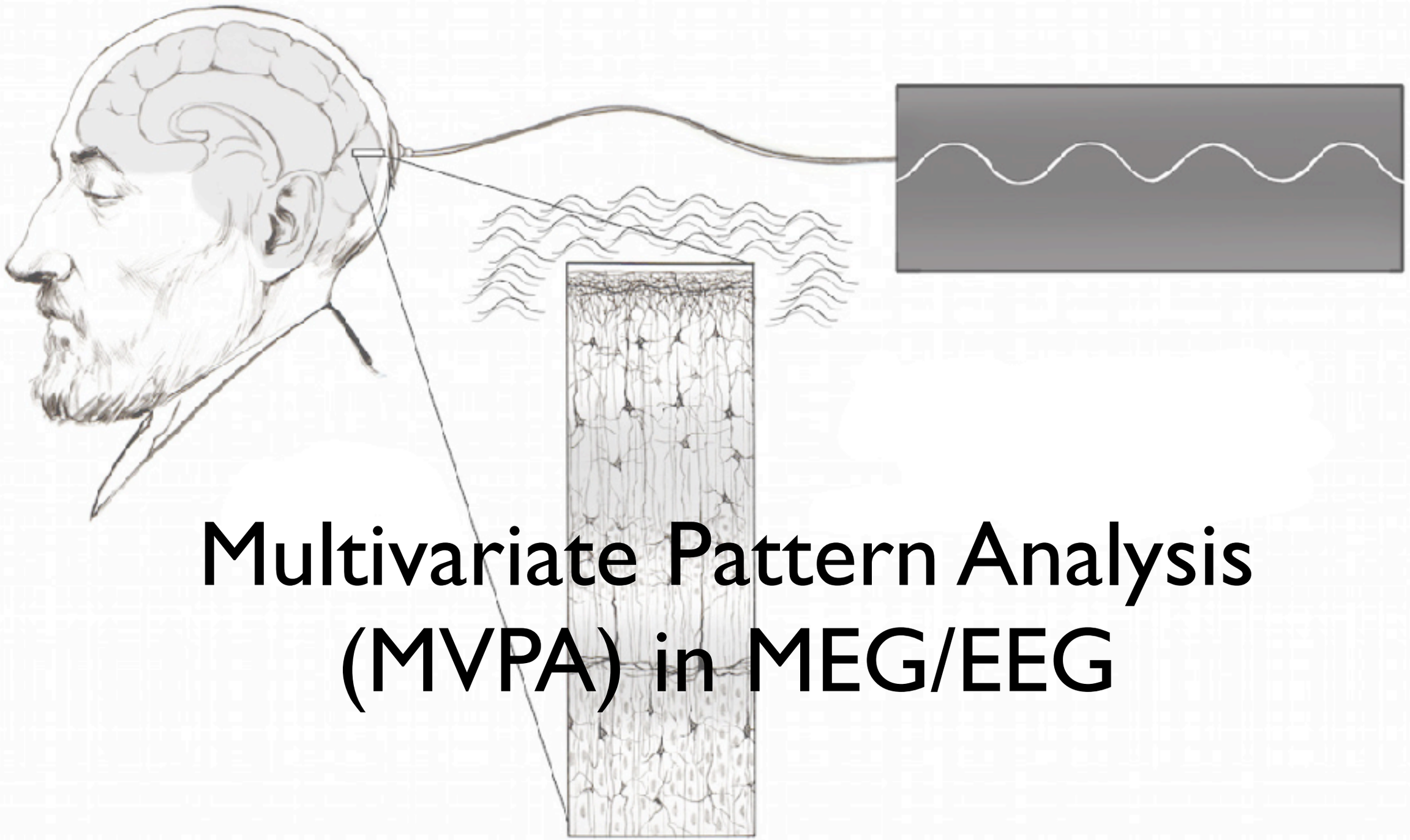


Multivariate Pattern Analysis (MVPA) in MEG/EEG

Johannes Fahrenfort

epdos

Graduate education network

Course overview

- **Day 1 (introduction)**
  - lecture 1: History and electrophysiological basis of EEG
  - lecture 2: Backward decoding models in MVPA: concepts and analytical approach
  - lecture 3: Advantages of MVPA, the temporal generalization method
  - lecture 4: Architecture of the ADAM toolbox and explaining the experiment of the practical
  - *Afternoon: practical*
- **Day 2 (advanced)**
  - lecture 1: Multiple comparisons, MVPA experimental design, mapping brain to brain/behavior
  - lecture 2: Forward encoding models in MVPA: how do they work, concepts and analytical approach
  - *Afternoon: practical, analyze your own data and/or a supplied dataset*

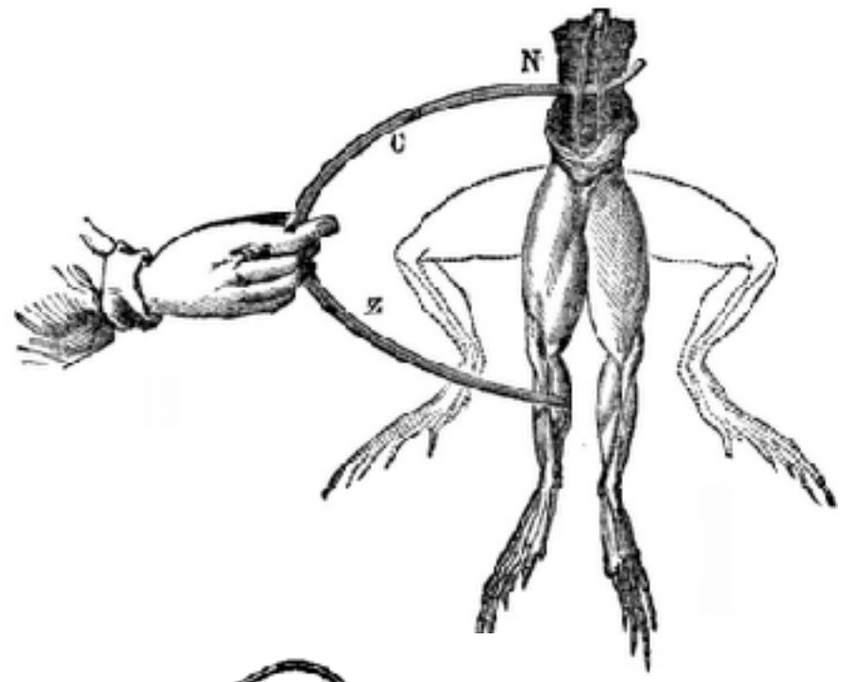
Lecture 1 -
History and neurophysiology of EEG/MEG

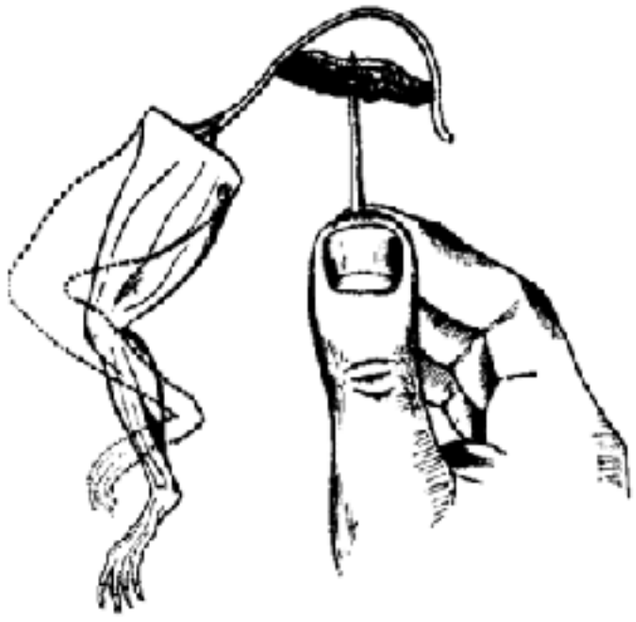
Luigi Galvani, the father of electrophysiology

- Electrophysiology: the study of the electrical properties of biological cells and tissues

- Galvani discovered in the late 1780s that stimulating the nerves of a dead frog with electricity resulted in muscle movement

- He coined the term ‘animal electricity’ to describe the force that activated these movements



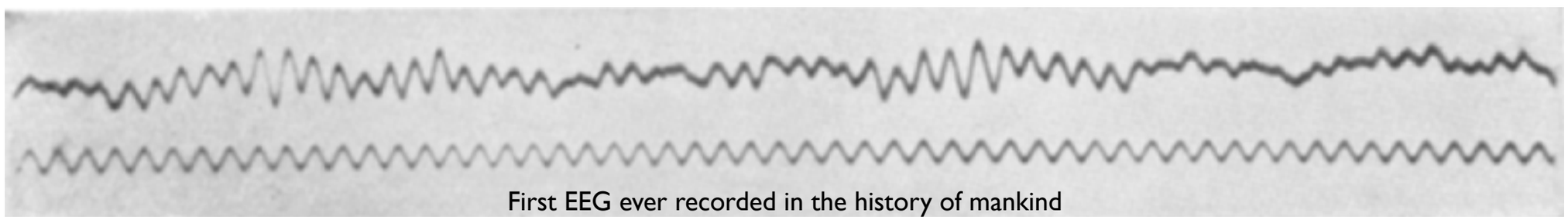


The history of electroencephalography (EEG)

- Gustov Fritsch and Eduard Hitzig (1870) showed that electrically stimulating the sensory-motor *cortex* of a dog produced movement

- Richard Caton (1875) showed the existence of electrical activity in exposed rabbit brain

- On July 6, 1924, **Hans Berger** (after 30 years of trying) for the first time recorded EEG from a human subject:



First EEG ever recorded in the history of mankind

Hans Berger

- It took another **5 years** (1929) before Berger dared to publish his results in "*Uber das elektrenkephalogramm des Menschen*" ("On the Electroencephalogram of Man")

- Between 1929 and 1938, Berger published 14 papers with the same title, distinguished from one another only numerically (report 1, report 2 etc)

- Many of the phenomena that Berger studied are still under investigation today

So what is EEG, really?

A few basic concepts

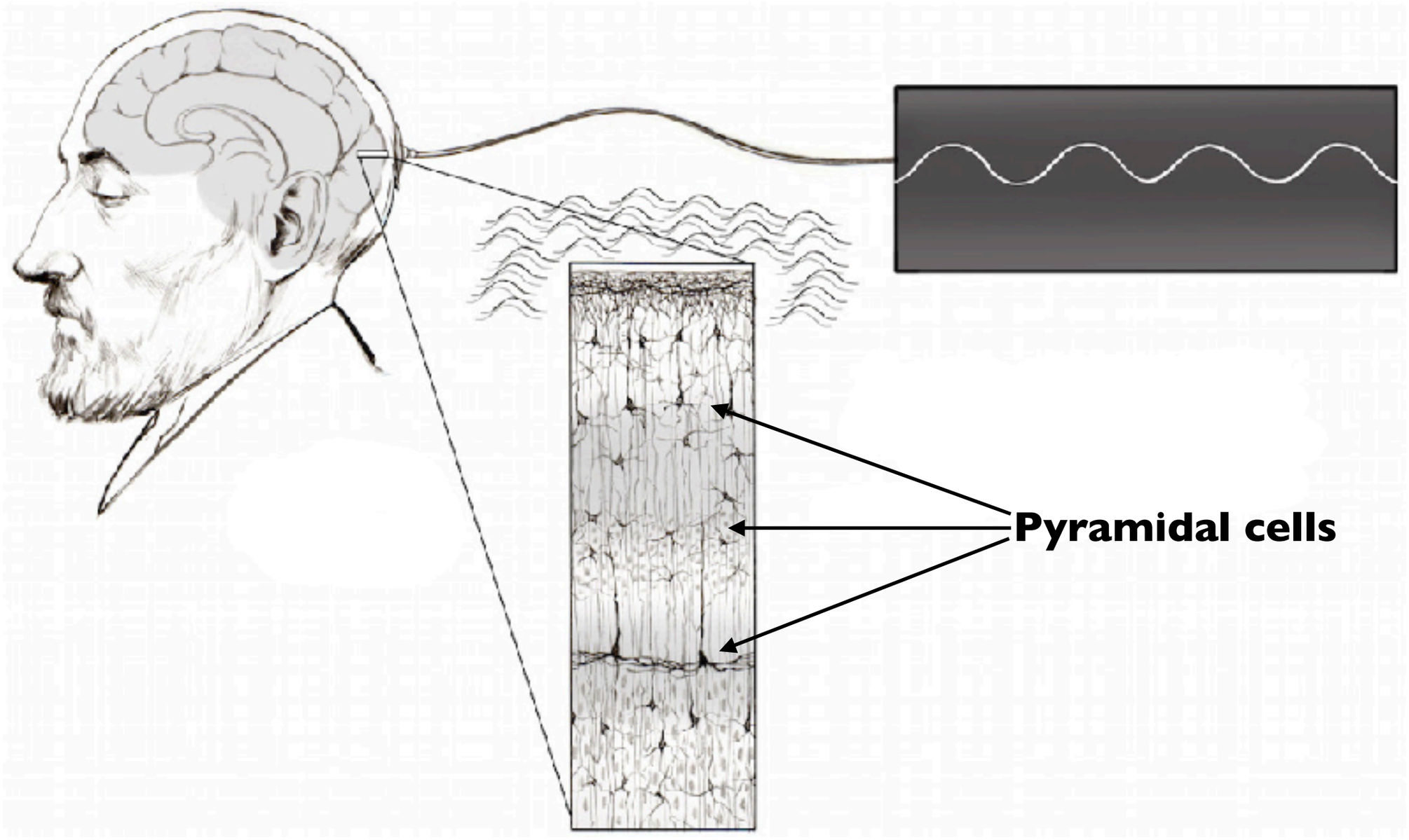
- Voltage
  - the *potential* of current to flow from one point to another.
  - think of it as "water pressure".
  - this is a relative measure!

- Current
  - number of charged particles (electrons, ions) that flow in a given time.
  - think of it as the volume of a "water flow".

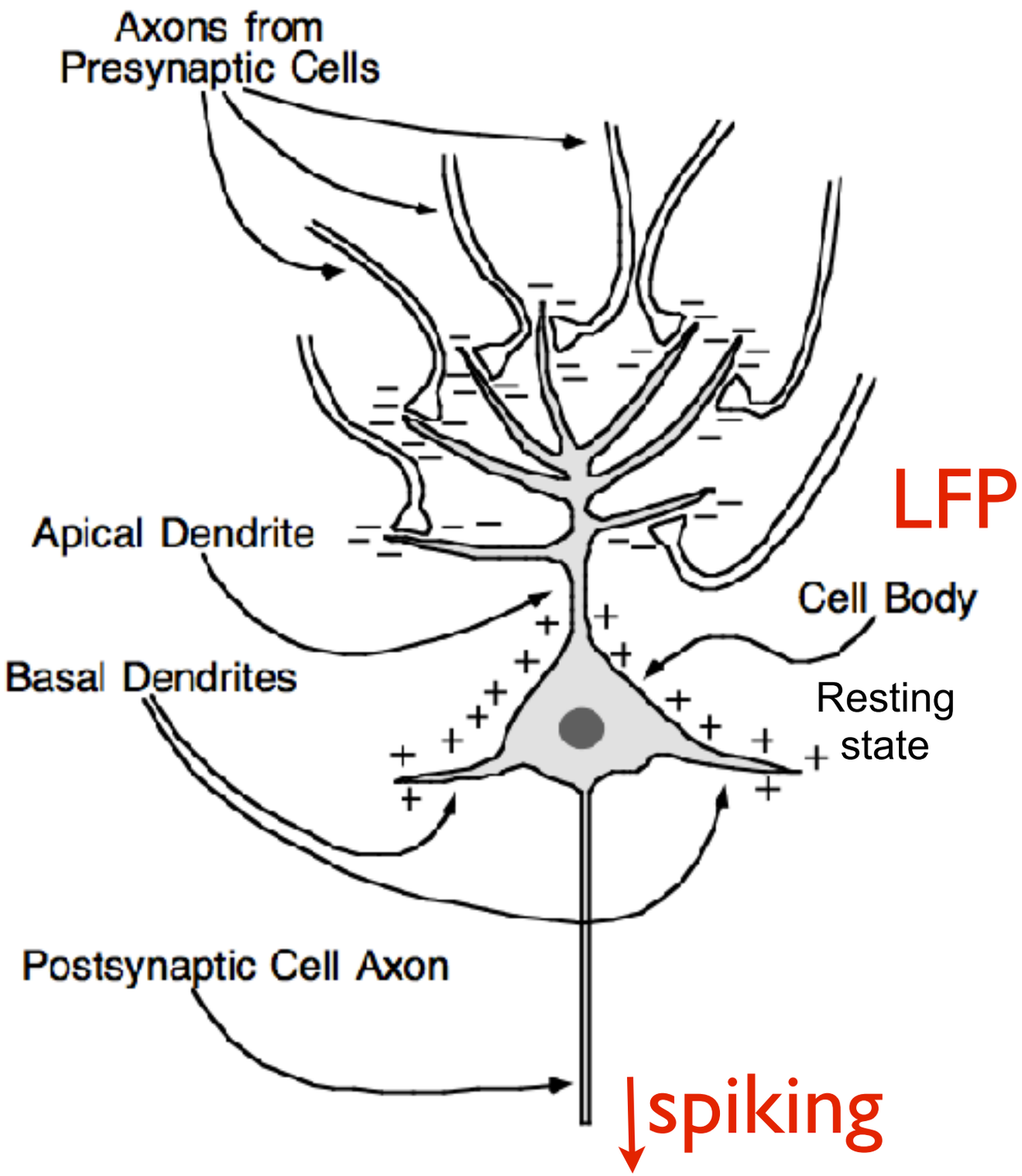
- Resistance
  - resistance to movement of charges
  - like having a skinny or blocked hose segment

- Ohm's Law: Voltage = Current * Resistance

So what does EEG measure?



Source of electric signal: cell polarization of pyramidal neurons

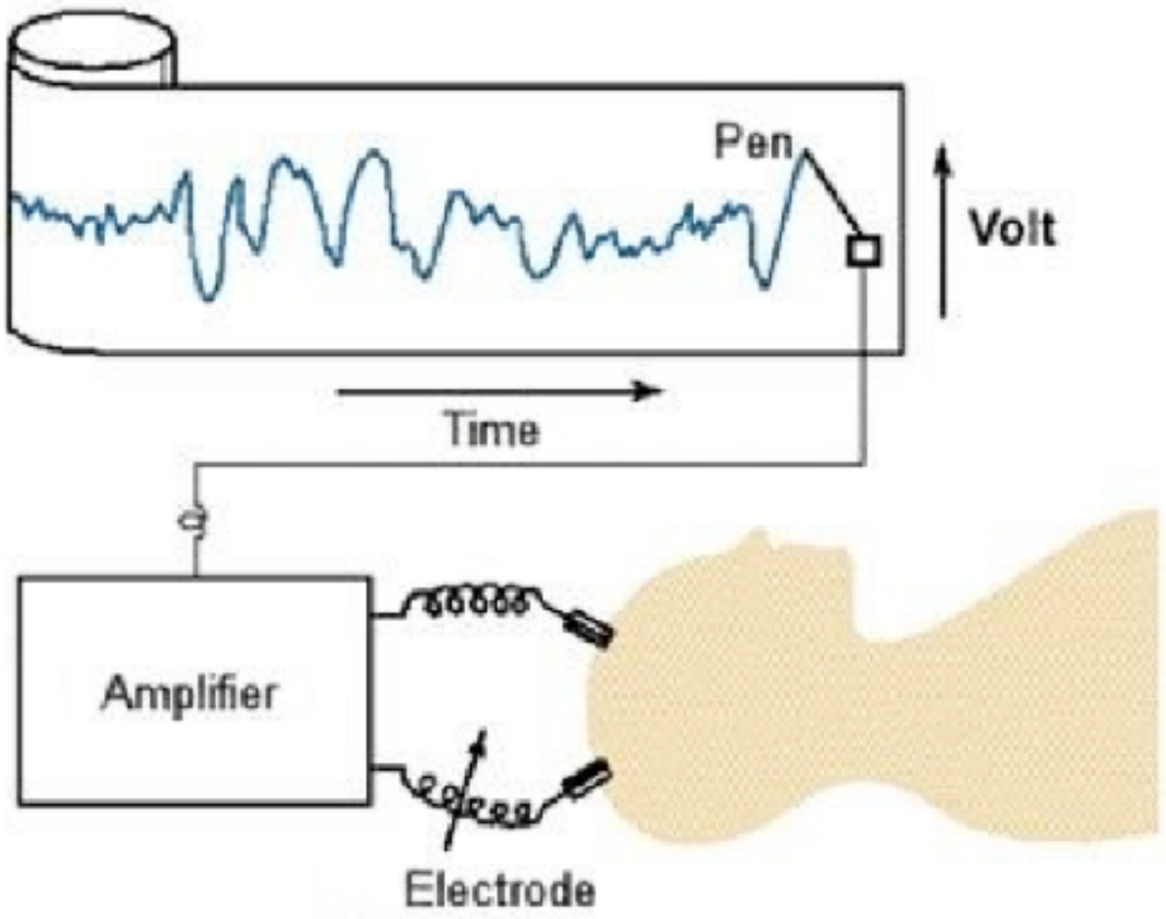


- Excitatory neurotransmitter released on dendrites causes positive charges to flow into dendrite
- Net negative on outside of dendrite
- Current flows through cell, leading to new spiking activity
- Polarity reverses with inhibitory neurotransmitter or postsynaptic potential on cell body / basal dendrites

Spiking activity versus local field potential (LFP)

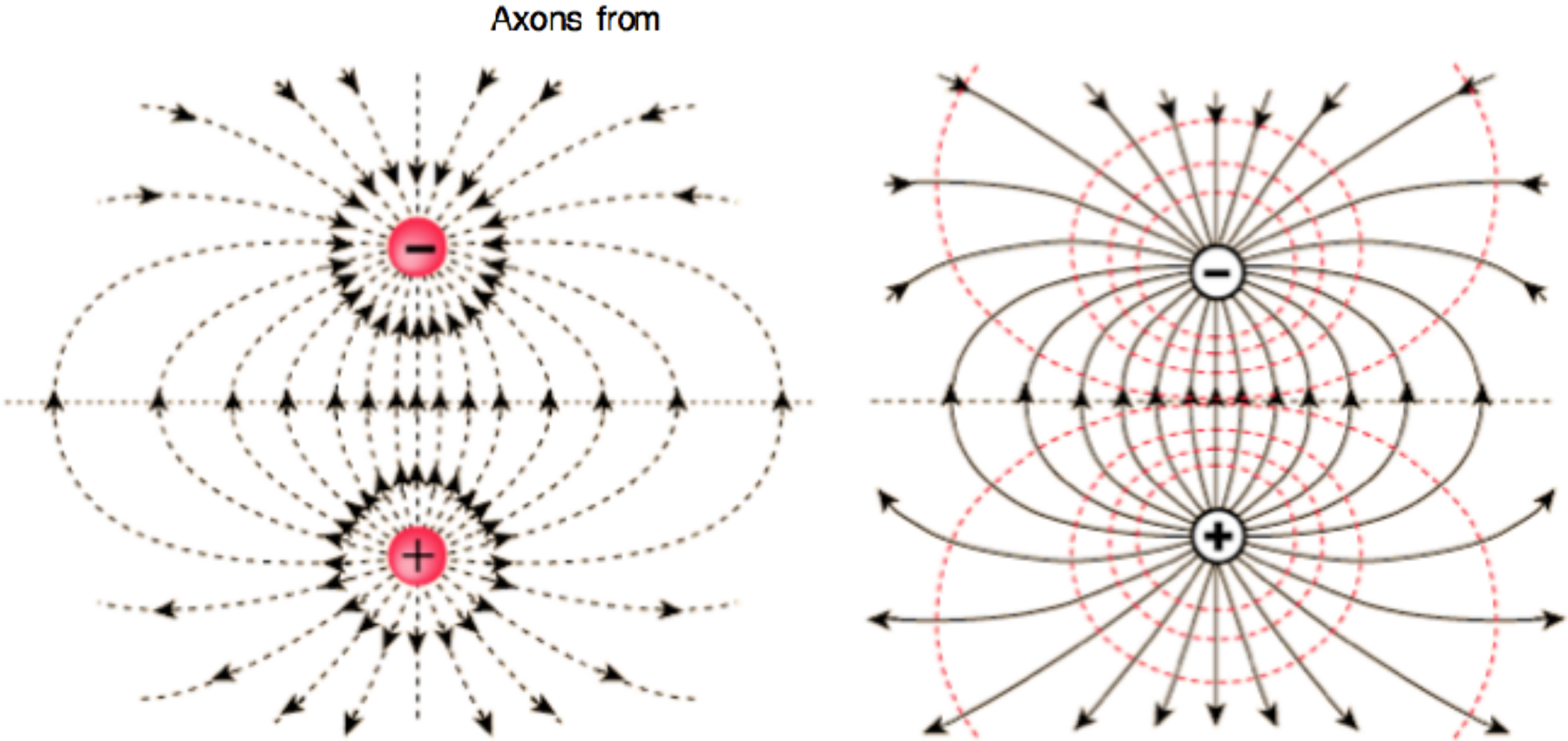
- Local Field Potential (LFP): the result of synchronized **input activity** of many dendrites into neurons
- Action potential (spiking): the **output activity** of a neuron

EEG



Measure volt difference between electrode and reference on the scalp

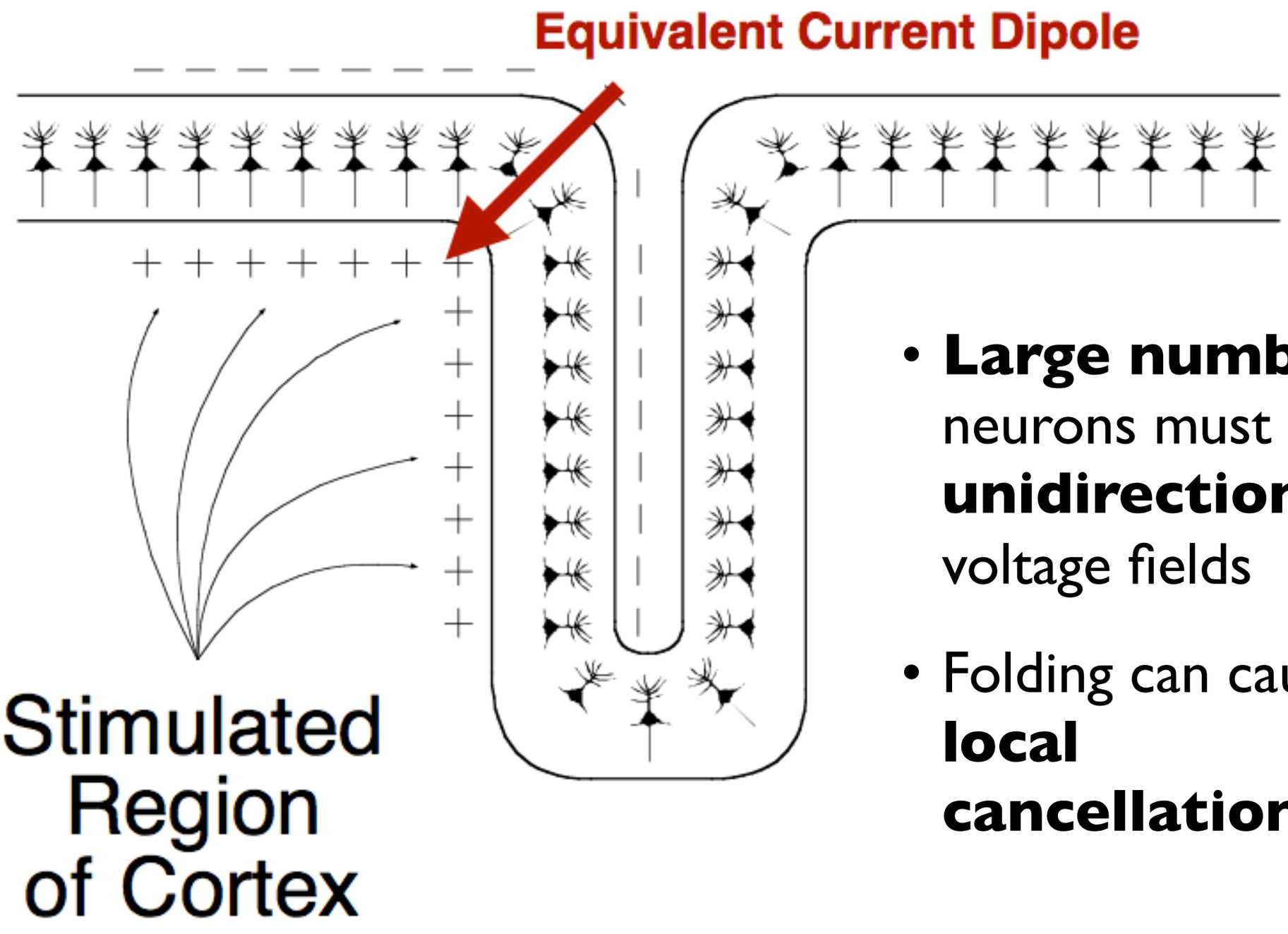
LFPs generate dipoles



Electric dipole field

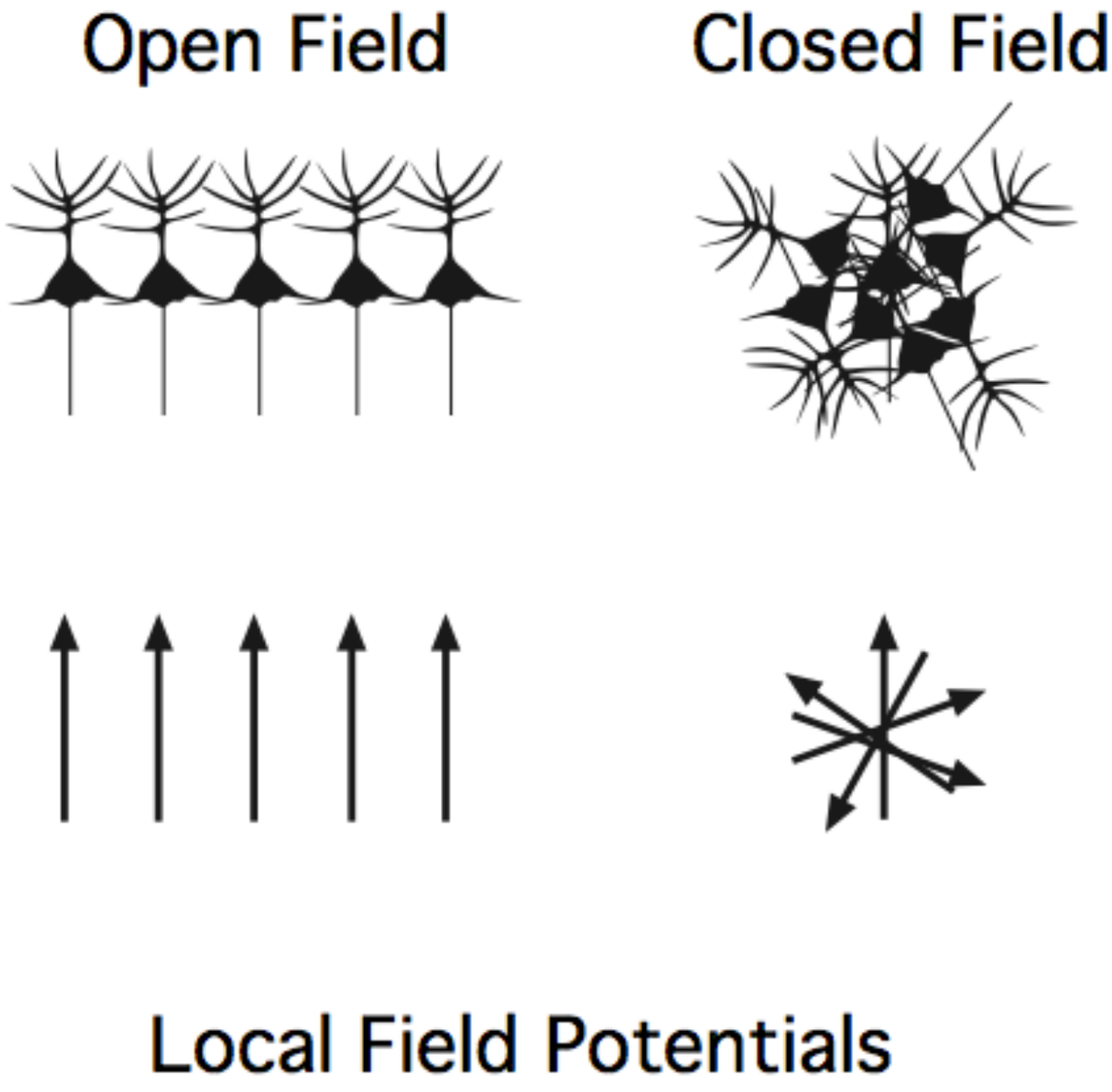
Equipotential lines

Cortical fold



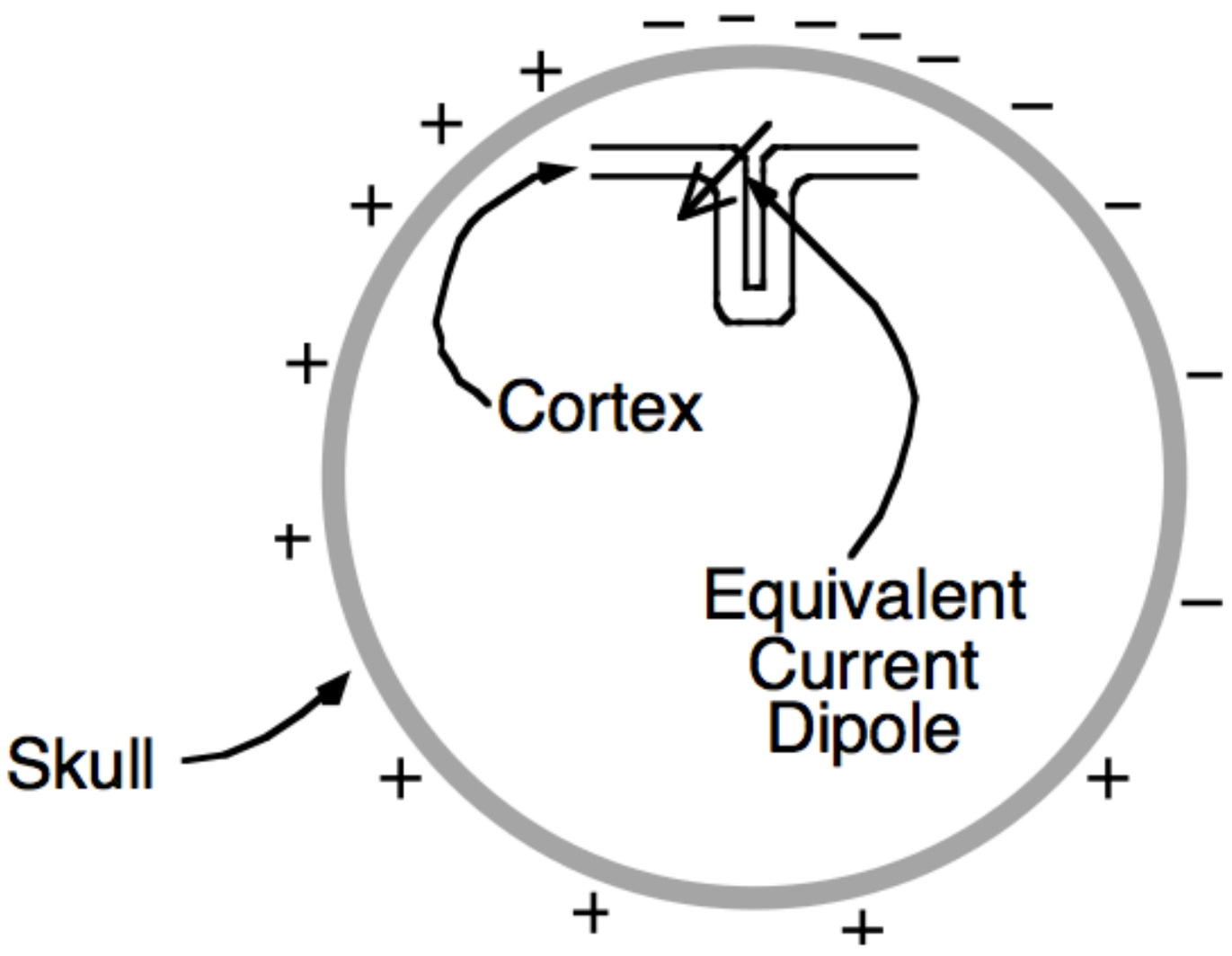
- **Large numbers** of neurons must have **unidirectional** voltage fields
- Folding can cause **local cancellation**

Source of EEG



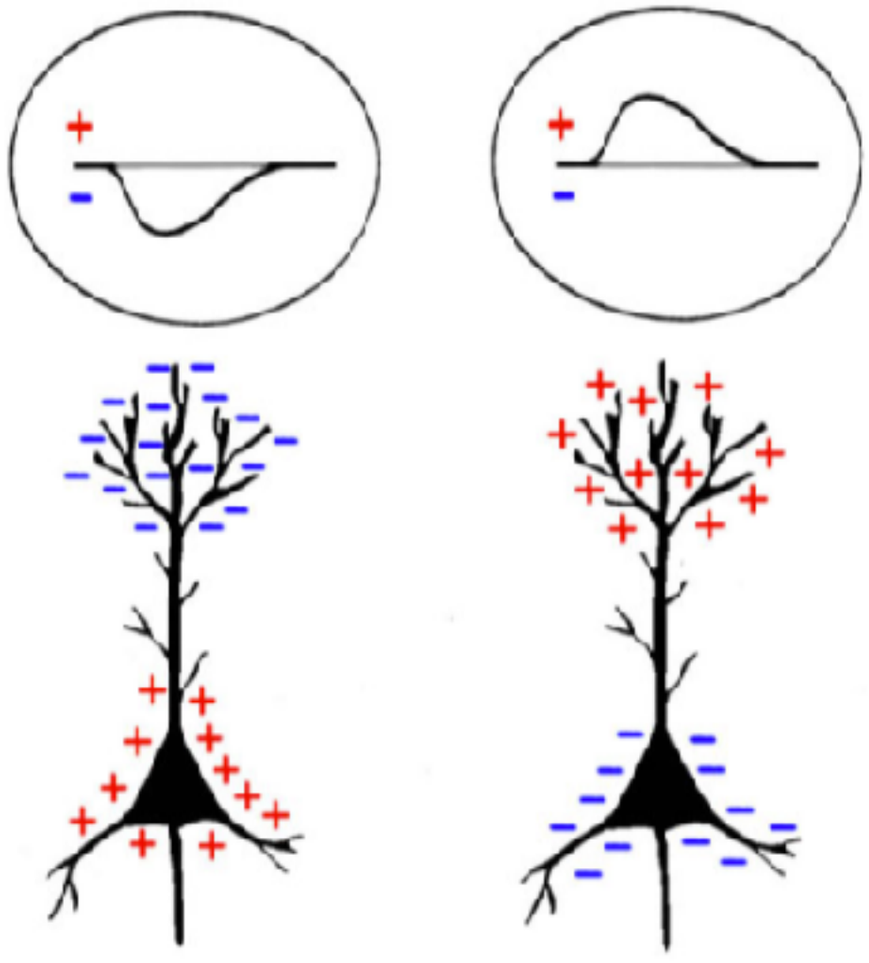
- **Local field potentials** (summation of postsynaptic inputs) **NOT** spiking activity / action potentials
- Scalp-recorded potentials only possible for layered structures with consistent orientations, which are mostly **cortical** (not subcortical)

Source of EEG



- Voltages spread through the head through **volume conduction**
- Voltage everywhere except at negative-positive transition
- Skull causes lateral spread (**blurring**)

Inhibition/excitation



Orientation of neurons with respect to electrode is in practice unknown

Moreover, either of these neurons may receive excitatory/inhibitory inputs at dendrite/soma

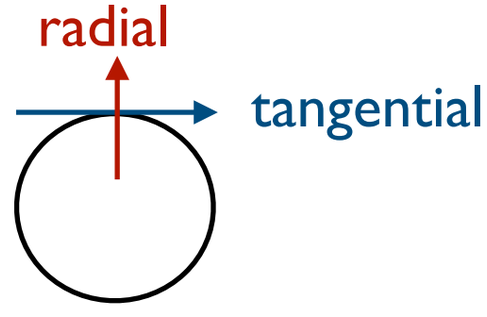
It is impossible to know whether a positive or negative EEG deflection is caused by inhibition or excitation

Three requirements for EEG

1. Many LFPs need to occur at the same time to create a sufficiently strong dipole (synchronous activity of many neurons)

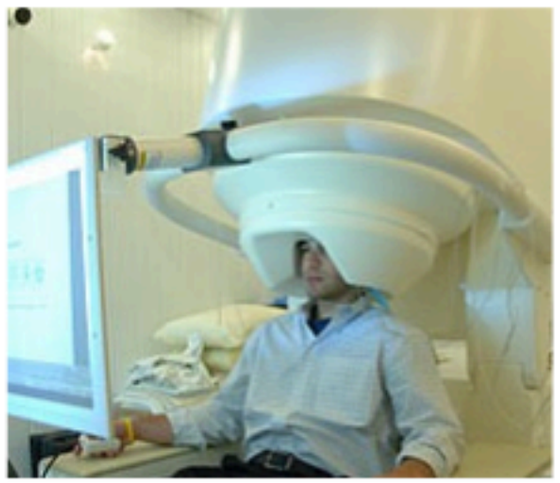
2. Dipoles (and thus neurons) need to have the same orientation

3. Can only measure radial dipoles. Neurons should not be oriented in parallel to the cortical surface.

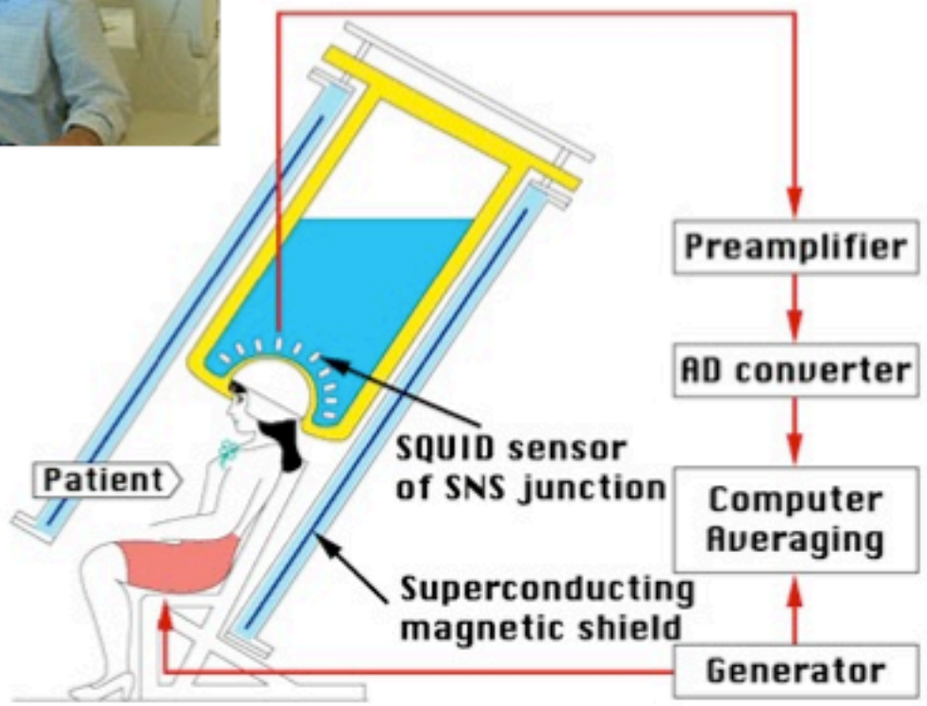


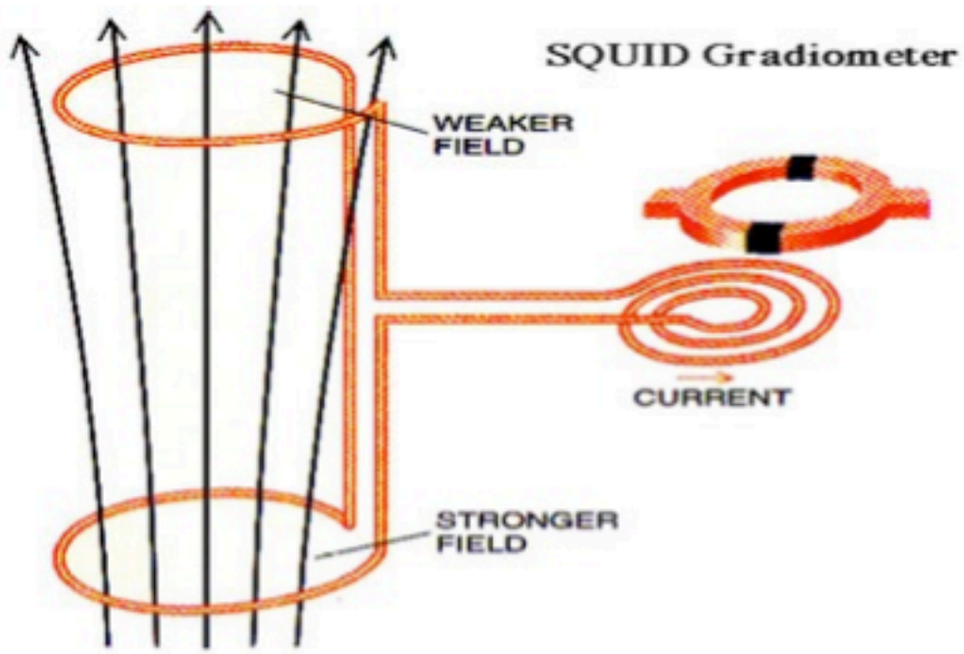
A lot of event-related neural activity does *not* meet these requirements; what does that mean for the interpretation of EEG?

Magnetoencephalography (MEG)



Measures large-scale magnetic activity using magnetometer coils placed on the head.

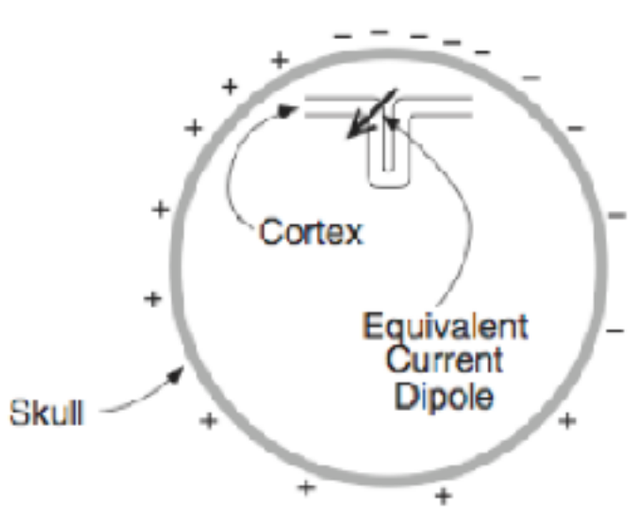


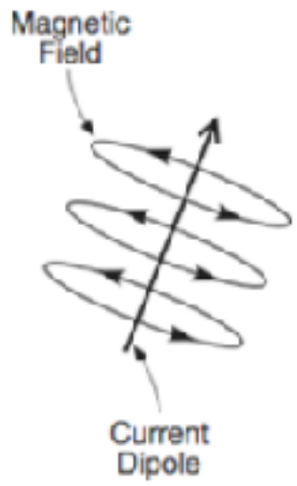


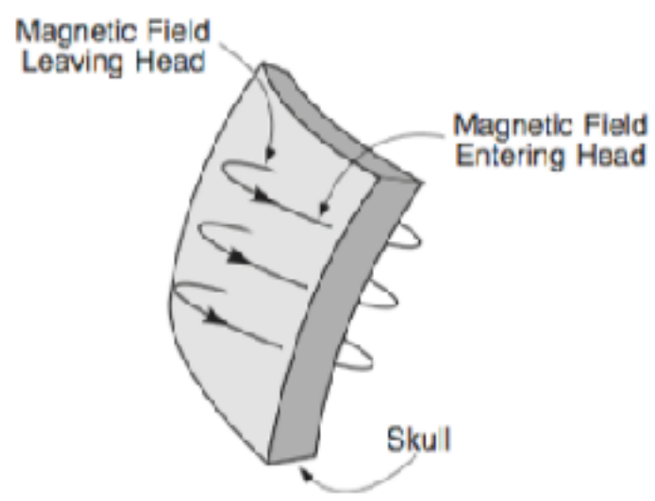
No blurring

SQUID: superconducting quantum interference device

Magnetoencephalography (MEG)

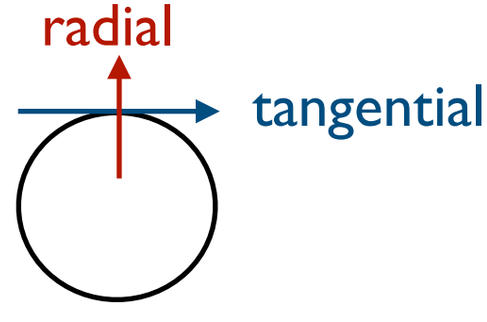






Right hand grip rule

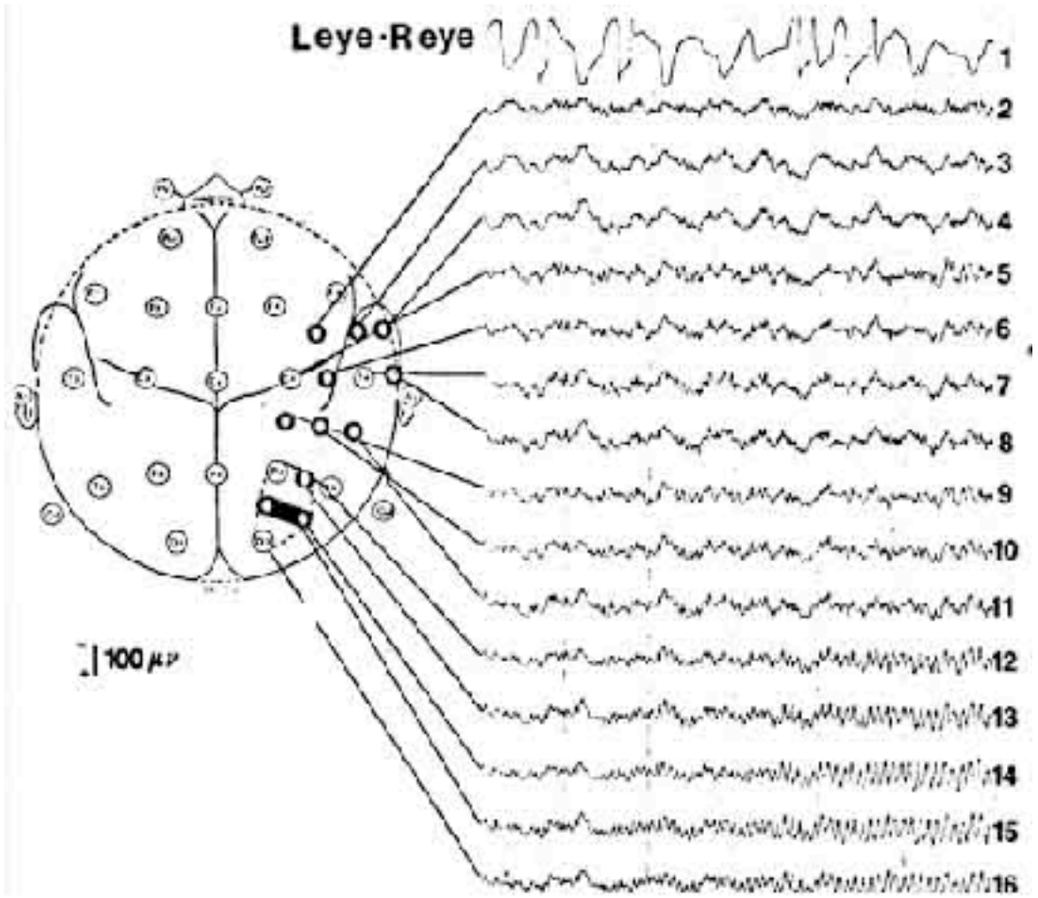
only measures magnetic fields that leave the skull, so ***cannot detect dipoles oriented perpendicular*** to the scalp
(= cannot measure radial dipoles)

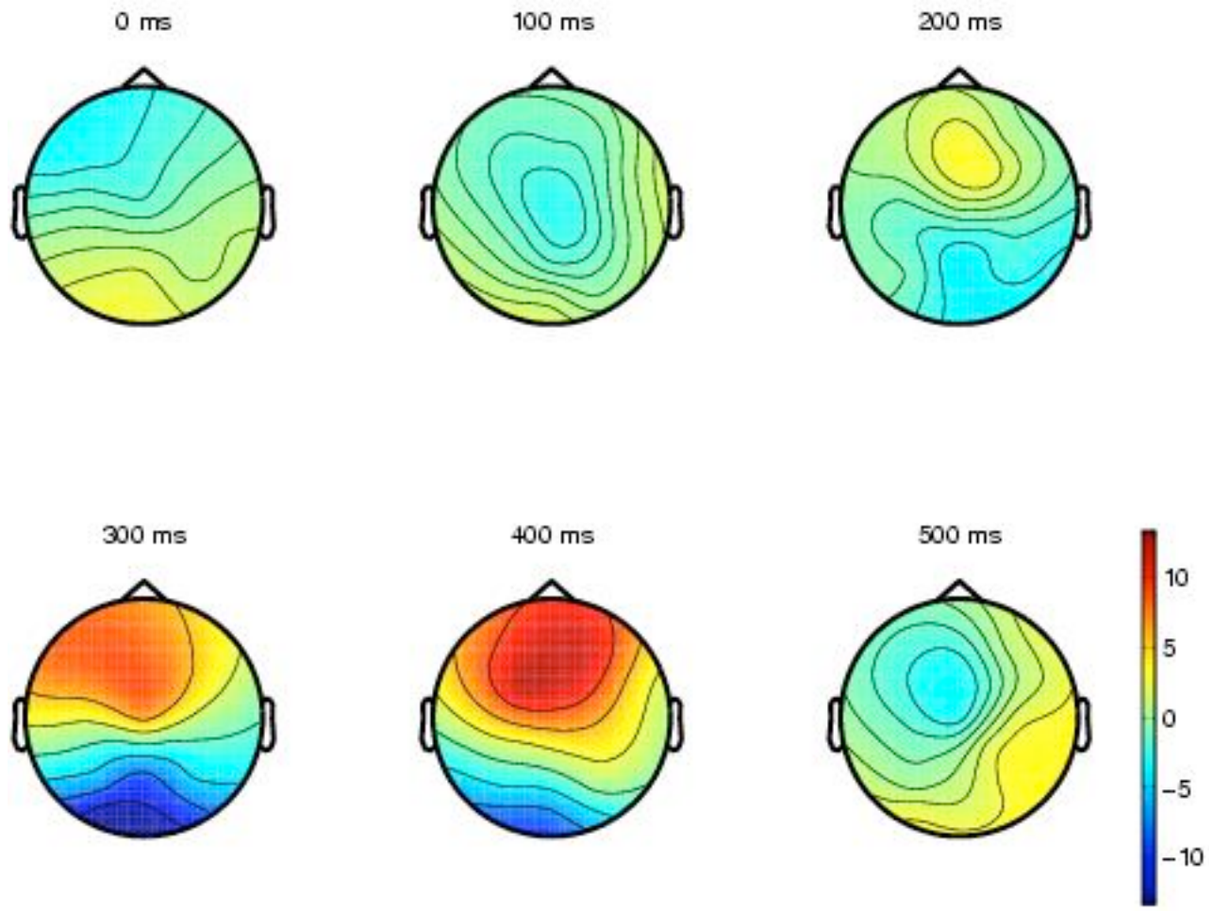


The MEG/EEG signal

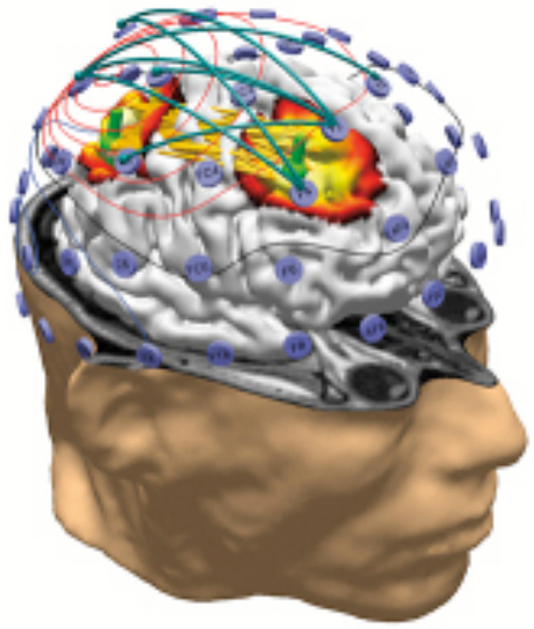
- MEG/EEG is primarily temporal, acquired at discrete moments in time, called **samples**
- The temporal resolution at which these samples are acquired is called the **sampling rate**
- MEG/EEG is also spatial, acquired across a varying number of **electrodes** (in MEG and analysis software these are often called **channels**)
- The number of electrodes in EEG can vary from anywhere between 1 to 256 channels, a typical number is **64**
- The signal in each of the channels can be plotted over time, as can be seen in an **event related potential (ERP)**
- The signal across channels for a particular time point can be plotted in a topographical map: the **topomap**

channels → topomap



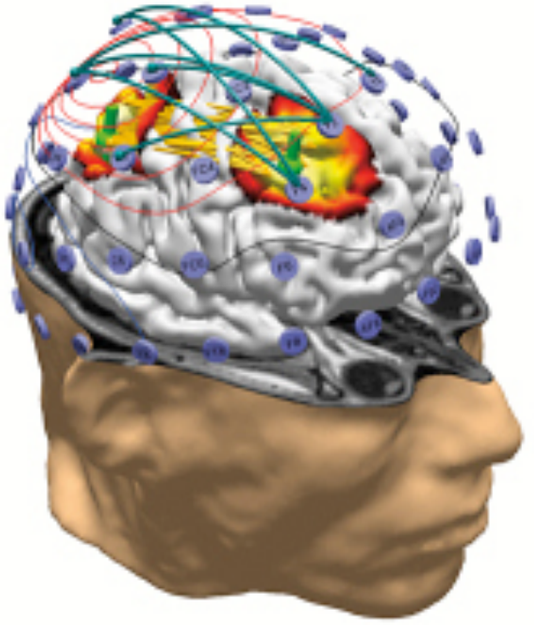


Forward problem versus Inverse problem



Forward problem
(relatively easy)

Inverse problem
(ill-posed)



MEG/EEG analysis

- Typically MEG/EEG is **pre-processed** to remove artifacts
- Next, there are many potential analytical approaches:
  - Event Related Potentials (won't talk much about **ERPs**)
  - Time-frequency representations (will talk a little about **TFRs**)
  - Multivariate approaches (will mostly talk about **MVPA**)
- Approaches are not mutually exclusive (can be combined)
- Ultimate goal in cognitive neuroscience is to characterize brain activity that subserves cognition and mental life (Hans Berger!)

Analysis software

- Standard main packages:
  - EEGLAB (user friendly, Matlab)
  - MNE (versatile, Python)
  - Brainstorm (Matlab, source reconstruction)
  - Brain Vision Analyzer (BVA, proprietary expensive, click and drag GUI)
- Some more dedicated toolboxes:
  - ERPLAB (Matlab, ERPs)
  - FieldTrip (Matlab, TFRs)
  - CoSMoMVPA (decoding, Matlab)
  - the Neural Decoding Toolbox (decoding, Matlab)
  - the ADAM toolbox (decoding/forward encoding, Matlab)
  - the PyMVPA toolbox (decoding, Python)

Questions?

Course overview

- **Day 1 (introduction)**
  - lecture 1: History and electrophysiological basis of EEG
  - lecture 2: Backward decoding models in MVPA: concepts and analytical approach
  - lecture 3: Advantages of MVPA, the temporal generalization method
  - lecture 4: Architecture of the ADAM toolbox and explaining the experiment of the practical
  - *Afternoon: practical*
- **Day 2 (advanced)**
  - lecture 1: Multiple comparisons, MVPA experimental design, mapping brain to brain/behavior
  - lecture 2: Forward encoding models in MVPA: how do they work, concepts and analytical approach
  - *Afternoon: practical, analyze your own data and/or a supplied dataset*

Lecture 2 -
Basic concepts and procedures in MVPA

- What is MVPA (decoding)? Univariate vs Multivariate (MVPA)
- Approaches and concepts:

  Train-test procedures, overfitting, K-fold cross validation, overfitting, classes, classifier, features, the decision boundary, weights, forward-transformed weights

- The confusion matrix, performance measures, balancing
- Balancing: undersampling and oversampling

Franciscus Donder's mental chronometry (1868)

- The first to measure reaction time in a laboratory

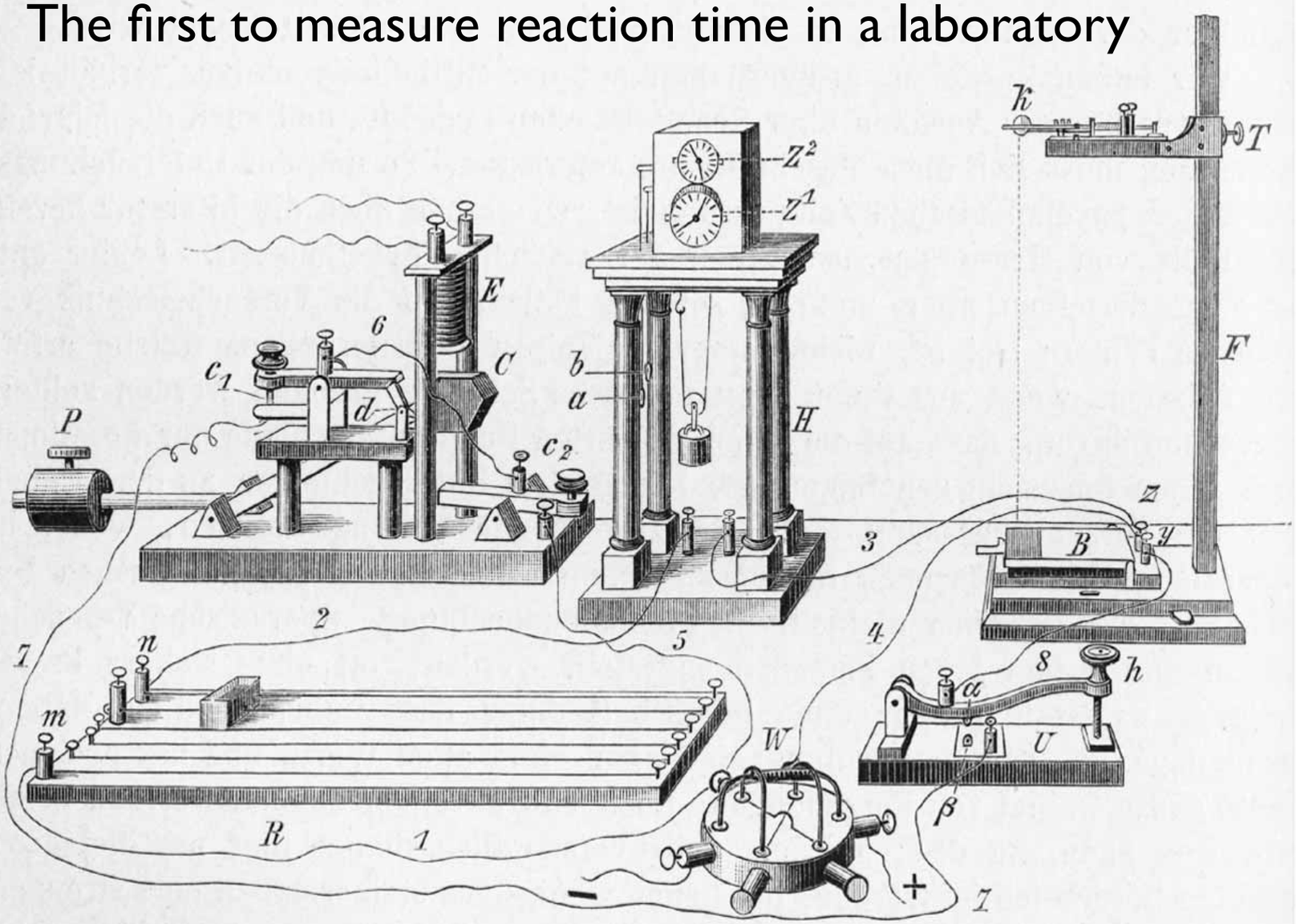
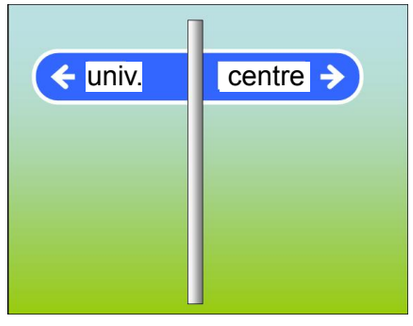
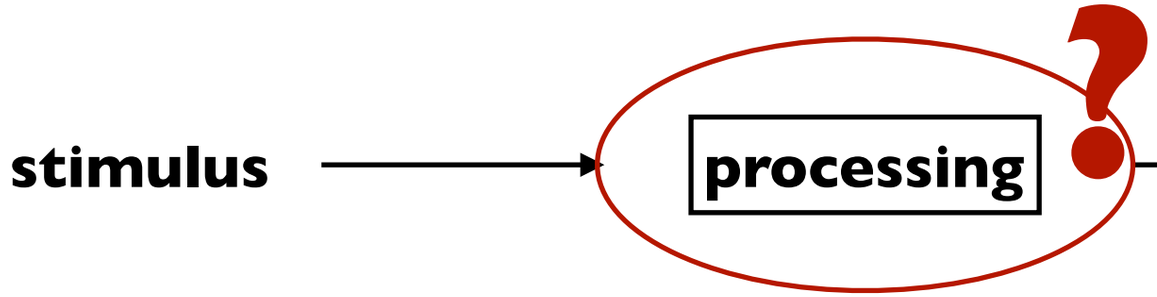


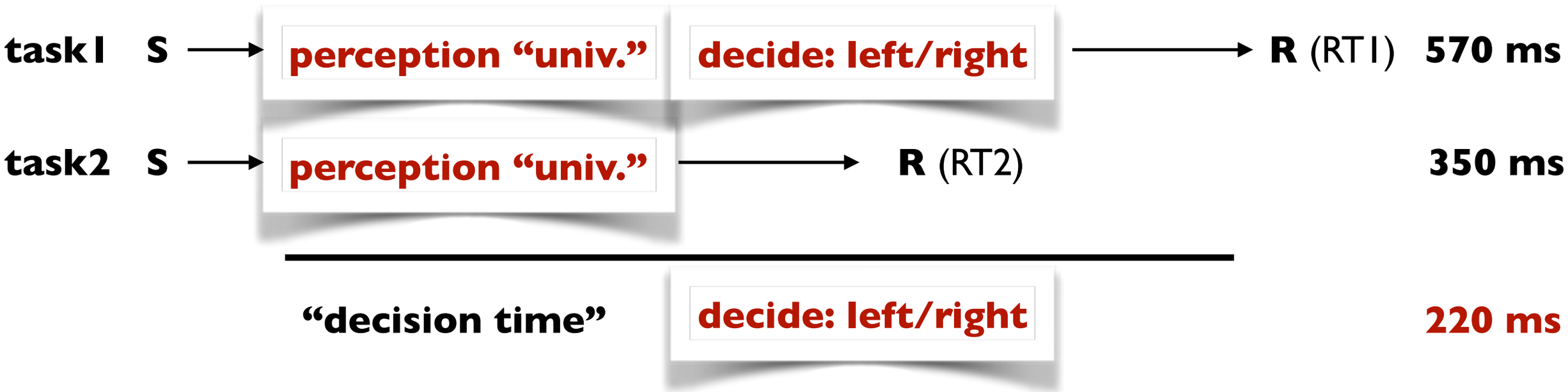
Fig. 215.

Subtraction



The difference between the task RTs should tell us the “decision time”!





Donders' subtraction methodology (reaction times)

"The idea occurred to me to interpose into the process of physiological time some new components of mental action. If I investigated how much this would lengthen the physiological time, this would, I judged, reveal the time required for the interposed term"

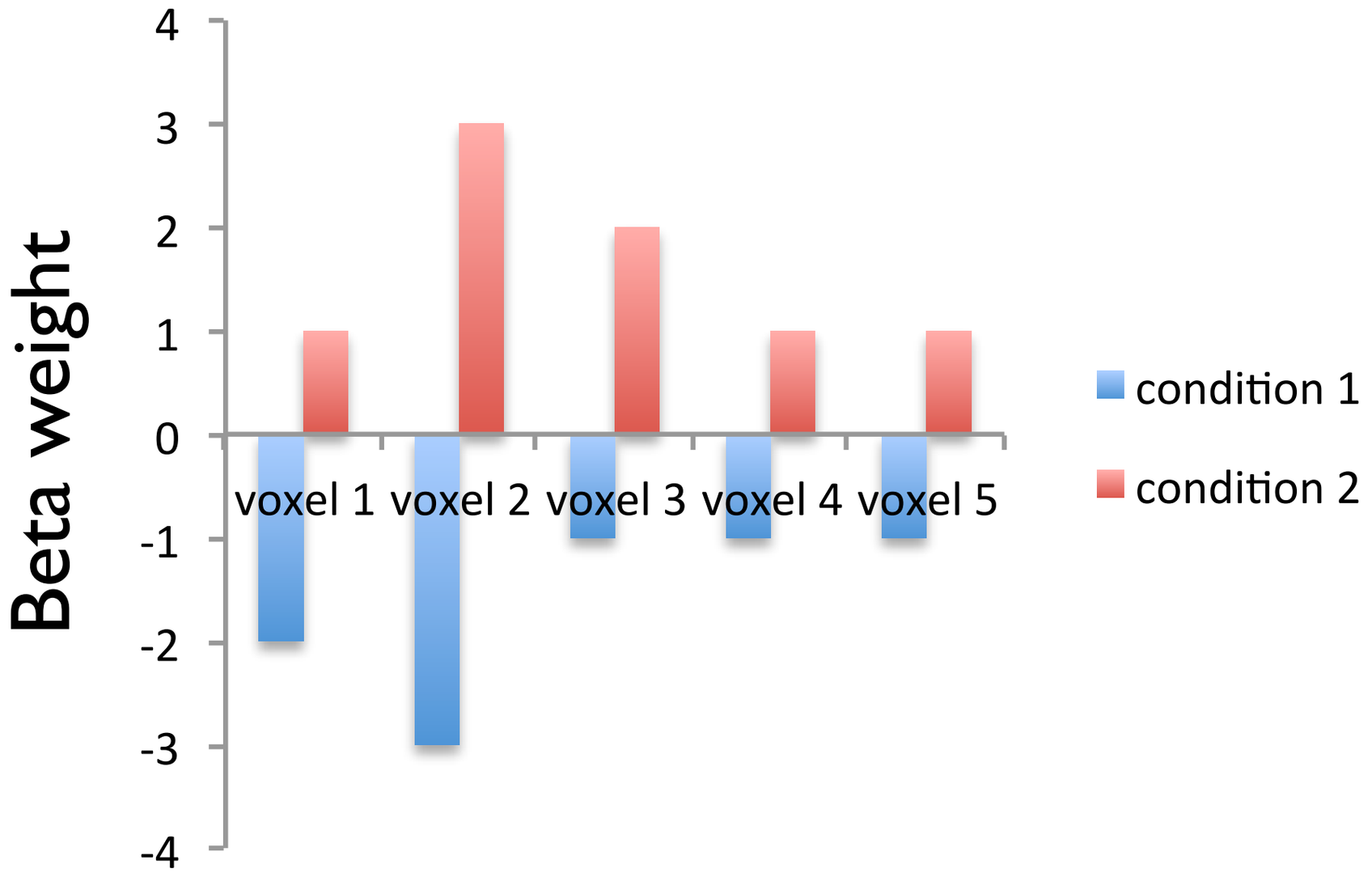
Donders (1868)

MVPA - what is it?

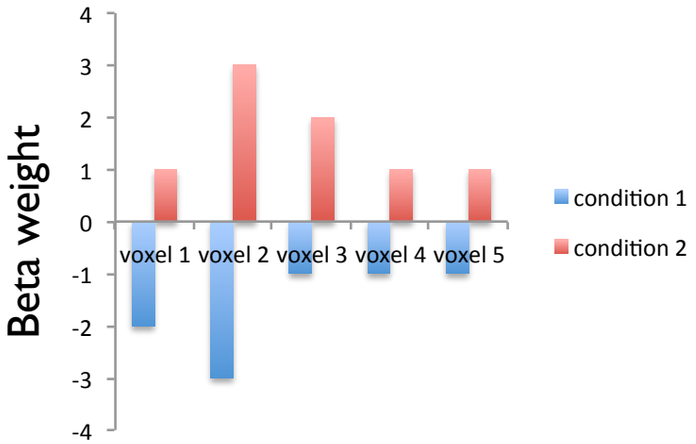
- Military Vehicle Preservation Association

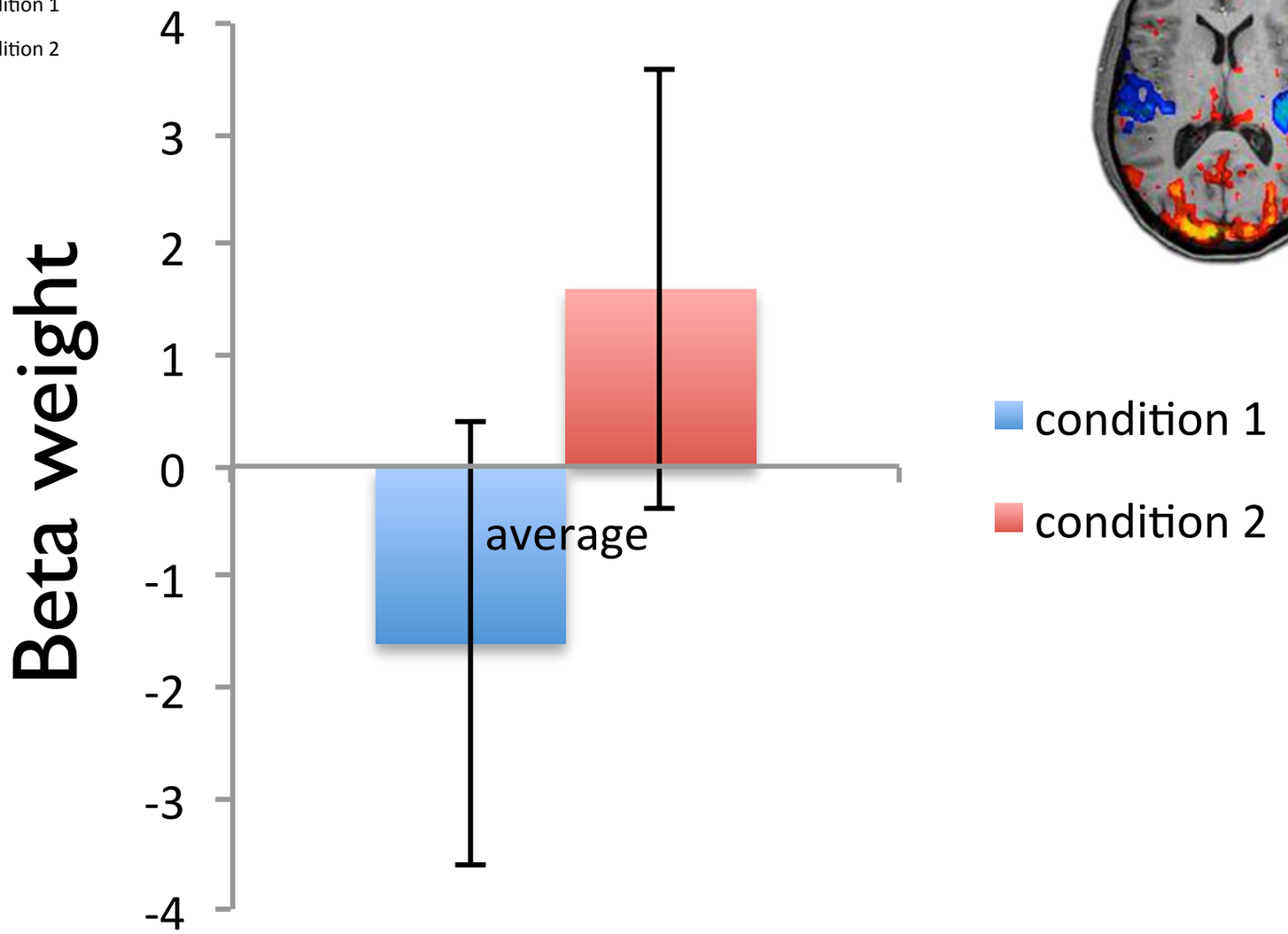
- Multivoxel pattern analysis (fMRI)
- Multivariate pattern analysis (EEG and fMRI)
- Univariate versus Multivariate

Univariate analysis

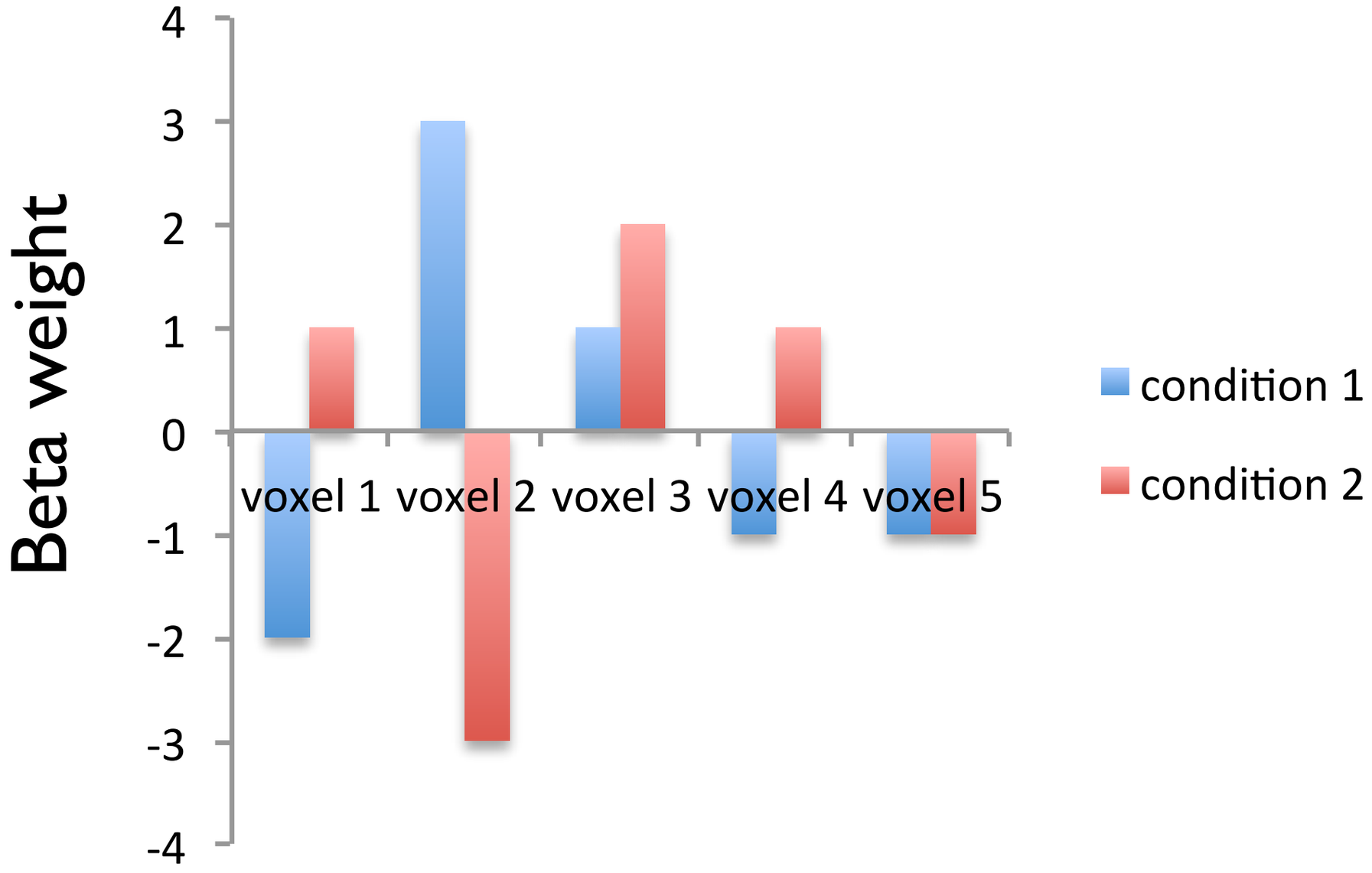


Univariate analysis

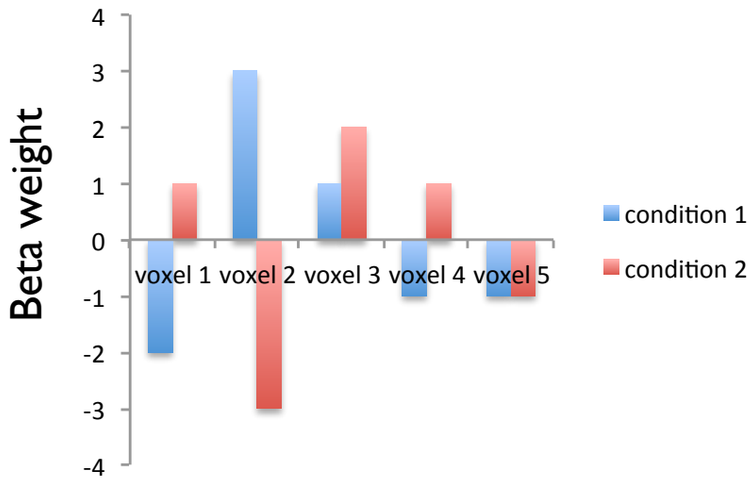


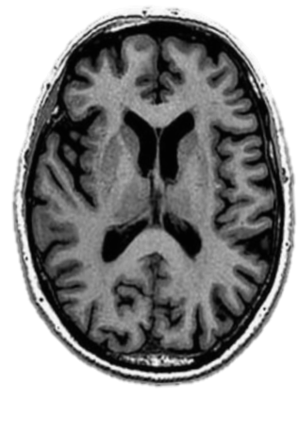


Univariate analysis

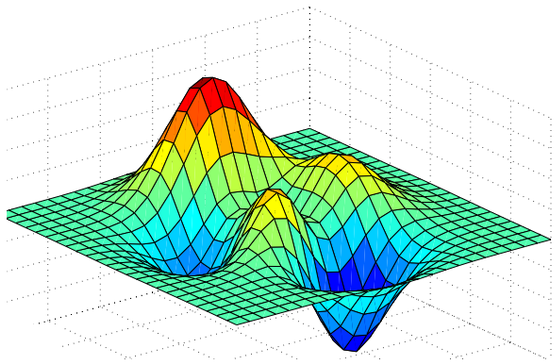


Univariate analysis

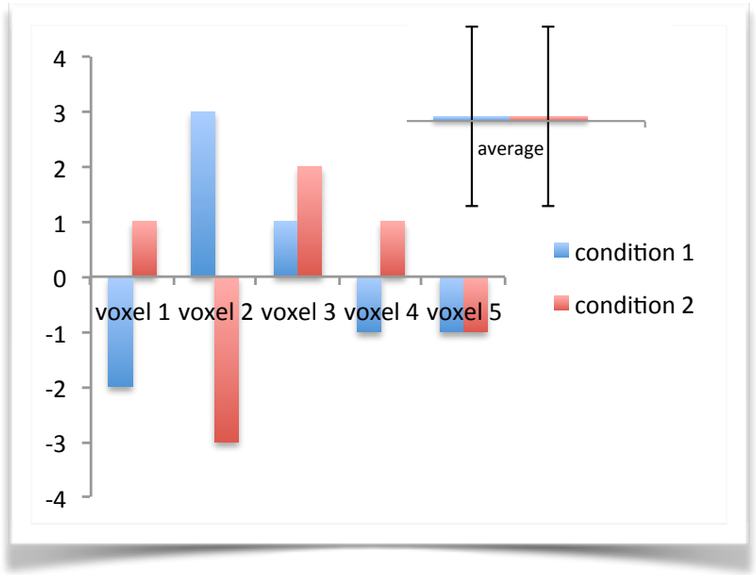


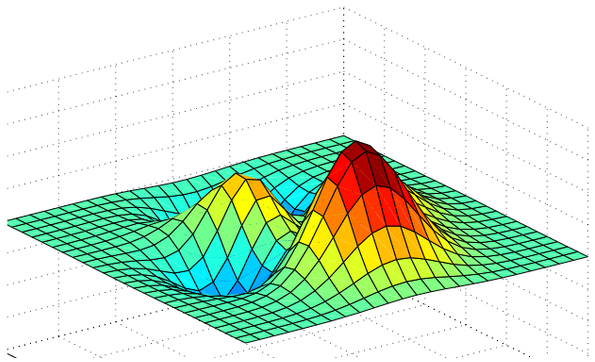


But multivariate signal may be consistent!

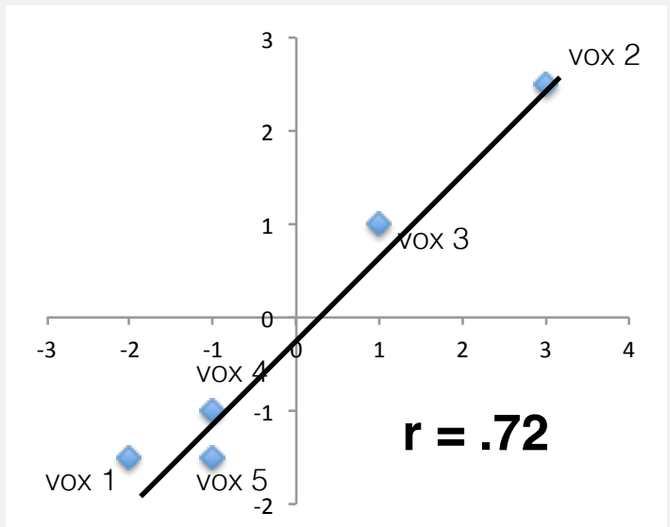


condition 1

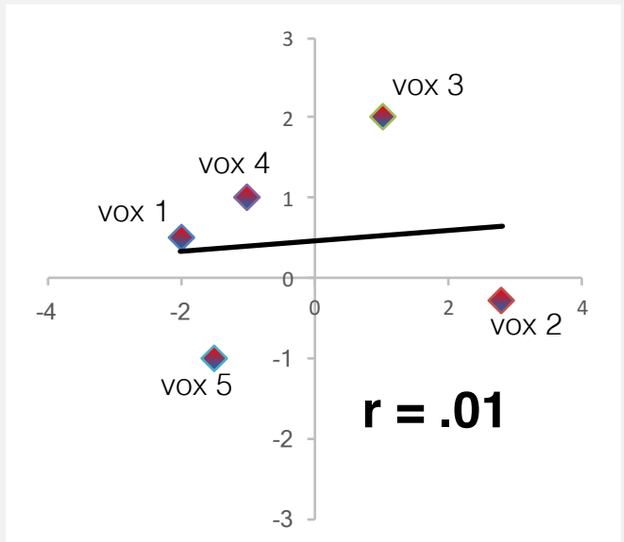




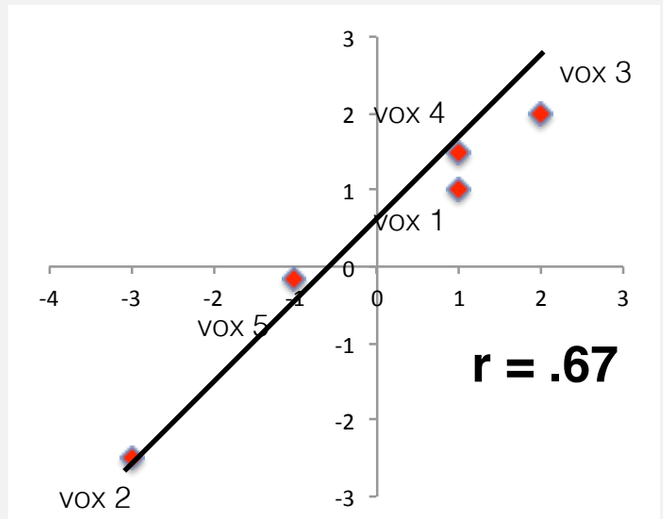
condition 2



condition 1 (trial 2)

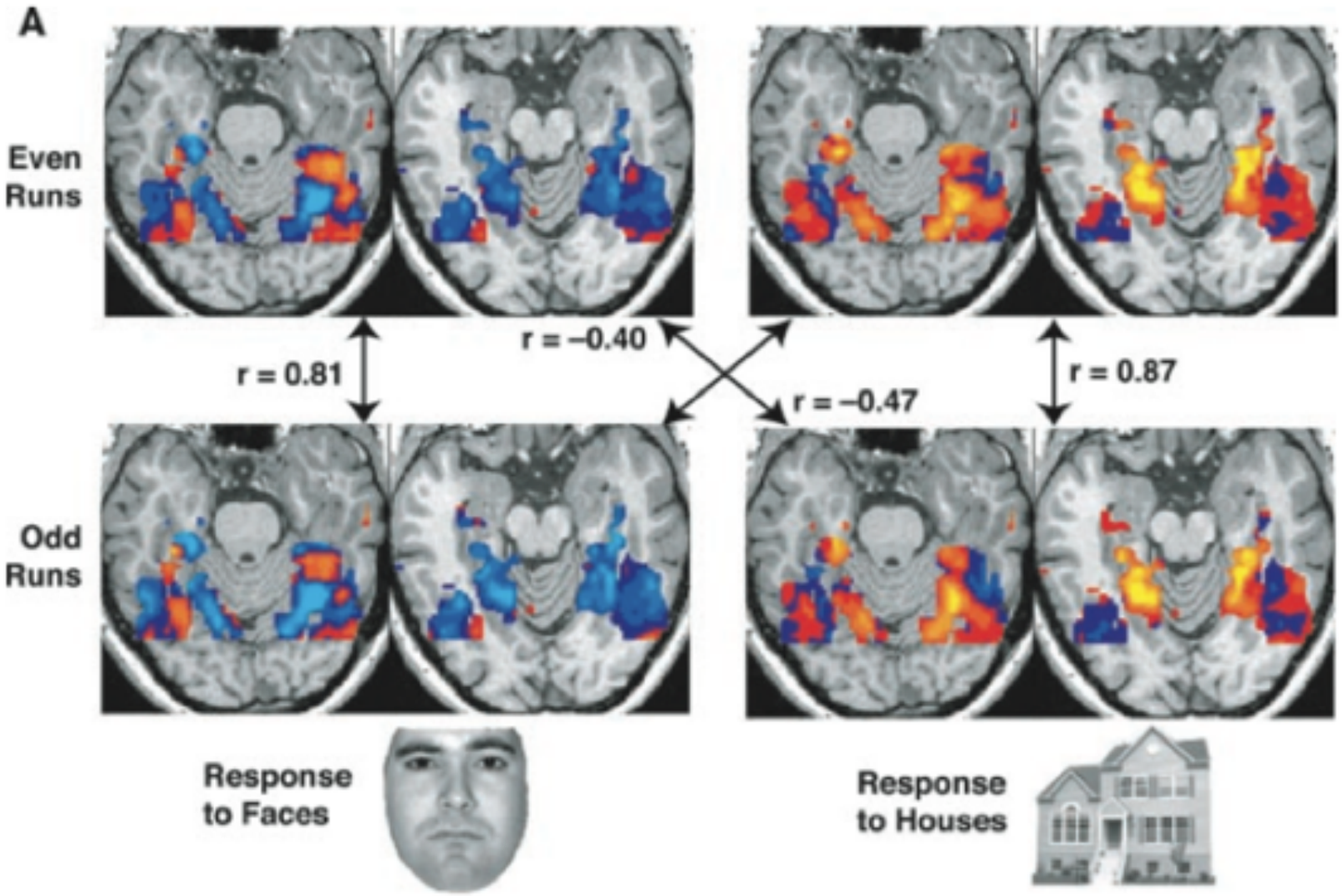


condition 2



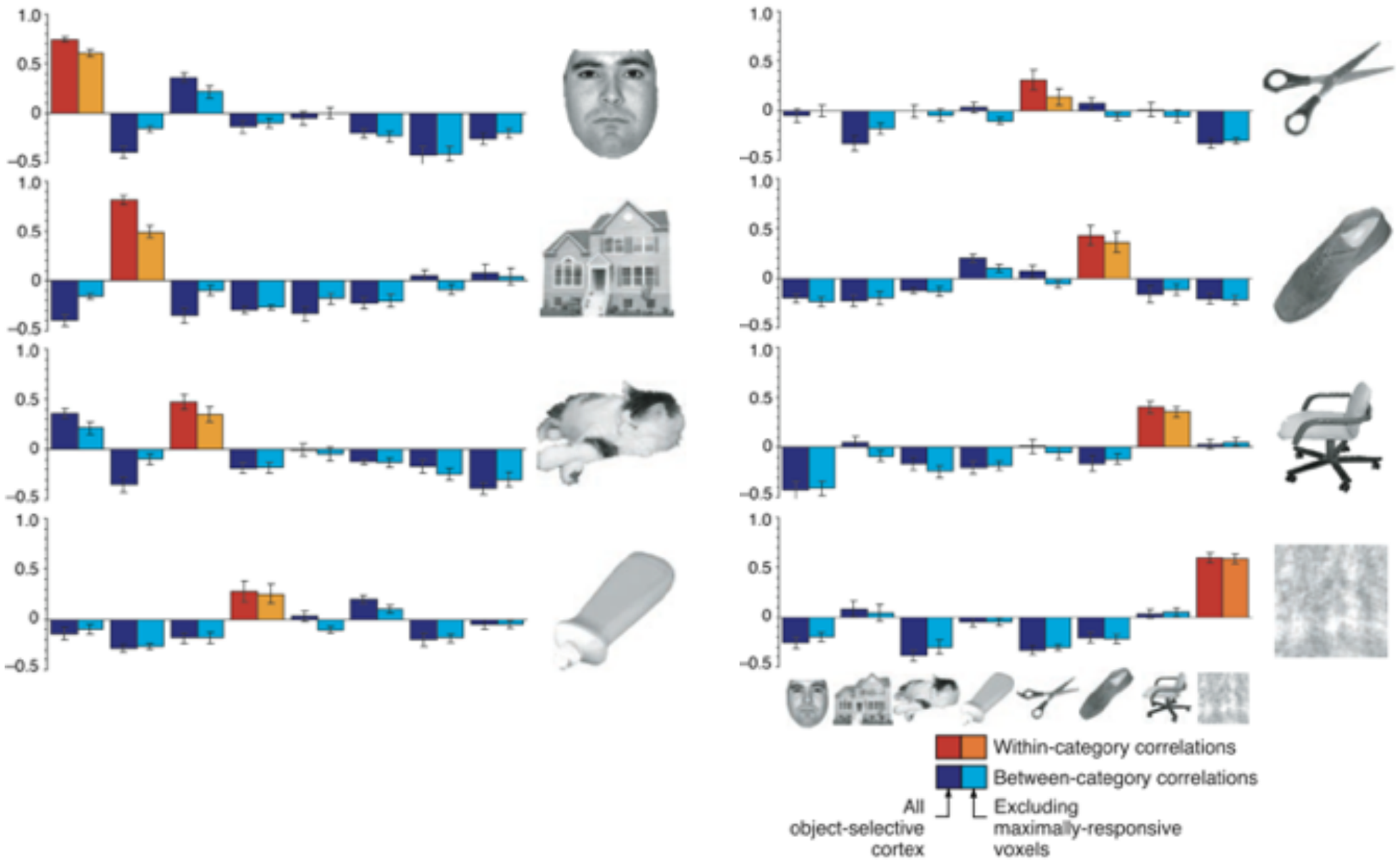
condition 2 (trial 2)

The first truly multivariate study in fMRI



Haxby, J.V., Gobbini, M. I., Furey, M. L., Ishai, A., Schouten, J. L., & Pietrini, P. (2001). Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science*, 293(5539), 2425–2430.

The first truly multivariate study in fMRI



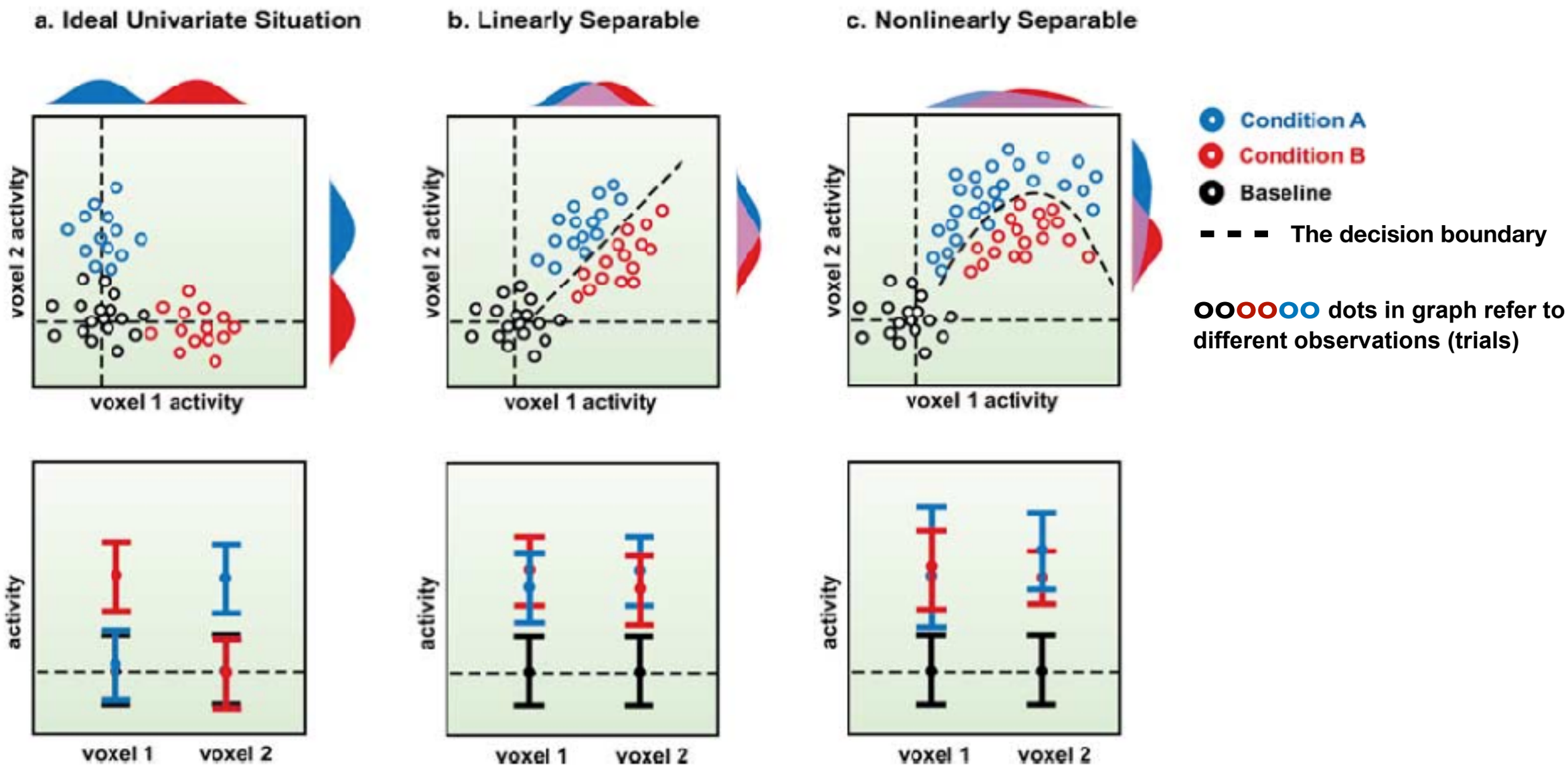
Haxby, J.V., Gobbini, M. I., Furey, M. L., Ishai, A., Schouten, J. L., & Pietrini, P. (2001). Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science*, 293(5539), 2425–2430.

How to identify multivariate patterns ('landscapes')?

- Representational Similarity Analysis (RSA, correlation) or Representational Dissimilarity Matrices (RDMs, 1-correlation)

- Train-test algorithms (e.g. LDA, SVM) :
  - linear classifiers
  - nonlinear classifiers

Univariate versus Multivariate analysis



Cox, D. D., & Savoy, R. L. (2003). Functional magnetic resonance imaging (fMRI) "brain reading": detecting and classifying distributed patterns of fMRI activity in human visual cortex. *NeuroImage*, 19(2), 261–270.

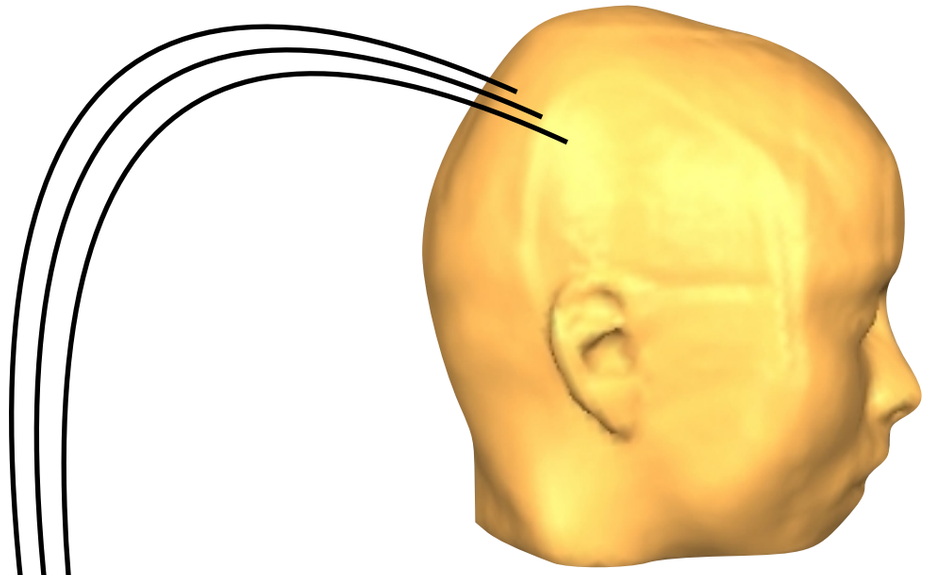
MVPA in EEG

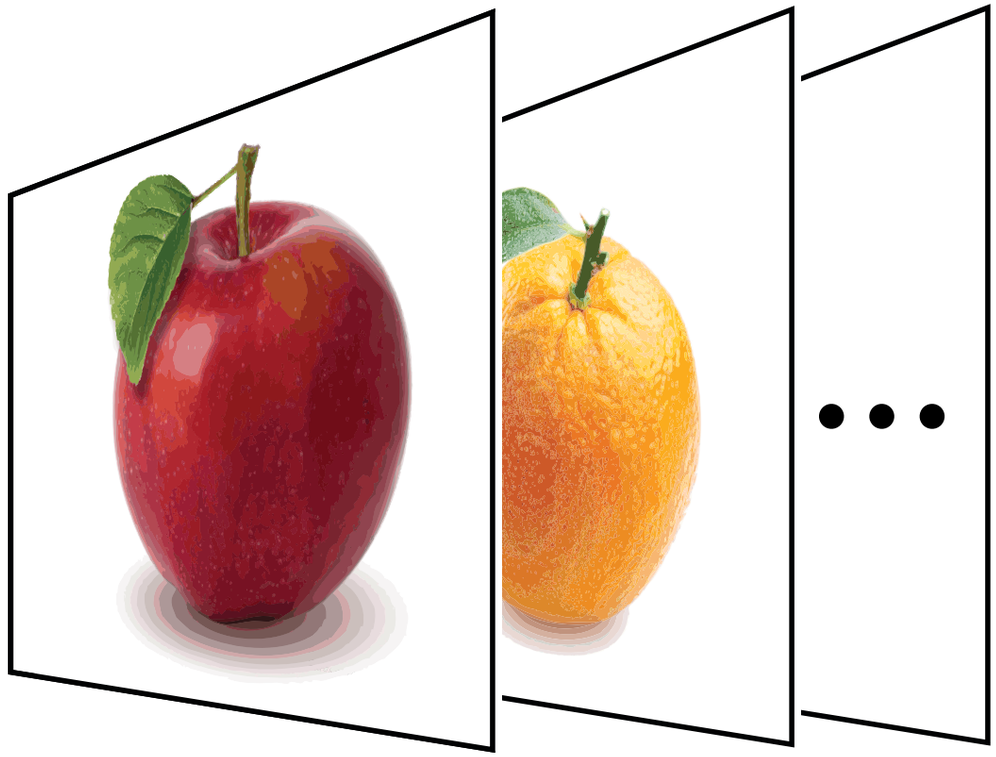
- Essentially the same thing as in fMRI, now using **electrodes** rather than **voxels** as **features**

- ERPs are conceptually similar to old school univariate GLM analysis in fMRI

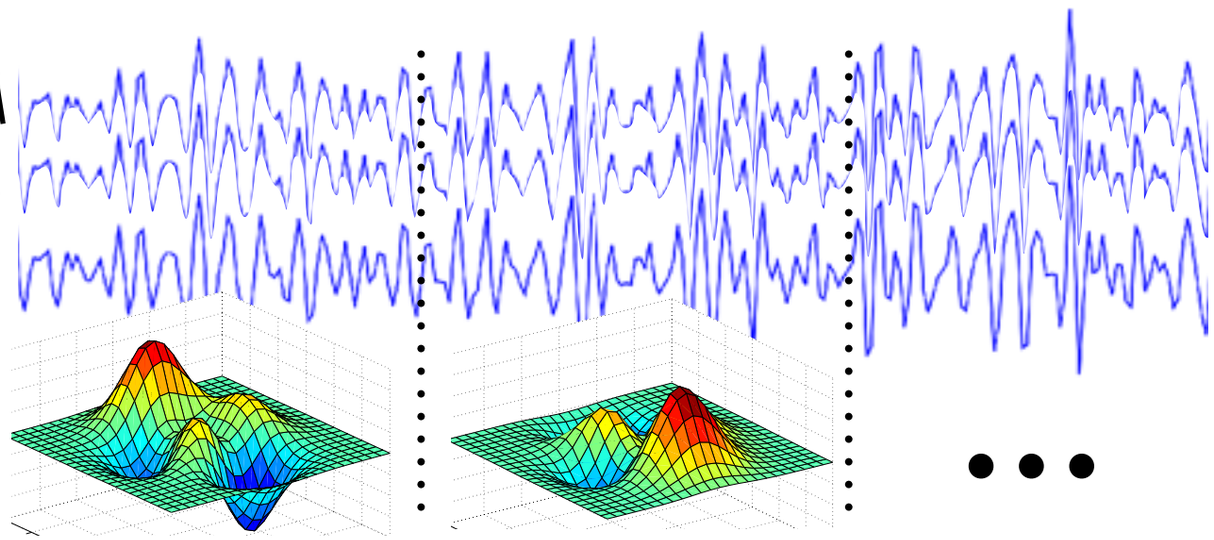
- Using MVPA, you can identify whether patterns of activation across the brain are different between conditions (even when specific ERPs would look highly similar)

Experiment: find a Neural Correlate of processing Apples and Oranges (NCAO)



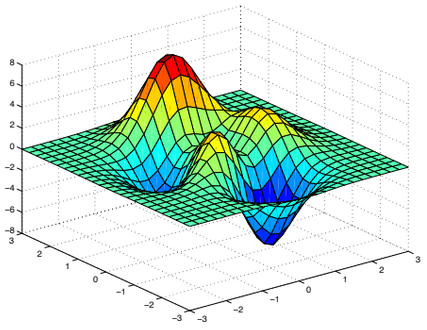


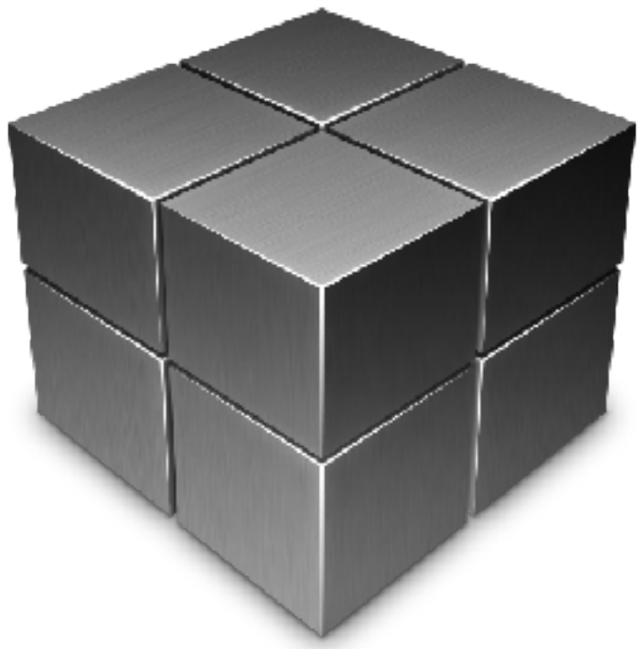
trial 1……trial 2……trial n



trial 1……trial 2……trial n

Training

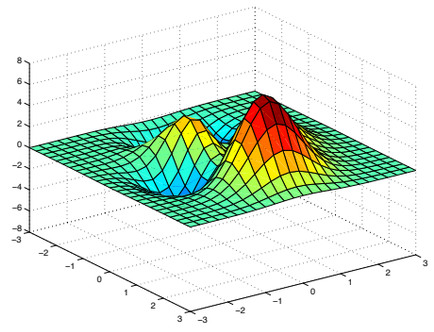


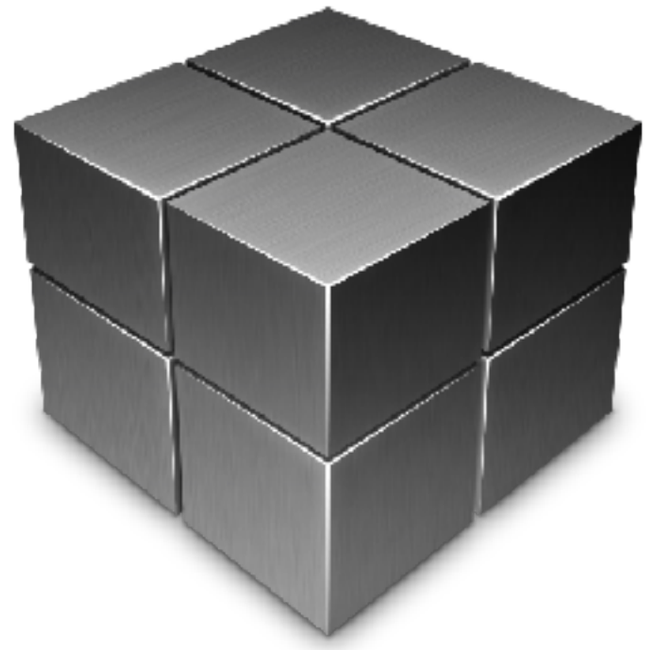


“apple”

Trial 1

Training

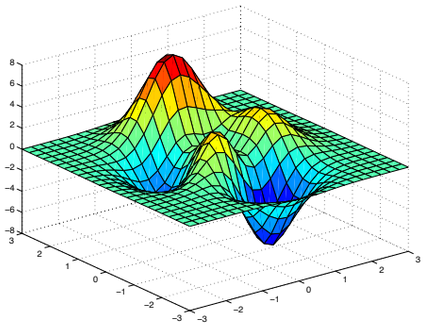


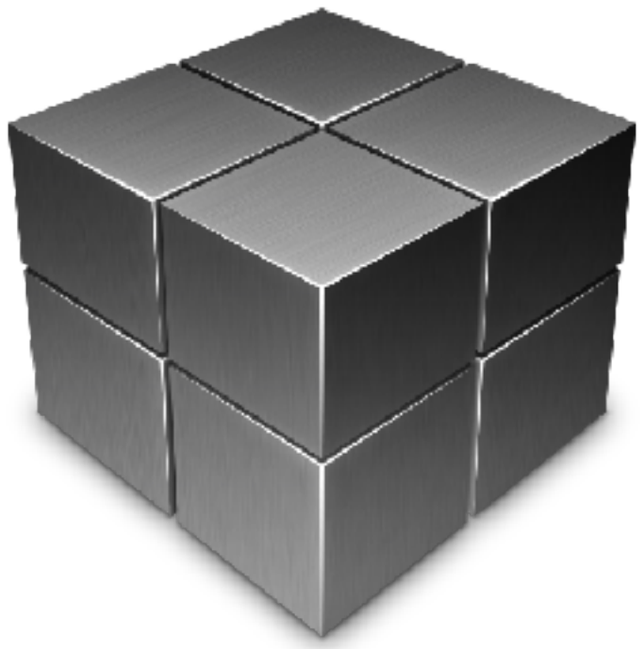


“orange”

Trial 2

Training

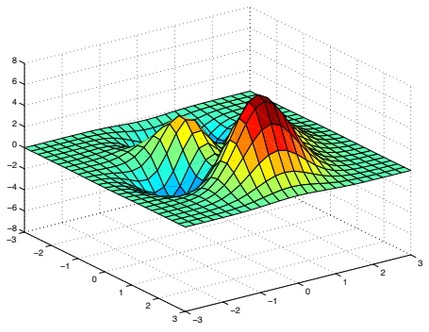


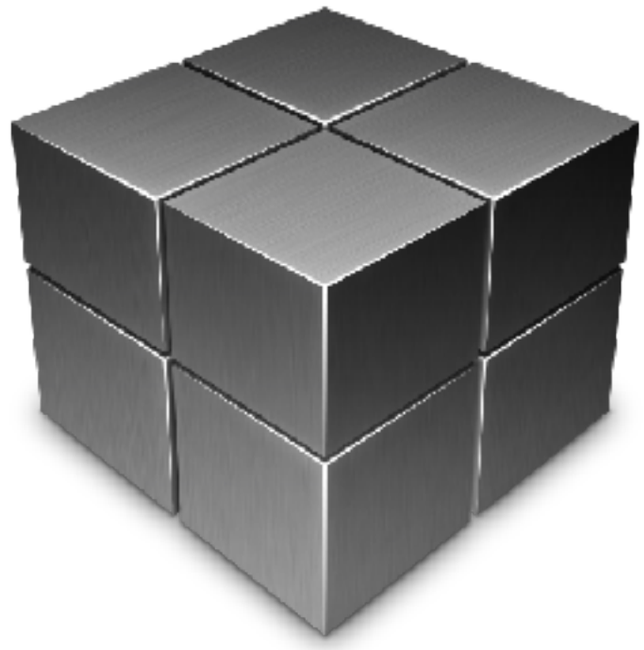


“apple”

Trial 3

Training

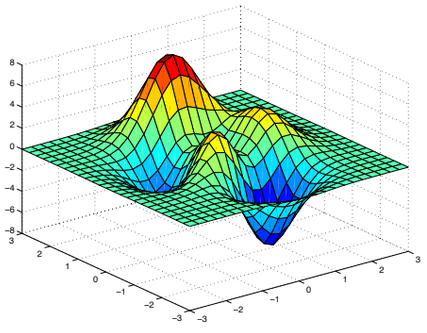


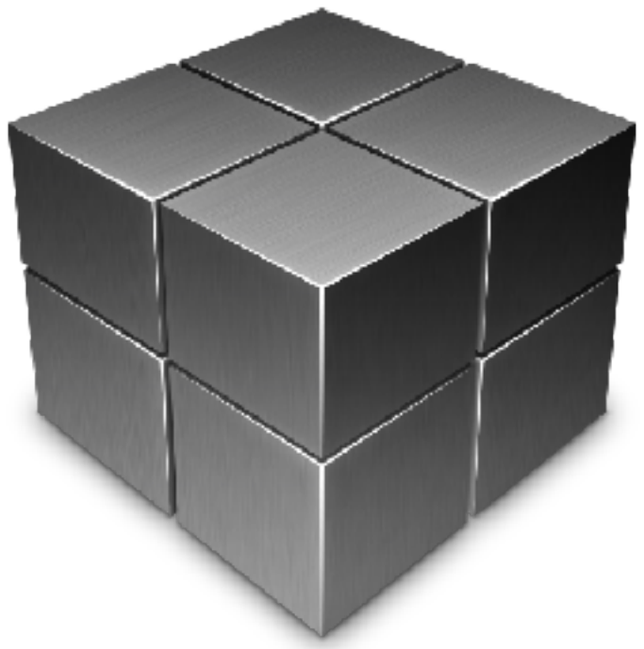


“orange”

Trial 4

Training

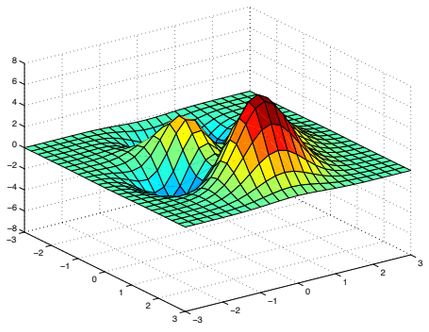


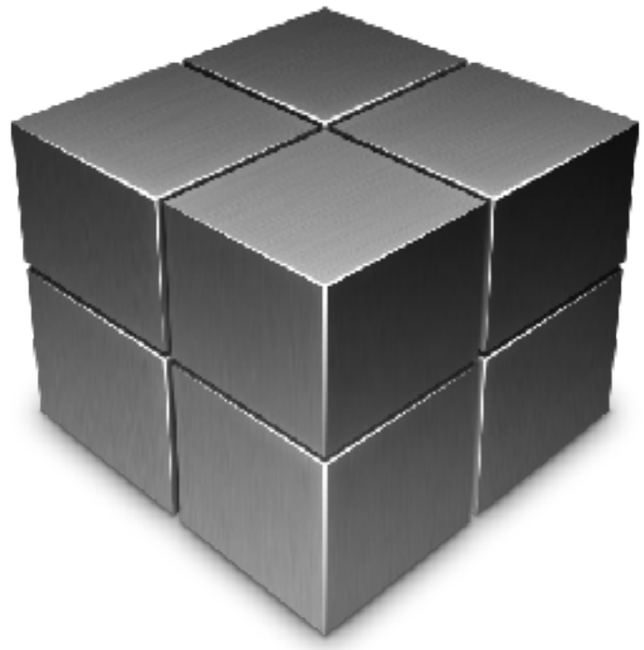


“apple”

Trial 5

Training

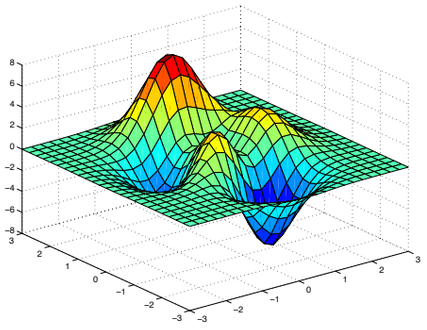


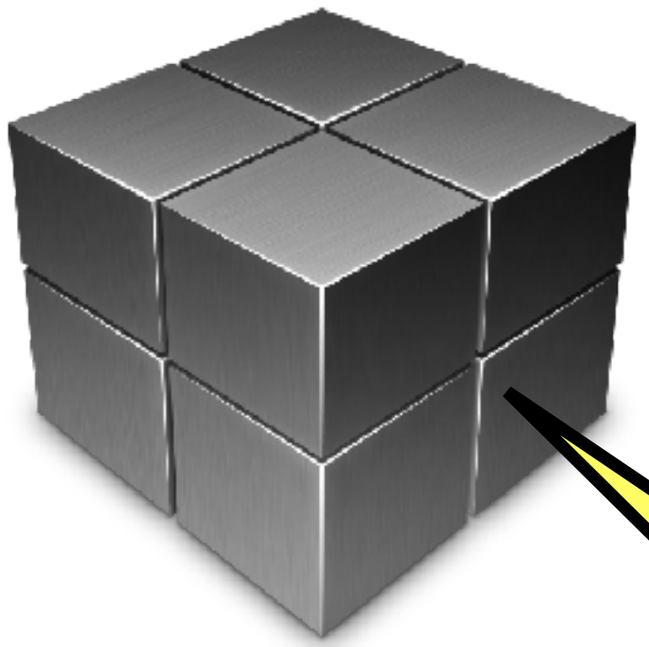


“orange”

Trial 6

Testing



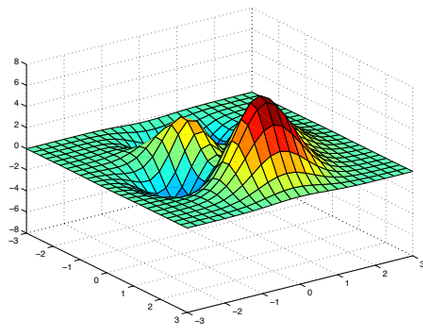


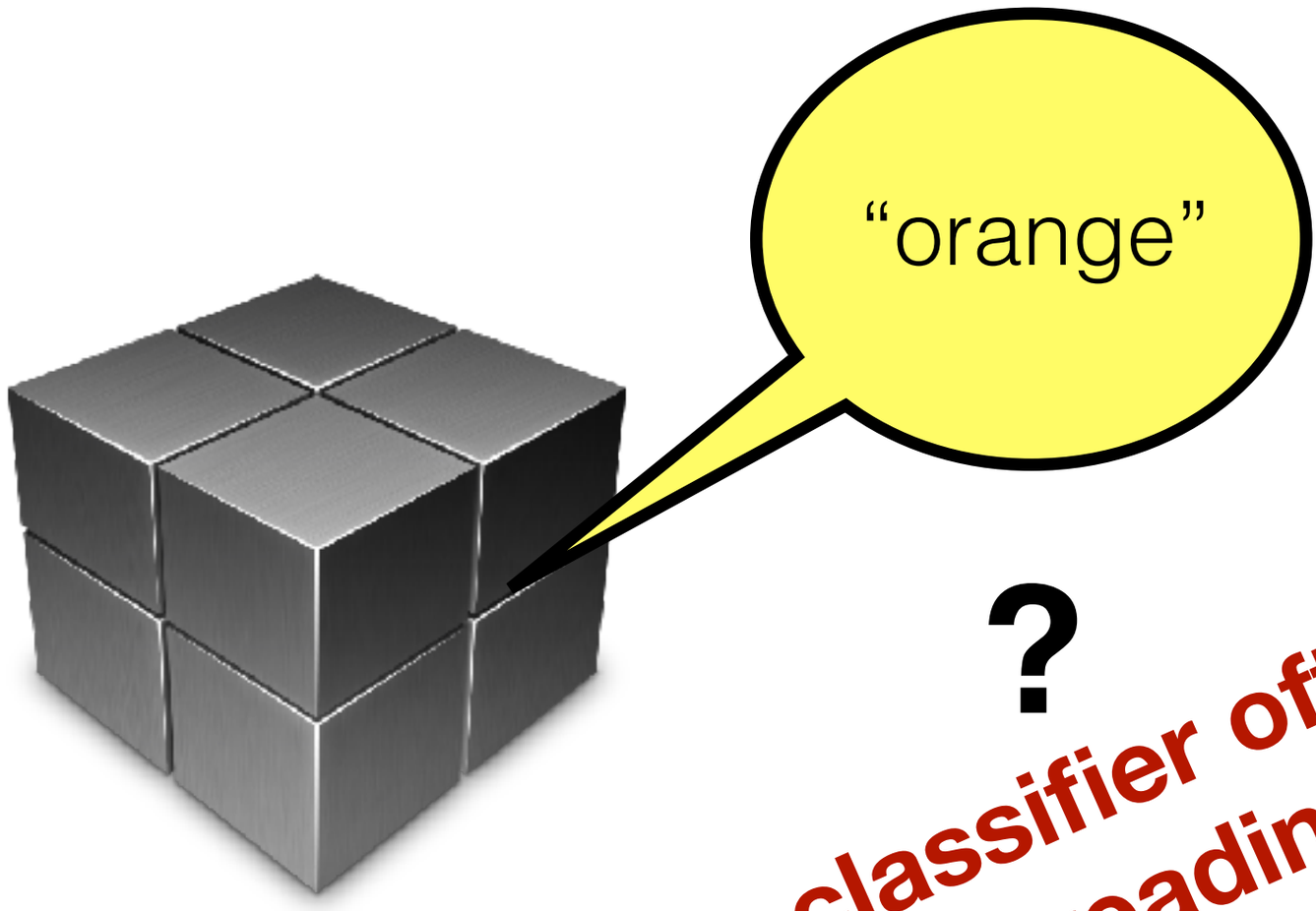
?

“apple”

Trial 1 (testing set)

Testing

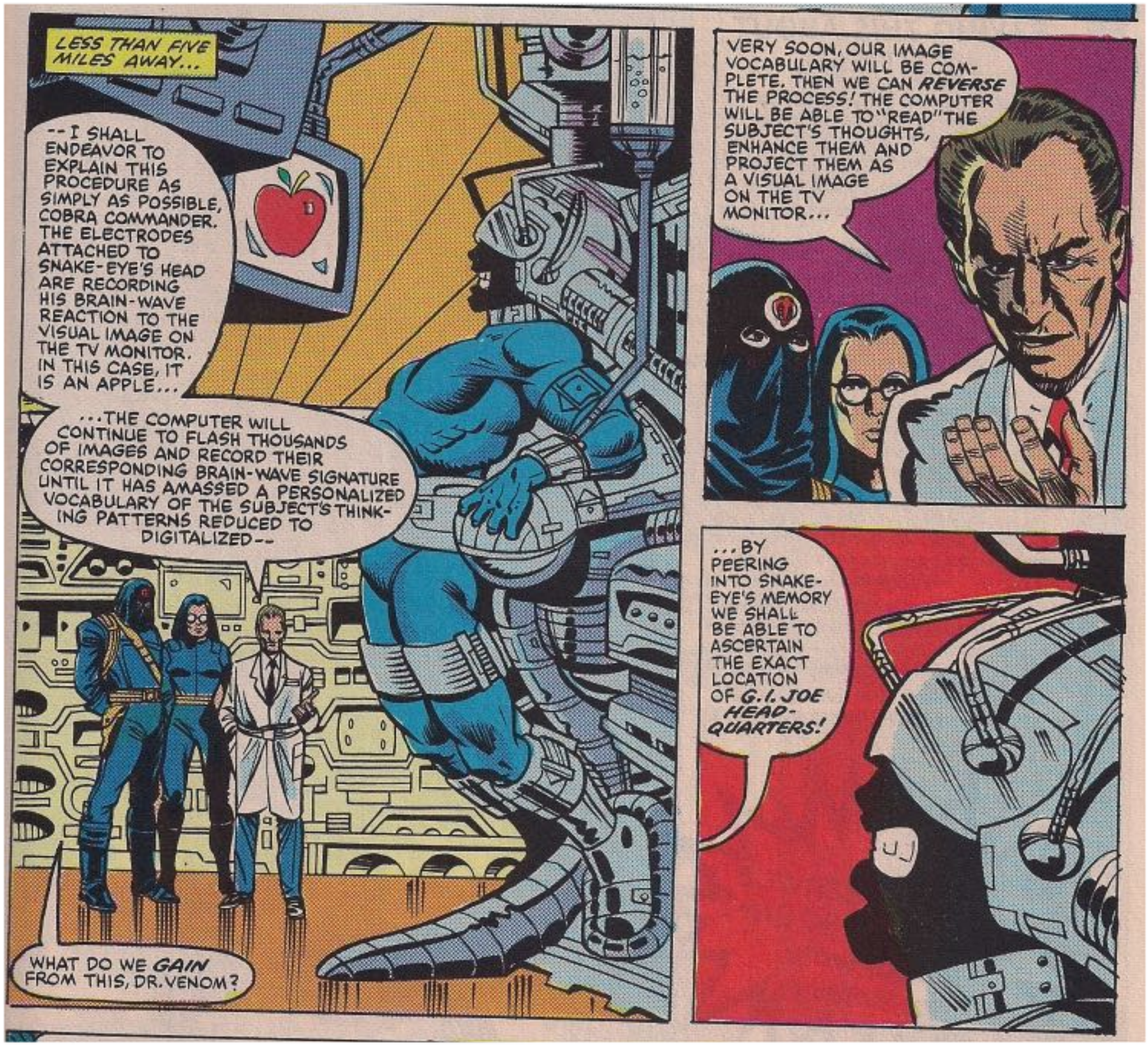




Trial 2 (testing set)

Predictions by classifier often pitched as ‘brain reading’

From a 1983 comic book



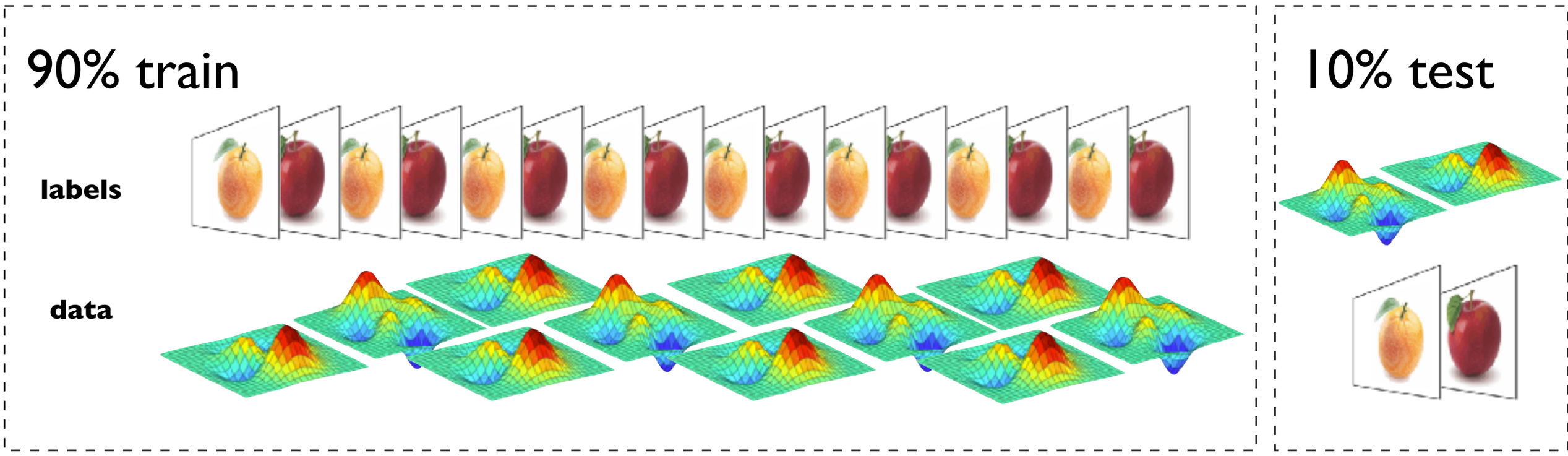
Decoding: computing classification performance on test data

Train-test schemes

- Your training set should always be *independent* from your testing set (never use the same data for training and for testing)

- Two ways of doing this:
  - Use the same dataset for training and testing, while never training on the same data as you tested on:
    k-fold cross-validation
  - Use two different datasets, a **separate training** set for training, and a different one for testing

K-fold cross-validation



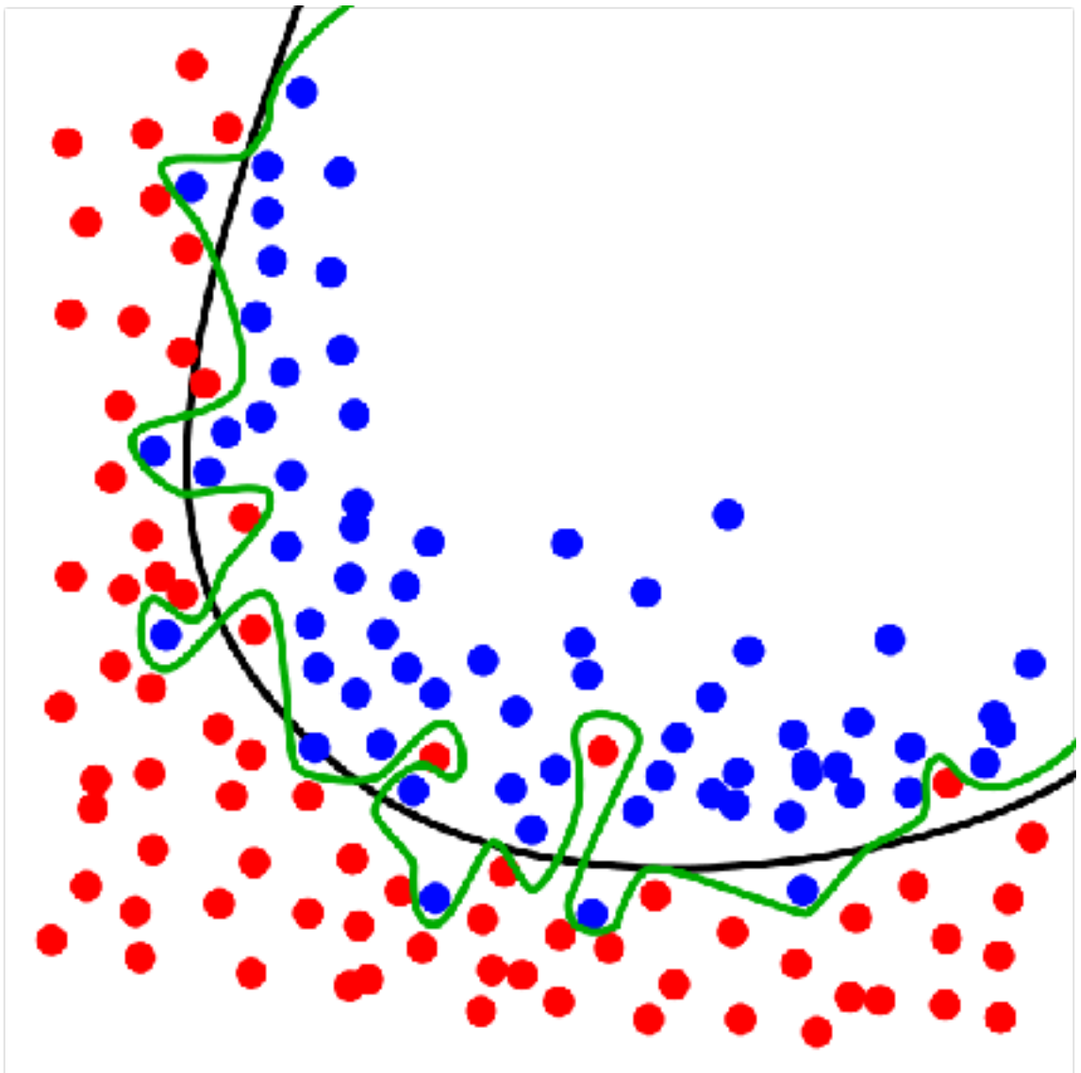
compute accuracy on test data

WHY?

Repeat until all data has been tested once

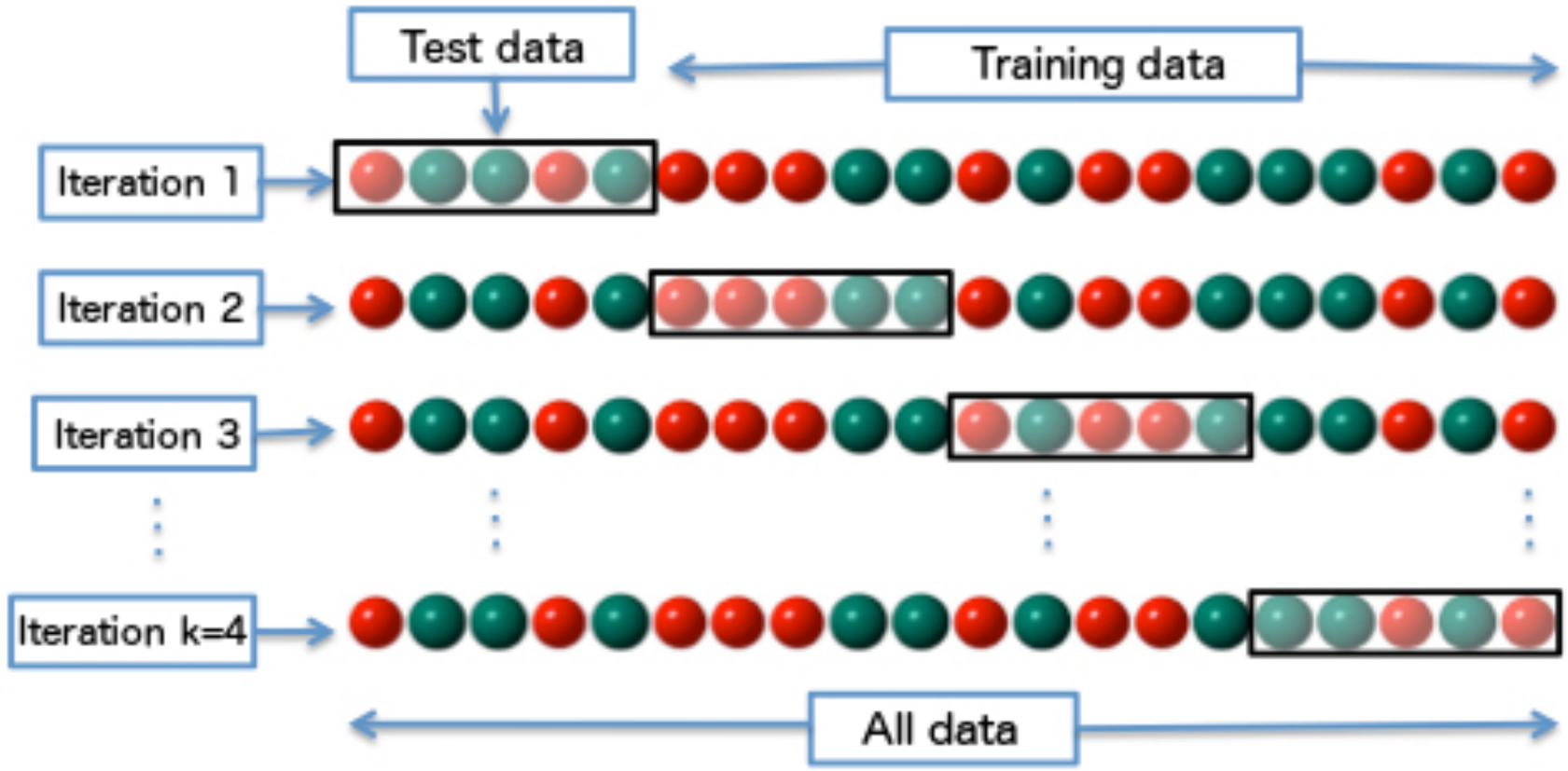
and **compute classification performance across folds**

Keep train and test data independent to prevent overfitting



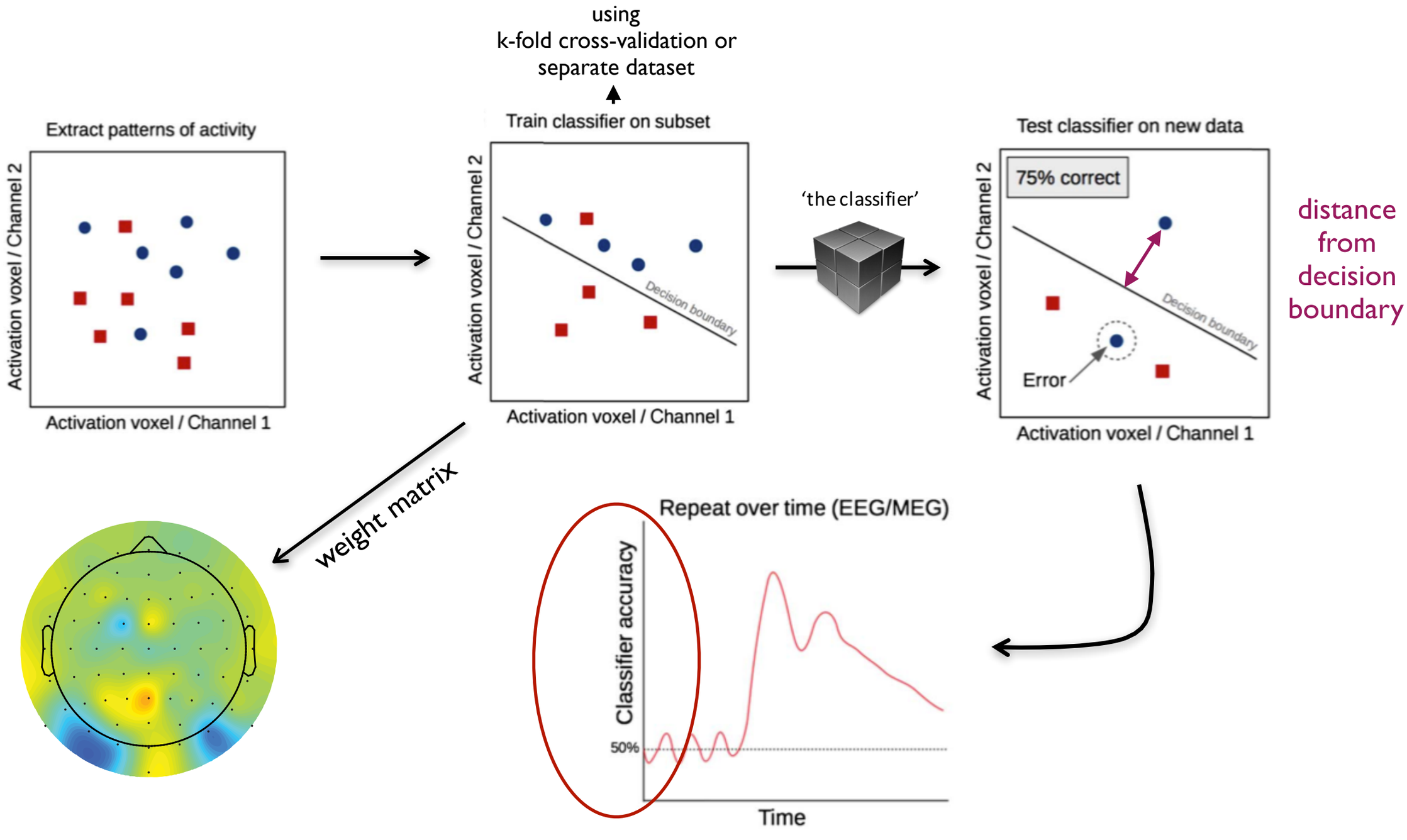
Conceptually similar to double dipping
(= using the same data for data selection and statistical testing)

Why is it called *K*-fold cross-validation?



- If ***k* = 4**, the classification will be ran **4 times**, each time training on **75%** of the data and testing on **25%** of the data
- The result will be averaged across the **4 folds**, thus testing all data exactly once
- If ***k*** is equal to the number of trials in your dataset, the method is called a **leave-one-out cross-validation procedure**
- Alternatively, you can collect a **separate** dataset for training your classifier (akin to a '*mapper*' in fMRI analysis)

What do EEG decoding results look like?

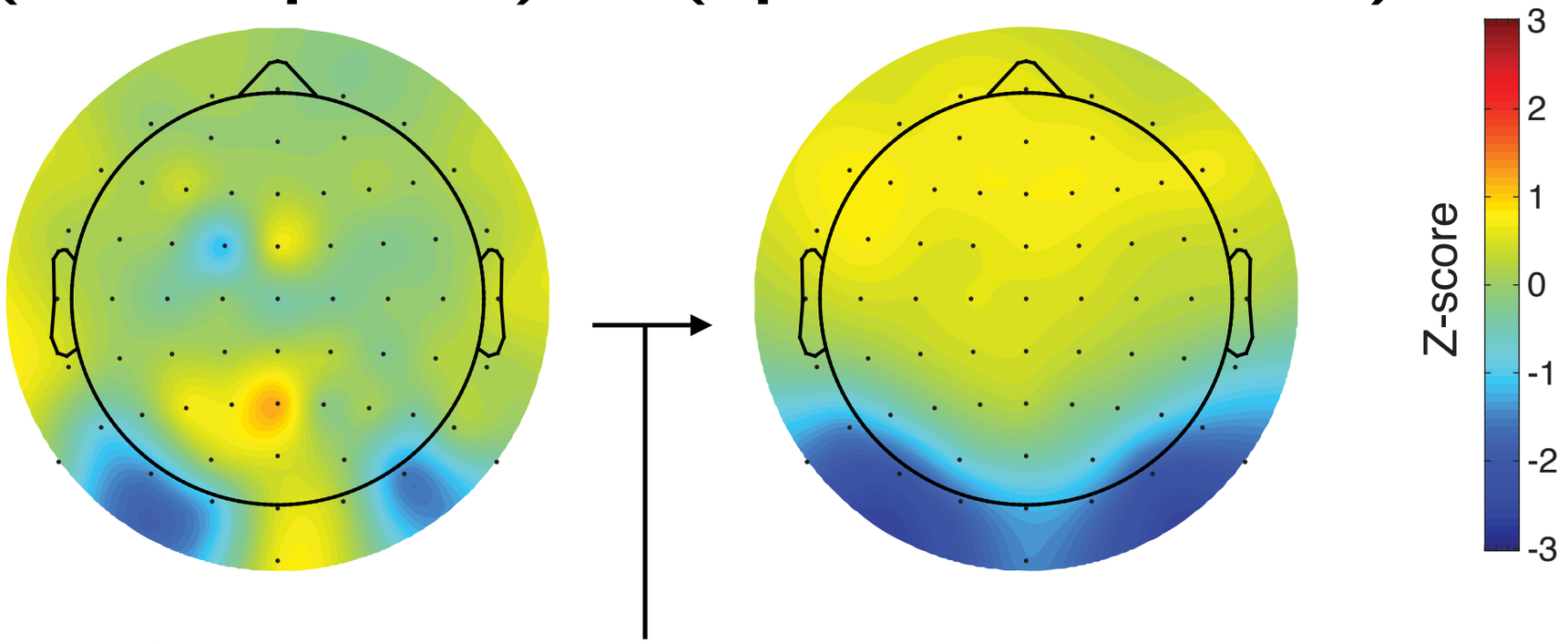


Grootswagers, T., Wardle, S. G., & Carlson, T. A. (2017). Decoding Dynamic Brain Patterns from Evoked Responses: A Tutorial on Multivariate Pattern Analysis Applied to Time Series Neuroimaging Data. *Journal of Cognitive Neuroscience*, *29*(4), 677–697.

Weight matrix

Weights (non-interpretable)

Activation pattern (equivalent to univariate)



Two options:

1) weights * covariance matrix = activation pattern
2) weights * correlation matrix = class/correlation separability map

Haufe, S., Meinecke, F., Goergen, K., Daehne, S., Haynes, J.-D., Blankertz, B., & Biessgmann, F. (2014). On the interpretation of weight vectors of linear models in multivariate neuroimaging. *NeuroImage*, 87, 96–110.

Performance measures: the confusion matrix

A confusion matrix is a table that is used to describe the performance of a classification model (or "***classifier***") on a set of test data for which the true values are known.

n=165	**Predicted: NO**	**Predicted: YES**
Actual: NO	correct rejections 50	false alarms 10
Actual: YES	misses 5	hits 100

Performance measures: overall accuracy

The percentage correctly classified instances across all data. Highly sensitive to bias (especially when classes are not balanced)
should not be used, ever.

n=165	**Predicted: NO**	**Predicted: YES**
Actual: NO	0	60
Actual: YES	0	105

Accuracy = 105 / 165 = 63%!

Performance measures: balanced accuracy

Balanced accuracy = average percentage correct for each class, averaged across classes

n=165	Predicted: NO	Predicted: YES
Actual: NO	0	60
Actual: YES	0	105

Accuracy = (0/60 + 105/105)/2 = (0 + 1)/2 = 50%

Performance measures

- ***Accuracy*** = the percentage correctly classified instances across all data. Highly sensitive to bias (especially when classes are not balanced) should not be used, ever.

- ***Balanced accuracy*** = average percentage correct for each class, averaged across classes

SDT measures

- Hit rate – False alarm rate
- d' = Z(HR) – Z(FAR)
- AUC = Area Under the Curve

decision boundary

n=165	Predicted: NO	Predicted: YES
Actual: NO	correct rejections 50	false alarms 10
Actual: YES	misses 5	hits 100

hit rate = 100 / 105 = .95

false alarm rate = 10/60 = 0.16

Signal Detection Theory (SDT) measures: Area Under the Curve (AUC)

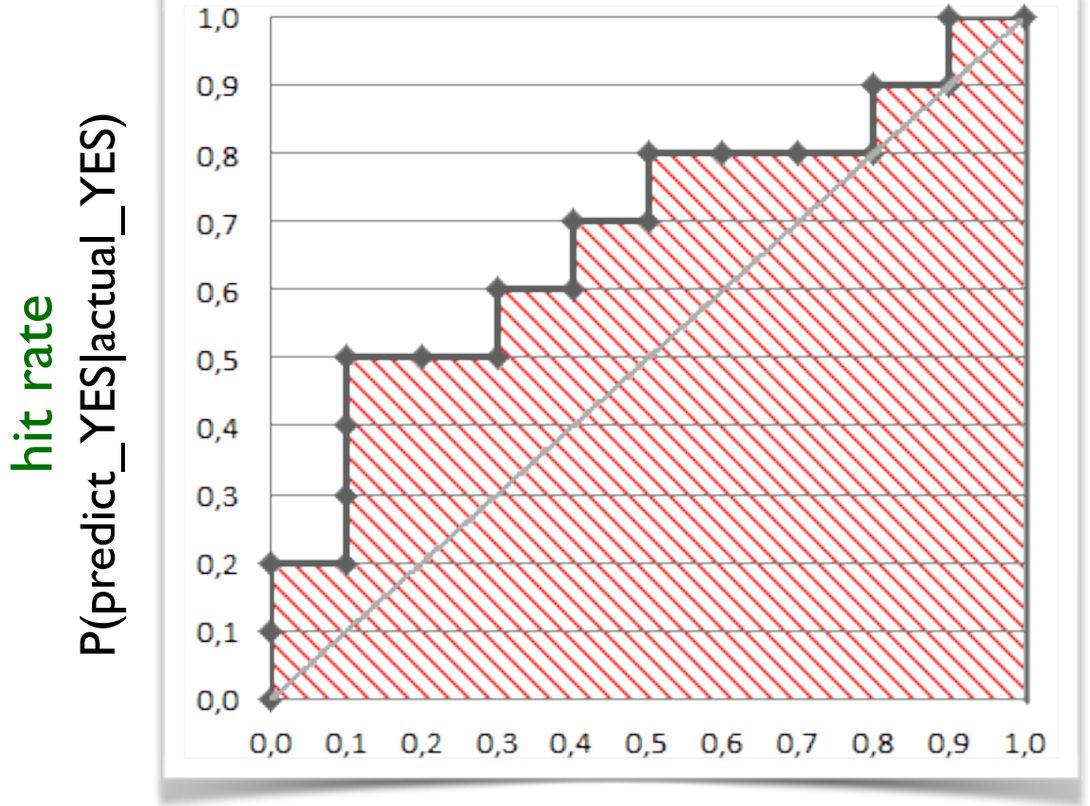
decision boundary

		Predict NO	Predict YES
N	Actual: NO	0 ⟷ Probability of YES	1
P	Actual: YES	0 ⟷	1

Using the 'score' (probability) that an instance is from a certain class

#	C	Score
1	P	0,9
2	P	0,8
3	N	0,7
4	P	0,6
5	P	0,55
6	P	0,54
7	N	0,53
8	N	0,52
9	P	0,51
10	N	0,505
11	P	0,4
12	N	0,39
13	P	0,38
14	N	0,37
15	N	0,36
16	N	0,35
17	P	0,34
18	N	0,33
19	P	0,3
20	N	0,1

sorted probabilities



Actual class label (P = Yes, N = No)

Probability of YES (according to the classifier)

Experimental design and analytical approach

- **conditions** versus **classes**

	Face	House
Face cue	25%	10%
House cue	10%	25%
Neutral cue	15%	15%

- Experimental design: the *condition labels* describe the cells in your factorial design

- Analytical approach using decoding: the *stimulus classes* are the relevant things that you are comparing when doing a decoding analysis (often these are the levels of a factor in your design)

Within class imbalances

Classes: face versus house

	Face	House	← Factor 'stimulus type'
Face cue	25%	10%	**35%**
House cue	10%	25%	**35%**
Neutral cue	15%	15%	**30%**
	50%	**50%**	100%

Factor 'expectation' ↑

Designs are often not balanced even when they seem to be at first glance

Within class imbalances

But the number of trials may not be balanced ***within stimulus classes***

	Face	House	
Face cue	25%	10%	**35%**
House cue	10%	25%	**35%**
Neutral cue	15%	15%	**30%**
	50%	**50%**	100%

← **Factor ‘stimulus type’**

↑ **Factor ‘expectation’**

Correctly cued items contribute more strongly to the comparison than neutrally or incorrectly cued items

Within class imbalances

The ADAM toolbox automatically *removes trials* within the underrepresented *conditions* to keep the design balanced (this is called ***undersampling***)

	Face	House	
Face cue	10%	10%	20%
House cue	10%	10%	20%
Neutral cue	10%	10%	20%
	30%	30%	**60%**

Factor ‘stimulus type’

Factor ‘expectation’

Between class imbalances

Your design may also be unbalanced ***between stimulus classes***

	Target	Foil	
Target cue	45%	5%	**50%**
Neutral cue	45%	5%	**50%**
	90%	**10%**	100%

Factor 'stimulus type'

Factor 'expectation'

Unbalanced accuracy = percentage correctly classified across all instances
Highly sensitive to bias when classes are not balanced!

Between class imbalances

By default, ADAM balances between classes by *duplicating/generating trials* of the underrepresented *classes* in the training set (this is called ***oversampling***)

	Target	Foil	
Target cue	45%	5% * 9	**50%**
Neutral cue	45%	5% * 9	**50%**
	50%	**50%**	100%

Factor ‘stimulus type’

Factor ‘expectation’

Some MVPA (machine learning) lingo

- The algorithm that is used to classify multivariate data patterns into distinct categories is called **the classifier**

- The categories (conditions) that the classifier discriminates are called **classes**

- The input data points used for classification are called the **features**. In the case of MEG/EEG, the channels/electrodes are usually the features.

- If a classifier is trained and tested on the same data this results in **overfitting** (in cognitive neuroscience we call this *double-dipping*)

- To prevent this, you either need to train and test on a different dataset, or use **k-fold cross-validation**

- After training, each feature is assigned a classifier **weight**, telling the classifier how useful that feature is to discriminate the relevant classes

- The performance of a classifier can be assessed in various ways, by making use of the **confusion matrix**

- **Undersampling** and **oversampling** can be used to keep a design **balanced** (preventing *bias* in classification accuracy)

Course overview

- Day 1 (introductory)
  - lecture 1: History and electrophysiological basis of EEG
  - lecture 2: Basic concepts in MVPA
  - lecture 3: Decoding measures and the temporal generalization method
  - lecture 4: Architecture of the ADAM toolbox and explaining the experiment of the practical
  - Afternoon: practical
- Day 2 (advanced)
  - lecture 1: Using classifier scores to map brain to behavior
  - lecture 2: Forward encoding models in MVPA: how do they work, concepts and analytical approach
  - Afternoon: practical, analyze your own data and/or a supplied dataset from scratch

Questions

Course overview

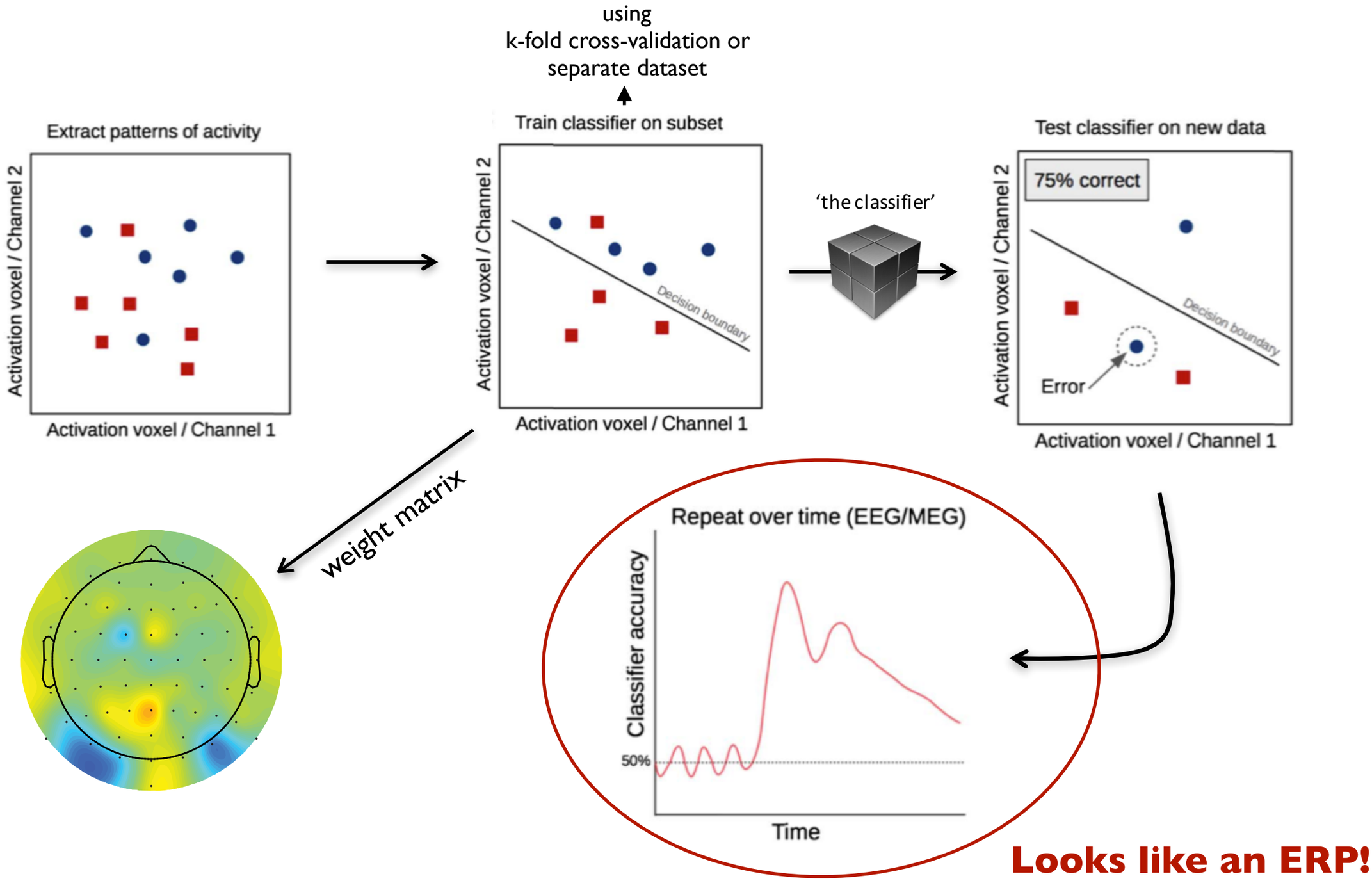
- **Day 1 (introduction)**
  - lecture 1: History and electrophysiological basis of EEG
  - lecture 2: Backward decoding models in MVPA: concepts and analytical approach
  - lecture 3: Advantages of MVPA, the temporal generalization method
  - lecture 4: Architecture of the ADAM toolbox and explaining the experiment of the practical
  - *Afternoon: practical*
- **Day 2 (advanced)**
  - lecture 1: Multiple comparisons, MVPA experimental design, mapping brain to brain/behavior
  - lecture 2: Forward encoding models in MVPA: how do they work, concepts and analytical approach
  - *Afternoon: practical, analyze your own data and/or a supplied dataset*

Lecture 3

- Advantages of MVPA

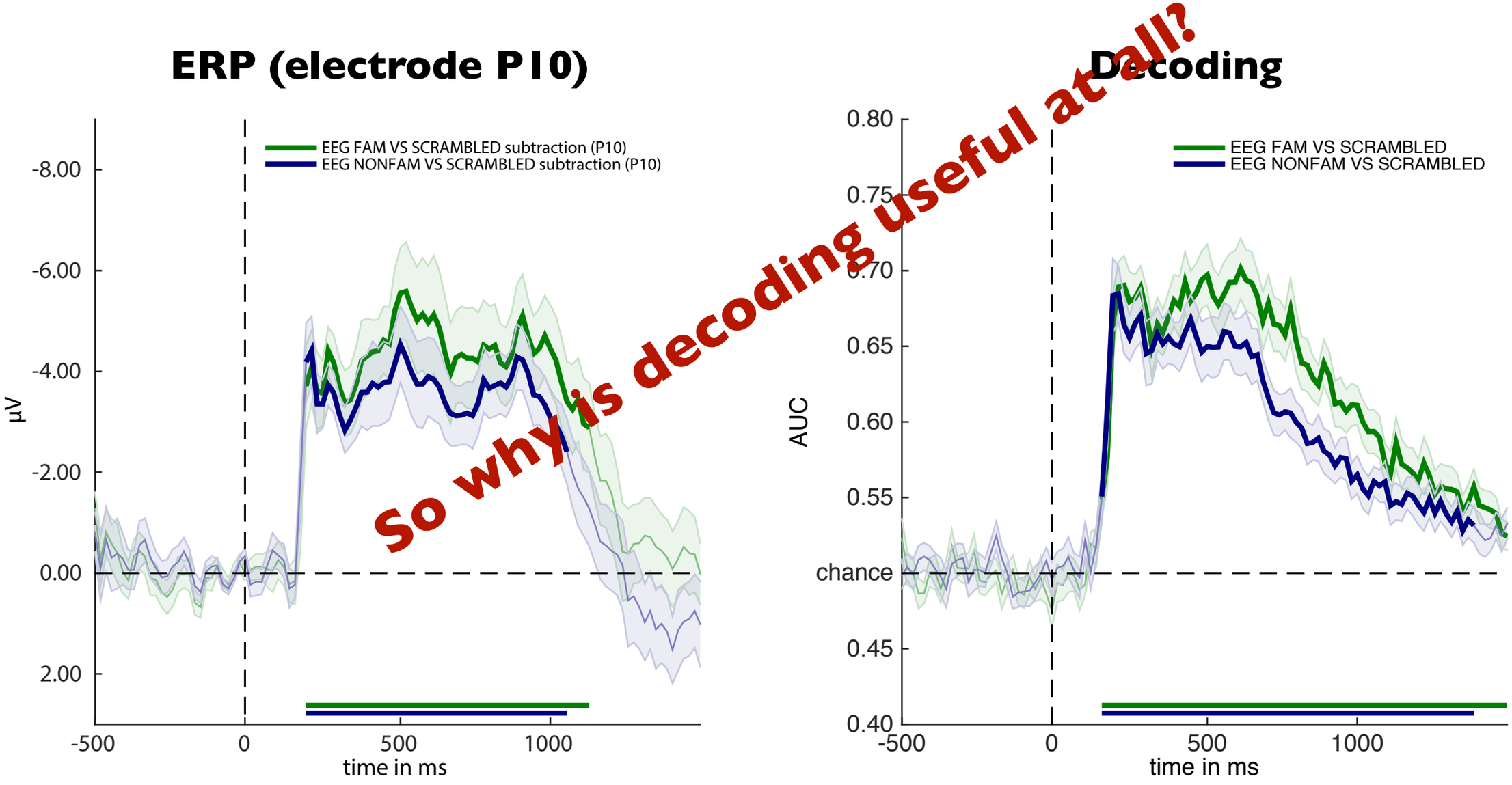
- The temporal generalization method

What do EEG decoding results look like?



Grootswagers, T., Wardle, S. G., & Carlson, T. A. (2017). Decoding Dynamic Brain Patterns from Evoked Responses: A Tutorial on Multivariate Pattern Analysis Applied to Time Series Neuroimaging Data. *Journal of Cognitive Neuroscience*, *29*(4), 677–697.

Result: a temporal signal



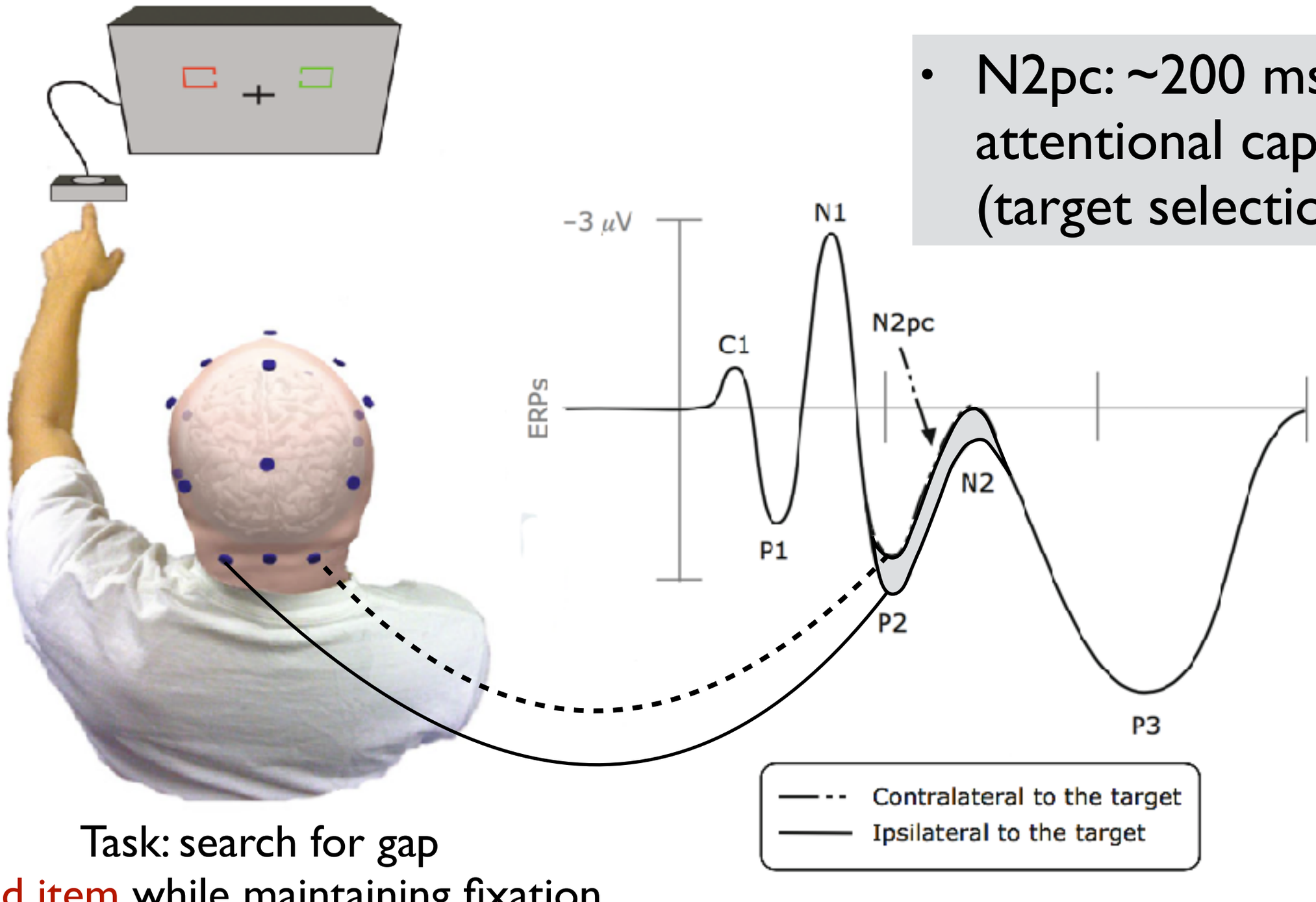
Advantages of MVPA

- Using MVPA, you do not have to specify or know *beforehand* which electrodes contain the experimental effect

- MVPA identifies differences that are not picked up by a regular ERP analysis, especially when the locus of the effect is unknown

- This also simplifies the multiple comparisons problem

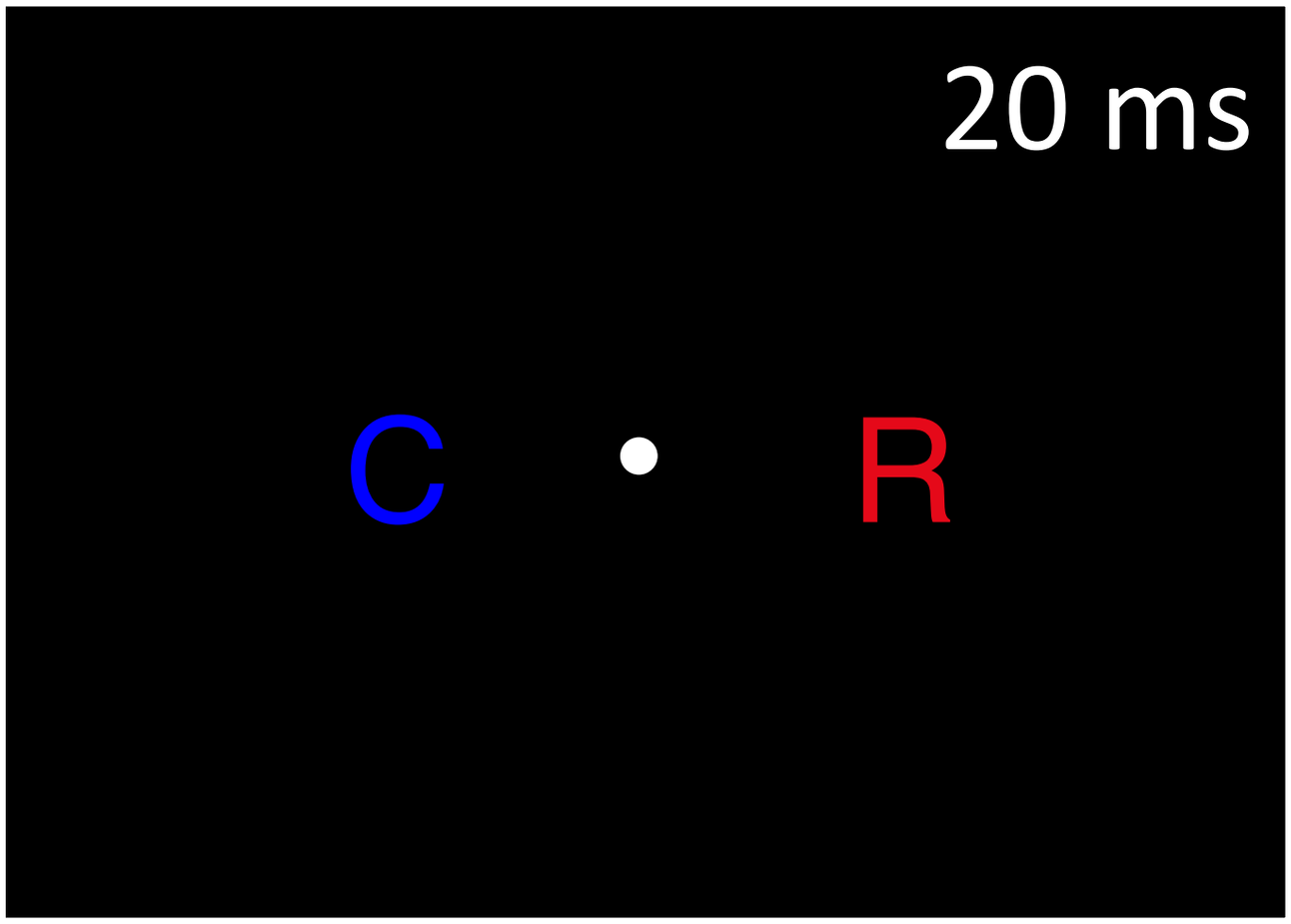
Example 1: attentional selection (N2pc)



- N2pc: ~200 ms attentional capture (target selection)

Task: search for gap
in red item while maintaining fixation

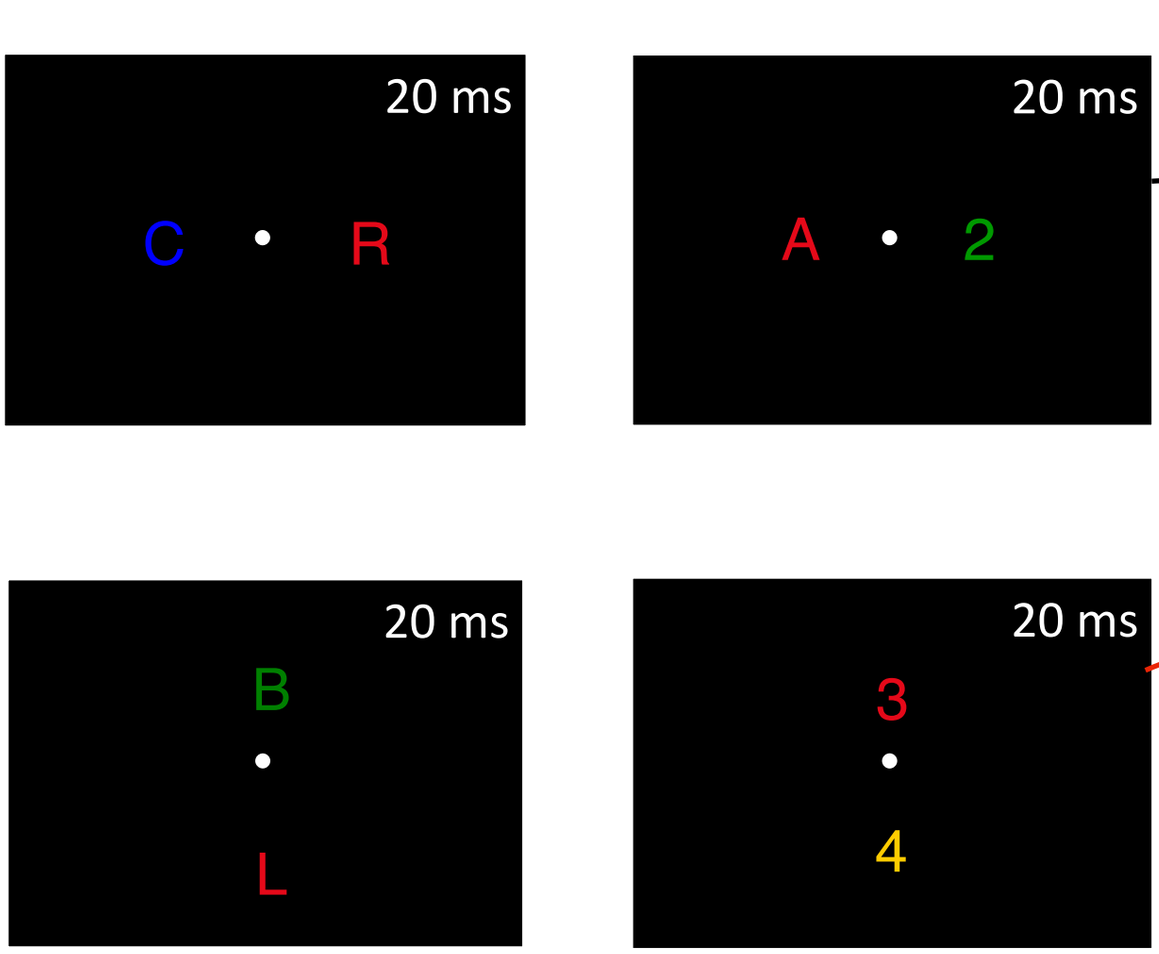
Attentional selection: N2pc

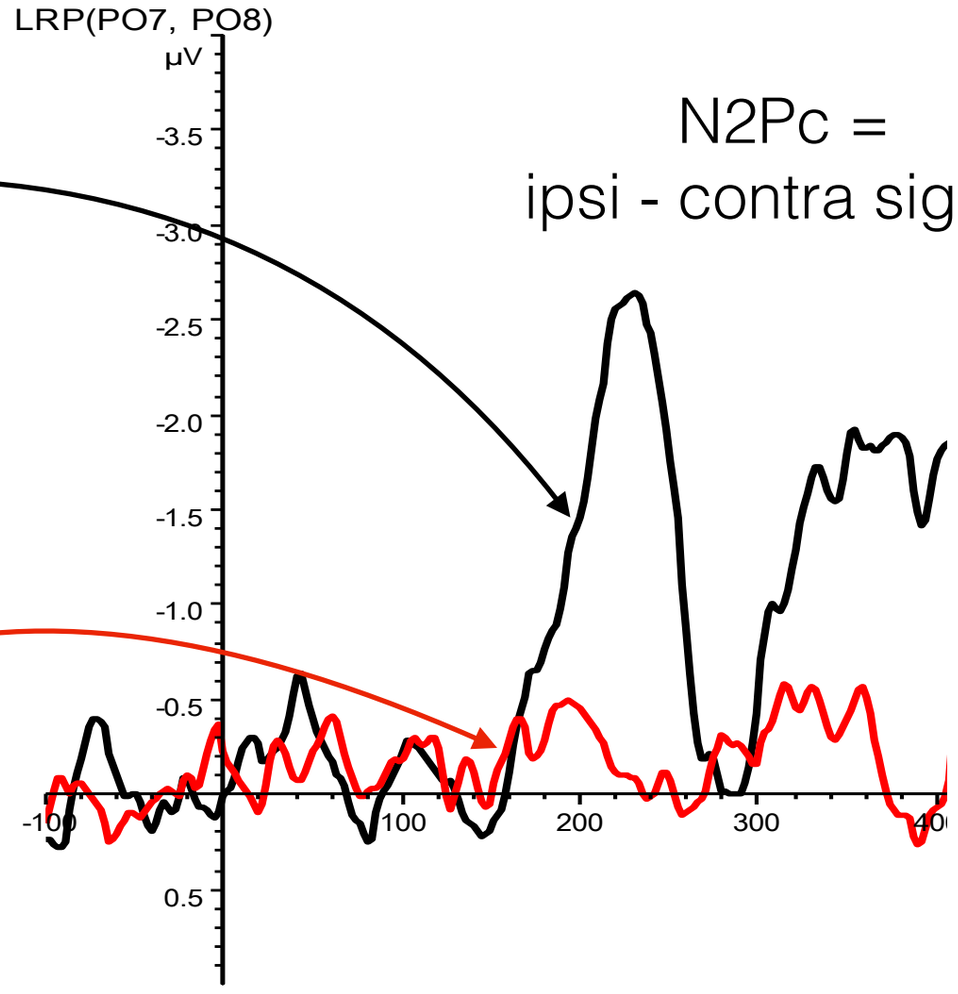


Task: identify the red item (digit or letter)

Example 1: ERP of contingent capture N2Pc

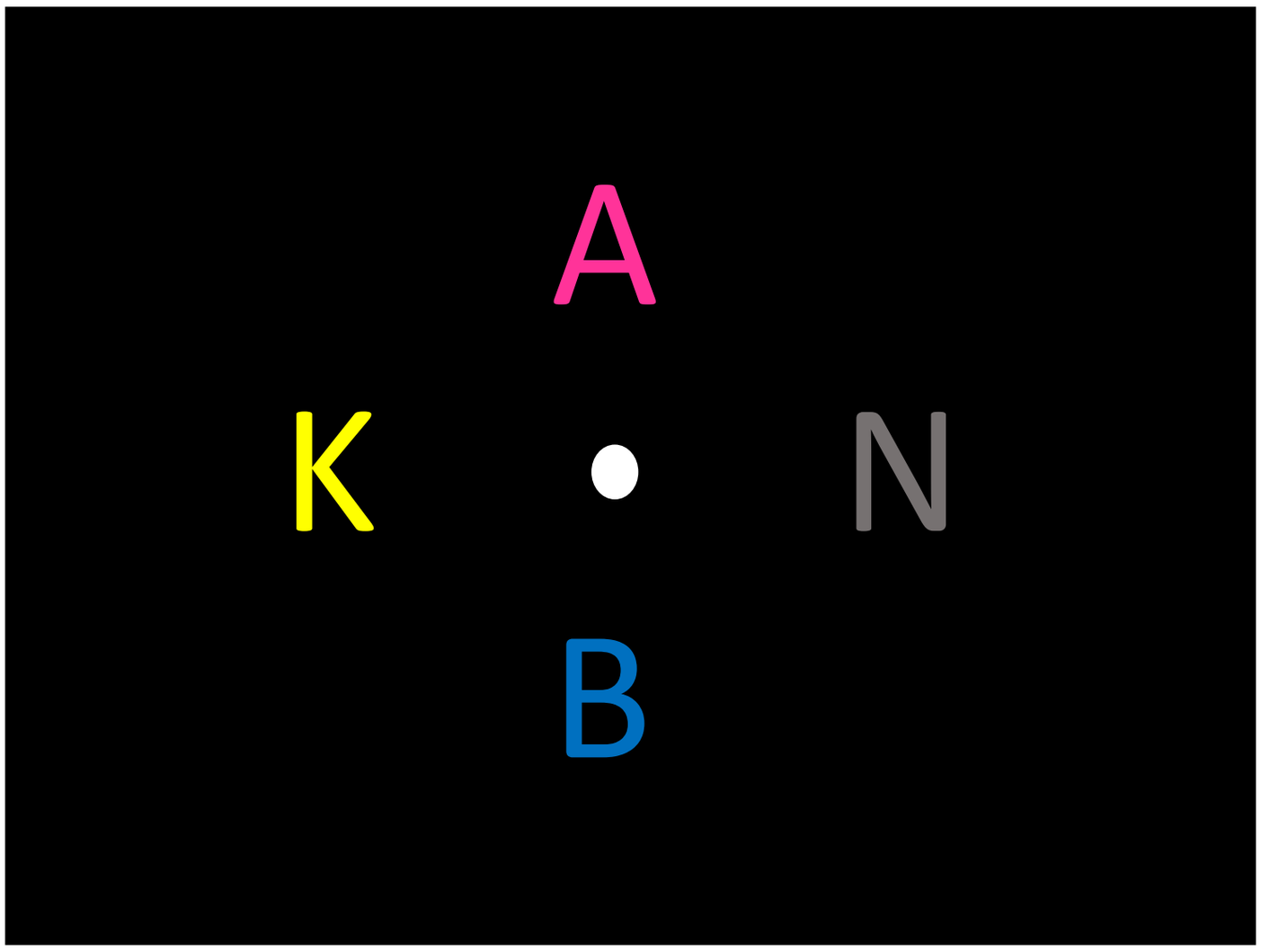
Task: detect whether red item is digit or letter





You cannot find an N2Pc for top vs bottom because the component is lateralized

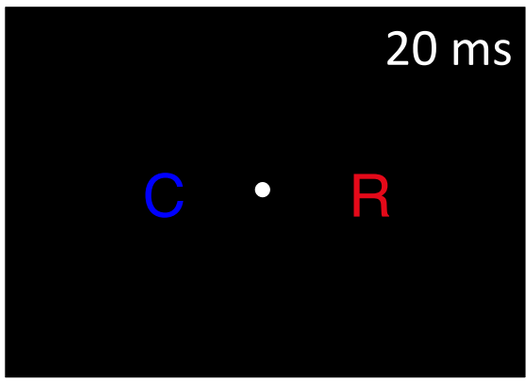
How to investigate parallel selection?
How to determine attention to vertical targets?

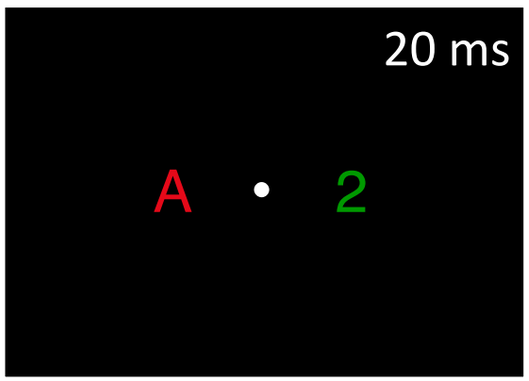


Task: identify the yellow and blue item

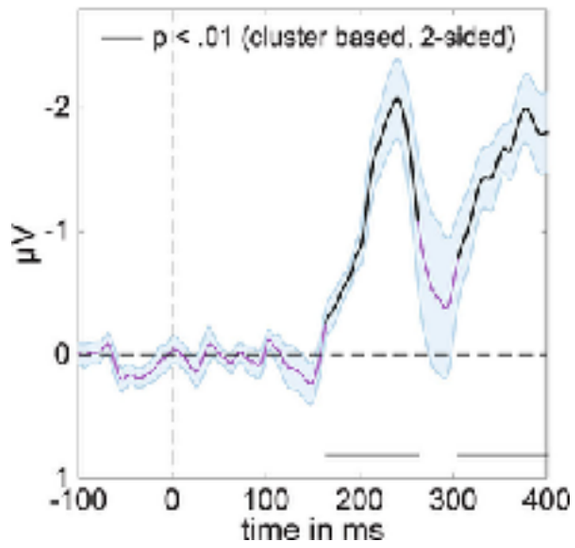
MVPA:classify where the target is

Task: detect whether red item is digit or letter





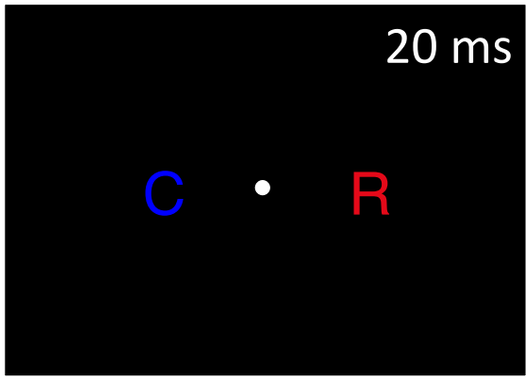
N2Pc

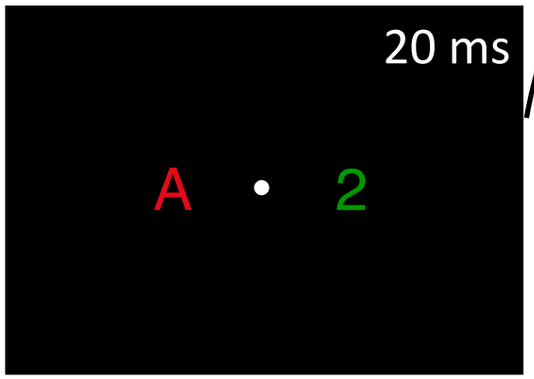


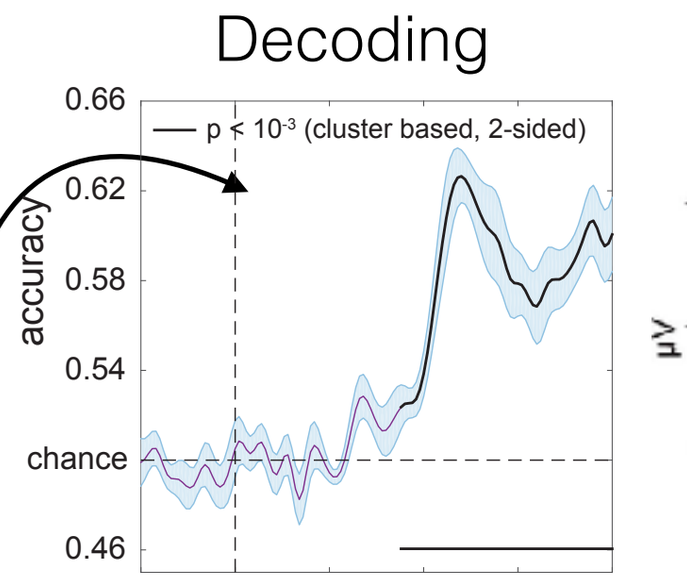
MVPA extracts ***any*** pattern from the data (does not have to be lateralized)

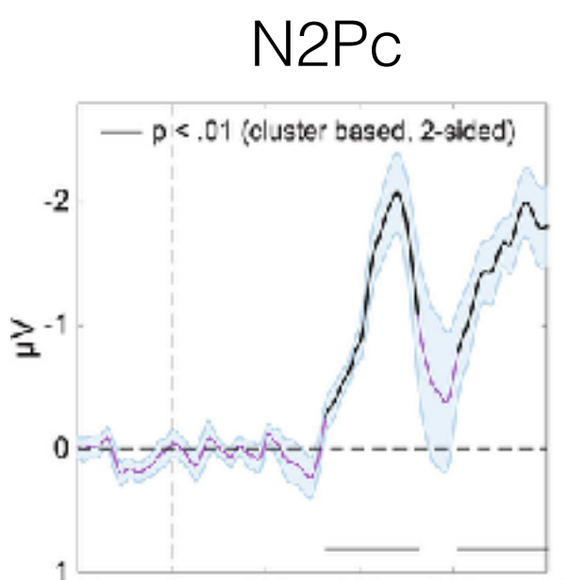
MVPA:classify where the target is

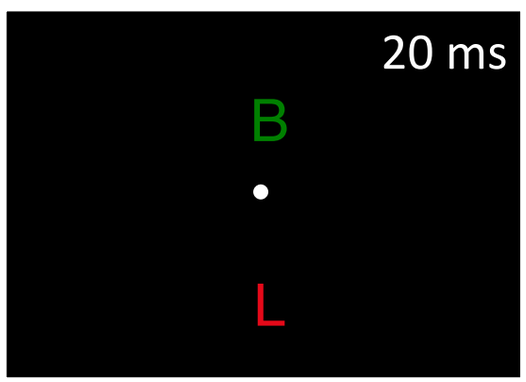
Task: detect whether red item is digit or letter

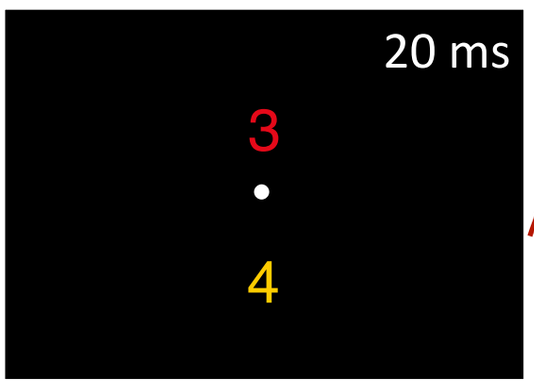


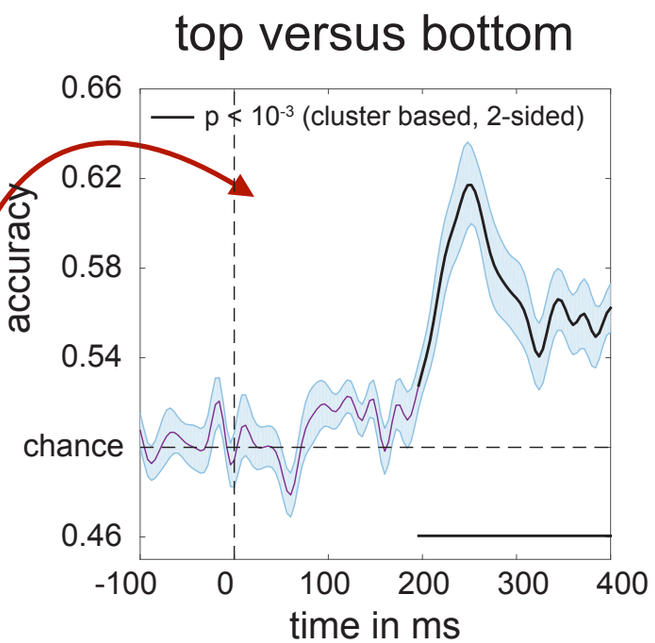








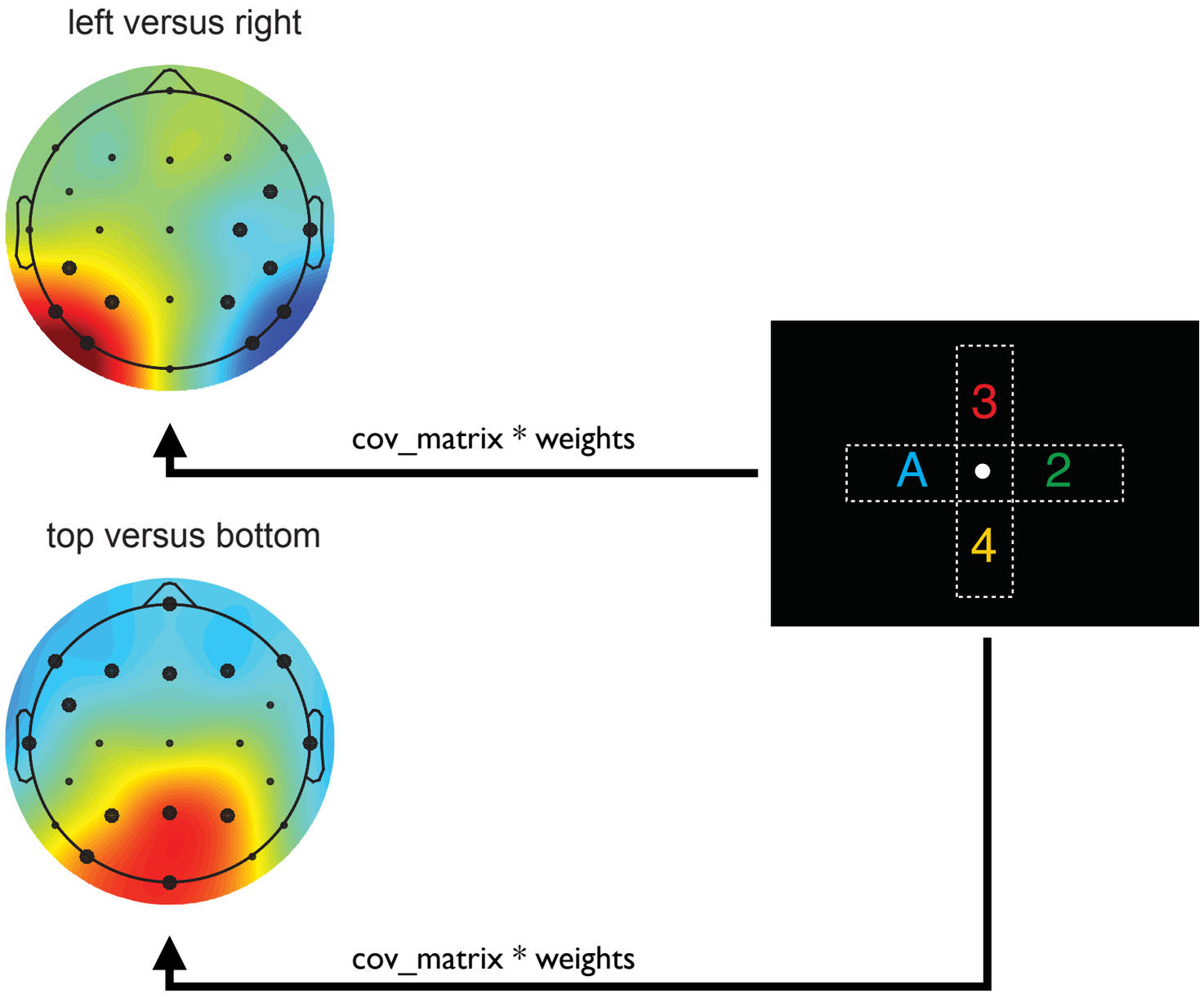




Also for
top vs bottom!

MVPA extracts ***any*** pattern from the data (does not have to be lateralized)

Use decoding to identify the ‘vertical’ N2pc



Fahrenfort, J. J., Grubert, A., Olivers, C. N. L., & Eimer, M. (2017). Multivariate EEG analyses support high-resolution tracking of feature-based attentional selection. *Scientific Reports*, 7(1), 1886.

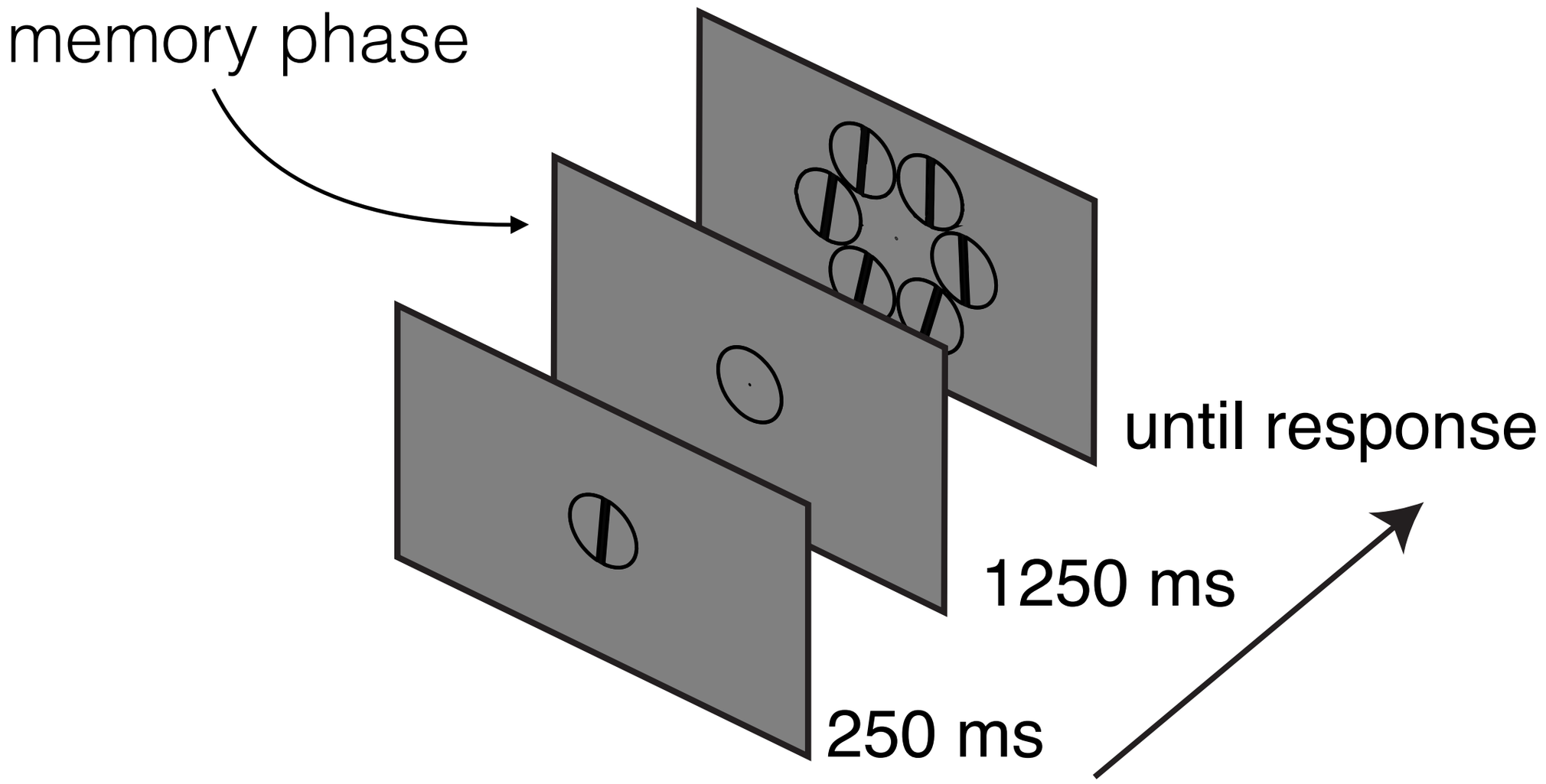
Why else is MVPA useful?

- You do not have to select electrodes

- The nice thing about EEG: it has high temporal resolution
  → look at the ***stability*** and ***dynamics*** of neural representations over time

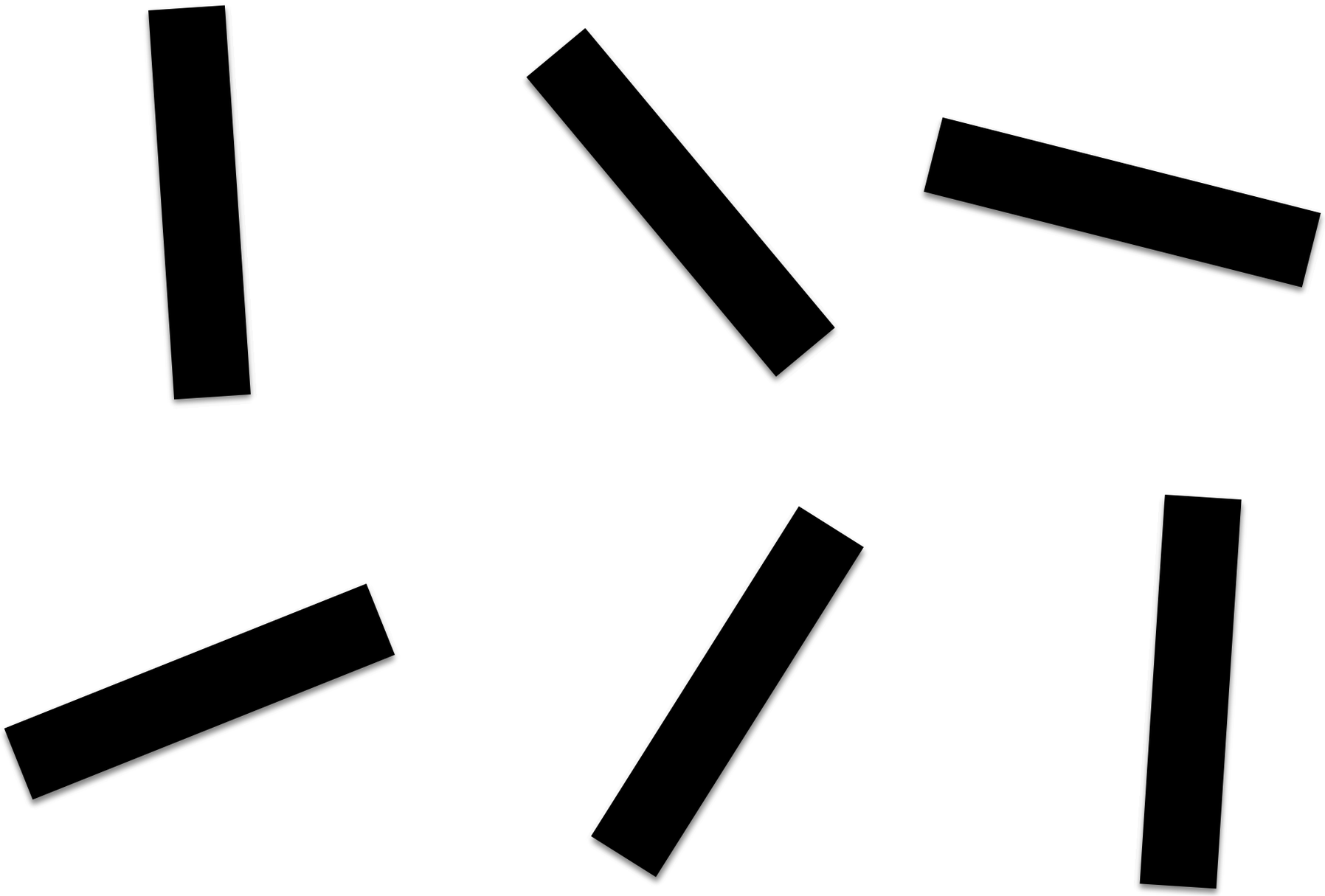
- Train on one time point, and test on all the others to assess stability of the signal!

Example 2: Tracking memory over time



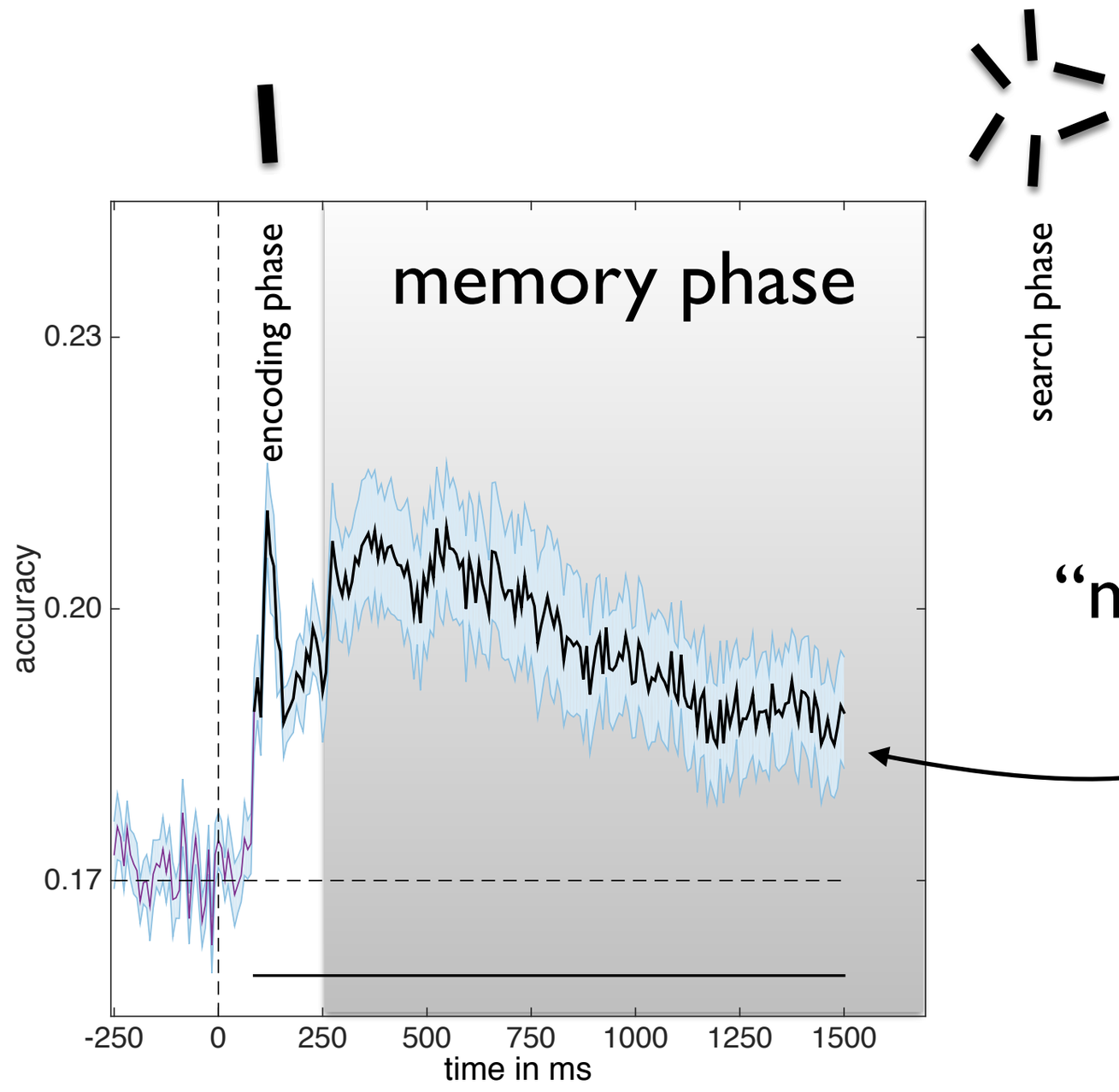
Task: remember orientation and identify by clicking on the correct item

Six potential orientations to remember



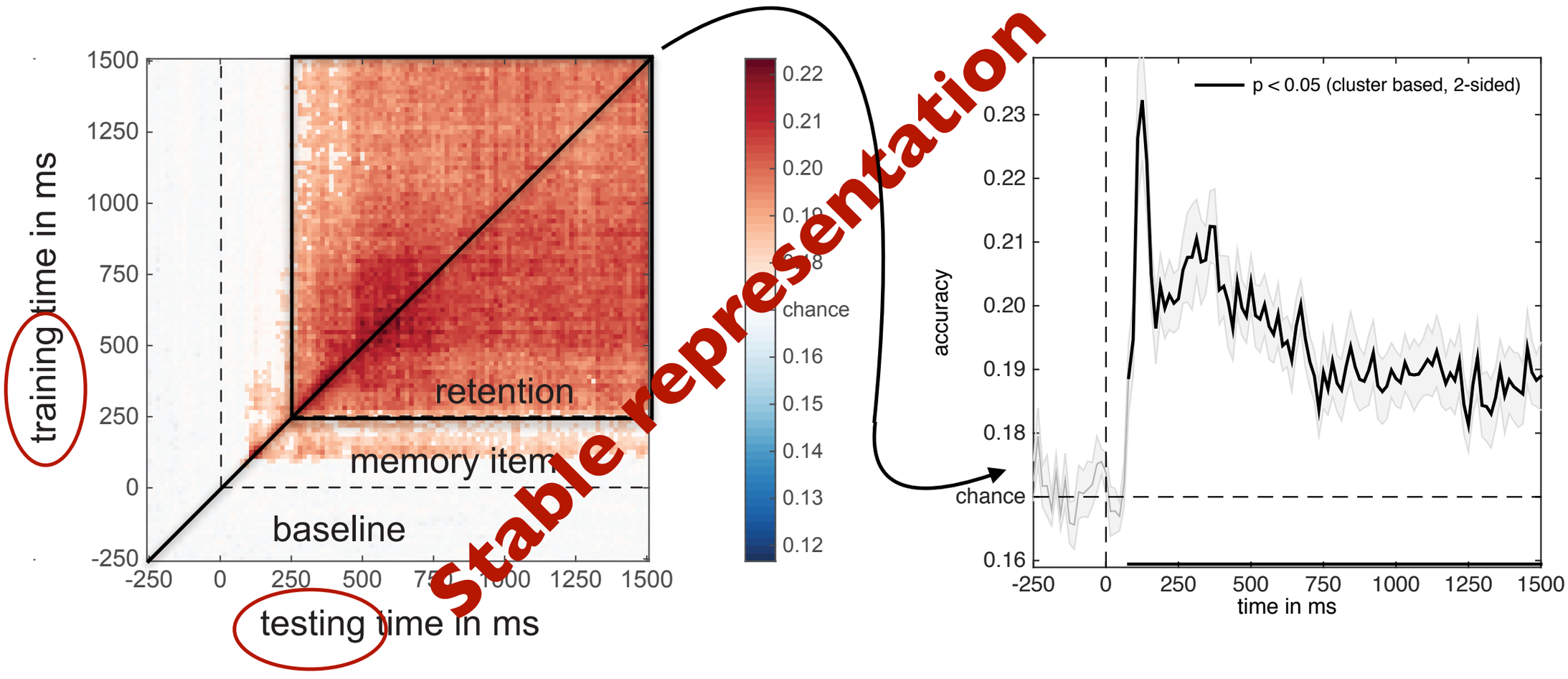
chance performance of classifier: 1/6th

Classification over time



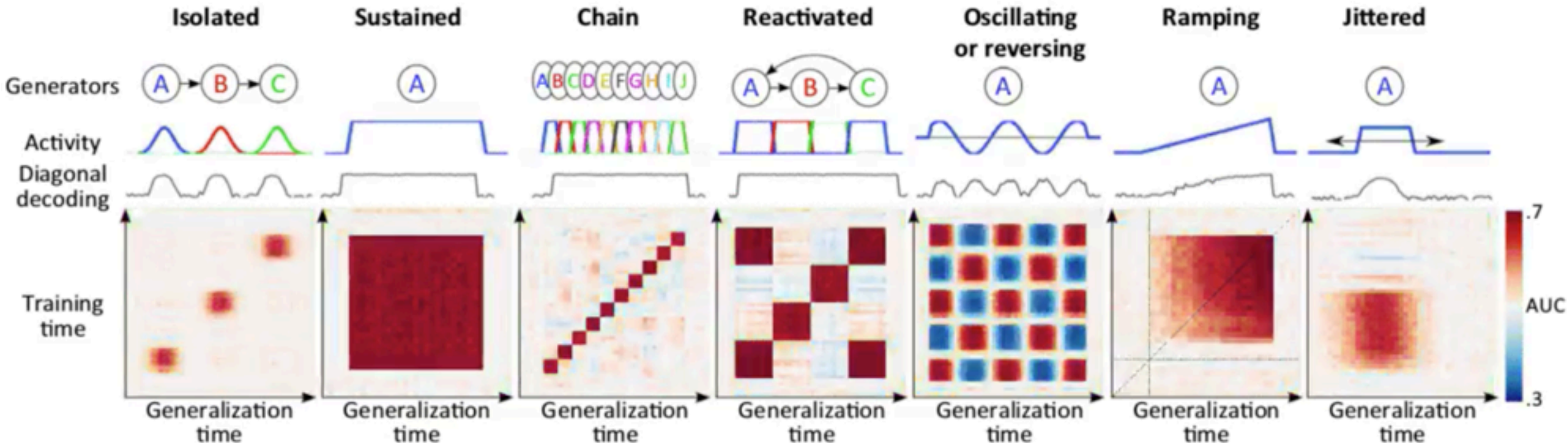
"mind reading"

Temporal generalization matrix



King, J. R., & Dehaene, S. (2014). Characterizing the dynamics of mental representations: the temporal generalization method. *Trends in Cognitive Sciences*, *18*(4), 203–210.

Temporal generalization: stability and dynamics over time

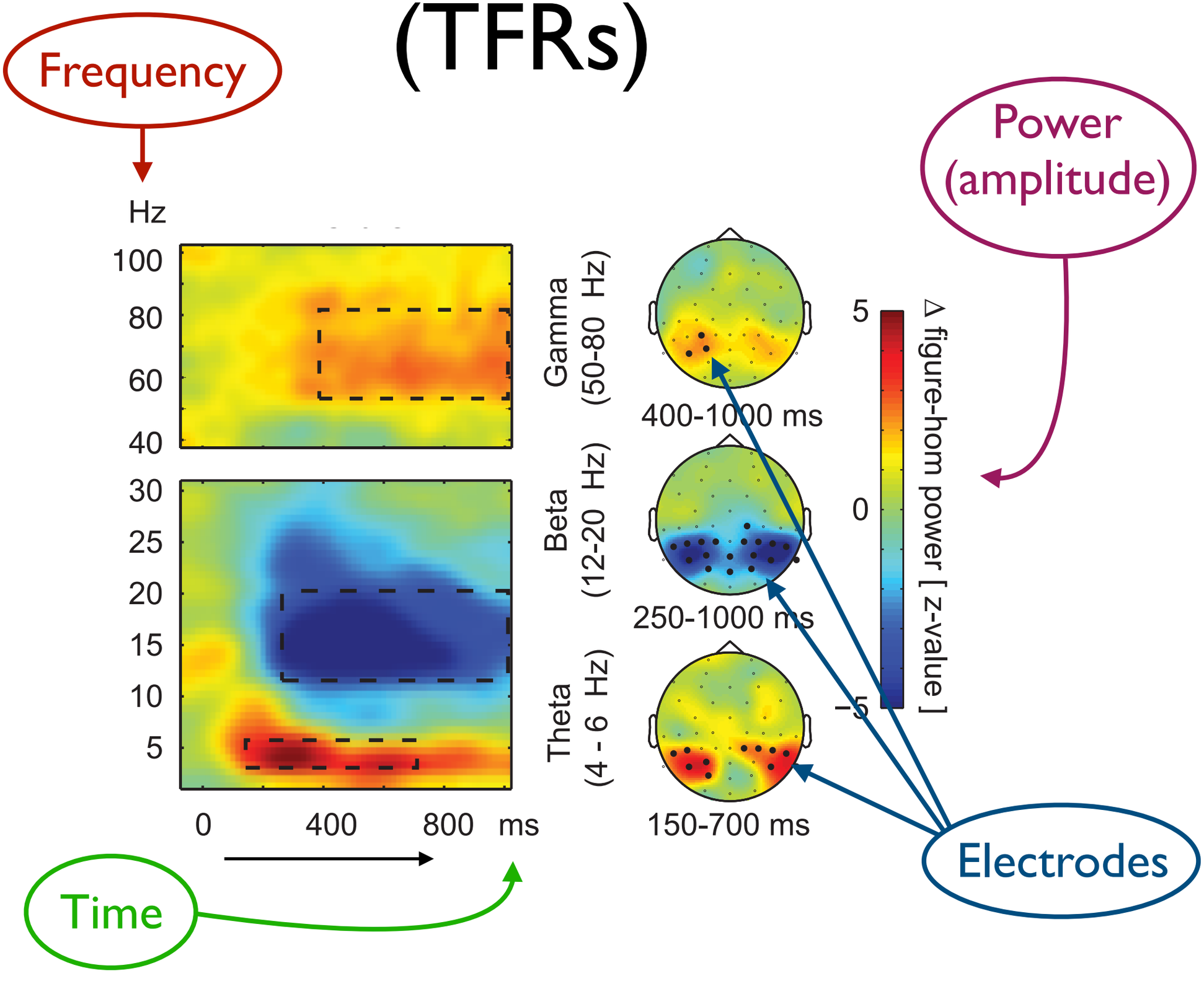


King, J. R., & Dehaene, S. (2014). Characterizing the dynamics of mental representations: the temporal generalization method. *Trends in Cognitive Sciences*, *18*(4), 203–210.

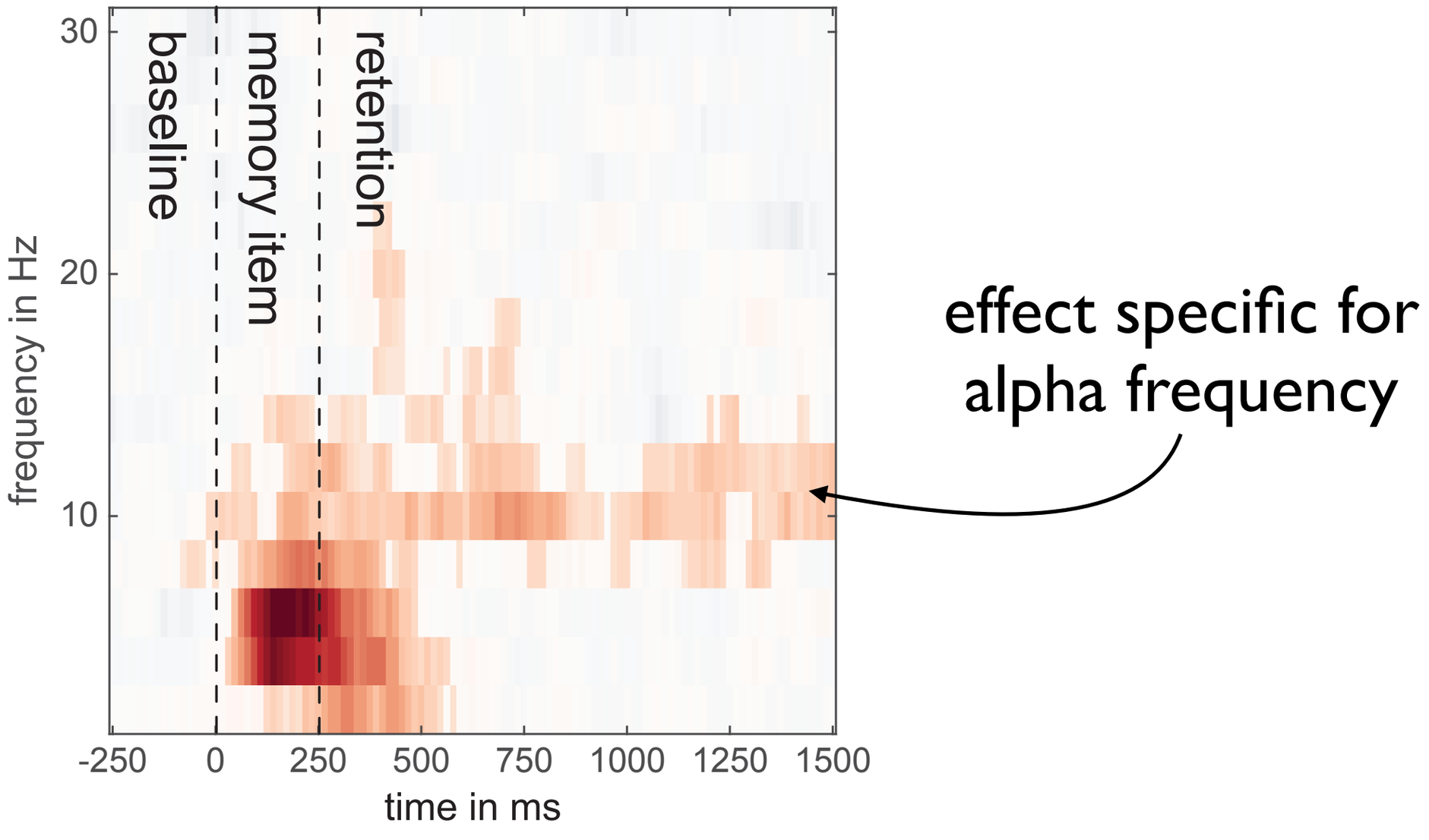
Why else is MVPA useful for MEG/EEG?

- You do not have to select electrodes
- Look at cortical stability and dynamics using temporal generalization
- Time-frequency representations (TFRs): perform MVPA on time-frequency data

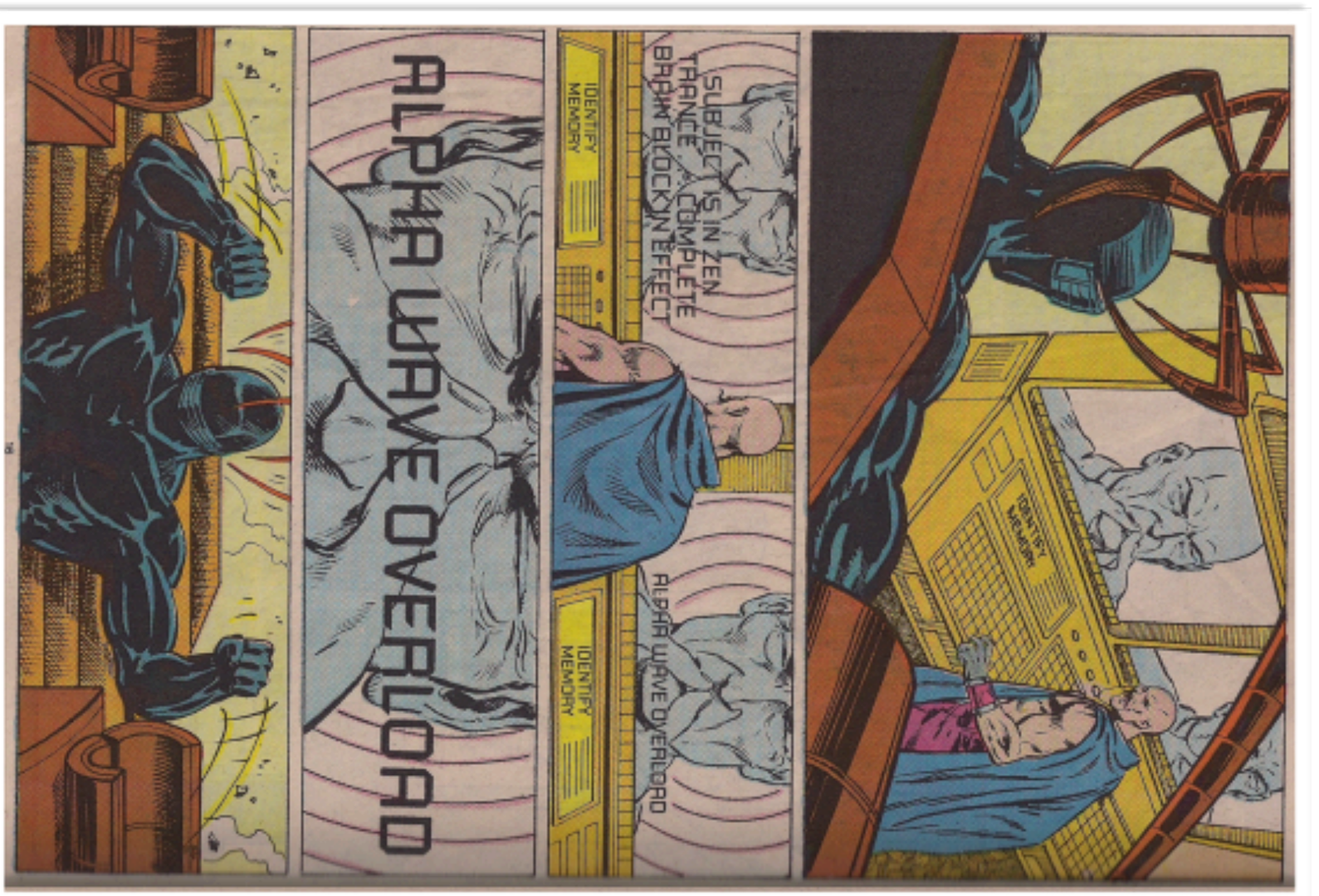
Time-frequency representations (TFRs)



Example of TFR decoding



The same comic book 4 years later



Conclusions

- MVPA allows you to find effects without selecting electrodes
- MVPA allows you to look at temporal generalization (stability / dynamics over time)
- MVPA also allows you to look at time frequency data, again without selecting electrodes

Course overview

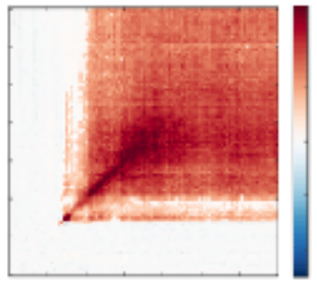
- **Day 1 (introduction)**
  - lecture 1: History and electrophysiological basis of EEG
  - lecture 2: Backward decoding models in MVPA: concepts and analytical approach
  - lecture 3: Advantages of MVPA, the temporal generalization method
  - lecture 4: Architecture of the ADAM toolbox and explaining the experiment of the practical
  - *Afternoon: practical*
- **Day 2 (advanced)**
  - lecture 1: Multiple comparisons, MVPA experimental design, mapping brain to brain/behavior
  - lecture 2: Forward encoding models in MVPA: how do they work, concepts and analytical approach
  - *Afternoon: practical, analyze your own data and/or a supplied dataset*

Lecture 4
ADAM architecture and practical

- Give overview of the ADAM toolbox

- Explain experiment from the practical

the Amsterdam Decoding and Modeling toolbox (ADAM)



1. Pre-processing
   (can do this using EEGLAB, do not need ADAM)

2. First level (single subject) analyses are computed and stored

3. Reading single subject results and compute group statistics

4. Visualize group statistics in a plot

ADAM architecture

Import and pre-process
(not part of ADAM)

Import native EEG or MEG data into EEGLAB or FieldTrip format, pre-process, e.g. highpass filter, epoching, artefact rejection. Baseline correction and muscle artefact rejection can also be applied by ADAM during first-level analysis.

adam_MVPA_firstlevel

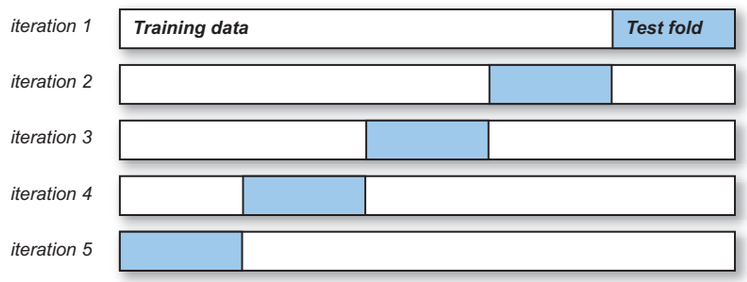
In: Epoched files in either EEGLAB or FieldTrip format
Out: ADAM result files (one for each subject), containing a performance metric for every train-test time sample (raw) or for every train-test sample of every frequency band (tfr)

Use RAW data

Compute time-frequency representations

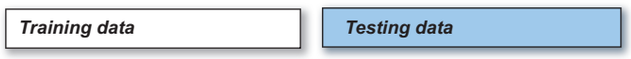
For every time point, build a backward decoding model (BDM) or forward encoding model (FEM) using training data, and compute performance metric on testing data. Weights of BDMs are forward transformed.

Option 1: K-fold cross-validation. Requires a single data file per subject.



The final performance metric is computed by averaging over test folds (in this example, *K=5*).

Option 2: Requires separate data sets for training and testing (either using separate files or separate event values for train and test data)



The performance metric is computed over the testing data.

Several transformations can be performed on the training and testing data, e.g. binning, whitening, computing induced power, etc. These transformations are either performed separately on training and testing data, or they are performed indiscriminately across all stimulus classes.

adam_compute_group_MVPA
adam_compute_group_ERP

In: ADAM result files computed by *adam_MVPA_firstlevel*
Out: ADAM stats variable(s) containing group statistics

adam_compare_MVPA_stats

In: ADAM stats variable(s) containing group statistics
Out: ADAM stats variable(s) containing group statistics

adam_plot_MVPA
adam_plot_BDM_weights

In: ADAM stats variable(s) containing group statistics
Out: publication-ready graphs of performance metrics and/or topographical maps of forward transformed weights

1. Import

Import and pre-process (not part of ADAM)

Import native EEG or MEG data into EEGLAB or FieldTrip format, pre-process, e.g. highpass filter, epoching, artefact rejection. Baseline correction and muscle artefact rejection can also be applied by ADAM during first-level analysis.

ADAM architecture

Import and pre-process (not part of ADAM)

Import native EEG or MEG data into EEGLAB or FieldTrip format, pre-process, e.g. highpass filter, epoching, artefact rejection. Baseline correction and muscle artefact rejection can also be applied by ADAM during first-level analysis.

adam_MVPA_firstlevel

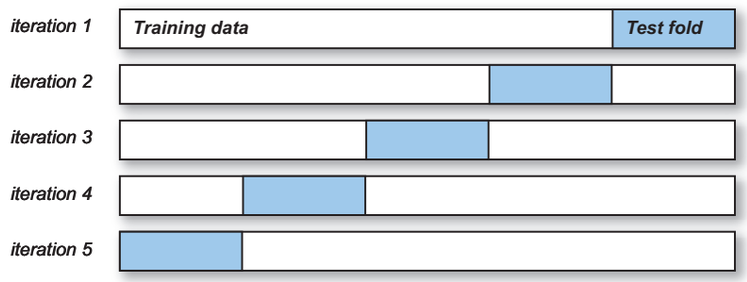
In: Epoched files in either EEGLAB or FieldTrip format
Out: ADAM result files (one for each subject), containing a performance metric for every train-test time sample (raw) or for every train-test sample of every frequency band (tfr)

Use RAW data

Compute time-frequency representations

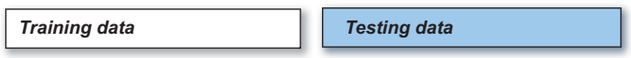
For every time point, build a backward decoding model (BDM) or forward encoding model (FEM) using training data, and compute performance metric on testing data. Weights of BDMs are forward transformed.

Option 1: K-fold cross-validation. Requires a single data file per subject.



The final performance metric is computed by averaging over test folds (in this example, *K=5*).

Option 2: Requires separate data sets for training and testing (either using separate files or separate event values for train and test data)



The performance metric is computed over the testing data.

Several transformations can be performed on the training and testing data, e.g. binning, whitening, computing induced power, etc. These transformations are either performed separately on training and testing data, or they are performed indiscriminately across all stimulus classes.

adam_compute_group_MVPA
adam_compute_group_ERP

In: ADAM result files computed by *adam_MVPA_firstlevel*
Out: ADAM stats variable(s) containing group statistics

adam_compare_MVPA_stats

In: ADAM stats variable(s) containing group statistics
Out: ADAM stats variable(s) containing group statistics

adam_plot_MVPA
adam_plot_BDM_weights

In: ADAM stats variable(s) containing group statistics
Out: publication-ready graphs of performance metrics and/or topographical maps of forward transformed weights

2. First level (single subject)

adam_MVPA_firstlevel

In: Epoched files in either EEGLAB or FieldTrip format

Out: ADAM result files (one for each subject), containing a performance metric for every train-test time sample (raw) or for every train-test sample of every frequency band (tfr)

read from disk

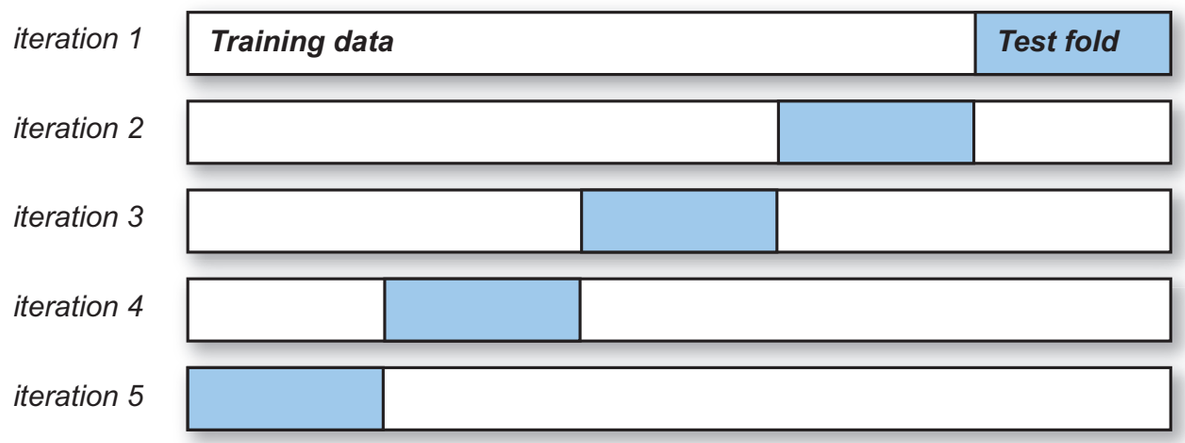
written to disk

Use RAW data

Compute time-frequency representations

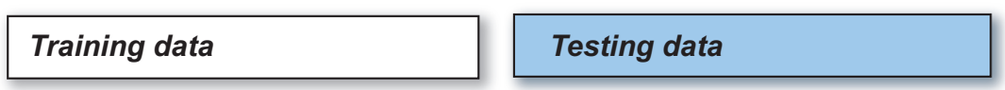
For every time point, build a backward decoding model (BDM) or forward encoding model (FEM) using training data, and compute performance metric on testing data. Weights of BDMs are forward transformed.

Option 1: K-fold cross-validation. Requires a single data file per subject.



The final performance metric is computed by averaging over test folds (in this example, *K=5*).

Option 2: Requires separate data sets for training and testing (either using separate files or separate event values for train and test data)



The performance metric is computed over the testing data.

Several transformations can be performed on the training and testing data, e.g. binning, whitening, computing induced power, etc. These transformations are either performed separately on training and testing data, or they are performed indiscriminately across all stimulus classes.

ADAM architecture

Import and pre-process
(not part of ADAM)

Import native EEG or MEG data into EEGLAB or FieldTrip format, pre-process, e.g. highpass filter, epoching, artefact rejection. Baseline correction and muscle artefact rejection can also be applied by ADAM during first-level analysis.

adam_MVPA_firstlevel

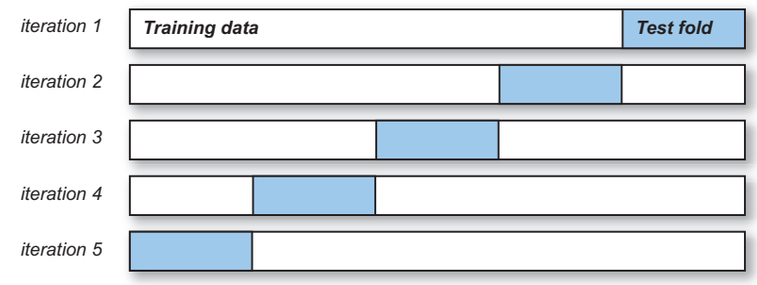
In: Epoched files in either EEGLAB or FieldTrip format
Out: ADAM result files (one for each subject), containing a performance metric for every train-test time sample (raw) or for every train-test sample of every frequency band (tfr)

Use RAW data

Compute time-frequency representations

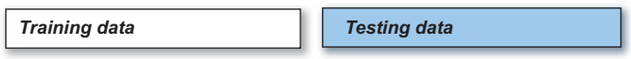
For every time point, build a backward decoding model (BDM) or forward encoding model (FEM) using training data, and compute performance metric on testing data. Weights of BDMs are forward transformed.

Option 1: K-fold cross-validation. Requires a single data file per subject.



The final performance metric is computed by averaging over test folds (in this example, *K=5*).

Option 2: Requires separate data sets for training and testing (either using separate files or separate event values for train and test data)



The performance metric is computed over the testing data.

Several transformations can be performed on the training and testing data, e.g. binning, whitening, computing induced power, etc. These transformations are either performed separately on training and testing data, or they are performed indiscriminately across all stimulus classes.

adam_compute_group_MVPA
adam_compute_group_ERP

In: ADAM result files computed by *adam_MVPA_firstlevel*
Out: ADAM stats variable(s) containing group statistics

adam_compare_MVPA_stats

In: ADAM stats variable(s) containing group statistics
Out: ADAM stats variable(s) containing group statistics

adam_plot_MVPA
adam_plot_BDM_weights

In: ADAM stats variable(s) containing group statistics
Out: publication-ready graphs of performance metrics and/or topographical maps of forward transformed weights

3. Group statistics
4. Plotting

read from disk

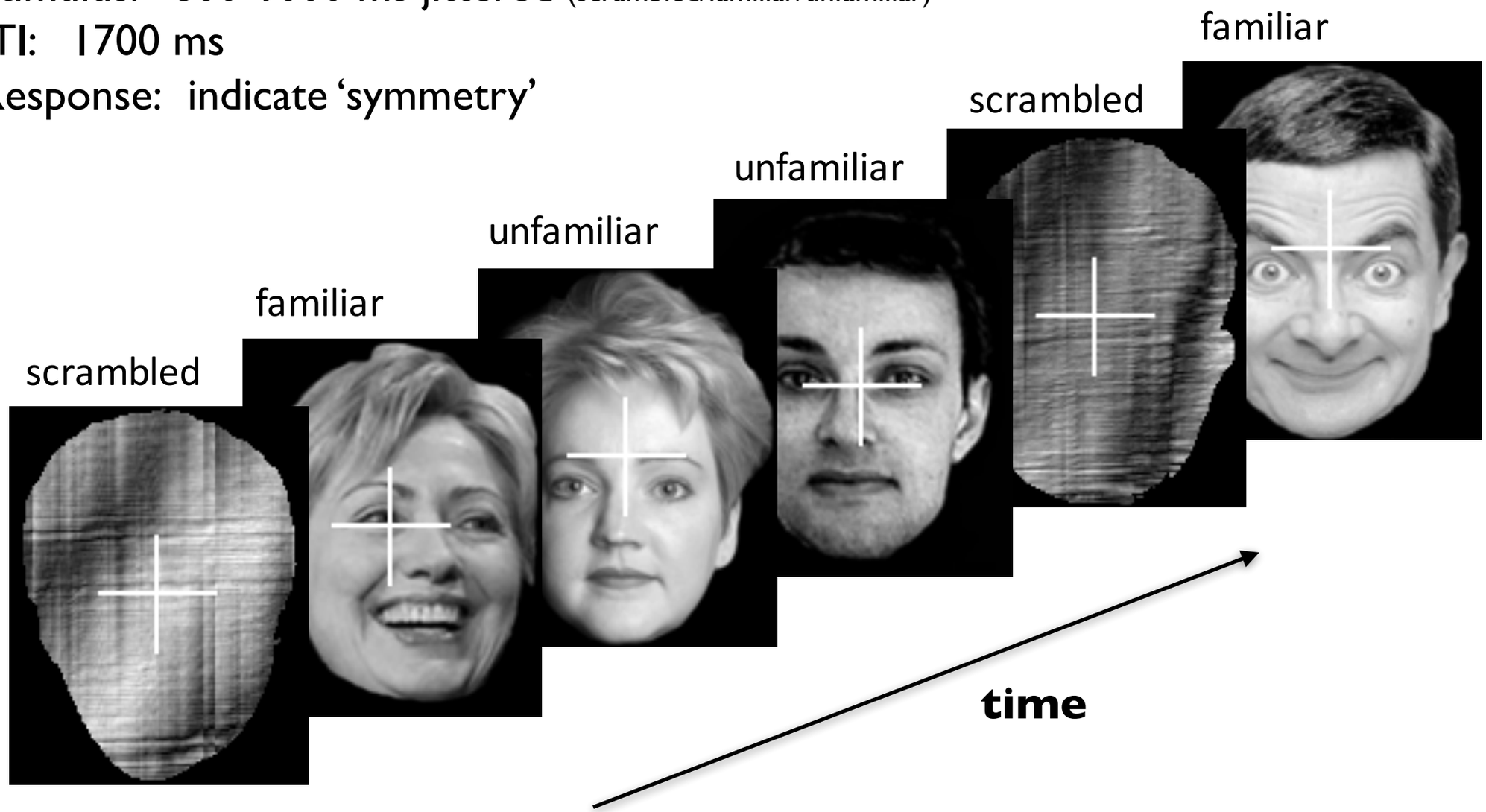
adam_compute_group_MVPA *adam_compute_group_ERP*	In: Out:	ADAM result files computed by *adam_MVPA_firstlevel* ADAM stats variable(s) containing group statistics
adam_compare_MVPA_stats	In: Out:	ADAM stats variable(s) containing group statistics ADAM stats variable(s) containing group statistics

adam_plot_MVPA *adam_plot_BDM_weights*	In: Out:	ADAM stats variable(s) containing group statistics publication-ready graphs of performance metrics and/or topographical maps of forward transformed weights

The experiment

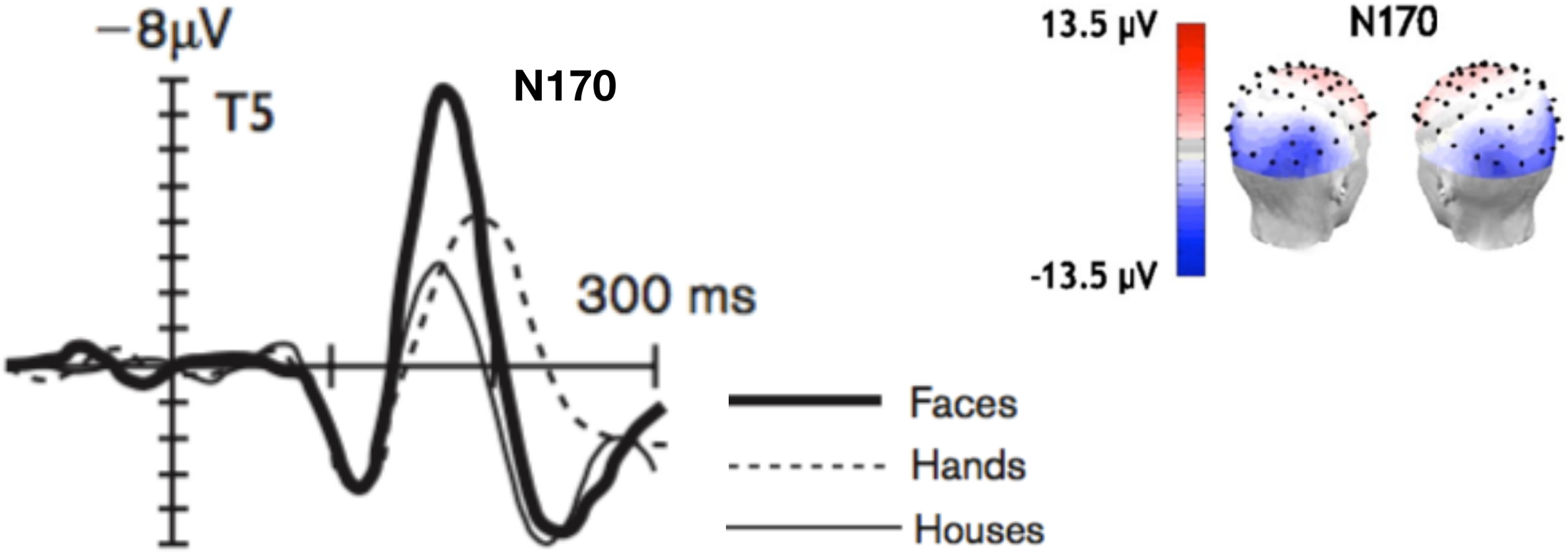
Open dataset, containing simultaneously recorded EEG/MEG

Pre-stimulus: 400-600 ms jittered
Stimulus: 800-1000 ms jittered (scrambled/familiar/unfamiliar)
ITI: 1700 ms
Response: indicate 'symmetry'



Wakeman, D. G., & Henson, R. N. OpenfMRI ds000117 (2014). https://openfmri.org/dataset/ds000117/
Wakeman, D. G., & Henson, R. N. (2015). A multi-subject, multi-modal human neuroimaging dataset. *Scientific Data*, 2.

Face-selective N170 component in EEG



Eimer, M. (2000). The face-specific N170 component reflects late stages in the structural encoding of faces. *Neuroreport*, 11(10), 2319–2324.

Table of experimental design

Numbers in the table denote event codes

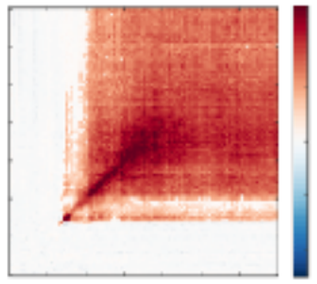
	Famous	Nonfamous	Scrambled
First presentation	5	13	17
Immediate repeat	6	14	18
Delayed repeat	7	15	19

Factor ‘stimulus type’

Factor ‘stimulus repetition’

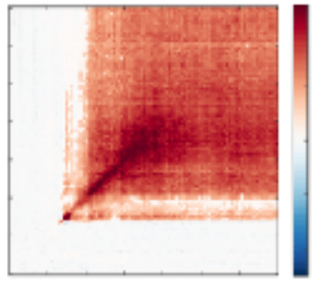
This factor ‘stimulus repetition’ exists in the experiment, but we do not analyze it, we only look at the first presentations

ADAM analysis pipeline



1. Pre-processing
   (can do this using EEGLAB, do not need ADAM)
2. First level (single subject) analyses are computed and stored
3. Compute group statistics after reading in single subject results
4. Visualize group statistics in a plot

Practical, use the ADAM toolbox



- **Part I:** Group analysis of raw EEG/MEG:
  - Comparing ERPs to MVPA
- **Part II:** Group analysis:
  - Temporal generalization time-by-time matrix
- **Part III:** First-level (single subject) analysis of raw data
- **Part IV:** Group analysis
  - Time-frequency (TFR), time-by-frequency / temporal generalization
- **Part V:** First-level (single subject) analysis of TFR data + play around with the scripts/data

unconventional order ☺

Questions?

Course overview

- **Day 1 (introduction)**
  - lecture 1: History and electrophysiological basis of EEG
  - lecture 2: Backward decoding models in MVPA: concepts and analytical approach
  - lecture 3: Advantages of MVPA, the temporal generalization method
  - lecture 4: Architecture of the ADAM toolbox and explaining the experiment of the practical
  - *Afternoon: practical*
- **Day 2 (advanced)**
  - lecture 1: Multiple comparisons, MVPA experimental design, mapping brain to brain/behavior
  - lecture 2: Forward encoding models in MVPA: how do they work, concepts and analytical approach
  - *Afternoon: practical, analyze your own data and/or a supplied dataset*