

Metric Learning in Optimal Transport for Domain Adaptation

Tanguy Kerdoncuff, Rémi Emonet and Marc Sebban

Univ Lyon, UJM-Saint-Etienne, CNRS, Institut d Optique Graduate School, Laboratoire
Hubert Curien UMR 5516, F-42023, SAINT-ETIENNE, France

June 28, 2020

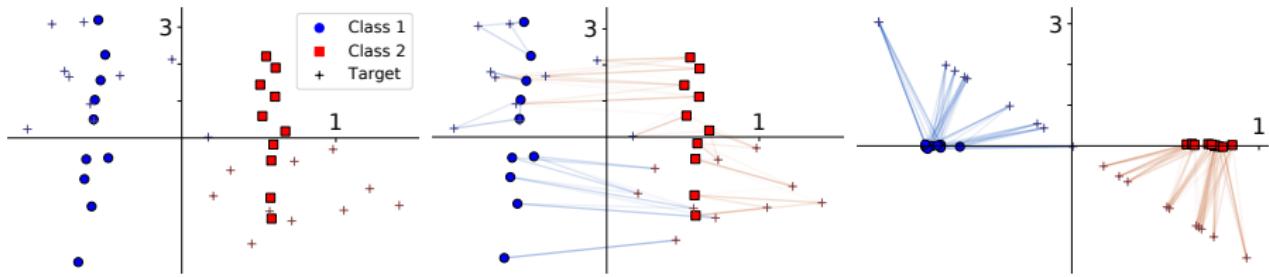


Table of Contents

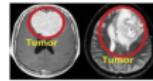
1 Background

2 Main contributions

- Bound on the Target Risk
- PCA and Wasserstein Distance
- Algorithm: Metric Learning for Optimal Transport

3 Experiments

Domain Adaptation (DA): Intuition



Machine Learning System
to detect tumors

Labeled Data



HOW TO TRANSFER ?



Domain Adaptation
(Hot topic in Machine Learning)



Unlabeled Data



tumor or not tumor

Domain Adaptation (DA): Notations



x_i^s : Image



x_j^t : Image

- In DA, both source $(x_i^s)_i$ and target $(x_j^t)_j$ samples are available

Domain Adaptation (DA): Notations



x_i^s : Image
 y_i^s : Chair



x_j^t : Image
 y_j^t : Unknown

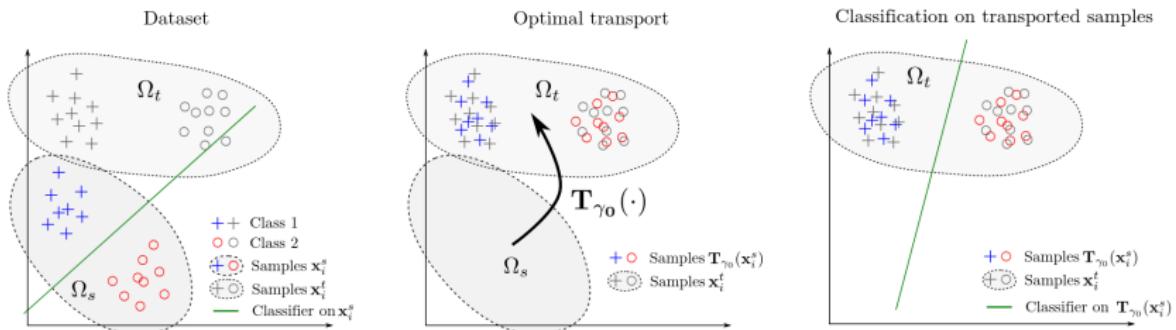
- In DA, both source $(x_i^s)_i$ and target $(x_j^t)_j$ samples are available
- Only source labels $(y_i^s)_i$ are given

Discrete Optimal Transport (OT)

Discrete Optimal Transport

$$\mathcal{W}^2(\hat{\mu}_s, \hat{\mu}_t) = \min_{\gamma \in \hat{\Pi}(\hat{\mu}_s, \hat{\mu}_t)} \sum_{i=1}^{m_s} \sum_{j=1}^{m_t} \|x_i^s - x_j^t\|_2^2 \gamma_{ij} = \min_{\gamma \in \hat{\Pi}(\hat{\mu}_s, \hat{\mu}_t)} \langle \gamma, C^2 \rangle.$$

With $\hat{\Pi}(\hat{\mu}_s, \hat{\mu}_t) = \{\gamma \in \mathbb{R}_+^{m_s \times m_t} | \gamma \mathbf{1}_{m_t} = \hat{\mu}_s, \gamma^T \mathbf{1}_{m_s} = \hat{\mu}_t\}$



Optimal Transport for Domain Adaptation (OTDA) [Courty et al., 2017]

Entropy regularization [Cuturi, 2013]

$$\min_{\gamma \in \hat{\Pi}(\hat{\mu}_s, \hat{\mu}_t)} \langle \gamma, C^2 \rangle - \lambda_e \left(- \sum_{i=1}^{m_s} \sum_{j=1}^{m_t} \gamma_{ij} \log(\gamma_{ij}) \right) \quad (1)$$

- ✓ Smooth solution, the Transport Plan γ is **not sparse** anymore

Previous methods: entropy and classes regularization

Entropy regularization [Cuturi, 2013]

$$\min_{\gamma \in \hat{\Pi}(\hat{\mu}_s, \hat{\mu}_t)} \langle \gamma, C^2 \rangle - \lambda_e \left(- \sum_{i=1}^{m_s} \sum_{j=1}^{m_t} \gamma_{ij} \log(\gamma_{ij}) \right) \quad (1)$$

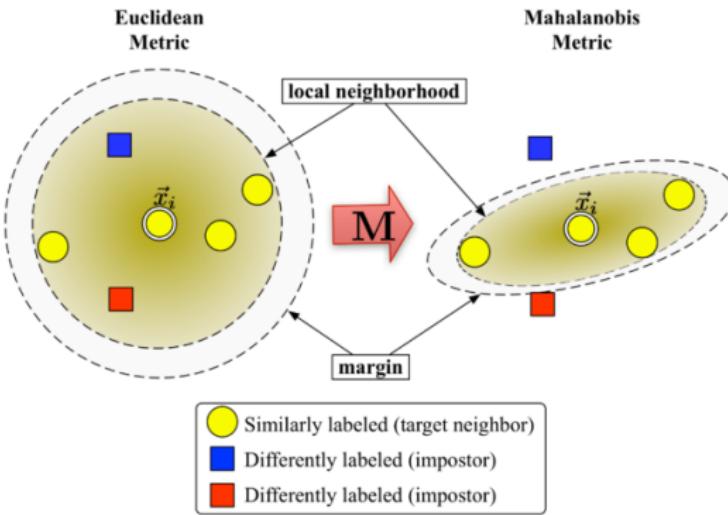
- ✓ Smooth solution, the Transport Plan γ is **not sparse** anymore

Optimal Transport for Domain Adaptation [Courty et al., 2017]

$$\min_{\gamma \in \hat{\Pi}(\hat{\mu}_s, \hat{\mu}_t)} \langle \gamma, C^2 \rangle - \lambda_e \Omega_e(\gamma) + \lambda_c \left(\sum_{j=1}^{m_t} \sum_{cl=1}^c \|\gamma(\mathcal{I}_{cl}, j)\|_2 \right) \quad (2)$$

- ✓ Points of different classes should **not** be send to same location

Metric learning



LMNN [Weinberger et al., 2006]

Mahalanobis metric

- $C_M^2(x, x') = (x - x')M(x - x')^T = \|Lx - Lx'\|_2^2$ with $L^T L = M$

Table of Contents

1 Background

2 Main contributions

- Bound on the Target Risk
- PCA and Wasserstein Distance
- Algorithm: Metric Learning for Optimal Transport

3 Experiments

Bound on the Target Risk

Classical form of Target Risk Bound [Ben-David et al., 2007]

$$\begin{aligned} \text{target risk} &\leq \text{source risk} \\ &+ \text{distance}(\mu_s, \mu_t) \\ &+ \lambda \text{ (error of the best classifier)} \end{aligned} \tag{3}$$

Bound on the Target Risk

Classical form of Target Risk Bound [Ben-David et al., 2007]

$$\text{target risk} \leq \text{source risk} + \text{distance}(\mu_s, \mu_t) + \lambda \text{ (error of the best classifier)} \quad (3)$$

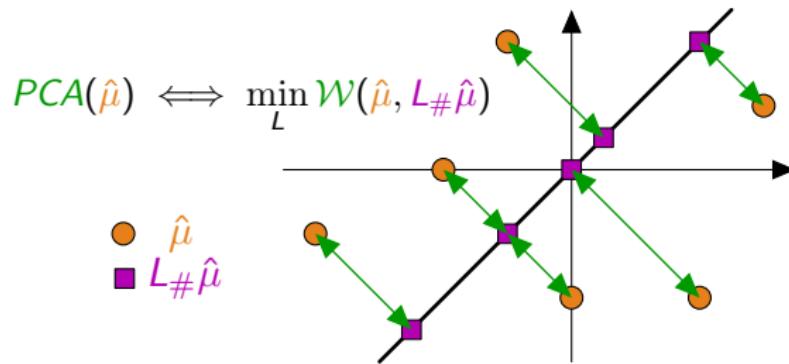
Our Target Risk Bound (based on [Redko et al., 2017])

$$\begin{aligned} \text{target risk} &\leq \text{source risk} + \mathcal{W}(L_{s\#}\hat{\mu}_s, L_{t\#}\hat{\mu}_t) \\ &\quad + \mathcal{W}(\hat{\mu}_s, L_{s\#}\hat{\mu}_s) + \mathcal{W}(L_{t\#}\hat{\mu}_t, \hat{\mu}_t) \\ &\quad + \mathcal{W}(\mu_s, \hat{\mu}_s) + \mathcal{W}(\hat{\mu}_t, \mu_t) + \lambda \end{aligned} \quad (4)$$

PCA and Wasserstein Distance

Our Target Risk Bound

$$\begin{aligned} \text{target risk} &\leq \text{source risk} + \mathcal{W}(L_s \# \hat{\mu}_s, L_t \# \hat{\mu}_t) \\ &\quad + \mathcal{W}(\hat{\mu}_s, L_s \# \hat{\mu}_s) + \mathcal{W}(\hat{\mu}_t, L_t \# \hat{\mu}_t) \\ &\quad + \mathcal{W}(\mu_s, \hat{\mu}_s) + \mathcal{W}(\hat{\mu}_t, \mu_t) + \lambda \end{aligned} \tag{5}$$



Metric Learning for Optimal Transport (MLOT)

Input: η (gradient step) X_s X_t Y_s

- 1: $V_s = PCA(X_s)$, $V_t = PCA(X_t)$
 - 2: $L_s = V_s^T V_s$, $L_t = V_t^T V_t$
 - 3: **for** $i = 1$ **to** N **do**
 - 4: $\gamma = \text{Solve } OT(C(L_s, L_t))$
 - 5: $L_s = L_s - \eta \nabla_{L_s} (\langle \gamma, C^2(L_s, L_t) \rangle + \lambda_i \Omega_i(L_s))$
 - 6: **end for**
 - 7: $\tilde{X}_s = \gamma L_t X_t$
 - 8: **return** classifier((\tilde{X}_s, Y_s) , X_t)
-

Metric Learning for Optimal Transport (MLOT)

Input: η (gradient step) X_s X_t Y_s

- 1: $V_s = PCA(X_s)$, $V_t = PCA(X_t)$
- 2: $L_s = V_s^T V_s$, $L_t = V_t^T V_t$
- 3: **for** $i = 1$ **to** N **do**
- 4: $\gamma = \text{Solve } OT(C(L_s, L_t))$
- 5: $L_s = L_s - \eta \nabla_{L_s} (\langle \gamma, C^2(L_s, L_t) \rangle + \lambda_i \Omega_i(L_s))$
- 6: **end for**
- 7: $\tilde{X}_s = \gamma L_t X_t$
- 8: **return** classifier((\tilde{X}_s, Y_s) , X_t)

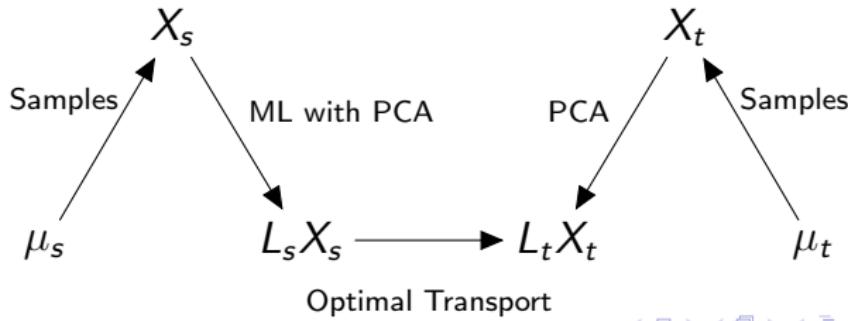


Illustration of the usefulness of Metric Learning

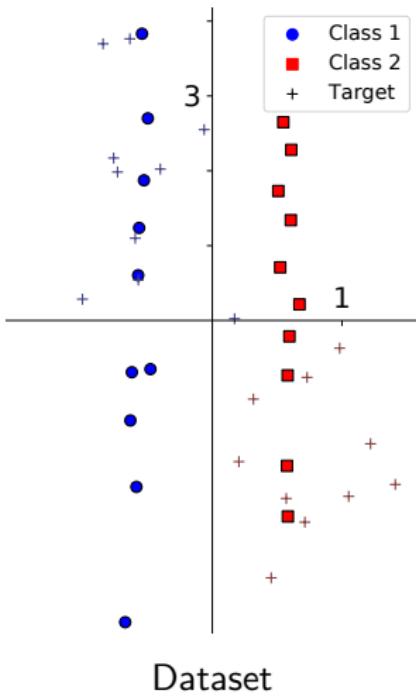


Illustration of the usefulness of Metric Learning

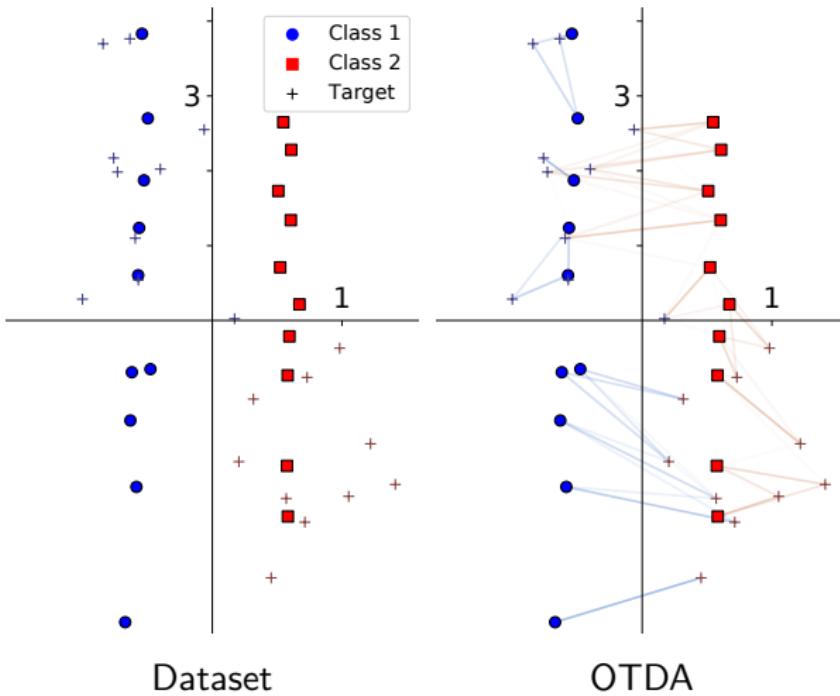


Illustration of the usefulness of Metric Learning

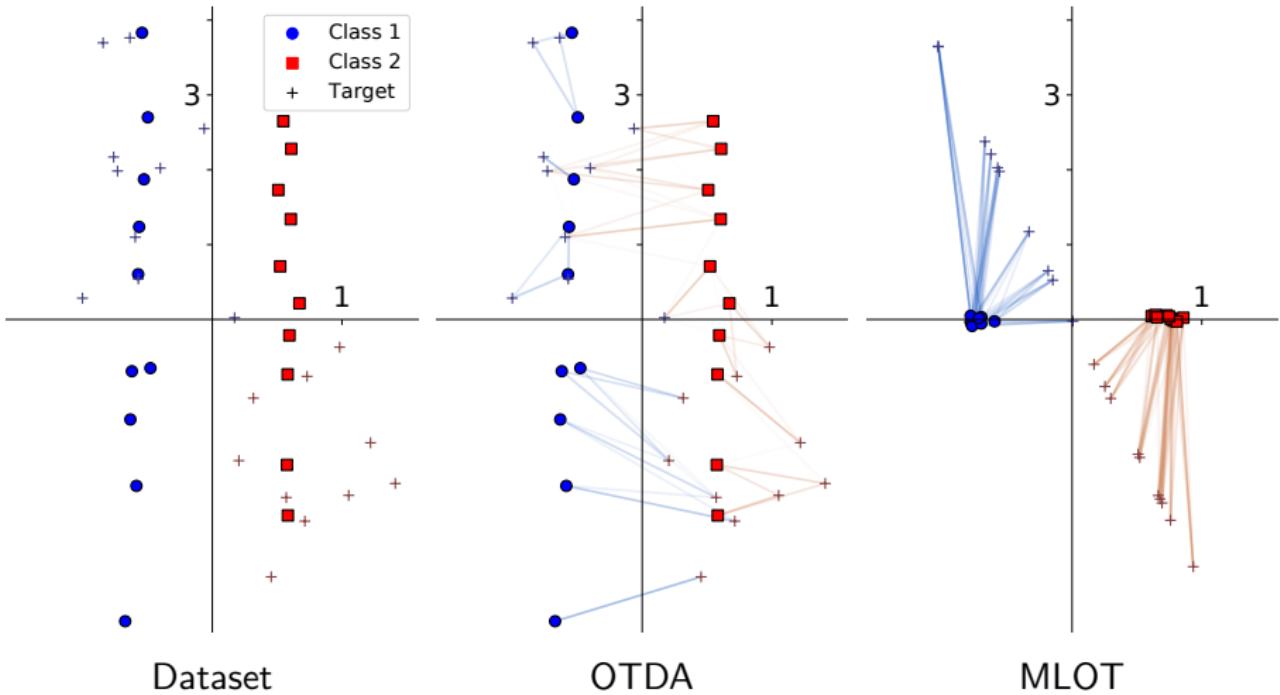


Table of Contents

1 Background

2 Main contributions

- Bound on the Target Risk
- PCA and Wasserstein Distance
- Algorithm: Metric Learning for Optimal Transport

3 Experiments

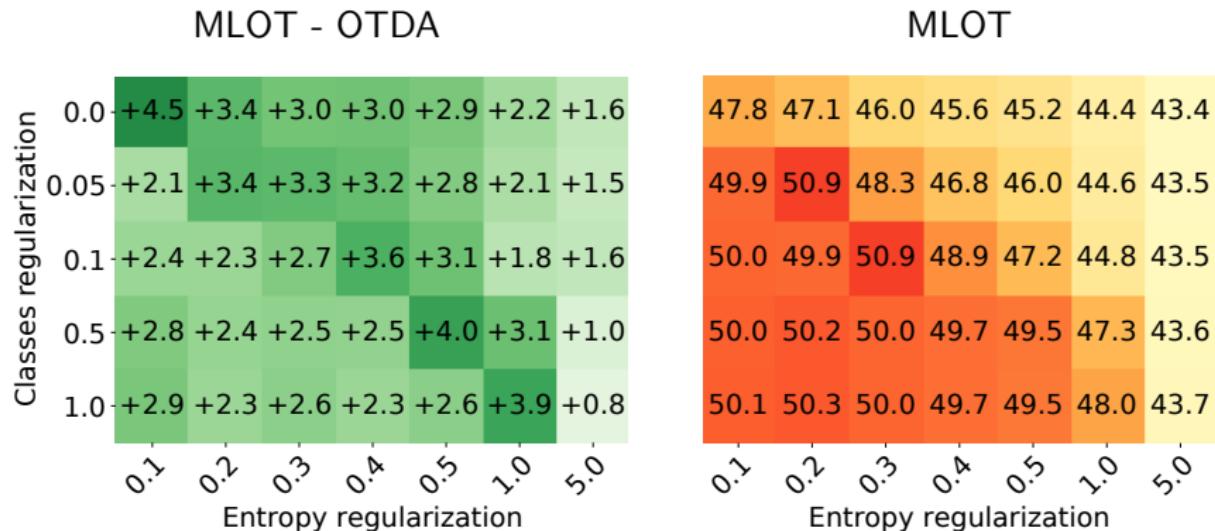
Cross validation of hyperparameters

- In **Unsupervised Domain Adaptation** the target label are **not available**
- We use the Reverse Validation of [Zhong et al., 2010]

Method	OT	TCA	LMNN	SA	JDOT	OTDA	OTDA _p	MLOT
Classical Validation	45.3	45.3	47.9	47.4	48.5	52.8	54	55.1
Reverse Validation	42.2	44.5	44.8	45.8	47.0	48.2	48.8	49.7

Table: Accuracy on Office-Caltech dataset [Gong et al., 2012] with SURF features.

OTDA vs MLOT



Accuracy of OTDA and MLOT on Office-Caltech dataset [Gong et al., 2012] with SURF features.

References

-  Ben-David, S., Blitzer, J., Crammer, K., and Pereira, F. (2007). Analysis of representations for domain adaptation.
In *NIPS*.
-  Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. (2017). Optimal transport for domain adaptation.
PAMI.
-  Cuturi, M. (2013). Sinkhorn distances: Lightspeed computation of optimal transport.
In *NIPS*.
-  Gong, B., Shi, Y., Sha, F., and Grauman, K. (2012). Geodesic flow kernel for unsupervised domain adaptation.
In *CVPR*.
-  Redko, I., Habrard, A., and Sebban, M. (2017). Theoretical analysis of domain adaptation with optimal transport.
In *ECML PKDD*.
-  Weinberger, K. Q., Blitzer, J., and Saul, L. K. (2006). Distance metric learning for large margin nearest neighbor classification.



MLOT and Optimal Transition between slide

- The code of the presented method is available on my github:
<https://github.com/Hv0nnus/MLOT>
- To use the Optimal Transport-based Slide Transition, see my github :
<https://github.com/Hv0nnus/TransitionPDF>