Fast Computation of Wasserstein Barycenters

Marco Cuturi[1], Arnaud Doucet[2]

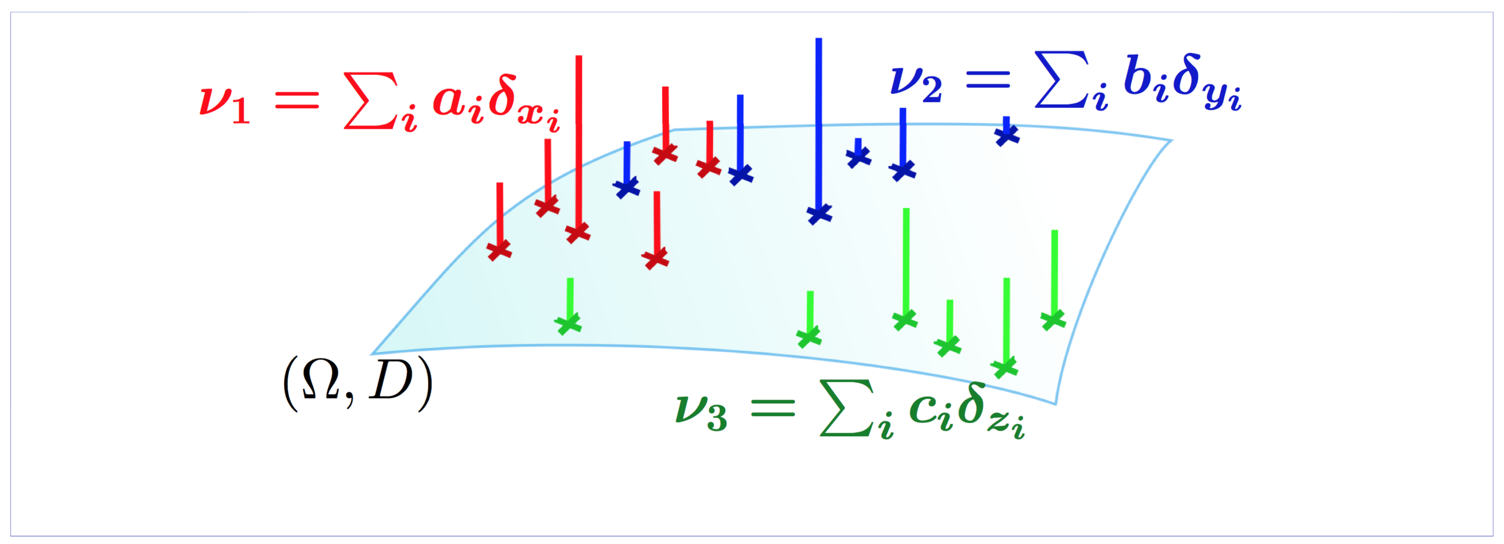
[1] Kyoto University - Graduate School of Informatics [2] University of Oxford - Department of Statistics

Motivation: Averaging Measures

Empirical Probability Measures
Play a Crucial Role in Machine Learning.

- A dataset, a sample = empirical measure.
- A bag-of-words, a histogram = empirical measure (finite probability space).

How can we average
a set of Empirical Probability Measures $\{\nu_1, \cdots, \nu_N\}$?



Ω: finite set (histograms), Hilbert, Metric...
D: Riemannian, Hilbert, *APSP* on a graph...

First question: how can we define averages?

- For vectors $\{x_1, \cdots, x_N\}$ in a Hilbert space, their average is

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad \textbf{(Explicit formula)}$$

$$= \operatorname*{argmin}_{u\in\mathbb{R}^d} \|u - x_i\|_2^2 = \operatorname*{argmin}_{u\in\mathbb{R}^d} D_{\text{Euclidean}}(u, x_i)^2 \qquad \textbf{(Variational formulation)}$$

- For non-Euclidean spaces (e.g. probability simplex) define a metric/ a divergence **[Banerjee et al'05, Nielsen'13]** and min. the variational formulation.

Our contribution: A Fast Computational Approach
to compute that average when
D = the Optimal Transport Distance
a.k.a Wasserstein, EMD, Monge-Kantorovich

Wasserstein Barycenters (theory by **[Agueh,Carlier'11]**)

- **Wasserstein...** : for $p \in [1, \infty)$, μ, ν in $P(\Omega)$,

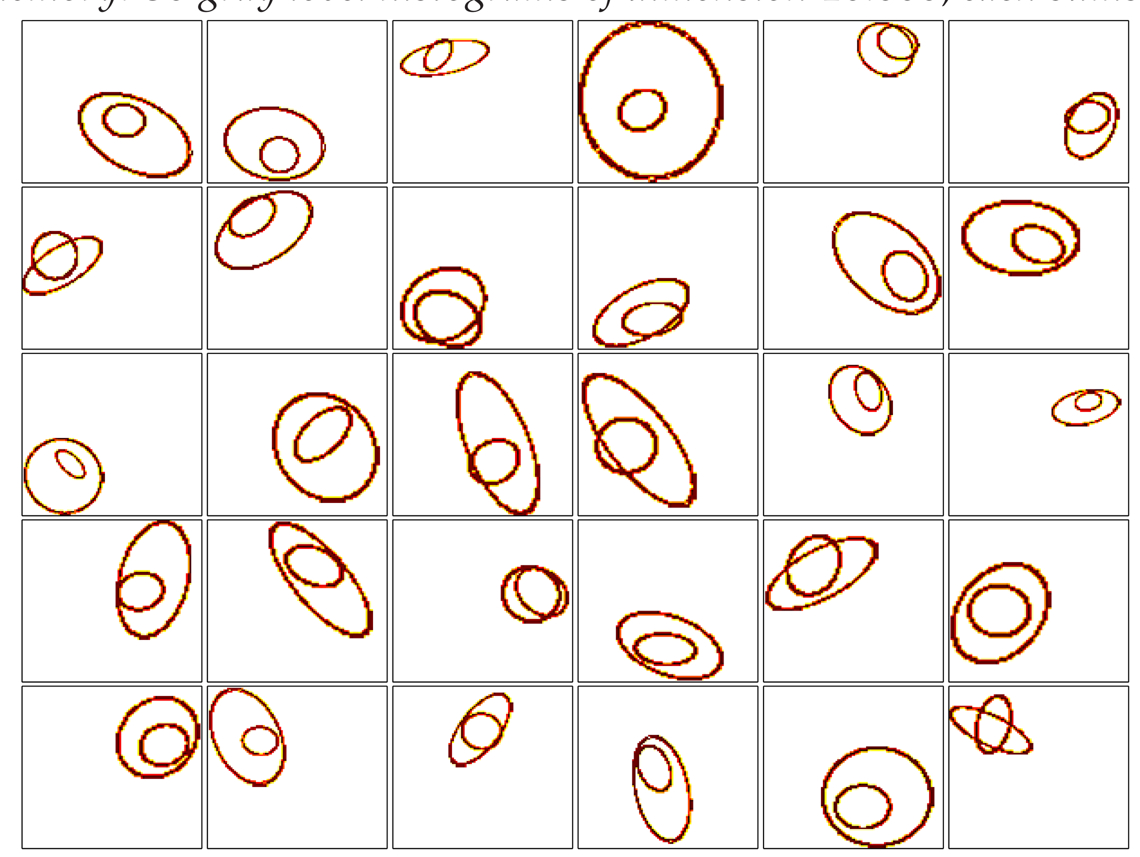
$$W_p(\mu,\nu) \stackrel{\text{def}}{=} \left(\inf_{\pi\in\Pi(\mu,\nu)} \int_{\Omega^2} D(x,y)^p d\pi(x,y)\right)^{1/p} \textbf{[Villani'09]},$$

where $\Pi(\mu,\nu)$ is the set of probability measures on Ω^2 with marginals μ, ν.

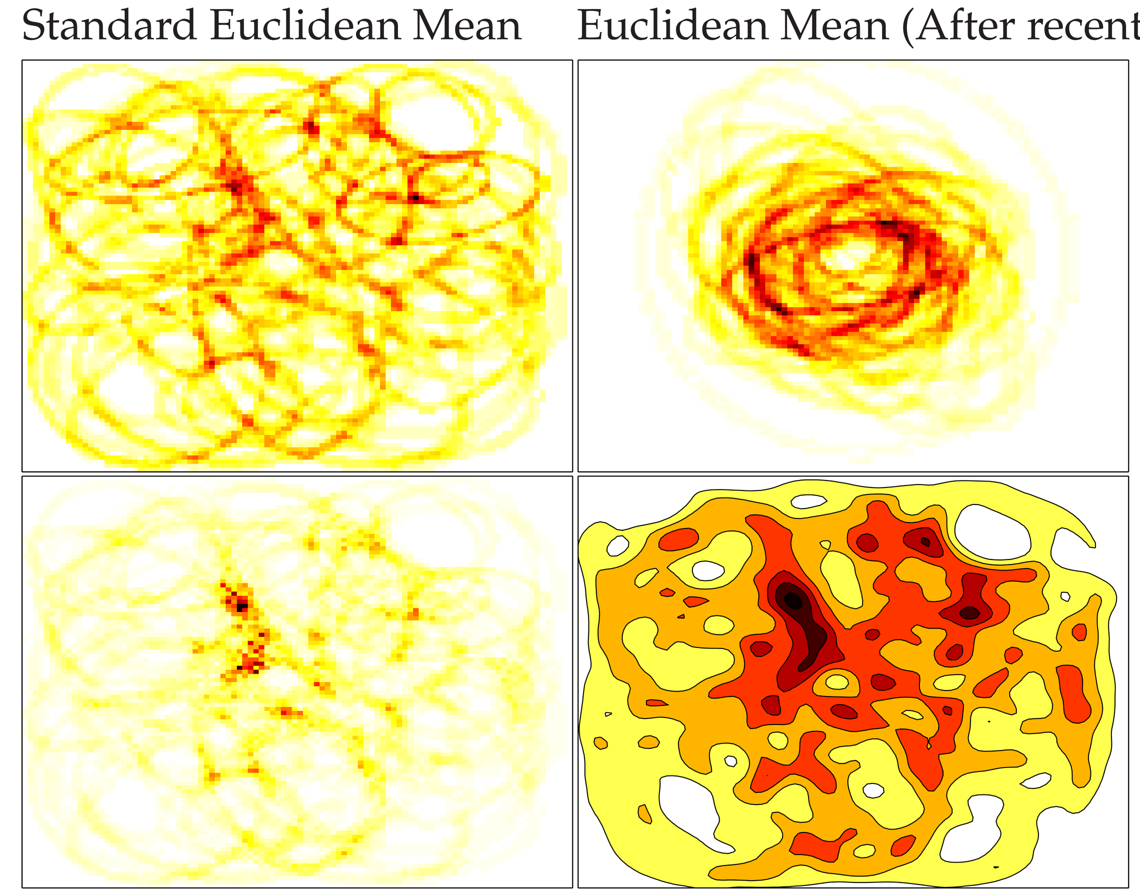
- **...Barycenters**: $\operatorname{argmin}_\mu f(\mu) \stackrel{\text{def}}{=} \frac{1}{N}\sum_{i=1}^N W_p^p(\mu, \nu_i)$.

Wasserstein Barycenters in One Example

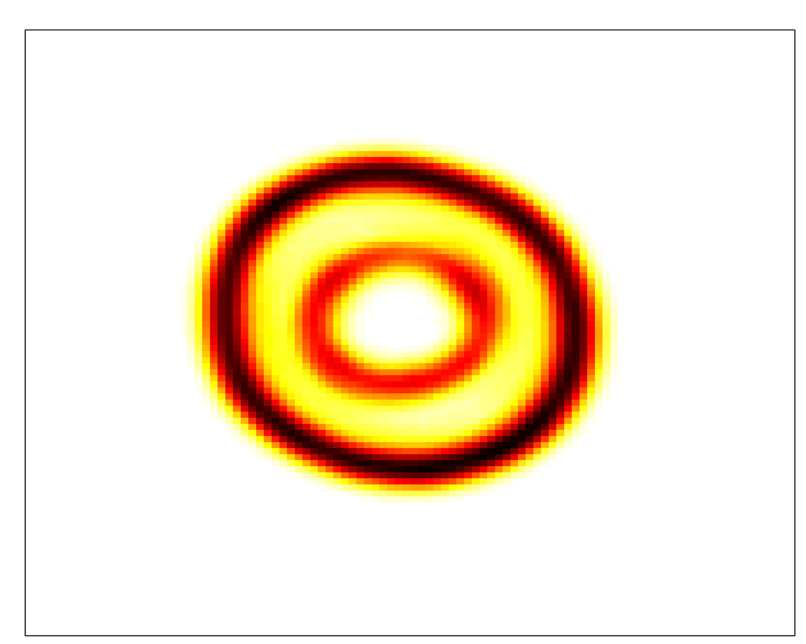
30 measures on the plane $[0,1]^2$, discretized as a 100×100 grid.
(in memory: 30 gray level histograms of dimension 10.000, each sums to 1)



Standard Euclidean Mean — Euclidean Mean (After recentering)



Symmetrized Kullback-Leibler — RKHS Mean (Gaussian $\sigma = 0.002$)



2-Wasserstein Mean

How can we get that? Duality, Sinkhorn's Matrix Scaling Algorithm to Solve Entropy Smoothed Optimal Transport, **GPGPU**.

Computation

Optimal Transport

- Let $\mu = \sum_{i=1}^n a_i \delta_{x_i}$ and $\nu = \sum_{j=1}^n b_j \delta_{y_j}$ be 2 probability measures.
- Let the **(pairwise distance matrix)**^{p} $M_{XY} \stackrel{\text{def}}{=} [D(x_i, y_j)^p]_{ij} \in \mathbb{R}^{n\times m}$
- Let the **transportation polytope** $U(a,b)$ of $a \in \Sigma_n$ and $b \in \Sigma_m$ be

$$U(a,b) \stackrel{\text{def}}{=} \{T \in \mathbb{R}_+^{n\times m} \mid T\mathbf{1}_m = a,\ T^T\mathbf{1}_n = b\}.$$

- Then, their p-Wasserstein distance is the solution (*either* primal or dual LP)

$$W_p^p(\mu,\nu) = \begin{cases} \mathbf{p}(a,b,M_{XY}) \stackrel{\text{def}}{=} \min_{T\in U(a,b)} \langle T, M_{XY}\rangle & \textbf{(primal)} \\ \mathbf{d}(a,b,M_{XY}) \stackrel{\text{def}}{=} \max_{(\alpha,\beta)\in C_{M_{XY}}} \alpha^T a + \beta^T b, & \textbf{(dual)} \\ \quad \text{where } C_M = \{(\alpha,\beta)\in\mathbb{R}^{n+m} \mid \alpha_i + \beta_j \le M_{ij}\}. \end{cases} = W(a, X)$$

(Sub)differentiability of Wasserstein Distance

- $\partial W|_a = \alpha^\star \Rightarrow$ *dual opt.* $\alpha^\star$ is a subgradient of $W|_a$
- $\partial W|_X = Y T^{\star T} \mathbf{diag}(a^{-1}) \Rightarrow$ *primal opt.* is a subgr. of $W|_X$ (in Euclidean case.)

Given ν_i with supp. Y_i and weights b_i, find support X and weight a to min. $f(a, X)$

$$f(\boldsymbol{a}, \boldsymbol{X}) \stackrel{\text{def}}{=} \frac{1}{N}\sum_{i=1}^N \mathbf{p}(\boldsymbol{a}, \boldsymbol{b_i}, M_{\boldsymbol{XY_i}})$$

Naive Subgradient Method (Hopeless...)

- $a \to f(a, X)$ is CONVEX: simple subgradient works (in theory...)
- $X \to f(a, X)$ is NOT CONVEX: can only converge to local minima (k-means)

Efficient Computations using Sinkhorn

- ENTROPY SMOOTHED **[Cuturi'13]** primal/dual optimal transports

$$\mathbf{p}_\lambda(a,b;M) = \min_{T\in U(a,b)} \langle X, M\rangle - \frac{1}{\lambda}h(T).$$

$$\mathbf{d}_\lambda(a,b;M) = \max_{(\alpha,\beta)\in\mathbb{R}^{n+m}} \alpha^T a + \beta^T b - \sum_{i\le n, j\le m} \frac{e^{-\lambda(m_{ij}-\alpha_i-\beta_j)}}{\lambda}$$

Proposition: Let $K \stackrel{\text{def}}{=} e^{-\lambda M_{XY}}$. Then there exists a pair of vectors $(u,v) \in \mathbb{R}_+^n \times \mathbb{R}_+^m$ recoverable with Sinkhorn's algorithm in $O(nm)$ such that

$$T_\lambda^\star = \mathbf{diag}(u) K\, \mathbf{diag}(v), \qquad \alpha_\lambda^\star = -\frac{\log(u)}{\lambda} + \frac{\log(u)^T\mathbf{1}_n}{\lambda n}\mathbf{1}_n.$$

- $\Rightarrow$ **do a simple (projected) gradient descent on smoothed objectives.**

More details (GPU parallelization, links with constrained clustering etc) in the paper.