Joint distribution optimal transportation for domain adaptation

R. Flamary

Joint work with N. Courty, A. Habrard, A. Rakotomamonjy,

Optimal Transport meets Probability, Statistics and Machine Learning
Oaxaca, Mexico, 2017

Table of content

Supervised learning

Amazon

Traditional supervised learning

- We want to learn predictor such that $y \approx f(\mathbf{x})$.
- Actual $\mathcal{P}(X, Y)$ unknown.
- We have access to training dataset $(\mathbf{x}_i, y_i)_{i=1,\ldots,n}$ $(\widehat{\mathcal{P}}(X, Y))$.
- We choose a loss function $\mathcal{L}(y, f(\mathbf{x}))$ that measure the discrepancy.

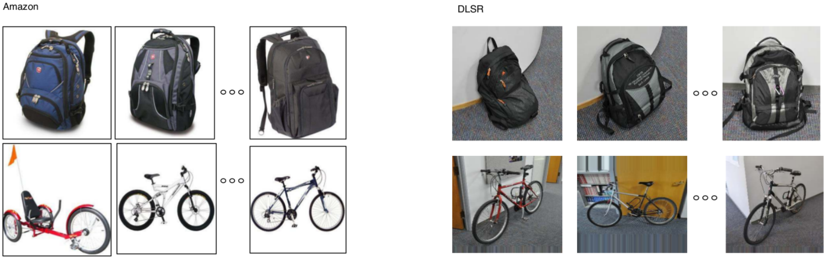
Empirical risk minimization

We week for a predictor f minimizing

$$\min_{f} \left\{ \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\widehat{\mathcal{P}}} \mathcal{L}(y, f(\mathbf{x})) = \sum_{j} \mathcal{L}(y_j, f(\mathbf{x}_j)) \right\} \tag{1}$$

- Well known generalization results for predicting on new data.
- Loss is usually $\mathcal{L}(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$ for least square regression and is $\mathcal{L}(y, f(\mathbf{x})) = \max(0, 1 - yf(\mathbf{x}))^2$ for squared Hinge loss SVM.

Domain Adaptation problem



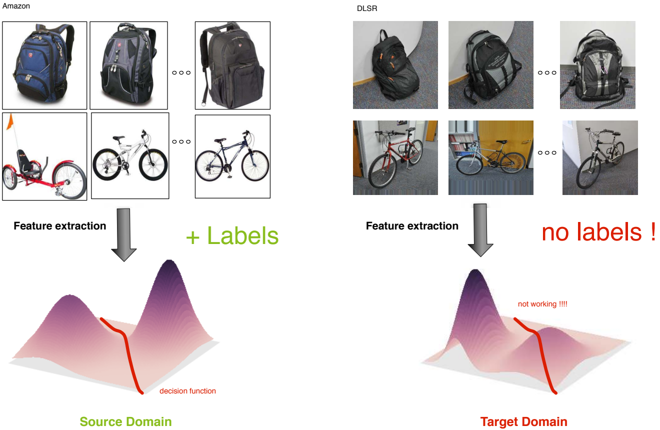
Our context

- Classification problem with data coming from different sources (domains).
- Distributions are different but related.

Problem

- 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

Unsupervised domain adaptation problem



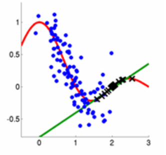
Problem

- 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

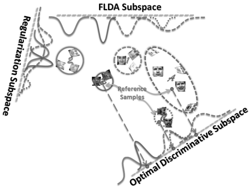
Reweighting schemes [Sugiyama et al., 2008]

- Distribution change between domains.
- Reweigh samples to compensate this change.



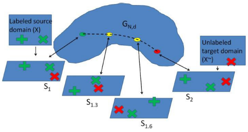
Subspace methods

- Data is invariant in a common latent subspace.
- Minimization of a divergence between the projected domains [Si et al., 2010].
- Use additional label information [Long et al., 2014b].

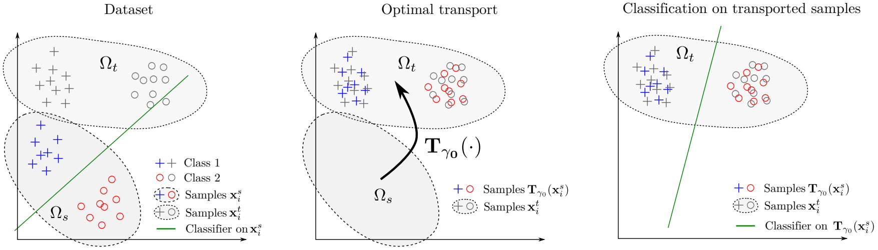


Gradual alignment

- Alignment along the geodesic between source and target subspace [R. Gopalan and Chellappa, 2014].
- Geodesic flow kernel [Gong et al., 2012].



Optimal transport for domain adaptation



Assumptions

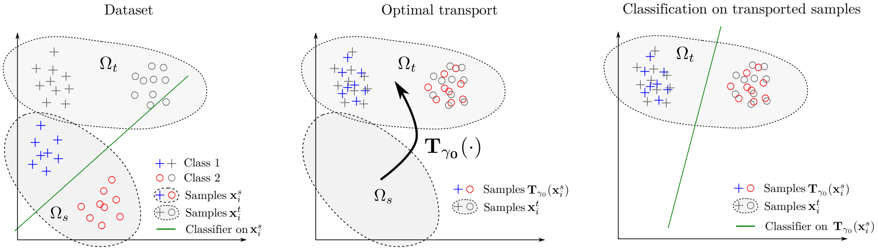
- There exist a transport in the feature space $\mathbf{T}$ between the two domains.
- The transport preserves the conditional distributions:

$$P_s(y|\mathbf{x}_s) = P_t(y|\mathbf{T}(\mathbf{x}_s)).$$

3-step strategy [Courty et al., 2016a]

1. Estimate optimal transport between distributions.
2. Transport the training samples with barycentric mapping .
3. Learn a classifier on the transported training samples.

Optimal transport for domain adaptation



Discussion

- Works very well in practice and handle large class of transformation.
- Step 1 and 2 can be fused by estimating the mapping [Perrot et al., 2016].

But

- Model transformation only in the feature space.
- Requires the same class proportion between domains [Tuia et al., 2015].
- Barycentric mapping is an approximation.
- In the end we search for a classifier $f : \mathbb{R}^d \rightarrow \mathbb{R}$, mapping is much more complex.

Joint distribution and classifier estimation

Objectives of JDOT

- Model the transformation of labels (allow change of proportion/value).
- Learn an optimal target predictor with no labels on target samples.
- Approach theoretically justified.

Joint distributions and dataset

- We work with the joint feature/label distributions.
- Let $\Omega \in \mathbb{R}^d$ be a compact input measurable space of dimension d and $\mathcal{C}$ the set of labels.
- Let $\mathcal{P}_s(X,Y) \in \mathcal{P}(\Omega \times \mathcal{C})$ and $\mathcal{P}_t(X,Y) \in \mathcal{P}(\Omega \times \mathcal{C})$ the source and target joint distribution.
- We have access to an empirical sampling $\hat{\mathcal{P}}_s = \frac{1}{N_s}\sum_{i=1}^{N_s} \delta_{\mathbf{x}_i^s,\mathbf{y}_i^s}$ of the source distribution defined by $\mathbf{X}_s = \{\mathbf{x}_i^s\}_{i=1}^{N_s}$ and label information $\mathbf{Y}_s = \{\mathbf{y}_i^s\}_{i=1}^{N_s}$.
- The target domain is defined only by an empirical distribution in the feature space with samples $\mathbf{X}_t = \{\mathbf{x}_i^t\}_{i=1}^{N_t}$.

Joint distribution OT (JDOT)

Proxy joint distribution

- Let f be a $\Omega \rightarrow \mathcal{C}$ function from a given class of hypothesis $\mathcal{H}$.
- We define the following joint distribution that use f as a proxy of y

$$\mathcal{P}_t^f = (\mathbf{x}, f(\mathbf{x}))_{\mathbf{x}\sim\mu_t} \tag{2}$$

and its empirical counterpart $\hat{\mathcal{P}_t}^f = \frac{1}{N_t}\sum_{i=1}^{N_t} \delta_{\mathbf{x}_i^t, f(\mathbf{x}_i^t)}$.

Learning with JDOT

We propose to learn the predictor f that minimize :

$$\min_f \quad \left\{ W_1(\hat{\mathcal{P}}_s, \hat{\mathcal{P}_t}^f) = \inf_{\boldsymbol{\gamma}\in\Delta} \sum_{ij} \mathcal{D}(\mathbf{x}_i^s, \mathbf{y}_i^s; \mathbf{x}_j^t, f(\mathbf{x}_j^t))\boldsymbol{\gamma}_{ij} \right\} \tag{3}$$

- Δ is the transport polytope.
- $\mathcal{D}(\mathbf{x}_i^s, \mathbf{y}_i^s; \mathbf{x}_j^t, f(\mathbf{x}_j^t)) = \alpha\|\mathbf{x}_i^s - \mathbf{x}_j^t\|^2 + \mathcal{L}(\mathbf{y}_i^s, f(\mathbf{x}_j^t))$ with $\alpha > 0$.
- We search for the predictor f that better align the joint distributions.
- Objective value is Transportation Lp [Thorpe et al., 2016], we optimize f.

Generalization bound (1)

Expected loss

The target expcted loss for a given predictor f is defined as

$$err_T(f) \stackrel{\text{def}}{=} \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{P}_t} \mathcal{L}(y, f(\mathbf{x})).$$

Similary we have on the target domain $err_T(f, g) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{P}_t} \mathcal{L}(g(\mathbf{x}), f(\mathbf{x}))$ and the inter function loss $err_T(f, g) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{P}_t} \mathcal{L}(g(\mathbf{x}), f(\mathbf{x}))$.

Probabilistic Lipschitzness [Urner et al., 2011, Ben-David et al., 2012]

Let $\phi : \mathbb{R} \to [0, 1]$. A labeling function $f : \Omega \to \mathbb{R}$ is ϕ-Lipschitz with respect to a distribution P over Ω if for all $\lambda > 0$

$$Pr_{x\sim P}\left[\exists y : [|f(x) - f(y)| > \lambda d(x, y)]\right] \leq \phi(\lambda).$$

Generalization bound (2)

Theorem 1
Let f_T^* and f_S^* be the two optimal labeling functions that verifies the ϕ-probabilistic Lipschitzness assumption. Let $\mathcal{L}$ be any loss function bounded by M, symmetric, k-lipschtiz and that satisfies the triangle inequality. Consider a sample of N_s labeled source instances drawn from $\mathcal{P}_s$ and N_t unlabeled instances drawn from μ_t, and any $f \in \mathcal{H}$, then for all $\lambda > 0$, with $\alpha = k\lambda$, we have with probability at least $1-\delta$ that:

$$\begin{aligned}\mathrm{err}_T(f) \leq\; & W_1(\hat{\mathcal{P}}_s, \hat{\mathcal{P}_t^f}) + \sqrt{\frac{2}{c'}\log(\frac{2}{\delta})}\left(\frac{1}{\sqrt{N_S}} + \frac{1}{\sqrt{N_T}}\right) \\ & + err_S(f_S^*) + err_T(f_S^*, f_T^*) + err_T(f_T^*) + k * M * \phi(\lambda).\end{aligned}$$

- First term is JDOT objective function.
- Second term is an empirical sampling bound.
- Last terms are usual in DA [Mansour et al., 2009, Ben-David et al., 2010].

Optimization problem

$$\min_{f \in \mathcal{H}, \boldsymbol{\gamma} \in \Delta} \sum_{i,j} \boldsymbol{\gamma}_{i,j} \left(\alpha d(\mathbf{x}_i^s, \mathbf{x}_j^t) + \mathcal{L}(y_i^s, f(\mathbf{x}_j^t)) \right) + \lambda \Omega(f) \tag{4}$$

Optimization procedure

- $\Omega(f)$ is a regularization for the predictor f
- We propose to use block coordinate descent (BCD)/Gauss Seidel.
- Provably converges to a stationary point of the problem.

γ update for a fixed f

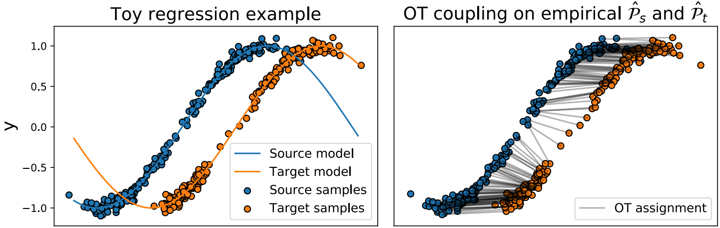
- Classical OT problem can be solved by network simplex.
- Regularized OT can also be used (just adds a term to problem (4))

f update for a fixed γ

$$\min_{f \in \mathcal{H}} \quad \sum_{i,j} \boldsymbol{\gamma}_{i,j} \mathcal{L}(y_i^s, f(\mathbf{x}_j^t)) + \lambda \Omega(f) \tag{5}$$

- Weighted loss from all source labels.
- γ performs label propagation.

Regression with JDOT



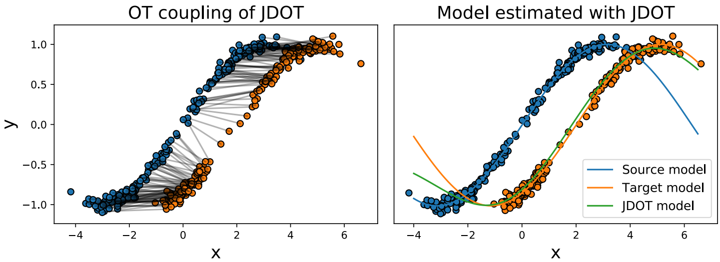
Least square regression with quadratic regularization

For a fixed γ the optimization problem is equivalent to

$$\min_{f \in \mathcal{H}} \quad \sum_j \frac{1}{n_t} \|\hat{y}_j - f(\mathbf{x}_j^t)\|^2 + \lambda \|f\|^2 \tag{6}$$

- $\hat{y}_j = n_t \sum_j \boldsymbol{\gamma}_{i,j} y_i^s$ is a weighted average of the source target values.
- Note that this problem is linear instead of quadratic.
- Can use any solver (linear, kernel ridge, neural network).

Regression with JDOT



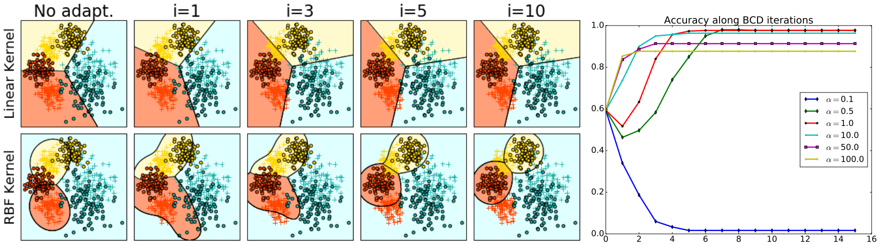
Least square regression with quadratic regularization

For a fixed γ the optimization problem is equivalent to

$$\min_{f \in \mathcal{H}} \quad \sum_j \frac{1}{n_t} \|\hat{y}_j - f(\mathbf{x}_j^t)\|^2 + \lambda \|f\|^2 \tag{6}$$

- $\hat{y}_j = n_t \sum_j \boldsymbol{\gamma}_{i,j} y_i^s$ is a weighted average of the source target values.
- Note that this problem is linear instead of quadratic.
- Can use any solver (linear, kernel ridge, neural network).

Classification with JDOT



Multiclass classification with Hinge loss

For a fixed γ the optimization problem is equivalent to

$$\min_{f_k\in\mathcal{H}} \quad \sum_{j,k} \hat{P}_{j,k}\mathcal{L}(1, f_k(\mathbf{x}_j^t)) + (1-\hat{P}_{j,k})\mathcal{L}(-1, f_k(\mathbf{x}_j^t)) + \lambda\sum_k \|f_k\|^2 \tag{7}$$

- $\hat{\mathbf{P}}$ is the class proportion matrix $\hat{\mathbf{P}} = \frac{1}{N_t}\boldsymbol{\gamma}^\top\mathbf{P}^s$.
- $\mathbf{P}^s$ and $\mathbf{Y}^s$ are defined from the source data with One-vs-All strategy as

$$Y_{i,k}^s = \begin{cases} 1 & \text{if } y_i^s = k \\ -1 & \text{else} \end{cases}, \qquad P_{i,k}^s = \begin{cases} 1 & \text{if } y_i^s = k \\ 0 & \text{else} \end{cases}$$

with $k \in 1, \cdots, K$ and K being the number of classes.

Caltech-Office classification dataset

Domains	Base	SurK	SA	OT-IT	OT-MM	JDOT
caltech→amazon	92.07	91.65	90.50	89.98	**92.59**	91.54
caltech→webcam	76.27	77.97	81.02	80.34	78.98	**88.81**
caltech→dslr	84.08	82.80	85.99	78.34	76.43	**89.81**
amazon→caltech	84.77	84.95	85.13	85.93	**87.36**	85.22
amazon→webcam	79.32	81.36	**85.42**	74.24	85.08	84.75
amazon→dslr	86.62	87.26	**89.17**	77.71	79.62	87.90
webcam→caltech	71.77	71.86	75.78	**84.06**	82.99	82.64
webcam→amazon	79.44	78.18	81.42	89.56	90.50	**90.71**
webcam→dslr	96.18	95.54	94.90	**99.36**	**99.36**	98.09
dslr→caltech	77.03	76.94	81.75	**85.57**	83.35	84.33
dslr→amazon	83.19	82.15	83.19	**90.50**	**90.50**	88.10
dslr→webcam	96.27	92.88	88.47	**96.61**	**96.61**	**96.61**
Mean	83.92	83.63	85.23	86.02	86.95	**89.04**
Avg. rank	4.50	4.75	3.58	3.00	2.42	**2.25**

Numerical experiments

- Classical dataset [Saenko et al., 2010] is dedicated to visual adaptation.
- Feature extraction by convolutional neural network [Donahue et al., 2014].
- Comparison with Surrogate Kernel [Zhang et al., 2013], Subspace Alignment [Fernando et al., 2013] and OT Domain Adaptation [Courty et al., 2016b].
- Parameter selected via reverse cross-validation [Zhong et al., 2010].
- SVM (Hinge loss) classifiers with linear kernel.
- Best ranking method and 2% accuracy gain in average.

Amazon Review Classification dataset

Domains	**NN**	**DANN**	**JDOT** (mse)	**JDOT** (Hinge)
books→dvd	0.805	**0.806**	0.794	0.795
books→kitchen	0.768	0.767	0.791	**0.794**
books→electronics	0.746	0.747	0.778	**0.781**
dvd→books	0.725	0.747	0.761	**0.763**
dvd→kitchen	0.760	0.765	0.811	**0.821**
dvd→electronics	0.732	0.738	0.778	**0.788**
kitchen→books	0.704	0.718	**0.732**	0.728
kitchen→dvd	0.723	0.730	0.764	**0.765**
kitchen→electronics	**0.847**	0.846	0.844	0.845
electronics→books	0.713	0.718	0.740	**0.749**
electronics→dvd	0.726	0.726	**0.738**	0.737
electronics→kitchen	0.855	0.850	0.868	**0.872**
Mean	0.759	0.763	0.783	**0.787**

Numerical experiments

- Dataset aim at predicting reviews across domains [Blitzer et al., 2006].
- Comparison with Domain adversarial neural network [Ganin et al., 2016].
- Classifier f is a neural network with same architecture as DANN.
- JDOT has better accuracy, classification loss is better than mean square error.

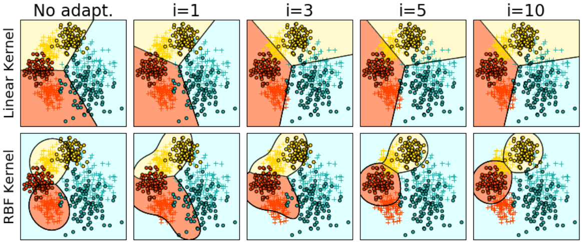
Wifi localization regression dataset

Domains	KRR	SurK	DIP	DIP-CC	GeTarS	CTC	CTC-TIP	JDOT
t1 $\rightarrow$ t2	80.84$\pm$1.14	90.36$\pm$1.22	87.98$\pm$2.33	91.30$\pm$3.24	86.76 $\pm$ 1.91	89.36$\pm$1.78	89.22$\pm$1.66	**93.03 $\pm$ 1.24**
t1 $\rightarrow$ t3	76.44$\pm$2.66	**94.97$\pm$1.29**	84.20$\pm$4.29	84.32$\pm$4.57	90.62$\pm$2.25	94.80$\pm$0.87	92.60 $\pm$ 4.50	90.06 $\pm$ 2.01
t2 $\rightarrow$ t3	67.12$\pm$1.28	85.83 $\pm$ 1.31	80.58 $\pm$ 2.10	81.22 $\pm$ 4.31	82.68 $\pm$ 3.71	87.92 $\pm$ 1.87	**89.52 $\pm$ 1.14**	86.76 $\pm$ 1.72
hallway1	60.02 $\pm$2.60	76.36 $\pm$ 2.44	77.48 $\pm$ 2.68	76.24$\pm$ 5.14	84.38 $\pm$ 1.98	86.98 $\pm$ 2.02	86.78 $\pm$ 2.31	**98.83$\pm$0.58**
hallway2	49.38 $\pm$ 2.30	64.69 $\pm$0.77	78.54 $\pm$ 1.66	77.8$\pm$ 2.70	77.38 $\pm$ 2.09	87.74 $\pm$ 1.89	87.94 $\pm$ 2.07	**98.45$\pm$0.67**
hallway3	48.42 $\pm$1.32	65.73 $\pm$ 1.57	75.10$\pm$ 3.39	73.40$\pm$ 4.06	80.64 $\pm$ 1.76	82.02$\pm$ 2.34	81.72 $\pm$ 2.25	**99.27$\pm$0.41**

Numerical experiments

- Objective is to predict position of a device on a discretized grid [Zhang et al., 2013].
- Same experimental protocol as [Zhang et al., 2013, Gong et al., 2016].
- Comparison with domain-invariant projection and its cluster regularized version ([Baktashmotlagh et al., 2013], **DIP** and **DIP-CC**), generalized target shift ([Zhang et al., 2015], **GeTarS**), and conditional transferable components, with its target information preservation regularization ([Gong et al., 2016], **CTC** and **CTC-TIP**).
- JDOT solve the adaptation problem for transfer across device (10% accuracy gain on Hallway).

Conclusion



Joint distribution optimal transportation for domain adaptation

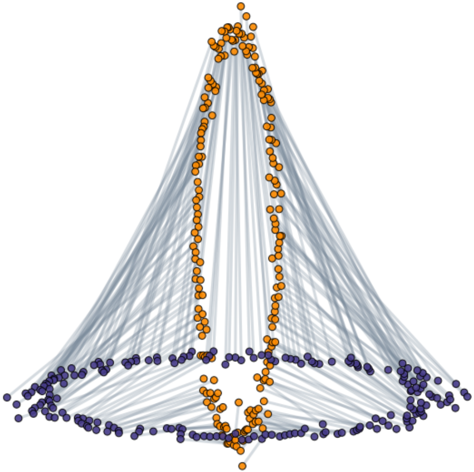
- General framework for domain adaptation.
- Model transformation of the joint distribution.
- Theoretical justification with generalization bound
- Similar in scope to [Long et al., 2014a] but use Wasserstein instead of MMD.
- Do not depend on the function hypothesis class (linear, kernel, neural network).

Thank you

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

- OT LP solver, Sinkhorn (stabilized, $\epsilon-$scaling, GPU)
- Domain adaptation with OT.
- Barycenters, Wasserstein unmixing.

Papers available on my website:
`https://remi.flamary.com/`



References I

Baktashmotlagh, M., Harandi, M., Lovell, B., and Salzmann, M. (2013).
Unsupervised domain adaptation by domain invariant projection.
In *ICCV*, pages 769–776.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. (2010).
A theory of learning from different domains.
Machine Learning, 79(1-2):151–175.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. (2010).
A theory of learning from different domains.
Machine Learning, 79(1-2):151–175.

Ben-David, S., Shalev-Shwartz, S., and Urner, R. (2012).
Domain adaptation–can quantity compensate for quality?
In *Proc of ISAIM*.

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

References II

Blitzer, J., McDonald, R., and Pereira, F. (2006).
Domain adaptation with structural correspondence learning.
In *Proc. of the 2006 conference on empirical methods in natural language processing*, pages 120–128.

Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. (2016a).
Optimal transport for domain adaptation.
Pattern Analysis and Machine Intelligence, IEEE Transactions on.

Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. (2016b).
Optimal transport for domain adaptation.
IEEE Transactions on Pattern Analysis and Machine Intelligence.

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

Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., and Darrell, T. (2014).
Decaf: A deep convolutional activation feature for generic visual recognition.
In *ICML*.

References III

Fernando, B., Habrard, A., Sebban, M., and Tuytelaars, T. (2013).
Unsupervised visual domain adaptation using subspace alignment.
In *ICCV*.

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

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(59):1–35.

Gong, B., Shi, Y., Sha, F., and Grauman, K. (2012).
Geodesic flow kernel for unsupervised domain adaptation.
In *CVPR*.

Gong, M., Zhang, K., Liu, T., Tao, D., Glymour, C., and Schölkopf, B. (2016).
Domain adaptation with conditional transferable components.
In *ICML*, volume 48, pages 2839–2848.

References IV

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

Long, M., Wang, J., Ding, G., Pan, S. J., and Philip, S. Y. (2014a).
Adaptation regularization: A general framework for transfer learning.
IEEE Transactions on Knowledge and Data Engineering, 26(5):1076–1089.

Long, M., Wang, J., Ding, G., Sun, J., and Yu, P. (2014b).
Transfer joint matching for unsupervised domain adaptation.
In *CVPR*, pages 1410–1417.

Mansour, Y., Mohri, M., and Rostamizadeh, A. (2009).
Domain adaptation: Learning bounds and algorithms.
In *Proc. of COLT*.

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

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

References V

R. Gopalan, R. L. and Chellappa, R. (2014).
Unsupervised adaptation across domain shifts by generating intermediate data representations.
IEEE Transactions on Pattern Analysis and Machine Intelligence, page To be published.

Redko, I., Habrard, A., and Sebban, M. (2016).
Theoretical Analysis of Domain Adaptation with Optimal Transport.
ArXiv e-prints.

Saenko, K., Kulis, B., Fritz, M., and Darrell, T. (2010).
Adapting visual category models to new domains.
In *ECCV*, LNCS, pages 213–226.

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., Suzuki, T., Nakajima, S., Kashima, H., von Bünau, P., and Kawanabe, M. (2008).
Direct importance estimation for covariate shift adaptation.
Annals of the Institute of Statistical Mathematics, 60(4):699–746.

References VI

Thorpe, M., Park, S., Kolouri, S., Rohde, G., and Slepcev, D. (2016).
A transportation l^p distance for signal analysis.
CoRR, abs/1609.08669.

Tuia, D., Flamary, R., Rakotomamonjy, A., and Courty, N. (2015).
Multitemporal classification without new labels: a solution with optimal transport.
In *8th International Workshop on the Analysis of Multitemporal Remote Sensing Images*.

Urner, R., Shalev-Shwartz, S., and Ben-David, S. (2011).
Access to unlabeled data can speed up prediction time.
In *Proceedings of ICML*, pages 641–648.

Zhang, K., Gong, M., and Schölkopf, B. (2015).
Multi-source domain adaptation: A causal view.
In *AAAI Conference on Artificial Intelligence*, pages 3150–3157.

Zhang, K., Zheng, V. W., Wang, Q., Kwok, J. T., Yang, Q., and Marsic, I. (2013).
Covariate shift in Hilbert space: A solution via surrogate kernels.
In *ICML*.

Zhong, E., Fan, W., Yang, Q., Verscheure, O., and Ren, J. (2010).
Cross validation framework to choose amongst models and datasets for transfer learning.
In *ECML/PKDD*.

Generalization error in domain adaptation

Theoretical bounds [Ben-David et al., 2010]

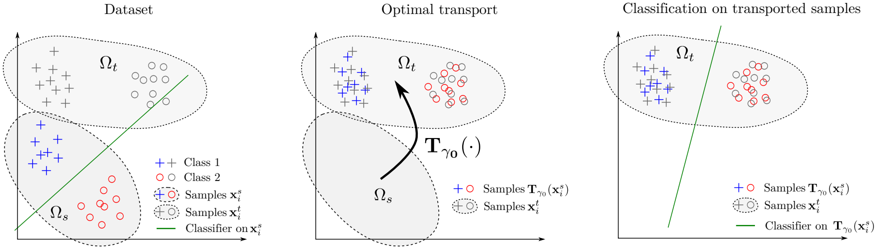
The error performed by a given classifier in the target domain is upper-bounded by the sum of three terms :

- Generalization error of the classifier in the source domain;
- Divergence measure between the densities the two domains (W_1 in [Redko et al., 2016]);
- A third term measuring how much the classification tasks are related to each other.

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

- Model the discrepancy between the distribution through a general transformation.
- Use **optimal transport** to estimate the transportation map between the two distributions.
- Use regularization terms for the optimal transport problem that exploits labels from the source domain.

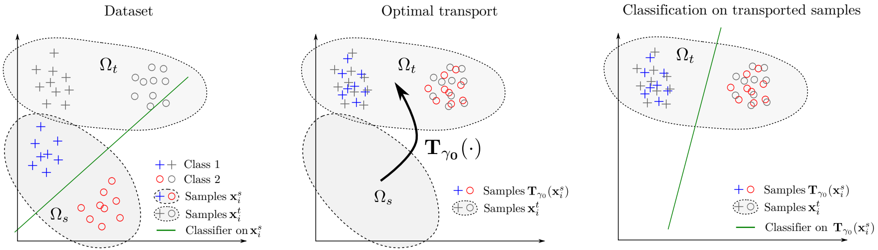
OT for domain adaptation : Step 1



Step 1 : Estimate optimal transport between distributions.

- Choose the ground metric (squared euclidean in our experiments).
- Using regularization allows
  - Large scale and regular OT with entropic regularization [Cuturi, 2013].
  - Class labels in the transport with group lasso [Courty et al., 2016a].
- Efficient optimization based on Bregman projections [Benamou et al., 2015] and
  - Majoration minimization for non-convex group lasso.
  - Generalized Conditionnal gradient for general regularization (cvx. lasso, Laplacian).

OT for domain adaptation : Steps 2 & 3



Step 2 : Transport the training samples onto the target distribution.

- The mass of each source sample is spread onto the target samples (line of γ_0).
- We estimate the transported position for each source [Ferradans et al., 2014] :

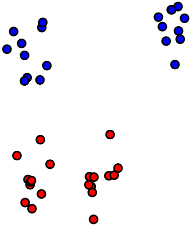
$$\widehat{\mathbf{x}_i^s} = \operatorname*{argmin}_{\mathbf{x}} \quad \sum_j \gamma_0(i,j) c(\mathbf{x}, \mathbf{x}_j^t). \tag{8}$$

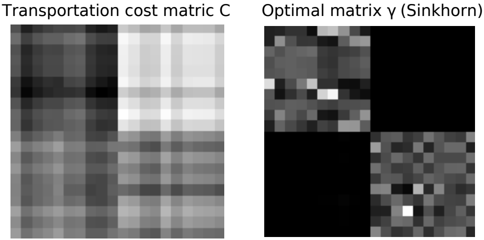
- Can be computed efficiently for a quadratic loss.

Step 3 : Learn a classifier on the transported training samples

- Classic ML problem when samples are well transported.

Efficient regularized optimal transport





Entropic regularization [Cuturi, 2013]

$$\gamma_0^\lambda = \underset{\gamma \in \mathcal{P}}{\operatorname{argmin}} \langle \gamma, \mathbf{C} \rangle_F - \lambda h(\gamma), \tag{9}$$

where $h(\gamma) = -\sum_{i,j} \gamma(i,j) \log \gamma(i,j)$ computes the entropy of γ.

- Entropy introduces smoothness in γ_0^λ.
- **Sinkhorn-Knopp** algorithm (efficient implementation in parallel, GPU).
- General framework using Bregman projections [Benamou et al., 2015].