Optimal Transport for Structured data

Applications on graphs

Titouan Vayer
Joint work with Laetitia Chapel, Remi Flamary, Romain Tavenard and Nicolas Courty
March 8, 2019

Table of content

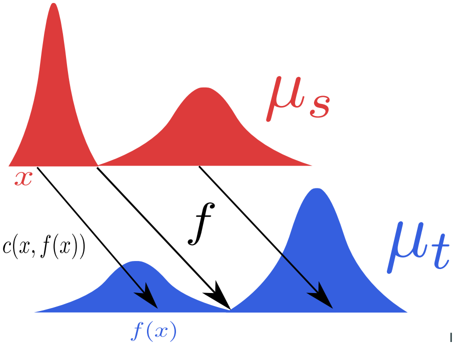
Introduction

Probability measures μ_s and μ_t on and a cost function $c : \Omega_s \times \Omega_t \to \mathbb{R}^+$.

Monge formulation

The Monge formulation [Monge, 1781] aim at finding a mapping $f : \Omega_s \to \Omega_t$ which transports the measure μ_s into μ_t with the less effort.

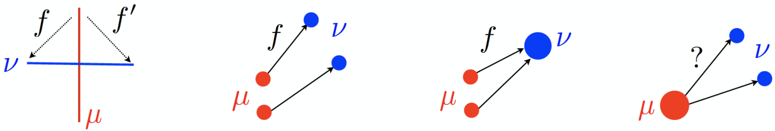
$$\inf_{T\#\mu_s=\mu_t} \int_{\Omega_s} c(\mathbf{x}, f(\mathbf{x}))\mu_s(\mathbf{x})d\mathbf{x} \tag{1}$$



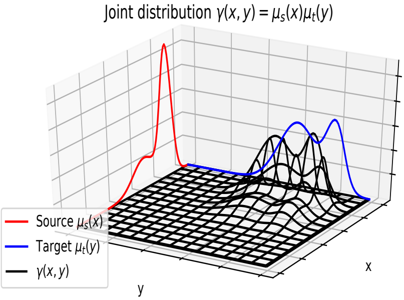
Inspired from Gabriel Peyré

[Brenier, 1991] proved existence and unicity of the Monge map for $c(x, y) = \|x - y\|^2$ and distributions with densities.

However with non regular distributions :



Optimal transport (Kantorovich formulation)



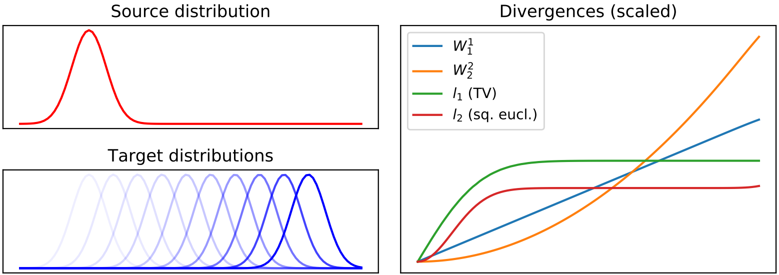
- The Kantorovich formulation [Kantorovich, 1942] seeks for a probabilistic coupling $\boldsymbol{\pi} \in \mathcal{P}(\Omega_s \times \Omega_t)$ between Ω_s and Ω_t:

$$\boldsymbol{\pi}_0 = \underset{\boldsymbol{\pi}}{\text{argmin}} \int_{\Omega_s \times \Omega_t} c(\mathbf{x}, \mathbf{y}) \boldsymbol{\pi}(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y}, \tag{2}$$

$$\text{s.t.} \quad \boldsymbol{\pi} \in \Pi = \left\{ \boldsymbol{\pi} \geq 0, \int_{\Omega_t} \boldsymbol{\pi}(\mathbf{x}, \mathbf{y}) d\mathbf{y} = \mu_s, \int_{\Omega_s} \boldsymbol{\pi}(\mathbf{x}, \mathbf{y}) d\mathbf{x} = \mu_t \right\}$$

- $\boldsymbol{\pi}$ is a joint probability measure with marginals μ_s and μ_t.
- Linear Program that always have a solution.

Wasserstein distance



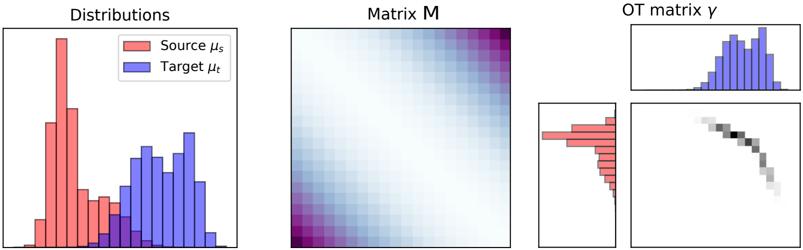
Wasserstein distance

$$W_p^p(\mu_s, \mu_t) = \min_{\boldsymbol{\pi} \in \Pi} \int_{\Omega_s \times \Omega_t} c(\mathbf{x}, \mathbf{y}) \boldsymbol{\pi}(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y} = E_{(\mathbf{x}, \mathbf{y}) \sim \boldsymbol{\pi}}[c(\mathbf{x}, \mathbf{y})] \quad (3)$$

where $c(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|^p$ is the ground metric.

- A.K.A. Earth Mover's Distance (W_1^1) [Rubner et al., 2000].
- Do not need the distribution to have overlapping support.
- Works for continuous and discrete distributions (histograms, empirical).

Optimal transport with discrete distributions



$\mu_s = \sum_{i=1}^{n_s} a_i \delta_{x_i^s}$ and $\mu_t = \sum_{j=1}^{n_t} b_j \delta_{x_j^t}$

OT Linear Program

$$\boldsymbol{\pi}_0 = \underset{\boldsymbol{\pi} \in \Pi}{\operatorname{argmin}} \left\{ \langle \boldsymbol{\pi}, M \rangle_F = \sum_{i,j} \pi_{i,j} M_{i,j} \right\}$$

where M is a cost matrix with $M_{i,j} = c(x_i^s, x_j^t)$ and the marginals constraints are

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

Solved with Network Flow solver of complexity $O(n^3 \log(n))$.

$$\pi_0^\lambda = \underset{\pi \in \Pi}{\text{argmin}} \quad \langle \pi, M \rangle_F + \lambda \Omega(\pi), \qquad (4)$$

Regularization term $\Omega(\pi)$

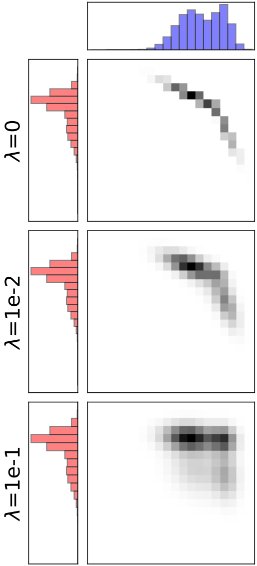
- Entropic regularization [Cuturi, 2013].

$$\Omega(\pi) = \sum_{i,j} \pi(i,j)(\log \pi(i,j) - 1)$$

- Group Lasso [Courty et al., 2016a], KL, Itakura Saito, β-divergences, [Dessein et al., 2016].

Why regularize?

- Smooth the "distance" estimation:
  $$W_\lambda(\mu_s, \mu_t) = \langle \pi_0^\lambda, M \rangle_F$$
- Encode prior knowledge on the data.
- Better posed problem (convex, stability).
- Fast algorithms to solve the OT problem.



Entropy-regularized transport

The solution of entropy regularized optimal transport problem is of the form

$$\boldsymbol{\pi}_0^\lambda = \text{diag}(\mathbf{u}) \exp(-M/\lambda) \text{diag}(\mathbf{v})$$

Why ? Consider the Lagrangian of the optimization problem:

$$\mathcal{L}(\boldsymbol{\pi}, \alpha, \beta) = \sum_{ij} \boldsymbol{\pi}_{ij} M_{ij} + \lambda \boldsymbol{\pi}_{ij} (\log \boldsymbol{\pi}_{ij} - 1) + \alpha^{\mathbf{T}} (\boldsymbol{\pi} \mathbf{1}_{n_t} - a) + \beta^{\mathbf{T}} (\boldsymbol{\pi}^T \mathbf{1}_{n_s} - b)$$

$$\begin{aligned} \partial \mathcal{L}(\boldsymbol{\pi}, \alpha, \beta) / \partial \boldsymbol{\pi}_{ij} \quad &= \quad M_{ij} + \lambda \log \boldsymbol{\pi}_{ij} + \alpha_i + \beta_j \\ \partial \mathcal{L}(\boldsymbol{\pi}, \alpha, \beta) / \partial \boldsymbol{\pi}_{ij} = 0 \quad &\Longrightarrow \quad \boldsymbol{\pi}_{ij} = \exp(\frac{\alpha_i}{\lambda}) \exp(-\frac{M_{ij}}{\lambda}) \exp(\frac{\beta_j}{\lambda}) \end{aligned}$$

- Through the **Sinkhorn theorem** $\text{diag}(\mathbf{u})$ and $\text{diag}(\mathbf{v})$ exist and are unique.
- Can be solved by the **Sinkhorn-Knopp** algorithm (implementation in parallel, GPU).

Sinkhorn-Knopp algorithm

The Sinkhorn-Knopp algorithm performs alternatively a scaling along the rows and columns of $\mathbf{K} = \exp(-\frac{M}{\lambda})$ to match the desired marginals.

Algorithm 1 Sinkhorn-Knopp Algorithm (SK).

Require: $\mathbf{a}, \mathbf{b}, M, \lambda$

$\mathbf{u}^{(0)} = \mathbf{1}, \mathbf{K} = \exp(-M/\lambda)$

for i in $1, \dots, n_{it}$ **do**

$\mathbf{v}^{(i)} = \mathbf{b} \oslash \mathbf{K}^\top \mathbf{u}^{(i-1)}$ // Update right scaling

$\mathbf{u}^{(i)} = \mathbf{a} \oslash \mathbf{K}\mathbf{v}^{(i)}$ // Update left scaling

end for

return $\mathcal{T} = \text{diag}(\mathbf{u}^{(n_{it})})\mathbf{K}\text{diag}(\mathbf{v}^{(n_{it})})$

- Complexity $O(kn^2)$, where k iterations are required to reach convergence
- Fast implementation in parallel, GPU friendly
- Allows automatic-differentiation for any loss *w.r.t* $\pi, \mathbf{a}, \mathbf{b}, M$

Benamou *et al.* [Benamou et al., 2015] showed that solving for the reg OT problem is actually a Bregman projection

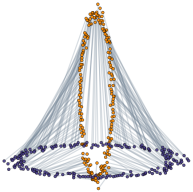
OT as a Bregman projection

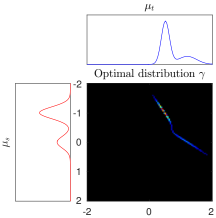
$\boldsymbol{\pi}^\star$ is the solution of the following Bregman projection

$$\boldsymbol{\pi}^\star = \operatorname*{argmin}_{\boldsymbol{\pi}\in\Pi} \mathrm{KL}(\boldsymbol{\pi}, \zeta), \tag{5}$$

where $\zeta = \exp(-\frac{M}{\lambda})$.

Sinkhorn in this case is an iterative projection scheme, with alternative projections on marginal constraints.





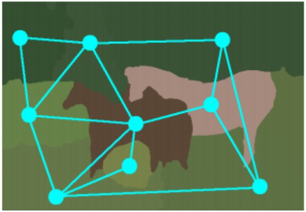
Transporting with optimal transport

- Color adaptation in image [Ferradans et al., 2014a].
- Domain adaptation [Courty et al., 2016b].
- OT mapping estimation [Perrot et al., 2016].

Divergence between distributions

- Use the ground metric to encode complex relations between the bins.
- Loss for multilabel classifier [Frogner et al., 2015]
- Loss for spectral unmixing [Flamary et al., 2016b].
- Non parametric divergence between non overlapping distributions.
- Objective function for GAN [Arjovsky et al., 2017].
- Estimate discriminant subspace [Flamary et al., 2016a].

Optimal Transport on structured data



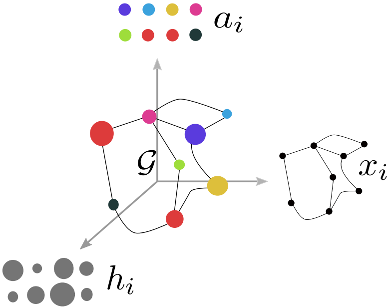
[Harchaoui and Bach, 2012]

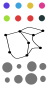
Structured data

- A structure data is viewed as a combination of features informations linked within each other by some structural information.
- Example : labeled graph.

Meaningful distances on structured data

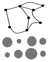
- Us both features (labels) and structure (graph).
- Allows for comparison, classification.
- Data science (statistics, means)





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

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



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

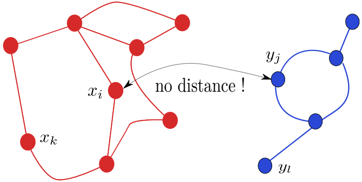
Graph data representation

$$\mu = \sum_{i=1}^{n} h_i \delta_{(x_i, a_i)}$$

- Nodes are weighted by their mass h_i.
- for two $\mu_s = \sum_{i=1}^{n} h_i \delta_{x_i, a_i}$ and $\mu_t = \sum_{j=1}^{m} g_j \delta_{y_j, b_j}$
  - Features values a_i and b_j can be compared through the common metric
  - But no common between the structure points x_i and y_j.

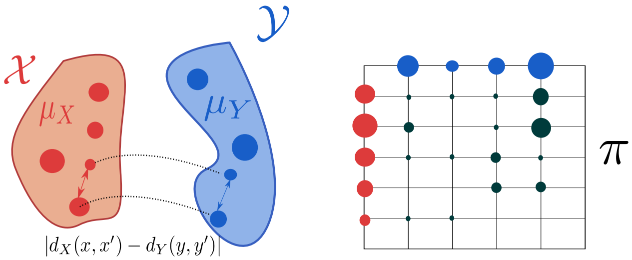
Structured data as distributions

Wasserstein distance deals with distribution but can not leverage the specific relation among the component of the distribution.



- How to include this structural information in the optimal transportation formulation ?
- How to use the new formulation in order to compare structured data (graphs, times series...)

Almost saved: Gromov-Wasserstein distance



Inspired from Gabriel Peyré

GW distance [Mémoli, 2011]

$\mathcal{X} = (X, d_X, \mu_X)$ and $\mathcal{Y} = (Y, d_Y, \mu_Y)$, two mesurable metric spaces.

$$\mathcal{GW}_p(\mu_X, \mu_Y) = \Big(\inf_{\pi \in \Pi(\mu_X, \mu_Y)} \int_{X \times Y \times X \times Y} |d_X(x,x') - d_Y(y,y')|^p d\pi(x,y) d\pi(x',y') \Big)^{\frac{1}{p}}$$

- Distance over measures with no common ground space.
- Compare the intrinsic distances in each space.
- Invariant to rotations and translation in either spaces.

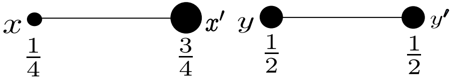
Mathematical properties

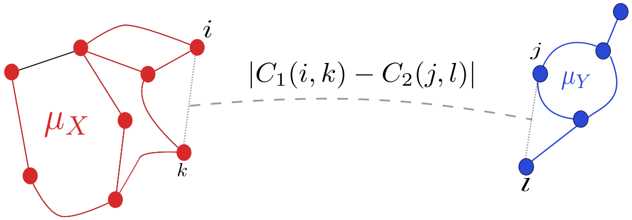
$\mathcal{GW}$ is a distance over the space of all mesurable metric spaces quotient by the measure preserving isometries (called *isomorphisms*) :

- $\mathcal{GW}$ is symmetric and satisfies the triangle inequality.
- $\mathcal{GW}_p(\mu_X, \mu_Y) = 0$ *iff* there exists a Monge Map $f : \mathcal{X} \to \mathcal{Y}$ such that :
  - $f\#\mu_X = \mu_Y$ (measure preserving).
  - $\forall x, x' \in X^2 \quad d_X(x, x') = d_Y(f(x), f(x'))$ (isometry between $\mathcal{X}$ and $\mathcal{Y}$).

Figure 1: Two isometric objects

Figure 2: Two isometric but not isomorphic objects





GW in discrete case

$$\mathcal{GW}_p(C_1, C_2, \mu_X, \mu_Y) = \left(\min_{\pi \in \Pi(\mu_X, \mu_Y)} \sum_{i,j,k,l} |C_1(i,k) - C_2(j,l)|^p \pi_{i,j}\, \pi_{k,l} \right)^{\frac{1}{p}}$$

$\mu_X = \sum_i h_i \delta_{x_i}$ and $\mu_Y = \sum_j g_j \delta_{y_j}$ and $C_1(i,k) = d_X(x_i, x_k), C_2(j,l) = d_Y(y_j, y_l)$

- This is related to a Quadratic Assignment Problem (QAP), opposed to the linear assignment problem as with the classical OT problem.
- Soft QAP : non-convex problem, often NP-hard
- Similarity measure between pair to pair distances : $L(C^1_{i,k}, C^2_{j,l}) = |C_1(i,k) - C_2(j,l)|^p$

Computing GW coupling (I) : entropic reguarization

Peyré and colleagues consider the entropic regularization of this problem [Peyré et al., 2016] :

$$\mathcal{GW}_p(C_1, C_2, \mu_X, \mu_Y) = \underset{\boldsymbol{\pi}\in\Pi}{\operatorname{argmin}} \left(\sum_{i,j,k,l} L(C^1_{i,k}, C^2_{j,l}) \boldsymbol{\pi}_{i,j} \boldsymbol{\pi}_{k,l} - \lambda H(\boldsymbol{\pi}) \right)$$

One can easily compute **GW** by using projected gradient descent where each iteration can be solved using a Sinkhorn algorithm !

Algorithm 2 Sinkhorn-Knopp Algorithm for GW

Require: g, h, C_1, C_2, λ
  $\boldsymbol{\pi}_0 = gh^T$
  for k in $1, \ldots, n_{it}$ **do**
    $\mathbf{u}^{(0)} = \mathbf{1}, \mathbf{K} = \exp(-\mathcal{L}(C_1, C_2) \otimes \boldsymbol{\pi}_{k-1}/\lambda)$
    for i in $1, \ldots, n'_{it}$ **do**
      $\mathbf{v}^{(i)} = h \oslash \mathbf{K}^\top \mathbf{u}^{(i-1)}$ // Update right scaling
      $\mathbf{u}^{(i)} = g \oslash \mathbf{K}\mathbf{v}^{(i)}$ // Update left scaling
    end for
  end for
  return $\mathcal{T} = \text{diag}(\mathbf{u}^{(n_{it})})\mathbf{K}\text{diag}(\mathbf{v}^{(n_{it})})$


SOON

Applications in ML

- Metric alignment and shape matching [Solomon et al., 2016]
- Barycenter of domains with different dimension [Peyré et al.,]
- Heterogeneous domain adaptation [Yan et al., 2018]
- Unsupervised word embeddings alignment [Alvarez-Melis and Jaakkola, 2018]
- CNN on 3D point clouds [Ezuz et al., 2017]

Illustration of applications of GW [Solomon et al., 2016]

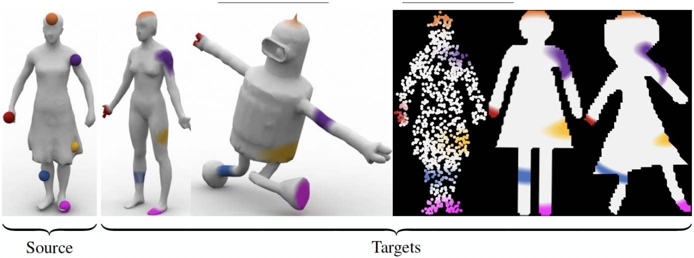
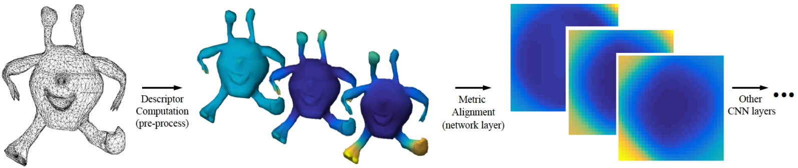


Figure 3: Shape matching between 3D and 2D objects

How to handle unstructured geometric data such as 3D mesh ?

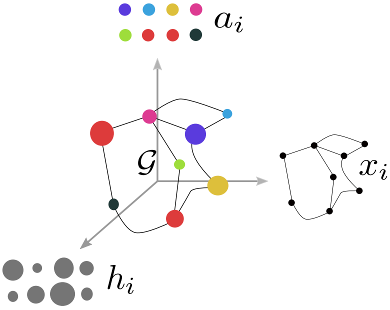
- Converting point clouds, meshes, or polygon soups into regular representations (multi-view images, volumetric grids or planar parameterizations..)
- Leads to fixed pre-process disconnected from the machine learning tool

Idea : use GW to optimize the geometric representation during the network learning process



Fused Gromov-Wasserstein distance

Get back to the roots



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

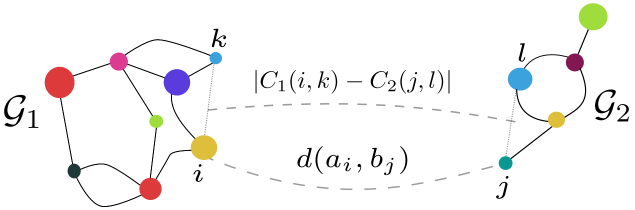
$$\mu_A = \sum_i h_i \delta_{a_i}$$

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

Graph data representation

$$\mu = \sum_{i=1}^{n} h_i \delta_{(x_i, a_i)}$$

- Nodes are weighted by their mass h_i.
- Features values a_i and b_j can be compared through the common metric
- But no common between the structure points x_i and y_j.



Fused Gromov Wasserstein distance

Parameters $q \geq 1$, $p \geq 1$.

$$\mathcal{FGW}_{p,q,\alpha}(C_1, C_2, \mu_s, \mu_t) = \left(\min_{\pi \in \Pi(\mu_s, \mu_t)} \sum_{i,j,k,l} \left((1-\alpha) M_{i,j}^q + \alpha |C_1(i,k) - C_2(j,l)|^q\right)^p \pi_{i,j}\, \pi_{k,l} \right)^{\frac{1}{p}}$$

$\mu_s = \sum_{i=1}^n h_i \delta_{x_i, a_i}$ and $\mu_t = \sum_{j=1}^m g_j \delta_{y_j, b_j}$

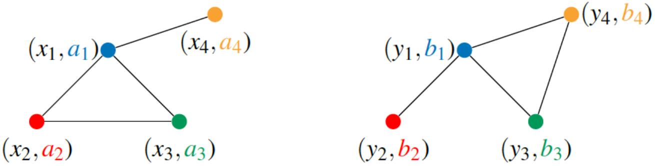
- $M_{i,j} = d(a_i, b_j)$ is the distance betweens the features
- $C_1(i,k) = d_X(x_i, x_k), C_2(j,l) = d_Y(y_j, y_l)$ distances in the manifolds of the structures (e.g shortest path)
- $\alpha \in [0,1]$ is a trade off parameter between structure and features.

FGW Properties (1)

$$\mathcal{FGW}_{p,q,\alpha}(C_1, C_2, \mu_s, \mu_t) = \left(\min_{\pi \in \Pi(\mu_s, \mu_t)} \sum_{i,j,k,l} \left((1-\alpha) M_{i,j}^q + \alpha |C_1(i,k) - C_2(j,l)|^q\right)^p \pi_{i,j}\, \pi_{k,l} \right)^{\frac{1}{p}}$$

Metric properties

- $\mathcal{FGW}$ defines a metric over structured data with **measure and features preserving isometries** as invariants.
- $\mathcal{FGW}$ is a metric for $q = 1$ a semi metric for $q > 1$, $\forall p \geq 1$.
- The distance is nul *iff* :
  - There exists a Monge map $T\#\mu_s = \mu_t$.
  - Structures are equivalent through this Monge map (isometry).
  - Features are equal through this Monge map.



FGW Properties (2)

Other properties for sontinuous distributions

- Interpolation between $\mathcal{W}$ ($\alpha = 0$) and $\mathcal{GW}$ ($\alpha = 1$) distances.
- Geodesic properties (constant speed, unicity).

Bounds and convergence to finite samples

- The following inequalities hold:

$$\mathcal{FGW}(\mu_s, \mu_t) \geq (1-\alpha)\mathcal{W}(\mu_A, \mu_B)^q$$

$$\mathcal{FGW}(\mu_s, \mu_t) \geq \alpha\mathcal{GW}(\mu_X, \mu_Y)^q$$

- Bound when $\mathcal{X} = \mathcal{Y}$:

$$\mathcal{FGW}(\mu_s, \mu_t)^p \leq 2\mathcal{W}(\mu_s, \mu_t)^p$$

- Convergence of finite samples when $\mathcal{X} = \mathcal{Y}$ with $d = Dim(\mathcal{X}) + Dim(\Omega)$:

$$\mathbb{E}[\mathcal{FGW}(\mu, \mu_n)] = O\left(n^{-\frac{1}{d}}\right)$$

$$\pi^* = \underset{\pi \in \Pi(\mu_s, \mu_t)}{\arg\min} \quad \text{vec}(\pi)^T Q \text{vec}(\pi) + \text{vec}((1-\alpha)M)^T \text{vec}(\pi) \tag{6}$$

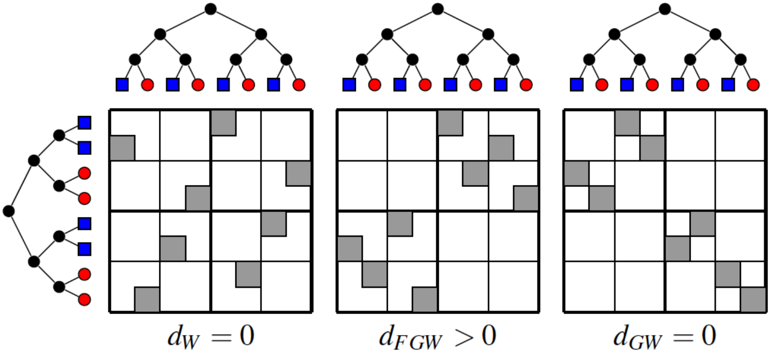
where $Q = -2\alpha C_2 \otimes C_1$

Algorithmic resolution ($p = 1$)

- Non convex QP : we use CG [Ferradans et al., 2014b] with OT solver.
- Convergence to a local minima [Lacoste-Julien, 2016].
- With entropic regularization, projected gradient descent [Peyré et al., 2016].

Algorithm 3 Conditional Gradient (CG) for FGW

$\pi^{(0)} \leftarrow \mu_X \mu_Y^\top$
for $i = 1, \ldots,$ **do**
  $G \leftarrow$ Gradient from Eq. (6) *w.r.t.* $\pi^{(i-1)}$
  $\tilde{\pi}^{(i)} \leftarrow$ Solve OT with ground loss G
  $\tau^{(i)} \leftarrow$ Line-search for loss with $\tau \in (0, 1)$
  $\pi^{(i)} \leftarrow (1 - \tau^{(i)})\pi^{(i-1)} + \tau^{(i)}\tilde{\pi}^{(i)}$
end for



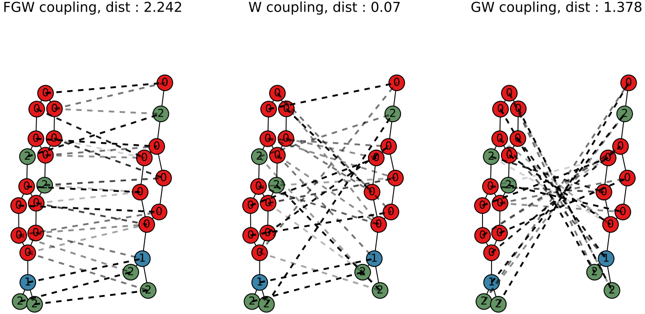
FGW maps on toy tree

- Uniform weights on the leafs of the tree.
- Structure distance taken as shortest path on the tree.
- Only FGW can encode both features and structures.

Graph classification

- We want to classify of a dataset of labeled graphs : $(\mathcal{G}_i, y_i)_i$
- Discrete labels : e.g atoms, continuous labels : e.g $\mathbb{R}^d$ vectors
- We use shortest path for C_1, C_2 to encode the structure
- We use ℓ_2 for continuous attributes and distance based on Weisfeler-Lehman labeling for discrete attributes.

MUTAG dataset : couplings between graphs from two different classes



Vector attributes	BZR	COX2	CUNEIFORM	ENZYMES	PROTEIN	SYNTHETIC
FGW sp	**85.12±4.15**	77.23±4.86	**76.67±7.04**	71.00±6.76	**74.55±2.74**	**100.00±0.00**
HOPPERK	84.15±5.26	**79.57±3.46**	32.59±8.73	45.33±4.00	71.96±3.22	90.67±4.67
PROPAK	79.51±5.02	77.66±3.95	12.59±6.67	**71.67±5.63**	61.34±4.38	64.67±6.70
PSCN k=10	80.00±4.47	71.70±3.57	25.19±7.73	26.67±4.77	67.95±11.28	**100.00±0.00**
PSCN k=5	82.20±4.23	71.91±3.40	24.81±7.23	27.33±4.16	71.79±3.39	**100.00±0.00**

Graph classification

- Classifiation accuracy on classical graph datasets.
- Comparison with state-of-the-art graph kernel approaches and Graph CNN.
- We use $\exp(-\gamma \mathcal{FGW})$ as a non-positive kernel for an SVM [Loosli et al., 2016] (FGW).

Discrete attr.	MUTAG	NCI1	PTC
FGW raw sp	83.26±10.30	72.82±1.46	55.71±6.74
FGW wl h=2 sp	86.42±7.81	85.82±1.16	63.20±7.68
FGW wl h=4 sp	**88.42±5.67**	**86.42±1.63**	**65.31±7.90**
GK k=3	82.42±8.40	60.78±2.48	56.46±8.03
RWK	79.47±8.17	58.63±2.44	55.09±7.34
SPK	82.95±8.19	74.26±1.53	60.05±7.39
WLK	86.21±8.48	85.77±1.07	62.86±7.23
WLK h=2	86.21±8.15	81.85±2.28	61.60±8.14
WLK h=4	83.68±9.13	85.13±1.61	62.17±7.80
PSCN k=10	83.47±10.26	70.65±2.58	58.34±7.71
PSCN k=5	83.05±10.80	69.85±1.79	55.37±8.28

Without attribute	IMDB-B	IMDB-M
GW sp	**63.80±3.49**	**48.00±3.22**
GK k=3	56.00±3.61	41.13±4.68
SPK	55.80±2.93	38.93±5.12

Graph classification

- Classifiation accuracy on classical graph datasets.
- Comparison with state-of-the-art graph kernel approaches and Graph CNN.
- We use $\exp(-\gamma \mathcal{FGW})$ as a non-positive kernel for an SVM [Loosli et al., 2016] (FGW).

FGW barycenter

Euclidean vs FGW barycenter

- Euclidean barycenter :

$$\min_{\hat{x} \in \mathbb{R}^d} \sum_i \lambda_i \|\hat{x} - x_i\|^2$$

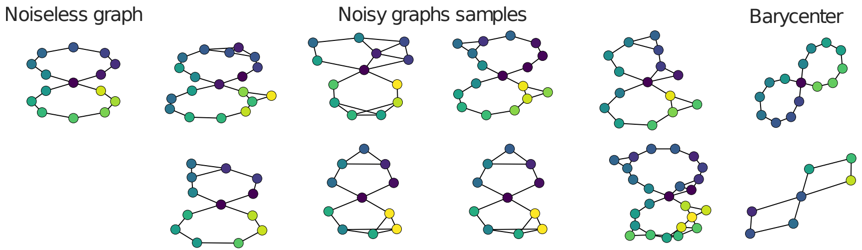
- FGW barycenter (Fréchet means) :

$$\min_{\hat{\mu}} \sum_i \lambda_i \mathcal{FGW}(\hat{\mu}, \mu_i)$$

Equivalent to find the structure and the feature minimizing the Fréchet means

FGW barycenter $p = 1, q = 2$

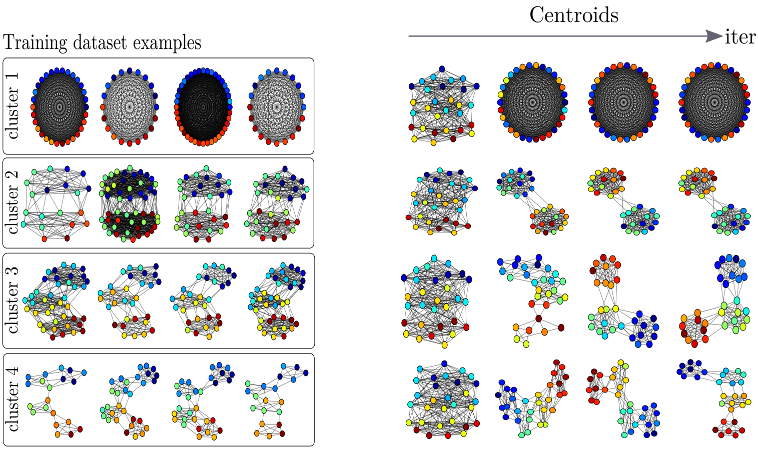
- Barycenter optimization solved via block coordinate descent (on $\boldsymbol{\pi}, \hat{C}, \{\hat{a_i}\}_i$).
- Can chose to fix the structure ($\hat{C}$) or the features $\{\hat{a_i}\}_i$ in the barycenter.
- $\{\hat{a_i}\}_i$, and $\hat{C}$ updates are weighted averages using $\boldsymbol{\pi}$.

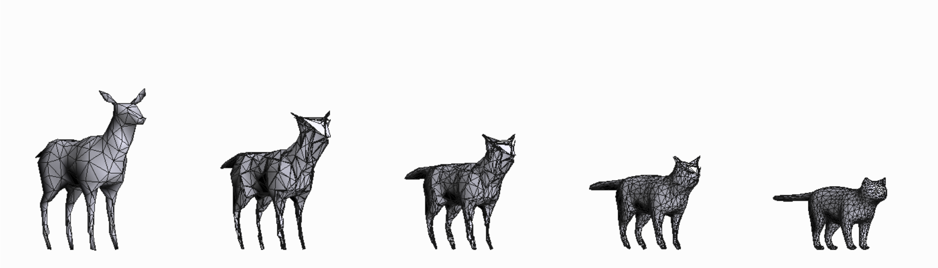


Barycenter of noisy graphs

- We select a clean graph, change the number of nodes and add label noise and random connections.
- We compute the barycenter on $n = 15$ and $n = 7$ nodes.
- Barycenter graph is obtained through thresholding of the $\hat{C}$ matrix.

- Clustering of multiple real-valued graphs. Dataset composed of 40 graphs (10 graphs $\times$ 4 types of communities)
- k-means clustering using the FGW barycenter

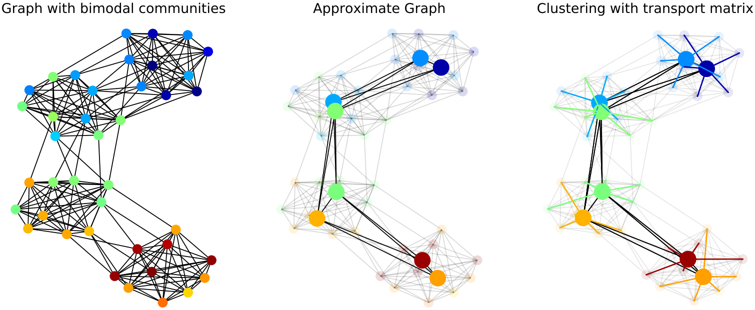




Mesh interpolation

- Two meshes (deer and cat).
- Fix structure from cat, estimate barycenter for the positions of the edges.
- Wasserstien ($\alpha = 0$) do not respect the graph (mesh neighborhood).
- FGW conserve the graph, regularized FGW smoothes the surface.

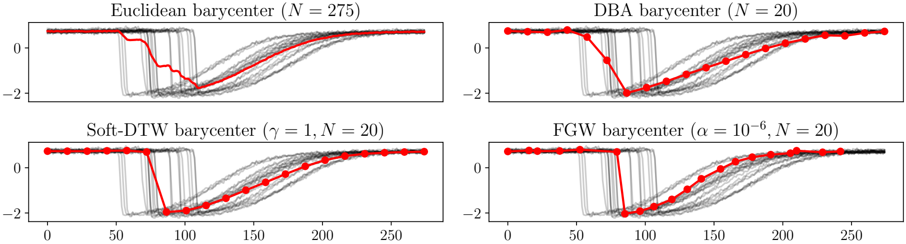
FGW for community clustering



Graph approximation and comunity clustering

$$\min_{C,\mu} \quad \mathcal{FGW}(C, C_0, \mu, \mu_0)$$

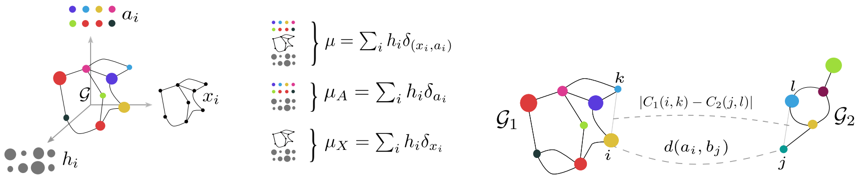
- Approximate the graph (C_0, μ_0) with a small number of nodes.
- OT matrix give the clustering affectation.
- Works for signle and multiple modes in the clusters.



Time series averaging

- Comparsion with Euclidean, DBA [Petitjean et al., 2011] and Soft-DTW [Cuturi and Blondel, 2017].
- Structure is time position of samples, fetaure value of the signal.
- Temporal position of nodes recovered with a MDS of C.
- Barycenter have non-regular sampling.

Conclusion for FGW



Fused Gromov-Wasserstein distance [Vayer et al., 2018],[Vayer et al., 2018]

- Model structured data as distributions.
- New versatile and differentiable method for comparing structured data
- Many desirable distance properties
- New notion of barycenter of structured data such as graphs or time series
- No need for embeddings and same sized graphs
- Interpretable distance via optimal map

What next ?

- Devise efficient optimization shemes for large structures.
- Add interpretability to deep neural networks on graph.

Alvarez-Melis, D. and Jaakkola, T. S. (2018).
Gromov-Wasserstein Alignment of Word Embedding Spaces.
arXiv:1809.00013 [cs].
arXiv: 1809.00013.

Arjovsky, M., Chintala, S., and Bottou, L. (2017).
Wasserstein gan.
arXiv preprint arXiv:1701.07875.

Benamou, J.-D., Carlier, G., Cuturi, M., Nenna, L., and Peyré, G. (2015).
Iterative Bregman projections for regularized transportation problems.
SISC.

Brenier, Y. (1991).
Polar factorization and monotone rearrangement of vector-valued functions.
Communications on pure and applied mathematics, 44(4):375–417.

Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. (2016a).
Optimal transport for domain adaptation.
IEEE Transactions on Pattern Analysis and Machine Intelligence.

Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. (2016b).
Optimal transport for domain adaptation.
Pattern Analysis and Machine Intelligence, IEEE Transactions on.

Cuturi, M. (2013).
Sinkhorn distances: Lightspeed computation of optimal transportation.
In *Neural Information Processing Systems (NIPS)*, pages 2292–2300.

Cuturi, M. and Blondel, M. (2017).
Soft-DTW: a differentiable loss function for time-series.
volume 70, pages 894–903, International Convention Centre, Sydney, Australia. PMLR.

Dessein, A., Papadakis, N., and Rouas, J.-L. (2016).
Regularized optimal transport and the rot mover's distance.
arXiv preprint arXiv:1610.06447.

Ezuz, D., Solomon, J., Kim, V. G., and Ben-Chen, M. (2017).
GWCNN: A Metric Alignment Layer for Deep Shape Analysis.
Computer Graphics Forum, 36(5):49–57.

Ferradans, S., Papadakis, N., Peyré, G., and Aujol, J.-F. (2014a).
Regularized discrete optimal transport.
SIAM Journal on Imaging Sciences, 7(3).

Ferradans, S., Papadakis, N., Peyré, G., and Aujol, J.-F. (2014b).
Regularized discrete optimal transport.
SIAM Journal on Imaging Sciences, 7(3):1853–1882.

Flamary, R., Cuturi, M., Courty, N., and Rakotomamonjy, A. (2016a).
Wasserstein discriminant analysis.
arXiv preprint arXiv:1608.08063.

Flamary, R., Fevotte, C., Courty, N., and Emyia, V. (2016b).
Optimal spectral transportation with application to music transcription.
In *Neural Information Processing Systems (NIPS)*.

Frogner, C., Zhang, C., Mobahi, H., Araya, M., and Poggio, T. A. (2015).
Learning with a wasserstein loss.
In *Advances in Neural Information Processing Systems*, pages 2053–2061.

Harchaoui, Z. and Bach, F. (2012).
Tree-walk kernels for computer vision.
Theory and Practice, page 32.

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

Lacoste-Julien, S. (2016).
Convergence rate of frank-wolfe for non-convex objectives.
arXiv preprint arXiv:1607.00345.

Loosli, G., Canu, S., and Ong, C. S. (2016).
Learning svm in krein spaces.
IEEE transactions on pattern analysis and machine intelligence, 38(6):1204–1216.

Mémoli, F. (2011).
Gromov-Wasserstein distances and the metric approach to object matching.
Foundations of Computational Mathematics, pages 1–71.

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

Perrot, M., Courty, N., Flamary, R., and Habrard, A. (2016).
Mapping estimation for discrete optimal transport.
In *Neural Information Processing Systems (NIPS)*.

Petitjean, F., Ketterlin, A., and Gançarski, P. (2011).
A global averaging method for dynamic time warping, with applications to clustering.
44(3):678–693.

Peyré, G., Cuturi, M., and Solomon, J. (2016).
Gromov-Wasserstein Averaging of Kernel and Distance Matrices.
In *ICML 2016*, Proc. 33rd International Conference on Machine Learning, New-York, United States.

Peyré, G., Cuturi, M., and Solomon, J.
Gromov-Wasserstein Averaging of Kernel and Distance Matrices.
page 10.

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, J., Peyré, G., Kim, V. G., and Sra, S. (2016).
Entropic metric alignment for correspondence problems.
ACM Transactions on Graphics, 35(4):1–13.

Vayer, T., Chapel, L., Flamary, R., Tavenard, R., and Courty, N. (2018).
Fused gromov-wasserstein distance for structured objects: theoretical foundations and mathematical properties.

Vayer, T., Chapel, L., Flamary, R., Tavenard, R., and Courty, N. (2018).
Fused Gromov-Wasserstein distance for structured objects: theoretical foundations and mathematical properties.
arXiv e-prints, page arXiv:1811.02834.

Yan, Y., Li, W., Wu, H., Min, H., Tan, M., and Wu, Q. (2018).
Semi-Supervised Optimal Transport for Heterogeneous Domain Adaptation.
In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pages 2969–2975, Stockholm, Sweden. International Joint Conferences on Artificial Intelligence Organization.