CRITEO

AI Lab

Optimal Transport and Machine Learning

A. Rakotomamonjy

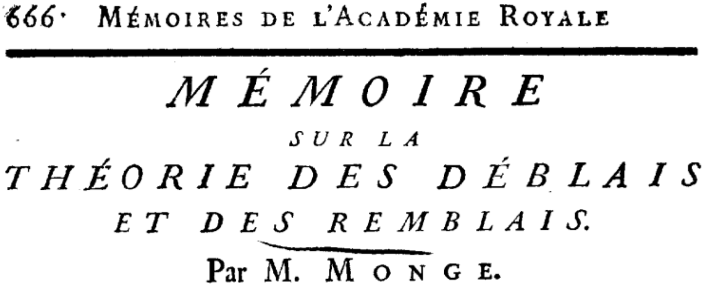
Journées MAS 2022

OPEN TOGETHER IMPACTFUL

What is optimal transport ?

A geometry of probability measures

The origins of optimal transport


666· MÉMOIRES DE L'ACADÉMIE ROYALE

MÉMOIRE

SUR LA

THÉORIE DES DÉBLAIS

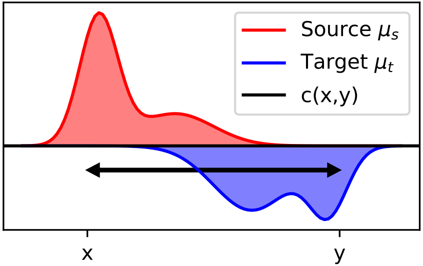
ET DES REMBLAIS.

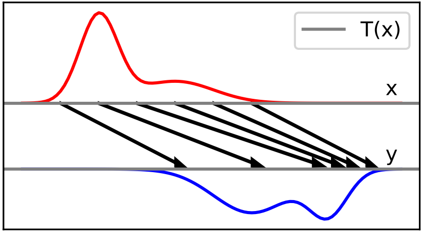
Par M. MONGE.

Problem [Monge, 1781]

- How to move dirt from one place (déblais) to another (remblais) while minimizing the effort ?
- Find a mapping T between the two distributions of mass (transport).
- Optimize with respect to a displacement cost $c(x, y)$ (optimal).

The origins of optimal transport

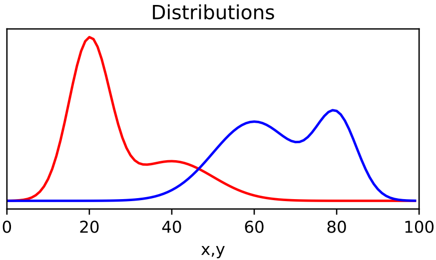


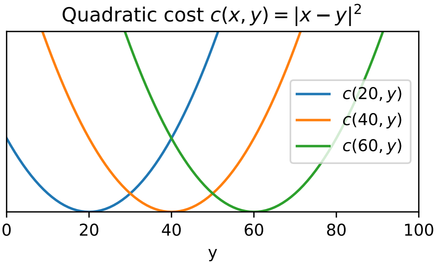


Problem [Monge, 1781]

- How to move dirt from one place (déblais) to another (remblais) while minimizing the effort ?
- Find a mapping T between the two distributions of mass (transport).
- Optimize with respect to a displacement cost $c(x, y)$ (optimal).

Optimal transport (Monge formulation)



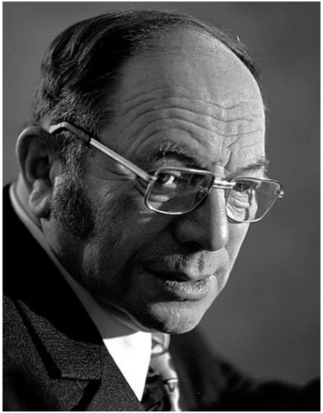


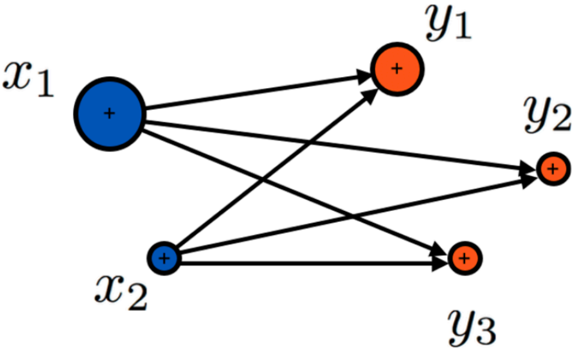
- Probability measures μ_s and μ_t on and a cost function $c : \Omega_s \times \Omega_t \rightarrow \mathbb{R}^+$.
- The Monge formulation [Monge, 1781] aims at finding a mapping $T : \Omega_s \rightarrow \Omega_t$

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

- mapping does not exist in the general case.
- [Brenier, 1991] proved existence and unicity of the Monge map for $c(x,y) = \|x-y\|^2$ and distributions with densities.

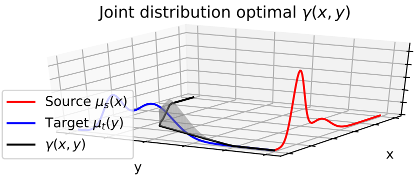
Kantorovich relaxation

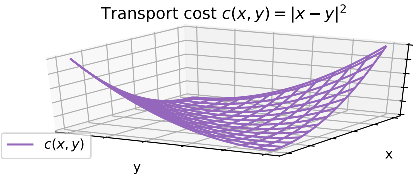




- Leonid Kantorovich (1912--1986), Economy nobelist in 1975
- Focus on where the mass goes, allow splitting [Kantorovich, 1942].
- Applications mainly for resource allocation problems

Optimal transport (Kantorovich formulation)





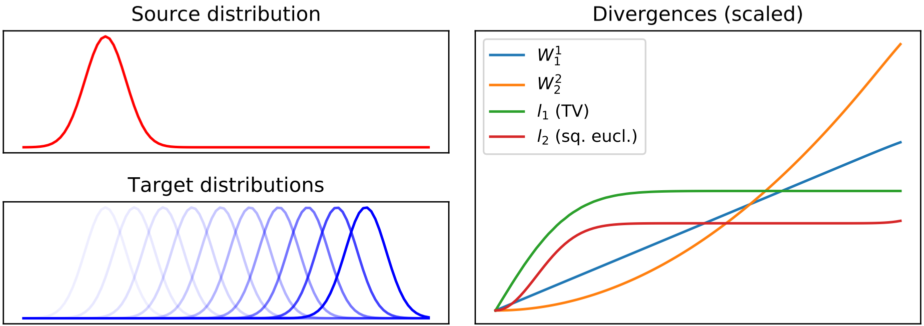
- The Kantorovich formulation [Kantorovich, 1942] seeks for a probabilistic coupling $\gamma \in \mathcal{P}(\Omega_s \times \Omega_t)$ between Ω_s and Ω_t:

$$\gamma_0 = \arg\min_{\gamma} \int_{\Omega_s \times \Omega_t} c(\mathbf{x}, \mathbf{y}) \gamma(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y}, \tag{2}$$

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

- γ is a joint probability measure with marginals μ_s and μ_t.
- Linear Program that always has a solution.

Wasserstein distance



Wasserstein distance

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

where $c(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|^p$

- Do not need the distribution to have overlapping support.
- Subgradients can be computed with the dual variables of the LP.
- can be made scalable using a dual form.

The 3 ways of optimal transport

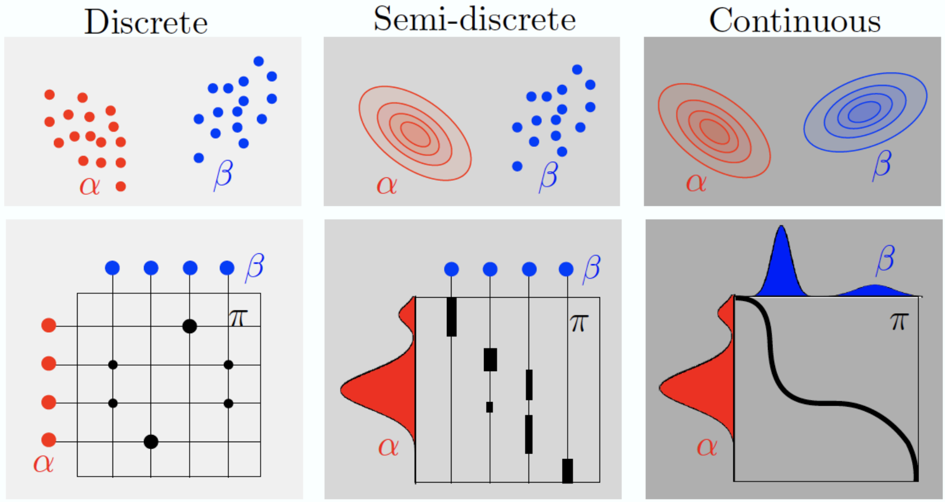
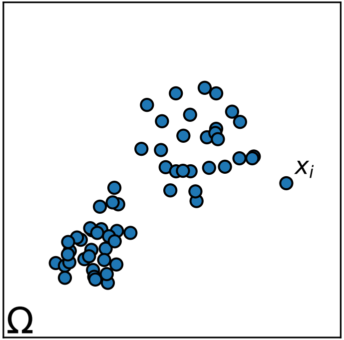


Image from Gabriel Peyré

Discrete distributions: Empirical vs Histogram

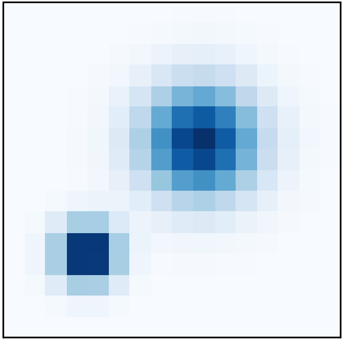
Discrete measure: $\mu = \sum_{i=1}^{n} a_i \delta_{\mathbf{x}_i}, \quad \mathbf{x}_i \in \Omega, \quad \sum_{i=1}^{n} a_i = 1$

Lagrangian (point clouds)



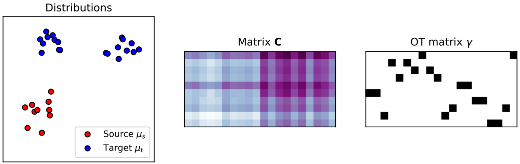
- Constant weight: $a_i = \frac{1}{n}$

Eulerian (histograms)



- Fixed positions $\mathbf{x}_i$ e.g. grid
- Convex polytope Σ_n (simplex): $\{(a_i)_i \geq 0; \sum_i a_i = 1\}$

Optimal transport with discrete distributions



OT Linear Program

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

$$\gamma_0 = \operatorname*{arg\,min}_{\gamma \in \mathbf{P}} \quad \left\{ \langle \gamma, \mathbf{C} \rangle_F = \sum_{i,j} \gamma_{i,j} c_{i,j} \right\}$$

where $\mathbf{C}$ is a cost matrix with $c_{i,j} = c(\mathbf{x}_i^s, \mathbf{x}_j^t)$ and the marginals constraints are

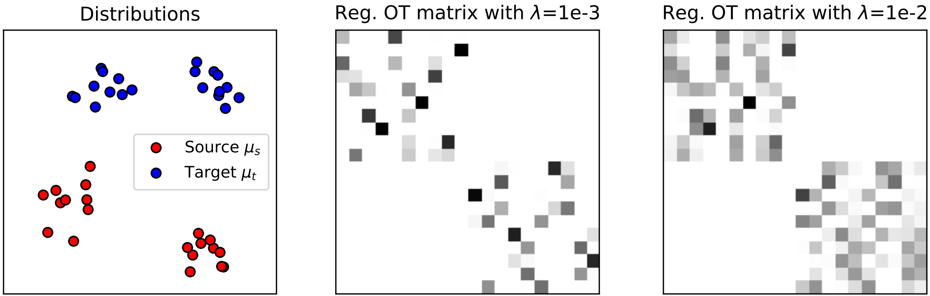
$$\mathbf{P} = \left\{ \gamma \in (\mathbb{R}^+)^{n_s \times n_t} | \, \gamma \mathbf{1}_{n_t} = \mathbf{a}, \gamma^T \mathbf{1}_{n_s} = \mathbf{b} \right\}$$

Linear program with $n_s n_t$ variables and $n_s + n_t$ constraints.

Optimal assignment

when $n_s = n_t$, and a_i and b_i are uniform, we have an optimal assignement problem and the solution is a 1-to-1 matching. γ is a permutation matrix.

Entropic regularized optimal transport

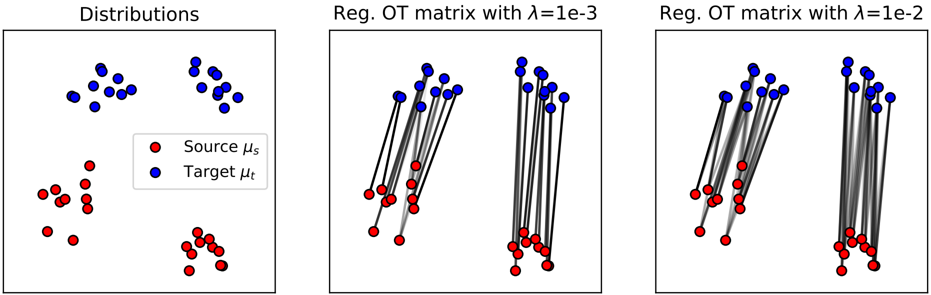


Entropic regularization [Cuturi, 2013]

$$\gamma_0^\lambda = \underset{\gamma \in \mathbf{P}}{\arg\min} \quad \langle \gamma, \mathbf{C} \rangle_F + \lambda \sum_{i,j} \gamma(i,j)(\log \gamma(i,j) - 1)$$

- Regularization with the negative entropy of γ.
- Looses sparsity, gains stability.
- Strictly convex optimization problem.
- Loss and OT matrix are differentiable.

Entropic regularized optimal transport



Entropic regularization [Cuturi, 2013]

$$\gamma_0^\lambda = \arg\min_{\gamma \in \mathbf{P}} \quad \langle \gamma, \mathbf{C} \rangle_F + \lambda \sum_{i,j} \gamma(i,j)(\log \gamma(i,j) - 1)$$

- Regularization with the negative entropy of γ.
- Looses sparsity, gains stability.
- Strictly convex optimization problem.
- Loss and OT matrix are differentiable.

Solving the entropy regularized problem

Lagrangian of the optimization problem

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

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

Entropy-regularized transport

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

$$\boldsymbol{\gamma}_0^{\lambda}=\mathrm{diag}(\mathbf{u})\exp(-\mathbf{C}/\lambda)\mathrm{diag}(\mathbf{v})$$

- Through the **Sinkhorn theorem** $\mathrm{diag}(\mathbf{u})$ and $\mathrm{diag}(\mathbf{v})$ exist and are unique.
- Relation with dual variables: $u_i=\exp(\alpha_i/\lambda),\quad v_j=\exp(\beta_j/\lambda)$.
- Can be solved by the **Sinkhorn-Knopp** algorithm.

Sinkhorn-Knopp algorithm

Algorithm 1 Sinkhorn-Knopp Algorithm (SK).

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

  $\mathbf{u}^{(0)} = \mathbf{1}, \mathbf{K} = \exp(-\mathbf{C}/\lambda)$
  for i in $1, \ldots, n_{it}$ **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 $\mathbf{T} = \text{diag}(\mathbf{u}^{(n_{it})})\mathbf{K}\text{diag}(\mathbf{v}^{(n_{it})})$

- The algorithm performs alternatively a scaling along the rows and columns of $\mathbf{K} = \exp(-\frac{\mathbf{C}}{\lambda})$ to match the desired marginals.
- Complexity $O(kn^2)$, where k iterations are required to reach convergence
- Fast implementation in parallel, GPU friendly

General case for entropic OT: autodifferentiation

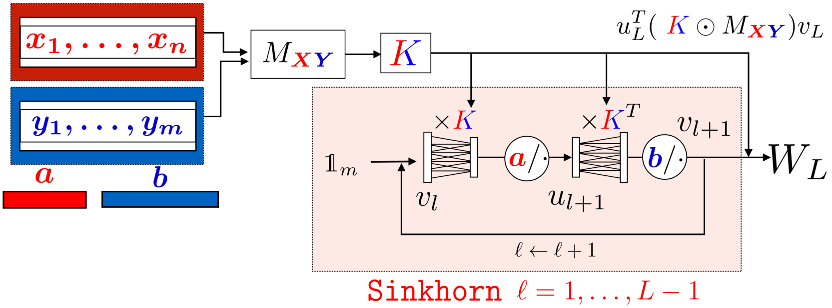


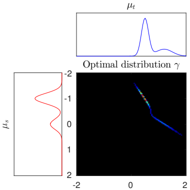
Image from Marco Cuturi

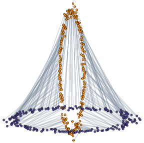
Sinkhorn Autodiff [Genevay et al., 2017]

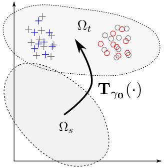
- Computing gradients through implicit function theorem can be costly [Luise et al., 2018].
- Each iteration of the Sinkhorn algorithm is differentiable.
- Modern neural network toolboxes can perform autodiff (PyTorch, Tensorflow).
- Fast but needs log-stabilization for numerical stability.

Table of contents

Some aspects of optimal transport in machine learning







Divergence between histograms

- Use the ground metric to encode complex relations between the bins.
- Loss for multilabel classifier [Frogner et al., 2015]
- Document-Topic representation [Zhao et al., 2020]

Divergence between empirical distributions

- Objective function for generative models [Arjovsky et al., 2017].
- Missing data imputation [Muzellec et al., 2020].
- Learn with train/test mismatch [Courty et al., 2016, **?**, Rakotomamonjy et al., 2020]

Transporting with optimal transport

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

Wasserstein distance as a multilabel loss

Siberian husky

Eskimo dog

Leveraging output space structure [Frogner et al., 2015]

- Classes of a multiclass (multi-label) problem have structure
- Takes into account semantic of classes in the output distribution probability
- Error in ``similar" class is less penalized than to dissimilar one
- can be represented as a Wasserstein distance between true label and output of a model. ground metric represent the distance between classes

$$\min_{f_\theta} \frac{1}{n} \sum_{i=1}^{n} W(f_\theta(x_i), y_i)$$

Wasserstein loss for generative modelling

Generative modelling as a matching distribution problem

- Learn a model $f_\theta(\cdot)$ that maps random vector to target space
- Distribution of the model output is targeted to be similar to the learning samples
- Similarity as Wasserstein sense [Arjovsky et al., 2017, Deshpande et al., 2018, Nguyen et al., 2020]

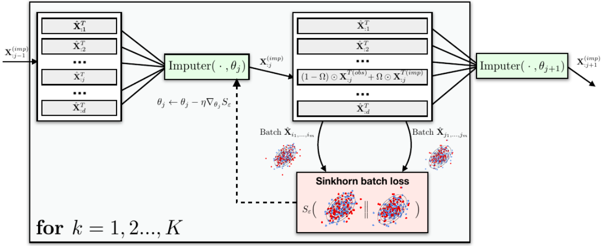
$$\min_{f_\theta} W\left(\{f_\theta(z_i)\}_{i=1}^{K}, \{x_j\}_{j=1}^{K}\right)$$

$\{z_i\}$ some random vectors, $\{x_j\}$ some samples from the target distribution

Missing Data Imputation

Impute missing data under matching distribution loss [Muzellec et al., 2020], [Kirchmeyer et al. 2021]

- Impute missing data so that distributions of imputed data and full ones match
- Sinkhorn divergence as a discrepancy measure



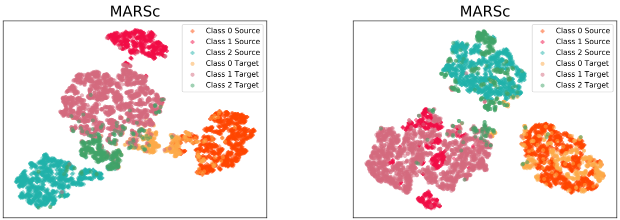
Learning with mismatch in train and test sets

Domain Adaptation

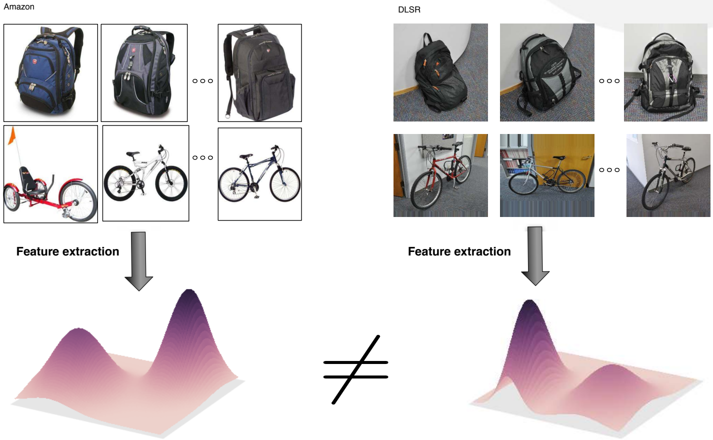
- several ML applications break the hypothesis that $P_{train} = P_{test}$
- Goal of domain adaptation : learn a representation mapping $g(\cdot)$ and a classifier $h(\cdot)$ so that representations of train/test data in the latent space matches
- Learning problem [Shen et al., 2018, Courty et al., 2016, Rakotomamonjy et al., 2020]

$$\min_{h,g} \frac{1}{n} \sum_i L(h(g(x_i^S)), y_i) + \lambda W(P(h(X^S), h(X^T))$$

- Representation when learning only on source and then after adaptation :



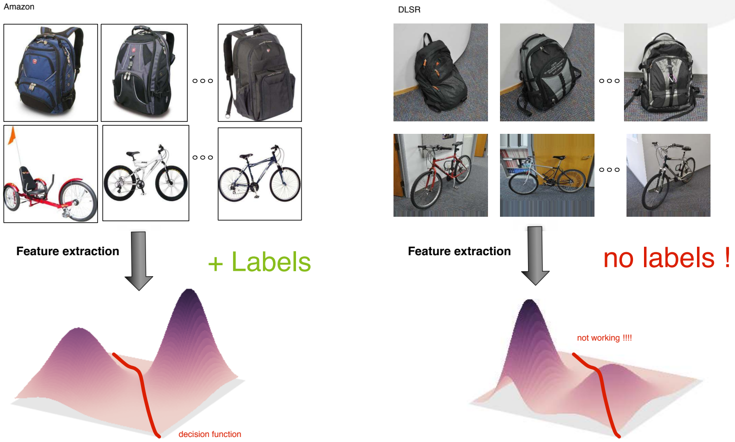
Domain Adaptation problem



Context

- Classification problem with data coming from different sources (domains).
- usual DA context : marginal distributions are different but related.

Unsupervised domain adaptation problem



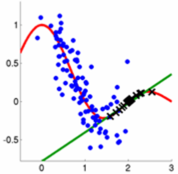
Problems

- Labels only available in the **source domain**, and classification is conducted in the **target domain**.
- Classifier trained on the source domain data performs badly in the target domain

Domain adaptation short state of the art

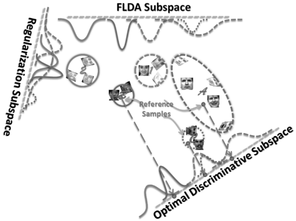
Reweighting schemes

- Distribution change between domains.
- Reweigh samples to compensate this change [Sugiyama et al., 2008].



Subspace methods

- Data is invariant in a common latent subspace.
- Minimization of a divergence between the projected domains
  [Si et al., 2010, Ganin et al., 2016, Tzeng et al., 2017].
- Use additional label information
  [Long et al., 2014, Long and Wang, 2015].



Domain-invariant Unsupervised domain adaptation

Classical approaches

- Learn representation mapping $g(\cdot)$ that matches source and target and a classifier $h(\cdot)$

$$\min_{h,g} \frac{1}{n_s}\sum_{i=1}^{n_s} L(y_i^s, h(g(x_i^s))) + \lambda D(p_S^g, p_T^g) + \Omega(h, g)$$

- $D(\cdot,\cdot)$ is a distance between distributions. it can be Jenssen-Shannon approximation, Maximum Mean discrepancy or Optimal Transport or any Integral Probability Metric.

Why this approach may break?

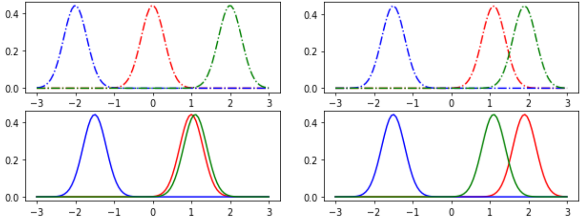
- Aligning marginals may not match class-conditionals
- when label proportions in source and target domains are different

Illustration of domain-invariance failure

mismatch when aligning just marginals

- top/bottom panels : source/target
- left/right figure : before/after optimization

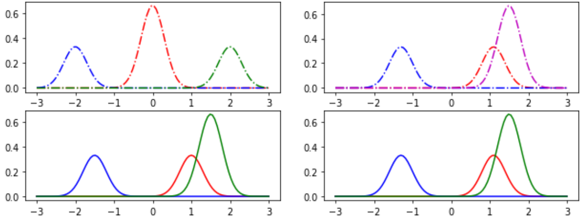
$\Longrightarrow$ we can have a mismatch in class-conditionals



mismatch induced by label shift

- top/bottom panels : source/target
- left/right figure : before/after optimization

$\Longrightarrow$ source classes have to be mixed



Domain Adaptation : Generalized Target Shift

General DA situation

- label shift : $p_S(y=k) \neq p_T(y=k)$
- class-conditional shift : $p_s(z|y=k) \neq p_T(z|y=k)$, z being the latent space representation

Our contribution

- proposes a learning model that matches class-conditionals without labels in target
- uses OT as a distance between distributions. it helps providing guarantees.

Generalized Target Shift

Goal

- a labeled source dataset $\{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ with $y_i^s \in \{1 \ldots C\}$
- unlabeled examples from the target domain $\{x_i^t\}_{i=1}^{n_t}$ with all $x_i \in \mathcal{X}$, sampled i.i.d from their respective distributions.
- We learn a representation through a representation mapping $g : \mathcal{X} \rightarrow \mathcal{Z}$ and a classifier h

Assumptions

- when g is learned only on source domains $P_s(z|y=k) \neq P_t(z|y=k)$

Notations

- f is the true labelling fonction
- marginal distributions of the source and target domains in the latent space as $p_S^g(z)$ and $p_T^g(z)$. Class-conditionals are noted $p_U^j \triangleq p_U(z|y=j)$
- Label proportions $p_U^{y=j} \triangleq p_U(y=j)$ with $U \in \{S, T\}$.

Theoretical results for Generalized Target Shift

Target risk bound

Assuming that any function $h \in \mathcal{H}$ is K-Lipschitz and g is a continuous function then for every function h and g, we have

$$\varepsilon_T(h \circ g, f) \leq \varepsilon_S(h \circ g, f) + 2K \cdot WD_1(p_S^g, p_T^g) + \left[1 + \sup_{k,z} w(z) S_k(z)\right] \varepsilon_S(h^\star \circ g, f) + \varepsilon_T^z(f_S^g, f_T^g)$$

Intuitions

- First term : expected risk in source domain
- Wasserstein distance between marginals in latent space
- product of label proportion ratio $w(z)$ and class-conditionals ratio $S_k(z)$
- optimal classifier $h^\star$ expected risk in the source
- Last term : how good the true labelling function in source and target are similar in the latent space.

Learning problem

Optimizing the bound

- apply the bound with label reweighted source so that no label shift occur $\implies w(z) = 1$
- estimate label proportions in target p_T^y
- minimize the empirical risk in source
- minimize distance between marginals and class-conditionals

Resulting learning problem

$$\min_{g,h} \frac{1}{n} \sum_{i=1}^{n_s} w^{\dagger}(x_i^s) L(y_i^s, h(g(x_i^s))) + \lambda WD_1(p_{\tilde{S}}^g, p_T^g) + \Omega(h, g) \tag{4}$$

where the importance weight $w^{\dagger}(x_i^s) = \frac{p_T^{y=y_i}}{p_S^{y=y_i}}$ allows to simulate sampling from $p_{\tilde{S}}^g$ given p_S^g, and the discrepancy between marginals is the Wasserstein distance

Solving the learning problem

Algorithm

- train g and h through SGD and backprop
- for scalability, we use the Kantorovich dual for the WD

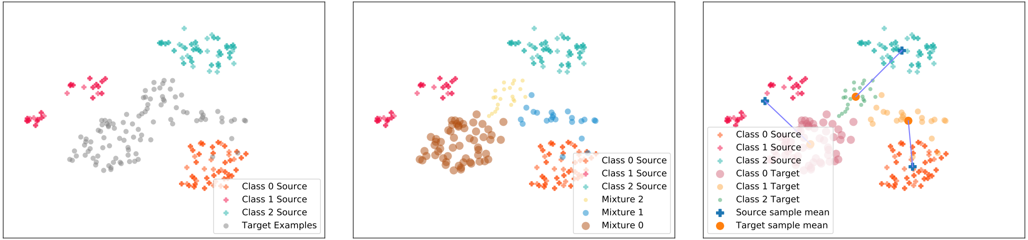
$$WD_1(\tilde{p}_s^g, p_t^g) = \sup_{\|v\|_L \leq 1} \mathbf{E}_{z \sim p_S^g} w^\dagger(z) v(z) - \mathbf{E}_{z \sim p_T^g} v(z). \tag{5}$$

- we still need to estimate p_t^Y and ensure that class-conditionals match.

Match and reweight strategy

- Cluster target domain data
- Match clusters with source class-conditionals
- identify target class-conditionals
- estimate target label proportion

Match and Reweight illustration



Steps

- **left** we have the source and target samples in the latent Space
- **middle** Target samples are clustered. Classes are assigned arbitrarily.
- **right** Optimal assignment of $p_S(z|y=k)$ to $p_T(z|y=k)$ mean vectors to, so that label propagation relates source and target classes.

Match and Reweight Guarantee

- label propagation is based on optimal assignment
- geometry of source and target classes should follow a specific pattern.
- When are we ensured to have correct match of classes ?

Proposition

Denote as $\nu = \frac{1}{C}\sum_{j=1}^{C} \delta_{p_S^j}$ and $\mu = \frac{1}{C}\sum_{j=1}^{C} \delta_{p_T^j}$ the empirical measures built from class-conditionals probabilities in source and target domains.

Choose $\mathcal{D}$ a distance over probability distributions

if we have the following assumption, known as the $\mathcal{D}$-cyclical monotonicity relation, holds for any permutation σ

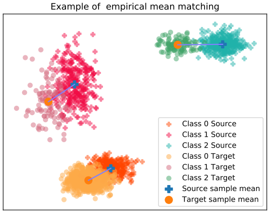
$$\sum_j \mathcal{D}(p_S^j, p_T^j) \leq \sum_j \mathcal{D}(p_S^j, p_T^{\sigma(j)})$$

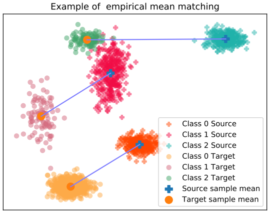
then then solving the optimal transport problem between ν and μ using $\mathcal{D}$ as the ground cost matches correctly class-conditional probabilities.

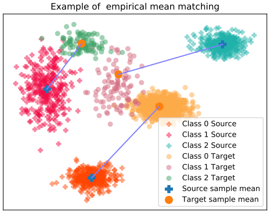
Sufficient condition

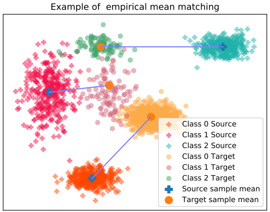
$$\forall j, k \quad \mathcal{D}(p_S^j, p_T^j) \leq \mathcal{D}(p_S^j, p_T^k)$$

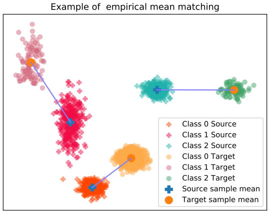
Illustration of correct matching

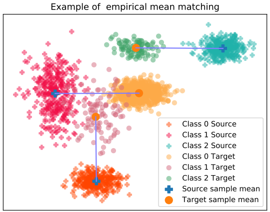












From matching marginals to matching class-conditionals

Question

we minimize distance between marginals. what happen to the class-conditionals?

Proposition

Denote as γ the optimal coupling plan for distributions $\nu = \frac{1}{C}\sum_{j=1}^{C} \delta_{p_S^j}$ and $\mu = \frac{1}{C}\sum_{j=1}^{C} \delta_{p_T^j}$.

Assume that the classes are ordered so that we have $\gamma = \frac{1}{C}$ diag(1) and that cyclical monotonicity holds.

Then $\gamma' =$ diag$(\mathbf{a})$ is also optimal for the transportation problem with marginals $\nu' = \sum_{j=1}^{C} a_j \delta_{p_S^j}$ and $\mu' = \sum_{j=1}^{C} a_j \delta_{p_T^j}$, with $a_j > 0, \forall j$.

- In addition, if the Wasserstein distance between ν' and μ' is 0, it implies that the distance between class-conditionals are all 0.

Hence

Optimal assignment does not change with weights. Achieving 0 distance between reweighted source and target marginals $\Longrightarrow$ 0 distance between class-conditionals.

Experimental setting

Baselines

- Source only
- Domain adversarial NN (DANN) : no adaptation to label shift

Competitors

- Different ways of estimating p_T^y for use in

$$WD_1(\tilde{p}_s^g, p_t^g) = \sup_{\|v\|_L \leq 1} \mathbf{E}_{z \sim p_S^g} w(z) v(z) - \mathbf{E}_{z \sim p_T^g} v(z).$$

- $WD_\beta = 1/(1+\beta)$ with β user-defined constant, and should depend on the label shift [Wu et al., 2019]
- IW-WD : $w(z) = \frac{p_T}{p_S}$ with p_T estimated assuming class-conditionals are equal [Combes et al., 2020].

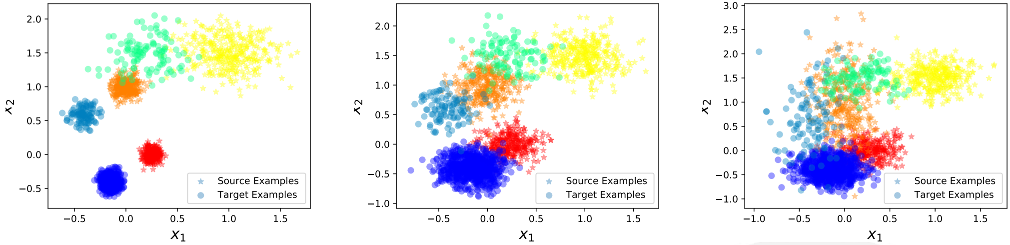
Architecture

- Feature extractor $g(\cdot)$ and classifier $h(\cdot)$ are same for all methods
- SGD and WD + gradient penalty for WD

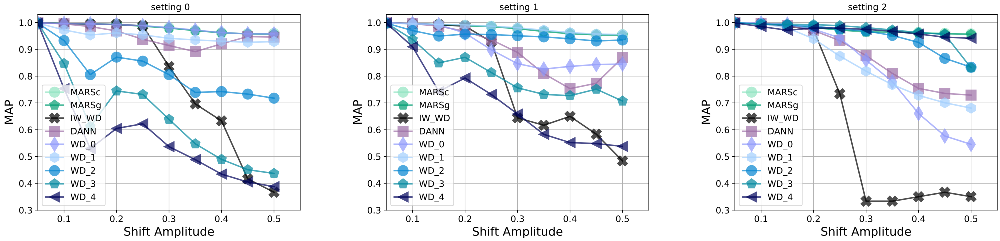
Experiments on toy data

Toy

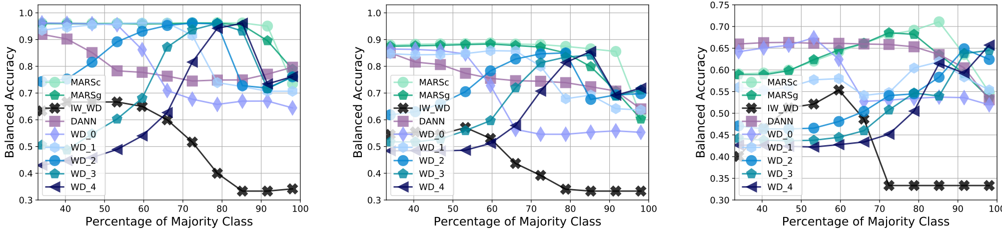
- Source : 3 Gaussians -- Target : same Gaussians with translated mean
- different label proportion between source and target
- different distances from sources (breaking cyclical monotonicity)



Examples and Results



With respects to the problem hardness



Computer Vision Tasks

Setting

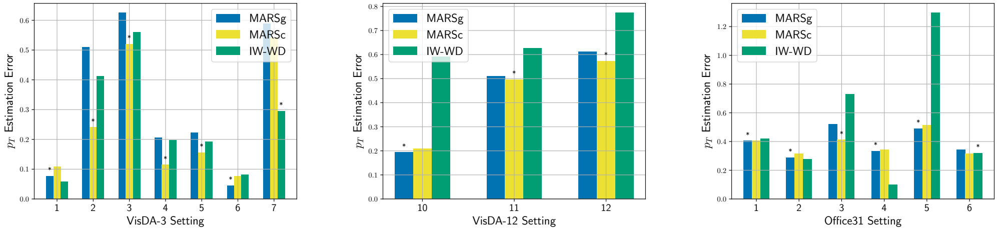
- Classical CV datasets
- Performance averaged over 10 random seed + statistical test

Setting	Source	DANN	WD$_{\beta=0}$	WD$_{\beta=1}$	WD$_{\beta=2}$	WD$_{\beta=3}$	WD$_{\beta=4}$	IW-WD	MARSg	MARSc
MNIST-USPS 10 modes										
Balanced	76.9±3.7	79.7±3.5	93.7±0.7	74.3±4.3	51.3±4.0	76.6±3.3	71.9±5.7	95.3±0.4	**95.6±0.7**	**95.6±1.0**
Mid	80.4±3.1	78.7±3.0	94.3±0.7	75.4±3.4	55.6±4.3	79.0±3.1	72.3±4.2	**95.6±0.5**	89.7±2.3	90.4±2.6
High	78.1±4.9	81.8±4.0	**93.9±1.1**	87.4±1.7	83.8±5.2	85.7±2.5	83.6±3.0	**94.1±1.0**	88.3±1.5	89.7±2.3
USPS-MNIST 10 modes										
Balanced	77.0±2.6	80.5±2.2	73.4±2.8	66.7±2.9	49.9±2.8	55.8±2.9	52.1±3.5	80.5±2.2	**84.6±1.7**	**85.5±2.1**
Mid	**79.5±2.8**	**78.9±1.8**	75.8±1.6	63.3±2.3	53.2±2.8	47.2±2.4	48.3±2.9	**78.4±3.5**	**79.7±3.6**	**78.5±2.5**
High	**78.5±2.4**	**77.8±2.0**	**76.1±2.7**	63.0±3.3	57.6±4.8	51.2±4.4	49.3±3.3	71.5±4.7	75.6±1.8	**77.1±2.4**
MNIST-MNISTM 10 modes										
Setting 1	58.3±1.3	**61.2±1.1**	57.4±1.7	50.2±4.4	47.0±2.0	57.9±1.1	60.0±1.3	**63.1±3.1**	58.1±2.3	56.6±4.6
Setting 2	60.0±1.1	61.1±1.0	58.1±1.4	53.4±3.5	48.6±2.4	59.7±0.7	58.1±0.8	65.0±3.5	57.7±2.3	55.7±2.1
Setting 3	58.1±1.2	**60.4±1.4**	57.7±1.2	47.7±4.9	42.2±7.3	57.1±1.0	53.5±1.1	52.5±14.8	53.7±7.2	53.7±3.3
VisdDA 12 modes										
setting 1	41.9±1.5	52.8±2.1	45.8±4.3	44.2±3.0	35.5±4.6	41.0±3.0	37.6±3.4	50.4±2.3	53.3±0.9	**55.1±1.6**
setting 2	41.8±1.5	50.8±1.6	45.7±8.9	40.5±4.8	36.2±5.0	36.1±4.6	31.9±5.7	48.6±1.8	53.1±1.6	**55.3±1.6**
setting 3	40.6±4.3	49.2±1.3	47.1±1.6	42.1±3.0	36.3±4.4	37.3±3.5	35.0±5.4	46.6±1.3	50.8±1.6	**52.1±1.2**
Office 31										
A - D	73.7±1.4	74.3±1.8	**77.2±0.7**	65.1±2.0	62.7±2.6	71.5±1.2	63.9±1.1	75.7±1.6	76.1±0.9	**78.2±1.3**
D - W	83.7±1.1	81.9±1.5	82.6±0.6	83.5±0.8	82.8±0.7	80.1±0.5	**87.1±0.9**	78.9±1.5	**86.3±0.6**	86.2±0.8
W - A	54.1±0.9	52.2±1.0	48.9±0.4	56.8±0.4	53.0±0.5	58.8±0.4	54.9±0.5	52.2±0.7	**60.7±0.8**	55.2±0.8
W - D	92.8±0.9	87.8±1.4	95.1±0.3	93.1±0.5	87.6±0.9	94.7±0.6	91.2±0.6	**97.0±0.9**	95.1±0.8	93.8±0.6
D - A	52.5±0.9	48.1±1.2	49.8±0.4	48.8±0.5	50.1±0.4	50.3±0.7	50.8±0.5	41.4±1.8	**54.7±0.9**	**55.0±0.9**
A - W	67.5±1.5	70.2±1.0	67.1±0.6	60.6±2.1	52.9±1.4	64.0±1.3	59.7±0.8	68.8±1.6	**73.1±1.5**	**71.9±1.2**
#Wins (/34)	7	9	5	0	1	0	2	9	12	21
Aver. Rank	4.16	4.73	5.32	6.97	8.38	6.59	7.57	4.95	3.38	2.95

Ablation study

Label proportion estimation

Estimating label proportion in target domains is key for : label propagation and matching marginals



Findings

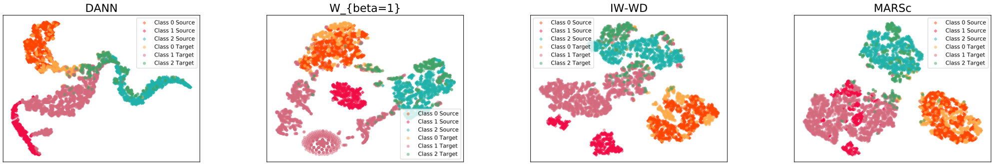
- Our approach using agglomerative clustering seems to work better than other approaches (Gaussian mixture models and using the confusion matrix as in Des Combes et al. [Combes et al., 2020]
- The method proposed by Des Combes assume that class-conditionals are equal (which is not true)

Ablation study

Low-dimensional representation in the latent space (VisDA-3)

Before Matching

After Matching



Conclusion

Python code available on GitHub

`https://github.com/rflamary/POT`

Summary

- a model that handles Conditional and label shift in DA
- guarantees under some geometrical assumptions in the latent space
- needs label proportion

Paper and code

- `https://arxiv.org/abs/2006.08161`
- `https://github.com/arakotom/mars_domain_adaptation`

References I

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

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

Combes, R. T. d., Zhao, H., Wang, Y.-X., and Gordon, G. (2020).
Domain adaptation with conditional distribution matching and generalized label shift.
arXiv preprint arXiv:2003.04475.

Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. (2016).
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.

References II

Deshpande, I., Zhang, Z., and Schwing, A. G. (2018).
Generative modeling using the sliced wasserstein distance.
In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

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

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.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. (2016).
Domain-adversarial training of neural networks.
Journal of Machine Learning Research, 17:1--35.

Genevay, A., Peyré, G., and Cuturi, M. (2017).
Sinkhorn-autodiff: Tractable wasserstein learning of generative models.
arXiv preprint arXiv:1706.00292.

References III

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

Long, M. and Wang, J. (2015).
Learning transferable features with deep adaptation networks.
CoRR, abs/1502.02791.

Long, M., Wang, J., Ding, G., Sun, J., and Yu, P. (2014).
Transfer joint matching for unsupervised domain adaptation.
In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1410--1417.

Luise, G., Rudi, A., Pontil, M., and Ciliberto, C. (2018).
Differential properties of sinkhorn approximation for learning with wasserstein distance.
In Advances in Neural Information Processing Systems, pages 5864--5874.

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

References IV

Muzellec, B., Josse, J., Boyer, C., and Cuturi, M. (2020).
Missing data imputation using optimal transport.
In International Conference on Machine Learning, pages 7130--7140. PMLR.

Nguyen, K., Ho, N., Pham, T., and Bui, H. (2020).
Distributional sliced-wasserstein and applications to generative modeling.
arXiv preprint arXiv:2002.07367.

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

Rakotomamonjy, A., Flamary, R., Gasso, G., Alaya, M. Z., Berar, M., and Courty, N. (2020).
Optimal transport for conditional domain matching and label shift.
CoRR, abs/2006.08161.

Shen, J., Qu, Y., Zhang, W., and Yu, Y. (2018).
Wasserstein distance guided representation learning for domain adaptation.
In Proceedings of the AAAI Conference on Artificial Intelligence, volume 32.

References V

Si, S., Tao, D., and Geng, B. (2010).
Bregman divergence-based regularization for transfer subspace learning.
IEEE Transactions on Knowledge and Data Engineering, 22(7):929--942.

Sugiyama, M., Nakajima, S., Kashima, H., Buenau, P., and Kawanabe, M. (2008).
Direct importance estimation with model selection and its application to covariate shift adaptation.
In Neural Information Processing Systems (NIPS).

Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. (2017).
Adversarial discriminative domain adaptation.
CoRR, abs/1702.05464.

Wu, Y., Winston, E., Kaushik, D., and Lipton, Z. (2019).
Domain adaptation with asymmetrically-relaxed distribution alignment.
In Chaudhuri, K. and Salakhutdinov, R., editors, Proceedings of the 36th International Conference on Machine Learning, volume 97 of Proceedings of Machine Learning Research, pages 6872--6881, Long Beach, California, USA. PMLR.

Zhao, H., Phung, D., Huynh, V., Le, T., and Buntine, W. (2020).
Neural topic model via optimal transport.