MRC Cognition and Brain Sciences Unit
UNIVERSITY OF CAMBRIDGE

fMRI
Statistical analysis

Dace Apšvalka
@dcdace dcdace.net

•Feb, 2022

The Plan

- fMRI files and data
- Pre-processing
- Statistical analysis
- Recap

Hands-on materials
https://github.com/dcdace/fMRI_training

Hypothesis

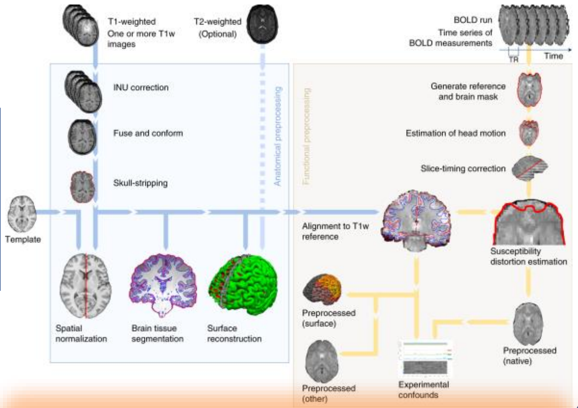
Design an experiment

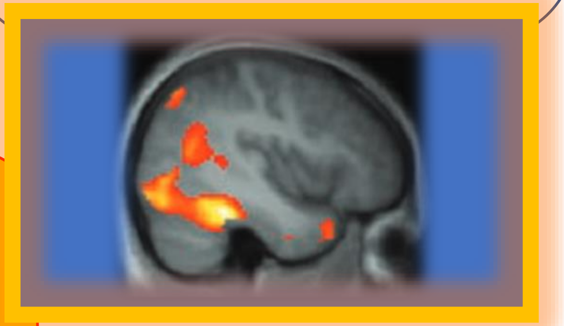
Stimuli
Timing

Collect the data

Anatomical image
Functional images
Event details

Pre-process & Analyse





The final push

fMRI Aanalysis

The General Linear Model (GLM)

March 12, 2019, Memory Control Lab, CBU

Dace Apšvalka

Largely based on Idan Blank's materials

https://cbmm.mit.edu/videos?field_video_grouping_tid%5B%5D=770

What do we want to find out?

Which brain regions are engaged when people try to **STOP IT!!!**

https://www.youtube.com/watch?v=Ow0lr63y4Mw

Finding No-Think regions in the brain

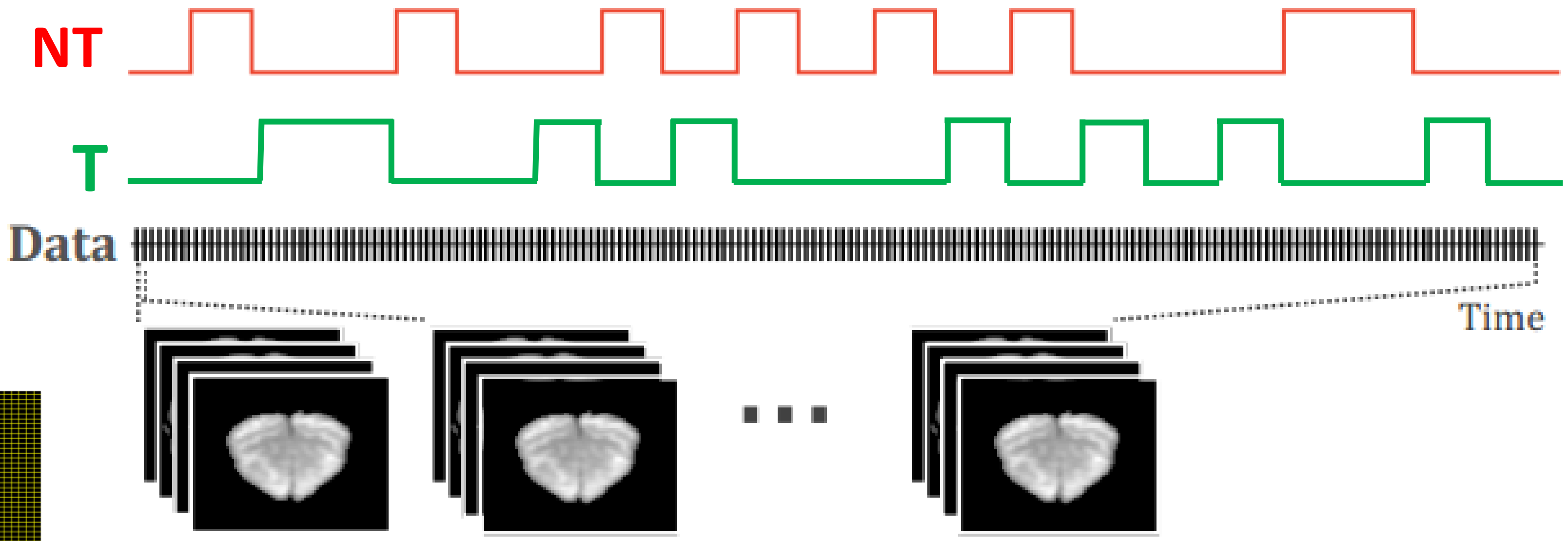
Which brain regions are engaged when people try to stop memory retrieval

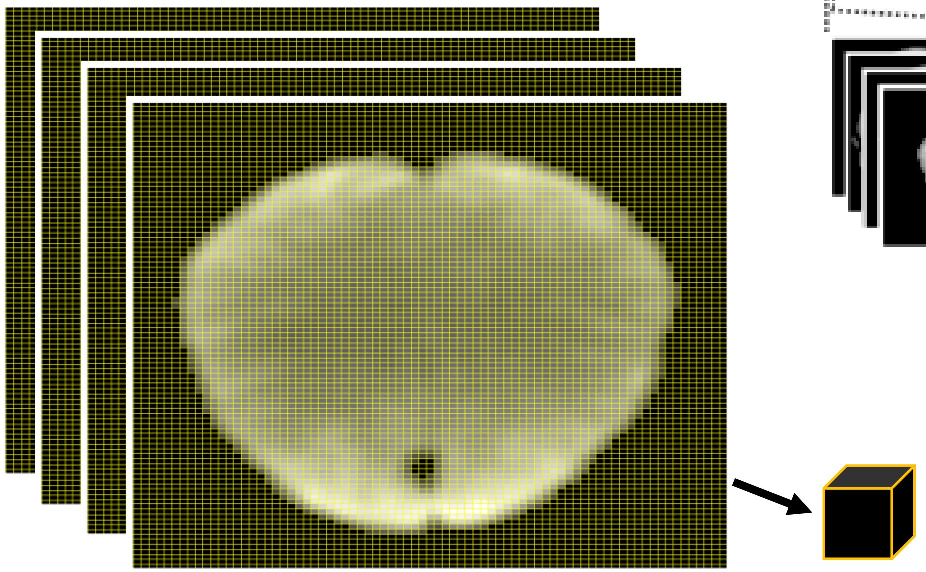
- With fMRI the meaningful questions are questions that **compare two conditions**
  - We need some sort of control condition –> Think condition

- **Which brain regions respond more to stopping memories than to retrieving memories**
  - The control question hopefully help to **wash out all the regions we are not interested in**. Because regions that we are NOT interested in should activate both conditions to the same extend (e.g. visual areas)

The structure of our data

- **Question:** Which brain regions respond more to stopping memories than to retrieving memories
- **Experiment**





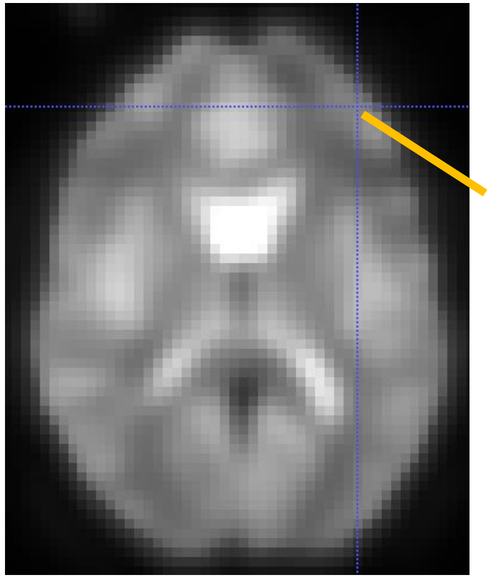
The brain data that we collect are a stack of 2D slices that form a 3D volume -> a large 3D cube with it's smallest unit –> a small 3D cube -> **a voxel**

The structure of our data

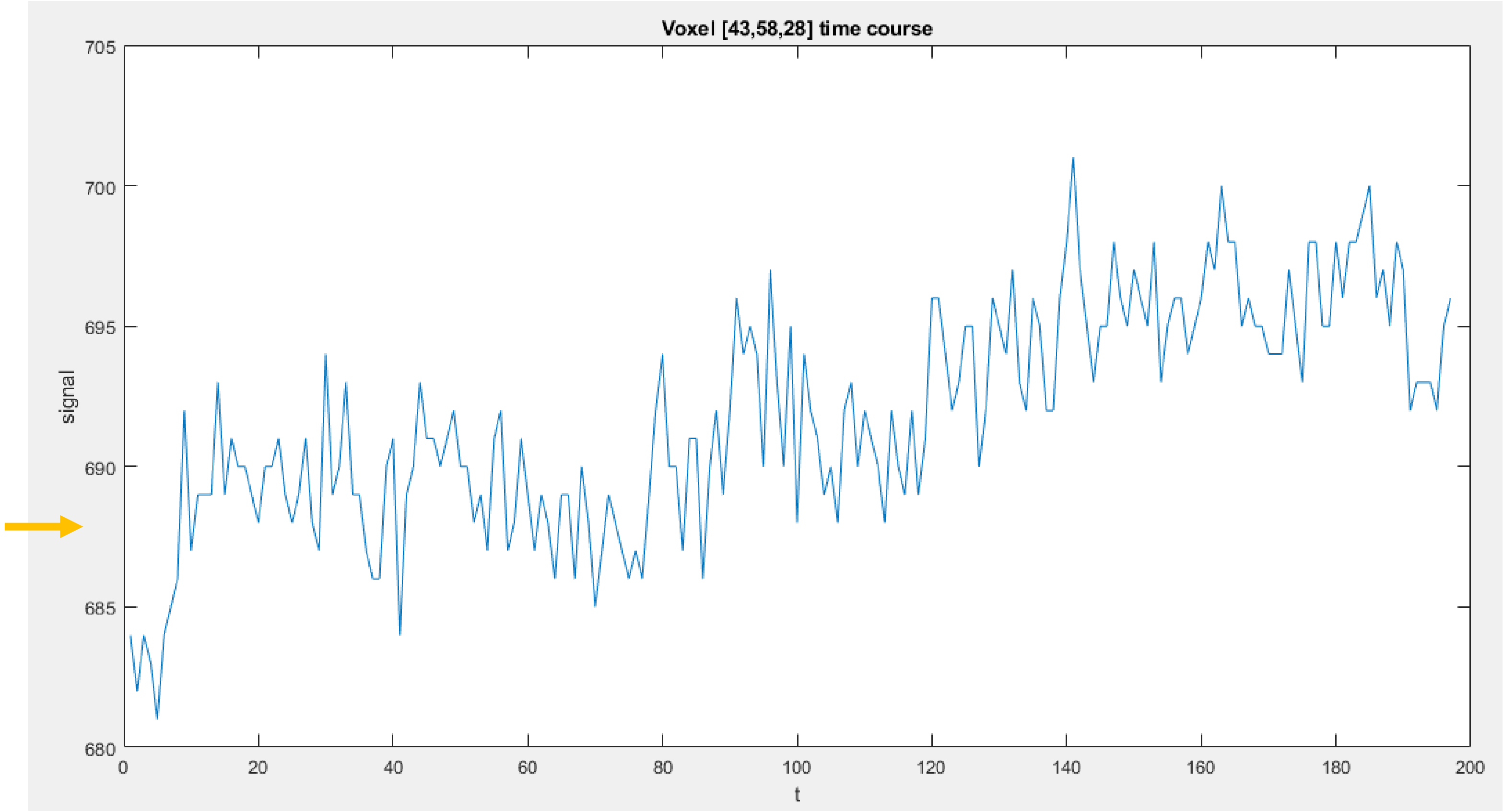
MATLAB script

```
img_name    = fullfile(root_dir, 's01_run4_swar4D.nii');
img_struct  = spm_vol(img_name); % structure 197 x 1
img_data    = spm_read_vols(img_struct); % 4D matrix
dim         = size(img_data);
disp(dim)
    61    73    61   197
y1 = squeeze(img_data(43,58,28,:)); % x y z t
% alternative function, returning the same results
% y1 = spm_get_data(img_struct, [43;58;28]); % x y z t

plot(y1)
title('Voxel [43,58,28] time course')
xlabel('t')
ylabel('signal')
```

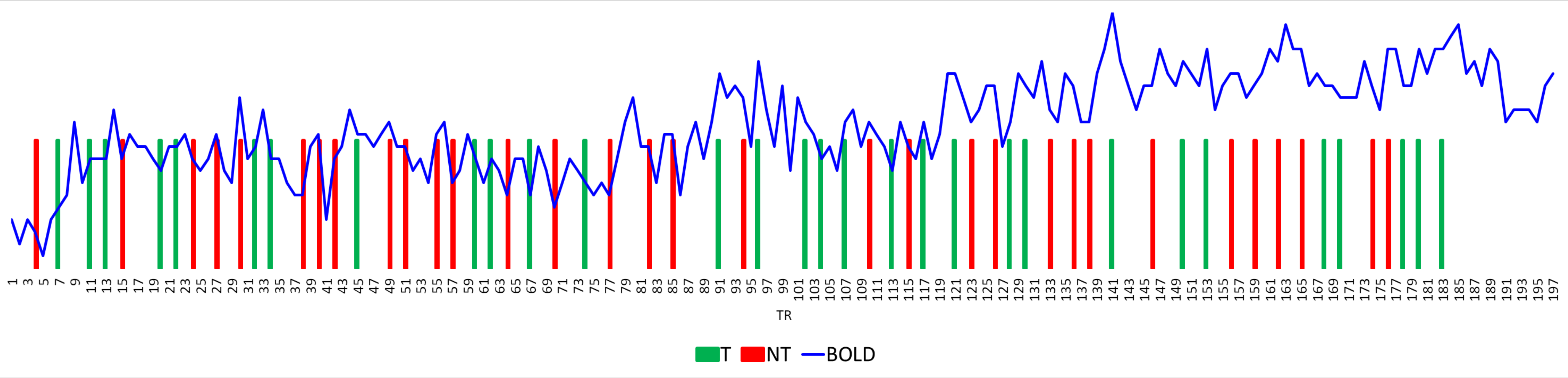


Time course of the voxel located at 43, 58, 28 (these are not MNI coordinates!)

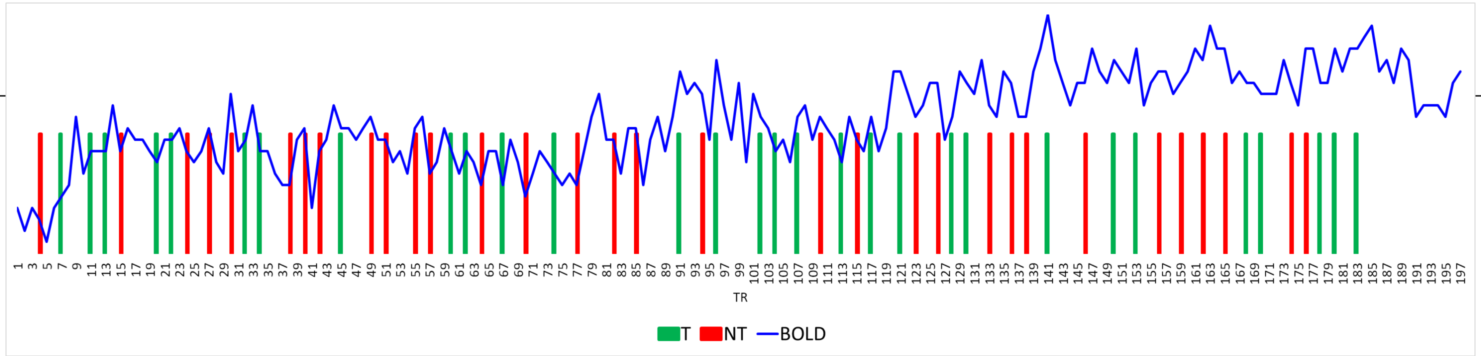


Question: Which brain regions respond more to stopping memories than to retrieving memories

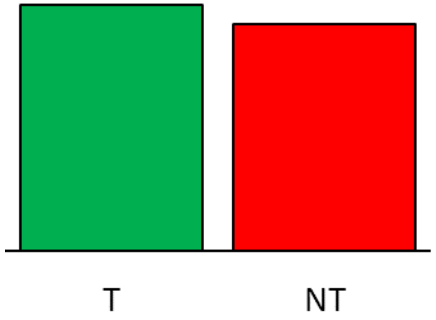
- Event onsets and BOLD signal (that we extracted using the code in the previous slide).
- **How do we analyse this?**



Question: Which brain regions respond more to stopping memories than to retrieving memories



- A raw BOLD signal is noisy
- Let's start with an intuitive way, and that's how people started to analyse fMRI data
- **Analysis:** an intuitive approach
  1. For each voxel, look at its signal time-series (activity across time)
  2. Average the signal across volumes that were collected while participants were trying to **Not-Think**
     - Mean NT = 691.92, Mean T = 692.35, NT-T = -0.43



BOLD signal = task-related activity changes + noise (other changes)

explained variation   unexplained variation

Question: Which brain regions respond more to stopping memories than to retrieving memories

- **Analysis:** a somewhat better approach

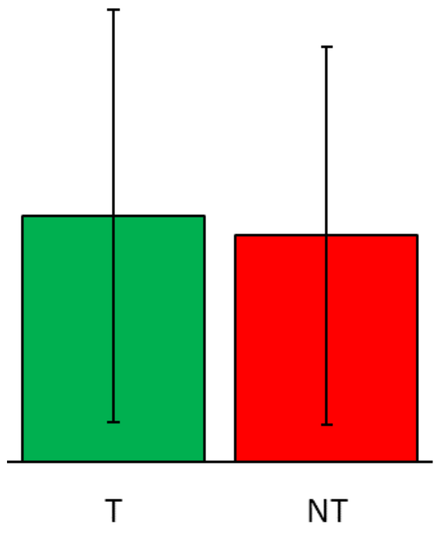
  - The difference in response to No-Think and Think might be due to chance (not real)

  - To test how likely the differences is to be real, we **take noise into account**

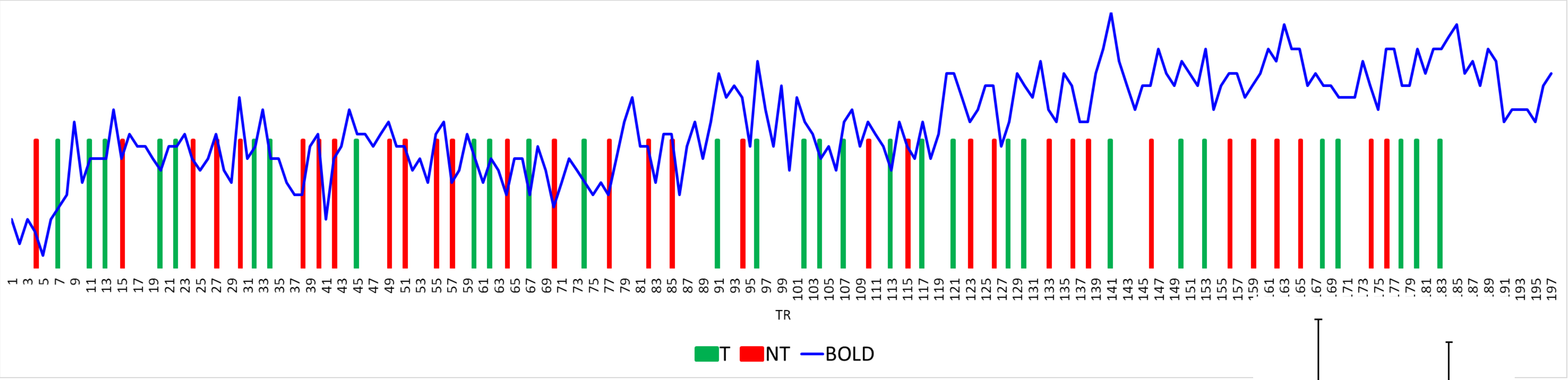
  - The simplest way to do this is to compute a t-value

$$t = \frac{\text{explained variation}}{\text{unexplained variation}} = \frac{\text{Mean}(\textbf{NT}) - \text{Mean}(\textbf{T})}{\text{noise}}$$

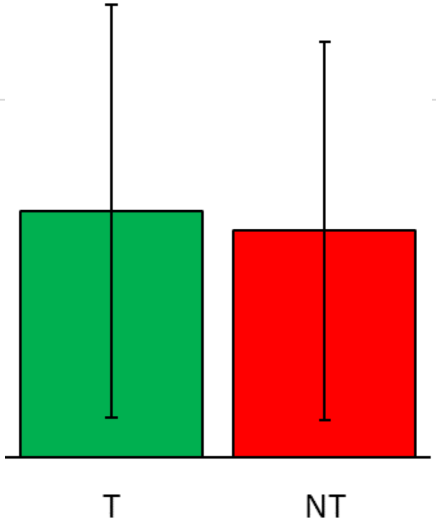
  - **If there is a lot of unexplained variation within conditions, the difference is likely to be due to chance**



Question: Which brain regions respond more to stopping memories than to retrieving memories

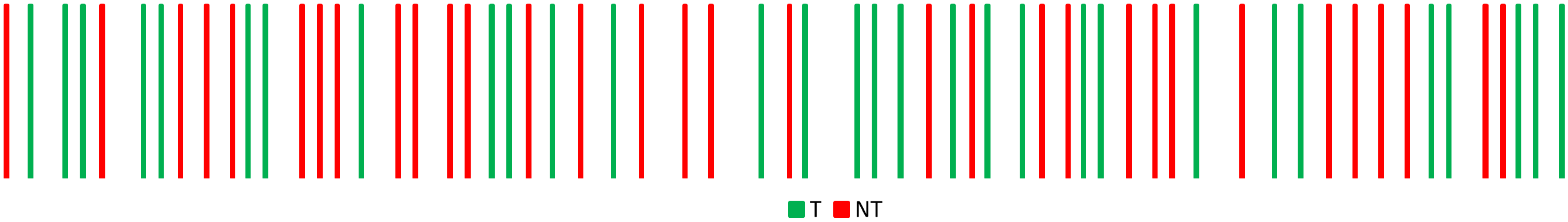


Why does the averaging method does not work?



Question: Which brain regions respond more to stopping memories than to retrieving memories

- **Stimuli**



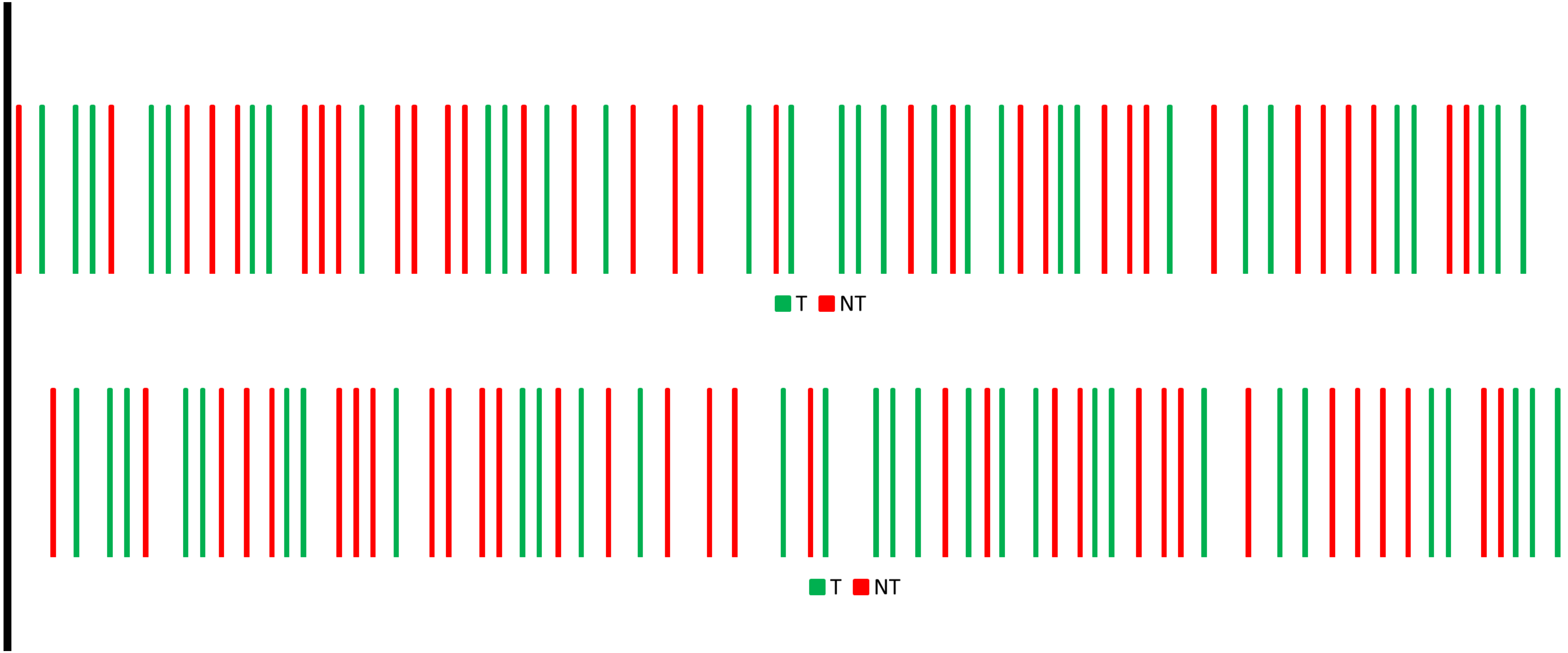
- **Neural activity**
  - Starts briefly after the onset of the stimulus
- **BOLD signal** (not a direct measure of neural activity)

  - When neurons are active, the fMRI BOLD signal will rise ______
    A. Immediately
    B. 1-3 seconds after the neural activity
    C. 6-12 seconds after the neural activity
    D. 25-30 seconds after the neural activity

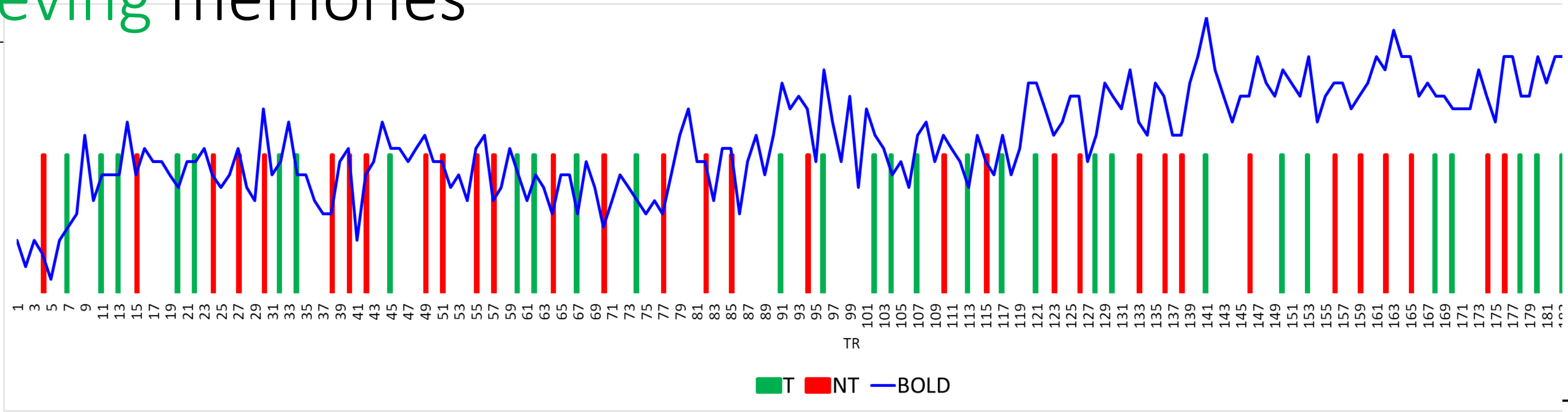
We were averaging the wrong points! The points in time when the cognitive control system would respond are actually little later than the onset of the stimulus; about 6-12 s later then the onset.

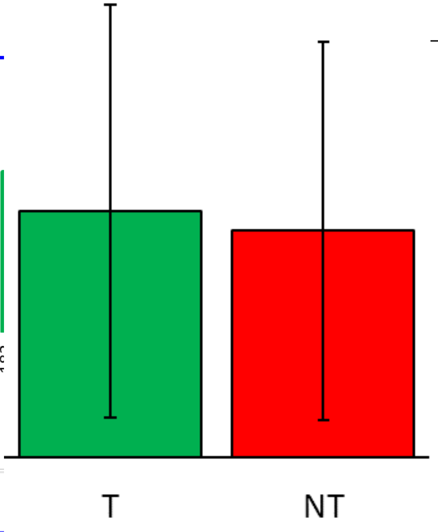
Question: Which brain regions respond more to stopping memories than to retrieving memories

The points that we are looking at should be shifted in time

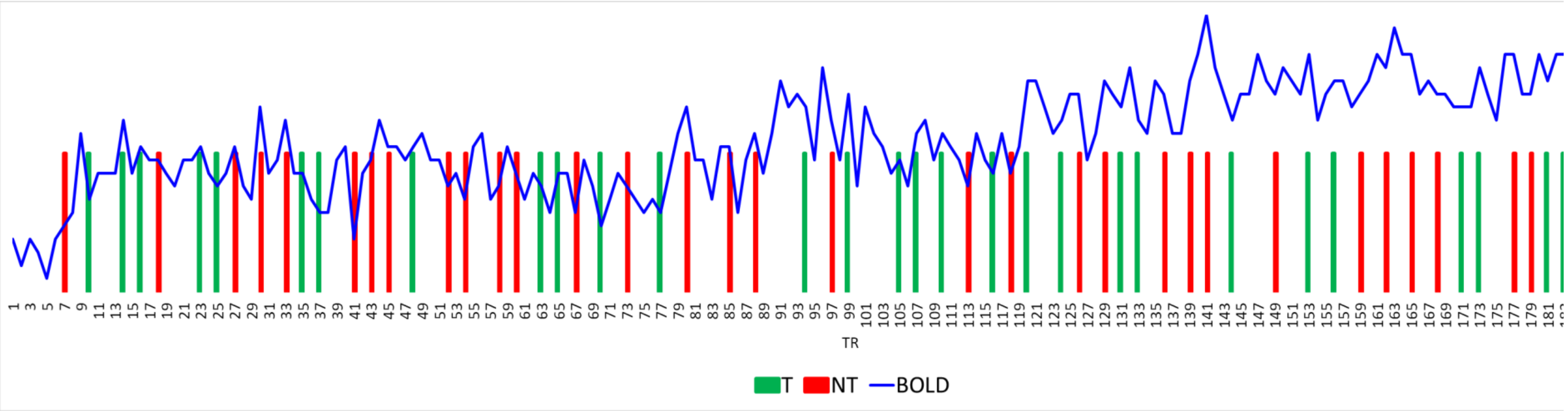


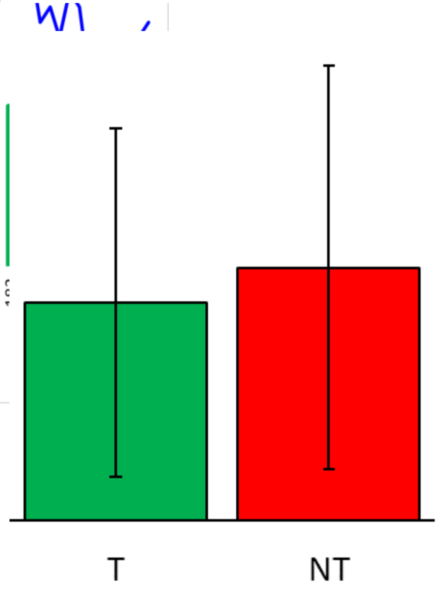
Question: Which brain regions respond more to stopping memories than to retrieving memories





+6s shift



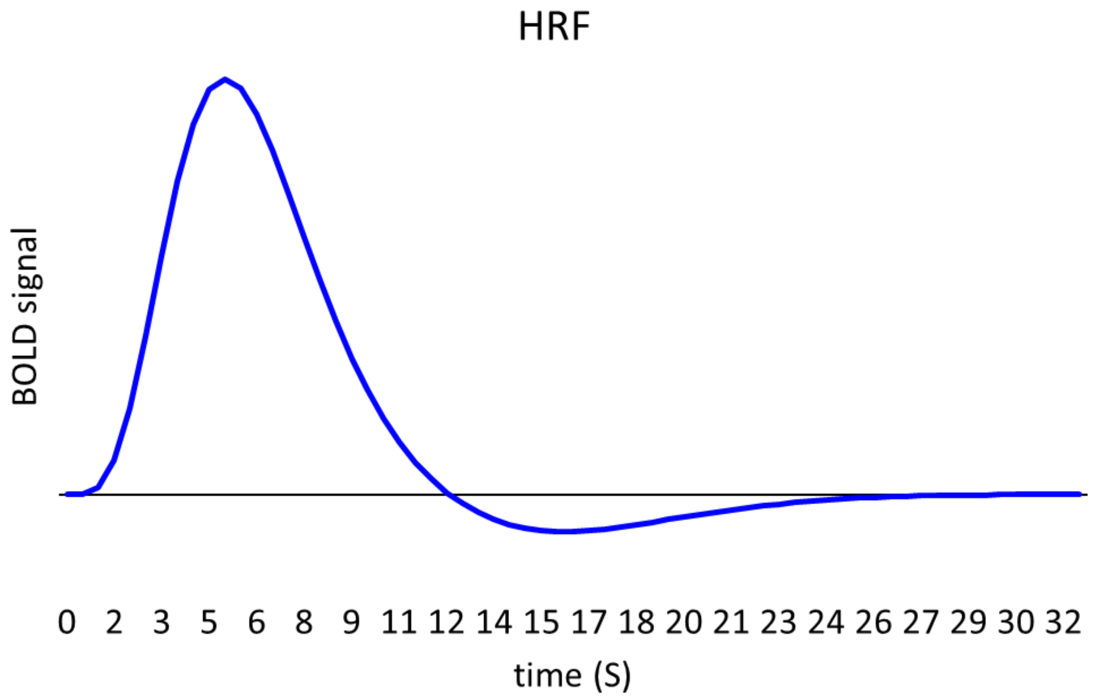


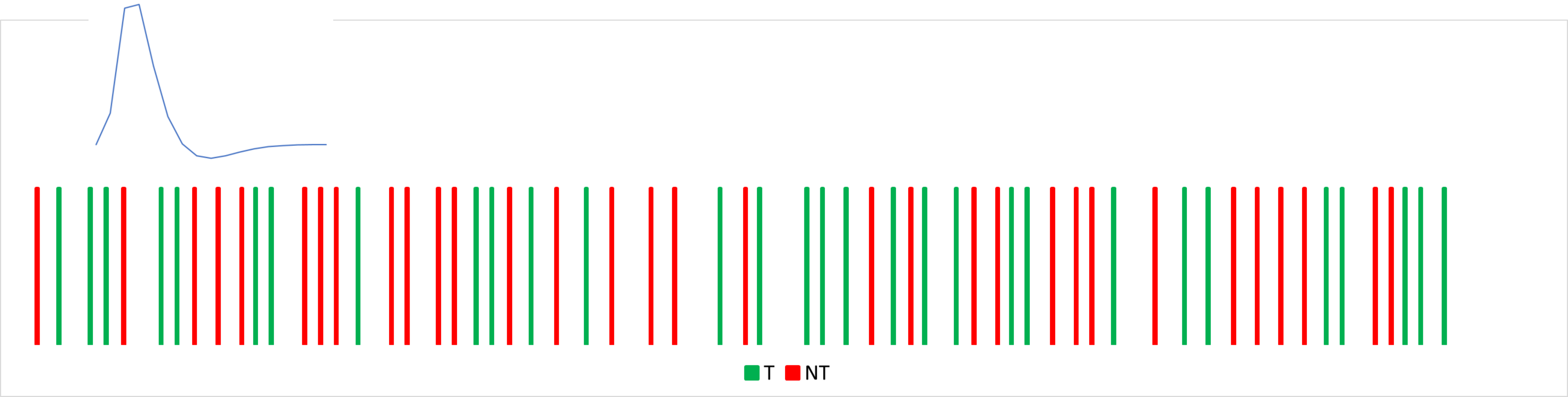
Looks slightly better, but still not working!

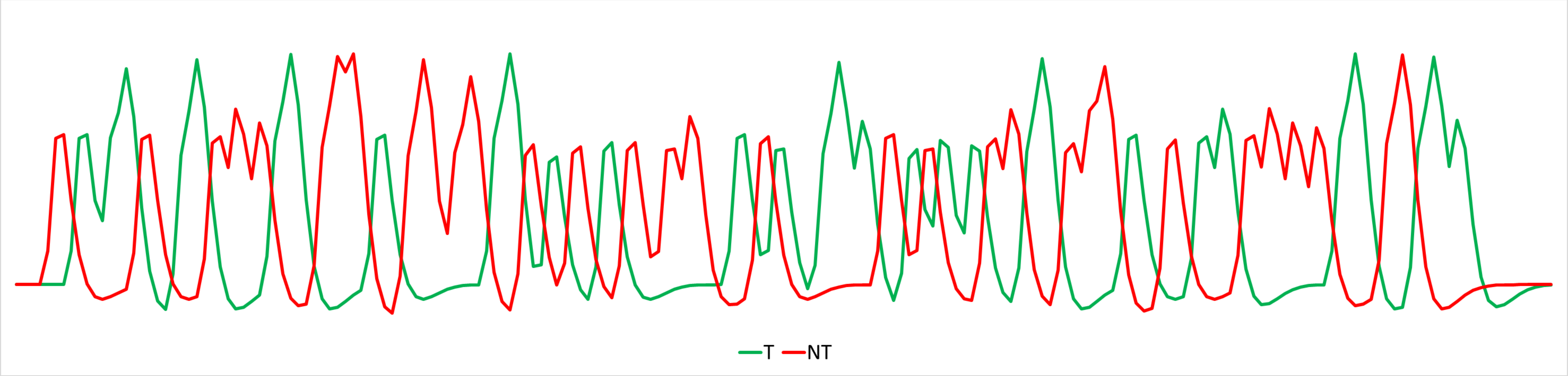
Bold signal does not change from 0 to 1 (on or off)!

Question: Which brain regions respond more to stopping memories than to retrieving memories

- The haemodynamic response function looks like this
- It is not binary, but smooth
- We need to make our prediction to look more like this
- How do we do that? This is done by mathematical process called **convolution**
- It is filtering the signal to look more like this shape

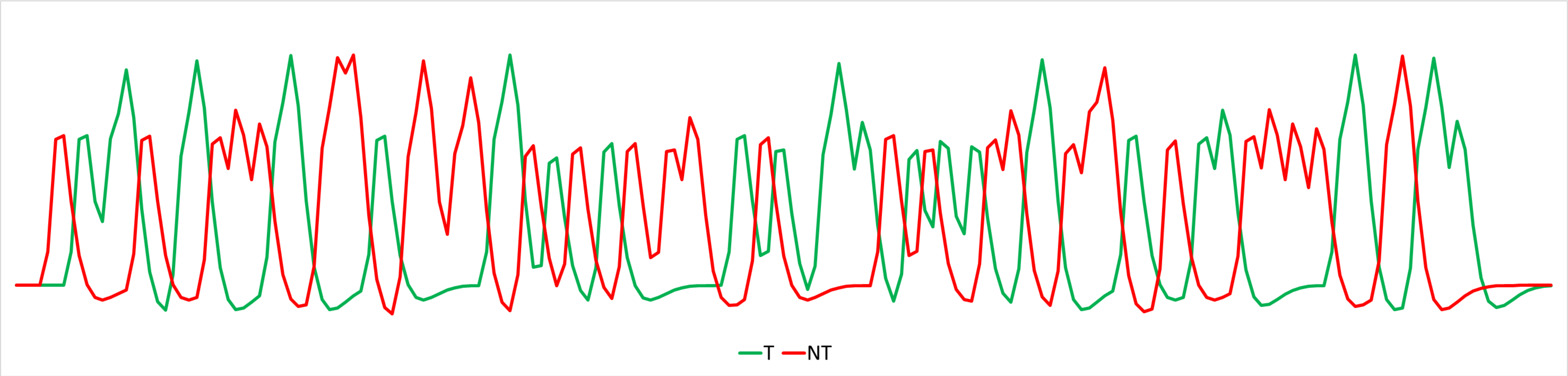




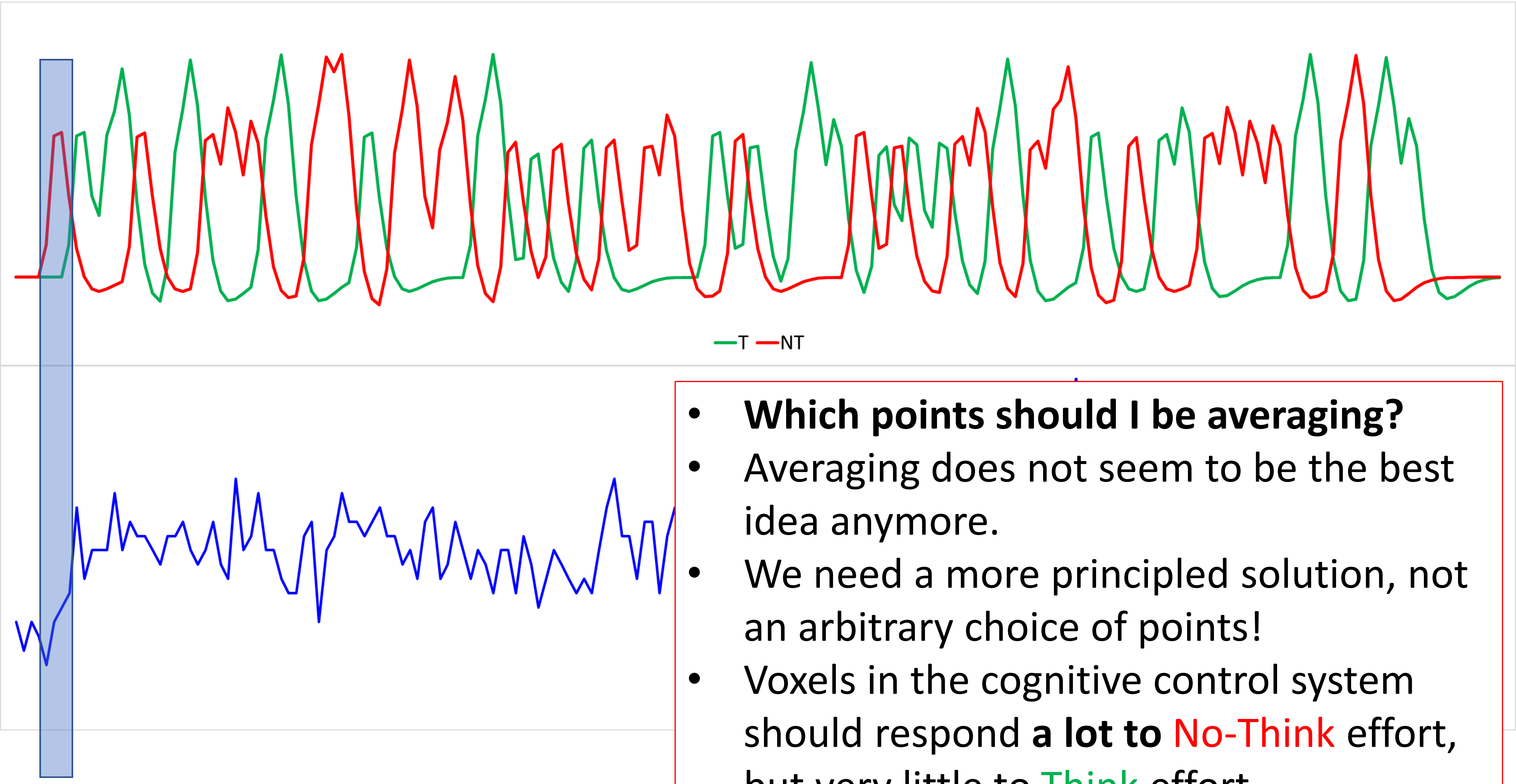


These are our BOLD predictions. **This is how we should expect our signal to look like!**

Question: Which brain regions respond more to stopping memories than to retrieving memories

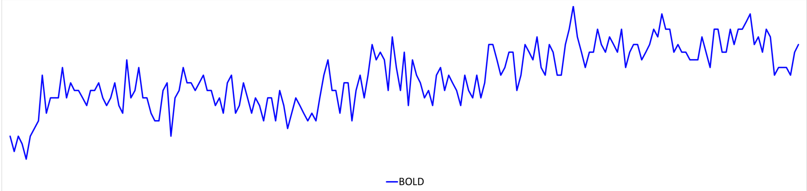


The time series of a voxel that responds more to No-Think than to Think should look like a combination of the two signals above, with No-Think having more weight than Think.



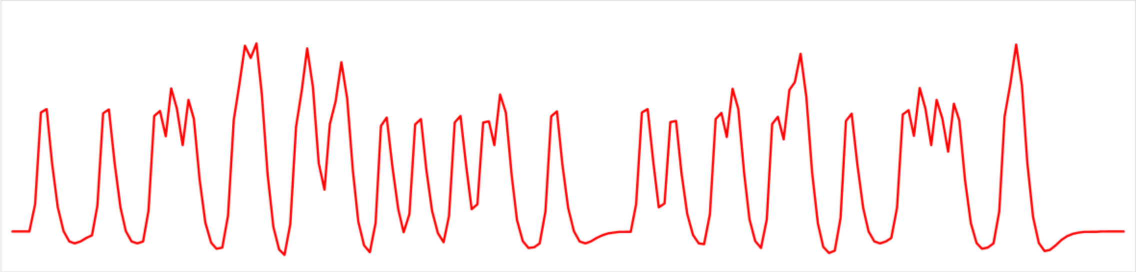
- **Which points should I be averaging?**
- Averaging does not seem to be the best idea anymore.
- We need a more principled solution, not an arbitrary choice of points!
- Voxels in the cognitive control system should respond **a lot to** No-Think effort, but very little to Think effort.

(de)Constructing the BOLD signal

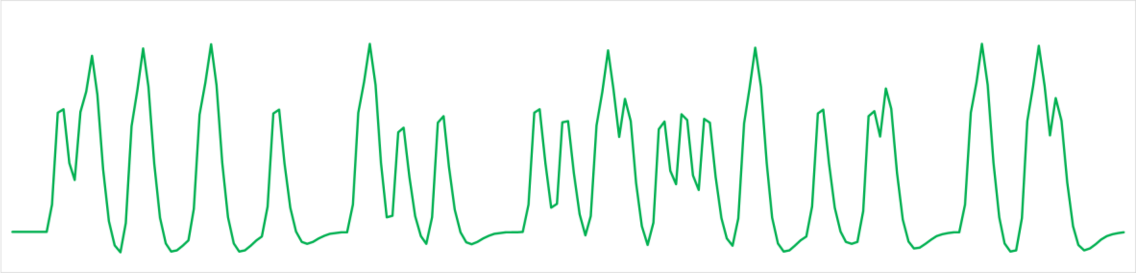


- We can approximate the signal time-series of a voxel by **combining these 3 signals**:

  - Baseline signal:
    - an average activity in that voxel when you are doing nothing
  - Response to No-Think:



  - Response to Think:

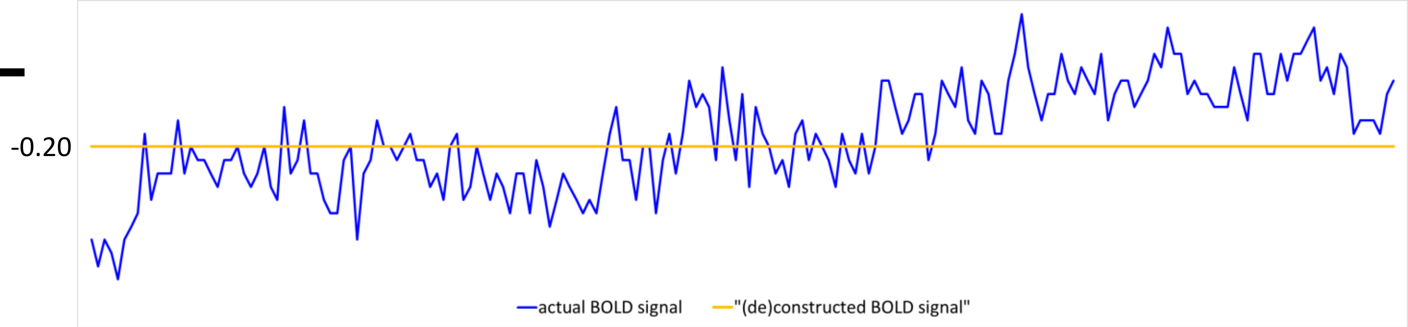


(de)Constructing the BOLD signal

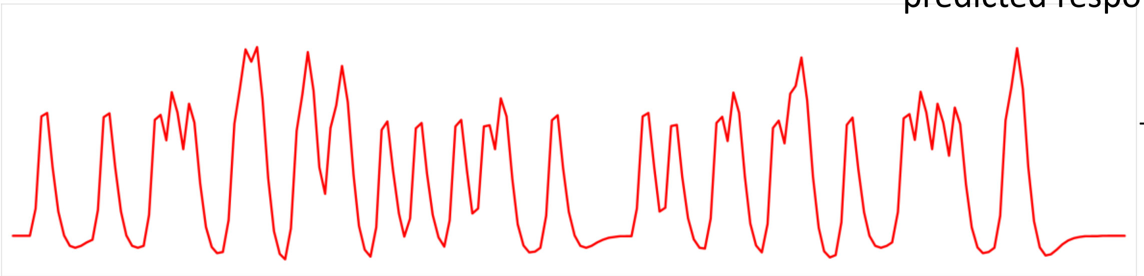
The true BOLD signal
Deconstructed BOLD signal

Baseline * -0.20

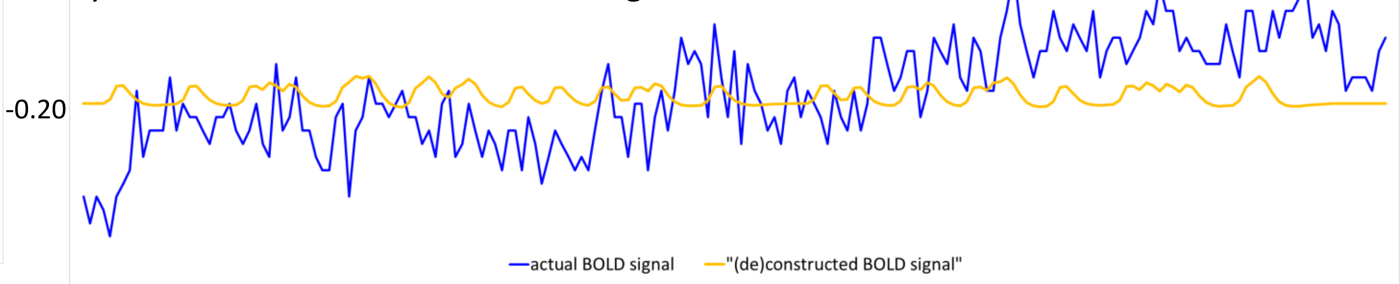
1. Let’s say the average signal at baseline was -0.2. We add this baseline measure to our deconstructed BOLD signal and see how it compares with the actual BOLD signal.



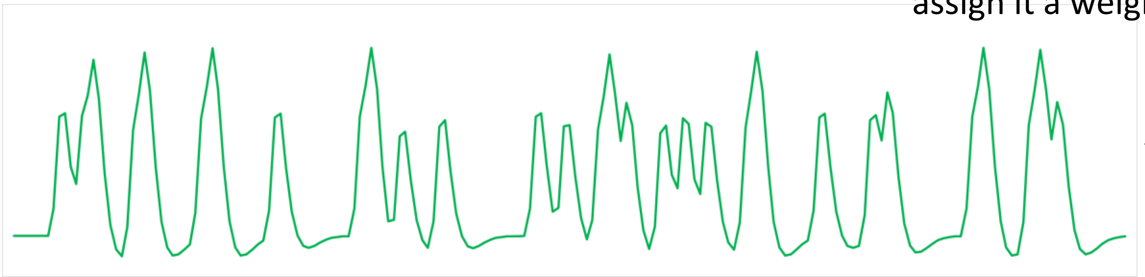
+ **No-Think** * 0.89



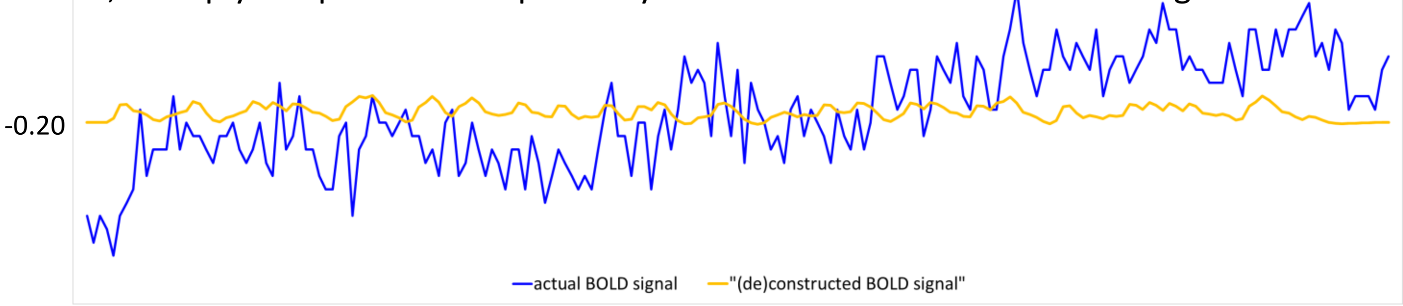
2. We predict that in this voxel No-Think condition contributes quite a lot to the true BOLD signal. Let’s assign it a weight 0.89 (we will find out later how exactly we find these weights), multiply the predicted response by it and add to the deconstructed signal.



+ **Think** * 0.36



3. We predict that in this voxel Think condition has little contribution to the true BOLD signal. Let’s assign it a weight 0.36, multiply the predicted response by it and add to the deconstructed signal.

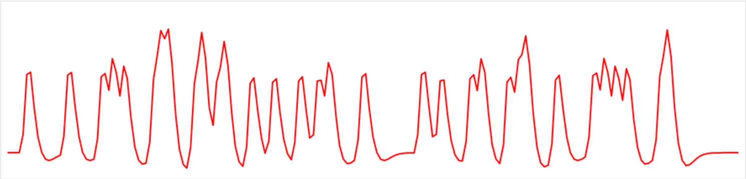


(de)Constructing the BOLD signal

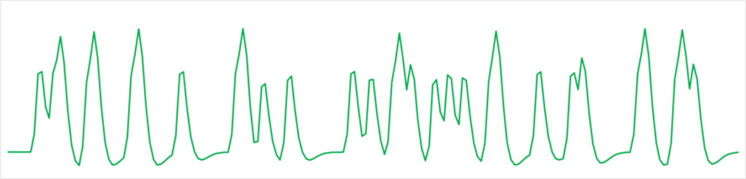
The true BOLD signal
Deconstructed BOLD signal

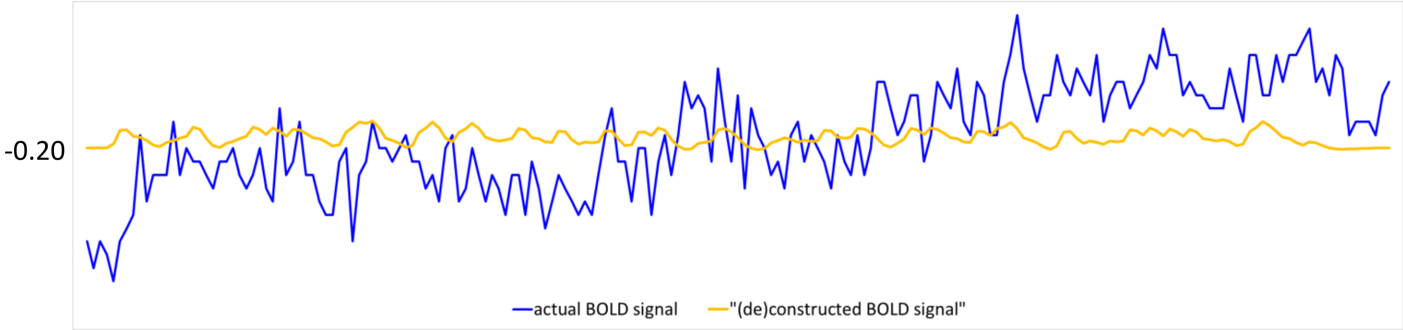
Baseline * -0.20

+ **No-Think** * 0.89



+ **Think** * 0.36

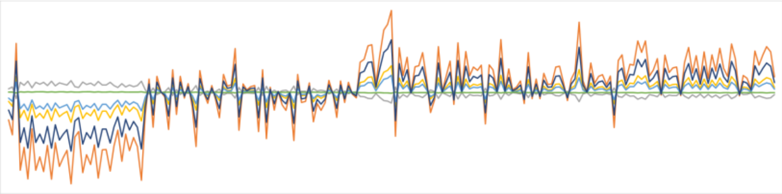




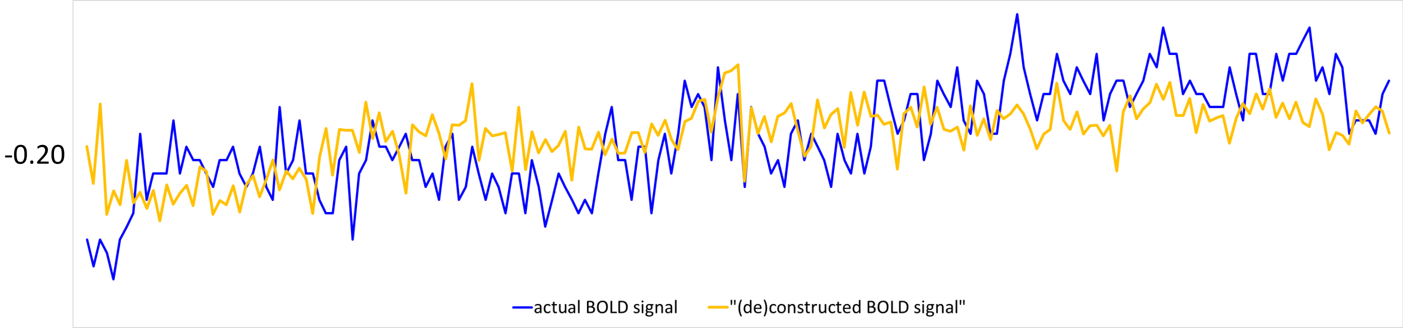
Let's assign some weights and add the **6 movement parameters** (mp) to the deconstructed signal.

+

Participant's movement parameters

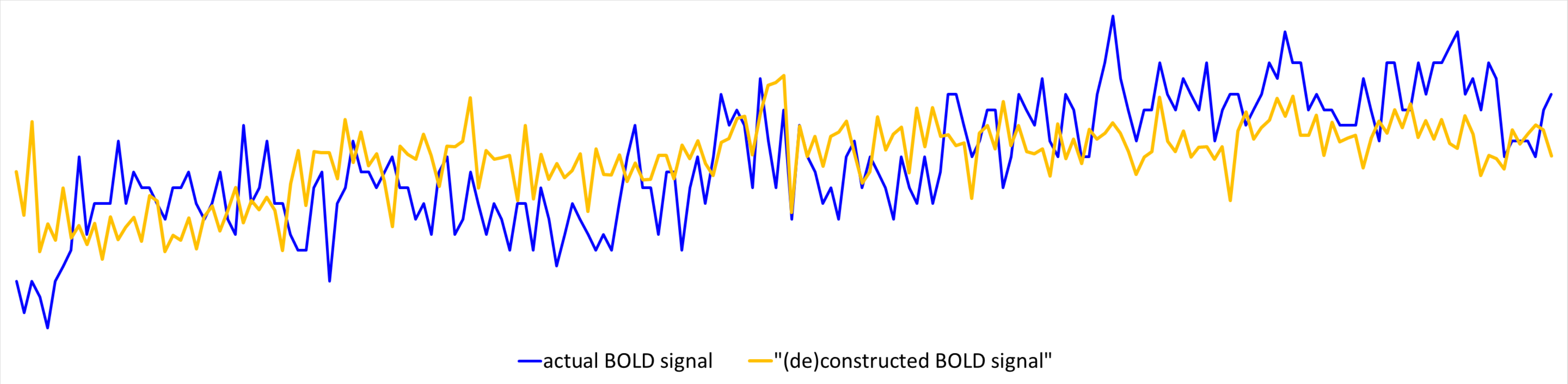


mp1*-0.30 +
mp2*-0.13 +
mp3* 0.33 +
mp4* 0.21 +
mp5*-0.01 +
mp6* 0.64 +



The deconstructed signal matches the true signal much better now!

(de)Constructing the BOLD signal



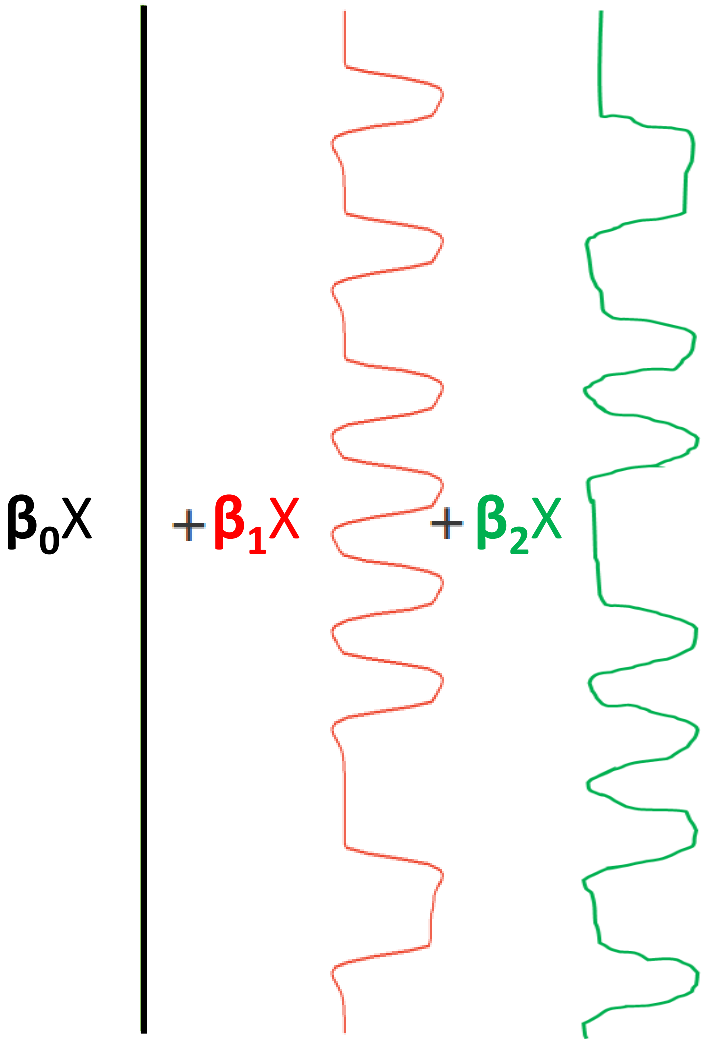
- These signals that we use to construct the approximation are called **predictors**
- One predictor predicts a **constant response**, just **baseline**
- Another predictor predicts how you should **respond to the task**

- Each predictor is associated with a weight called a **beta-weight**

- To create a **linear combination** of predictors, which **approximates a true signal**, we multiply each predictor by its beta-weight and then sum the results.

(de)Constructing the BOLD signal

These signals are just columns (arrays) of numbers stored in a matrix.

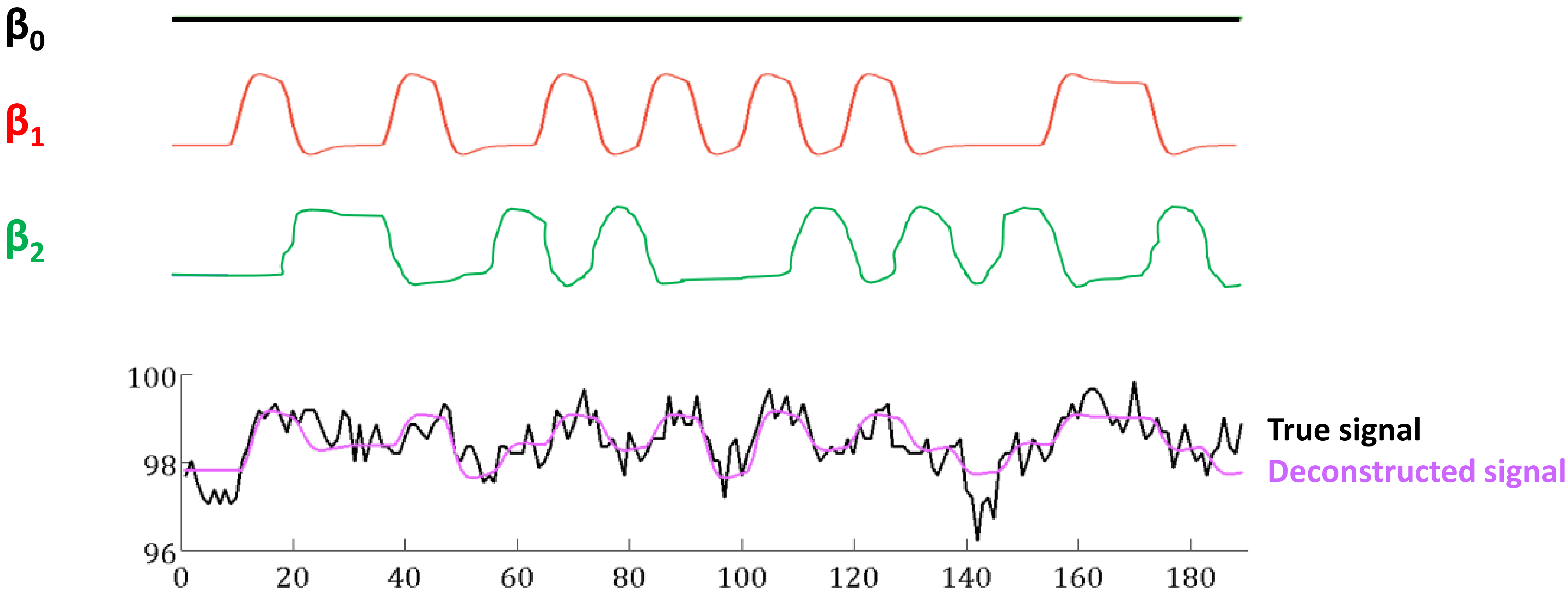


$= \beta_0 X + \beta_1 X + \beta_2 X =$

1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0.0866	0
1	0.4615	0
1	0.8464	0
1	1.0625	0
1	1.0528	0
1	0.6795	0
1	0.2640	0.0866
1	0.0106	0.4615
1	-0.0971	0.8464
1	-0.1193	1.0625
1	-0.1003	1.1394
1	-0.0688	1.0544
1	0.0460	0.6489
1	0.4403	0.2267
1	0.8365	-0.0203
1	1.0582	-0.1176
1	1.0512	-0.1309
1	0.6790	-0.0196
1	0.2639	0.3900
1	0.0106	0.8047
1	-0.0971	1.0409

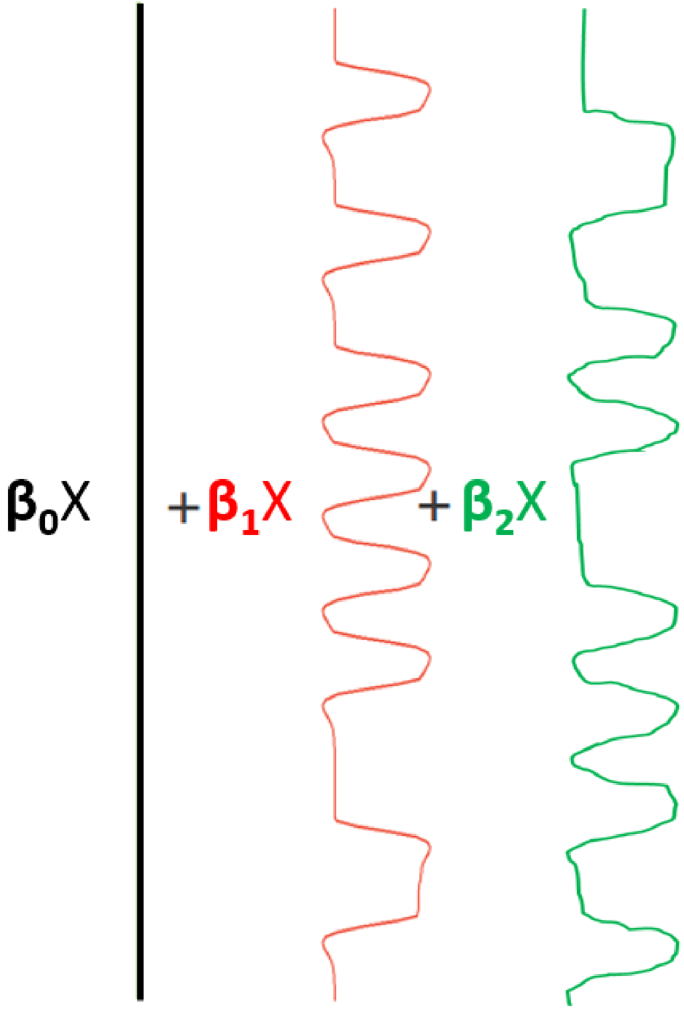
Deconstructed signal

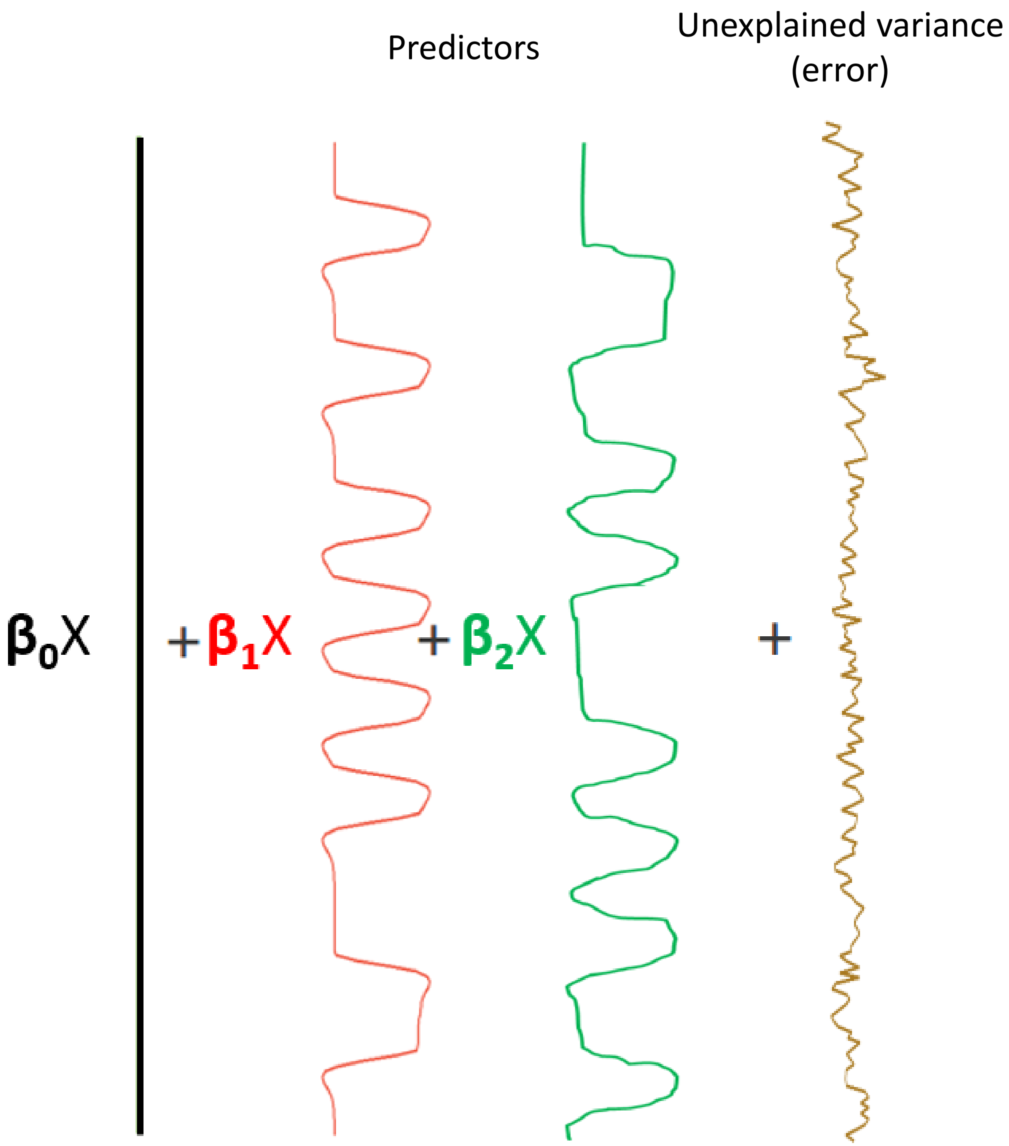
(de)Constructing the BOLD signal

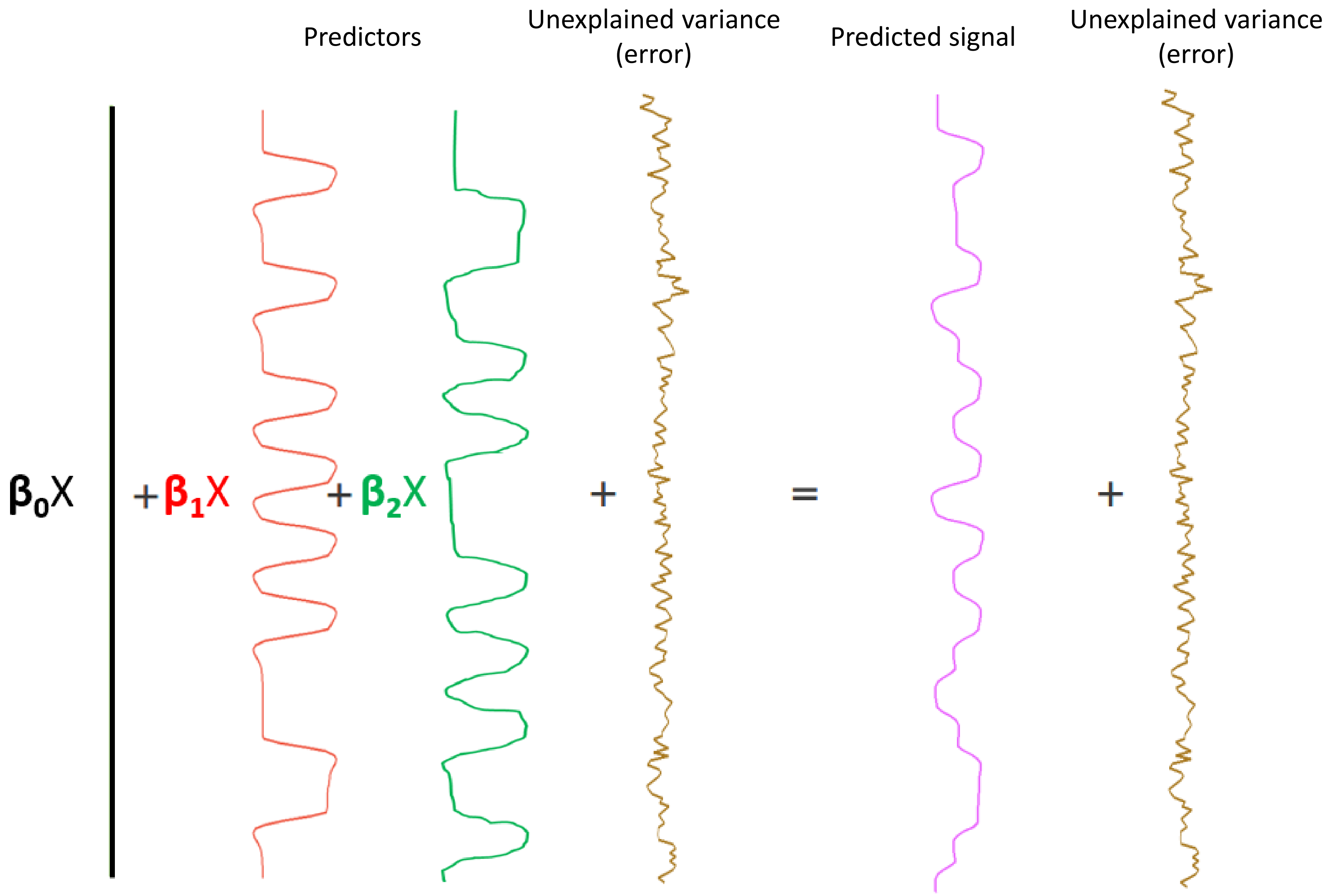


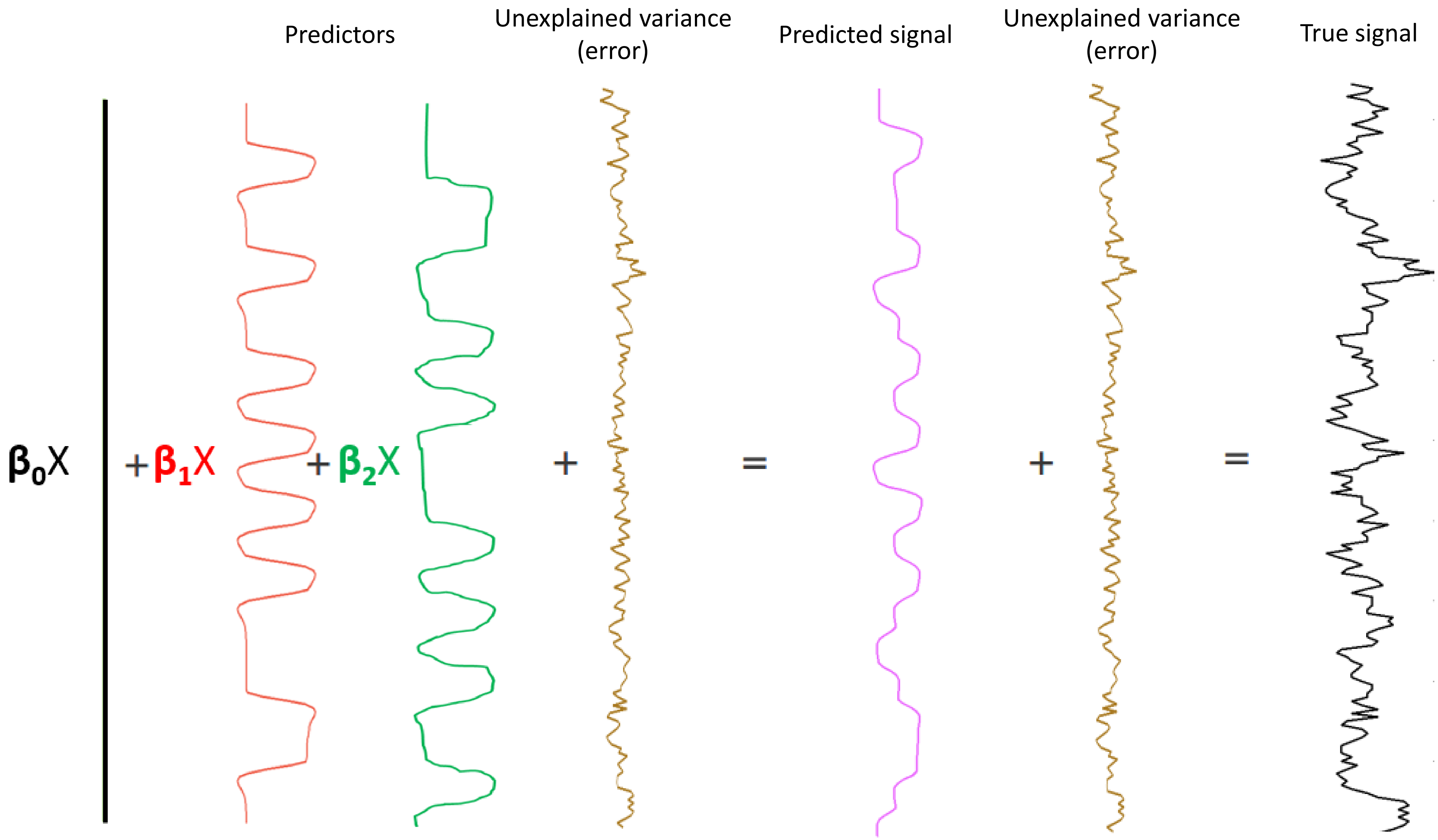
For every time point: **signal(t)** – **prediction(t)** = **error(t)**

Predictors



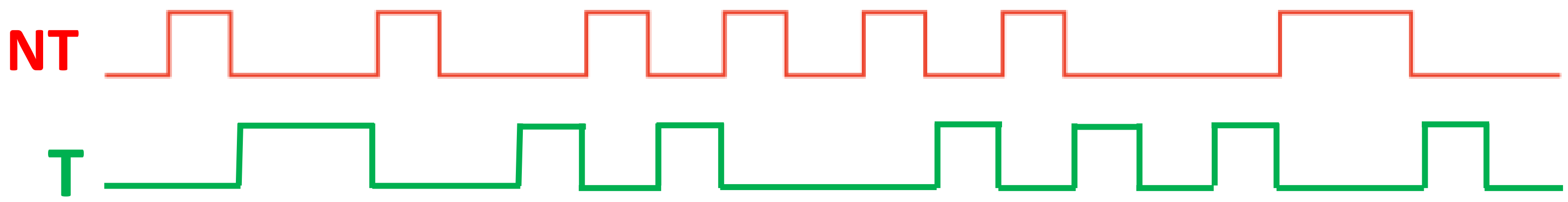

Predictors
Unexplained variance
(error)
β₀X
+β₁X
+β₂X
+


Predictors
Unexplained variance (error)
Predicted signal
Unexplained variance (error)
$\beta_0 X$
$+ \beta_1 X$
$+ \beta_2 X$
$+$
$=$
$+$


Predictors
Unexplained variance (error)
Predicted signal
Unexplained variance (error)
True signal
$\beta_0 X + \beta_1 X + \beta_2 X + = + =$

(de)Constructing the BOLD signal

Experiment



Analysis: the approach that works

- Find the beta-weights that best approximate a voxel's signal time-series The best approximation is the one with the least errors
- Compare the beta-weights for estimated response to **No-Think** to beta-weights for estimated response to **Think**

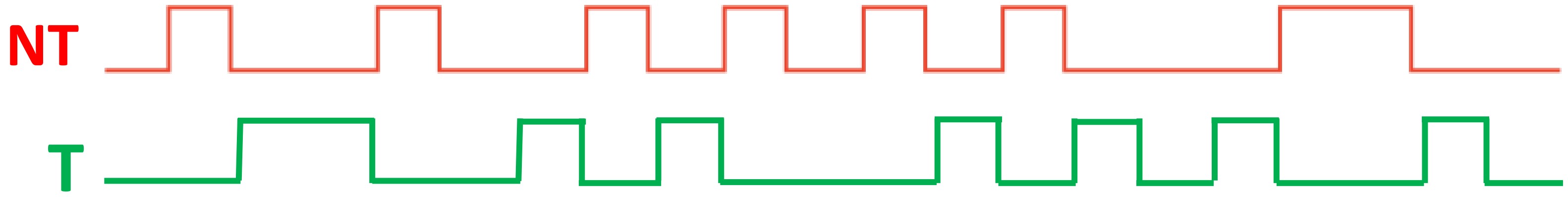
BOLD signal = task-related activity changes + noise (other changes)

explained variation | unexplained variation

Linear combination of predictors (deconstructed signal) + errors

(de)Constructing the BOLD signal

Experiment



Analysis: the approach that works

- Find the beta-weights that best approximate a voxel's signal time-series The best approximation is the one with the least errors
- Compare the beta-weights for estimated response to **No-Think** to beta-weights for estimated response to **Think**

To find beta-weights, we use **General Linear Model (GLM)**

How to find betas: **GLM** $Y = X\beta + \varepsilon$

BOLD signal = X * b + errors

explained variation — task-related activity changes

unexplained variation — noise (other changes)

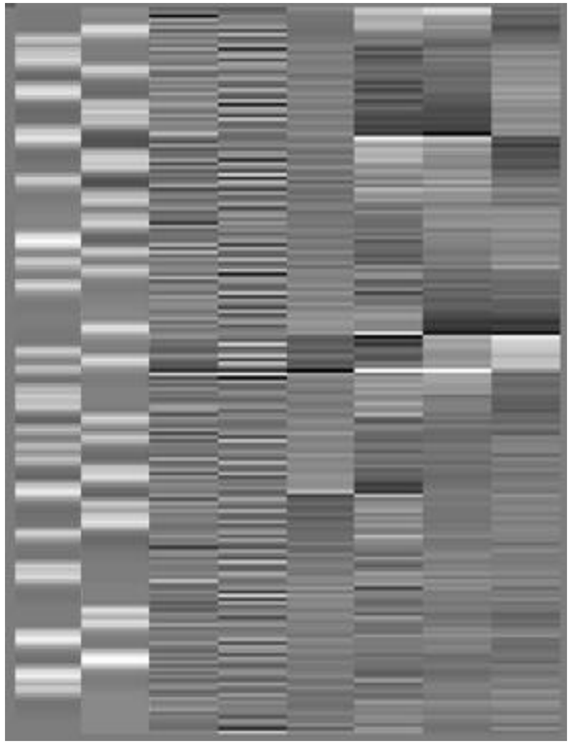
- **What we know?**
  - **BOLD signal**: we collect this from the brain (functional data)
  - **X**: the design matrix (each column is a predictor that we built ourselves)
- **What we want to find?**
  - **b**: vector of beta-weights (one weight for predictor in X) that give the best approximation of the BOLD signal
- **How we find it?**
  - By minimising the sum of squared errors. In practice, the **GLM has a formula, which guarantees to find these beta-weights**

How to find betas: **GLM** $Y = X\beta + \varepsilon$

- Any predictor that can help approximate the BOLD signal will decrease the Sum of Squared Errors
- Therefore, we include additional predictors:
  - The 6 head-motion parameters

An example design matrix.
Each column is a predictor:

- Think
- No-Think
- 6 movement parameters

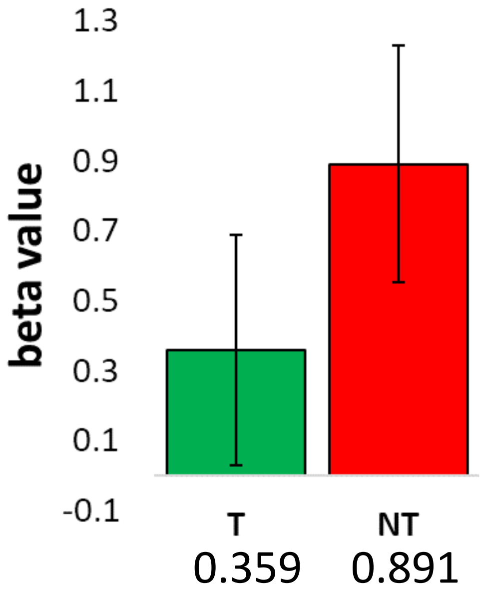


Multiple regression, to find beta-weights

voxel1	T	NT	rp1	rp2	rp3	rp4	rp5	rp6
-1.98409	0	0	0.744	-0.968	-0.815	-0.263	2.13	-0.0816
-2.49237	0	0	1.15	-0.399	-1.42	-0.365	1.95	0.0903
-1.98409	0	0	-1.33	0.482	-1.81	-0.729	0.666	-0.304
-2.23823	0	0	2.11	0.35	-1.81	-0.748	0.177	-0.157
-2.74651	0	0.0865661	1.49	0.722	-1.99	-0.63	-0.0958	-0.202
-1.98409	0	0.374888	2.35	1.17	-1.98	-0.718	0.0911	-0.509
-1.72996	0	0.384923	0.989	-0.953	-2.12	-1.12	-0.369	-0.378
-1.47582	0.0865661	0.216117	2.11	1.49	-2	-1.41	-0.556	-0.0755

Coefficients

Model		Unstandardized	Standard Error	Standardized	t	p
1	intercept	−0.199	0.106		−1.877	0.062
	T	0.359	0.329	0.070	1.091	0.277
	NT	0.891	0.338	0.171	2.637	0.009
	rp1	−0.302	0.077	−0.302	−3.916	< .001
	rp2	−0.129	0.051	−0.129	−2.544	0.012
	rp3	0.326	0.063	0.327	5.223	< .001
	rp4	0.210	0.074	0.210	2.829	0.005
	rp5	−0.008	0.059	−0.008	−0.135	0.893
	rp6	0.643	0.064	0.643	9.978	< .001



A comparison of beta-weights is called a **contrast**.
Formally, a contrast is a vector indicating which beta-weights we are testing. β(NT) - β(T): [-1 1 0 0 0 0 0 0]

How to find betas: **GLM**

1. **Extract the signal time-series** from a given voxel

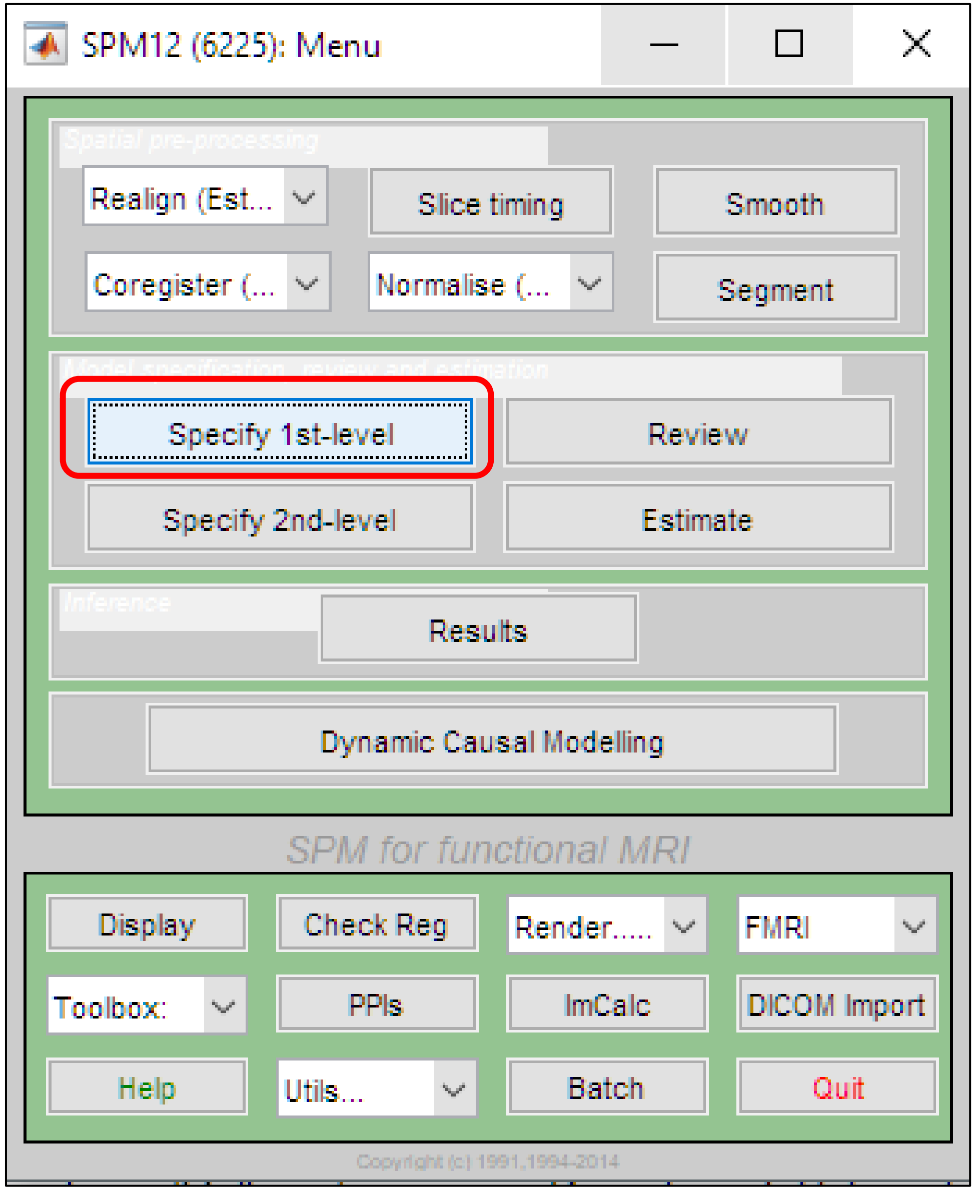
2. **Run GLM** (the signal and your design matrix are the inputs) to **find beta-weights** that best approximate the true signal

3. Define your **contrast** and test it

$$t = \frac{\beta(\mathbf{NT}) - \beta(\mathbf{T})}{\text{error(ish)}}$$

4. Repeat for **all voxels**
   - Produces
     - An image file for each predictor with beta values for each voxel: **beta-maps**
     - An image file with contrast values for each voxel: **contrast-maps**
     - An image file with contrast-specific t-values for each voxel: **t-maps**

First-level (subject specific) analysis



First-level analysis: model specification

Help on: fMRI model specification	
Directory	...FirstLevel\stats\model01\s01
Timing parameters	
. Units for design	Seconds
. Interscan interval	2
. Microtime resolution	31
. Microtime onset	15
Data & Design	
. Subject/Session	
. . Scans	207 files
. . Conditions	
. . Multiple conditions	...\TNT1\s01_run1_onsets.mat
. . Regressors	
. . Multiple regressors	...1_006_TNT1_pace_0001.txt
. . High-pass filter	128
. Subject/Session	
. . Scans	204 files
. . Conditions	
. . Multiple conditions	...\TNT2\s01_run2_onsets.mat
. . Regressors	
. . Multiple regressors	...1_008_TNT2_pace_0001.txt
. . High-pass filter	128
Factorial design	
Basis Functions	
. Canonical HRF	
. . Model derivatives	No derivatives
Model Interactions (Volterra)	Do not model Interactions
Global normalisation	None
Masking threshold	0.8
Explicit mask	
Serial correlations	AR(1)

Where to save the output

TR

t: A number of time-bins per scan (number of slices)
t0: The first slice (or reference slice)

Change these only if slice-time correction was performed at pre-processing! If you have 31 slices and have made slice 15 the reference slice you would set t=31, t0=15.

The pre-processed files for this session

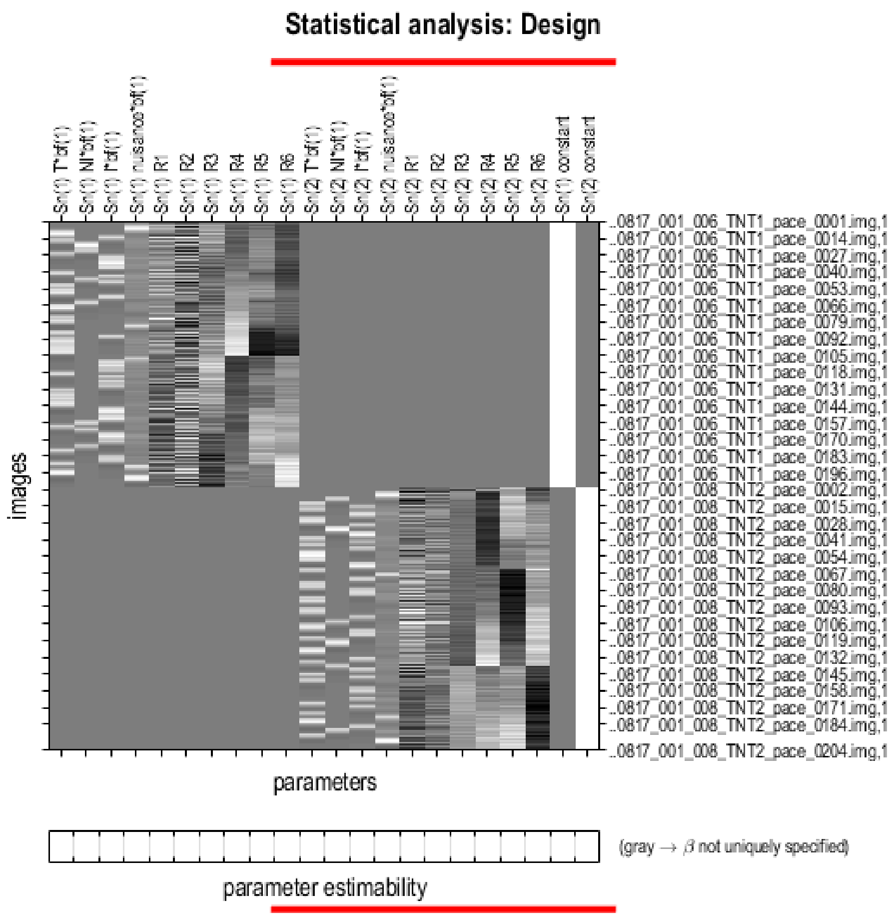
.mat files with condition onset and duration information

Movement parameter .txt file

Slow signal drifts with a period longer than 128 s will be removed. It is a way to remove possible confounds.

Accounting for serial correlations in fMRI time series due to aliased biorhythms and unmodelled neuronal activity. Accounts for non-independency in the BOLD response.

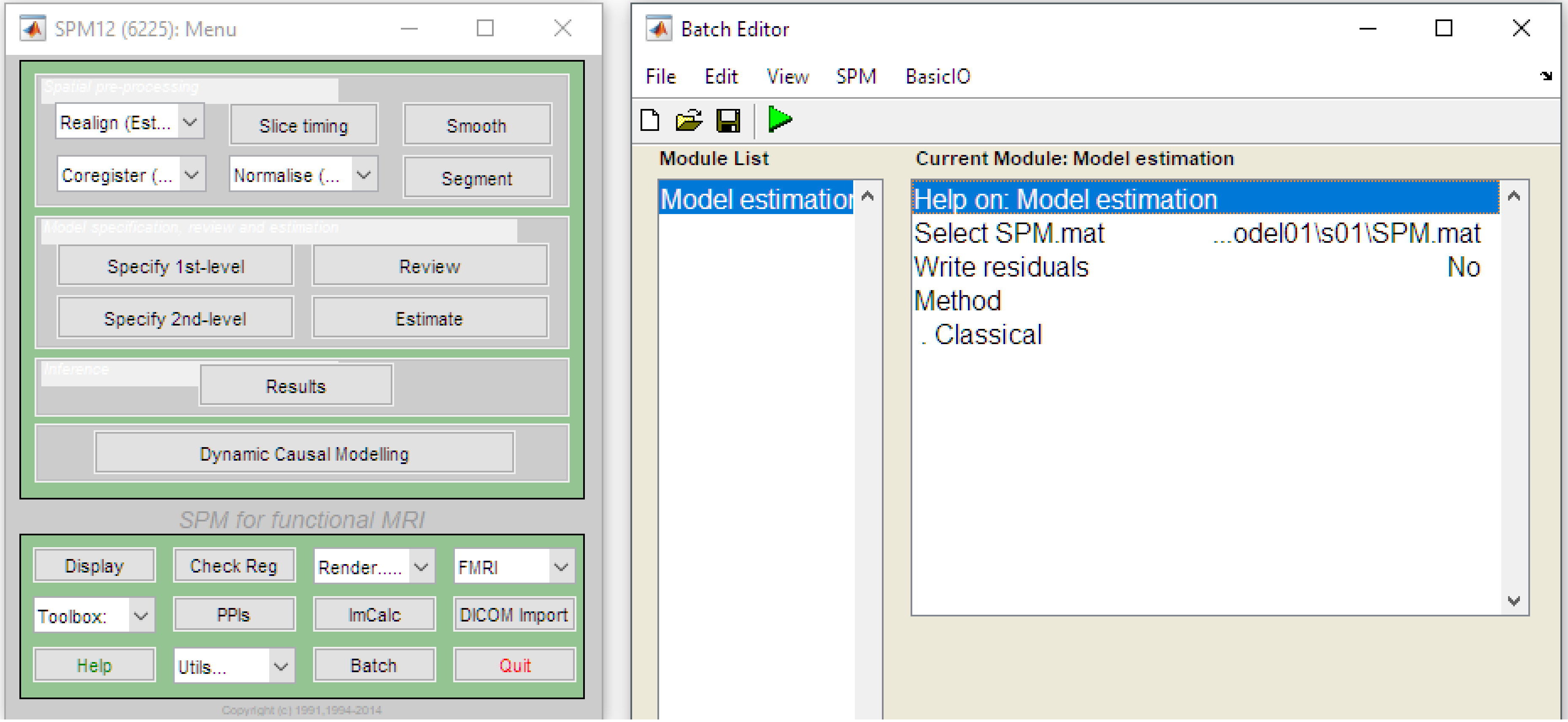
First-level analysis: design matrix



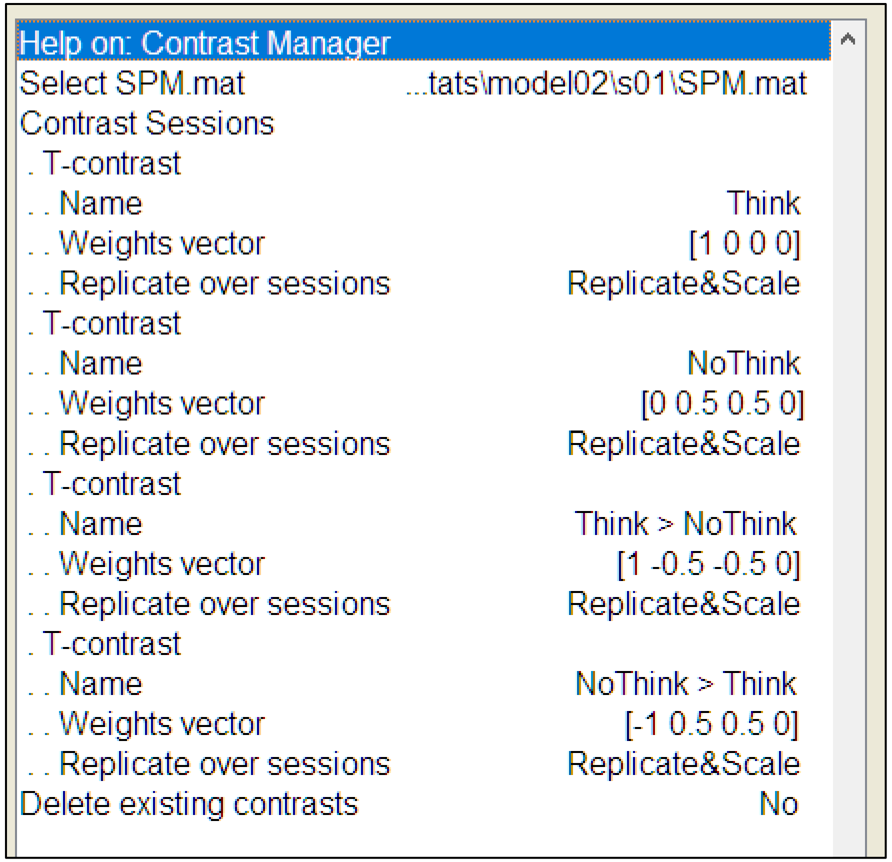
Design description...

Basis functions : hrf
Number of sessions : 2
Trials per session : 4 4
Interscan interval : 2.00 {s}
High pass Filter : [min] Cutoff: 128 {s}
Global calculation : mean voxel value
Grand mean scaling : session specific
Global normalisation : None

First-level analysis: model estimation

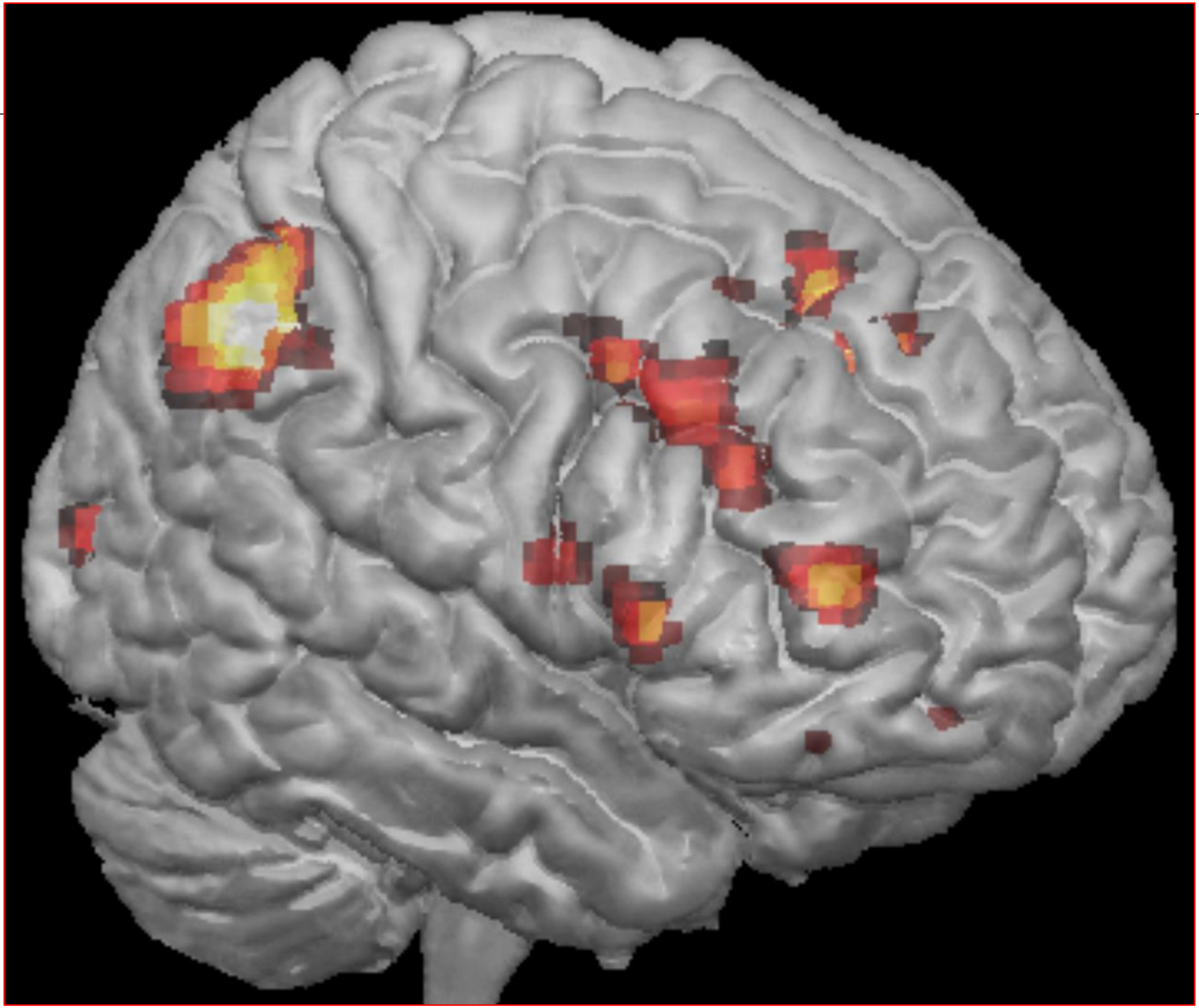


First-level analysis: define contrasts



No-Think > Think

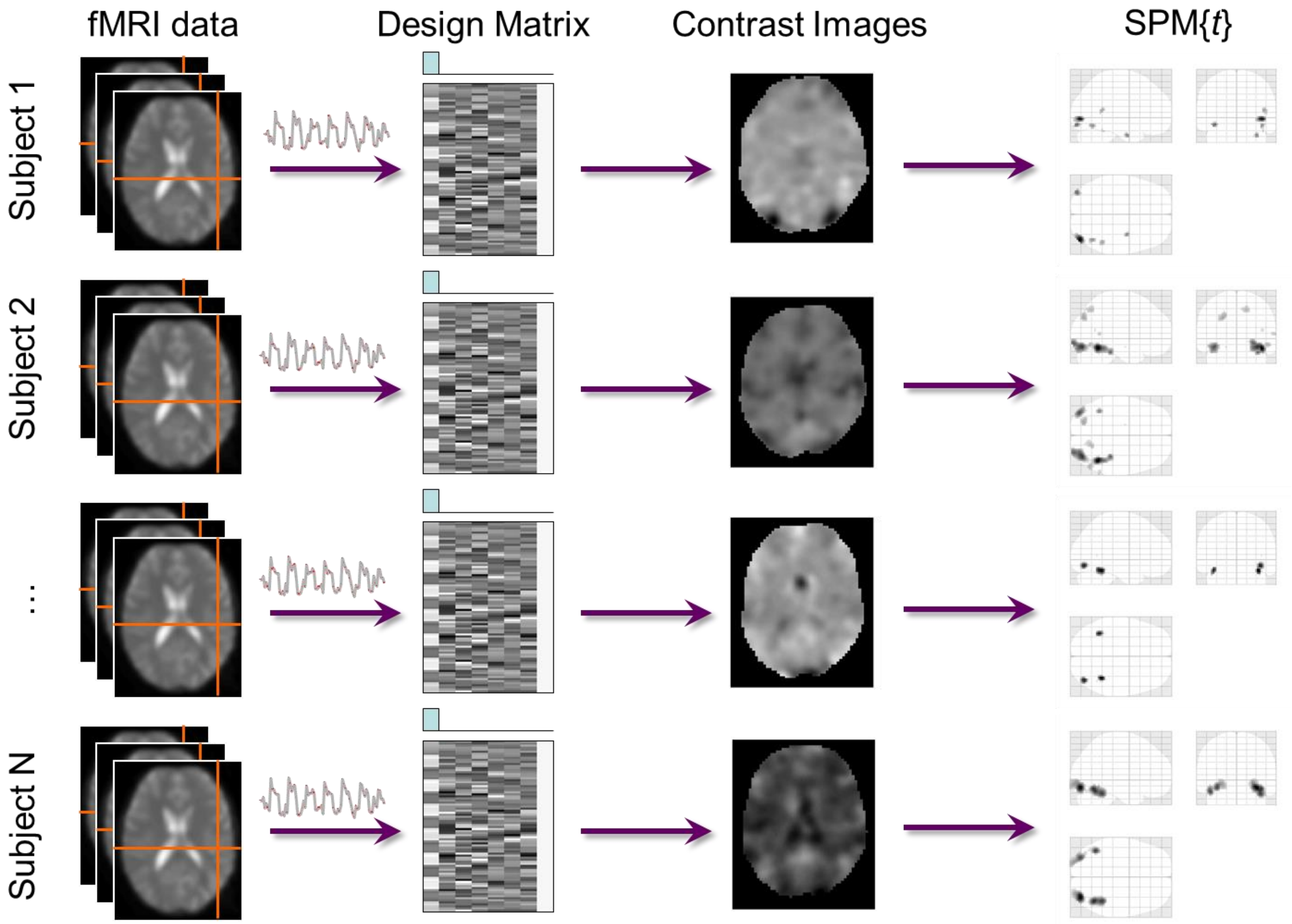
FWE corrected



Statistical brain map (t-map) of regions showing significant positive difference between No-Think and Think

First-level analysis

- Run the GLM for each subject

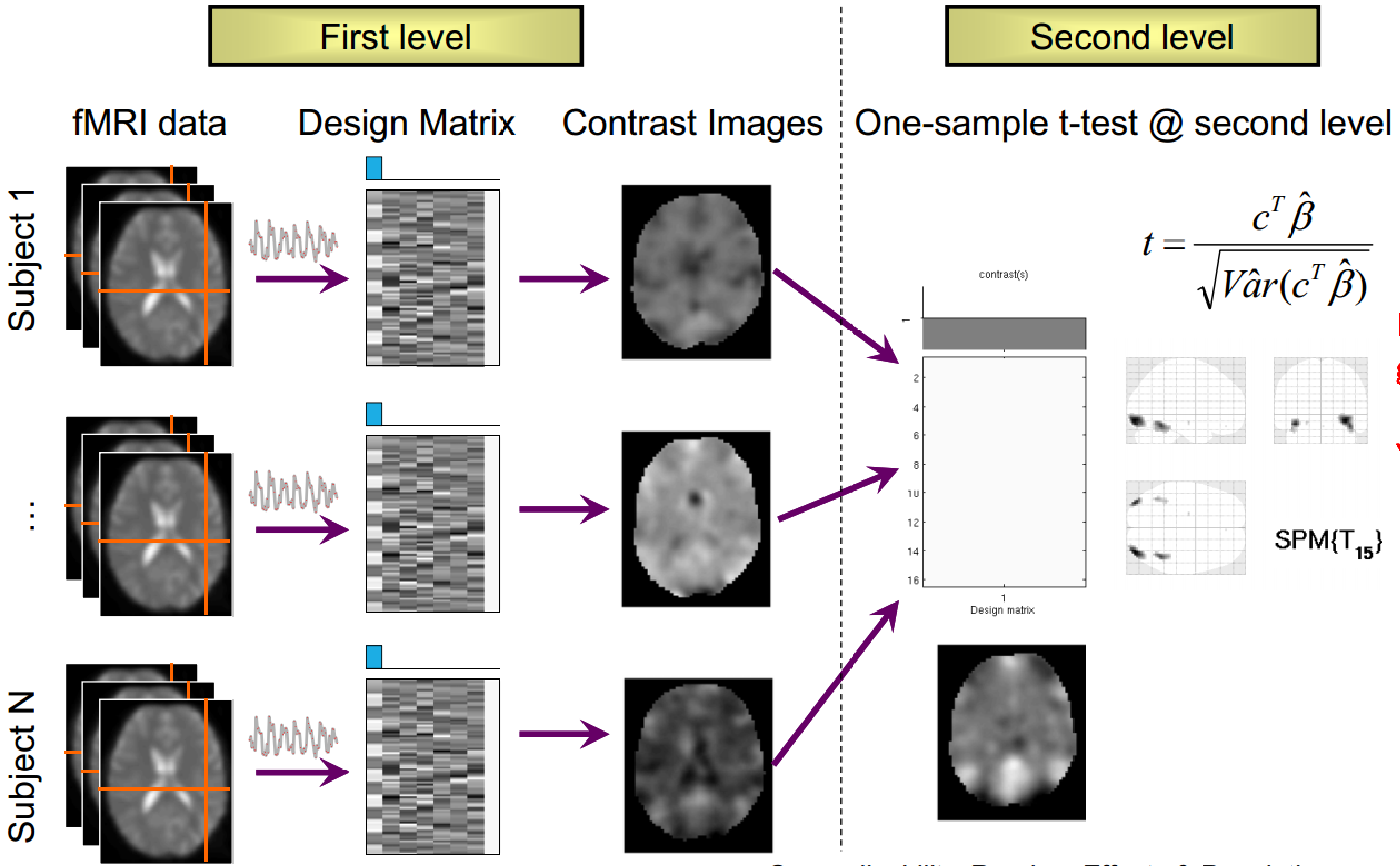


Group level (2nd level) analysis is across subjects

- Which voxels are showing a significant activation differences between our conditions consistently **within a group**
  - Average contrast effect across sample (e.g., one-sample t-test)

- Importantly, all subject brains need to be in common space, e.g. MNI, to perform voxel-wise group analyses
  - That was achieved by the **Normalisation step in** pre-processing

Summary statistics, Random effects approach



$$t = \frac{c^T \hat{\beta}}{\sqrt{\hat{Var}(c^T \hat{\beta})}}$$

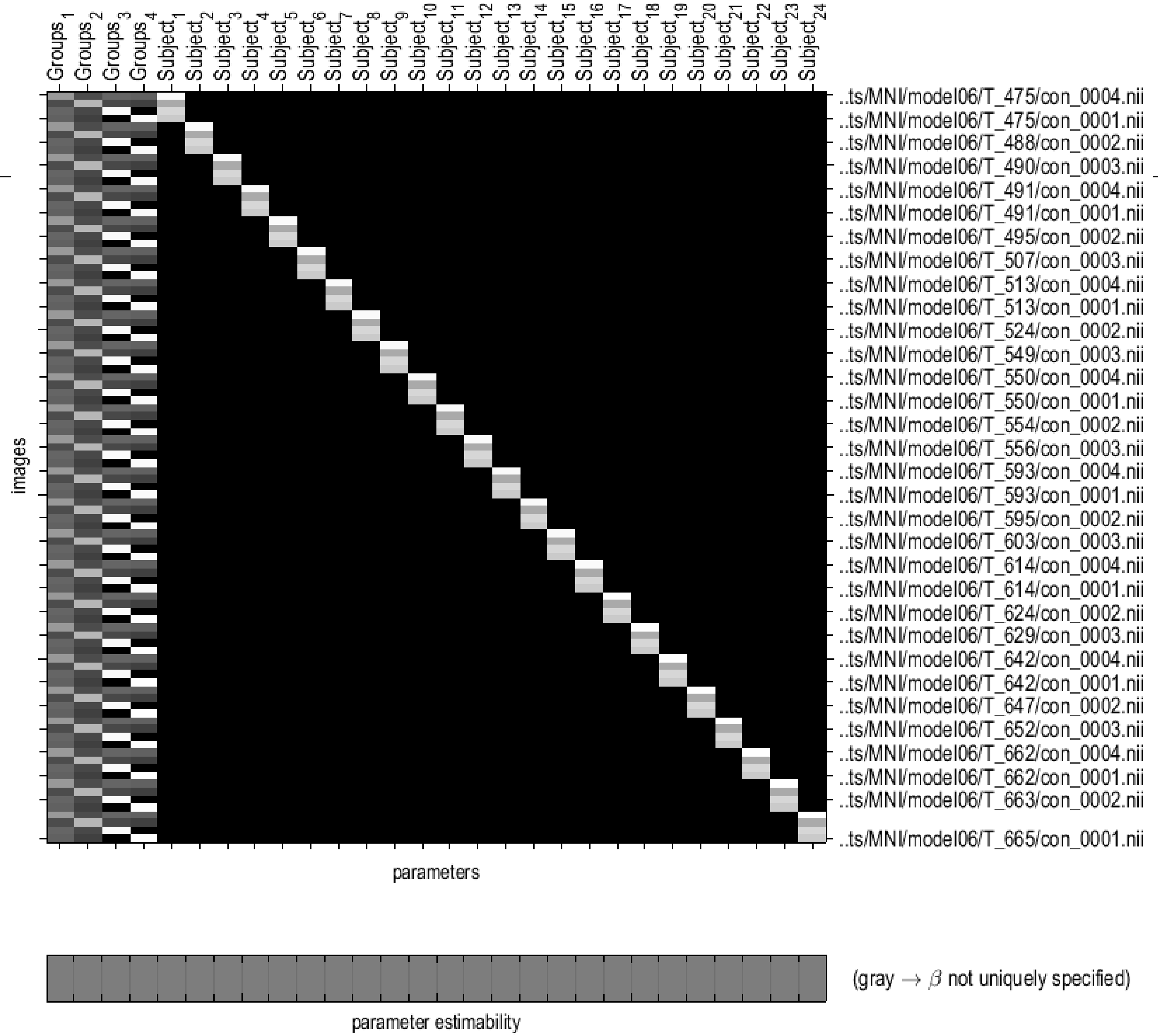
It is still a GLM model, and we get group level beta estimates!

$Y_{group} = X_{group}\ \beta_{group} + \varepsilon_{group}$

Generalisability, Random Effects & Population Inference. Holmes & Friston, NeuroImage,1998.

Within-subjects ANOVA

- **conditions**: Stop, Go, NT, T



Design description...

Design : ANOVA - within subject
Global calculation : omit
Grand mean scaling : <no grand Mean scaling>
Global normalisation : <no global normalisation>
Parameters : 4 condition, +0 covariate, +24 block, +0 nuisance
28 total, having 27 degrees of freedom
leaving 69 degrees of freedom from 96 images

Stats tests at the 2nd level

- Condense where possible
  - If a factor can be collapsed through a contrast at the 1st level, do so and use the simplest possible 2nd level model
  - T-tests at the 2nd level are preferred
    - Avoids need to estimate non-sphericity to account for within-subject correlations across repeated measures
    - Generally more accurate estimation of error
- However, if more than 2 factors or levels exist, a single t-contrast cannot capture main effects and interactions
  - 2nd level ANOVA will be necessary

Stats tests at the 2nd level

SPM RECIPE

Design	1^{st} Level	2^{nd} Level
1 group, 1 factor, 2 levels	A1 – A2	One-sample t-test
1 group, 1 factor, 2+ levels	A1,A2, …, A_n	One-way ANOVA (within-subjects)
1 group, 2 factors, 2 levels each	(A1B1+A1B2)-(A2B1+A2B2): ME A (A1B1+A2B1)-(A1B2+A2B2): ME B (A1B1+A2B2)-(A1B2+A2B1): A x B	One-sample t-tests
1 group, 2+ factors/2+ levels	Multiple contrasts for each ME and interaction	One-way ANOVA
2 groups, 1 factor, 2 levels	A1 – A2	Two-sample t-test
2 groups, 1 factors, 2+ levels	A1, A2, …, An	Two-way ANOVA (mixed)
2 groups, 2 factors, 2 levels each	(A1B1+A1B2)-(A2B1+A2B2): ME A (A1B1+A2B1)-(A1B2+A2B2): ME B (A1B1+A2B2)-(A1B2+A2B1): A x B	Two-sample t-tests
2 groups, 2+ factors/2+ levels	Multiple contrasts for each ME and interaction	Two-way ANOVA

SPM interface: possible designs

- One-sample t-test
- Two-sample t-test
- Paired t-test
- Multiple regression
- One-way ANOVA
- One-way ANOVA – within subject
- Full factorial
- Flexible factorial

Hypothesis

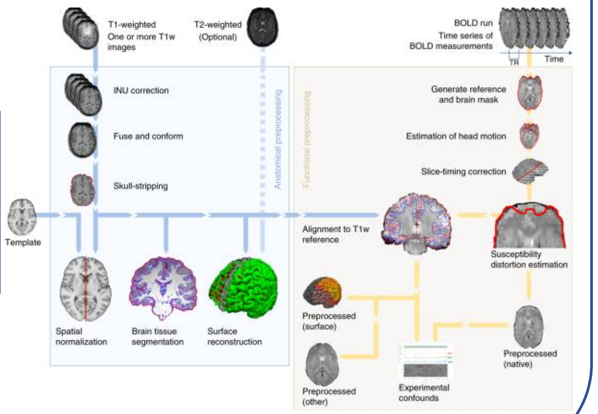
Design an experiment

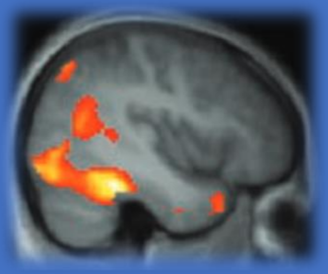
Stimuli
Timing

Collect the data

Anatomical image
Functional images
Event details

Pre-process & Analyse





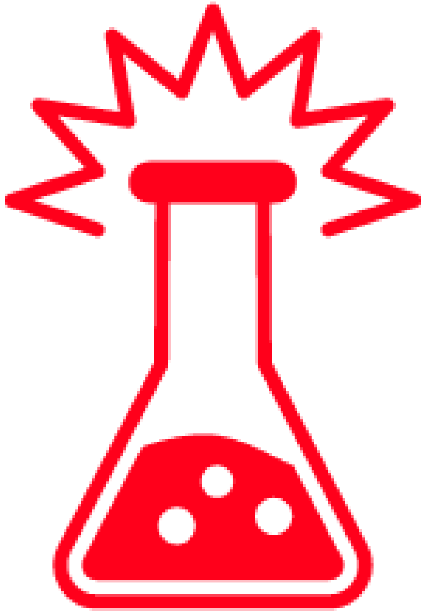
The final push

The Plan

- fMRI files and data
- Pre-processing
- Statistical analysis
- Recap

Hands-on materials
https://github.com/dcdace/fMRI_training

http://etc.ch/EEKG



Statistical analysis with Nilearn

https://github.com/dcdace/fMRI_training launch binder

04_GLM_analysis.ipynb