Data normalization and its impact on source modelling and multivariate analysis

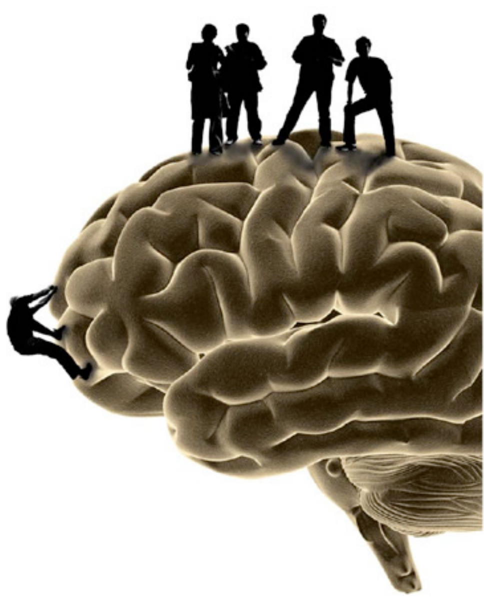
Alexandre Gramfort
http://alexandre.gramfort.net

Inria, Parietal Team

PARIETAL

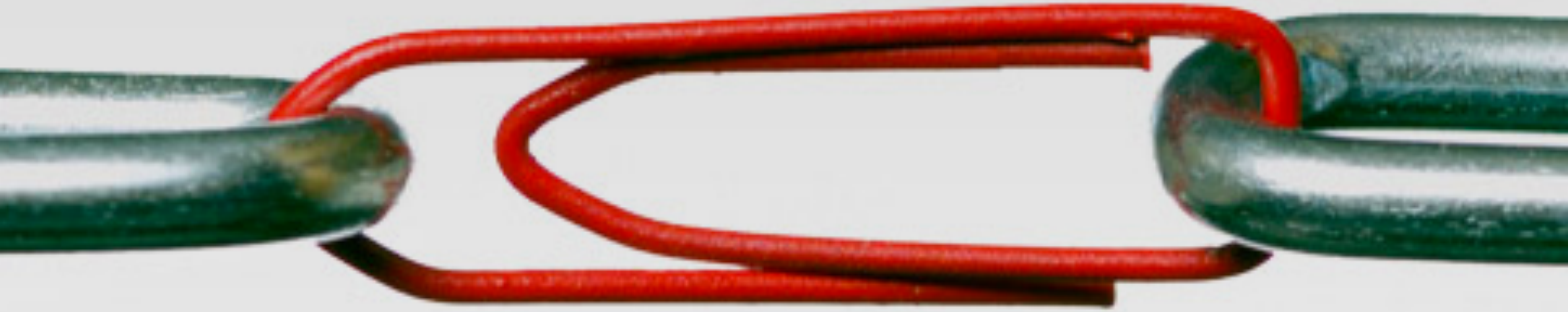
informatics mathematics Inria

université PARIS-SACLAY



OHBM - 2020

Your analysis pipeline will break at the weakest link so every step matters

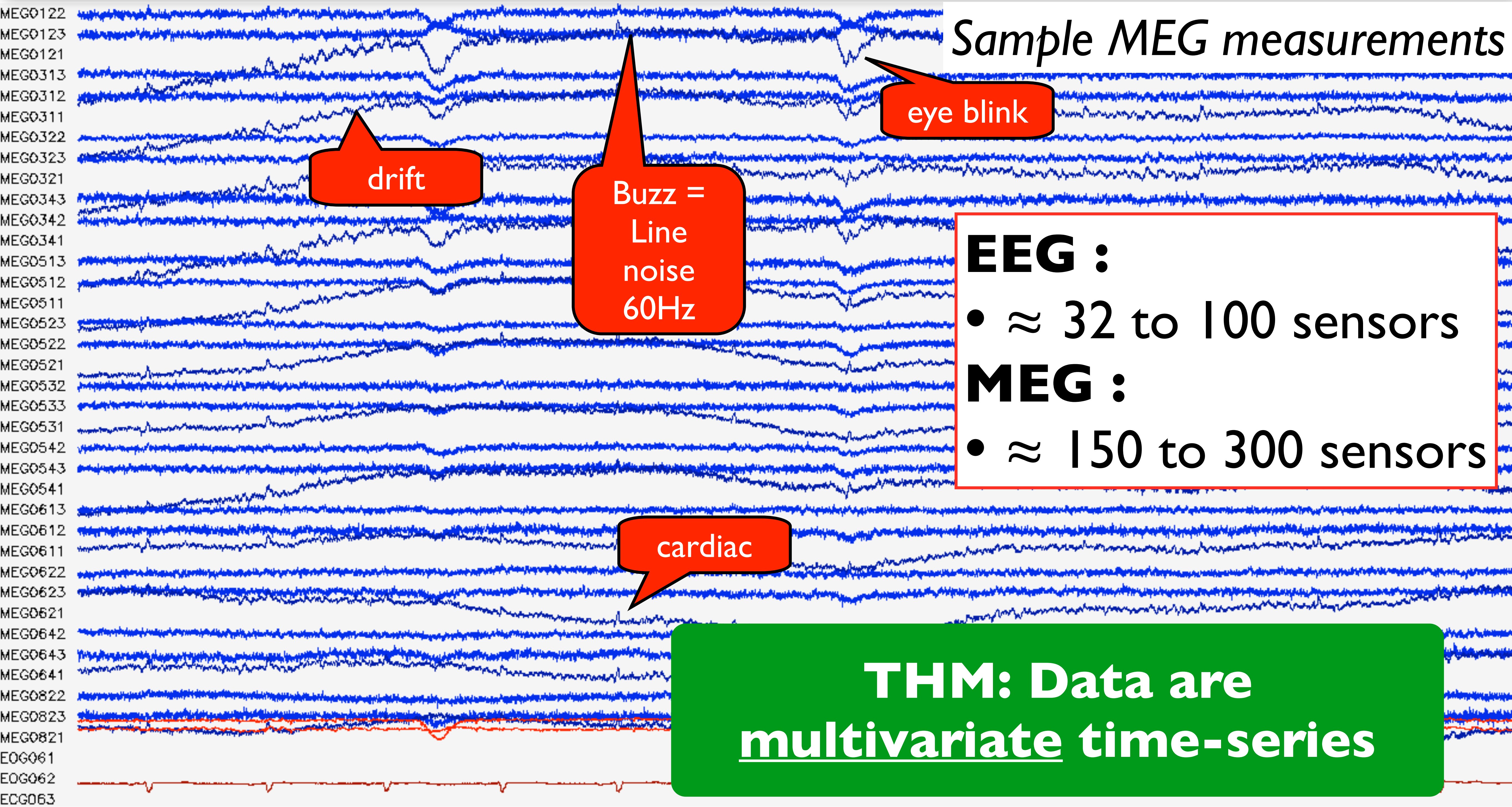


THM means Take Home Message (not Theorem)

This logo means that you have a URL to a page with a notebook

M/EEG Measurements

Sample MEG measurements



EEG :

- ≈ 32 to 100 sensors

MEG :

- ≈ 150 to 300 sensors

THM: Data are <u>multivariate</u> time-series

Time frame: 10 seconds

≈ 1000 samples / s

Multivariate models boost SNR

Consider 2 EEG signals such that:

$$x_1(t) = 2s(t) + n(t)$$
$$x_2(t) = s(t) + n(t)$$

where $s(t)$ is the source and $n(t)$ the additive noise, which is common.

Multivariate models boost SNR

Consider 2 EEG signals such that:

$$x_1(t) = 2s(t) + n(t)$$
$$x_2(t) = s(t) + n(t)$$

where $s(t)$ is the source and $n(t)$ the additive noise, which is common.

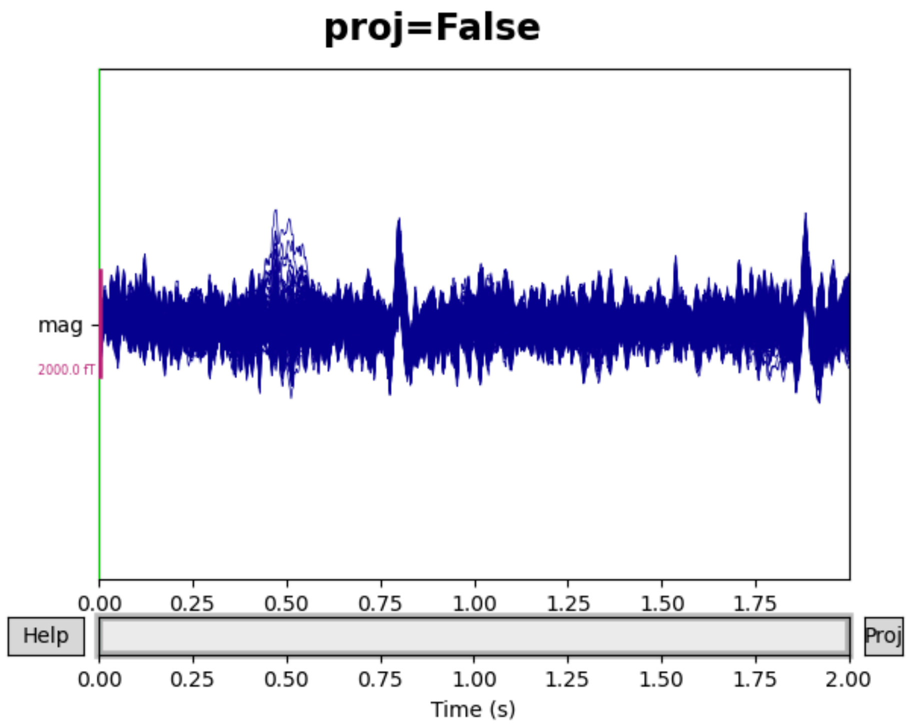
This implies that:

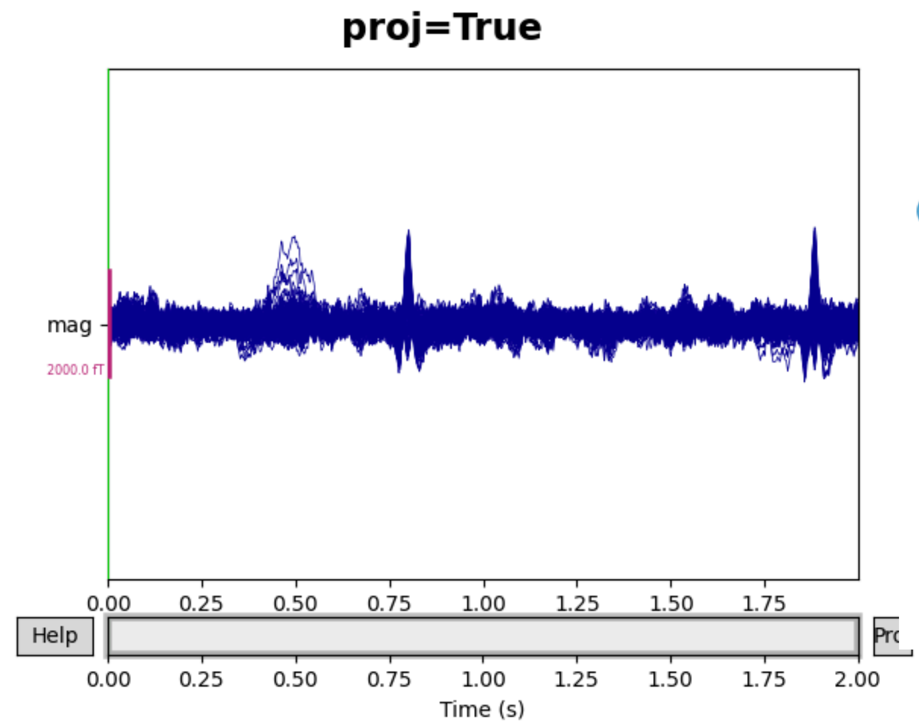
$$x_2(t) - x_1(t) = s(t)$$

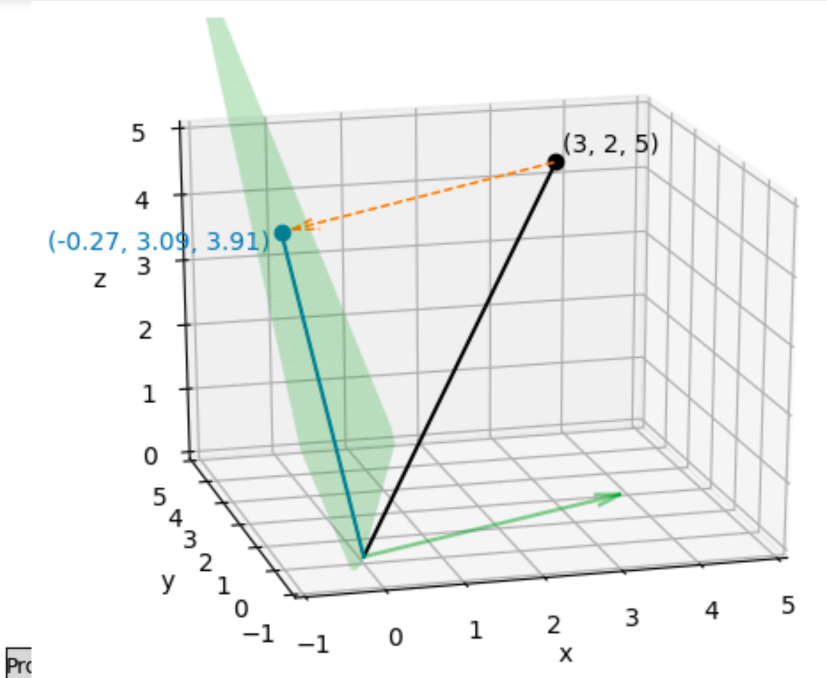
Only signal remains !

THM: By subtracting the signals on can improve the SNR even if noise is huge

Signal Subspace Projector

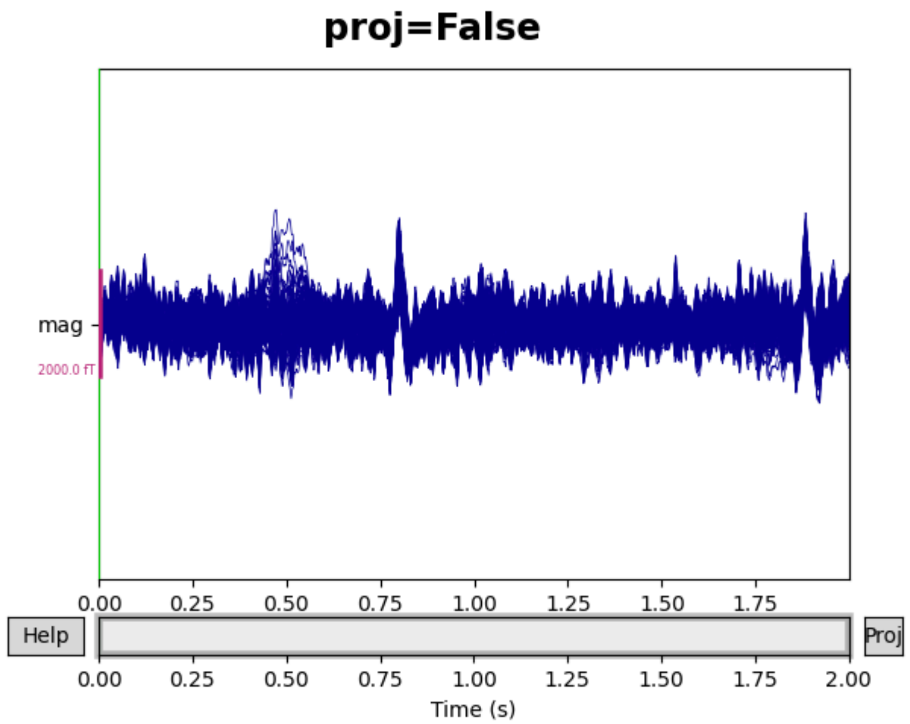


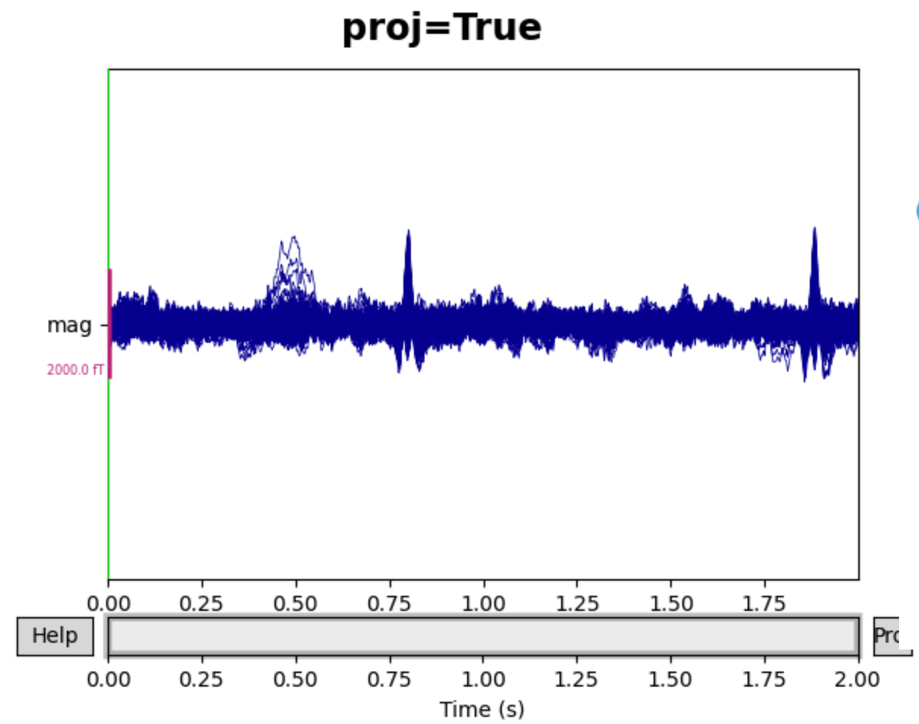


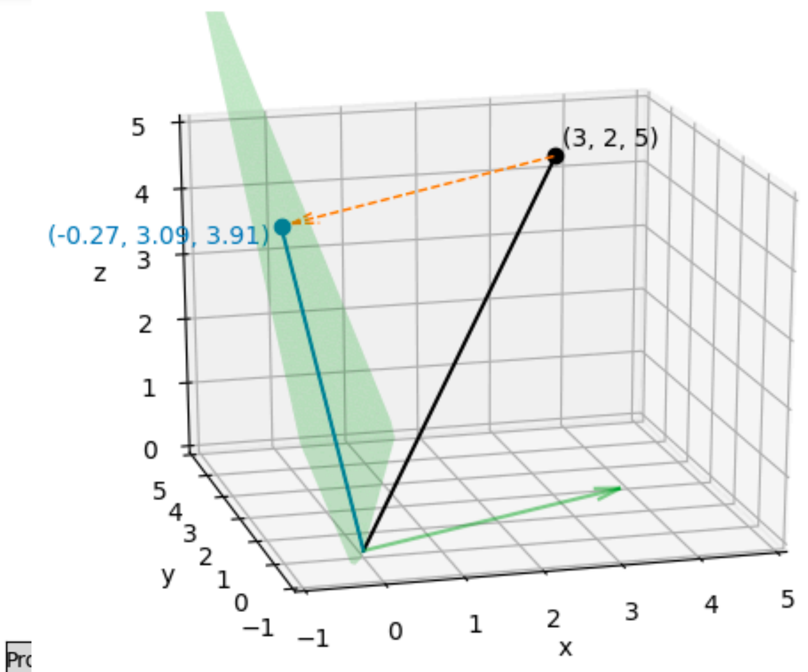


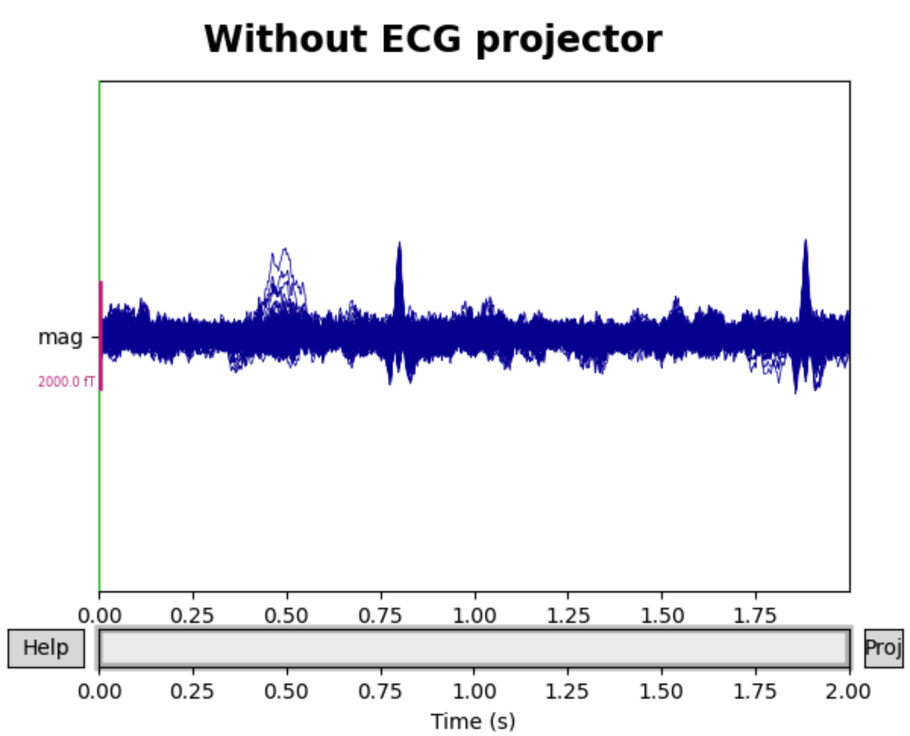
https://mne.tools/stable/auto_tutorials/preprocessing/plot_45_projectors_background.html

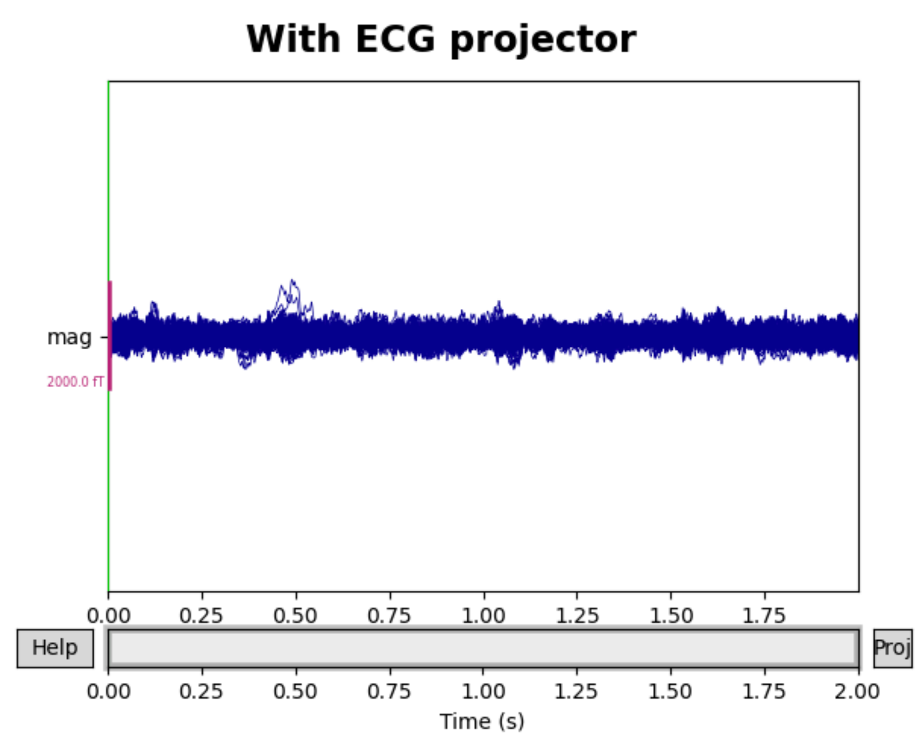
Signal Subspace Projector





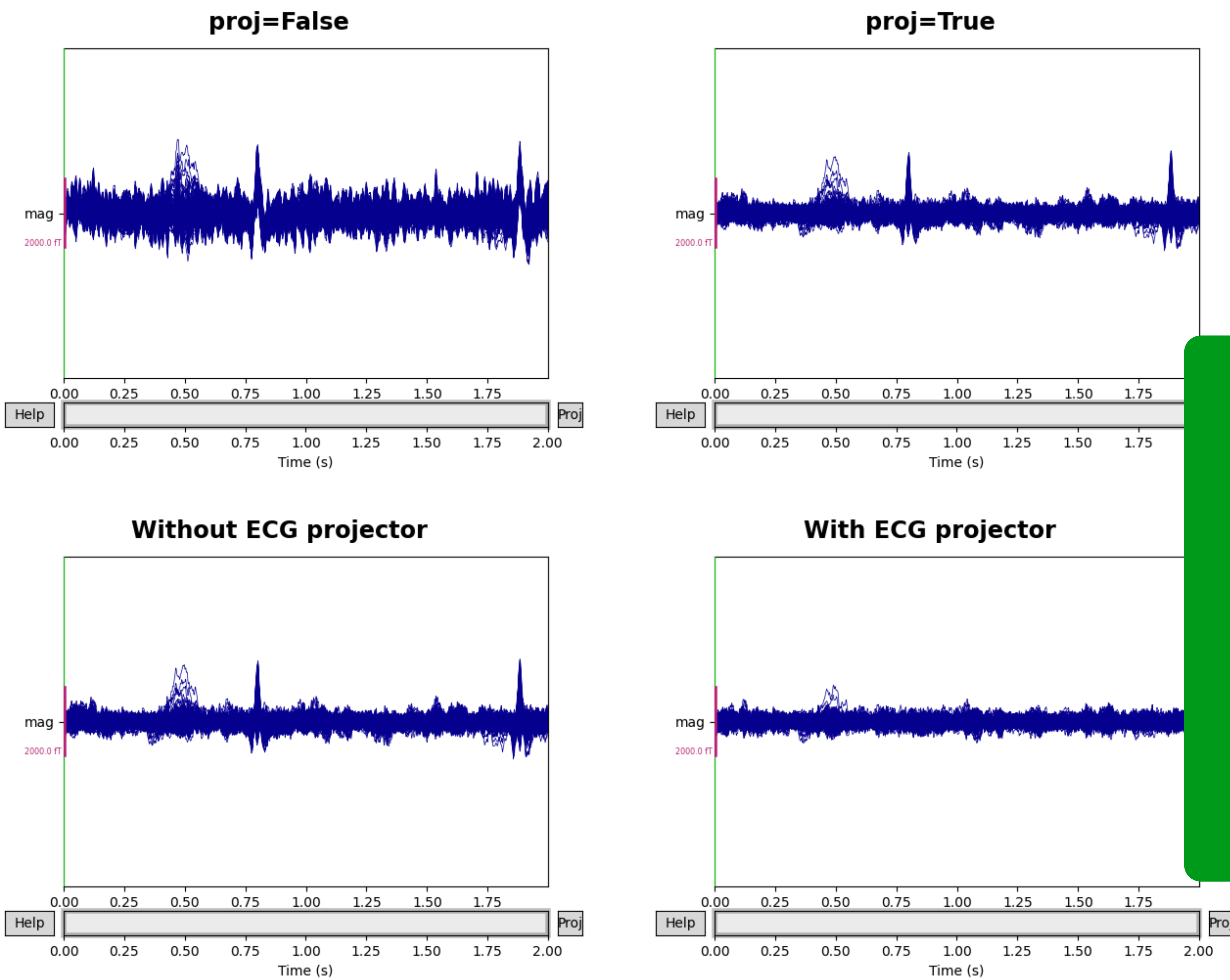


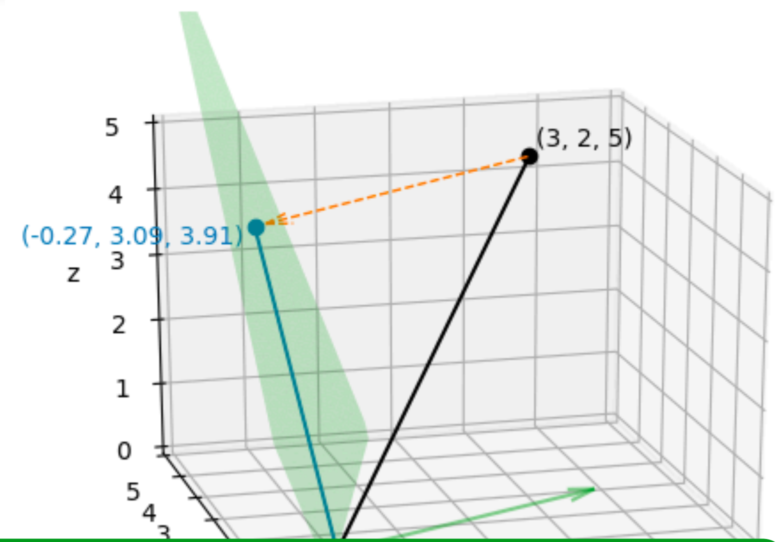




https://mne.tools/stable/auto_tutorials/preprocessing/plot_45_projectors_background.html

Signal Subspace Projector

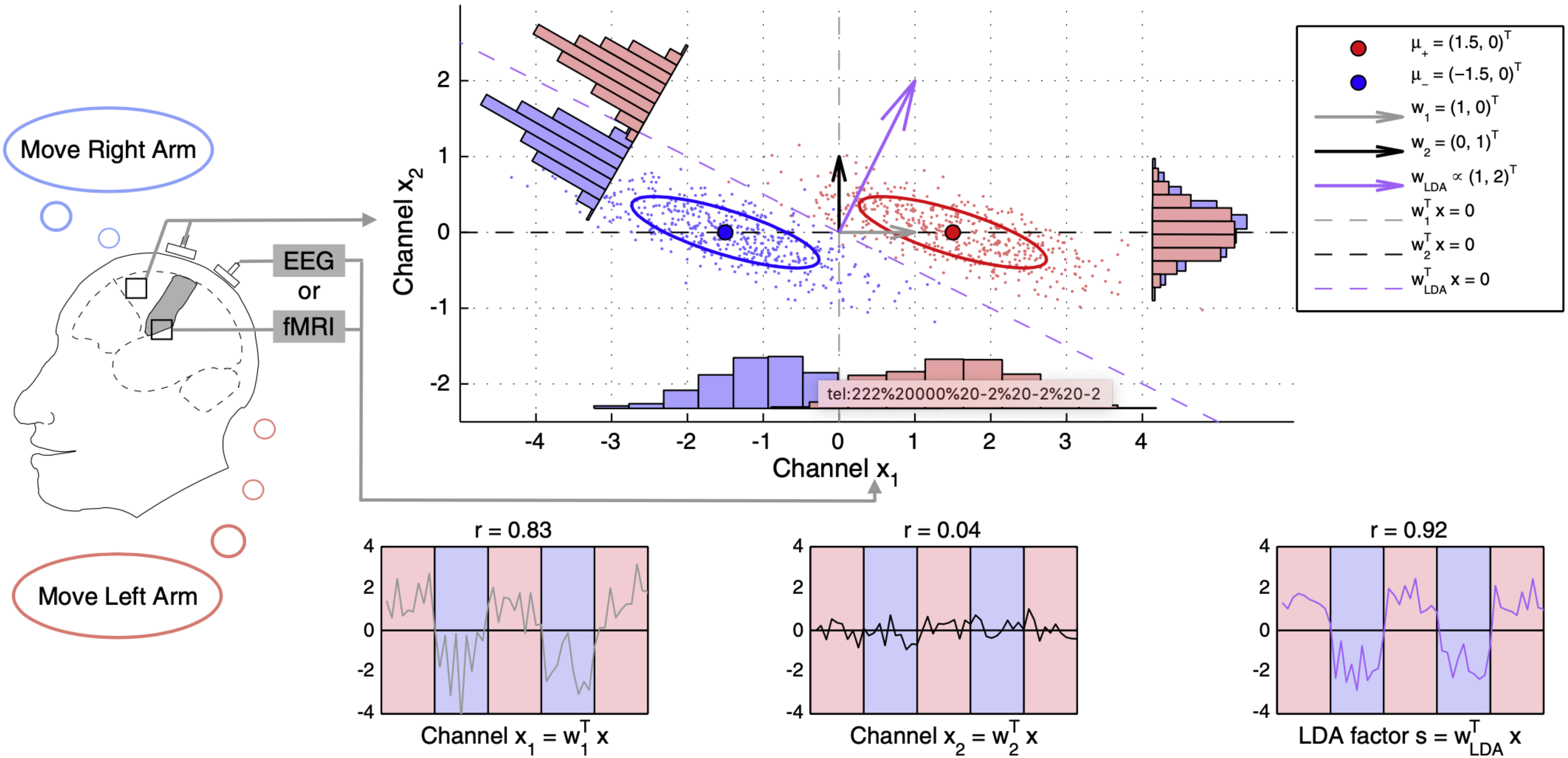




Using multiples signals one can remove environmental noise but also physiological noise

https://mne.tools/stable/auto_tutorials/preprocessing/plot_45_projectors_background.html

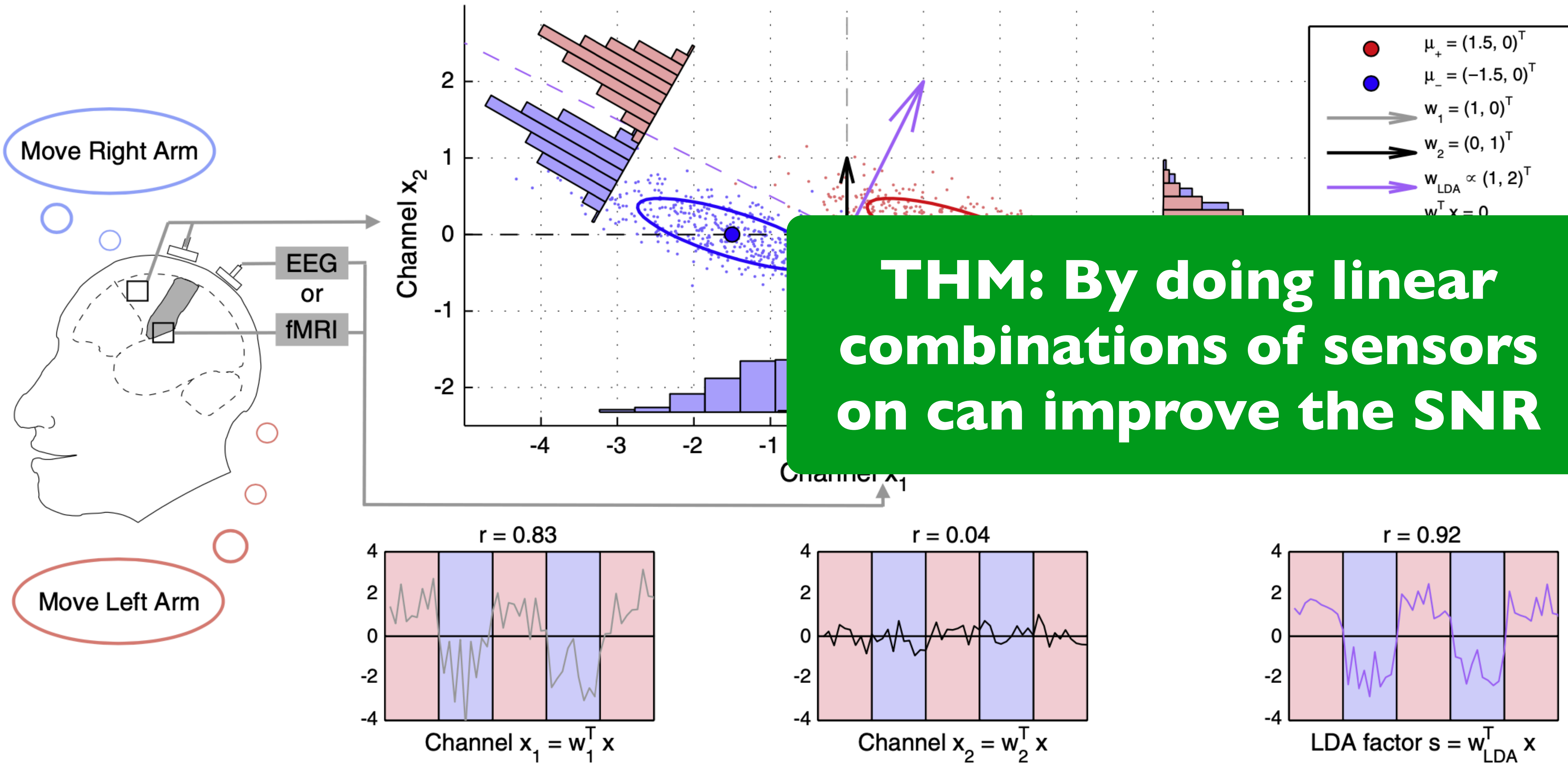
Multivariate models boost SNR



from [Haufe et al. (2014) On the interpretation of weight vectors of linear models in multivariate neuroimaging, NeuroImage]

https://mne.tools/stable/auto_examples/decoding/plot_linear_model_patterns.html

Multivariate models boost SNR



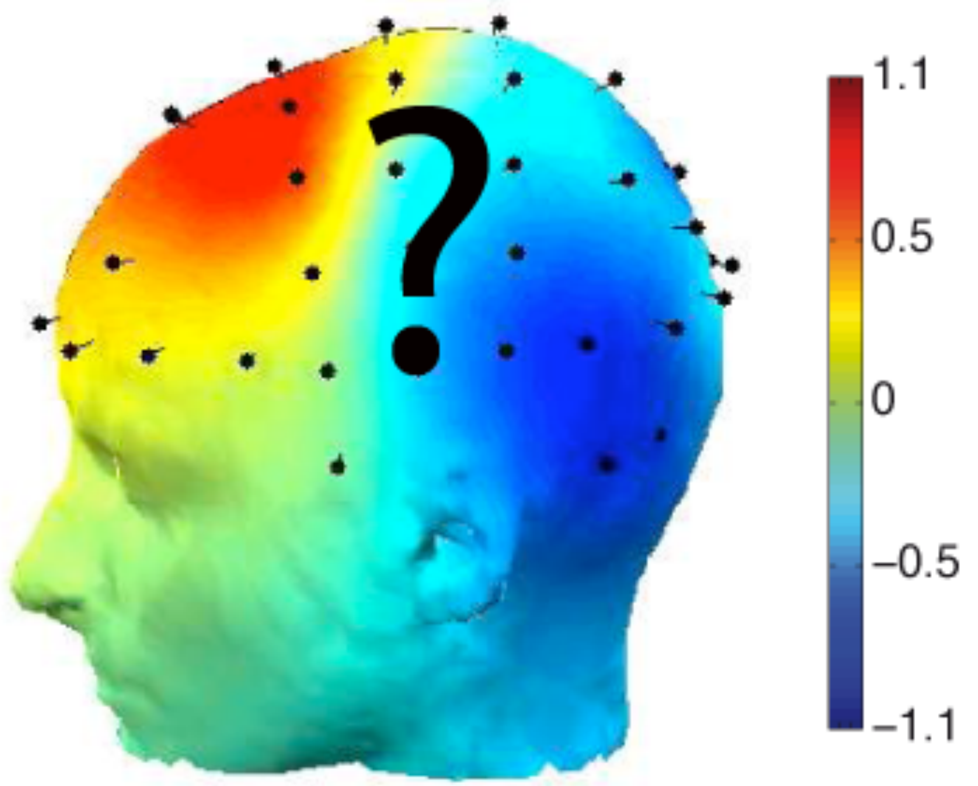
THM: By doing linear combinations of sensors on can improve the SNR

from [Haufe et al. (2014) On the interpretation of weight vectors of linear models in multivariate neuroimaging, NeuroImage]

https://mne.tools/stable/auto_examples/decoding/plot_linear_model_patterns.html

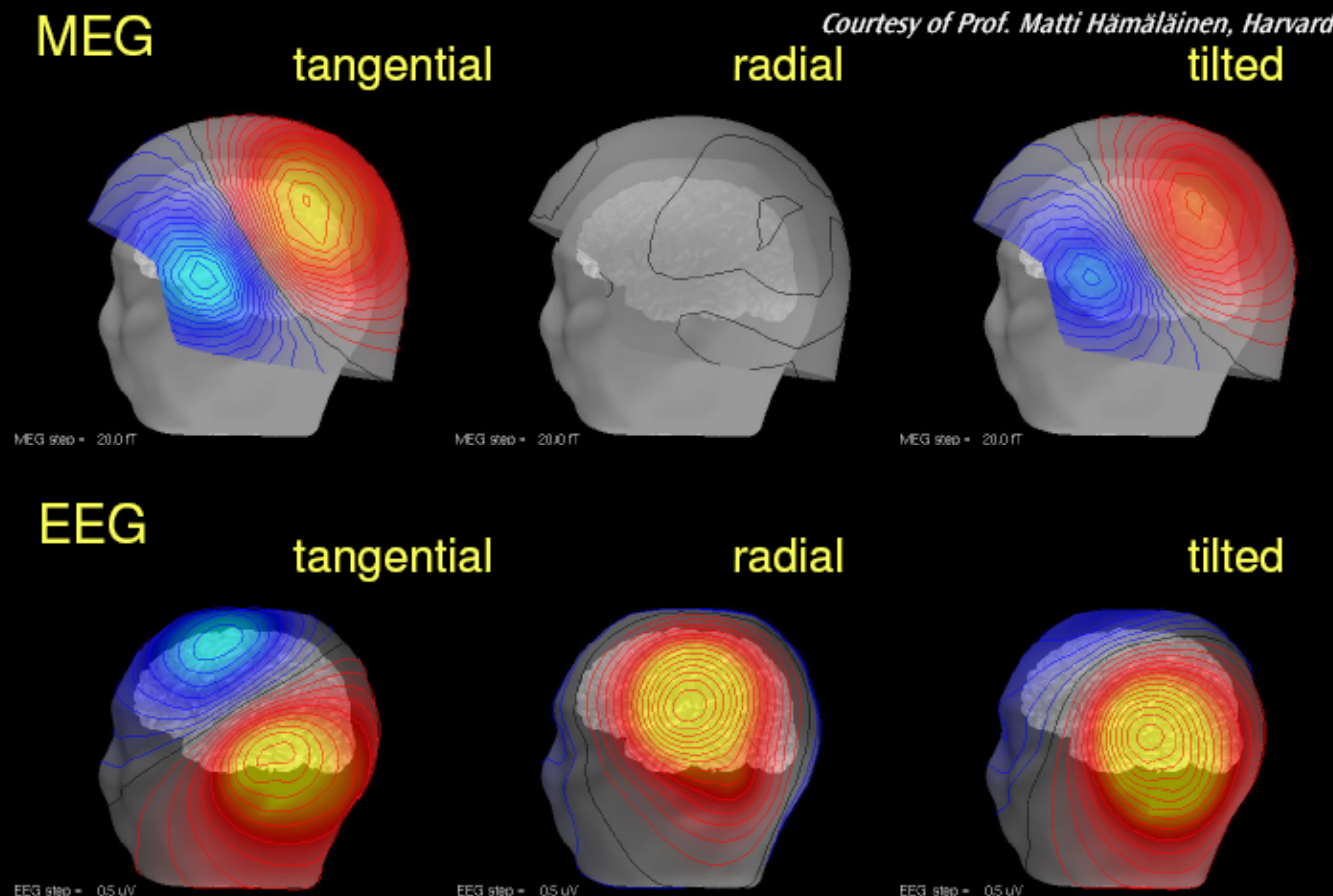
EEG/MEG Source imaging

Find the current generators that produced the M/EEG measurements

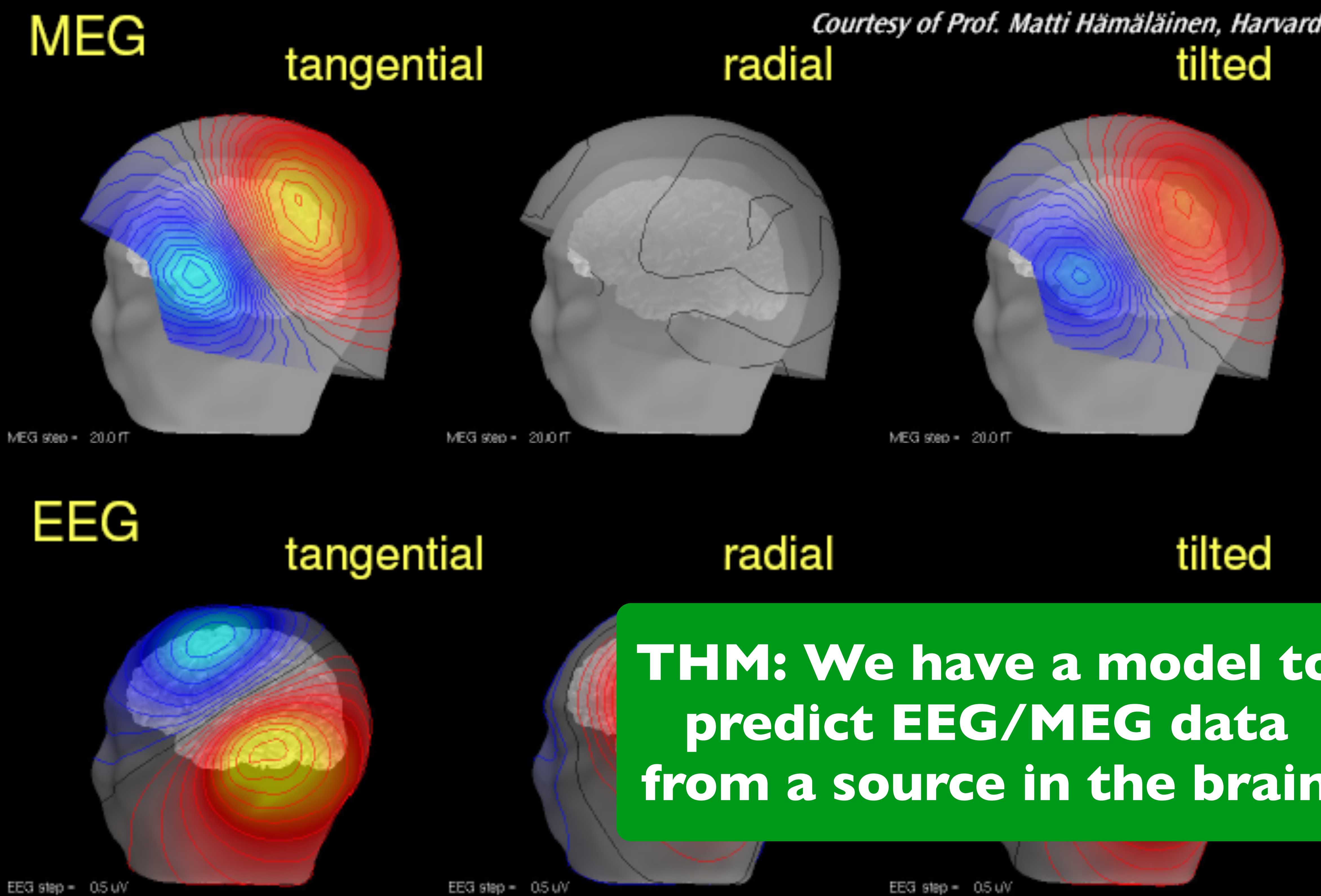


THM: <u>Multivariate</u> models are also at the core of source imaging

MEG vs. EEG

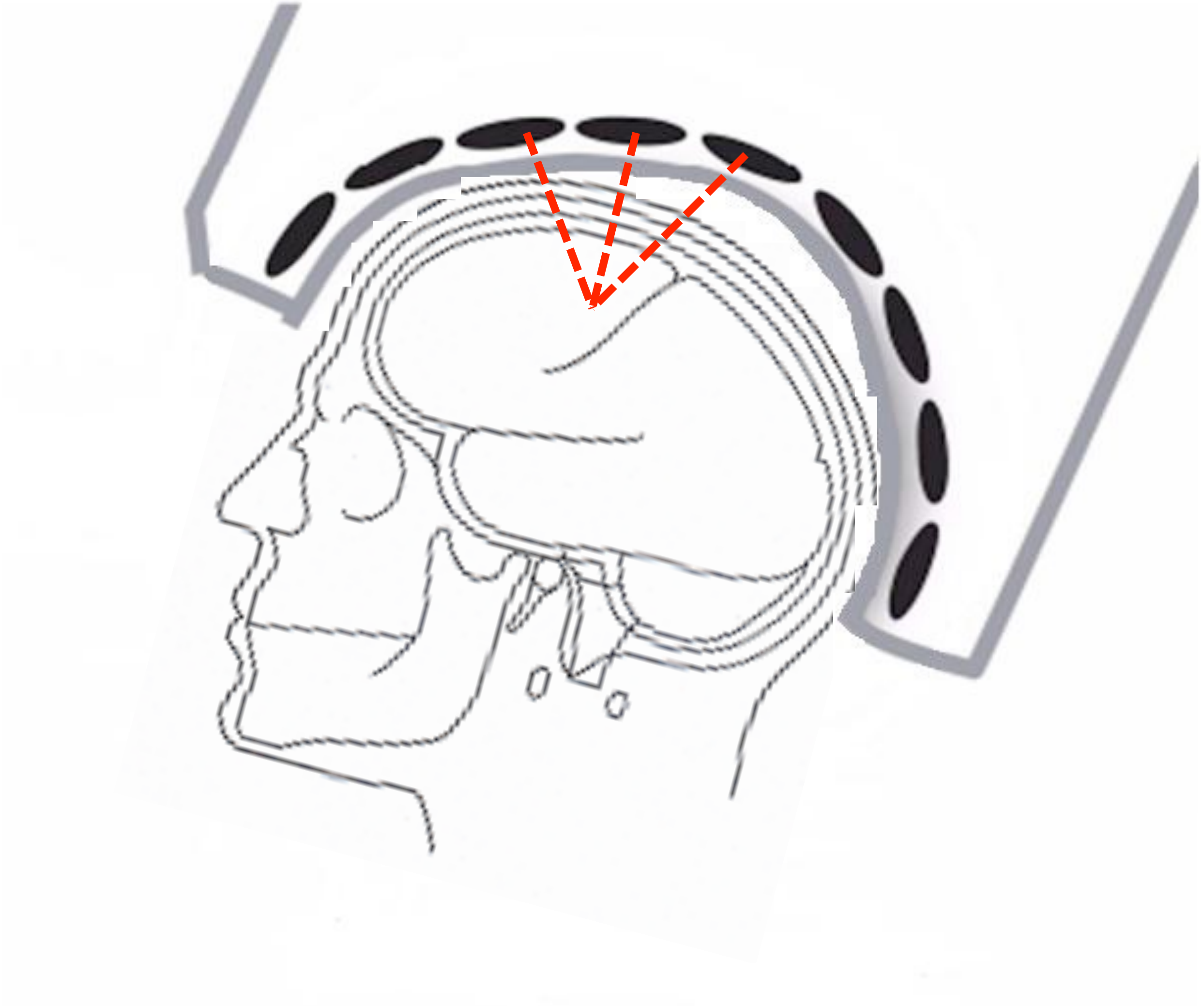

Courtesy of Prof. Matti Hämäläinen, Harvard
MEG
tangential
radial
tilted
MEG step = 20.0 fT
MEG step = 20.0 fT
MEG step = 20.0 fT
EEG
tangential
radial
tilted
EEG step = 0.5 uV
EEG step = 0.5 uV
EEG step = 0.5 uV

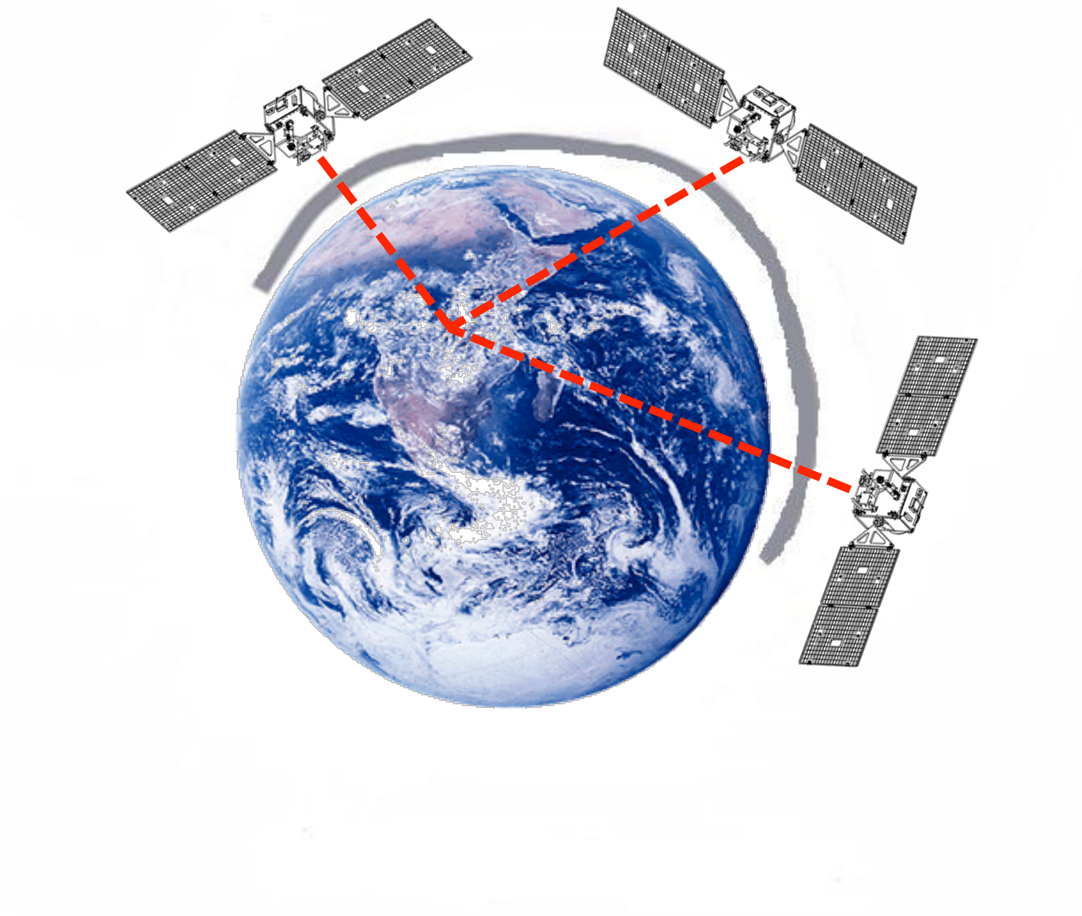
MEG vs. EEG



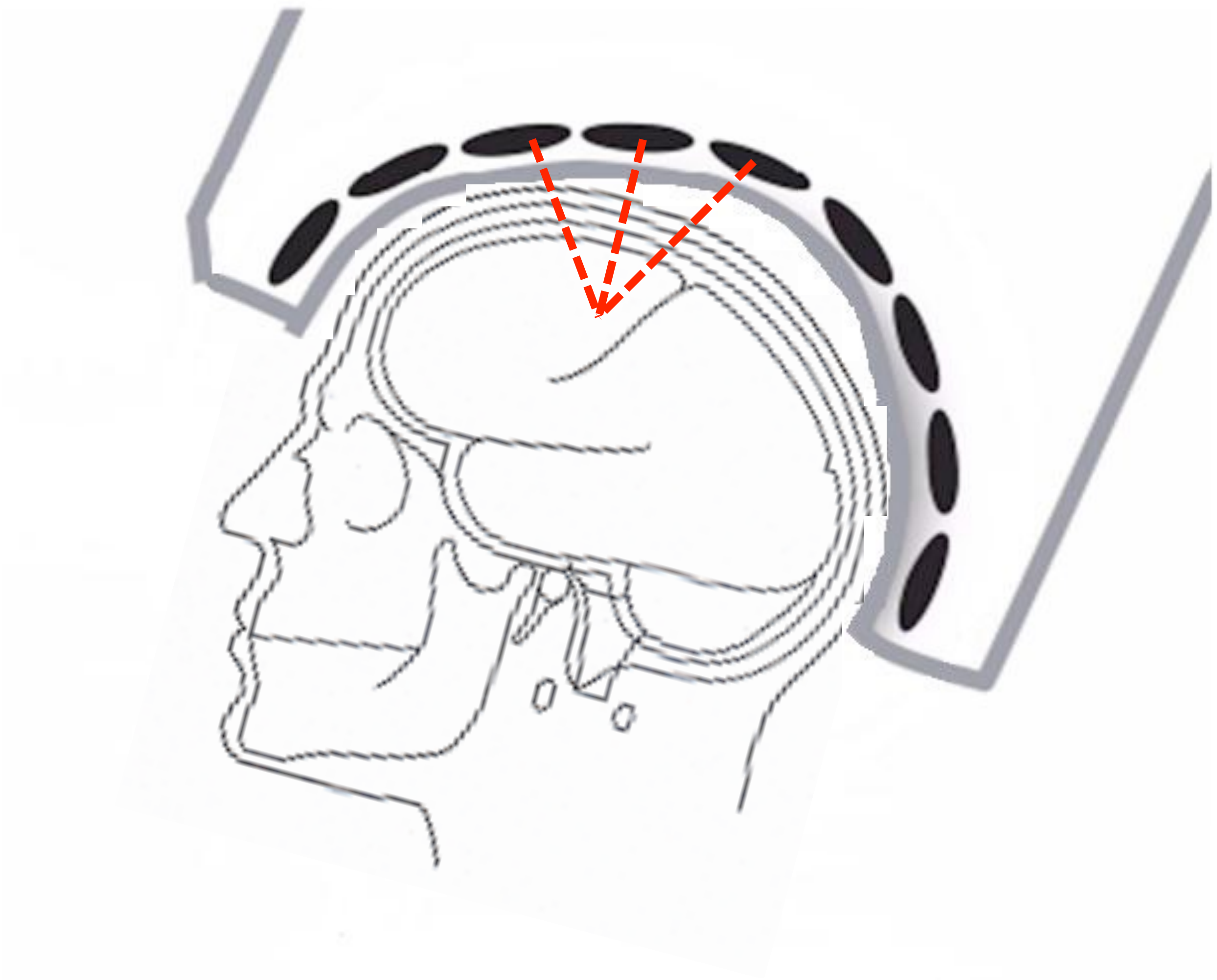
THM: We have a model to predict EEG/MEG data from a source in the brain

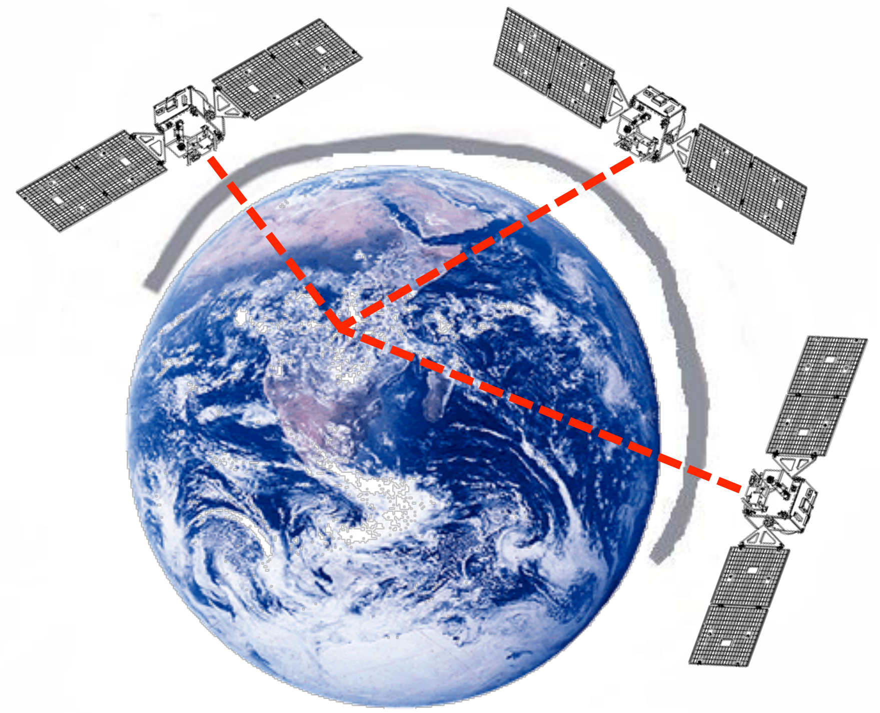
Dipole fitting





Dipole fitting





How would you solve?

Dipole fitting

How would you solve?

Try every possible location and orientation and take the “best”

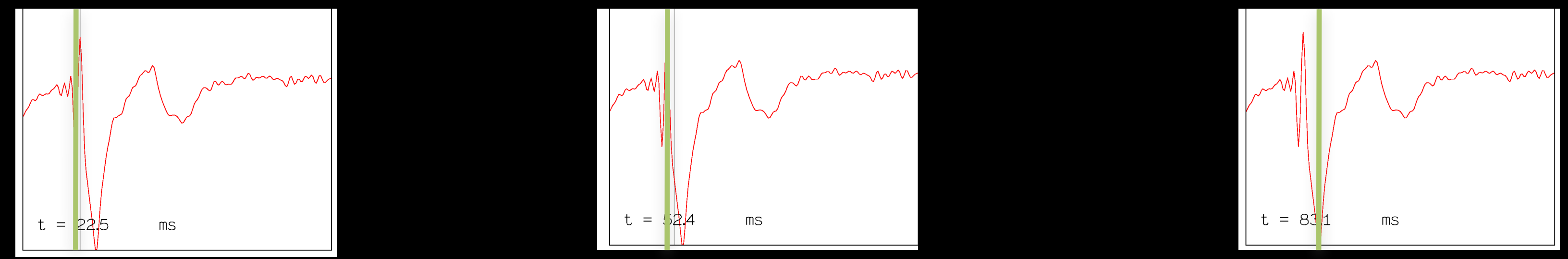
$$\boxed{\hat{\vec{r}} = \arg\min_{\vec{r} \in \mathrm{Brain}} \|m - g(\vec{r})\|_2^2}$$

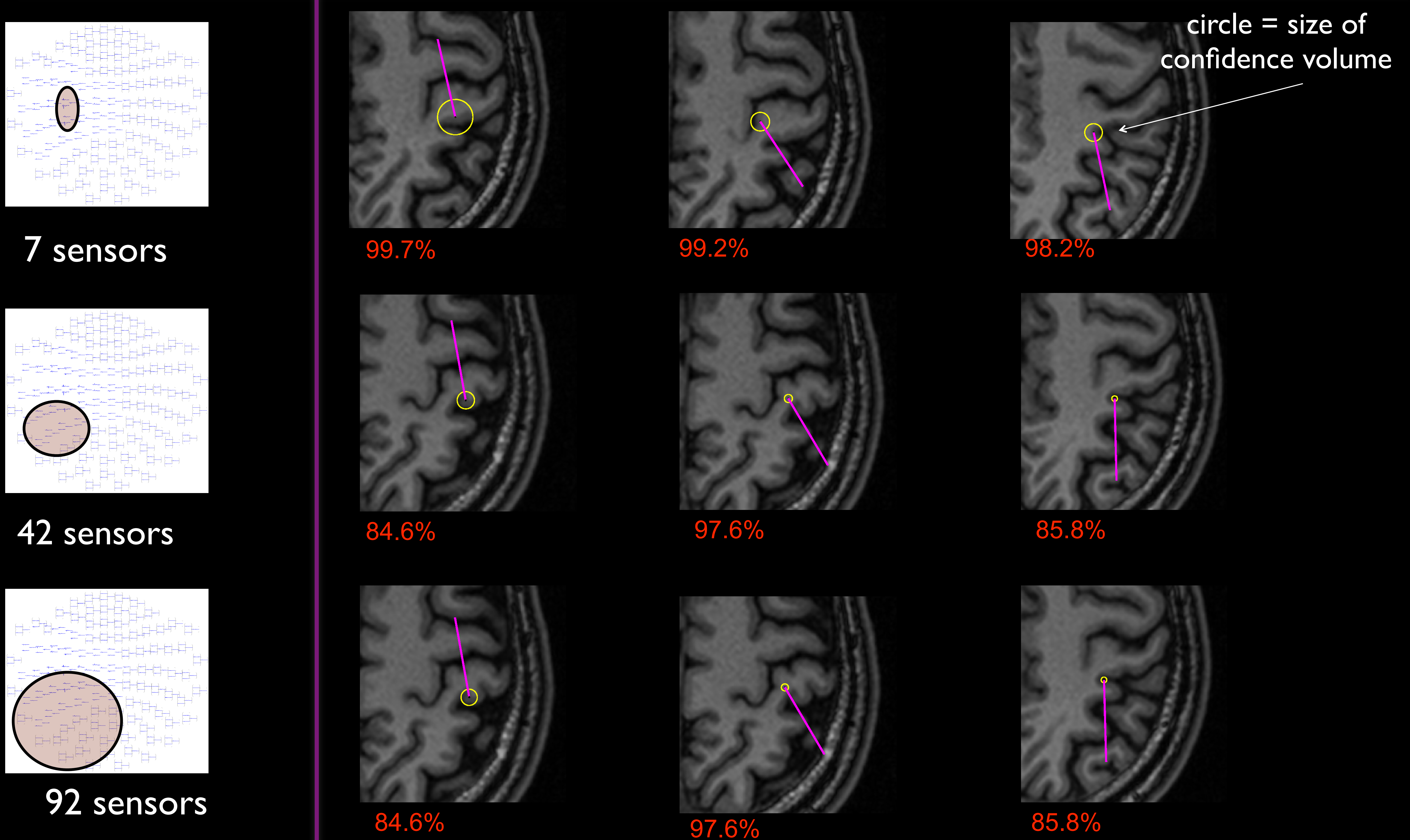
where m is the EEG data

and g is forward model

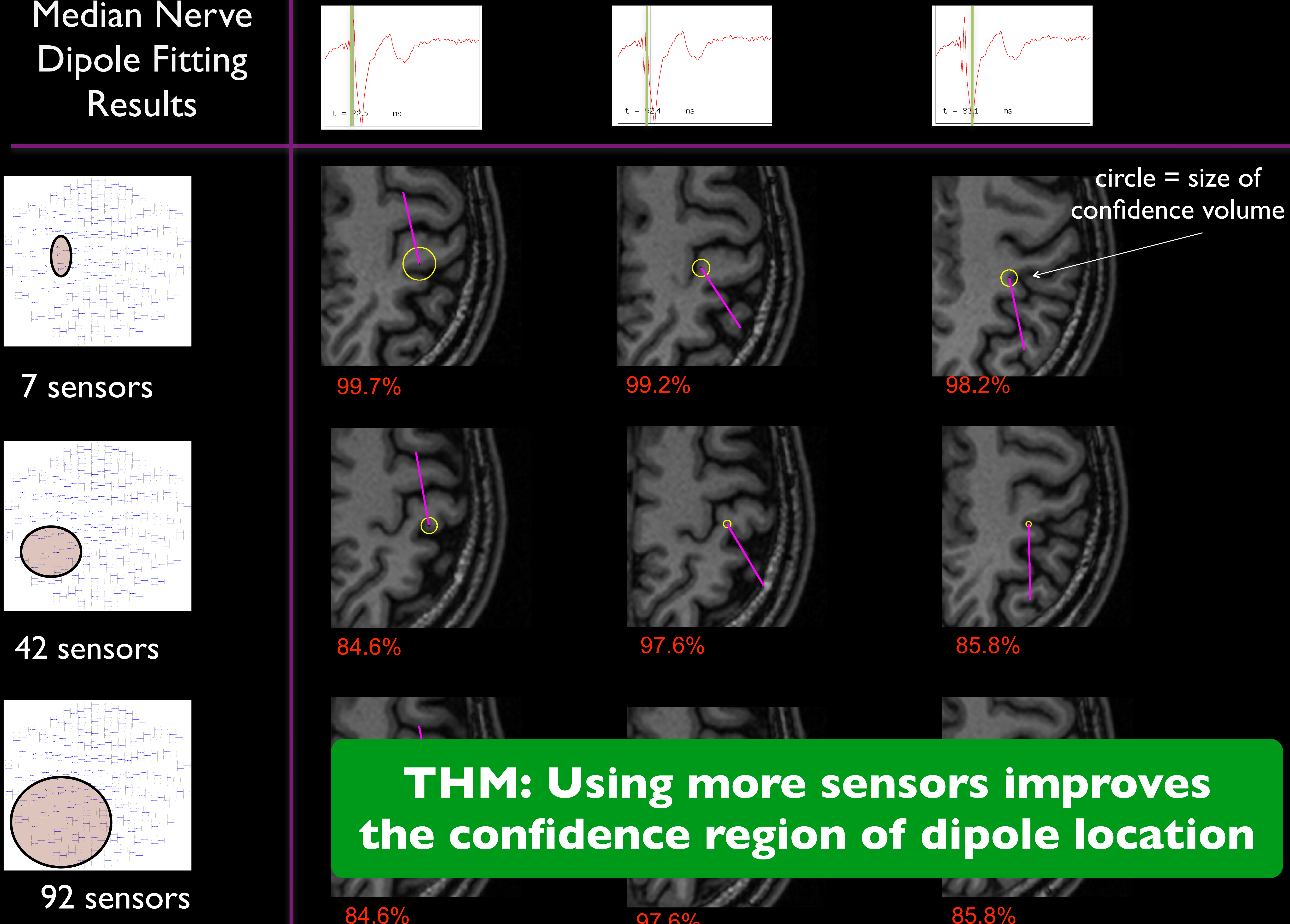
with $\|x\|_2^2 = \sum_i x_i^2$ (Euclidean norm)

Median Nerve Dipole Fitting Results



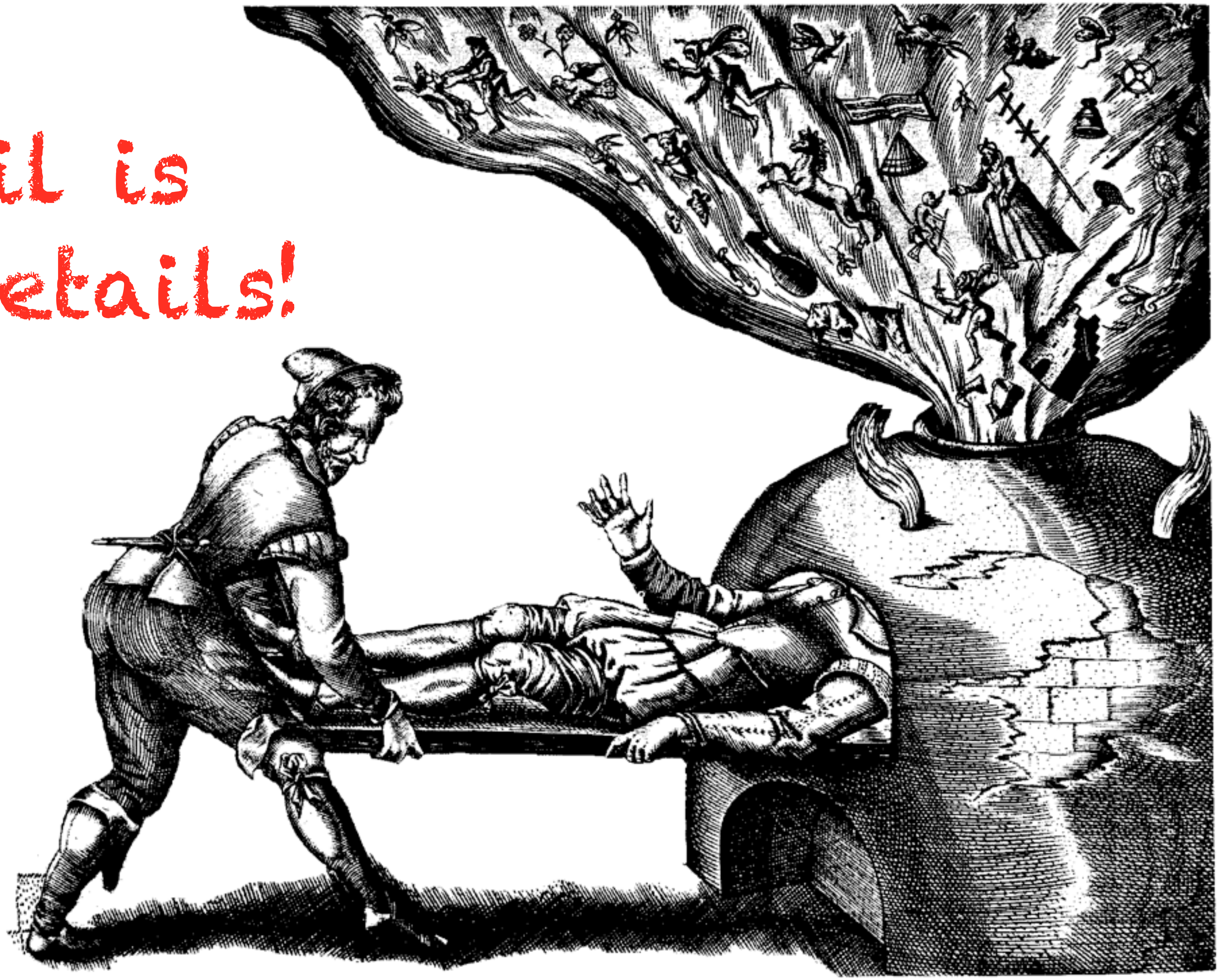


Median Nerve Dipole Fitting Results

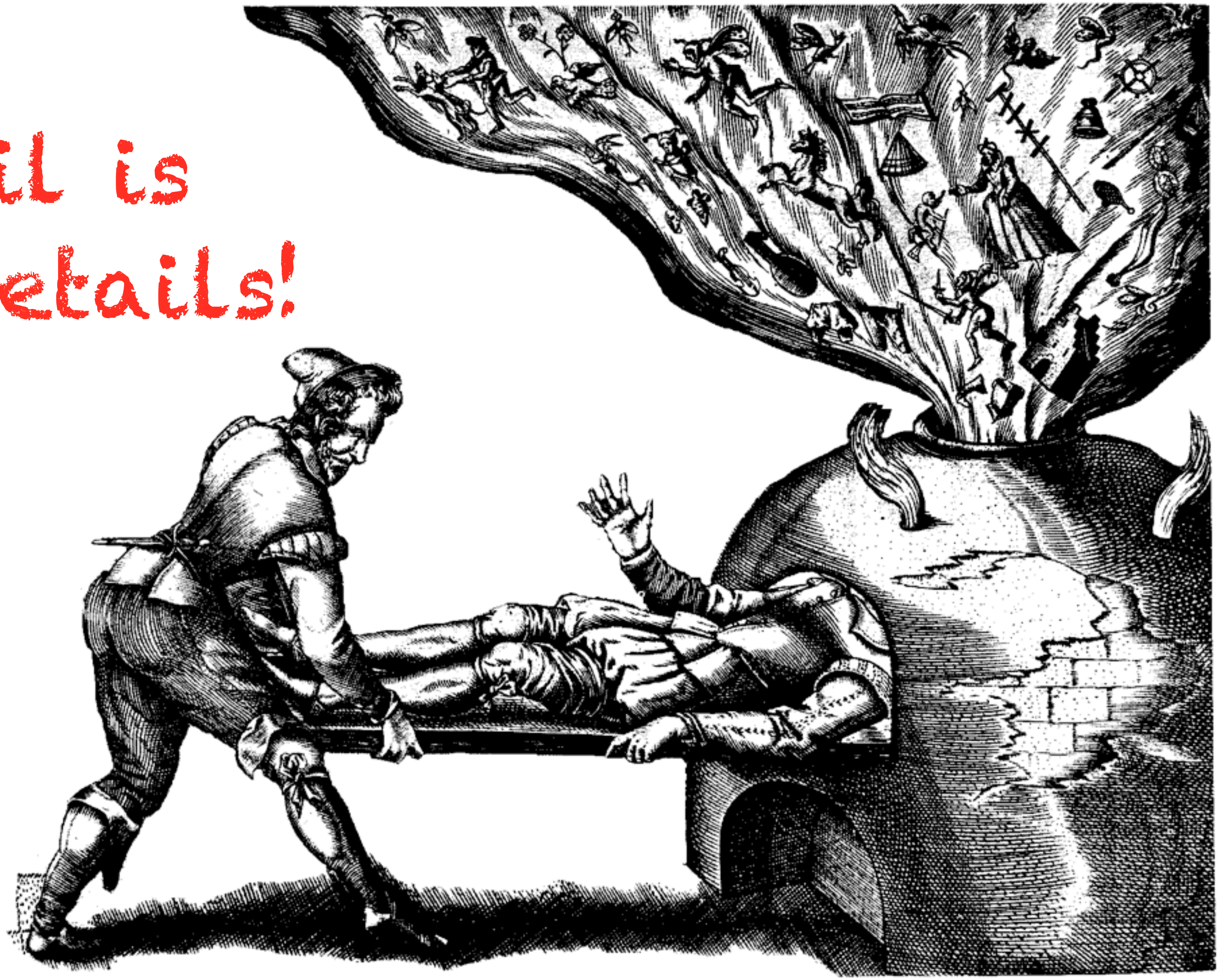


THM: Using more sensors improves the confidence region of dipole location

The devil is
in the details!



The devil is
in the details!

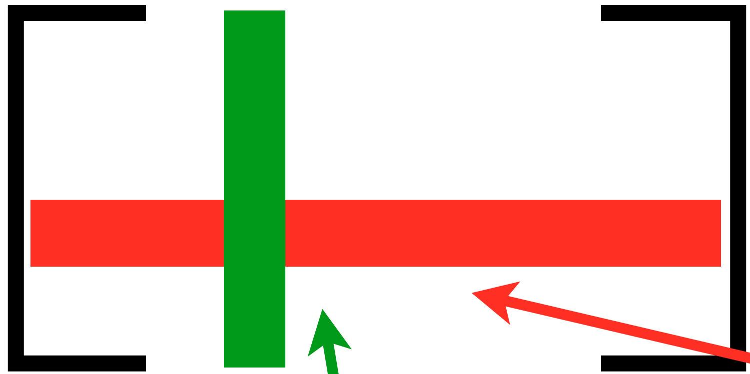


IDEA

If you have a noisy sensor you
do <u>not</u> want to "fit it" as well

M/EEG Measurements: Notation

M =

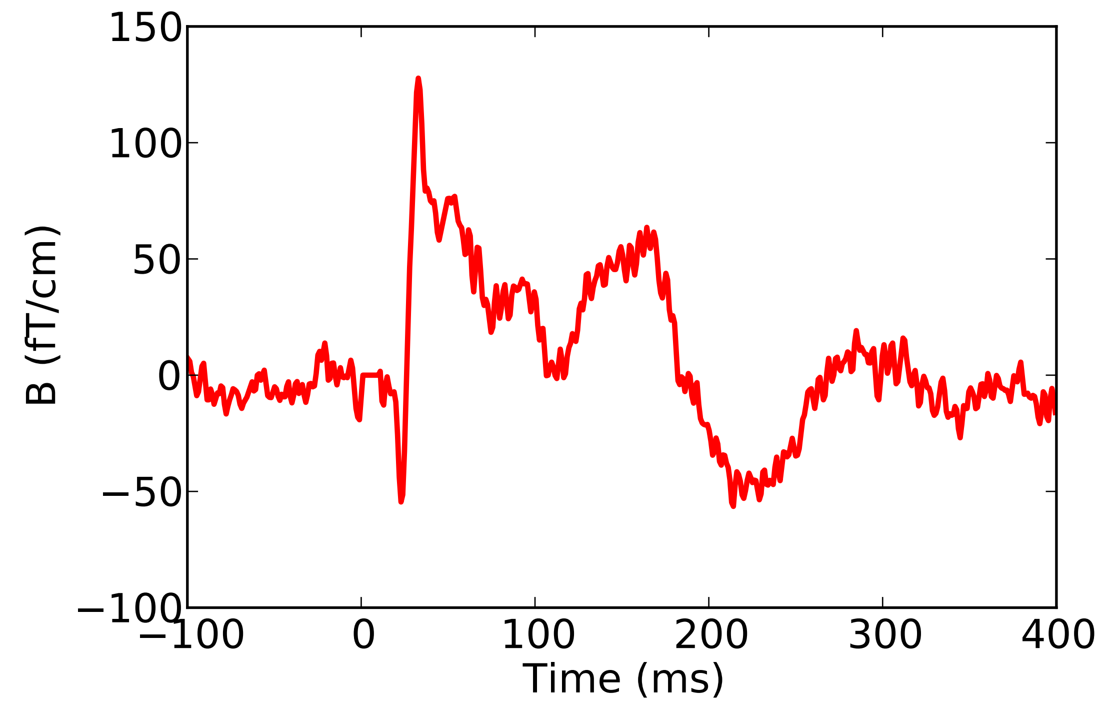


MEG and/or EEG

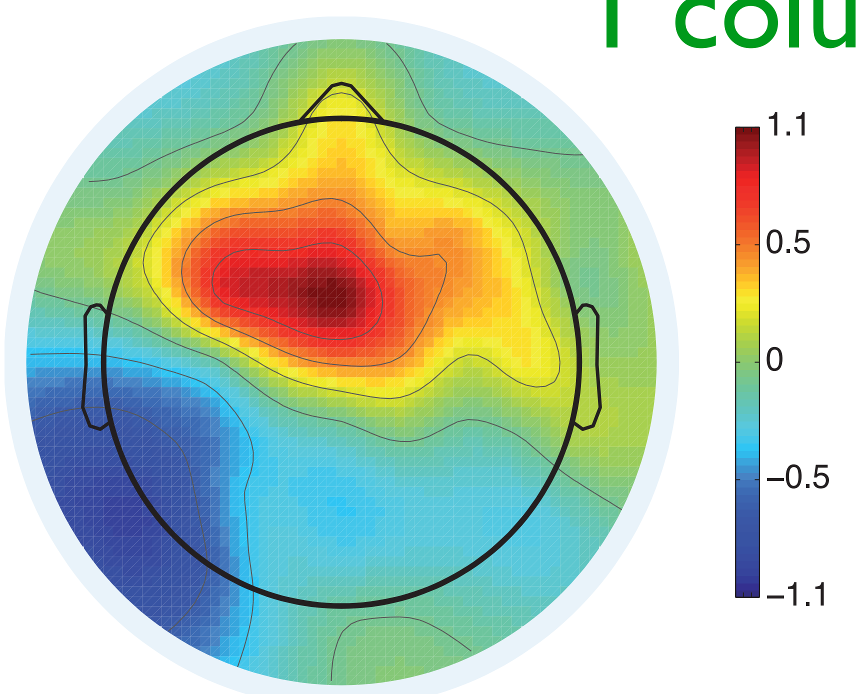
one row = one sensor

one column = one time instant

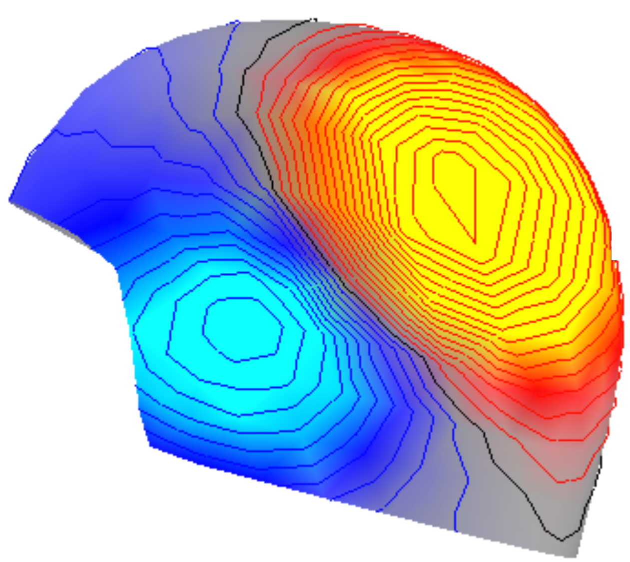
1 column



1 time series on 1 sensor

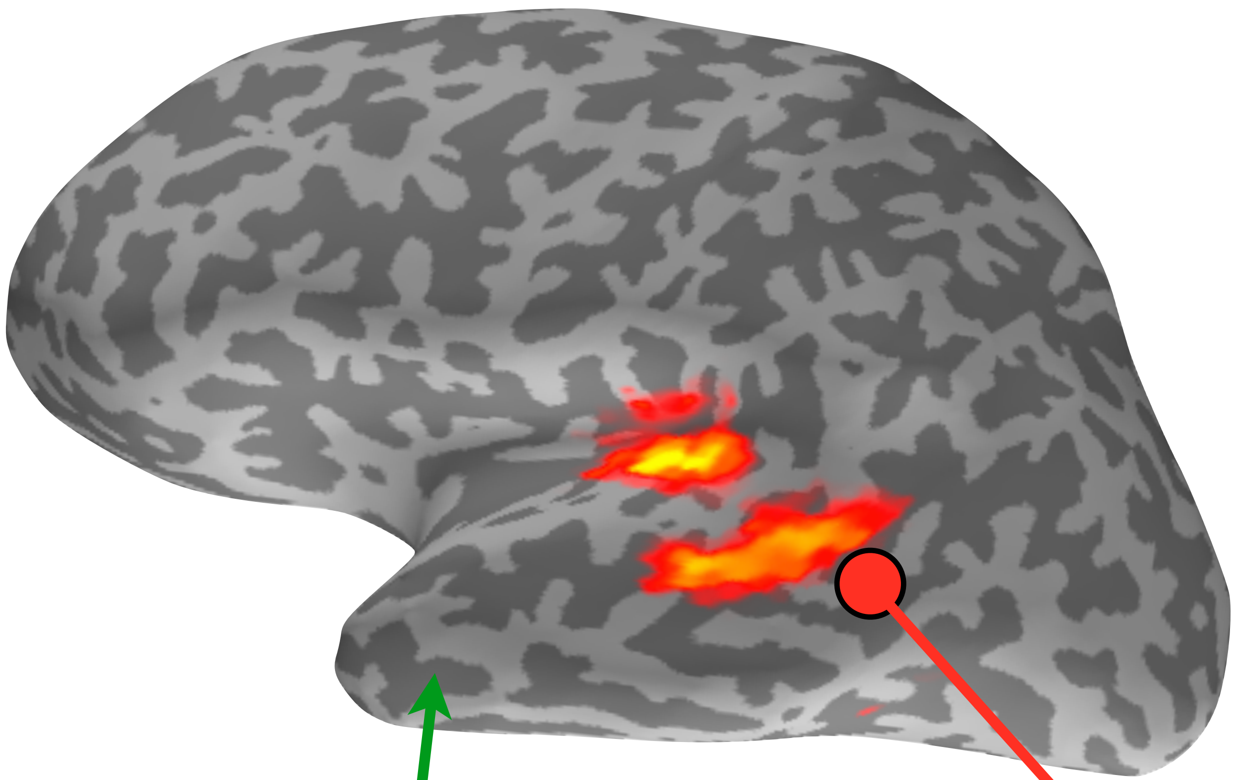


2D topography

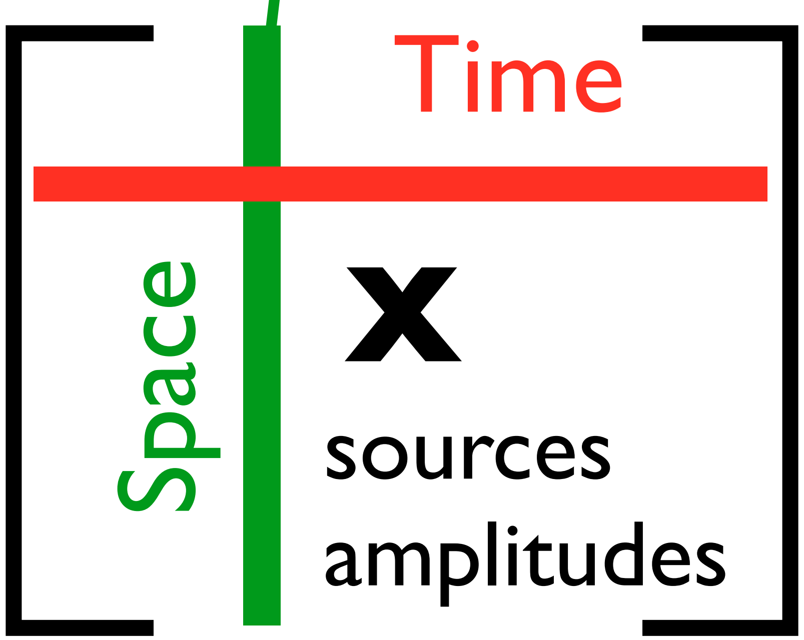


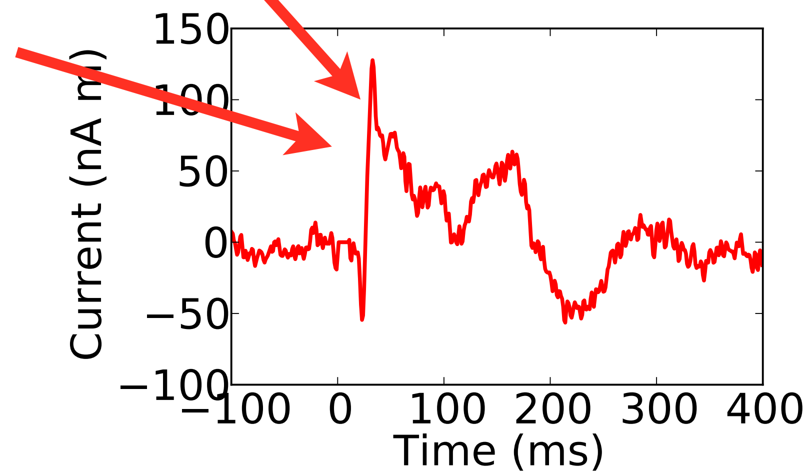
3D topography

The source model



Position 5000 candidate sources over each hemisphere (e.g. every 5mm)



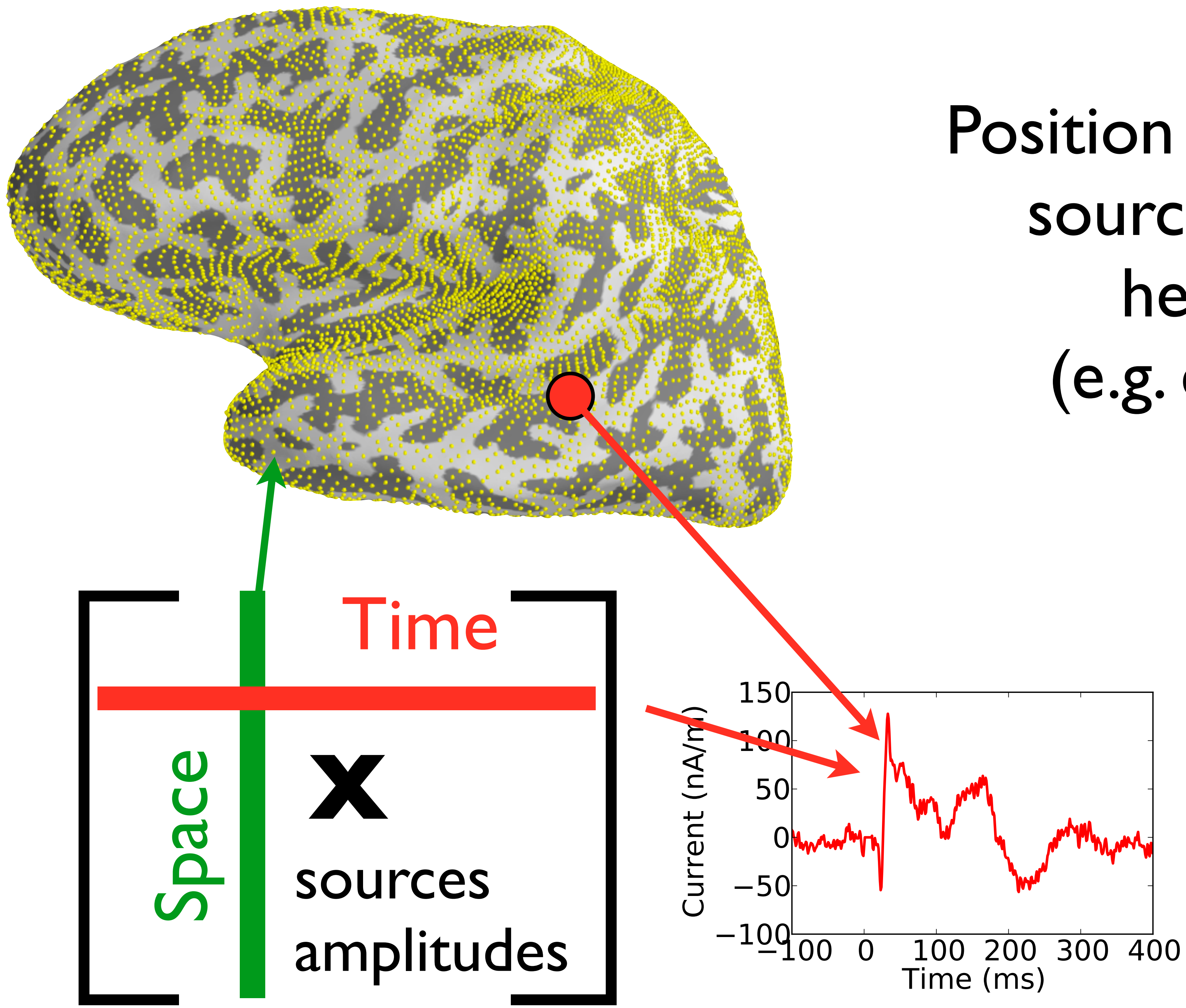


Scalar field defined over time

$$\mathbf{X} \in \mathbb{R}^{\mathbf{P}\times\mathbf{T}}$$

[Dale and Sereno 93]

The source model

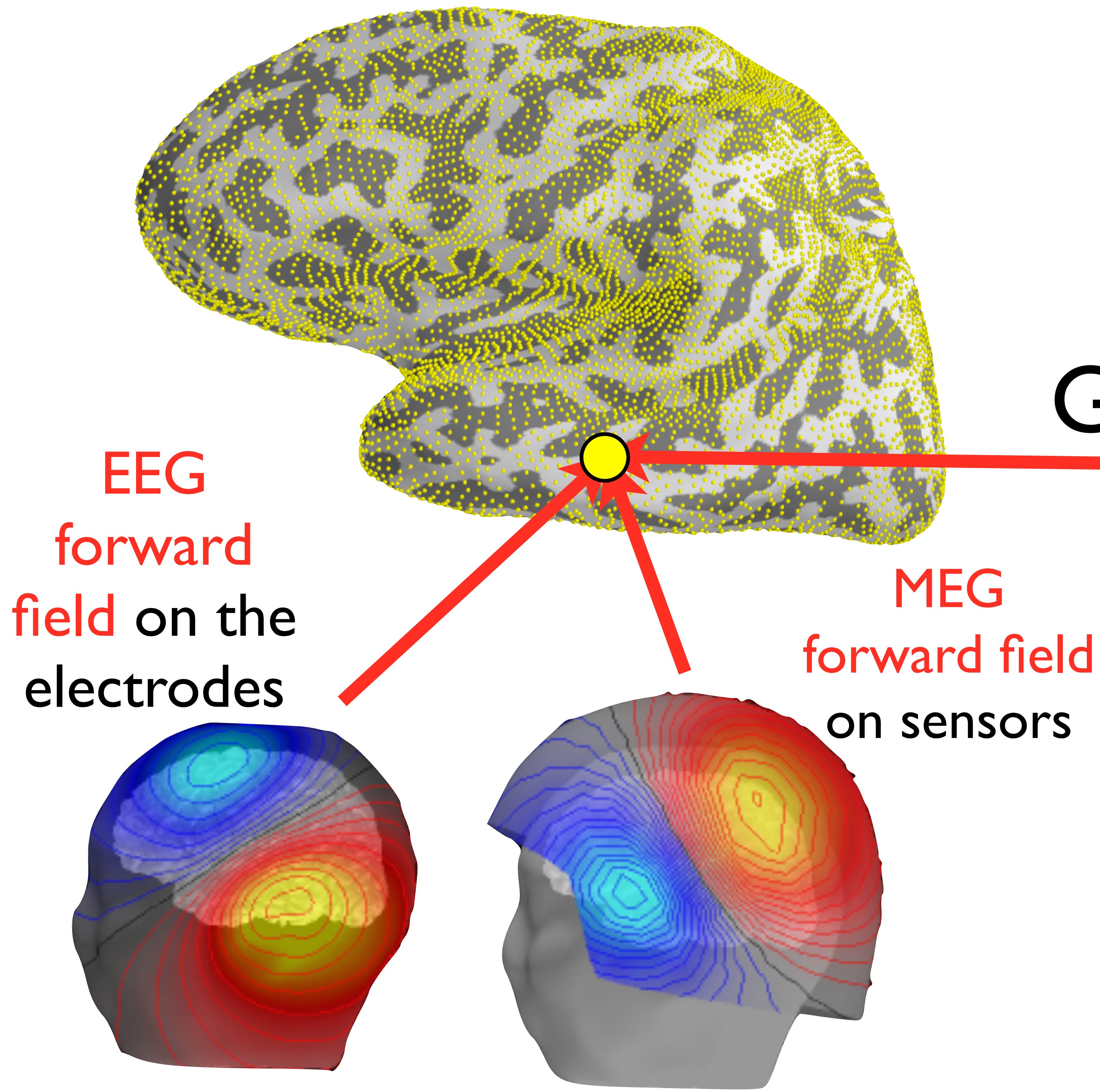


Position 5000 candidate sources over each hemisphere (e.g. every 5mm)

Scalar field defined over time

[Dale and Sereno 93]

The source model



one column = Forward field of one dipole

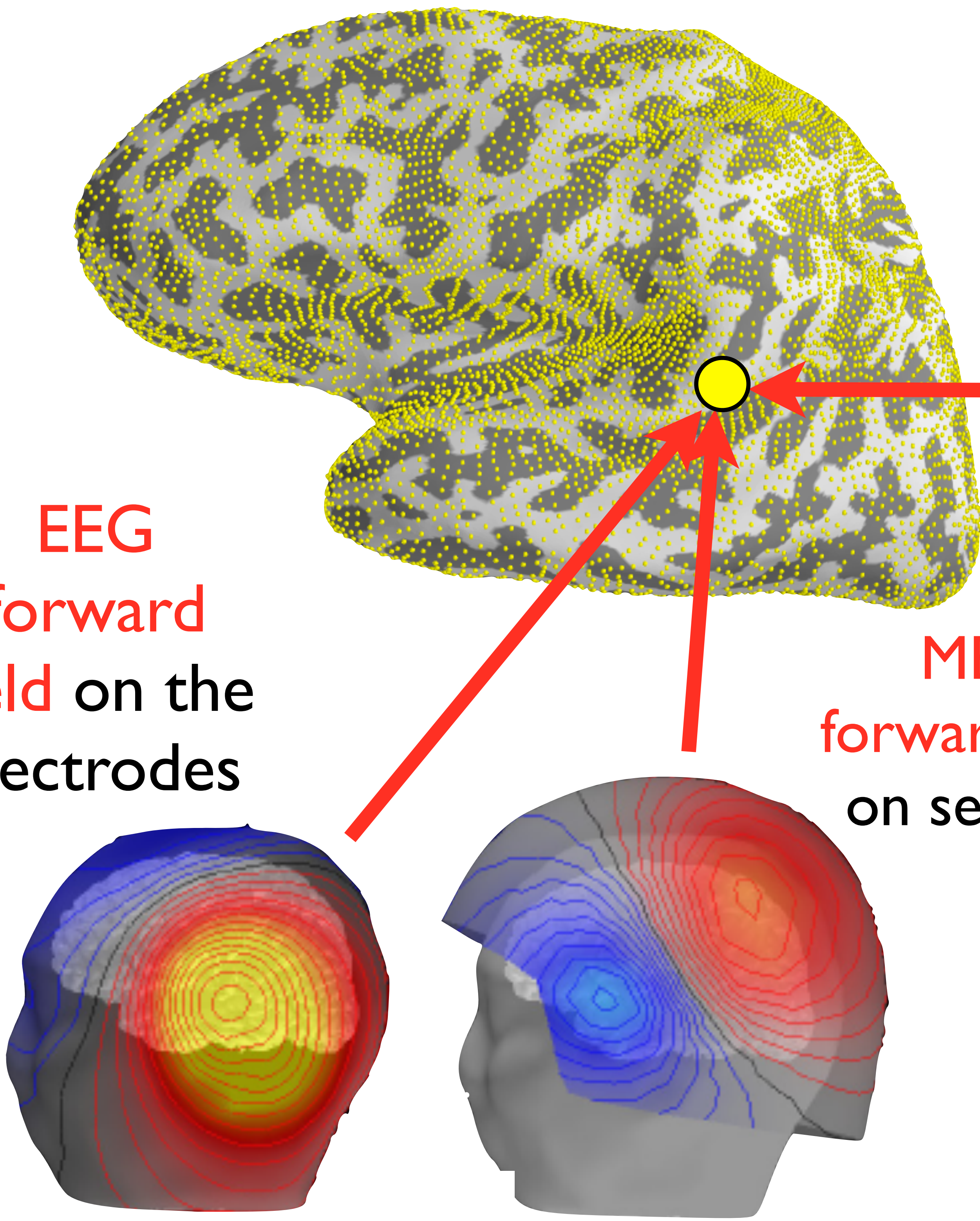
$$G = \left[\begin{array}{c} G_{EEG} \\ \hdashline G_{MEG} \end{array} \right] \in \mathbb{R}^{\mathbf{N} \times \mathbf{P}}$$

G is the gain matrix / forward operator obtained by concatenation of the forward fields

Distributed model

one column = Forward field of one dipole

$$G = \begin{bmatrix} G_{EEG} \\ \hline G_{MEG} \end{bmatrix}$$

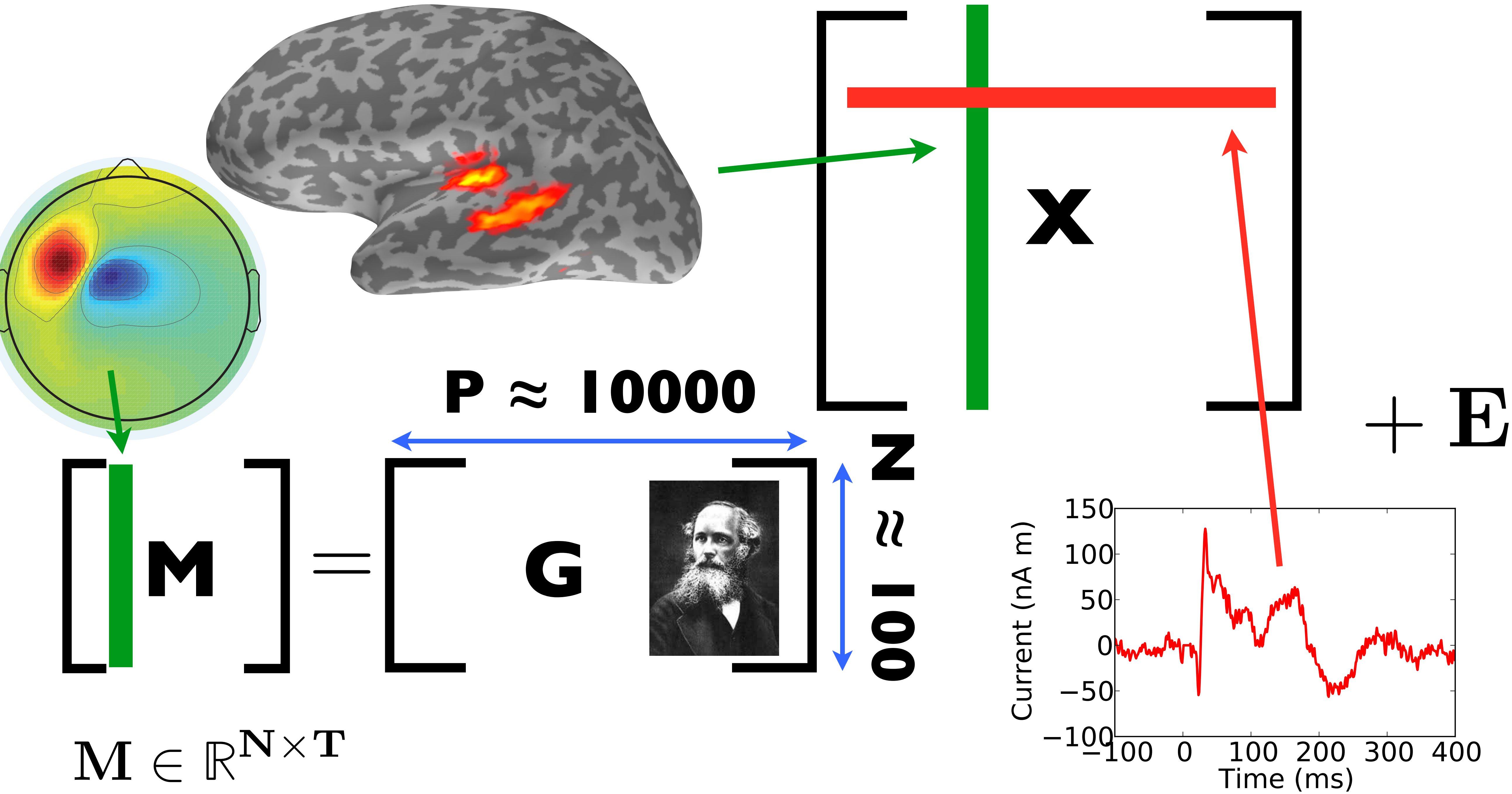


EEG forward field on the electrodes

MEG forward field on sensors

G is the gain matrix obtained by concatenation of the forward fields

Inverse problem: M = GX+E



Objective: Estimate X given M and G

Inverse problem framework

Penalized (variational) formulation (with whitened data):

$$\mathbf{X}^* = \arg\min_{\mathbf{X}} \|\mathbf{M} - \mathbf{G}\mathbf{X}\|_F^2 + \lambda\phi(\mathbf{X}), \lambda > 0$$

Data fit Regularization

λ : Trade-off between the data fit and the regularization

where $\|A\|_F^2 = \sum_{ij} A_{ij}^2$

Inverse problem framework

Penalized (variational) formulation (with whitened data):

$$\mathbf{X}^* = \arg\min_{\mathbf{X}} \|\mathbf{M} - \mathbf{G}\mathbf{X}\|_F^2 + \lambda\phi(\mathbf{X}), \lambda > 0$$

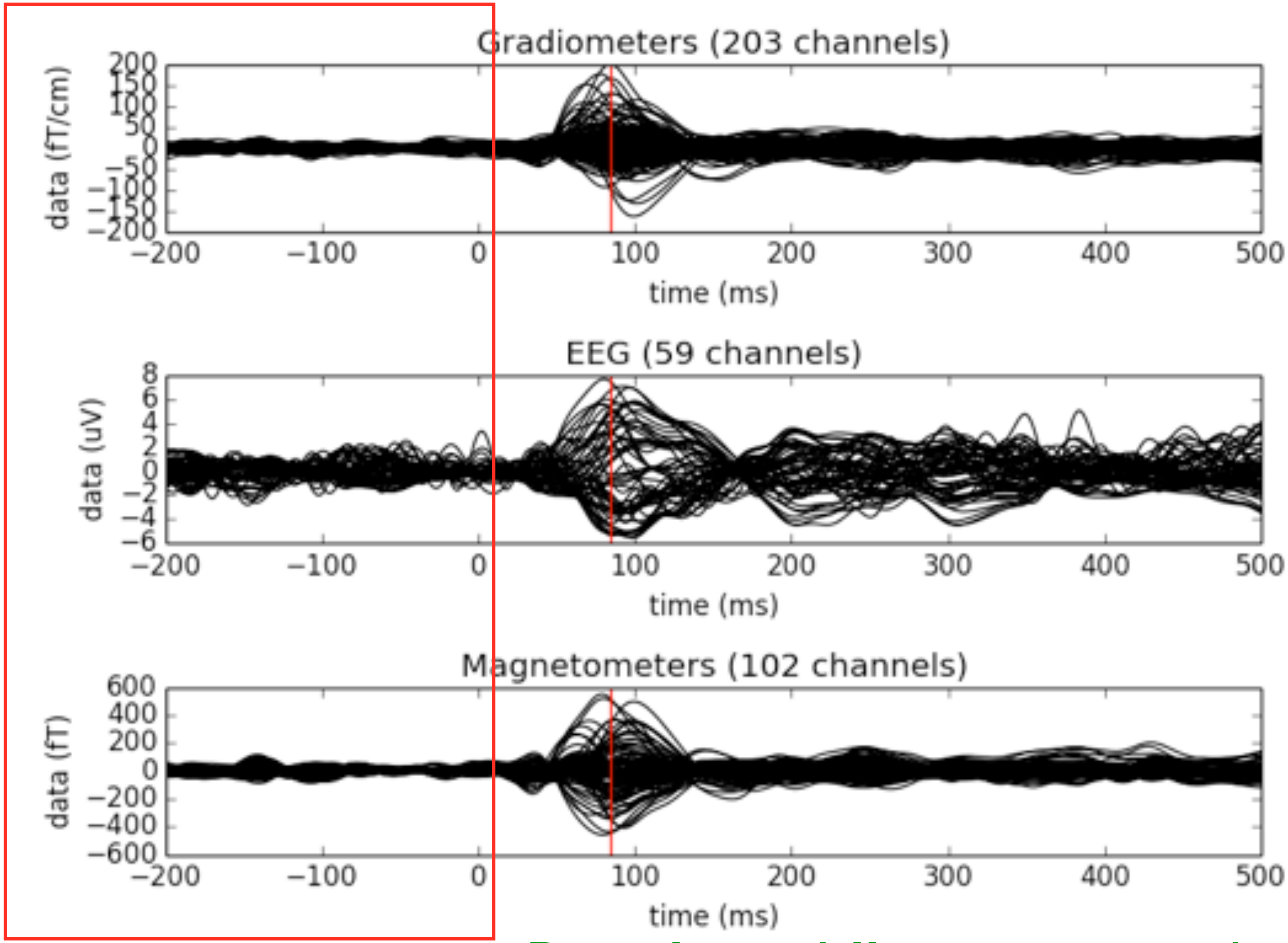
Data fit   Regularization

λ : Trade-off between the data fit and the regularization

where $\|A\|_F^2 = \sum_{ij} A_{ij}^2$

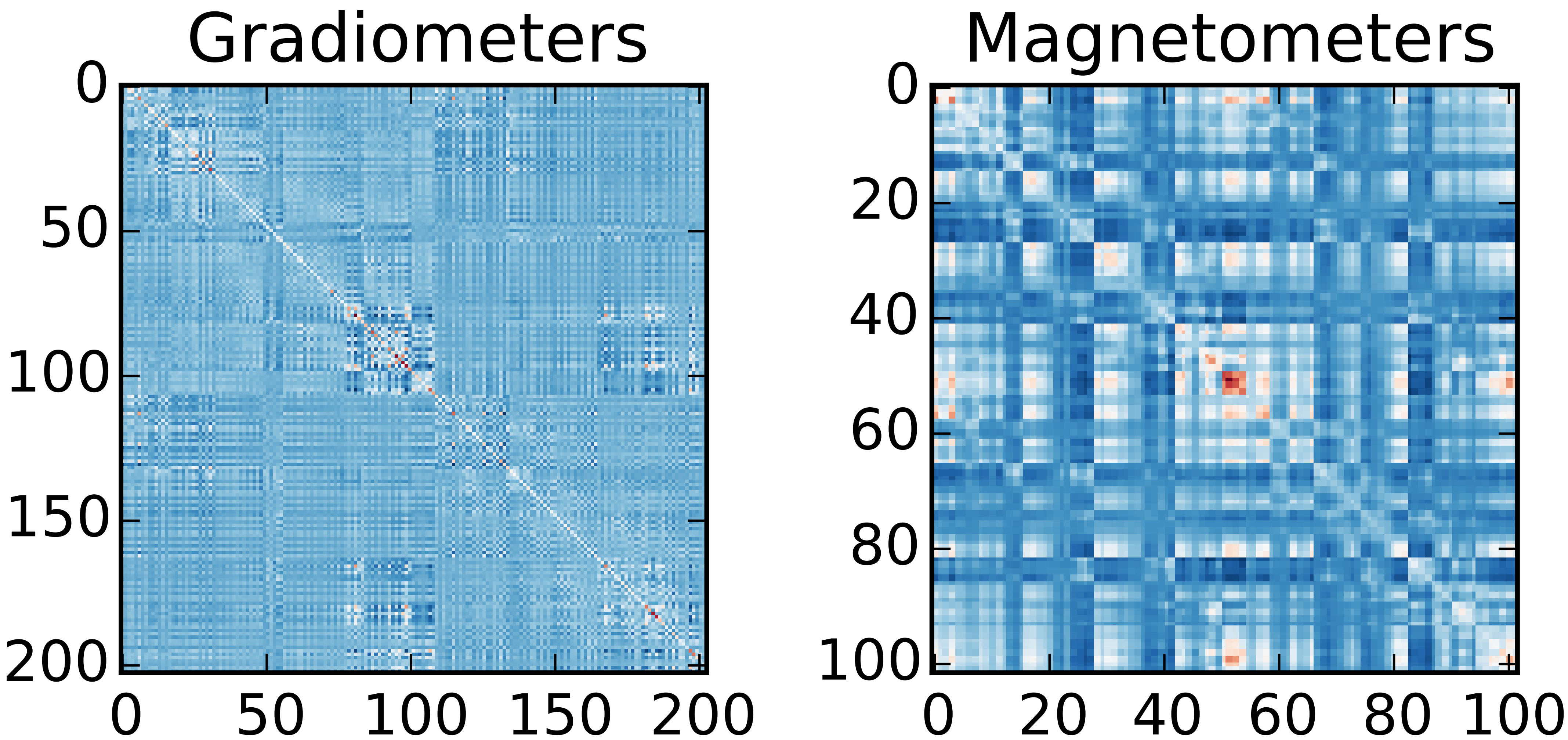
How do you whiten / normalize / scale the data?

To know what is signal look at the noise



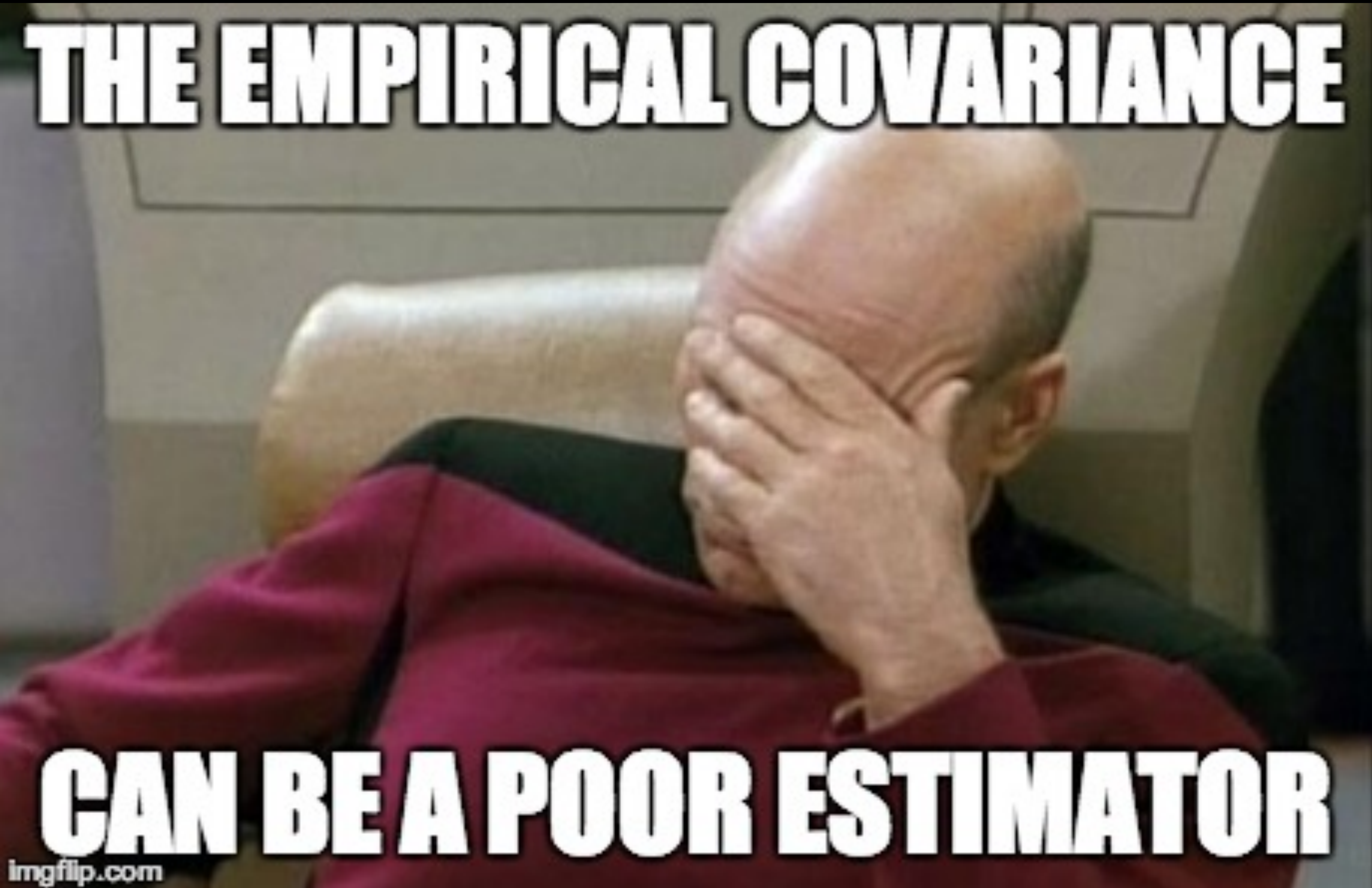
Data from different sensors have to be "normalized" to be on the same scale.

To know what is signal look at the noise



$$C = \frac{MM^T}{d_t}$$

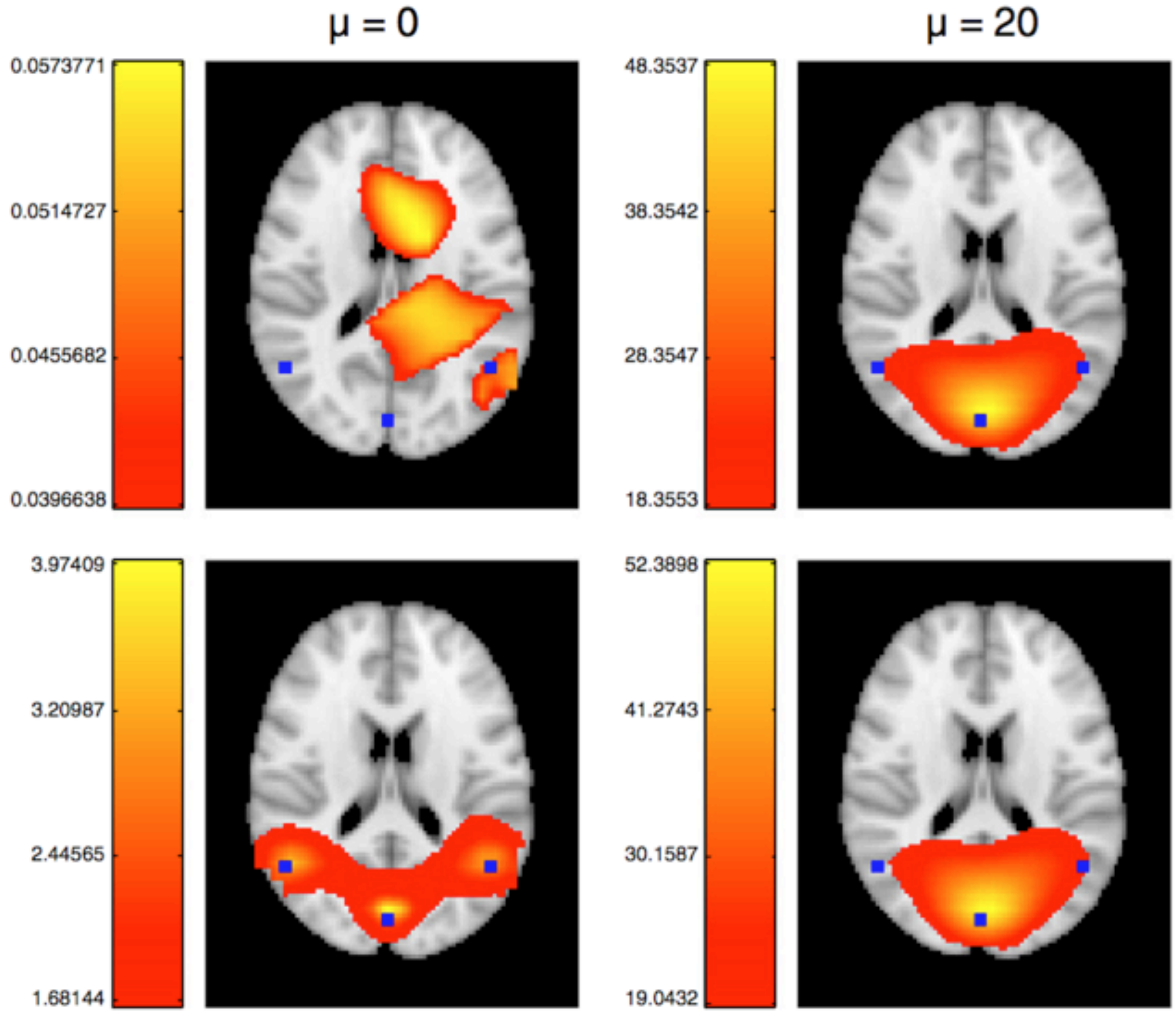
With whitened data the covariance would be diagonal


THE EMPIRICAL COVARIANCE
CAN BE A POOR ESTIMATOR
imgflip.com

True sources VS LCMV Beamformer given C

10s

80s



$$C_\mu = \mu\sigma_e^2 I + \frac{1}{T} M M^t$$

[Woolrich, 2011, Neuroimage]

Many strategies exist:

simple, fast

1. Hand-set (REG)

$$C' = C + \alpha I, \quad \alpha > 0$$

2. Ledoit-Wolf (LW)

$$C_{LW} = (1 - \alpha)C + \alpha \mu I \qquad \mu = \mathrm{mean}(\mathrm{diag}(C))$$

3. Cross-validated shrinkage (SC)

$$C_{SC} = (1 - \alpha_{CV})C + \alpha_{CV} \mu I$$

4. Probabilistic PCA (PPCA)

$$C_{PPCA} = HH^t + \sigma^2 I_N$$

5. Factor Analysis (FA)

$$C_{FA} = HH^t + \mathrm{diag}(\psi_1, \dots, \psi_D)$$

complex, slow

[Engemann, D. and Gramfort, A., NeuroImage 2015]

Model selection: Log-likelihood

Given my model C how likely are unseen data Y?

$$\mathcal{L}(Y|C) = -\frac{1}{2T}\mathrm{Trace}(YY^tC^{-1}) - \frac{1}{2}\log((2\pi)^N\det(C))$$

Higher log likelihood = better C & better whitening

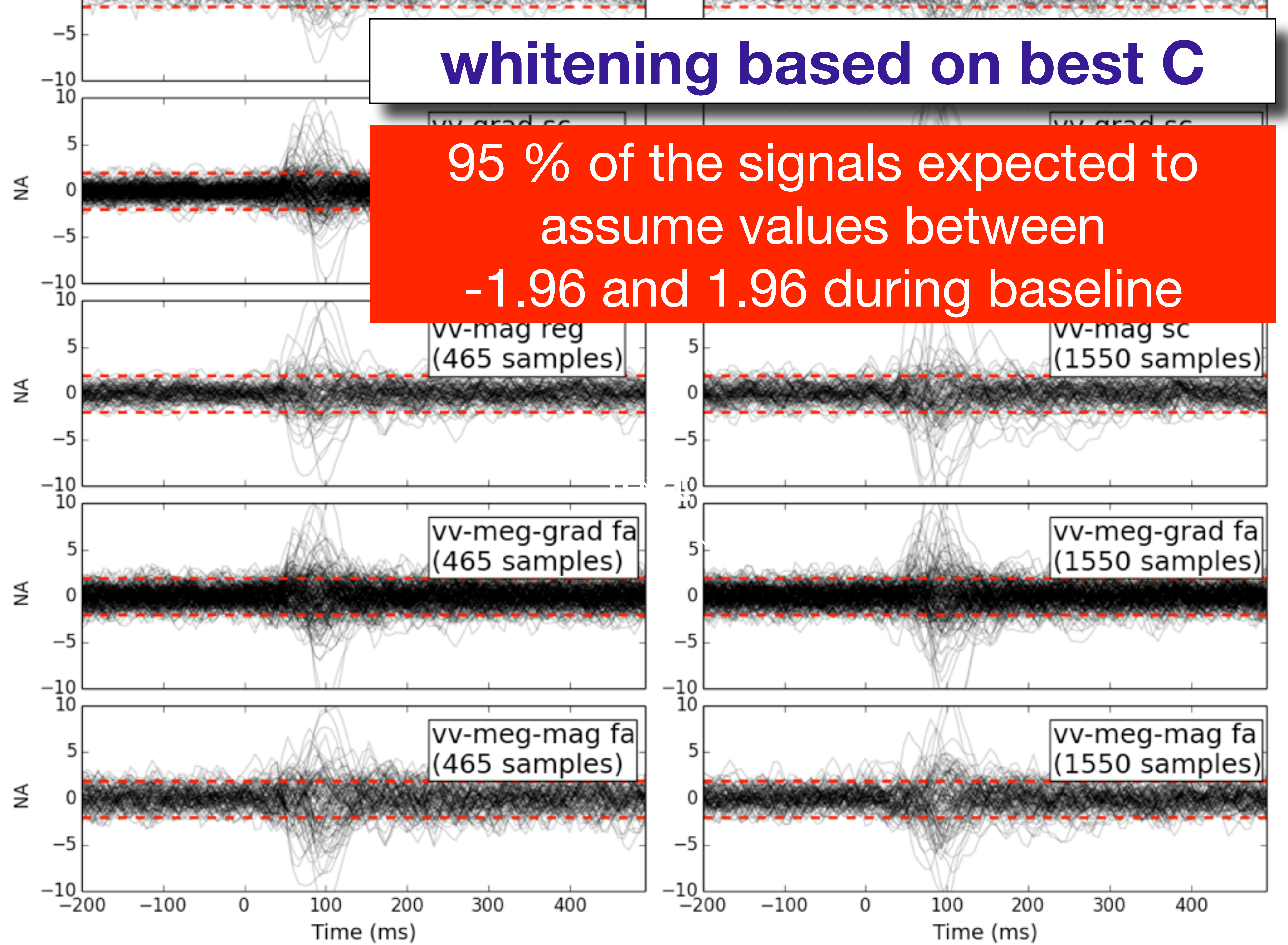
[Engemann, D. and Gramfort, A., NeuroImage 2015]

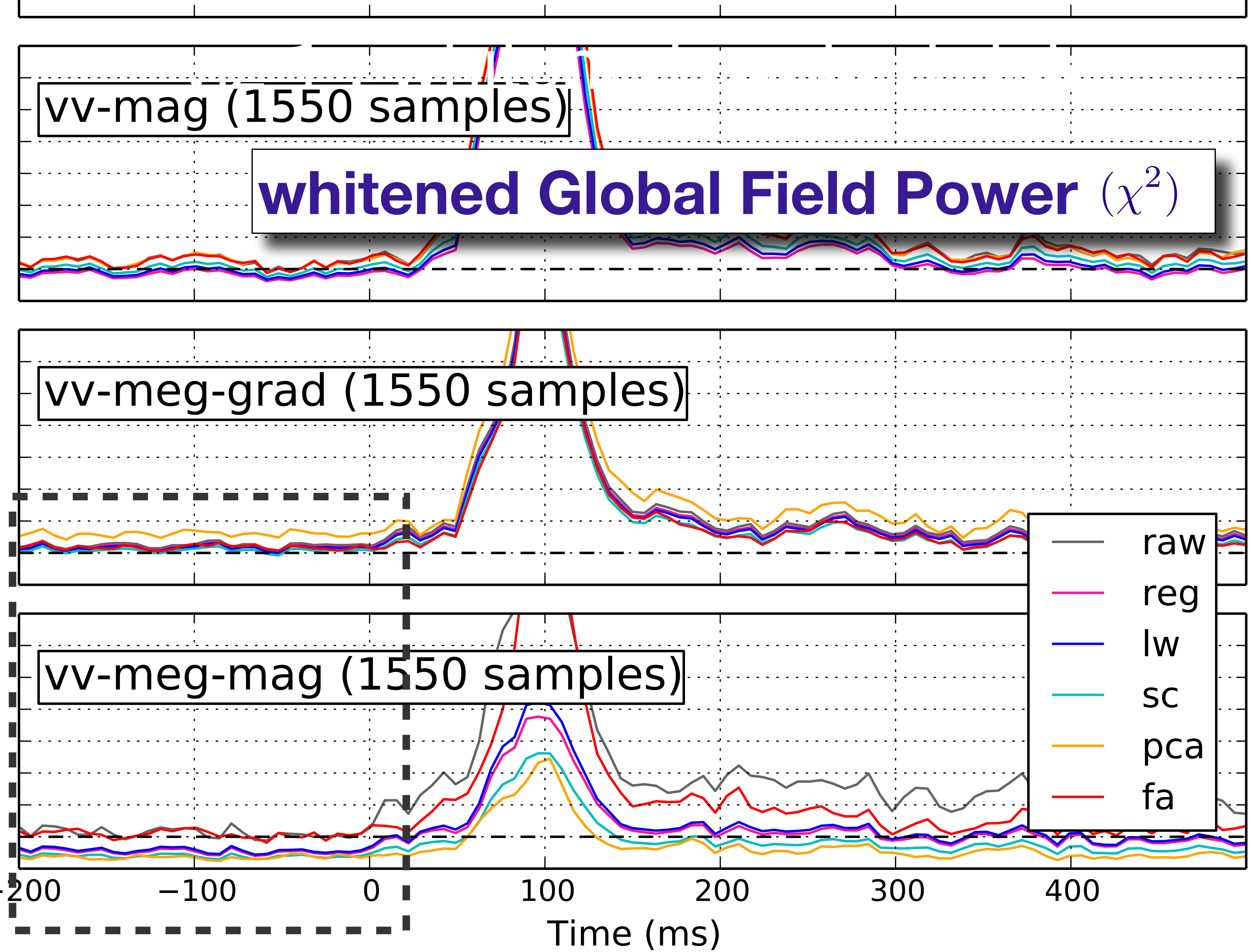
MEG and EEG data

key	**dataset and channel type**
bti-mag	4D Magnes 3600 WH magnetometers
ctf-mag	CTF-275 axial gradiometers
vv-eeg	VectorView EEG electrodes
vv-grad	VectorView planar gradiometers
vv-mag	VectorView magnetometers
vv-meg-grad	VectorView planar gradiometers, combined estimation
vv-meg-mag	VectorView magnetometers, combined estimation

whitening based on best C

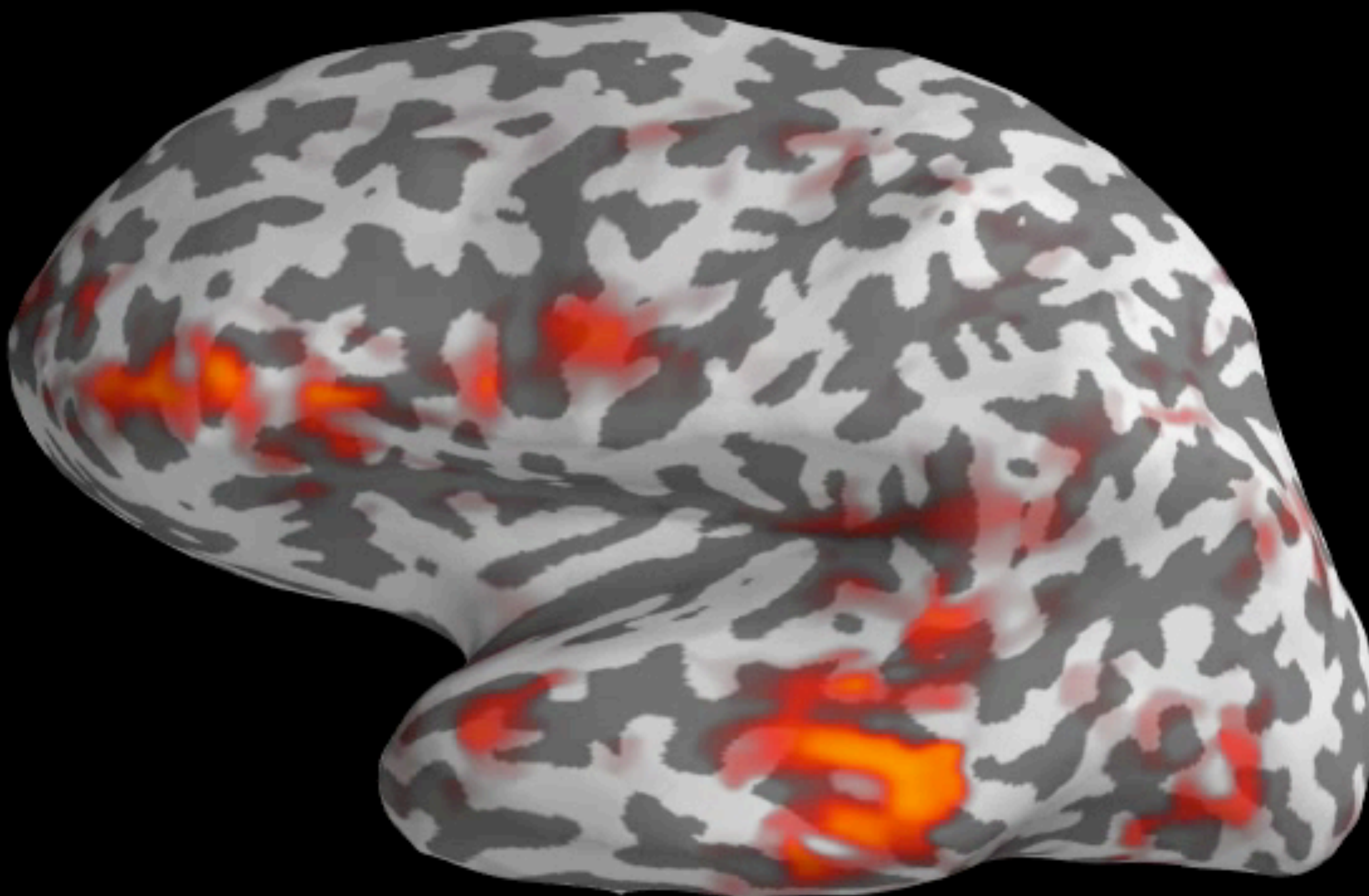
95 % of the signals expected to assume values between -1.96 and 1.96 during baseline





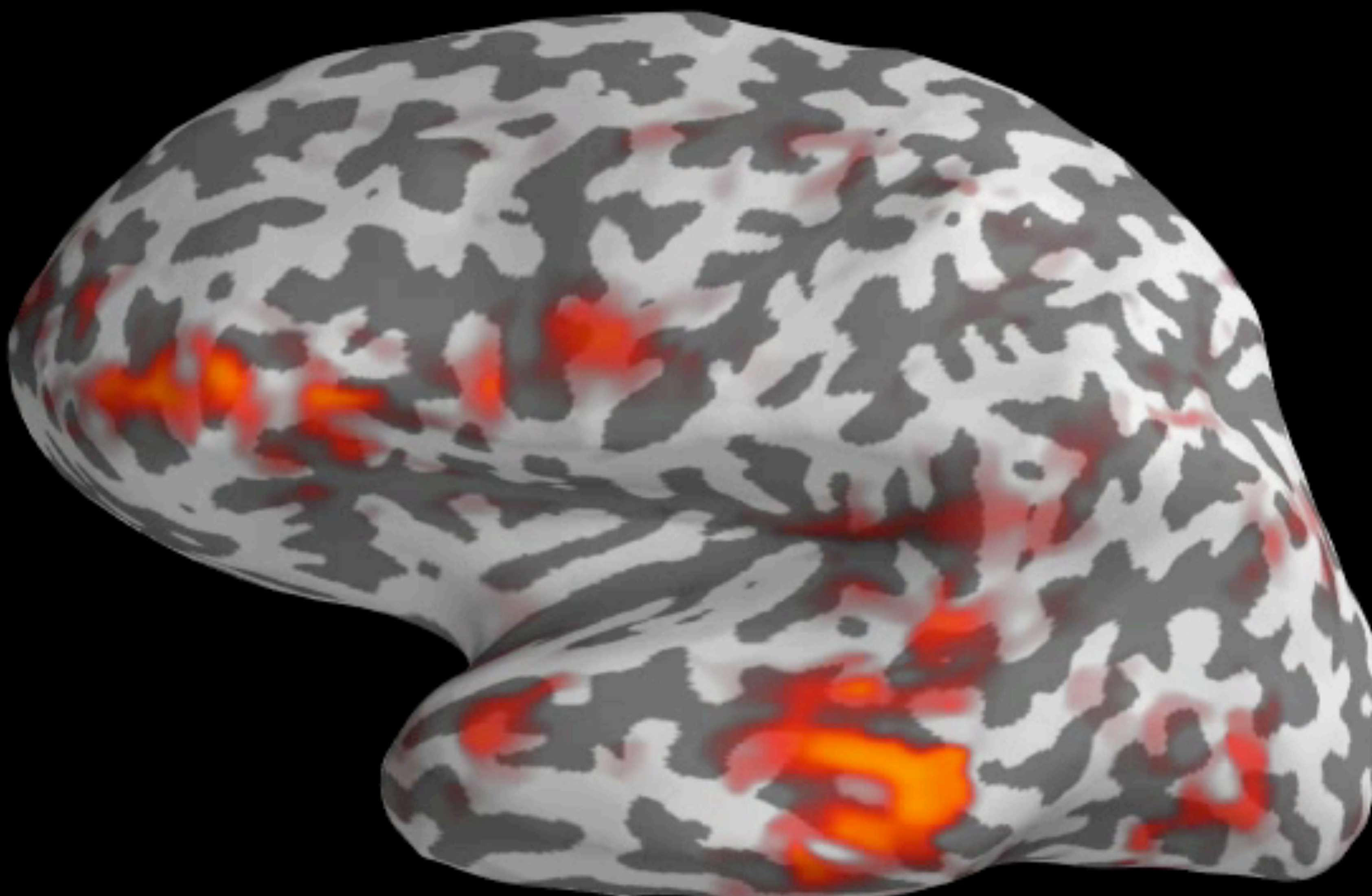
vv-mag (1550 samples)
whitened Global Field Power (χ^2)
vv-meg-grad (1550 samples)
vv-meg-mag (1550 samples)
raw
reg
lw
sc
pca
fa
−200
−100
0
100
200
300
400
Time (ms)

http://youtu.be/Uxr5Pz7JPrs



time=0.00 ms

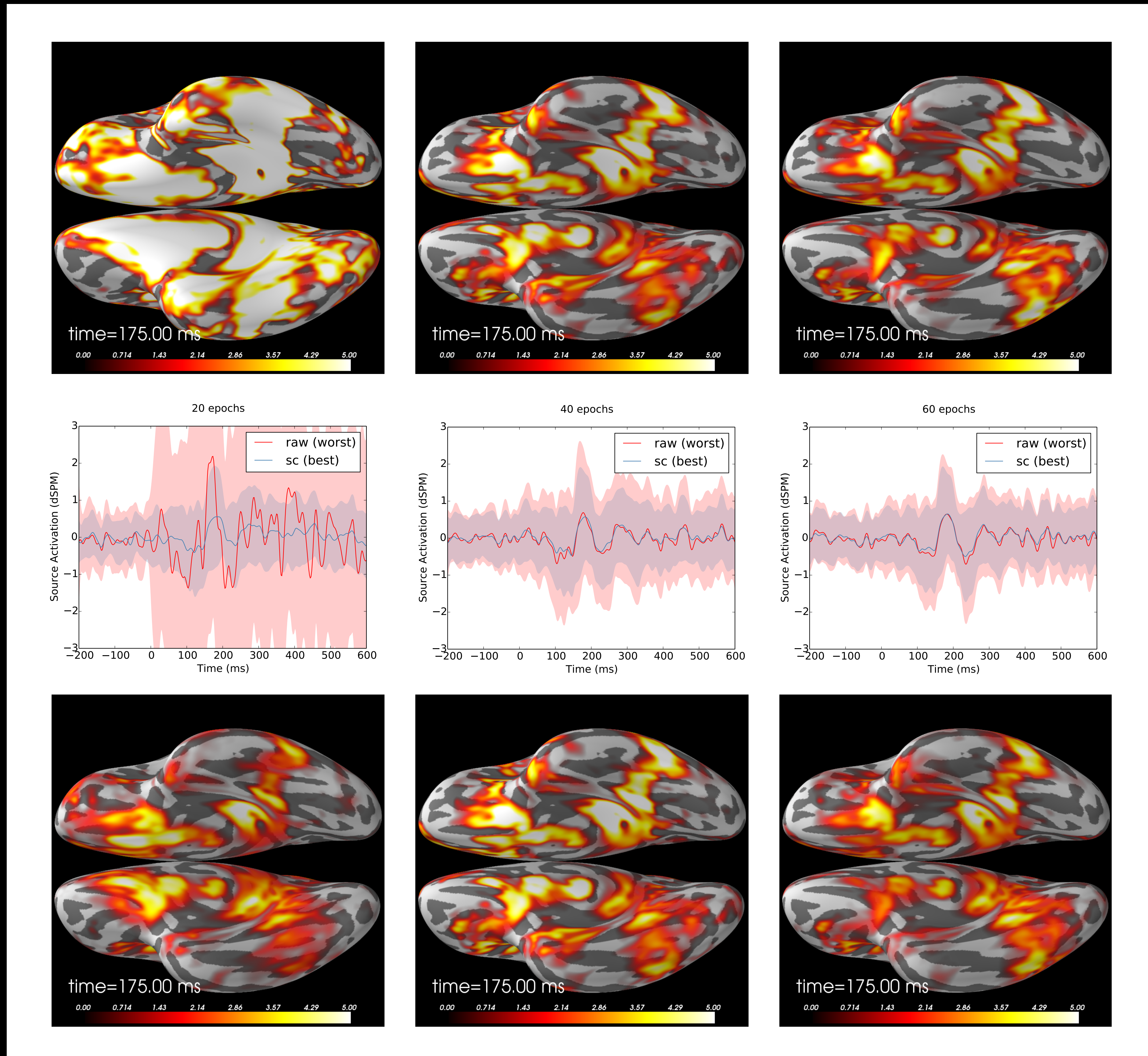
http://youtu.be/Uxr5Pz7JPrs



time=0.00 ms

20 epochs 40 epochs 60 epochs

worst

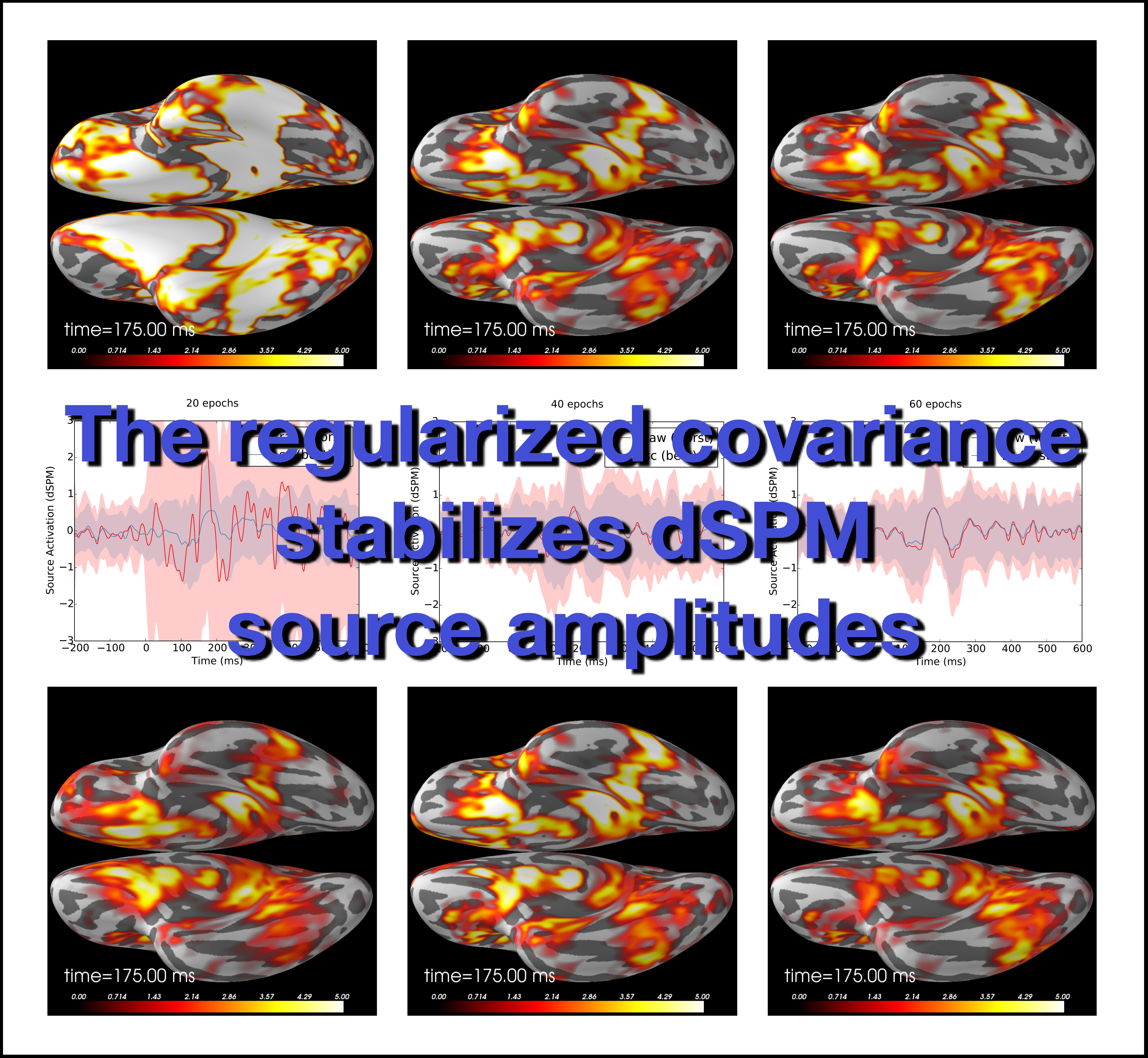


std

best

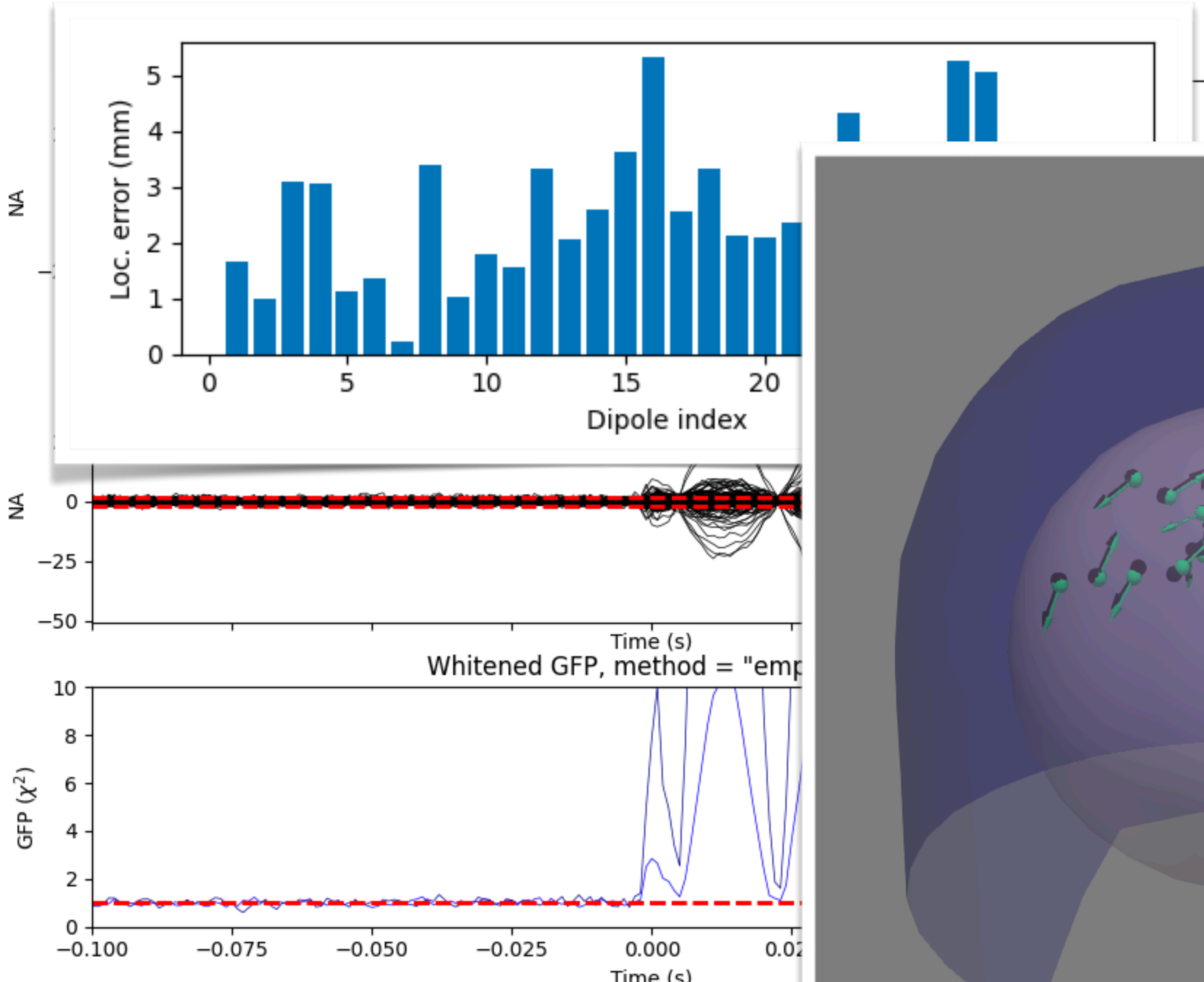
faces > scrambled SPM faces dataset, Henson (2003)

The regularized covariance stabilizes dSPM source amplitudes

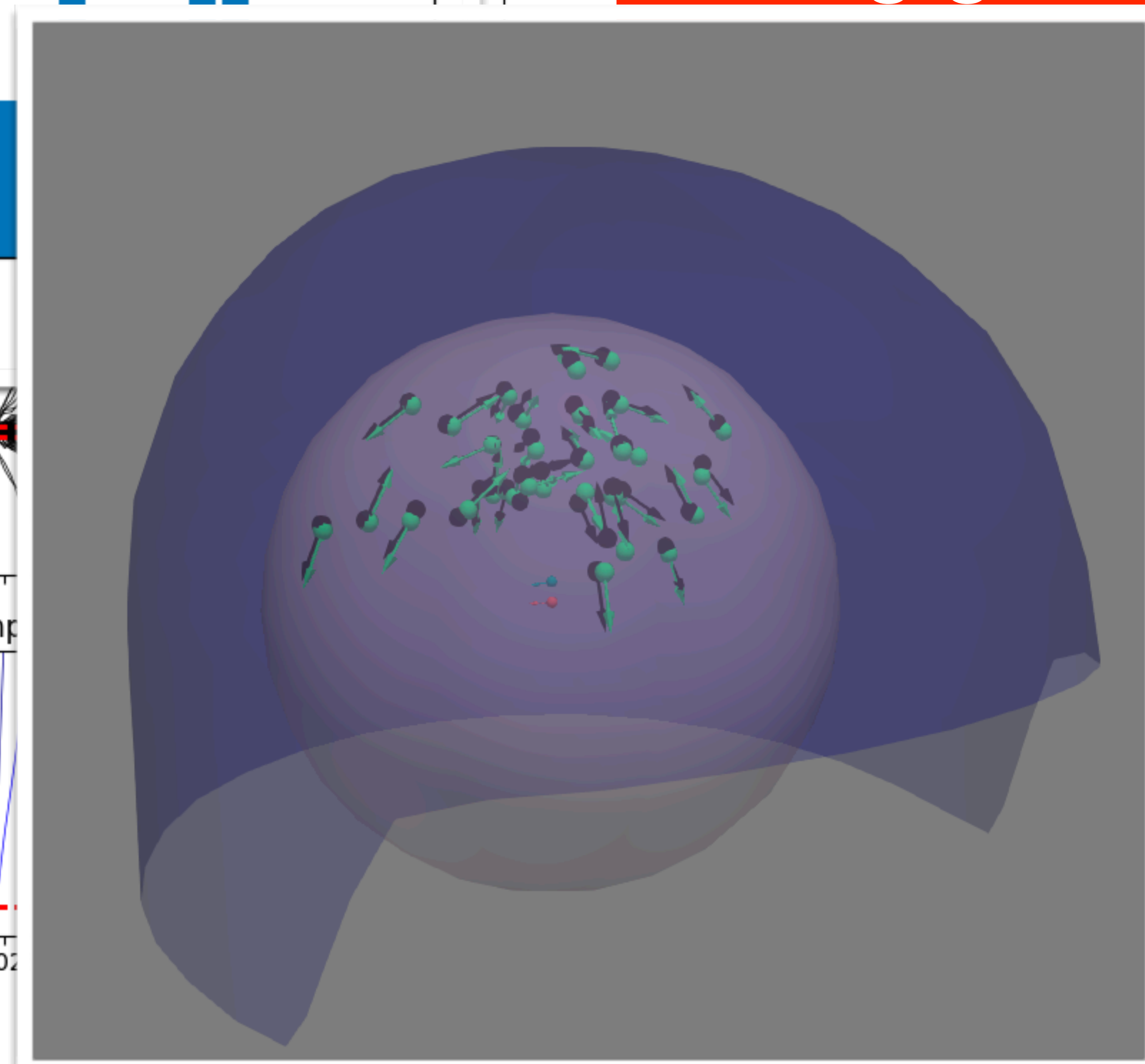


https://bit.ly/ohbm2020_mne_cov

Dipole fitting and noise covariance

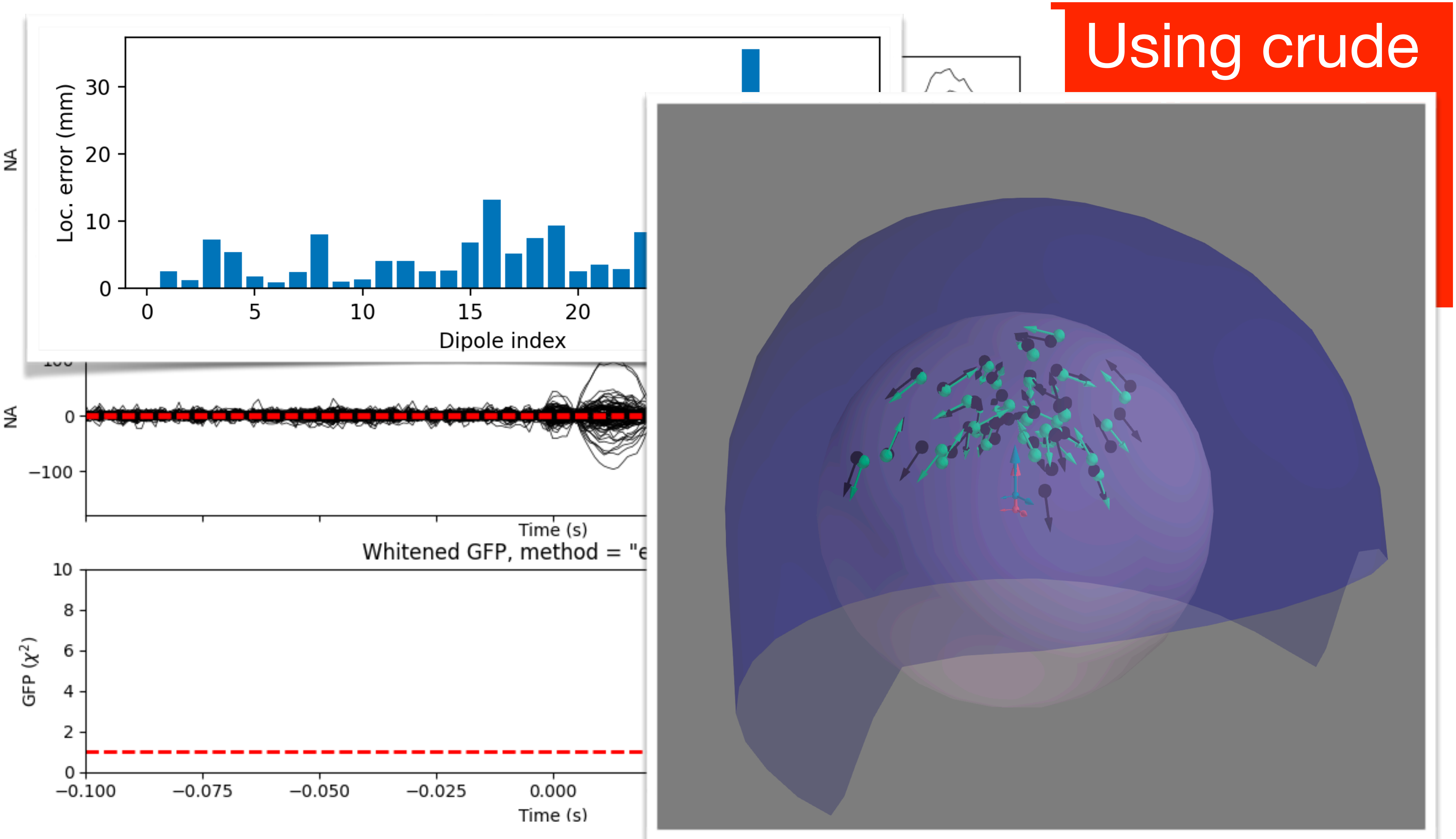


Using good



https://mne.tools/stable/auto_tutorials/sample-datasets/plot_brainstorm_phantom_elekta.html

Dipole fitting and noise covariance



https://mne.tools/stable/auto_tutorials/sample-datasets/plot_brainstorm_phantom_elekta.html

Conclusion

Conclusion

- “Garbage in garbage out”
- To make the best of EEG/MEG data **do not** process channel independently!
- Check statistical hypothesis and use quality control plots

frontiers in Neuroscience

METHODS ARTICLE

Front. Neurosci., 06 August 2018 | https://doi.org/10.3389/fnins.2018.00530

Check for updates

A Reproducible MEG/EEG Group Study With the MNE Software: Recommendations, Quality Assessments, and Good Practices

Mainak Jas[1], Eric Larson[2], Denis A. Engemann[3,4], Jaakko Leppäkangas[1], Samu Taulu[2,5], Matti Hämäläinen[6] and Alexandre Gramfort[1,3,4*]

[1]Telecom ParisTech, Université Paris-Saclay, Paris, France

[2]Institute for Learning and Brain Sciences, University of Washington, Seattle, WA, United States

[3]NeuroSpin, CEA, Université Paris-Saclay, Gif-sur-Yvette, France

[4]INRIA, Université Paris-Saclay, Saclay, France

[5]Department of Physics, University of Washington, Seattle, WA, United States

[6]Athinoula A. Martinos Center for Biomedical Imaging, Massachusetts General Hospital, Harvard Medical School, Charlestown, MA, United States

https://www.frontiersin.org/articles/10.3389/fnins.2018.00530/full

NeuroImage
Volume 108, March 2015, Pages 328-342

Automated model selection in covariance estimation and spatial whitening of MEG and EEG signals

Denis A. Engemann [a, b, c, d], Alexandre Gramfort [a, e]

Show more

https://doi.org/10.1016/j.neuroimage.2014.12.040

https://www.sciencedirect.com/science/article/abs/pii/S1053811914010325?via%3Dihub

MNE Tutorials Gallery Manual API FAQ Cite Site Page

Search

MNE

MEG + EEG ANALYSIS & VISUALIZATION

http://www.martinos.org/mne

MNE is a community-driven software package designed for for **processing electroencephalography (EEG) and magnetoencephalography (MEG) data** providing comprehensive tools and workflows for:

1. Preprocessing
2. Source estimation
3. Time-frequency analysis
4. Statistical testing
5. Estimation of functional connectivity
6. Applying machine learning algorithms
7. Visualization of sensor- and source-space data

MNE includes a comprehensive Python package (provided under the simplified BSD license), supplemented by tools compiled from C code for the LINUX and Mac OSX operating systems, as well as a MATLAB toolbox.

Documentation

- Getting Started
- What's new
- Cite MNE
- Related publications
- Tutorials
- Examples Gallery
- Manual
- API Reference
- Frequently Asked Questions
- Advanced installation and setup
- MNE with CPP

MNE software for processing MEG and EEG data, A. Gramfort, M. Luessi, E. Larson, D. Engemann, D. Strohmeier, C. Brodbeck, L. Parkkonen, M. Hämäläinen, Neuroimage 2013

Thanks !

Contact http://alexandre.gramfort.net

GitHub : @agramfort *Twitter : @agramfort*

Support ERC SLAB, ANR THALAMEEG ANR-14-NEUC-0002-01
NIH R01 MH106174, DFG HA 2899/21-1.