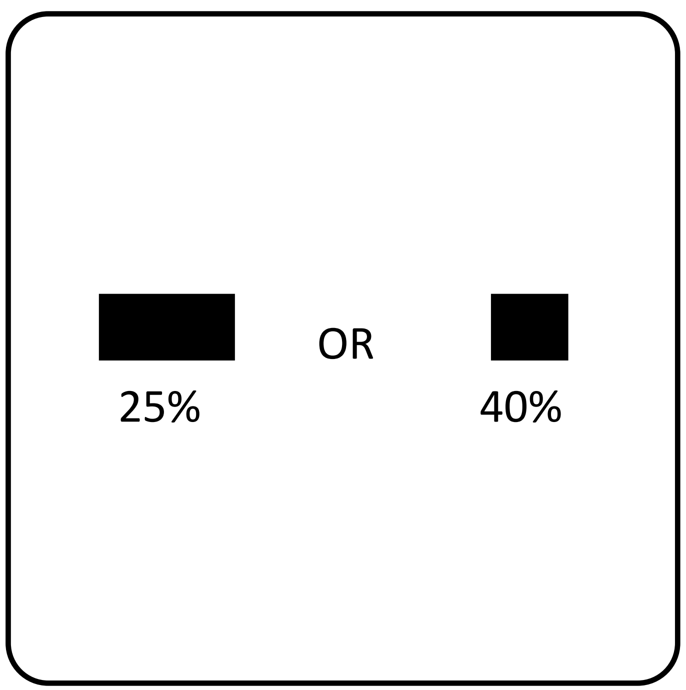
Using OSL: Sensor Space Analyses

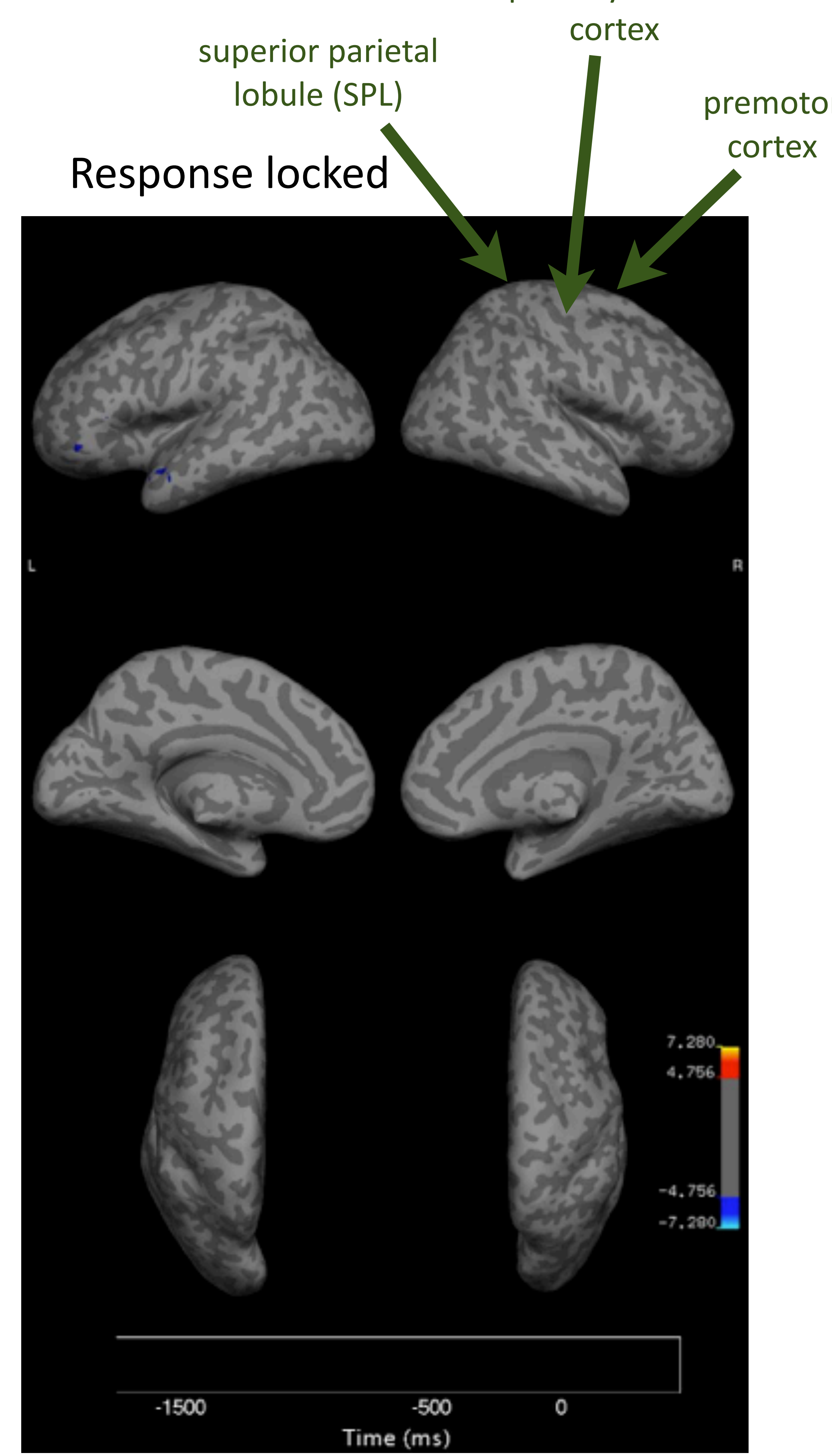
OHBA Analysis Workshop

Mark Woolrich

- Induced analysis of the decision making period:
  - ***source reconstruction***
  - *epoching*: time-locked to when the response is given
  - compute the ***average evoked power*** (the induced response, ERD/ERS) from 1-12Hz
  - ***group averaged*** over 30 subjects



Hunt et al., Nature Neuroscience, 2012.



Talk Outline

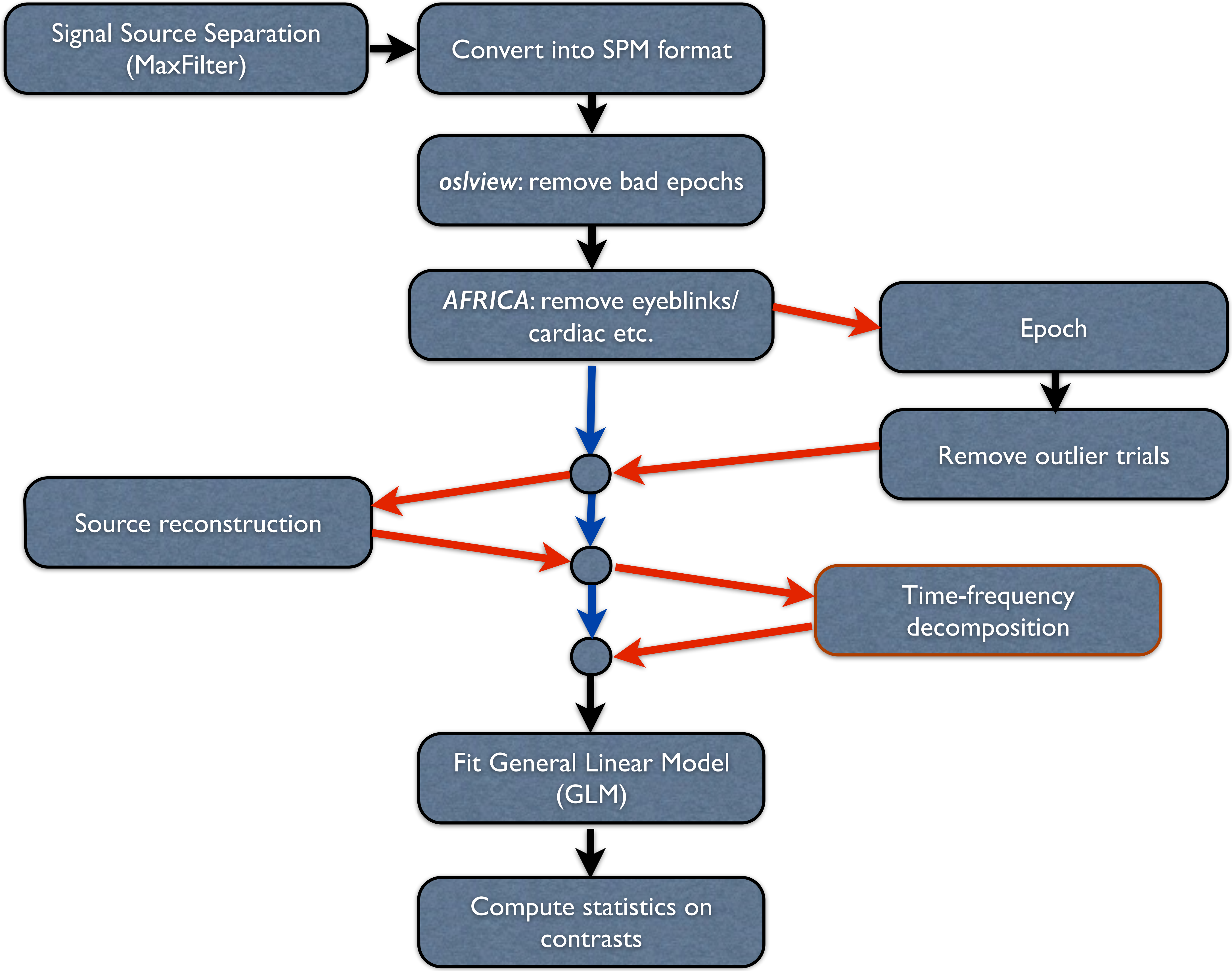
- Analysing continuous data in sensor space
    - time-wise GLM

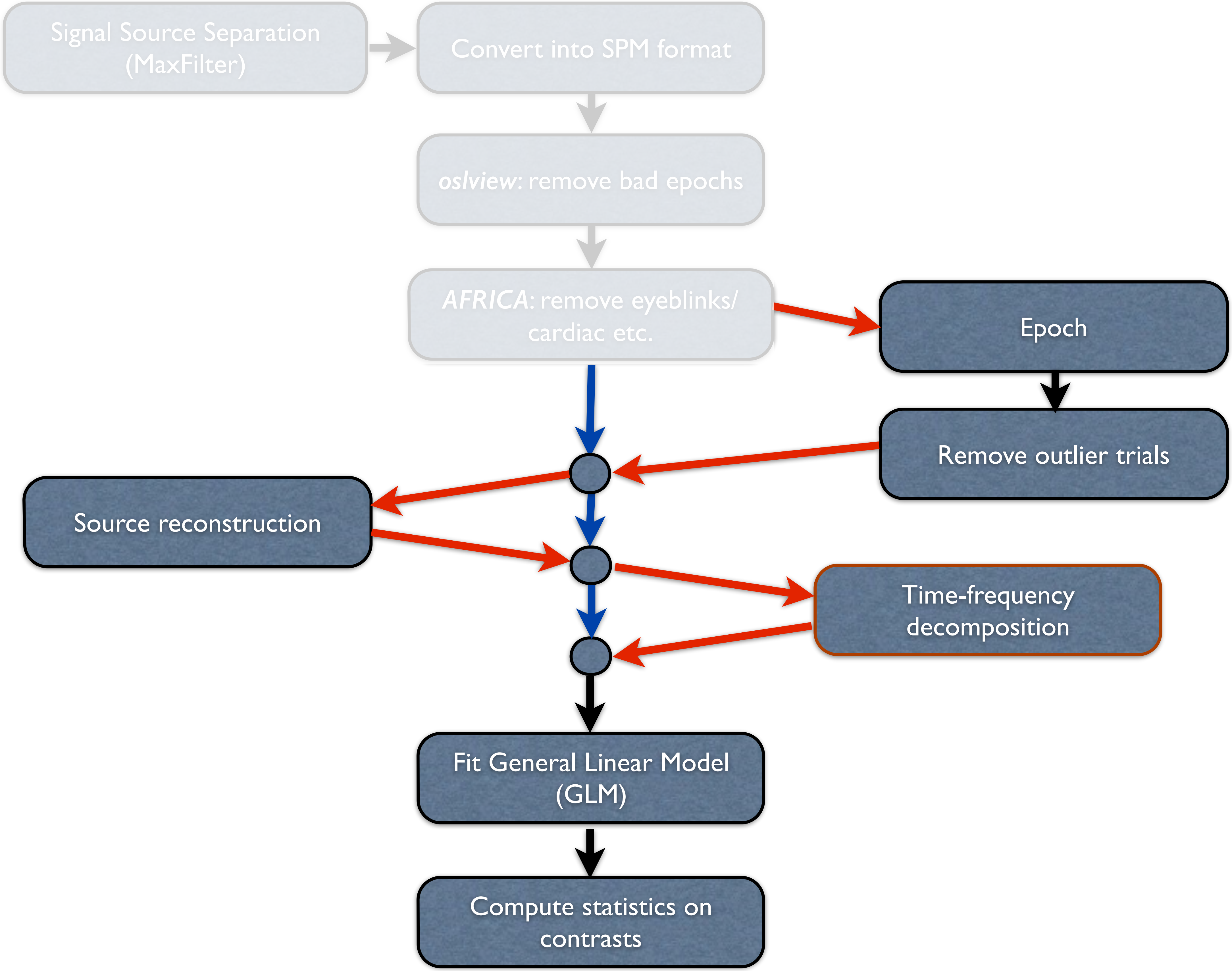
- Analysing epoched data in sensor space
    - trial-wise GLM

- OSL (OHBA's Software Library):
    - OAT (OSL's Analysis Tool)

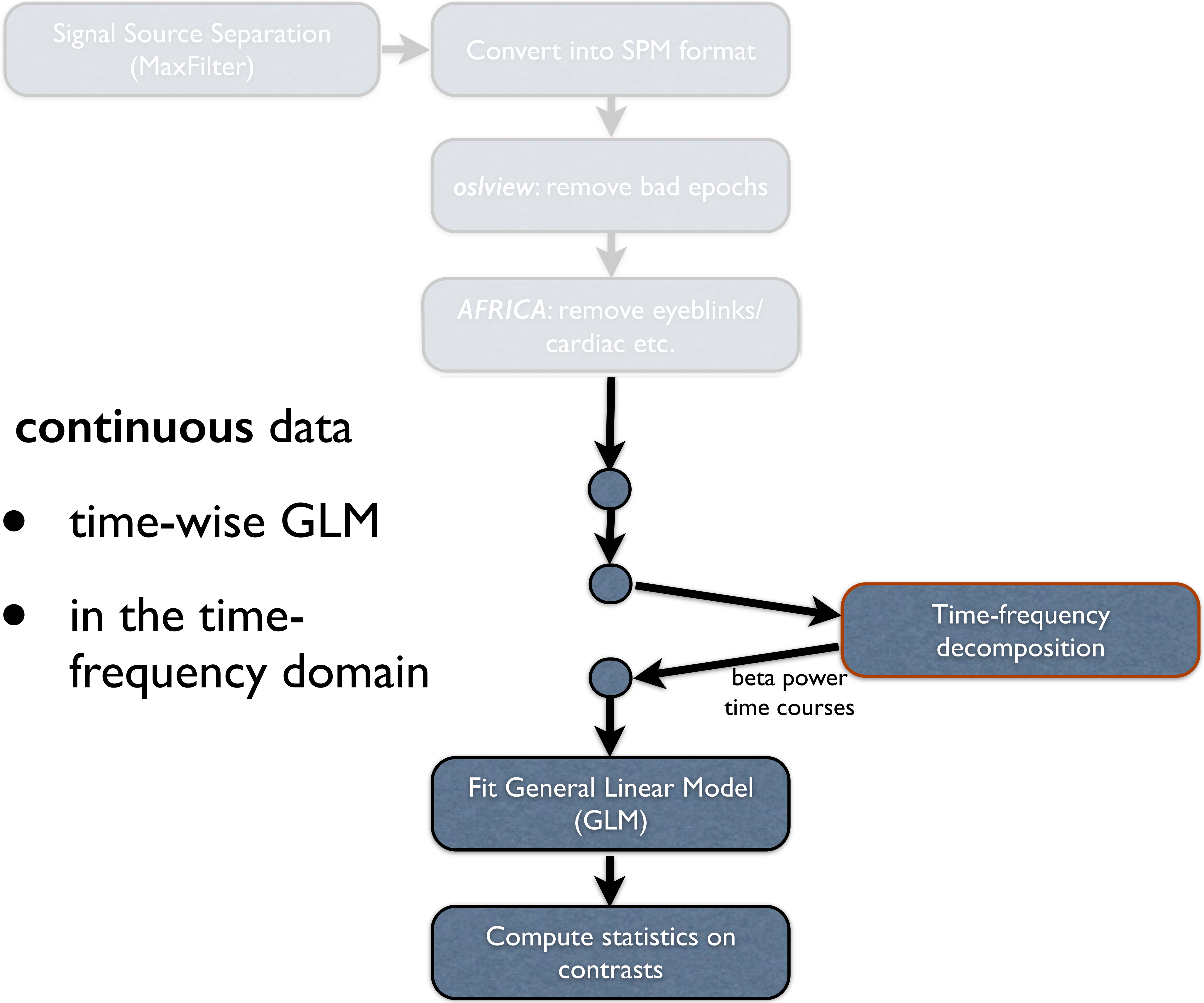
Continuous Data Example

- Finger tapping versus rest
- 20sec blocks of alternating:
  - ➡ rest
  - ➡ finger tapping
- We want to compare changes in beta power (13-30 Hz)

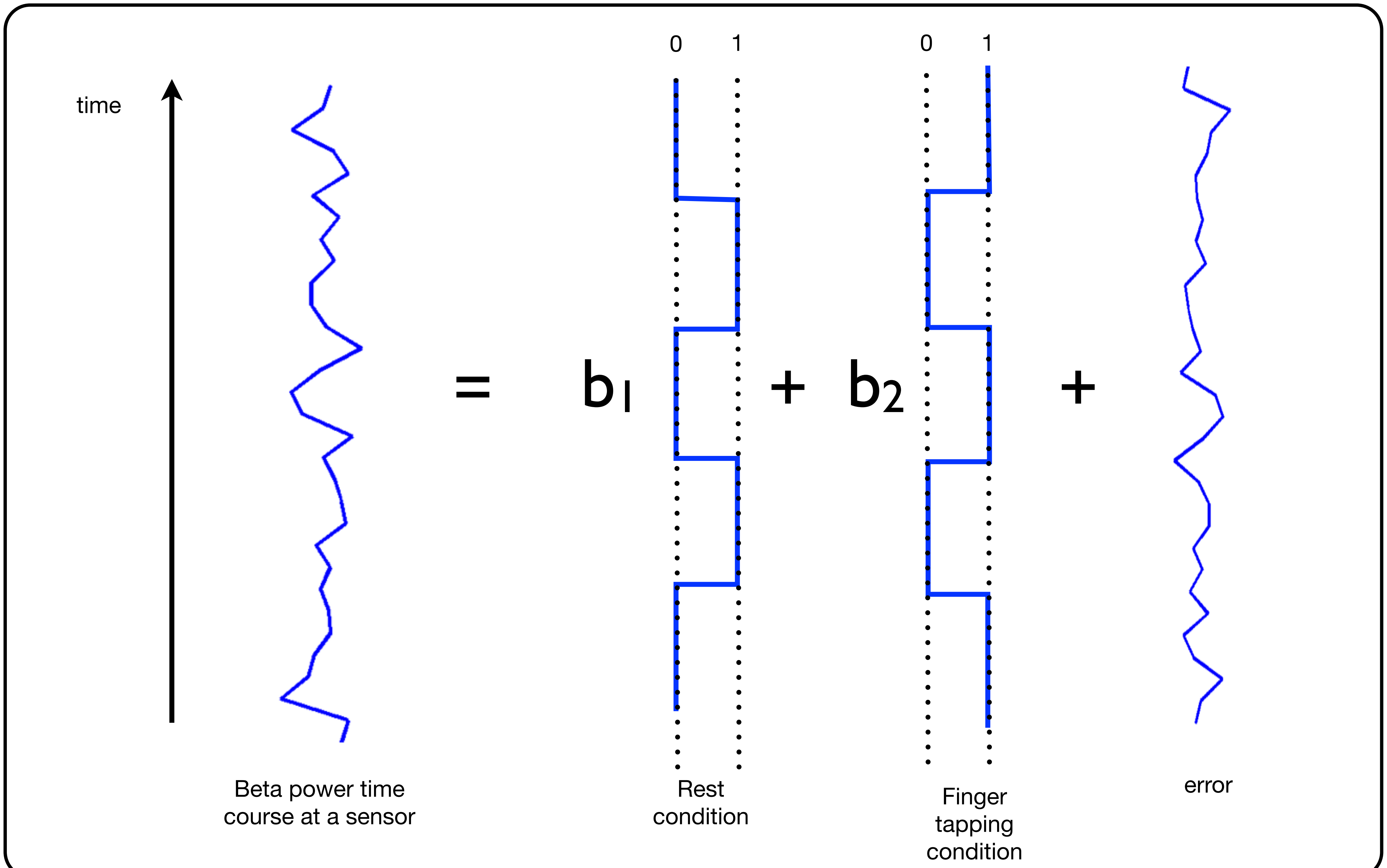

Signal Source Separation (MaxFilter)
Convert into SPM format
oslview: remove bad epochs
AFRICA: remove eyeblinks/ cardiac etc.
Epoch
Remove outlier trials
Source reconstruction
Time-frequency decomposition
Fit General Linear Model (GLM)
Compute statistics on contrasts


Signal Source Separation (MaxFilter)
Convert into SPM format
oslview: remove bad epochs
AFRICA: remove eyeblinks/ cardiac etc.
Epoch
Remove outlier trials
Source reconstruction
Time-frequency decomposition
Fit General Linear Model (GLM)
Compute statistics on contrasts

- **continuous** data
  - time-wise GLM
  - in the time-frequency domain

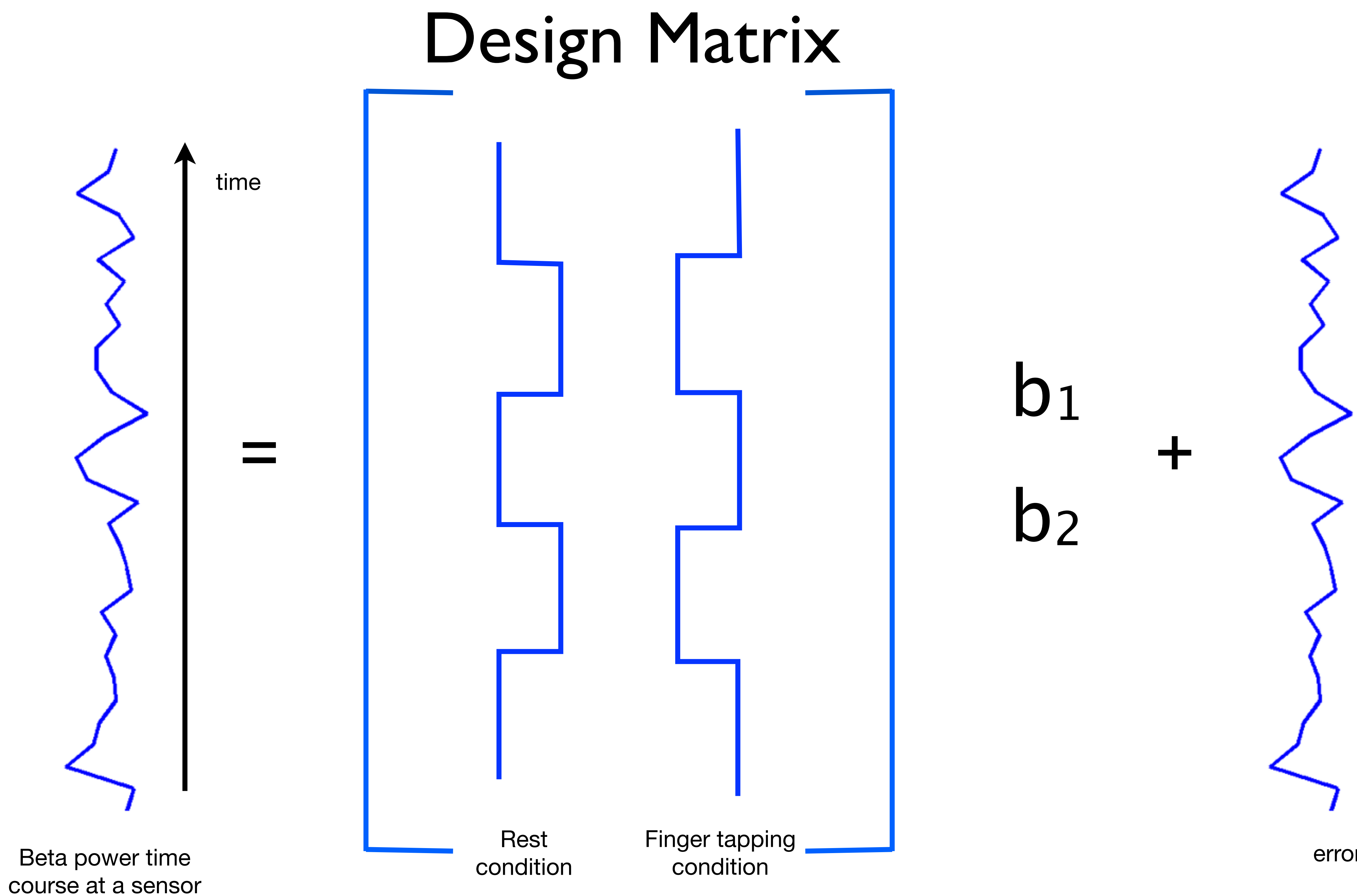


Continuous Data Time-wise Example



Time-wise GLM Example

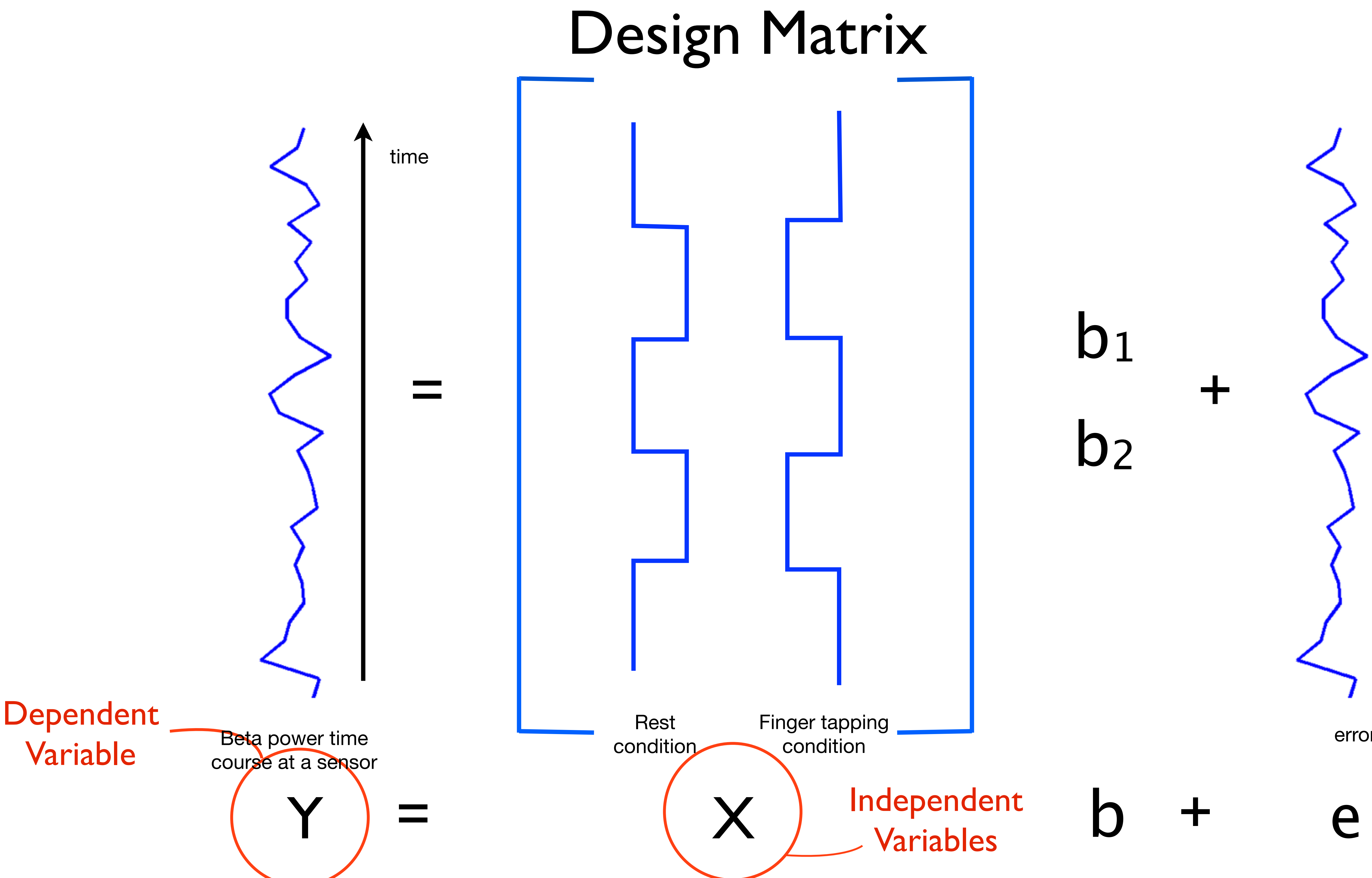
More formally we fit (for each sensor) a General Linear Model (GLM):



$$Y = X \; b + e$$

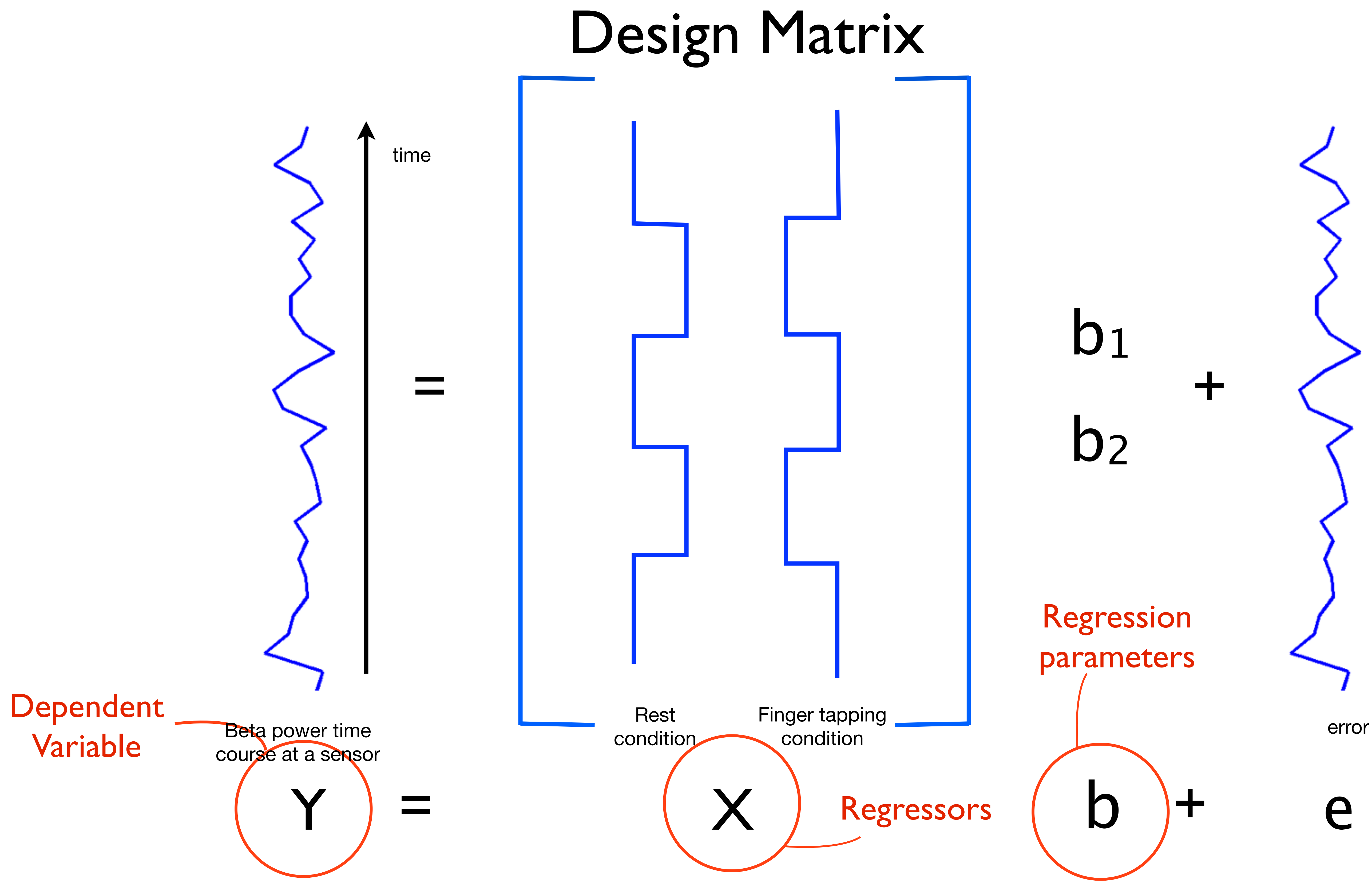
Time-wise GLM Example

More formally we fit (for each sensor) a General Linear Model (GLM):



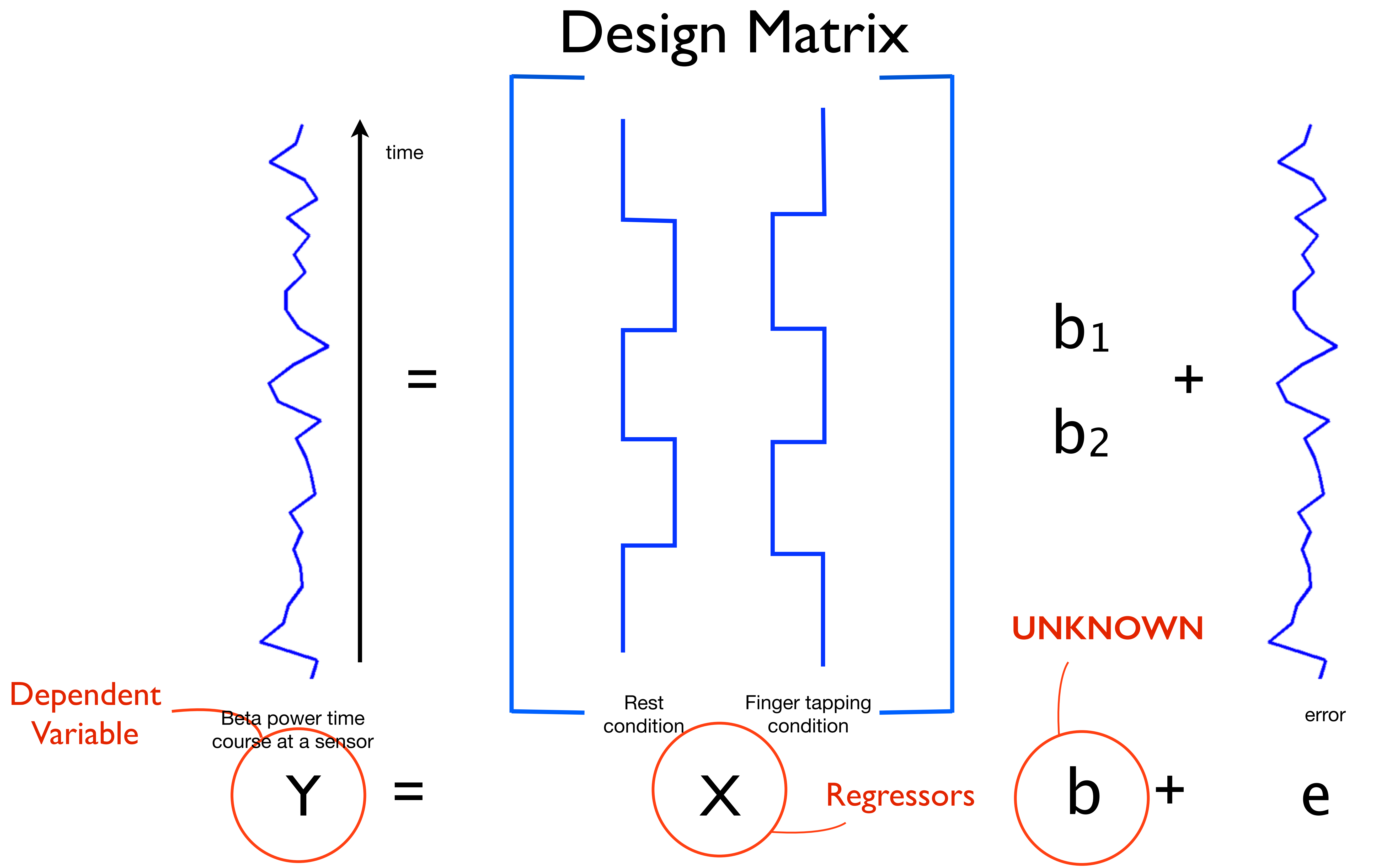
Time-wise GLM Example

More formally we fit (for each sensor) a General Linear Model (GLM):

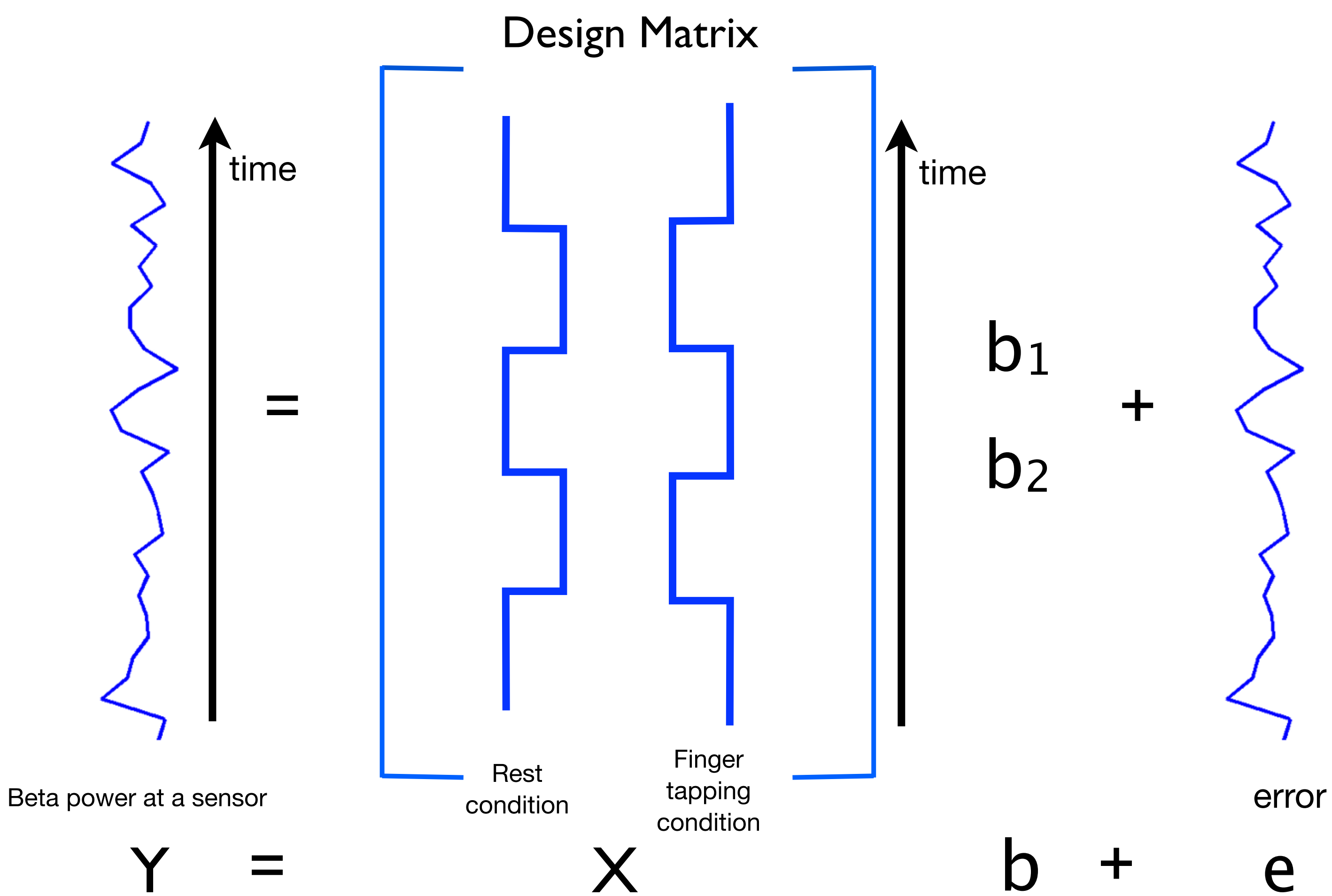


Time-wise GLM Example

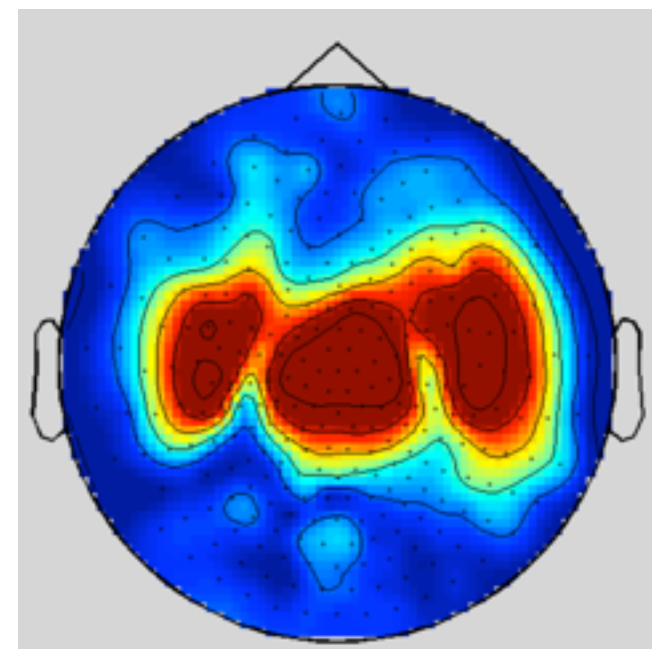
More formally we fit (for each sensor) a General Linear Model (GLM):



GLM can be fitted at each sensor



Sensor (topographic) map of mean beta power decrease between finger tapping and rest, i.e. B_1-B_2



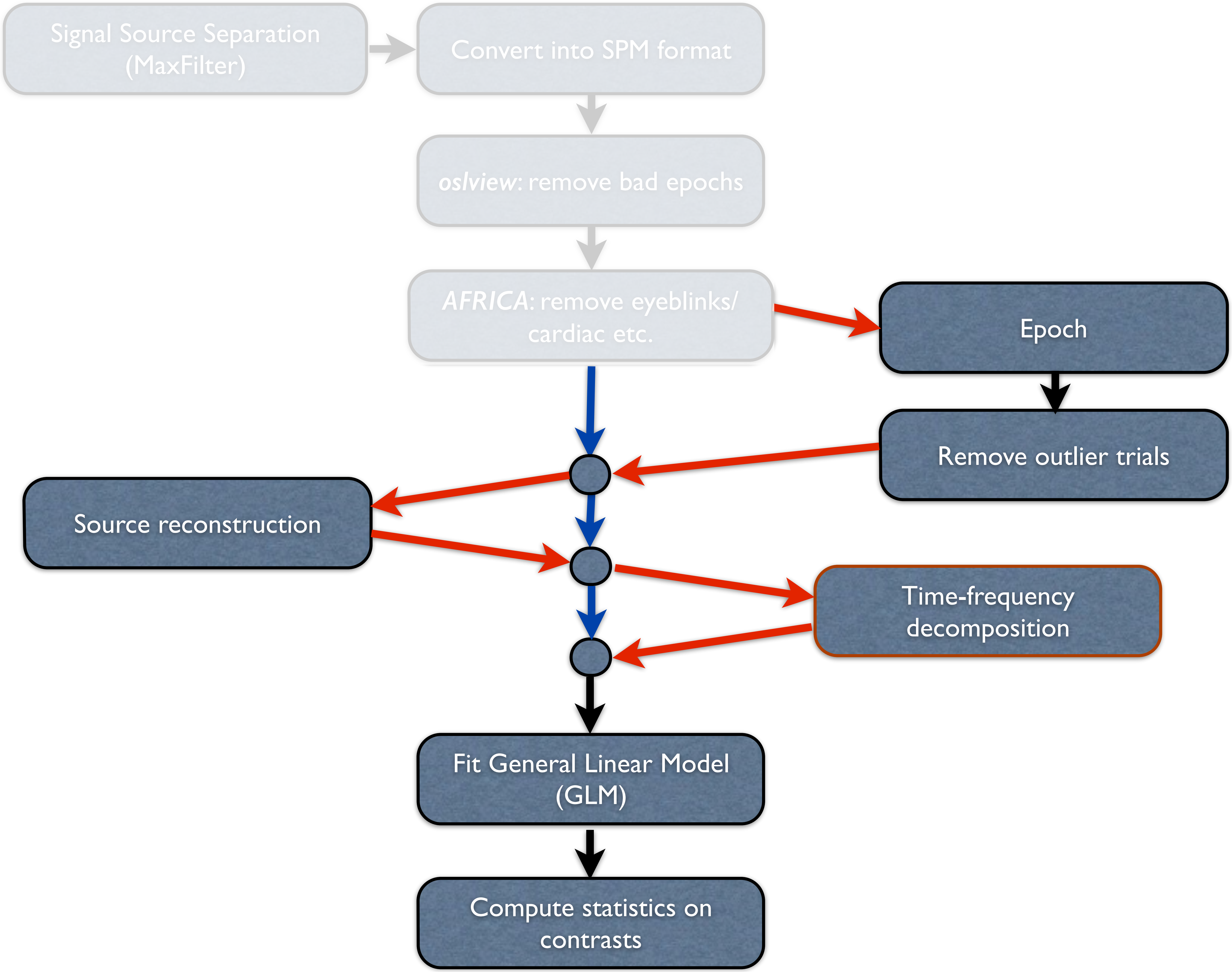
Repeat for all sensors

$$Y=Xb+e$$

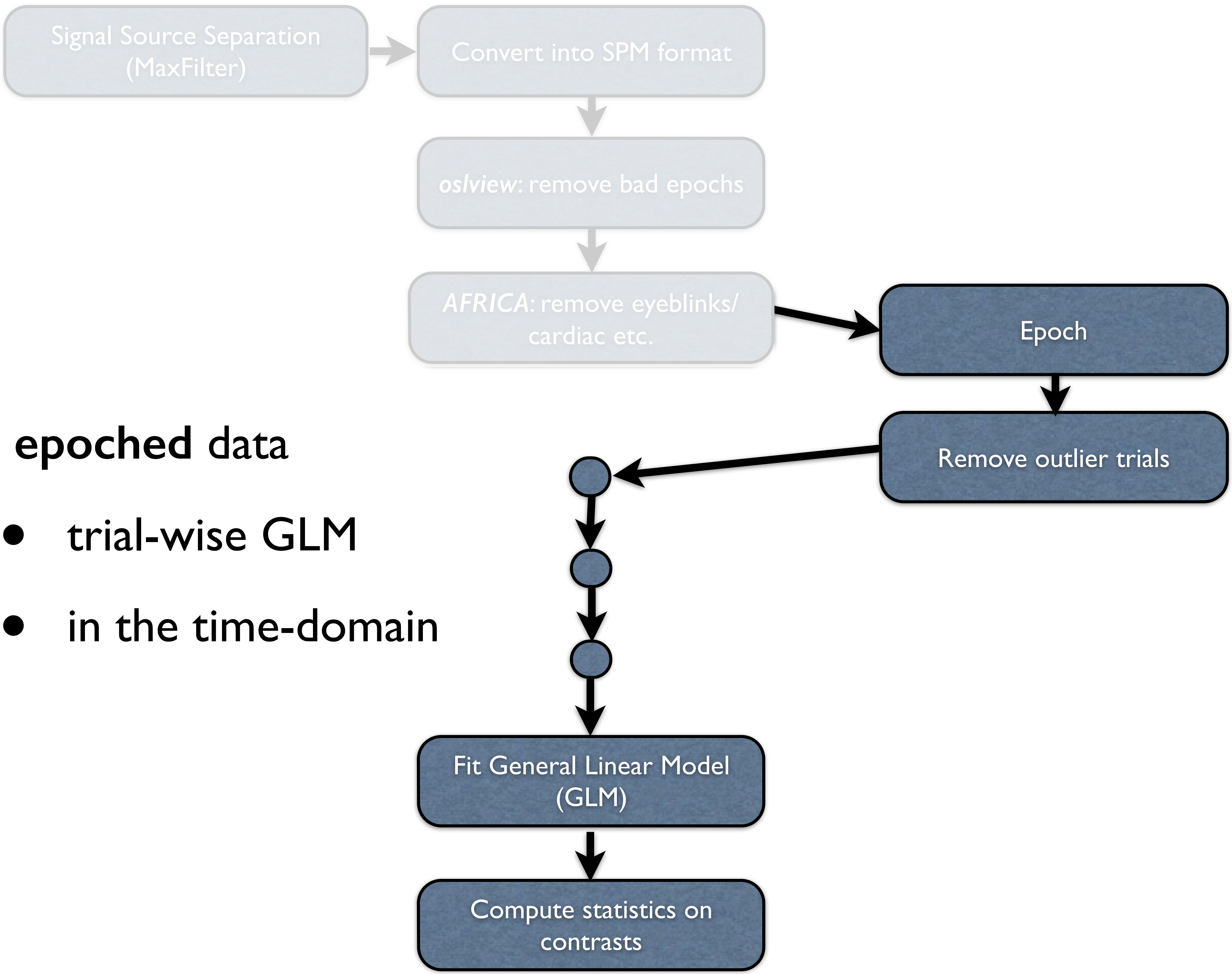
Ordinary Least Squares (OLS) estimate $B=[B_1, B_2]$, which are the Parameter Estimates (PEs) of b

Epoched Data Example

- Faces versus motorbikes
  - ➡ 240 trials (epochs) of presenting pictures of faces
  - ➡ 120 trials (epochs) of presenting pictures of motorbikes
- We want to compare the responses time-locked to stimulus presentation (i.e. the Event-Related Fields (ERFs))



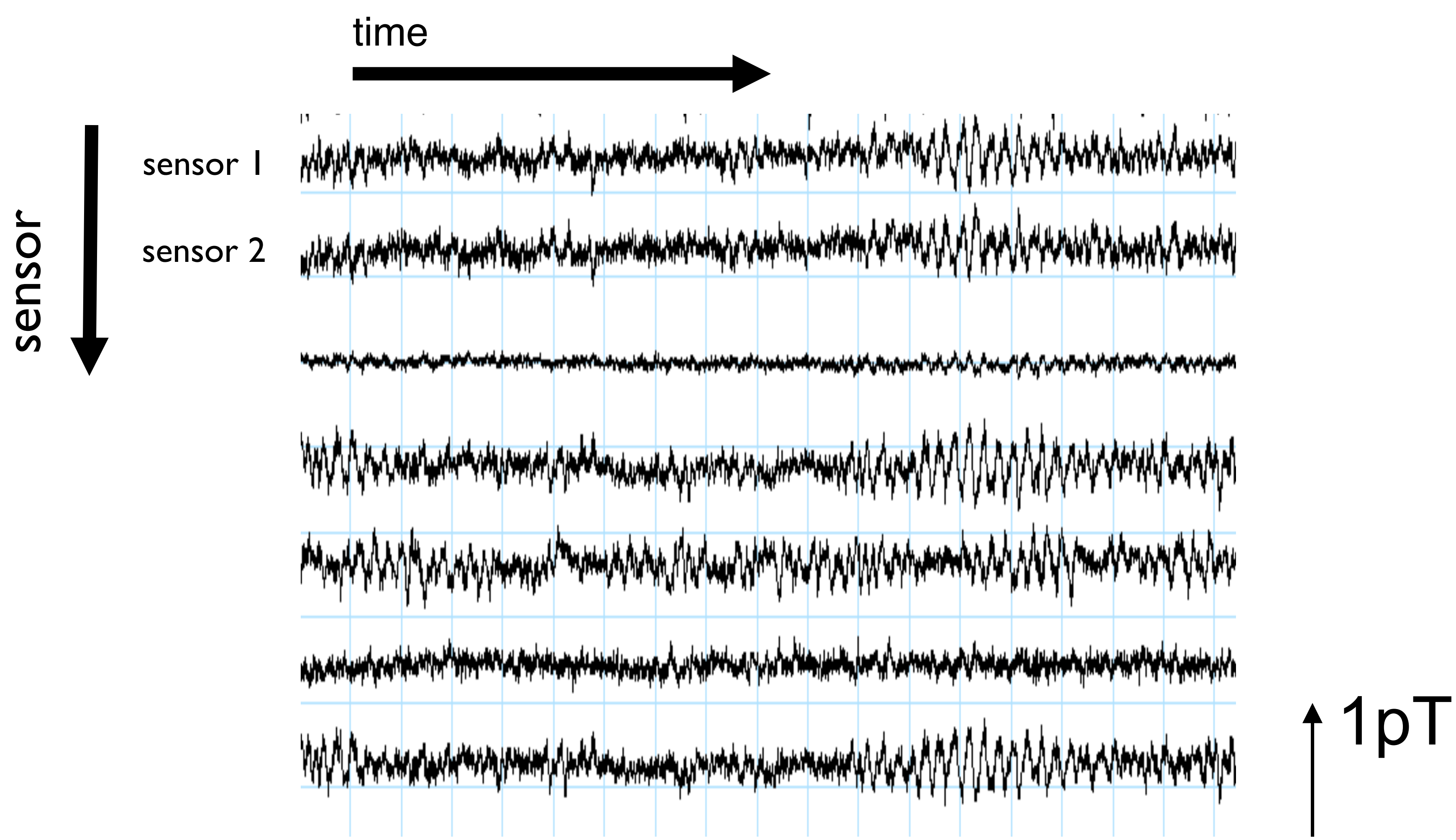
Signal Source Separation (MaxFilter)
Convert into SPM format
oslview: remove bad epochs
AFRICA: remove eyeblinks/ cardiac etc.
Epoch
Remove outlier trials
Source reconstruction
Time-frequency decomposition
Fit General Linear Model (GLM)
Compute statistics on contrasts



- **epoched** data
    - trial-wise GLM
    - in the time-domain

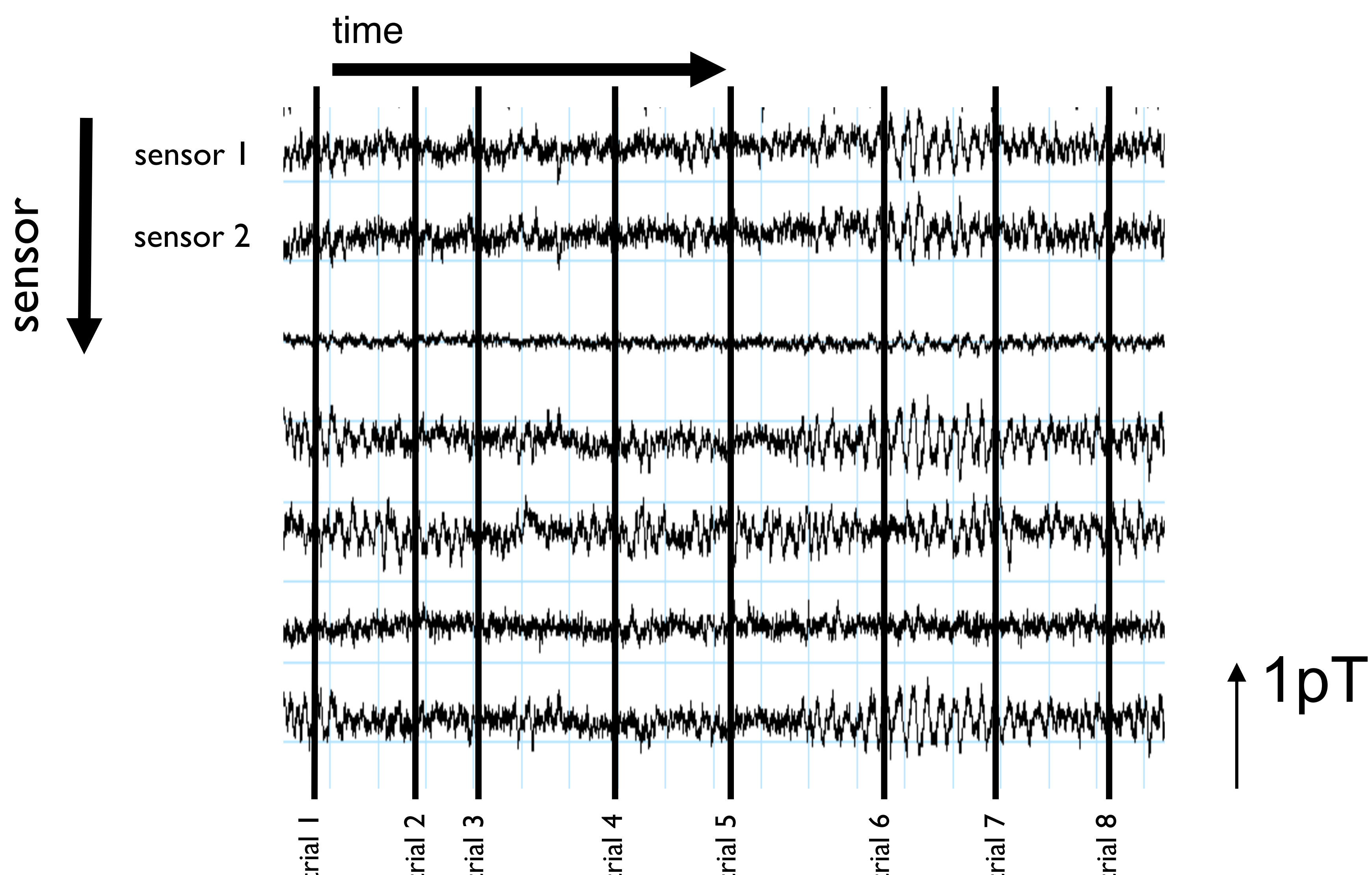
Epoching and ERFs

Epoching takes continuous 2D data:
sensors x timepoints



Epoching and ERFs

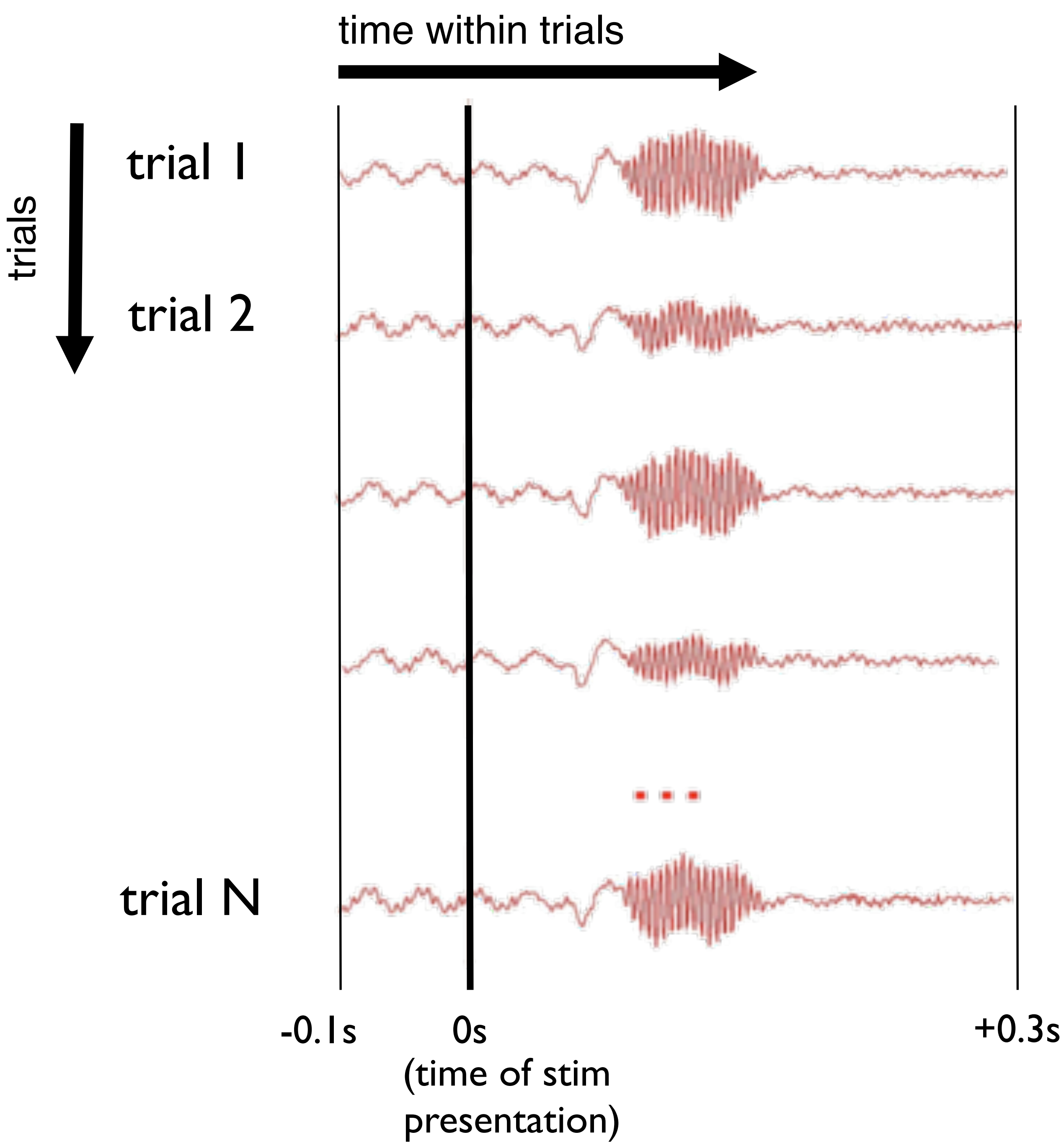
Epoching takes continuous 2D data:
sensors x timepoints



Epoching and ERFs

Epoching takes continuous 2D data:
sensors x timepoints
and produces epoched 3D data:
sensors x trials x timepoints-within-trial

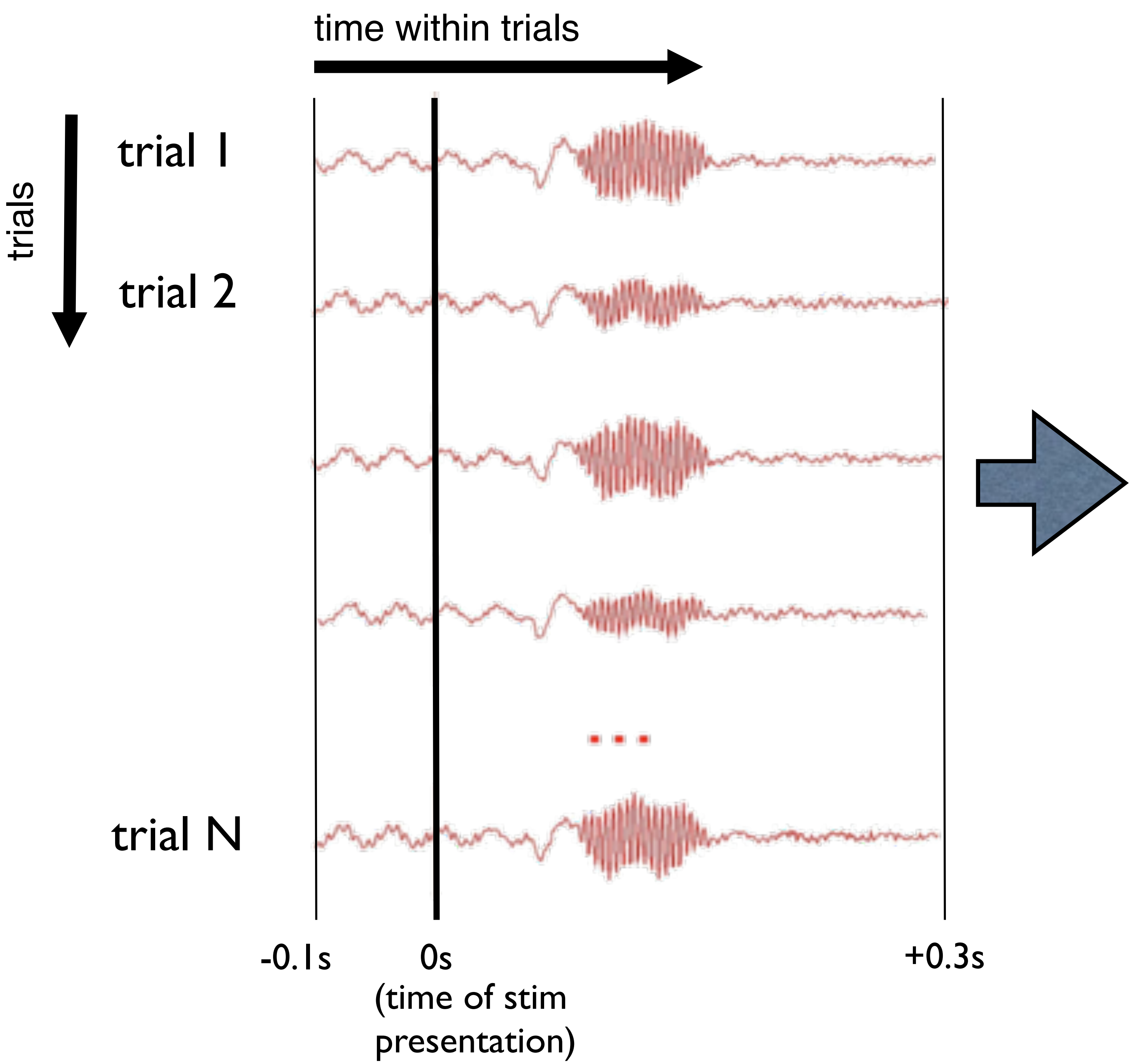
E.g. at one sensor:



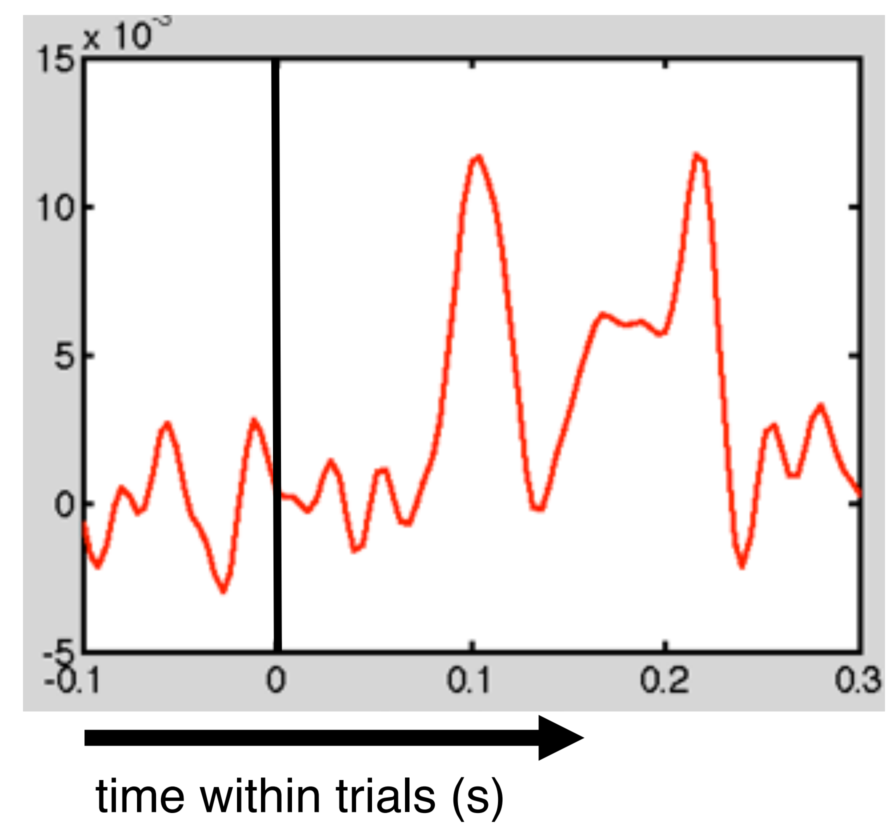
Epoching and ERFs

Epoching takes continuous 2D data:
sensors x timepoints
and produces epoched 3D data:
sensors x trials x timepoints-within-trial

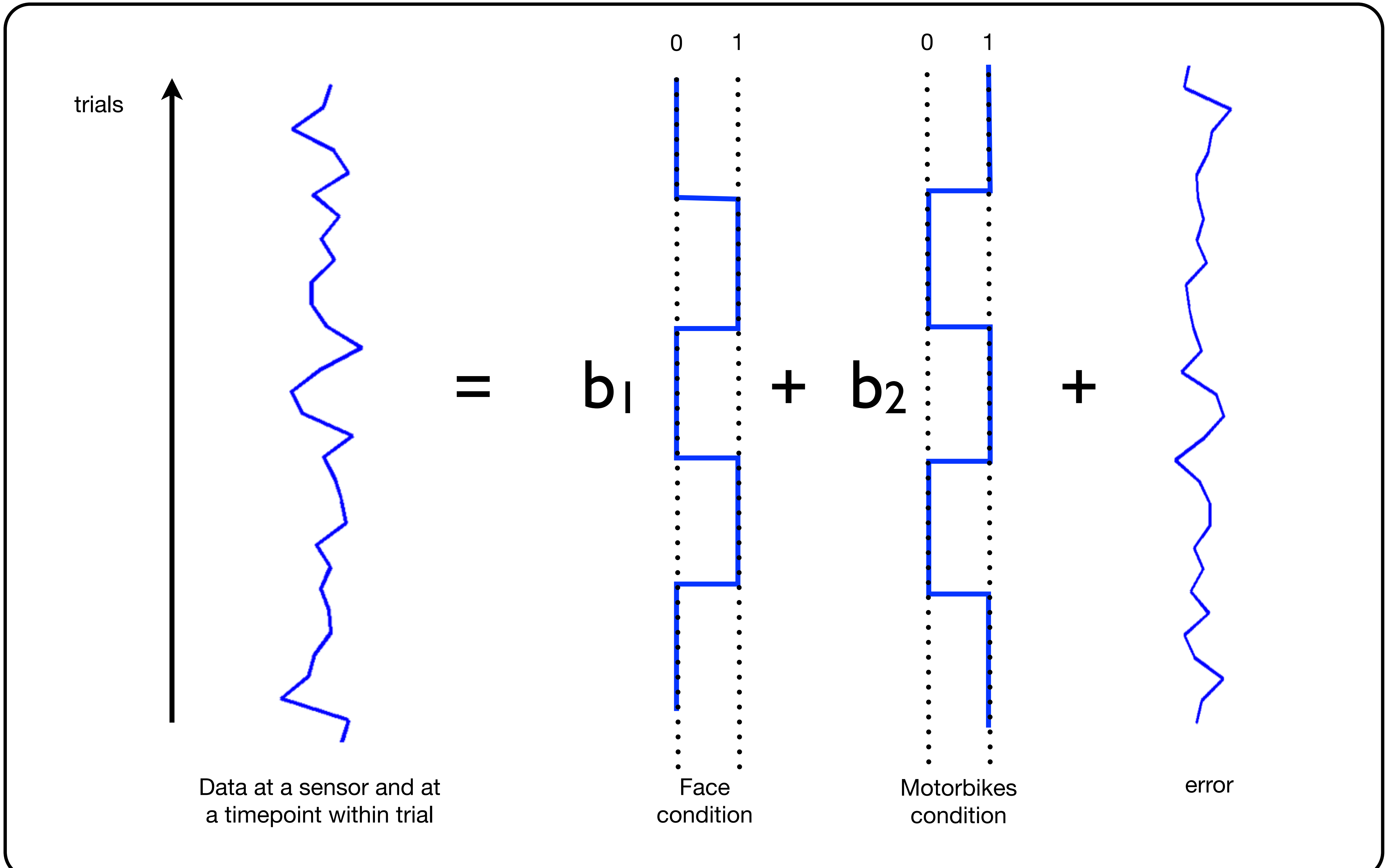
E.g. at one sensor:



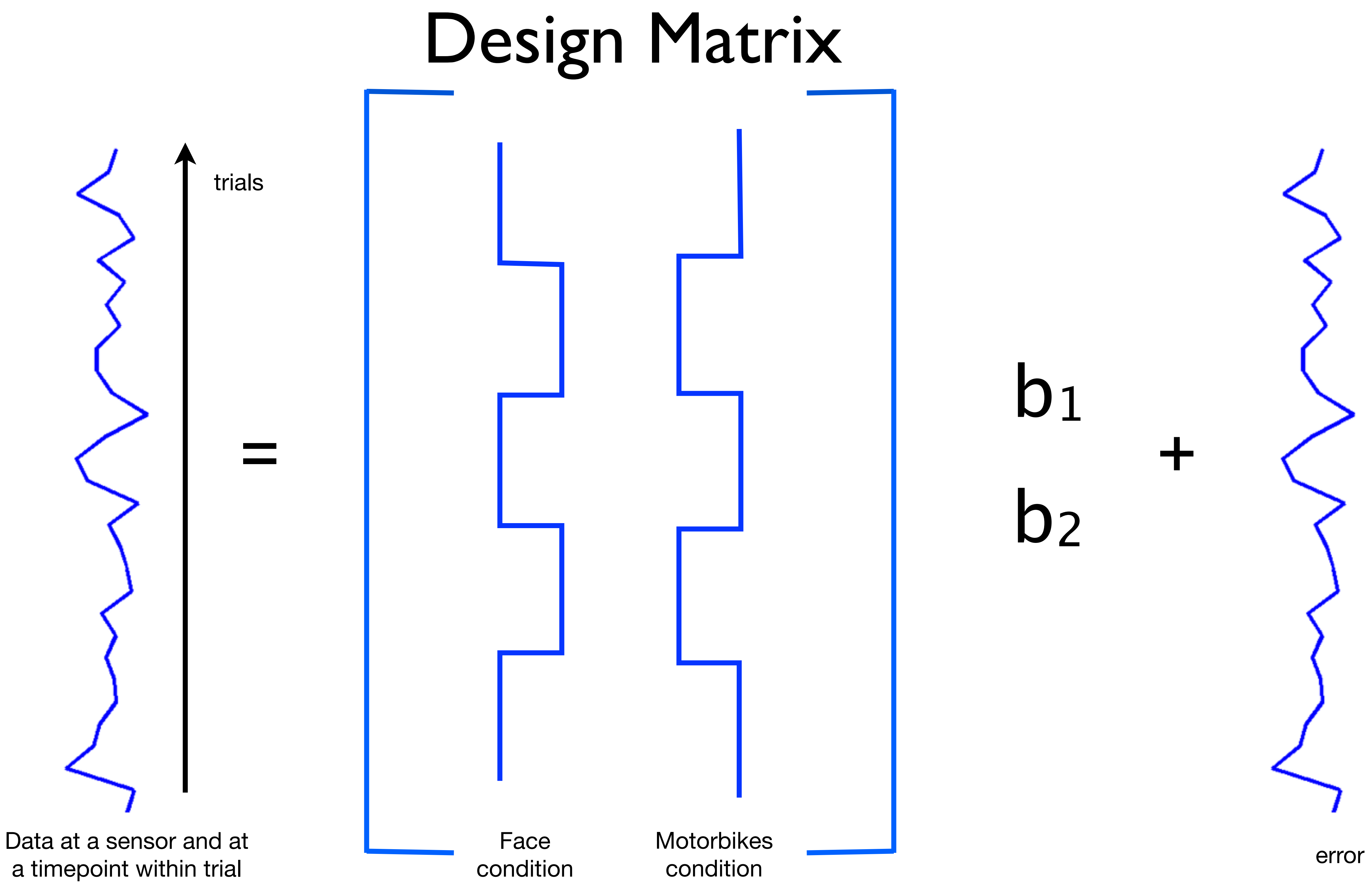
Average over all "face" epochs/ trials to get a face Event Related Field (ERF):



Trial-wise GLM

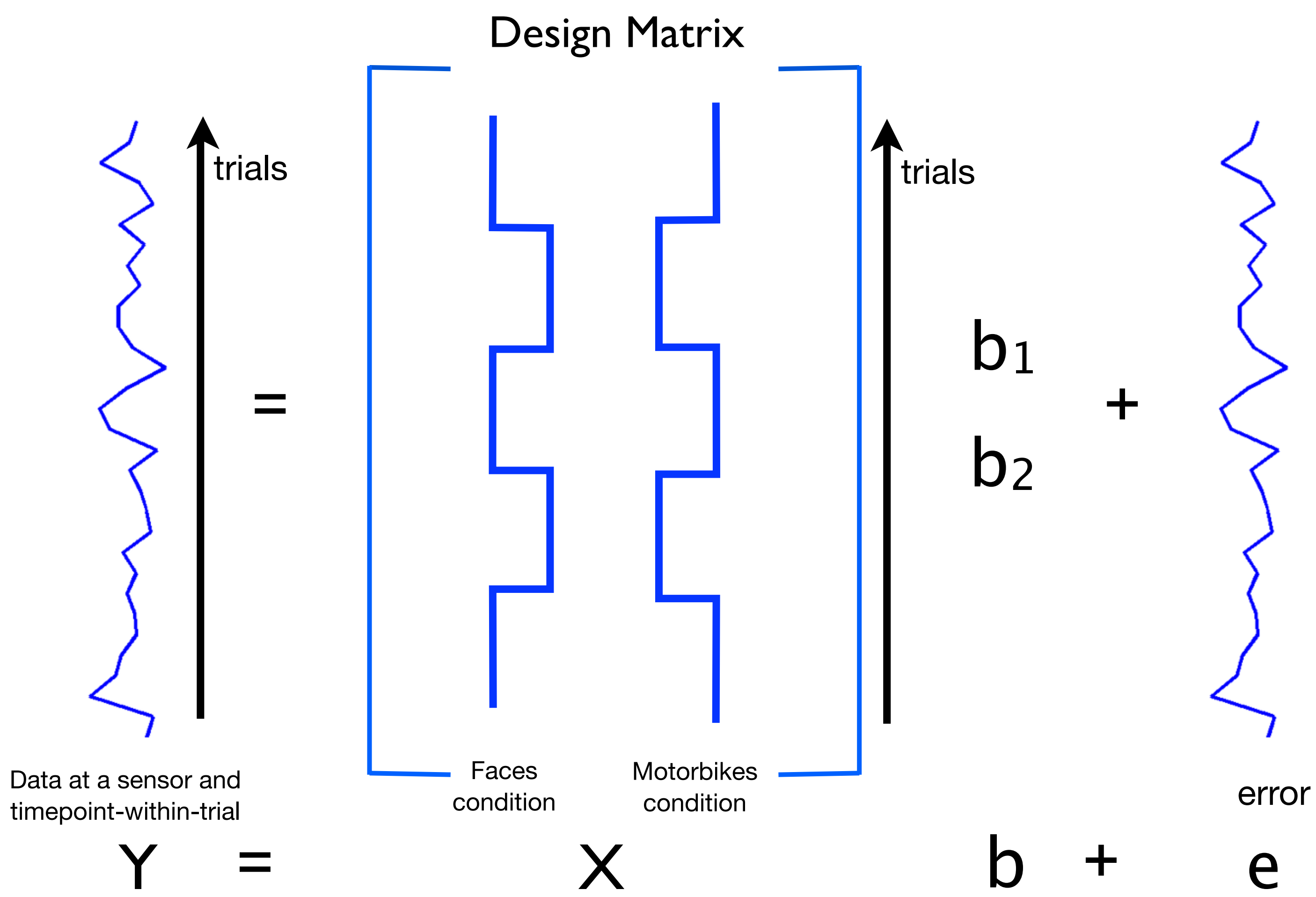


Trial-wise GLM

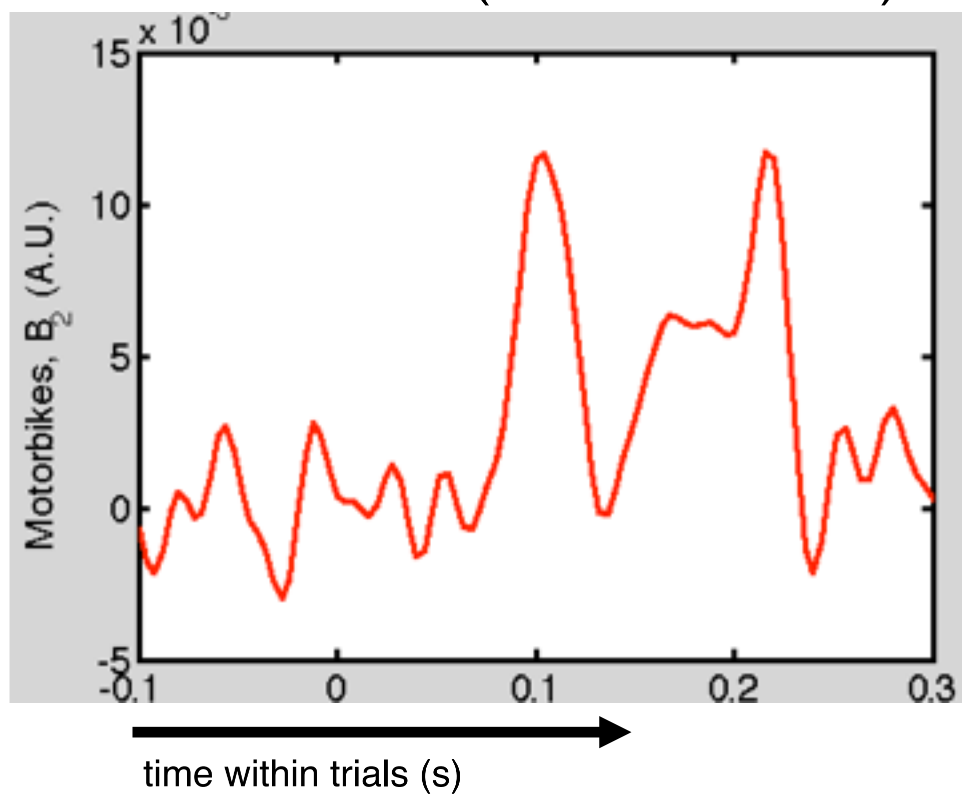


$$Y = X b + e$$

GLM can be fit to each time-point within trial



Motorbikes (B_2) at a sensor near the visual cortex (motorbike ERF)

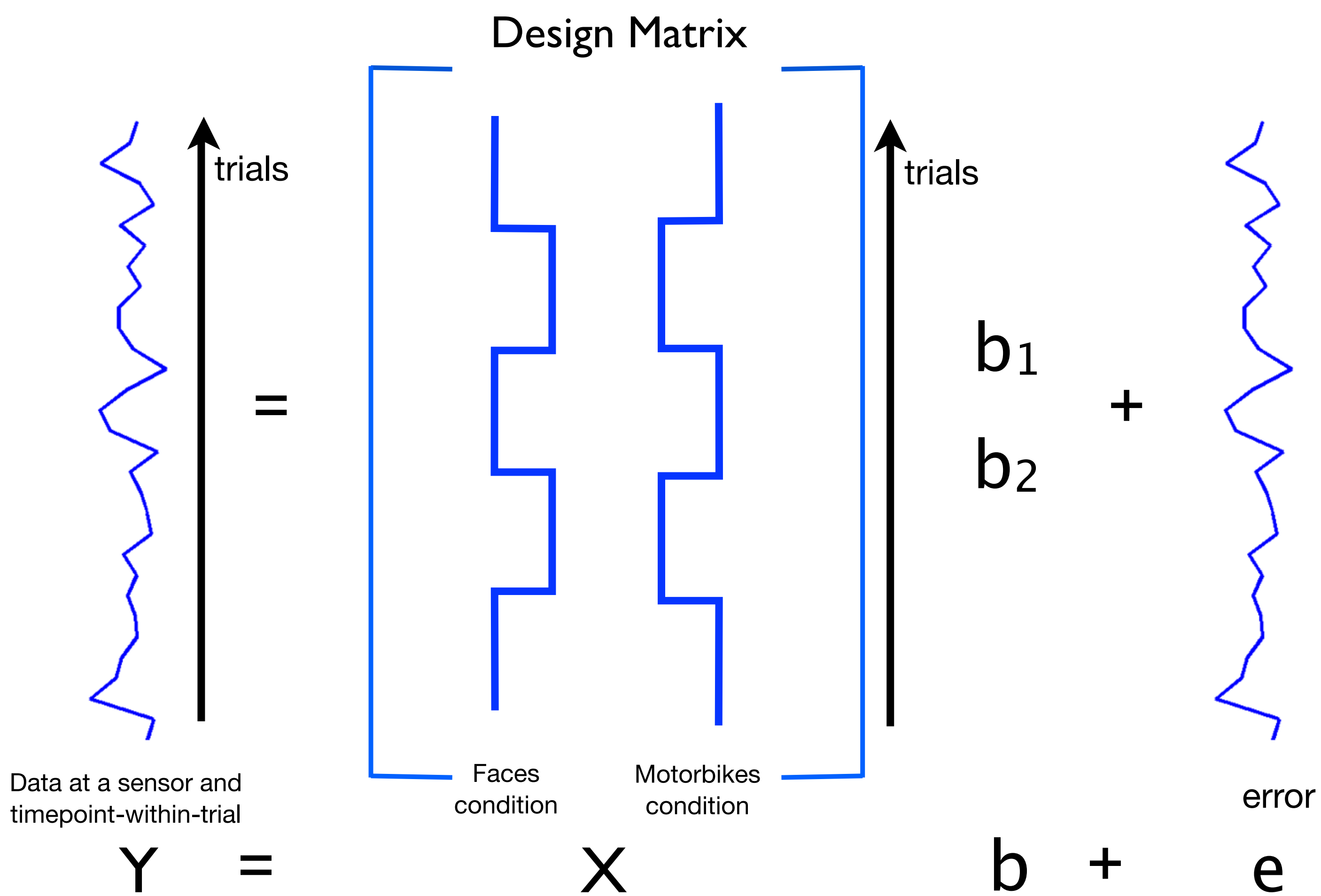


Repeat for all timepoints within-trial at a sensor

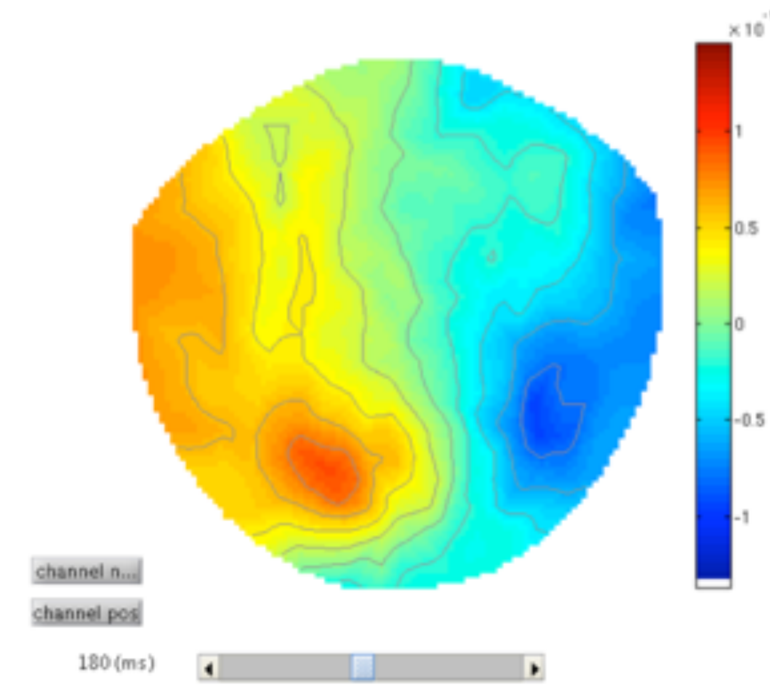
$$Y=Xb+e$$

→ Estimate B, which are the Parameter Estimates (PEs) of b

The GLM can be fit to each sensor



Motorbikes (B_2) at 100ms post-stimulus



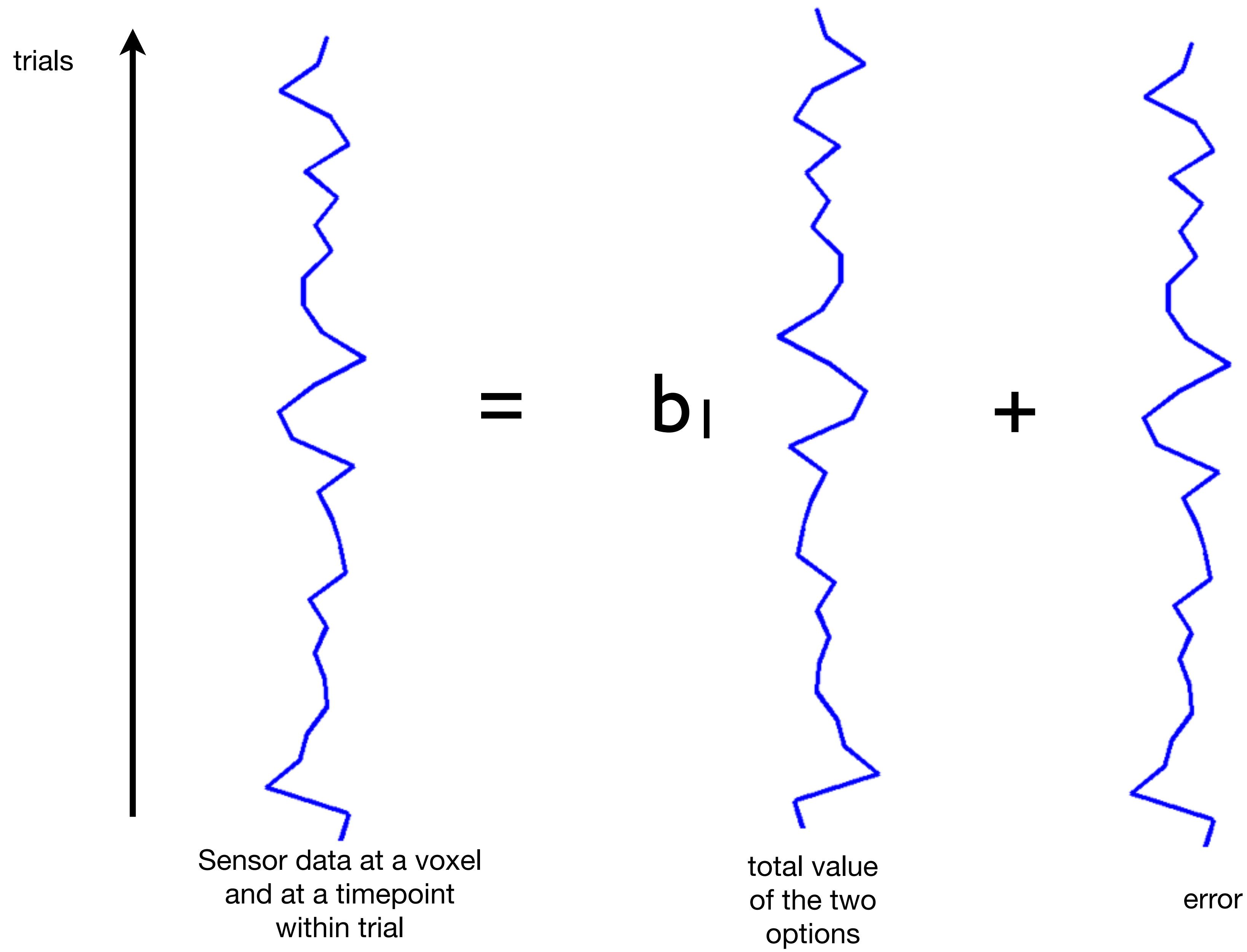
Repeat for all sensors at a time-within-trial

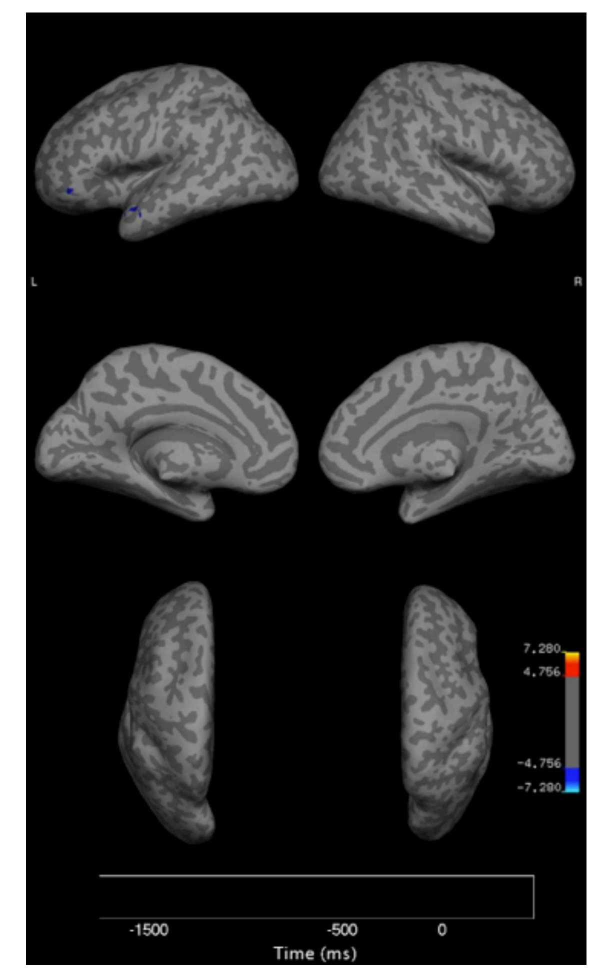
$$Y=Xb+e$$

Estimate B, which are the Parameter Estimates (PEs) of b

GLM

Note that the GLM is a general framework, e.g. in which we can also fit continuous variables:





Hunt, Nature Neuroscience, 2012

Contrasts

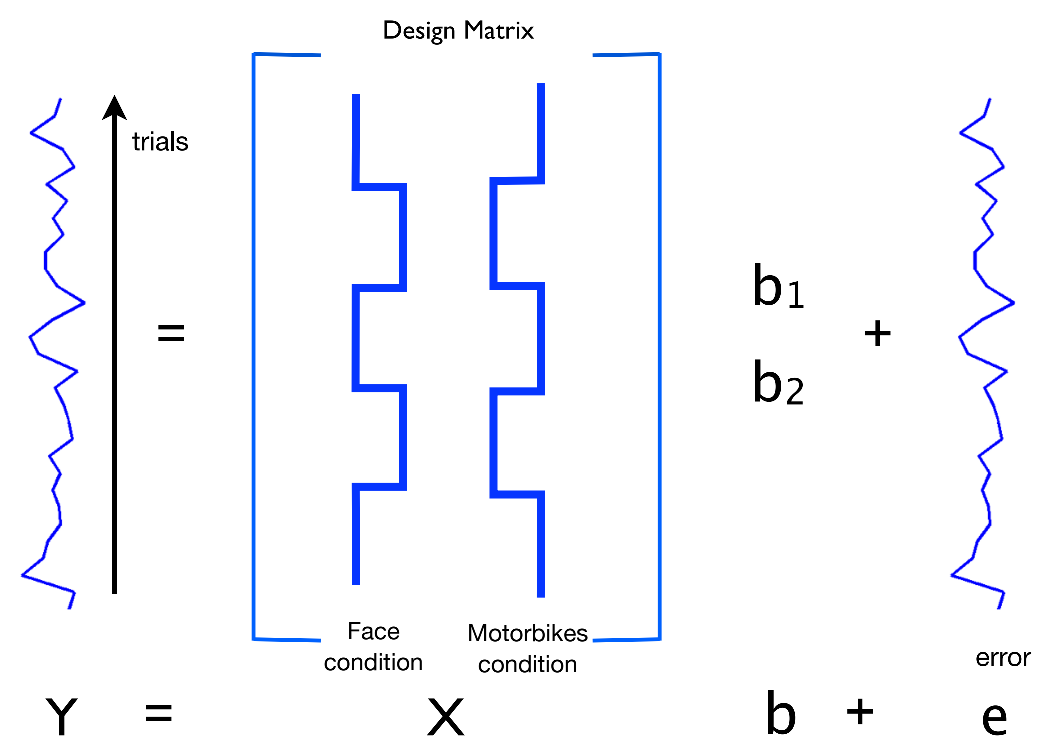
A COntrast of Parameter Estimates (COPE) is a linear combination of the regression parameter estimates, e.g.

Contrast [1 0] gives a COPE $= 1 x B_1 + 0 x B_2$

$= B_1$

Contrast [1 -1] gives a COPE $= 1 x B_1 - 1 x B_2$

$= B_1 - B_2$



Contrasts

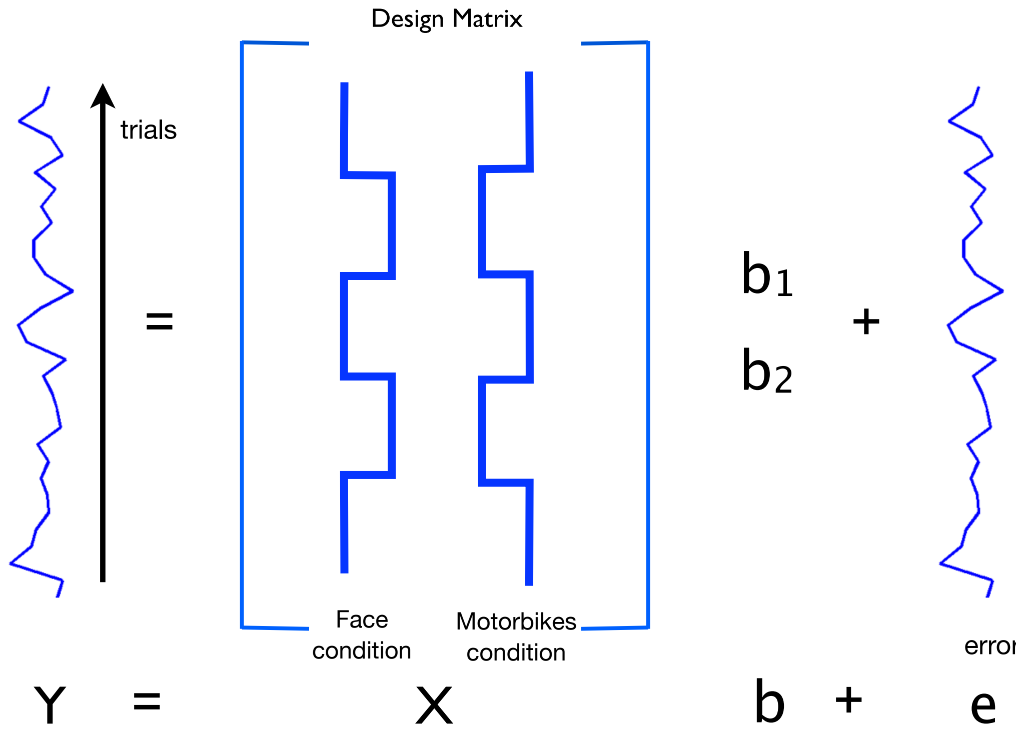
A COntrast of Parameter Estimates (COPE) is a linear combination of the regression parameter estimates, e.g.

Contrast [1 0] gives a COPE $=1xB_1+0xB_2$

$= B_1$

Contrast [1 -1] gives a COPE $=1xB_1-1xB_2$

$= B_1 - B_2$



Use a t-test to test the null hypothesis that COPE=0:

t-statistic:

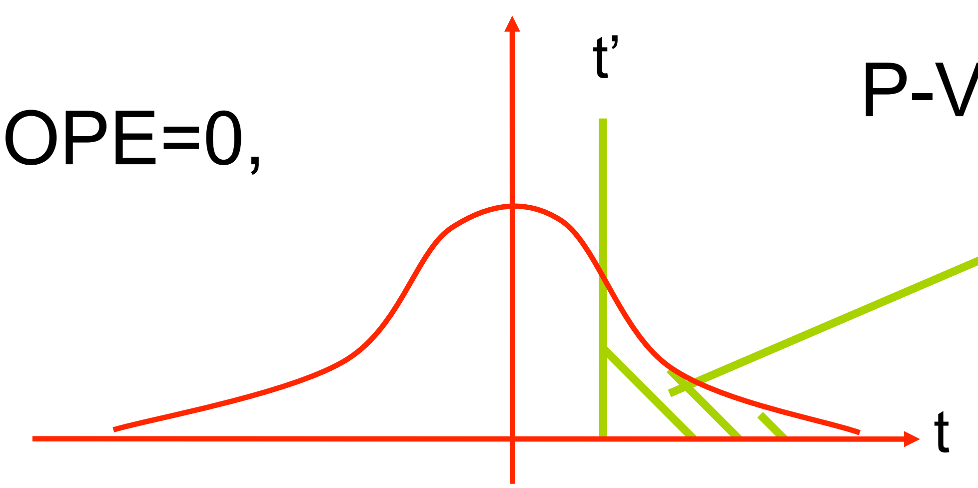
$$t = \frac{COPE}{std(COPE)}$$

Null Hypothesis Test

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

Under null hypothesis, COPE=0,

t' is from a t-distribution



P-Value=prob(t>t'|COPE=0)
(one-tailed test)

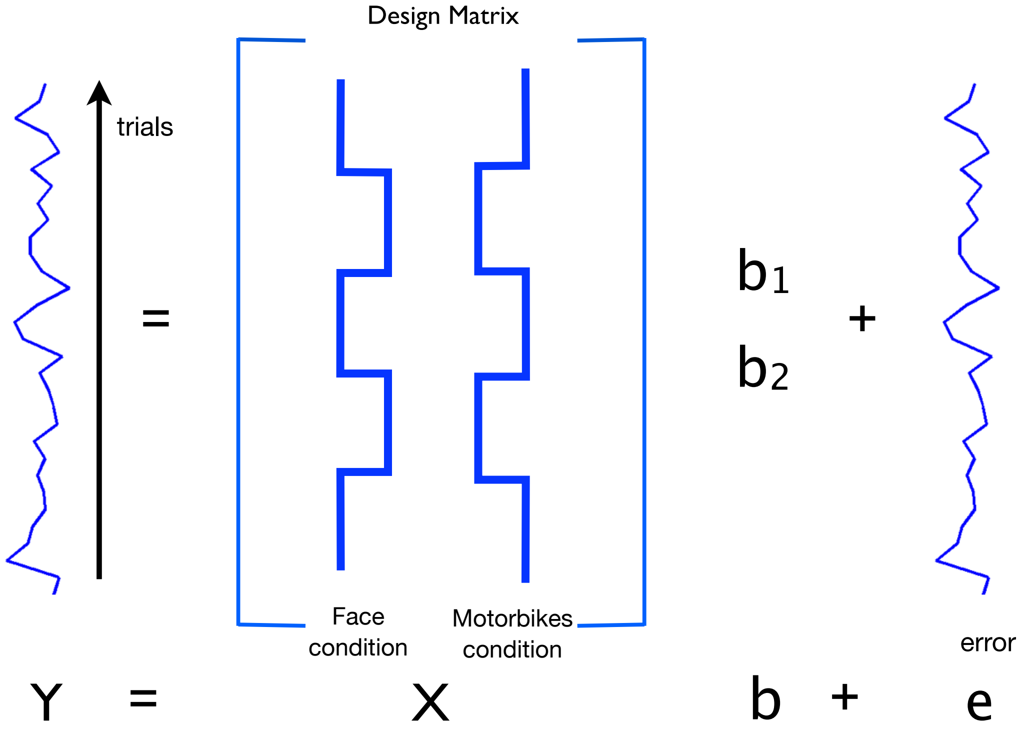
Small P-Value => null hypothesis unlikely
If P-Value < P-threshold then reject null hypothesis
P-threshold corresponds to False Positive Rate (FPR)

Contrasts

A COntrast of Parameter Estimates (COPE) is a linear combination of parameter estimates, e.g.

Contrast [0 1] gives a COPE $= 0 \times B_1 + 1 \times B_2$

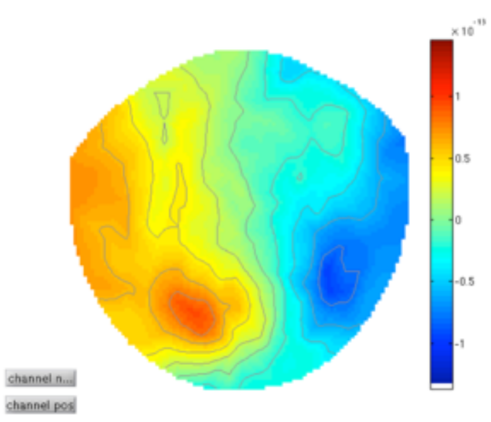
$= B_2$

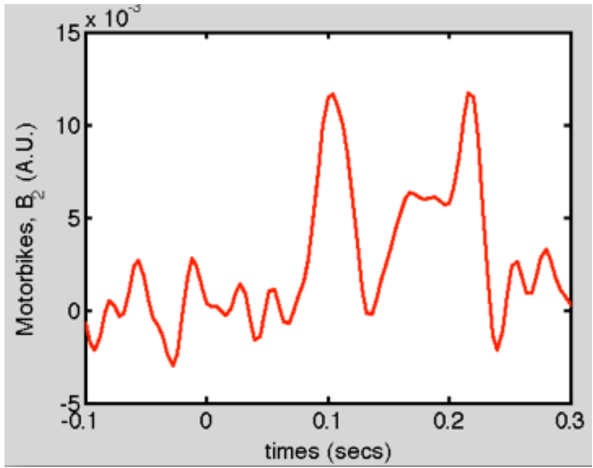


Test the null hypothesis that $B_2=0$

e.g. where in time and space is there significant positive activity in response to the motorbike condition?*

** as we are doing a one-tailed t-test*



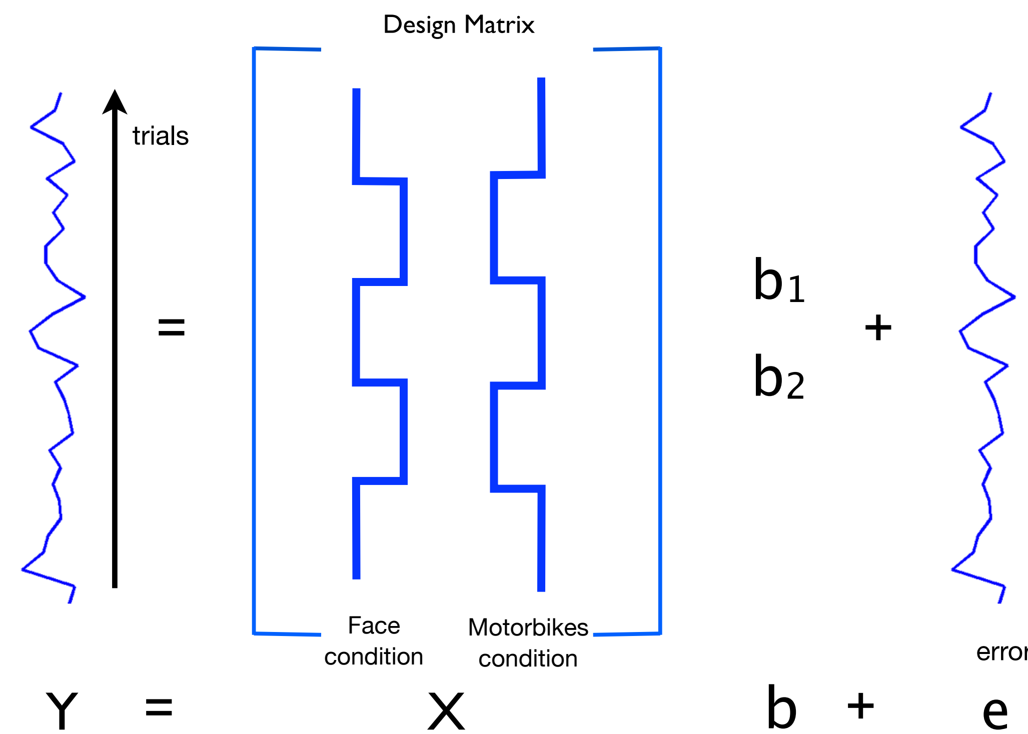


Contrasts

A COntrast of Parameter Estimates (COPE) is a linear combination of parameter estimates, e.g.

Contrast [1 -1] gives a COPE $= 1 \times B_1 - 1 \times B_2$

$= B_1 - B_2$



Test the null hypothesis that $B_1 - B_2 = 0$

e.g. where in time and space is there more activity in response to the faces than the motorbike condition?*

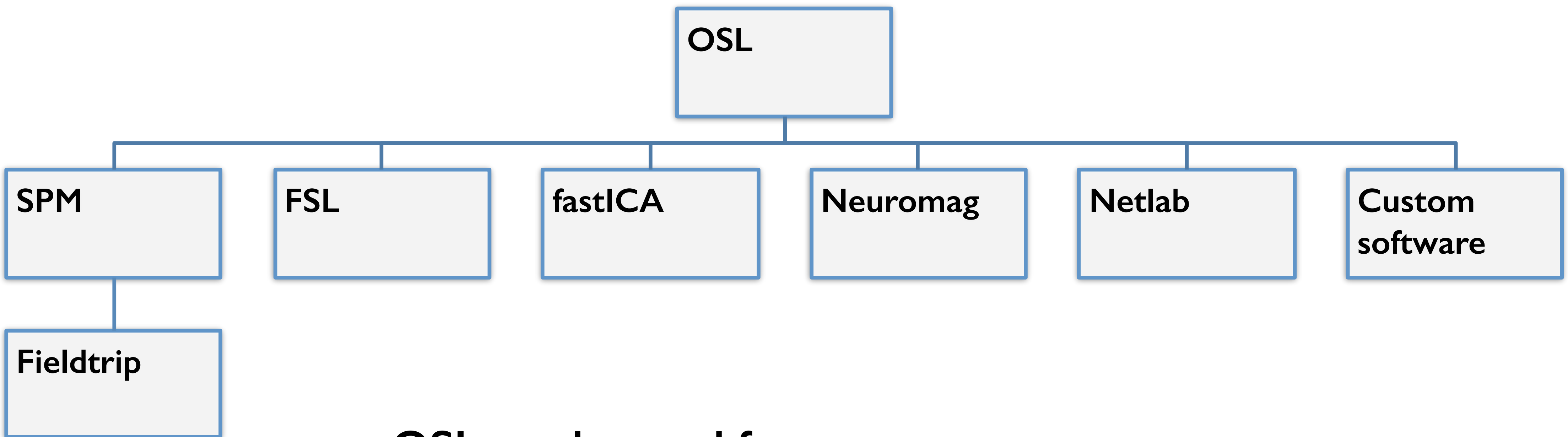
** as we are doing a one-tailed t-test*

Other stuff the GLM can do

- Linear trends between conditions
  - ➡ plus quadratic, higher order trends
- Factorial designs (interaction effects)
- F-tests (combined explanatory power over multiple contrasts)
- Subject-wise GLMs at the group level (e.g. patients vs controls)
- See the FSL course FEAT/FMRI Preprocessing and Model-Based slides at:

http://www.fmrib.ox.ac.uk/fslcourse

OHBA's Software Library



OSL can be used for

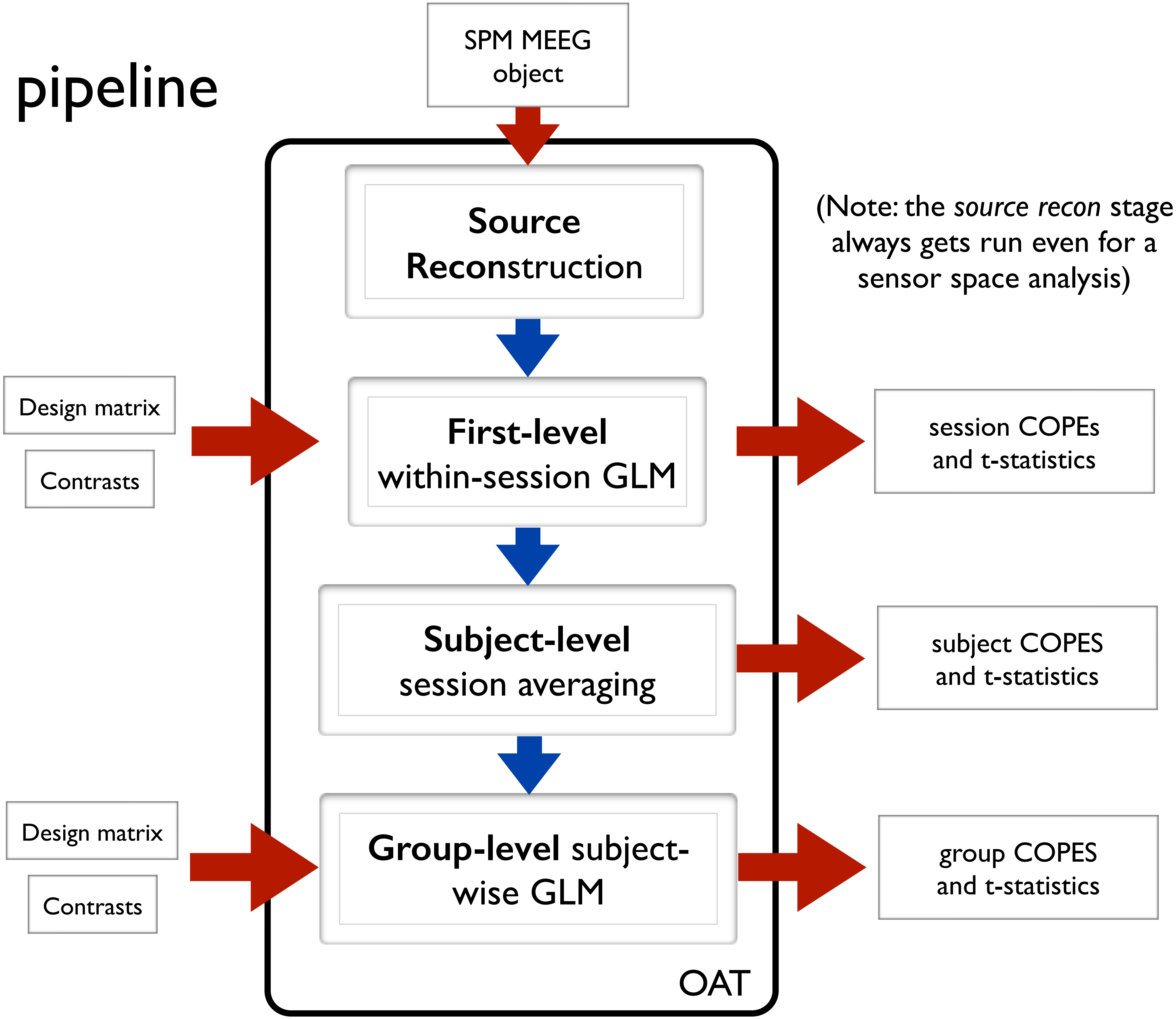
- task (GLM) and rest (e.g. ICA) analyses
- preprocessing
- sensor space analysis
- source space analysis
- statistics

OAT - OSL's (easy) Analysis Tool

- **Task-based** analysis in:
  - ➡ sensor space, or
  - ➡ source space (e.g. via beamforming)
- In:
  - ➡ in time-frequency domain (e.g. induced responses)
  - ➡ time domain (e.g. ERF-style), or
- First-level (within-subject) analysis, using:
  - ➡ trial-wise GLM on epoched data
  - ➡ time-wise GLM on continuous data
- Group-level (between-subject) subject-wise GLM analysis

OAT Pipeline Stages

- 4 distinct pipeline stages:

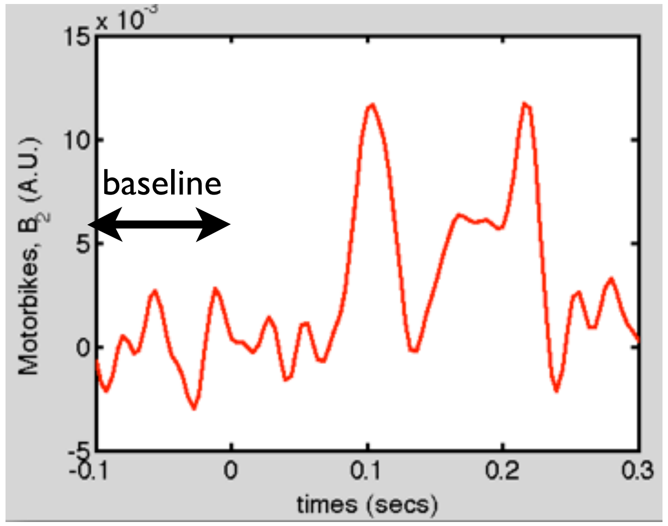


OAT Setup and Pipeline Stages

- Set some mandatory fields, and then use osl_check_oat call to setup an OAT struct:
  - ➡ *oat= osl_check_oat(oat);*
- 4 distinct stages to the pipeline, with corresponding settings:
  - ➡ *oat.source_recon, e.g.*
    - *oat.source_recon.method='beamform';* % for beamforming
    - *oat.source_recon.method='none';* % for a sensor space analysis
  - ➡ *oat.first_level* (GLM within-session analysis)
  - ➡ *oat.subject_level* (within-subject averaging)
  - ➡ *oat.group_level* (GLM subject-wise analysis)

Some oat.first_level settings

- baseline correction (BC), subtracts mean COPE in baseline time window (e.g. t<0s)



- *oat.first_level.bc* specifies whether BC should be done for each contrast
  - ➡ BC is computed on the COPE time course
  - ➡ no need to do BC on differential contrasts (e.g. [1 -1] contrast)
  - ➡ *recommended* to do BC on main effect contrasts (e.g. [1 0] contrast)

Some oat.first_level settings

- Set time range and freq range using:
  - *oat.first_level.time_range* (for TF analyses need to make this time range is smaller than *oat.source_recon.time_range* to remove edge effects)
  - *oat.first_level.tf_freq_range*

Some oat.first_level settings

- Set time range and freq range using:
  - *oat.first_level.time_range* (for TF analyses need to make this time range is smaller than *oat.source_recon.time_range* to remove edge effects)
  - *oat.first_level.tf_freq_range*
- To do an ERF analysis set *oat.first_level.tf_method='none'*
- To do a Hilbert Time-Frequency (TF) analysis set *oat.first_level.tf_method='hilbert'*
- To do a Morlet Wavelet Time-Frequency (TF) analysis set *oat.first_level.tf_method='morlet'*
- For TF analyses set the number of frequency bins using *oat.first_level.tf_num_freqs=10*

Running OAT

- Use osl_run_oat to run an OAT:
  - ➡ *oat=osl_run_oat(oat);*
- This runs the stages specified in ***oat.to_do***, e.g.:
  - ➡ *oat.to_do=[1 1 0 0];* only runs source_recon and first-level stages

OAT output

- Results are stored in the directory specified in *oat.source_recon.dirname*, with a '*.oat*' suffix
- This directory includes a **.mat file*, can load this into Matlab with: *oat=osl_load_oat(oat);*

OAT output

- Results are stored in the directory specified in *oat.source_recon.dirname*, with a *'.oat'* suffix

- This directory includes a **.mat file*, can load this into Matlab with: *oat=osl_load_oat(oat);*

- This loads in a struct containing the settings used and the filenames of the outputs for each stage of the pipeline:

  - ➡ *oat.source_recon.results_fnames*
  - ➡ *oat.first_level.results_fnames*
  - ➡ *oat.subject_level.results_fnames*
  - ➡ *oat.group_level.results_fnames*

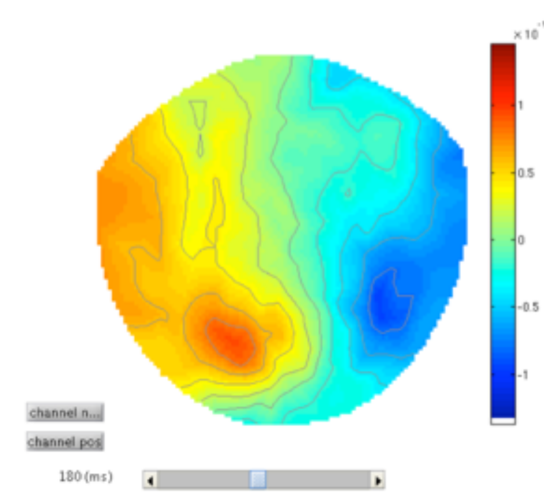
- *These can be loaded into Matlab, e.g. to load session 2's first level results use the call:*

  - ➡ *res=osl_load_oat_results(oat, oat.first_level.results_fnames{2})*

Viewing OAT output

- It is highly recommended that you inspect *oat.results.report* (an HTML page), to ensure that OAT has run successfully (See the practical)

- In sensor space, use:
  - Use *osl_stats_multiplotER* and *osl_stats_multiplotTFR* to call Fieldtrip interactive topoplots
  - The two orientations of the *MEGPLANARs* are combined (in the first_level stage) by rectifying and adding



Practical

1) Sensor space trial-wise GLM using OAT on **epoched** data:

a) Time-domain (ERF) analysis

b) Time-frequency (induced response) analysis

2) Sensor space time-wise GLM using OAT on **continuous** data.

Finally

- If you end up using OSL for your own data analysis then please get in touch with me
- Look at and use the OSL Wiki!
    - https://sites.google.com/site/ohbaosl/
    - sign up for OSL email list