Optimal transport for domain adaptation

R. Flamary

Joint work with N. Courty, D. Tuia, A. Rakotomamonjy

Statlearn 2016, Vannes, April 8, 2016

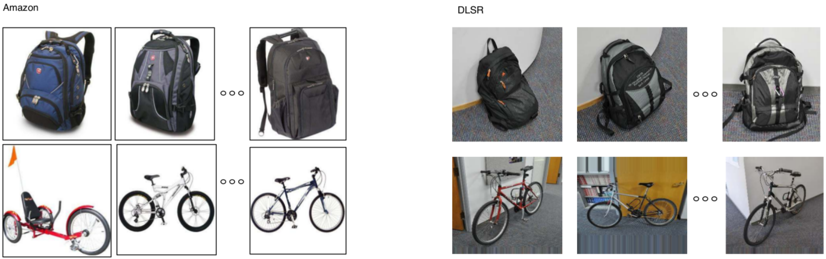
Table of content

Domain Adaptation problem

Traditional machine learning hypothesis

- We have access to training data.
- Probability distribution of the training set and the testing are the same.
- We want to learn a classifier that generalizes to new data.

Domain Adaptation problem



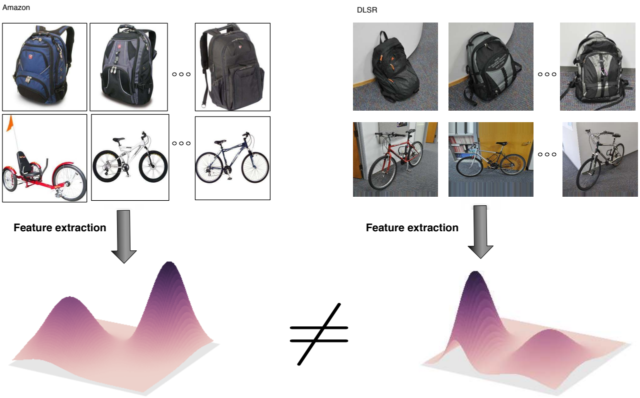
Traditional machine learning hypothesis

- We have access to training data.
- Probability distribution of the training set and the testing are the same.
- We want to learn a classifier that generalizes to new data.

Our context

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

Domain Adaptation problem

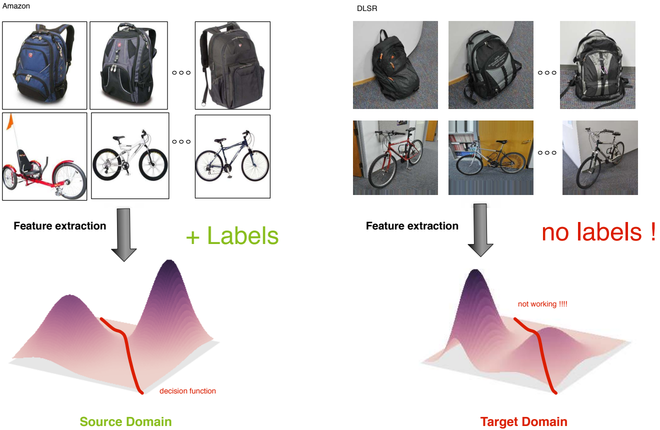


Probability Distribution Functions over the domains

Our context

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

Unsupervised domain adaptation problem



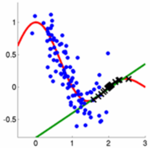
Problems

- 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

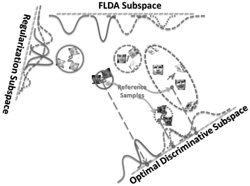
Reweighting schemes [Sugiyama et al., 2008]

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



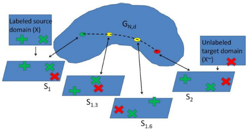
Subspace methods

- 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., 2014].



Gradual alignment

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



Generalization error in domain adaptation

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

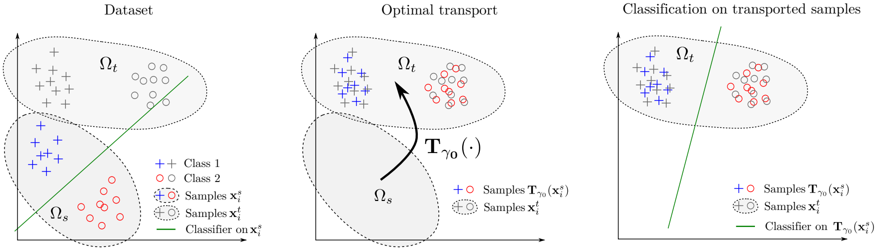
The error performed by a given classifier in the target domain is upper-bounded by the sum of three terms :

- Error of the classifier in the source domain;
- Divergence measure between the two pdfs in the two domains;
- A third term measuring how much the classification tasks are related to each other.

Our proposal

- 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.

Optimal transport for domain adaptation



Assumptions

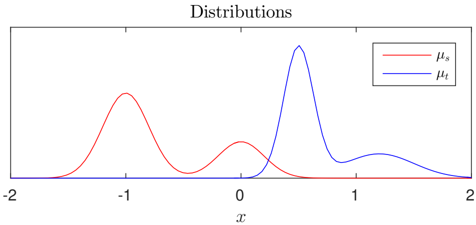
- There exist a transport $\mathbf{T}$ between the source and target domain.
- The transport preserves the conditional distributions:

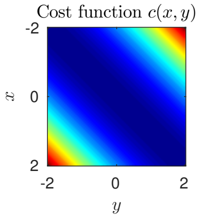
$$P_s(y|\mathbf{x}_s) = P_t(y|\mathbf{T}(\mathbf{x}_s)).$$

3-step strategy

1. Estimate optimal transport between distributions.
2. Transport the training samples onto the target distribution.
3. Learn a classifier on the transported training samples.

Optimal transport





- Given two probability measures μ_s and μ_t on $\Omega_s \times \Omega_t$ and a cost function $c : \Omega_s \times \Omega_t \to \mathbb{R}^+$.
- The Kantorovich formulation [Kantorovich, 1942] seeks for a probabilistic coupling $\boldsymbol{\gamma} \in \mathcal{P}(\Omega_s \times \Omega_t)$ between Ω_s and Ω_t:

$$\begin{array}{lll} \boldsymbol{\gamma}_0 = & \arg\min_{\boldsymbol{\gamma}} & \int_{\Omega_s \times \Omega_t} c(\mathbf{x}, \mathbf{y}) \boldsymbol{\gamma}(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y}, \\ & \text{s.t.} & \int_{\Omega_t} \boldsymbol{\gamma}(\mathbf{x}, \mathbf{y}) d\mathbf{y} = \mu_s, \\ & & \int_{\Omega_s} \boldsymbol{\gamma}(\mathbf{x}, \mathbf{y}) d\mathbf{x} = \mu_t, \end{array} \tag{1}$$

- $\boldsymbol{\gamma}$ can be understood as a joint probability measure with marginals μ_s and μ_t.

Optimal transport, discrete case

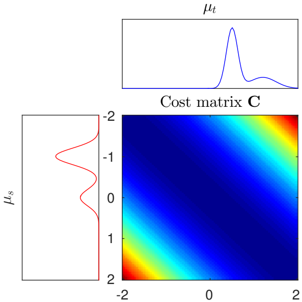
- When μ_s ad μ_t are discrete histograms with n_s and n_t bins.
- The optimization problem becomes

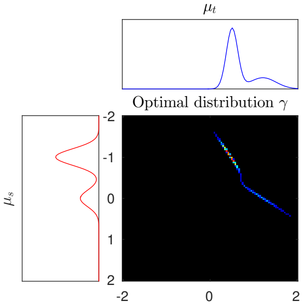
$$\boldsymbol{\gamma}_0 = \underset{\boldsymbol{\gamma} \in \mathcal{P}}{\arg\min} \quad \langle \boldsymbol{\gamma}, \mathbf{C} \rangle_F$$

  where $\mathbf{C}$ is a transportation cost matrix and

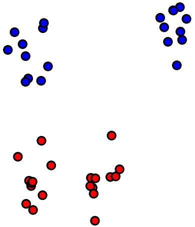
$$\mathcal{P} = \left\{ \boldsymbol{\gamma} \in (\mathbb{R}^+)^{\mathbf{n_s} \times \mathbf{n_t}} | \, \boldsymbol{\gamma} \mathbf{1_{n_t}} = \boldsymbol{\mu_s}, \boldsymbol{\gamma}^{\mathbf{T}} \mathbf{1_{n_s}} = \boldsymbol{\mu_t} \right\}$$

- Classical LP problem (Linear cost, linear constraints).
- On the right optimal matrix $\boldsymbol{\gamma}_0$ for two examples (black is exactly zero).
- In machine learning we often have access only to samples !

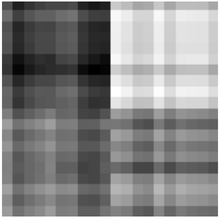




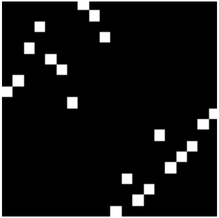
Optimal transport for empirical distributions



Transportation cost matric $\mathbf{C}$



Optimal matrix γ (LP)

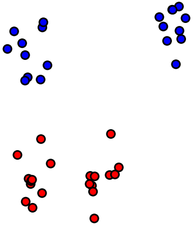


Empirical distributions

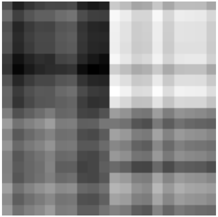
$$\mu_s = \sum_{i=1}^{n_s} p_i^s \delta_{\mathbf{x}_i^s}, \quad \mu_t = \sum_{i=1}^{n_t} p_i^t \delta_{\mathbf{x}_i^t} \tag{2}$$

- $\delta_{\mathbf{x}_i}$ is the Dirac at location $\mathbf{x}_i \in \mathbb{R}^d$ and p_i^s and p_i^t are probability masses.
- $\sum_{i=1}^{n_s} p_i^s = \sum_{i=1}^{n_t} p_i^t = 1$, in this work $p_i^s = \frac{1}{n_s}$ and $p_i^t = \frac{1}{n_t}$.
- Samples stored in matrices: $\mathbf{X}_s = [\mathbf{x}_1^s, \ldots, \mathbf{x}_{ns}^s]^\top$ and $\mathbf{X}_t = [\mathbf{x}_1^t, \ldots, \mathbf{x}_{nt}^t]^\top$
- The cost is set to the square euclidean distance between sample positions.

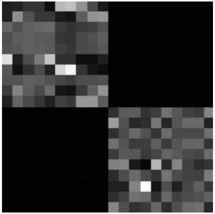
Efficient regularized optimal transport



Transportation cost matric C



Optimal matrix γ (Sinkhorn)



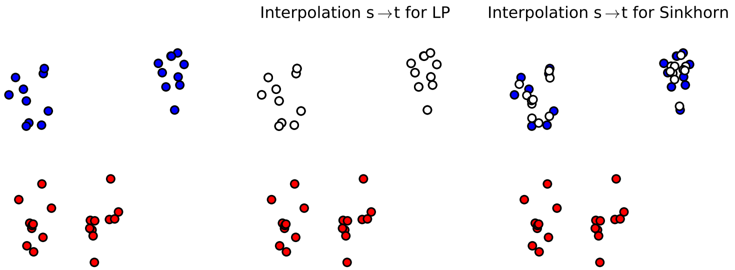
Entropic regularization [Cuturi, 2013]

$$\gamma_0^\lambda = \underset{\gamma \in \mathcal{P}}{\arg\min} \, \langle \gamma, \mathbf{C} \rangle_F - \lambda h(\gamma), \tag{3}$$

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 GPU).
- General framework using Bregman projections [Benamou et al., 2015].

Transporting the discrete samples



Barycentric mapping [Ferradans et al., 2014]

- The mass of each source sample is spread onto the target samples (line of γ_0).
- The source samples becomes a weighted sum of dirac (impractical for ML).
- We estimate the transported position for each source with:

$$\widehat{\mathbf{x}_i^s} = \arg\min_{\mathbf{x}} \quad \sum_j \gamma_0(i,j) c(\mathbf{x}, \mathbf{x}_j^t). \tag{4}$$

- Position of the transported samples for :

$$\hat{\mathbf{X}}_s = \text{diag}(\gamma_0 \mathbf{1}_{n_t})^{-1}\gamma_0 \mathbf{X}_t \quad \text{and} \quad \hat{\mathbf{X}}_t = \text{diag}(\gamma_0^\top \mathbf{1}_{n_s})^{-1}\gamma_0^\top \mathbf{X}_s. \tag{5}$$

Regularization for domain adaptation

Optimization problem

$$\min_{\boldsymbol{\gamma}\in\mathcal{P}} \quad \langle\boldsymbol{\gamma},\mathbf{C}\rangle_F + \lambda\Omega_s(\boldsymbol{\gamma}) + \eta\Omega(\boldsymbol{\gamma}), \tag{6}$$

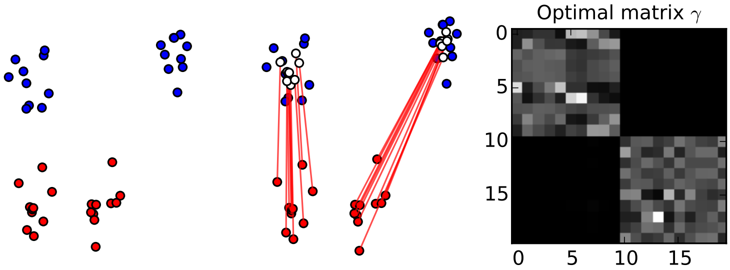
where

- $\Omega_s(\boldsymbol{\gamma})$ Entropic regularization [Cuturi, 2013].
- $\eta \geq 0$ and $\Omega_c(\cdot)$ is a DA regularization term.
- Regularization to avoid overfitting in high dimension and encode additional information.

Regularization terms for domain adaptation $\Omega(\boldsymbol{\gamma})$

- Class based regularization [Courty et al., 2014] to encode the source label information.
- Graph regularization [Ferradans et al., 2014] to promote local sample similarity conservation.
- Semi-supervised regularization when some target samples have known labels.

Entropic regularization

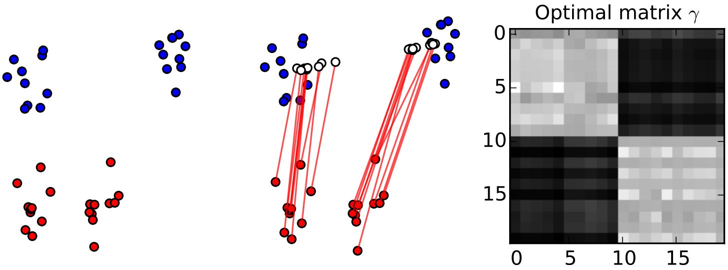


Entropic regularization [Cuturi, 2013]

$$\Omega_s(\boldsymbol{\gamma}) = \sum_{i,j} \boldsymbol{\gamma}(i,j) \log \boldsymbol{\gamma}(i,j)$$

- Extremely efficient optimization scheme (Sinkhorn Knopp).
- Solution is not sparse anymore due to the regularization.
- Strong regularization force the samples to concentrate on the center of mass of the target samples.

Entropic regularization

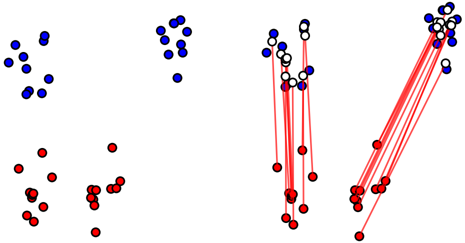


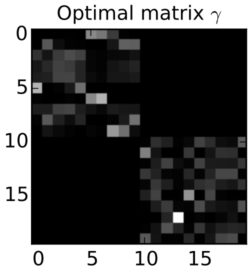
Entropic regularization [Cuturi, 2013]

$$\Omega_s(\boldsymbol{\gamma}) = \sum_{i,j} \boldsymbol{\gamma}(i,j) \log \boldsymbol{\gamma}(i,j)$$

- Extremely efficient optimization scheme (Sinkhorn Knopp).
- Solution is not sparse anymore due to the regularization.
- Strong regularization force the samples to concentrate on the center of mass of the target samples.

Class-based regularization





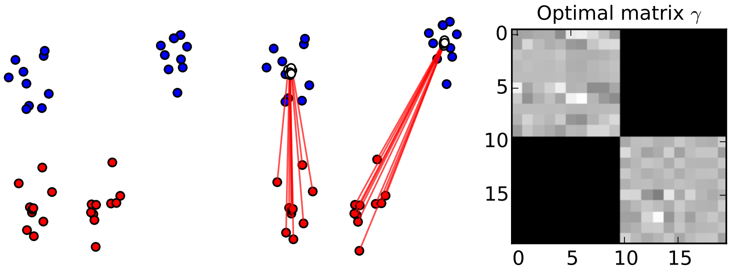
Group lasso regularization

- We group components of $\boldsymbol{\gamma}$ using classes from the source domain:

$$\Omega_c(\boldsymbol{\gamma}) = \sum_j \sum_c ||\boldsymbol{\gamma}(\mathcal{I}_c, j)||_q^p, \tag{7}$$

- $\mathcal{I}_c$ contains the indices of the lines related to samples of the class c in the source domain.
- $||\cdot||_q^p$ denotes the ℓ_q norm to the power of p.
- For $p \leq 1$, we encourage a target domain sample j to receive masses only from "same class" source samples.

Class-based regularization



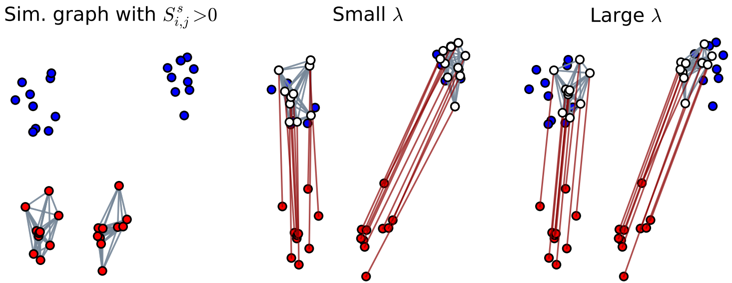
Group lasso regularization

- We group components of $\boldsymbol{\gamma}$ using classes from the source domain:

$$\Omega_c(\boldsymbol{\gamma}) = \sum_j \sum_c ||\boldsymbol{\gamma}(\mathcal{I}_c, j)||_q^p, \tag{7}$$

- $\mathcal{I}_c$ contains the indices of the lines related to samples of the class c in the source domain.
- $||\cdot||_q^p$ denotes the ℓ_q norm to the power of p.
- For $p \leq 1$, we encourage a target domain sample j to receive masses only from "same class" source samples.

Laplacian regularization for sample displacement



Graph regularization for the sample displacement

- Proposed in [Ferradans et al., 2014] for color transfer in images.
- $\hat{\mathbf{x}}^s_i - \mathbf{x}^s_i$ is the displacement of source sample $\mathbf{x}^s_i$ during transport.
- We want similar samples defined in $\mathbf{S}^s$ to have similar displacements:

$$\Omega(\boldsymbol{\gamma}) = \frac{1}{N_s^2}\sum_{i,j} S^s_{i,j}\|(\hat{\mathbf{x}}^s_i - \mathbf{x}^s_i) - (\hat{\mathbf{x}}^s_j - \mathbf{x}^s_j)\|^2$$

- Similarity graph $\mathbf{S}^s$ is pruned using the classes in the source domain.
- Quadratic regularization term with possible regularization of the transported target samples ($\mathbf{S}^t$).

Semi-supervised domain adaptation

Principle

- A few target samples have a known label.
- How to include this information in the OT problem?

Semi-supervised learning [Rousselle and Canu, 2015]

- Learn a regularized OT matrix.
- Prune the matrix components according to the known classes.

Our proposal: Semi supervised transport

- Regularize (again?) the OT matrix during its estimation.
- Forbid inter-class mass transfer.
- Regularization term: $\Omega_{ss}(\boldsymbol{\gamma}) = \langle \boldsymbol{\gamma}, \mathbf{M} \rangle_F$
- $M_{ij} = +\infty$ whenever $y_i^s \neq y_j^t$ and $M_{ij} = 0$ otherwise (same or unknown label).
- Boils down to modifying the cost matrix $\mathbf{C}$.

Optimization problem

$$\min_{\boldsymbol{\gamma}\in\mathcal{P}} \quad \langle\boldsymbol{\gamma},\mathbf{C}\rangle_F + \lambda\Omega_s(\boldsymbol{\gamma}) + \eta\Omega(\boldsymbol{\gamma}),$$

Special cases

- $\eta = 0$: Sinkhorn Knopp [Cuturi, 2013].
- $\lambda = 0$ and Laplacian regularization: Large quadratic program solved with conditionnal gradient [Ferradans et al., 2014].
- Non convex group lasso $\ell_p - \ell_1$: Majoration Minimization with Sinkhorn Knopp [Courty et al., 2014].

General framework with convex regularization $\Omega(\gamma)$

- Can we use efficient Sinkhorn Knopp scaling to solve the global problem?
- Yes using generalized conditional gradient [Bredies et al., 2009].
- Linearization of the second regularization term but not the entropic regularization.

Generalized conditionnal gradient

- Proposed in [Bredies et al., 2009].
- Composite minimization:

$$\min_{\gamma\in\mathcal{P}} \quad f(\gamma)+g(\gamma),$$

where $f(\cdot)$ is differentiable, possibly non-convex, $g(\cdot)$ convex, possibly non-differentiable.

- Application to optimal transport:

$$f(\gamma)=\langle\gamma,\mathbf{C}\rangle_F+\eta\Omega_c(\gamma)$$

$$g(\gamma)=\lambda\Omega_s(\gamma)$$

- Step 3 in Algorithm becomes

$$\gamma^\star=\underset{\gamma\in\mathcal{P}}{\arg\min} \quad \left\langle\gamma,\mathbf{C}+\eta\nabla\Omega_c(\gamma^k)\right\rangle_F+\lambda\Omega_s(\gamma)$$

Entropic regularized OT with efficient solver.

Algorithm

1: Initialize $k=0$ and $\gamma^0\in\mathcal{P}$
2: **repeat**
3: With $\mathbf{G}\in\nabla f(\gamma^k)$, solve

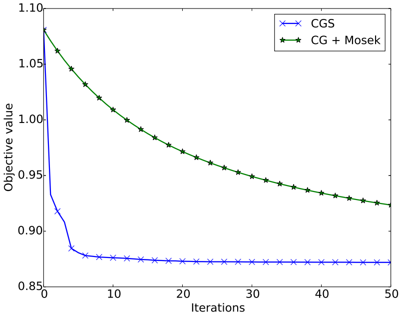
$$\gamma^\star=\underset{\gamma\in\mathcal{P}}{\arg\min} \quad \langle\gamma,\mathbf{G}\rangle_F+g(\gamma)$$

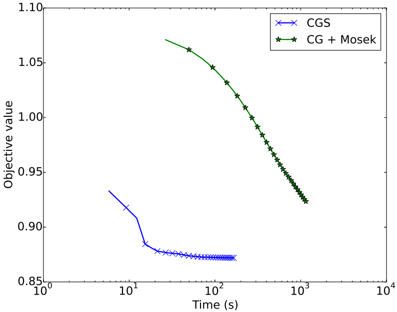
4: Find the optimal step α^k

$$\alpha^k=\underset{0\le\alpha\le1}{\arg\min} f(\gamma^k+\alpha\Delta\gamma)+g(\gamma^k+\alpha\Delta\gamma)$$

with $\Delta\gamma=\gamma^*-\gamma^k$
5: $\gamma^{k+1}\leftarrow\gamma^k+\alpha^k\Delta\gamma$, set $k\leftarrow k+1$
6: **until** Convergence

Computationnal performance

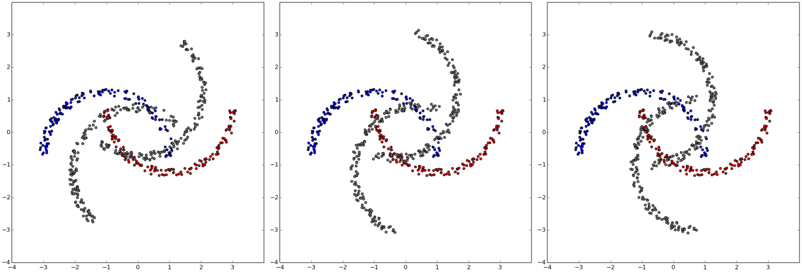




Comparison between CG and Generalized CG

- Experiments with Group Lasso regularization (200 samples in source and target).
- CG used Mosek for solving Linear Program.
- Objective value as a function of iterations and computational time.

Simulated problem with controllable complexity



Two moons problem [Germain et al., 2013]

- Two entangled moons with a rotation between domains.
- The rotation angle allow a control of the adaptation difficulty.
- Comparison with Domain Adaptation SVM[Bruzzone and Marconcini, 2010] and [Germain et al., 2013].

OT domain adaptation:

- **OT-exact** non-regularized OT.
- **OT-IT** Entropic reg.
- **OT-GL** Group-lasso + entropic reg.
- **OT-Lap** Laplacian + entropic reg.

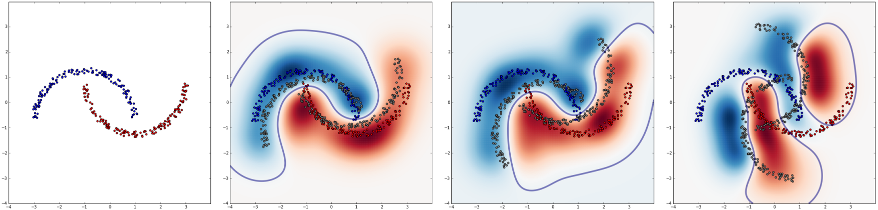
Results on the two moons dataset

	10°	20°	30°	40°	50°	70°	90°
SVM (no adapt.)	0	0.104	0.24	0.312	0.4	0.764	0.828
DASVM	0	**0**	0.259	0.284	0.334	0.747	0.820
PBDA	0	0.094	0.103	0.225	0.412	0.626	0.687
OT-exact	0	0.028	0.065	0.109	0.206	0.394	**0.507**
OT-IT	0	0.007	0.054	0.102	0.221	0.398	**0.508**
OT-GL	0	**0**	**0**	**0.013**	**0.196**	**0.378**	**0.508**
OT-Lap	0	**0**	0.004	0.062	0.201	0.402	0.524

Discussion

- Average prediction error for adaptation from 10° to 90°.
- Clear advantage of the optimal transport techniques.
- Regularization helps (a lot) up to 40°.
- 90° is the theoretical limit (positive definite Jacobian of the transformation).

Results on the two moons dataset

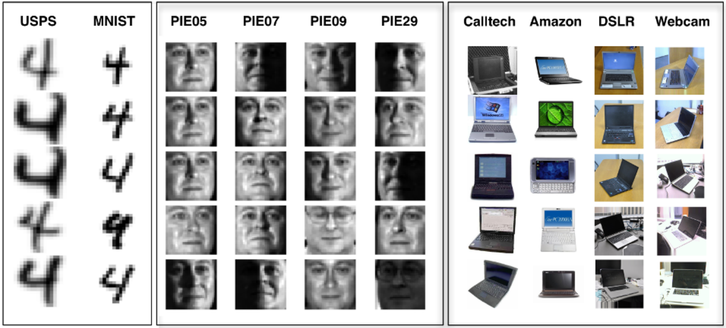


(a) source domain (b) rotation=20° (c) rotation=40° (d) rotation=90°

Discussion

- Average prediction error for adaptation from 10° to 90°.
- Clear advantage of the optimal transport techniques.
- Regularization helps (a lot) up to 40°.
- 90° is the theoretical limit (positive definite Jacobian of the transformation).

Visual adaptation datasets



Datasets

- **Digit recognition**, MNIST VS USPS (10 classes, d=256, 2 dom.).
- **Face recognition**, PIE Dataset (68 classes, d=1024, 4 dom.).
- **Object recognition**, Caltech-Office dataset (10 classes, d=800/4096, 4 dom.).

Numerical experiments

- Comparison with state of the art on the 3 datasets.
- Comparison on object recognition with deep invariant features.
- Semi supervised extension.

Experimental setup

Compared methods

- **1NN**, original classifier without adaptation
- **PCA**, projection on the first principal components of the joint source/target distribution (estimated from a concatenation of source and target samples);
- **GFK**, Geodesic Flow Kernel [Gong et al., 2012];
- **TSL**, Transfer Subspace Learning [Si et al., 2010], minimizing the Bregman divergence between the domains embedded in lower dimensional spaces;
- **JDA**, Joint Distribution Adaptation [Long et al., 2013].

Parameter validation

- In unsupervised DA, no target labels are available.
- For fair comparison, parameters validated on a validation target set.
- Performance estimated with the validated parameters on an independent test set in the target domain.
- Average recognition accuracy on 10 validation/test splits.

Comparison on vision datasets

Datasets	Digits		Faces		Objects	
Methods	ACC	Nb best	ACC	Nb best	ACC	Nb best
1NN	48.66	0	26.22	0	28.47	0
PCA	42.94	0	34.55	0	37.98	0
GFK	52.56	0	26.15	0	39.21	0
TSL	47.22	0	36.10	0	42.97	1
JDA	57.30	0	**56.69**	**7**	44.34	1
OT-exact	49.96	0	50.47	0	36.69	0
OT-IT	59.20	0	54.89	0	42.30	0
OT-Lap	61.07	0	56.10	3	43.20	0
OT-LpLq	**64.11**	**1**	55.45	0	46.42	1
OT-GL	63.90	**1**	55.88	2	**47.70**	**9**

Discussion

- We report mean accuracy (ACC) and the number of time the method have been the best among all possible adaptation pairs.
- OT works very well on digits and object recognition (+7% and +3% wrt JDA).
- Good but not best on face recognition (-.5% wrt JDA).

Deep achitecture features on Caltech-Office

	Layer 6				Layer 7			
Domains	**DeCAF**	**JDA**	**OT-IT**	**OT-GL**	**DeCAF**	**JDA**	**OT-IT**	**OT-GL**
C→A	79.25	88.04	88.69	**92.08**	85.27	89.63	91.56	**92.15**
C→W	48.61	79.60	75.17	**84.17**	65.23	79.80	82.19	**83.84**
C→D	62.75	84.12	83.38	**87.25**	75.38	85.00	85.00	**85.38**
A→C	64.66	81.28	81.65	**85.51**	72.80	82.59	84.22	**87.16**
A→W	51.39	80.33	78.94	**83.05**	63.64	83.05	81.52	**84.50**
A→D	60.38	**86.25**	85.88	85.00	75.25	85.50	**86.62**	85.25
W→C	58.17	**81.97**	74.80	81.45	69.17	79.84	81.74	**83.71**
W→A	61.15	90.19	80.96	**90.62**	72.96	90.94	88.31	**91.98**
W→D	97.50	**98.88**	95.62	96.25	98.50	**98.88**	98.38	91.38
D→C	52.13	81.13	77.71	**84.11**	65.23	81.21	82.02	**84.93**
D→A	60.71	91.31	87.15	**92.31**	75.46	91.92	92.15	**92.92**
D→W	85.70	**97.48**	93.77	96.29	92.25	**97.02**	96.62	94.17
mean	65.20	86.72	83.64	**88.18**	75.93	87.11	87.53	**88.11**

Discussion

- Invariant features provided by a deep learning architecture [Donahue et al., 2014].
- Comparison with features obtained on different layers.
- Important gain when using OT in addition to invariant features.

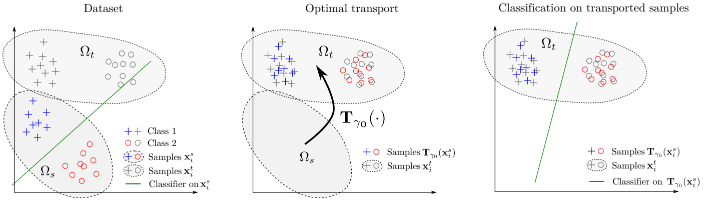
Semi-supervised domain adaptation

	Unsupervised + labels		Semi-supervised		
Domains	**OT-IT**	**OT-GL**	**OT-IT**	**OT-GL**	**MMDT**
C→A	37.0 ± 0.5	41.4 ± 0.5	46.9 ± 3.4	47.9 ± 3.1	**49.4 ± 0.8**
C→W	28.5 ± 0.7	37.4 ± 1.1	64.8 ± 3.0	**65.0 ± 3.1**	63.8 ± 1.1
C→D	35.1 ± 1.7	44.0 ± 1.9	59.3 ± 2.5	**61.0 ± 2.1**	56.5 ± 0.9
A→C	32.3 ± 0.1	36.7 ± 0.2	36.0 ± 1.3	**37.1 ± 1.1**	36.4 ± 0.8
A→W	29.5 ± 0.8	37.8 ± 1.1	63.7 ± 2.4	**64.6 ± 1.9**	**64.6 ± 1.2**
A→D	36.9 ± 1.5	46.2 ± 2.0	57.6 ± 2.5	**59.1 ± 2.3**	56.7 ± 1.3
W→C	35.8 ± 0.2	36.5 ± 0.2	38.4 ± 1.5	**38.8 ± 1.2**	32.2 ± 0.8
W→A	39.6 ± 0.3	41.9 ± 0.4	47.2 ± 2.5	47.3 ± 2.5	**47.7± 0.9**
W→D	77.1 ± 1.8	**80.2 ± 1.6**	79.0 ± 2.8	79.4 ± 2.8	67.0 ± 1.1
D→C	32.7 ± 0.3	34.7 ± 0.3	35.5 ± 2.1	3**6.8 ± 1.5**	34.1 ± 1.5
D→A	34.7 ± 0.3	37.7 ± 0.3	45.8 ± 2.6	46.3 ± 2.5	**46.9 ± 1.0**
D→W	81.9 ± 0.6	**84.5 ± 0.4**	83.9 ± 1.4	84.0 ± 1.5	74.1 ± 0.8
mean	41.8	46.6	54.8	**55.6**	52.5

Discussion

- Some target samples have a known label (3 labels per class).
- We compare with unsupervised adaptation where the known labels are used in the classifier training.
- In semi-supervised case we use the modified metric matrix.
- Competitive when compared to state of the art [Hoffman et al., 2013].

Conclusion



Optimal transport for domain adaptation

- General framework for adapting between domains (transport the samples).
- Can handle very complex transformation between domains.
- Works very well but needs regularization (class based).
- Deep learning friendly + semi-supervised version.

Current and future works

- Extension to multi-domain/multi-task learning.
- What about domains with different class proportion ? [Tuia et al., 2015].
- What about the cost matrix $\mathbf{C}$? Can we do better than euclidean?
- Theoretical generalization bounds?

Collaborators

N. Courty

R. Flamary

D. Tuia

A. Rakotomamonjy

Barycenters

Collaborators

N. Courty

R. Flamary

D. Tuia

A. Rakotomamonjy

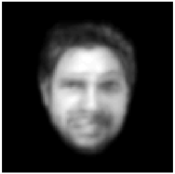
Barycenters

L2 Barycenter

L1 Barycenter

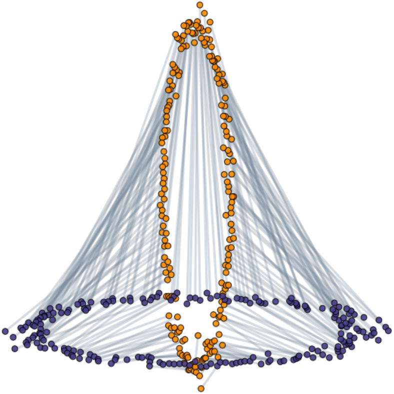
KL Barycenter

Wass. Barycenter



Thank you

Code available on the following web site:
`http://remi.flamary.com/soft/soft-transp.html`
Paper available on ArXiv
`http://arxiv.org/abs/1507.00504`



References I

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.

Benamou, J.-D., Carlier, G., Cuturi, M., Nenna, L., and Peyré, G. (2015).
Iterative bregman projections for regularized transportation problems.
SIAM Journal on Scientific Computing, 37(2):A1111–A1138.

Bredies, K., Lorenz, D. A., and Maass, P. (2009).
A generalized conditional gradient method and its connection to an iterative shrinkage method.
Computational Optimization and Applications, 42(2):173–193.

Bruzzone, L. and Marconcini, M. (2010).
Domain adaptation problems: A dasvm classification technique and a circular validation strategy.
Pattern Analysis and Machine Intelligence, IEEE Transactions on, 32(5):770–787.

Courty, N., Flamary, R., and Tuia, D. (2014).
Domain adaptation with regularized optimal transport.
In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*.

References II

Cuturi, M. (2013).
Sinkhorn distances: Lightspeed computation of optimal transportation.
In *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 *Proceedings of The 31st International Conference on Machine Learning*, pages 647–655.

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

Germain, P., Habrard, A., Laviolette, F., and Morvant, E. (2013).
A PAC-Bayesian Approach for Domain Adaptation with Specialization to Linear Classifiers.
In *ICML*, pages 738–746, Atlanta, USA.

Gong, B., Shi, Y., Sha, F., and Grauman, K. (2012).
Geodesic flow kernel for unsupervised domain adaptation.
In *CVPR*, pages 2066–2073. IEEE.

References III

Hoffman, J., Rodner, E., Donahue, J., Saenko, K., and Darrell, T. (2013).
Efficient learning of domain-invariant image representations.
In *International Conference on Learning Representations*.

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., Sun, J., and Yu, P. (2013).
Transfer feature learning with joint distribution adaptation.
In *ICCV*, pages 2200–2207.

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

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

References IV

Rousselle, D. and Canu, S. (2015).
Optimal transport for semi-supervised domain adaptation.
In *ESANN*.

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

Sugiyama, M., Nakajima, S., Kashima, H., Buenau, P. V., and Kawanabe, M. (2008).
Direct importance estimation with model selection and its application to covariate shift adaptation.
In *Advances in neural information processing systems*, pages 1433–1440.

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*.