Stochastic Methods for Optimal Transport and Applications in Machine Learning

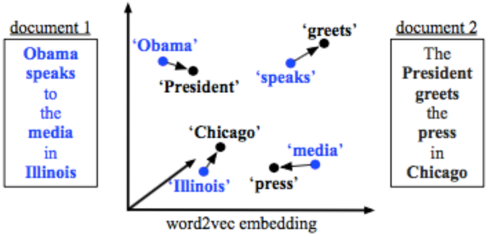
Aude Genevay

CEREMADE - Université Paris Dauphine
INRIA - Mokaplan project-team
DMA - Ecole Normale Supérieure

Journées IOPS - Juillet 2017

Joint work with F. Bach, M.Cuturi, G. Peyré

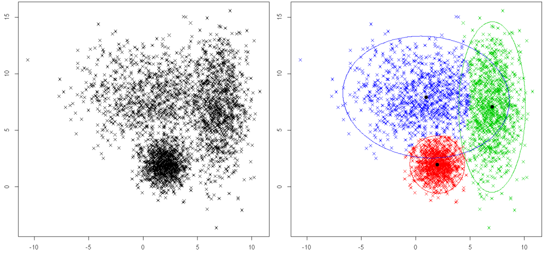
Motivations : Large-Scale Discrete Optimal Transport



- Document d_i = histogram of words
- Word w_k = point in $\mathbb{R}^d$ for a certain embedding (usually learnt with neural networks, e.g. Word2Vec)
- Document $\sim$ weighted cloud of points in $\mathbb{R}^d \Rightarrow d_i \sim \mu_i = \sum \alpha_{k,i} \delta_{w_k}$
- Distance between 2 documents d_1, d_2 is the optimal transport distance between the associated point clouds μ_1, μ_2.

Motivations : Semi-Discrete Optimal Transport

- Given a cloud of points $(x_1, \ldots, x_M)$ in $\mathbb{R}^d$
- We want to fit a (parametric) statistical model to this cloud : we choose a family of probability measures with parametric densities $\mathrm{d}\mu(x, \theta) = f(x, \theta)\mathrm{d}x$
- Find θ that minimizes the optimal transport distance between μ and $\nu = \sum_{i=1}^{N} \frac{1}{N}\delta_{x_i}$

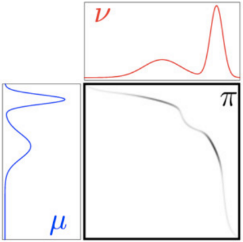


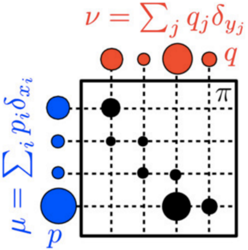
Optimal Transport

Two positive Radon measures μ on $\mathcal{X}$ and ν on $\mathcal{Y}$ of mass 1
Cost $c(x, y)$ to move a unit of mass from x to y
Set of couplings with marginals μ and ν

$$\Pi(\mu,\nu) \stackrel{\text{def.}}{=} \{\pi \in \mathcal{M}_+^1(\mathcal{X}\times\mathcal{Y}) \mid \pi(A\times\mathcal{Y}) = \mu(A), \pi(\mathcal{X}\times B) = \nu(B)\}$$

What's the coupling that minimizes the total cost?





Kantorovitch Formulation of OT

The optimal overall cost for transporting μ to ν is given by

$$W(\mu, \nu) = \min_{\pi \in \Pi(\mu,\nu)} \int_{\mathcal{X}\times\mathcal{Y}} c(x, y) \mathrm{d}\pi(x, y) \qquad (\mathcal{P}_\varepsilon)$$

Kantorovitch Formulation of OT

The optimal overall cost for transporting μ to ν is given by

$$W_\varepsilon(\mu,\nu) = \min_{\pi \in \Pi(\mu,\nu)} \int_{\mathcal{X}\times\mathcal{Y}} c(x,y)\mathrm{d}\pi(x,y) + \varepsilon\,\mathrm{KL}(\pi|\mu\otimes\nu) \qquad (\mathcal{P}_\varepsilon)$$

where

$$\mathrm{KL}(\pi|\mu\otimes\nu) \overset{\text{def.}}{=} \int_{\mathcal{X}\times\mathcal{Y}} \big(\log\big(\frac{\mathrm{d}\pi}{\mathrm{d}\mu\mathrm{d}\nu}(x,y)\big) - 1\big)\mathrm{d}\pi(x,y)$$

Kantorovitch Formulation of OT

The optimal overall cost for transporting μ to ν is given by

$$W_\varepsilon(\mu,\nu) = \min_{\pi\in\Pi(\mu,\nu)} \int_{\mathcal{X}\times\mathcal{Y}} c(x,y)\mathrm{d}\pi(x,y) + \varepsilon\,\mathrm{KL}(\pi|\mu\otimes\nu) \qquad (\mathcal{P}_\varepsilon)$$

where

$$\mathrm{KL}(\pi|\mu\otimes\nu) \overset{\text{def.}}{=} \int_{\mathcal{X}\times\mathcal{Y}} \Big(\log\Big(\frac{\mathrm{d}\pi}{\mathrm{d}\mu\mathrm{d}\nu}(x,y)\Big) - 1\Big)\mathrm{d}\pi(x,y)$$

Adding an entropic regularization smoothes the constraint. In particular it yields an unconstrained dual problem.

Reminder on convex duality

Primal problem:

$$\begin{array}{ll} \min_x & f(x) \\ \text{subject to} & h_i(x) = 0 \quad \text{for i} = 1 \ldots \text{m} \end{array}$$

Lagrange dual function:

$$g(\lambda) = \min_x f(x) + \sum_{i=1}^{m} \lambda_i h_i(x)$$

Dual problem:

$$\max_\lambda g(\lambda)$$

Under good assumptions, both problems are equivalent.

Dual formulation of OT

$$W(\mu, \nu) = \max_{u \in \mathcal{C}(\mathcal{X}), v \in \mathcal{C}(\mathcal{Y})} \int_{\mathcal{X}} u(x) \mathrm{d}\mu(x) + \int_{\mathcal{Y}} v(y) \mathrm{d}\nu(y) - \iota_{U_c}(u, v) \quad (\mathcal{D}_\varepsilon)$$

where the constraint set U_c is defined by

$$U_c \stackrel{\text{def.}}{=} \{(u, v) \in \mathcal{C}(\mathcal{X}) \times \mathcal{C}(\mathcal{Y}) \; ; \; \forall (x, y) \in \mathcal{X} \times \mathcal{Y}, u(x) + v(y) \leq c(x, y)\}$$

Dual formulation of OT (with entropy)

$$W_\varepsilon(\mu, \nu) = \max_{u \in \mathcal{C}(\mathcal{X}), v \in \mathcal{C}(\mathcal{Y})} \int_{\mathcal{X}} u(x) \mathrm{d}\mu(x) + \int_{\mathcal{Y}} v(y) \mathrm{d}\nu(y) - \iota^{\varepsilon}_{U_c}(u, v)$$

and the smoothed indicator is

$$\iota^{\varepsilon}_{U_c}(u, v) \overset{\text{def.}}{=} \varepsilon \int_{\mathcal{X} \times \mathcal{Y}} \exp\left(\frac{u(x) + v(y) - c(x, y)}{\varepsilon}\right) \mathrm{d}\mu(x) \mathrm{d}\nu(y)$$

Semi-Dual formulation of OT

The dual problem is convex in u and v. We fix v and minimize over u.

Semi-Dual formulation of OT

The dual problem is convex in u and v. We fix v and minimize over u. This yields :

$$u(x) \stackrel{\text{def.}}{=} \min_{y \in \mathcal{Y}} c(x, y) - v(y)$$

Semi-Dual formulation of OT

The dual problem is convex in u and v. We fix v and minimize over u. This yields :

$$u(x) \stackrel{\text{def.}}{=} \min_{y \in \mathcal{Y}} c(x,y) - v(y)$$

Plugging back in the dual :

$$\begin{aligned} W_\varepsilon(\mu,\nu) &= \max_{v \in \mathcal{C}(\mathcal{Y})} \int_{\mathcal{X}} \min_{y \in \mathcal{Y}} \left(c(x,y) - v(y)\right) \mathrm{d}\mu(x) + \int_{\mathcal{Y}} v(y) \mathrm{d}\nu(y) - \varepsilon \\ &= \max_{v \in \mathcal{C}(\mathcal{Y})} \mathbb{E}_\mu \Big[\min_{y \in \mathcal{Y}} \left(c(x,y) - v(y)\right) + \int_{\mathcal{Y}} v(y) \mathrm{d}\nu(y) - \varepsilon \Big] \end{aligned}$$

Semi-Dual formulation of OT (with entropy)

The dual problem is convex in u and v. We fix v and minimize over u.

Semi-Dual formulation of OT (with entropy)

The dual problem is convex in u and v. We fix v and minimize over u. This yields :

$$u(x) \overset{\text{def.}}{=} -\varepsilon \log \left(\int_{\mathcal{Y}} \exp(\frac{v(y) - c(x,y)}{\varepsilon}) \mathrm{d}\nu(y) \right)$$

Semi-Dual formulation of OT (with entropy)

The dual problem is convex in u and v. We fix v and minimize over u. This yields :

$$u(x) \stackrel{\text{def.}}{=} -\varepsilon \log\left(\int_{\mathcal{Y}} \exp\left(\frac{v(y) - c(x,y)}{\varepsilon}\right)\mathrm{d}\nu(y)\right)$$

Plugging back in the dual :

$$\begin{aligned} W_\varepsilon(\mu,\nu) &= \max_{v\in\mathcal{C}(\mathcal{Y})} \int_{\mathcal{X}} -\varepsilon \log\left(\int_{\mathcal{Y}} \exp\left(\frac{v(y) - c(x,y)}{\varepsilon}\right)\mathrm{d}\nu(y)\right) \mathrm{d}\mu(y) \\ &\quad + \int_{\mathcal{Y}} v(y)\mathrm{d}\nu(y) - \varepsilon \\ &= \max_{v\in\mathcal{C}(\mathcal{Y})} \mathbb{E}_\mu\Big[-\varepsilon \log\left(\int_{\mathcal{Y}} \exp\left(\frac{v(y) - c(x,y)}{\varepsilon}\right)\right) \\ &\quad + \int_{\mathcal{Y}} v(y)\mathrm{d}\nu(y) - \varepsilon\Big] \end{aligned}$$

We consider 2 frameworks :

- Semi-Discrete : μ is continuous and $\nu = \sum_{j=1}^{M} \nu_i \delta y_j$ The optimization problem is

$$\max_{v \in \mathbb{R}^M} \mathbb{E}_\mu \left[-\varepsilon \log \left(\sum_{j=1}^{M} \exp\left(\frac{v(y_j) - c(x, y_j)}{\varepsilon} \right) \right) + \sum_{j=1}^{M} v(y_j) \nu_j - \varepsilon \right]$$

We consider 2 frameworks :

- Semi-Discrete : μ is continuous and $\nu = \sum_{j=1}^{M} \nu_i \delta y_j$ The optimization problem is

$$\max_{v \in \mathbb{R}^M} \mathbb{E}_\mu \left[-\varepsilon \log \left(\sum_{j=1}^{M} \exp\left(\frac{v(y_j) - c(x, y_j)}{\varepsilon}\right) \right) + \sum_{j=1}^{M} v(y_j)\nu_j - \varepsilon \right]$$

- Discrete : $\mu = \sum_{i=1}^{N} \mu_i \delta x_i$ and $\nu = \sum_{j=1}^{M} \nu_i \delta y_j$ The optimization problem is

$$\max_{v \in \mathbb{R}^M} \sum_{i=1}^{N} \left[-\varepsilon \log \left(\sum_{j=1}^{M} \exp\left(\frac{v(y_j) - c(x_i, y_j)}{\varepsilon}\right) \right) + \sum_{j=1}^{M} v(y_j)\nu_j - \varepsilon \right] \mu_i$$

Stochastic Optimization

Computing the full gradient is

- Hard in the semi-discrete setting (even impossible if we don't know μ explicitly)

Stochastic Optimization

Computing the full gradient is

- Hard in the semi-discrete setting (even impossible if we don't know μ explicitly)
- Very costly in the discrete case since we need to compute N gradients and sum them.

The idea of stochastic optimization is to use approximate gradients so that each iteration is inexpensive.

Stochastic Optimization I

- Goal : maximize $H_\varepsilon(v) = \mathbb{E}_\mu \left[h_\varepsilon(X, v) \right]$ over v in $\mathbb{R}^M$.
- Standard gradient ascent :

$$v^{(k)} = v^{(k-1)} + \nabla_v H_\varepsilon(v^{(k-1)})$$

- The whole gradient $\nabla_v H_\varepsilon(v)$ is too costly/complicated to compute
- Idea : Sample x from μ and use $\nabla_v h_\varepsilon(x, v)$ as a proxy for the full gradient in the gradient ascent.

Stochastic Optimization II

Algorithm 1 Averaged SGD

Input: C

Output: v

$v \leftarrow \mathbb{0}_M$, $\bar{v} \leftarrow v$

for $k = 1, 2, \ldots$ **do**

Sample x_k from μ

$v \leftarrow v + \frac{C}{\sqrt{k}} \nabla_v h_\varepsilon(x_k, v)$ (gradient ascent step)

$\bar{v} \leftarrow \frac{1}{k} v + \frac{k-1}{k} \bar{v}$ (averaging)

end for

- cost of each iteration M
- convergence rate $O(1/\sqrt{(}k))$

Stochastic Optiization : Case of a Finite Sum I

In the specific case where μ is also a discrete measure, we are minimizing a finite sum of N functionals :

$$\max_{v \in \mathbb{R}^M} \sum_{i=1}^{N} \left[-\varepsilon \log \left(\sum_{j=1}^{M} \exp\left(\frac{v(y_j) - c(x_i, y_j)}{\varepsilon}\right) \right) + \sum_{j=1}^{M} v(y_j)\nu_j - \varepsilon \right] \mu_i$$

A more efficient algorithm consists in using an average of the past gradients as a proxy for the full gradient :

- At iteration k, an index i is drawn. Its gradient $\nabla_v h_\varepsilon(x_i, v^{(k)})$ is updated in the vector of partial gradients (vector with N entries kept in memory).
- The average gradient is updated accordingly, and used in a step of the gradient ascent

Stochastic Optiization : Case of a Finite Sum II

Algorithm 2 SAG for Discrete OT

Input: C

Output: $\mathbf{v}$

$\mathbf{v} \leftarrow \mathbb{0}_M$, $\mathbf{d} \leftarrow \mathbb{0}_J$, $\forall i, \mathbf{g}_i \leftarrow \mathbb{0}_M$

for $k = 1, 2, \dots$ **do**

  Sample $i \in \{1, 2, \dots, I\}$ uniform.

  $\mathbf{d} \leftarrow \mathbf{d} - \mathbf{g}_i$

  $\mathbf{g}_i \leftarrow \mu_i \nabla_v \bar{h}_\varepsilon(x_i, \mathbf{v})$

  $\mathbf{d} \leftarrow \mathbf{d} + \mathbf{g}_i$; $\mathbf{v} \leftarrow \mathbf{v} + C\mathbf{d}$

end for

- cost of each iteration M
- convergence rate $O(1/k)$

Numerical Results for Word Mover's Distance (Discrete OT)

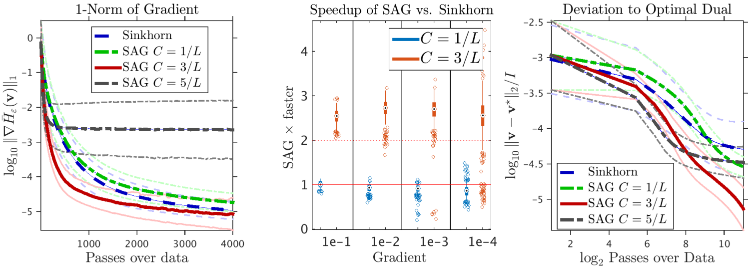
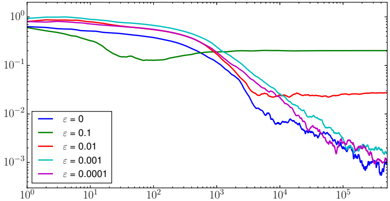
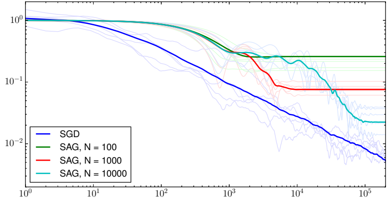


Figure 1: Results for the computation of 595 pairwise word mover's distances between 35 very large corpora of text, each represented as a cloud of $I = 20,000$ word embeddings.

Numerical Results for Density Fitting (Semi-discrete OT)



(a) SGD



(b) SGD vs. SAG

Figure 2: (a) Plot of $\|\mathbf{v}_k - \mathbf{v}_0^\star\|_2 \,/\, \|\mathbf{v}_0^\star\|_2$ as a function of k, for SGD and different values of ε ($\varepsilon = 0$ being un-regularized). (b) Plot of $\|\mathbf{v}_k - \mathbf{v}_\varepsilon^\star\|_2 \,/\, \|\mathbf{v}_\varepsilon^\star\|_2$ averaged over 40 runs as a function of k, for SGD and SAG with different number N of samples, for regularized OT using $\varepsilon = 10^{-2}$.

Dual Formulation as an Expectation

Recall the dual objective function to be maximized, for $\varepsilon > 0$

$$\begin{aligned} F_\varepsilon(u, v) &= \int_{\mathcal{X}} u(x)\mathrm{d}\mu(x) + \int_{\mathcal{Y}} v(y)\mathrm{d}\nu(y) \\ &\quad -\varepsilon \int_{\mathcal{X}\times\mathcal{Y}} \exp(\frac{u(x) + v(y) - c(x,y)}{\varepsilon})\mathrm{d}\mu(x)\mathrm{d}\nu(y) \end{aligned}$$

Let $X \sim \mu$ and $Y \sim \nu$ be two independent random variables, we get

$$F_\varepsilon(u, v) = \mathbb{E}_{\mu\otimes\nu}\left[f_\varepsilon(X, Y, u, v)\right]$$

where $\forall\varepsilon > 0$,

$$f_\varepsilon(x, y, u, v) \stackrel{\text{def.}}{=} u(x) + v(y) - \varepsilon \exp\left(\frac{u(x) + v(y) - c(x,y)}{\varepsilon}\right).$$

Reminder on RKHS I

We consider two reproducing kernel Hilbert spaces (RKHS) $\mathcal{H}$ and $\mathcal{G}$ on $\mathcal{X}$ and on $\mathcal{Y}$, with kernels κ and ℓ.

Properties of RKHS

(a) if $u \in \mathcal{H}$, then $u(x) = \langle u, \kappa(\cdot, x)\rangle_{\mathcal{H}}$
(b) $\kappa(x, x') = \langle \kappa(\cdot, x), \kappa(\cdot, x')\rangle_{\mathcal{H}}$.

The Gaussian Kernel

For the Gaussian Kernel i.e. $\kappa(x, x') = \exp(\frac{\|x-x'\|^2}{2\sigma^2})$ the associated RKHS is dense in the space of continuous functions. This means that any continuous function can be approximated by a linear combination of Gaussian Kernels.

Reminder on RKHS II

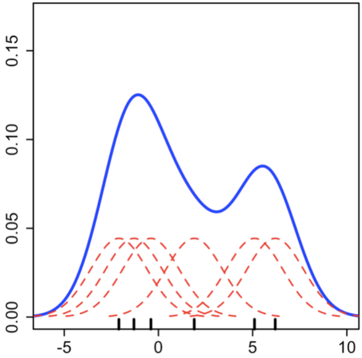


Figure 3: Approximation of a function by a sum of gaussian kernels. The choice of the bandwidth is crucial.

Continuous OT I

$$f_\varepsilon(x,y,u,v) \overset{\text{def.}}{=} u(x)+v(y)-\varepsilon \exp\left(\frac{u(x)+v(y)-c(x,y)}{\varepsilon}\right).$$

Rewriting $u(x)$ and $v(y)$ as scalar products in $\mathcal{H}$ and $\mathcal{G}$ we get

$$\begin{aligned} f_\varepsilon(x,y,u,v) \overset{\text{def.}}{=}\ & \langle u, \kappa(\cdot,x)\rangle_{\mathcal{H}} + \langle v, \ell(\cdot,y)\rangle_{\mathcal{G}} \\ & -\varepsilon \exp\left(\frac{\langle u, \kappa(\cdot,x)\rangle_{\mathcal{H}} + \langle v, \ell(\cdot,y)\rangle_{\mathcal{G}} - c(x,y)}{\varepsilon}\right). \end{aligned}$$

we can apply the SGD algorithm in the RKHS :

$$(u_k, v_k) \overset{\text{def.}}{=} (u_{k-1}, v_{k-1}) + \frac{C}{\sqrt{k}} \nabla f_\varepsilon(x_k, y_k, u_{k-1}, v_{k-1}) \in \mathcal{H}\times\mathcal{G}, \qquad (1)$$

where (x_k, y_k) are i.i.d. samples from $\mu \otimes \nu$.

Continuous OT II

Algorithm 3 Kernel SGD for continuous OT

Input: C, kernels κ and ℓ

Output: $(\alpha_k, x_k, y_k)_{k=1,\dots}$

for $k = 1, 2, \dots$ **do**

Sample x_k from μ

Sample y_k from ν

$u_{k-1}(x_k) \overset{\text{def.}}{=} \sum_{i=1}^{k-1} \alpha_i \kappa(x_k, x_i)$

$v_{k-1}(y_k) \overset{\text{def.}}{=} \sum_{i=1}^{k-1} \alpha_i \ell(y_k, y_i)$

$\alpha_k \overset{\text{def.}}{=} \frac{C}{\sqrt{k}} \left(1 - e^{\frac{u_{k-1}(x_k) + v_{k-1}(y_k) - c(x_k, y_k)}{\varepsilon}} \right)$

end for

Continuous OT III

Proposition : Convergence of SGD in the RKHS

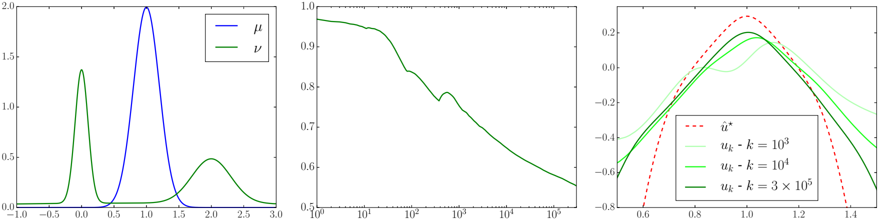
The iterates (u_k, v_k) defined in (1) satisfy

$$(u_k, v_k) \stackrel{\text{def.}}{=} \sum_{i=1}^{k} \alpha_i(\kappa(\cdot, x_i), \ell(\cdot, y_i)) \tag{2}$$

$$\text{where } \alpha_i \stackrel{\text{def.}}{=} \Pi_{B_r}\left(\frac{C}{\sqrt{i}}\left(1 - e^{\frac{u_{i-1}(x_i)+v_{i-1}(y_i)-c(x_i,y_i)}{\varepsilon}}\right)\right), \tag{3}$$

where $(x_i, y_i)_{i=1\dots k}$ are i.i.d samples from $\mu \otimes \nu$ and Π_{B_r} is the projection on the centered ball of radius r. If the solutions of $(\mathcal{D}_\varepsilon)$ are in $\mathcal{H} \times \mathcal{G}$ and if r is large enough, the iterates (u_k, v_k) converge to a solution of $(\mathcal{D}_\varepsilon)$.

Continuous OT : Numerical Results



(a) setting (b) convergence of u_k (c) plots of u_k

Figure 4: (a) Plot of $\frac{\mathrm{d}\mu}{\mathrm{d}x}$ (blue) and $\frac{\mathrm{d}\nu}{\mathrm{d}x}$ (green). (b) Plot of $\|\mathbf{u}_k - \hat{\mathbf{u}}^\star\|_2 / \|\hat{\mathbf{u}}^\star\|_2$ as a function of k with SGD in the RKHS, for regularized OT using $\varepsilon = 10^{-1}$. (c) Plot of the iterates u_k for $k = 10^3, 10^4, 10^5$ and the proxy for the true potential $\hat{\mathbf{u}}^\star$, evaluated on a grid where μ has non negligible mass.

Conclusion

- Dual formulations of OT can be rewritten as expectation maximization problems
- This allows the use of stochastic optimization methods
- Surpass Sinkhorn in the discrete setting (online method more efficient than batch)
- Tackle semi-discrete and continuous problems without requiring discretization