Introduction to (Python) Optimal Transport

Rémi Flamary, École polytechnique

January 28 2025

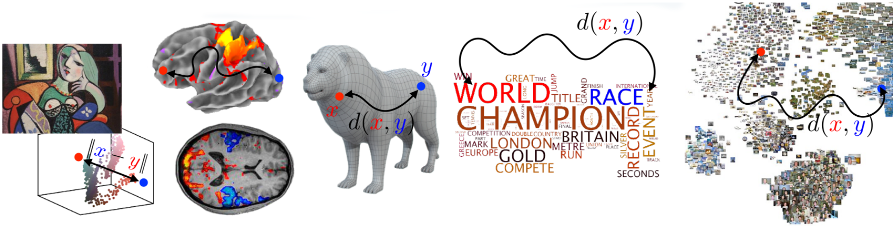
Distributions are everywhere

Distributions are everywhere in machine learning

- Images, vision, graphics, Time series, text, genes, proteins.
- Many datum and datasets can be seen as distributions.
- Important questions:
  - How to compare distributions?
  - How to use the geometry of distributions?
- Optimal transport provides many tools that can answer those questions.

Illustration from the slides of Gabriel Peyré.

Distributions are everywhere

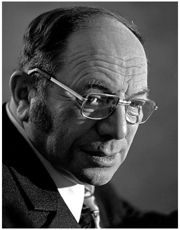


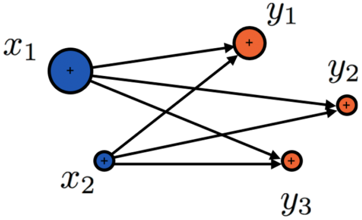
Distributions are everywhere in machine learning

- Images, vision, graphics, Time series, text, genes, proteins.
- Many datum and datasets can be seen as distributions.
- Important questions:
  - How to compare distributions?
  - How to use the geometry of distributions?
- Optimal transport provides many tools that can answer those questions.

Illustration from the slides of Gabriel Peyré.

Optimal transport





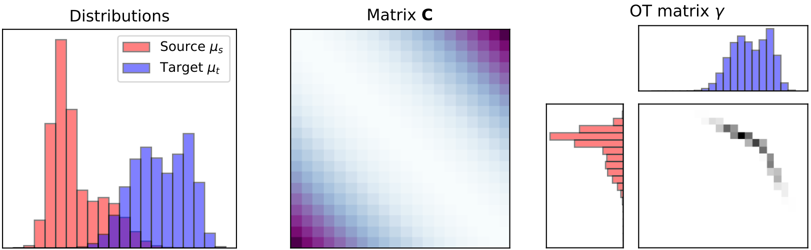
- Problem introduced by Gaspard Monge in his memoire [Monge, 1781].
- How to move mass while minimizing a cost (mass + cost)
- Monge formulation seeks for a mapping between two mass distribution.
- Reformulated by Leonid Kantorovich (1912–1986), Economy nobelist in 1975
- Focus on where the mass goes, allow splitting [Kantorovich, 1942].
- Applications originally for resource allocation problems

Python Optimal Transport (POT)

The toolbox

- Website/documentation: `https://pythonot.github.io/`
- Github: `https://github.com/PythonOT/POT`
- Activity: 76 contributors, 2.5k stars, 2.8 M PyPI downloads, 1000 citations.
- Features: OT solvers from 73 papers, 58 examples in gallery.
- CI-CD: 95% test coverage, 100% PEP8 compliant with pre-commit.
- Maintained since 2017: 2 releases/year, 1.5k commits.
- Deep learning features: Pytorch/Tensorflow/Jax support with autodiff.

Optimal transport between discrete distributions



Kantorovitch formulation : OT Linear Program

When $\mu_s = \sum_{i=1}^{n_s} a_i \delta_{\mathbf{x}_i^s}$ and $\mu_t = \sum_{i=1}^{n_t} b_i \delta_{\mathbf{x}_i^t}$

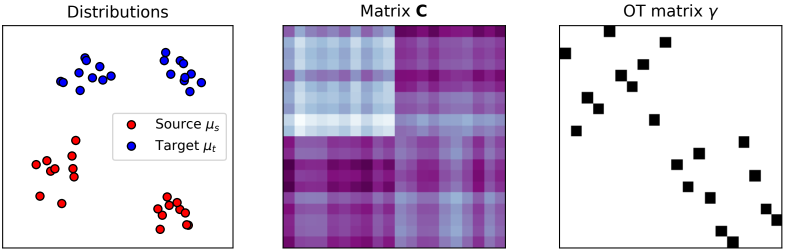
$$W_p^p(\mu_s, \mu_t) = \min_{\boldsymbol{T} \in \Pi(\mu_s, \mu_t)} \left\{ \langle \boldsymbol{T}, \mathbf{C} \rangle_F = \sum_{i,j} T_{i,j} c_{i,j} \right\}$$

where $\mathbf{C}$ is a cost matrix with $c_{i,j} = c(\mathbf{x}_i^s, \mathbf{x}_j^t) = \|\mathbf{x}_i^s - \mathbf{x}_j^t\|^p$ and the constraints are

$$\Pi(\mu_s, \mu_t) = \left\{ \boldsymbol{T} \in (\mathbb{R}^+)^{n_s \times n_t} | \, \boldsymbol{T}\mathbf{1}_{n_t} = \mathbf{a}, \boldsymbol{T}^T \mathbf{1}_{n_s} = \mathbf{b} \right\}$$

- Solving the OT problem with network simplex is $O(n^3 \log(n))$ for $n = n_s = n_t$.
- $W_p(\mu_s, \mu_t)$ is called the Wasserstein distance (EMD for $p = 1$).

Optimal transport between discrete distributions



Kantorovitch formulation : OT Linear Program

When $\mu_s = \sum_{i=1}^{n_s} a_i \delta_{\mathbf{x}_i^s}$ and $\mu_t = \sum_{i=1}^{n_t} b_i \delta_{\mathbf{x}_i^t}$

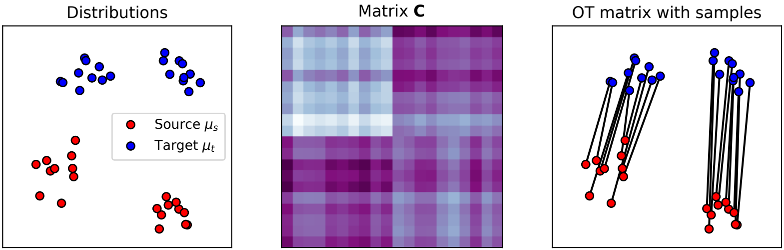
$$W_p^p(\mu_s, \mu_t) = \min_{\boldsymbol{T} \in \Pi(\mu_s, \mu_t)} \left\{ \langle \boldsymbol{T}, \mathbf{C} \rangle_F = \sum_{i,j} T_{i,j} c_{i,j} \right\}$$

where $\mathbf{C}$ is a cost matrix with $c_{i,j} = c(\mathbf{x}_i^s, \mathbf{x}_j^t) = \|\mathbf{x}_i^s - \mathbf{x}_j^t\|^p$ and the constraints are

$$\Pi(\mu_s, \mu_t) = \left\{ \boldsymbol{T} \in (\mathbb{R}^+)^{n_s \times n_t} | \, \boldsymbol{T}\mathbf{1}_{n_t} = \mathbf{a}, \boldsymbol{T}^T \mathbf{1}_{n_s} = \mathbf{b} \right\}$$

- Solving the OT problem with network simplex is $O(n^3 \log(n))$ for $n = n_s = n_t$.
- $W_p(\mu_s, \mu_t)$ is called the Wasserstein distance (EMD for $p = 1$).

Optimal transport between discrete distributions



Kantorovitch formulation : OT Linear Program

When $\mu_s = \sum_{i=1}^{n_s} a_i \delta_{\mathbf{x}_i^s}$ and $\mu_t = \sum_{i=1}^{n_t} b_i \delta_{\mathbf{x}_i^t}$

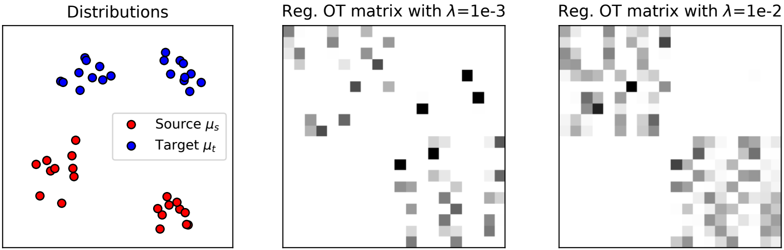
$$W_p^p(\mu_s, \mu_t) = \min_{\boldsymbol{T} \in \Pi(\mu_s, \mu_t)} \left\{ \langle \boldsymbol{T}, \mathbf{C} \rangle_F = \sum_{i,j} T_{i,j} c_{i,j} \right\}$$

where $\mathbf{C}$ is a cost matrix with $c_{i,j} = c(\mathbf{x}_i^s, \mathbf{x}_j^t) = \|\mathbf{x}_i^s - \mathbf{x}_j^t\|^p$ and the constraints are

$$\Pi(\mu_s, \mu_t) = \left\{ \boldsymbol{T} \in (\mathbb{R}^+)^{n_s \times n_t} | \, \boldsymbol{T}\mathbf{1}_{n_t} = \mathbf{a}, \boldsymbol{T}^T \mathbf{1}_{n_s} = \mathbf{b} \right\}$$

- Solving the OT problem with network simplex is $O(n^3 \log(n))$ for $n = n_s = n_t$.
- $W_p(\mu_s, \mu_t)$ is called the Wasserstein distance (EMD for $p = 1$).

Entropic regularized optimal transport



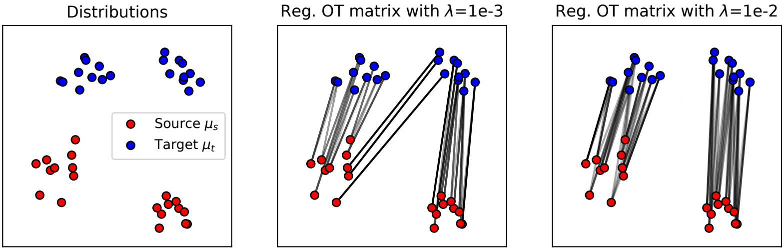
Entropic regularization [Cuturi, 2013]

$$\mathbf{T}_0^\lambda = \underset{\mathbf{T}\in\Pi(\mu_s,\mu_t)}{\arg\min} \quad \langle \mathbf{T}, \mathbf{C}\rangle_F + \lambda \sum_{i,j} T_{i,j}(\log T_{i,j} - 1)$$

- Regularization with the negative entropy of $\boldsymbol{T}$.
- Looses sparsity but smooth and strictly convex optimization problem.
- Can be solved efficiently with Sinkhorn's matrix scaling algorithm with $\mathbf{u}^{(0)} = \mathbf{1}, \mathbf{K} = \exp(-\mathbf{C}/\lambda)$ and $\mathbf{T} = \text{diag}(\mathbf{u}^\star)\mathbf{K}\text{diag}(\mathbf{v}^\star)$

$$\mathbf{v}^{(k)} = \mathbf{b} \oslash \mathbf{K}^\top \mathbf{u}^{(k-1)}, \quad \mathbf{u}^{(k)} = \mathbf{a} \oslash \mathbf{K}\mathbf{v}^{(k)}$$

Entropic regularized optimal transport



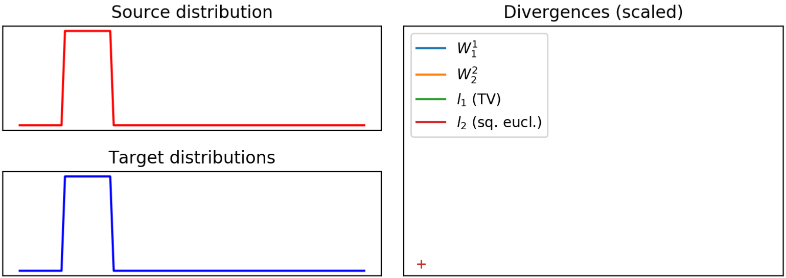
Entropic regularization [Cuturi, 2013]

$$\mathbf{T}_0^\lambda = \underset{\mathbf{T}\in\Pi(\mu_s,\mu_t)}{\arg\min} \quad \langle \mathbf{T},\mathbf{C}\rangle_F + \lambda\sum_{i,j} T_{i,j}(\log T_{i,j} - 1)$$

- Regularization with the negative entropy of $\boldsymbol{T}$.
- Looses sparsity but smooth and strictly convex optimization problem.
- Can be solved efficiently with Sinkhorn's matrix scaling algorithm with $\mathbf{u}^{(0)} = \mathbf{1}, \mathbf{K} = \exp(-\mathbf{C}/\lambda)$ and $\mathbf{T} = \text{diag}(\mathbf{u}^\star)\mathbf{K}\text{diag}(\mathbf{v}^\star)$

$$\mathbf{v}^{(k)} = \mathbf{b} \oslash \mathbf{K}^\top\mathbf{u}^{(k-1)}, \quad \mathbf{u}^{(k)} = \mathbf{a} \oslash \mathbf{K}\mathbf{v}^{(k)}$$

Wasserstein distance



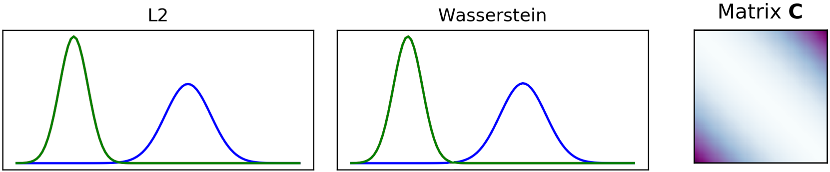
Wasserstein distance

$$W_p^p(\mu_s, \mu_t) = \min_{\gamma \in \mathcal{P}} \int_{\Omega_s \times \Omega_t} \|\mathbf{x} - \mathbf{y}\|^p \gamma(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y} = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\gamma}[\|\mathbf{x} - \mathbf{y}\|^p] \quad (1)$$

In this case we have $c(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|^p$

- A.K.A. Earth Mover's Distance (W_1^1) [Rubner et al., 2000].
- Useful between discrete distribution even without overlapping support.
- Smooth approximation can be computed with Sinkhorn [Cuturi, 2013].
- **Wasserstein barycenter**: $\overline{\mu} = \arg\min_\mu \sum_i w_i W_p^p(\mu, \mu_i)$

Wasserstein distance



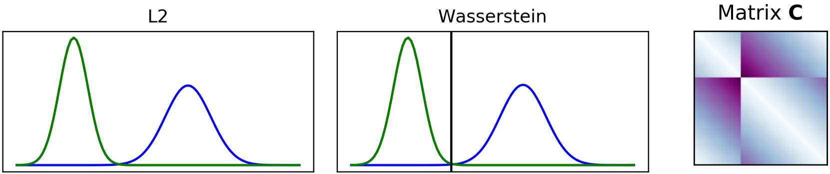
Wasserstein distance

$$W_p^p(\mu_s, \mu_t) = \min_{\gamma \in \mathcal{P}} \int_{\Omega_s \times \Omega_t} \|\mathbf{x} - \mathbf{y}\|^p \gamma(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y} = \mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim \gamma}[\|\mathbf{x} - \mathbf{y}\|^p] \quad (1)$$

In this case we have $c(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|^p$

- A.K.A. Earth Mover's Distance (W_1^1) [Rubner et al., 2000].
- Useful between discrete distribution even without overlapping support.
- Smooth approximation can be computed with Sinkhorn [Cuturi, 2013].
- **Wasserstein barycenter**: $\overline{\mu} = \arg\min_\mu \sum_i w_i W_p^p(\mu, \mu_i)$

Wasserstein distance

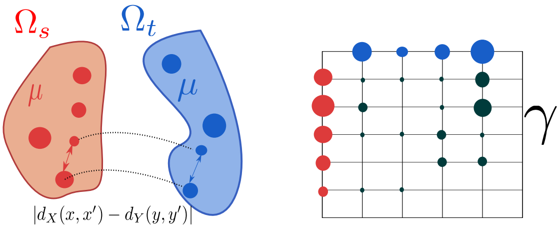


Wasserstein distance

$$W_p^p(\mu_s, \mu_t) = \min_{\gamma \in \mathcal{P}} \int_{\Omega_s \times \Omega_t} \|\mathbf{x} - \mathbf{y}\|^p \gamma(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y} = \mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim \gamma}[\|\mathbf{x} - \mathbf{y}\|^p] \quad (1)$$

In this case we have $c(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|^p$

- A.K.A. Earth Mover's Distance (W_1^1) [Rubner et al., 2000].
- Useful between discrete distribution even without overlapping support.
- Smooth approximation can be computed with Sinkhorn [Cuturi, 2013].
- **Wasserstein barycenter**: $\overline{\mu} = \arg\min_\mu \sum_i w_i W_p^p(\mu, \mu_i)$



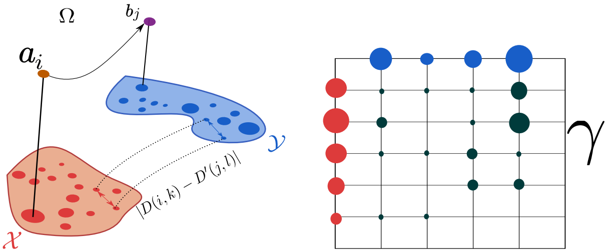
Inspired from Gabriel Peyré

GW for discrete distributions [Memoli, 2011]

$$\mathcal{GW}_p^p(\mu_s,\mu_t) = \min_{T\in\Pi(\mu_s,\mu_t)} \sum_{i,j,k,l} |D_{i,k} - D'_{j,l}|^p T_{i,j}\, T_{k,l}$$

with $\mu_s = \sum_i a_i \delta_{\mathbf{x}_i^s}$ and $\mu_t = \sum_j b_j \delta_{x_j^t}$ and $D_{i,k} = \|\mathbf{x}_i^s - \mathbf{x}_k^s\|$, $D'_{j,l} = \|\mathbf{x}_j^t - \mathbf{x}_l^t\|$

- Distance between metric measured spaces : across different spaces.
- Search for an OT plan that preserve the pairwise relationships between samples.
- Entropy regularized GW proposed in [Peyré et al., 2016].
- Fused GW interpolates between Wass. and GW [Vayer et al., 2018].



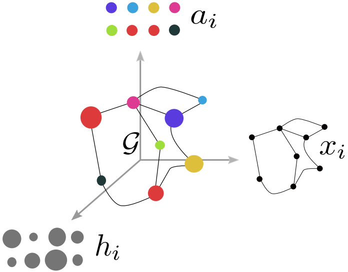
FGW for discrete distributions [Vayer et al., 2018]

$$\mathcal{FGW}_p^p(\mu_s, \mu_t) = \min_{T \in \Pi(\mu_s, \mu_t)} \sum_{i,j,k,l} \left((1-\alpha) C_{i,j}^q + \alpha |D_{i,k} - D'_{j,l}|^q\right)^p T_{i,j}\, T_{k,l}$$

with $\mu_s = \sum_i a_i \delta_{\mathbf{x}_i^s}$ and $\mu_t = \sum_j b_j \delta_{x_j^t}$ and $D_{i,k} = \|\mathbf{x}_i^s - \mathbf{x}_k^s\|$, $D'_{j,l} = \|\mathbf{x}_j^t - \mathbf{x}_l^t\|$

- Distance between metric measured spaces : across different spaces.
- Search for an OT plan that preserve the pairwise relationships between samples.
- Entropy regularized GW proposed in [Peyré et al., 2016].
- Fused GW interpolates between Wass. and GW [Vayer et al., 2018].

Gromov-Wasserstein between graphs



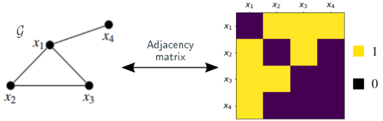
$$\mu = \sum_i h_i \delta_{(x_i, a_i)}$$

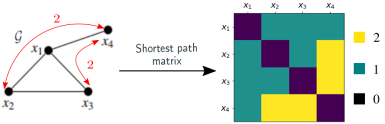
$$\mu_A = \sum_i h_i \delta_{a_i}$$

$$\mu_X = \sum_i h_i \delta_{x_i}$$

Graph as a distribution $(\boldsymbol{D}, \mathbf{F}, \boldsymbol{h})$

- The positions x_i are implicit and represented as the pairwise matrix $\boldsymbol{D}$.
- Possible choices for $\boldsymbol{D}$: Adjacency matrix, Laplacian, Shortest path, ...

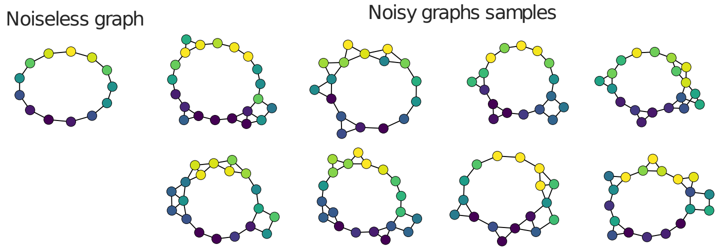




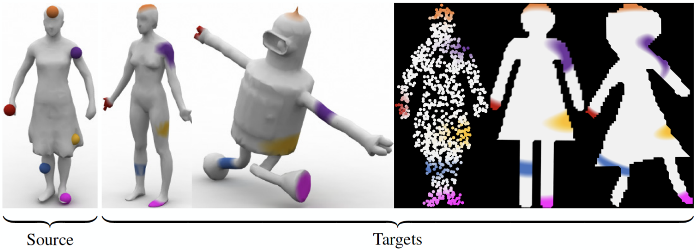
- The node features can be compared between graphs and stored in $\mathbf{F}$.
- h_i are the masses on the nodes of the graphs (uniform by default).

Applications of (F)GW

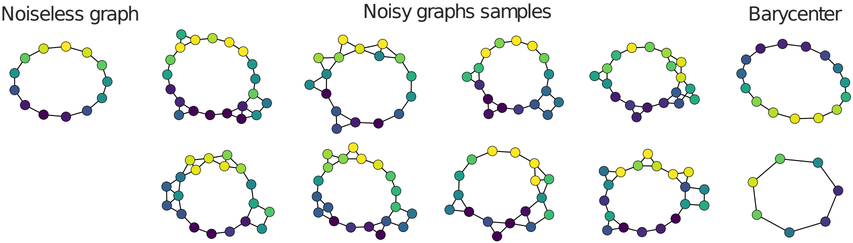
Barycenter/averaging of labeled graphs [Vayer et al., 2018]



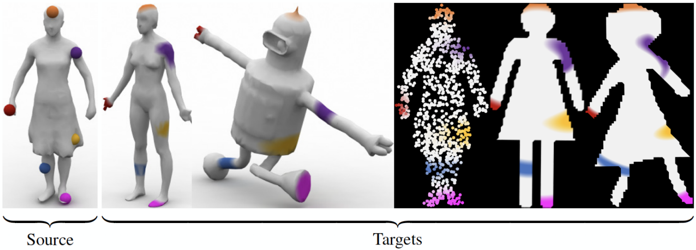
Shape matching between surfaces [Solomon et al., 2016, Thual et al., 2022]



Barycenter/averaging of labeled graphs [Vayer et al., 2018]

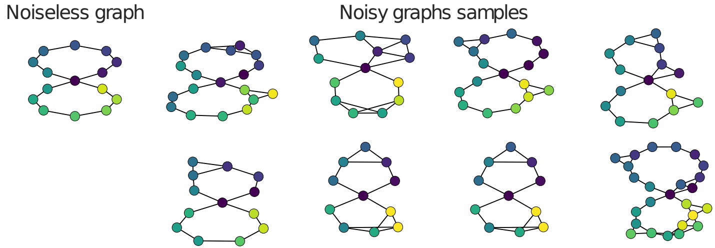


Shape matching between surfaces [Solomon et al., 2016, Thual et al., 2022]

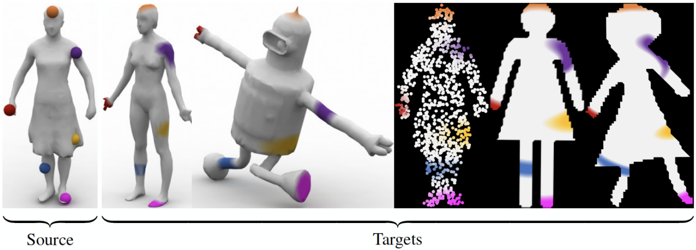


Applications of (F)GW

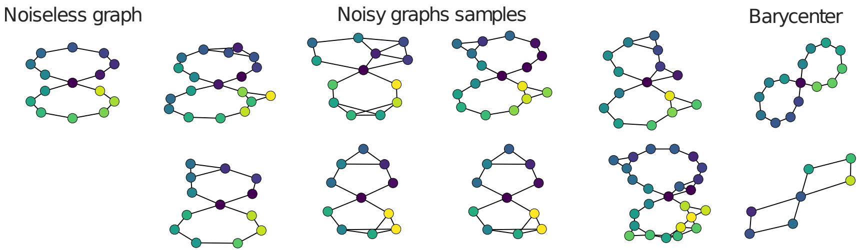
Barycenter/averaging of labeled graphs [Vayer et al., 2018]



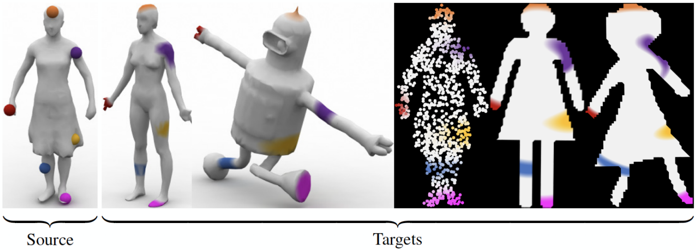
Shape matching between surfaces [Solomon et al., 2016, Thual et al., 2022]



Barycenter/averaging of labeled graphs [Vayer et al., 2018]

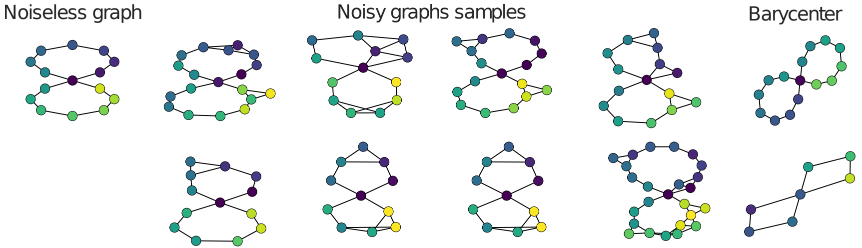


Shape matching between surfaces [Solomon et al., 2016, Thual et al., 2022]

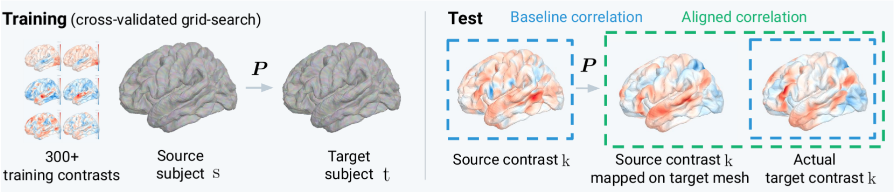


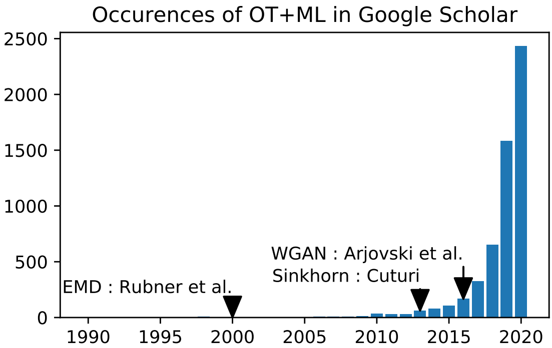
Applications of (F)GW

Barycenter/averaging of labeled graphs [Vayer et al., 2018]



Shape matching between surfaces [Solomon et al., 2016, Thual et al., 2022]

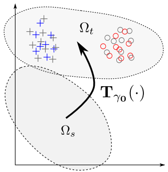


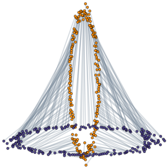


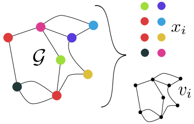
Short history of OT for ML

- Proposed in in image processing by [Rubner et al., 2000] (EMD).
- Entropic regularized OT allows fast approximation [Cuturi, 2013].
- Deep learning/ stochastic optimization [Arjovsky et al., 2017].
- Generative models with diffusion/Schrödinger bridges.

Three aspects of optimal transport







Transporting with optimal transport

- Learn to map between distributions.
- Estimate a smooth mapping from discrete distributions.
- Applications in domain adaptation.

Divergence between histograms/empirical distributions

- Use the ground metric to encode complex relations between the bins of histograms for data fitting.
- OT losses are non-parametric divergences between non overlapping distributions.
- Used to train minimal Wasserstein estimators.

Divergence between structured objects and spaces

- Modeling of structured data and graphs as distribution.
- OT losses (Wass. or (F)GW) measure similarity between distributions/objects.
- OT find correspondance across spaces for adaptation.

POT in scientific research

Optimal transport for single-cell and spatial omics

Charlotte Bunne, Geoffrey Schiebinger, Andreas Krause, Aviv Regev & Marco Cuturi

Nature Reviews Methods Primers **4**, Article number: 58 (2024) | Cite this article

SCOT: Single-Cell Multi-Omics Alignment with Optimal Transport

Authors: Pinar Demetci, Rebecca Santorella, Björn Sandstede, William Stafford Noble, and Ritambhara Singh | AUTHORS

Publication: Journal of Computational Biology • https://doi.org/10.1089/cmb.2021.0446

EMB IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, VOL. 69, NO. 2, FEBRUARY 2022 807

Transfer Learning Based on Optimal Transport for Motor Imagery Brain-Computer Interfaces

Victoria Peterson, Nicolás Nieto, Dominik Wyser, Olivier Lambercy, *Member, IEEE*, Roger Gassert, *Senior Member, IEEE*, Diego H. Milone, and Rubén D. Spies

JHEP Published for SISSA by Springer

Received: *May 29, 2023*
Revised: *August 30, 2023*
Accepted: *October 2, 2023*
Published: *October 10, 2023*

Measurements of multijet event isotropies using optimal transport with the ATLAS detector

ATLAS EXPERIMENT

The ATLAS collaboration

E-mail: atlas.publications@cern.ch

Simulation-Free Schrödinger Bridges via Score and Flow Matching

Alexander Tong[†] Mila – Québec AI Institute, Université de Montréal

Nikolay Malkin[†] Mila – Québec AI Institute, Université de Montréal

Kilian Fatras[†] Mila – Québec AI Institute, McGill University

Lazar Atanackovic University of Toronto, Vector Institute

Yanlei Zhang Mila – Québec AI Institute, Université de Montréal

Guillaume Huguet Mila – Québec AI Institute, Université de Montréal

Guy Wolf Mila – Québec AI Institute, Université de Montréal, Canada CIFAR AI Chair

Yoshua Bengio Mila – Québec AI Institute, Université de Montréal, CIFAR Senior Fellow

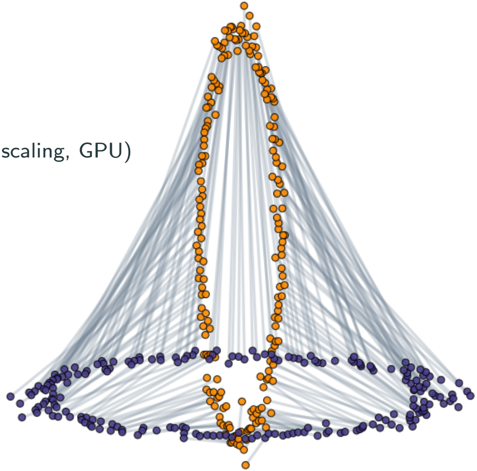
Thank you

Python code available on GitHub:

Python code available on GitHub:

`https://github.com/PythonOT/POT`

- OT LP solver, Sinkhorn (stabilized, $\epsilon-$scaling, GPU)
- Domain adaptation with OT.
- Barycenters, Wasserstein unmixing.
- Wasserstein Discriminant Analysis.



Tutorial on OT for ML:

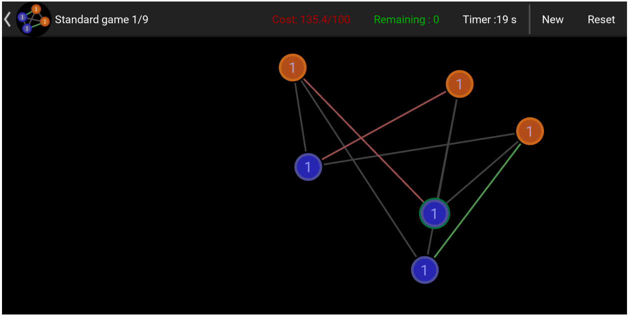
`http://tinyurl.com/otml-isbi`

Papers available on my website:

`https://remi.flamary.com/`

OTGame (OT Puzzle game on android)

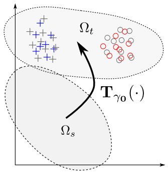
OTGame

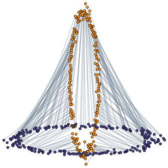


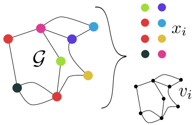
https://play.google.com/store/apps/details?id=com.flamary.otgame

Supplementary material

Three aspects of optimal transport







Transporting with optimal transport

- Learn to map between distributions.
- Estimate a smooth mapping from discrete distributions.
- Applications in domain adaptation.

Divergence between histograms/empirical distributions

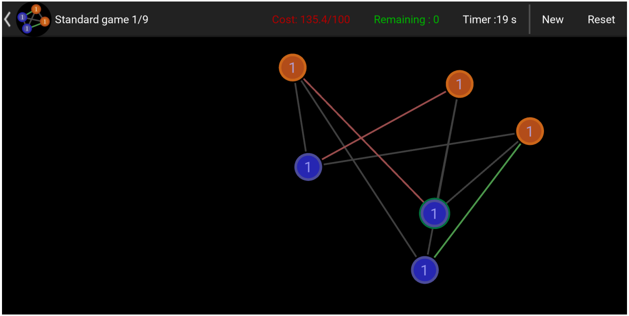
- Use the ground metric to encode complex relations between the bins of histograms for data fitting.
- OT losses are non-parametric divergences between non overlapping distributions.
- Used to train minimal Wasserstein estimators.

Divergence between structured objects and spaces

- Modeling of structured data and graphs as distribution.
- OT losses (Wass. or (F)GW) measure similarity between distributions/objects.
- OT find correspondance across spaces for adaptation.

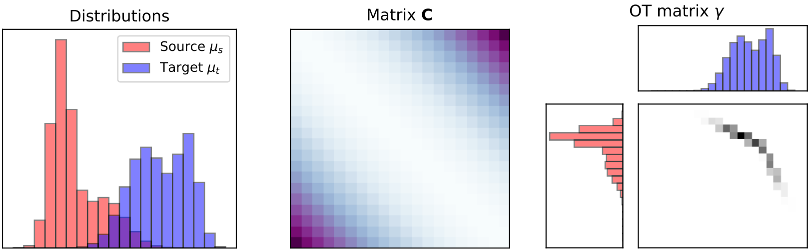
OTGame (OT Puzzle game on android)

OTGame



https://play.google.com/store/apps/details?id=com.flamary.otgame

Optimal transport between discrete distributions



Kantorovitch formulation : OT Linear Program

When $\mu_s = \sum_{i=1}^{n_s} a_i \delta_{\mathbf{x}_i^s}$ and $\mu_t = \sum_{i=1}^{n_t} b_i \delta_{\mathbf{x}_i^t}$

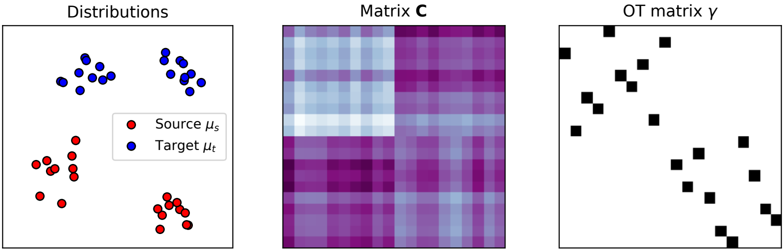
$$W_p^p(\mu_s, \mu_t) = \min_{\boldsymbol{T} \in \Pi(\mu_s, \mu_t)} \left\{ \langle \boldsymbol{T}, \mathbf{C} \rangle_F = \sum_{i,j} T_{i,j} c_{i,j} \right\}$$

where $\mathbf{C}$ is a cost matrix with $c_{i,j} = c(\mathbf{x}_i^s, \mathbf{x}_j^t) = \|\mathbf{x}_i^s - \mathbf{x}_j^t\|^p$ and the constraints are

$$\Pi(\mu_s, \mu_t) = \left\{ \boldsymbol{T} \in (\mathbb{R}^+)^{n_s \times n_t} | \, \boldsymbol{T}\mathbf{1}_{n_t} = \mathbf{a}, \boldsymbol{T}^T \mathbf{1}_{n_s} = \mathbf{b} \right\}$$

- Solving the OT problem with network simplex is $O(n^3 \log(n))$ for $n = n_s = n_t$.
- $W_p(\mu_s, \mu_t)$ is called the Wasserstein distance (EMD for $p = 1$).

Optimal transport between discrete distributions



Kantorovitch formulation : OT Linear Program

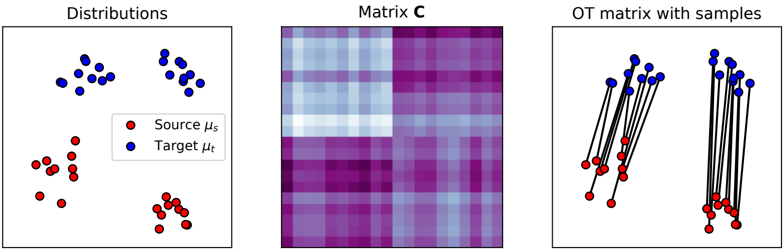
When $\mu_s = \sum_{i=1}^{n_s} a_i \delta_{\mathbf{x}_i^s}$ and $\mu_t = \sum_{i=1}^{n_t} b_i \delta_{\mathbf{x}_i^t}$

$$W_p^p(\mu_s, \mu_t) = \min_{\boldsymbol{T} \in \Pi(\mu_s, \mu_t)} \left\{ \langle \boldsymbol{T}, \mathbf{C} \rangle_F = \sum_{i,j} T_{i,j} c_{i,j} \right\}$$

where $\mathbf{C}$ is a cost matrix with $c_{i,j} = c(\mathbf{x}_i^s, \mathbf{x}_j^t) = \|\mathbf{x}_i^s - \mathbf{x}_j^t\|^p$ and the constraints are

$$\Pi(\mu_s, \mu_t) = \left\{ \boldsymbol{T} \in (\mathbb{R}^+)^{n_s \times n_t} | \, \boldsymbol{T}\mathbf{1}_{n_t} = \mathbf{a}, \boldsymbol{T}^T \mathbf{1}_{n_s} = \mathbf{b} \right\}$$

- Solving the OT problem with network simplex is $O(n^3 \log(n))$ for $n = n_s = n_t$.
- $W_p(\mu_s, \mu_t)$ is called the Wasserstein distance (EMD for $p = 1$).



Kantorovitch formulation : OT Linear Program

When $\mu_s = \sum_{i=1}^{n_s} a_i \delta_{\mathbf{x}_i^s}$ and $\mu_t = \sum_{i=1}^{n_t} b_i \delta_{\mathbf{x}_i^t}$

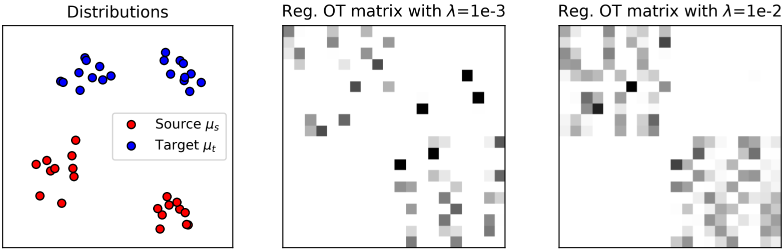
$$W_p^p(\mu_s, \mu_t) = \min_{\boldsymbol{T} \in \Pi(\mu_s, \mu_t)} \left\{ \langle \boldsymbol{T}, \mathbf{C} \rangle_F = \sum_{i,j} T_{i,j} c_{i,j} \right\}$$

where $\mathbf{C}$ is a cost matrix with $c_{i,j} = c(\mathbf{x}_i^s, \mathbf{x}_j^t) = \|\mathbf{x}_i^s - \mathbf{x}_j^t\|^p$ and the constraints are

$$\Pi(\mu_s, \mu_t) = \left\{ \boldsymbol{T} \in (\mathbb{R}^+)^{n_s \times n_t} \,|\, \boldsymbol{T}\mathbf{1}_{n_t} = \mathbf{a}, \boldsymbol{T}^T \mathbf{1}_{n_s} = \mathbf{b} \right\}$$

- Solving the OT problem with network simplex is $O(n^3 \log(n))$ for $n = n_s = n_t$.
- $W_p(\mu_s, \mu_t)$ is called the Wasserstein distance (EMD for $p = 1$).

Entropic regularized optimal transport



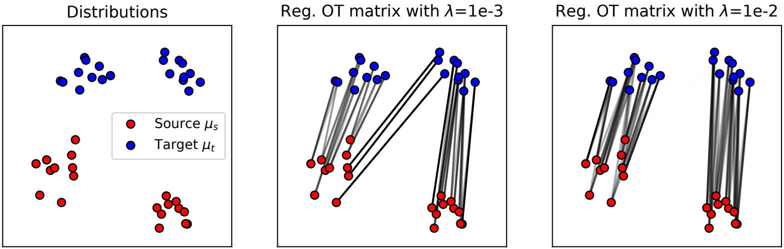
Entropic regularization [Cuturi, 2013]

$$\mathbf{T}_0^\lambda = \underset{\mathbf{T}\in\Pi(\mu_s,\mu_t)}{\arg\min} \quad \langle \mathbf{T}, \mathbf{C}\rangle_F + \lambda \sum_{i,j} T_{i,j}(\log T_{i,j} - 1)$$

- Regularization with the negative entropy of $\boldsymbol{T}$.
- Looses sparsity but smooth and strictly convex optimization problem.
- Can be solved efficiently with Sinkhorn's matrix scaling algorithm with $\mathbf{u}^{(0)} = \mathbf{1}, \mathbf{K} = \exp(-\mathbf{C}/\lambda)$ and $\mathbf{T} = \text{diag}(\mathbf{u}^\star)\mathbf{K}\text{diag}(\mathbf{v}^\star)$

$$\mathbf{v}^{(k)} = \mathbf{b} \oslash \mathbf{K}^\top \mathbf{u}^{(k-1)}, \quad \mathbf{u}^{(k)} = \mathbf{a} \oslash \mathbf{K}\mathbf{v}^{(k)}$$

Entropic regularized optimal transport



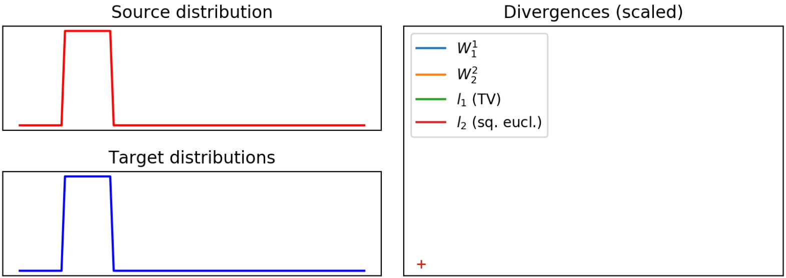
Entropic regularization [Cuturi, 2013]

$$\mathbf{T}_0^\lambda = \underset{\mathbf{T}\in\Pi(\mu_s,\mu_t)}{\arg\min} \quad \langle \mathbf{T}, \mathbf{C}\rangle_F + \lambda \sum_{i,j} T_{i,j}(\log T_{i,j} - 1)$$

- Regularization with the negative entropy of $\boldsymbol{T}$.
- Looses sparsity but smooth and strictly convex optimization problem.
- Can be solved efficiently with Sinkhorn's matrix scaling algorithm with $\mathbf{u}^{(0)} = \mathbf{1}, \mathbf{K} = \exp(-\mathbf{C}/\lambda)$ and $\mathbf{T} = \text{diag}(\mathbf{u}^\star)\mathbf{K}\text{diag}(\mathbf{v}^\star)$

$$\mathbf{v}^{(k)} = \mathbf{b} \oslash \mathbf{K}^\top \mathbf{u}^{(k-1)}, \quad \mathbf{u}^{(k)} = \mathbf{a} \oslash \mathbf{K}\mathbf{v}^{(k)}$$

Wasserstein distance



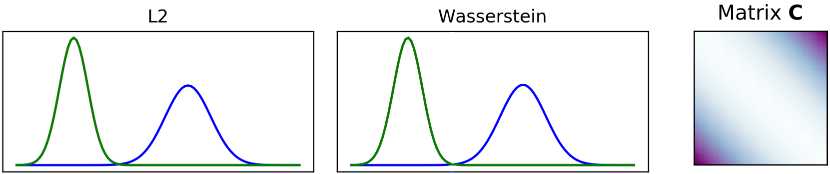
Wasserstein distance

$$W_p^p(\mu_s, \mu_t) = \min_{\gamma \in \mathcal{P}} \int_{\Omega_s \times \Omega_t} \|\mathbf{x} - \mathbf{y}\|^p \gamma(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y} = \mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim \gamma}[\|\mathbf{x} - \mathbf{y}\|^p] \quad (2)$$

In this case we have $c(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|^p$

- A.K.A. Earth Mover's Distance (W_1^1) [Rubner et al., 2000].
- Useful between discrete distribution even without overlapping support.
- Smooth approximation can be computed with Sinkhorn [Cuturi, 2013].
- **Wasserstein barycenter**: $\overline{\mu} = \arg\min_\mu \sum_i w_i W_p^p(\mu, \mu_i)$

Wasserstein distance



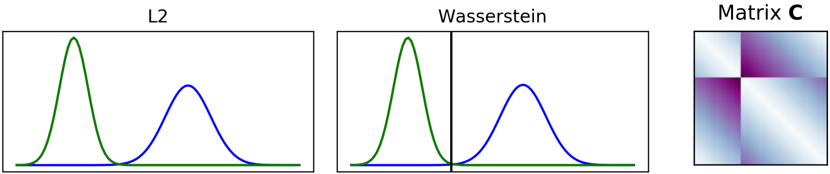
Wasserstein distance

$$W_p^p(\mu_s, \mu_t) = \min_{\gamma \in \mathcal{P}} \int_{\Omega_s \times \Omega_t} \|\mathbf{x} - \mathbf{y}\|^p \gamma(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y} = \mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim \gamma}[\|\mathbf{x} - \mathbf{y}\|^p] \qquad (2)$$

In this case we have $c(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|^p$

- A.K.A. Earth Mover's Distance (W_1^1) [Rubner et al., 2000].
- Useful between discrete distribution even without overlapping support.
- Smooth approximation can be computed with Sinkhorn [Cuturi, 2013].
- **Wasserstein barycenter**: $\overline{\mu} = \arg\min_\mu \sum_i w_i W_p^p(\mu, \mu_i)$

Wasserstein distance



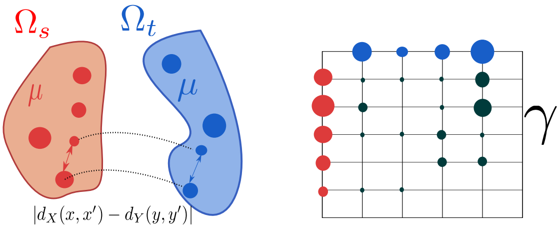
Wasserstein distance

$$W_p^p(\mu_s, \mu_t) = \min_{\gamma \in \mathcal{P}} \int_{\Omega_s \times \Omega_t} \|\mathbf{x} - \mathbf{y}\|^p \gamma(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y} = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\gamma}[\|\mathbf{x} - \mathbf{y}\|^p] \tag{2}$$

In this case we have $c(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|^p$

- A.K.A. Earth Mover's Distance (W_1^1) [Rubner et al., 2000].
- Useful between discrete distribution even without overlapping support.
- Smooth approximation can be computed with Sinkhorn [Cuturi, 2013].
- **Wasserstein barycenter**: $\overline{\mu} = \arg\min_\mu \sum_i w_i W_p^p(\mu, \mu_i)$

Gromov-Wasserstein and extensions



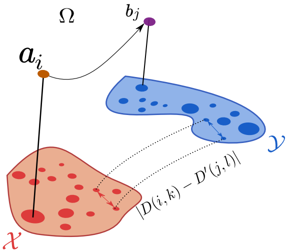
Inspired from Gabriel Peyré

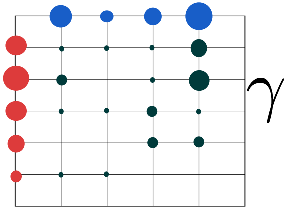
GW for discrete distributions [Memoli, 2011]

$$\mathcal{GW}_p^p(\mu_s,\mu_t) = \min_{T\in\Pi(\mu_s,\mu_t)} \sum_{i,j,k,l} |D_{i,k} - D'_{j,l}|^p T_{i,j}\, T_{k,l}$$

with $\mu_s = \sum_i a_i \delta_{\mathbf{x}_i^s}$ and $\mu_t = \sum_j b_j \delta_{x_j^t}$ and $D_{i,k} = \|\mathbf{x}_i^s - \mathbf{x}_k^s\|$, $D'_{j,l} = \|\mathbf{x}_j^t - \mathbf{x}_l^t\|$

- Distance between metric measured spaces : across different spaces.
- Search for an OT plan that preserve the pairwise relationships between samples.
- Entropy regularized GW proposed in [Peyré et al., 2016].
- Fused GW interpolates between Wass. and GW [Vayer et al., 2018].



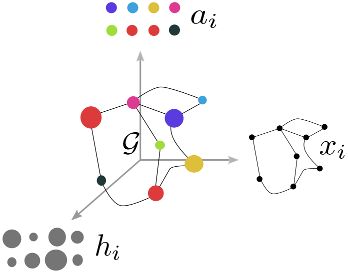


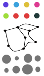
FGW for discrete distributions [Vayer et al., 2018]

$$\mathcal{FGW}_p^p(\mu_s, \mu_t) = \min_{T \in \Pi(\mu_s, \mu_t)} \sum_{i,j,k,l} \left((1-\alpha) C_{i,j}^q + \alpha |D_{i,k} - D'_{j,l}|^q\right)^p T_{i,j}\, T_{k,l}$$

with $\mu_s = \sum_i a_i \delta_{\mathbf{x}_i^s}$ and $\mu_t = \sum_j b_j \delta_{x_j^t}$ and $D_{i,k} = \|\mathbf{x}_i^s - \mathbf{x}_k^s\|, D'_{j,l} = \|\mathbf{x}_j^t - \mathbf{x}_l^t\|$

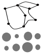
- Distance between metric measured spaces : across different spaces.
- Search for an OT plan that preserve the pairwise relationships between samples.
- Entropy regularized GW proposed in [Peyré et al., 2016].
- Fused GW interpolates between Wass. and GW [Vayer et al., 2018].





$$\mu = \sum_i h_i \delta_{(x_i, a_i)}$$

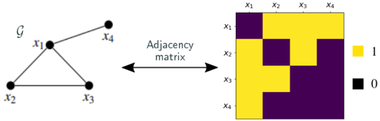
$$\mu_A = \sum_i h_i \delta_{a_i}$$

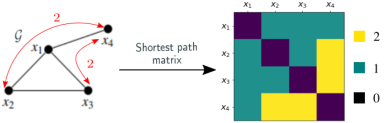


$$\mu_X = \sum_i h_i \delta_{x_i}$$

Graph as a distribution $(\boldsymbol{D}, \mathbf{F}, \boldsymbol{h})$

- The positions x_i are implicit and represented as the pairwise matrix $\boldsymbol{D}$.
- Possible choices for $\boldsymbol{D}$: Adjacency matrix, Laplacian, Shortest path, ...

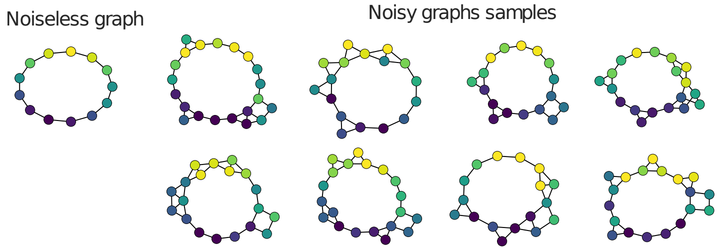




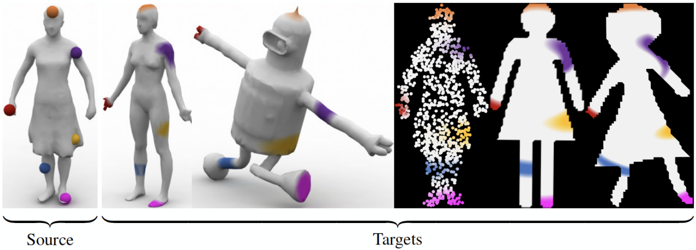
- The node features can be compared between graphs and stored in $\mathbf{F}$.
- h_i are the masses on the nodes of the graphs (uniform by default).

Applications of (F)GW

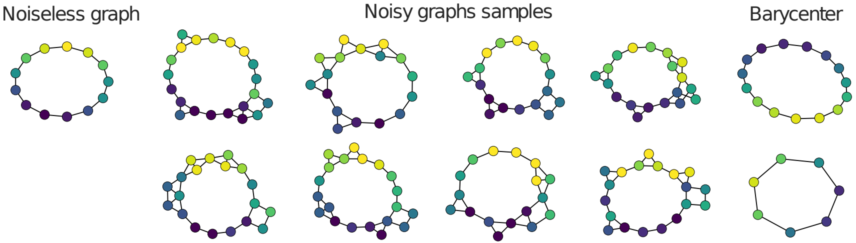
Barycenter/averaging of labeled graphs [Vayer et al., 2018]



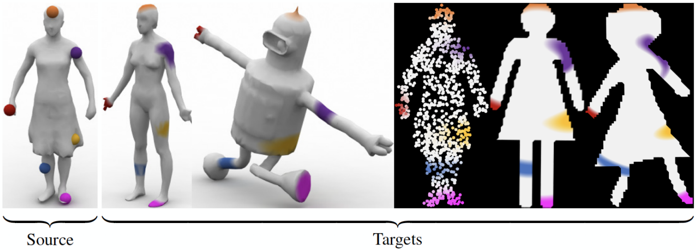
Shape matching between surfaces [Solomon et al., 2016, Thual et al., 2022]



Barycenter/averaging of labeled graphs [Vayer et al., 2018]



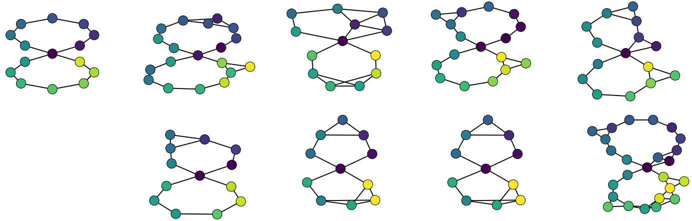
Shape matching between surfaces [Solomon et al., 2016, Thual et al., 2022]



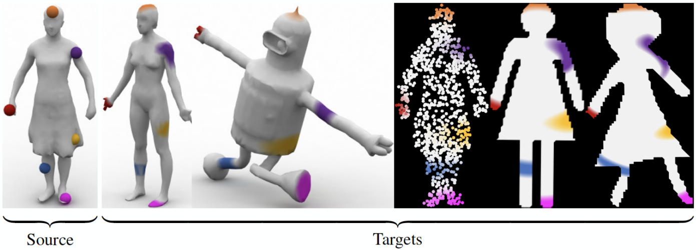
Applications of (F)GW

Barycenter/averaging of labeled graphs [Vayer et al., 2018]

Noiseless graph　　　　　　　　Noisy graphs samples

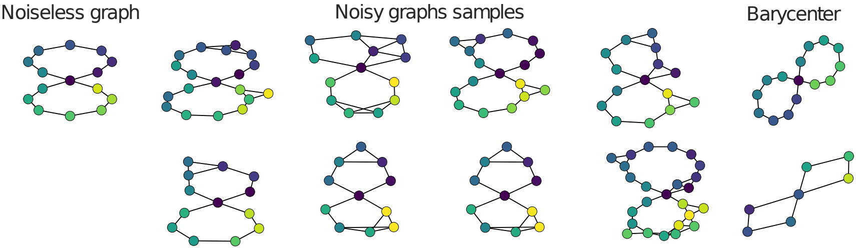


Shape matching between surfaces [Solomon et al., 2016, Thual et al., 2022]

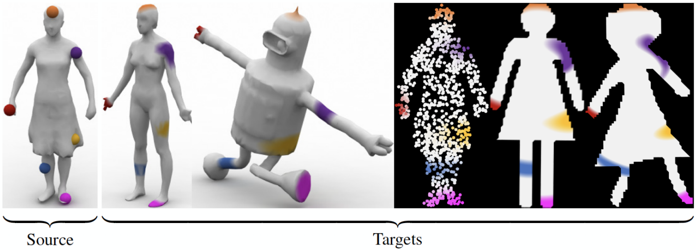


Applications of (F)GW

Barycenter/averaging of labeled graphs [Vayer et al., 2018]

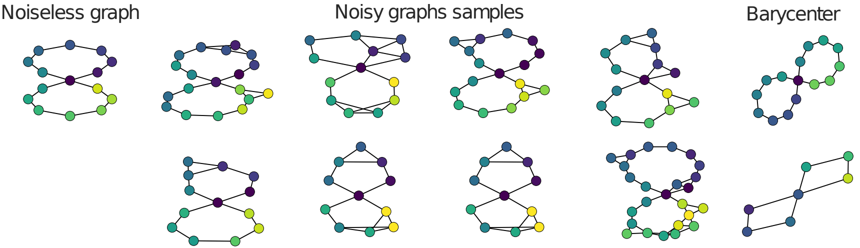


Shape matching between surfaces [Solomon et al., 2016, Thual et al., 2022]

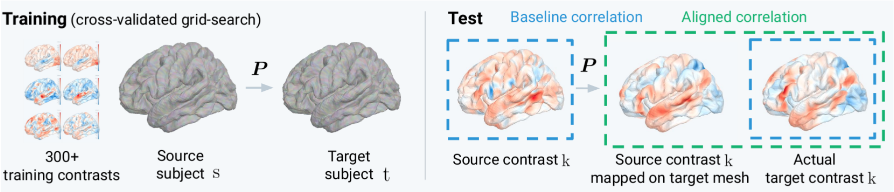


Applications of (F)GW

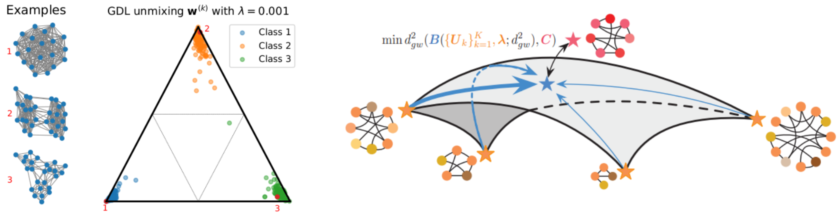
Barycenter/averaging of labeled graphs [Vayer et al., 2018]



Shape matching between surfaces [Solomon et al., 2016, Thual et al., 2022]



Graph Dictionary Learning



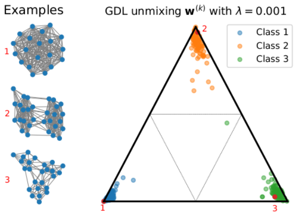
Representation learning for graphs

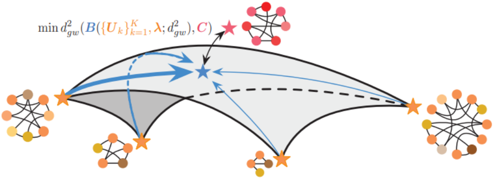
- Learn a dictionary $\{\overline{\mathbf{C}_i}\}_i$ of graph templates to describe a continuous manifold.
- The representation is learned by minimizing the (F)GW distance between the graph reconstruction from the embedding in the dictionary.
- Online Graph Dictionary learning : Linear model [Vincent-Cuaz et al., 2021].

$$\widehat{\mathbf{C}} = \sum_i w_i \overline{\mathbf{C}_i}$$

- GW Factorization : Nonlinear (GW barycenter) model [Xu, 2020].
- Dictionary for structured prediction with GW bary. [Brogat-Motte et al., 2022].

Graph Dictionary Learning





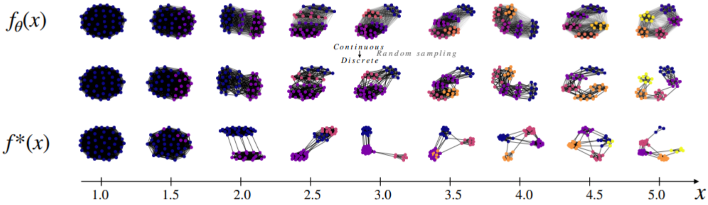
Representation learning for graphs

- Learn a dictionary $\{\overline{\mathbf{C}_i}\}_i$ of graph templates to describe a continuous manifold.
- The representation is learned by minimizing the (F)GW distance between the graph reconstruction from the embedding in the dictionary.
- Online Graph Dictionary learning : Linear model [Vincent-Cuaz et al., 2021].
- GW Factorization : Nonlinear (GW barycenter) model [Xu, 2020].

$$\widehat{\mathbf{C}} = \arg\min_{\mathbf{C}} \sum_i w_i GW(\mathbf{C}, \overline{\mathbf{C}_i})$$

- Dictionary for structured prediction with GW bary. [Brogat-Motte et al., 2022].

Graph Dictionary Learning

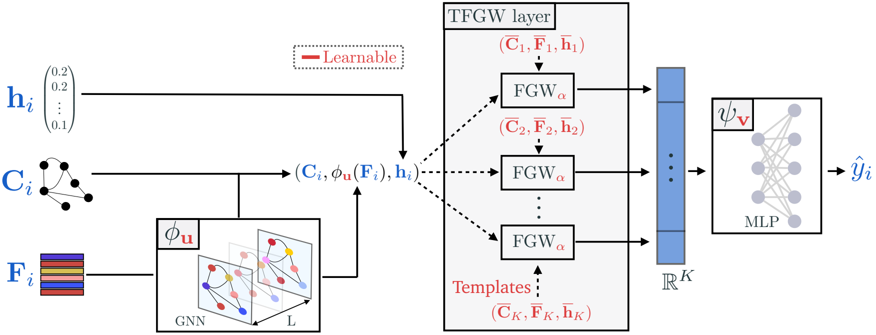


Representation learning for graphs

- Learn a dictionary $\{\overline{\mathbf{C}_i}\}_i$ of graph templates to describe a continuous manifold.
- The representation is learned by minimizing the (F)GW distance between the graph reconstruction from the embedding in the dictionary.
- Online Graph Dictionary learning : Linear model [Vincent-Cuaz et al., 2021].
- GW Factorization : Nonlinear (GW barycenter) model [Xu, 2020].
- Dictionary for structured prediction with GW bary. [Brogat-Motte et al., 2022].

$$f(\mathbf{x}) = \widehat{\mathbf{C}}(\mathbf{x}) = \arg\min_{\mathbf{C}} \sum_i w_i(\mathbf{x}) GW(\mathbf{C}, \overline{\mathbf{C}_i})$$

FGW for a pooling layer in GNN



Template based FGW layer (TFGW) [Vincent-Cuaz et al., 2022]

- Principle: represent a graph through its distances to learned templates.
- Learnable parameters are illustrated in red above.
- New end-to-end GNN models for graph-level tasks.
- Sate-of-the-art (still!) on graph classification (1×#1, 3×#2 on paperwithcode).

References i

[Arjovsky et al., 2017] Arjovsky, M., Chintala, S., and Bottou, L. (2017).
Wasserstein generative adversarial networks.
In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pages 214–223, Sydney, Australia.

[Brogat-Motte et al., 2022] Brogat-Motte, L., Flamary, R., Brouard, C., Rousu, J., and d'Alché Buc, F. (2022).
Learning to predict graphs with fused gromov-wasserstein barycenters.
In *International Conference in Machine Learning (ICML)*.

[Cuturi, 2013] Cuturi, M. (2013).
Sinkhorn distances: Lightspeed computation of optimal transport.
In *NIPS*, pages 2292–2300.

[Kantorovich, 1942] Kantorovich, L. (1942).
On the translocation of masses.
C.R. (Doklady) Acad. Sci. URSS (N.S.), 37:199–201.

References ii

[Memoli, 2011] Memoli, F. (2011).
Gromov wasserstein distances and the metric approach to object matching.
Foundations of Computational Mathematics, pages 1–71.

[Monge, 1781] Monge, G. (1781).
Mémoire sur la théorie des déblais et des remblais.
De l'Imprimerie Royale.

[Peyré et al., 2016] Peyré, G., Cuturi, M., and Solomon, J. (2016).
Gromov-wasserstein averaging of kernel and distance matrices.
In *ICML*, pages 2664–2672.

[Rubner et al., 2000] Rubner, Y., Tomasi, C., and Guibas, L. J. (2000).
The earth mover's distance as a metric for image retrieval.
International journal of computer vision, 40(2):99–121.

[Solomon et al., 2016] Solomon, J., Peyré, G., Kim, V. G., and Sra, S. (2016).
Entropic metric alignment for correspondence problems.
ACM Transactions on Graphics (TOG), 35(4):72.

References iii

[Thual et al., 2022] Thual, A., Tran, H., Zemskova, T., Courty, N., Flamary, R., Dehaene, S., and Thirion, B. (2022).

Aligning individual brains with fused unbalanced gromov-wasserstein.

In *Neural Information Processing Systems (NeurIPS)*.

[Vayer et al., 2018] Vayer, T., Chapel, L., Flamary, R., Tavenard, R., and Courty, N. (2018).

Fused gromov-wasserstein distance for structured objects: theoretical foundations and mathematical properties.

[Vincent-Cuaz et al., 2022] Vincent-Cuaz, C., Flamary, R., Corneli, M., Vayer, T., and Courty, N. (2022).

Template based graph neural network with optimal transport distances.

In *Neural Information Processing Systems (NeurIPS)*.

[Vincent-Cuaz et al., 2021] Vincent-Cuaz, C., Vayer, T., Flamary, R., Corneli, M., and Courty, N. (2021).

Online graph dictionary learning.

In *International Conference on Machine Learning (ICML)*.

References iv

[Xu, 2020] Xu, H. (2020).
Gromov-wasserstein factorization models for graph clustering.
In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6478–6485.