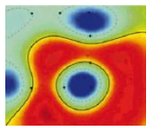
Machine Learning and Signal Processing tools for Brain Computer Interfacing

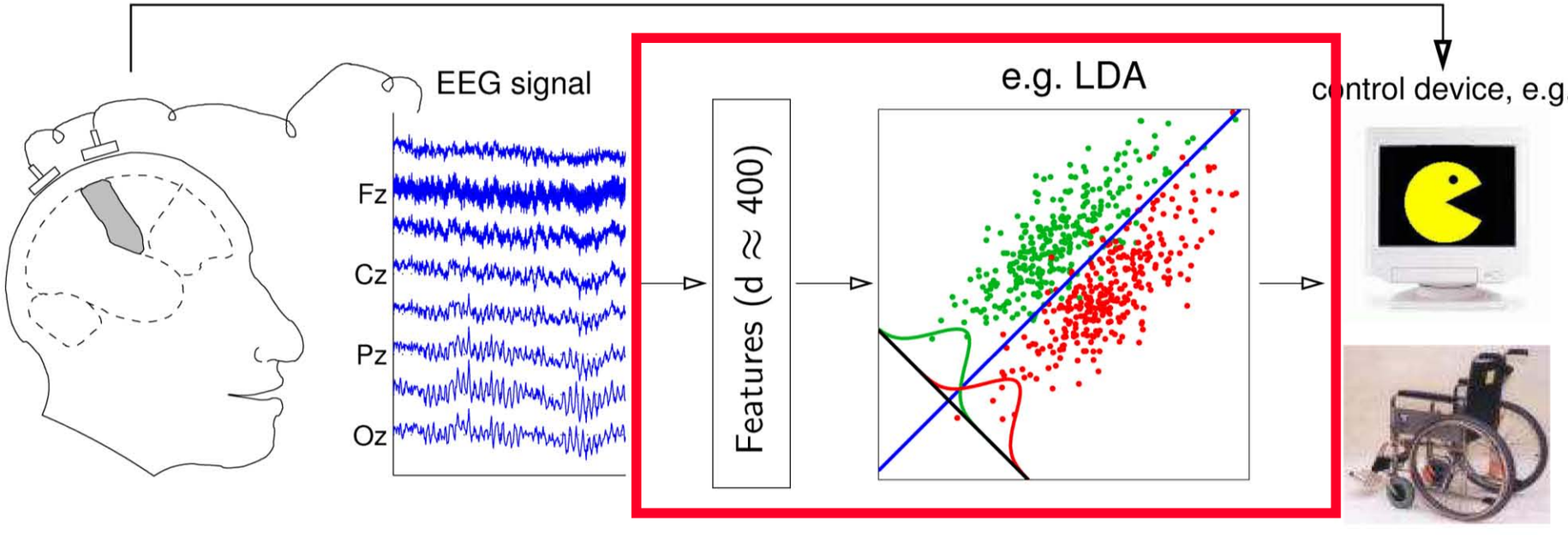
Klaus-Robert Müller, Benjamin Blankertz, Gabriel Curio et al.

Overview

- Lecture 1: Machine Learning Tools for BCI
  - Introduction to BCI
  - Sensory motor rhythms classification
  - Non-stationarity, mixed effects
- Lecture 2
  - P300 BCI, ERP classification
  - Patterns vs Filters
  - Shrinkage
  - Multimodal analysis
  - Pitfalls

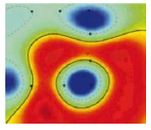


Noninvasive Brain-Computer Interface



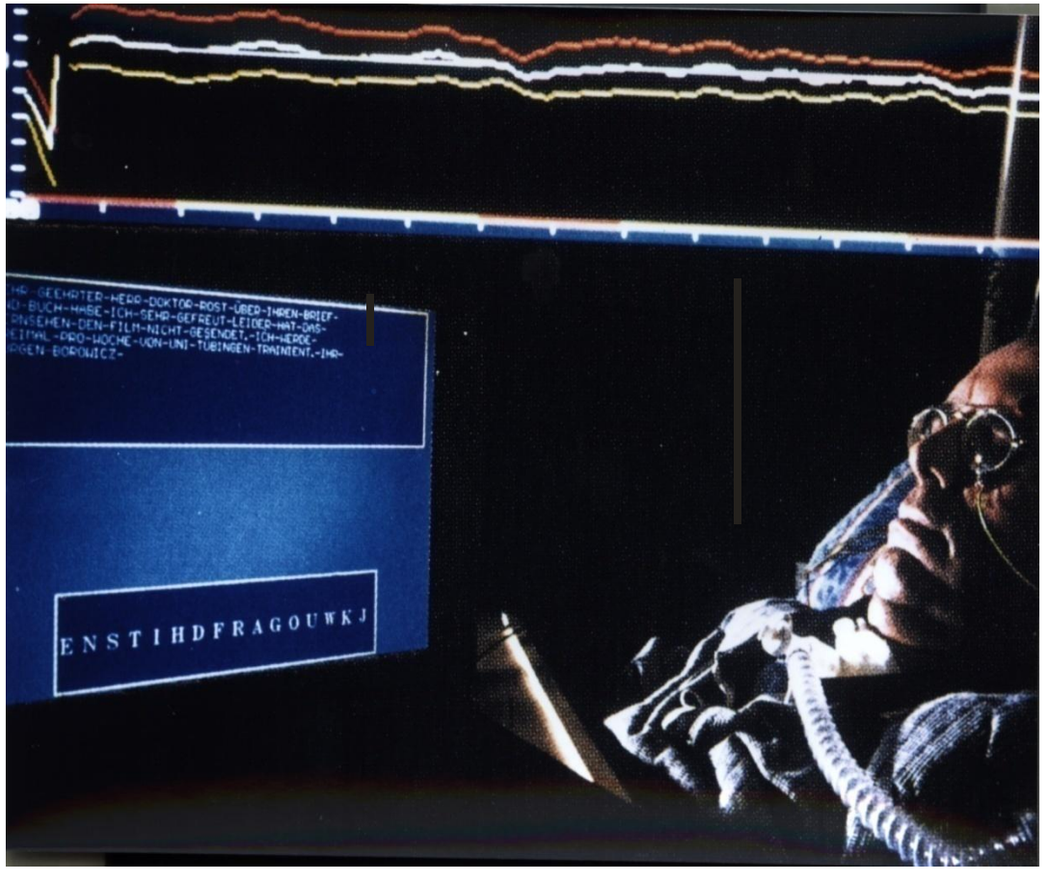
DECODING

BCI: Translation of human intentions into a technical control signal
without using activity of muscles or peripheral nerves

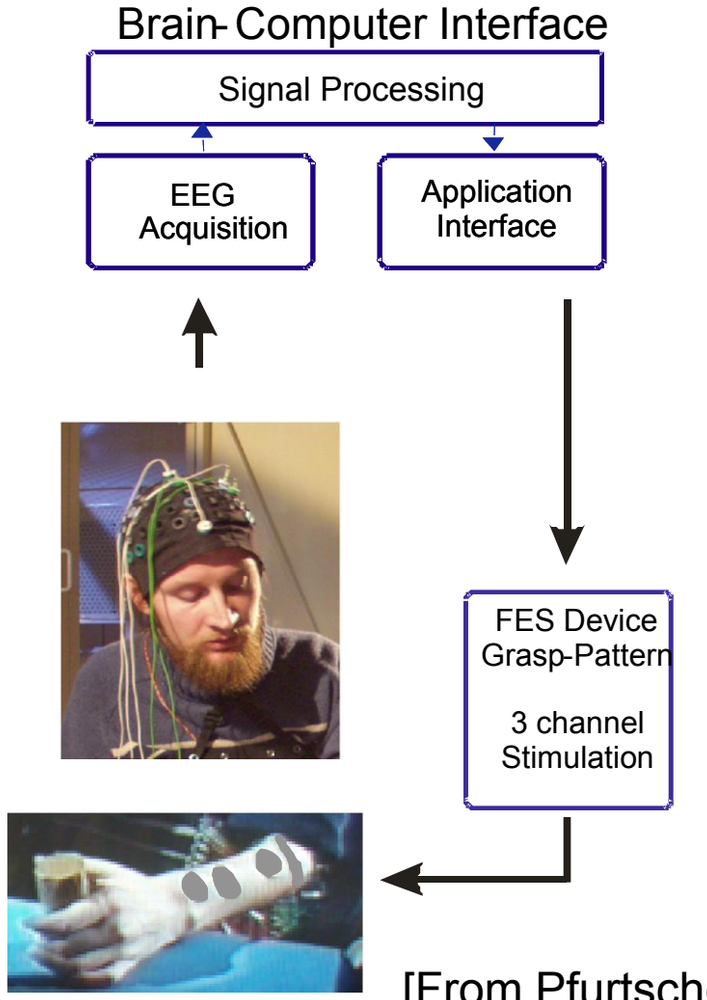


‚Brain Pong‘ with BBCI

Noninvasive BCI: clinical applications



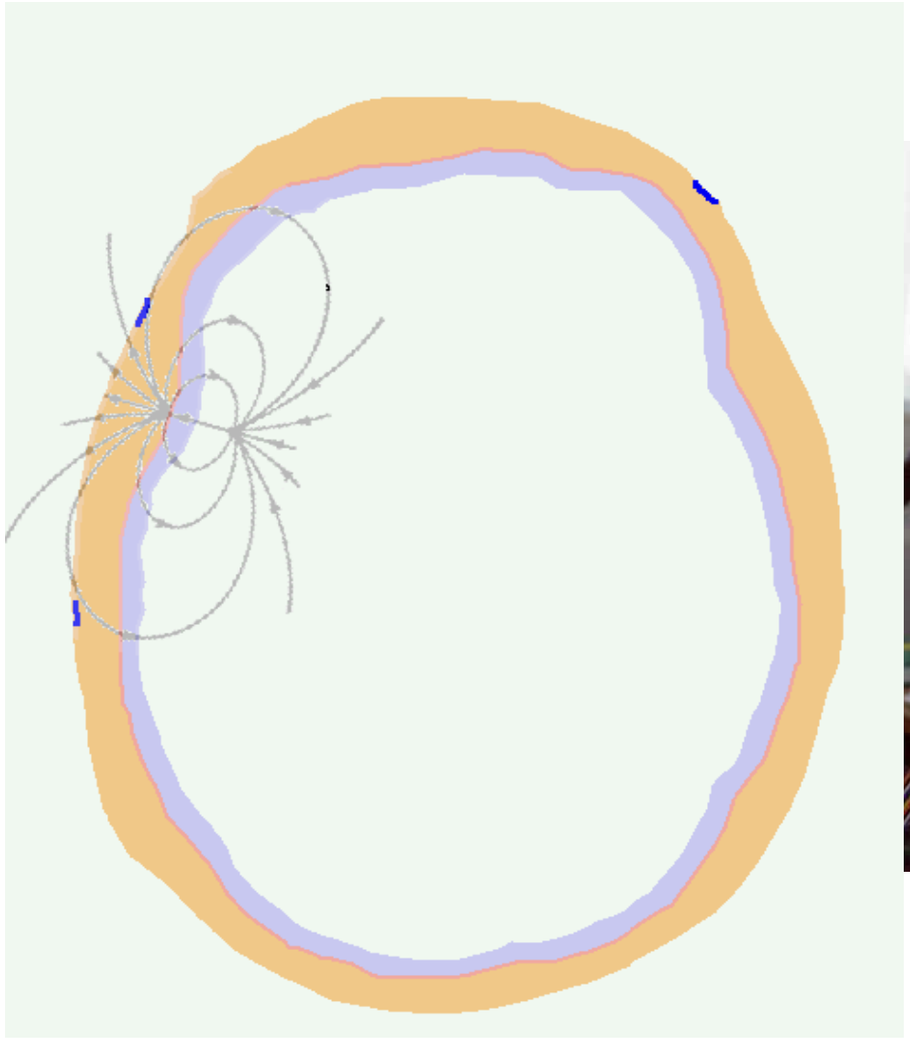
[From Birbaumer et al.]

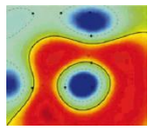


[From Pfurtscheller et al.]

BBCI: Leitmotiv: *›let the machines learn‹*

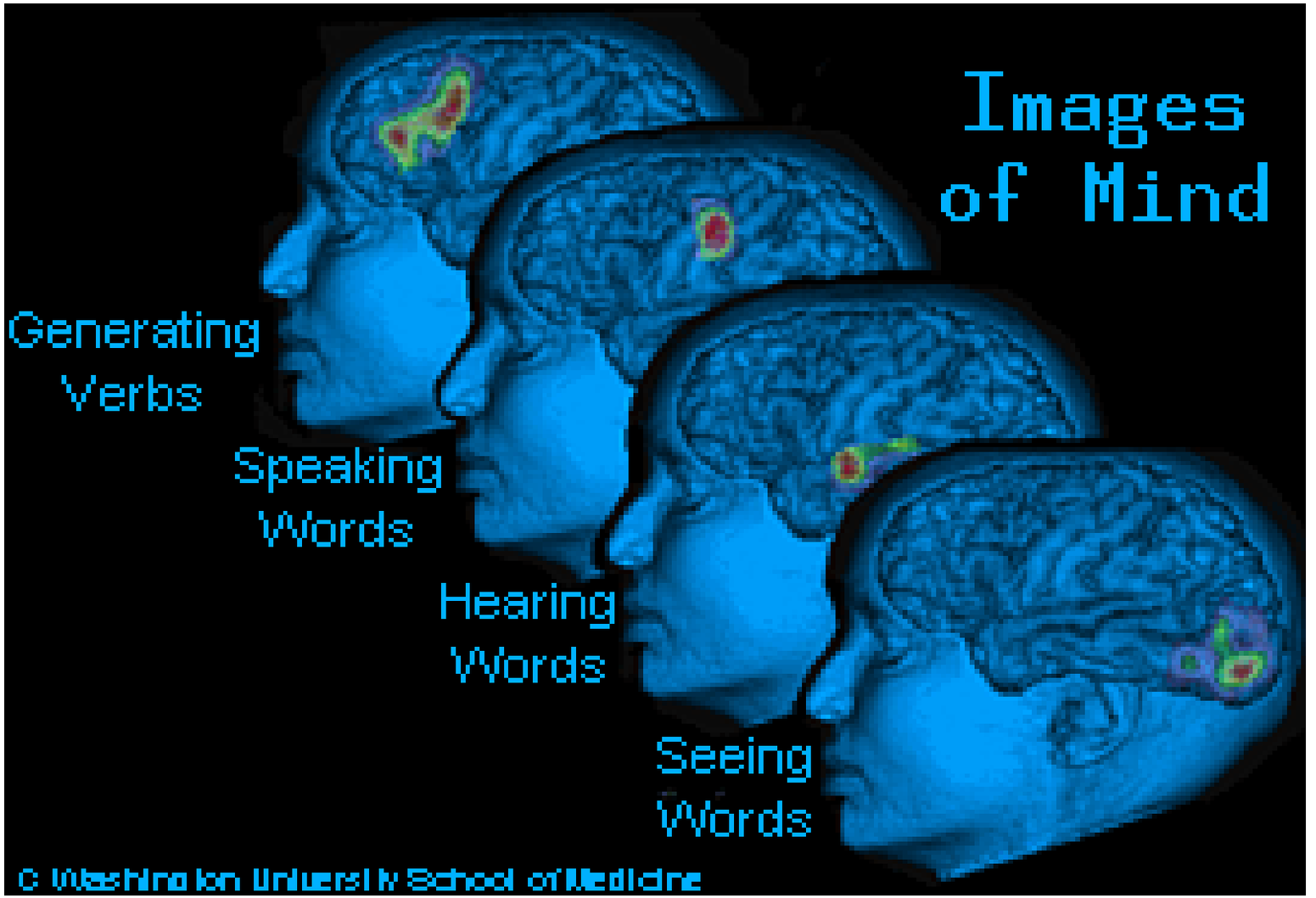
EEG basiertes nichtinvasives BCI

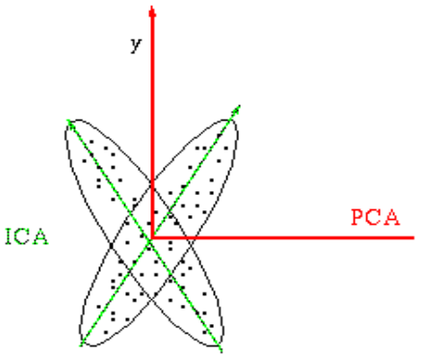




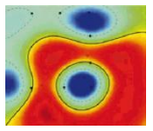
[From Vigario]

The cerebral cocktail party problem





- **use ICA/NGCA projections for artifact and noise removal**
- **feature extraction and selection**

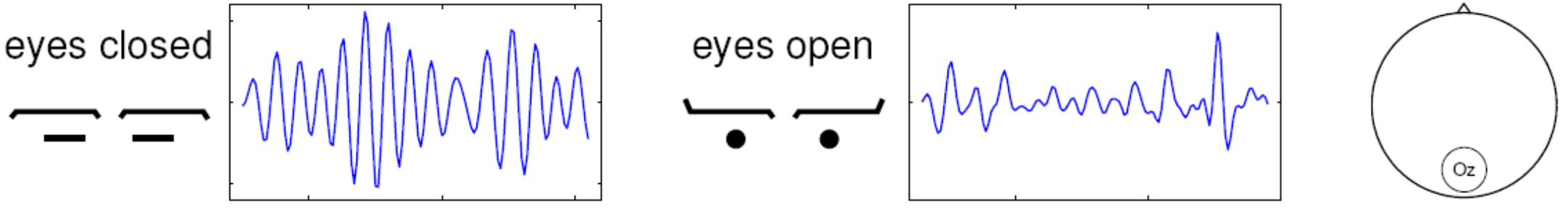


[cf. Ziehe et al. 2000, Blanchard et al. 2006]

Towards imaginations: Modulation of Brain Rhythms

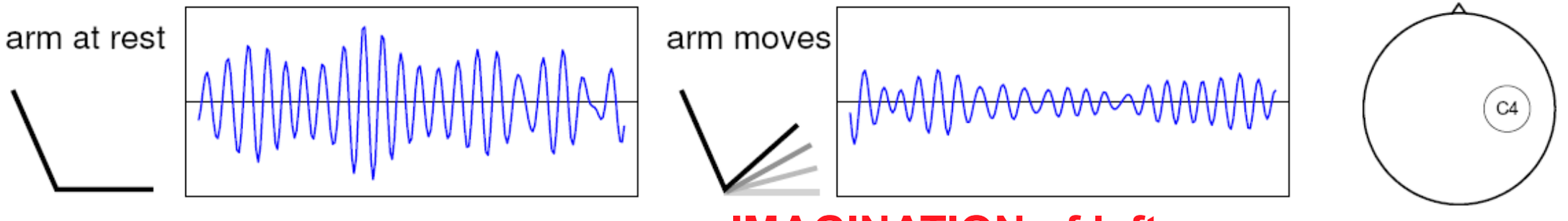
Most rhythms are idle rhythms, i.e., they are **attenuated** during activation.

- α-rhythm (around 10 Hz) in visual cortex:



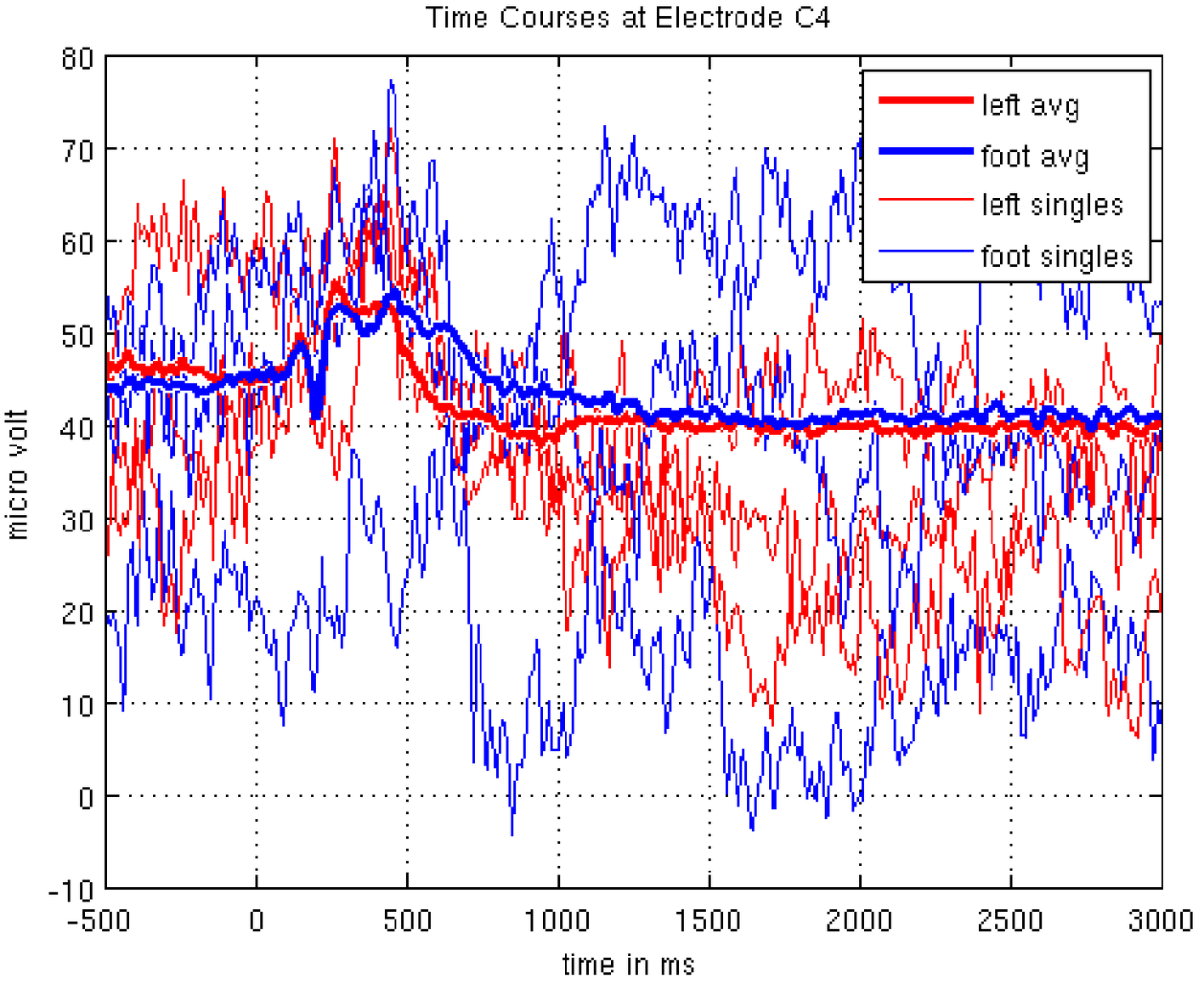
Single channel

- μ-rhythm (around 10 Hz) in motor and sensory cortex:



IMAGINATION of left arm

Variance I: Single-trial vs. Averaging

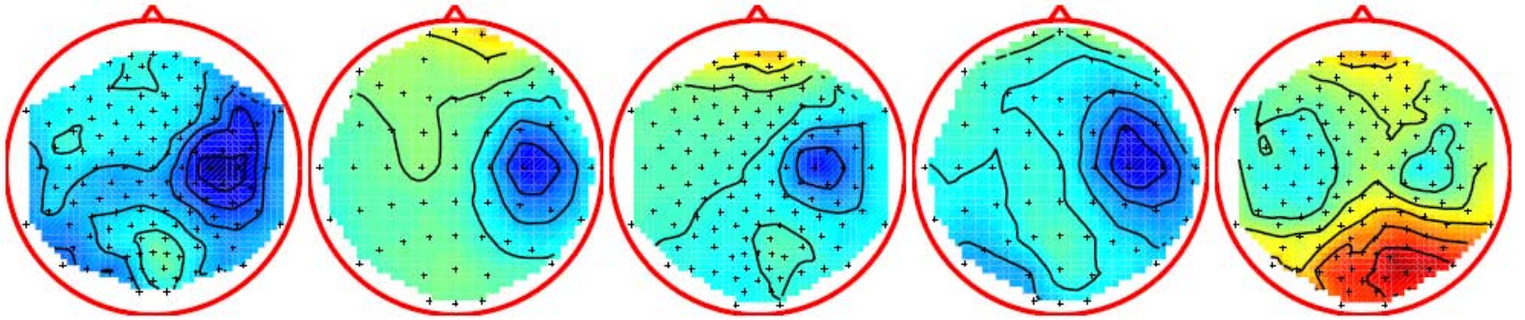


Single channel

Variance II: Session to Session Variability

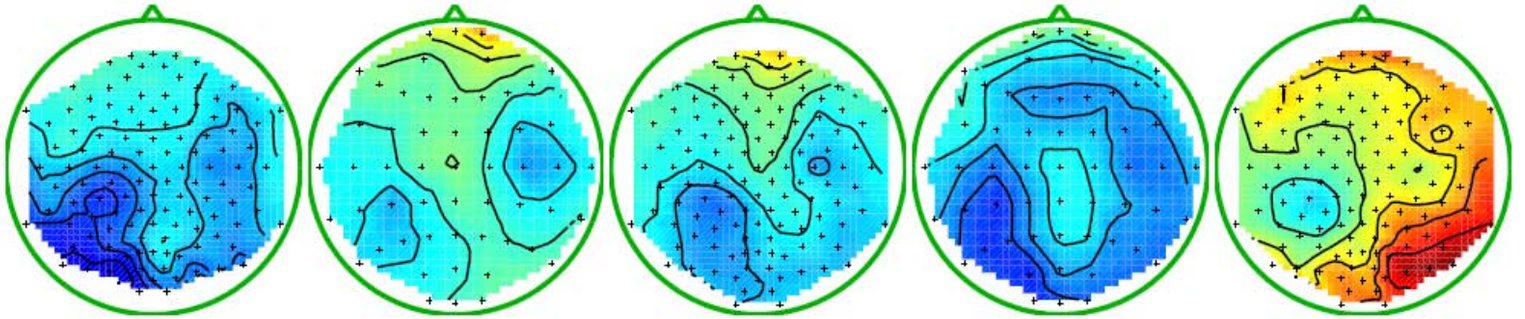
- Experiment: **One subject** imagined **left** vs. **right** hand movements on different days.
- Even though each ERD map represents an **average** across 140 trials, they exhibit an apparent diversity.

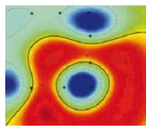
left



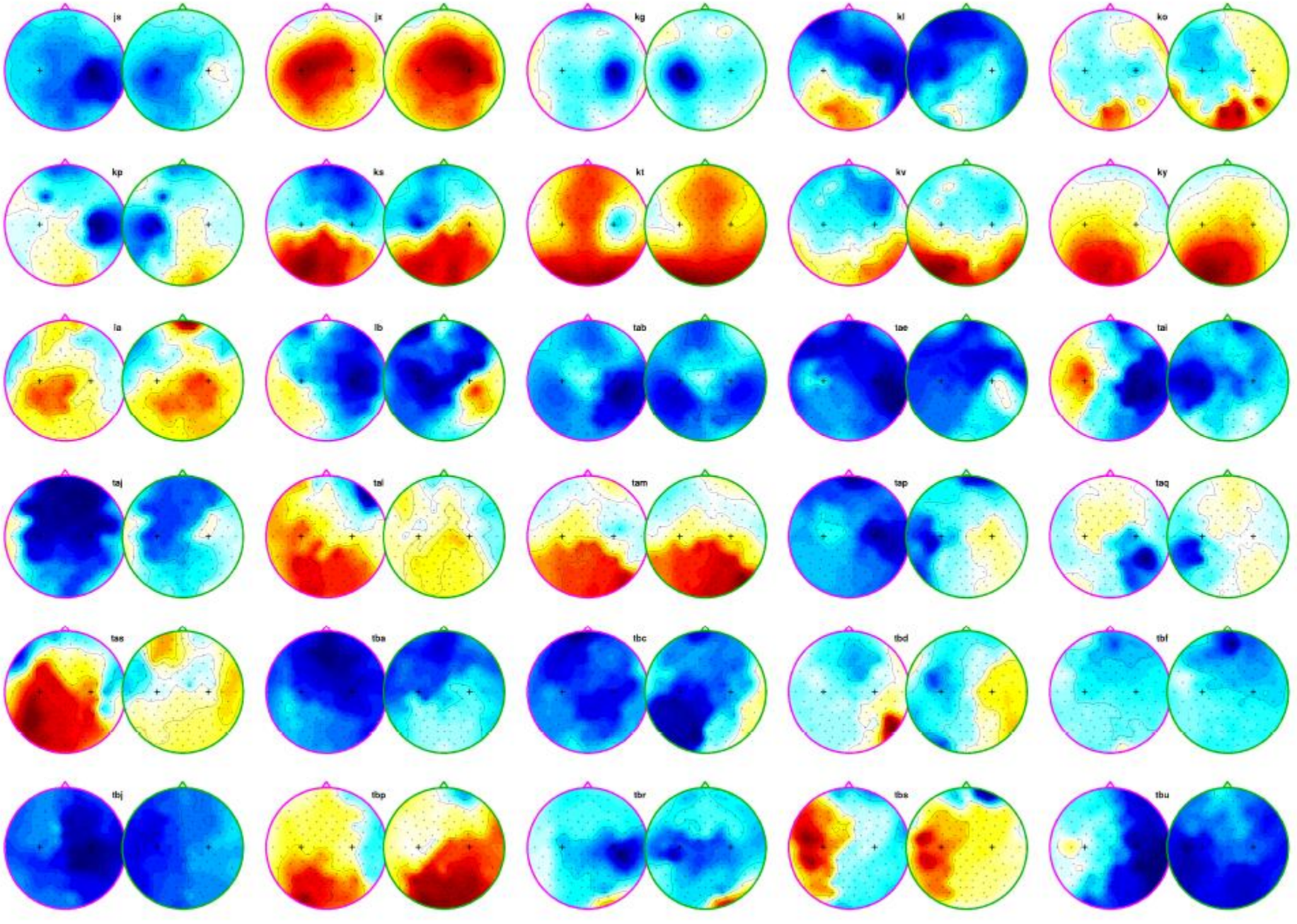
maps

right

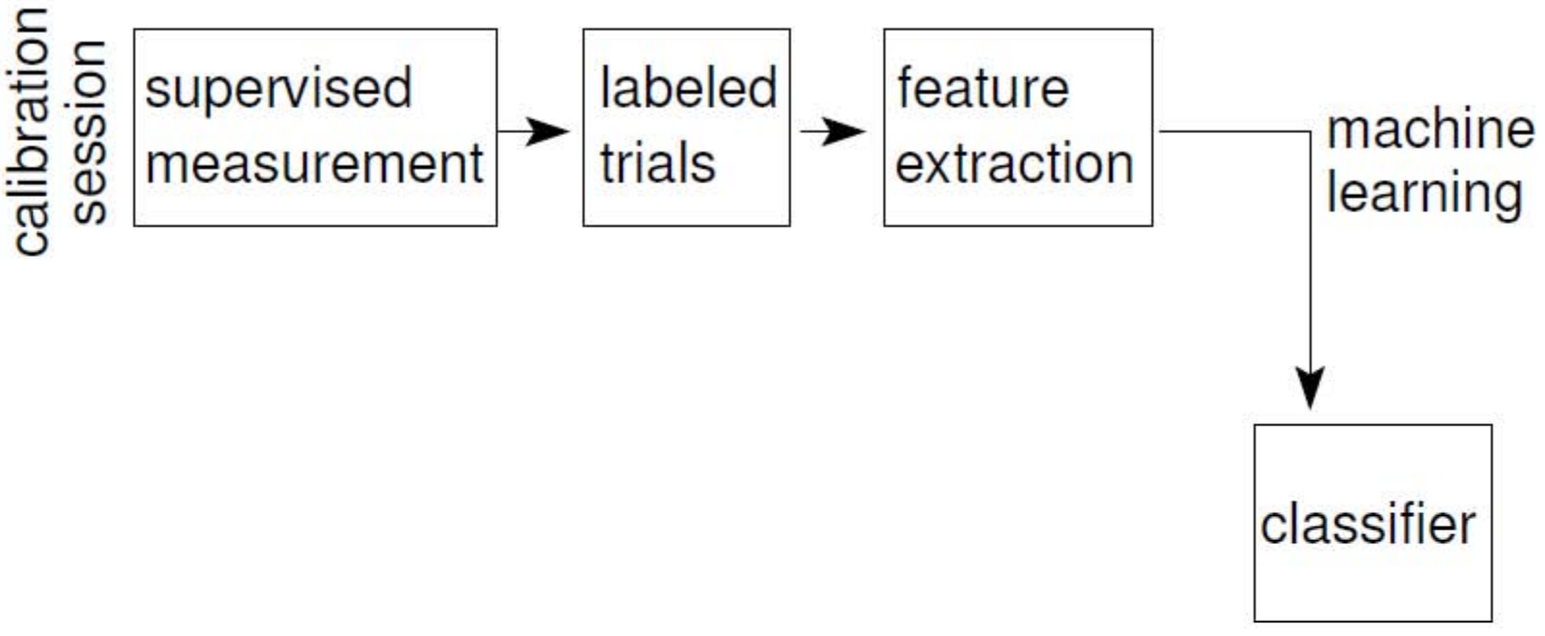




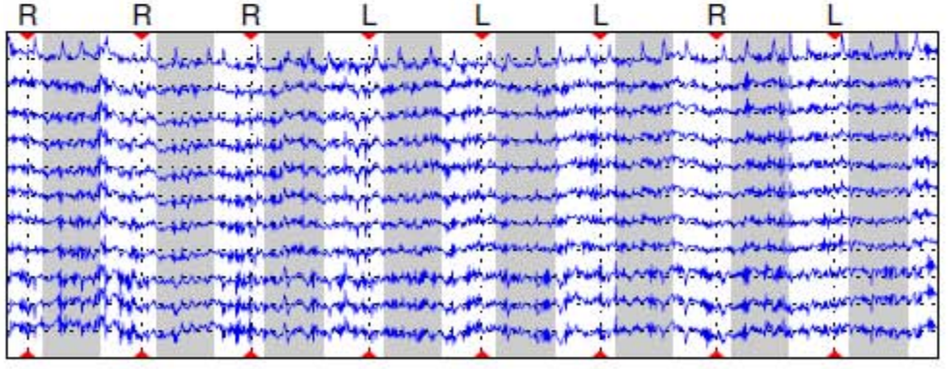
Variance III: inter subject variability [l vs r]



BCI with machine learning: training



offline: calibration (10–20 minutes)



collect training samples

BBCI paradigms

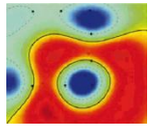
Leitmotiv: *›let the machines learn‹*

- healthy subjects *untrained* for BCI

A: training 20min: right/left hand **imagined** movements

→ infer the respective brain acivities (ML & SP)

B: online feedback session

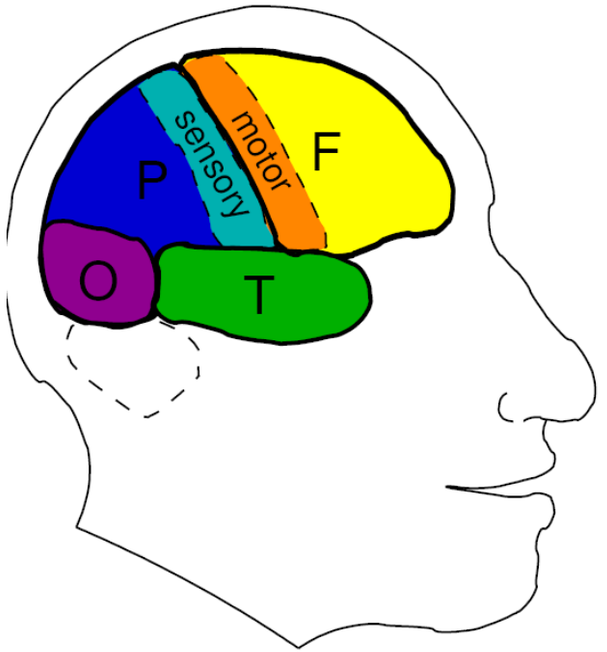


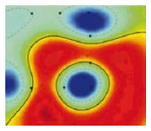
BBCI paradigms

Leitmotiv: *›let the machines learn‹*

- healthy subjects (BCI *untrained*) perform "imaginary" movements (ERD/ERS)

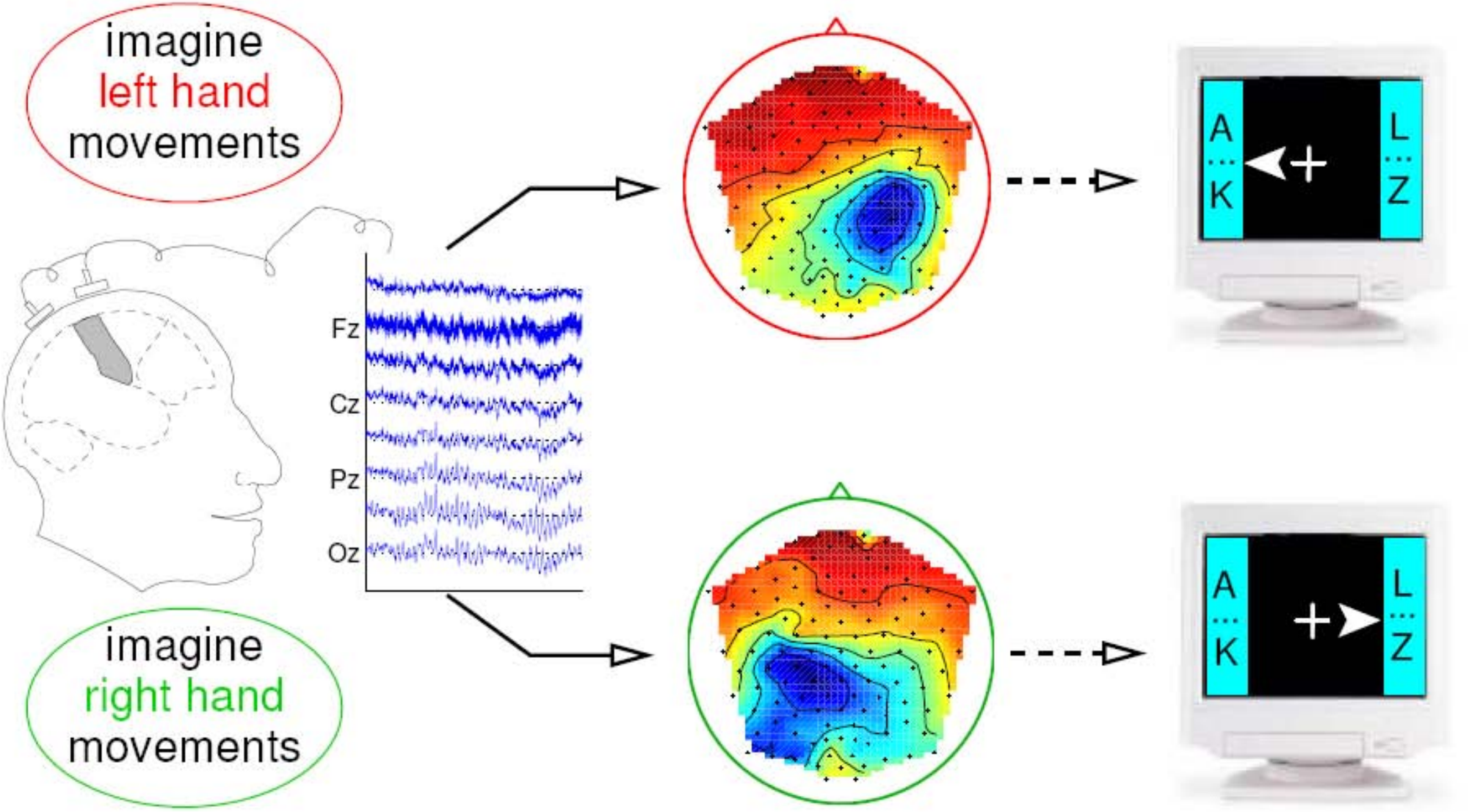
- instruction: imagine
  - squezzing a ball,
  - kicking a ball,
  - feel touch





Playing with BCI: training session (20 min)

Machine learning approach to BCI: infer prototypical pattern

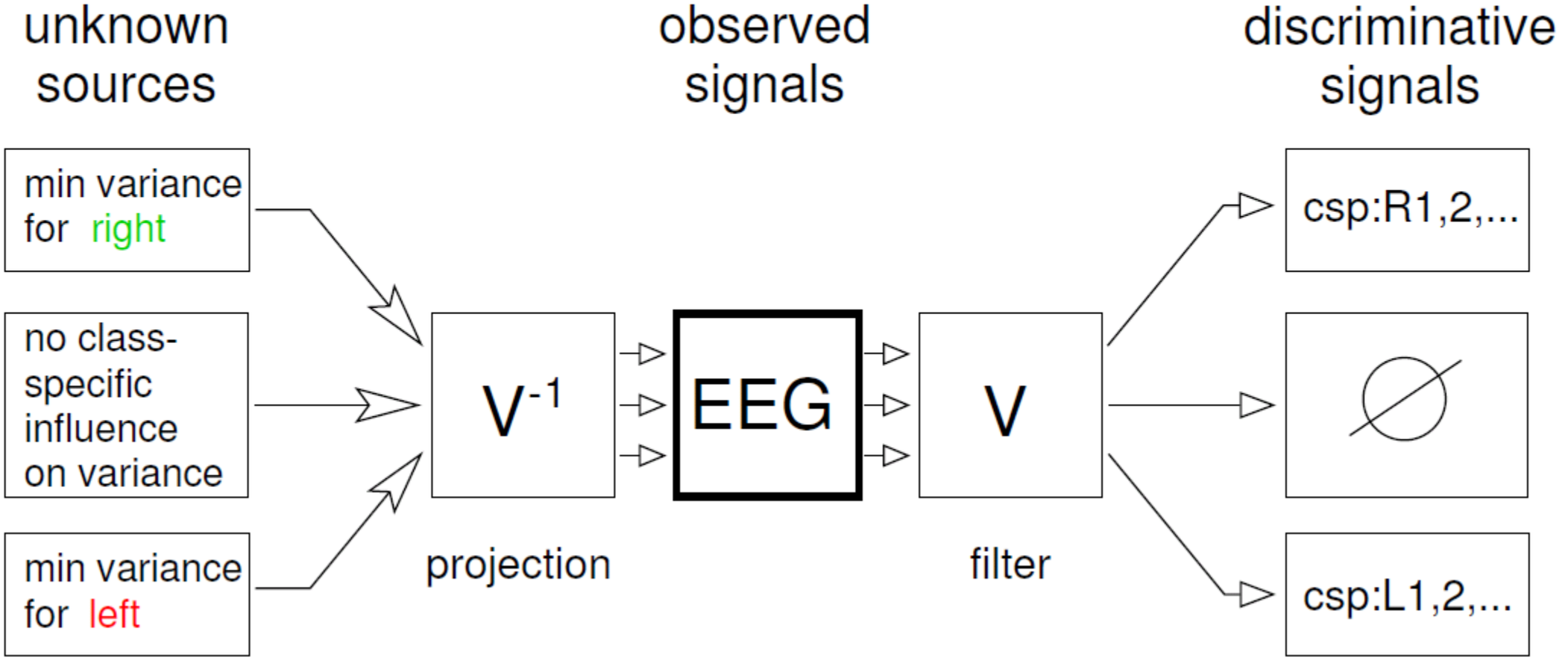


Inference by CSP Algorithm

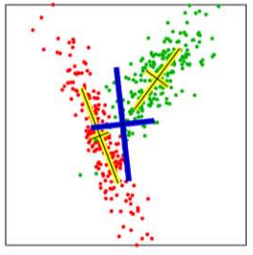
Common Spatial Pattern Analysis

Goal: Find spatial filters that optimally capture modulations of brain rhythms

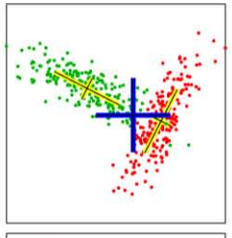
Observation: power of a brain rhythm $\sim$ variance of band-pass filtered signal.



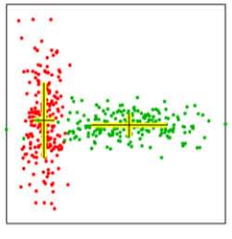
Common Spatial Patterns for 2 classes



Original data: Each class has a specific spatial extension.
Let Σ_1 and Σ_2 be the covariance matrices of the two classes.
The blue cross visualizes the covarianz matrix of $\Sigma_1 + \Sigma_2$.

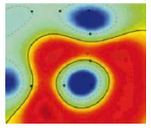


Make a whitening of $\Sigma_1 + \Sigma_2$, i.e., determine matrix P such that $P(\Sigma_1 + \Sigma_2)P^\top = I$ (possible due to positive definiteness of $\Sigma_1 + \Sigma_2$).
➤ Principal axis of the classes are perpendicular. Define: $\hat{\Sigma}_i = P\Sigma_i P^\top$.

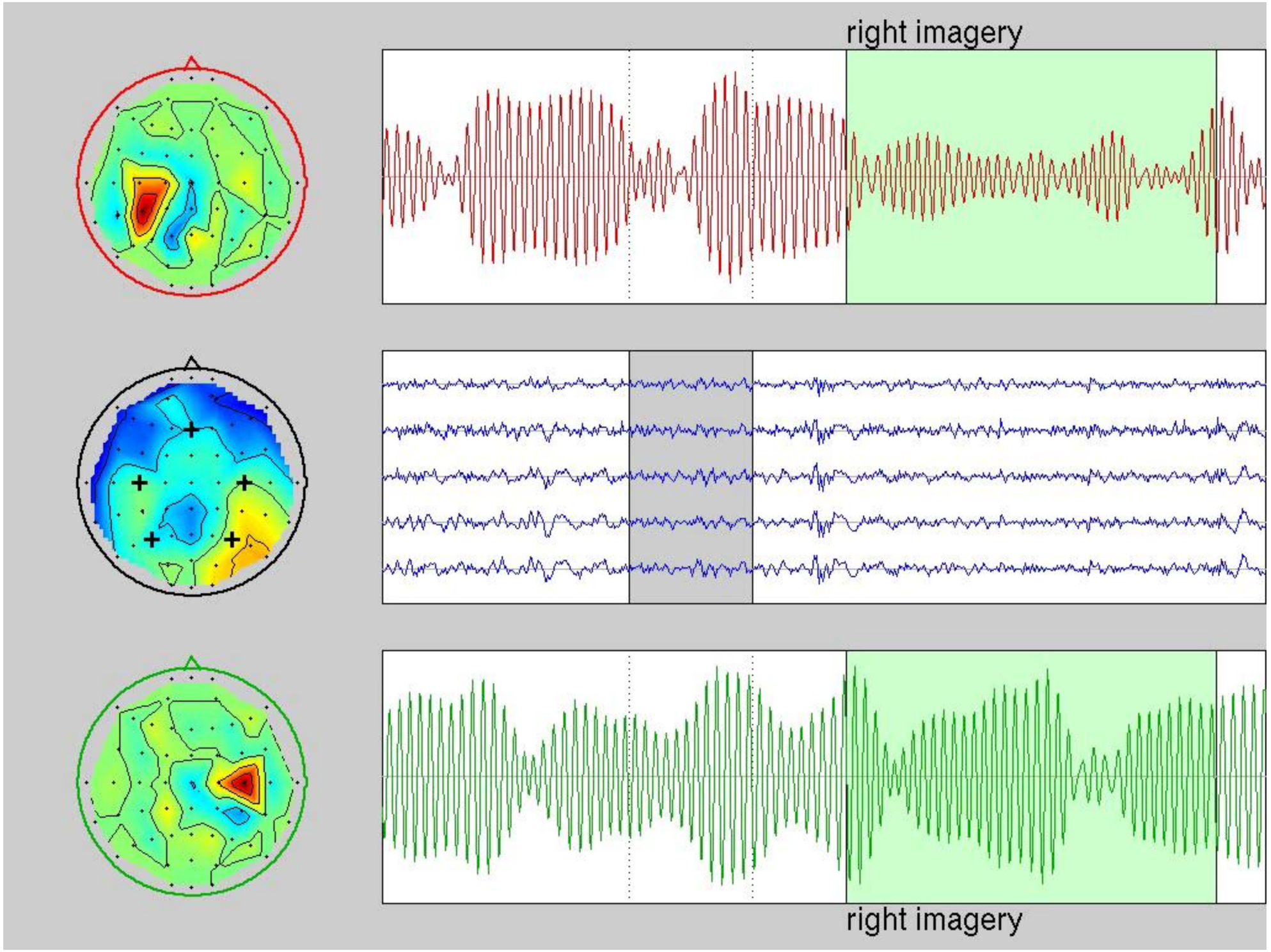


Calculate orthogonal matrix R and diagonal maxtrix D by spectral theory such that $\hat{\Sigma}_1^\top = RDR^\top$. Therefore $\hat{\Sigma}_2^\top = R(1-D)R^\top$ since $\hat{\Sigma}_1 + \hat{\Sigma}_2 = I$.
➤ Variance along the axis of input space is complementatory with respect to the two classes.

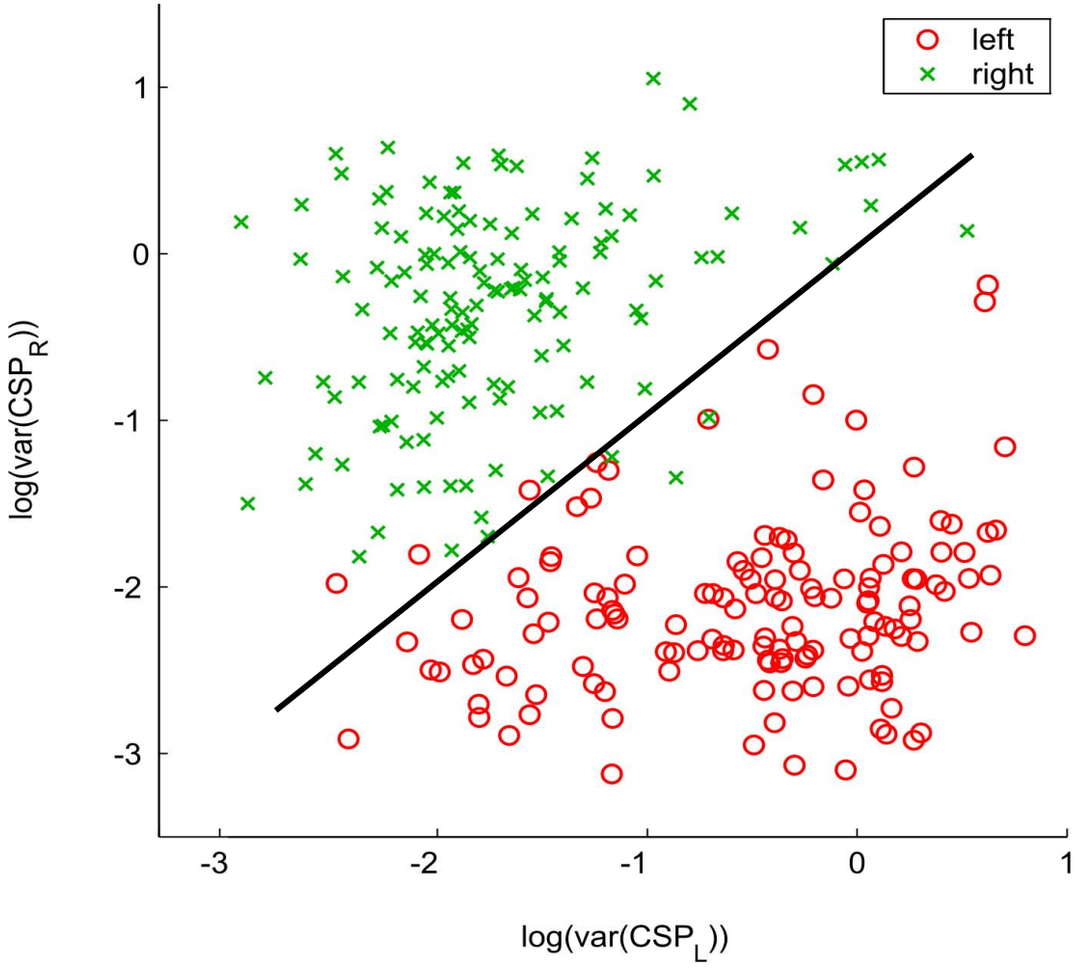
Essential idea for multi-class extension:
CSP is based on the **simultaneous diagonalization** of two covariance matrices with corresponding eigenvalues summing up to 1.

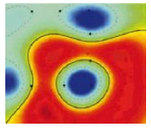


[cf. Blankertz et al. 2008, Lemm et al. 2005, Dornhege et al. 2006, Tomioka & Müller in Press]

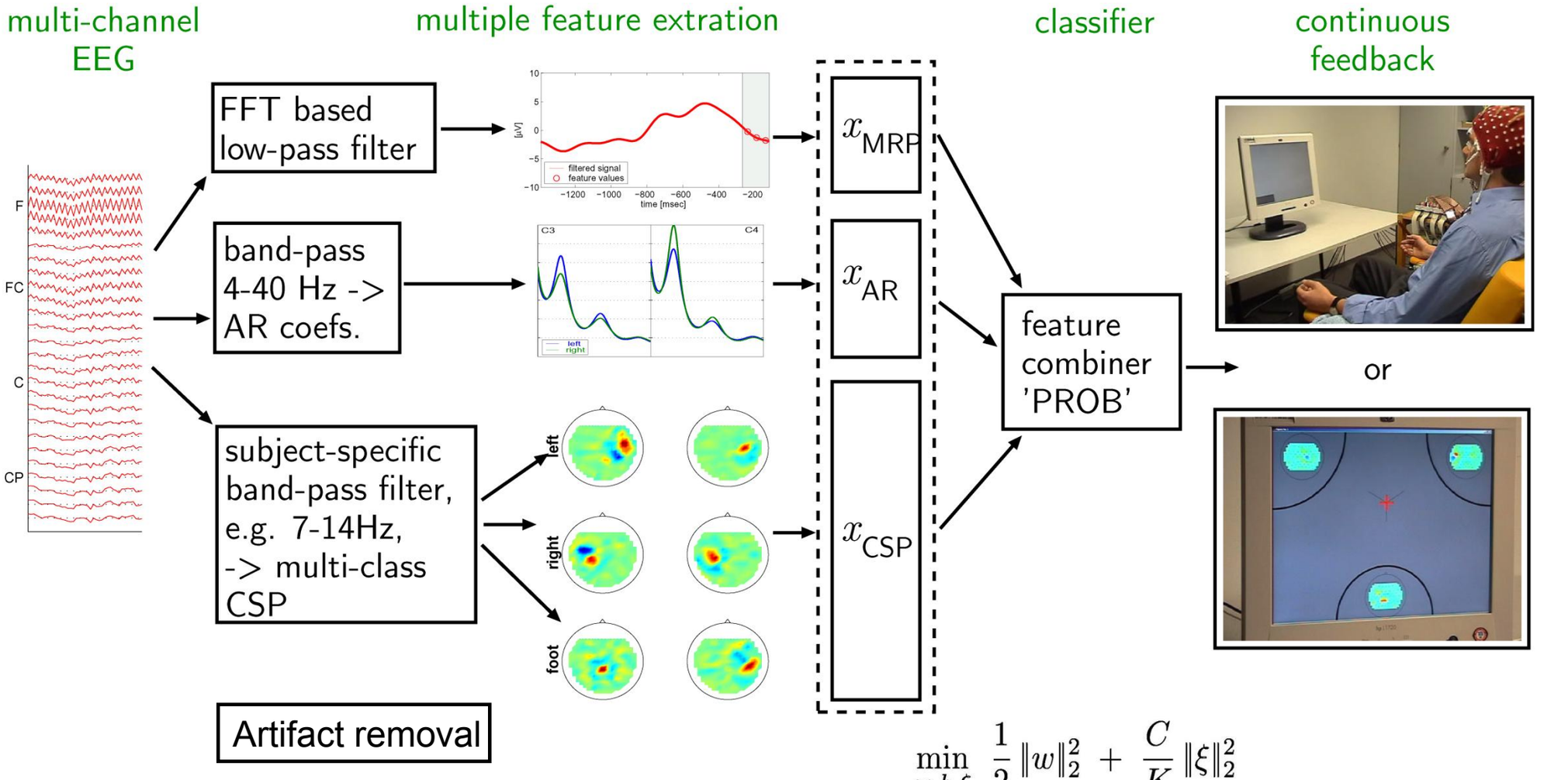

right imagery
right imagery

Distribution of EEG features





BBCI Set-up

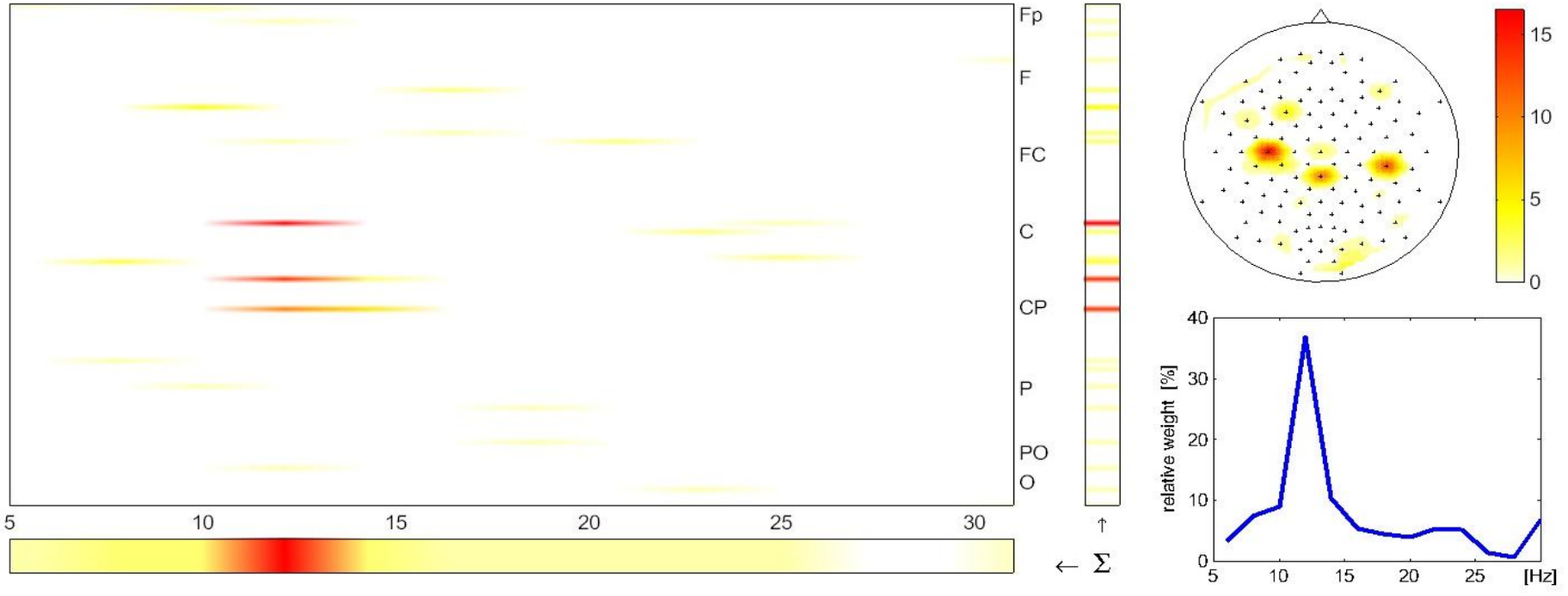


$$\min_{w,b,\xi} \ \frac{1}{2}\|w\|_2^2 + \frac{C}{K}\|\xi\|_2^2$$

$$\text{subject to} \quad y_k(w^\top x_k + b) = 1 - \xi_k \quad \text{for } k = 1, \ldots, K$$

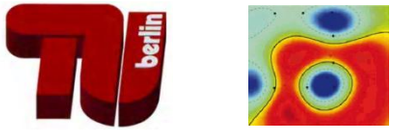
[cf. Müller et al. 2001, 2007, 2008, Dornhege et al. 2003, 2007, Blankertz et al. 2004, 2005, 2006, 2007, 2008]

What can Machine Learning tell us about physiology?



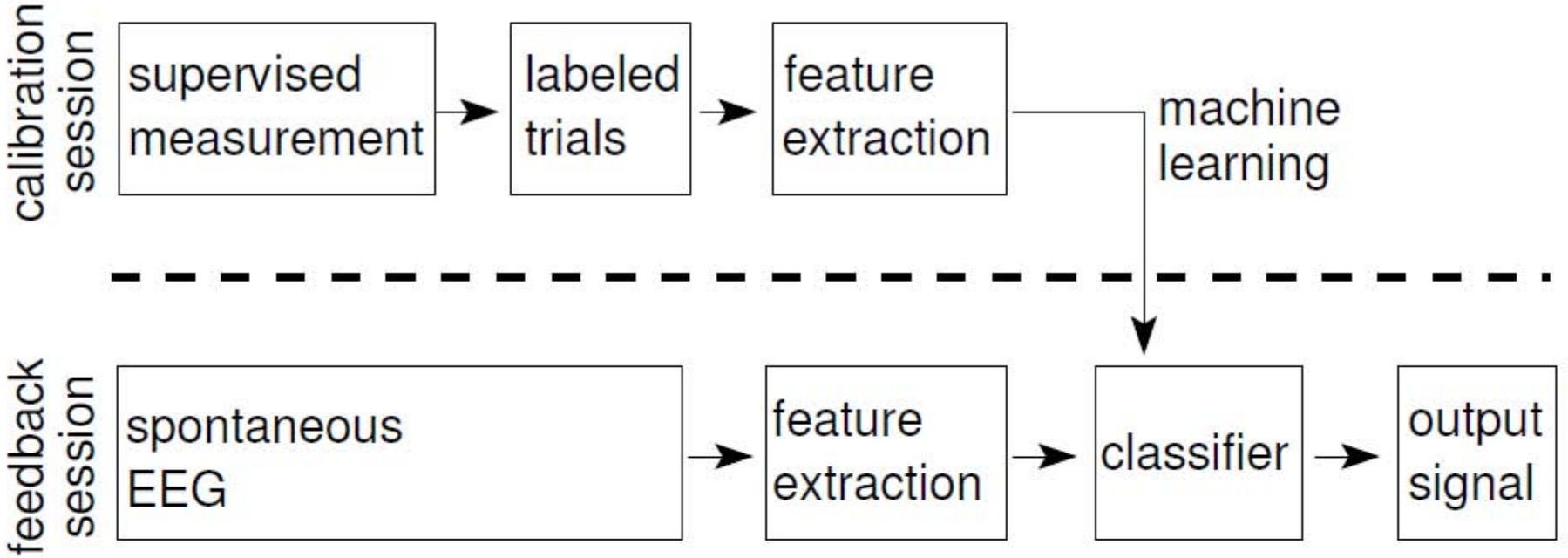
$$\min_{w,b,\xi} \quad \frac{1}{2}\|w\|_1 + \frac{C}{K}\|\xi\|_1$$

$$\text{subject to} \qquad y_k(w^\top x_k + b) = 1 - \xi_k \quad \text{for } k = 1, \ldots, K$$

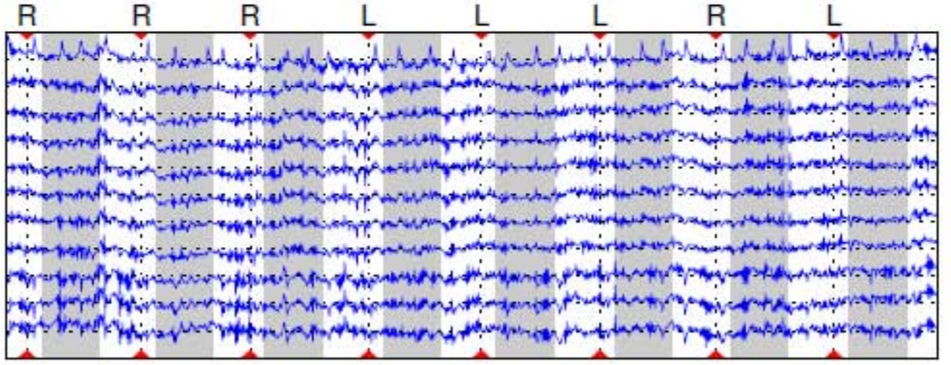


[cf. Blankertz et al. 2001, 2006]

BCI with machine learning: feedback

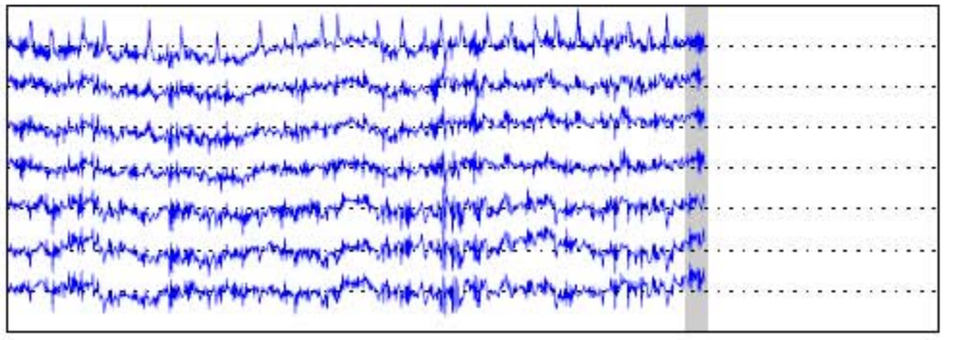


offline: calibration (10–20 minutes)



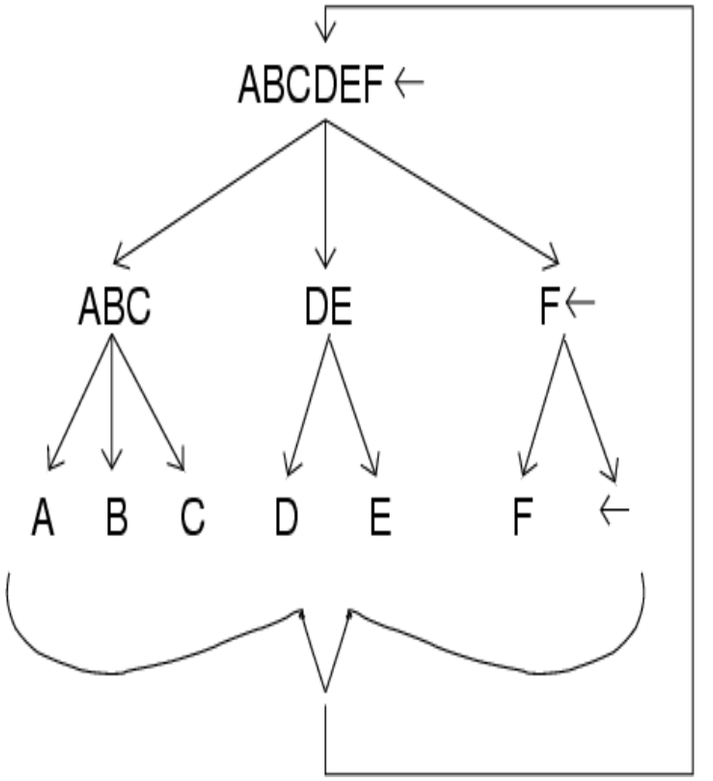
collect training samples

online: feedback (up to 6 hours)

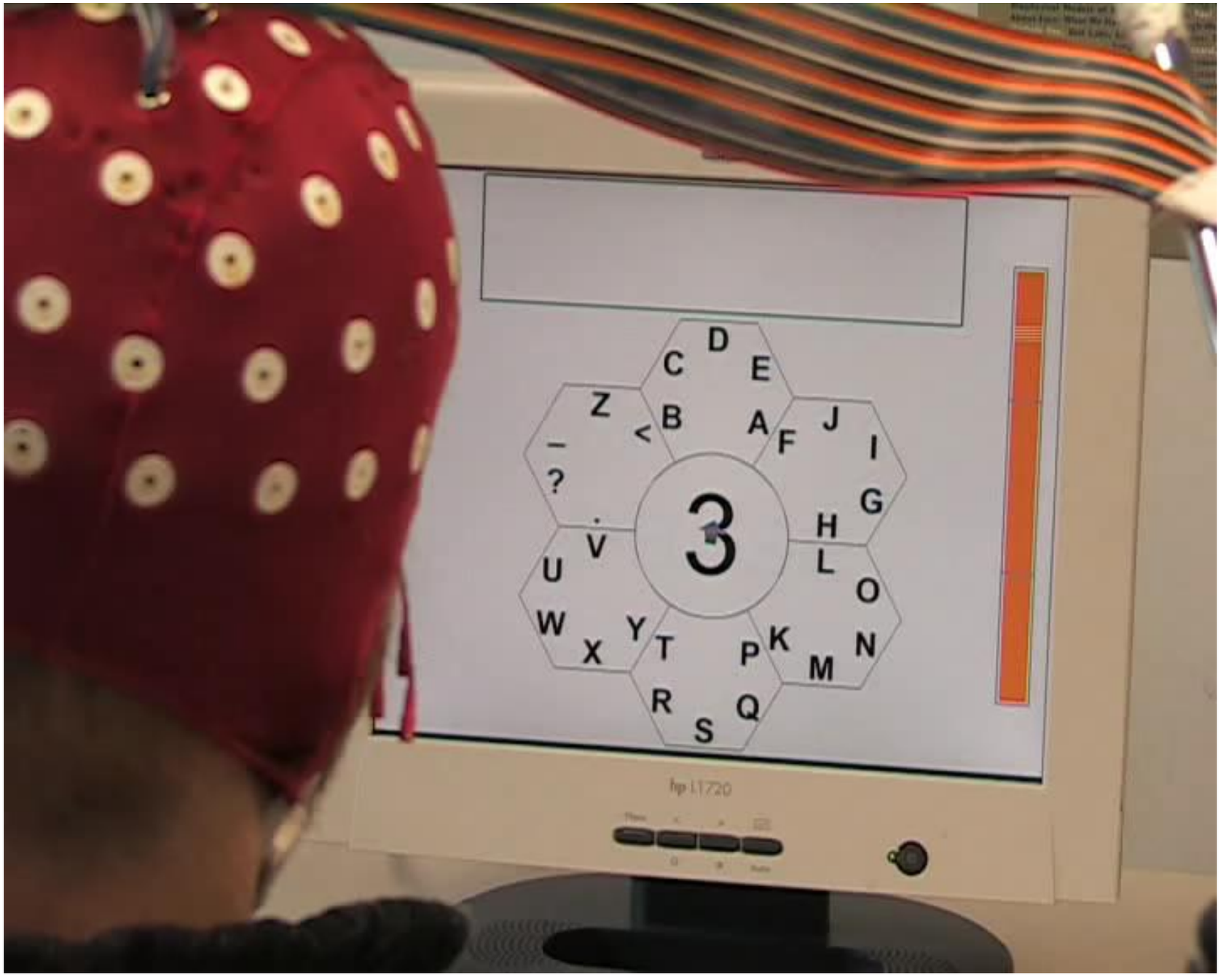


classification of sliding windows ($\leq$ 1s)

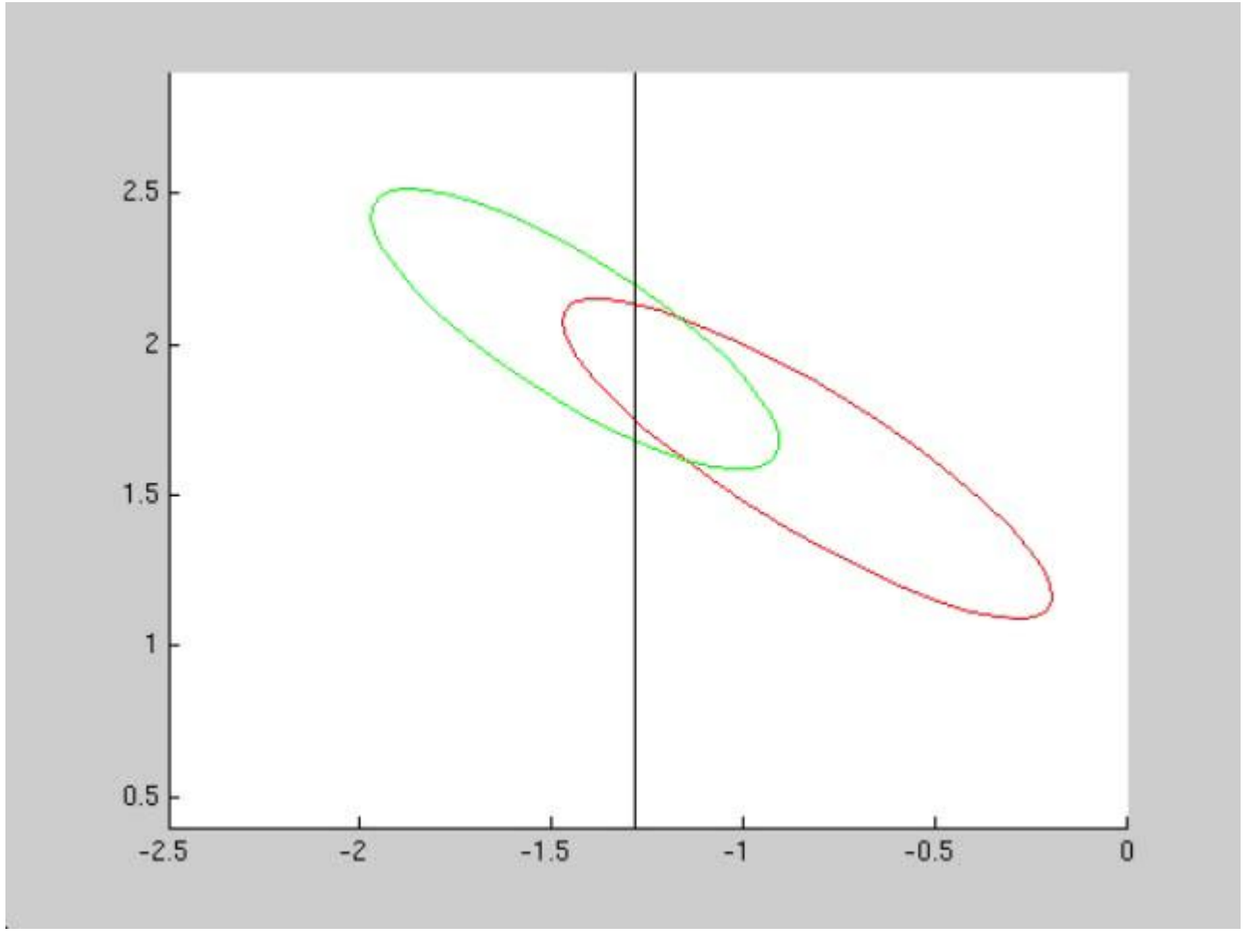
Spelling with BBCI: a communication for the disabled I



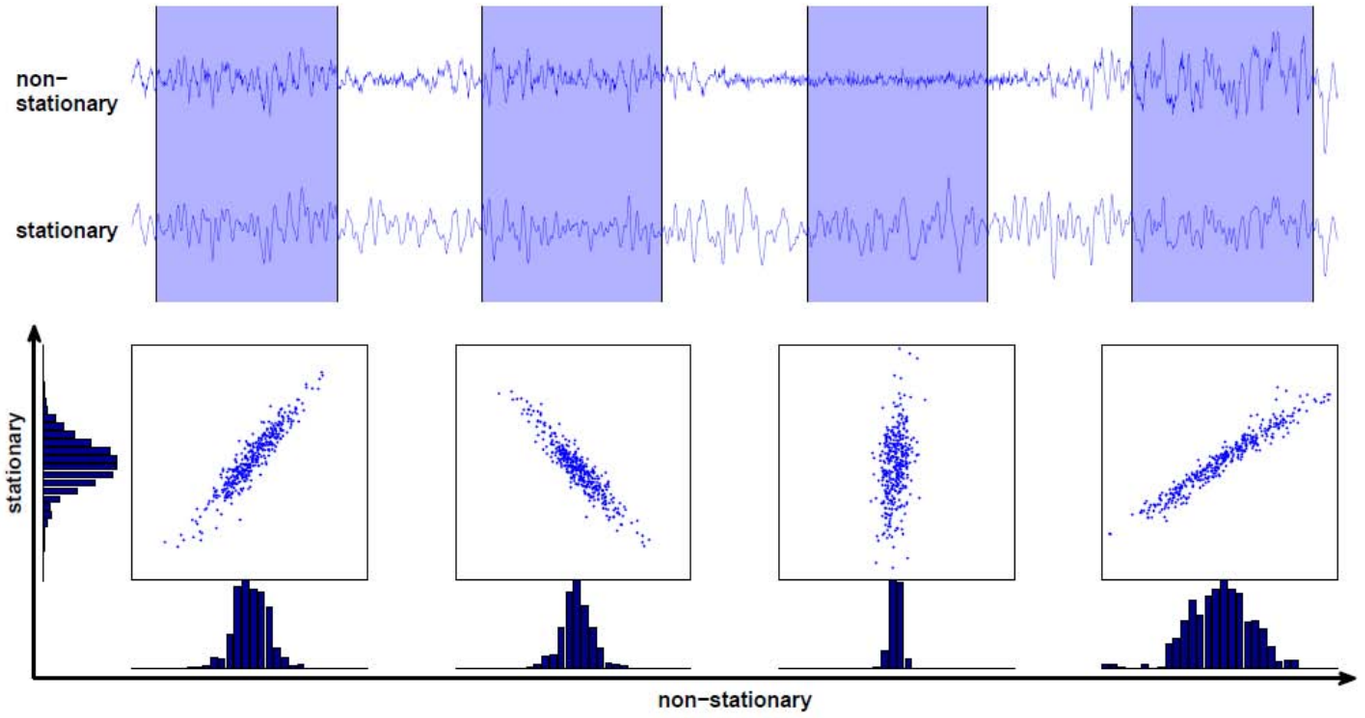
Spelling with BBCI: a communication for the disabled II



Future Issues: Shifting distributions within experiment



Splitting into stationary and nonstationary subspace: SSA



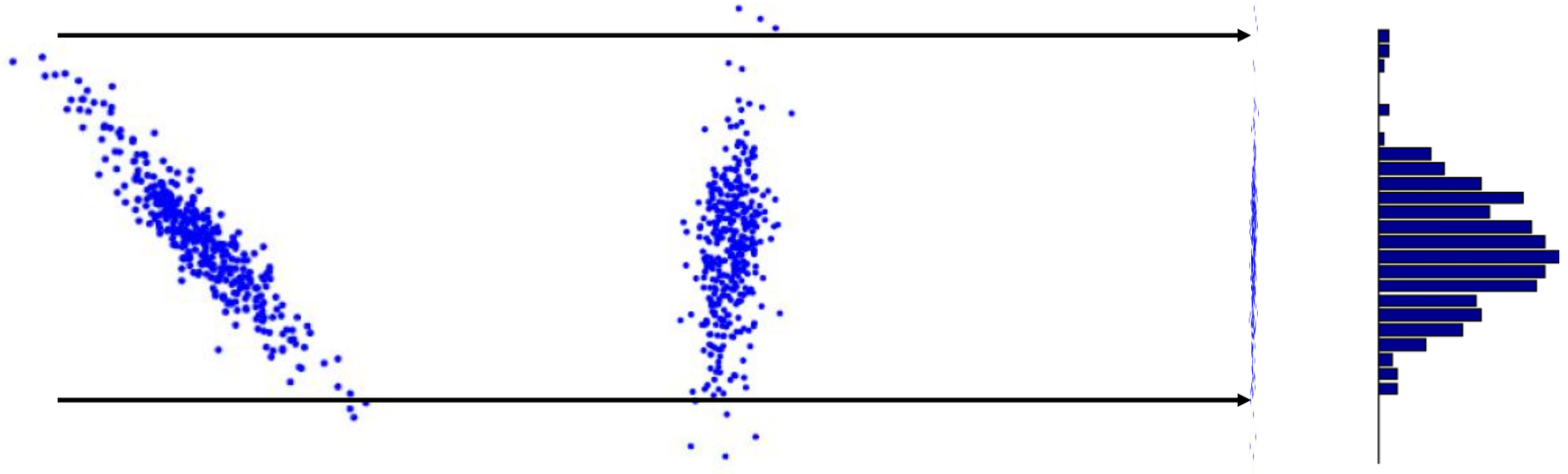
- d stationary source signals $s^{\mathfrak{s}}(t) \in \mathbb{R}^d$
- $D - d$ non-stationary source signals $s^{\mathfrak{n}}(t) \in \mathbb{R}^{(D-d)}$
- Observed signals: instantaneous linear superpositions of sources

$$x(t) = As(t) = \begin{bmatrix} A^{\mathfrak{s}} & A^{\mathfrak{n}} \end{bmatrix} \begin{bmatrix} s^{\mathfrak{s}}(t) \\ s^{\mathfrak{n}}(t) \end{bmatrix}$$

invert

[cf. Bünau, Meinecke, Kiraly, Müller PRL 09]

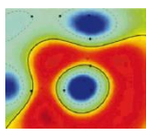
SSA



given: Epochs X_i of Data points in $\mathbb{C}^n$

wanted: Linear subspace S of $\mathbb{C}^n$ such that
marginalized data sets $X_i \mid_S$ look the same

„stationary projection”



Inverting the SSA Mixing Model

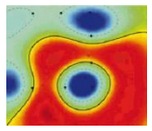
Model

$$x(t) = As(t) = \begin{bmatrix} A^{\mathfrak{s}} & A^{\mathfrak{n}} \end{bmatrix} \begin{bmatrix} s^{\mathfrak{s}}(t) \\ s^{\mathfrak{n}}(t) \end{bmatrix}$$

Goal of SSA

Given only $x(t)$, find an estimate for the demixing matrix $\hat{B} = \hat{A}^{-1}$ that separates $\mathfrak{s}$-sources from $\mathfrak{n}$-sources.

$$\begin{bmatrix} \hat{s}^{\mathfrak{s}}(t) \\ \hat{s}^{\mathfrak{n}}(t) \end{bmatrix} = \hat{B}x(t) = \begin{bmatrix} \hat{B}^{\mathfrak{s}} \\ \hat{B}^{\mathfrak{n}} \end{bmatrix} x(t)$$



SSA: Algorithm idea

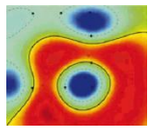
Stationarity in the context of SSA

A timeseries $x(t)$ is *weakly stationary*, if its mean and covariance is constant over time, i.e.

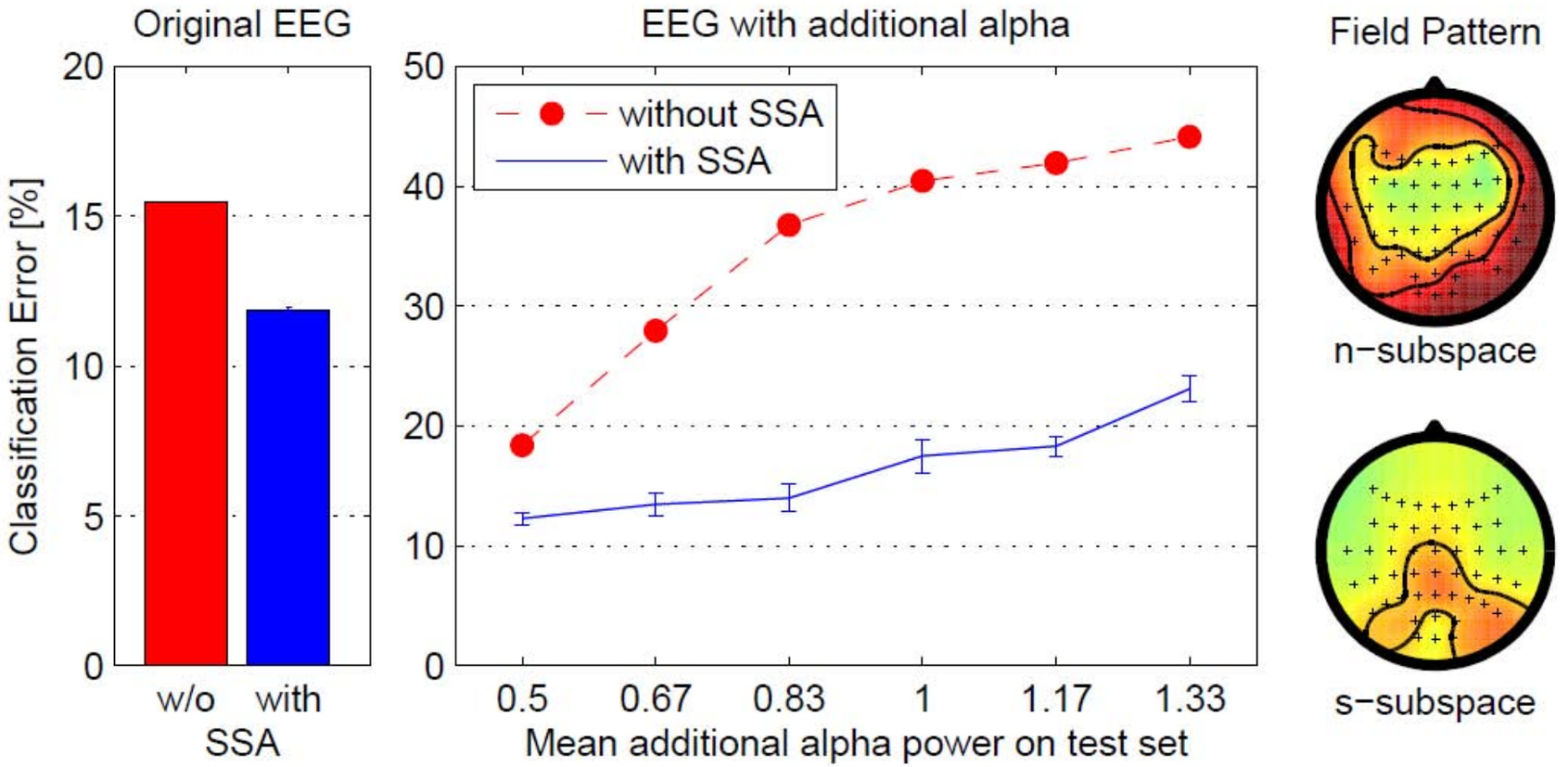
$$\mathbb{E}[x(t)] = \mathbb{E}[x(t+\tau)] \text{ and}$$
$$\mathbb{E}[x(t)^\top x(t)] = \mathbb{E}[x(t+\tau)^\top x(t+\tau)] \; \forall t, \tau.$$

Algorithmic Approach

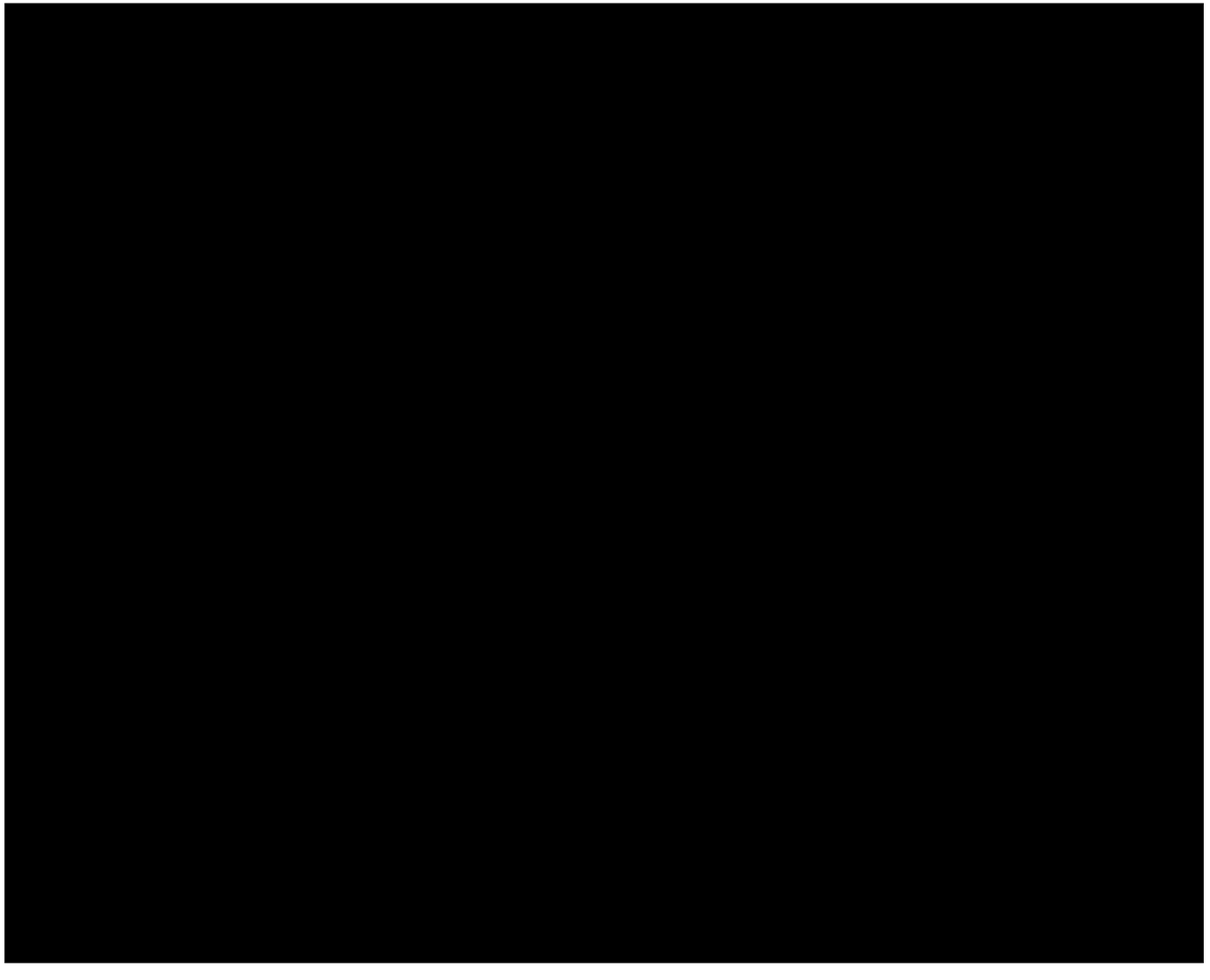
Divide the timeseries into N epochs. Find the projection $\hat{B}^{\mathfrak{s}}$ to the stationary sources which minimizes the difference in mean and covariance between each epoch $(\hat{\mu}_i^{\mathfrak{s}}, \hat{\Sigma}_i^{\mathfrak{s}})$ and the whole dataset $(\bar{\hat{\mu}}^{\mathfrak{s}}, \bar{\hat{\Sigma}}^{\mathfrak{s}})$ for the estimated stationary sources.



Application to Brain-Computer-Interfacing

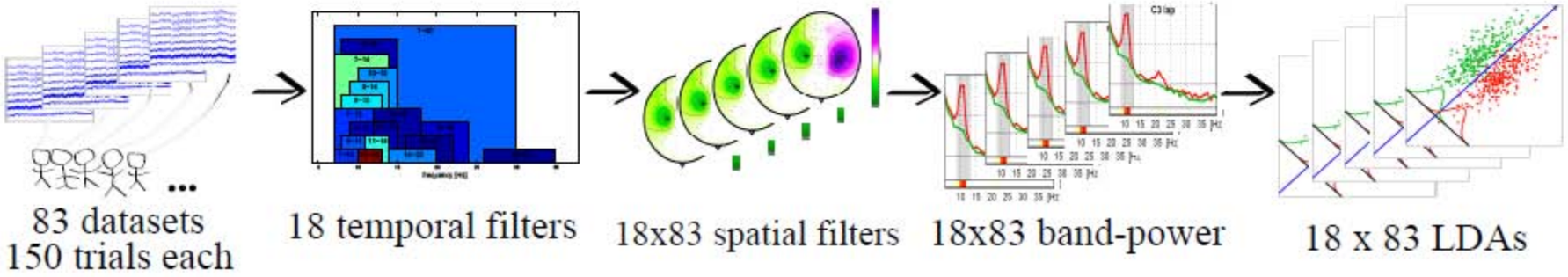


Real Man Machine Interaction



Towards a subject independent BCI decoder

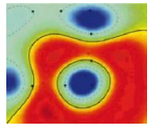
- we end up with **1494 features** and $83 \cdot 150 = $ **12450 trials**
- to find a **subject-independent BCI**, we can perform ℓ_1-regularized regression (or others like LMM) using **leave-one-subject-out cross-validation**
- note that our trials have a **grouping** structure



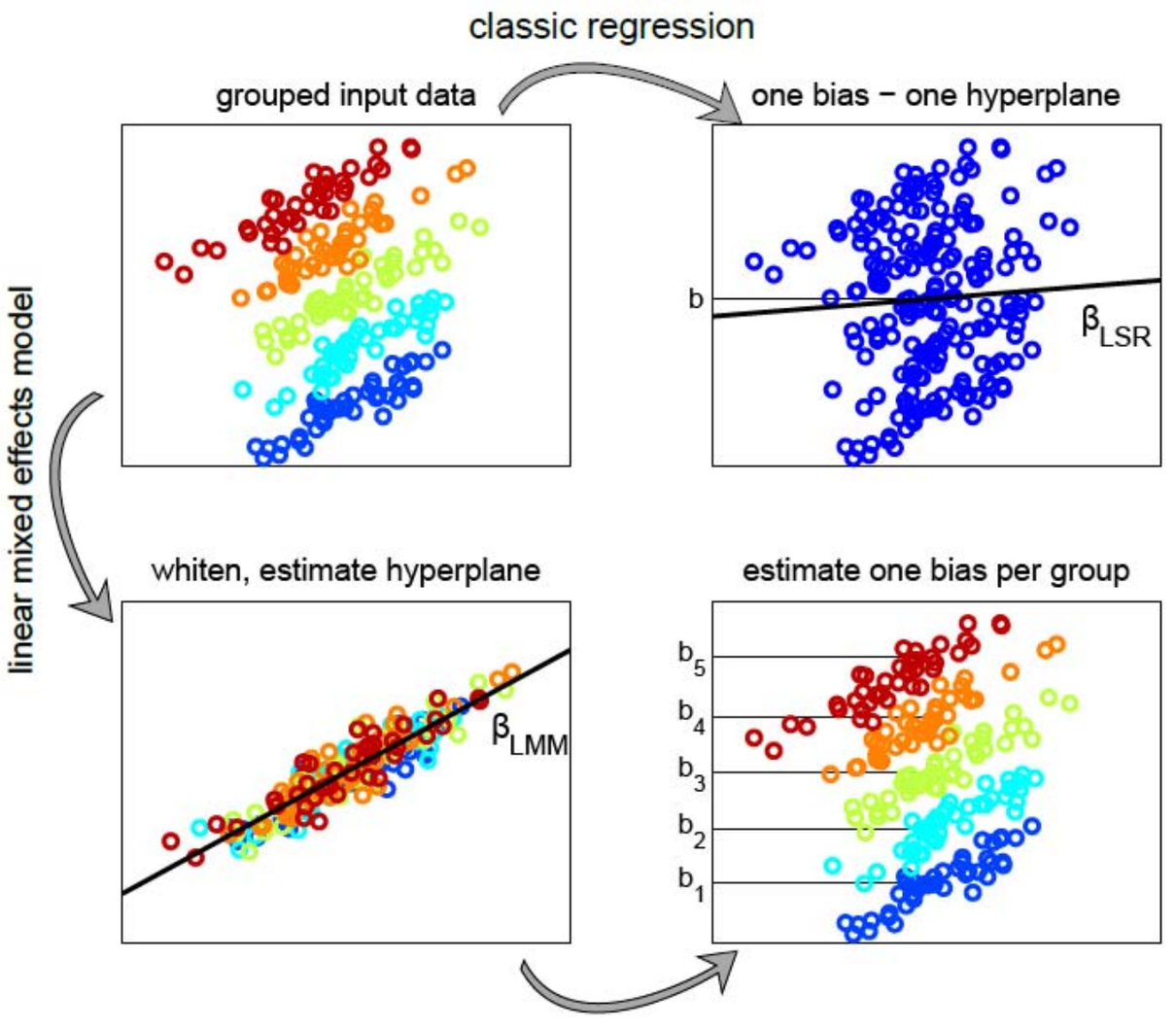
Model formulation

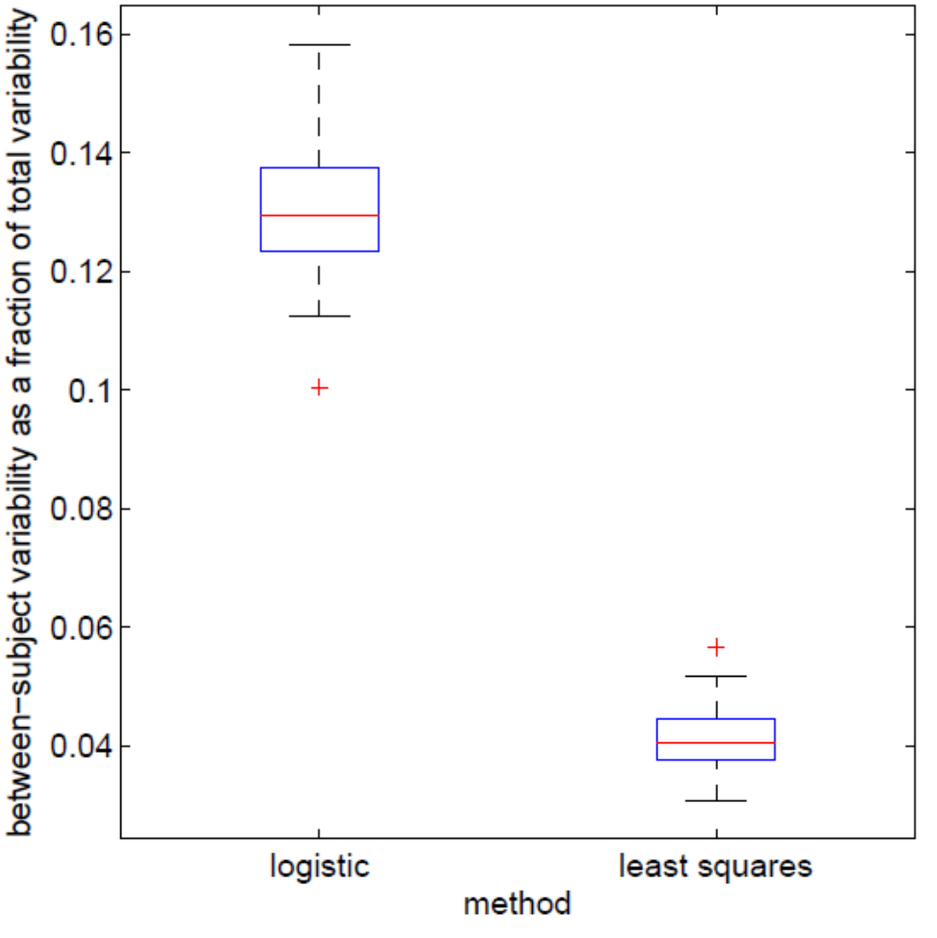
- Reminder – Linear regression:
  - $\mathbf{y} = \mathbf{X}\beta + \varepsilon$
- Mixed effects model with n groups:
  - $\mathbf{y}_i = \mathbf{X}\beta + \mathbf{Z}_i\mathbf{b}_i + \varepsilon_i \quad \forall i \in \{1 \dots n\}$
  - Consists of n simultaneous equations, one for each group
  - The equations are coupled by the common term $\mathbf{X}\beta$
  - Each equation has a group-dependent term $\mathbf{Z}_i\mathbf{b}_i$
  - In our case, each $\mathbf{Z}_i$ is simply a vector of ones, i.e. the corresponding $\mathbf{b}_i$ is scalar and represents the bias of group i
  - So-called **random intercepts model**
- Since we expect our features to be redundant and are aiming for better interpretability, we enforce sparsity by adding an ℓ_1 penalty

$$b_i \sim \mathcal{N}_q(0, \tau^2 I_q)$$
$$\varepsilon_i \sim \mathcal{N}_{n_i}(0, \sigma^2 I_{n_i})$$



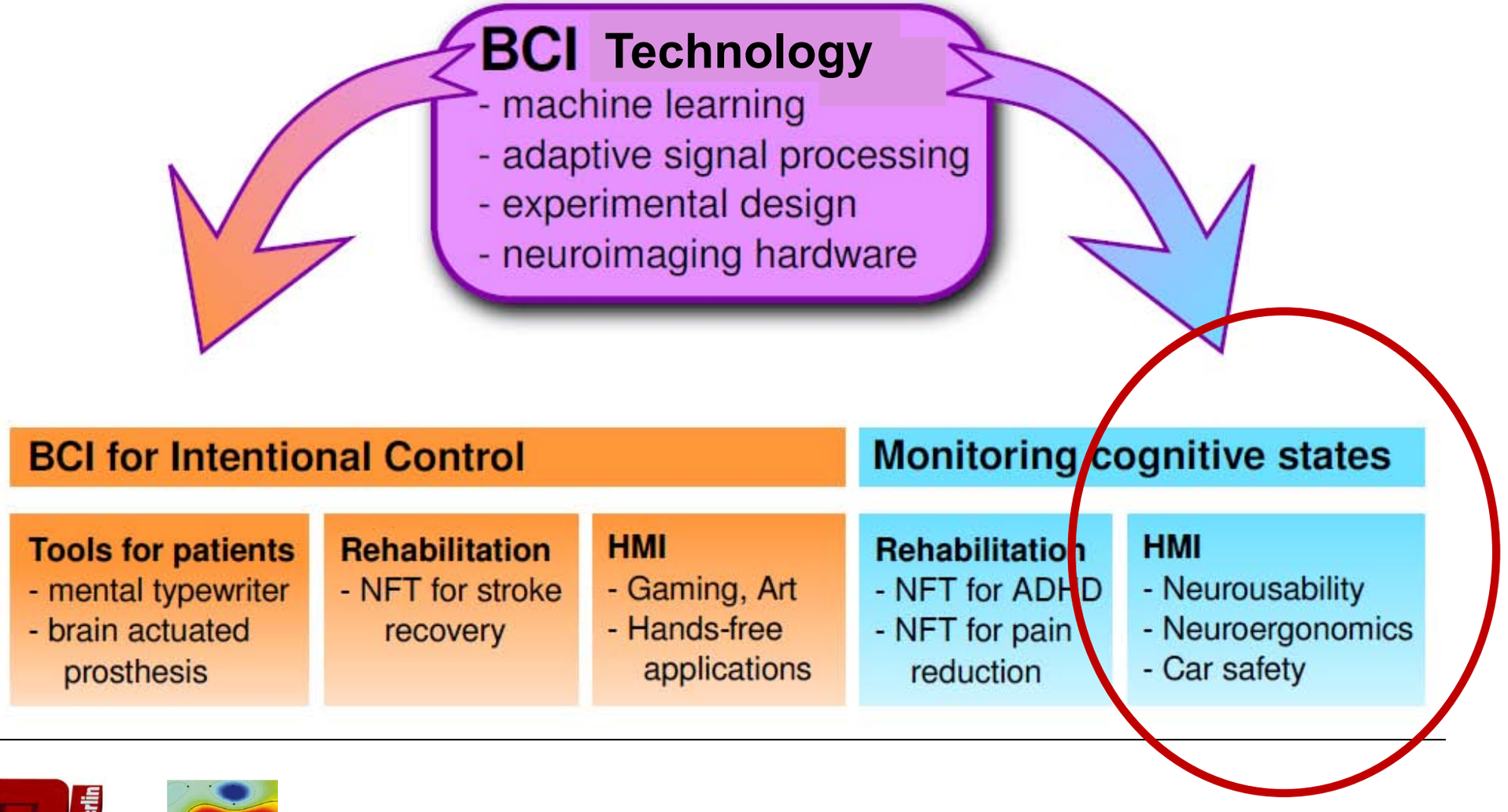
Linear Mixed Effects Model: intuition





[Fazli, Müller et al. 2011]

Towards industrial applications of BCI Technology



[Blankertz et al 2010 Front. Neurosci.]

Towards Application: Predicting drowsiness

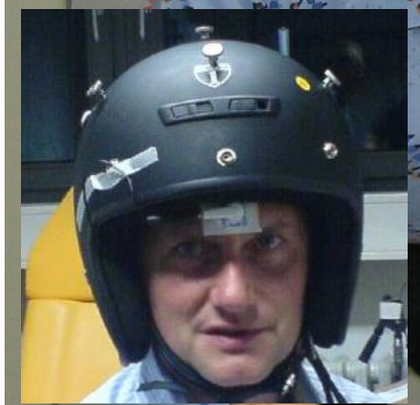
Application: Cognitive workload and drowsyness assessment

Assess workload with BCI and balance it by smart driver assistent system

Assess cognitive alertness

[Kohlmorgen, Müller et al 2007]

Future issues: sensors



Popescu et al 2007

Conclusion

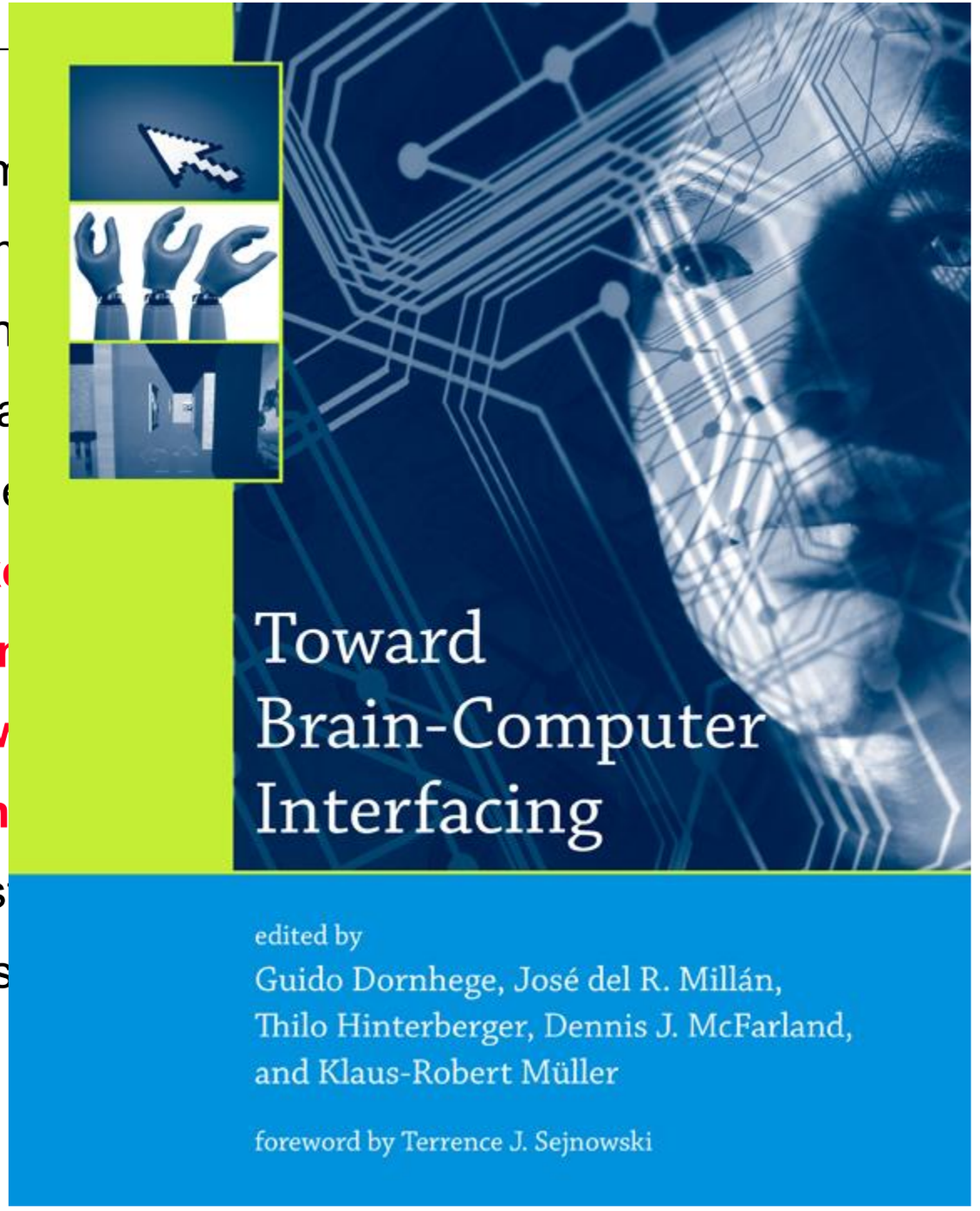
- BBCI: non-invasive with high Inform
- BBCI: Untrained, Calibration < 20m
- 5-8 letters/min mental typewriter on
- Machine Learning and modern data
- Applications: communication vs. me

  Rehabilitation: **TOBI EU IP, stroke**

  Computational Neuroscience: **Berr**

  Man Machine Interaction: **brain@v**
- BBCI Sensors, software: **IDA spin**
- towards no training, towards indus
- ‚illiterates', nonstationarity, wireless

FOR INFORMATION SEE:
www.bbci.de



Thanks to BBCI core team:

Gabriel Curio
Florian Losch
Volker Kunzmann
Frederike Holefeld
Vadim Nikulin@Charite

Florin Popescu
Andreas Ziehe
Steven Lemm
Motoaki Kawanabe
Guido Nolte@FIRST

Yakob Badower@Pico Imaging
Marton Danozci

Benjamin Blankertz
Michael Tangermann
Claudia Sannelli
Carmen Vidaurre
Siamac Fazli
Martijn Schreuter
Stefan Haufe
Thorsten Dickhaus
Frank Meinecke
Felix Biessmann@TUB

Matthias Krauledat
Guido Dornhege
Roman Krepki@industry

Collaboration with: U Tübingen, Bremen, Albany, TU Graz, EPFL, Daimler, Siemens, MES, MPIs, U Tokyo, TIT, RIKEN, Bernstein Center for Computational Neuroscience Berlin, Columbia, CUNY
Funding by: EU, BMBF and DFG

Overview of BCI Competitions

BCI competition I	BCI competition II
December 2001 – June 2002	December 2003 – June 2004
3 datasets	6 datasets
10 submissions	59 submissions
[Sajda et al., 2003]	[Blankertz et al., 2004]

BCI Competition III

- Dec 12th 2004 – May 31st 2005
- announcement of the results: between June 14th and 19th 2005
- 8 datasets from 5 different BCI groups with different tasks

For BCI IV Competition see www.bbci.de

berlin
brain computer
interface

FOR INFORMATION SEE: www.bbci.de

Machine Learning open source software initiative: MLOSS see

www.jmlr.org

Machine Learning and Signal Processing tools for Brain Computer Interfacing II

Klaus-Robert Müller, Benjamin Blankertz, Gabriel Curio et al.

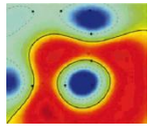
ERP Analysis

Method:

- classification of **spatio-temporal** features;
- *shrinkage* of the sample covariance matrix to counterbalance the estimation bias

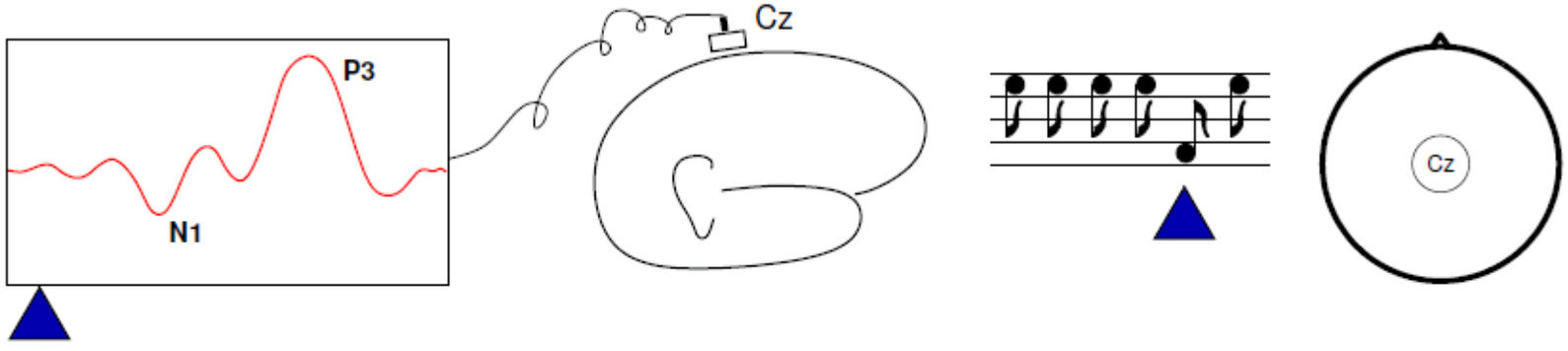
Application:

- classification of single-trial ERPs in an attention-based speller

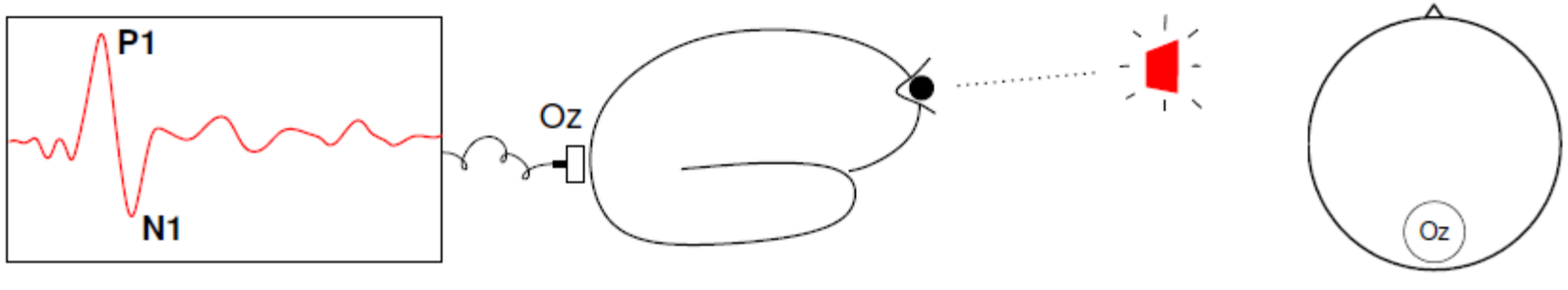


Neurophysiological Background for ERPs

An infrequent stimulus in a series of standard stimuli evokes a P300 component at central scalp position *if attended*:



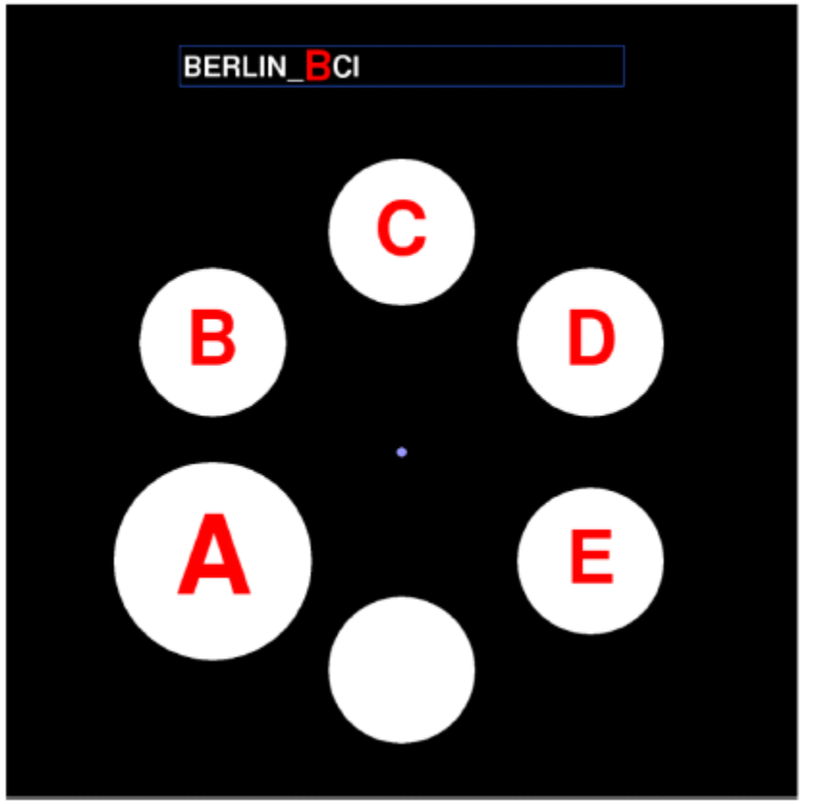
The presentation of a visual stimulus elicits a Visual Evoked Potential (VEP) in visual cortex *if focused*:



Experimental Design

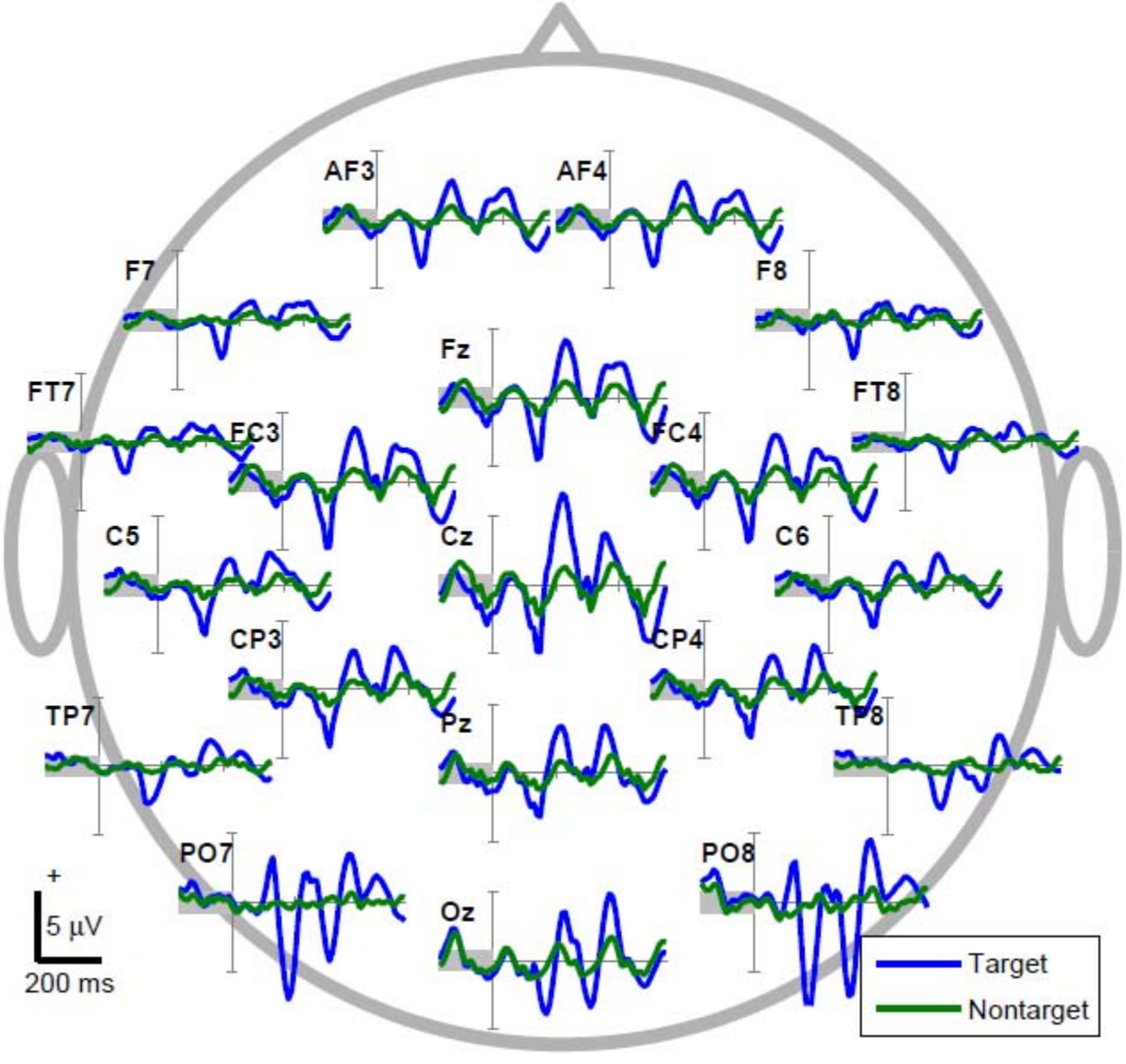
Classic Matrix Speller

Attention-based Hex-o-Spell



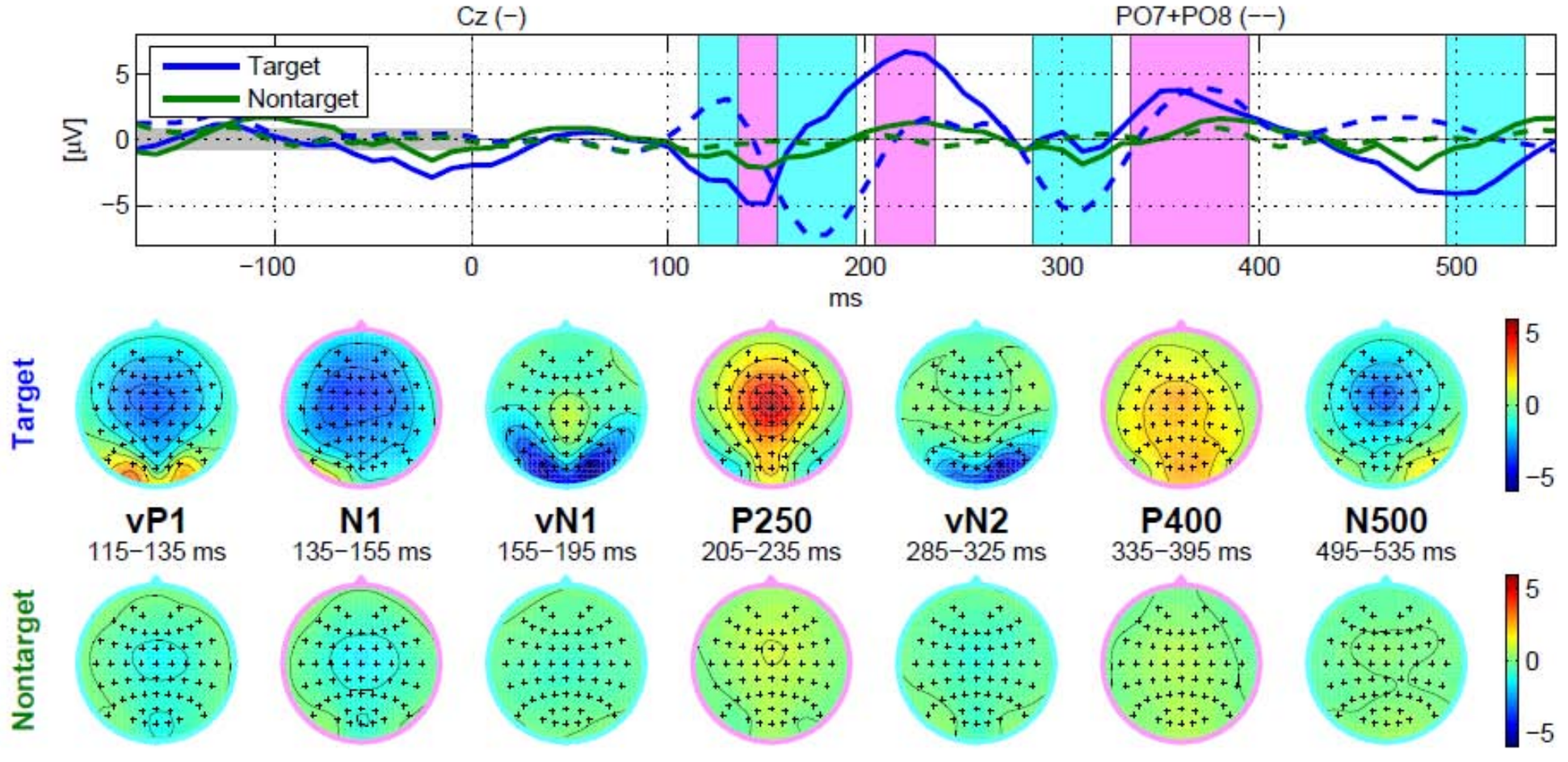
Single subject ERPs for Hex-o-spell

Data set for illustration of classification methods:



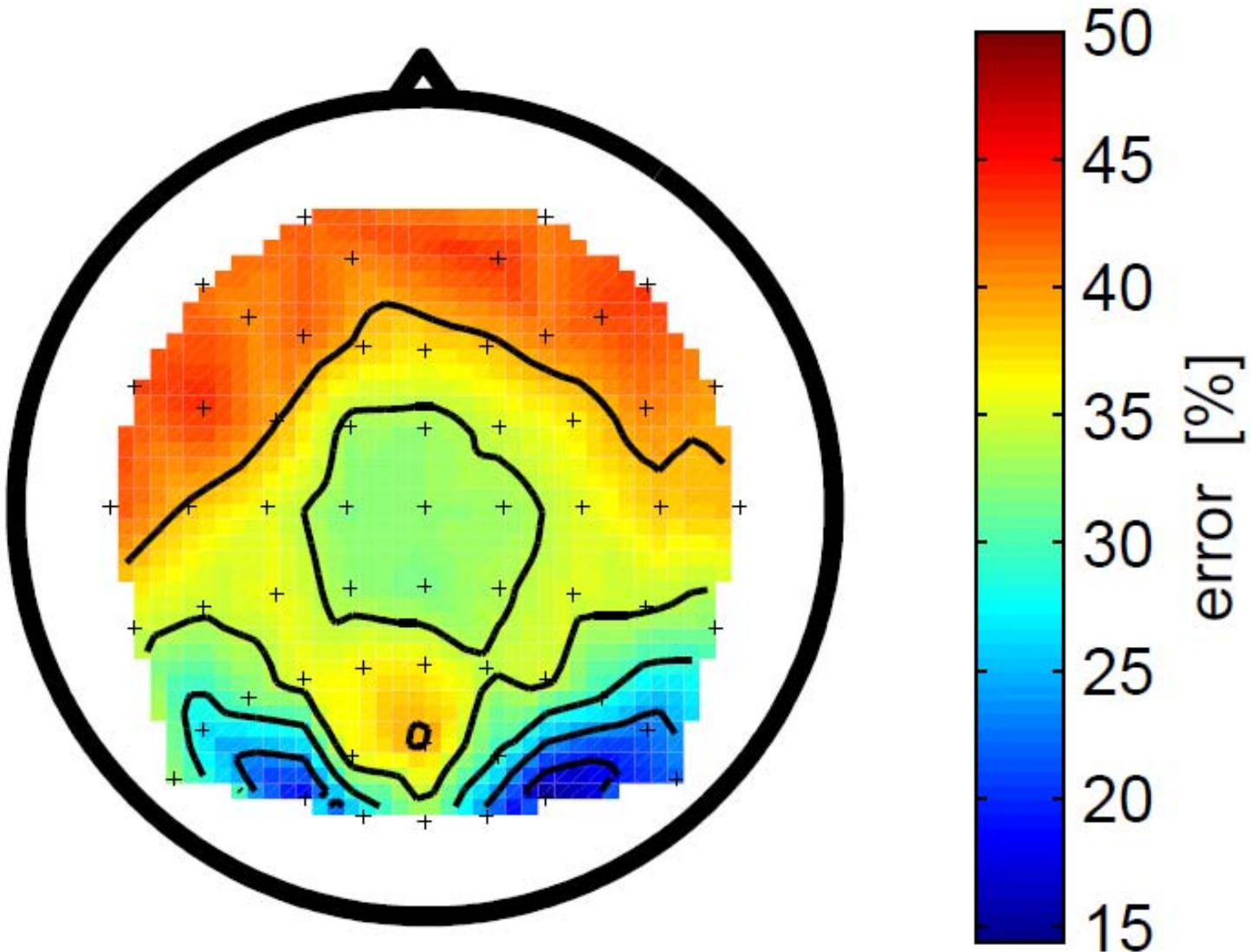
Topographies of ERP components

There are several ERP components that can be used to determine the attended symbol:

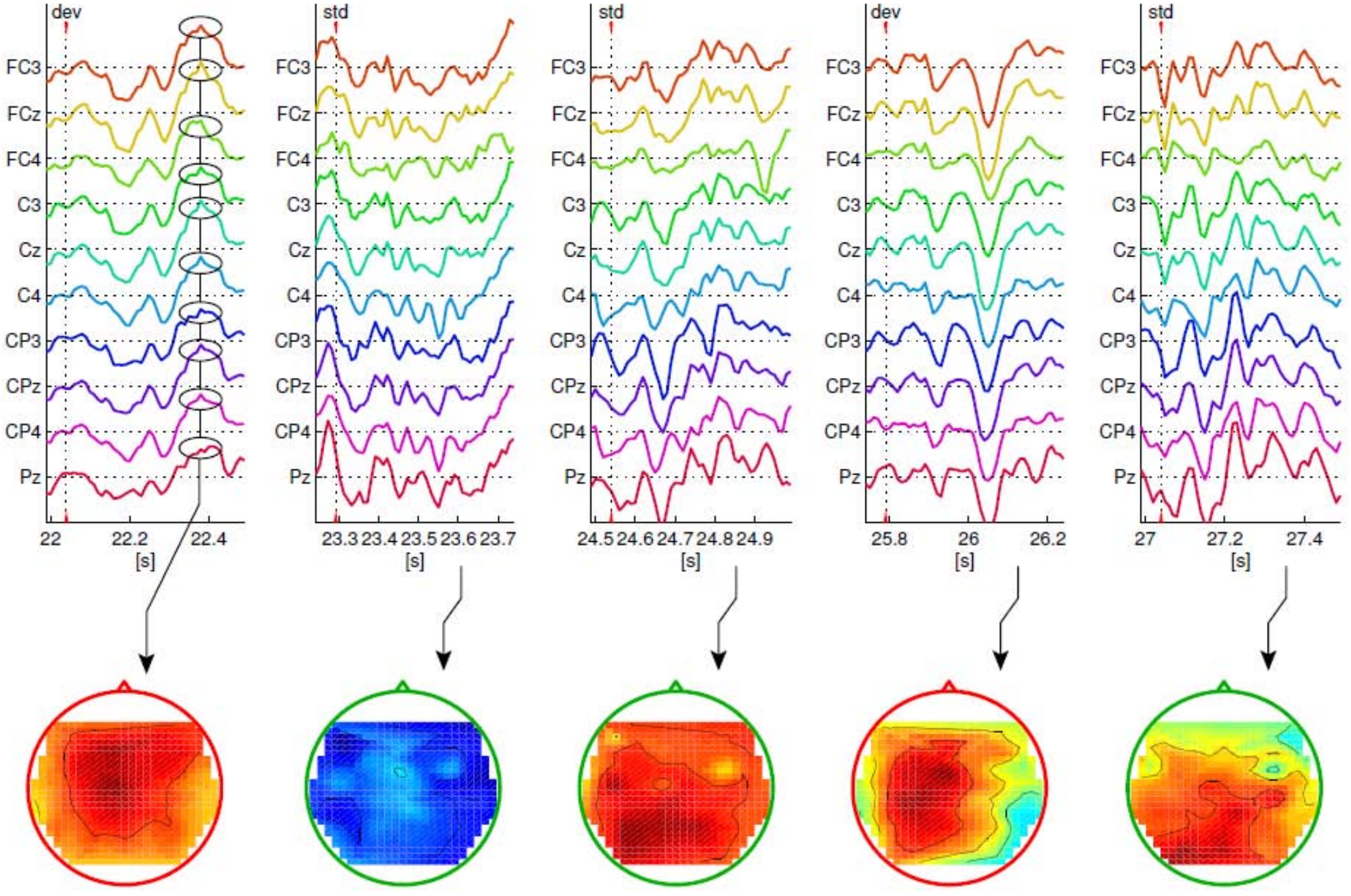


Classification of temporal features

As a first step: classification on raw time courses (115–535 ms) in single channels. The result is displayed as scalp map:

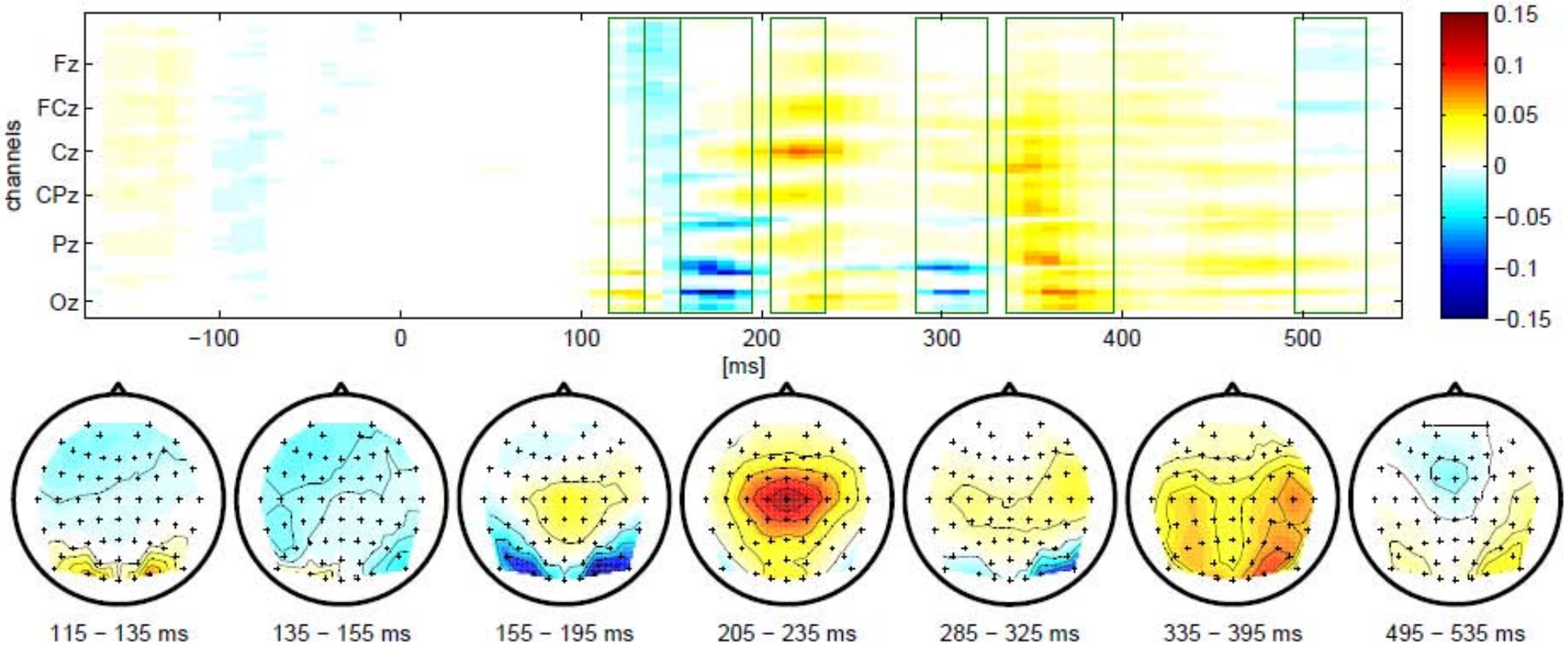


Extraction of spatial features



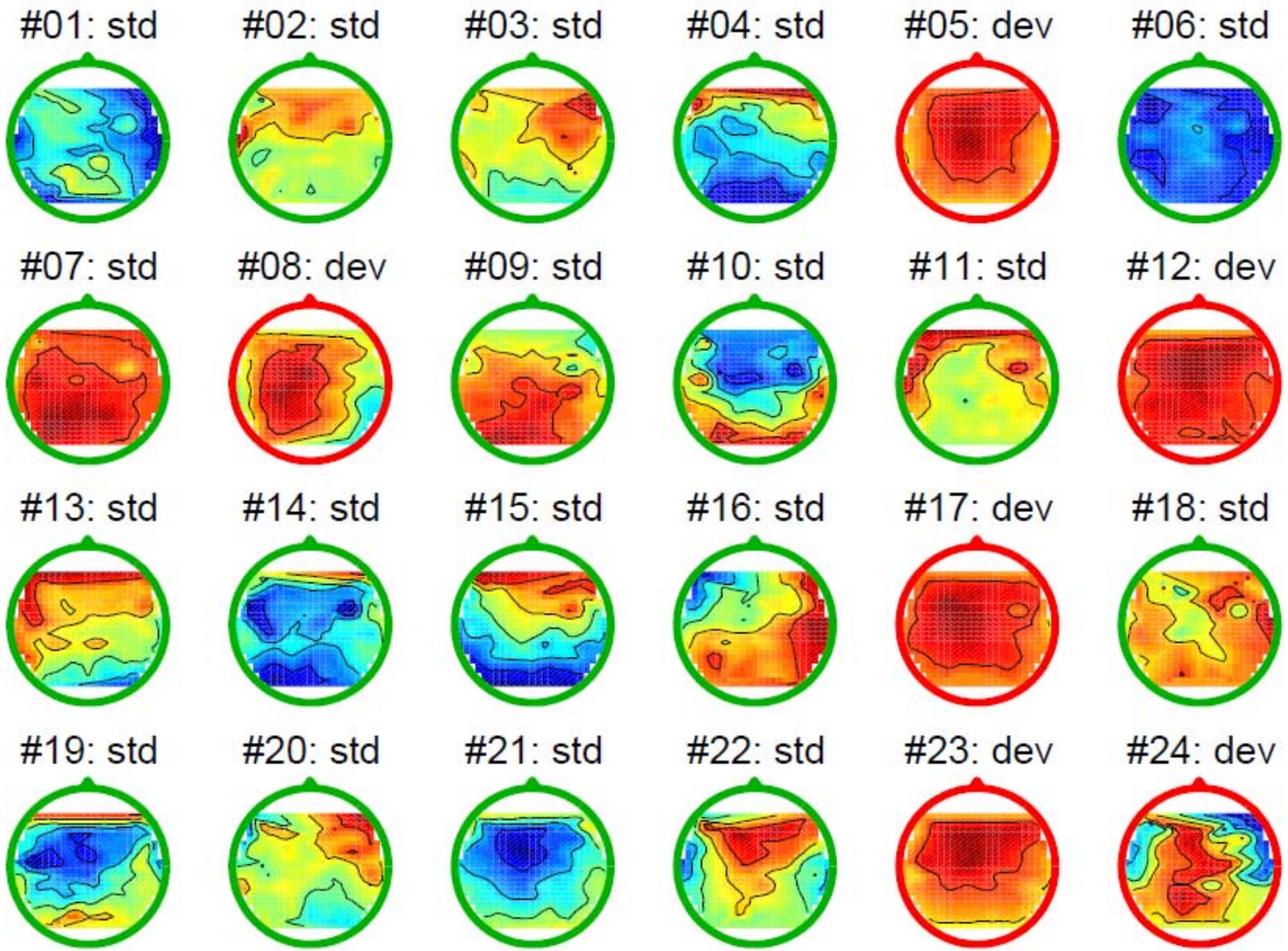
The r^2 matrix of differences

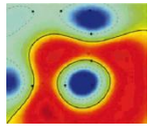
The temporal and spatial structure of the difference between ERPs of different conditions can be investigated by the signed r^2-matrix:



$$r(x) := \frac{\sqrt{N_1 \cdot N_2}}{N_1 + N_2} \frac{\text{mean}\{x_i \,|\, y_i = 1\} - \text{mean}\{x_i \,|\, y_i = 2\}}{\text{std}\{x_i\}}$$

Spatial features





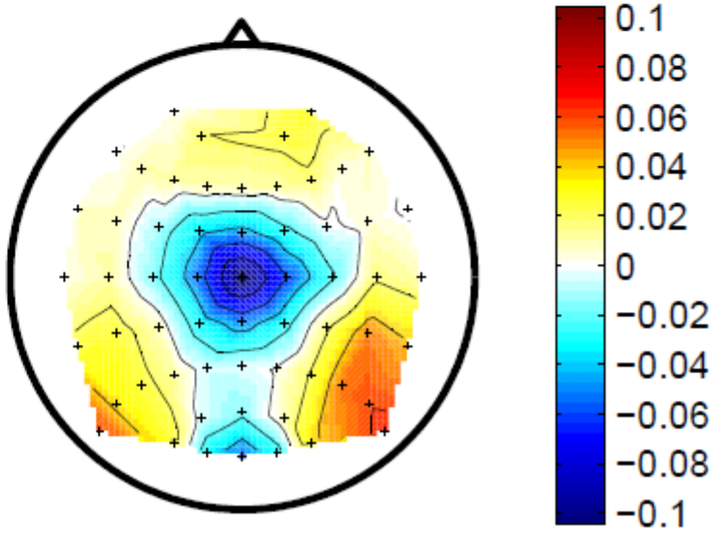
A linear classifier as a spatial filter

A linear classifier that was trained on *spatial features* can also be regarded as a **spatial filter**.
Let $\mathbf{w}$ be the LDA weight vector and $\mathbf{X} \in \mathbb{R}^{\#\text{chans} \times \#\text{time points}}$ be continuous EEG signals. Then

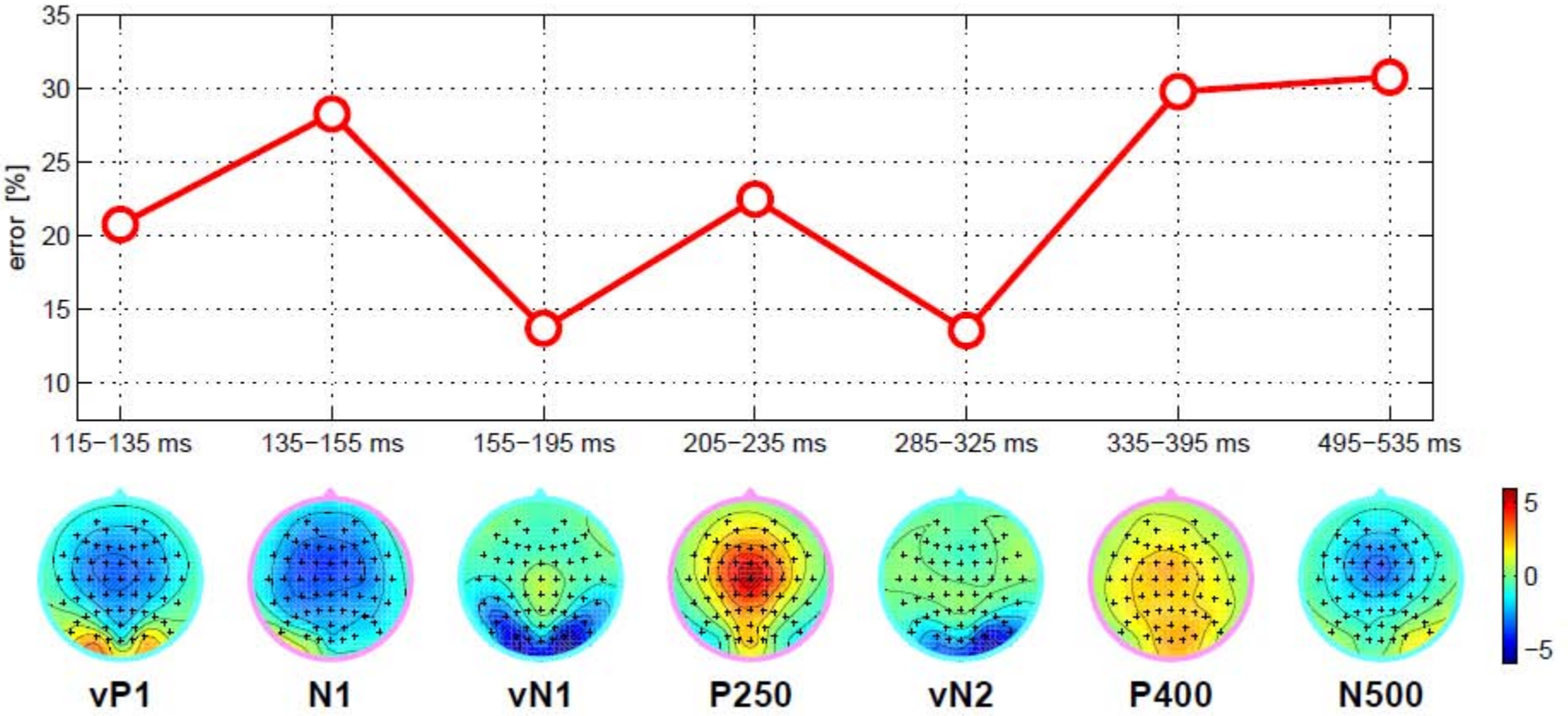
$$\mathbf{X}_f := \mathbf{w}^\top \mathbf{X} \quad \in \mathbb{R}^{1 \times \#\text{time points}}$$

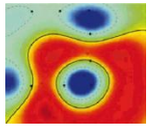
is the result of spatial filtering: each channel of $\mathbf{X}$ is weighted with the corresponding component of $\mathbf{w}$ and summed up.

The weight vector of the classifier can be display as scalp map:

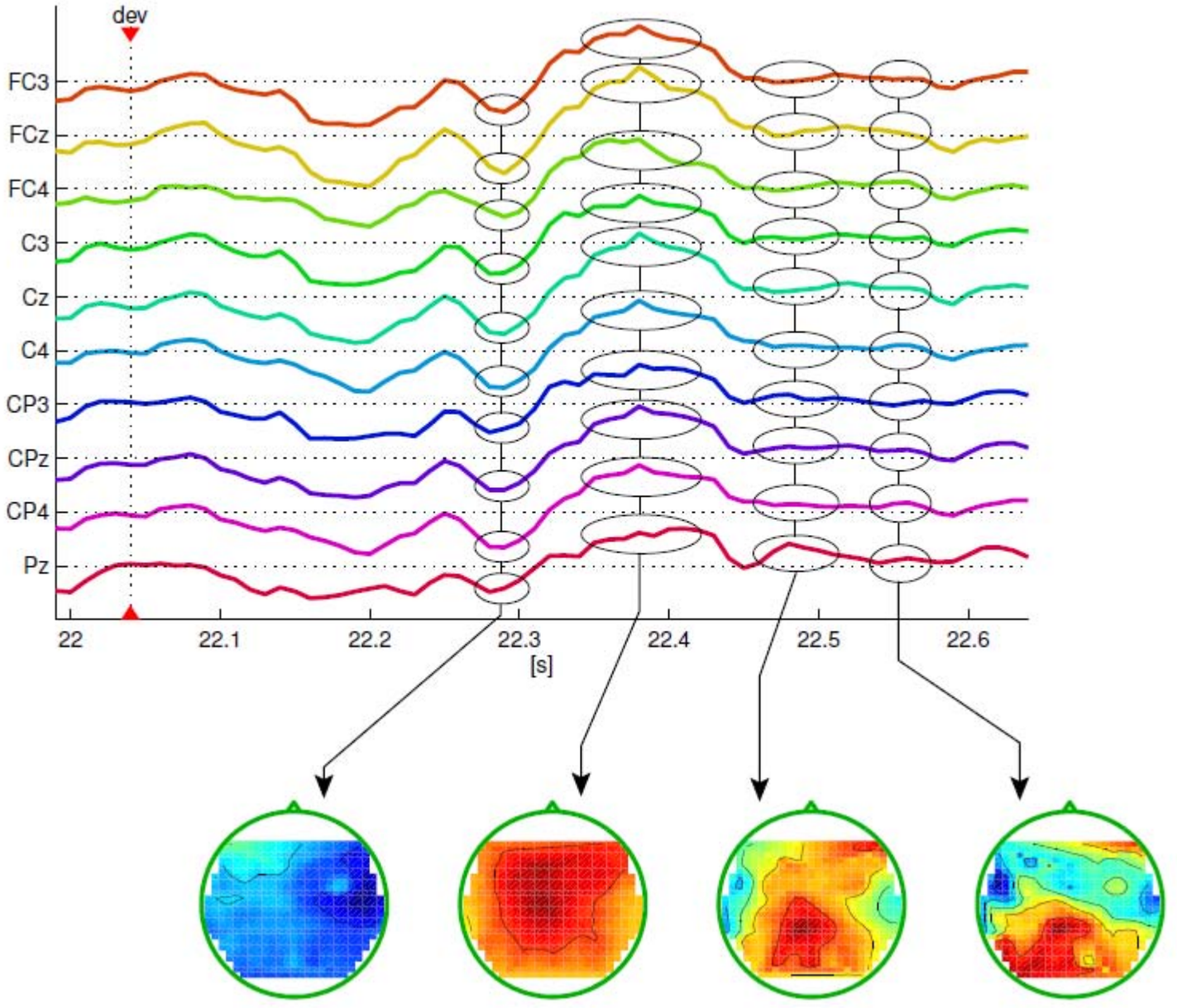


Classification results of spatial features



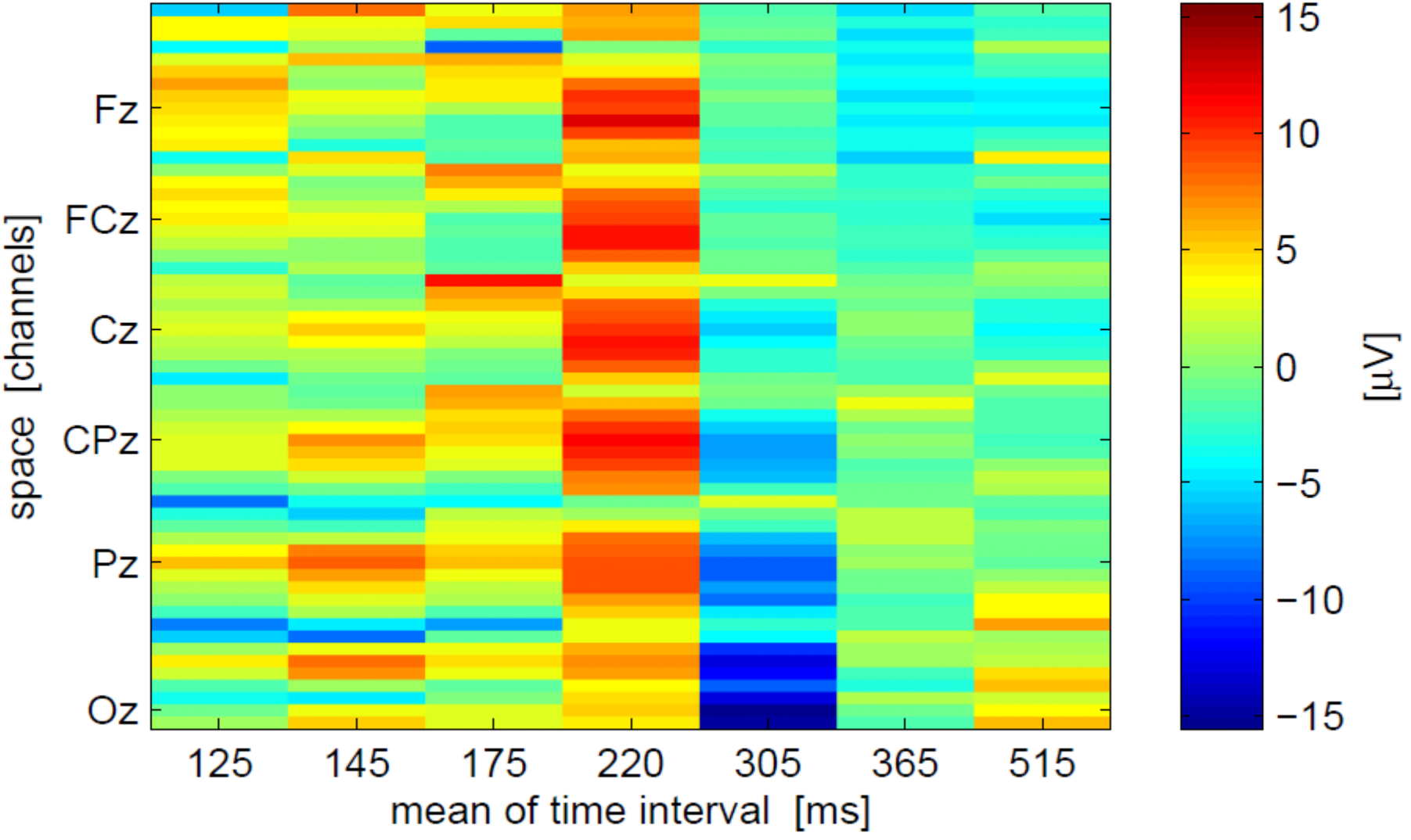


Extraction of spatio-temporal features

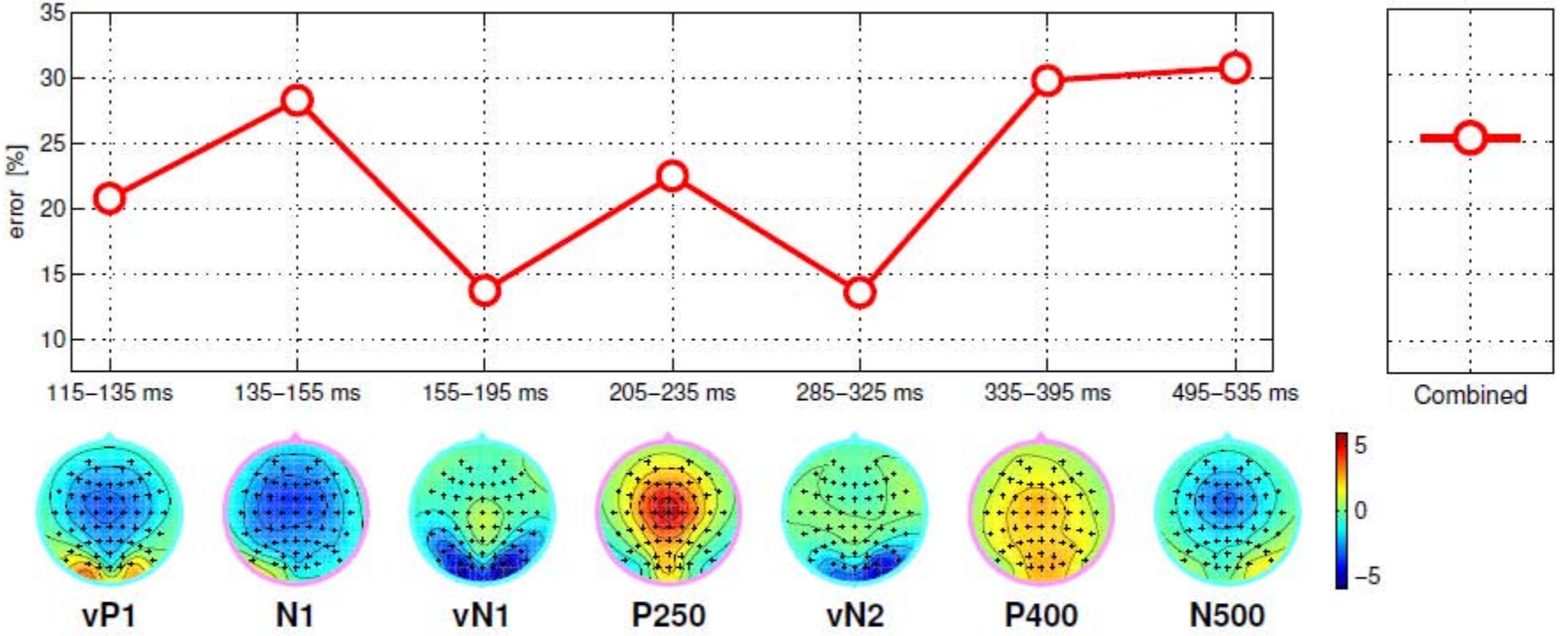


Spatio-temporal features

Spatio-temporal features are typically high-dimensional (here 59 EEG channels $\times$ 7 time intervals $=$ 413 dimensional features):



Classification results for spatio-temporal features



Although information was added, classification on the concatenated feature becomes worse: *overfitting*.

Bias in estimating covariances

Let $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$ be n vectors drawn from a d-dimensional Gaussian distribution $\mathcal{N}(\mu, \mathbf{\Sigma})$.
For classification μ and $\mathbf{\Sigma}$ have to be estimated from the data:

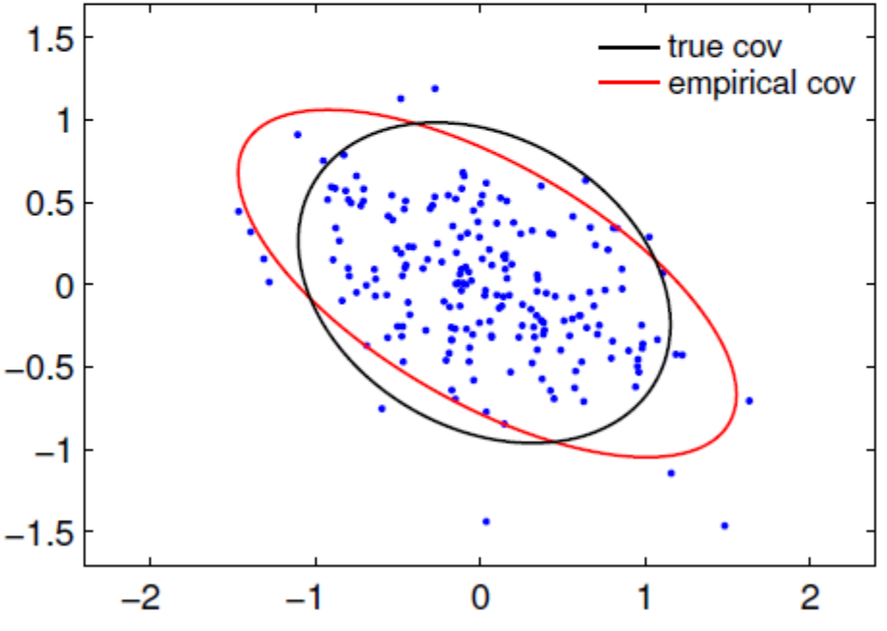
- $\hat{\mu} = \frac{1}{n} \sum_{k=1}^{n} \mathbf{x}_k$
- $\hat{\mathbf{\Sigma}} = \frac{1}{n-1} \sum_{k=1}^{n} (\mathbf{x}_k - \hat{\mu})(\mathbf{x}_k - \hat{\mu})^\top$

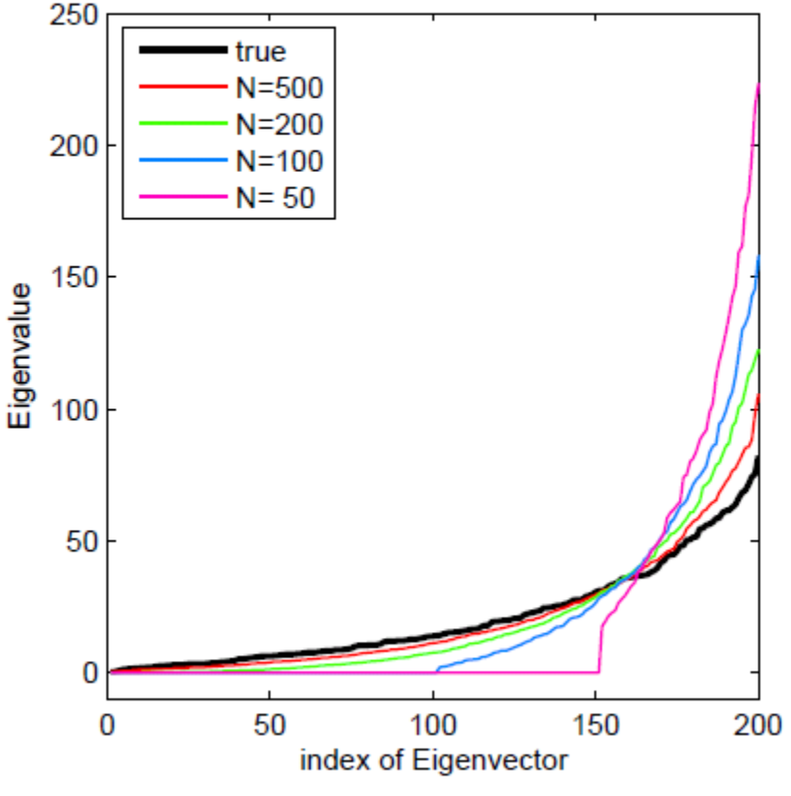
But, if the number of samples n is not large relative to the dimension d, the estimation is error-prone.
There is a systematical bias:

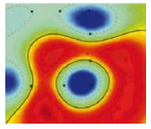
- Large Eigenvalues of $\hat{\mathbf{\Sigma}}$ are too large
- Small Eigenvalues of $\hat{\mathbf{\Sigma}}$ are too small

This affects, e.g., classification with LDA:
Normal vector of LDA: $w = \hat{\mathbf{\Sigma}}^{-1}(\mu_1 - \mu_2)$.

Bias in estimating covariances II







A remedy for classification

A simple way that can partly fix the bias is **shrinkage**: the empirical covariance matrix is modified to be more spherical. In LDA the empirical covariance matrix $\hat{\mathbf{\Sigma}}$ is replaced by

$$\tilde{\mathbf{\Sigma}}(\gamma) = (1-\gamma)\hat{\mathbf{\Sigma}} + \gamma\nu\mathbf{I}$$

for a $\gamma \in [0, 1]$ and ν defined as average Eigenvalue $\mathrm{trace}(\mathbf{S}_i)/d$. Since $\hat{\mathbf{\Sigma}}$ is positive semi-definite we can have an Eigenvalue decomposition $\hat{\mathbf{\Sigma}} = \mathbf{V}\mathbf{D}\mathbf{V}^\top$ with orthonormal $\mathbf{V}$ and diagonal $\mathbf{D}$. From

$$\tilde{\mathbf{\Sigma}} = (1-\gamma)\mathbf{V}\mathbf{D}\mathbf{V}^\top + \gamma\nu\mathbf{I} = \mathbf{V}\left((1-\gamma)\mathbf{D} + \gamma\nu\mathbf{I}\right)\mathbf{V}^\top$$

we see that

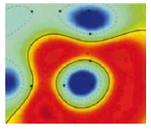
- $\tilde{\mathbf{\Sigma}}(\gamma)$ and $\hat{\mathbf{\Sigma}}$ have the same Eigenvectors (columns of $\mathbf{V}$)
- extreme Eigenvalues (large/small) are shrunk/extended towards the average ν.
- $\gamma = 0$ yields LDA without shrinkage, $\gamma = 1$ assumes spherical covariance matrices.

Modelselection

LDA with shrinkage of the empirical covariance matrix has one free parameter (γ), also called hyperparameter, that needs to be selected. There is no general way to do it.
Numerous strategies with different properties exist, e.g.

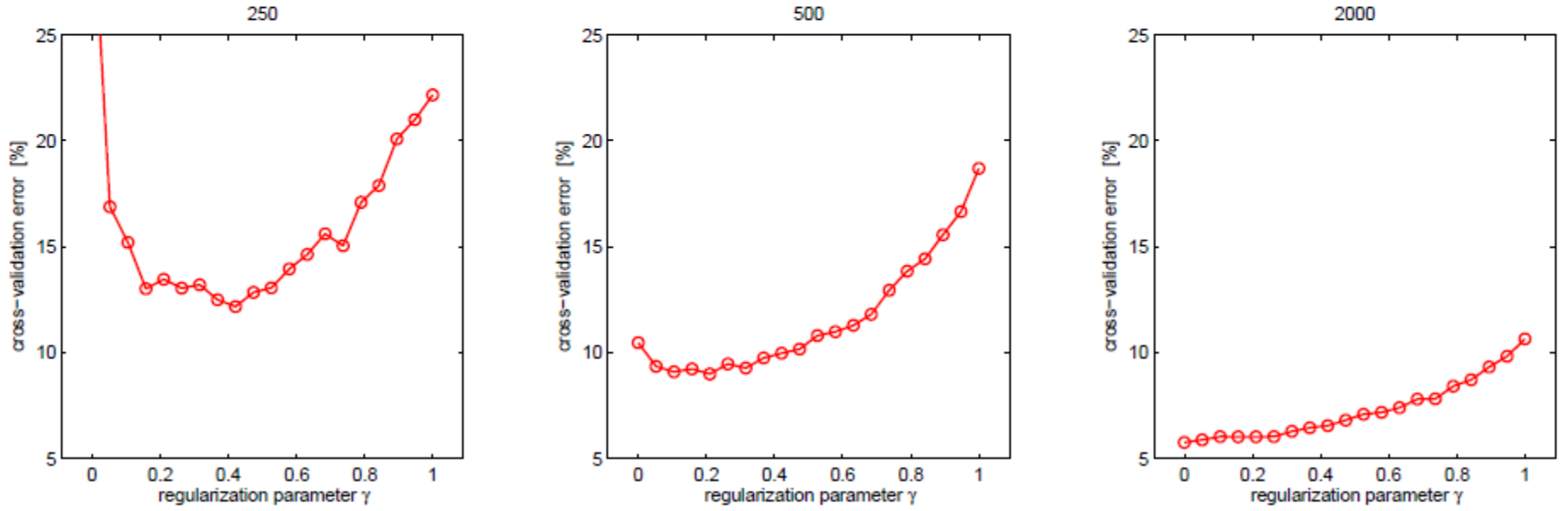
- empirical Bayes shrinkage estimator
- MDL: Minimum Description Length
- Model-selection based on cross-validation.
- ...

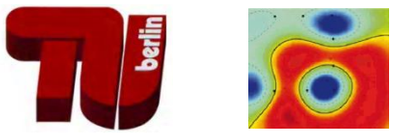
An easy (and also time-consuming) way is model-selection based on **cross-validation**.



Regularized LDA at work

Cross-validation results for different sizes of training data (250, 500, 2000) for different values of the regularization parameter γ (x-axis). Features vectors have 250 dimensions.



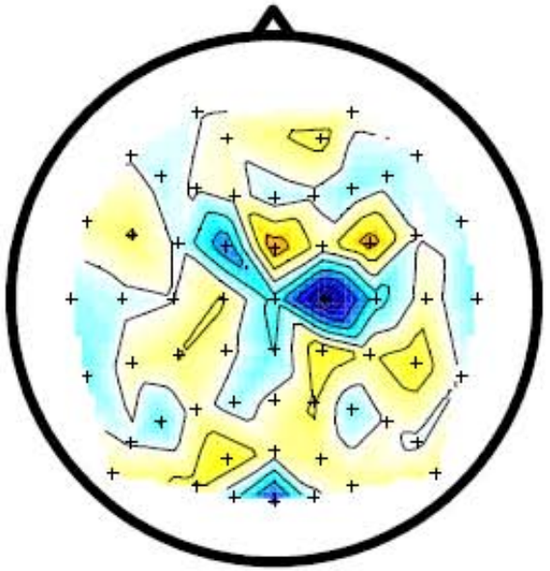


Investigating the impact of shrinkage

LDA: $w = \hat{\Sigma}^{-1}(\mu_1 - \mu_2)$; **shrinkage:** $\tilde{\Sigma}(\gamma) = (1-\gamma)\hat{\Sigma} + \gamma\nu\mathbf{I}$

$\gamma = 0$

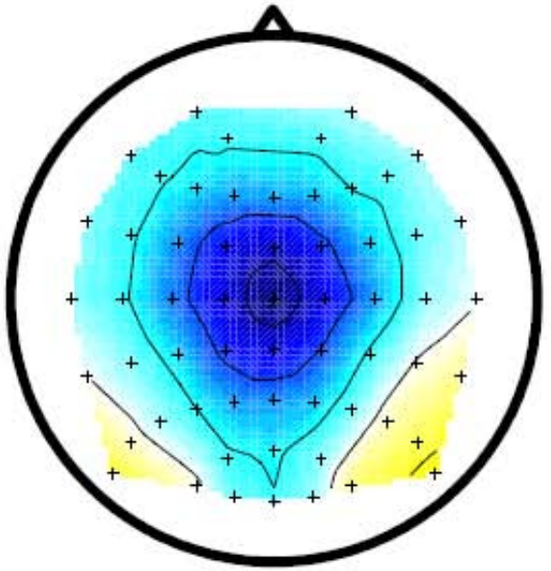
$w \sim \hat{\Sigma}^{-1}(\mu_1 - \mu_2)$



accounting for spatial structure of the noise

$\gamma = 1$

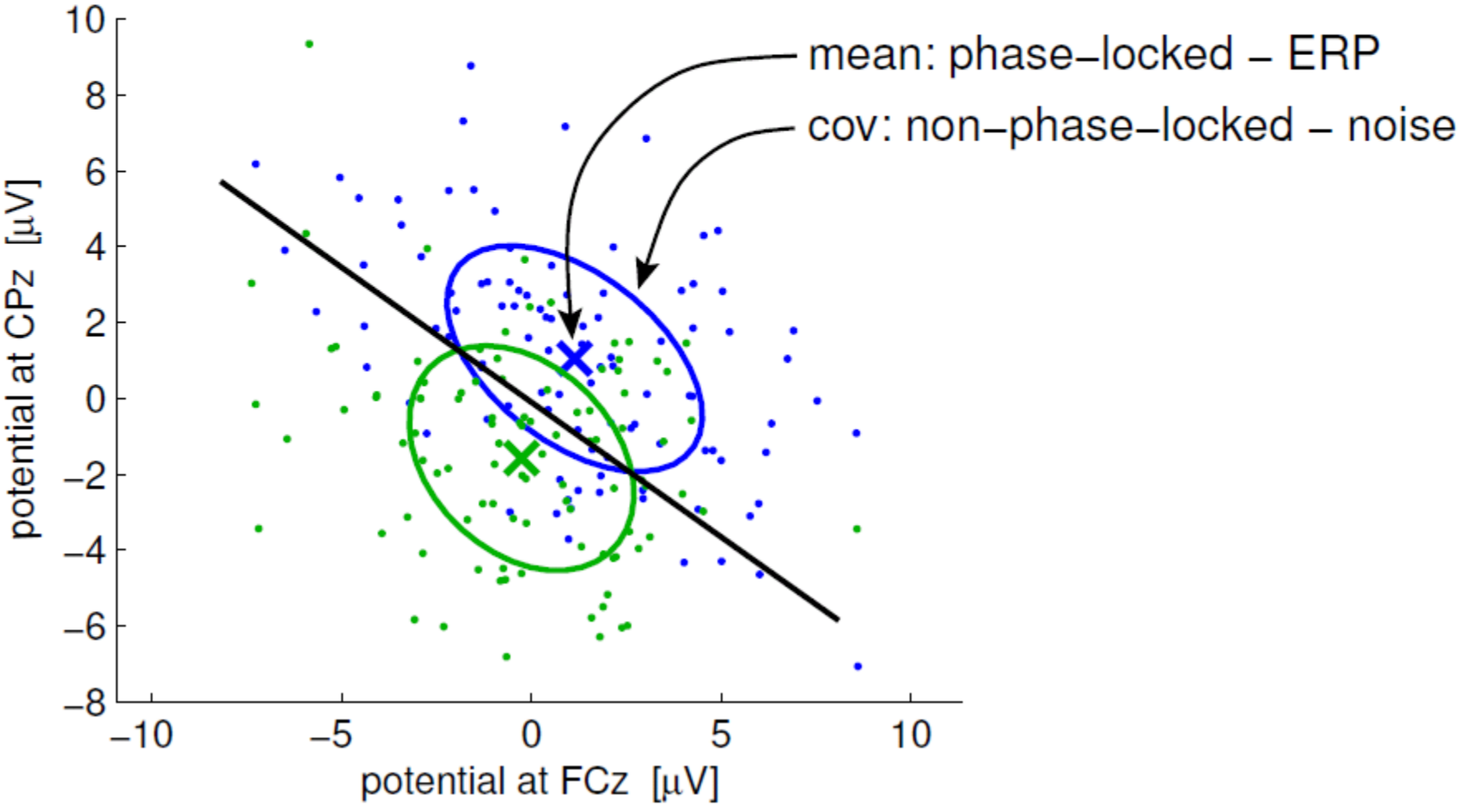
$w \sim \mu_1 - \mu_2$



ERP and noise

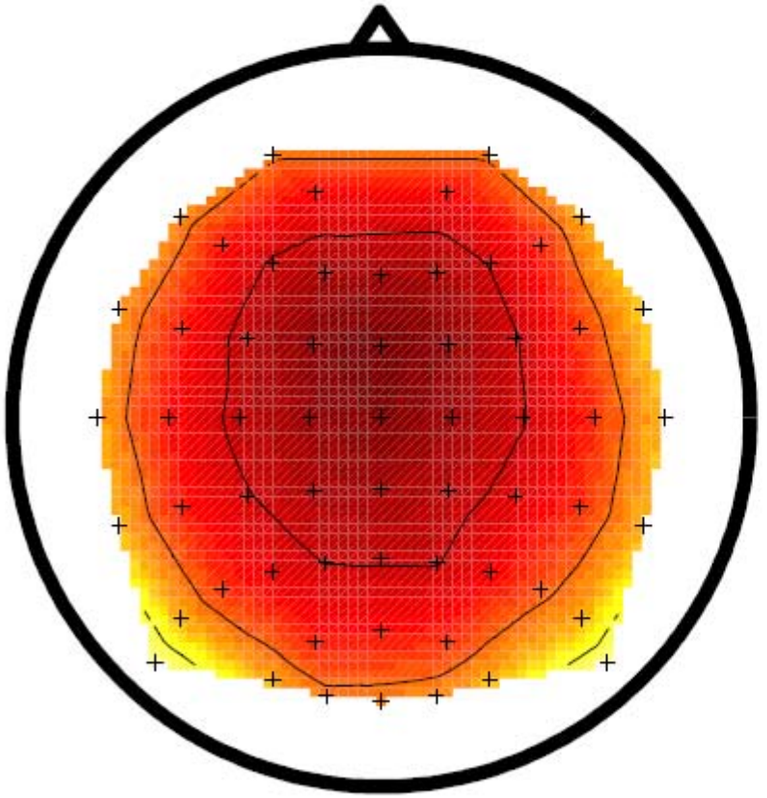
Simple assumption for ERPs: single trial $x_k(t)$ is composed of an ERP $s(t)$ and Gaussian 'noise' $\mathbf{n}_k(t)$:

$$\mathbf{x}_k(t) = \mathbf{s}(t) + \mathbf{n}_k(t) \qquad \text{for all trials } k = 1, \ldots, K$$

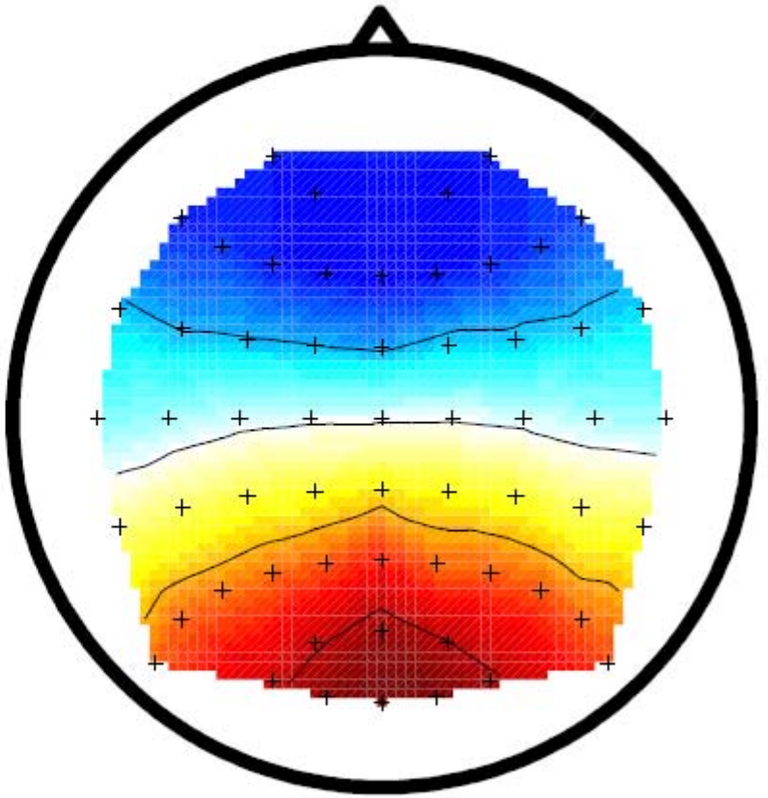


Spatial structure of noise

The two strongest principal components of the noise (covariance matrix) in this data set:

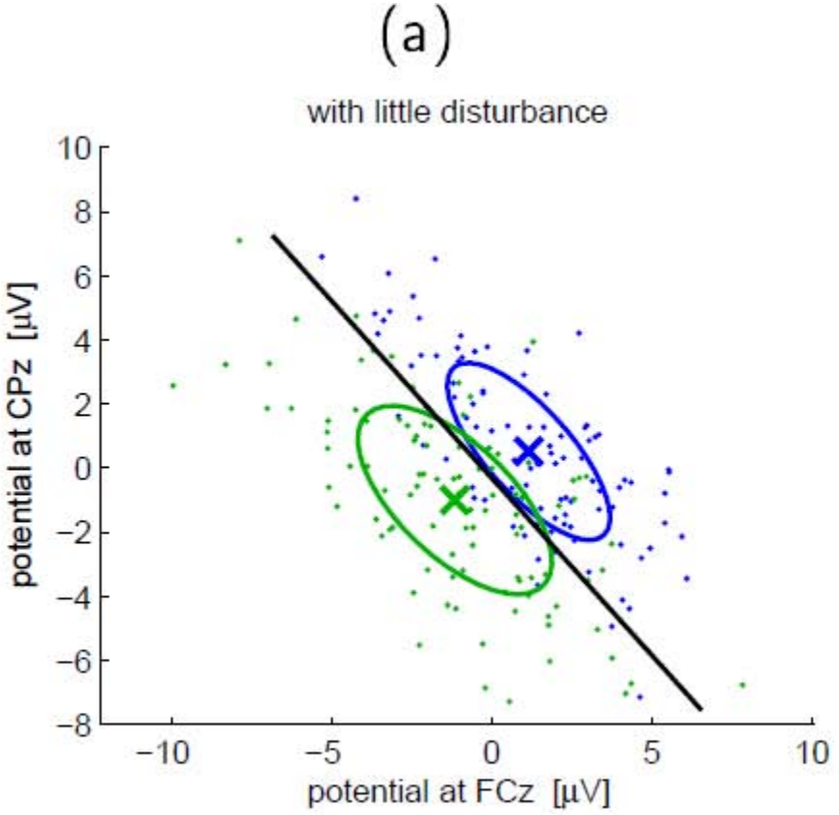


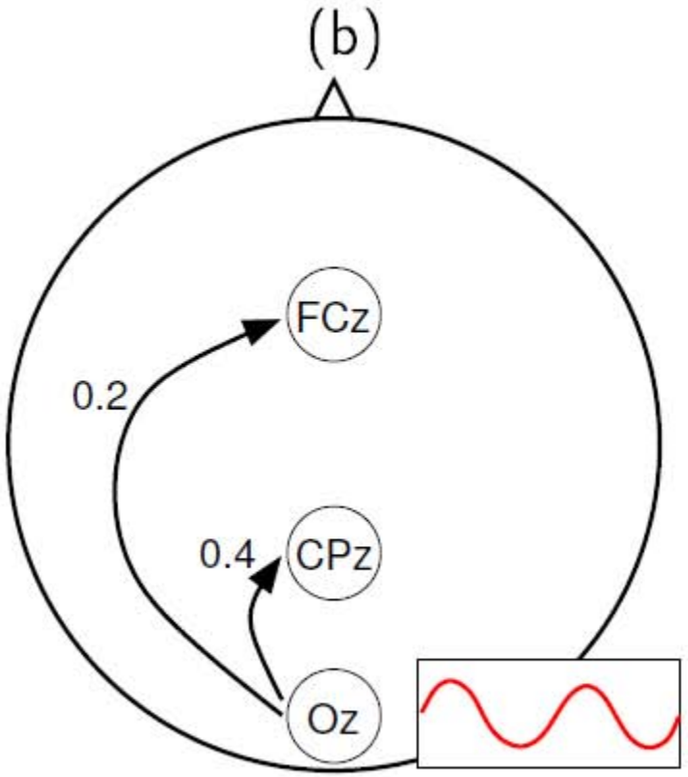
Trial-to-trial variation of P3

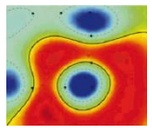


Visual alpha

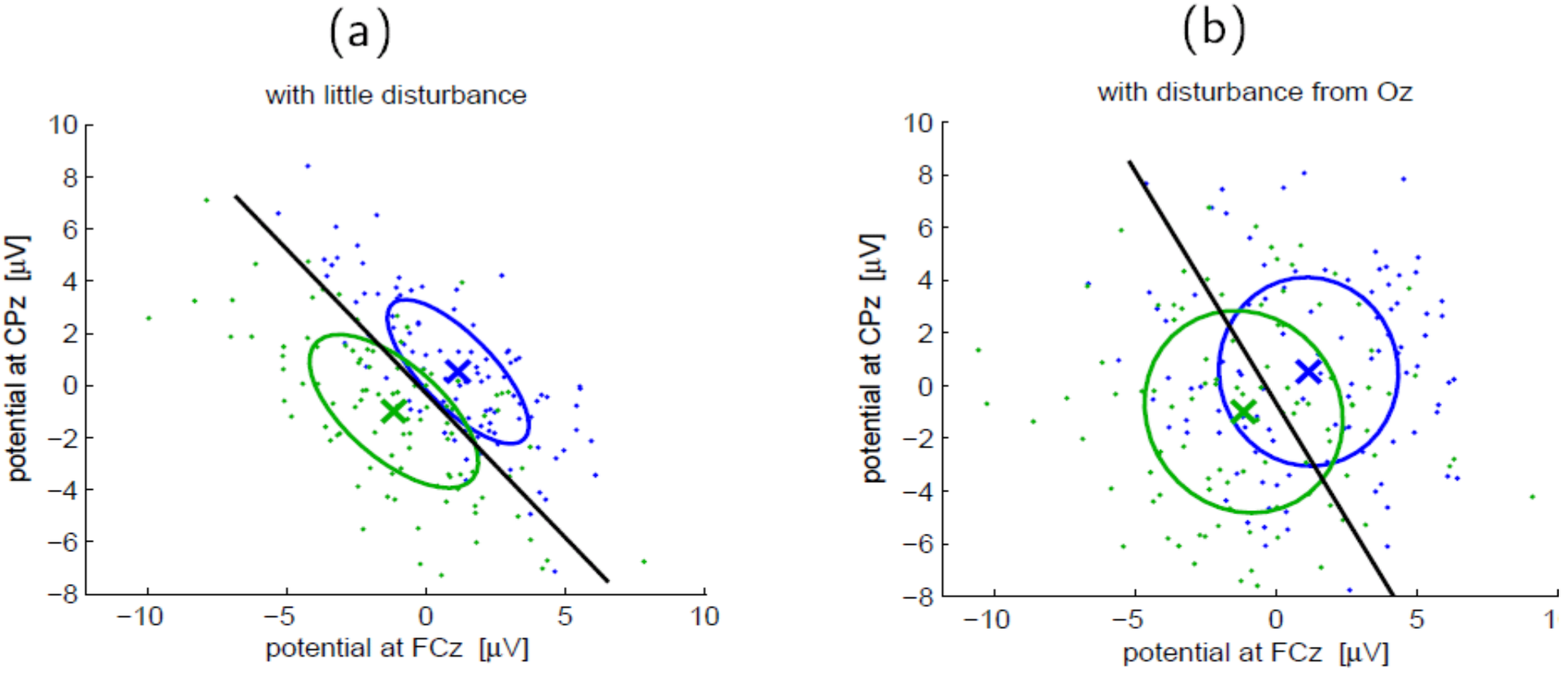
Understanding spatial filters







Understanding spatial filters II

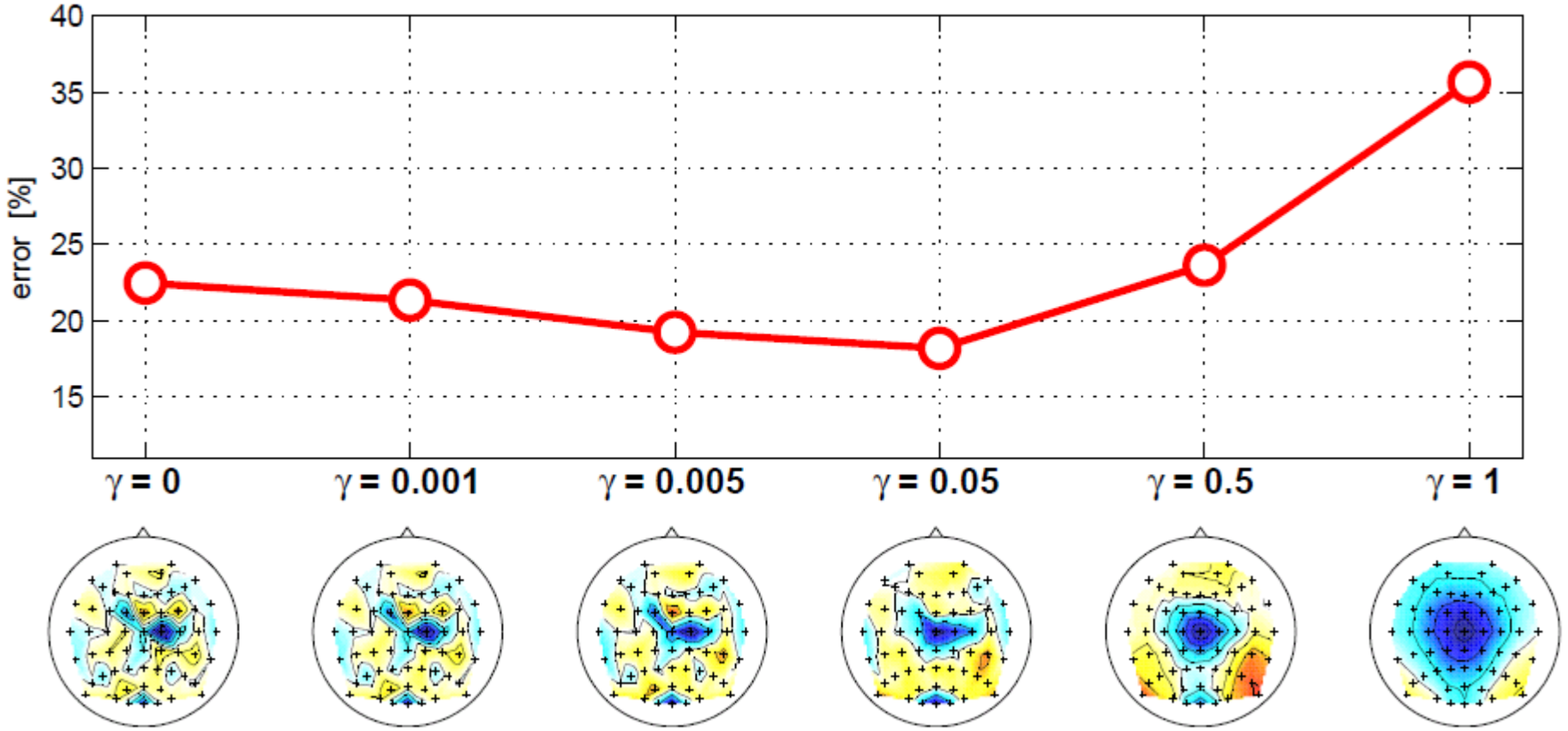


Two channel classification of (a): 15% error, (b): 37% error

When disturbing channel Oz is added to the data (3D): 16% error. Here, channel Oz is required for good classification although itself is not discriminative.

Impact of shrinkage on the spatial filters

With increasing shrinkage, the spatial filters (classifier) look smoother, but classification may degrade with too much shrinkage.



Maps of spatial filters for different values of γ.

Optimal selection of shrinkage parameters

Let $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$ be n feature vectors and let $\hat{\mu} = \frac{1}{n} \sum_{k=1}^{n} \mathbf{x}_k$ be the empirical mean.

Aim: get a better estimate of the true covariance matrix $\mathbf{\Sigma}$ (especially in case $n < d$) than the sample covariance matrix $\hat{\mathbf{\Sigma}} = \frac{1}{n-1} \sum_{k=1}^{n} (\mathbf{x}_k - \hat{\mu})(\mathbf{x}_k - \hat{\mu})^\top$ by selecting a γ in

$$\tilde{\mathbf{\Sigma}}(\gamma) := (1 - \gamma)\hat{\mathbf{\Sigma}} + \gamma \nu \mathbf{I}.$$

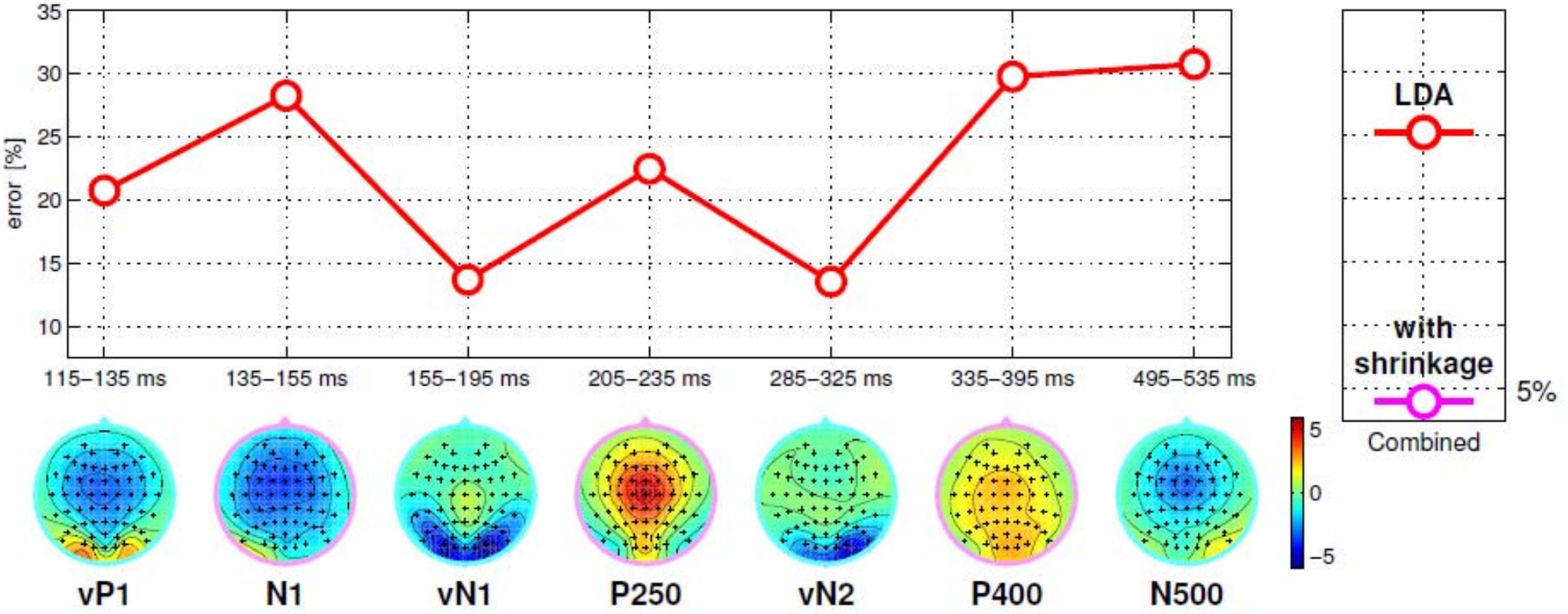
We denote by $(\mathbf{x}_k)_i$ resp. $(\hat{\mu})_i$ the i-th element of the vector $\mathbf{x}_k$ resp. $\hat{\mu}$. Furthermore we denote by s_{ij} the element in the i-th row and j-th column of $\hat{\mathbf{\Sigma}}$. We define

$$z_{ij}(k) = ((\mathbf{x}_k)_i - (\hat{\mu})_i)\ ((\mathbf{x}_k)_j - (\hat{\mu})_j)$$

Then the optimal shrinkage parameter $\gamma^\star$ for which $\tilde{\mathbf{\Sigma}}(\gamma^\star) = \operatorname{argmin}_{\mathbf{S}} \|\mathbf{S} - \mathbf{\Sigma}\|_F^2$ can be analytically calculated ([2]) as

$$\gamma^\star = \frac{n}{(n-1)^2} \frac{\sum_{i,j=1}^{d} \operatorname{var}_k(z_{ij}(k))}{\sum_{i \neq j} s_{ij}^2 + \sum_i (s_{ii} - \nu)^2}$$

Result of Classification with shrinkage

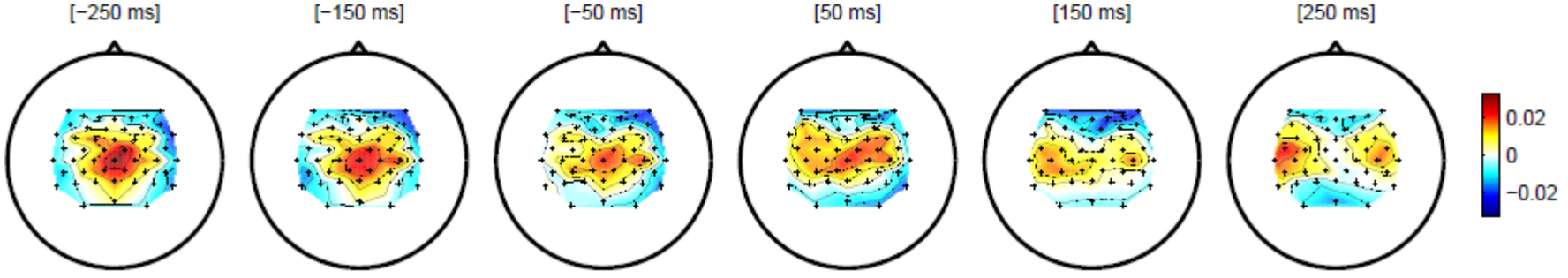


Using shrinkage the classification error could be drastically reduced to 4%.

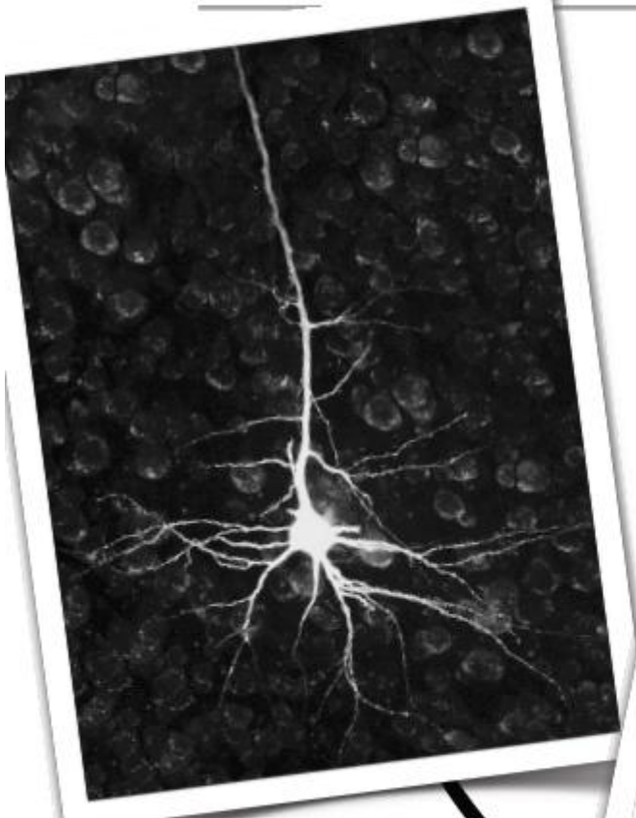
Summary spatio-temporal classification

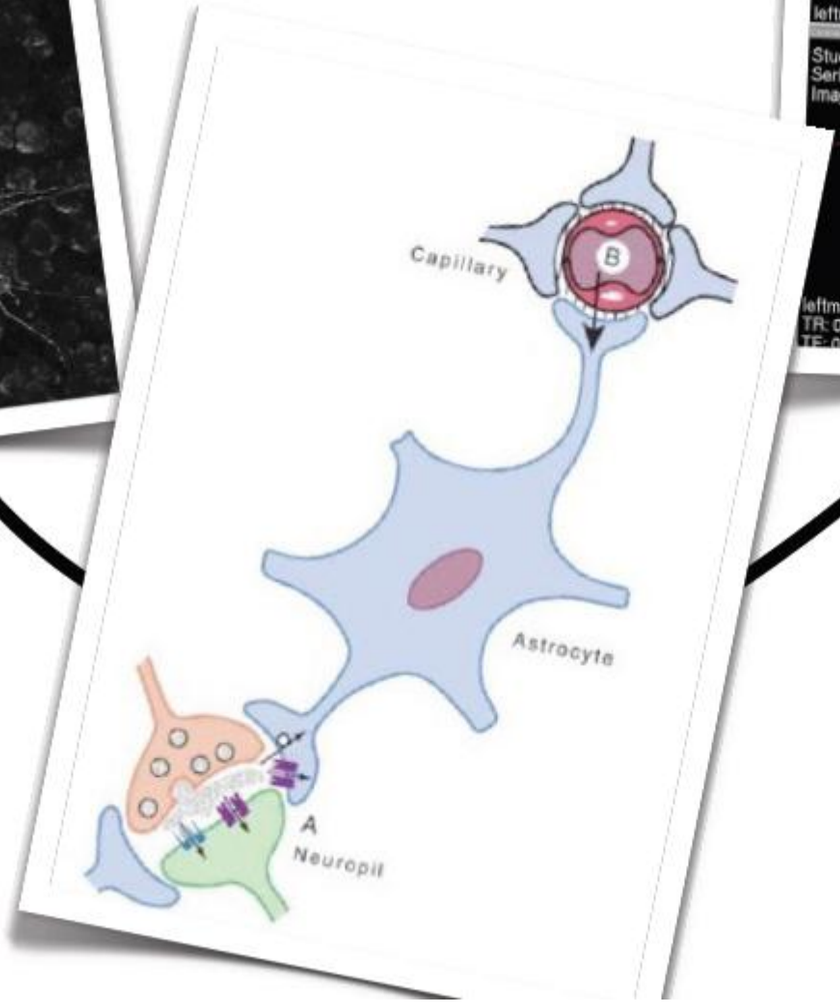
- Linear classification with shrinkage is a powerful method.
- Complete shrinkage ($\gamma = 1$) means neglecting the structure of the noise. In this case the classifier is the difference of the ERPs.
- The appropriateness of a linear separation depends on the way features are extracted and transformed.
- In contrast to non-linear classifiers, the weights of a linear classifier are informative.

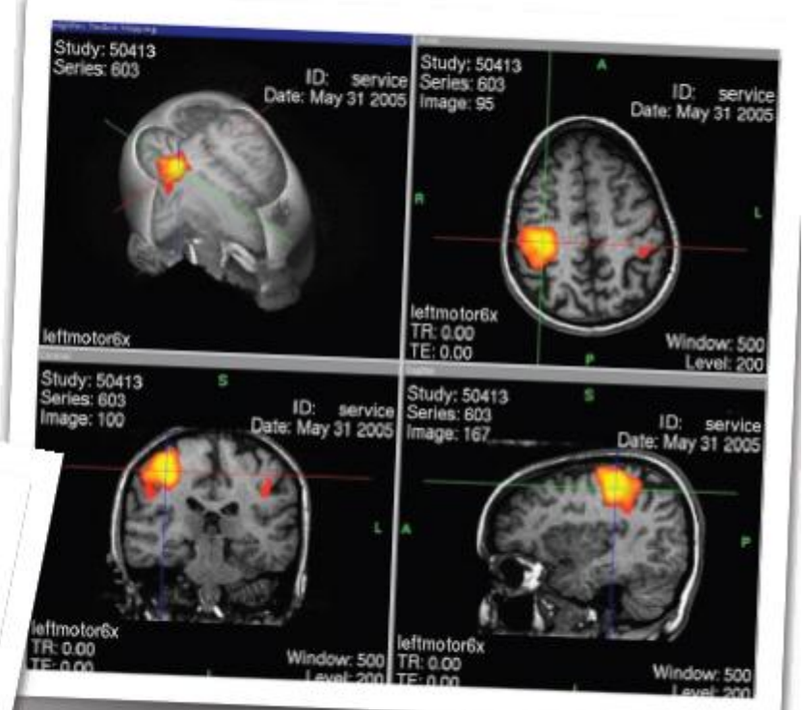
The weights of the trained classifier can be visualized as a sequence of scalp topographies:



Application: Neuro-Vascular Coupling







CCA: correlating apples and oranges

Given two (or more) multivariate variables

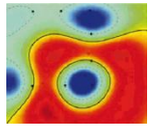
$$X \in \mathbb{R}^M, Y \in \mathbb{R}^N$$

CCA finds projections

$$w_x \in \mathbb{R}^M, w_y \in \mathbb{R}^N$$

that maximise the covariance between the variables

$$\underset{u}{\text{ar}} \begin{bmatrix} 0 & C_{xy} \\ C_{yx} & 0 \end{bmatrix} \begin{bmatrix} w_x \\ w_y \end{bmatrix} = \alpha \begin{bmatrix} C_{xx} & 0 \\ 0 & C_{yy} \end{bmatrix} \begin{bmatrix} w_x \\ w_y \end{bmatrix}$$



kCCA: solving CCA on data kernels

Intuition behind the Kernel Trick:

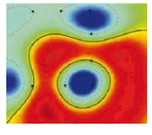
The solution of CCA in kernel space is obtained by solving the generalised eigenvalue problem

$$\begin{bmatrix} 0 & K_x K_y \\ K_y K_x & 0 \end{bmatrix} \begin{bmatrix} \alpha_x \\ \alpha_y \end{bmatrix} = \rho \begin{bmatrix} K_x^2 & 0 \\ 0 & K_y^2 \end{bmatrix} \begin{bmatrix} \alpha_x \\ \alpha_y \end{bmatrix}$$

The solutions in the input space can be recovered by

$$w_x = X\alpha_x$$
$$w_y = Y\alpha_y$$

No need to compute big covariance matrices!

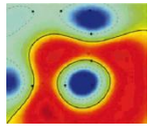


tkCCA: correlating apples and oranges over time

$$\underset{w_x(\tau), w_y}{\operatorname{argmax}} \operatorname{Corr} \left(\sum_{\tau} w_x(\tau)^\top x(t-\tau),\ w_y^\top y(t) \right)$$

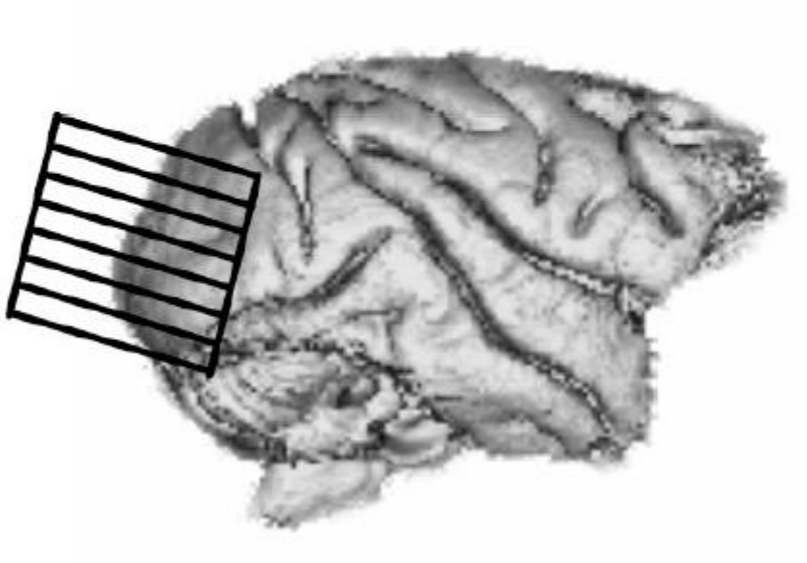
$$\tilde{X} = \begin{bmatrix} X_{\tau_1} \\ X_{\tau_2} \\ \vdots \\ X_{\tau_T} \end{bmatrix} \qquad \tilde{w}_x = \begin{bmatrix} w_x(\tau_1) \\ w_x(\tau_2) \\ \vdots \\ w_x(\tau_T) \end{bmatrix}$$

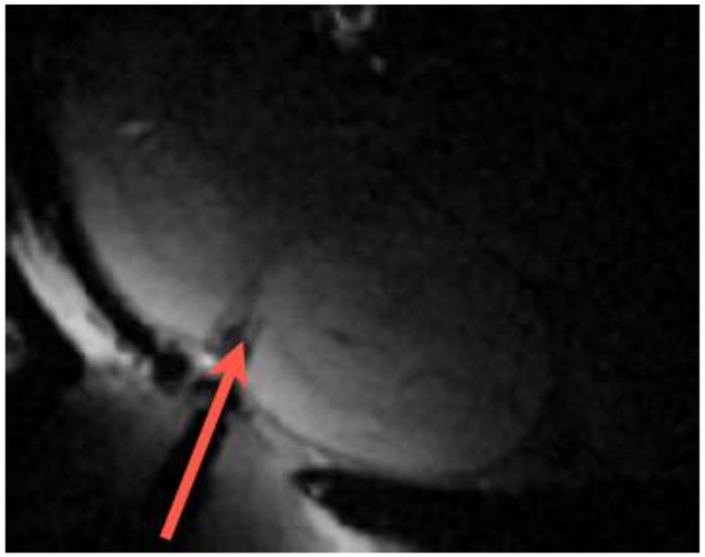
$$\underset{w_{\tilde{x}}, w_y}{\operatorname{argmax}} \operatorname{Corr} \left(\tilde{w}_x^\top \tilde{X}, w_y^\top Y \right)$$



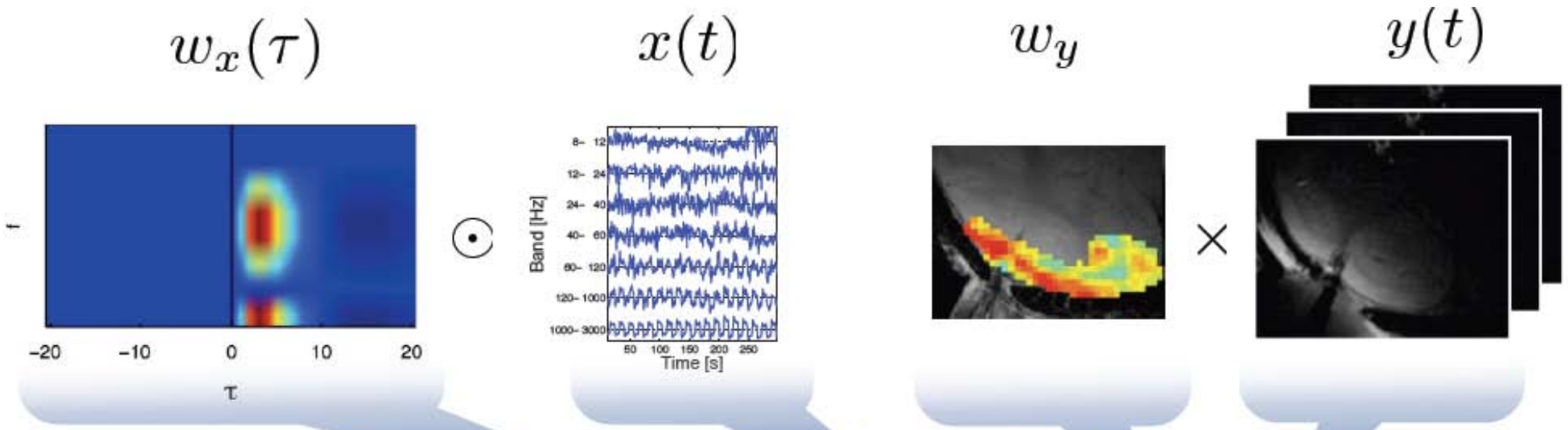
Experimental Setup

- **Simultaneous measurements of**
  - fMRI/ BOLD signal
  - Intracortical neural activity





Temporal Kernel CCA

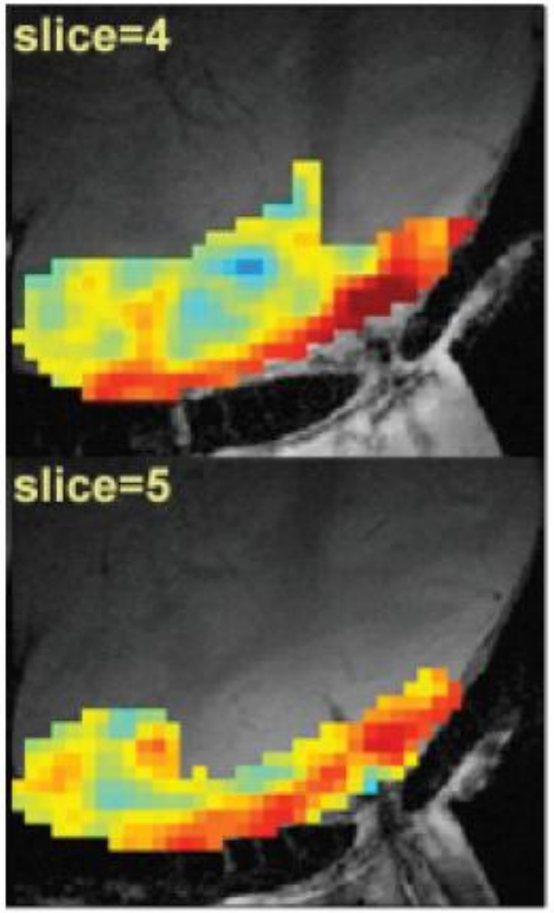


$$\underset{w_x(\tau), w_y}{\operatorname{argmax}} \operatorname{Corr}\left(\sum_{\tau} w_x(\tau)^\top x(t-\tau),\ w_y^\top y(t)\right)$$

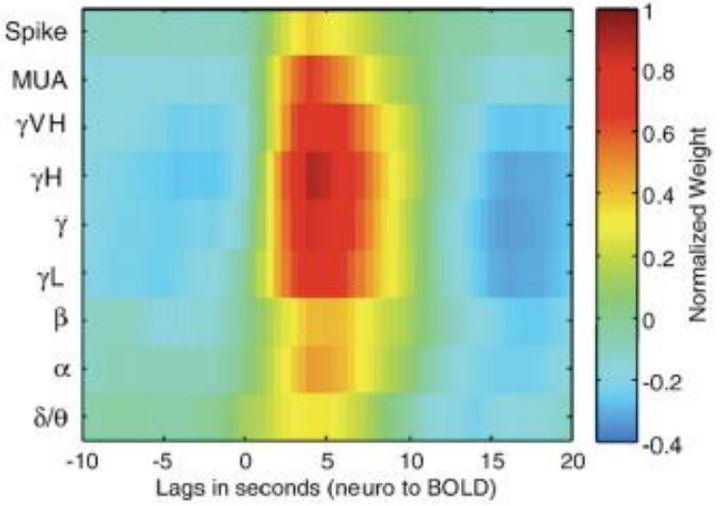
multivariate convolution of the neurophysiological signal with frequency dependent HRF

spatial weighting of voxels with activation pattern.

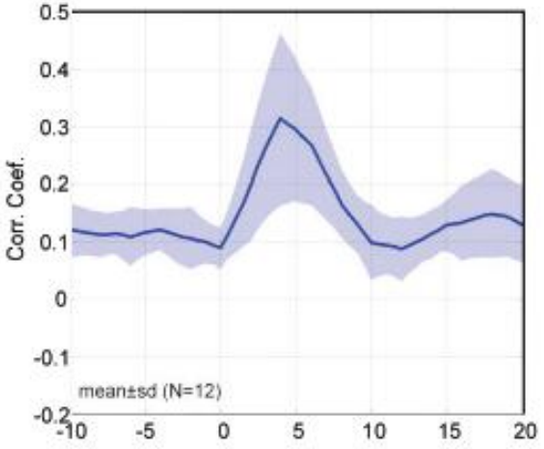
Results tkCCA: spatial dependencies and HRF



Spatial Dependencies



Haemodynamic Response Function



Canonical Correlogram

Murayama et al., "*Relationship between neural and haemodynamic signals during spontaneous activity studied with temporal kernel CCA*", Magnetic Resonance Imaging, 2010

Finale: Caveats in Validation

When machine learning techniques are used for classification of EEG single-trials, the expected performance of a method has to be evaluated carefully, and there are several possible pitfalls.

The estimation of generalization performance requires a training and a test set. The estimation is only proper

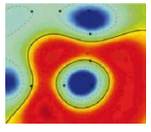
- if the test set was not used in any way to determine parameters of the method, and
- if the samples in the test set are independent from the samples in the training set.

Although these principles are quite obvious, it happens that they are violated.
Unfortunately, even some published journal articles lack a proper validation of the proposed methods.

Hall of pitfalls in single-trial EEG analysis (and beyond)

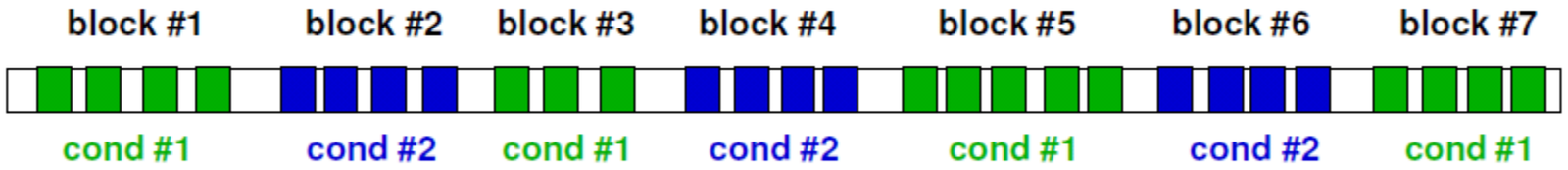
- preprocessing methods that use statistics of the whole data set like ICA, or normalization of features
  (particularly severe for methods that use label information)
- features are selected on the whole data set, including trials that are later in the test set
- select parameters by cross validation on the whole data set and report the performance for the selected values
- artifacts/outliers are rejected from the whole data set
  (resulting in a simplified test set)
- unsufficient validation for paradigms with block design

In this presentation we highlight the last issue.

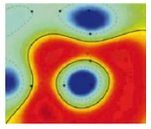


Block design

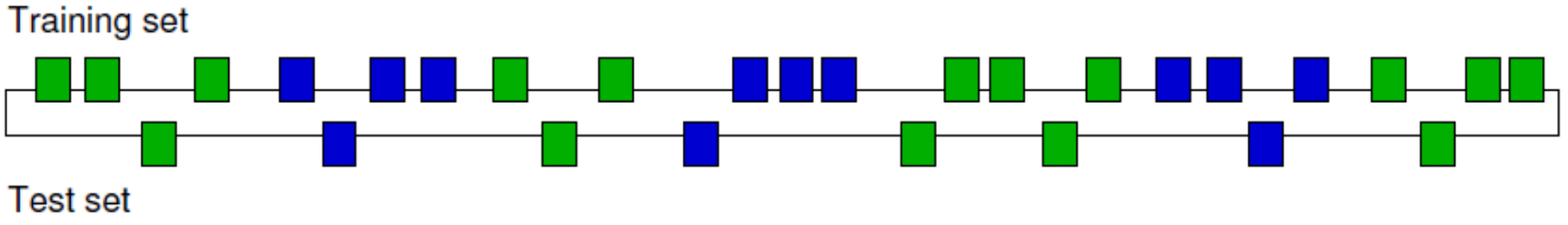
Assume the task is to discriminate between mental states in different conditions.
We say that an experiment has a block design, if the periods for which there is no alternation between conditions are longer than the intended change of states in online operation.



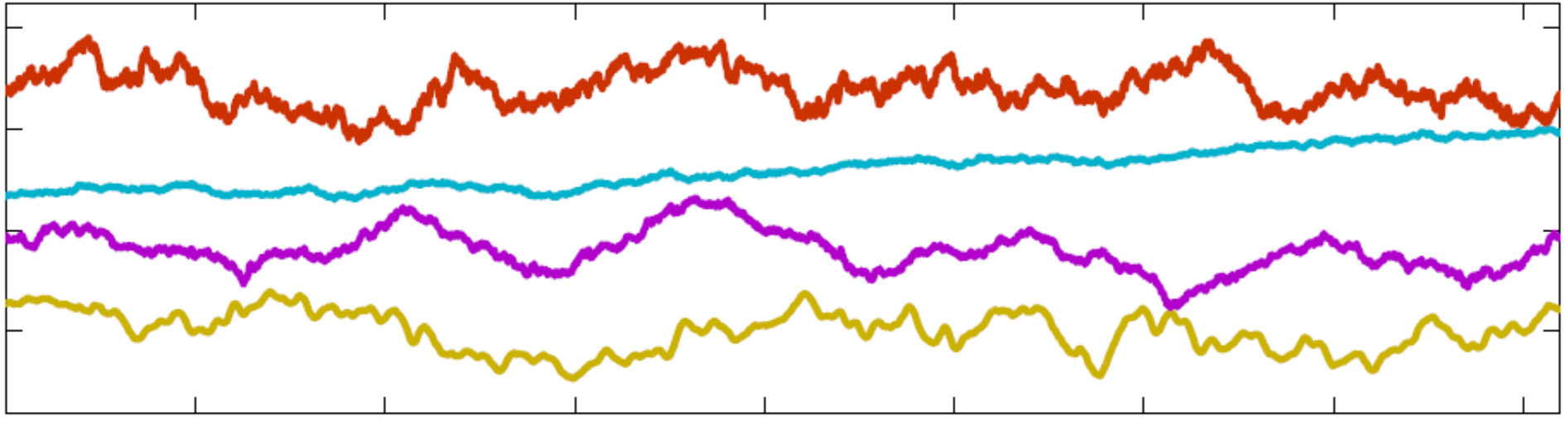
A problem arises, if the performance is estimated for such a data set by cross validation.



Slowly varying variables



In EEG there are many slowly changing variables of background activity, therefore the single-trials are not independent. For an ordinary cross validation in a block design data set, the requirement of independence between training and test set is violated.



A validation test

To demonstrate impact of block design in cross validation, we perform cross validation in the following setting. Taking an arbitrary EEG data set, we assign **fake** labels (regardless of what happened during the recording) like this:

nBlocksPerClass=1:

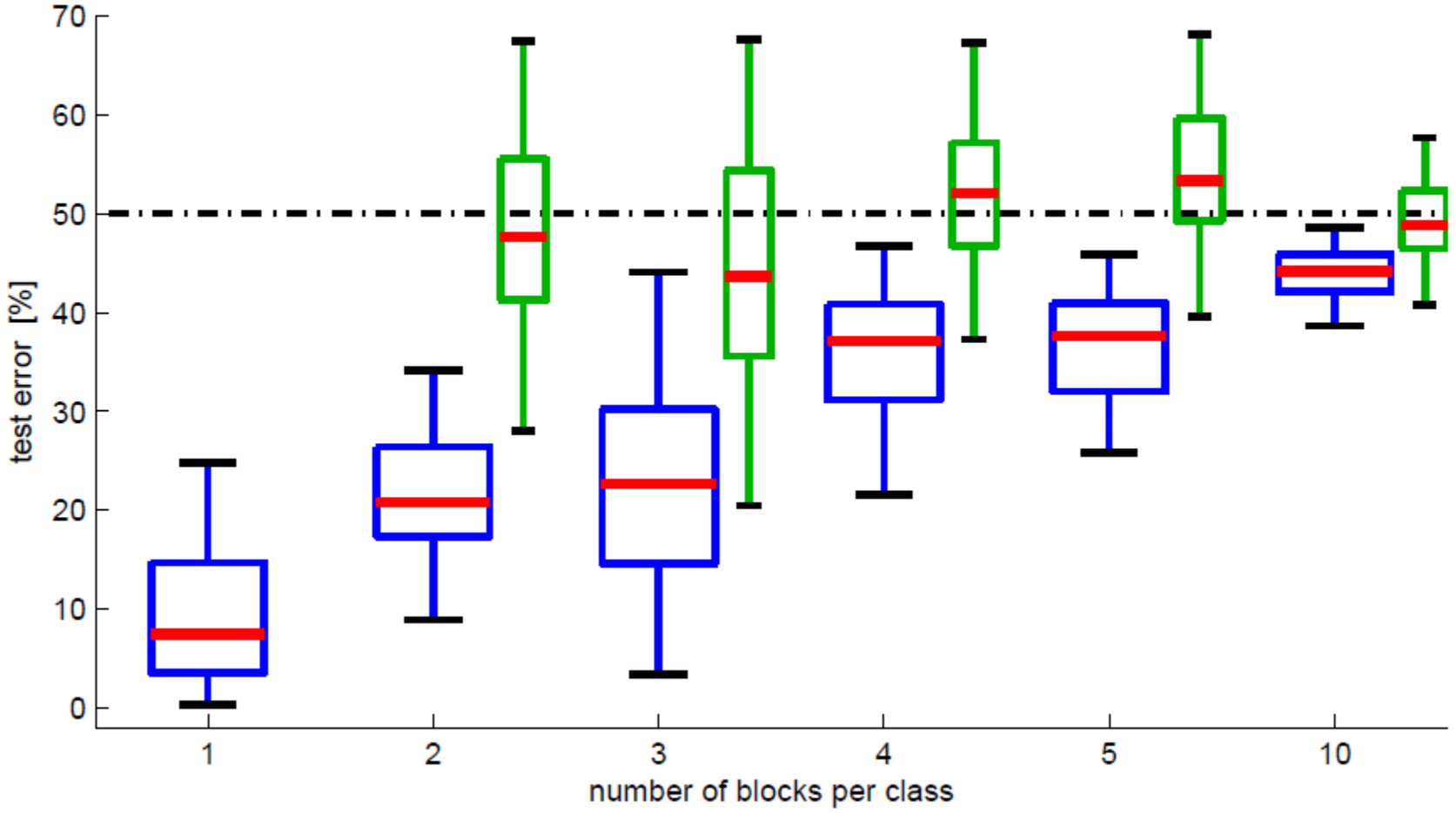
nBlocksPerClass=2:

nBlocksPerClass=3:

and so on.

Results of validation test

From each block single-trials are extracted of length 1s. This procedure was performed for 80 EEG data sets. Blue boxplots show the results of cross-validation:



For comparison, results for **leave-one-block-out** validation are shown in green.

Further remarks & summary

- The severeness of the underestimation of the true error depends on the complexity of the features and the classifier.
- Cross validation in block design data might also give the correct result – but alternative evaluation is required.
- The situation gets worse if trials are extracted from overlapping segments.
- The most realistic validation is to train the methods on the first $N - 1$ runs and to evaluate on the last run.
- Leave-one-block-out and leave-one-run-out have larger standard errors than cross validation.