Partial Optimal Transport with Applications on Positive-Unlabeled Learning (NeurIPS 2020)

Laetitia Chapel, Mokhhtar Z. Alaya, Gilles Gasso

November 10, 2023

Presented by Mengying Yan @ ML in practice reading group

Introduction

Purpose

- Address partial Wasserstein and partial Gromov-Wasserstein problems and propose algorithms to solve them
- Formulate positive-unlabeled (PU) learning using partial optimal transport
- Show that partial-GW metrics are efficient for PU learning when positive and unlabeled datsets come from different domains or have different features

Intuition: Seeing PU learning problem as transporting a probability mass from unlabeled (source) dataset to the positive (target) dataset

What is optimal transport?

Optimal transport is a tool to compare probability distributions

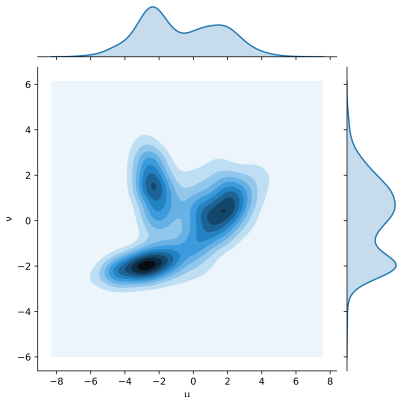


Figure: Two one-dimensional distributions and one possible joint distribution that defines a transport plan between them (not unique). Source: Wikipedia.

Background on optimal transport

- Two empirical distributions over $\mathcal{X}$ and $\mathcal{Y}$

$$\mathbf{p} = \sum_{i=1}^{n} p_i \delta_{x_i}, \textit{and} \quad \mathbf{q} = \sum_{j=1}^{m} q_i \delta_{y_i}$$

- $\mathbf{T}$ is a transport plan (see it as a joint distribution with marginal distribution p and q) belongs to couplings $\Pi(p, q)$
- Cost $\mathbf{C}$
- Total cost $\langle \mathbf{C}, \mathbf{T} \rangle_F = \sum_{i=1}^{n} \sum_{j=1}^{m} C_{ij} T_{ij}$
- The **optimal transport plan** is the plan with the minimal cost out of all possible transport plans

p-Wasserstein distance

Couplings $\Pi(p,q) = \{T \in \mathbb{R}_+^{|p|\times|q|} | T\mathbb{1}_{|q|} = p,\, T^T\mathbb{1}_{|p|} = q\}$
Cost $\mathbf{C} = \mathbf{D}^p = (D_{ij}^p)_{i,j}$, where D_{ij} is geometric distance between x_i and y_j

p-Wasserstein distance

$$W_p^p(\mathbf{p},\mathbf{q}) = \min_{T\in\Pi(p,q)} \langle \mathbf{C}, \mathbf{T}\rangle_F = \min_{T\in\Pi(p,q)} \sum_{i=1}^{n}\sum_{j=1}^{m} C_{ij}T_{ij}$$

Gromov-Wasserstein distance

For x_i and y_j not in the same underlying space.

Intra-domain distance of source $\mathbf{C}^s = (C^s(x_i, x_k))_{i,k}$, and target $\mathbf{C}^t = (C^t(y_j, y_l))_{i,k}$

Gromov-Wasserstein distance

$$GW_p^p(\mathbf{p}, \mathbf{q}) = \min_{T \in \Pi(p,q)} \sum_{i,k=1}^{n} \sum_{j,l=1}^{m} |C_{ik}^s - C_{jl}^t| T_{ij} T_{jl}$$

Partial Wasserstein distance

- Transporting only a fraction of the mass instead of requiring all the mass has to be transported.
- Couplings $\Pi^u(p,q) = \{T \in \mathbb{R}_+^{|p|\times|q|} \,|\, T\mathbb{1}_{|q|} \leq p,\, T^T\mathbb{1}_{|p|} \leq q,\, \mathbb{1}_{|p|}^T T\mathbb{1}_{|q|} = s\}$

Partial Wasserstein distance

$$PW_p^p(\mathbf{p},\mathbf{q}) = \min_{T\in\Pi(p,q)} \sum_{i=1}^{n}\sum_{j=1}^{m} C_{ij}T_{ij}$$

Propose to solve partial-W by adding dummy points and extending the cost matrix

$$\bar{C} = \begin{bmatrix} \boldsymbol{C} & \xi\mathbb{1}_{|\boldsymbol{q}|} \\ \xi\mathbb{1}_{|\boldsymbol{p}|}^\top & 2\xi + A \end{bmatrix}$$

Partial Gromov-Wasserstein distance

$$PGW_2^2(\boldsymbol{p},\boldsymbol{q}) = \min_{\boldsymbol{T}\in\Pi^u(\boldsymbol{p},\boldsymbol{q})} \mathcal{J}_{\boldsymbol{C}^s,\boldsymbol{C}^t}(\boldsymbol{T})$$

$$\mathcal{J}_{\boldsymbol{C}^s,\boldsymbol{C}^t}(\boldsymbol{T}) = \frac{1}{2}\sum_{i,k=1}^{n}\sum_{j,l=1}^{m}(C_{ik}^s - C_{jl}^t)^2 T_{ij}T_{kl}.$$

Frank-Wolfe algorithm for partial-GW

Conditional gradient method

Algorithm 1 Frank-Wolfe algorithm for partial-GW

Input: Source and target samples: $(\mathcal{X}, \boldsymbol{p})$ and $(\mathcal{Y}, \boldsymbol{q})$, mass s, $p = 2$, initial guess $\boldsymbol{T}^{(0)}$
Compute cost matrices $\boldsymbol{C}^s$ and $\boldsymbol{C}^t$, build $\bar{\boldsymbol{p}} = [\boldsymbol{p}, \|\boldsymbol{q}\|_1 - s]$ and $\bar{\boldsymbol{q}} = [\boldsymbol{q}, \|\boldsymbol{p}\|_1 - s]$
for $k = 0, 1, 2, 3, \ldots$ **do**
  $\boldsymbol{G}^{(k)} \leftarrow \mathcal{M}(\boldsymbol{C}^s, \boldsymbol{C}^t) \circ \boldsymbol{T}^{(k)}$ `// Compute the gradient` $\nabla \mathcal{J}_{\boldsymbol{C}^s, \boldsymbol{C}^t}(\boldsymbol{T}^{(k)})$
  $\bar{\boldsymbol{T}}^{(k)} \leftarrow \operatorname{argmin}_{\boldsymbol{T} \in \Pi(\bar{\boldsymbol{p}}, \bar{\boldsymbol{q}})} \langle \bar{\boldsymbol{G}}^{(k)}, \boldsymbol{T} \rangle_F$ `// Compute partial-W, with` $\bar{\boldsymbol{G}}$ `as in eq. (1)`
  Get $\widetilde{\boldsymbol{T}}^{(k)}$ from $\bar{\boldsymbol{T}}^{(k)}$ `// Remove last row and column`
  Compute $\gamma^{(k)}$ as in Eq. (5) `// Line-search`
  $\boldsymbol{T}^{(k+1)} \leftarrow (1 - \gamma^{(k)})\boldsymbol{T}^{(k)} + \gamma^{(k)} \widetilde{\boldsymbol{T}}^{(k)}$ `// Update`
end for
Return: $\boldsymbol{T}^{(k)}$

$$\widetilde{\boldsymbol{T}}^{(k)} \leftarrow \underset{\boldsymbol{T} \in \Pi^u(\boldsymbol{p}, \boldsymbol{q})}{\operatorname{argmin}} \langle \nabla \mathcal{J}_{\boldsymbol{C}^s, \boldsymbol{C}^t}(\boldsymbol{T}^{(k)}), \boldsymbol{T} \rangle_F$$

$$\gamma^{(k)} \leftarrow \underset{\gamma \in [0,1]}{\operatorname{argmin}} \{ \mathcal{J}_{\boldsymbol{C}^s, \boldsymbol{C}^t}((1-\gamma)\boldsymbol{T}^{(k)} + \gamma \widetilde{\boldsymbol{T}}^{(k)}) \}$$

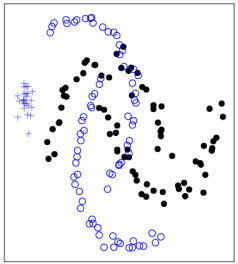
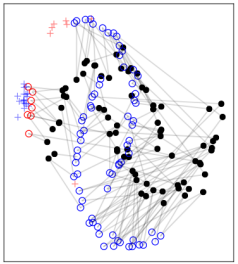
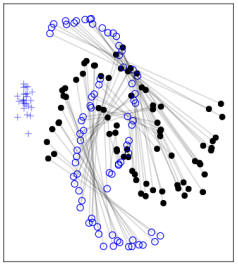
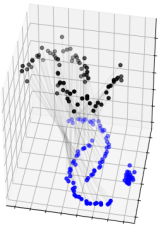
Positive and Unlabeled (PU) learning

- Positive samples: $Pos = \{x_i\}_{i=1}^{n_P}$ from $p(x|y=1)$
- Unlabeled samples: $Unl = \{x_i^U\}_{i=1}^{n_U}$ from $p(x) = \pi p(x|y=1) + (1-\pi)p(x|y=-1)$
- Class prior (true proportion of positives): $\pi = p(y=1)$
- Goal: To learn a binary classifier solely using Pos and Unl.

Idea:
Transporting a probability mass from unlabeled (source) dataset to the positive (target) one.

Optimal transport for PU learning

- (Left) Source (in black) and target (in blue) samples that have been collected under distinct environments. The source domain contains only positive points (o) whereas the target domain contains both positives and negatives (+)
- (Middle left) Partial-W fails to assign correctly all the labels in such context, red symbols indicating wrong assignments
- (Middle right) Partial-GW recovers the correct labels of the unlabeled samples, with a consistent transportation plan (gray lines), even when the datasets do not live in the same state space (Right).

PU learning formulation using partial optimal transport

- Unl as source distribution $\mathbb{X}$, Pos as target distribution $\mathbb{Y}$
- Total probability mass to be transported as the proportion of positives in the unlabeled set: $s = \pi$
- $n = n_U, m = n_P, p_i = 1/n, q_j = s/m$
- Unlabeled positives points mapped are to the positive samples while the negatives are discarded (not transported). This is done by enforce condition $T\mathbb{1}_{|q|} \leq \{p, 0\}$

Couplings $\Pi^{PU}(p, q) = \{T \in \mathbb{R}_+^{|p|\times|q|} | T\mathbb{1}_{|q|} \leq \{p, 0\}, T^T\mathbb{1}_{|p|} \leq q, \mathbb{1}_{|p|}^T T\mathbb{1}_{|q|} = s\}$

$$PUW_p^p(\mathbf{p}, \mathbf{q}) = \min_{T\in\Pi(p,q)} \sum_{i=1}^{n}\sum_{j=1}^{m} C_{ij}T_{ij}$$

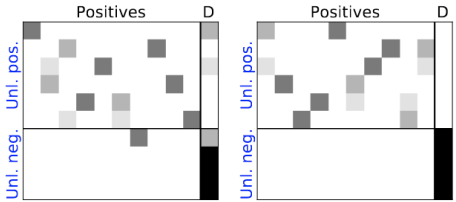
Solving the PU problem

Solving the following problem provides the solution to PU learning using partial optimal transport.

$$\bar{\boldsymbol{T}}^* = \underset{\bar{\boldsymbol{T}} \in \Pi(\bar{\boldsymbol{p}}, \bar{\boldsymbol{q}})}{\operatorname{argmin}} \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} \bar{C}_{ij} \bar{T}_{ij} + \eta \Omega(\bar{\boldsymbol{T}})$$

- Enforce $T\mathbb{1}_{|q|} \leq \{p, 0\}$
- Group-lasso regularization leads to a sparse transportation map

$$\Omega(\bar{\boldsymbol{T}}) = \sum_{i=1}^{n} \left(\|\bar{\boldsymbol{T}}_{i(:m)}\|_2 + \|\bar{\boldsymbol{T}}_{i(m+1)}\|_2 \right)$$



Experiments

1. Selected completely at random (SCAR): labeled positives are randomly drawn from positive distribution
2. Selected at random (SAR) assumption: labeled positives are selected according to some features of the samples. Distributions of Pos and Unl are from different (domains) metric space.
   - Coloted MNIST: 90%red, 10%green
   - Pos only from green, Unl mostly from red
3. Domain adaption
   - Pos: from Caltech 256
   - Unl: Amazon, Webcam, DSLR and Caltech 256 (exclude Pos)

Experimental results

Table 1: Average accuracy rates on various datasets. (G)-PW 0 indicates no noise and (G)P-W 0.025 stands for a noise level of $\alpha = 0.025$. Best values are indicated boldface.

DATASET	π	PU	PUSB	P-W 0	P-W 0.025	P-GW 0	P-GW 0.025
MUSHROOMS	0.518	91.1	90.8	96.3	**96.4**	95.0	93.1
SHUTTLE	0.786	90.8	90.3	**95.8**	94.0	94.2	91.8
PAGEBLOCKS	0.898	92.1	90.9	**92.2**	91.6	90.9	90.8
USPS	0.167	95.4	95.1	**98.3**	98.1	94.9	93.3
CONNECT-4	0.658	**65.6**	58.3	55.6	61.7	59.5	60.8
SPAMBASE	0.394	**84.3**	84.1	78.0	76.4	70.2	71.2
ORIGINAL MNIST	0.1	97.9	97.8	**98.8**	98.6	98.2	97.9
COLORED MNIST	0.1	87.0	80.0	91.5	91.5	97.3	**98.0**
SURF C→SURF C	0.1	89.3	89.4	90.0	**90.2**	87.2	87.0
SURF C→SURF A	0.1	**87.7**	85.6	81.6	81.8	85.6	85.6
SURF C→SURF W	0.1	84.4	80.5	82.2	82.0	**85.6**	85.0
SURF C→SURF D	0.1	82.0	83.2	80.0	80.0	87.6	**87.8**
DECAF C→DECAF C	0.1	93.9	**94.8**	94.0	93.2	86.4	87.2
DECAF C→DECAF A	0.1	80.5	82.2	80.2	80.2	**89.2**	88.8
DECAF C→DECAF W	0.1	82.4	83.8	80.2	80.2	**89.2**	88.6
DECAF C→DECAF D	0.1	82.6	83.6	80.8	80.6	**94.2**	93.2

Recommendations

This paper provides a new perspective of looking at positive and unlabeled learning. Optimal transport can also be used for domain adaptation.

Not easy to follow for those who are not familiar with optimal transport. Some additional sources:

- https://alexhwilliams.info/itsneuronalblog/2020/10/09/optimal-transport/
- https://hongtengxu.github.io/docs/IJCAI2023_Tutorial_OTML.pdf
- https://en.wikipedia.org/wiki/Wasserstein_metric
- Montesuma, E. F., Mboula, F. N., & Souloumiac, A. (2023). Recent Advances in Optimal Transport for Machine Learning (arXiv:2306.16156). arXiv. http://arxiv.org/abs/2306.16156

Implementation: Python Optimal Transport (POT) toolbox (https://pythonot.github.io/)