

Multi-Source Domain Adaptation for Learning on Biosignals

Adaptation de Domaine Multi-Sources pour l'Apprentissage sur
les Bio-Signaux

Théo Gnassounou

Jury: Marteen De Vos, Nicolas Thome, Agnès Desolneux, Alain Rakotomamonjy,
Romain Tavenard

Introduction

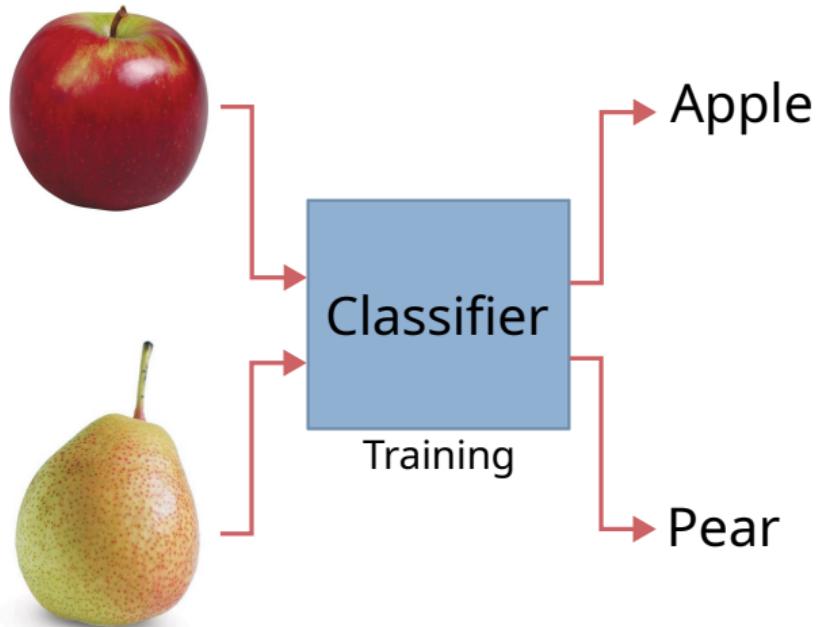
SKADA & Benchmark

Monge mapping for
Biosignals

Introduction to Domain Adaptation

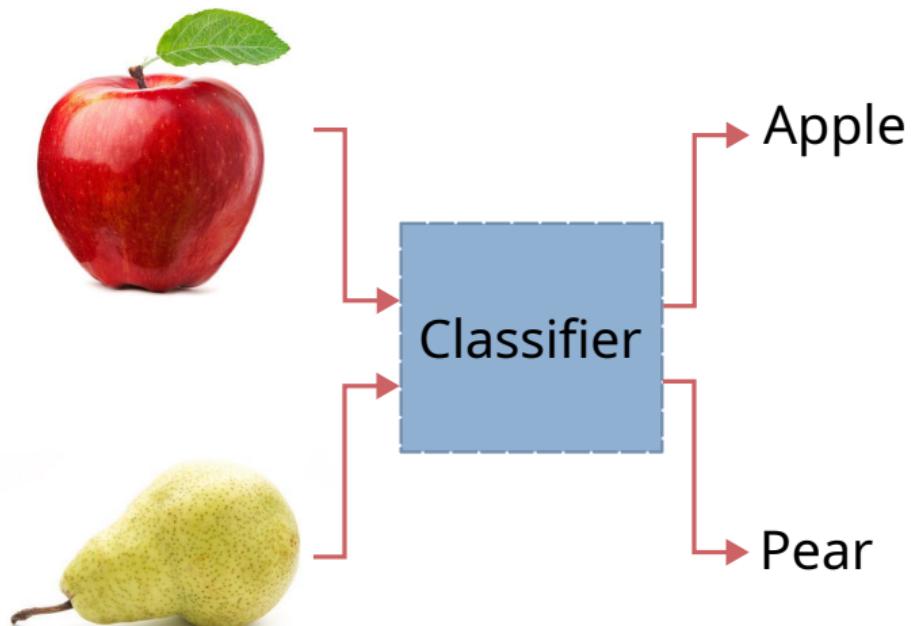
Machine Learning: a powerful tool

Le machine learning permet, par exemple, d'apprendre à un modèle à reconnaître des pommes et des poires.



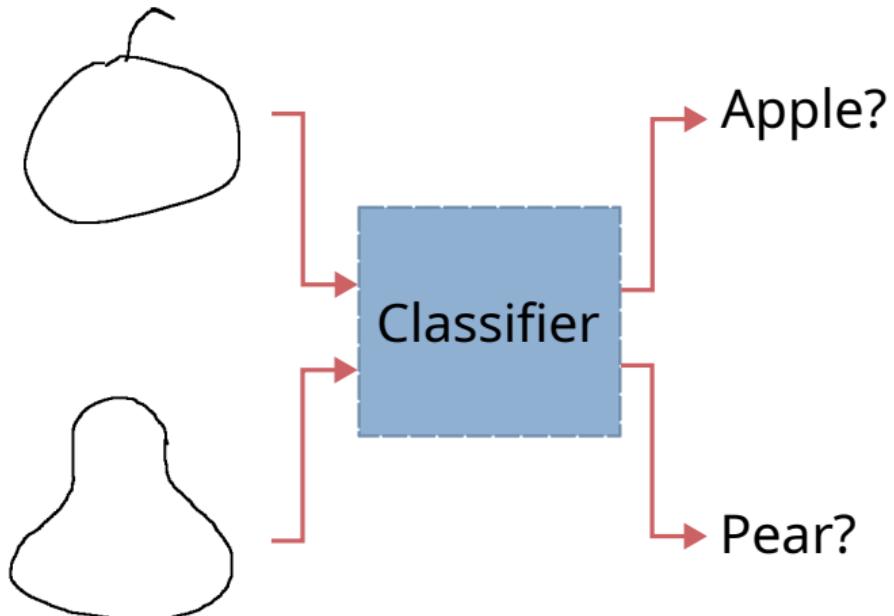
Machine Learning: a powerful tool

Le modèle peut ensuite reconnaître de nouvelles images de pommes et poires.



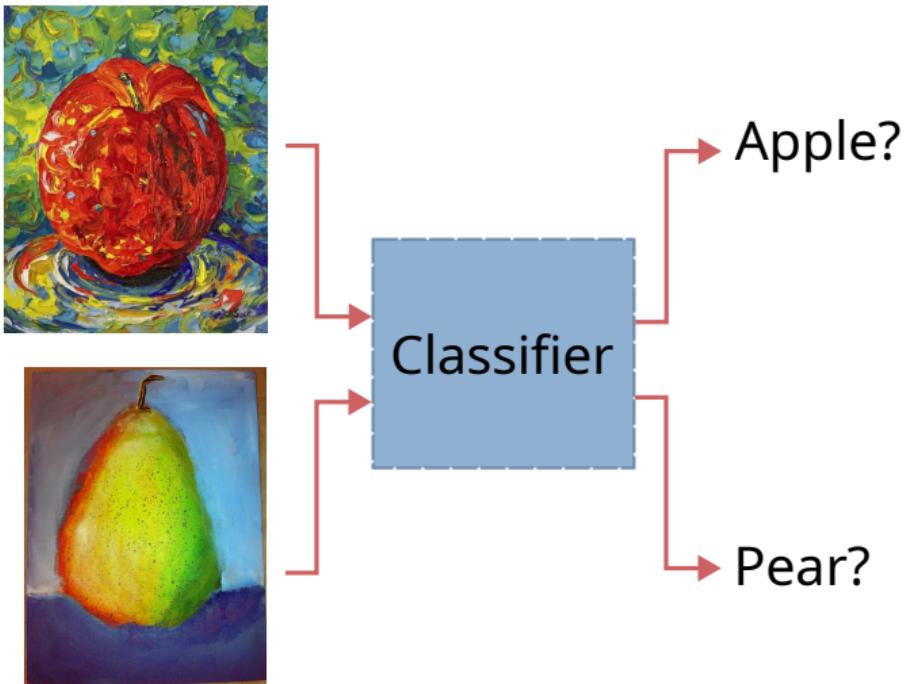
Machine Learning: but shift happens ...

Cependant, le modèle qui n'a pas été entraîné sur des images de pommes et poires peintes peut avoir des difficultés à les reconnaître.

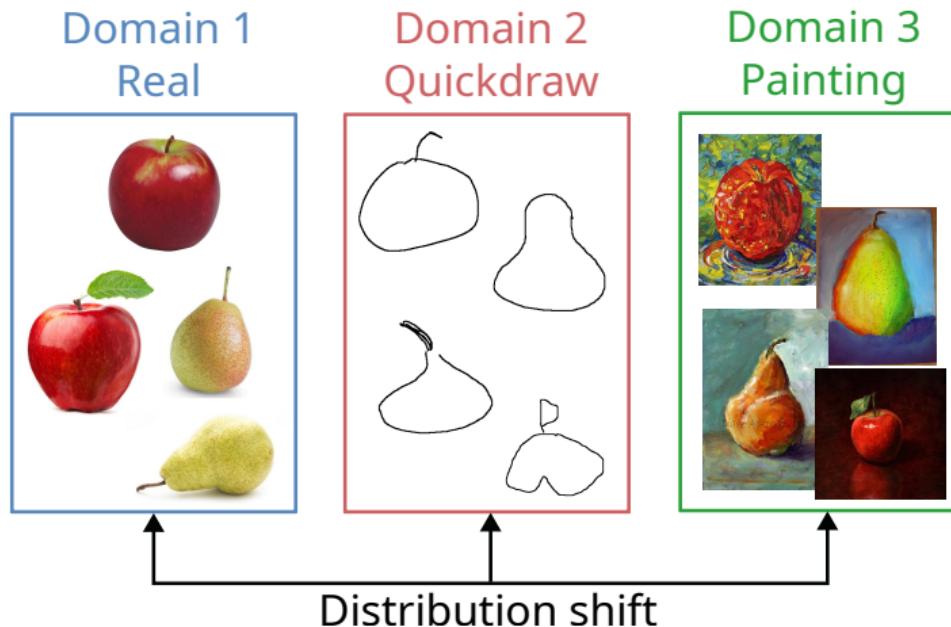


Machine Learning: but shift happens ...

Pareil pour des peintures de pommes et de poires.

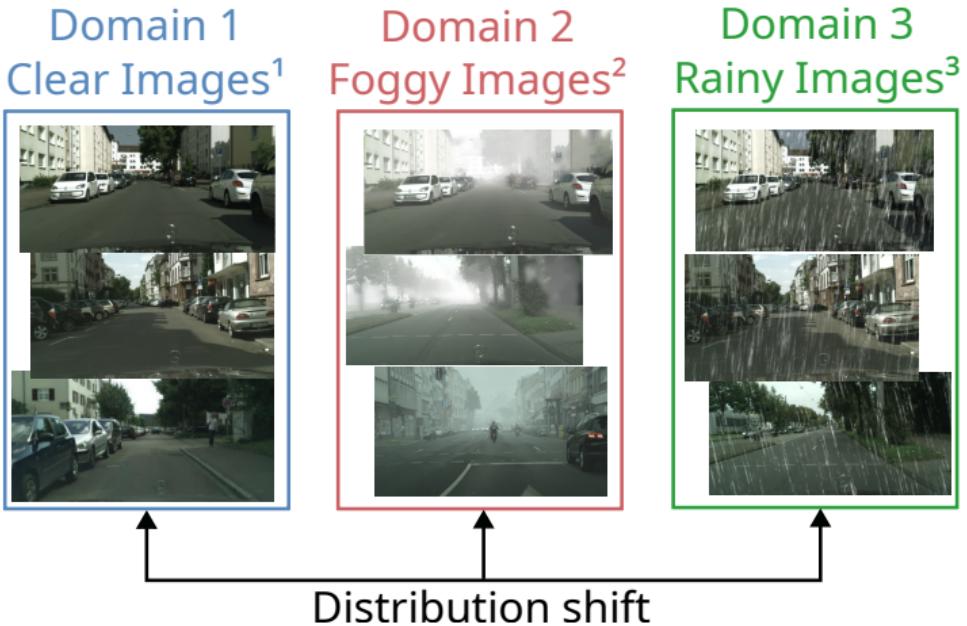


Distribution shift in Image classification¹



¹ Dataset: *DomainNet* (Peng et. al., 2019)

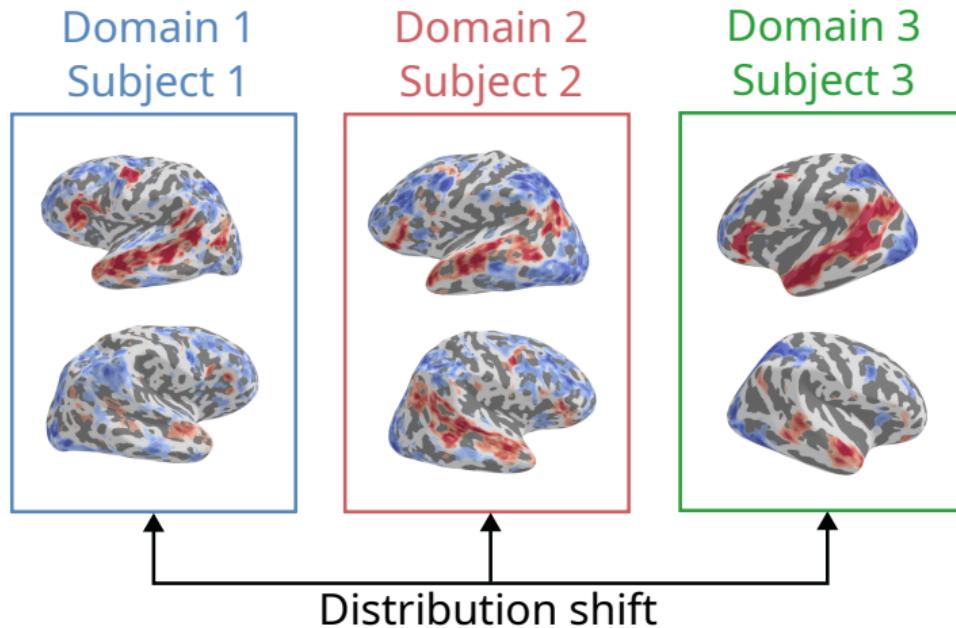
Distribution shift in Autonomous Driving



Dataset: ¹*Cityscapes (Cordts et. al., 2016)*

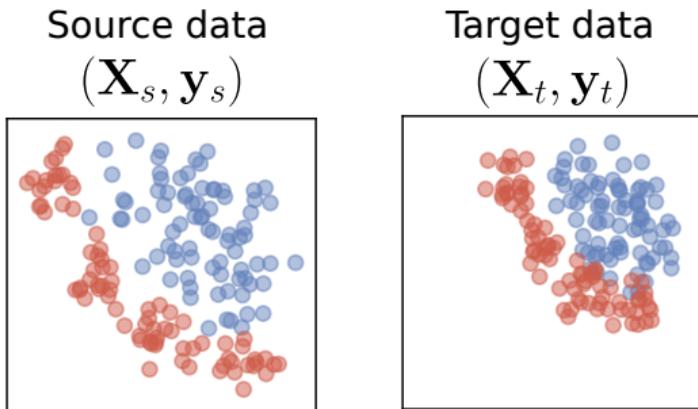
²*Foggy Cityscapes (Sakaridis et. al., 2018)* ³*Rainy Cityscapes (Li et. al., 2024)*

Distribution shift in Functional MRI¹



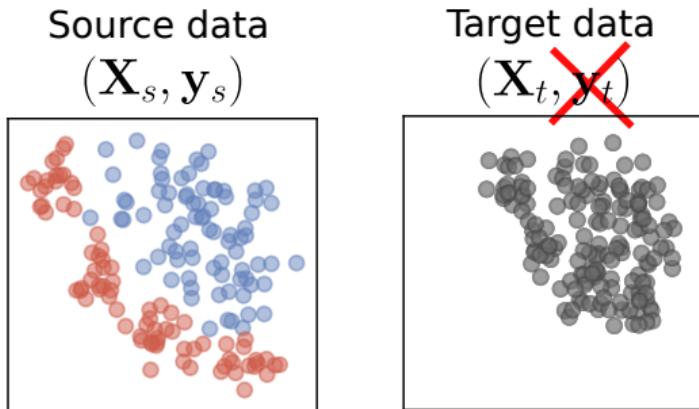
¹ Dataset: IBC Project (Pinho et. al., 2018)

Impact of Distribution Shift



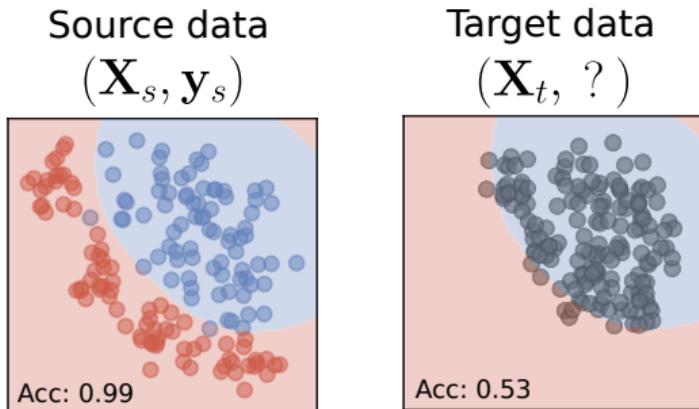
- Suppose classification problem with two classes (**blue** and **red**).
- Source domain (left): **labeled** data.

Impact of Distribution Shift



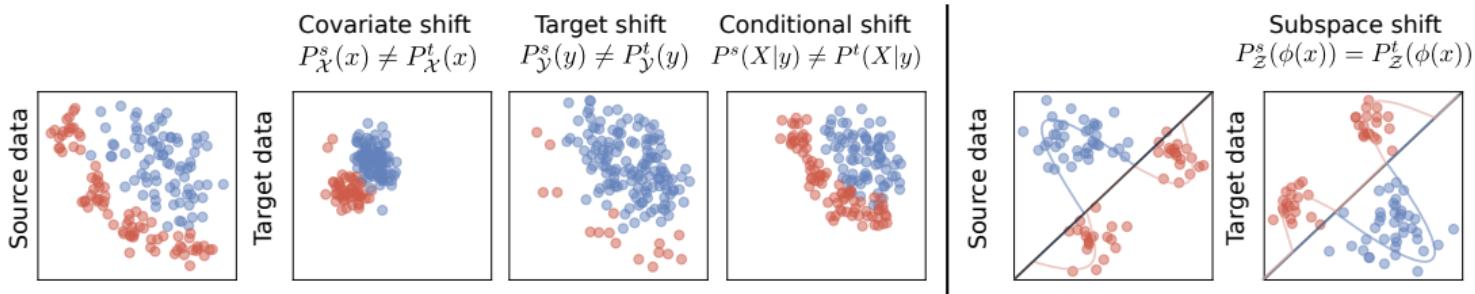
- Suppose classification problem with two classes (**blue** and **red**).
- Source domain (left): **labeled** data.
- Target domain (right): **unlabeled** data.

Impact of Distribution Shift



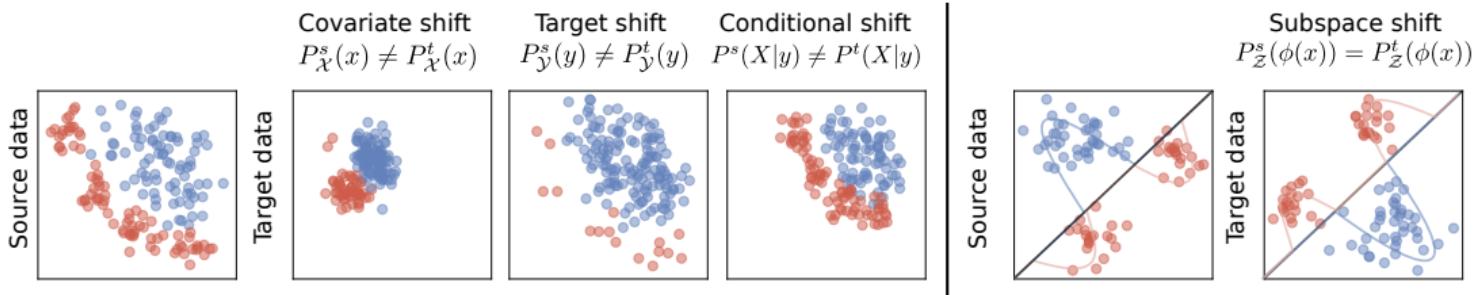
- Suppose classification problem with two classes (**blue** and **red**).
- Source domain (left): **labeled** data.
- Target domain (right): **unlabeled** data.
- Drop in performance when applying source classifier to target data.

Existing Distribution Shifts



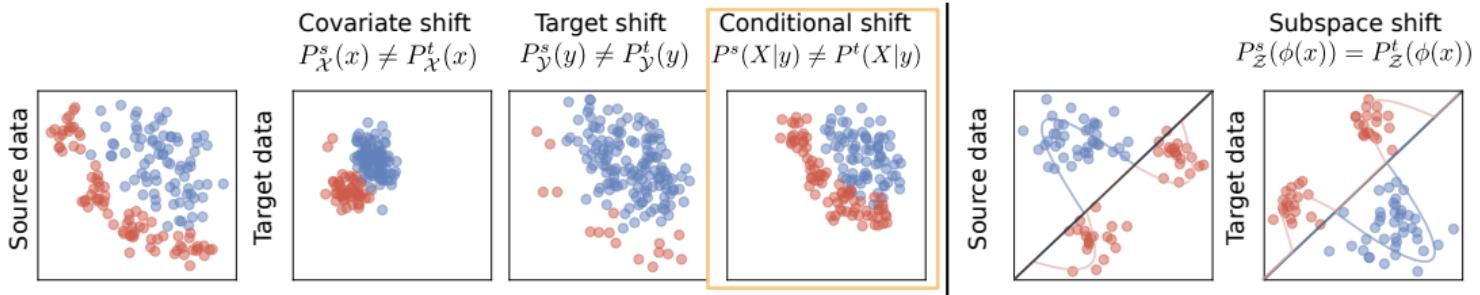
- **Covariate shift** : Distribution of **data** changes over domains.
- **Target shift** : Distribution of **labels** changes over domains.
- **Conditional shift** : Distribution of **data conditioned on labels** changes over domains.
- **Subspace Assumption** : Exist a **subspace that is invariant** over domains.

Tackling Shift with Domain Adaptation (DA)



- **Covariate shift** → Reweighting methods
- **Target shift** → Reweighting methods
- **Conditional shift** → Mapping methods
- **Subspace Assumption** → Subspace methods + Deep Learning Methods

Tackling Shift with Domain Adaptation (DA)

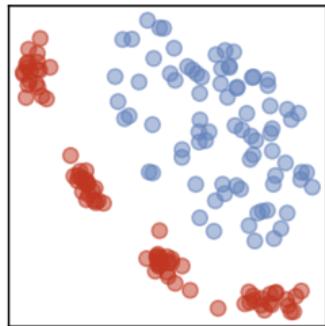


- **Covariate shift** → Reweighting methods
- **Target shift** → Reweighting methods
- **Conditional shift** → Mapping methods
- **Subspace Assumption** → Subspace methods + Deep Learning Methods

Example of Mapping DA method

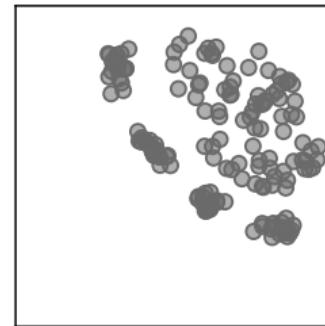
Source data

$$(\mathbf{X}_s, \mathbf{y}_s)$$



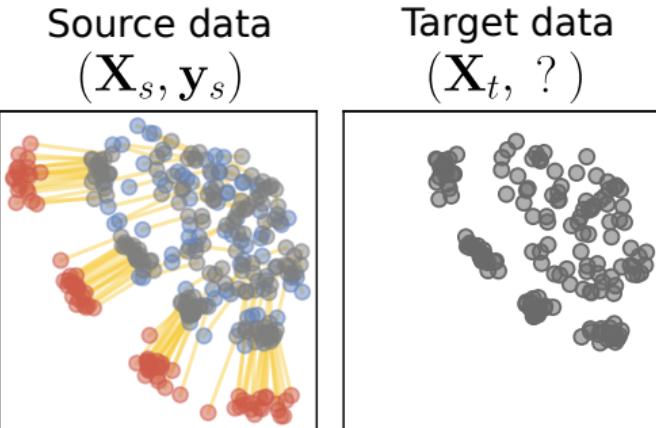
Target data

$$(\mathbf{X}_t, ?)$$



- Goal: Estimates a **mapping** from source to target domain.

Example of Mapping DA method



- Goal: Estimates a **mapping** from source to target domain.
- **Covariance mapping**¹:

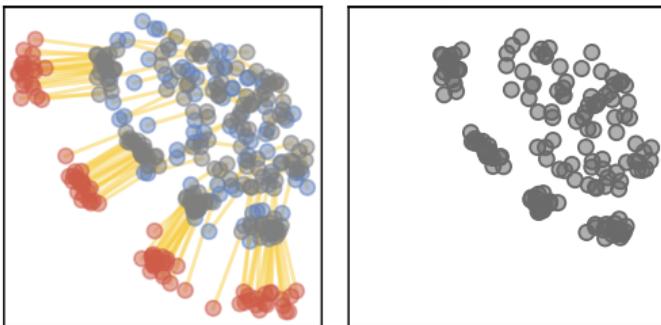
$$m(\mathbf{X}_s^i) = \boldsymbol{\Sigma}_t^{1/2} \boldsymbol{\Sigma}_s^{-1/2} \mathbf{X}_s^i$$

¹ Sun et. al., 2017

Example of Mapping DA method

Source data
 $(\mathbf{X}_s, \mathbf{y}_s)$

Target data
 $(\mathbf{X}_t, ?)$

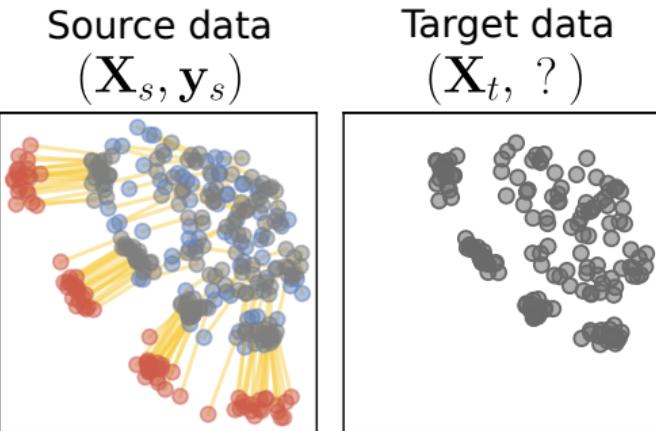


- Goal: Estimates a **mapping** from source to target domain.
- **Optimal Transport**²:

$$m(\mathbf{X}_s^i) = \sum_j \mathbf{T}_{ij} \mathbf{X}_t^j \quad \text{with} \quad \mathbf{T} = \underset{\mathbf{T}}{\operatorname{argmin}} \langle \mathbf{C}, \mathbf{T} \rangle_F$$

²Peyré et. al., 2020

Example of Mapping DA method

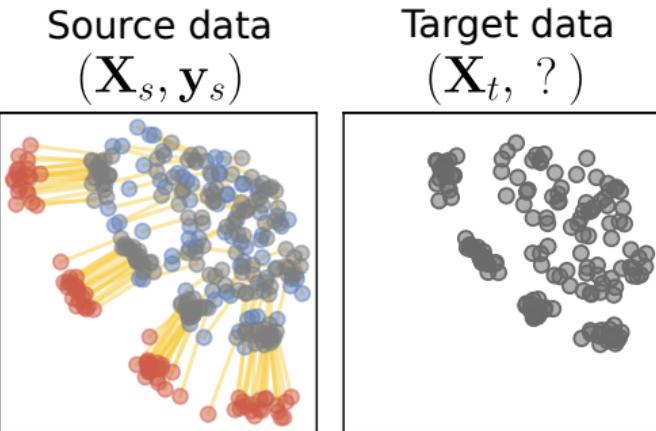


- Goal: Estimates a **mapping** from source to target domain.
- **Entropic Optimal Transport**³:

$$m(\mathbf{X}_s^i) = \sum_j \mathbf{T}_{ij} \mathbf{X}_t^j \quad \text{with} \quad \mathbf{T} = \operatorname{argmin}_{\mathbf{T}} \langle \mathbf{C}, \mathbf{T} \rangle_F + \lambda \sum_{i,j} \mathbf{T}_{ij} \log(\mathbf{T}_{ij} - 1)$$

³Cuturi et. al., 2013

Example of Mapping DA method



- Goal: Estimates a **mapping** from source to target domain.
- **Linear Optimal Transport**⁴:

$$m(\mathbf{X}_s^i) = \mathbf{A} \mathbf{X}_s^i \quad \text{with} \quad \mathbf{A} = \boldsymbol{\Sigma}_s^{-\frac{1}{2}} \left(\boldsymbol{\Sigma}_s^{\frac{1}{2}} \boldsymbol{\Sigma}_t \boldsymbol{\Sigma}_s^{\frac{1}{2}} \right)^{\frac{1}{2}} \boldsymbol{\Sigma}_s^{-\frac{1}{2}}$$

⁴Bhatia et. al., 2019

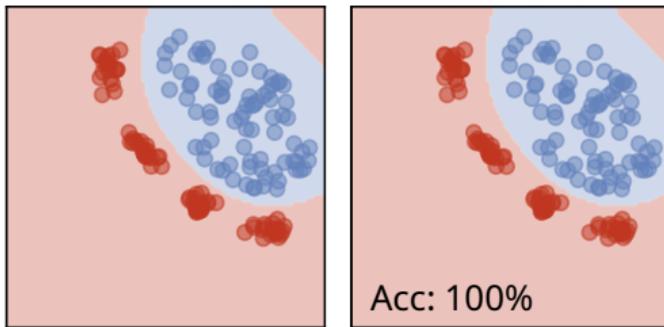
Example of Mapping DA method

Source data

$$(\mathbf{X}_s, \mathbf{y}_s)$$

Target data

$$(\mathbf{X}_t, \hat{\mathbf{y}}_t)$$



- Goal: Estimates a **mapping** from source to target domain.
- **Linear Optimal Transport**⁴:

$$m(\mathbf{X}_s^i) = \mathbf{A} \mathbf{X}_s^i \quad \text{with} \quad \mathbf{A} = \boldsymbol{\Sigma}_s^{-\frac{1}{2}} \left(\boldsymbol{\Sigma}_s^{\frac{1}{2}} \boldsymbol{\Sigma}_t \boldsymbol{\Sigma}_s^{\frac{1}{2}} \right)^{\frac{1}{2}} \boldsymbol{\Sigma}_s^{-\frac{1}{2}}$$

- Train the classifier on mapped source data & apply it to target data.

⁴Bhatia et. al., 2019

Challenges in Domain Adaptation

Lack of realistic and reproducible benchmarks in DA

	Method
Reweighting	Density Reweight
	Discriminative Reweight
	Gaussian Reweight
	KLIEP
	KMM
	NN Reweight
Mapping	MMDTarS
	Coral
	OT mapping
	Lin. OT mapping
Subsp.	MMD-LS
	SA
	TCA
Other	TSL
	JDOT
	OT label prop
	DASVM

Lack of realistic and reproducible benchmarks in DA

Method	
Reweighting	Density Reweight
	Discriminative Reweight
	Gaussian Reweight
	KLIEP
	KMM
	NN Reweight
Mapping	MMDTarS
	Coral
	OT mapping
	Lin. OT mapping
Subsp.	MMD-LS
	SA
	TCA
Other	TSL
	JDOT
	OT label prop
	DASVM

1. Many existing DA methods in the literature. and few open-source implementations:

- Introduce **SKADA**, a new open-source DA library¹.

¹ Gnassounou et. al. 2024

Lack of realistic and reproducible benchmarks in DA

	Method	Validation Procedure
Reweighting	Density Reweight	None
	Discriminative Reweight	NA
	Gaussian Reweight	None
	KLIEP	Integrated CV
	KMM	None
	NN Reweight	None
Mapping	MMDTarS	CV
	Coral	NA
	OT mapping	CV target/CircCV
	Lin. OT mapping	NA
Subsp.	MMD-LS	CV
	SA	2-fold CV on source
	TCA	Validation on target
Other	TSL	None
	JDOT	Reverse CV
	OT label prop	NA
	DASVM	Circular Validation

1. Many existing DA methods in the literature, and few open-source implementations:

- Introduce **SKADA**, a new open-source DA library¹.

¹ Gnassounou et. al. 2024

Lack of realistic and reproducible benchmarks in DA

	Method	Validation Procedure
Reweighting	Density Reweight	None
	Discriminative Reweight	NA
	Gaussian Reweight	None
	KLIEP	Integrated CV
	KMM	None
	NN Reweight	None
Mapping	MMDTarS	CV
	Coral	NA
	OT mapping	CV target/CircCV
	Lin. OT mapping	NA
Subsp.	MMD-LS	CV
	SA	2-fold CV on source
	TCA	Validation on target
Other	TSL	None
	JDOT	Reverse CV
	OT label prop	NA
	DASVM	Circular Validation

1. Many existing DA methods in the literature. and few open-source implementations:
 - Introduce **SKADA**, a new open-source DA library¹.
2. Hard to validate DA methods in practice:
 - Introduce **SKADA-bench**, a new DA benchmark².

¹ Gnassounou et. al. 2024

² Lalou et. al. 2025

Introduction to Electroencephalography (EEG)



- Non-invasive technique to **record brain activity**.
- **Different electrodes** placed on the scalp measure voltage fluctuations.
- Used in various applications: **sleep staging**, epilepsy detection, **brain-computer interfaces**, cognitive load monitoring.

How to tackle distribution shift in biosignals

Possible variability in biosignals:

- Variability in the **patient population** : age, gender, height, diseased or healthy, different sleep stage proportion.

How to tackle distribution shift in biosignals

Possible variability in biosignals:

- Variability in the **patient population** : age, gender, height, diseased or healthy, different sleep stage proportion.
- Variability in **recording and preprocessing** : different sensors, sensor positions, impedance, noise, interference, sampling rates, filtering.

How to tackle distribution shift in biosignals

Possible variability in biosignals:

- Variability in the **patient population** : age, gender, height, diseased or healthy, different sleep stage proportion.
- Variability in **recording and preprocessing** : different sensors, sensor positions, impedance, noise, interference, sampling rates, filtering.
- Variability in **data interpretation by specialist** : different scoring criteria, subjectivity, tiredness.

How to tackle distribution shift in biosignals

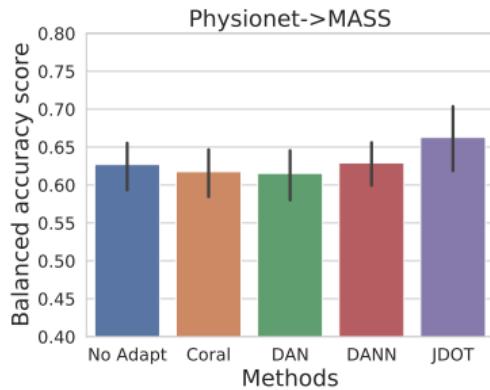
Possible variability in biosignals:

- Variability in the **patient population** : age, gender, height, diseased or healthy, different sleep stage proportion.
- Variability in **recording and preprocessing** : different sensors, sensor positions, impedance, noise, interference, sampling rates, filtering.
- Variability in **data interpretation by specialist** : different scoring criteria, subjectivity, tiredness.

→ **Distribution shift** .

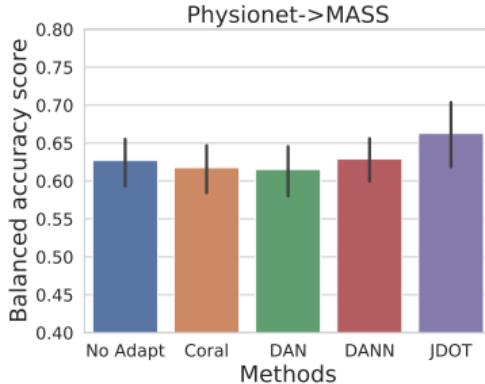
Tackle distribution shift in Biosignals

		Sources	SHHS
		Physionet	
Targets	MASS	0.74	0.63
	Physionet	0.51	0.62
	SHHS	0.49	0.59



Tackle distribution shift in Biosignals

		Sources Physionet	SHHS
Targets	MASS	0.74	0.63
	Physionet	0.51	0.62
	SHHS	0.49	0.59

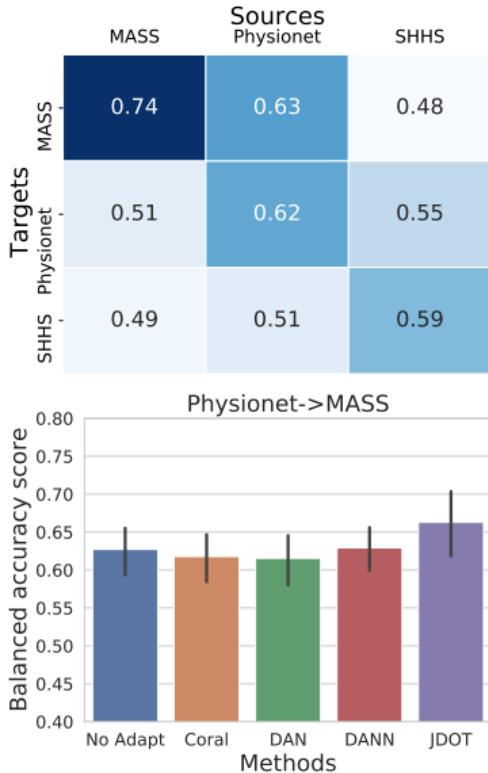


1. How to reduce the drop in performance when testing on a new subject?

- Introduce **Monge Mapping Normalisation** based on Optimal Transport¹.

¹ Gnassounou et. al., 2023

Tackle distribution shift in Biosignals



1. How to reduce the drop in performance when testing on a new subject?
 - Introduce **Monge Mapping Normalisation** based on Optimal Transport¹.
2. How to deal with multiple channels?
 - Extend Monge Mapping Normalisation to a **multi-channels** setting².
3. How to incorporate in deep learning architectures?
 - Introduce **PSDNorm**, a new layer for deep learning architectures³.

¹ Gnassounou et. al., 2023

² Gnassounou et. al., 2024

³ Gnassounou et. al., 2025

SKADA and **SKADA-bench**

SKADA: a Open-source Python library for DA methods.

- **Homogeneous API** for all DA methods (Shallow and Deep learning).
- **Sklearn-like API** with estimator class (.fit, .predict, ...), pipeline, grid search ...
- **DA scorer** to validate hyper-parameters without using target label.



SKADA: a Open-source Python library for DA methods.

- **Homogeneous API** for all DA methods (Shallow and Deep learning).
- **Sklearn-like API** with estimator class (.fit, .predict, ...), pipeline, grid search ...
- **DA scorer** to validate hyper-parameters without using target label.

```
1 from skada.datasets import make_shifted_datasets  
2  
3 X, y, sample_domain = make_shifted_datasets(  
4     shift='conditional_shift',  
5 )
```

SKADA: a Open-source Python library for DA methods.

- **Homogeneous API** for all DA methods (Shallow and Deep learning).
- **Sklearn-like API** with estimator class (.fit, .predict, ...), pipeline, grid search ...
- **DA scorer** to validate hyper-parameters without using target label.

```
1 from skada import OTMapping  
2  
3 estimator = OTMapping()
```

SKADA: a Open-source Python library for DA methods.

- **Homogeneous API** for all DA methods (Shallow and Deep learning).
- **Sklearn-like API** with estimator class (.fit, .predict, ...), pipeline, grid search ...
- **DA scorer** to validate hyper-parameters without using target label.

```
1 from skada import CORAL  
2  
3 estimator = CORAL()
```

SKADA: a Open-source Python library for DA methods.

- **Homogeneous API** for all DA methods (Shallow and Deep learning).
- **Sklearn-like API** with estimator class (.fit, .predict, ...), pipeline, grid search ...
- **DA scorer** to validate hyper-parameters without using target label.

```
1 from skada import CORAL  
2  
3 estimator = CORAL()  
4 estimator.fit(X, y, sample_domain=sample_domain)
```

Validation procedure in DA

Source data
 $(\mathbf{X}_s, \mathbf{y}_s)$

Target data
 $(\mathbf{X}_t, ?)$

Validation procedure in DA

Source data

$$(\mathbf{X}_s, \mathbf{y}_s)$$

Target data

$$(\mathbf{X}_t, ?)$$

1st solution : Use **only source data** to validate hyper-parameters.

$$\begin{array}{c} (\mathbf{X}_s, \mathbf{y}_s) \\ \downarrow \\ (\mathbf{X}_{\text{train}}, \mathbf{y}_{\text{train}}) \end{array}$$

$$\begin{array}{c} (\mathbf{X}_t, ?) \\ \downarrow \\ (\mathbf{X}_{\text{val}}, \mathbf{y}_{\text{val}}) \end{array}$$

Validation procedure in DA

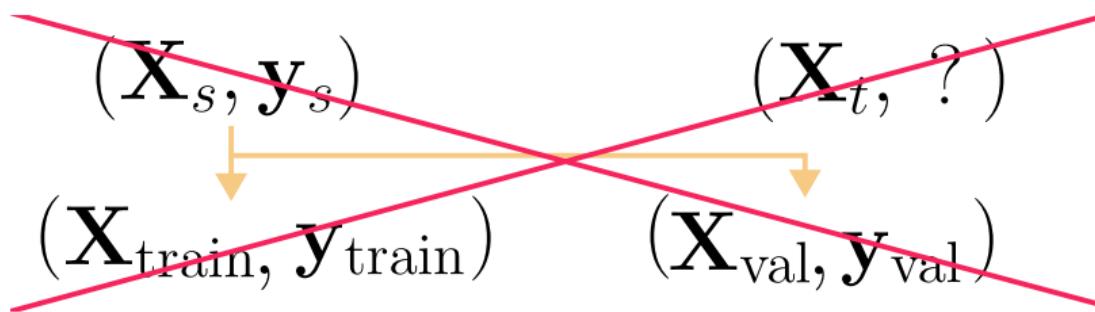
Source data

$$(\mathbf{X}_s, \mathbf{y}_s)$$

Target data

$$(\mathbf{X}_t, ?)$$

1st solution : Use **only source data** to validate hyper-parameters.



Validation procedure in DA

Source data

$$(\mathbf{X}_s, \mathbf{y}_s)$$

Target data

$$(\mathbf{X}_t, ?)$$

2nd solution : Use **target data** to validate hyper-parameters.

$$(\mathbf{X}_s, \mathbf{y}_s)$$

$$(\mathbf{X}_t, ?)$$

$$(\mathbf{X}_{\text{train}}, \mathbf{y}_{\text{train}})$$

$$(\mathbf{X}_{\text{val}}, \mathbf{y}_{\text{val}})$$

Validation procedure in DA

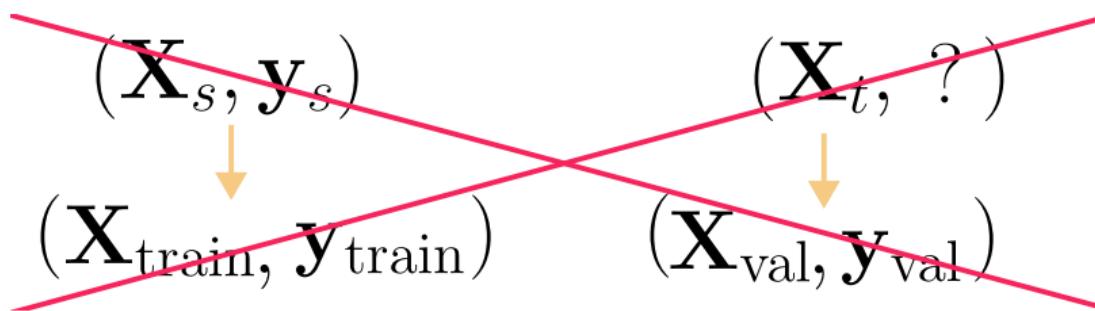
Source data

$$(\mathbf{X}_s, \mathbf{y}_s)$$

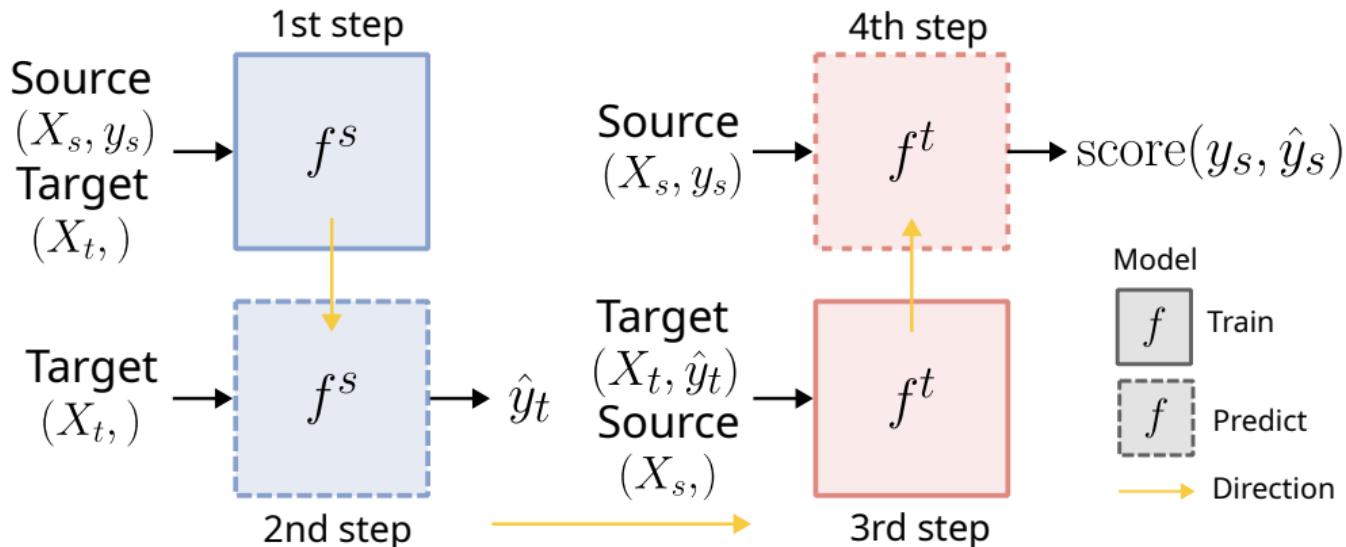
Target data

$$(\mathbf{X}_t, ?)$$

2nd solution : Use **target data** to validate hyper-parameters.



Circular Validation (CircV)¹



¹Bruzonne et. al., 2009

SKADA-bench: a new benchmark for DA

SKADA Methods

- 20 shallow methods
- 7 deep DA methods

SKADA Scorers

- Circular Validation
- 2 Reweighting scorers
- 2 Entropy Minimization scorers

Modalities

- 2 Computer Vision
- 2 Tabular
- 2 NLP
- 1 Biosignals

SKADA-bench: a new benchmark for DA

SKADA Methods

- 20 shallow methods
- 7 deep DA methods

SKADA Scorers

- Circular Validation
- 2 Reweighting scorers
- 2 Entropy Minimization scorers

Modalities

- 2 Computer Vision
- 2 Tabular
- 2 NLP
- 1 Biosignals



BenchOpt

The word "BenchOpt" is written in a large, black, sans-serif font. A red circle with a black outline is positioned behind the letter "o". A black arrow points towards the center of the circle, hitting the bullseye.

Benchmark results

	Cov. shift	Tar. shift	Cond. shift	Sub. shift	Office31	OfficeHome	MNIST /USPS	20NewsGroups	AmazonReview	Mushrooms	Phishing	BCI	Selected Scorer	Rank	
Train Src	0.88	0.85	0.66	0.19	0.59	0.56	0.54	0.59	0.7	0.72	0.91	0.55		9.75	
Train Tgt	0.92	0.93	0.82	0.98	0.88	0.8	0.96	1.0	0.73	1.0	0.97	0.64		1.06	
Reweighting	Dens. RW	0.88	0.86	0.66	0.18	0.57	0.55	0.54	0.58	0.7	0.71	0.91	0.55	IW	10.76
	Gauss. RW	0.89	0.86	0.65	0.21	0.2	0.44	0.11	0.54	0.6	0.51	0.46	0.25	CircV	19.42
	KLIEP	0.88	0.86	0.66	0.19	0.59	0.56	0.54	0.6	0.69	0.72	0.91	0.55	CircV	10.36
	KMM	0.89	0.87	0.64	0.15	0.58	0.55	0.52	0.7	0.57	0.74	0.91	0.52	CircV	12.11
	MMDTarS	0.88	0.86	0.64	0.2	0.56	0.55	0.54	0.59	0.7	0.74	0.91	0.55	IW	9.51
Mapping	CORAL	0.66	0.84	0.66	0.19	0.59	0.57	0.62	0.73	0.69	0.72	0.92	0.62	CircV	7.10
	MapOT	0.72	0.57	0.82	0.02	0.55	0.51	0.61	0.76	0.67	0.63	0.84	0.47	PE	10.98
	EntOT	0.71	0.6	0.82	0.12	0.58	0.58	0.6	0.83	0.62	0.75	0.86	0.54	CircV	9.75
	LinOT	0.73	0.73	0.76	0.18	0.59	0.57	0.64	0.82	0.7	0.76	0.91	0.61	CircV	5.33
	MMD-LS	0.65	0.68	0.81	0.52	0.55	0.54	0.52	0.97	0.68	0.86	0.88	0.56	IW	9.66
Sub.	JPCA	0.88	0.85	0.66	0.15	0.55	0.47	0.51	0.77	0.69	0.78	0.9	0.54	PE	8.77
	SA	0.74	0.68	0.8	0.11	0.59	0.57	0.56	0.88	0.66	0.88	0.89	0.53	CircV	8.53
	TCA	0.46	0.48	0.55	0.56	0.04	NA	0.11	0.57	0.6	0.45	NA	0.27	CircV	19.57

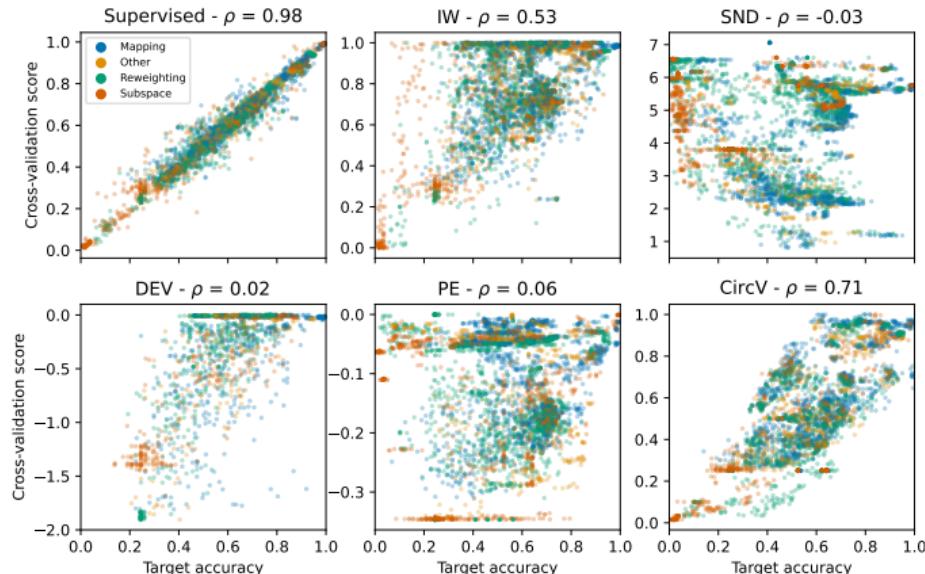
Benchmark results

		Cov. shift	Tar. shift	Cond. shift	Sub. shift	Office31	OfficeHome	MNIST /USPS	20NewsGroups	AmazonReview	Mushrooms	Phishing	BCI	Selected Scorer	Rank
	Train Src	0.88	0.85	0.66	0.19	0.59	0.56	0.54	0.59	0.7	0.72	0.91	0.55		9.75
	Train Tgt	0.92	0.93	0.82	0.98	0.88	0.8	0.96	1.0	0.73	1.0	0.97	0.64		1.06
Reweighting	Dens. RW	0.88	0.86	0.66	0.18	0.57	0.55	0.54	0.58	0.7	0.71	0.91	0.55	IW	10.76
	Gauss. RW	0.89	0.86	0.65	0.21	0.2	0.44	0.11	0.54	0.6	0.51	0.46	0.25	CircV	19.42
	KLIEP	0.88	0.86	0.66	0.19	0.59	0.56	0.54	0.6	0.69	0.72	0.91	0.55	CircV	10.36
	KMM	0.89	0.87	0.64	0.15	0.58	0.55	0.52	0.7	0.57	0.74	0.91	0.52	CircV	12.11
	MMDTarS	0.88	0.86	0.64	0.2	0.56	0.55	0.54	0.59	0.7	0.74	0.91	0.55	IW	9.51
Mapping	CORAL	0.66	0.84	0.66	0.19	0.59	0.57	0.62	0.73	0.69	0.72	0.92	0.62	CircV	7.10
	MapOT	0.72	0.57	0.82	0.02	0.55	0.51	0.61	0.76	0.67	0.63	0.84	0.47	PE	10.98
	EntOT	0.71	0.6	0.82	0.12	0.58	0.58	0.6	0.83	0.62	0.75	0.86	0.54	CircV	9.75
	LinOT	0.73	0.73	0.76	0.18	0.59	0.57	0.64	0.82	0.7	0.76	0.91	0.61	CircV	5.33
	MMD-LS	0.65	0.68	0.81	0.52	0.55	0.54	0.52	0.97	0.68	0.86	0.88	0.56	IW	9.66
Sub.	JPCA	0.88	0.85	0.66	0.15	0.55	0.47	0.51	0.77	0.69	0.78	0.9	0.54	PE	8.77
	SA	0.74	0.68	0.8	0.11	0.59	0.57	0.56	0.88	0.66	0.88	0.89	0.53	CircV	8.53
	TCA	0.46	0.48	0.55	0.56	0.04	NA	0.11	0.57	0.6	0.45	NA	0.27	CircV	19.57

Benchmark results

	Cov. shift	Tar. shift	Cond. shift	Sub. shift	Office31	OfficeHome	MNIST / USPS	20NewsGroups	AmazonReview	Mushrooms	Phishing	BCI	Selected Scorer	Rank	
Train Src	0.88	0.85	0.66	0.19	0.59	0.56	0.54	0.59	0.7	0.72	0.91	0.55		9.75	
Train Tgt	0.92	0.93	0.82	0.98	0.88	0.8	0.96	1.0	0.73	1.0	0.97	0.64		1.06	
Reweighting	Dens. RW	0.88	0.86	0.66	0.18	0.57	0.55	0.54	0.58	0.7	0.71	0.91	0.55	IW	10.76
	Gauss. RW	0.89	0.86	0.65	0.21	0.2	0.44	0.11	0.54	0.6	0.51	0.46	0.25	CircV	19.42
	KLIEP	0.88	0.86	0.66	0.19	0.59	0.56	0.54	0.6	0.69	0.72	0.91	0.55	CircV	10.36
	KMM	0.89	0.87	0.64	0.15	0.58	0.55	0.52	0.7	0.57	0.74	0.91	0.52	CircV	12.11
	MMDTarS	0.88	0.86	0.64	0.2	0.56	0.55	0.54	0.59	0.7	0.74	0.91	0.55	IW	9.51
Mapping	CORAL	0.66	0.84	0.66	0.19	0.59	0.57	0.62	0.73	0.69	0.72	0.92	0.62	CircV	7.10
	MapOT	0.72	0.57	0.82	0.02	0.55	0.51	0.61	0.76	0.67	0.63	0.84	0.47	PE	10.98
	EntOT	0.71	0.6	0.82	0.12	0.58	0.58	0.6	0.83	0.62	0.75	0.86	0.54	CircV	9.75
	LinOT	0.73	0.73	0.76	0.18	0.59	0.57	0.64	0.82	0.7	0.76	0.91	0.61	CircV	5.33
	MMD-LS	0.65	0.68	0.81	0.52	0.55	0.54	0.52	0.97	0.68	0.86	0.88	0.56	IW	9.66
Sub.	JPCA	0.88	0.85	0.66	0.15	0.55	0.47	0.51	0.77	0.69	0.78	0.9	0.54	PE	8.77
	SA	0.74	0.68	0.8	0.11	0.59	0.57	0.56	0.88	0.66	0.88	0.89	0.53	CircV	8.53
	TCA	0.46	0.48	0.55	0.56	0.04	NA	0.11	0.57	0.6	0.45	NA	0.27	CircV	19.57

DA scorers



- **Supervised** scorers (cheating) correlate well with target accuracy.
- **Unsupervised** scorers are **less correlated** with target accuracy.
- Only **CircV** is correlated with target accuracy.

SKADA-bench: Conclusion and perspectives

Key takeaways:

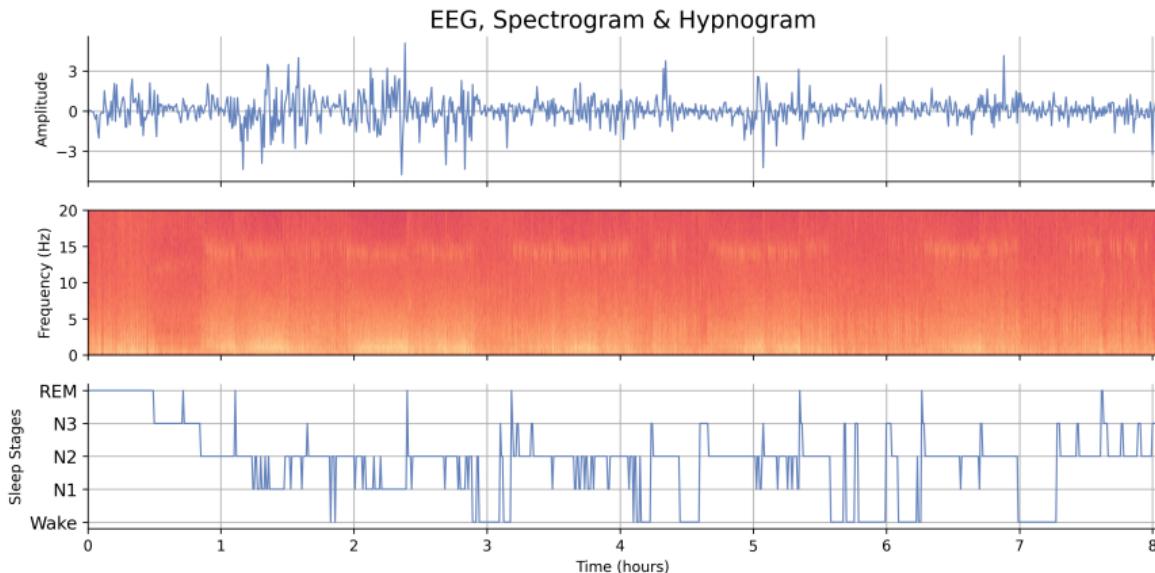
- SKADA makes easy to use DA
- Extensive benchmark show the **limit of DA field**
- DA methods need better **scorers** to be properly evaluated

Perspectives:

- **Detect the shift** to apply the right DA method
- Propose **new DA scorers** better correlated with target accuracy
- Extend SKADA-bench to **Deep DA** methods

Monge Mapping for biosignals

Impact of distribution shift in Sleep Staging



- **Classification** problem with **five** classes: Wake, N1, N2, N3, REM for each **30s** epoch.
- **Frequency** helps to classify sleep stage

Multi-source Domain Adaptation for Sleep Staging



Subject S_1
(X_1, y_1)



Subject S_2
(X_2, y_2)



Subject T
(X_T)

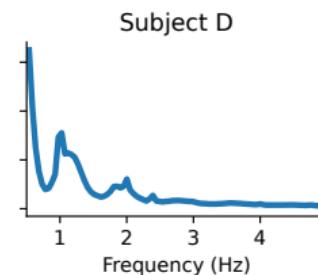
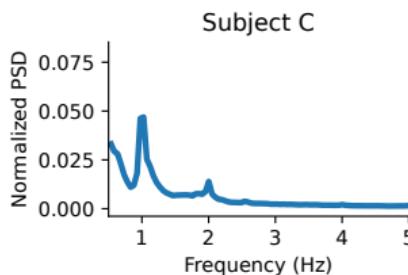
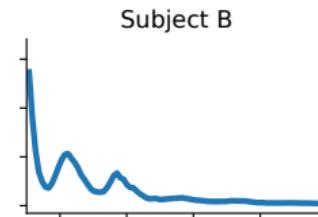
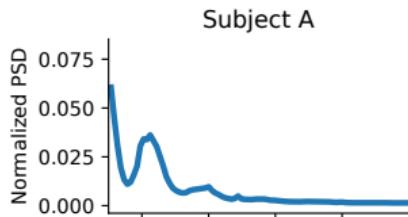


Subject S_3
(X_3, y_3)



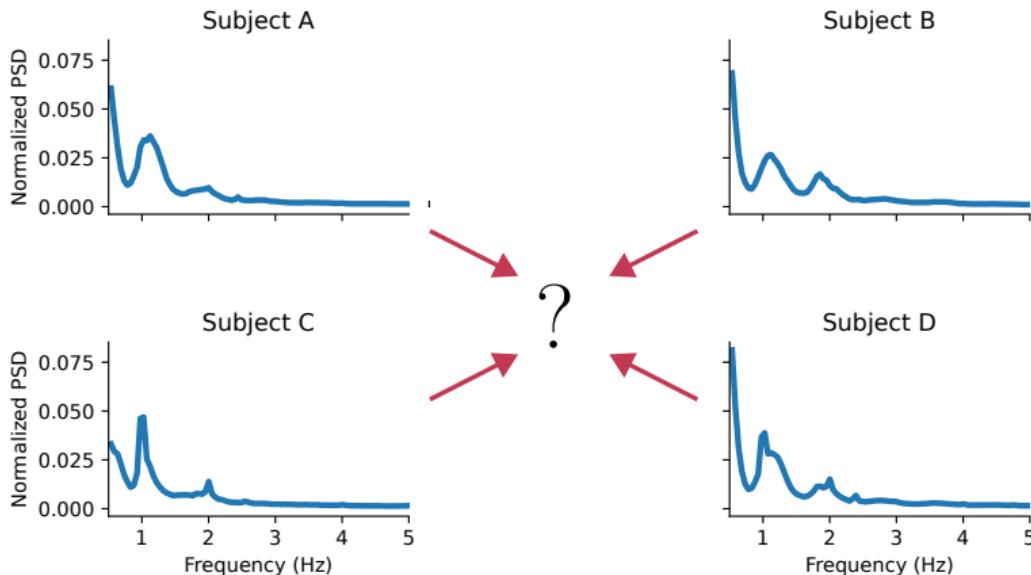
Subject S_4
(X_4, y_4)

Shift in PSD for Sleep Staging



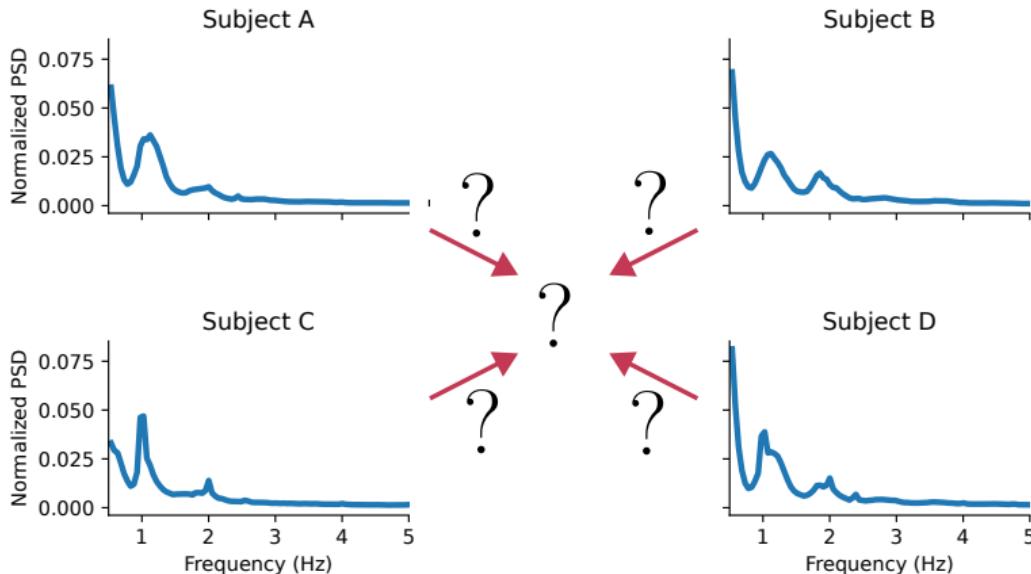
- Power Spectral Density (PSD) : representation of the signal in the frequency domain.

Shift in PSD for Sleep Staging



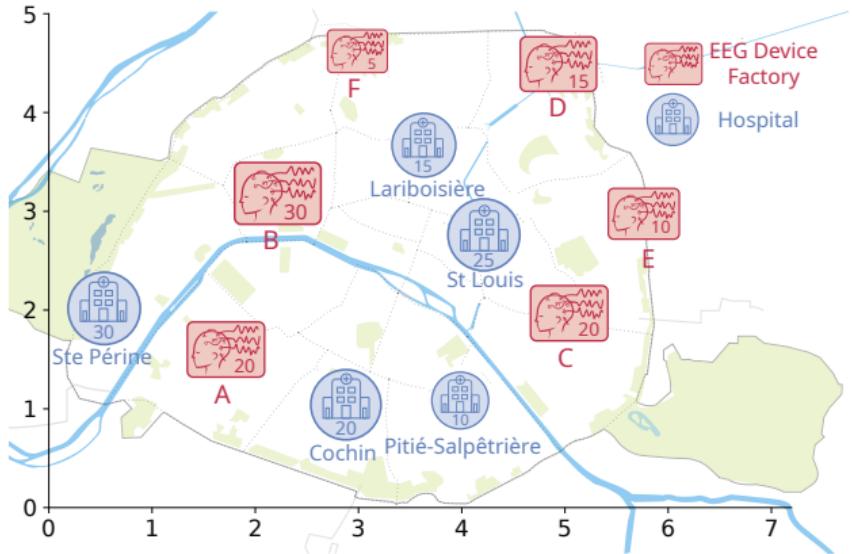
- **Power Spectral Density (PSD)** : representation of the signal in the frequency domain.
- Project each PSD to a common reference: **Which Reference ?**

Shift in PSD for Sleep Staging



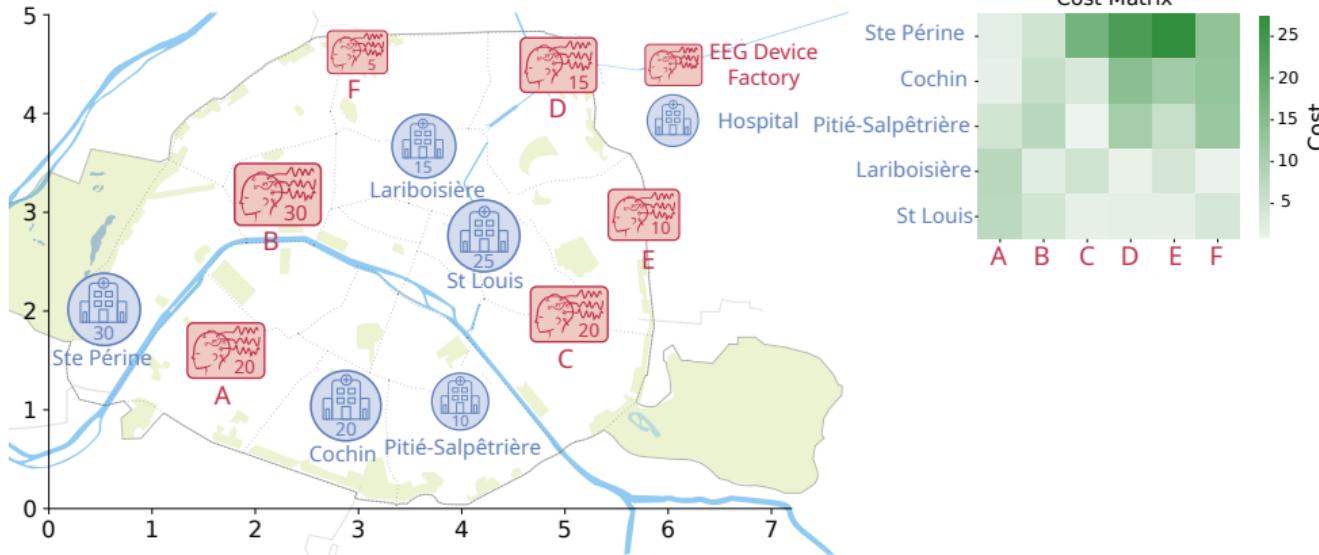
- **Power Spectral Density (PSD)** : representation of the signal in the frequency domain.
- Project each PSD to a common reference: **Which Reference ? How to project ?**

What is Optimal Transport?



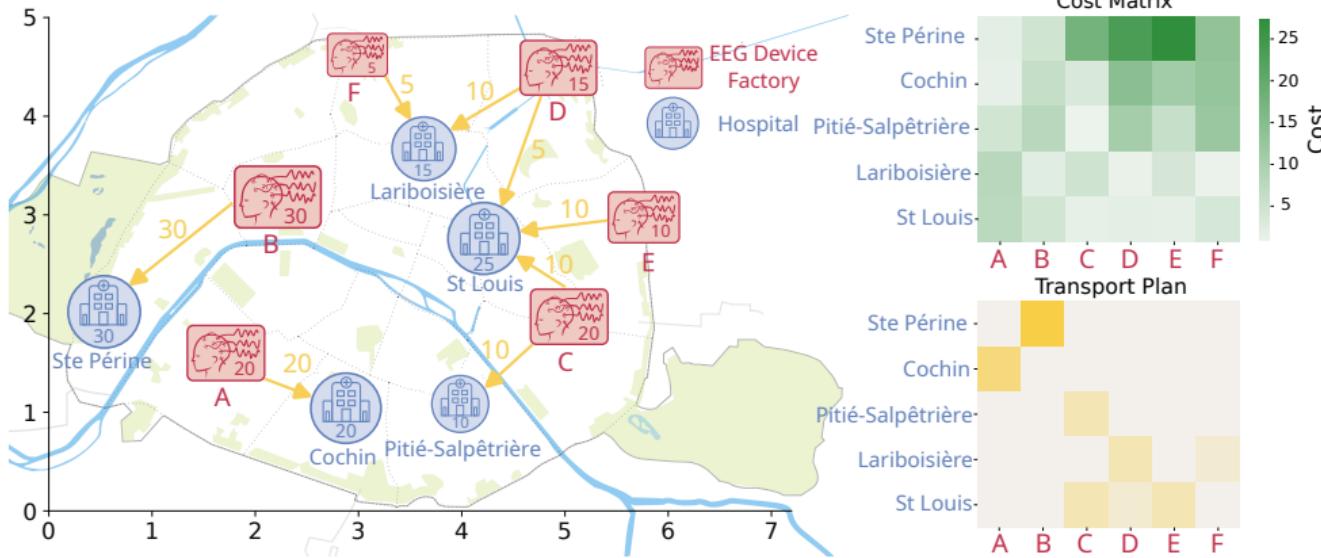
- Consider two distributions μ_s (**EEG device factory**) and μ_t (**hospital**).

What is Optimal Transport?



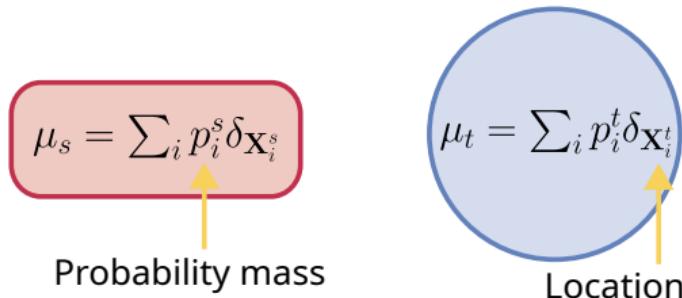
- A **ground metric** defines the cost of transporting mass between points.

What is Optimal Transport?



- Optimal Transport aim to find the **best way** to transport source distribution to target distribution.

Optimal Transport formulation



- We can define a **Wasserstein distance** with a transport cost matrix $\mathbf{C}_{ij} = \|\mathbf{x}_s^i - \mathbf{x}_t^j\|^p$

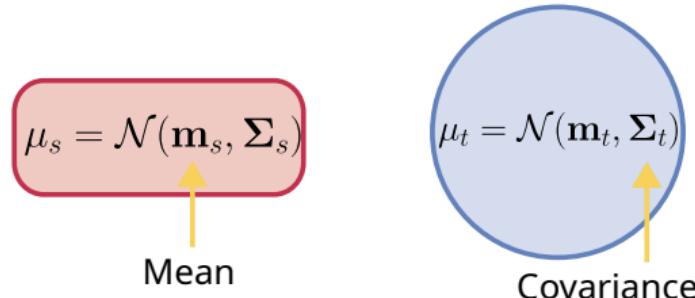
$$\mathcal{W}_p^p(\mu_s, \mu_t) = \min_{\mathbf{T} \in \Pi(\mathbf{a}, \mathbf{b})} \langle \mathbf{C}, \mathbf{T} \rangle_F$$

where $\Pi(\mathbf{a}, \mathbf{b}) = \{\mathbf{T} \in \mathbb{R}_+^{n_s \times n_t} \mid \mathbf{T}\mathbf{1}_{n_t} = \mu_s, \mathbf{T}^\top \mathbf{1}_{n_s} = \mu_t\}$ is the set of valid transport plans.

- And a **Monge Mapping** from source to target samples:

$$m(\mathbf{x}_t^s) = \frac{1}{a_i} \sum_{j=1}^{n_t} \mathbf{T}_{ij} \mathbf{x}_j^t$$

Monge mapping for Gaussian distributions



- We can define a **Wasserstein distance** or Bures-Wasserstein distance:

$$\mathcal{W}_2^2(\mu_s, \mu_t) = \text{Tr} \left(\Sigma_s + \Sigma_t - 2 \left(\Sigma_t^{\frac{1}{2}} \Sigma_s \Sigma_t^{\frac{1}{2}} \right)^{\frac{1}{2}} \right)$$

- And a **Monge Mapping** from source to target samples:

$$m(\mathbf{x}) = \mathbf{A} (\mathbf{x} - \mathbf{m}_s) + \mathbf{m}_t, \quad \text{with} \quad \mathbf{A} = \Sigma_s^{-\frac{1}{2}} \left(\Sigma_s^{\frac{1}{2}} \Sigma_t \Sigma_s^{\frac{1}{2}} \right)^{\frac{1}{2}} \Sigma_s^{-\frac{1}{2}} = \mathbf{A}^T$$

↑
Transport Map
↑
Source Covariance
↑
Target Covariance

Wasserstein barycenter between Gaussian distributions

Considering multiple Gaussian distributions μ_k . The barycenter $\bar{\mu}$ is expressed as

$$\bar{\mu} = \arg \min_{\mu} \frac{1}{K} \sum_{k=1}^K \mathcal{W}_2^2(\mu, \mu_k) . \quad (1)$$

Wasserstein barycenter between Gaussian distributions

Considering multiple Gaussian distributions μ_k . The barycenter $\bar{\mu}$ is expressed as

$$\bar{\mu} = \arg \min_{\mu} \frac{1}{K} \sum_{k=1}^K \mathcal{W}_2^2(\mu, \mu_k) . \quad (1)$$

The barycenter is still a Gaussian distribution $\bar{\mu} = \mathcal{N}(\bar{\mathbf{m}}, \bar{\Sigma})$.

Wasserstein barycenter between Gaussian distributions

Considering multiple Gaussian distributions μ_k . The barycenter $\bar{\mu}$ is expressed as

$$\bar{\mu} = \arg \min_{\mu} \frac{1}{K} \sum_{k=1}^K \mathcal{W}_2^2(\mu, \mu_k) . \quad (1)$$

The barycenter is still a Gaussian distribution $\bar{\mu} = \mathcal{N}(\bar{\mathbf{m}}, \bar{\Sigma})$.

⇒ **No closed-form** for computing the covariance $\bar{\Sigma}$.

Wasserstein barycenter between Gaussian distributions

Considering multiple Gaussian distributions μ_k . The barycenter $\bar{\mu}$ is expressed as

$$\bar{\mu} = \arg \min_{\mu} \frac{1}{K} \sum_{k=1}^K \mathcal{W}_2^2(\mu, \mu_k) . \quad (1)$$

The barycenter is still a Gaussian distribution $\bar{\mu} = \mathcal{N}(\bar{\mathbf{m}}, \bar{\Sigma})$.

⇒ **No closed-form** for computing the covariance $\bar{\Sigma}$.

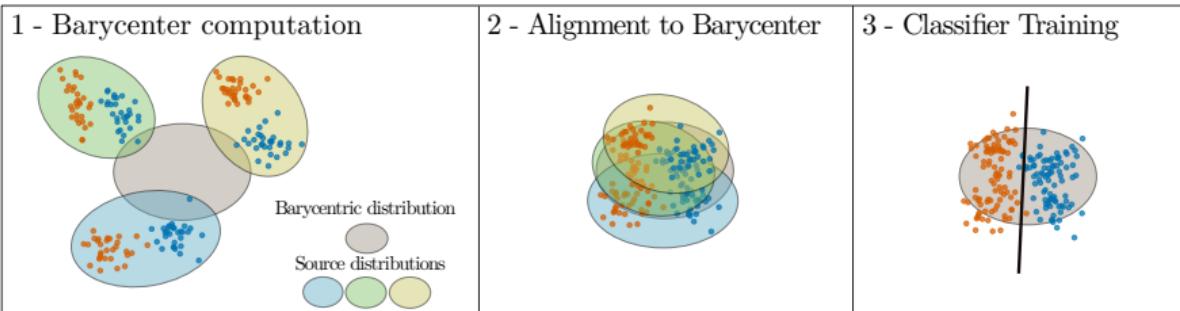
One uses the following optimality condition¹:

$$\bar{\Sigma} = \frac{1}{K} \sum_{k=1}^K \left(\bar{\Sigma}^{\frac{1}{2}} \Sigma_k \bar{\Sigma}^{\frac{1}{2}} \right)^{\frac{1}{2}} , \quad (2)$$

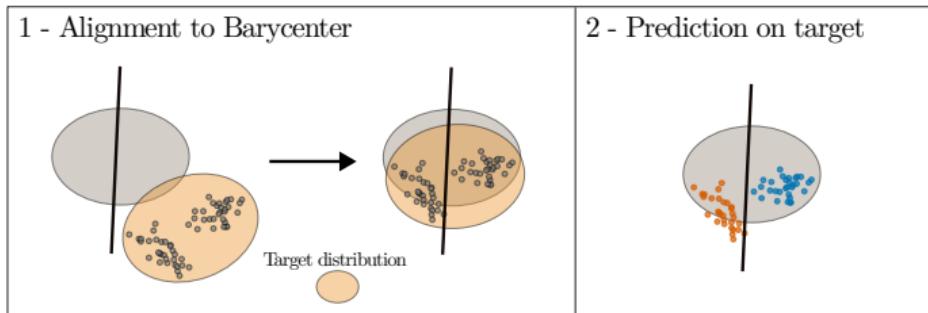
¹Agueh et. al., 2011

Monge Mapping Normalisation

Train-time

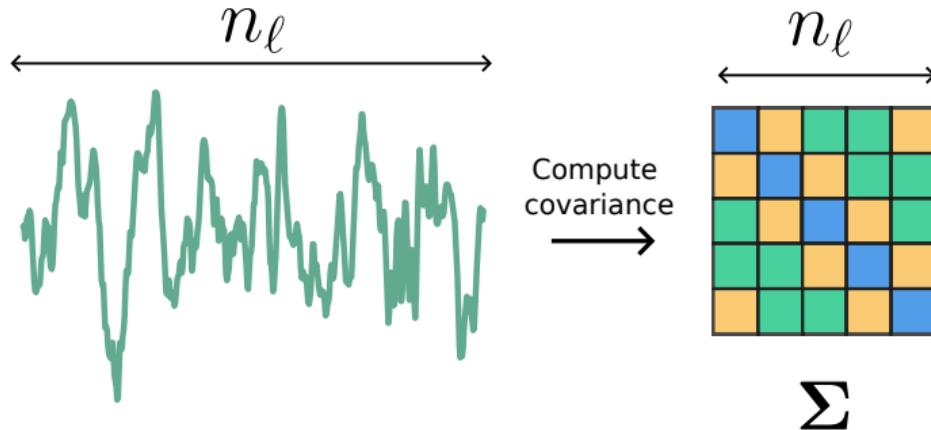


Test-time



Assumptions on the signals

- Centered **Gaussian** distributions $\rightarrow \mathbf{X} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ with $\boldsymbol{\Sigma} \in \mathcal{S}_{n_\ell}^{++}$
- $\boldsymbol{\Sigma}$ is the "auto-covariance", computed with time-lagged. $\boldsymbol{\Sigma}_{i,j} = \mathbf{X}_i \mathbf{X}_j$
- **Stationarity+Periodicity** \rightarrow Covariance matrices are **Toeplitz circulant** matrices.



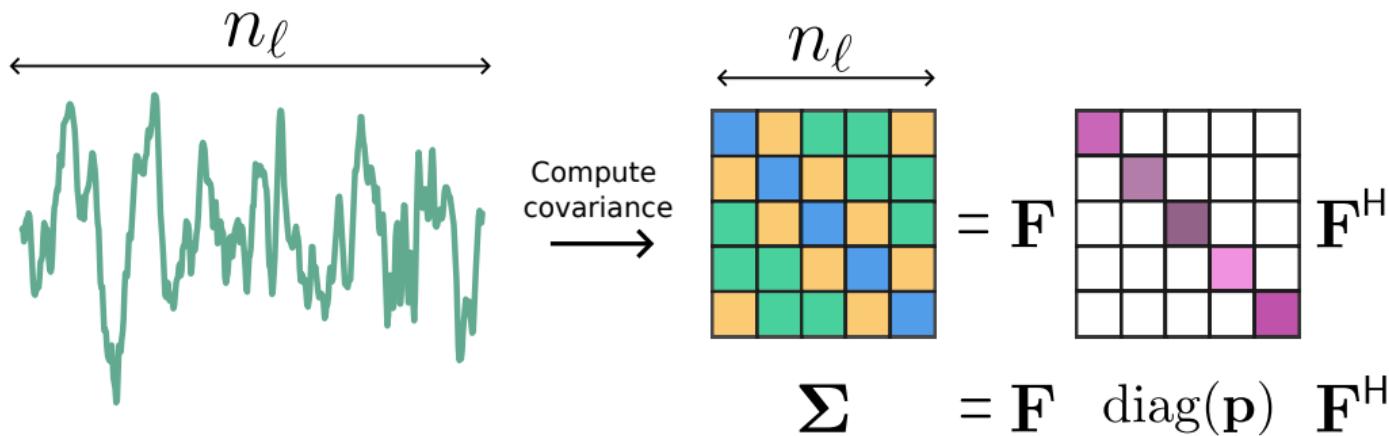
Assumptions on the signals

The Discrete Fourier Transform (DFT) can diagonalize the circulant matrix

$$\Sigma = \mathbf{F} \text{diag}(\mathbf{p}) \mathbf{F}^*$$

Power Spectral Density (PSD)

with \mathbf{F} and \mathbf{F}^* the Fourier transform operator and its inverse, and \mathbf{p} the Power Spectral Density (PSD) of the signal.



Convolutional Monge Mapping Normalization

Consider K centered stationary Gaussian signals of covariance $\Sigma_k = \text{Fdiag}(\mathbf{p}_k)\mathbf{F}^*$ and PSD \mathbf{p}_k with $k \in [K]$.

- First step: **barycenter computation**:

$$\bar{\Sigma} = \frac{1}{K} \sum_{k=1}^K \left(\bar{\Sigma}^{\frac{1}{2}} \Sigma_k \bar{\Sigma}^{\frac{1}{2}} \right)^{\frac{1}{2}},$$

Barycenter Covariance Domain k Covariance

- Second step: **map** each **signal k** to barycenter:

$$m(x) = \mathbf{A}x, \quad \text{with} \quad \mathbf{A} = \Sigma_s^{-\frac{1}{2}} \left(\Sigma_s^{\frac{1}{2}} \Sigma_t \Sigma_s^{\frac{1}{2}} \right)^{\frac{1}{2}} \Sigma_s^{-\frac{1}{2}} = \mathbf{A}^T.$$

Transport Map Source Covariance Target Covariance

Convolutional Monge Mapping Normalization

Consider K centered stationary Gaussian signals of covariance $\Sigma_k = \mathbf{F} \text{diag}(\mathbf{p}_k) \mathbf{F}^*$ and PSD \mathbf{p}_k with $k \in [K]$.

- First step: **barycenter computation** :

$$\bar{\mathbf{p}} = \left(\frac{1}{K} \sum_{k=1}^K \mathbf{p}_k^{\odot \frac{1}{2}} \right)^{\odot 2}.$$

Barycenter PSD Domain k PSD

- Second step: **map** each **signal k** to barycenter:

$$m(\mathbf{x}) = \mathbf{h} * \mathbf{x}, \quad \text{with} \quad \mathbf{h} = \mathbf{F}^* \left(\bar{\mathbf{p}}^{\odot \frac{1}{2}} \odot \mathbf{p}_k^{\odot -\frac{1}{2}} \right).$$

Filter Source PSD Target PSD

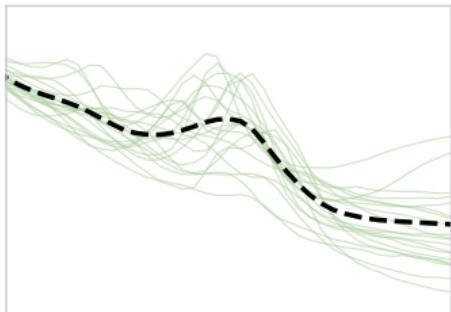
Convolutional Monge Mapping Normalization

Consider K centered stationary Gaussian signals of covariance $\Sigma_k = \mathbf{F} \text{diag}(\mathbf{p}_k) \mathbf{F}^*$ and PSD \mathbf{p}_k with $k \in [K]$.

- First step: **barycenter computation**:

$$\bar{\mathbf{p}} = \left(\frac{1}{K} \sum_{k=1}^K \mathbf{p}_k^{\odot \frac{1}{2}} \right)^{\odot 2}$$

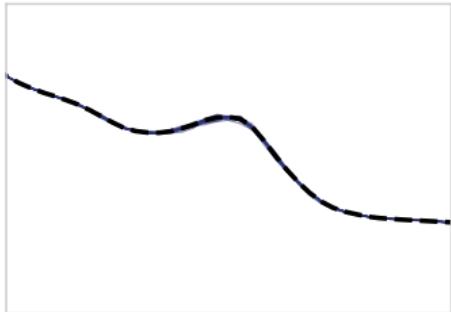
Barycenter PSD Domain k PSD



- Second step: map each signal k to barycenter:

$$m(\mathbf{x}) = \mathbf{h} * \mathbf{x}, \quad \text{with} \quad \mathbf{h} = \mathbf{F}^* \left(\bar{\mathbf{p}}^{\odot \frac{1}{2}} \odot \mathbf{p}_k^{\odot -\frac{1}{2}} \right).$$

Filter Source PSD Target PSD

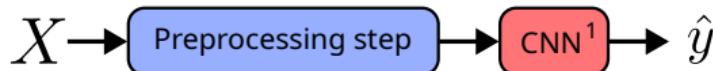


Discussion on Stationarity and gaussiannity assumptions

- EEG signals are **non-stationary** and **non Gaussian**
- **CMMN** relies on these assumptions for effective learning
- The "real" assumption is: **The shift is comprised in the second order statistics.**
- Monge Mapping reduce the shift but keep **individual information** of each subject in the highest orders.

Limit of CMMN setup

- Three different datasets: MASS, Physionet and SHHS
- Only 200 subjects in total
- Use simple CNN architecture from¹
- Only for univariate signals



Architecture	Chambon [6]	
	No Adapt	CMMN
MASS→MASS	75.1 ± 1.0	76.2 ± 2.2
Phys.→Phys.	69.2 ± 2.7	71.7 ± 2.4
SHHS→SHHS	61.2 ± 3.8	64.3 ± 2.7
MASS→Phys.	58.4 ± 2.4	62.3 ± 1.5
MASS→SHHS	41.8 ± 3.6	47.6 ± 4.0
Phys.→MASS	64.0 ± 2.7	68.3 ± 2.5
Phys.→SHHS	45.6 ± 2.1	51.6 ± 1.8
SHHS→MASS	57.0 ± 2.8	64.5 ± 2.8
SHHS→Phys.	55.0 ± 2.7	58.3 ± 1.7
Mean	58.6 ± 2.6	62.7 ± 2.4

→ Problems: Low number of subjects, not SOTA network, 2 steps preprocess than train .

¹Chambon et. al., 2018

Extension of CMMN setup

Monge Mapping for Multi-channel signals:

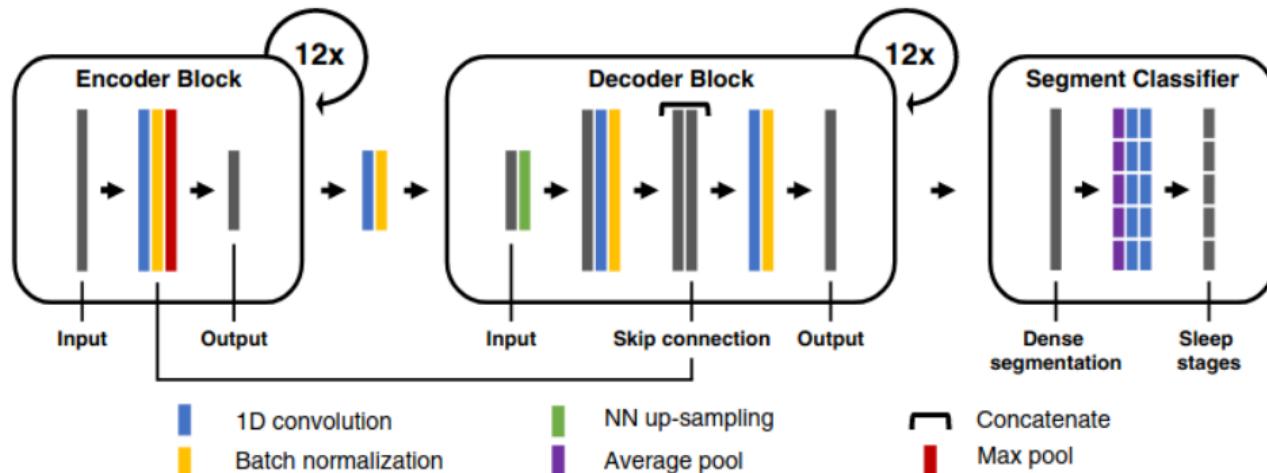
- Use **cross-PSD** to capture the **spatial** information
- Application on Sleep staging and BCI.
- Proof of concept on images

End-to-end training with PSDNorm:

- **PSDNorm** as a new normalization layer in deep learning
- Application on Sleep staging with **SOTA architecture** .

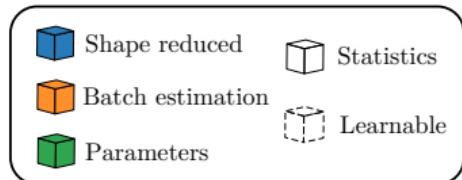
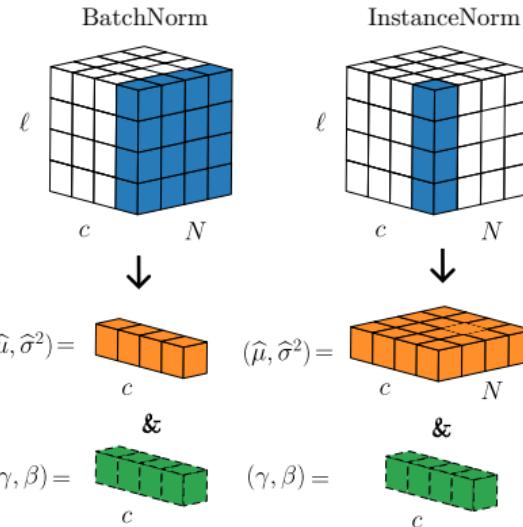
End-to-end training with PSDNorm

Better architecture



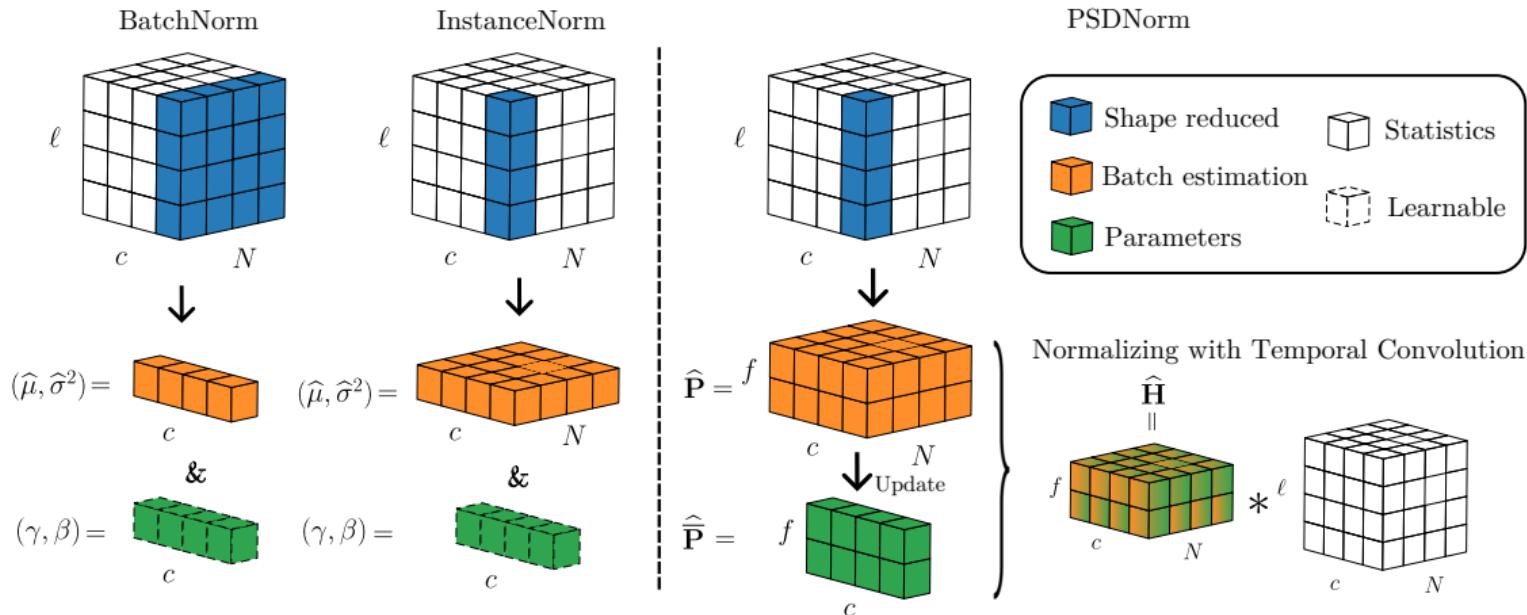
- **UNet** architecture: encoder-decoder with skip connections
- Take as input **sequence of 17min of sleep** (one annotation every 30seconds)
- Very slow training in the original paper but we improved the implementation to be **faster**
 - How can we use the **CMMN** to improve the training of this architecture?

Classical Normalizations in Deep Learning



- **Batch Normalization (BN):** Normalize the input of each layer with the mean and variance of the batch
- **Instance Normalization (IN):** Normalize the input of each layer with the mean and variance of the instance. **Adapted to time series**

PSDNorm as a new normalization layer



- Compute **PSD over the sequence** of sleep (vs. one on night before)
- Compute the barycenter with a **geodesic update on the batch**.

Increase the number of subjects

Dataset	Subj.	Rec.	Age \pm std	Sex (F/M)
ABC	44	117	48.8 \pm 9.8	43%/57%
CCSHS	515	515	17.7 \pm 0.4	50%/50%
CFS	681	681	41.7 \pm 20.0	55%/45%
HPAP	166	166	46.5 \pm 11.9	43%/57%
MROS	2101	2698	76.4 \pm 5.5	0%/100%
PHYS	70	132	58.8 \pm 22.0	33%/67%
SHHS	5730	8271	63.1 \pm 11.2	52%/48%
MASS	61	61	42.5 \pm 18.9	55%/45%
CHAT	1230	1635	6.6 \pm 1.4	52%/48%
SOF	434	434	82.8 \pm 3.1	100%/0%
Total	11032	14710	-	-

→ **10** datasets, **10k** subjects, **10M** of samples

Results on LODO sleep staging

	Dataset	BatchNorm	InstanceNorm	CMMN	PSDNorm
All subjects	ABC	78.49 \pm 0.42	78.83\pm0.59	78.33 \pm 0.12	78.56 \pm 0.67
	CCSHS	88.79\pm0.21	88.75 \pm 0.04	88.61 \pm 0.10	88.56 \pm 0.36
	CFS	84.97 \pm 0.37	85.73\pm0.29	84.85 \pm 0.13	85.42 \pm 0.09
	CHAT	64.72 \pm 3.94	68.86 \pm 2.49	69.76 \pm 1.62	70.57\pm1.24
	HOMEPAF	76.39 \pm 0.29	76.70 \pm 0.35	76.77\pm0.66	76.72 \pm 0.27
	MASS	73.71 \pm 0.62	72.12 \pm 0.70	73.90\pm0.69	72.51 \pm 1.68
	MROS	81.30 \pm 0.25	81.49 \pm 0.18	80.91 \pm 0.42	81.57\pm0.34
	PhysioNet	76.13 \pm 0.57	76.15 \pm 0.52	76.48\pm0.37	75.96 \pm 1.02
	SHHS	77.97 \pm 1.46	79.05 \pm 0.89	78.21 \pm 0.39	79.14\pm1.01
	SOF	81.33 \pm 0.54	81.98 \pm 0.22	81.84 \pm 0.49	82.50\pm0.34
Balanced@400	Mean(Dataset)	78.38 \pm 0.47	78.97 \pm 0.11	78.98 \pm 0.14	79.15\pm0.14
	Mean(Subject)	78.14 \pm 1.01	79.26 \pm 0.48	78.77 \pm 0.07	79.51\pm0.62
	ABC	78.26 \pm 1.33	78.73\pm0.42	78.04 \pm 0.51	78.18 \pm 0.68
	CCSHS	87.42 \pm 0.16	87.62\pm0.42	87.57 \pm 0.20	87.58 \pm 0.30
	CFS	84.32 \pm 0.57	84.72\pm0.33	84.58 \pm 0.20	84.29 \pm 0.36
	CHAT	66.55 \pm 0.88	64.43 \pm 4.41	68.73 \pm 2.48	70.28\pm1.70
	HOMEPAF	75.25 \pm 0.50	76.47\pm0.63	76.10 \pm 0.32	76.83\pm0.61
	MASS	70.00 \pm 1.91	71.52 \pm 1.13	71.63 \pm 1.92	72.77\pm1.09
	MROS	80.37\pm0.20	80.28 \pm 0.21	80.09 \pm 0.40	80.26 \pm 0.11
	PhysioNet	75.81\pm0.13	74.68 \pm 0.55	75.31 \pm 1.54	74.82 \pm 2.11
	SHHS	76.44 \pm 0.92	78.68 \pm 0.37	77.00 \pm 0.39	78.88\pm0.68
	SOF	81.08 \pm 1.14	80.68 \pm 1.38	81.25\pm0.71	79.49 \pm 0.41
	Mean(Dataset)	77.55 \pm 0.34	77.78\pm0.46	78.03 \pm 0.35	78.34\pm0.42
	Mean(Subject)	77.22 \pm 0.34	78.17 \pm 0.28	77.74 \pm 0.36	78.85\pm0.59

- Balanced@400 have **10x less** subjects than all subjects
- PSDNorm reaches performances of BatchNorm with **10x less** data
- PSDNorm outperforms CMMN, **better adaptation with end-to-end training**

Distribution shift in biosignals: Conclusion

Key takeaways:

- Distribution shift is a major challenge in biosignal applications.
- While large datasets can mitigate this shift, they are often unavailable due to **privacy and logistical constraints**.
- PSDNorm (Monge Mapping) **efficiently reduces** this shift, achieving high performance even with **limited data** .

Perspectives:

- Apply PSDNorm/CMMN to **other applications** : BCI, ECG classification, ...
- Comparison with **Foundational models** for biosignals

My key takeaways from 3-year PhD journey:

- **Understand the societal and environmental impact of my work :**
 - 0.416T of CO₂ emitted for PSDNorm paper training
- **Understanding the problem :**
 - Characterize the data (and possible shifts) to inform model design.
 - Design experiments thoughtfully to avoid unnecessary and resource-intensive computations.
- **Embracing Open Science :**
 - Share code and data to foster collaboration and reproducibility.
 - Collaborate on open-source packages .