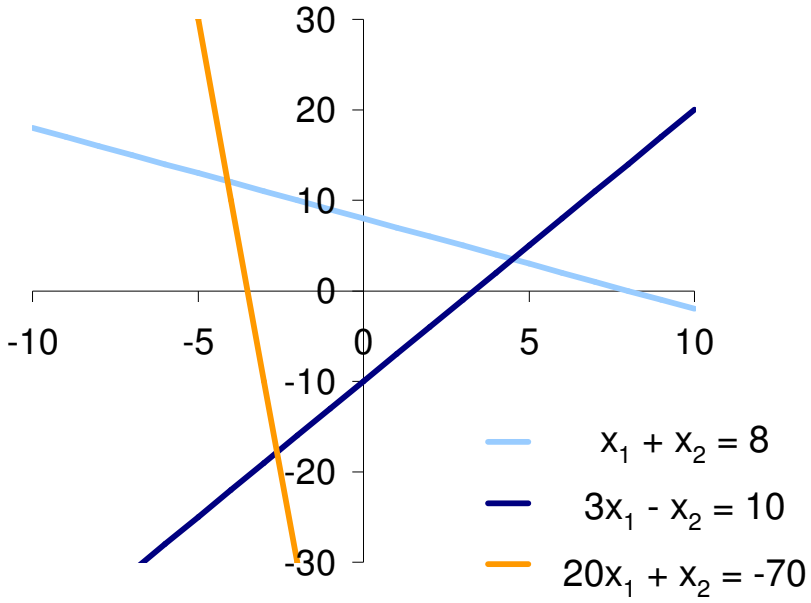
Design Matrix

- Extremely flexible, accommodates many types of predictor variables:
  - ANCOVA is simply an ANOVA-like design matrix with additional columns for covariates
  - Can easily model interactions between ANOVA-esqe categorical predictors and covariates
  - Can model functions of predictor variables – e.g. could include powers of predictor variables to get a polynomial expansion
- Can include both experimentally manipulated variables and nuisance variables of no interest – any known source of variance that you want to model

Estimating the parameters

- How to determine the beta values?
- Generally have more observations than predictors, which means the design is over-determined – there's no unique solution





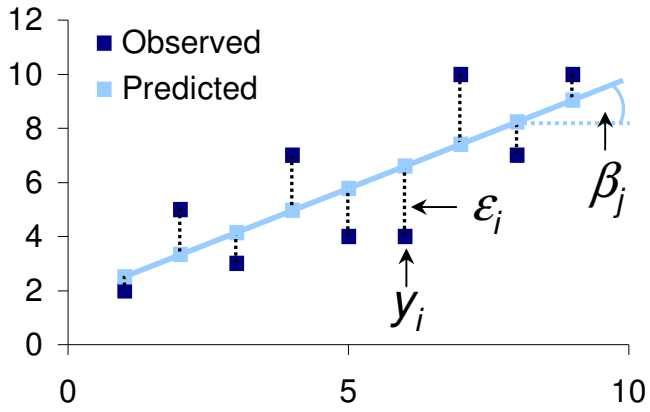
Estimating the parameters

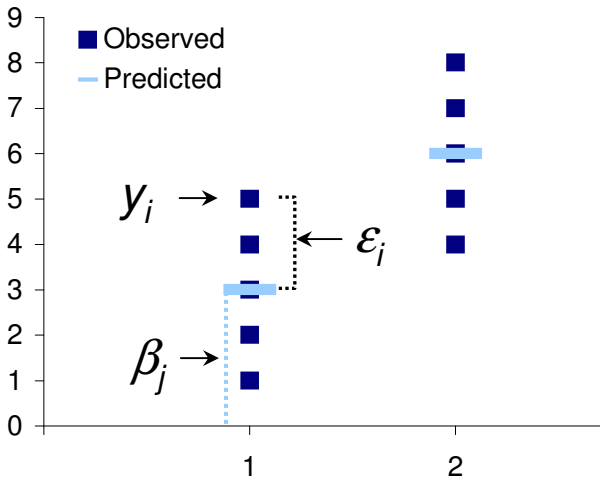
- Want to find the set that are “best” in some way
- The most common definition is to find the beta values that minimise the sum of squared residuals (error terms, unexplained variance) – i.e. the difference between observed and predicted values
- Ordinary Least Squares (OLS)

OLS estimate: $\beta = (X^T X)^{-1} XY$

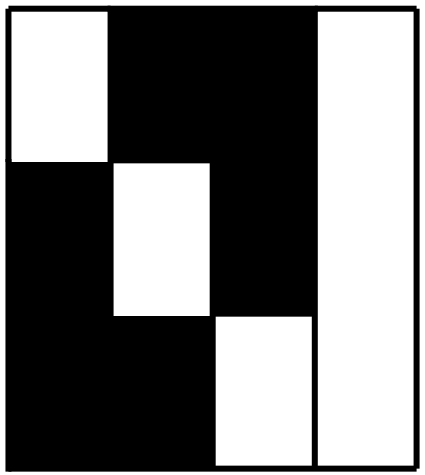
Predicted data: $\hat{Y} = X\beta$

Residuals: $E = Y - \hat{Y}$





Estimating the parameters



x_1 x_2 x_3 μ

$x_1 + x_2 + x_3 = \mu$

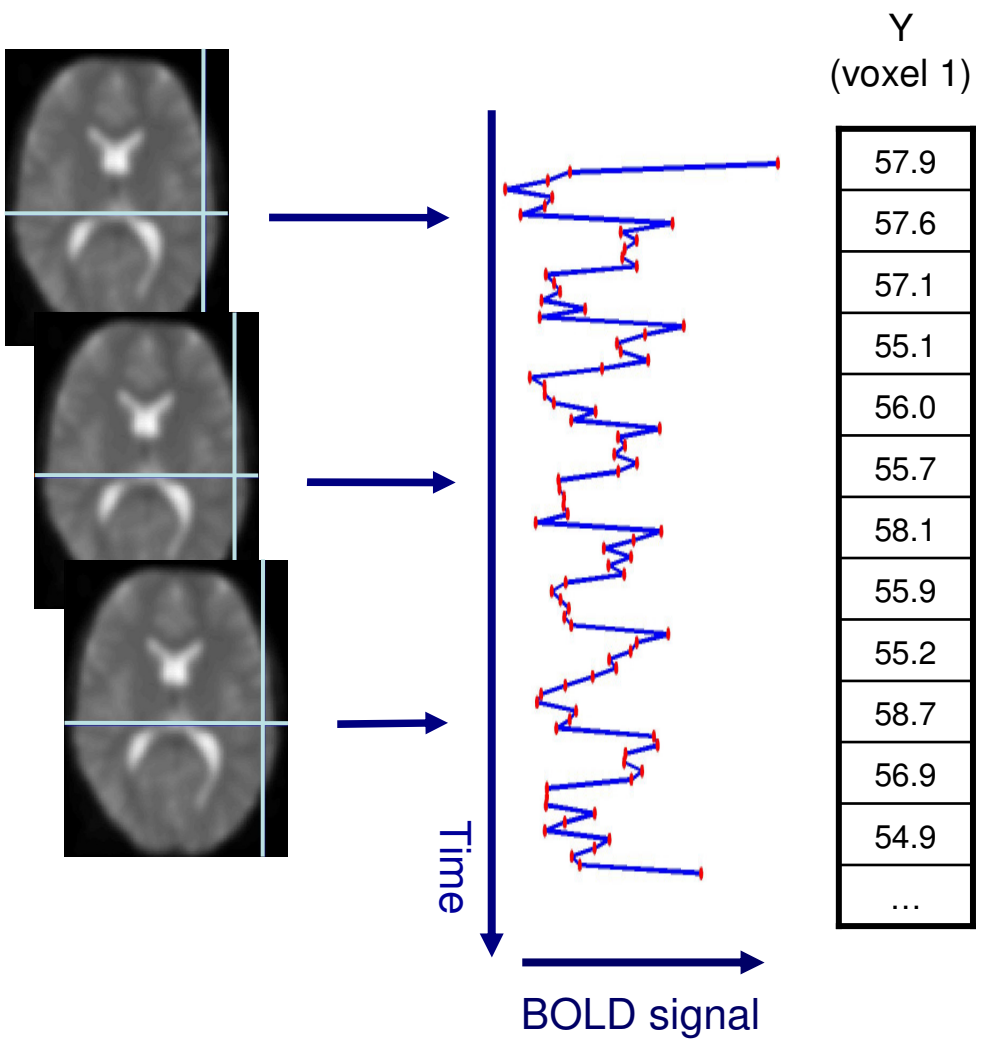
- Main caveat is that the inverse is not defined for rank deficient matrix
- Rank = number of linearly independent columns
- If one column (or set of columns) is a linear combination of another column (or set of columns), the matrix is rank deficient
- Have to solve with additional constraints
- Moore-Penrose pseudoinverse – solves for minimum $\Sigma\beta^2$ (L_2 minimum norm)
- This limits the statistical comparisons that can be performed

- Correlations between design matrix columns produce inefficient estimation

  β values represent partial correlations between predictor and the observed data

- i.e. they represent the unique contribution of each predictor to explaining the observed data

Application to fMRI data



Most common approach is mass univariate

Model estimated for each voxel independently

Y is a vector of values showing BOLD signal strength at a single voxel in successive scans (a voxel time-series)

Output is one beta value for each predictor in the model

Iterate over all voxels to build up a series of beta images, one beta image for each predictor in the model

Application to fMRI data

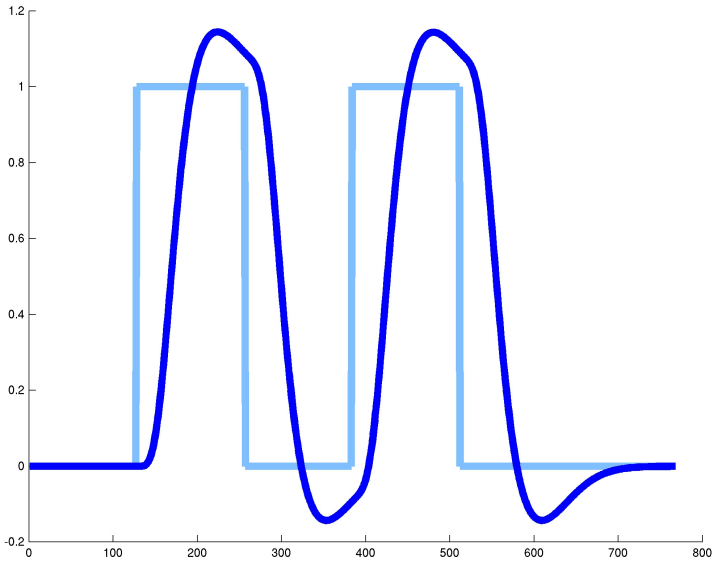
One complication in fMRI is that the BOLD signal responds slowly and is temporally extended

Peak response is 4-6 seconds after stimulus presentation

Simply putting 1's in the design matrix rows corresponding to stimulus presentation can miss considerable amounts of variance in the signal

0
1
1
0
0
1
1
0

Simple "boxcar" stimulus function



Application to fMRI data

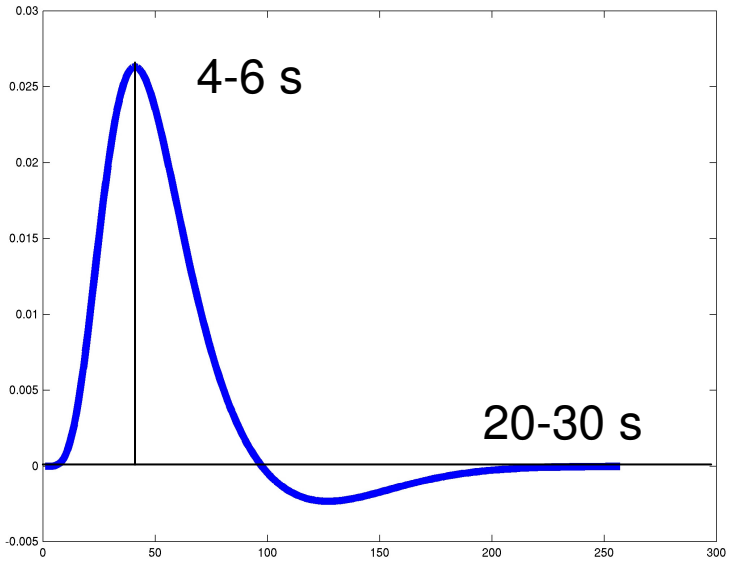
How to create a design matrix with predictor functions that capture this variability?

Fortunately, the BOLD response shows a stereotyped pattern of response (especially in primary sensory areas)

Canonical Haemodynamic Response Function

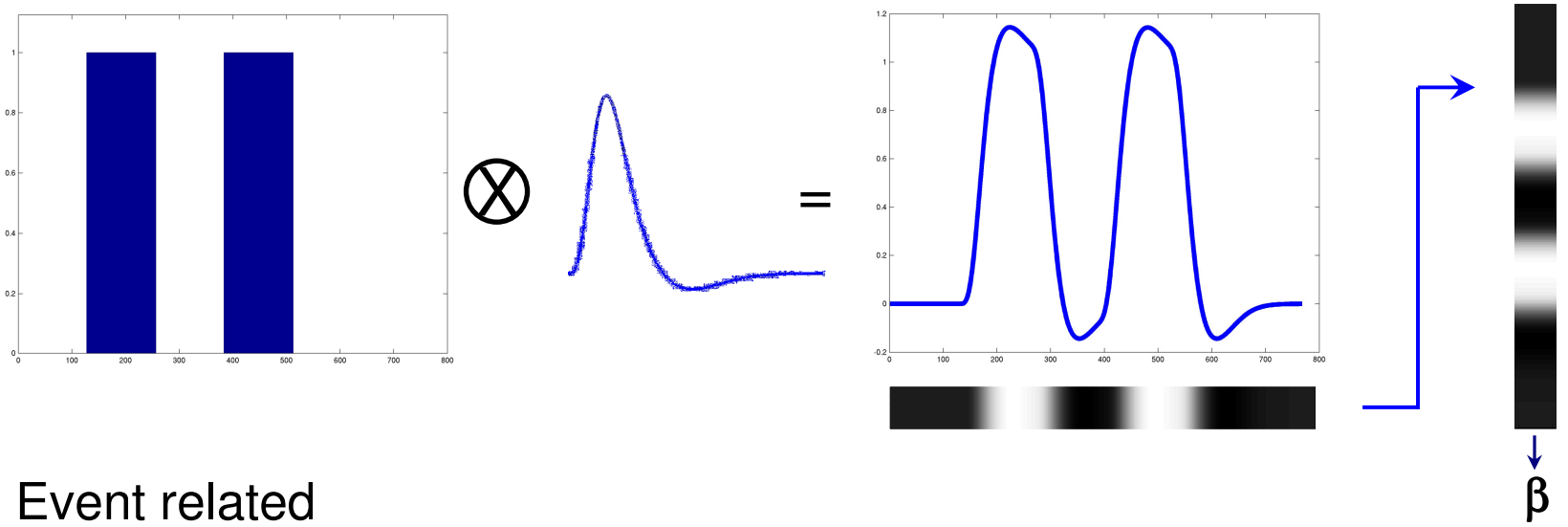
Create a function representing stimulus presentation, then convolve it with the Href function

Href used as a basis function (i.e. a simple function used to describe the response space of a more complex function, in this case the complex time course)

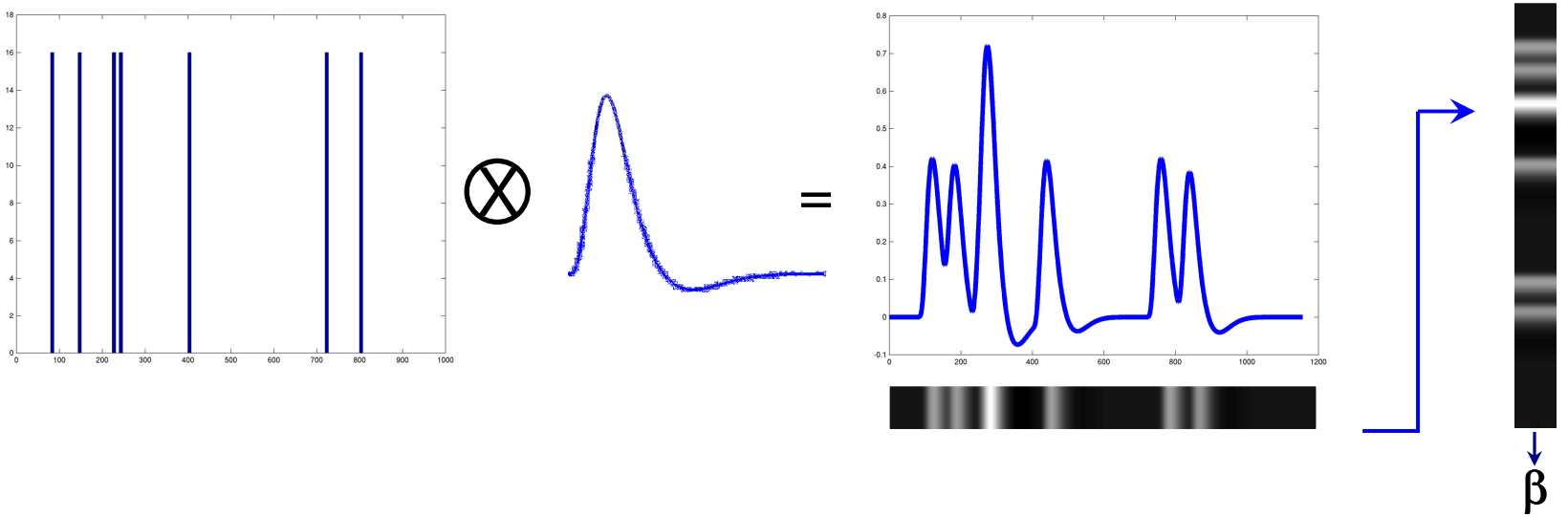


Convolved regressors

Block design (epochs)

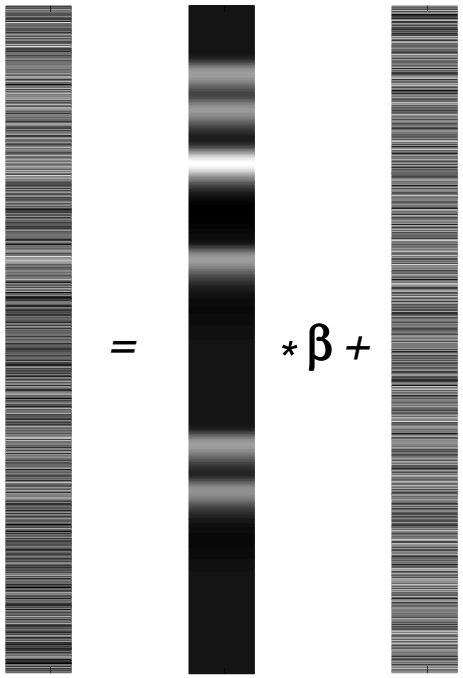


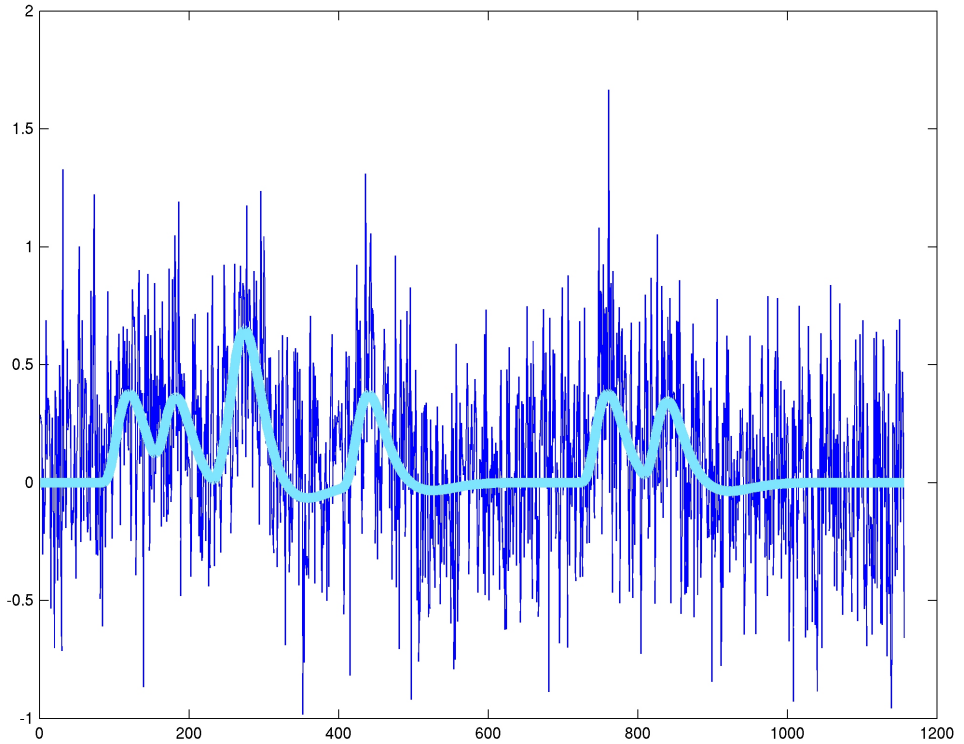
Event related



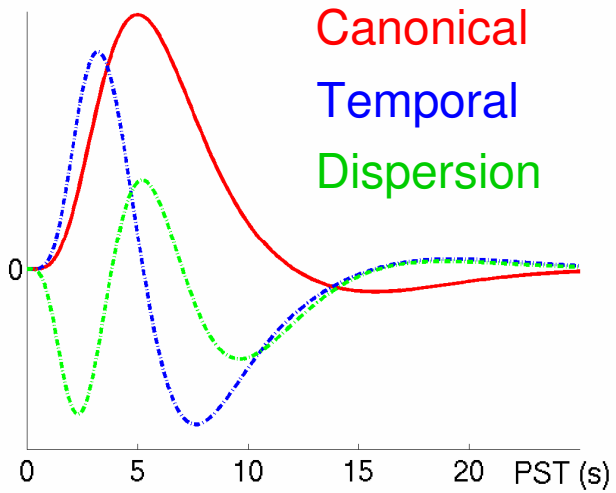
Convolved regressors

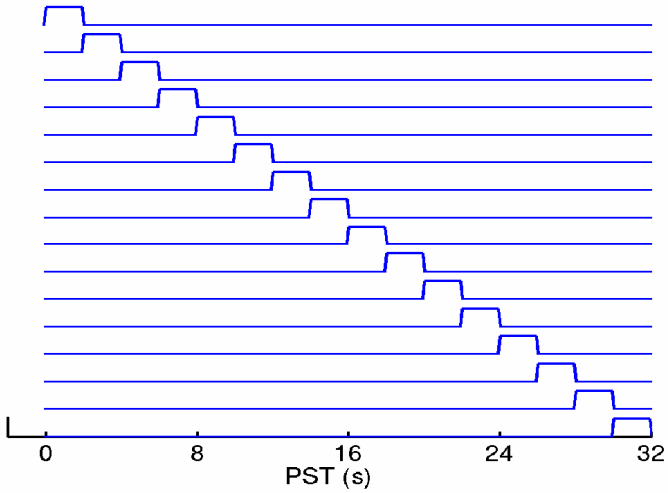
Beta values can be thought of as scaling the regression function to give the best fit to the data





Other basis sets





Canonical Href on its own does a reasonable job, but it's inflexible (only 1df), and won't capture differences in shape or timing of BOLD signal

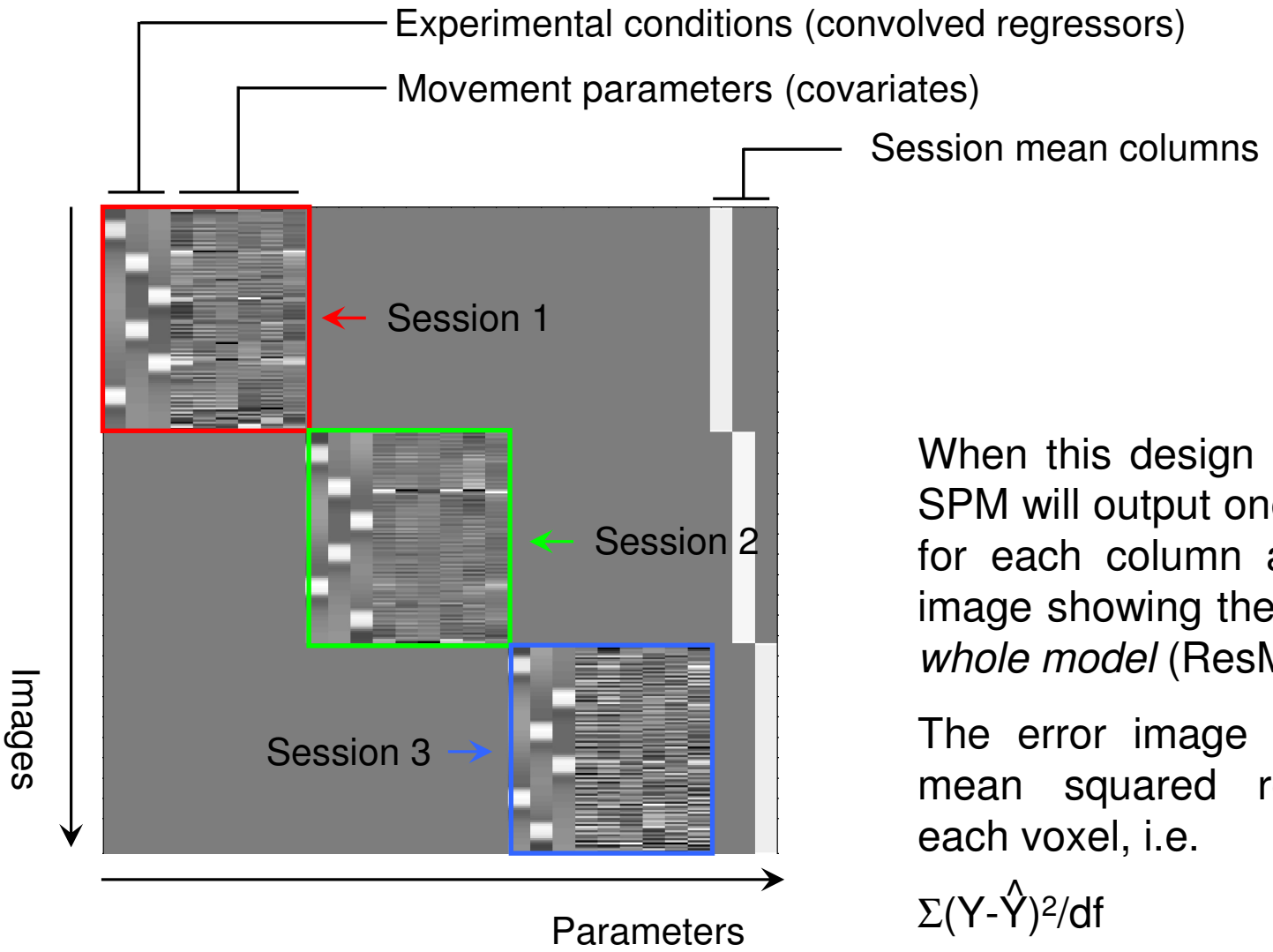
In practice, can use any basis function you want, including sets of multiple basis functions

Adding temporal and dispersion derivatives to Href allows for flexibility in modelling latency and shape of response

Finite Impulse Response (FIR) – series of mini-boxcars, each modelling a discrete time-bin (similar to selective averaging)

End up with 1 column (and one beta image) per basis function per predictor

The Design Matrix Revisited



When this design is estimated, SPM will output one beta image for each column and a single image showing the error *for the whole model* (ResMS.img)

The error image contains the mean squared residuals for each voxel, i.e.

$\Sigma(Y-\hat{Y})^2/df$

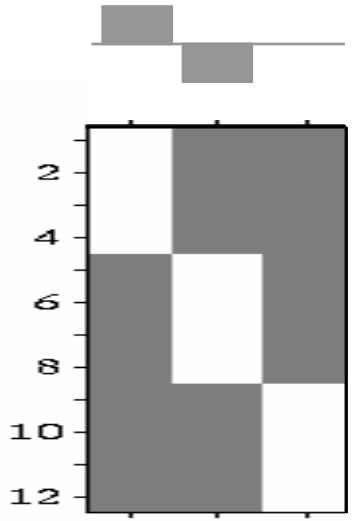
Contrasts

After the model parameters have been estimated, specific hypotheses are tested using t or F statistics.

Calculate these using contrasts - weighted combinations of beta values.

A contrast vector defines how to weight each column of the design matrix

c = | 1 | -1 | 0 |



β = | 4.2 | 0.6 | 3.5 |

E.g. to find areas more active in condition 1 than condition 2, use the contrast [1 -1 0]

If c is a row vector of contrast weights, and β is a row vector of beta values,

Contrast = cβ'

1	-1	0	*	4.2
				0.6
				3.5

= 4.2 – 0.6 = 3.6

Contrasts

Comparisons between conditions can take any form as long as the contrast vector sums to zero:

- [2 -1 -1] = areas more active during condition 1 than conditions 2 and 3
- [-3 -1 1 3] = areas showing a linear increase over 4 conditions

For a 2x2 factorial design with factors A and B, if the design matrix columns correspond to A_1B_1, A_1B_2, A_2B_1,A_2B_2 then:

- [1 1 -1 -1] = main effect of A
- [1 -1 1 -1] = main effect of B
- [1 -1 -1 1] = interaction

Conditions can also be compared to an unmodelled baseline – e.g. to find areas that are active during condition 1, use the contrast [1 0 0]

t-contrasts

T contrasts contain a single row (single degree of freedom)

Directional, so [1 -1 0] shows where c1>c2, but [-1 1 0] shows where c2>c1.

A t-statistic is obtained by dividing the contrast value by the standard error of the contrast:

$$t = c\beta'/se(c\beta')$$

Standard error of $(c\beta')$ can be estimated using $\sqrt{(\sigma^2 * c * (X'*X)^{-1} * c')}$

Where σ^2= mean squared error = $\Sigma(Y-\hat{Y})^2/df$

H_0: $c\beta = 0$

Running a t-contrast creates one image containing the calculated contrast values (con*.img) and one image containing the calculated t-statistics (spmT*.img)

F-contrasts

F contrasts can contain more than one row

Non-directional – test for any differences between conditions (+ve or –ve)

Additional rows can be thought of as adding a logical OR to the contrast e.g:

[1 -1 0;

0 1 -1]

= Where is condition 1 greater than condition 2 OR condition 2 greater than condition 3

[1 0 0;
0 1 0]

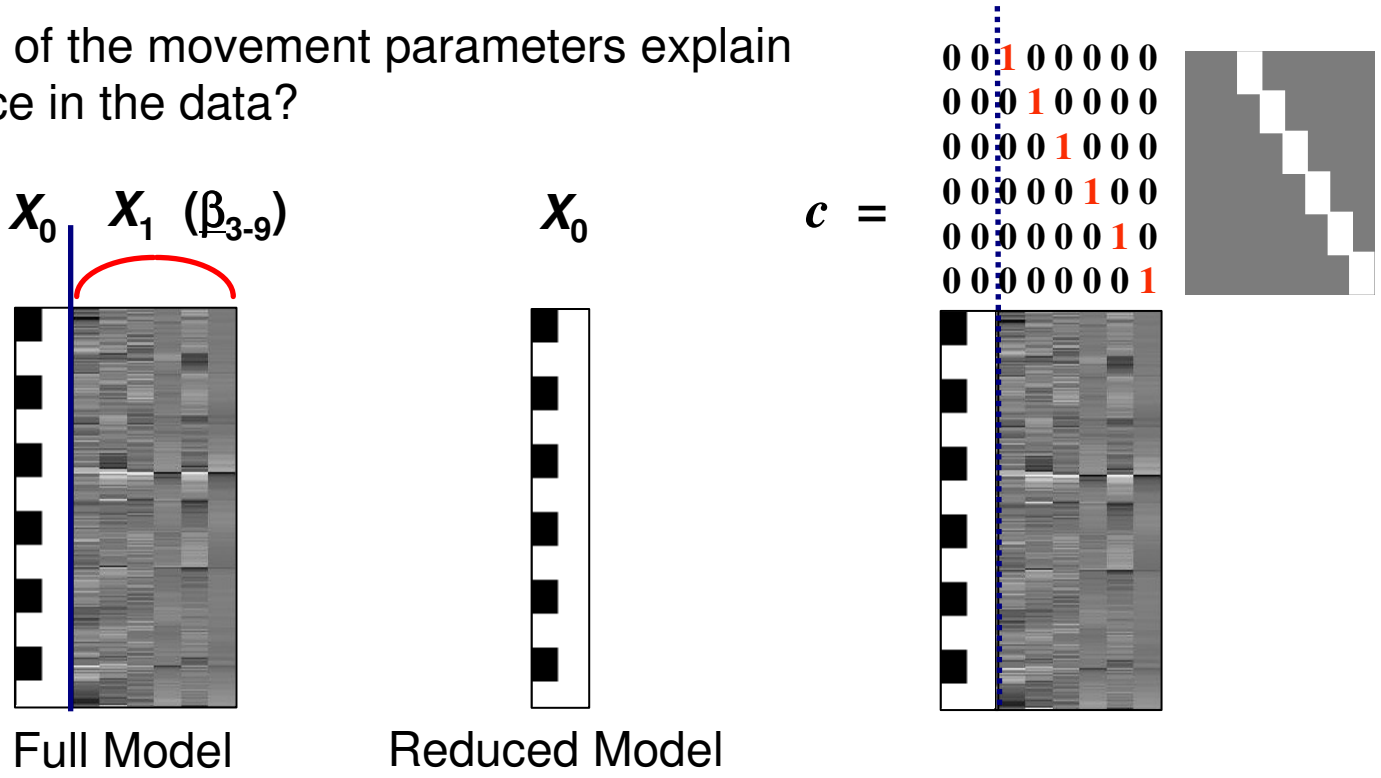
= Where is condition 1 greater than baseline OR condition 2 greater than baseline

F-contrasts

F statistics represent the ratio of variance explained by a model that includes the effects of interest (the full model) and a model that doesn't (the reduced model) – i.e. they're asking how much additional variance is explained by modelling the effects of interest.

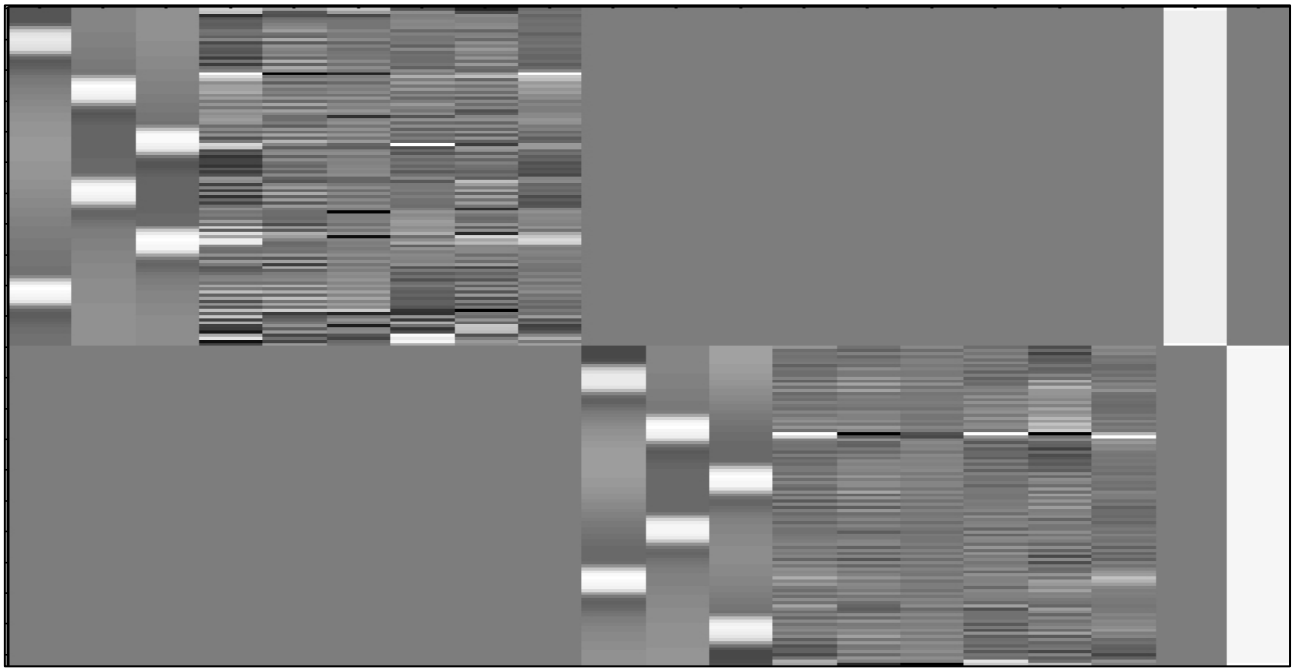
Running F contrasts creates an ess*.img (Extra Sum of Squares) and an spmF*.img

e.g. do any of the movement parameters explain any variance in the data?



F-contrasts

First 3 columns in each session represent the levels of a single factor:



Main effect of factor:

1	-1	0	0	0	0	0	0	0	1	-1	0	0	0	0	0	0	0	0	0
0	1	-1	0	0	0	0	0	0	0	1	-1	0	0	0	0	0	0	0	0

Or:

2	-1	-1	0	0	0	0	0	0	2	-1	-1	0	0	0	0	0	0	0	0
-1	2	-1	0	0	0	0	0	0	-1	2	-1	0	0	0	0	0	0	0	0
-1	-1	2	0	0	0	0	0	0	-1	-1	2	0	0	0	0	0	0	0	0

Statistical Inference

Statistic images evaluated by applying a threshold, and observing the spatial distribution of statistics that survive the threshold.

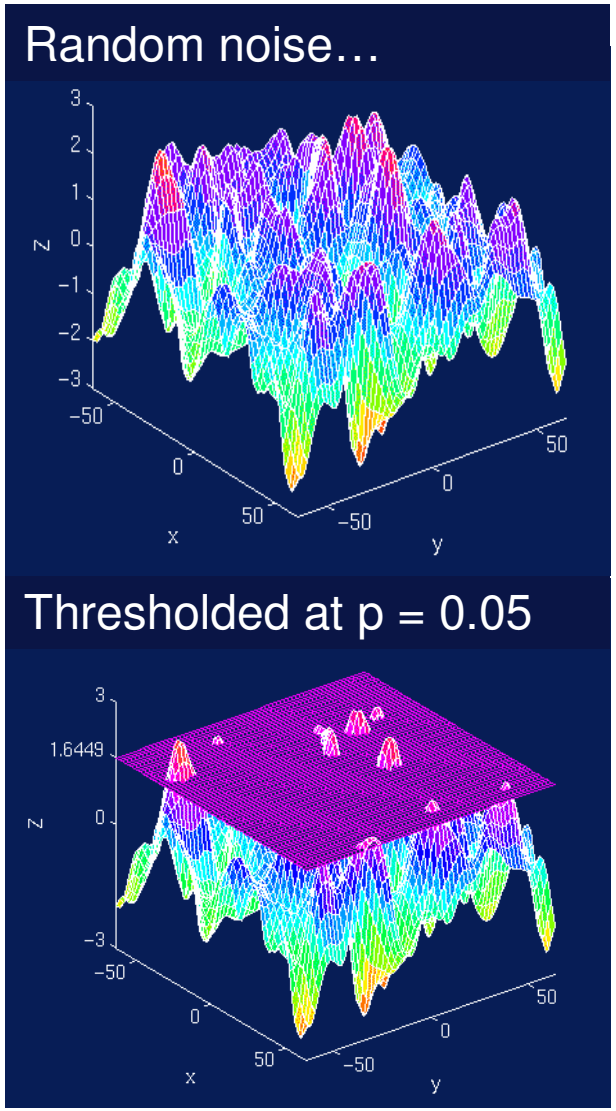
But there's a massive issue with multiple comparisons. Testing *100,000* voxels at p<0.05 means that on average, 5000 will be "significant" by chance (false positives, or "type I" errors)

Need to correct somehow for multiple comparisons

Bonferroni correction:

$p_{(corrected)} = p_{(uncorrected)}/n$ comparisons

This is only appropriate when the *n* tests independent. SPMs are smooth, meaning that nearby voxels are correlated, making Bonferroni overly conservative in fMRI.



Statistical Inference

SPM uses the “Theory of Random Fields” to calculate an estimate of the true number of independent comparisons.

This depends greatly on the smoothness of the data (smoother data = fewer independent comparisons)

Actual spatial resolution measured in “resels” (resolution elements).

As well as beta and error images, estimating an SPM design also outputs an image called RPV.img – this shows the “Resels per Voxel” (i.e. the smoothness of the data) at each voxel.

In the results interface, the FWE corrected (=Family Wise Error corrected) p value controls for multiple comparisons in a similar way to Bonferroni correction, but using the number of resels rather than the number of voxels to adjust the p value.

Depending on the smoothness, 100,000 voxels ~ 500 resels.

Statistical Inference

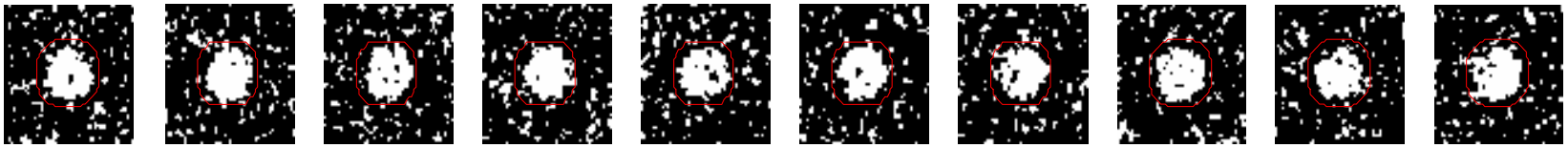
Familywise Error Rate (FWER)

- Chance of *any* false positives
- Controlled by Bonferroni & Random Field Methods

False Discovery Rate (FDR)

- Proportion of false positives among rejected tests

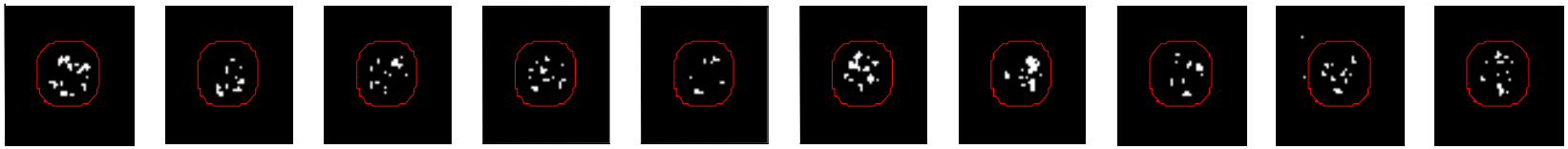
Control of Per Comparison Rate at 10%



11.3% 11.3% 12.5% 10.8% 11.5% 10.0% 10.7% 11.2% 10.2% 9.5%

Percentage of Null Pixels that are False Positives

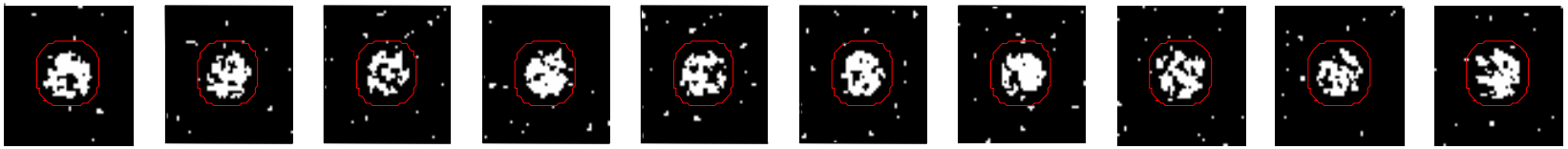
Control of Familywise Error Rate at 10%



FWE

Occurrence of Familywise Error

Control of False Discovery Rate at 10%

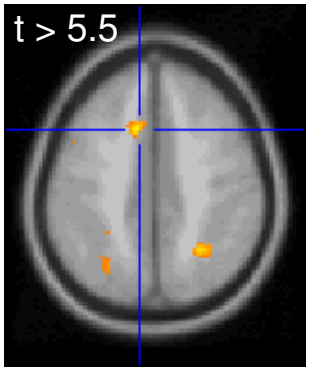


6.7% 10.4% 14.9% 9.3% 16.2% 13.8% 14.0% 10.5% 12.2% 8.7%

Percentage of Activated Pixels that are False Positives

Statistical Inference

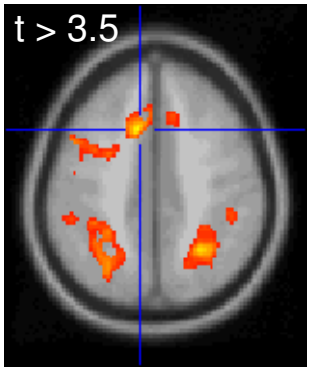
High Threshold



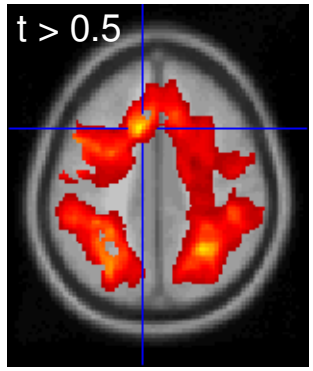
Good Specificity

Poor Power
(risk of false
negatives)

Med. Threshold



Low Threshold



Poor Specificity
(risk of false
positives)

Good Power