Learning under distribution shift using optimal transport

Gilles Gasso

March 15, 2022

joint work with M. Alaya, L. Chapel, N. Courty, R. Flamary, R. Herault, M. Kechaou, A. Rakotomamonjy

litis

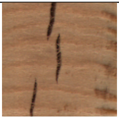
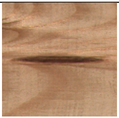
INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES ROUEN

Table of contents

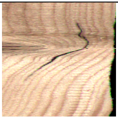
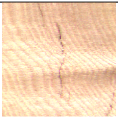
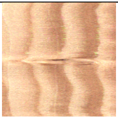
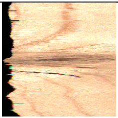
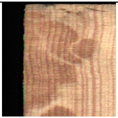
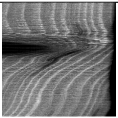
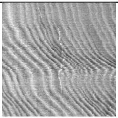
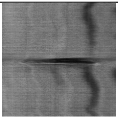
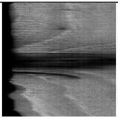
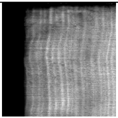
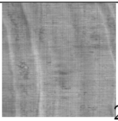
Illustration

Task: classification of defaults affecting wood's pieces

Source domain (Pine): 2 classes with only RGB images

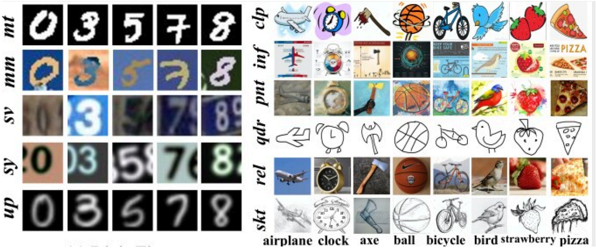
Label 1		Label 2	
			

Target domain (Spruce): 3 classes, multi-view (RGB and scanner) images

Label 1		Label 2		Label 3	
					
					

Illustration

Task: image classification



Dmain adaptation?

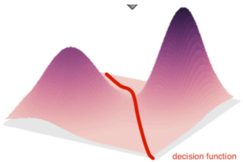
- Differences in instances $\not\Longrightarrow$ difference in the predictions
- Transfer knowledge from previous domain to a new domain to overcome the differences
- Domains are somehow related

Domain adaptation problem

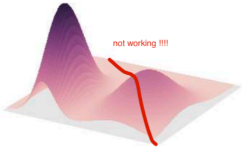
Our context

- Source Domain: data are from the joint distribution $P_s(\mathbf{x}^s, y^s)$

  Target domain: data follow the distribution $P_t(\mathbf{x}^t, y^t)$
- P_s and P_t are different but *sufficiently* related



Source Domain



Target Domain

Goal

Leverage on labeled source data to learn a classifier effective for unlabeled target data

Use Optimal Transport to measure the domain relatedness

Summary of optimal transport

The origins of optimal transport

666· MÉMOIRES DE L'ACADÉMIE ROYALE

MÉMOIRE

SUR LA

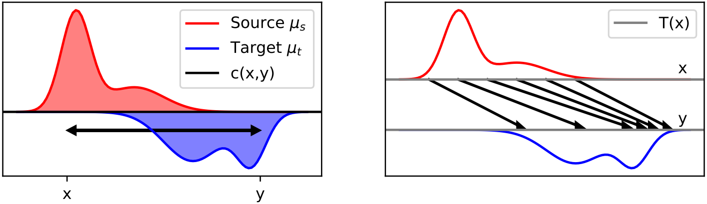
THÉORIE DES DÉBLAIS

ET DES REMBLAIS.

Par M. MONGE.

Problem [Monge, 1781]

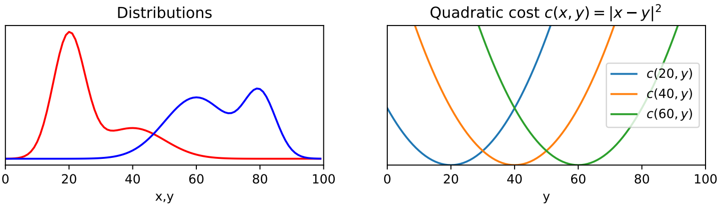
- Move dirt from one place to another while minimizing the effort
- Find a mapping T between the two distributions of mass
- Optimize with respect to a given displacement cost $c(\boldsymbol{x}, \boldsymbol{z})$



Problem [Monge, 1781]

- Move dirt from one place to another while minimizing the effort
- Find a mapping T between the two distributions of mass
- Optimize with respect to a given displacement cost $c(\boldsymbol{x}, \boldsymbol{z})$

Optimal transport: Monge formulation

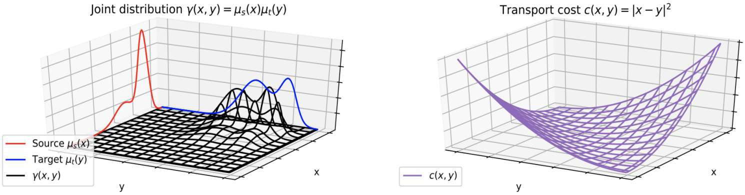


- Probability measures μ_s on $\mathcal{X}_s$ and μ_t on $\mathcal{X}_t$ and a cost function $c : \mathcal{X}_s \times \mathcal{X}_t \rightarrow \mathbb{R}^+$
- The [Monge, 1781] formulation seeks a mapping $T : \mathcal{X}_s \rightarrow \mathcal{X}_t$

$$\inf_{T\#\mu_s=\mu_t} \quad \int_{\mathcal{X}_s} c(\mathbf{x}, T(\mathbf{x}))\mu_s(\mathbf{x})d\mathbf{x}$$

- Non-convex problem, mapping does not exist in the general case
- Brenier [1991] proved existence and unicity of the Monge map for $c(\mathbf{x}, \mathbf{z}) = \|\mathbf{x} - \mathbf{z}\|^2$ and distributions with densities

Optimal transport - Kantorovitch relaxation



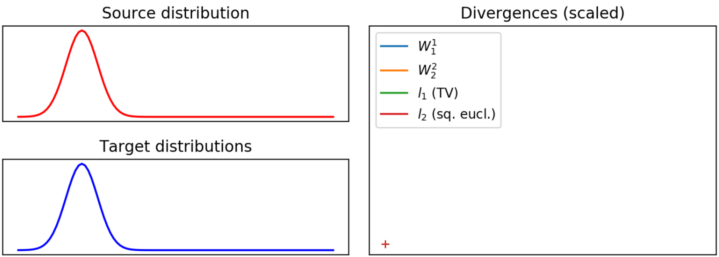
- The Kantorovitch formulation solves for the joint coupling

$$\hat{\gamma} = \operatorname{argmin}_{\gamma} \int_{\mathcal{X}_s \times \mathcal{X}_t} c(\boldsymbol{x}^s, \boldsymbol{x}^t)\gamma(\boldsymbol{x}^s, \boldsymbol{x}^t) d\boldsymbol{x}^s d\boldsymbol{x}^t,$$

$$\text{s.t. } \gamma \in \mathcal{U} = \{\gamma \geq 0 \,|\, \int_{\mathcal{X}_\mathbf{t}} \gamma(\boldsymbol{x}^\mathrm{s}, \boldsymbol{x}^\mathrm{t}) \mathrm{d}\boldsymbol{x}^\mathrm{t} = \mu_\mathrm{s}, \int_{\mathcal{X}_\mathbf{s}} \gamma(\boldsymbol{x}^\mathrm{s}, \boldsymbol{x}^\mathrm{t}) \mathrm{d}\boldsymbol{x}^\mathrm{s} = \mu_\mathrm{t}\}$$

- γ: joint probability measure with marginals μ_s and μ_t

Wasserstein distance



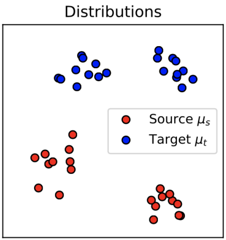
Wasserstein distance

$$W_p^p(\mu_s, \mu_t) = \min_{\gamma \in \mathcal{U}} \int_{\mathcal{X}_s \times \mathcal{X}_t} c(\mathbf{x}^s, \mathbf{x}^t)\gamma(\mathbf{x}^s, \mathbf{x}^t) d\mathbf{x}^s d\mathbf{x}^t$$

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

- Do not need the distribution to have overlapping supports
- Similar definition holds for discrete distributions (histograms, empirical).

The discrete distribution case



Source distribution $\mu_s = \sum_{i=1}^{n_s} a_i \delta_{x_i^s}, \ \sum_i a_i = 1$

Target one $\mu_t = \sum_{j=1}^{n_t} b_j \delta_{x_j^t}, \ \sum_j b_j = 1$

Problem

Measure the distance between μ_s and μ_t

- Find a joint probabilistic coupling γ

$$\min_{\gamma \in \mathcal{U}(\mu_s, \mu_t)} \langle C, \gamma \rangle_F$$

$C \in \mathbb{R}^{n_s \times n_t}$ is the transportation cost matrix with entries $c(\boldsymbol{x}_i^s, \boldsymbol{x}_j^t)$

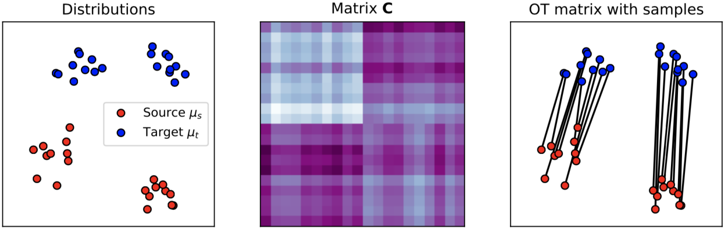
- $\mathcal{U}(\mu_s, \mu_t) = \{\gamma \in \mathbb{R}_+^{n_s \times n_t} | \gamma 1_{n_t} = \mu_s, \gamma^\top 1_{n_s} = \mu_t\}$

Discrete Optimal transport

$$\min_{\gamma \in \mathcal{U}(\mu_s, \mu_t)} \langle C, \gamma \rangle_F$$

with $\mathcal{U}(\mu_s, \mu_t) = \{\gamma \in \mathbb{R}_+^{n_s \times n_t} | \gamma 1_{n_t} = \mu_s, \gamma^\top 1_{n_s} = \mu_t\}$

- Linear programming problem with solution in $O(n^3 logn)$

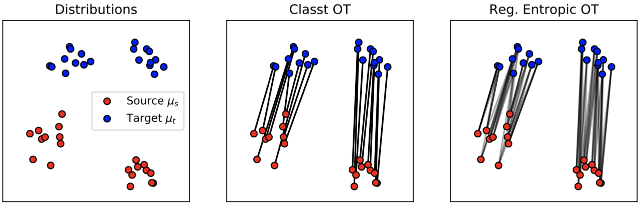


OT with discrete distributions

Regularized OT

$$\min_{\gamma \in \mathcal{U}(\mu_{\mathbf{s}}, \mu_{\mathbf{t}})} \langle C, \gamma \rangle_F + \lambda \Omega(\gamma)$$

- Generally use of convex regularization $\Omega(\gamma)$
  - Entropy regularization that leads to Sinkhorn iterations
  - Quadratic, Group-lasso $\cdots$
- Better computation speed or enforce prior knowledge
  - OT DA uses group-lasso to map source samples with the same labels onto the same subset of target instances



OT in ML applications

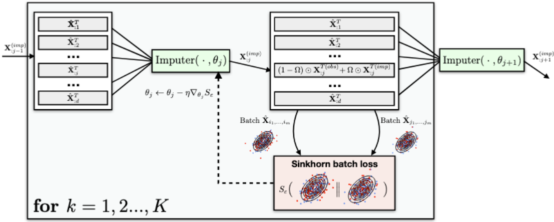
Generative modeling as a problem of distribution matching

- Learn a model f_θ that maps a random vector ξ to target space
- Distribution of the model output should be similar to the one of the learning source samples
- Similarity as Wasserstein distance sense [Arjovsky et al., 2017]

$$min_{f_\theta} \; W\left(\{\boldsymbol{x}_i^s\}_{i=1}^{n_s}, \{f_\theta(\xi_j)\}_{j=1}^{n_t}\right)$$

Impute missing data

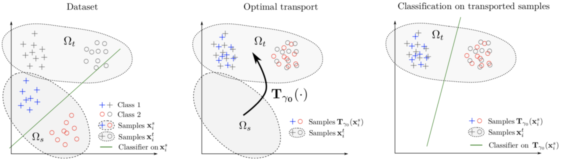
- Impute missing data so that to match distributions of imputed data and the full ones [Muzellec et al., 2020]
- Sinkhorn divergence is used to measure similarity



Domain adaptation

- Several ML applications do not fulfill the assumption $P_{train} = P_{test}$
- Common objective: learn sample representation mapping function $g(\cdot)$ and the prediction model $h(\cdot)$ so that the learned features of train/test data match in the latent space
- Learning problem

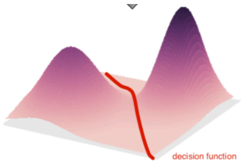
$$\min_{h,g} \frac{1}{n_s} \sum_{i=1}^{n_s} L\left(h(g(\boldsymbol{x}_i^s)), y_i^s\right) + \lambda\, W\left(P_{train}\left(g(\boldsymbol{x}^s)\right), P_{test}\left(g(\boldsymbol{x}^t)\right)\right)$$



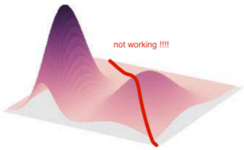
Optimal transport and domain adaptation

Recall the context

- Source Domain: data are from the joint distribution $P_s(\boldsymbol{x}^s, y^s)$
- Target domain: data follow the distribution $P_t(\boldsymbol{x}^t, y^t)$
- Classification task: $\mathcal{Y}_s = \{1, \cdots, K\}$
- P_s and P_t are different but *sufficiently* related



Source Domain



Target Domain

Domain adaptation: common settings

Notations

Source data are labeled $\mathcal{D}_s = \{(\mathbf{x}_i^s, y_i^s) \in \mathcal{X}_s \times \mathcal{Y}_s\}_{i=1}^{n_s}$

Target samples are unlabeled $\mathcal{D}_t = \{\mathbf{x}_j^t \in \mathcal{X}_t\}_{j=1}^{n_t}$

	Joint dis.	Marginal dis.	Conditional dis.	Label dis.
Source	$P_s(\mathbf{x}, y)$	$P_s(\mathbf{x})$	$P_s(y/\mathbf{x})$	$P_s(y)$
Target	$P_t(\mathbf{x}, y)$	$P_t(\mathbf{x})$	$P_t(y/\mathbf{x})$	$P_t(y)$

Common assumptions

- Same instance and label spaces $\mathcal{X}_s = \mathcal{X}_t$ and $\mathcal{Y}_s = \mathcal{Y}_t$
- Joint distributions are drifted $P_s(\mathbf{x}, y) \neq P_t(\mathbf{x}, y)$
  - Covariate shift: $P_s(\mathbf{x}) \neq P_t(\mathbf{x})$ but $P_s(y/\mathbf{x}) \simeq P_t(y/\mathbf{x})$
  - Label shift: $P_s(y) \neq P_t(y)$ but $P_s(\mathbf{x}/y) \simeq P_t(\mathbf{x}/y)$

Domain adaptation: the goal

- Let $D(\cdot, \cdot)$ be a distance between distributions
- Assume $L(\cdot, \cdot) : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ is a loss function

Learning problem

- Learn a function $f(\cdot) : \mathcal{X} \to \mathcal{Y}$ that minimizes the risk on source domain while aligning the source and target distributions

$$\min_f R_s(f) + D(P_s, P_t)$$

- $R_s(f) = \mathbb{E}_{(x,y)\sim P_s} L(y, f(x))$ is the expected risk on source domain

Expected outcome

Such learned f adapts well to target domain

Empirical DA risk minimization

In practice

Model $f(\cdot) = h \circ g(\cdot)$ consists of

- a representation learning function $g(\cdot) : \mathcal{X} \rightarrow \mathcal{Z}$
- and a classifier $h(\cdot) : \mathcal{Z} \rightarrow \mathcal{Y}$

Regularized empirical risk minimization

$$\min_{h,g} \frac{1}{n_s} \sum_{i=1}^{n_s} L\left(h(g(\mathbf{x}_i^s)), y_i^s\right) + \lambda D(P_s^g, P_t^g) + \Omega(h, g)$$

- The distributions are aligned in the representation space $\mathcal{Z}$
- Ω is a regularization term
- Problem usually solved using stochastic gradient descent

Why does it work?

Bounding the target risk [Ben-David et al., 2010]

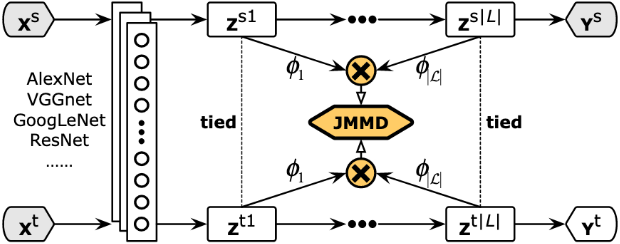
$$R_t(f) \leq R_s(f) + D(P_s(\boldsymbol{x}), P_t(\boldsymbol{x})) + \alpha$$

- What we should care about: measure of distribution shift $D(P_s(\boldsymbol{x}), P_t(\boldsymbol{x}))$
- What we expect: domain relatedness measured by $\alpha = \inf_f R_s(f) + R_t(f)$

Most DA strategies

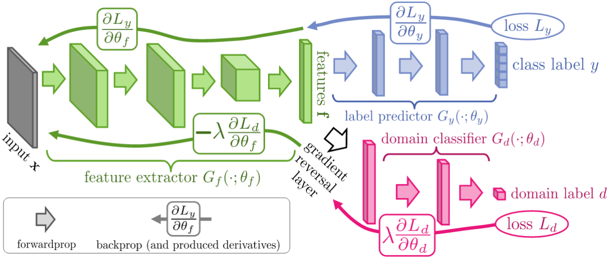
- Choose f with good properties (to get α minimal)
- Minimize distribution discrepancy

Joint adaptation network [Long et al., 2017]



- Jointly align feature distributions across layers
- Based on kernel Maximum Mean Discrepancy [Gretton et al., 2012] between layer activation distributions $D(P_s(\boldsymbol{x}), P_t(\boldsymbol{x})) \equiv \|m_z(P_s) - m_z(P_t)\|^2_{\mathcal{H}}$

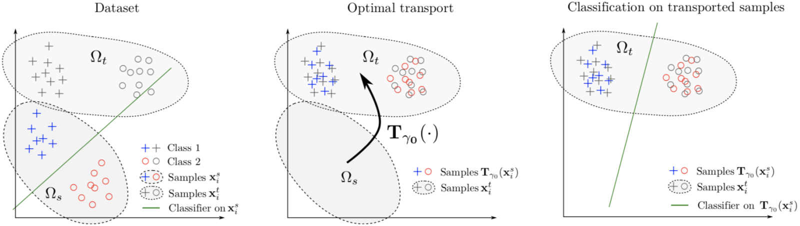
Domain adversarial network [Ganin et al., 2016]



- Mapping source and target instances onto a domain-invariant latent subspace
- Ensure good prediction on source domain
- Approach issued from the target risk bound

Some domain-invariant adaptation methods

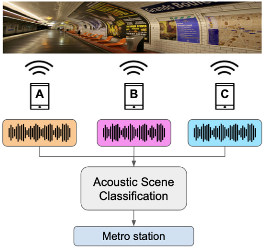
Optimal transport domain adaptation [Courty et al., 2016]



- Estimate a push-forward operator T between source and target distributions
- Map source samples onto target domain
- Learn a classification function

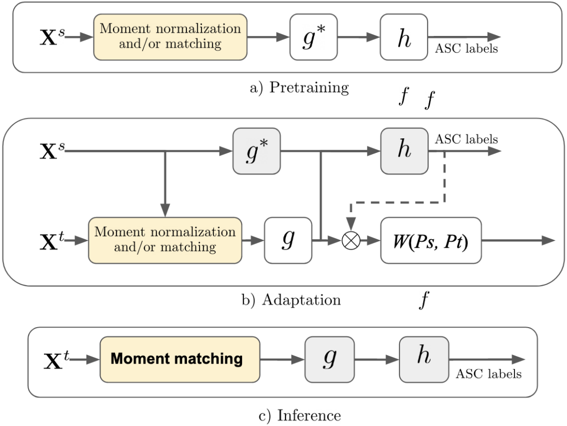
Learning problem

- ASC: classify an audio recording into a class (metro, bus...)
- Issue: different recording devices may impede performances
- Goal: adapt the ASC system to account for data recorded with different devices

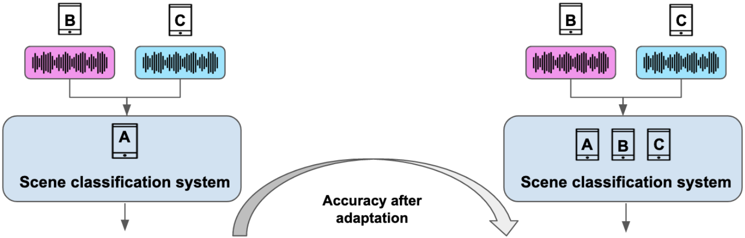


- Source domain: device A
- Target one: devices B and C

Proposed approach [Olvera et al., 2022]



Classification accuracy



True label \ Predicted label	airport	bus	metro	metro_station	park	public_square	shopping_mall	street_pedestrian	street_traffic	tram
airport	0.06	0.00	0.19	0.69	0.00	0.00	0.00	0.00	0.06	0.00
bus	0.00	0.22	0.28	0.39	0.03	0.00	0.00	0.00	0.08	0.00
metro	0.00	0.00	0.36	0.61	0.00	0.00	0.00	0.00	0.03	0.00
metro_station	0.00	0.06	0.22	0.67	0.00	0.00	0.00	0.00	0.06	0.00
park	0.00	0.06	0.47	0.17	0.06	0.00	0.00	0.00	0.22	0.03
public_square	0.03	0.06	0.33	0.25	0.03	0.00	0.00	0.00	0.28	0.03
shopping_mall	0.00	0.00	0.08	0.83	0.00	0.00	0.00	0.03	0.06	0.00
street_pedestrian	0.00	0.00	0.39	0.47	0.00	0.00	0.03	0.00	0.11	0.00
street_traffic	0.00	0.00	0.03	0.19	0.08	0.00	0.00	0.03	0.67	0.00
tram	0.00	0.07	0.36	0.54	0.00	0.00	0.00	0.00	0.04	0.00

True label \ Predicted label	airport	bus	metro	metro_station	park	public_square	shopping_mall	street_pedestrian	street_traffic	tram
airport	0.61	0.00	0.00	0.06	0.00	0.06	0.22	0.00	0.06	0.00
bus	0.00	0.58	0.19	0.03	0.00	0.06	0.00	0.00	0.00	0.14
metro	0.00	0.00	0.69	0.11	0.03	0.03	0.00	0.03	0.03	0.08
metro_station	0.08	0.14	0.08	0.47	0.00	0.00	0.03	0.00	0.14	0.06
park	0.00	0.00	0.03	0.00	0.69	0.06	0.00	0.00	0.11	0.11
public_square	0.00	0.03	0.00	0.00	0.17	0.42	0.00	0.03	0.22	0.14
shopping_mall	0.06	0.00	0.00	0.00	0.00	0.06	0.83	0.06	0.00	0.00
street_pedestrian	0.11	0.00	0.00	0.03	0.08	0.17	0.11	0.33	0.00	0.17
street_traffic	0.00	0.03	0.00	0.03	0.06	0.03	0.00	0.06	0.81	0.00
tram	0.03	0.06	0.17	0.00	0.00	0.03	0.00	0.06	0.11	0.56

13.6% → 60.1%

(Source domain accuracy: 59.3 - 64.7%)

Limitations of the classical DA methods

Causes of failure

- They learn a mapping function g such that the conditional distributions are preserved (covariate shift)

$$P_s(y/g(\boldsymbol{x})) \simeq P_t(y/g(\boldsymbol{x}))$$

- This amounts to align the marginal distributions $P_s^g \simeq P_t^g$

But what if

- the label distributions change across domains $P_s(y) \neq P_t(y)$?

  $\Rightarrow$ Aligning marginals may not match class-conditionals

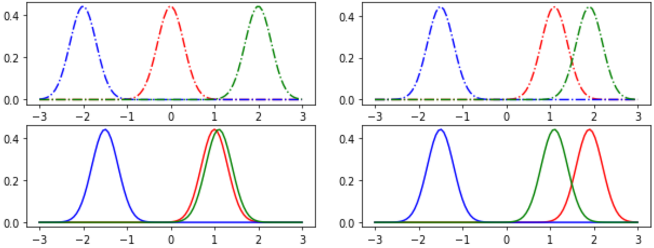
- the input spaces are not similar $\mathcal{X}_s \neq \mathcal{X}_t$?

Optimal Transport for Conditional Domain Matching and Label Shift

Illustration of domain-invariance breaking

- top/bottom panels: source/target domains
- left/right: before/after adaptation

Mismatch when aligning marginals

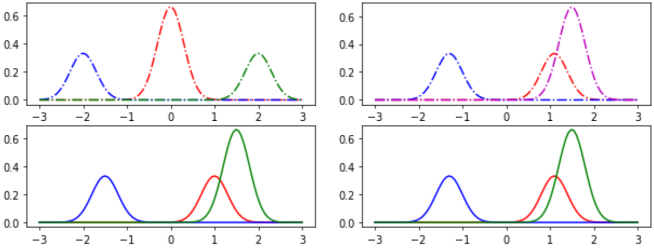


$\Rightarrow$ Class conditionals are mismatched

Illustration of domain-invariance breaking

- top/bottom panels: source/target domains
- left/right: before/after adaptation

Mismatch induced by label shift $P_s(y) \neq P_t(y)$



$\Rightarrow$ source domain classes are mixed

Generalized domain adaptation

Considered setting

- label shift: $P_s(y = k) \neq P_t(y = k)$
- class conditional shift: $P_s(z/y = k) \neq P_t(z/y = k)$
- $z = g(\boldsymbol{x})$ is the latent space representation

Contributions

- learning framework that matches class-conditionals *without labels in target domain*
- learn OT mapping between source and target class-conditionals
- estimate the class-proportion in target by modeling target samples by a mixture of models (cluster assumption)

Formulation

Goal

- given a labeled source dataset $\mathcal{D}_s = \{(\boldsymbol{x}_i^s, y_i^s)\}_{i=1}^{n_s}$ and unlabeled target one $\mathcal{D}_t = \{\boldsymbol{x}_j^t\}_{j=1}^{n_t}$
- learn a latent representation mapping $g : \mathcal{X} \to \mathcal{Z}$
- and a classifier $h : \mathcal{Z} \to \mathcal{Y}$ that performs well on target samples

Approach

- re-weighting scheme of source samples to deal with the label shift
- mapping class-conditionals i.e. $P_s(g(\boldsymbol{x})/y = k) \simeq P_t(g(\boldsymbol{x})/y = k)$

Target risk bound

Assuming that $P_s(y = k) > 0$, $P_s(\mathbf{z}/y = k) > 0$ for all class k and h is K-Lipschitz and g is continuous, we have

$$R_t(f) \leq \underbrace{R_s(f)}_{\text{source risk}} + 2K \underbrace{W_1(P_s^g, P_t^g)}_{\text{alignment}} + \underbrace{\mathbb{E}_{\mathbf{z}\sim P_t^g}|f_t^g(\mathbf{z}) - f_s^g(\mathbf{z})|}_{\text{can't be estimated}}$$
$$+ \left[1 + \sup_{k,\mathbf{z}} \underbrace{\omega(\mathbf{z})S_k(\mathbf{z})}_{\text{reweighting}}\right] R_t(h^\star \circ g)$$

- $\omega(\mathbf{z}) = \dfrac{P_t(y = k)}{P_s(y = k)}$, if $\mathbf{z}$ is of class k, is the label proportion ratio
- $S_k(\mathbf{z}) = \dfrac{P_t(\mathbf{z}/y = k)}{P_s(\mathbf{z}/y = k)}$, class-conditional ratio
- and $\sup_{k,\mathbf{z}} \omega(\mathbf{z})S_k(\mathbf{z}) \geq 1$ (the lower bound is attained when there is no shift)

Generalized DA via a match and re-weight strategy

Derived learning problem

- Principle: to avoid label shift, we match the target marginal P_t^g with the **re-weighted source one** $\tilde{P}_s^g = \sum_{k=1}^K \underbrace{P_t(y=k)}_{\text{unknown}} P_s(\mathbf{z}/y=k)$
- Hence, we solve the weighted problem

$$\min_{h,g} \frac{1}{n_s} \sum_{i=1}^{n_s} \omega_i L\left(h(g(\mathbf{x}_i^s)), y_i^s\right) + \lambda\, W_1(\tilde{P}_s^g, P_t^g) + \Omega(h, g)$$

with $\omega_i = \dfrac{P_t(y=y_i)}{P_s(y=y_i)}$

- **Notice:** the procedure requires to estimate the unknown target class proportion

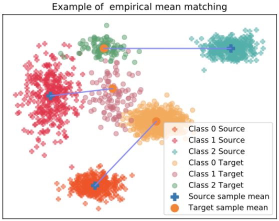
The principle

- Given the target sample representations $\{\mathbf{z}_j^t = g(\mathbf{x}_j^t)\}_j$, learn the target marginal distribution as a mixture with K modes

$$P_t^g(\mathbf{z}) = \sum_{k=1}^{K} \alpha_k p_k^t(\mathbf{z}), \quad \alpha_k > 0, \sum_k \alpha_k = 1$$

- Use OT to find the permutation σ that aligns the source class-conditionals $\{P_s(\mathbf{z}/y = k)\}_{k=1}^K$ with the target ones $\{p_k\}_{k=1}^K$

Target class proportion $P_t(y = k) = \sigma(\alpha_k)$

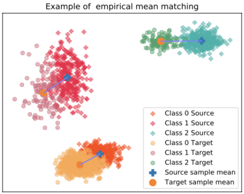


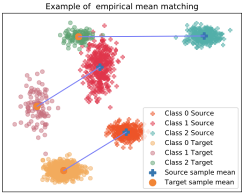
Assumptions for correct class-conditional matching

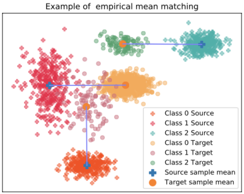
Assumptions

- cluster assumption on the source domain
- Cyclical monotonicity between source and target class-conditionals

Examples of correct/incorrect geometrical arrangment







Experimental evaluation

Baselines

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

Competitors that account for label shift

- WD_β: $\omega = 1/(1+\beta)$ with β user-defined constant
- IW-WD: $\omega = \frac{P_t}{P_s}$ estimated assuming class-conditionals are equal

Datasets

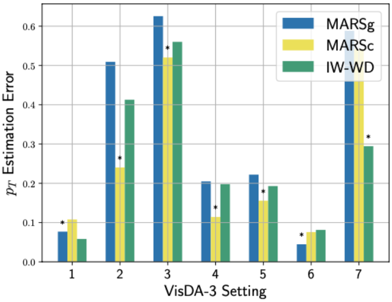
- Evaluation on computer vision tasks (Digits, VisDA)

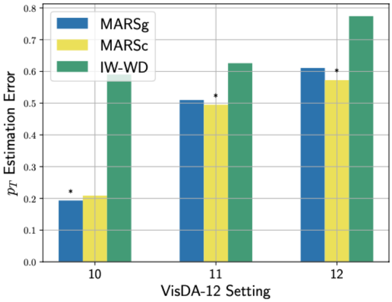
Quantitative results

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 3 modes					
setting 1	79.3±4.3	78.9±9.1	91.8±0.7	73.8±2.0	61.7±2.2	65.6±2.7	58.6±2.6	**94.1±0.6**	92.5±1.2	92.1±1.8
setting 4	80.2±5.3	75.5±9.3	72.8±1.2	86.9±7.5	86.8±1.2	80.2±6.9	75.7±2.0	85.9±5.7	87.7±3.0	**91.3±4.8**
setting 2	81.5±3.5	68.5±14.7	68.8±1.3	84.5±1.2	**93.2±0.4**	73.7±14.2	60.7±0.9	78.7±10.8	84.0±4.3	**91.8±3.4**
setting 3	78.4±3.2	59.0±15.9	64.1±1.9	79.2±0.8	77.1±10.3	90.0±0.5	**94.4±0.3**	78.0±9.3	75.7±4.1	73.9±13.2
setting 5	83.5±3.5	80.9±14.5	63.9±0.6	73.7±7.3	50.9±1.1	76.5±6.7	59.3±1.0	**90.4±3.6**	**89.0±0.9**	**89.0±3.5**
setting 6	80.9±4.2	54.8±19.8	45.3±2.4	63.7±5.1	67.1±6.1	42.9±11	62.2±1.4	**94.4±1.0**	**93.7±0.4**	**93.9±1.0**
setting 7	79.2±3.7	42.9±2.5	57.5±1.5	55.4±2.0	50.2±4.3	43.7±8.3	62.5±0.8	**88.5±4.9**	78.6±3.2	**82.3±7.5**
					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**

Balanced accuracy. The best performing method is indicated in bold

Estimation of target label proportion



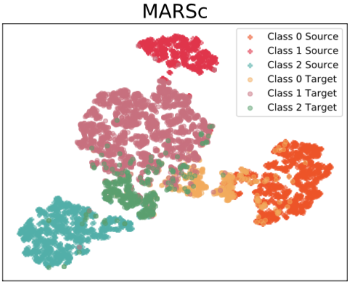


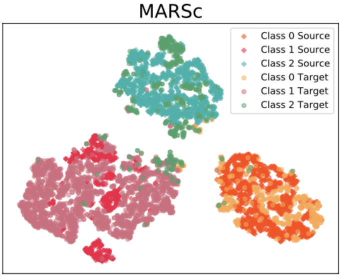
- best performance is correlated to better label proportion estimation

Ablation studies

Embedding visualisation

- left: before adaptation, right: after





- almost correct matching of class conditionals

Partial OT and domain adaptation

Moving beyond class-conditional matching

What if

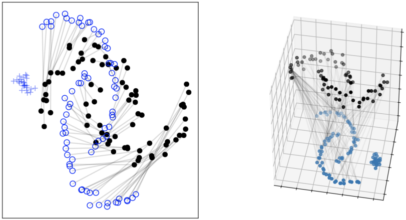
- the label distributions change $P_s(y) \neq P_t(y)$? ✓
- the input spaces are not similar $\mathcal{X}_s \neq \mathcal{X}_t$?
- the label spaces are different $\mathcal{Y}_s \neq \mathcal{Y}_t$? $\longrightarrow$ Open set DA

Our approach

- Optimal transport as a measure of distribution discrepancy
- Open set DA: detect unknown target classes and map known class instances
- Different input spaces: use Gromov-Wasserstein optimal transport

Adaptation when input spaces differ

- Labeled source data $\mathcal{D}_s = \{(\mathbf{x}_i^s, y_i) \in \mathcal{X}_s \times \mathcal{Y}_s\}_{i=1}^{n_s}$
- Target samples are unlabeled $\mathcal{D}_t = \{\mathbf{x}_j^t \in \mathcal{X}_t\}_{j=1}^{n_t}$
- Our setting
  - Label spaces differ $\mathcal{Y}_s \neq \mathcal{Y}_t$
  - Input instances belong to different spaces $\mathcal{X}_s \neq \mathcal{X}_t$
  - Ex: multi-view data with some views absent across domains



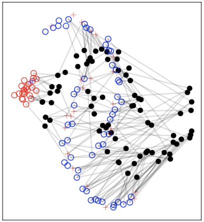
Goal

Find a mapping accounting for different input spaces and target shift

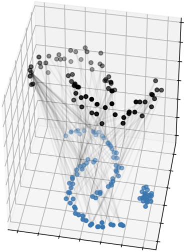
Issues

- Classical OT deals with distributions defined over the same metric space

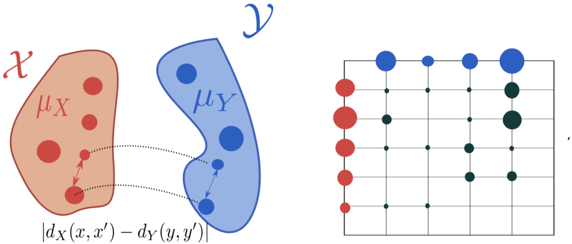
$$\min_{\gamma \in \mathcal{U}(\mu_{\mathbf{s}}, \mu_{\mathbf{t}})} \langle C, \gamma \rangle_F$$



- How to deal with $\mathcal{X}_s \neq \mathcal{X}_t$?
  - Use a Gromov-Wasserstein optimal transport
- How to deal with $\mathcal{Y}_s \neq \mathcal{Y}_t$?
  - Optimize over the marginals
  - or resort to partial transport of probability mass



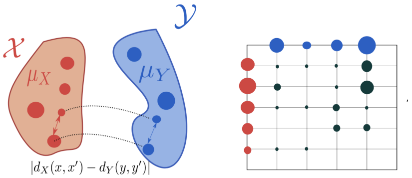
Gromov-Wasserstein optimal transport



Inspired from Gabriel Peyré

- Measure distance between distributions with no common ground space
- Based on pairwise distances in each space
- Invariant to rotation and translation of the samples

Discrete Gromov-Wasserstein optimal transport



Inspired from Gabriel Peyré

$$\min_{\gamma \in \mathcal{U}(\mu_s,\mu_t)} J(\gamma) = \sum_{i,k=1}^{n_s} \sum_{j,\ell=1}^{n_t} (C^s_{ik} - C^t_{j\ell})^2 \gamma_{ij}\gamma_{k\ell}$$

with $C^s_{ik} = d_{\mathcal{X}_s}(x^s_i, x^s_k)$ and $C^t_{j\ell} = d_{\mathcal{X}_t}(x^t_j, x^t_\ell)$ ground distances

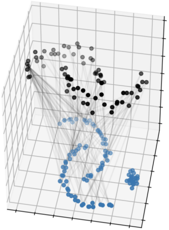
- Non-convex problem
- Practical computation considers Gromov-Wasserstein optimal transport with entropic regularization

Partial Gromov-Wasserstein optimal transport

- How to deal with $\mathcal{Y}_s \neq \mathcal{Y}_t$?
  - Avoid transferring all probability mass from source to target
  - $\rightarrow$ Transport only a fraction of probability mass

Partial GW OT

$$\min_{\gamma \in \mathcal{U}^{\mathrm{u}}(\mu_{\mathrm{s}}, \mu_{\mathrm{t}})} J(\gamma) = \sum_{i,k=1}^{n_s} \sum_{j,\ell=1}^{n_t} (C^s_{ik} - C^t_{j\ell})^2 \gamma_{ij} \gamma_{k\ell}$$



the set of coupling matrices is defined now as

$$\mathcal{U}^{\mathrm{u}}(\mu_{\mathrm{s}}, \mu_{\mathrm{t}}) = \{\gamma \in \mathbb{R}_+^{n_s \times n_t} \mid \gamma 1 \leq \mu_s, \gamma^\top 1 \leq \mu_t, 1_n^\top \gamma 1 = \beta\}$$

$0 \leq \beta \leq \min(\|\mu_s\|_1, \|\mu_t\|_1)$: fraction of probability mass to be transported

Solving Partial Gromov-Wasserstein problem

Frank-Wolfe iterations

1. Compute a linear minimization oracle over the set $\mathcal{U}^{\mathrm{u}}(\mu_{\mathrm{s}}, \mu_{\mathrm{t}})$

$$\tilde{\gamma} \leftarrow \mathrm{argmin}_{\gamma \in \mathcal{U}^{\mathrm{u}}(\mu_{\mathrm{s}}, \mu_{\mathrm{t}})} \langle \nabla_{\gamma} J(\gamma^{(k)}), \gamma \rangle$$

2. Find step size

$$\eta^{(k)} \leftarrow \mathrm{argmax}_{\eta \in [0,1]} J((1-\eta)\gamma^{(k)} + \eta \tilde{\gamma})$$

3. Update the solution

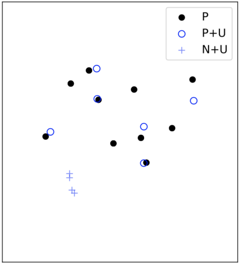
$$\gamma^{(k+1)} \leftarrow (1-\eta^{(k)})\gamma^{(k)} + \eta^{(k)} \tilde{\gamma}$$

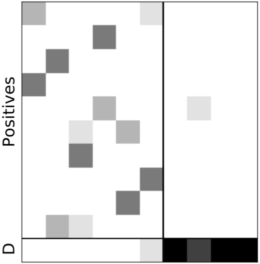
- The most difficult part is solving step 1
- Step 1 is a partial (Wasserstein) optimal transport

Strategy for partial Wasserstein OT

$$(PW) \qquad \min_\gamma \langle \mathrm{M}, \gamma \rangle,$$
$$\text{s.t.} \qquad \gamma \in \mathcal{U}(\mu_s, \mu_t) = \{\gamma \in \mathbb{R}_+^{n_s \times n_t} \,|\, \gamma 1 \leq \mu_s, \gamma^\top 1 \leq \mu_t, 1_n^\top \gamma 1 = \beta\}$$

- Key element
  - Turn the partial inequality constraints into equality ones
  - Introduce dummy points that will receive the excess of mass





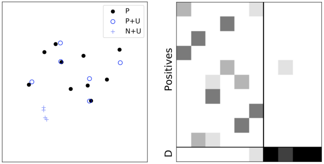
Strategy for partial Wasserstein OT

Equivalent problem

$$\min_{\tilde{\gamma}\in\mathcal{U}(\tilde{\mu}_s,\tilde{\mu}_t)}\langle\tilde{M},\tilde{\gamma}\rangle,\quad \mathcal{U}(\tilde{\mu}_s,\tilde{\mu}_t)=\{\tilde{\gamma}\geq 0,\tilde{\gamma}1=\tilde{\mu}_s,\tilde{\gamma}^\top 1=\tilde{\mu}_t\}$$

$$\text{with } \tilde{M}=\begin{bmatrix} M & e^\top \\ e & \infty \end{bmatrix},\quad \tilde{\mu}_s=\begin{bmatrix} \mu_s \\ \hline \|\mu_t\|_1-\beta \end{bmatrix},\quad \tilde{\mu}_t=\begin{bmatrix} \mu_t \\ \hline \|\mu_s\|_1-\beta \end{bmatrix}$$

- Interpretation
  - $e=\xi 1$ is a vector such that $\xi\geq\frac{1}{2}\max_{i,j}M_{ij}$
  - Marginals $\tilde{\mu}_s$ and $\tilde{\mu}_t$ have the same mass $\|\mu_s\|_1+\|\mu_t\|_1-\beta$
- The new problem is a linear program solved with network flow solver
- Provably it provides the solution to (PW) problem



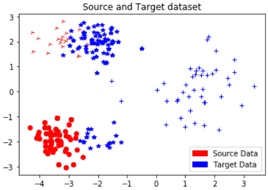
Application to PU learning

Positive Unlabeled learning

- P $= \{\boldsymbol{x}_i\}_{i=1}^{n_p}$ set of positive samples with $x \sim p(\boldsymbol{x}|y=1)$
- U $= \{\boldsymbol{x}_i^u\}_{i=1}^{n_u}$ unlabeled set with $x^u \sim p(x) = \beta p(\boldsymbol{x}|y=1) + (1-\beta)p(\boldsymbol{x}|y=-1)$
- $\beta = p(y=1)$ true proportion of positives

Link with open set DA

- Identifying the target unseen class amounts to PU learning
  - P represents source samples
  - U corresponds to target samples with the unknown class (negatives)



Partial GW on Caltech data - same input space

Data set	$\beta(\%)$	PU	PUSB	P-W	P-GW
Original Mnist	10	89.3	82.8	**99.1**	96.3
Colored Mnist	10	87.0	80.0	86.5	**96.5**
Surf C$\to$Surf C	10	89.3	**89.4**	82.3	86.4
Surf C$\to$Surf A	10	**87.7**	85.6	82.2	87.2
Surf C$\to$Surf W	10	84.4	80.5	80.8	**89.0**
Surf C$\to$Surf D	10	82.0	83.2	80.2	**94.2**
Decaf C$\to$Decaf C	10	93.9	**94.8**	83.8	85.8
Decaf C$\to$Decaf A	10	80.5	82.2	83.8	**88.6**
Decaf C$\to$Decaf W	10	82.4	83.8	87.0	**90.8**
Decaf C$\to$Decaf D	10	82.6	83.6	84.8	**95.2**

- Datasets: Caltech 256 (C), Amazon (A), Webcam (W), DSLR (D)
- Methods: Vanilla PU [Du Plessis et al., 2014], PU with sampling bias [Kato et al., 2019]
- Partial GW provides better classification accuracy even when source space and target domains share the same input space

Partial GW on Caltech data - different spaces

- Source $\mathcal{X}_s$ = Surf features $\rightarrow$ Target $\mathcal{X}_s$ = Decaf features[Donahue et al., 2014]
- or source $\mathcal{X}_s$ = Decaf features $\rightarrow$ Target $\mathcal{X}_s$ = Surf features

Scenario	*=C	*=A	*=W	* = D
Surf C$\rightarrow$ Decaf *	88.0	95.0	93.2	95.0
Decaf C$\rightarrow$ Surf *	87.4	87.4	86.6	94.0

- Previous methods (PU, PUSB, Partial-W) do not apply
- Partial GW yields similar performances as in setting where $\mathcal{X}_s = \mathcal{X}_t$

$\rightarrow$ Partial GW is able to leverage on the discriminative information conveyed by intra-domain similarity matrices.

Conclusion

Conclusion

- A framework that handles conditional and label shift in DA
- Joint estimation of label proportion and source/target mapping
- Theoretical guarantees under some geometrical assumptions in the latent space
- A framework that accounts for DA applied on data in incomparable spaces and with unknown classes

References

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1-2):151–175, 2010.

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

Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1853–1865, 2016.

Nicolas Courty, Rémi Flamary, Amaury Habrard, and Alain Rakotomamonjy. Joint distribution optimal transportation for domain adaptation. In *Advances in Neural Information Processing Systems*, pages 3730–3739, 2017a.

Nicolas Courty, Rémi Flamary, Amaury Habrard, and Alain Rakotomamonjy. Joint distribution optimal transportation for domain adaptation. In *Advances in Neural Information Processing Systems*, pages 3730–3739, 2017b.

Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655, 2014.

Marthinus C Du Plessis, Gang Niu, and Masashi Sugiyama. Analysis of learning from positive and unlabeled data. In *Advances in neural information processing systems*, pages 703–711, 2014.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, Francois Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

Masahiro Kato, Takeshi Teshima, and Junya Honda. Learning from positive and unlabeled data with a selection bias. In *ICLR*, 2019.

Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In *International Conference on Machine Learning*, pages 3122–3130, 2018.

Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2208–2217. JMLR. org, 2017.

Gaspard Monge. *Mémoire sur la théorie des déblais et des remblais*. De l'Imprimerie Royale, 1781.

Boris Muzellec, Julie Josse, Claire Boyer, and Marco Cuturi. Missing data imputation using optimal transport. In *International Conference on Machine Learning*, pages 7130–7140. PMLR, 2020.

Michel Olvera, Emmanuel Vincent, and Gilles Gasso. On the impact of normalization strategies in unsupervised adversarial domain adaptation for acoustic scene classification. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022.

Ievgen Redko, Nicolas Courty, Rémi Flamary, and Devis Tuia. Optimal transport for multi-source domain adaptation under target shift. In *Proceedings of Machine Learning Research*, volume 89, pages 849–858, 2019.

Yifan Wu, Ezra Winston, Divyansh Kaushik, and Zachary Lipton. Domain adaptation with asymmetrically-relaxed distribution alignment. *arXiv preprint arXiv:1903.01689*, 2019.