Geometric deep learning to bridge Riemannian transfer learning with end-to-end learning

Reinmar J. KOBLER[1,2], Motoaki KAWANABE[1,2]

[1]ATR, Kyoto, Japan

[2]RIKEN AIP, Tokyo, Japan

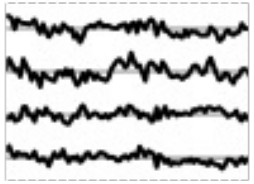
Goals

- Infer target $y_j \in \mathcal{Y}$ from brain activity (e.g., M/EEG) $\mathbf{X}_j \in \mathcal{X} \subset \mathbb{R}^{P \times Q}$
- Invariances: **sessions/subjects**/hardware/artifacts/**linear mixing/**…

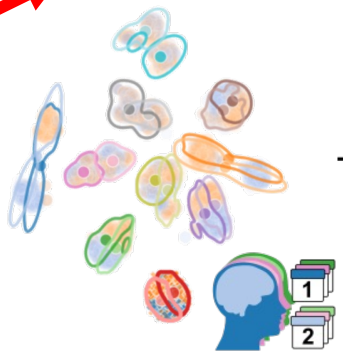
How to impose invariances in neural nets?

② Constrain latent space
(Riemannian geometry)

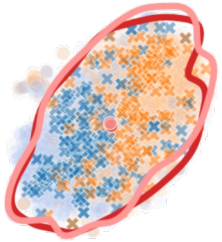
$$\mathbf{X}_j \in \mathcal{X} \subset \mathbb{R}^{P \times Q} \longrightarrow f_\theta \longrightarrow \mathbf{z}_j \in \mathcal{Z} \subset \mathbb{R}^{L} \longrightarrow g_\psi \longrightarrow y_j \in \mathcal{Y}$$



① Metric learning
(Pierre's talk)



$\longrightarrow$



③ Explicit alignment in
(input $\mathcal{X}$) or latent $\mathcal{Z}$ space

Explicit alignment in latent space $\mathcal{Z}$

Motivation:

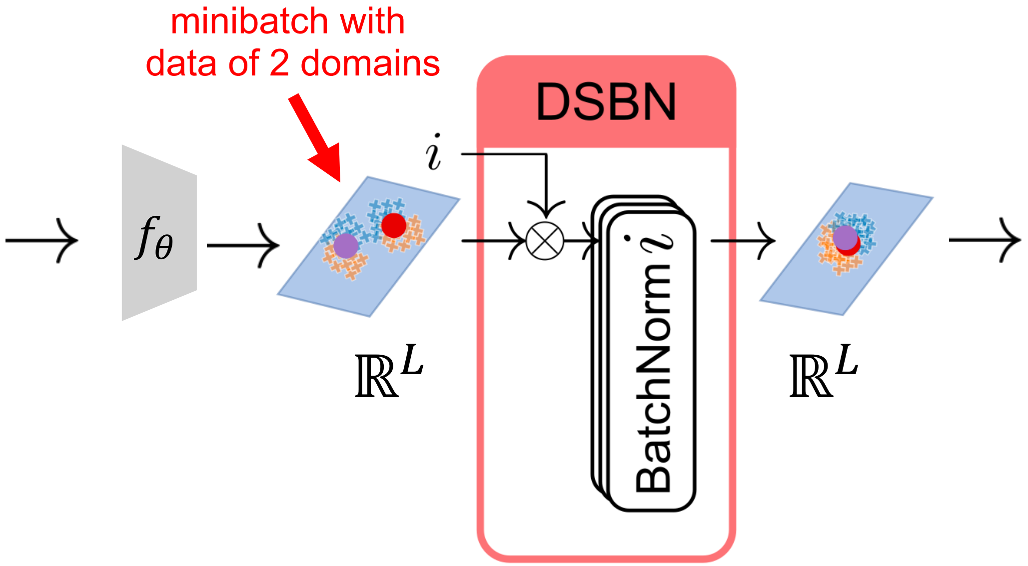
align mean and std along **relevant** (=discriminative) dimensions

Constraint:

backpropagate gradients

Domain-specific batch-normalization (DSBN) [Chang+2019,*CVPR*]

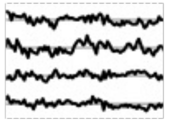
- Transfer to new domains
- Online unsupervised domain adaptation



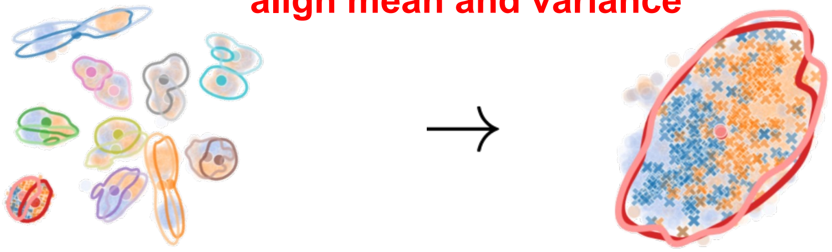
Effective for EEG: [Li+2021,*Methods*; Bakas+2022,*arXiv*]

Constrain latent space to SPD manifold $\mathcal{S}_d^+$

$$\mathbf{X}_j \in \mathcal{X} \longrightarrow \boxed{f_\theta} \longrightarrow \mathbf{z}_j \in \mathcal{Z} \subset \mathbb{R}^L \longrightarrow \boxed{\text{DSBN}} \longrightarrow \tilde{\mathbf{z}}_j \in \mathbb{R}^L \longrightarrow \boxed{g_\psi} \longrightarrow y_j \in \mathcal{Y}$$



align mean and variance



constrain latent space and align Fréchet mean and variance

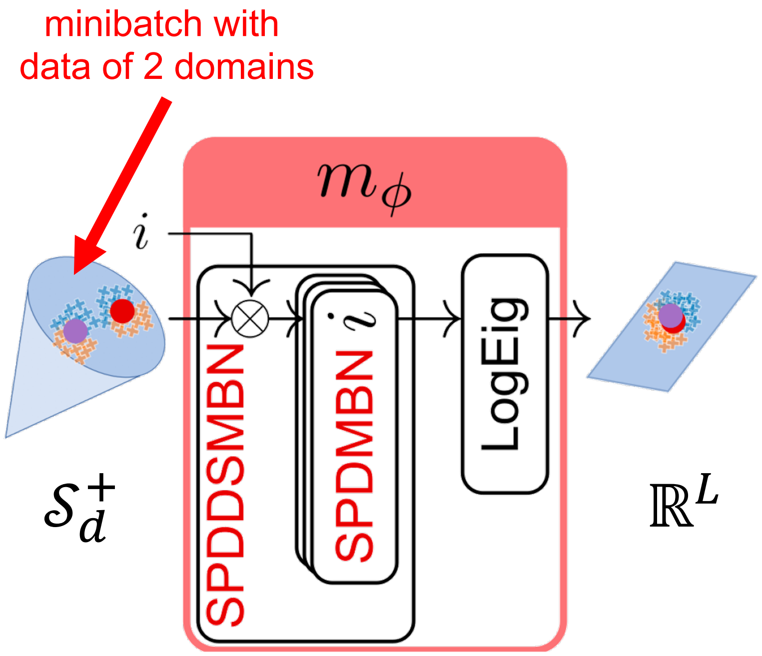
$$\mathbf{X}_j \in \mathcal{X} \longrightarrow \boxed{f'_\theta} \underbrace{\boxed{\text{Cov-Pool}}\ \boxed{\text{ReEig}}}_{\substack{\text{SPDNet layers} \\ \text{[Huang\&Gool2017,}\textit{AAAI}\text{]}}} \longrightarrow \mathbf{Z}_j \in \mathcal{S}_d^+ \longrightarrow \underbrace{\boxed{m_\phi}}_{\substack{\text{alignment and} \\ \text{tangent space projection}}} \longrightarrow \tilde{\mathbf{z}}_j \in \mathbb{R}^L \longrightarrow \boxed{g_\psi} \longrightarrow y_j \in \mathcal{Y}$$

Alignment on $\mathcal{S}_d^+$ and projection to $\mathbb{R}^L$

[Kobler+2022,*NeurIPS*]

Alignment

- New layer (SPDDSMBN) that combines ideas of DSBN [Chang+2019,*CVPR*], momentum batch norm [Yong+2020,*ECCV*] and SPDBN [Kobler+2022,*ICASSP*]
- Exponential smoothing to track domain's Fréchet mean $\mathbf{G}_i$ and variance ν_i across minibatches
- Standardize and re-bias each domain's data so that the Fréchet mean $\mathbf{G}_i \to \mathbf{I}$ and variance $\nu_i \to \nu_\phi \in \mathbb{R}^+$ (learnable parameter).



Projection: LogEig layer [Huang&Gool2017,*AAAI*]

Properties of m_ϕ:

- Bridge to tangent space mapping (TSM) models on $(\mathcal{S}_d^+, \delta_{\text{AIRM}})$ [Barachant+2012, *TBME*]
- Invariance to shared affine mixing

TSM in a nutshell

Setting

Riemannian manifold $(\mathcal{S}_d^+, \delta_{\mathrm{AIRM}})$

$$\mathcal{S}_d^+ = \{\mathbf{Z} \in \mathbb{R}^{d\times d} : \mathbf{Z} = \mathbf{Z}^T, \mathbf{Z} \succ 0\}$$

$$\delta_{\mathrm{AIRM}}(\mathbf{Z}_1, \mathbf{Z}_2) = \left\| \mathrm{Log}\left(\mathbf{Z}_1^{-\frac{1}{2}} \mathbf{Z}_2 \mathbf{Z}_1^{-\frac{1}{2}}\right) \right\|_F$$

For all invertible $\mathbf{A}$:

$$\delta_{\mathrm{AIRM}}(\mathbf{A}\mathbf{Z}_1\mathbf{A}^T, \mathbf{A}\mathbf{Z}_2\mathbf{A}^T) = \delta_{\mathrm{AIRM}}(\mathbf{Z}_1, \mathbf{Z}_2)$$

N observations from a single domain

$$\mathcal{D}_i = \{(\mathbf{X}_j, y_j) \mid j = 1, \ldots, N\}$$

Limitations

Expensive to compute Fréchet mean $\mathbf{G}$ and variance ν.
⇒ $\{\theta, \phi, \psi\}$ are typically fitted sequentially

Typical tangent space mapping (TSM) models

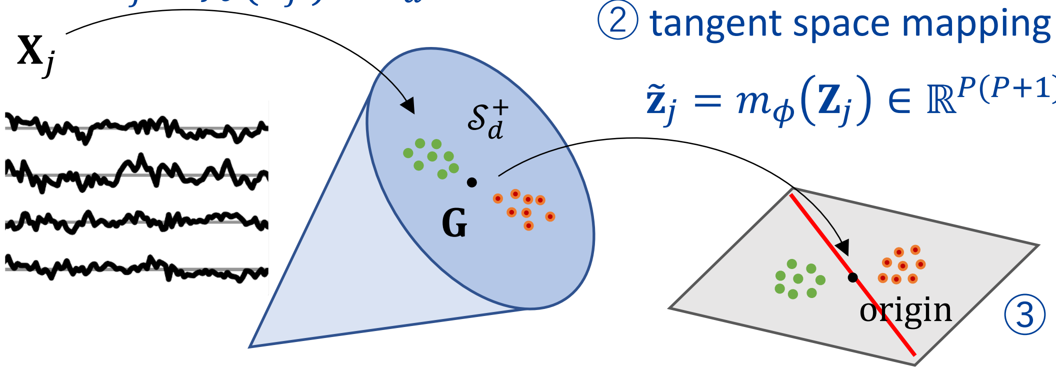
① feature extraction

$$\mathbf{Z}_j = f_\theta(\mathbf{X}_j) \in \mathcal{S}_d^+$$

$$m_{\boldsymbol{\phi}}(\mathbf{Z}_j) = \mathrm{upper} \circ \Gamma_{\mathbf{G}\to\mathbf{I}} \circ \mathrm{Log}_{\mathbf{G}}(\mathbf{Z}_j) = \mathrm{upper} \circ \mathrm{Log}\left(\mathbf{G}^{-\frac{1}{2}} \mathbf{Z}_j \mathbf{G}^{-\frac{1}{2}}\right)$$

② tangent space mapping

$$\tilde{\mathbf{z}}_j = m_\phi(\mathbf{Z}_j) \in \mathbb{R}^{P(P+1)/2}$$



③ classification

$$y_j = g_\psi(\mathbf{v}_j)$$

Fréchet mean $\mathbf{G}$ and variance ν

$$\mathbf{G} = \arg\min_{\mathbf{Z}\in\mathcal{S}_P^+} L(\mathbf{Z}) \qquad \nu = L(\mathbf{G})$$

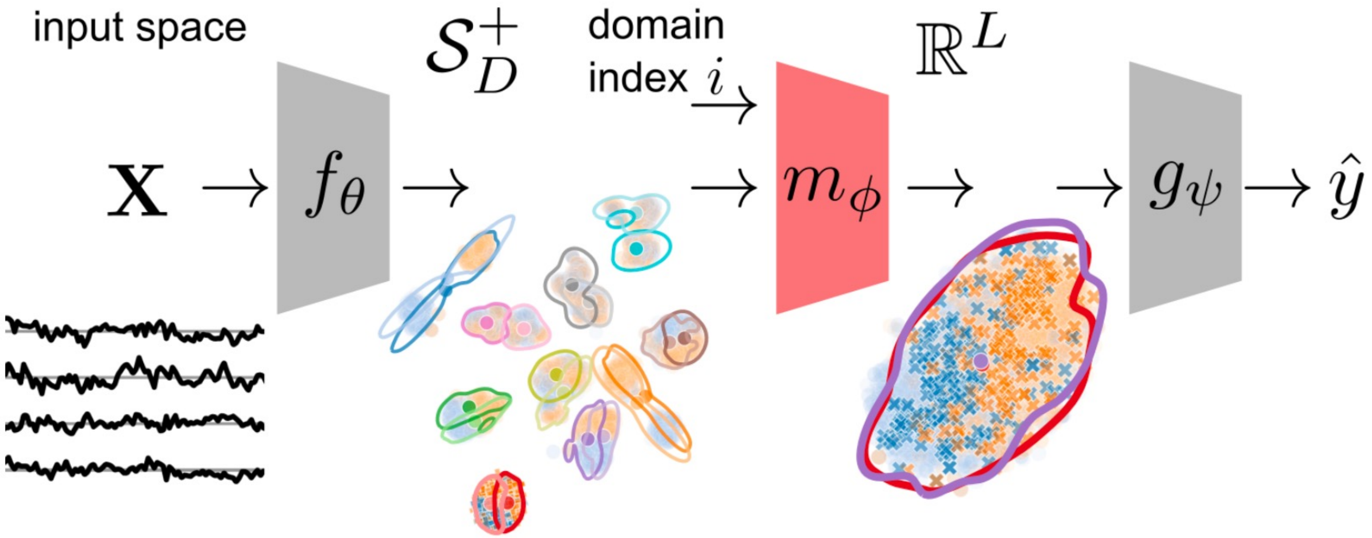
$$L(\mathbf{Z}) = \frac{1}{N}\sum_{j=1}^{N} \delta_{\mathrm{AIRM}}^2(\mathbf{Z}, \mathbf{Z}_j)$$

Tangent space distances are…

- … cheap to compute
- … locally approximate δ_{AIRM}
  ⇒ inherit *invariance* properties

TSMNet: learning TSM end-to-end

[Kobler+2022,*NeurIPS*]



Training:

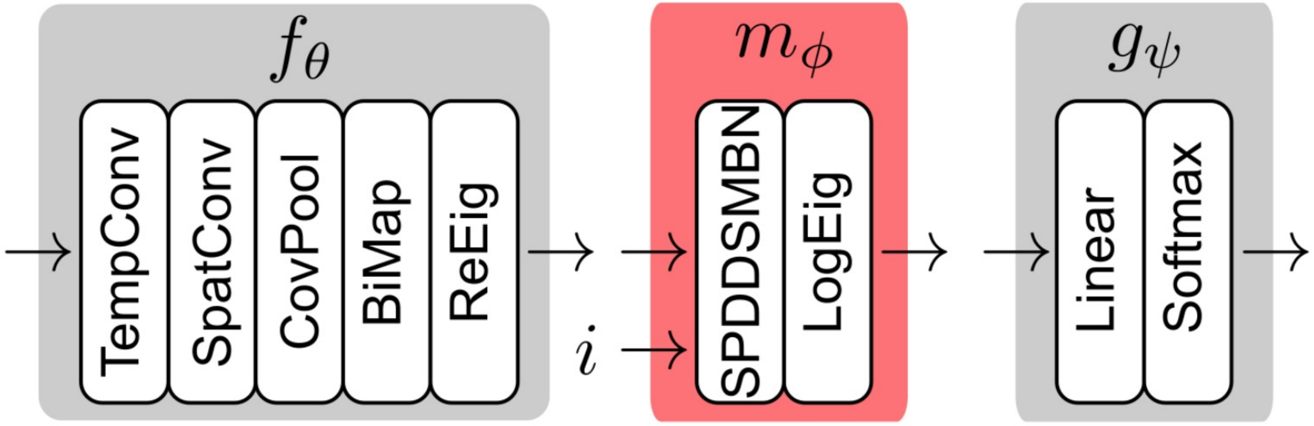
- **Loss**: cross-entropy error (classification setting)
- **Strategy**: minibatch-based gradient descent
- **Optimizer**: Riemannian ADAM [Becigneul+2019,*ICML*]

Inference: keep learnable parameters fixed $\{\theta,\phi,\psi\}$

- Source domains: use learned Fréchet mean and variance
- New domains: latent space projection + estimate Fréchet mean and variance
- Online: update Fréchet mean and variance iteratively [**poster on Friday**]

TSMNet: parametrization and interpretation

[Kobler+2022,*NeurIPS*]



linear + temporal covariance TSM **linear**

Equivalent to typical TSM [Barachant+2012, *TBME*]

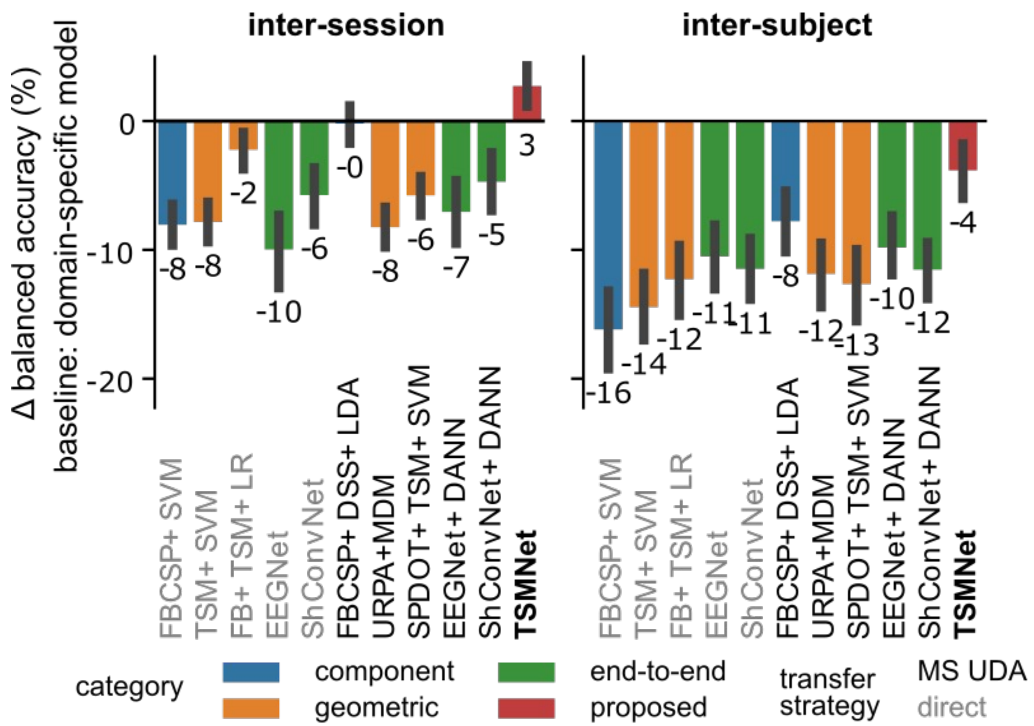
⇒ **global model interpretation feasible** (i.e., $\{\theta,\phi,\psi\} \rightarrow$ spatial and spectral patterns) [Kobler+2021, *EMBC*]

Results: motor imagery

[Kobler+2022,*NeurIPS*]

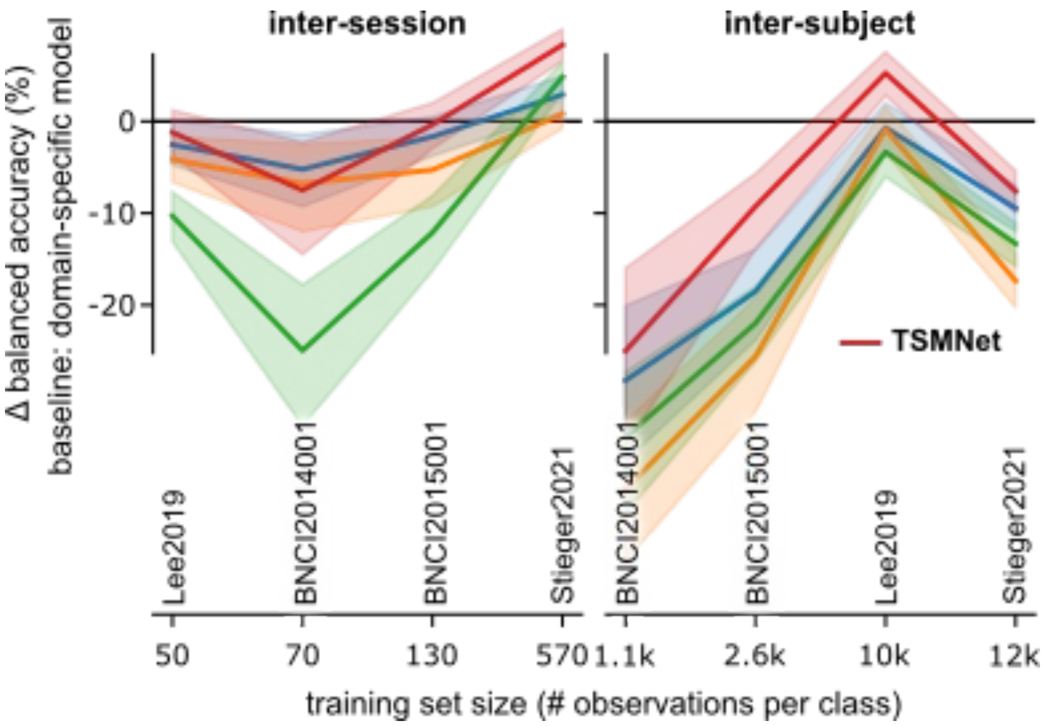
Grand average

5 dataset, 573 sessions, 138 subjects



Performance over dataset size

Small to large-scale MI datasets



Results: BCNI2014001 dataset

[Kobler+2022,*NeurIPS*]

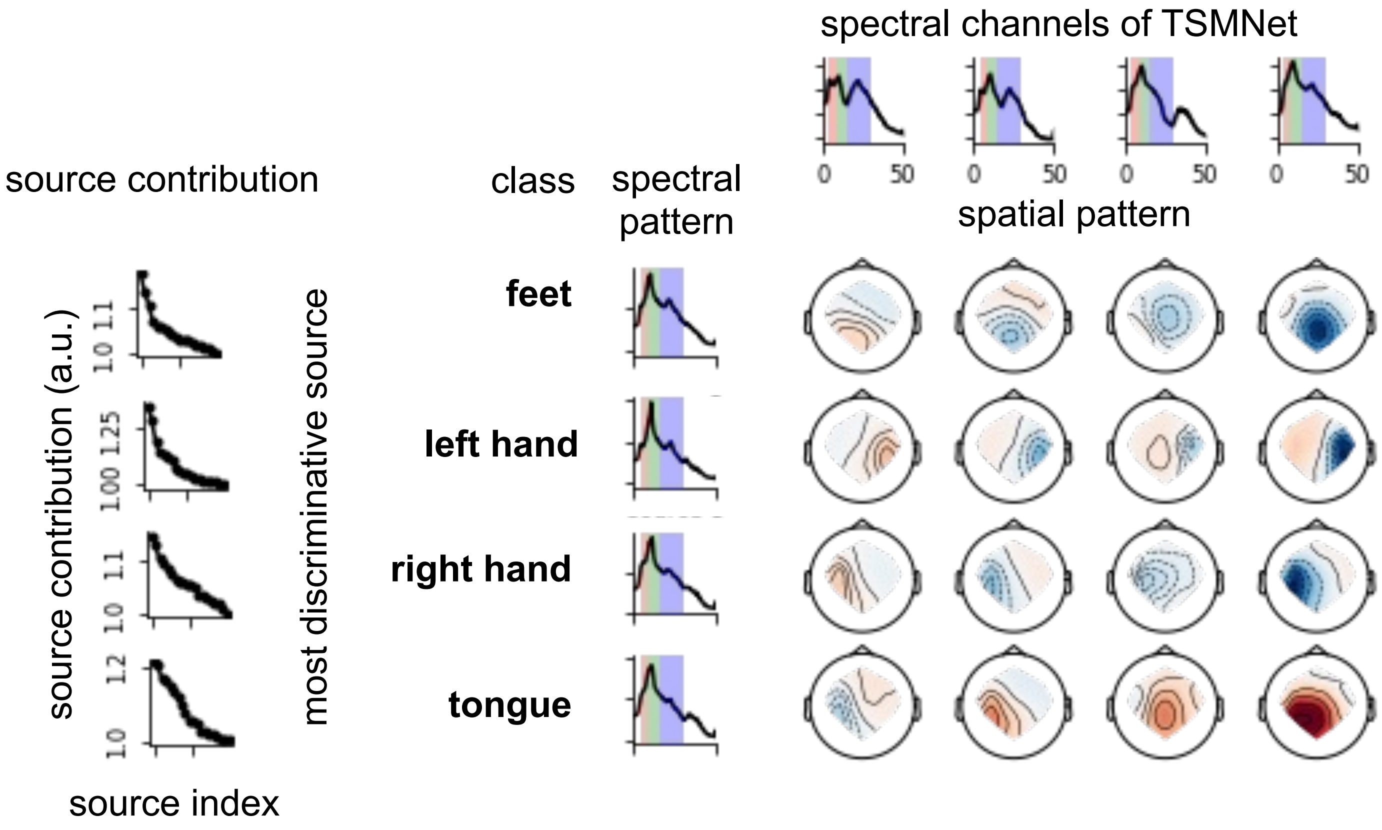
 # Parameters

inter-subject transfer

model	# parameters
TSMNet	5099
EEGNet	3660
EEGNet+DANN	10302
ShConvNet	43404
ShConvNet+DANN	75102

Model interpretation

Patterns for TSMNet fitted to session 1 of subject 9.



https://github.com/rkobler/TSMNet

Summary

Highlights

- Bridge TSM on $(\mathcal{S}_d^+, \delta_{\mathrm{AIRM}})$ and deep learning
- SoA TL performance (small and large datasets)
  AND global model interpretation
- Seamless transfer to new domains and online adaptation

Limitations

- Computations (matrix logarithms and powers)
- Similar class priors across domains

https://github.com/rkobler/TSMNet

Acknowledgments

Acknowledgments
This work was supported by Innovative Science and Technology Initiative for Security Grant Number JPJ004596, ATLA, Japan.

References

Chang et al., “Domain-Specific Batch Normalization for Unsupervised Domain Adaptation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*,2019.

Li et al. “Subject matching for cross-subject EEG-based recognition of driver states related to situation awareness,” *Methods*, vol. 202, pp. 136–143, 2022.

Bakas *et al.*, “Team Cogitat at NeurIPS 2021: Benchmarks for EEG Transfer Learning Competition.” arXiv, 2022.

Huang and Gool, “A Riemannian Network for SPD Matrix Learning,” in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

Yong et al., “Momentum Batch Normalization for Deep Learning with Small Batch Size,” in *Computer Vision – ECCV,* 2020.

Barachant et al., “Multiclass brain-computer interface classification by Riemannian geometry.,” *IEEE transactions on bio-medical engineering*, vol. 59, no. 4, pp. 920–928, 2012.

Becigneul et al., “Riemannian Adaptive Optimization Methods,” in *International Conference on Learning Representations*, 2019.

Kobler et al., “On the interpretation of linear Riemannian tangent space model parameters in M/EEG,” in *Proceedings of the 43rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2021.

Kobler et al., “Controlling The Fréchet Variance Improves Batch Normalization on the Symmetric Positive Definite Manifold,” in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.

Kobler et al., “SPD domain-specific batch normalization to crack interpretable unsupervised domain adaptation in EEG,” in *Advances in Neural Information Processing Systems*, 2022.