Optimal transport for graph data

Barycenters and dictionary learning

R. Flamary - CMAP, École Polytechnique, Institut Polytechnique de Paris

January 26 2022

Statistics & Computer Science Day for Data Science, Paris-Saclay

Collaborators


N. Courty


A. Rakotomamonjy


D. Tuia


A. Habrard


M. Perrot


M. Ducoffe


M. Cuturi


K. Lounici


A. Férrari


C. Févotte


V. Emiya


V. Seguy


B. Damodaran


T. Vayer


L. Chapel


R. Tavenard


K. Fatras


I. Redko


H. Janati


T. Séjourné


H. Tran

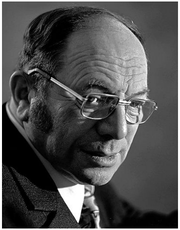
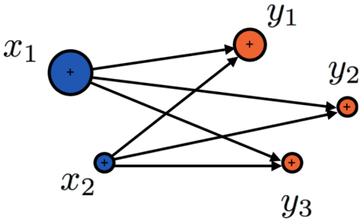

G. Gasso


M. Corneli

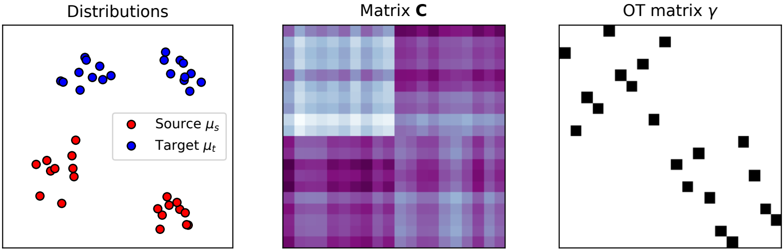

C. Vincent-Cuaz

Table of content

Optimal Transport and divergences between graphs

- 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



Kantorovitch formulation : OT Linear Program

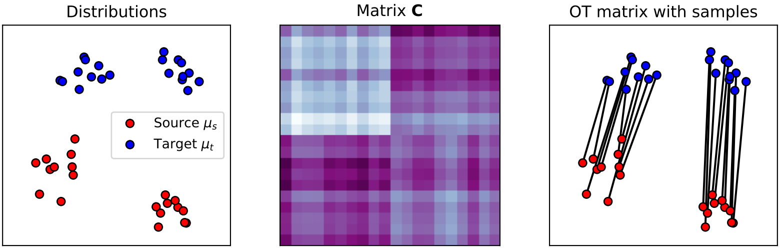
When $\mu_s = \sum_{i=1}^{n_s} a_i \delta_{\mathbf{x}_i^s}$ and $\mu_t = \sum_{i=1}^{n_t} b_i \delta_{\mathbf{x}_i^t}$

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

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

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

- $W_p(\mu_s, \mu_t)$ is called the Wasserstein distance (EMD for $p = 1$).
- Entropic regularization solved efficiently with Sinkhorn [Cuturi, 2013].
- Classical OT needs distributions lying in the same space → Gromov-Wasserstein.



Kantorovitch formulation : OT Linear Program

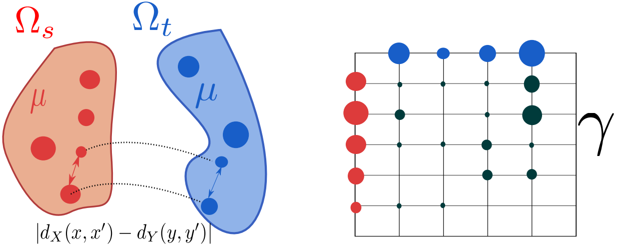
When $\mu_s = \sum_{i=1}^{n_s} a_i \delta_{\mathbf{x}_i^s}$ and $\mu_t = \sum_{i=1}^{n_t} b_i \delta_{\mathbf{x}_i^t}$

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

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

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

- $W_p(\mu_s, \mu_t)$ is called the Wasserstein distance (EMD for $p = 1$).
- Entropic regularization solved efficiently with Sinkhorn [Cuturi, 2013].
- Classical OT needs distributions lying in the same space $\rightarrow$ Gromov-Wasserstein.



Inspired from Gabriel Peyré

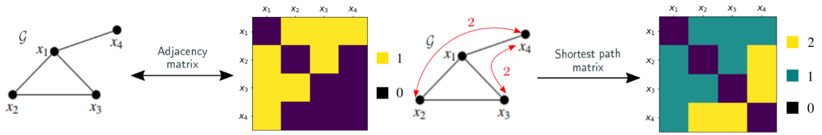
GW for discrete distributions [Memoli, 2011]

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

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.
- Invariant to isometry in either spaces (e.g. rotations and translation).
- Entropy regularize GW proposed in [Peyré et al., 2016].

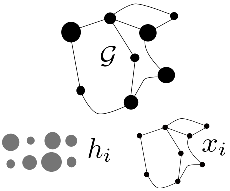
Gromov-Wasserstein between graphs



Modeling the graph structure with a pairwise matrix $\boldsymbol{D}$

- An undirected graph $\mathcal{G} := (\mathbf{V}, \mathbf{E})$ is defined by $\mathbf{V} = \{\mathbf{x_i}\}_{\mathbf{i} \in [\mathbf{N}]}$ set of the $\mathbf{N}$ nodes and $\mathbf{E} = \{(\mathbf{x_i}, \mathbf{x_j}) | \mathbf{x_i} \leftrightarrow \mathbf{x_j}\}$ set of edges.
- Structure represented as a symmetric matrix $\boldsymbol{D}$ of relations between the nodes.
- Possible choices : **Adjacency matrix** (used in this study), Laplacian matrix, Shortest path or geodesic distance matrix.

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

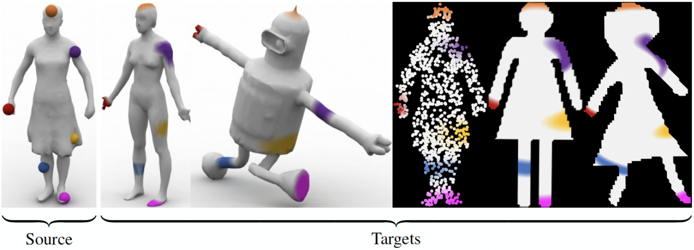


- Graph represented as a discrete distribution:

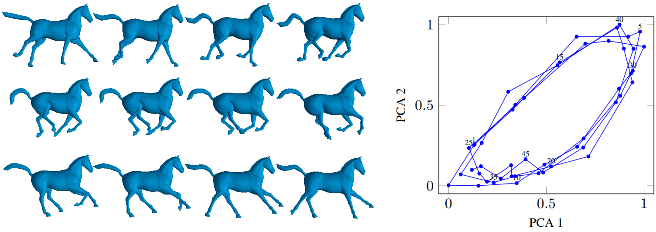
$$\mu_X = \sum_i h_i \delta_{x_i}$$

- The positions x_i are implicit and represented as the pairwise matrix $\boldsymbol{D}$.
- h_i are the masses on the nodes of the graphs (uniform by default).

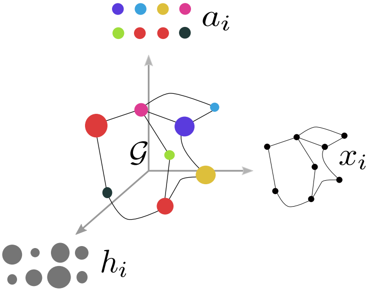
Shape matching between 3D and 2D surfaces

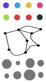


Multidimensional scaling (MDS) of shape collection



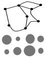
Attributed graphs as distributions





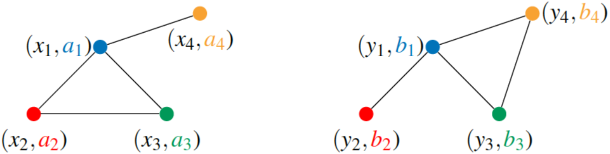
$$\mu = \sum_i h_i \delta_{(x_i, a_i)}$$

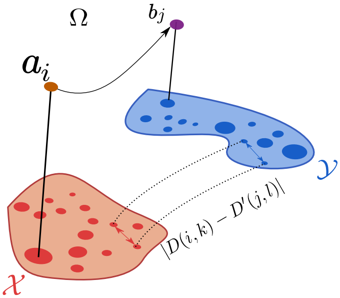
$$\mu_A = \sum_i h_i \delta_{a_i}$$



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

- Joint distribution μ in the feature/structure space.
  - Nodes are weighted by their mass h_i.
  - Structure encoded by x_i (no common metric between two different graphs).
  - Features values a_i can be compared through the common metric.
- Importance of the joint modeling:





Fused Gromov Wasserstein distance [Vayer et al., 2020]
$\mu_s = \sum_{i=1}^{n} h_i \delta_{x_i, a_i}$ and $\mu_t = \sum_{j=1}^{m} g_j \delta_{y_j, b_j}$

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

with $D_{i,k} = \|x_i - x_k\|$ and $D'_{j,l} = \|y_i - y_l\|$ and $C_{i,j} = \|a_i - b_j\|$

- Parameters $q > 1$, $\forall p \geq 1$.
- $\alpha \in [0, 1]$ is a trade off parameter between structure and features.

FGW Properties

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

Metric properties [Vayer et al., 2020]

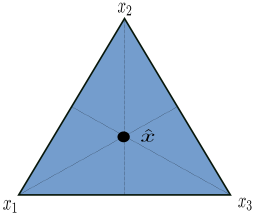
- $\mathcal{FGW}$ defines a metric over structured data with **measure and features preserving isometries** as invariants.
- $\mathcal{FGW}$ is a metric for $q = 1$ a semi metric for $q > 1$, $\forall p \geq 1$.
- The distance is nul *iff* :
  - There exists a Monge map $T\#\mu_s = \mu_t$.
  - Structures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

Bounds and convergence to finite samples [Vayer et al., 2020]

- $\mathcal{FGW}(\mu_s, \mu_t)$ is lower bounded by $(1-\alpha)\mathcal{W}(\mu_A, \mu_B)^q$ and $\alpha\mathcal{GW}(\mu_X, \mu_Y)^q$
- Convergence of finite samples when $\mathcal{X} = \mathcal{Y}$ with $d = Dim(\mathcal{X}) + Dim(\Omega)$:

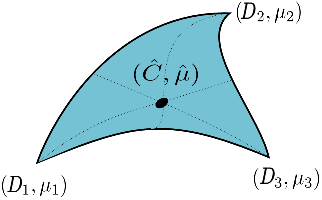
$$\mathbb{E}[\mathcal{FGW}(\mu, \mu_n)] = O\left(n^{-\frac{1}{d}}\right)$$

Euclidean barycenter



$$\min_x \sum_k \lambda_k \|x - x_k\|^2$$

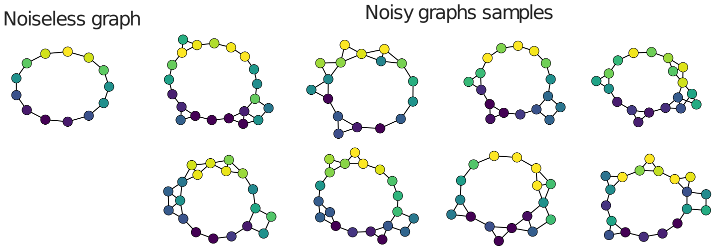
FGW barycenter



$$\min_{D \in \mathbb{R}^{n \times n}, \mu} \sum_i \lambda_i \mathcal{FGW}(D_i, D, \mu_i, \mu)$$

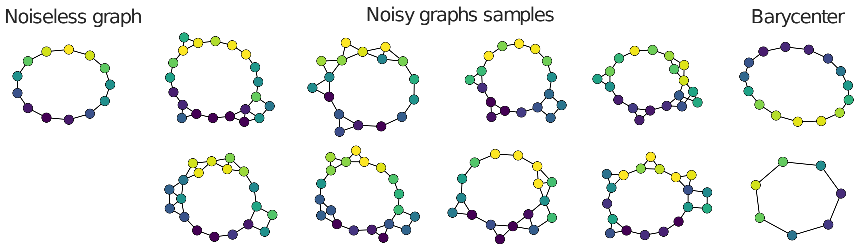
FGW barycenter $p = 1, q = 2$

- Estimate FGW barycenter using Frechet means (similar to [Peyré et al., 2016]).
- Barycenter optimization solved via block coordinate descent (on $\mathbf{T}, D, \{a_i\}_i$).
- Can chose to fix the structure (D) or the features $\{a_i\}_i$ in the barycenter.
- a_{ii}, and D updates are weighted averages using $\mathbf{T}$.



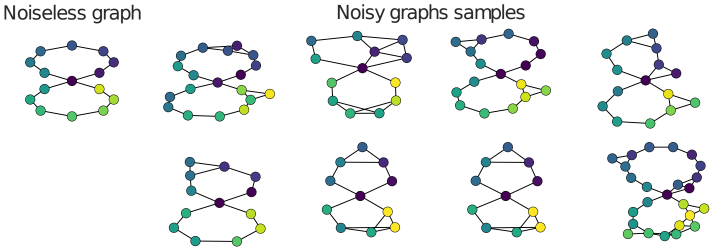
Barycenter of noisy graphs

- 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 D matrix.



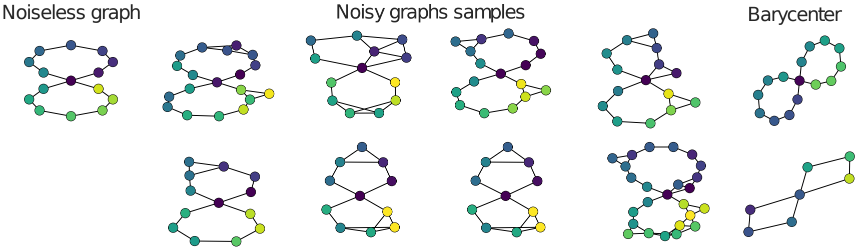
Barycenter of noisy graphs

- 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 D matrix.



Barycenter of noisy graphs

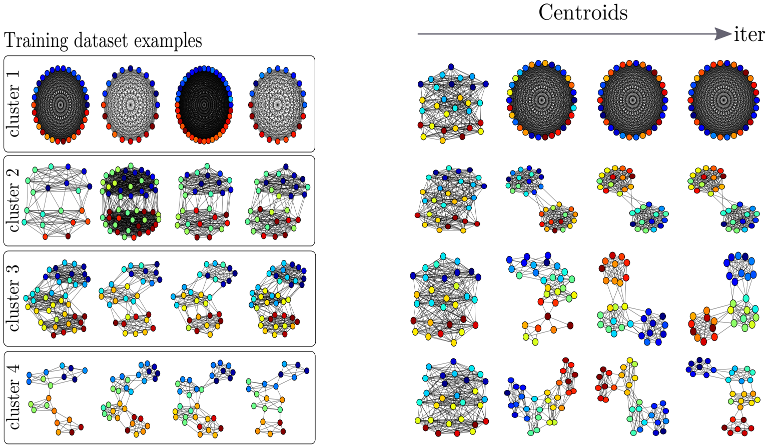
- We select a clean graph, change the number of nodes and add label noise and random connections.
- We compute the barycenter on $n = 15$ and $n = 7$ nodes.
- Barycenter graph is obtained through thresholding of the D matrix.



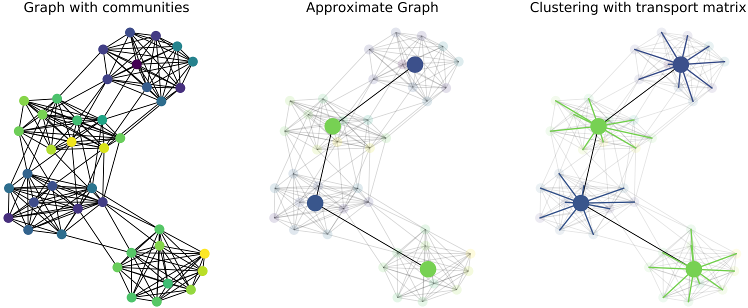
Barycenter of noisy graphs

- 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 D matrix.

FGW for graphs based clustering



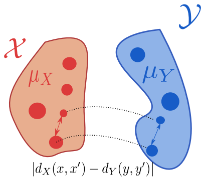
- Clustering of multiple real-valued graphs. Dataset composed of 40 graphs (10 graphs $\times$ 4 types of communities)
- k-means clustering using the FGW barycenter

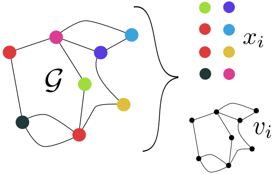


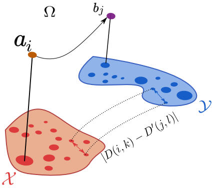
Graph approximation and community clustering

$$\min_{\mathbf{D},\mu} \quad \mathcal{FGW}(\mathbf{D}, \mathbf{D}_0, \mu, \mu_0)$$

- Approximate the graph $(\mathbf{D}_0, \mu_0)$ with a small number of nodes.
- Can be seen as a FGW (compressed) barycenter for one graph.
- OT matrix give the clustering affectation.
- Works for signle and multiple modes in the clusters.







Gromov-Wasserstein distance [Memoli, 2011]

- Divergence between distributions across metric spaces.
- Can be used to measure similarity between graphs seen as distribution their pairwise node relationship.

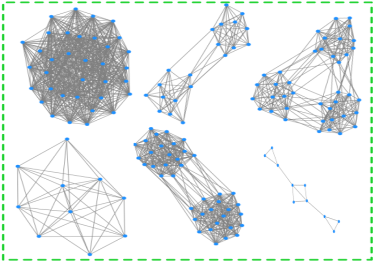
Fused Gromov-Wasserstein distance [Vayer et al., 2018]

- Model labeled structured data as joint structure/labels distributions.
- New versatile method for comparing structured data based on Optimal Transport
- New notion of barycenter of structured data such as graphs or time series

How to use GW/FGW to model data variability in a dataset of graphs?

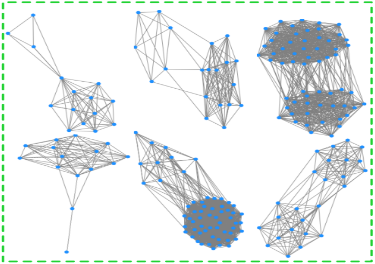
Online Graph Dictionary Learning

Dataset 1



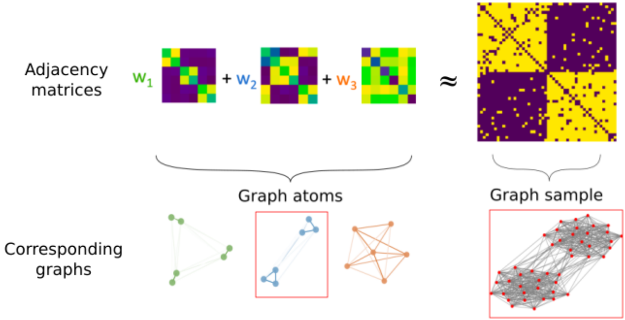
SBM with balanced communities $\{1, 2, 3\}$.

Dataset 2



Two communities of variable proportions.

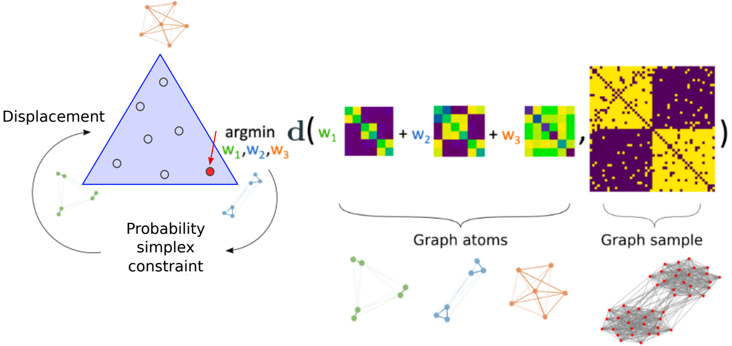
- We have access to **large datasets of graphs** with variable number of nodes.
- How to model the variability of those graphs?
- A natural formulation is to use **factorization**.
- We propose to use a **linear** model for representing te graph associated to and estimation of the linear basis : **Dictionary learning**.



Linear modeling of graphs

$$\boldsymbol{D} \approx \sum_{s \in [S]} w_s \overline{\boldsymbol{D}_s} \tag{1}$$

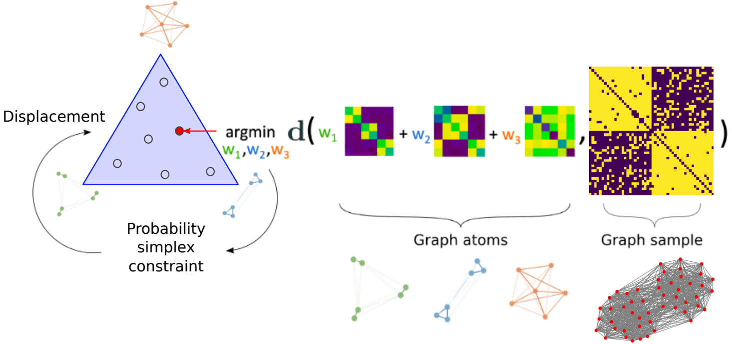
- Approximate a given graph structure $\boldsymbol{D}$ as a non-negative weighted sum of template graphs $\overline{\boldsymbol{D}_s}$.
- $\mathbf{w} \in \Sigma_S$ are the weights in the simplex.
- $\{\overline{\boldsymbol{D}_s}\}_s$ is the dictionary of templates that all have the same order (nb. of nodes).



Sparse linear unmixing with Gromov-Wasserstein

$$\min_{\mathbf{w} \in \Sigma_S} \quad \mathcal{GW}_2^2\left(\sum_{s \in [S]} w_s \overline{\boldsymbol{D}_s} \ , \ \boldsymbol{D} \right) \tag{2}$$

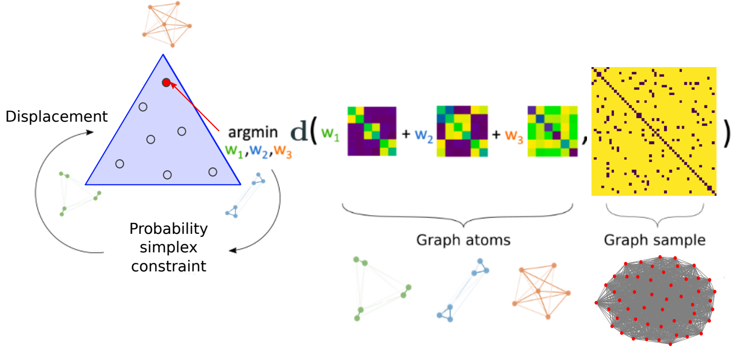
- Estimate the linear representation on the simplex $\mathbf{w}$ minimizing the GW distance *w.r.t.* the target graph $\boldsymbol{D}$ (non-negative unmixing).
- $\mathbf{w}$ is a vector embedding of the graph $\boldsymbol{D}$ in the dictionary.
- GW between graphs



Sparse linear unmixing with Gromov-Wasserstein

$$\min_{\mathbf{w}\in\Sigma_S} \quad \mathcal{GW}_2^2\left(\sum_{s\in[S]} w_s\overline{\boldsymbol{D}_s}\ ,\ \boldsymbol{D}\right) \tag{2}$$

- Estimate the linear representation on the simplex $\mathbf{w}$ minimizing the GW distance *w.r.t.* the target graph $\boldsymbol{D}$ (non-negative unmixing).
- $\mathbf{w}$ is a vector embedding of the graph $\boldsymbol{D}$ in the dictionary.
- GW between graphs



Sparse linear unmixing with Gromov-Wasserstein

$$\min_{\mathbf{w}\in\Sigma_S} \quad \mathcal{GW}_2^2\left(\sum_{s\in[S]} w_s \overline{\boldsymbol{D}_s}\ ,\ \boldsymbol{D}\right) \tag{2}$$

- Estimate the linear representation on the simplex $\mathbf{w}$ minimizing the GW distance *w.r.t.* the target graph $\boldsymbol{D}$ (non-negative unmixing).
- $\mathbf{w}$ is a vector embedding of the graph $\boldsymbol{D}$ in the dictionary.
- GW between graphs

GDL optimization problem

$$\min_{\substack{\{\mathbf{w}^{(k)}\}_{k\in[K]} \\ \{\overline{\boldsymbol{D}}_s\}_{s\in[S]}}} \sum_{k=1}^{K} \mathcal{GW}_2^2\left(\boldsymbol{D}^{(k)}, \sum_{s\in[S]} w_s^{(k)} \overline{\boldsymbol{D}}_s\right) - \lambda \|\mathbf{w}^{(k)}\|_2^2 \tag{3}$$

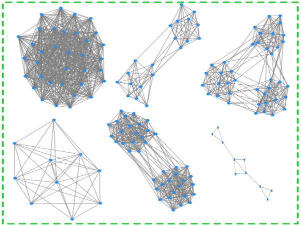
- On a dataset of K undirected graphs $\{\boldsymbol{D}^{(k)} \in S_{N^{(k)}}(\mathbb{R})\}_{k\in[K]}$.
- We want to estimate simultaneously the unmixing $\mathbf{w}^{(k)}$ of each graphs and the optimal dictionary $\{\overline{\boldsymbol{D}}_s\}_{s\in[S]}$.
- Very similar to classical DL (Non-negative Matrix Factorization) approach but with GW as a data fitting term.
- We propose to solve it an adaptation of the online algorithm [Mairal et al., 2009]

Stochastic/Online update [Vincent-Cuaz et al., 2021]

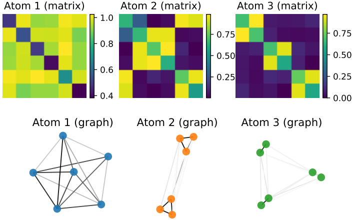
1: Sample a minibatch of graphs $\mathcal{B} := \{\boldsymbol{D}^{(k)}\}_{k\in\mathcal{B}}$.
2: Compute $\{(\mathbf{w}^{(k)}, \boldsymbol{T}^{(k)})\}_{k\in[B]}$ from solving B independent unmixings.
3: Compute the gradient $\widetilde{\nabla}_{\overline{\boldsymbol{D}}_s}$ on the minibatch with fixed $\{(\mathbf{w}^{(k)}, \boldsymbol{T}^{(k)})\}_{k\in[B]}$.
4: Projected gradient step , $\forall s \in [S], \overline{\boldsymbol{D}}_s \leftarrow Proj_{S_N(\mathbb{R})}(\overline{\boldsymbol{D}}_s - \eta_C \widetilde{\nabla}_{\overline{\boldsymbol{D}}_s})$

- Stochastic block model with $\{1, 2, 3\}$ blocks

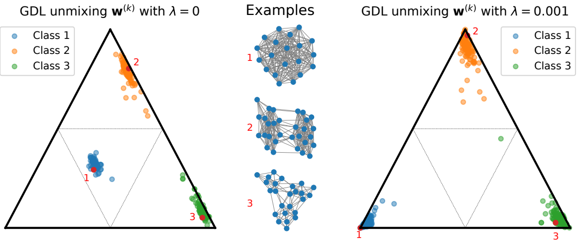
Dataset

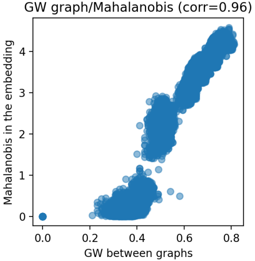


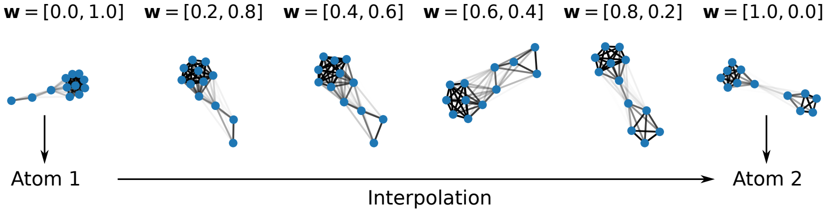
Learned atoms



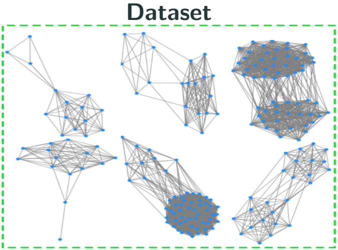
Embedding space





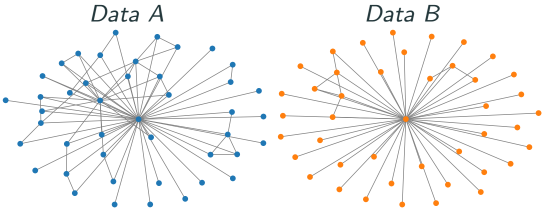


Learned Dictionary: Interpolation $\sim$ 1D Manifold

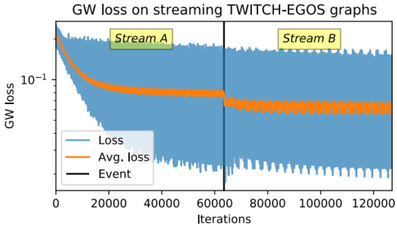


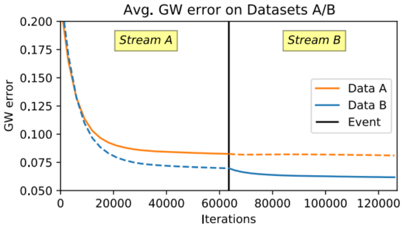
- Stochastic block model with 2 blocks and varying proportions of block size.
- GDL with 2 atoms can recover the extreme points.
- Linear interpolation recover a continuous variation of proportion.

- **Streaming graphs**: Stochastic update for each new incoming graph
- Dataset: **TWITCH-EGOS**
  - 120.000+ graphs
  - 2 classes
  - shared hub structure



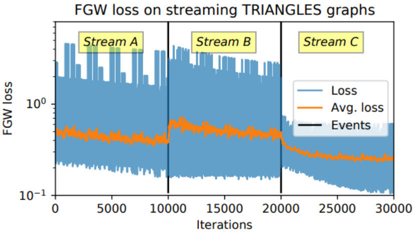
- **Simulated stream**: data A (class 1) → data B (class 2)

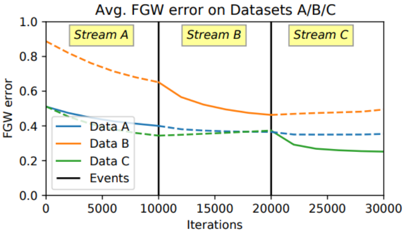


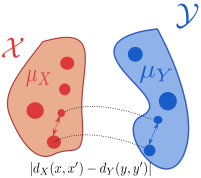


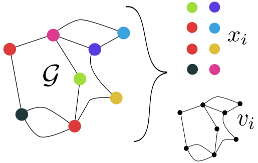
Experiments - Online Learning

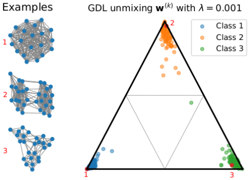
- **Streaming graphs**: Stochastic update for each new incoming graph
- Dataset : **TRIANGLES**
  - $30.000+$ labeled graphs
  - 10 classes
- **Simulated stream**: data A (4 classes) $\rightarrow$ data B (3 classes) $\rightarrow$ data C (3 classes)











Gromov-Wasserstein family for graph modeling

- Graphs modelled as distributions, $\mathcal{GW}$ can measure their similarity.
- Extensions of GW for labeled graphs and Frechet means can be computed.
- Nonlinear and linear dictionaries of graphs using $\mathcal{GW}$ provide a good modeling.
- Weights on the nodes are important but rarely available : relax the constraints [Séjourné et al., 2020] or even remove one of them [Vincent-Cuaz et al., 2022].

Open questions and new research

- Stability of the $\mathcal{GW}$ plan to perturbations of $\boldsymbol{D}$ (related to the GDL upper bound).
- Use $\mathcal{GW}$ as a "kernel" for structured prediction (conditional $\mathcal{GW}$ barycenters).

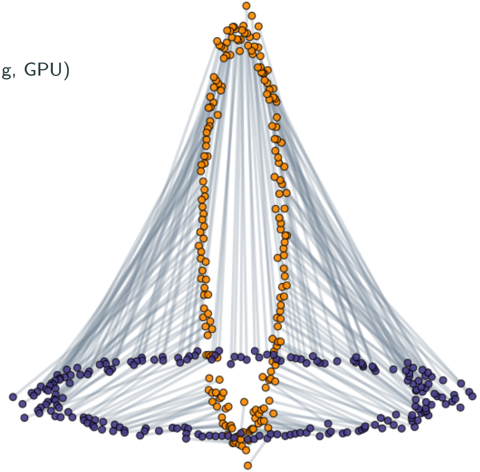
Thank you

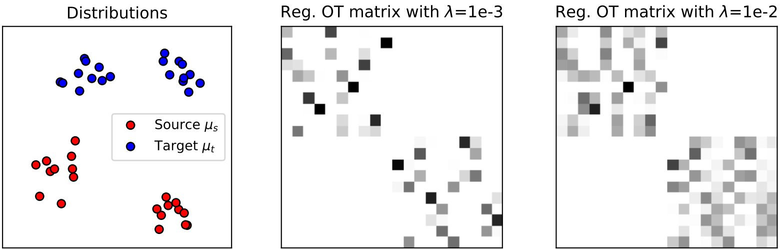
Python code available on GitHub:
`https://github.com/PythonOT/POT`

- 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/`

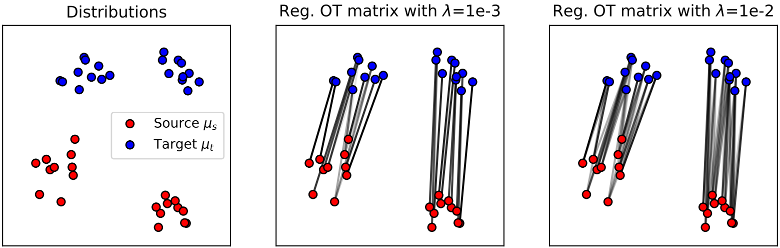




Entropic regularization [Cuturi, 2013]

$$W_\epsilon(\mu_s, \mu_t) = \min_{\mathbf{T}\in\Pi(\mu_s,\mu_t)} \quad \langle \mathbf{T}, \mathbf{C}\rangle_F + \epsilon \sum_{i,j} T_{i,j} \log T_{i,j}$$

- Regularization with the negative entropy $-H(\mathbf{T})$.
- Looses sparsity, but strictly convex optimization problem [Benamou et al., 2015].
- Can be solved with the very efficient Sinkhorn-Knopp matrix scaling algorithm.
- Loss and OT matrix are differentiable and have better statistical properties [Genevay et al., 2018].



Entropic regularization [Cuturi, 2013]

$$W_\epsilon(\mu_s, \mu_t) = \min_{\mathbf{T}\in\Pi(\mu_s,\mu_t)} \quad \langle \mathbf{T}, \mathbf{C}\rangle_F + \epsilon \sum_{i,j} T_{i,j} \log T_{i,j}$$

- Regularization with the negative entropy $-H(\mathbf{T})$.
- Looses sparsity, but strictly convex optimization problem [Benamou et al., 2015].
- Can be solved with the very efficient Sinkhorn-Knopp matrix scaling algorithm.
- Loss and OT matrix are differentiable and have better statistical properties [Genevay et al., 2018].

GW Upper bond [Vincent-Cuaz et al., 2021]
Let two graphs of order N in the linear embedding $\left(\sum_s w_s^{(1)}\overline{\boldsymbol{D}_s}\right)$ and $\left(\sum_s w_s^{(2)}\overline{\boldsymbol{D}_s}\right)$, the $\mathcal{GW}$ divergence can be upper bounded by

$$\mathcal{GW}_2\left(\sum_{s\in[S]} w_s^{(1)}\overline{\boldsymbol{D}_s}, \sum_{s\in[S]} w_s^{(2)}\overline{\boldsymbol{D}_s}\right) \leq \|\mathbf{w}^{(1)} - \mathbf{w}^{(2)}\|_{\boldsymbol{M}} \tag{4}$$

with $\boldsymbol{M}$ a PSD matrix of components $M_{p,q} = \left\langle \boldsymbol{D_h}\overline{\boldsymbol{D}_p}, \overline{\boldsymbol{D}_q}\boldsymbol{D_h}\right\rangle_F$, $\boldsymbol{D_h} = diag(\boldsymbol{h})$.

Discussion

- The upper bound is the value of GW for a transport $T = diag(\boldsymbol{h})$ assuming that the nodes are already aligned.
- The bound is exact when the weights $\mathbf{w}^{(1)}$ and $\mathbf{w}^{(2)}$ are close.
- Solving $\mathcal{GW}$ with FW si $O(N^3\log(N))$ at each iterations.
- Computing the Mahalanobis upper bound is $O(S^2)$: very fast alterative to GW for nearest neighbors retrieval.

Optimization problem

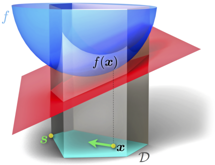
$$\mathcal{GW}_p^p(\mu_s,\mu_t)=\min_{\mathbf{T}\in\Pi(\mu_s,\mu_t)}\sum_{i,j,k,l}|D_{i,k}-D'_{j,l}|^pT_{i,j}\,T_{k,l}$$

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

- Quadratic Program (Wasserstein is a linear program).
- Nonconvex, NP-hard, related to Quadratic Assignment Problem (QAP).
- Large problem and non convexity forbid standard QP solvers.

Optimization algorithms

- Local solution with conditional gradient algorithm (Frank-Wolfe) [Frank and Wolfe, 1956].
- Each FW iteration requires solving an OT problems.
- Gromov in 1D has a close form (solved in discrete with a sort) [Vayer et al., 2019].
- With entropic regularization, one can use mirror descent [Peyré et al., 2016] or fast low rank approximations [Scetbon et al., 2021].



Optimization Problem

$$\mathcal{GW}_{p,\epsilon}^p(\mu_s, \mu_t) = \min_{\mathbf{T}\in\Pi(\mu_s,\mu_t)} \sum_{i,j,k,l} |D_{i,k} - D'_{j,l}|^p T_{i,j}\, T_{k,l} + \epsilon \sum_{i,j} T_{i,j} \log T_{i,j} \qquad (5)$$

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

- Smoothing the original GW with a convex and smooth entropic term.

Solving the entropic $\mathcal{GW}$ [Peyré et al., 2016]

- Problem (5) can be solved using a KL mirror descent.
- This is equivalent to solving at each iteration t

$$\mathbf{T}^{(t+1)} = \min_{\mathbf{T}\in\mathcal{P}} \left\langle \mathbf{T}, \mathbf{G}^{(t)} \right\rangle_F + \epsilon \sum_{i,j} T_{i,j} \log T_{i,j}$$

  Where $G_{i,j}^{(t)} = 2\sum_{k,l} |D_{i,k} - D'_{j,l}|^p T_{k,l}^{(t)}$ is the gradient of the GW loss at previous point $\mathbf{T}^{(k)}$.
- Problem above solved using a Sinkhorn-Knopp algorithm of entropic OT.
- Very fast approximation exist for low rank distances [Scetbon et al., 2021].

Optimization problem

$$\min_{\mathbf{w}\in\Sigma_S} \quad \mathcal{GW}_2^2\left(\sum_{s\in[S]} w_s \overline{\boldsymbol{D}_s}\ ,\ \boldsymbol{D}\right) - \lambda\|\mathbf{w}\|_2^2$$

- Non-convex Quadratic Program *w.r.t.* $\mathbf{T}$ and $\mathbf{w}$.
- GW for fixed $\mathbf{w}$ already have an existing Frank-Wolfe solver.
- We proposed a Block Coordinate Descent algorithm

BCD Algorithm for sparse GW unmixing [Tseng, 2001]

1: **repeat**
2: Compute OT matrix $\boldsymbol{T}$ of $\mathcal{GW}_2^2(\boldsymbol{D}, \sum_s w_s \overline{\boldsymbol{D}_s})$, with FW [Vayer et al., 2018].
3: Compute the optimal $\mathbf{w}$ given $\boldsymbol{T}$ with Frank-Wolfe algorithm.
4: **until** convergence

- Since the problem is quadratic optimal steps can be obtained for both FW.
- BCD convergence in practice in a few tens of iterations.

GDL on labeled graphs

- For datasets with labeled graphs, on can learn simultaneously a dictionary of the structure $\{\overline{\boldsymbol{D}}_s\}_{s\in[S]}$ and a dictionary on the labels/features $\{\overline{\mathbf{F}}_s\}_{s\in[S]}$.
- Data fitting is Fused Gromov-Wasserstein distance $\mathcal{FGW}$, same stochastic algorithmm.

Dictionary on weights

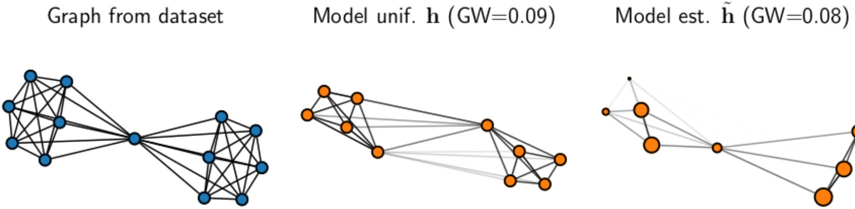
$$\min_{\substack{\{(\mathbf{w}^{(k)},\mathbf{v}^{(k)})\}_k \\ \{(\overline{\boldsymbol{D}}_s,\overline{\boldsymbol{h}}_s)\}_s}} \sum_{k=1}^{K} \mathcal{GW}_2^2\left(\boldsymbol{D}^{(k)}, \sum_s w_s^{(k)}\overline{\boldsymbol{D}_s}, \boldsymbol{h}^{(k)}, \sum_s v_s^{(k)}\overline{\boldsymbol{h}_s}\right) - \lambda\|\mathbf{w}^{(k)}\|_2^2 - \mu\|\mathbf{v}^{(k)}\|_2^2$$

- We model the graphs as a linear model on the structure and the node weights

$$(\boldsymbol{D}^{(k)}, \boldsymbol{h}^{(k)}) \longrightarrow \quad \left(\sum_s w_s^{(k)}\boldsymbol{D}_s, \sum_s v_s^{(k)}\overline{\boldsymbol{h}_s}\right)$$

- This allows for sparse weights $\boldsymbol{h}$ so embedded graphs with different order.
- We provide in [Vincent-Cuaz et al., 2021] subgradients of GW *w.r.t.* the mass $\boldsymbol{h}$.

Experiments - Unsupervised representation learning



Comparison of fixed and learned weights dictionaries

- Graph taken from the IMBD dataset.
- Show original graph and representation after projection on the embedding.
- Uniform weight $\boldsymbol{h}$ has a hard time representing a central node.
- Estimated weights $\tilde{\boldsymbol{h}}$ recover a central node.
- In addition some nodes are discarded with 0 weight (graphs can change order).

Table 1. Clustering: Rand Index computed for benchmarked approaches on real datasets.

	no attribute		discrete attributes		real attributes			
models	IMDB-B	IMDB-M	MUTAG	PTC-MR	BZR	COX2	ENZYMES	PROTEIN
GDL(ours)	**51.64(0.59)**	55.41(0.20)	**70.89(0.11)**	**51.90(0.54)**	**66.42(1.96)**	**59.48(0.68)**	66.97(0.93)	**60.49(0.71)**
GWF-r	51.24 (0.02)	**55.54(0.03)**	-	-	52.42(2.48)	56.84(0.41)	**72.13(0.19)**	59.96(0.09)
GWF-f	50.47(0.34)	54.01(0.37)	-	-	51.65(2.96)	52.86(0.53)	71.64(0.31)	58.89(0.39)
GW-k	50.32(0.02)	53.65(0.07)	57.56(1.50)	50.44(0.35)	56.72(0.50)	52.48(0.12)	66.33(1.42)	50.08(0.01)
SC	50.11(0.10)	54.40(9.45)	50.82(2.71)	50.45(0.31)	42.73(7.06)	41.32(6.07)	70.74(10.60)	49.92(1.23)

Clustering Experiments on real datasets

- Different data fitting losses:
  - Graphs without node attributes : Gromov-Wasserstein.
  - Graphs with node attributes (discrete and real): Fused Gromov-Wasserstein.
- We learn a dictionary on the dataset and perform K-means in the embedding using the Mahalanobis distance approximation.
- Compared to GW Factorization (GWF) [Xu, 2020] and spectral clustering.
- Similar performance for supervised classification (using GW in a kernel).

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

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

Frank, M. and Wolfe, P. (1956).
An algorithm for quadratic programming.
Naval research logistics quarterly, 3(1-2):95–110.

Genevay, A., Chizat, L., Bach, F., Cuturi, M., and Peyré, G. (2018).
Sample complexity of sinkhorn divergences.
arXiv preprint arXiv:1810.02733.

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

Mairal, J., Bach, F., Ponce, J., and Sapiro, G. (2009).
Online dictionary learning for sparse coding.
In *Proceedings of the 26th annual international conference on machine learning*, pages 689–696.

Memoli, 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.

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

Scetbon, M., Peyré, G., and Cuturi, M. (2021).
Linear-time gromov wasserstein distances using low rank couplings and costs.
arXiv preprint arXiv:2106.01128.

Séjourné, T., Vialard, F.-X., and Peyré, G. (2020).
The unbalanced gromov wasserstein distance: Conic formulation and relaxation.
arXiv preprint arXiv:2009.04266.

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.

Tseng, P. (2001).
Convergence of a block coordinate descent method for nondifferentiable minimization.
Journal of optimization theory and applications, 109(3):475–494.

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. (2020).
Fused gromov-wasserstein distance for structured objects.
Algorithms, 13 (9):212.

Vayer, T., Flamary, R., Tavenard, R., Chapel, L., and Courty, N. (2019).
Sliced gromov-wasserstein.
In *Neural Information Processing Systems (NeurIPS)*.

Vincent-Cuaz, C., Flamary, R., Corneli, M., Vayer, T., and Courty, N. (2022).
Semi-relaxed gromov wasserstein divergence with applications on graphs.
In *International Conference on Learning Representations (ICLR)*.

Vincent-Cuaz, C., Vayer, T., Flamary, R., Corneli, M., and Courty, N. (2021).
Online graph dictionary learning.
In *International Conference on Machine Learning (ICML)*.

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