Facing the small data reality in event-related potential BCI protocols

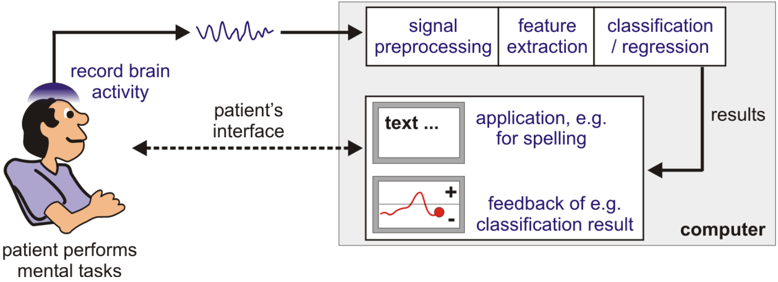
Michael Tangermann

Radboud University, Nijmegen, The Netherlands
Donders Institute
Data-Driven NeuroTech Lab
https://neurotechlab.socsci.ru.nl

michael.tangermann@donders.ru.nl

Nijmegen, CuttingGardens, Oct 17, 2023

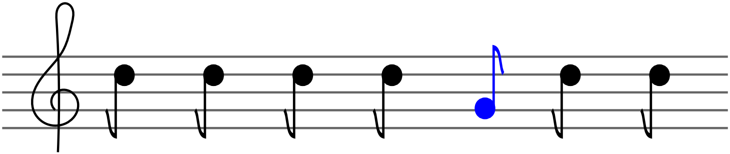
Context: Brain-Computer Interface System (**BCI**)



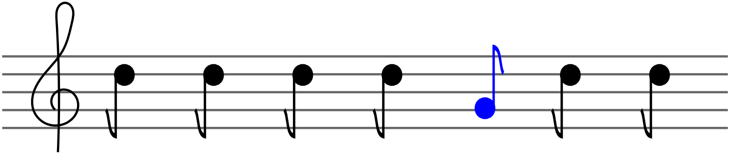
Predominantly used brain signal features:

- **Oscillatory power** upon imagery paradigms, steady-state evoked potentials
- **Event-related potentials (ERP)**: visual / auditory / haptic stimulation, noise-tagging, code-modulated protocols, ...

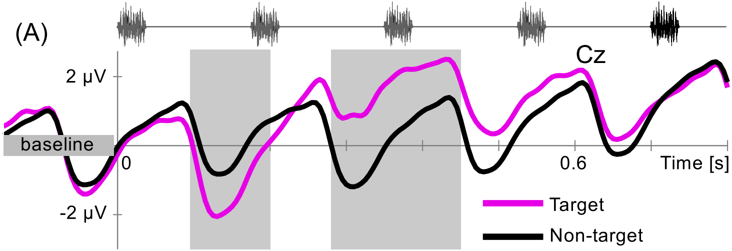
Context: auditory event-related potentials (ERP)



Context: auditory event-related potentials (ERP)

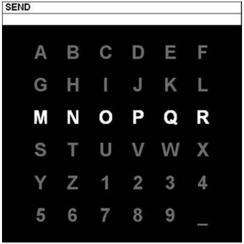


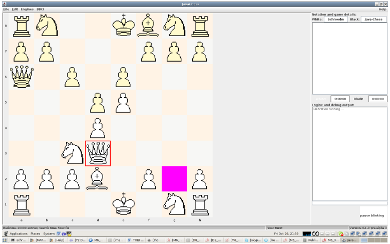
(after averaging out noise...)

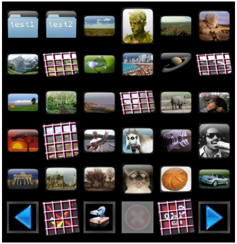


Attentive processing of a tone makes a difference!

Context: visual ERP applications for BCI





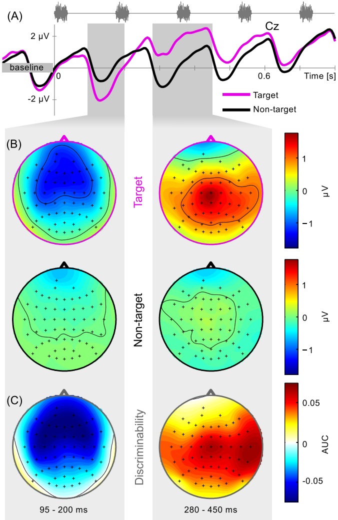


(video photobrowser) (video row-col speller)

Improvements compared to row-column speller:

- grid overlay instead of brightness highlighting
- **MSE decrease by 50 %** [Tangermann et al., IJ Bioelectromagnetism 2011], [Hübner et al., Brain Computer Interface, 2020]
- Compare: Face speller [Kaufmann et al., JNE 2011]

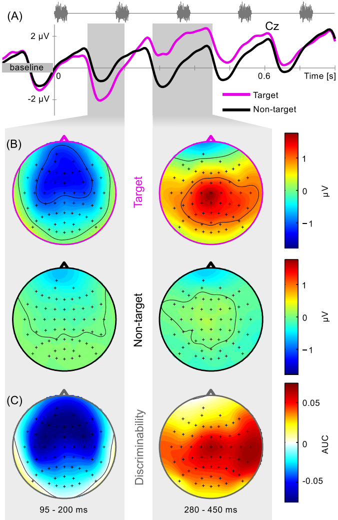
Classify target vs. non-target epochs



(Textbook) discriminative ERPs:

- early negative component (N100-N200)
- late positive component (P300a/b)

Classify target vs. non-target epochs



(Textbook) discriminative ERPs:

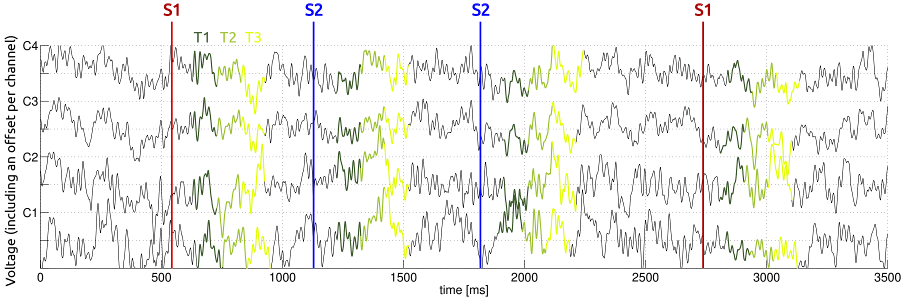
- early negative component (N100-N200)
- late positive component (P300a/b)

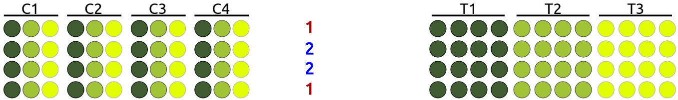
Typically used features and classification models:

1. few intervals per channel, average the potential per interval ($\rightarrow$ Linear Discriminant Analysis)
2. per-epoch covariance matrix ($\rightarrow$ Riemannian methods)
3. raw data ($\rightarrow$ Deep Learning)

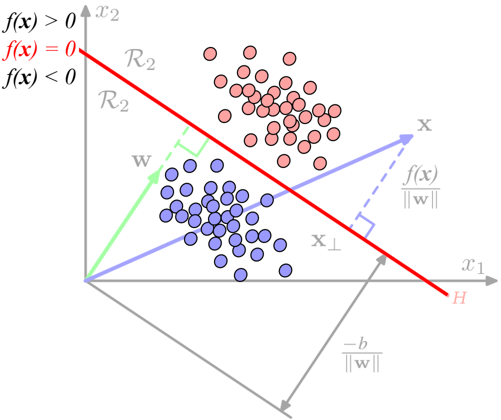
From time series data to ERP features

Toy example with target stimuli, non-target stimuli:
3 time features (T1, T2, T3) per channel, 4 channels (C1, C2, C3, C4).
$\rightarrow$ Data matrix $\mathbf{X}$ has 4 rows, each containing $3 * 4 = 12$ features.



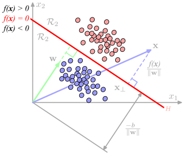


ERP classification with linear discriminant analysis (LDA): assumptions, parameters



Normal vector **w** and bias b need to be learned based on labeled training data.

ERP classification with linear discriminant analysis (LDA): assumptions, parameters



Normal vector **w** and bias b need to be learned based on labeled training data.

Assumptions made by LDA:

- Feature distributions are Gaussian
- Both classes share the same distribution

Trained LDA model is obtained by:

$$\mathbf{w} = \mathbf{\Sigma}_W^{-1}(\mathbf{m}_2 - \mathbf{m}_1)$$

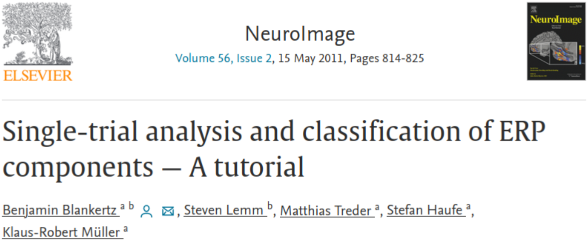
and

$$b = -\frac{1}{2}\mathbf{w}^T(\mathbf{m}_1 + \mathbf{m}_2)$$

(*Note: formulation for balanced classes*)

State of the art: LDA with shrinkage regularization

Seminal paper **on shrinkage regularization** of the empirical sample covariance matrix forms the basis of the following approaches.
`https://doi.org/10.1016/j.neuroimage.2010.06.048`

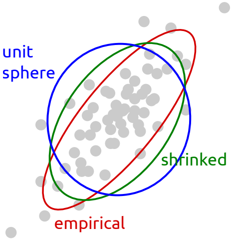

ELSEVIER

NeuroImage

Volume 56, Issue 2, 15 May 2011, Pages 814-825

Single-trial analysis and classification of ERP components — A tutorial

Benjamin Blankertz [a b], Steven Lemm [b], Matthias Treder [a], Stefan Haufe [a], Klaus-Robert Müller [a]



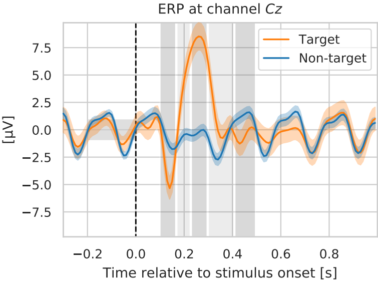
Motivation: we often face VERY small datasets!

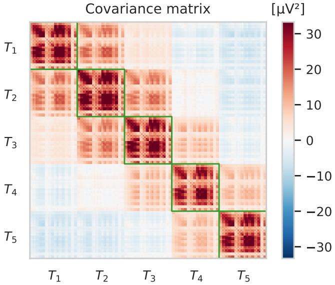
- New subject or new protocol and you want to **try out different experimental conditions**.
- Patient with limited stamina allows for **short sessions only**.
- You expect your data to change over time (**non-stationary feature distributions**), e.g., during a rehabilitation training of patients with aphasia.

Three methods to classify based on small ERP datasets

1. Time-decoupled LDA (supervised)
2. Block-Toeplitz with tapering (supervised)
3. Unsupervised mean-difference maximization (UMM)

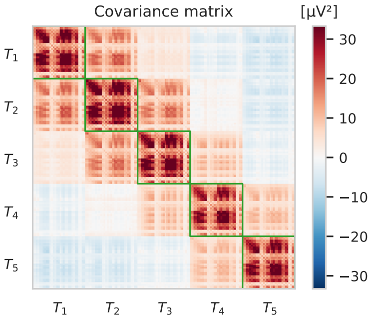
Very small auditory ERP datasets





- Training data: 78 epochs (13 target, 65 non-target), 155 features (31 channels and 5 time windows T_i) in *channel-prime* order.
- Upfront: Shrinkage regularization does a decent job.
- Can we do even better?

Estimating the empirical covariance matrix



- Observation: diagonal blocks of size 31×31 are similar
- Reminder: covariance matrix is calculated from **mean-free** data

$$\mathbf{\Sigma} = \frac{1}{N-1} \sum_N (\mathbf{x}_n - \mathbf{m})(\mathbf{x}_n - \mathbf{m})^T$$

(symmetric and positive semi-definite)

- $\rightarrow$ Covariance matrix describes background noise characteristic.

Two assumptions about background noise in ERP data

Given artifact-free data...

A1 : The noise on top of the ERP features is normally distributed.

Two assumptions about background noise in ERP data

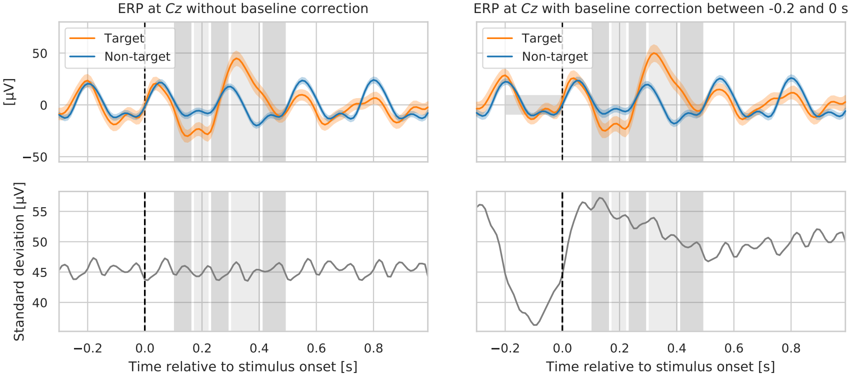
Given artifact-free data...

A1 : The noise on top of the ERP features is normally distributed.

A2 : Noise is unrelated to current user task, i.e.

- target or non-target epoch
- stimulation or no stimulation

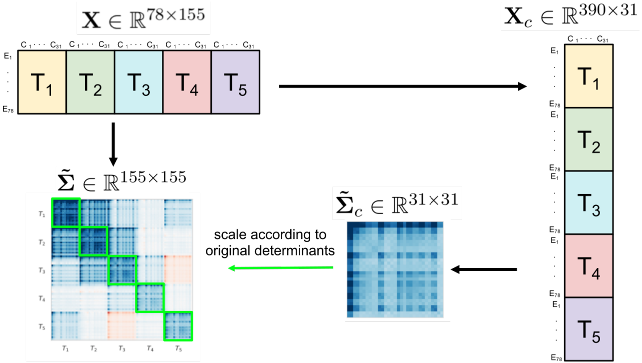
Checking assumption A2 by analyzing the noise



- A2 seems realistic if no baseline correction is performed.
- Not problematic: Can use *high-pass filter* instead (e.g., 0.5 Hz).
- High-pass filtering also helps to ensure A1!

Assuming independence between time intervals (A2) → **Time-Decoupled LDA** (TD-LDA)

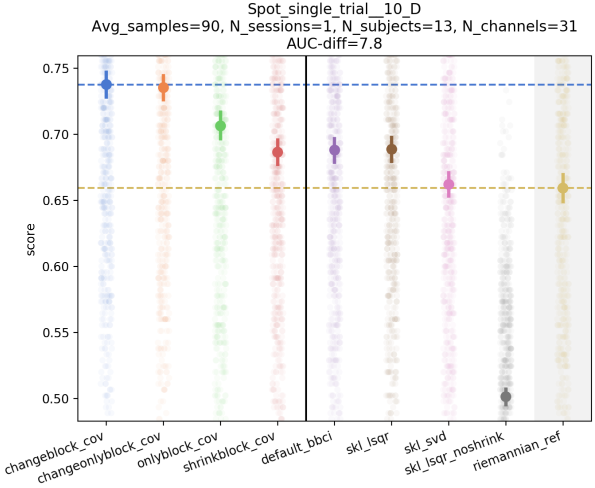
(1) Apply shrinkage regularization to empirical covariance matrix. **(2)** Improve estimate of diagonal blocks using *additional virtual* data points. **(3)** Exchange and re-scale the original diagonal blocks:



Using TD-LDA, the diagonal blocks are estimated from 5 times more data!

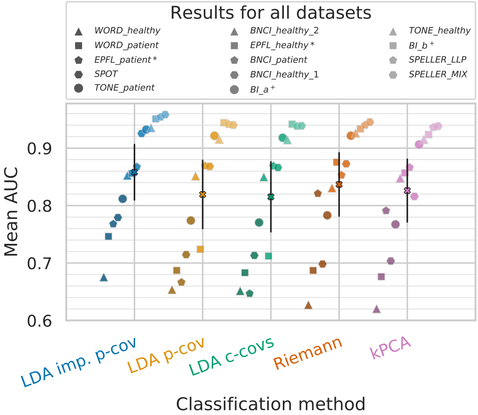
TD-LDA Results: High Performance for Small Datasets

Improved $\tilde{\Sigma}$ together with the standard class means $\rightarrow$ **TD-LDA**.
Applied to small auditory ERP datasets:



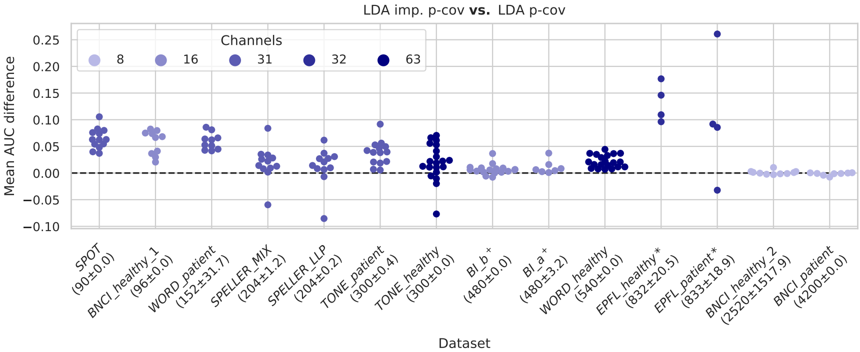
Results on various ERP datasets (auditory, visual)

TD-LDA ("LDA imp. p-cov") improves on many different ERP datasets compared to shrinkage regularized LDA ("LDA p-cov"):



[Sosulski et al.(2021), Neuroinformatics, 19(3):461-476.]

Improvements depend on size of training dataset



- TD-LDA is **specifically effective for small datasets** but does not hurt for large datasets.
- TD-LDA uses domain-specific regularization of covariance matrix.
- TD-LDA does **not** improve the quality of the class means.

Three methods to classify based on small ERP datasets

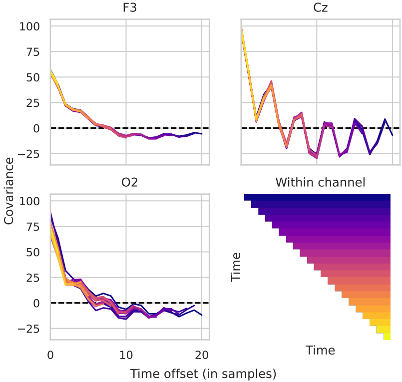
1. Time-decoupled LDA (supervised)
2. Block-Toeplitz with tapering (supervised)
3. Unsupervised mean-difference maximization (UMM)

(Another) two assumptions about noise in ERP data

A3 : Only ERP signal is time-locked, EEG background is stationary.
$\rightarrow$ covariance across time depends only on temporal distance δ between samples, i.e. $cov(x^{t_j}, x^{t_i}) = cov(x^{t_j+\delta}, x^{t_i+\delta})\ \forall\delta \in \mathbb{R}$.

A4 For increasing temporal distances, i.e. $|t_i - t_j| \rightarrow \infty$, the covariance goes towards zero.

Checking assumption A4

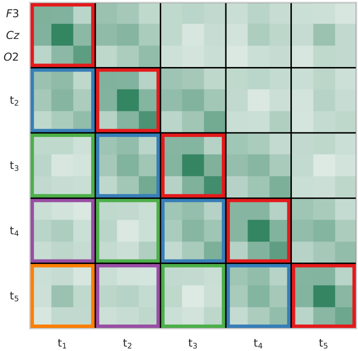


- Plots show covariances within three different channels across different temporal distances δ.
- Assumptions seems to hold for EEG channels F3 and Cz, but less so for channel O2 (Ideally: curves should overlap and approach zero).

Implementing assumption A3

With features in channel-prime order and after initial shrinkage:

If same temporal distances imply the same covariance within one channel, then we can average along the diagonal blocks AND along each of the off-diagonal blocks separately → **Toeplitz structure**.

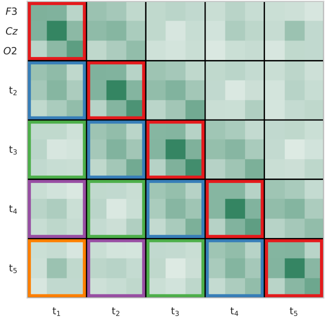


Memory requirements?

Implementing assumption A4

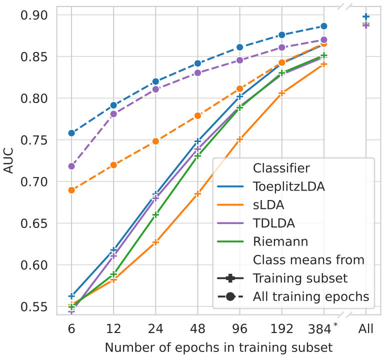
If covariance within one channel goes to zero with increased temporal distance, then we can taper down the blocks from the main diagonal to the corners. Practically:

- use a **linear tapering function**: strong weight on main diagonal, small weight on covariance blocks describing large temporal distance.
- simple implementation: add up the blocks across each diagonal



Results: Block-Toeplitz with tapering

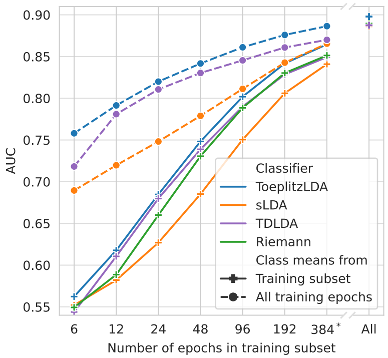
Evaluated on 13 ERP dataset with over 200 subjects:



- solid lines: realistic performances
- dashed lines: performance with improved covariances, but maximally (unrealistic) informative class means

Results: Block-Toeplitz with tapering

Evaluated on 13 ERP dataset with over 200 subjects:



- solid lines: realistic performances
- dashed lines: performance with improved covariances, but maximally (unrealistic) informative class means
- Toeplitz-LDA slightly outperforms TD-LDA, strongly outperforms shrinkage-regularized LDA (sLDA).
- Improved class mean estimates *could* boost performance further.

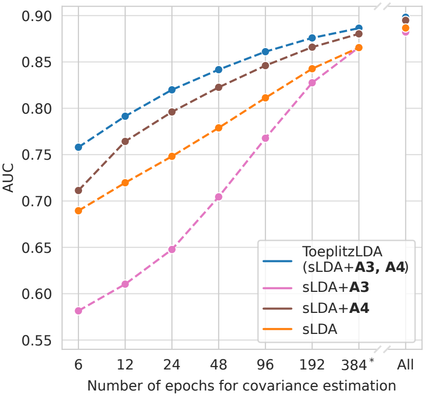
Block-Toeplitz with tapering: visual ERP speller

Observation: Block-Toeplitz LDA drastically outperforms shrinkage-regularized LDA on this application metric for an *unsupervised approach* (correctly spelled letters).

[Sosulski & Tangermann, Journal of Neural Eng., 2022, `https://doi.org/10.1088/1741-2552/ac9c98`]

How strong is the influence of the assumptions?

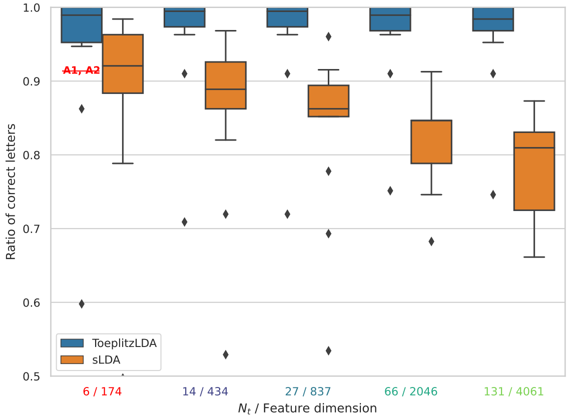
With (unrealistic) oracle for optimal class mean estimates:



- Major improvement by tapering alone (A4)
- Using A3 alone (block-wise averaging per diagonal without tapering) mimicks equally good estimates of covariances independent of temporal distance
  $\rightarrow$ performance drop!
- Combination of assumptions A3 and A4 works best.

Block-Toeplitz LDA scales with many temporal features:

Increasing the number of time intervals per channel:



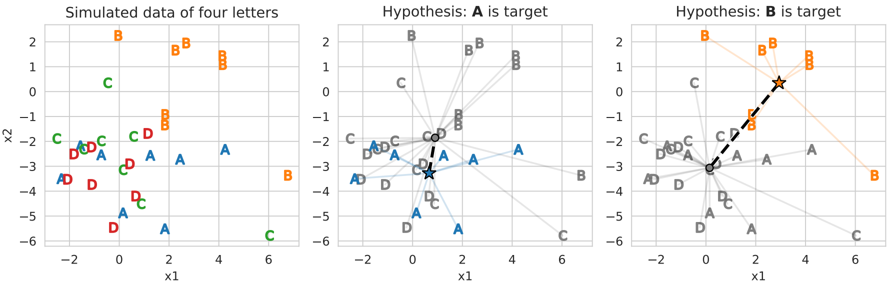
- sLDA suffers from higher feature dimensionality (as covariance matrix is harder to estimate).
- Block-Toeplitz LDA can cope with original samples! Definition of feature intervals dispensable?

Three methods to classify based on small ERP datasets

1. Time-decoupled LDA (supervised)
2. Block-Toeplitz with tapering (supervised)
3. Unsupervised mean-difference maximization (UMM)

Unsupervised mean-difference maximization

Toy example: which of the four symbols is the attended target?



- Idea: The true target mean is expected to have largest distance to the mean of the other (non-target) symbols.

UMM is unsupervised, and it acts instantaneously on a single trial

Algorithm 1 Pseudocode for the basic UMM method. Variants of blue lines are described in Sections 2.2.2 and 2.2.3.

Require: available symbols S, epochs of i-th trial $E^{(i)}$

for every trial i **do**
    $\Sigma^{-1} \leftarrow \text{cov}(E^{(i)})^{-1}$ ▷ no class labels needed
    $d^* \leftarrow -\infty$
    for s in S **do**
        $\Delta\boldsymbol{\mu}_s \leftarrow \text{mean}(E^{(i)}_{A^{s+}}) - \text{mean}(E^{(i)}_{A^{s-}})$
        $d \leftarrow \Delta\boldsymbol{\mu}_s \Sigma^{-1} \Delta\boldsymbol{\mu}_s^{\intercal}$
        if $d > d^*$ **then**
            $d^* \leftarrow d$
            $s^* \leftarrow s$
        end if
    end for ▷ s^* decoded symbol for trial i
end for

- Sequentially check all possible hypotheses for largest mean difference.
- Distances d are computed using a covariance correction (cp. to Mahalanobis distances).
- Covariance matrices are estimated using shrinkage regularization with following block-Toeplitz regularization with tapering.

UMM comes with a confidence and can learn across trials.

$$c = \frac{d^{\Sigma}(s^*) - d^{\Sigma}(s^r)}{\sigma_{S^-}}$$

- Confidence c is obtained by comparing the winner (*) distance to the runner-up (^{r}) distance.

$$\boldsymbol{\mu}^C_{s+} = \frac{\left[\sum_{l=1}^{N_t} (\hat{c}^{(l)} \cdot \boldsymbol{\mu}_+^{(l)}) + c^{(i)} \cdot \mathrm{mean}\left(E^{(i)}_{A^{s+}}\right)\right]}{\sum_{l=1}^{N_t} (\hat{c}^{(l)}) + c^{(i)}}$$

- Class means (and covariances) can be combined across trials either optimistically or based on the confidence obtained for each trial.

UMM results for binary classification

Target vs. non-target classification of multiple (MOABB) datasets:

UMM classification rates for all datasets and hyperparameters

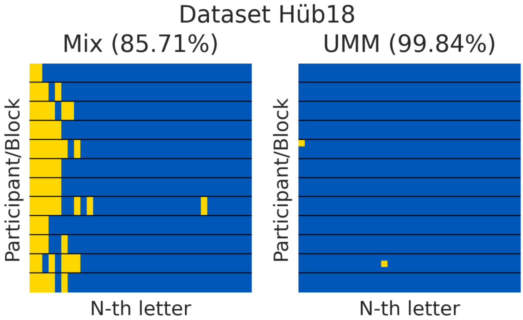
Dataset	Σ_s^1 μ^1	Σ_s^{all} μ^1	Σ_t^1 μ^1	Σ_t^{all} μ^1	Σ_s^1 μ^O	Σ_s^{all} μ^O	Σ_t^1 μ^O	Σ_t^{all} μ^O	Σ_s^1 μ^C	Σ_s^{all} μ^C	Σ_t^1 μ^C	Σ_t^{all} μ^C
Hüb17 (38)	76.02	72.18	92.19	91.44	80.58	95.95	99.37	**99.96**	81.83	98.66	99.37	**99.96**
Hüb18 (36)	80.16	74.84	96.11	94.44	72.70	98.97	99.21	**99.92**	73.65	99.05	99.52	99.84
Lee19 (107)	54.99	44.95	73.86	71.92	49.76	79.79	95.57	98.48	54.58	93.40	97.93	**99.47**
Ric13 (8)	50.00	42.14	62.50	59.29	42.86	47.14	58.93	59.64	41.43	**77.86**	60.71	59.64
Sch14 (21)	47.23	47.02	54.15	52.16	55.25	62.27	78.87	76.74	56.17	75.85	**82.52**	76.74

Used estimators

Classification rate: 0 – 25 – 50 – 75 – 100

- Good instantaneous performances.
- Confidence-based history of means outperforms state-of-the-art for visual datasets Hüb17, Hüb18 and Lee19.
- Auditory datasets are harder (known).
- Patient dataset Ric13 can run into problems, if initial hypothesis is wrong. (For repair, see Poster on UMM)

UMM results for letter selection



- For visual ERP datasets, basically error-free letter selection (offline replay of MIX dataset obtained by Hübner et al.).

Wrap-up I

- Many BCIs require reliable ERP classification. Small (training) datasets are a common problem.
- Novel LDA variants with domain-specific regularizations can perform extremely well (**TD-LDA**, **Toeplitz-LDA**).
- With **UMM**, a novel *unsupervised* classification approach is available with potential to:
  - completely omit calibration
  - completely omit warm-up period (as in other unsupervised methods)
  - mitigate non-stationarity (UMM can be used instantaneously)
  - use confidence for, e.g., dynamic stopping, outlier detection, ...

Wrap-Up II

- Please approach me for code on the LDA methods presented, for thesis projects etc.
- A lot of the results presented have been investigated by my PhD student **Jan Sosulski**

- Alternative approach for small datasets (and not limited to ERP data) is to use transfer learning from earlier sessions / other subjects / different ERP tasks, see talks by **Pierre Guetschel** (today, local talk in Nijmegen), **Reinmar Kobler** (global talk today) and **Hubert Banville** (global talk on Thursday)