Optimal Transport for Machine Learning

10 years of least effort

Rémi Flamary, École polytechnique
October 2nd 2023

Mathematical Foundations of AI, 2023, Paris
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 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.
# Table of content

## Introduction
- OT problem and mathematical tools
- Optimal Transport and Machine Learning

## Mapping with optimal transport
- Mapping with optimal transport from discrete samples
- Optimal transport for domain adaptation

## Optimal Transport as a distance between distributions
- OT between histogram data
- OT between empirical distributions

## Optimal Transport between spaces and structures
- Gromov-Wasserstein and extensions
- Applications of OT between graphs

## Conclusion
Outline

Introduction

- OT problem and mathematical tools
- Optimal Transport and Machine Learning

Mapping with optimal transport

- Mapping with optimal transport from discrete samples
- Optimal transport for domain adaptation

Optimal Transport as a distance between distributions

- OT between histogram data
- OT between empirical distributions

Optimal Transport between spaces and structures

- Gromov-Wasserstein and extensions
- Applications of OT between graphs

Conclusion
Optimal transport

- Problem introduced by Gaspard Monge in his memoir [Monge, 1781].
- How to move mass while minimizing a cost (mass + cost)
- Monge formulation seeks for a mapping between two mass distributions.
- Focus on where the mass goes, allow splitting [Kantorovich, 1942].
- Applications originally for resource allocation problems.
Kantorovitch formulation : OT Linear Program

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

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

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

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

- 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 \( \mu_s = \sum_{i=1}^{n_s} a_i \delta_{x_i^s} \) and \( \mu_t = \sum_{i=1}^{n_t} b_i \delta_{x_i^t} \)

\[
W_p^P(\mu_s, \mu_t) = \min_{T \in \Pi(\mu_s, \mu_t)} \left\{ \langle T, C \rangle_F = \sum_{i,j} T_{i,j} c_{i,j} \right\}
\]

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

\[
\Pi(\mu_s, \mu_t) = \left\{ T \in (\mathbb{R}^+)^{n_s \times n_t} \mid T1_{n_t} = a, T^T1_{n_s} = b \right\}
\]

- 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

Distributions

Source $\mu_s$

Target $\mu_t$

Matrix $C$

OT matrix with samples

Kantorovitch formulation : OT Linear Program

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

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

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

$$\Pi(\mu_s, \mu_t) = \left\{ T \in (\mathbb{R}^+)^{n_s \times n_t} \mid T1_{n_t} = a, T^T 1_{n_s} = b \right\}$$

- 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 regularization [Cuturi, 2013]

\[
T_0^\lambda = \arg \min_{T \in \Pi(\mu_s, \mu_t)} \langle T, C \rangle_F + \lambda \sum_{i,j} T_{i,j} (\log T_{i,j} - 1)
\]

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

\[
v^{(k)} = b \odot K^T u^{(k-1)}, \quad u^{(k)} = a \odot K v^{(k)}
\]
Entropic regularization [Cuturi, 2013]

\[ T_0^\lambda = \arg \min_{T \in \Pi(\mu_s, \mu_t)} \langle T, C \rangle_F + \lambda \sum_{i,j} T_{i,j} (\log T_{i,j} - 1) \]

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

\[ v^{(k)} = b \odot K^\top u^{(k-1)}, \quad u^{(k)} = a \odot K v^{(k)} \]
Wasserstein distance

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

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

- 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**: \(\bar{\mu} = \arg \min_{\mu} \sum_i w_i W_p^p(\mu, \mu_i)\)
Wasserstein distance

\[ W_p^p(\mu_s, \mu_t) = \min_{\gamma \in P} \int_{\Omega_s \times \Omega_t} \|x - y\|^p \gamma(x, y) dx dy = \mathbb{E}_{(x,y) \sim \gamma}[\|x - y\|^p] \tag{1} \]

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

- 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**: \( \bar{\mu} = \arg \min_{\mu} \sum_i w_i W_p^p(\mu, \mu_i) \)
Wasserstein distance

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

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

- 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**: \(\bar{\mu} = \arg\min_{\mu} \sum_i w_i W_p^p(\mu, \mu_i)\)
Short history of OT for ML

- Proposed 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 correspondence across spaces for adaptation.
Introduction

OT problem and mathematical tools

Optimal Transport and Machine Learning

Mapping with optimal transport

Mapping with optimal transport from discrete samples

Optimal transport for domain adaptation

Optimal Transport as a distance between distributions

OT between histogram data

OT between empirical distributions

Optimal Transport between spaces and structures

Gromov-Wasserstein and extensions

Applications of OT between graphs

Conclusion
Mapping with optimal transport

Distributions

Classt OT

Reg. Entropic OT

Mapping estimation

- Barycentric mapping using the OT matrix $T$ [Ferradans et al., 2014].

$$\hat{m}_T(x_i^s) = \arg \min_x \sum_j T_{i,j} c(x, x_j^t)$$

- Smooth entropic mapping [Seguy et al., 2017, Pooladian and Niles-Weed, 2021].

- Linear Monge mapping when data supposed Gaussian [Flamary et al., 2019].

- Estimation for $W_2$ using input convex neural networks [Makkuva et al., 2020].
Mapping estimation

- Barycentric mapping using the OT matrix $T$ [Ferradans et al., 2014].

\[ \hat{m}_T(x^s_i) = \arg \min_x \sum_j T_{i,j} c(x, x^t_j) \]

- Smooth entropic mapping [Seguy et al., 2017, Pooladian and Niles-Weed, 2021].
- Linear Monge mapping when data supposed Gaussian [Flamary et al., 2019].
- Estimation for $W_2$ using input convex neural networks [Makkuva et al., 2020].
Mapping with optimal transport

Distributions

Source $\mu_s$

Target $\mu_t$

Classic OT (LP)

Reg. Entropic OT

Mapping estimation

- Barycentric mapping using the OT matrix $T$ [Ferradans et al., 2014].

$$\hat{m}_T(x^s_i) = \arg \min_{x} \sum_j T_{i,j} c(x, x^t_j)$$

- Smooth entropic mapping [Seguy et al., 2017, Pooladian and Niles-Weed, 2021].

- Linear Monge mapping when data supposed Gaussian [Flamary et al., 2019].

- Estimation for $W_2$ using input convex neural networks [Makkuva et al., 2020].
Mapping with optimal transport

Distributions

Classic OT (LP)

Reg. Entropic OT

Mapping estimation

- Barycentric mapping using the OT matrix $T$ [Ferradans et al., 2014].

$$\hat{m}_T(x^s_i) = \arg\min_{x} \sum_j T_{i,j} c(x, x^t_j)$$

- Smooth entropic mapping [Seguy et al., 2017, Pooladian and Niles-Weed, 2021].
- Linear Monge mapping when data supposed Gaussian [Flamary et al., 2019].
- Estimation for $W_2$ using input convex neural networks [Makkuva et al., 2020].
Mapping with optimal transport

- Barycentric mapping using the OT matrix $T$ [Ferradans et al., 2014].
- Smooth entropic mapping [Seguy et al., 2017, Pooladian and Niles-Weed, 2021].

$$\hat{m}(x) = \frac{\sum_j x_j^t v_j \exp(-\|x - x_j^t\|^2/\lambda)}{\sum_j v_j \exp(-\|x - x_j^t\|^2/\lambda)}, \quad \text{with } v \text{ sol. of Sinkhorn}$$

- Linear Monge mapping when data supposed Gaussian [Flamary et al., 2019].
- Estimation for $W_2$ using input convex neural networks [Makkuva et al., 2020].
Mapping estimation

- Barycentric mapping using the OT matrix $T$ [Ferradans et al., 2014].
- Smooth entropic mapping [Seguy et al., 2017, Pooladian and Niles-Weed, 2021].
- Linear Monge mapping when data supposed Gaussian [Flamary et al., 2019].

$$m(x) = m_2 + A(x - m_1) \quad \text{with} \quad A = \Sigma_1^{-1/2} (\Sigma_1^{1/2} \Sigma_2 \Sigma_1^{1/2})^{1/2} \Sigma_1^{-1/2}$$

- Estimation for $W_2$ using input convex neural networks [Makkuva et al., 2020].
Mapping with optimal transport

Mapping estimation

- Barycentric mapping using the OT matrix $T$ [Ferradans et al., 2014].
- Smooth entropic mapping [Seguy et al., 2017, Pooladian and Niles-Weed, 2021].
- Linear Monge mapping when data supposed Gaussian [Flamary et al., 2019].

$$m(x) = m_2 + A(x - m_1) \text{ with } A = \Sigma_1^{-1/2} (\Sigma_1^{1/2} \Sigma_2 \Sigma_1^{1/2})^{1/2} \Sigma_1^{-1/2}$$

- Estimation for $W_2$ using input convex neural networks [Makkuva et al., 2020].
Mapping estimation

- Barycentric mapping using the OT matrix $T$ [Ferradans et al., 2014].
- Smooth entropic mapping [Seguy et al., 2017, Pooladian and Niles-Weed, 2021].
- Linear Monge mapping when data supposed Gaussian [Flamary et al., 2019].
- Estimation for $W_2$ using input convex neural networks [Makkuva et al., 2020].
Histogram matching in images

Pixels as empirical distribution [Ferradans et al., 2014]
Histogram matching in images

Image colorization [Ferradans et al., 2014]

Original $X^0$

Original $Y^0$

Proposed method
OT mapping for Image-to-Image translation

Principle

- Encode image as a distribution in a DNN embedding.
- Transform between images using estimated Monge mapping.
- Linear Monge Mapping (Wasserstein Style Transfer [Mroueh, 2019]).
- Nonlinear Monge Mapping using input Convex Neural Networks [Korotin et al., 2019].
- Allows for transformation between two images but also style interpolation with Wasserstein barycenters.
Domain Adaptation

- Classification problem with data coming from different sources (domains).
- Distributions are different but related.
- Labels only available in the source domain, but prediction is conducted in the target domain.
- Objective: Train a classifier that performs well in the target domain.
Domain Adaptation problem

- Classification problem with data coming from different sources (domains).
- Distributions are different but related.
- Labels only available in the source domain, but prediction is conducted in the target domain.
- Objective: Train a classifier that performs well in the target domain.
Assumptions

1. There exist an OT mapping $m$ in the feature space between the two domains.

2. The transport preserves the joint distributions:

$$P^{s}(x, y) = P^{t}(m(x), y).$$

3-step strategy [Courty et al., 2014, Courty et al., 2016]

1. Estimate optimal transport between distributions (use regularization).

2. Transport the training samples on target domain.

3. Learn a classifier on the transported training samples.
Extensions and related works

- JDOT [Courty et al., 2017b] : Joint OT and target predictor estimation.
- [Shen et al., 2018] : Wasserstein Distance Guided Representation Learning.
- DeepJDOT [Damodaran et al., 2018, Fatras et al., 2021] : Deep learning JDOT.
- [Gnassounou et al., 2023]: Convolutional Monge Mapping for Multi-source DA.
Extensions and related works

- JDOT [Courty et al., 2017b]: Joint OT and target predictor estimation.
- [Shen et al., 2018]: Wasserstein Distance Guided Representation Learning.
- DeepJDOT [Damodaran et al., 2018, Fatras et al., 2021]: Deep learning JDOT.
- [Gnassounou et al., 2023]: Convolutional Monge Mapping for Multi-source DA.
Outline

Introduction
  OT problem and mathematical tools
  Optimal Transport and Machine Learning

Mapping with optimal transport
  Mapping with optimal transport from discrete samples
  Optimal transport for domain adaptation

Optimal Transport as a distance between distributions
  OT between histogram data
  OT between empirical distributions

Optimal Transport between spaces and structures
  Gromov-Wasserstein and extensions
  Applications of OT between graphs

Conclusion
Discrete distributions: Empirical vs Histogram

Discrete measure:  \( \mu = \sum_{i=1}^{n} a_i \delta_{x_i}, \ x_i \in \Omega, \ \sum_{i=1}^{n} a_i = 1 \)

Lagrangian (point clouds)

- Constant weight:  \( a_i = \frac{1}{n} \)
- Quotient space:  \( \Omega^n, \ \Sigma_n \)

Eulerian (histograms)

- Fixed positions  \( x_i \) e.g. grid
- Convex polytope  \( \Sigma_n \) (simplex):  \( \{(a_i)_i \geq 0; \sum_i a_i = 1\} \)
Unsupervised learning on histogram data

- DL with Wasserstein distance [Sandler and Lindenbaum, 2011, Rolet et al., 2016]
- Nonlinear DL with Wasserstein barycenter [Schmitz et al., 2017]
- Geodesic PCA in Wasserstein space [Seguy and Cuturi, 2015, Bigot et al., 2017].
- Approximation using Wasserstein embedding [Courty et al., 2017a].
## Dictionary Learning and Principal Geodesics Analysis

<table>
<thead>
<tr>
<th>Class 0</th>
<th>Class 1</th>
<th>Class 4</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PCA</strong></td>
<td><strong>PGA</strong></td>
<td><strong>PCA</strong></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

### Unsupervised learning on histogram data

- DL with Wasserstein distance [Sandler and Lindenbaum, 2011, Rolet et al., 2016]
- Nonlinear DL with Wasserstein barycenter [Schmitz et al., 2017]
- Geodesic PCA in Wasserstein space [Seguy and Cuturi, 2015, Bigot et al., 2017].
- Approximation using Wasserstein embedding [Courty et al., 2017a].
Multi-label learning with Wasserstein Loss

Learning with a Wasserstein Loss [Froger et al., 2015]

$$\min_f \sum_{k=1}^{N} W_1^1(f(x_i), l_i)$$

- Empirical loss minimization with Wasserstein loss.
- Multi-label prediction (labels $l$ seen as histograms, $f$ output softmax).
- Cost between labels can encode semantic similarity between classes.
- Good performances in image tagging.
Wasserstein Generative Adversarial Networks (WGAN)

Wasserstein GAN [Arjovsky et al., 2017]

\[
\min_G \ W_1^1(G\#\mu_z, \mu_d), \quad \text{s.t. } z \sim \mathcal{N}(0, I)
\]  

- Minimizes the distance between the true \( \mu_d \) and generated data \( G\#\mu_z \).
- Better convergence in practice than classical GANs [Goodfellow et al., 2014].
- Wasserstein in the dual (separable w.r.t. the samples).

\[
\min_G \sup_{\phi \in \text{Lip}^1} \mathbb{E}_{x \sim \mu_d} [\phi(x)] - \mathbb{E}_{z \sim \mu_z} [\phi(G(z))]
\]

- Lipschitzness constrained or penalized [Gulrajani et al., 2017].
- State of the art for image generation with [Karras et al., 2019] (before diffusion).
Wasserstein Generative Adversarial Networks (WGAN)

Wasserstein GAN [Arjovsky et al., 2017]

\[
\min_G W_1^1(G\#\mu_z, \mu_d), \quad \text{s.t. } z \sim \mathcal{N}(0, I)
\]

- Minimizes the distance between the true \(\mu_d\) and generated data \(G\#\mu_z\).
- Better convergence in practice than classical GANs [Goodfellow et al., 2014].
- Wasserstein in the dual (separable w.r.t. the samples).

\[
\min_G \sup_{\phi \in \text{Lip}^1} \mathbb{E}_{x \sim \mu_d}[\phi(x)] - \mathbb{E}_{z \sim \mu_z}[\phi(G(z))]
\]

- Lipschitzness constrained or penalized [Gulrajani et al., 2017].
- State of the art for image generation with [Karras et al., 2019] (before diffusion).
Outline

Introduction
  OT problem and mathematical tools
  Optimal Transport and Machine Learning

Mapping with optimal transport
  Mapping with optimal transport from discrete samples
  Optimal transport for domain adaptation

Optimal Transport as a distance between distributions
  OT between histogram data
  OT between empirical distributions

Optimal Transport between spaces and structures
  Gromov-Wasserstein and extensions
  Applications of OT between graphs

Conclusion
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 \( \mu_s = \sum_i a_i \delta_{x_i^s} \) and \( \mu_t = \sum_j b_j \delta_{x_j^t} \) and \( D_{i,k} = \|x_i^s - x_k^s\|, D'_{j,l} = \|x_j^t - x_l^t\| \)

- 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].

Inspired from Gabriel Peyré
FGW for discrete distributions [Vayer et al., 2018]

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

with \( \mu_s = \sum_i a_i \delta_{x_i^s} \) and \( \mu_t = \sum_j b_j \delta_{x_j^t} \) and \( D_{i,k} = \|x_i^s - x_k^s\|, D'_{j,l} = \|x_j^t - x_l^t\| \)

- 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

Graph as a distribution \((D, F, h)\)

- The positions \(x_i\) are implicit and represented as the pairwise matrix \(D\).
- Possible choices for \(D\) : Adjacency matrix, Laplacian, Shortest path, ...
- The node features can be compared between graphs and stored in \(F\).
- \(h_i\) 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]

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]

Noiseless graph

Barycenter

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]

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]

Noiseless graph

Noisy graphs samples

Barycenter

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

Barycenter

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

Representation learning for graphs

- Learn a dictionary \( \{ \overline{C}_i \}_i \) 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].
  \[
  \hat{C} = \sum_i w_i \overline{C}_i
  \]
- 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 \( \{ \overline{C}_i \} \) 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].

\[
\hat{C} = \arg \min_C \sum_i w_i GW(C, \overline{C}_i)
\]

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

Representation learning for graphs

- Learn a dictionary \( \{ \overline{C}_i \}_i \) 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].

\[
f(x) = \hat{C}(x) = \arg \min_C \sum_i w_i(x)GW(C, \overline{C}_i)
\]
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.
- State-of-the-art (still!) on graph classification (1×#1, 3×#2 on paperwithcode).
Outline

Introduction
- OT problem and mathematical tools
- Optimal Transport and Machine Learning

Mapping with optimal transport
- Mapping with optimal transport from discrete samples
- Optimal transport for domain adaptation

Optimal Transport as a distance between distributions
- OT between histogram data
- OT between empirical distributions

Optimal Transport between spaces and structures
- Gromov-Wasserstein and extensions
- Applications of OT between graphs

Conclusion
Ten years of least effort

Optimal Transport for Machine Learning

• Very dynamic community (NeurIPS OTML workshop every 2 years).
• Distributions are everywhere, and geometry can help.
• OT can be used to map, find correspondances and measure similarity.
• Many extensions: sliced, unbalanced, multi-marginal, ...

What about the next ten years ?

• OT is here to stay, it is a tool that can be adapted/relaxed.
• We need better solvers (faster, more scalable, more robust).
Collaborators

N. Courty  A. Rakotomamonjy  D. Tuia  A. Habrard  M. Perrot  M. Ducoffe

M. Cuturi  K. Lounici  A. Férari  C. Févotte  V. Emiya  V. Seguy

B. Damodaran  T. Vayer  L. Chapel  R. Tavenard  K. Fatras  I. Redko

H. Janati  T. Séjourné  H. Tran  G. Gasso  M. Corneli  C. Vincent-Cuaz

+ H. Van Assel, Th. Gnassounou, A. Gramfort
Thank you

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)


**Wasserstein generative adversarial networks.**


**Geodesic pca in the wasserstein space by convex pca.**


**Learning to predict graphs with fused gromov-wasserstein barycenters.**

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


**Learning wasserstein embeddings.**

Joint distribution optimal transportation for domain adaptation.
In *Neural Information Processing Systems (NIPS)*.


Domain adaptation with regularized optimal transport.
In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*.


Optimal transport for domain adaptation.
*Pattern Analysis and Machine Intelligence, IEEE Transactions on*.


Sinkhorn distances: Lightspeed computation of optimal transport.
In *NIPS*, pages 2292–2300.

  Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation.


  Unbalanced minibatch optimal transport; applications to domain adaptation.

  In International Conference on Machine Learning (ICML).


  Regularized discrete optimal transport.

  SIAM Journal on Imaging Sciences, 7(3).


  Concentration bounds for linear monge mapping estimation and optimal transport domain adaptation.


**Learning with a wasserstein loss.**


**Convolutional monge mapping normalization for learning on biosignals.**
In *Neural Information Processing Systems*.


**Generative adversarial nets.**


**Improved training of wasserstein gans.**

On the translocation of masses.


A style-based generator architecture for generative adversarial networks.

In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410.


Wasserstein-2 generative networks.


Gromov wasserstein distances and the metric approach to object matching.


Mémoire sur la théorie des déblais et des remblais.
De l’Imprimerie Royale.


Wasserstein barycenter for multi-source domain adaptation.


Wasserstein style transfer.

Gromov-wasserstein averaging of kernel and distance matrices.
In ICML, pages 2664–2672.


Entropic estimation of optimal transport maps.


Fast dictionary learning with a smoothed wasserstein loss.
In Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, pages 630–638.


The earth mover’s distance as a metric for image retrieval.

Nonnegative matrix factorization with earth mover’s distance metric for image analysis.


Large-scale optimal transport and mapping estimation.


Principal geodesic analysis for probability measures under the optimal transport metric.


Template based graph neural network with optimal transport distances.


Online graph dictionary learning.
In International Conference on Machine Learning (ICML).


Gromov-wasserstein factorization models for graph clustering.