Online Graph Dictionary Learning

Ruike Zhu

2/9 2023

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

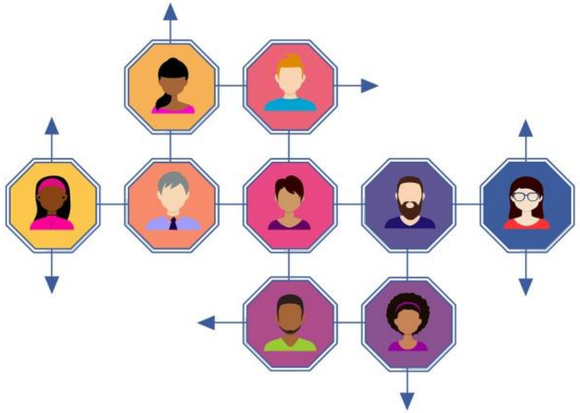
Outline

- Introduction
- Preliminaries
- Online graph dictionary learning
- Experiments
- Conclusion

Outline

- Introduction
- Preliminaries
- Online graph dictionary learning
- Experiments
- Conclusion

Introduction



- Graphs has been of great interest in the last decades
  - ◊ molecule compounds
  - ◊ brain connectivity
  - ◊ Social networks
  - ◊ …
- Designing good representations for these data is challenging
  - ◊ non-vectorial
  - ◊ requires dedicated modelling of their representing structures
- Introduce an unsupervised representation learning algorithm

Outline

- Introduction
- Preliminaries
    - ◊ Dictionary Learning
    - ◊ (Fused) Gromov-Wasserstein for structured data
- Online graph dictionary learning
- Experiments
- Conclusion

Preliminaries

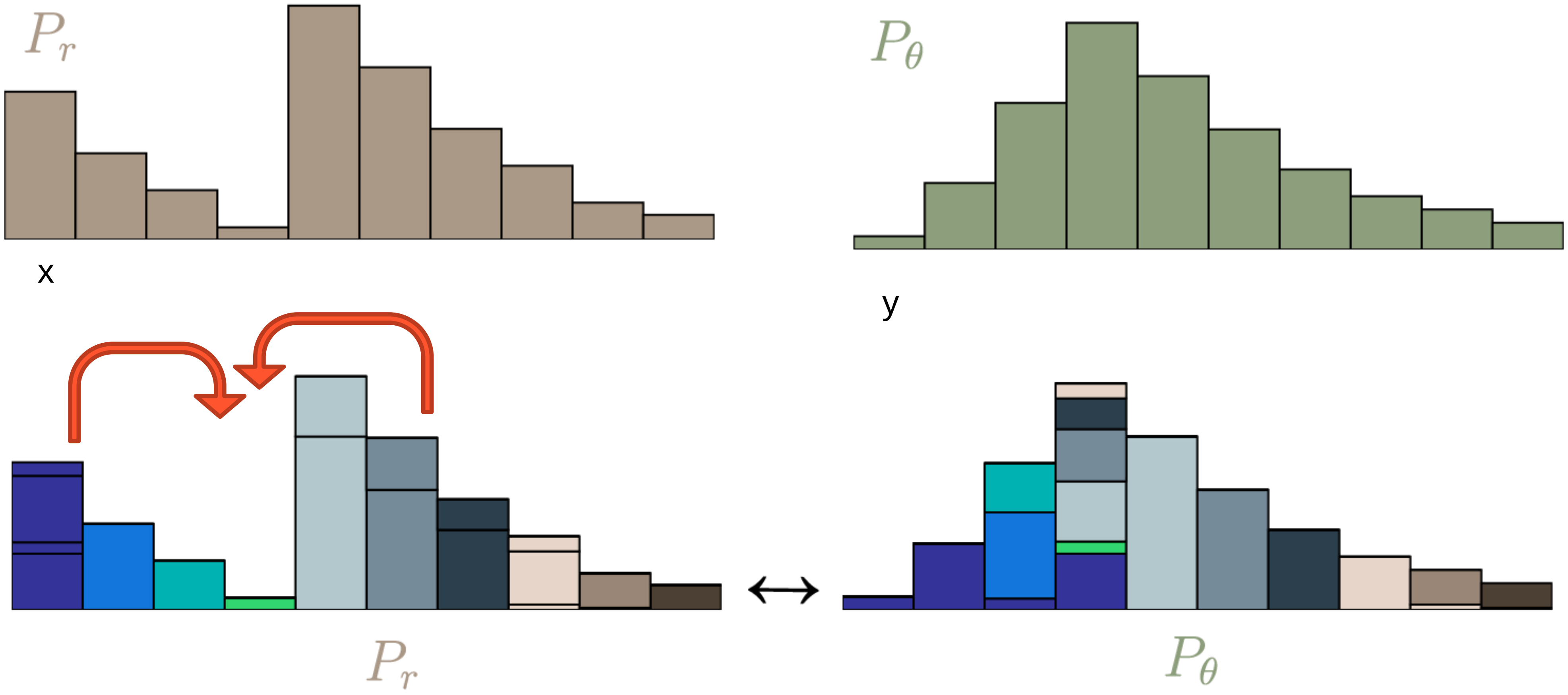
- Dictionary learning
  - ◊ A field of unsupervised learning that aims at estimating a linear representation of the data. The linear subspace is defined by the span of a family of vectors, called **atoms**, which constitute a **dictionary**
  - ◊ These atoms are inferred from the input data by minimizing a reconstruction error
- Linear modeling of graph:
  - ◊ Given: a graph $G = (\boldsymbol{C}, \boldsymbol{h})$ with intra-cost matrices $\boldsymbol{C} \in \mathbb{R}^{N\times N}$, node weight $\boldsymbol{h} \in \mathbb{R}^{N}$ and a dictionary $\{\overline{\boldsymbol{C}}_s\}_{s\in[S]}$
  - ◊ Output: a linear representation $\sum_{s\in[S]} \omega_s \overline{\boldsymbol{C}}_s$ of graph G with minimized reconstruction error

[1] Vincent-Cuaz, Cédric, et al. "Online graph dictionary learning." International Conference on Machine Learning. PMLR, 2021.

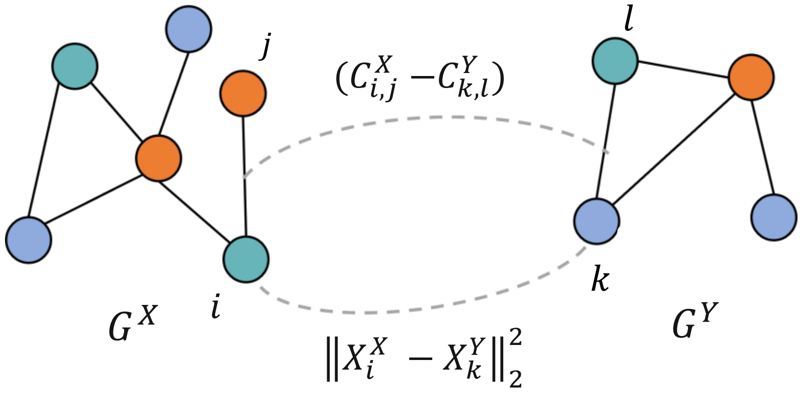
Preliminaries

$\sum_x \boldsymbol{T}(x,y) = \boldsymbol{P}_r(y)$,
$\sum_y \boldsymbol{T}(x,y) = \boldsymbol{P}_\theta(x)$

- Earth Mover's Distance: $EMD(\boldsymbol{P}_r, \boldsymbol{P}_\theta) = min \sum_{x,y} \|x-y\| \boldsymbol{T}(x,y)$



Preliminaries



- (Fused) Gromov-Wasserstein for graph data
  - ◊ Given two graphs $G^X = (\boldsymbol{C}^X, \boldsymbol{h}^X)$, $G^Y = (\boldsymbol{C}^Y, \boldsymbol{h}^Y)$

    - $\boldsymbol{C}^X \in \mathbb{R}^{N^X \times N^X}$ and $\boldsymbol{C}^Y \in \mathbb{R}^{N^Y \times N^Y}$ are the intra-cost matrices.

    - $\boldsymbol{h}^X, \boldsymbol{h}^Y$ is a vector of node weights in graph, usually define $\boldsymbol{h}^X = \frac{1}{N^X}\mathbf{1}_{N^X}$ if without any prior knowledge

Distribution density

  - ◊ The GW_p distance between G^X and G^Y is defined as:

$$\min_{\boldsymbol{T}\epsilon\Pi(\boldsymbol{h}^X,\boldsymbol{h}^Y)} \Big(\sum_{\substack{i,j\epsilon G^X \\ k,l\epsilon G^Y}} \left(C^X_{i,j} - C^Y_{k,l}\right)^p T_{ik}T_{jl}\Big)$$

T act as a probabilistic matching of nodes which tends to associate pairs of nodes that have similar pairwise relations in C^X and C^Y

  - ◊ $\Pi(\boldsymbol{h}^X, \boldsymbol{h}^Y) := \{\boldsymbol{T}\epsilon\mathbb{R}^{N^X \times N^Y} | \boldsymbol{T}\mathbf{1}_{N^Y} = \boldsymbol{h}_X, \boldsymbol{T}\mathbf{1}_{N^X} = \boldsymbol{h}_Y\}$
  - ◊ FGW distance: $\min_{\boldsymbol{T}\epsilon\Pi(\boldsymbol{h}^X,\boldsymbol{h}^Y)} \big(\sum_{\substack{i,j\epsilon G^X \\ k,l\epsilon G^Y}} \big((1-\alpha)\|X^X_i - X^Y_k\|^2_2 + \alpha\left(C^X_{i,j} - C^Y_{k,l}\right)^p\big) T_{ik}T_{jl}\big)$

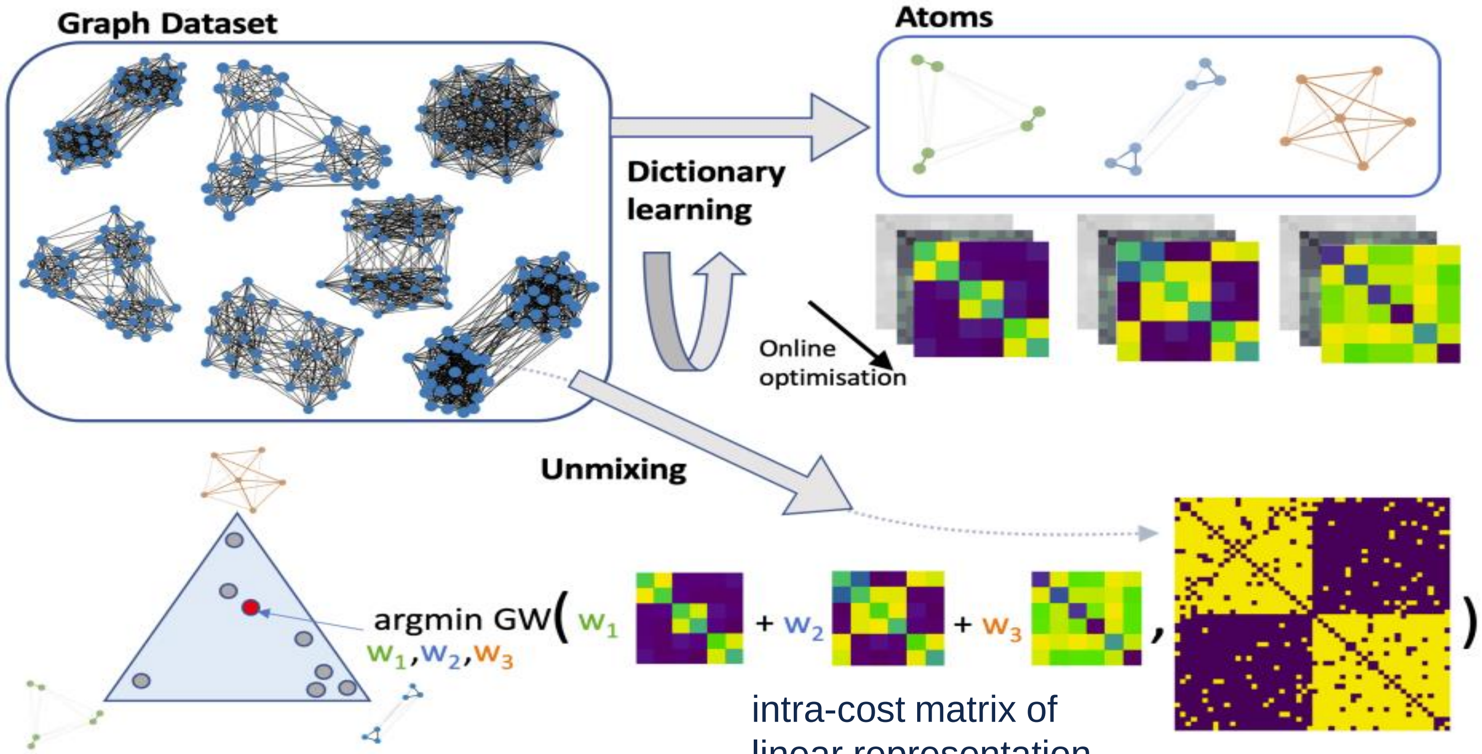
[2] Peyré, Gabriel, Marco Cuturi, and Justin Solomon. "Gromov-wasserstein averaging of kernel and distance matrices." International conference on machine learning. PMLR, 2016.

Outline

- Introduction
- Preliminaries
- Online graph dictionary learning
    - ◊ GW unmixing
    - ◊ Dictionary learning and online algorithm
- Experiments
- Conclusion

Algorithm overview

- Minimization of the GW distance between the intra-cost matrix associated to the input graph and its linear representation in the dictionary
- - Given a set of graph $\{G^{(k)}: (\boldsymbol{C}^{(k)}, \boldsymbol{h}^{(k)})\}_{k\epsilon[K]}$
  - Output the optimal embedding vector $\boldsymbol{\omega}\epsilon\Sigma_S$ of each graph G and a dictionary $\{\overline{\boldsymbol{C}}_s\}_{s\epsilon[S]}$



intra-cost matrix of linear representation

intra-cost matrix of input graph

[1] Vincent-Cuaz, Cédric, et al. "Online graph dictionary learning." International Conference on Machine Learning. PMLR, 2021.

Algorithm overview

- - Given a set of graph $\{G^{(k)}: (\boldsymbol{C}^{(k)}, \boldsymbol{h}^{(k)})\}_{k\epsilon[K]}$

  - Output the optimal embedding vector $\boldsymbol{\omega}\epsilon\Sigma_S$ of each graph G and a dictionary $\{\overline{\boldsymbol{C}}_s\}_{s\epsilon[S]}$

- Workflow of algorithm:

```
Randomly initialize {C̄_s}_{sϵ[S]}
For each batch of graph:
        Initialize ω = (1/S) 1_S
        repeat
                Fix dictionary {C̄_s}_{sϵ[S]} and embedding vector ω, update transport plan T
                Fix dictionary {C̄_s}_{sϵ[S]} and transport plan T, update embedding vector ω
        until converge
Fix ω and T, update {C̄_s}_{sϵ[S]}, projected gradient step
```

GW Unmixing

[1] Vincent-Cuaz, Cédric, et al. "Online graph dictionary learning." International Conference on Machine Learning. PMLR, 2021.

GW Unmixing

- Minimization of the GW distance between the similarity matrix associated to the graph and its linear representation in the dictionary
- - Given a graph $G = (\boldsymbol{C}, \boldsymbol{h})$

  a dictionary $\{\overline{\boldsymbol{C}}_s\}_{s\epsilon[S]}$ with S atoms of which each atom contains N_s nodes

  - Output the optimal transport matrix $\boldsymbol{T}$,

  embedding vector $\boldsymbol{\omega}$ of graph G
- Object function of unmixing problem given dictionary:

◊ $$\min_{\boldsymbol{\omega}\epsilon\Sigma_S} \left(GW_2^2\left(\boldsymbol{C}, \underline{\sum_{s\epsilon[S]} \omega_s \overline{\boldsymbol{C}}_s}\right) - \boxed{\lambda}\|\boldsymbol{\omega}\|_2^2 \right)$$

λ is the negative quadratic regularization promoting sparsity [1]

When C_s is adjacency matrix, this represent the probabilities of connection between the nodes in linear representation

[1] Vincent-Cuaz, Cédric, et al. "Online graph dictionary learning." International Conference on Machine Learning. PMLR, 2021.

Fix $\{\overline{C}_s\}_{s\epsilon[S]}, \omega$ – update T

◊ $\min_{\omega\epsilon\Sigma_S} \left(GW_2^2\left(C, \sum_{s\epsilon[S]} \omega_s \overline{C}_s\right) - \lambda\|\omega\|_2^2 \right)$ (1)

Algorithm 3 BCD for GW unmixing problem

1: Initialize $w = \frac{1}{S}\mathbf{1}_S$
2: **repeat**
3: Compute OT matrix T of $GW_2^2\left(C, \widetilde{C}(w)\right)$, with CG algorithm (Vayer et al., 2018, Alg.1 & 2).
4: Compute the optimal w solving equation 1 for a fixed T with CG algorithm 4
5: **until** convergence

◊ $GW_2^2\left(C, \sum_{s\epsilon[S]} \omega_s \overline{C}_s\right) = \min_{T\epsilon\Pi(h,\overline{h}_s)} \langle L, T\rangle; L = C^2 h \mathbf{1}_N^T + \mathbf{1}_{N_s} h_s^T (\widetilde{C}(\omega)^2)^T - 2CT\widetilde{C}(\omega)^T$ according to equation (5) in [3]

◊ $T^{(t)} = \underset{T\epsilon\Pi(h,\overline{h}_s)}{\operatorname{argmin}} \langle C^2 h \mathbf{1}_N^T + \mathbf{1}_{N_s} h_s^T (\widetilde{C}(\omega)^2)^T - 2CT\widetilde{C}(\omega)^T, T\rangle$; $\widetilde{C}(\omega) = \sum_{s\epsilon[S]} \omega_s \overline{C}_s$,

◊ This corresponds to an optimal transport problem, can be solved iteratively by the entropic regularization-based method

[1] Vincent-Cuaz, Cédric, et al. "Online graph dictionary learning." International Conference on Machine Learning. PMLR, 2021.
[3] Xu, Hongtengl. "Gromov-Wasserstein factorization models for graph clustering." Proceedings of the AAAI conference on artificial intelligence. Vol. 34. No. 04. 2020.

Fix $\{\overline{C}_s\}_{s\epsilon[S]}$, T – update $\boldsymbol{\omega}$

$\varepsilon(\boldsymbol{C}, \widetilde{\boldsymbol{C}}(\boldsymbol{\omega}), \boldsymbol{T}) = GW_2^2\left(\boldsymbol{C}, \sum_{s\epsilon[S]} \omega_s \overline{\boldsymbol{C}}_s\right) - \lambda\|\boldsymbol{\omega}\|_2^2$

Algorithm 4 CG for solving GW unmixing problem *w.r.t* $\boldsymbol{w}$ given $\boldsymbol{T}$

1: **repeat**

2: Compute $\boldsymbol{g}$, gradients *w.r.t* $\boldsymbol{w}$ of $\mathcal{E}(\boldsymbol{C}, \widetilde{\boldsymbol{C}}(\boldsymbol{w}), \boldsymbol{T})$ following equation 2

3: Find direction $\boldsymbol{x}^\star = \arg\min_{\boldsymbol{x}\in\Sigma_S} \boldsymbol{x}^T \boldsymbol{g}$

4: Line-search: denoting $\boldsymbol{z}(\gamma) = \gamma\boldsymbol{x}^\star + (1-\gamma)\boldsymbol{w}$, #ensure the constrain of $\|\boldsymbol{\omega}\|_1 = 1$

$$\gamma^\star = \arg\min_{\gamma\in(0,1)} \mathcal{E}(\boldsymbol{C}, \widetilde{\boldsymbol{C}}(\boldsymbol{z}(\gamma)), \boldsymbol{T}) = \arg\min_{\gamma\in(0,1)} a\gamma^2 + b\gamma + c$$

5: $\boldsymbol{w} \leftarrow \boldsymbol{z}(\gamma^\star)$

6: **until** convergence

$\varepsilon(\boldsymbol{C}, \widetilde{\boldsymbol{C}}(\boldsymbol{\omega}), \boldsymbol{T})$ can be converted to second-order polynomial equation with coefficients as below

Partial derivates of the GW objective ε

◊ $$\frac{\partial\varepsilon}{\partial\omega_s}(\boldsymbol{C}, \widetilde{\boldsymbol{C}}(\boldsymbol{\omega}), \boldsymbol{T}) = 2tr\left\{\left(\overline{\boldsymbol{C}}_s \odot \widetilde{\boldsymbol{C}}(\boldsymbol{\omega})\right)\boldsymbol{h}\boldsymbol{h}^T - \overline{\boldsymbol{C}}_s \boldsymbol{T}^T \boldsymbol{C}^T \boldsymbol{T}\right\} \quad (2)$$

$$a = tr\{\left(\widetilde{\boldsymbol{C}}(\boldsymbol{x}^\star - \boldsymbol{w}) \odot \widetilde{\boldsymbol{C}}(\boldsymbol{x}^\star - \boldsymbol{w})\right)\boldsymbol{h}\boldsymbol{h}^T\} - \lambda\|\boldsymbol{x}^\star - \boldsymbol{w}\|_2^2$$

$$b = 2tr\{\left(\widetilde{\boldsymbol{C}}(\boldsymbol{x}^\star - \boldsymbol{w}) \odot \widetilde{\boldsymbol{C}}(\boldsymbol{w})\right)\boldsymbol{h}\boldsymbol{h}^\top - \widetilde{\boldsymbol{C}}(\boldsymbol{x}^\star - \boldsymbol{w})\boldsymbol{T}^\top \boldsymbol{C}^T \boldsymbol{T}\} - 2\lambda\langle \boldsymbol{w}, \boldsymbol{x} - \boldsymbol{w}\rangle$$

[1] Vincent-Cuaz, Cédric, et al. "Online graph dictionary learning." International Conference on Machine Learning. PMLR, 2021.

Online algorithm

- Assume the dictionary is unknown and need to be estimated from dataset
- - Given K graphs $\{G^{(k)}: (\boldsymbol{C}^{(k)}, \boldsymbol{h}^{(k)})\}_{k\epsilon[K]}$

  - Output the optimal embedding vector $\boldsymbol{\omega}$ of each graph G and a dictionary $\{\overline{\boldsymbol{C}}_s\}_{s\epsilon[S]}$
- Object function:

◊ $$\min_{\substack{\{\boldsymbol{\omega}^{(k)}\}_{k\epsilon[K]} \\ \{\overline{\boldsymbol{C}}_s\}_{s\epsilon[S]}}} \sum_{k=1}^{K} \left(GW_2^2\left(\boldsymbol{C}^{(k)}, \ \sum_{s\epsilon[S]} \omega_s^{(k)} \overline{\boldsymbol{C}}_s\right) - \lambda \left\| \boldsymbol{\omega}^{(k)} \right\|_2^2 \right)$$

◊ $\boldsymbol{\omega}^{(k)} \epsilon \Sigma_S, \ \overline{\boldsymbol{C}}_s \epsilon S_N(\mathbb{R})$

Fix T, ω – update $\{\overline{C}_s\}_{s\epsilon[S]}$

$$\min_{\substack{\{\boldsymbol{\omega}^{(k)}\}_{k\epsilon[B]} \\ \{\overline{C}_s\}_{s\epsilon[S]}}} \sum_{k=1}^{B} \left(GW_2^2(\boldsymbol{C}^{(k)}, \sum_{s\epsilon[S]} \omega_s^{(k)} \overline{\boldsymbol{C}}_s) - \lambda \|\boldsymbol{\omega}^{(k)}\|_2^2\right)$$

Algorithm 5 GDL: stochastic update of atoms $\{\overline{C}_s\}_{s\in[S]}$

1: Sample a minibatch of graphs $\mathcal{B} := \{C^{(k)}\}_{k\in\mathcal{B}}$.
2: Compute optimal $\{(\boldsymbol{w}^{(k)}, \boldsymbol{T}^{(k)})\}_{k\in[B]}$ by solving B independent unmixing problems with Alg.3.
3: Projected gradient step with estimated gradients $\widetilde{\nabla}_{\overline{C}_s}$ (see equation 54), $\forall s \in [S]$:

$$\overline{C}_s \leftarrow Proj_{S_N(\mathbb{R})}(\overline{C}_s - \eta_C \widetilde{\nabla}_{\overline{C}_s})$$

ensure constrain of $\overline{C}_s$, need to be symmetric matrix

Estimated gradients *w.r.t* $\{\overline{C}_s\}$ over a minibatch of graphs $\mathcal{B} := \{C^{(k)}\}_{k\in\mathcal{B}}$ given unmixing solutions $\{(\boldsymbol{w}^{(k)}, \boldsymbol{T}^{(k)})\}_{k\in[B]}$ read:

$$\widetilde{\nabla}_{\overline{C}_s}\left(\sum_{k\in\mathcal{B}} \mathcal{E}(C^{(k)}, \widetilde{C}(\boldsymbol{w}^{(k)}), \boldsymbol{T}^{(k)})\right) = \frac{2}{B} \sum_{k\in\mathcal{B}} w_s^{(k)} \{\widetilde{C}(\boldsymbol{w}^{(k)}) \odot \boldsymbol{h}\boldsymbol{h}^\top - \boldsymbol{T}^{(k)\top} C^{(k)\top} \boldsymbol{T}^{(k)}\} \tag{54}$$

[1] Vincent-Cuaz, Cédric, et al. "Online graph dictionary learning." International Conference on Machine Learning. PMLR, 2021.

Algorithm overview

- \- Given a set of graph $\{G^{(k)}: (\boldsymbol{C}^{(k)}, \boldsymbol{h}^{(k)})\}_{k \epsilon [K]}$

  \- Output the optimal embedding vector $\boldsymbol{\omega} \epsilon \Sigma_S$ of each graph G and a dictionary $\{\overline{\boldsymbol{C}}_s\}_{s \epsilon [S]}$

- Workflow of algorithm:

```
Randomly initialize {C̄_s}_{sϵ[S]}
For each batch of graph:
        Initialize ω = (1/S) 1_S
        repeat
                Fix dictionary {C̄_s}_{sϵ[S]} and embedding vector ω, update transport plan T
                Fix dictionary {C̄_s}_{sϵ[S]} and transport plan T, update embedding vector ω
        until converge
Fix ω and T, update {C̄_s}_{sϵ[S]} , projected gradient step
```

Algorithm 3

Algorithm 4

Algorithm 5

[1] Vincent-Cuaz, Cédric, et al. "Online graph dictionary learning." International Conference on Machine Learning. PMLR, 2021.

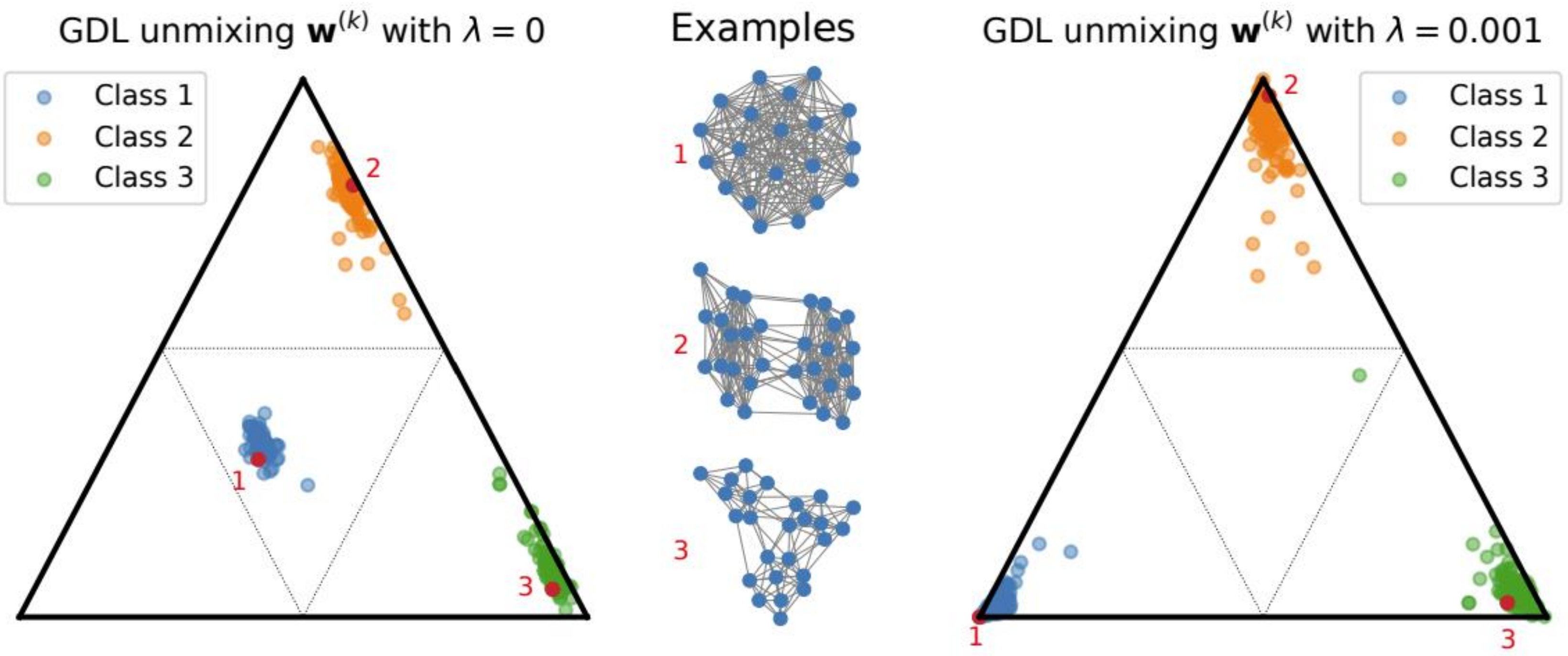
Outline

- Introduction
- Preliminaries
- Online graph dictionary learning
- Experiments
  - ◊ With synthesis graph
  - ◊ With real-life graph
  - ◊ With Stream graph
- Conclusion

With synthesis graph

$$\min_{\substack{\{\boldsymbol{\omega}^{(k)}\}_{k\epsilon[K]} \\ \{\overline{\boldsymbol{C}}_s\}_{s\epsilon[S]}}} \sum_{k=1}^{K} \left(GW_2^2\left(\boldsymbol{C}^{(k)}, \sum_{s\epsilon[S]} \omega_s^{(k)} \overline{\boldsymbol{C}}_s\right) - \lambda \left\|\boldsymbol{\omega}^{(k)}\right\|_2^2 \right)$$

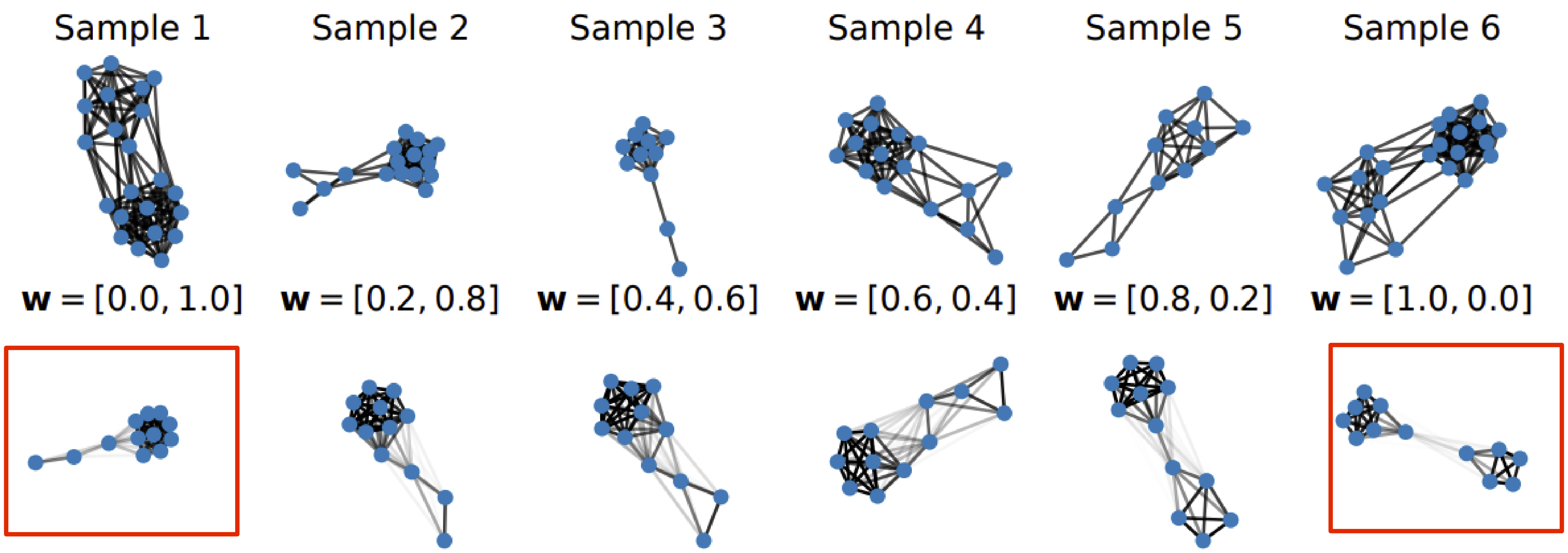
- Three different generative structures: dense, two clusters and three clusters.
- Nodes are assigned to clusters into equal proportions.
- For each generative structure 100 graphs are sampled.



[1] Vincent-Cuaz, Cédric, et al. "Online graph dictionary learning." International Conference on Machine Learning. PMLR, 2021.

With synthesis graph

- Estimate on D2 a dictionary with 2 atoms of order 12. The interpolation between the two estimated atoms for some samples is reported.



On the top, a random sample of real graphs from D2 (two blocks).
On the bottom, reconstructed graphs as linear combination of two estimated atoms (varying proportions for each atom).

[1] Vincent-Cuaz, Cédric, et al. "Online graph dictionary learning." International Conference on Machine Learning. PMLR, 2021.

With real-life graph

- Graph clustering

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

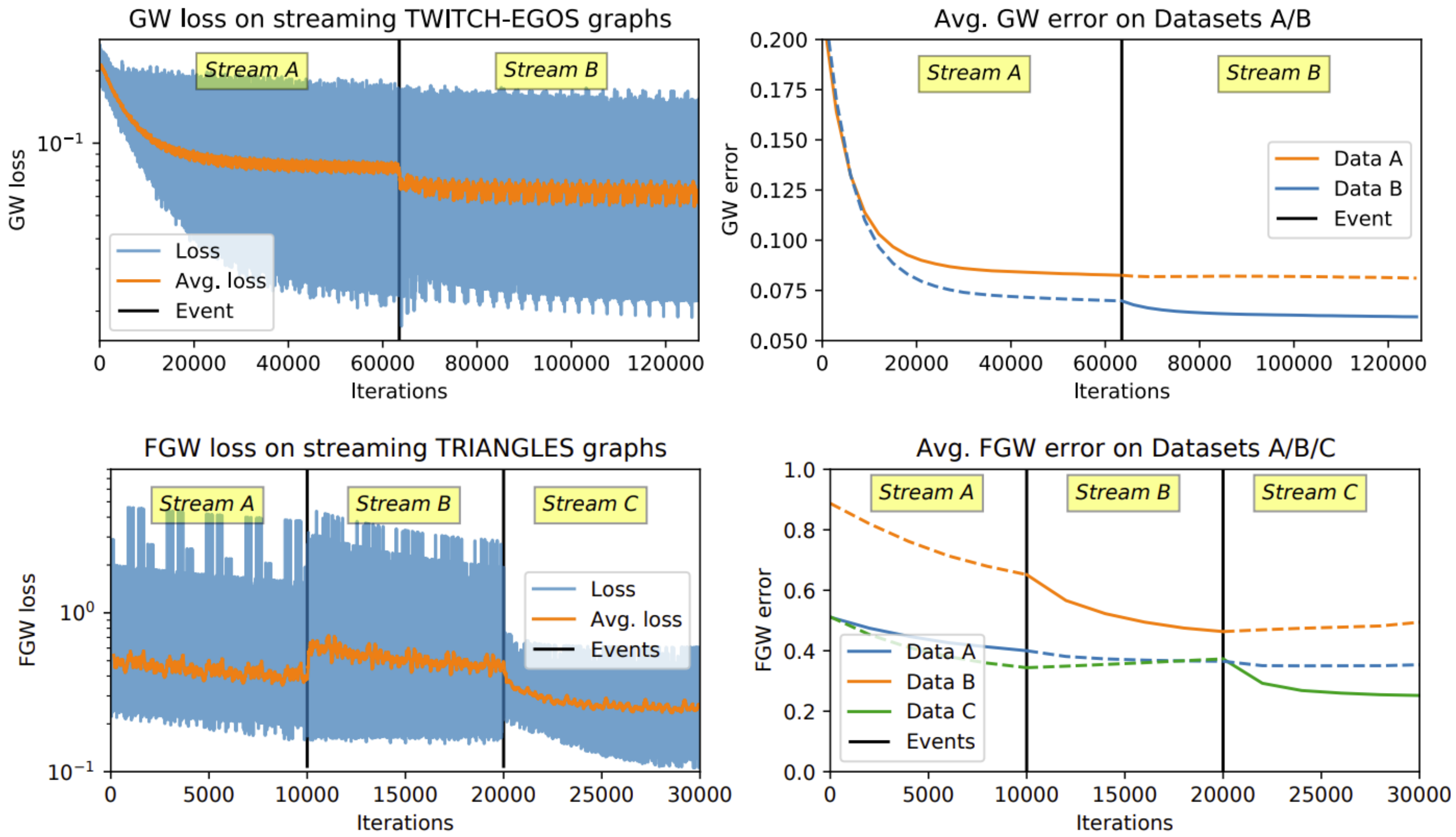
	NO ATTRIBUTE		DISCRETE ATTRIBUTES		REAL ATTRIBUTES			
MODELS	IMDB-B	IMDB-M	MUTAG	PTC-MR	BZR	COX2	ENZYMES	PROTEIN
GDL (ours)	51.32(0.30)	55.08(0.28)	70.02(0.29)	51.53(0.36)	62.59(1.68)	58.39(0.52)	66.97(0.93)	60.22(0.30)
$\boxed{\text{GDL}_\lambda \text{ (ours)}}$	**51.64(0.59)**	55.41(0.20)	**70.89(0.11)**	**51.90(0.54)**	**66.42(1.96)**	**59.48(0.68)**	66.79(1.12)	**60.49(0.71)**
GWF-r	51.24 (0.02)	**55.54(0.03)**	68.83(1.47)	51.44(0.52)	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)	58.96(1.91)	50.87(0.79)	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)

With negative quadratic regularization

[1] Vincent-Cuaz, Cédric, et al. "Online graph dictionary learning." International Conference on Machine Learning. PMLR, 2021.

Stream graph

- Dataset: TWITCH-EGOS with 2 atoms of 14 nodes each (top)
  TRIANGLES with 4 atoms of 17 nodes each (bottom).



[1] Vincent-Cuaz, Cédric, et al. "Online graph dictionary learning." International Conference on Machine Learning. PMLR, 2021.

Outline

- Introduction
- Preliminaries
- Online graph dictionary learning
- Experiments
- Conclusion

Conclusion

- Introduction
- Preliminaries
  - ◊ Dictionary Learning
  - ◊ (Fused) Gromov-Wasserstein for structured data
- Online graph dictionary learning
  - ◊ GW unmixing
  - ◊ Dictionary learning and online algorithm
- Experiments
  - ◊ With synthesis graph
  - ◊ With real-life graph
  - ◊ With Stream graph

Reference

[1] Vincent-Cuaz, Cédric, et al. "Online graph dictionary learning." International Conference on Machine Learning. PMLR, 2021.

[2] Peyré, Gabriel, Marco Cuturi, and Justin Solomon. "Gromov-wasserstein averaging of kernel and distance matrices." International conference on machine learning. PMLR, 2016.

[3] Xu, Hongtengl. "Gromov-Wasserstein factorization models for graph clustering." Proceedings of the AAAI conference on artificial intelligence. Vol. 34. No. 04. 2020.

Thanks!