Geodesic Optimization for Predictive Shift Adaptation on EEG data (GOPSA)

Apolline Mellot*, Antoine Collas*, Sylvain Chevallier, Alexandre Gramfort, Denis Engemann

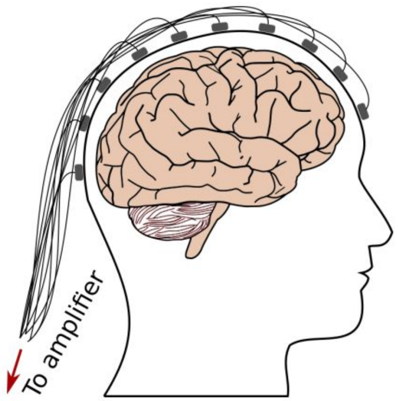
Accepted at NeurIPS 2024 (Spotlight)
http://arxiv.org/abs/arXiv:2407.03878

MIND team meeting - October 15, 2024

Table of contents

1. Context

1. Context: Analysis of complex biological signals



Electroencephalography (EEG): non invasive recording of brain activity

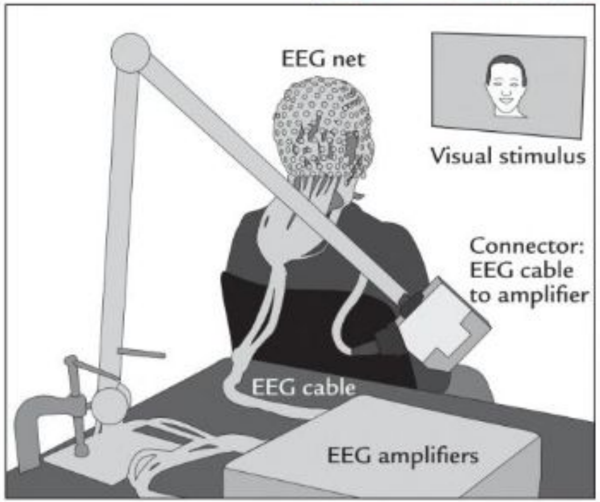
<u>ML applied to EEG</u>:

- **Classification:** Brain Computer Interface (BCI), epileptic seizure detection, sleep staging, etc.
- **Regression:** Risk scores, optimal drug-dosage, brain age, etc.

→ Focus on **regression context**

- EEG recording setup:
    - Recording protocol: rest, visual stimuli…
    - EEG cap with electrodes
    - Amplifier

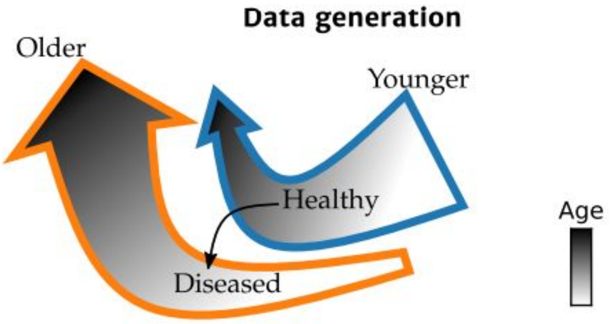
[Hari and Puce 2023]



1. **Context:** Generalization of ML models for EEG

- Generalize across **different context / populations**:

<u>Example:</u> extraction of validate biomarkers



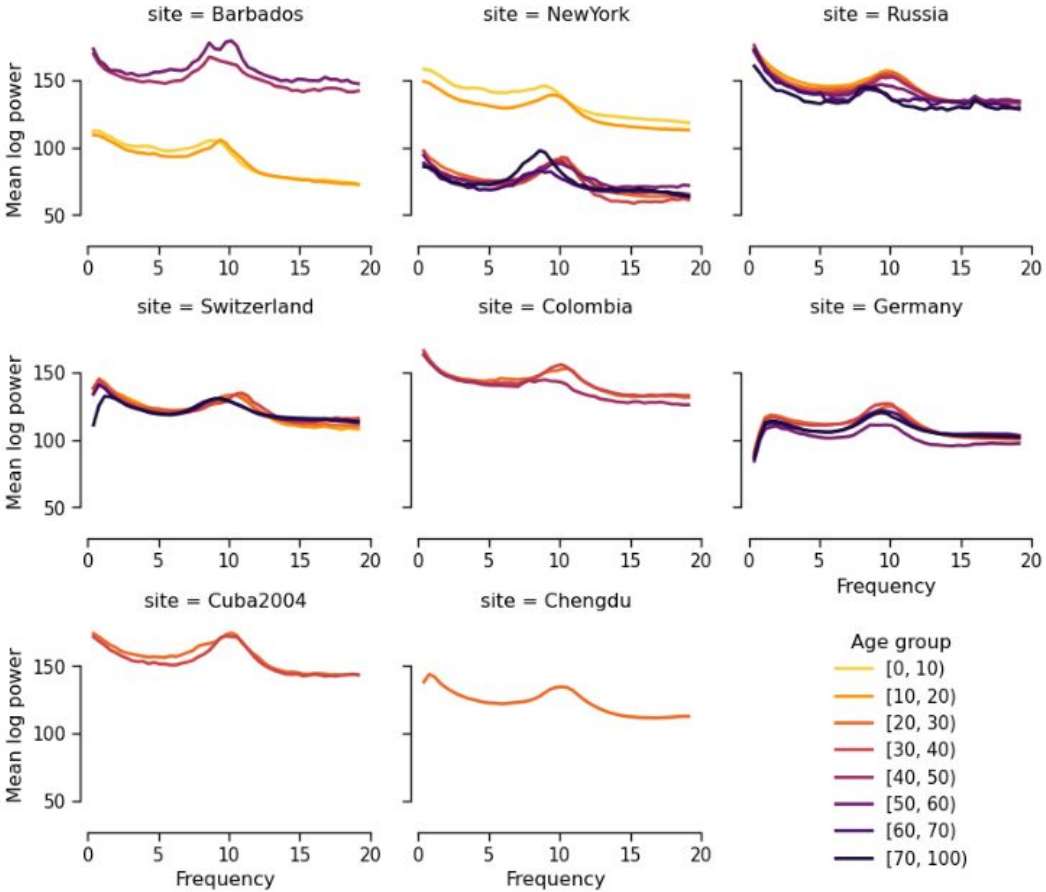
[Dockes *et al.* 2021]

- Accurate ML models:

Increasing the number of data = Increasing the performances

→ **Pooling** several existing datasets: recent emergence of large databases

1. **Context:** Inherent variability of EEG signals



- Many causes of **variability** in EEG data:
  - recording devices
  - populations
  - recording sites
  - preprocessing

[Li *et al* 2022]

1. **Context:** Dataset shift and domain Adaptation

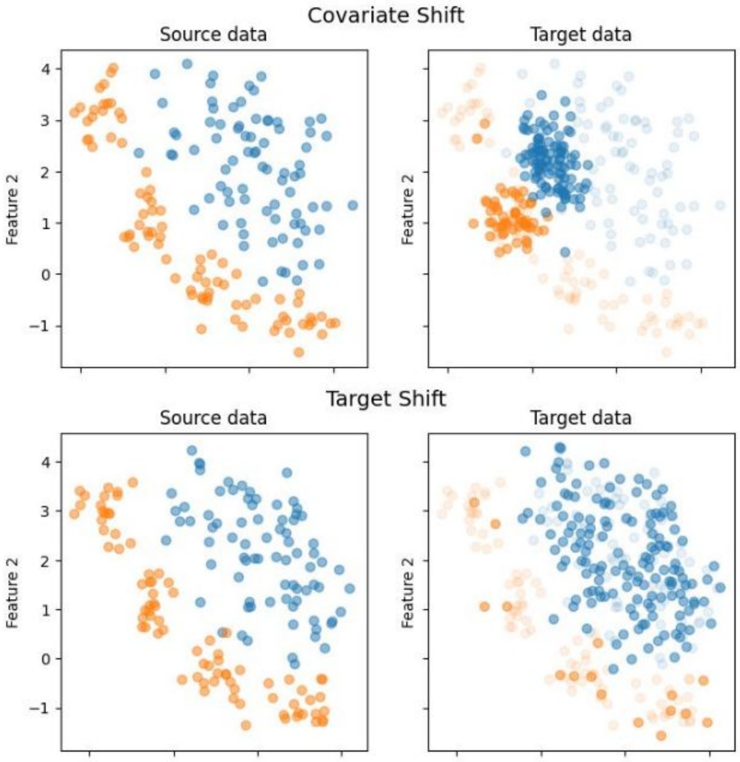
This variability leads to discrepancies across datasets called **dataset shifts.**

→ Problem: dataset shifts limit generalization of ML models

Domain adaptation (DA) tries to reduce shift between datasets.

But DA methods for EEG usually focus on **one** type of shift →not realistic

Different types of shift:



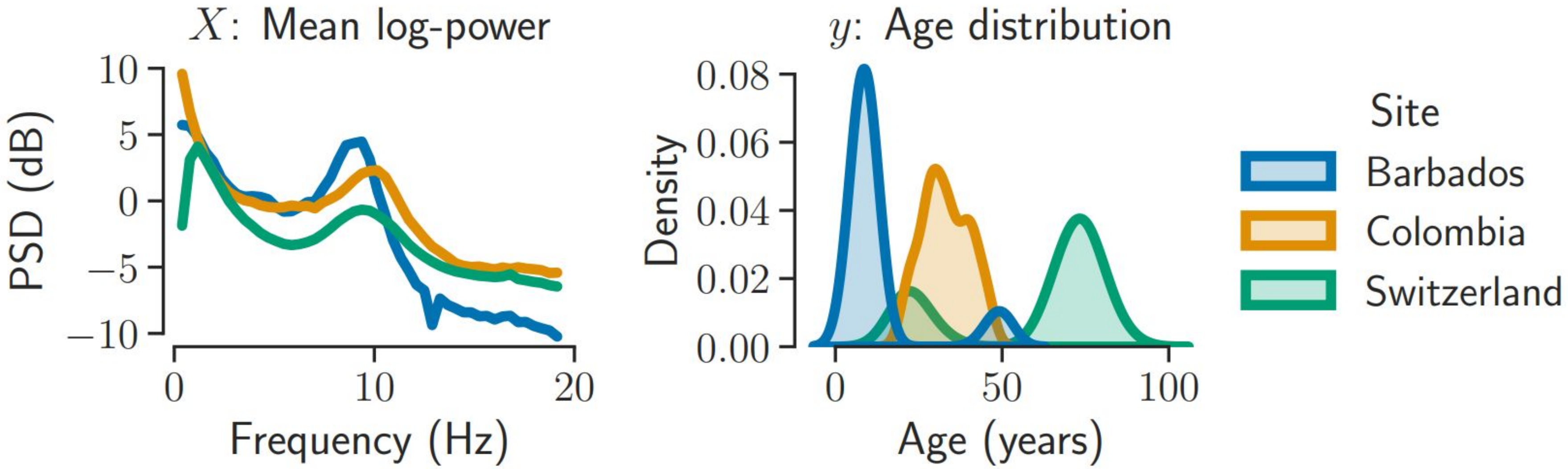
Source: SKADA
[Gnassounou *et al*, 2024b]

2. Motivation

2. Motivation: To deal with joint shift in X and y

- Real world applications like **multicenter studies**: both shift in X and y

Example: HarMNqEEG dataset for age prediction



Goal: Multi-source DA to tackle shifts in X and y **jointly**

[Li *et al* 2022]

2. Motivation: To deal with joint shift in X and y

Combine several datasets as train set (**source**) and test on a new unseen dataset (**target**)

Assume to know the **mean** of the target labels $\bar{y}_T$ to adapt the target data

Multi-source test-time semi-supervised Domain Adaptation

No need to re-train for a new target dataset

3. Related work

3. **Related work:** Riemannian geometry for EEG

Riemannian-based models proved effective with EEG in BCI, biomarker exploration…

$\longrightarrow$Use **spatial covariance matrix** as descriptor:

Spatial covariance representation

EEG signals are multivariate time series $X \in \mathbb{R}^{d \times T}$ recorded from d sensors over T time points.

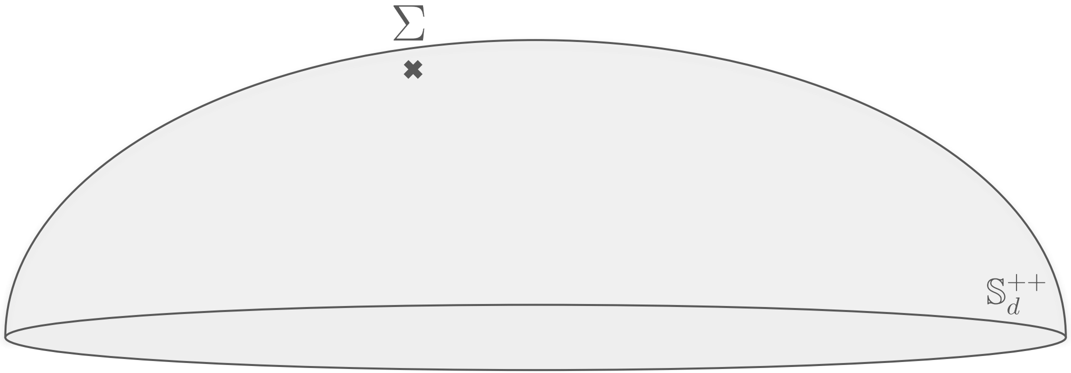
The spatial covariance matrix $\Sigma \in \mathbb{S}_d^{++}$ of X is defined as:

$$\Sigma = \frac{1}{T} X X^\top$$

- Covariance matrices are symmetric positive definite: **smooth manifold**

[Pennec *et al.* 2006]

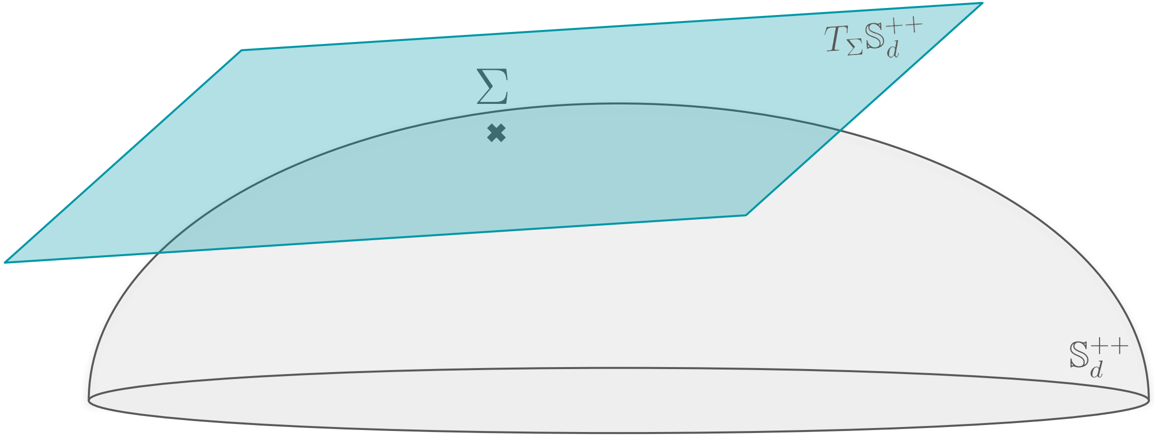
3. Related work: Riemannian geometry for EEG



- Covariance matrices are symmetric positive definite: **smooth manifold**

[Boumal 2023]

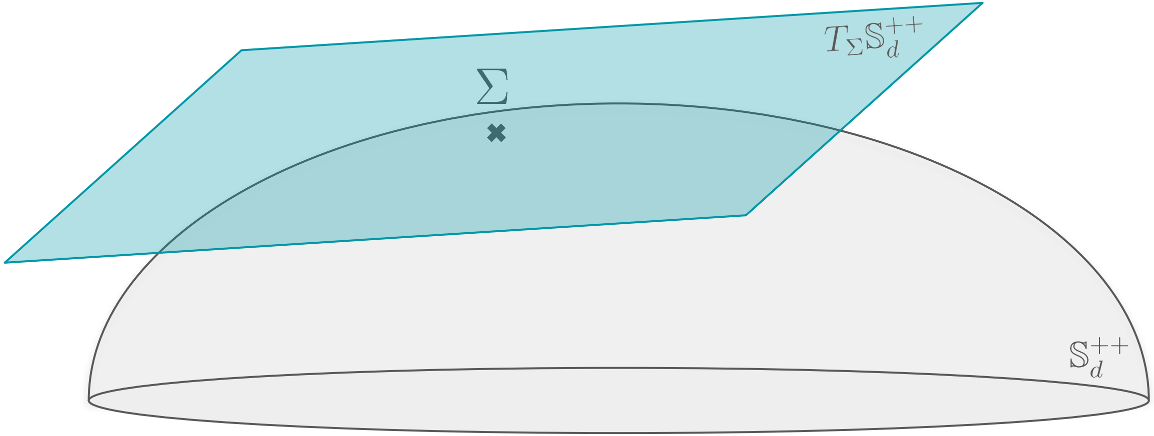
3. Related work: Riemannian geometry for EEG



- Covariance matrices are symmetric positive definite: smooth manifold
- Vector space defined at each point: **tangent space**

[Boumal 2023]

3. Related work: Riemannian geometry for EEG



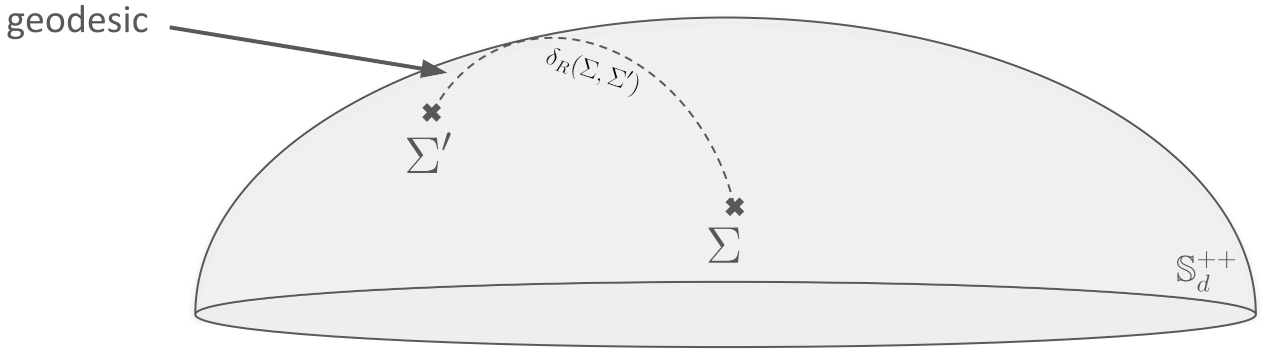
- Covariance matrices are symmetric positive definite: smooth manifold
- Vector space defined at each point: tangent space
- Equipped with smooth inner product: **Riemannian manifold**

[Pennec *et al.* 2006]

Affine-invariant Riemannian metric

Given $\Gamma, \Gamma' \in T_\Sigma \mathbb{S}_d^{++}$: $\langle \Gamma, \Gamma' \rangle_\Sigma = \operatorname{tr}\left(\Sigma^{-1}\Gamma\Sigma^{-1}\Gamma'\right)$

3. Related work: Riemannian geometry for EEG



Affine-invariant Riemannian distance

$$\delta_R(\Sigma, \Sigma') = || \log \left(\Sigma^{\frac{-1}{2}} \Sigma' \Sigma^{\frac{-1}{2}} \right) ||_F$$

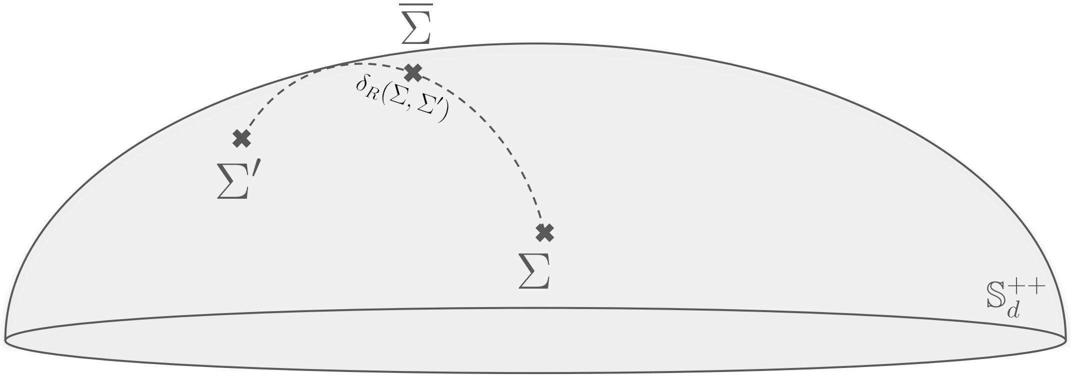
with $\log : \mathbb{S}_d^{++} \to \mathbb{S}_d$ the matrix logarithm:

$$\log(\Sigma) = U \log \Delta U^\top$$

being $\Sigma = U \Delta U^\top$ the SVD of Σ

[Pennec *et al.* 2006]

3. Related work: Riemannian geometry for EEG



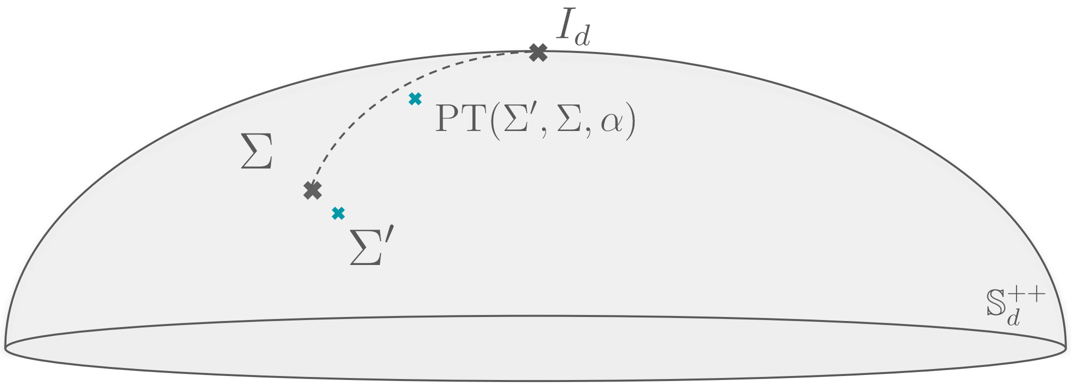
Riemannian mean

For a set $\{\Sigma_i\}_{i=1}^N \subset \mathbb{S}_d^{++}$:

$$\overline{\Sigma} = \arg\min_{\Sigma \in \mathbb{S}_d^{++}} \sum_{i=1}^{N} \delta_R(\Sigma, \Sigma_i)^2$$

[Pennec *et al.* 2006]

3. Related work: Riemannian geometry for EEG



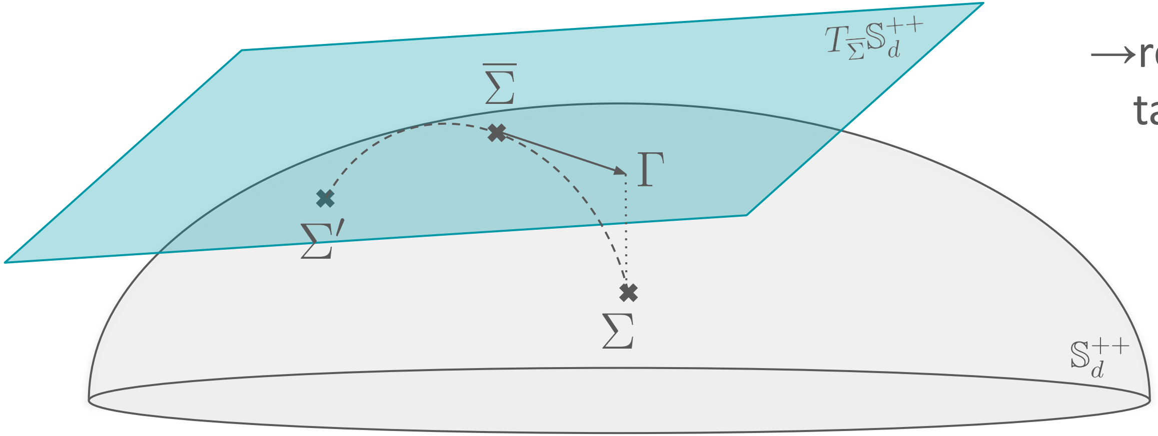
Lemma: Parallel transport (PT) to the identity

Given $\Sigma, \Sigma' \in \mathbb{S}_d^{++}$, the parallel transport of Σ' along the geodesic from Σ to the identity I_d at $\alpha \in [0, 1]$ is:

$$\mathrm{PT}\left(\Sigma', \Sigma, \alpha\right) = \Sigma^{\frac{-\alpha}{2}} \Sigma' \Sigma^{\frac{-\alpha}{2}}$$

- PT usually used to align distributions

3. Related work: Riemannian geometry for EEG



→regression on the tangent space

Riemannian logarithmic mapping and feature extraction

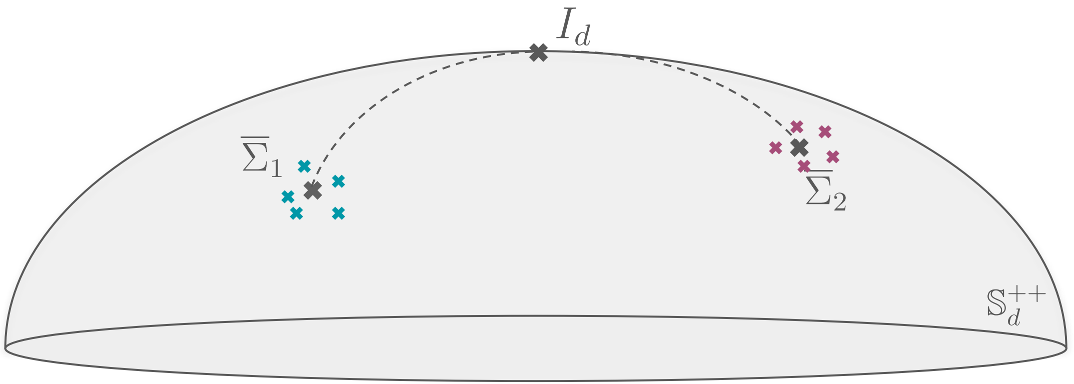
Tangent vector:

$$\Gamma = \log_{\overline{\Sigma}}(\Sigma) = \overline{\Sigma}^{\frac{1}{2}} \log\left(\overline{\Sigma}^{\frac{-1}{2}} \Sigma \overline{\Sigma}^{\frac{-1}{2}}\right) \overline{\Sigma}^{\frac{1}{2}} \in T_{\overline{\Sigma}}\mathbb{S}_d^{++}$$

Feature extraction:

$$\phi\left(\Sigma_i, \overline{\Sigma}\right) = \operatorname{uvec}\left(\log_{I_d}\left(\operatorname{PT}\left(\Sigma_i, \overline{\Sigma}, 1\right)\right)\right) = \operatorname{uvec}\left(\log\left(\overline{\Sigma}^{\frac{-1}{2}} \Sigma_i \overline{\Sigma}^{\frac{-1}{2}}\right)\right) \in \mathbb{R}^{\frac{d(d+1)}{2}}$$

where uvec is the upper triangular part with off-diagonal elements multiplied by $\sqrt{2}$



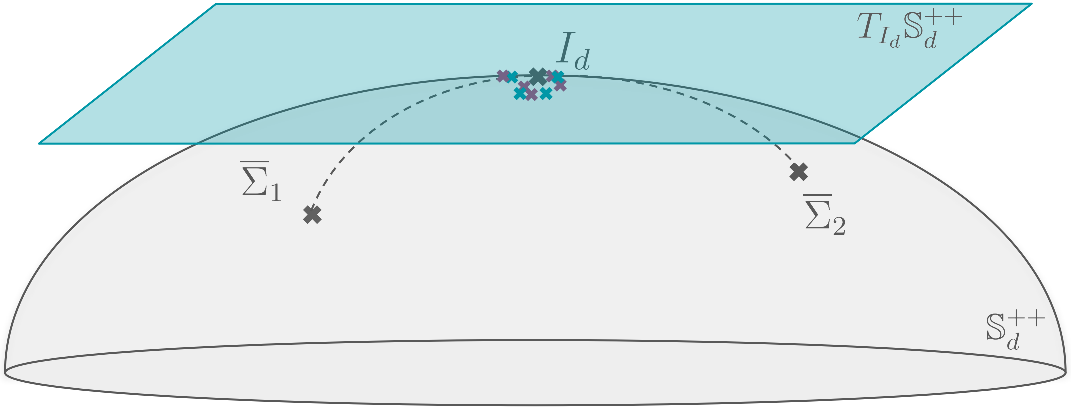
Re-center domains to identity

Each domain $k \in [\![1, K]\!]$ is parallel transported from its Riemannian mean $\overline{\Sigma}_k$ to the identity:

$$\phi(\Sigma_{k,i}, \overline{\Sigma}_k) = \mathrm{uvec}\left(\log\left(\overline{\Sigma}_k^{\frac{-1}{2}} \Sigma_{k,i} \overline{\Sigma}_k^{\frac{-1}{2}}\right)\right)$$

[Zanini et *al.*, 2018] [Yair et *al.*, 2019]

3. Related work: Riemannian DA for M/EEG



Re-center domains to identity

Each domain $k \in [\![1, K]\!]$ is parallel transported from its Riemannian mean $\overline{\Sigma}_k$ to the identity:

$$\phi(\Sigma_{k,i}, \overline{\Sigma}_k) = \mathrm{uvec}\left(\log\left(\overline{\Sigma}_k^{\frac{-1}{2}} \Sigma_{k,i} \overline{\Sigma}_k^{\frac{-1}{2}}\right)\right)$$

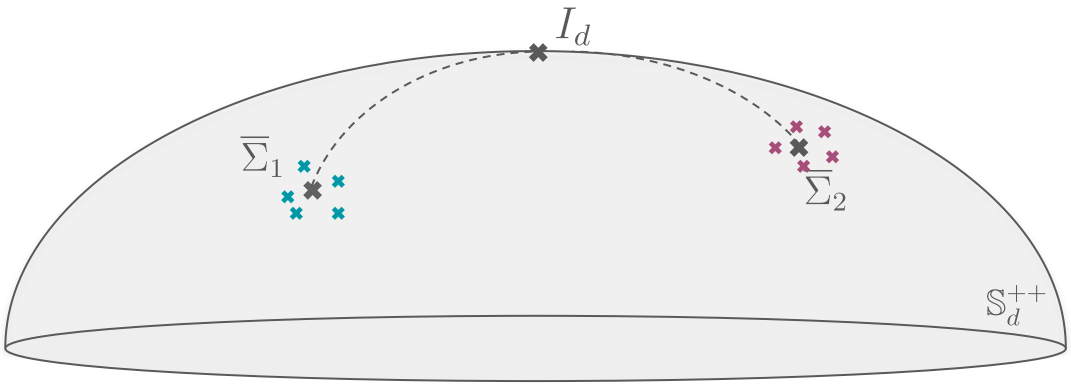
→ Strong baseline for brain when no shift in y

Problem: when different y distributions between domains, re-centering to common reference remove information of interest

[Zanini et *al.*, 2018] [Yair et *al.*, 2019]

4. Contribution

4. Contribution: Joint adaptation of shifts in X and y



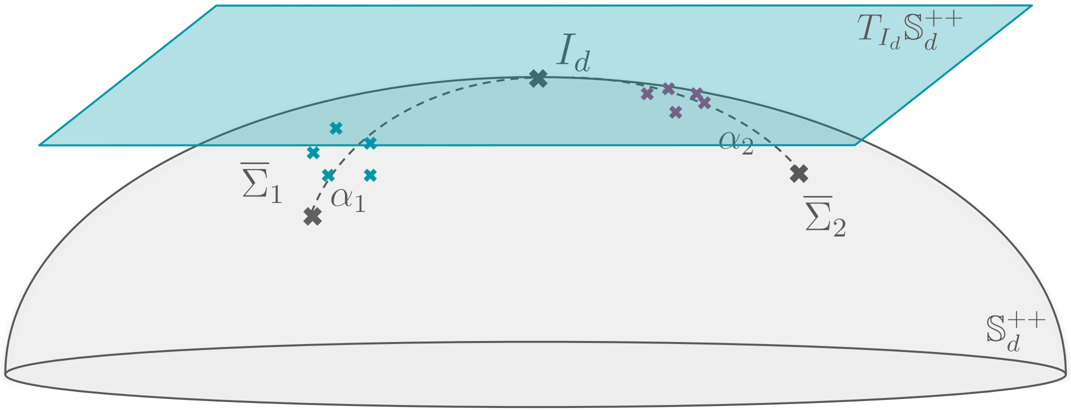
Parallel transport along the geodesic: GOPSA

Each domain $k \in [\![1, K]\!]$ is parallel transported along the geodesic between Riemannian mean $\overline{\Sigma}_k$ and the identity:

$$\phi(\Sigma_{k,i}, \overline{\Sigma}_k, \alpha) = \mathrm{uvec}\left(\log_{I_d}\left(\mathrm{PT}\left(\Sigma_{k,i}, \overline{\Sigma}_k, \alpha\right)\right)\right) = \mathrm{uvec}\left(\log\left(\overline{\Sigma}_k^{\frac{-\alpha}{2}} \Sigma_{k,i} \overline{\Sigma}_k^{\frac{-\alpha}{2}}\right)\right)$$

with $\alpha \in [0, 1]$

4. Contribution: Joint adaptation of shifts in X and y



Parallel transport along the geodesic: GOPSA

Each domain $k \in [\![1, K]\!]$ is parallel transported along the geodesic between Riemannian mean $\overline{\Sigma}_k$ and the identity:

$$\phi(\Sigma_{k,i}, \overline{\Sigma}_k, \alpha) = \mathrm{uvec}\left(\log_{I_d}\left(\mathrm{PT}\left(\Sigma_{k,i}, \overline{\Sigma}_k, \alpha\right)\right)\right) = \mathrm{uvec}\left(\log\left(\overline{\Sigma}_k^{\frac{-\alpha}{2}} \Sigma_{k,i} \overline{\Sigma}_k^{\frac{-\alpha}{2}}\right)\right)$$

with $\alpha \in [0, 1]$

4. Method: Train time

We have access to K labeled source domains $\{(\Sigma_{k,i}, y_{k,i})\}_{i=1}^{N_k}$

We define the concatenation of the source data:

$$Z_{\mathcal{S}}(\gamma) = \left[\phi\left(\Sigma_{1,1}, \overline{\Sigma}_1, \overset{\alpha_1}{\sigma(\gamma_1)}\right), \ldots, \phi\left(\Sigma_{K,N_K}, \overline{\Sigma}_K, \overset{\alpha_K}{\sigma(\gamma_K)}\right)\right]^\top \in \mathbb{R}^{N_{\mathcal{S}} \times d(d+1)/2}$$

Train time optimization problem

Simultaneously learns features and regression model:

$$\gamma_{\mathcal{S}}^\star = \underset{\gamma \in \mathbb{R}^K}{\arg\min} \left\{ \mathcal{L}_{\mathcal{S}}(\gamma) = ||y_{\mathcal{S}} - Z_{\mathcal{S}}(\gamma)\beta_{\mathcal{S}}^\star(\gamma)||_2^2 \right\}$$

subject to $\beta_{\mathcal{S}}^\star(\gamma) = Z_{\mathcal{S}}(\gamma)^\top(\lambda I_N + Z_{\mathcal{S}}(\gamma) Z_{\mathcal{S}}(\gamma)^\top)^{-1} y_{\mathcal{S}}$

1. Parallel transport the covariance matrices.
2. Vectorization and predicted output with linear regression.
3. Comparison with the true output values.

4. **Method:** Test time

We have access to a new unseen target domain $(\Sigma_{\mathcal{T},i})_{i=1}^{N_{\mathcal{T}}}$ and assume to know $\bar{y}_{\mathcal{T}}$.

We define the concatenation of the target data:

$$Z_{\mathcal{T}}(\gamma) = \left[\phi\left(\Sigma_{\mathcal{T},1}, \overline{\Sigma}_{\mathcal{T}}, \sigma(\gamma)\right), \dots, \phi\left(\Sigma_{\mathcal{T},N_{\mathcal{T}}}, \overline{\Sigma}_{\mathcal{T}}, \sigma(\gamma)\right)\right]^{\top} \in \mathbb{R}^{N_{\mathcal{T}} \times d(d+1)/2}$$

Test time optimization problem

Find the optimal parallel transport of the target domain:

$$\gamma_{\mathcal{T}}^{\star} = \underset{\gamma \in \mathbb{R}}{\arg\min} \left\{ \mathcal{L}_{\mathcal{T}}(\gamma) = \left(\bar{y}_{\mathcal{T}} - \frac{1}{N_{\mathcal{T}}} 1_{N_{\mathcal{T}}}^{\top} Z_{\mathcal{T}}(\gamma) \beta_{\mathcal{S}}^{\star}(\gamma_{\mathcal{S}}^{\star})\right)^2 \right\}$$

with $\beta_{\mathcal{S}}^{\star}$ the final regression coefficients of the train-time optimization.

1. Parallel transport the target covariance matrices
2. Vectorization and predicted output with pre-fitted linear regression:

$$\widehat{y}_{\mathcal{T}} = Z_{\mathcal{T}}(\gamma_{\mathcal{T}}^{\star}) \beta_{\mathcal{S}}^{\star}(\gamma_{\mathcal{S}}^{\star}) \in \mathbb{R}^{N_{\mathcal{T}}}$$

3. Comparison with the true mean output value

5. Empirical benchmarks

5. Empirical benchmarks: Baseline methods

- **DO Dummy**: always predict the mean value per domain
- **No DA**: all covariance matrices are projected to the tangent space at the source geometric mean computed from all source points, no matter their recording sites
- **Re-center** / **Re-scale**: all domains re-centered to I_d / dispersions set to 1
- **DO Intercept**: fitting one intercept $\square_0$ per domain
- **GREEN**: deep-learning architecture tailored for EEG applications ⟶ SPD network

[Paillard *et al.* 2024]

5. Empirical benchmarks: Simulated data

Simulated data and shifts

<u>Generative model:</u> $x_i(t) = As_i(t)$ $\Sigma_i = \mathbb{E}[x_i(t)x_i(t)^\top]$

with $x_i(t) \in \mathbb{R}^d$ the observed time series, $s_i(t) \in \mathbb{R}^d$ the underlying signal of the neural generators and A the mixing matrix

<u>Log-linear model:</u> $$y_i = \beta_0 + \sum_{l=1}^{d} \beta_l \log(p_{li})$$

with $p_{li} > 0$ the variance of the l-th element of $s_i(t)$

<u>Shift in the data:</u> $\Sigma_i \mapsto B_k^\xi \Sigma_i B_k^\xi$

with $B_k \in \mathbb{S}_d^{++}$

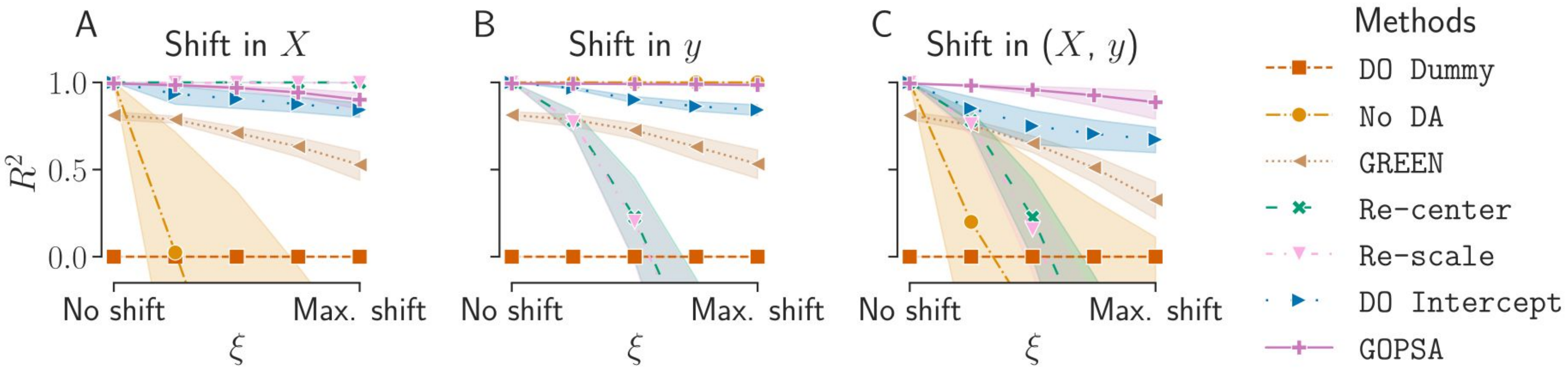
<u>Shift in the labels:</u> $p_{li} \mapsto p_{li}^{1+k\xi}$

$\xi \geq 0$ controls the amplitude of the shift

[Sabbagh *et al.* 2019, 2020]

5. Empirical benchmarks: Simulated data

- 6 domains generated: 5 source, 1 target.

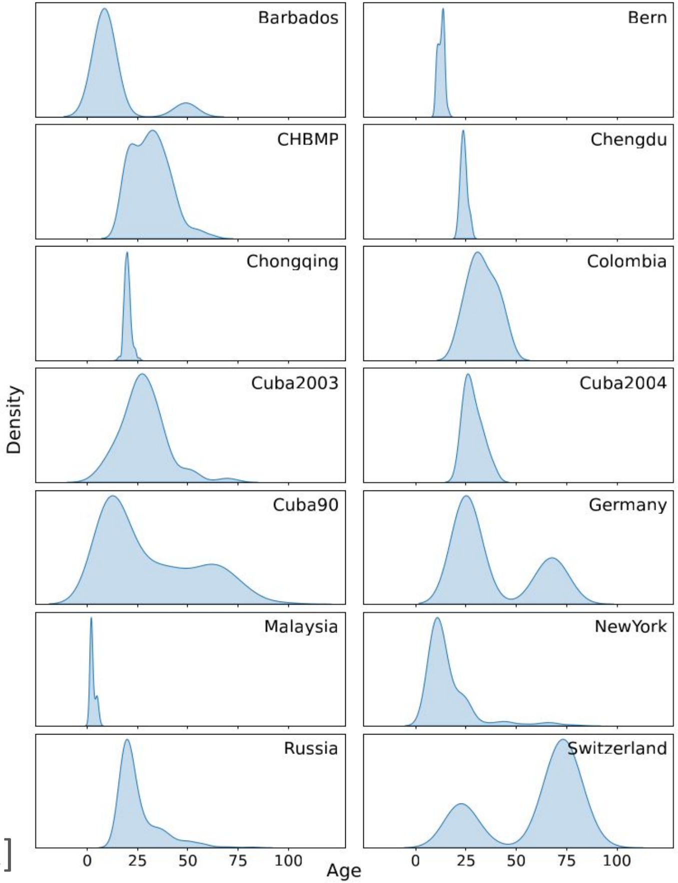


- No DA performs well with shifts in y only but fails when shifts in X are introduced
- Re-center correct shifts in X but perform poorly when shifts in y are added
- GOPSA outperforms other methods, handling both X and y shifts effectively

5. Empirical benchmarks: HarMNqEEG dataset

- 1564 participants - 14 studies - 9 countries
- Same montage with 19 channels
- No raw data →cross spectral matrices from 1.17Hz to 19.14Hz with a resolution of 0.39Hz
- Common artifact cleaning procedure
- Additional preprocessing:
  - Common Average Reference
  - Real part extraction
  - Shrinkage regularization

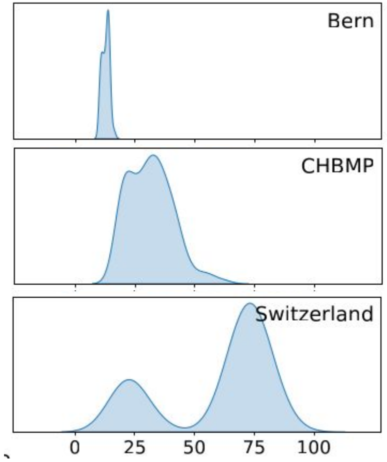
[Li *et al.* 2022]

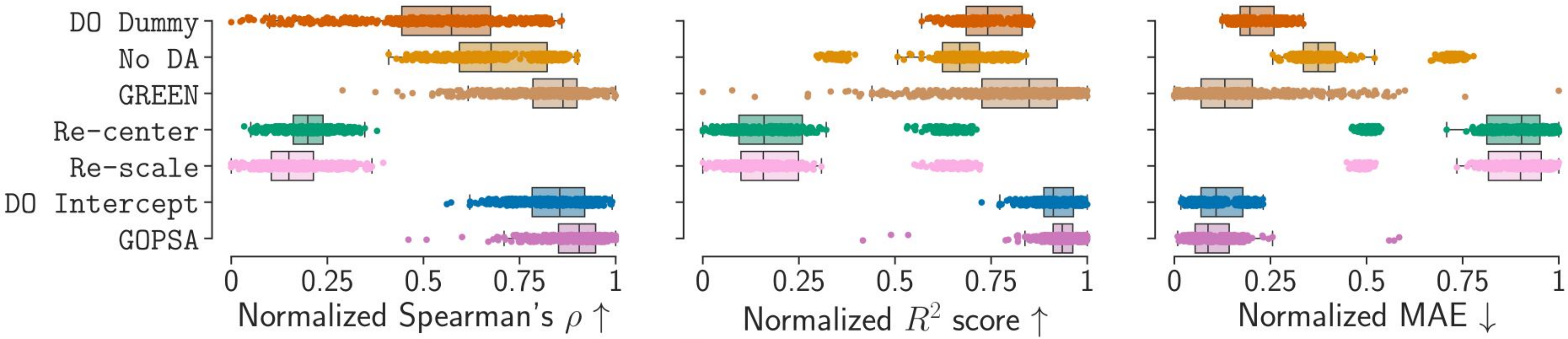


5. Empirical benchmarks: HarMNqEEG dataset

- 5 different sites combination
- Min-max normalization for each combination
- Re-center, Re-scale performed worse than Do Dummy and No DA
- GREEN performed better than No DA but with large variance
- DO Intercept and GOPSA showed similar performance

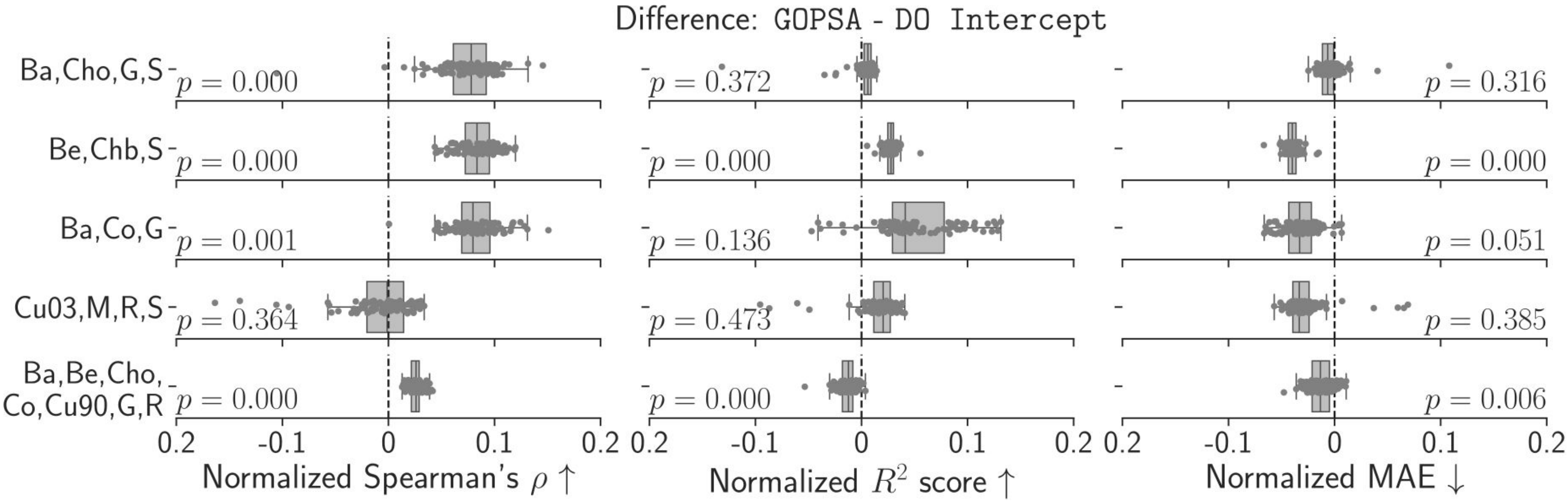
Example:





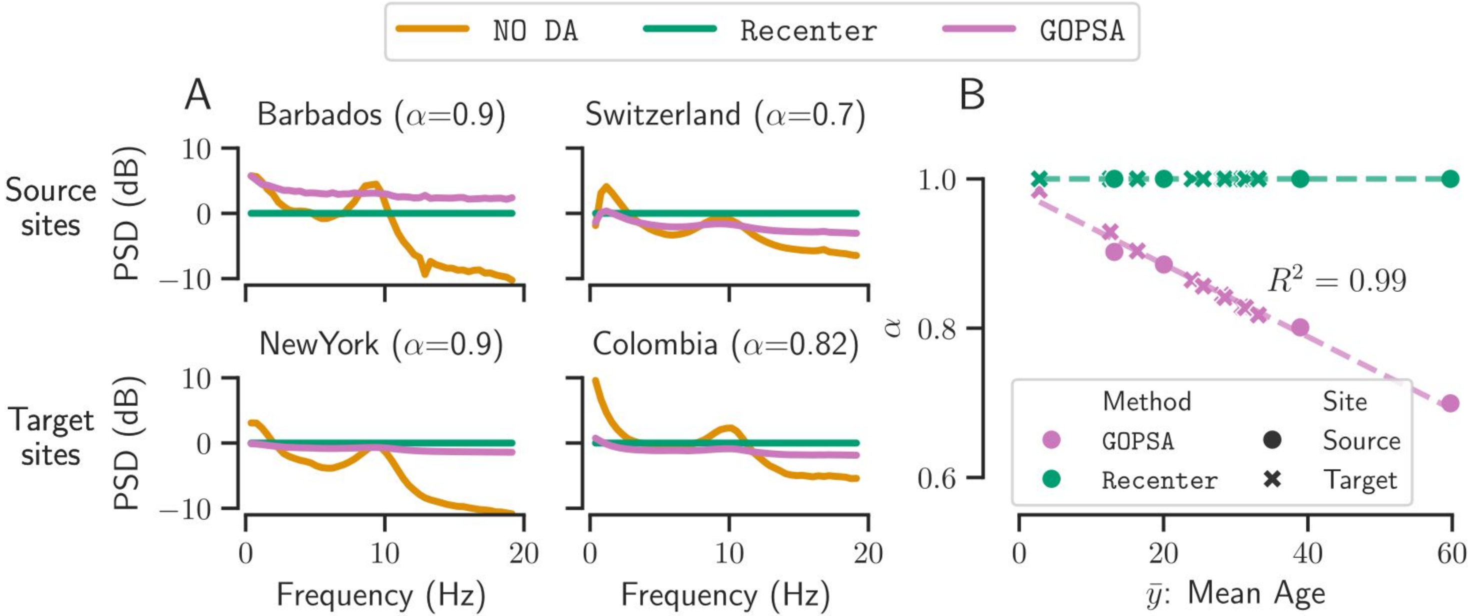
5. Empirical benchmarks: HarMNqEEG dataset

- Closer look at the difference between GOPSA and DO Intercept



- GOPSA significantly outperformed the baseline methods in some site combinations, but not all $\longrightarrow$ not all site combinations show joint (X, y) shifts

5. Empirical benchmarks: Model inspection



- No DA: high variability between sites
- Re-center flattens PSDs: too much information loss
- GOPSA harmonizes PSDs across sites + linear relationship between alpha values and age

6. Conclusion

- GOPSA handles joint shifts in X and y using by learning domain-specific re-centering operators and a global regression model.
- Achieved better performance on the HarMNqEEG dataset across multiple metrics in a majority of site combinations compared to baseline methods.
- Implementation using PyTorch which readily supports its inclusion in Riemannian deep learning models.

Reminder

You are all invited at my PhD defense on **Friday 8 November** at **10am**!