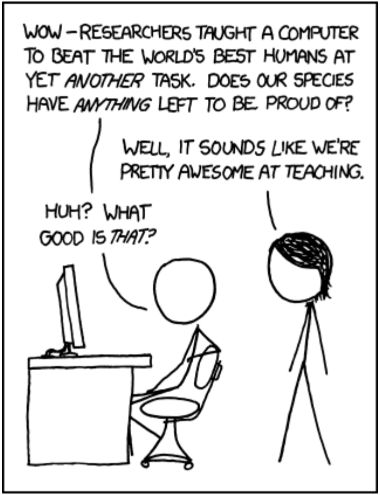
Optimal transport for machine learning

Rémi Flamary, Nicolas Courty

Statlearn 2018, Nice, April 5 2018

Introduction

Machine learning / Statistical learning / AI

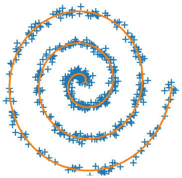


https://xkcd.com

Three aspects of Machine Learning

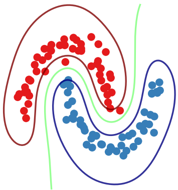
Unsupervised learning

- Extract information from unlabeled data
- Find labels (clustering) or subspaces/manifolds.
- Generate realistic data (GAN).



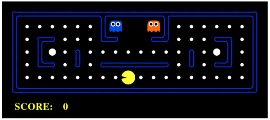
Supervised Learning

- Learning to predict from labeld dataset.
- Regression, Classification.
- Can use unsupervised information (DA, Semi-sup.)

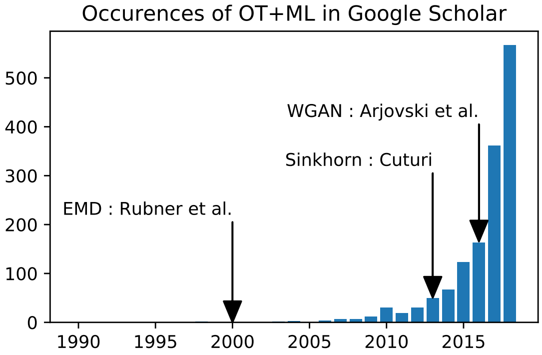


Reinforcement Learning

- Let the machine experiment.
- Learn from its mistakes.
- Framework for learning to play games.



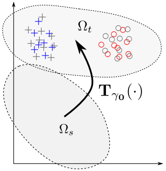
Optimal transport for machine learning

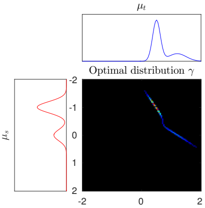


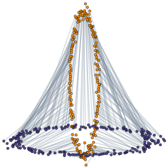
Short history of OT for ML

- Recently introduced to ML (well known in image processing since 2000s).
- Computationnal OT allow numerous applications (regularization).
- Deep learning boost (numerical optimization and GAN).

Three aspects of optimal transport







Transporting with optimal transport

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

Divergence between histograms

- Use the ground metric to encode complex relations between the bins.
- Loss for multilabel classifier [Frogner et al., 2015]
- Loss for spectral unmixing [Flamary et al., 2016b].

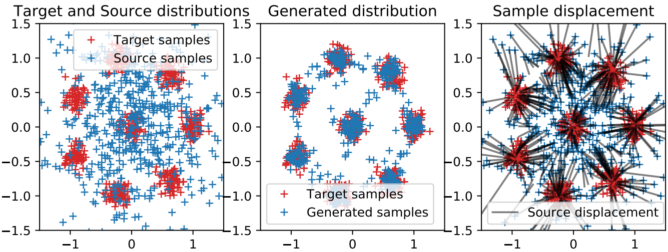
Divergence between empirical distributions

- Non parametric divergence between non overlapping distributions.
- Objective function for GAN [Arjovsky et al., 2017].
- Estimate discriminant subspace [Flamary et al., 2016a].

Table of content

Mapping with optimal transport

Mapping with optimal transport

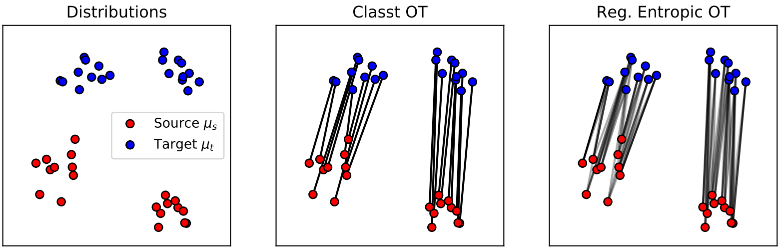


Mapping estimation

- Mapping do not exist in general between empirical distributions.
- Barycentric mapping [Ferradans et al., 2014].
- Smooth mapping estimation [Perrot et al., 2016, Seguy et al., 2017].

Why map ?

- Sensible displacement to align distributions.
- Color adaptation in image [Ferradans et al., 2014].
- Domain adaptation and transfer learning [Courty et al., 2016].

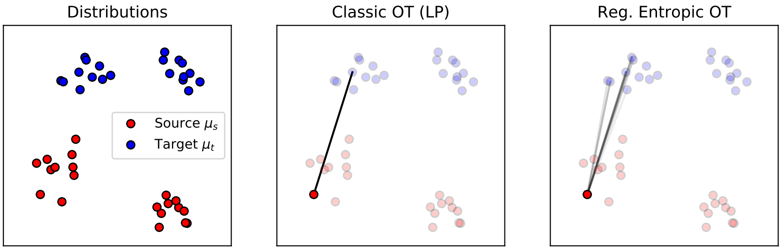


Barycentric mapping [Ferradans et al., 2014]

$$\widehat{T}_{\gamma_0}(\mathbf{x}_i^s) = \arg\min_{\mathbf{x}} \quad \sum_j \gamma_0(i,j) c(\mathbf{x}, \mathbf{x}_j^t). \tag{1}$$

- The mass of each source sample is spread onto the target samples (line of γ_0).
- The mapping is the barycenter of the target samples weighted by γ_0
- Closed form solution for the quadratic loss.
- Limited to the samples in the distribution (no out of sample).
- Trick: learn OT on few samples and apply displacement to the nearest point.

Transporting the discrete samples

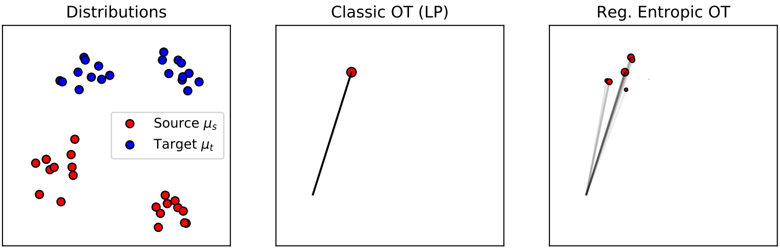


Barycentric mapping [Ferradans et al., 2014]

$$\widehat{T}_{\gamma_0}(\mathbf{x}_i^s) = \arg\min_{\mathbf{x}} \quad \sum_j \gamma_0(i,j)\|\mathbf{x} - \mathbf{x}_j^t\|^2. \tag{1}$$

- The mass of each source sample is spread onto the target samples (line of γ_0).
- The mapping is the barycenter of the target samples weighted by γ_0
- Closed form solution for the quadratic loss.
- Limited to the samples in the distribution (no out of sample).
- Trick: learn OT on few samples and apply displacement to the nearest point.

Transporting the discrete samples

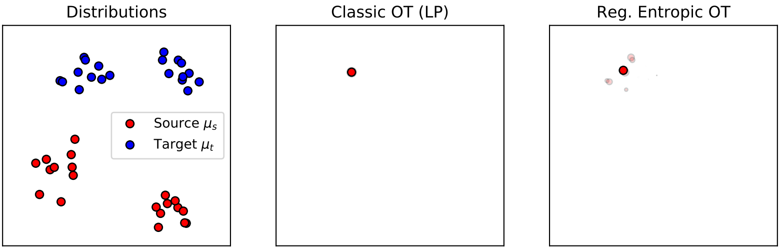


Barycentric mapping [Ferradans et al., 2014]

$$\widehat{T}_{\gamma_0}(\mathbf{x}_i^s) = \frac{1}{\sum_j \gamma_0(i,j)} \sum_j \gamma_0(i,j)\mathbf{x}_j^t. \tag{1}$$

- The mass of each source sample is spread onto the target samples (line of γ_0).
- The mapping is the barycenter of the target samples weighted by γ_0
- Closed form solution for the quadratic loss.
- Limited to the samples in the distribution (no out of sample).
- Trick: learn OT on few samples and apply displacement to the nearest point.

Transporting the discrete samples

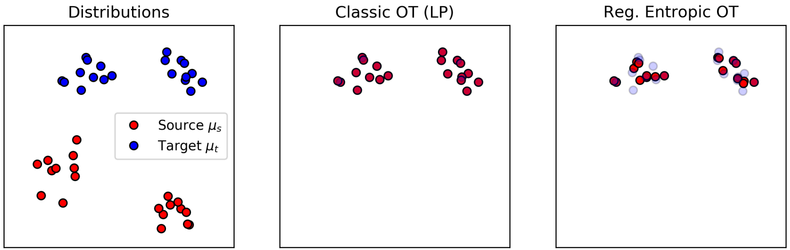


Barycentric mapping [Ferradans et al., 2014]

$$\widehat{T}_{\gamma_0}(\mathbf{x}_i^s) = \frac{1}{\sum_j \gamma_0(i,j)} \sum_j \gamma_0(i,j)\mathbf{x}_j^t. \tag{1}$$

- The mass of each source sample is spread onto the target samples (line of γ_0).
- The mapping is the barycenter of the target samples weighted by γ_0
- Closed form solution for the quadratic loss.
- Limited to the samples in the distribution (no out of sample).
- Trick: learn OT on few samples and apply displacement to the nearest point.

Transporting the discrete samples



Barycentric mapping [Ferradans et al., 2014]

$$\widehat{T}_{\gamma_0}(\mathbf{x}_i^s) = \frac{1}{\sum_j \gamma_0(i,j)} \sum_j \gamma_0(i,j)\mathbf{x}_j^t. \tag{1}$$

- The mass of each source sample is spread onto the target samples (line of γ_0).
- The mapping is the barycenter of the target samples weighted by γ_0
- Closed form solution for the quadratic loss.
- Limited to the samples in the distribution (no out of sample).
- Trick: learn OT on few samples and apply displacement to the nearest point.

Pixels as empirical distribution [Ferradans et al., 2014]

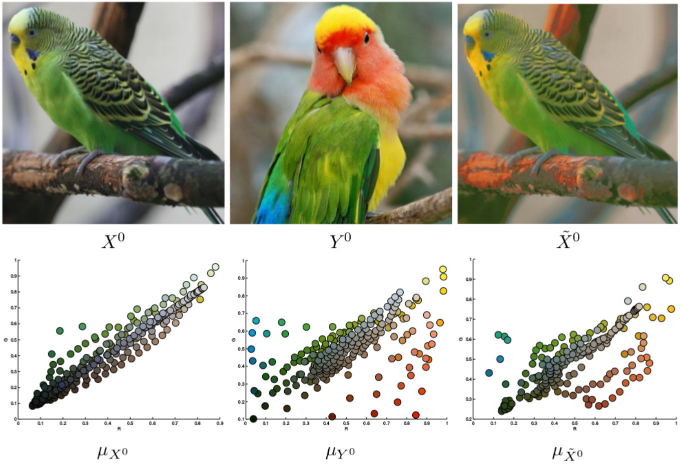
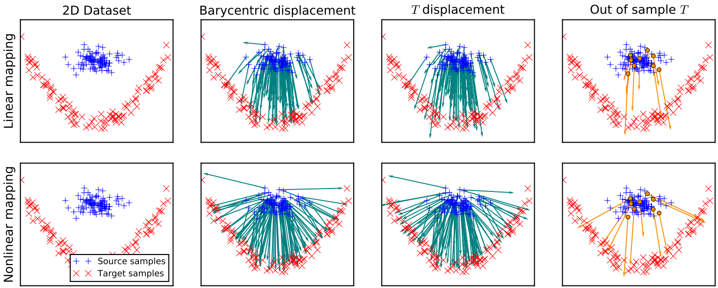


Image colorization [Ferradans et al., 2014]

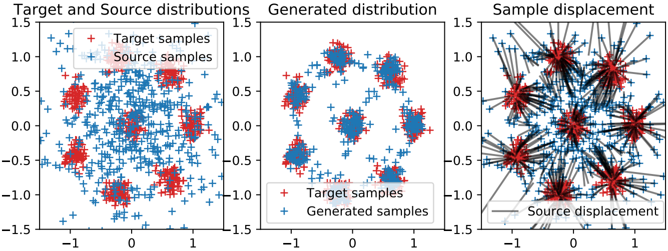
Joint OT and mapping estimation



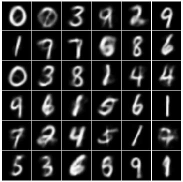
Simultaneous OT matrix and mapping [Perrot et al., 2016]

$$\min_{T, \boldsymbol{\gamma} \in \mathcal{P}} \quad \langle \boldsymbol{\gamma}, \mathbf{C} \rangle_F + \sum_i \|T(\mathbf{x}_i^s) - T_{\boldsymbol{\gamma}}(\mathbf{x}_i^s)\|^2 + \lambda \|T\|^2$$

- Estimate jointly the OT matrix and a smooth mapping approximating the barycentric mapping.
- The mapping is a regularization for OT.
- Controlled generalization error (statistical bound).
- Linear and kernel mappings T, limited to small scale datasets.

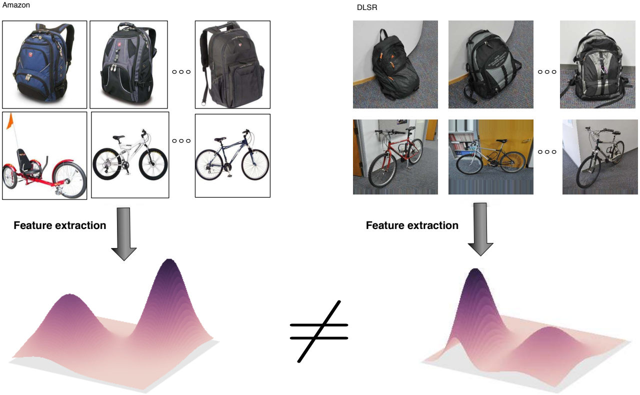


Large scale mapping estimation [Seguy et al., 2017]



- 2-step procedure:
  1. Stochastic estimation of regularized $\hat{\gamma}$.
  2. Stochastic estimation of f with a neural
- OT solved with Stochastic Gradient Ascent in the dual.
- Convergence to the true OT and mapping for small regularization.

Domain Adaptation problem

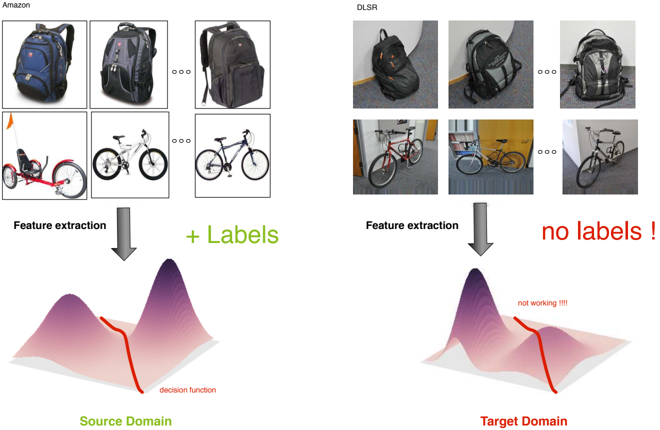


Probability Distribution Functions over the domains

Our context

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

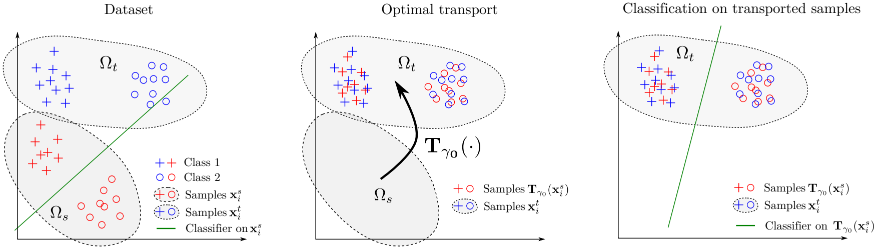
Unsupervised domain adaptation problem



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

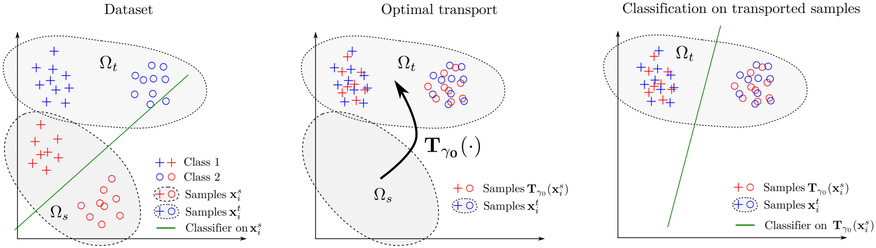
OT for domain adaptation : Step 1



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., 2016].
- 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



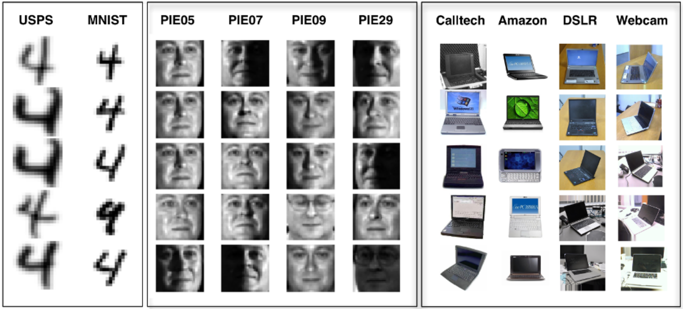
Step 2 : Transport the training samples onto the target distribution.

- The mass of each source sample is spread onto the target samples (line of γ_0).
- Transport using barycentric mapping [Ferradans et al., 2014].
- The mapping can be estimated for out of sample prediction [Perrot et al., 2016, Seguy et al., 2017].

Step 3 : Learn a classifier on the transported training samples

- Transported sample keep their labels.
- Classic ML problem when samples are well transported.

Visual adaptation datasets



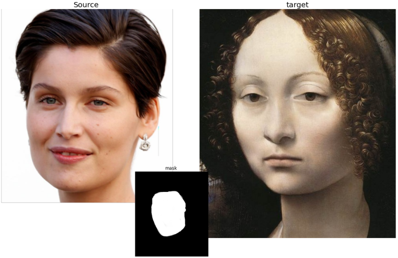
Datasets

- **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

- State of the art performances on the 3 datasets.
- Works well on deep features adaptation and extension to semi-supervised DA.

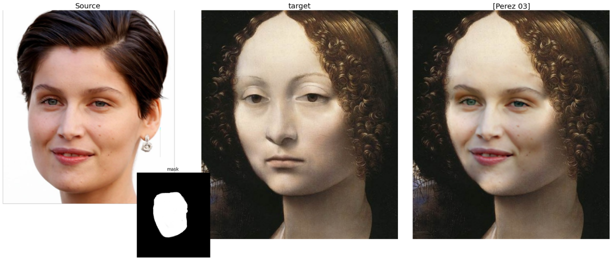
Seamless copy in images



Poisson image editing [Pérez et al., 2003]

- Use the color gradient from the source image.
- Use color border conditions on the target image.
- Solve Poisson equation to reconstruct the new image.

Seamless copy in images



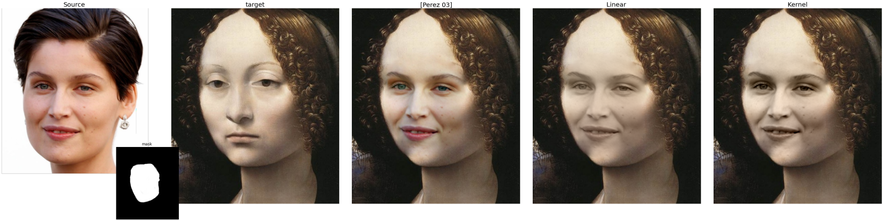
Poisson image editing [Pérez et al., 2003]

- Use the color gradient from the source image.
- Use color border conditions on the target image.
- Solve Poisson equation to reconstruct the new image.

Seamless copy with gradient adaptation [Perrot et al., 2016]

- Transport the gradient from the source to target color gradient distribution.
- Solve the Poisson equation with the mapped source gradients.
- Better respect of the color dynamic and limits false colors.

Seamless copy in images



Poisson image editing [Pérez et al., 2003]

- Use the color gradient from the source image.
- Use color border conditions on the target image.
- Solve Poisson equation to reconstruct the new image.

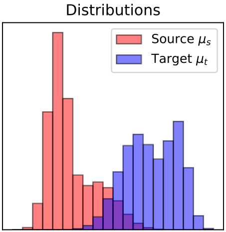
Seamless copy with gradient adaptation [Perrot et al., 2016]

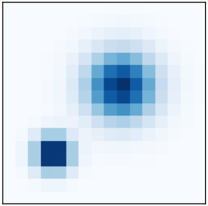
- Transport the gradient from the source to target color gradient distribution.
- Solve the Poisson equation with the mapped source gradients.
- Better respect of the color dynamic and limits false colors.

Seamless copy with gradient adaptation

Learning from histograms with Optimal Transport

Learning from histograms

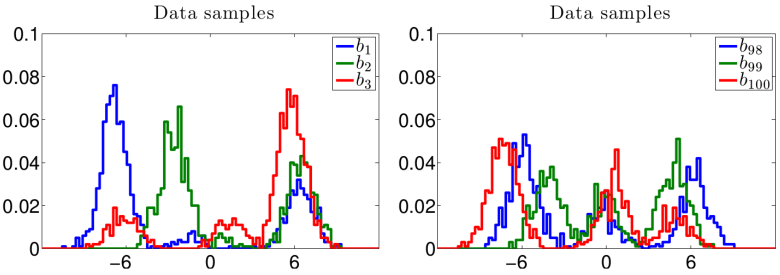




Data as histograms

- Fixed bin positions $\mathbf{x}_i$ e.g. grid, simplex $\Delta = \left\{(\mu_i)_i \geq 0; \sum_i \mu_i = 1\right\}$
- A lot of datasets comes under the form of histograms.
- Images are photo counts (black and white), text as word counts.
- Natural divergence is Kullback–Leibler.
- Not all data can be seen as histograms (positivity+constant mass)!

Dictionary learning on histograms

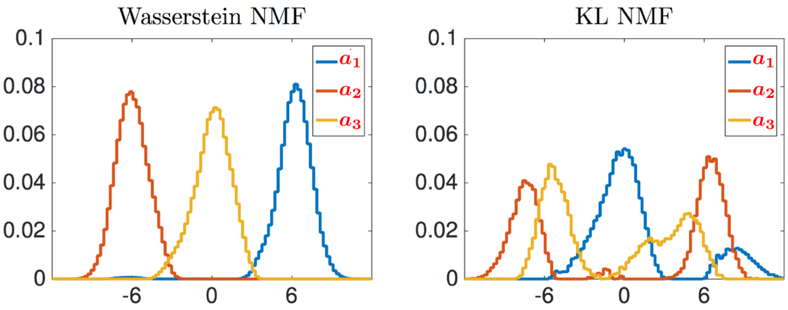


DL with Wasserstein distance [Sandler and Lindenbaum, 2011]

$$\min_{\mathbf{D},\mathbf{H}} \quad \sum_i W_{\mathbf{C}}(\mathbf{v}_i, \mathbf{D}\mathbf{h}_i)$$

- NMF: columns of $\mathbf{D}$ and $\mathbf{H}$ are on the simplex.
- Metric $\mathbf{C}$ can encode spatial relations btewwen the bins of the histograms.
- Ground metric learning [Zen et al., 2014].
- Fast DL with regularized OT [Rolet et al., 2016].

Dictionary learning on histograms

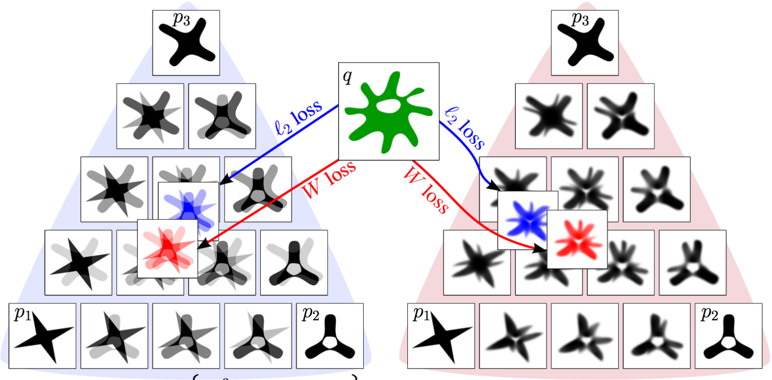


DL with Wasserstein distance [Sandler and Lindenbaum, 2011]

$$\min_{\mathbf{D},\mathbf{H}} \quad \sum_i W_{\mathbf{C}}(\mathbf{v}_i, \mathbf{D}\mathbf{h}_i)$$

- NMF: columns of $\mathbf{D}$ and $\mathbf{H}$ are on the simplex.
- Metric $\mathbf{C}$ can encode spatial relations btewwen the bins of the histograms.
- Ground metric learning [Zen et al., 2014].
- Fast DL with regularized OT [Rolet et al., 2016].

Wasserstein dictionary learning

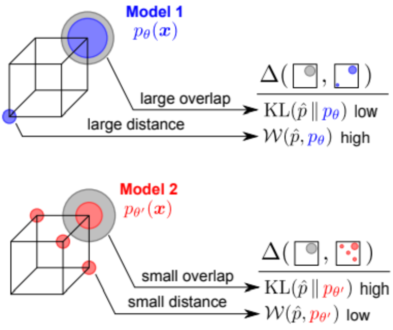


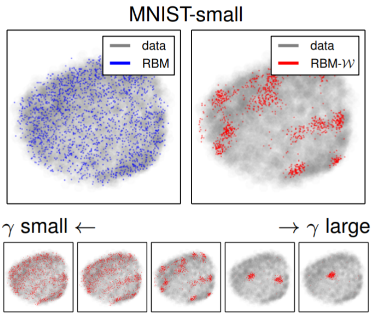
Euclidean Simplex: $\left\{\sum_{i=1}^{3} \lambda_i p_i, \lambda \in \Sigma_3\right\}$ Wasserstein simplex: $\{P(\lambda), \lambda \in \Sigma_3\}$

Nonlinear unmixing with Wasserstein simplex [Schmitz et al., 2017]

- Linear model is a barycenter for the squared ℓ_2 distance.
- Use Wassersyein barycenter for modeling.

Training Restricted Boltzman Machine with Wasserstein





Wasserstein training of RBM [Montavon et al., 2016]

- Use Wassersteien instead of KL for training RBM.
- Estimation of RBM generative models $p_\theta(\mathbf{x})$.
- Used for completion or denoising.

Multi-label learning with Wasserstein Loss

Siberian husky

Eskimo dog

Flickr : street, parade, dragon
Prediction : people, protest, parade

Flickr : water, boat, ref ection, sun-shine
Prediction : water, river, lake, summer;

Learning with a Wasserstein Loss [Frogner et al., 2015]

$$\min_{f} \quad \sum_{k=1}^{N} W_1^1(f(\mathbf{x}_i), \mathbf{l}_i)$$

- Empirical loss minimization with Wasserstein loss.
- Multi-label prediction (labels $\mathbf{l}$ seen as histograms, f output softmax).
- Cost between labels can encode semantic similarity between classes.
- Good performances in image tagging.

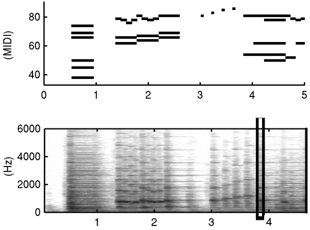
Linear unmixing

$$\min_{\mathbf{h}\in\Delta} \quad W_{\mathbf{C}}(\mathbf{v}, \mathbf{D}\mathbf{h}) \tag{2}$$

- Δ is the probability simplex (positivity, sum to one).
- $\mathbf{v}$ is the observation, $\mathbf{D}$ the dictionary, $\mathbf{h}$ the mixing coefficients.
- Supervised when the dictionary is known designed.
- Classical problem in remote sensing, signal processing.

Musical spectral unmixing

- State of the art: KL $+$ designed dictionary.
- Spectra with harmonic structure.
- Variability in the fundamental frequency.
- Variability in the magnitude of the harmonics.



$\Rightarrow$ Optimal spectral transportation [Flamary et al., 2016b].

Linear unmixing with optimal transport

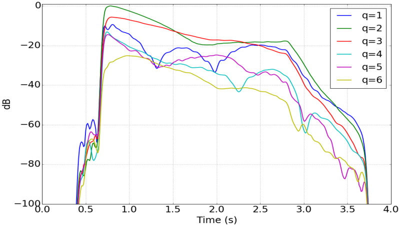
Linear unmixing

$$\min_{\mathbf{h}\in\Delta} \quad W_{\mathbf{C}}(\mathbf{v}, \mathbf{D}\mathbf{h}) \tag{2}$$

- Δ is the probability simplex (positivity, sum to one).
- $\mathbf{v}$ is the observation, $\mathbf{D}$ the dictionary, $\mathbf{h}$ the mixing coefficients.
- Supervised when the dictionary is known designed.
- Classical problem in remote sensing, signal processing.

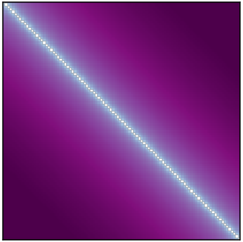
Musical spectral unmixing

- State of the art: KL + designed dictionary.
- Spectra with harmonic structure.
- Variability in the fundamental frequency.
- Variability in the magnitude of the harmonics.



⇒ Optimal spectral transportation [Flamary et al., 2016b].

Quadratic cost **C** (log)



Quadratic cost between frequencies

- Allows small shift in frequencies.
- Very sensitive to harmonics magnitude.

Harmonic invariant cost

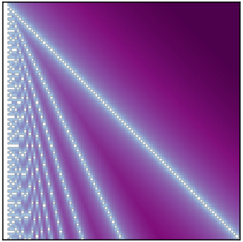
$$c_{ij} = \min_{q=1,\dots,\left\lceil \frac{f_i}{f_j} \right\rceil} (f_i - q f_j)^2 + \epsilon\, \delta_{q \neq 1},$$

- Allow mass transfer between harmonics.
- $\epsilon > 0$ discriminates between octaves.

Solving the optimization problem

- A good invariant cost allows for extremely simple dictionary elements (diracs on the fundamental frequency).
- We take $\mathbf{D}$ as diracs on the fundamental frequencies of the notes.
- Closed form for solving the OT problem.
- Non-convex Group lasso for sparse estimates and/or entropic regularization.

Optimal spectral transportation (OST)



Quadratic cost between frequencies

- Allows small shift in frequencies.
- Very sensitive to harmonics magnitude.

Harmonic invariant cost

$$c_{ij} = \min_{q=1,\ldots,\left\lceil \frac{f_i}{f_j} \right\rceil} (f_i - q f_j)^2 + \epsilon\, \delta_{q \neq 1},$$

- Allow mass transfer between harmonics.
- $\epsilon > 0$ discriminates between octaves.

Solving the optimization problem

- A good invariant cost allows for extremely simple dictionary elements (diracs on the fundamental frequency).
- We take $\mathbf{D}$ as diracs on the fundamental frequencies of the notes.
- Closed form for solving the OT problem.
- Non-convex Group lasso for sparse estimates and/or entropic regularization.

OST in action

Simulated data

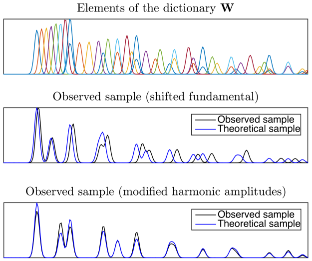
- Robust to shifted fundamental frequency.
- Robust to harmonics magnitude variability.
- Very fast ($\sim$ms per frame).

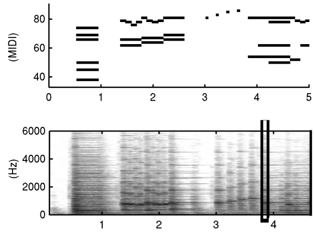
MAPS Dataset [Emiya et al., 2010]

- Several piano sequence from classical music ($m = 60$ notes)
- Comparison with ground truth given as MIDI.
- OST similar of better than KL+Dico while ≥ 70 times quicker.

Real time demonstration

- Python+Pygame implementation.
- `https://github.com/rflamary/OST`





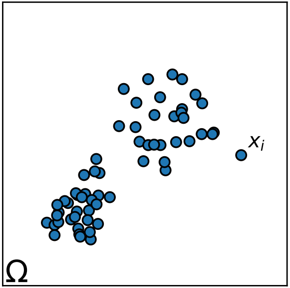
Learning from empirical distributions with Optimal Transport

Empirical distributions A.K.A datasets

$$\mu = \sum_{i=1}^{n} \mu_i \delta_{\mathbf{x}_i}, \quad \mathbf{x}_i \in \Omega, \quad \sum_{i=1}^{n} \mu_i = 1$$

Empirical distribution

- Two realizations never overlap.
- Training base of all machine learning approaches.
- How to measure discrepancy?
- Maximum Mean Discrepancy (ℓ_2 after convolution).
- Wasserstein distance.

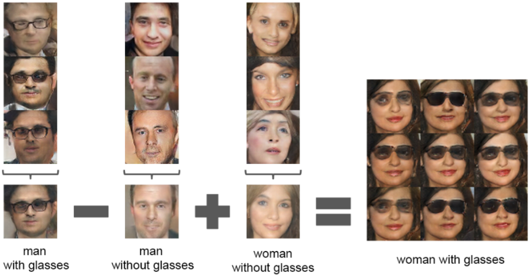


Generative Adversarial Networks (GAN)

Generative Adversarial Networks (GAN) [Goodfellow et al., 2014]

$$\min_G \max_D \quad E_{\mathbf{x}\sim\mu_d}[\log D(\mathbf{x})] + E_{\mathbf{z}\sim\mathcal{N}(0,\mathbf{I})}[\log(1 - D(G(\mathbf{z})))]$$

- Learn a generative model G that outputs realistic samples from data μ_d.
- Learn a classifier D to discriminate between the generated and true samples.
- Make those models compete (Nash equilibrium [Zhao et al., 2016]).
- Generator space has semantic meaning [Radford et al., 2015].
- But extremely hard to train (vanishing gradients).

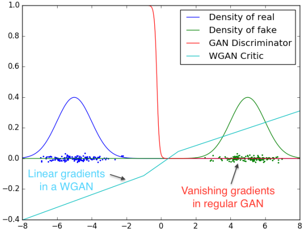


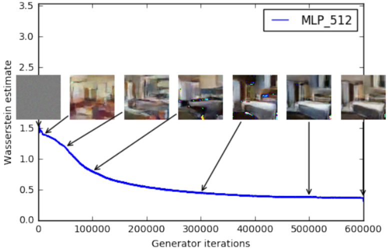
Generative Adversarial Networks (GAN) [Goodfellow et al., 2014]

$$\min_G \max_D \quad E_{\mathbf{x}\sim\mu_d}[\log D(\mathbf{x})] + E_{\mathbf{z}\sim\mathcal{N}(0,\mathbf{I})}[\log(1 - D(G(\mathbf{z})))]$$

- Learn a generative model G that outputs realistic samples from data μ_d.
- Learn a classifier D to discriminate between the generated and true samples.
- Make those models compete (Nash equilibrium [Zhao et al., 2016]).
- Generator space has semantic meaning [Radford et al., 2015].
- But extremely hard to train (vanishing gradients).

Wasserstein Generative Adversarial Networks (WGAN)





Wasserstein GAN [Arjovsky et al., 2017]

$$\min_{G} \quad W_1^1(G(\mathbf{z}), \mu_d), \quad \text{s.t. } \mathbf{z} \sim \mathcal{N}(0, \mathbf{I}) \tag{3}$$

- Minimizes the Wasserstein distance between the data and the generated data.
- No vanishing gradients ! Far better convergence in practice.
- Wasserstein in the dual (separable w.r.t. the samples).

$$\min_{G} \sup_{\phi \in \mathsf{Lip}^1} \quad \mathbb{E}_{\mathbf{x}\sim\mu_d}[\phi(\mathbf{x})] - \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(0,\mathbf{I})}[\phi(G(\mathbf{z}))]$$

- ϕ is a neural network that acts as an *actor critic*

WGAN: the devil in the approximation

Neural network belonging to Lip^1 ?

- Not really! [Arjovsky et al., 2017] proposes to do weight clipping that force an upper bound on the Lipschitz constant.
- It is actually the supremum over K-Lipschitz functions that is approximated by a neural network

$$\max_{f \in \text{NN class}} L_{WGAN}(f, G) \leq \sup_{\|\phi\|_L \leq K} L_{WGAN}(\phi, G) \quad = \quad K \cdot W_1^1(G(\mathbf{z}), \mu_d)$$

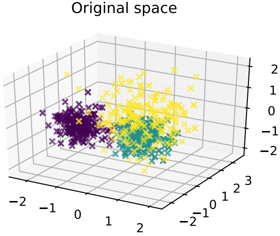
- Actually **not** equivalent to solve the optimal transport, but gradients are aligned.

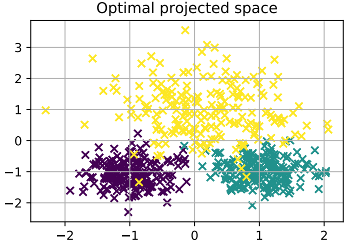
Improved WGAN [Gulrajani et al., 2017]

$$\min_G \sup_{f \in \text{NN class}} \mathbb{E}_{\mathbf{x}\sim\mu_d}[f(\mathbf{x})] - \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(0,\mathbf{I})}[f(G(\mathbf{z}))] + \lambda \mathbb{E}_{\mathbf{x}\sim\mu_d}[(||\nabla f(\mathbf{x})||_2 - 1)^2]$$

Relaxation of the constraint (for W_1 the gradient of the potential is 1 almost everywhere).

Wasserstein Discriminant Analysis (WDA)



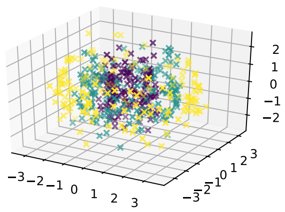


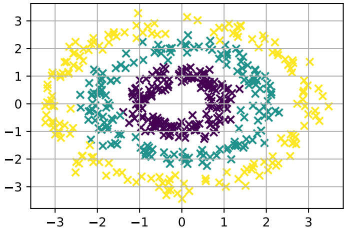
$$\max_{\mathbf{P}\in\mathcal{S}} \quad \frac{\sum_{c,c'>c} W_\lambda(\mathbf{P}\mathbf{X}^c, \mathbf{P}\mathbf{X}^{c'})}{\sum_c W_\lambda(\mathbf{P}\mathbf{X}^c, \mathbf{P}\mathbf{X}^c)} \quad (4)$$

- $\mathbf{X}^c$ are samples from class c.
- $\mathbf{P}$ is an orthogonal projection;

- Converges to Fisher Discriminant when $\lambda \to \infty$.
- Non parametric method that allows nonlinear discrimination.
- Problem solved with gradient ascent in the Stiefel manifold $\mathcal{S}$.
- Gradient computed using automatic differentiation of Sinkhorn algorithm.

Wasserstein Discriminant Analysis (WDA)





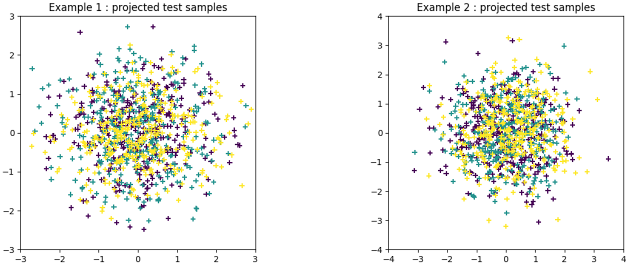
$$\max_{\mathbf{P}\in\mathcal{S}} \quad \frac{\sum_{c,c'>c} W_\lambda(\mathbf{PX}^c, \mathbf{PX}^{c'})}{\sum_c W_\lambda(\mathbf{PX}^c, \mathbf{PX}^c)} \quad (4)$$

- $\mathbf{X}^c$ are samples from class c.
- $\mathbf{P}$ is an orthogonal projection;

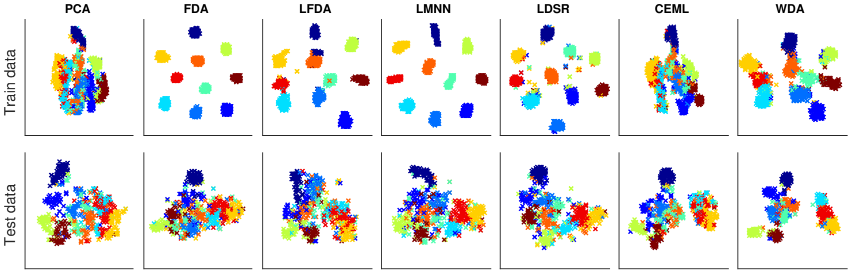
- Converges to Fisher Discriminant when $\lambda \to \infty$.
- Non parametric method that allows nonlinear discrimination.
- Problem solved with gradient ascent in the Stiefel manifold $\mathcal{S}$.
- Gradient computed using automatic differentiation of Sinkhorn algorithm.

WDA in action

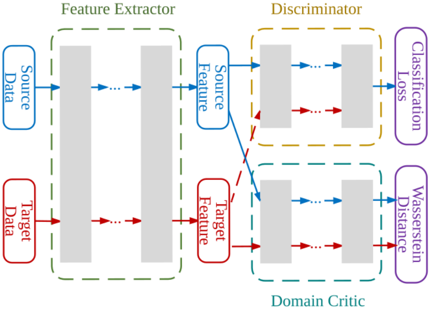
Simulated datasets : 10→2

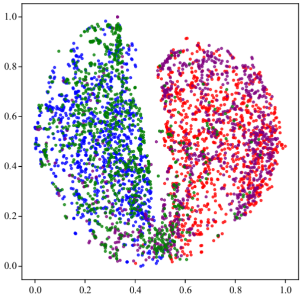


MNIST Dataset: 784→10(→2 TSNE)



Domain adaptation with Wasserstein distance





(d) t-SNE of WDGRL features

Domain adaptation for deep learning [Shen et al., 2018]

- Modern DA aim at aligning source and target in the deep representation : DANN [Ganin et al., 2016], MMD [Tzeng et al., 2014], CORAL [Sun and Saenko, 2016].
- Wasserstein distance used as objective for the adaptation [Shen et al., 2018].

Joint Distribution Optimal Transport for DA

Learning with JDOT [Courty et al., 2017]

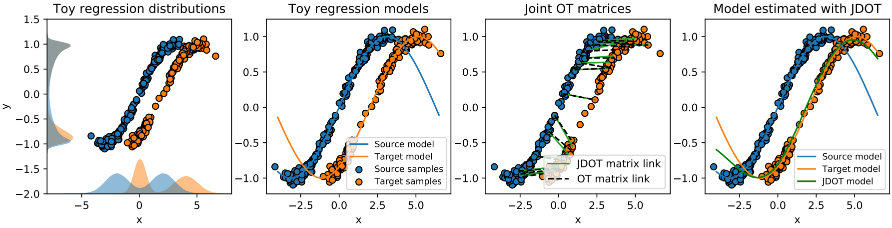
$$\min_{f} \quad \left\{ W_1(\hat{\mathcal{P}}_s, \hat{\mathcal{P}_t}^f) = \inf_{\boldsymbol{\gamma}\in\Pi} \sum_{ij} \mathcal{D}(\mathbf{x}_i^s, y_i^s; \mathbf{x}_j^t, f(\mathbf{x}_j^t))\boldsymbol{\gamma}_{ij} \right\} \tag{5}$$

- $\hat{\mathcal{P}_t}^f = \frac{1}{N_t}\sum_{i=1}^{N_t} \delta_{\mathbf{x}_i^t, f\mathbf{x}_i^t}$ is the proxy joint feature/label distribution.
- Π is the transport polytope, $\hat{\mathcal{P}}_s$ the empirical source distribution.
- $\mathcal{D}(\mathbf{x}_i^s, y_i^s; \mathbf{x}_j^t, f(\mathbf{x}_j^t)) = \alpha\|\mathbf{x}_i^s - \mathbf{x}_j^t\|^2 + \mathcal{L}(y_i^s, f(\mathbf{x}_j^t))$ with $\alpha > 0$.
- We search for the predictor f that better align the joint distributions.
- JDOT can be seen as minimizing a generalization bound.

Optimizing JDOT

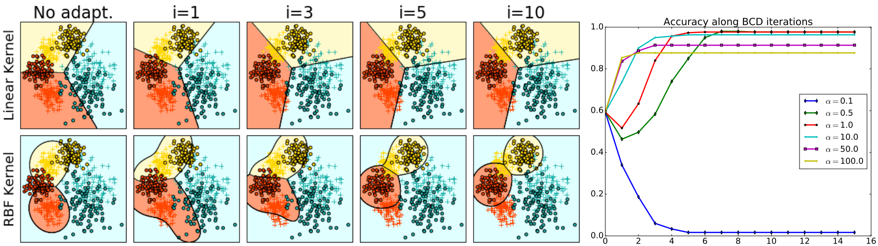
- Can be solved by block coordinate descent $(f, \boldsymbol{\gamma})$ [Courty et al., 2017].
- Solving with fixed f is classical OT.
- Solving with fixed $\boldsymbol{\gamma}$ is weighted empirical loss minimization.

JDOT in action



Numerical experiments

- Examples on toy regression and classification problems.
- State of the art in Visual adaptation (Caltech/office), review score prediction (Amazon) and Wifi localization.
- Works very well but limited to small datasets.
- OT performed with euclidean distance in the feature space.



Numerical experiments

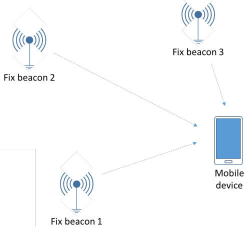
- Examples on toy regression and classification problems.
- State of the art in Visual adaptation (Caltech/office), review score prediction (Amazon) and Wifi localization.
- Works very well but limited to small datasets.
- OT performed with euclidean distance in the feature space.

JDOT in action

☆☆☆☆☆ **Excellent product that I completely hate**, Apr 1, 2013
By Thirsty - See all my reviews
This review is from: Strollmaster 3000 (Baby Product)

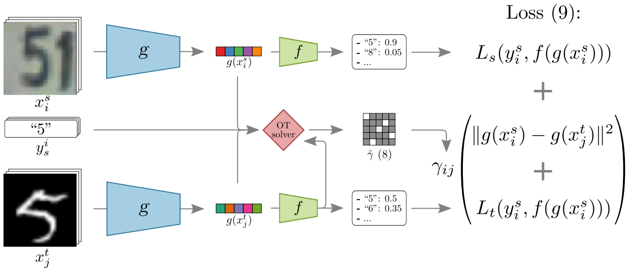
The Strollmaster 3000 is every parent's dream - roomy, durable, safe, and easy to fold, with a unique 17-point harness. Best yet, it weighs just 1.6 lbs. and sells for an unbelievable $17.99. Unfortunately, it has one fatal flaw - the cupholder can only handle beverages up to 64 oz. I was dumbstruck as well. Is this America? I was left holding my 128 oz. Big Gulp like some kind of sucker. So, if you're into amazing, durable products that are a steal and virtually idyllic, then, sure, buy it. If you want to down a bathtub of Dr. Pepper, though. I'd pass.

☆☆☆☆☆ **Let it go... in the trash**
By VPI1977 on December 24, 2017
Had high expectation, too much snow, too many animals, wish it had more ninjas. Also it would be better if these people ate more, I mean how are we suppose to make society better if people don't sit down to eat and socialize.
2 people found this helpful



Numerical experiments

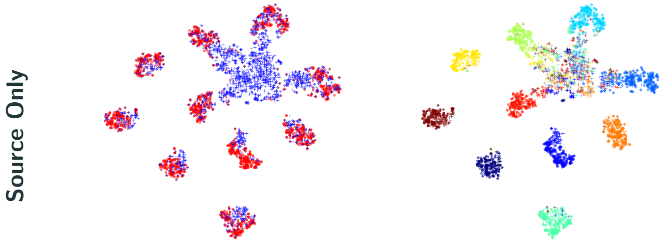
- Examples on toy regression and classification problems.
- State of the art in Visual adaptation (Caltech/office), review score prediction (Amazon) and Wifi localization.
- Works very well but limited to small datasets.
- OT performed with euclidean distance in the feature space.



DeepJDOT [Damodaran et al., 2018]

- Learn simultaneously the embedding g and the classifier f.
- JDOT performed in the joint embedding/label space.
- Use minibatch to estimate OT and update g, f at each iterations.
- Scales to large datasets and estimate a representation for both domains.
- TSNE projections of embeddings (MNIST→MNIST-M).

JDOT for large scale deep learning



DeepJDOT [Damodaran et al., 2018]

- Learn simultaneously the embedding g and the classifier f.
- JDOT performed in the joint embedding/label space.
- Use minibatch to estimate OT and update g, f at each iterations.
- Scales to large datasets and estimate a representation for both domains.
- TSNE projections of embeddings (MNIST$\rightarrow$MNIST-M).

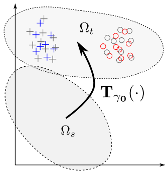
JDOT for large scale deep learning

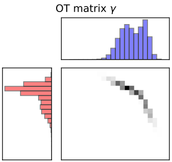
DeepJDOT [Damodaran et al., 2018]

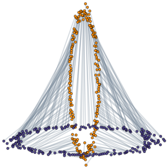
- Learn simultaneously the embedding g and the classifier f.
- JDOT performed in the joint embedding/label space.
- Use minibatch to estimate OT and update g, f at each iterations.
- Scales to large datasets and estimate a representation for both domains.
- TSNE projections of embeddings (MNIST→MNIST-M).

Conclusion

Optimal transport for machine learning





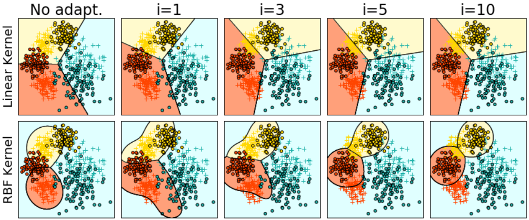


Mapping with optimal transport

- Optimal displacement from one distribution to another.
- Can estimate smooth mapping for out of sample displacement.
- Domain, color and gradient adaptation, transfer learning.

Learning with optimal transport

- Natural divergence for machine learning and estimation.
- Cost encode complex relations in an histogram.
- Regularization is the key (performance, smoothness).
- Recent optimization procedures opened it to medium/large scale datasets.
- Sensible loss between non overlapping distributions.
- Works with both histograms and empirical distributions.



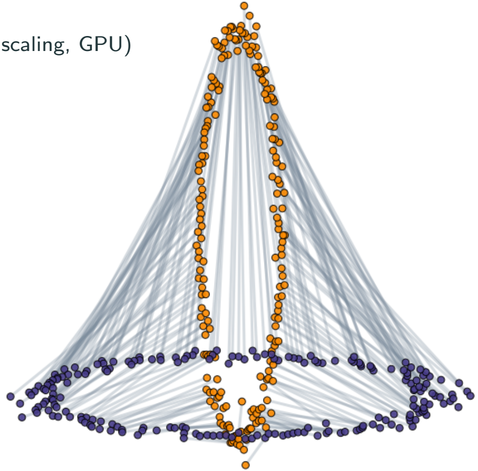
Open questions

- Generalization bounds for learning with OT.
- Concentration inequalities of regularized OT.
- Learning the ground metric (supervised, unsupervised, adversarial?).
- Large scale OT and mapping estimation, accelerated stochastic optimization.

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.
- Wasserstein Discriminant Analysis.



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

Post docs available in:
Nice, Rouen, Rennes (France)

References I

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

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

Courty, N., Flamary, R., Habrard, A., and Rakotomamonjy, A. (2017).
Joint distribution optimal transportation for domain adaptation.
In *Neural Information Processing Systems (NIPS)*.

References II

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

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

Damodaran, B. B., Kellenberger, B., Flamary, R., Tuia, D., and Courty, N. (2018).
Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation.

References III

Emiya, V., Badeau, R., and David, B. (2010).
Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle.
IEEE Transactions on Audio, Speech, and Language Processing, 18(6):1643–1654.

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

Flamary, R., Cuturi, M., Courty, N., and Rakotomamonjy, A. (2016a).
Wasserstein discriminant analysis.
arXiv preprint arXiv:1608.08063.

References IV

Flamary, R., Fevotte, C., Courty, N., and Emyia, V. (2016b).
Optimal spectral transportation with application to music transcription.
In *Neural Information Processing Systems (NIPS)*.

Frogner, C., Zhang, C., Mobahi, H., Araya, M., and Poggio, T. A. (2015).
Learning with a wasserstein loss.
In *Advances in Neural Information Processing Systems*, pages 2053–2061.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. (2016).
Domain-adversarial training of neural networks.
Journal of Machine Learning Research, 17(59):1–35.

References V

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014).
Generative adversarial nets.
In *Advances in neural information processing systems*, pages 2672–2680.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. (2017).
Improved training of wasserstein gans.
NIPS.

Montavon, G., Müller, K.-R., and Cuturi, M. (2016).
Wasserstein training of restricted boltzmann machines.
In *Advances in Neural Information Processing Systems*, pages 3718–3726.

References VI

Pérez, P., Gangnet, M., and Blake, A. (2003).
Poisson image editing.
ACM Trans. on Graphics, 22(3).

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

Radford, A., Metz, L., and Chintala, S. (2015).
Unsupervised representation learning with deep convolutional generative adversarial networks.
arXiv preprint arXiv:1511.06434.

References VII

Rolet, A., Cuturi, M., and Peyré, G. (2016).
Fast dictionary learning with a smoothed wasserstein loss.
In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 630–638.

Sandler, R. and Lindenbaum, M. (2011).
Nonnegative matrix factorization with earth mover's distance metric for image analysis.
IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(8):1590–1602.

References VIII

Schmitz, M. A., Heitz, M., Bonneel, N., Mboula, F. M. N., Coeurjolly, D., Cuturi, M., Peyré, G., and Starck, J.-L. (2017).
Wasserstein dictionary learning: Optimal transport-based unsupervised non-linear dictionary learning.
arXiv preprint arXiv:1708.01955.

Seguy, V., Bhushan Damodaran, B., Flamary, R., Courty, N., Rolet, A., and Blondel, M. (2017).
Large-scale optimal transport and mapping estimation.

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

References IX

Sun, B. and Saenko, K. (2016).
Deep CORAL: Correlation Alignment for Deep Domain Adaptation, pages 443–450.
Springer International Publishing, Cham.

Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., and Darrell, T. (2014).
Deep domain confusion: Maximizing for domain invariance.
arXiv preprint arXiv:1412.3474.

Zen, G., Ricci, E., and Sebe, N. (2014).
Simultaneous ground metric learning and matrix factorization with earth mover's distance.
In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 3690–3695.

References X

Zhao, J., Mathieu, M., and LeCun, Y. (2016).
Energy-based generative adversarial network.
arXiv preprint arXiv:1609.03126.