Primal Parallel Heuristics for Computing Wasserstein Barycenters

Stefano Gualandi[a]

Joint works:
P.Y. Bouchet[c], L.M. Rousseau[c]

G. Auricchio[a], E. Chenchen[a], M. Mascherpa[a], M. Veneroni[a], F. Bassetti[b],
([a]) Università di Pavia, Dipartimento di Matematica, Italy
([b]) Politecnico di Milano, Dipartimento di Matematica
([c]) École Polytechnique de Montréal

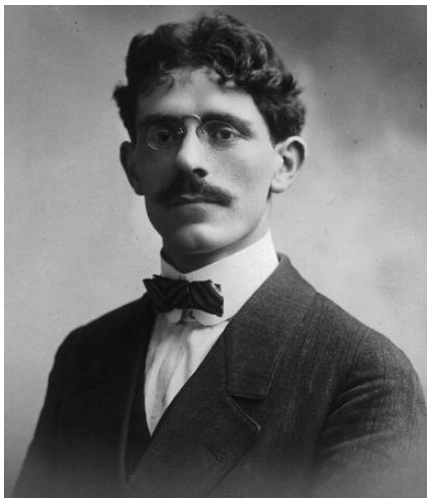
Gaoling School of Artificial Intelligence, RUC **Online seminar, Nov 9th, 2022**

email: stefano.gualandi@unipv.it
mastodon: @famo2spaghi@mastodon.sigmoid
twitter: @famo2spaghi
blog: http://stegua.github.com

Outline

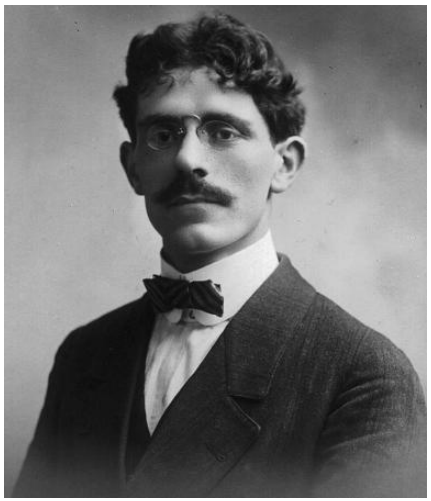
Section 1

Statistical Inference: *Éléments typiques*



Éléments typiques. *La statistique a constamment besoin de donner une idée de ce que sont, en gros, les éléments d'une collection complexe, difficile à saisir dans son ensemble, en les assimilant à un seul élément choisi aussi ressemblant que possible à tous les éléments de la collection et qui, pour cette raison, soit appelé élément typique de cette collection.* *Maurice Fréchet, 1945.*

Statistical Inference: *Éléments typiques*



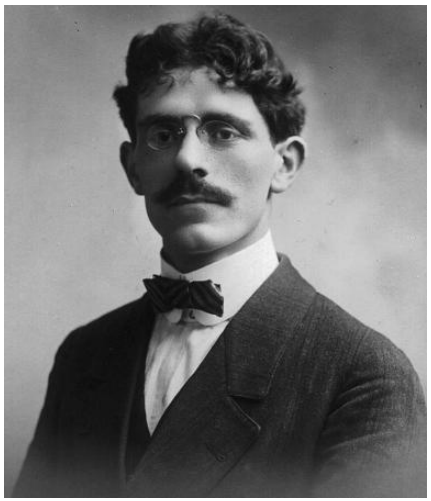
Éléments typiques. *La statistique a constamment besoin de donner une idée de ce que sont, en gros, les éléments d'une collection complexe, difficile à saisir dans son ensemble, en les assimilant à un seul élément choisi aussi ressemblant que possible à tous les éléments de la collection et qui, pour cette raison, soit appelé élément typique de cette collection.* *Maurice Fréchet, 1945.*

Example 1: Numbers in $\mathbb{R}$

Consider the collection of numbers $C = \{2, 3, 4, 6, 7, 12\}$.

QUESTION: Which is the *Élément typique* of C?

Statistical Inference: *Éléments typiques*



Éléments typiques. *La statistique a constamment besoin de donner une idée de ce que sont, en gros, les éléments d'une collection complexe, difficile à saisir dans son ensemble, en les assimilant à un seul élément choisi aussi ressemblant que possible à tous les éléments de la collection et qui, pour cette raison, soit appelé élément typique de cette collection.* *Maurice Fréchet, 1945.*

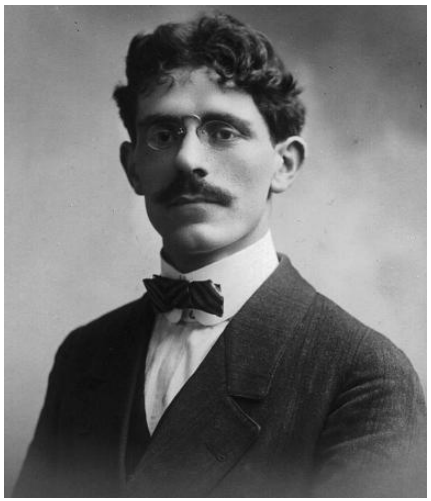
Example 1: Numbers in $\mathbb{R}$

Consider the collection of numbers $C = \{2, 3, 4, 6, 7, 12\}$.

QUESTION: Which is the *Élément typique* of C?

Mean:

Statistical Inference: *Éléments typiques*



Éléments typiques. *La statistique a constamment besoin de donner une idée de ce que sont, en gros, les éléments d'une collection complexe, difficile à saisir dans son ensemble, en les assimilant à un seul élément choisi aussi ressemblant que possible à tous les éléments de la collection et qui, pour cette raison, soit appelé élément typique de cette collection.* *Maurice Fréchet, 1945.*

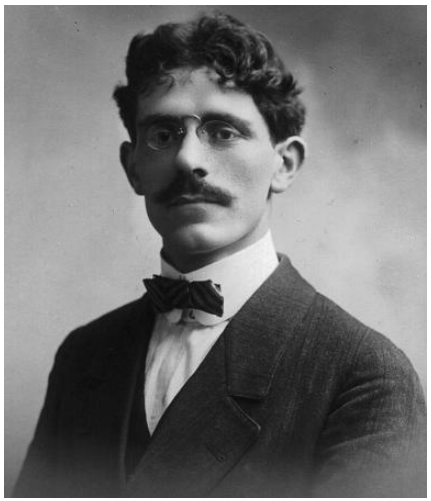
Example 1: Numbers in $\mathbb{R}$

Consider the collection of numbers $C = \{2, 3, 4, 6, 7, 12\}$.

QUESTION: Which is the *Élément typique* of C?

Mean: $\rho^* = \arg\min_{\rho \in \mathbb{R}} \frac{1}{2} \sum_{c \in C} ||\rho - c||_2^2$

Statistical Inference: *Éléments typiques*



Éléments typiques. *La statistique a constamment besoin de donner une idée de ce que sont, en gros, les éléments d'une collection complexe, difficile à saisir dans son ensemble, en les assimilant à un seul élément choisi aussi ressemblant que possible à tous les éléments de la collection et qui, pour cette raison, soit appelé élément typique de cette collection.* *Maurice Fréchet, 1945.*

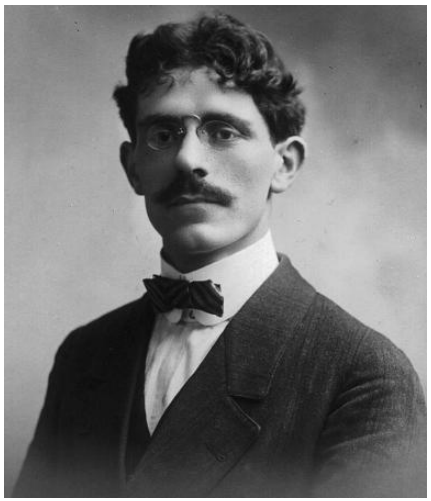
Example 1: Numbers in $\mathbb{R}$

Consider the collection of numbers $C = \{2, 3, 4, 6, 7, 12\}$.

QUESTION: Which is the *Élément typique* of C?

Mean: $\rho^* = \arg\min_{\rho \in \mathbb{R}} \frac{1}{2} \sum_{c \in C} ||\rho - c||_2^2 \quad \rightarrow$

Statistical Inference: *Éléments typiques*



Éléments typiques. *La statistique a constamment besoin de donner une idée de ce que sont, en gros, les éléments d'une collection complexe, difficile à saisir dans son ensemble, en les assimilant à un seul élément choisi aussi ressemblant que possible à tous les éléments de la collection et qui, pour cette raison, soit appelé élément typique de cette collection.* *Maurice Fréchet, 1945.*

Example 1: Numbers in $\mathbb{R}$

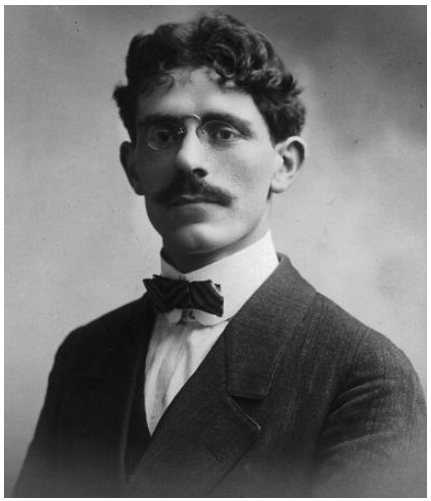
Consider the collection of numbers $C = \{2, 3, 4, 6, 7, 12\}$.

QUESTION: Which is the *Élément typique* of C?

Mean: $\rho^* = \arg\min_{\rho \in \mathbb{R}} \frac{1}{2} \sum_{c \in C} ||\rho - c||_2^2 \quad \rightarrow \quad \rho^* = \frac{\sum_{c \in C} c}{|C|} \approx 5.66$

Median:

Statistical Inference: *Éléments typiques*



Éléments typiques. *La statistique a constamment besoin de donner une idée de ce que sont, en gros, les éléments d'une collection complexe, difficile à saisir dans son ensemble, en les assimilant à un seul élément choisi aussi ressemblant que possible à tous les éléments de la collection et qui, pour cette raison, soit appelé élément typique de cette collection.* *Maurice Fréchet, 1945.*

Example 1: Numbers in $\mathbb{R}$

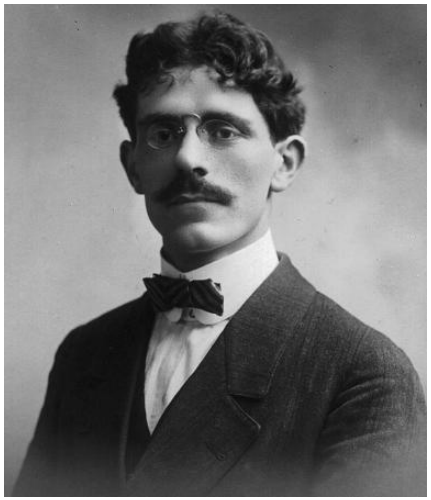
Consider the collection of numbers $C = \{2, 3, 4, 6, 7, 12\}$.

QUESTION: Which is the *Élément typique* of C?

Mean: $\rho^* = \arg\min_{\rho \in \mathbb{R}} \frac{1}{2} \sum_{c \in C} ||\rho - c||_2^2 \quad \rightarrow \quad \rho^* = \frac{\sum_{c \in C} c}{|C|} \approx 5.66$

Median: $\rho^* = \arg\min_{\rho \in \mathbb{R}} \frac{1}{2} \sum_{c \in C} ||\rho - c||_2$

Statistical Inference: *Éléments typiques*



Éléments typiques. *La statistique a constamment besoin de donner une idée de ce que sont, en gros, les éléments d'une collection complexe, difficile à saisir dans son ensemble, en les assimilant à un seul élément choisi aussi ressemblant que possible à tous les éléments de la collection et qui, pour cette raison, soit appelé élément typique de cette collection.* *Maurice Fréchet, 1945.*

Example 1: Numbers in $\mathbb{R}$

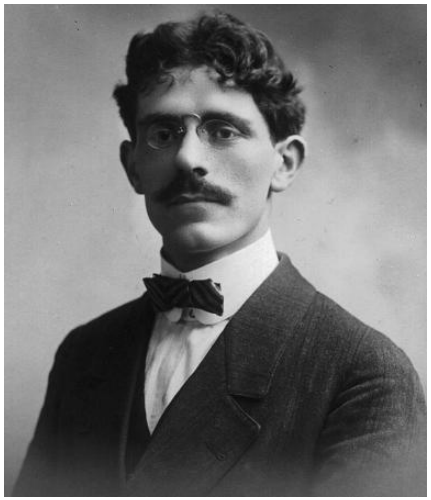
Consider the collection of numbers $C = \{2, 3, 4, 6, 7, 12\}$.

QUESTION: Which is the *Élément typique* of C?

Mean: $\rho^* = \arg\min_{\rho \in \mathbb{R}} \frac{1}{2} \sum_{c \in C} ||\rho - c||_2^2 \quad \rightarrow \quad \rho^* = \frac{\sum_{c \in C} c}{|C|} \approx 5.66$

Median: $\rho^* = \arg\min_{\rho \in \mathbb{R}} \frac{1}{2} \sum_{c \in C} ||\rho - c||_2 \quad \rightarrow$

Statistical Inference: *Éléments typiques*



Éléments typiques. *La statistique a constamment besoin de donner une idée de ce que sont, en gros, les éléments d'une collection complexe, difficile à saisir dans son ensemble, en les assimilant à un seul élément choisi aussi ressemblant que possible à tous les éléments de la collection et qui, pour cette raison, soit appelé élément typique de cette collection.* *Maurice Fréchet, 1945.*

Example 1: Numbers in $\mathbb{R}$

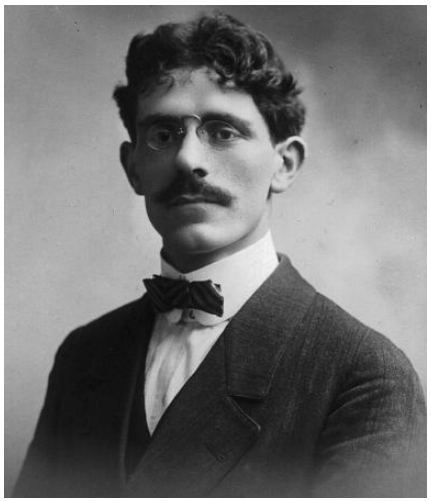
Consider the collection of numbers $C = \{2, 3, 4, 6, 7, 12\}$.

QUESTION: Which is the *Élément typique* of C?

Mean: $\rho^* = \arg\min_{\rho \in \mathbb{R}} \frac{1}{2} \sum_{c \in C} ||\rho - c||_2^2 \quad \rightarrow \quad \rho^* = \frac{\sum_{c \in C} c}{|C|} \approx 5.66$

Median: $\rho^* = \arg\min_{\rho \in \mathbb{R}} \frac{1}{2} \sum_{c \in C} ||\rho - c||_2 \quad \rightarrow \quad \rho^* \in [4 \dots 6]$ (...many solutions!)

Statistical Inference: *Éléments typiques*



Éléments typiques. *La statistique a constainment besoin de donner une idée de ce que sont, en gros, les éléments d'une collection complexe, difficile à saisir dans son ensemble, en les assimilant à un seul élément choisi aussi ressemblant que possible à tous les éléments de la collection et qui, pour cette raison, soit appelé élément typique de cette collection.* *Maurice Fréchet, 1945.*

Example 1: Numbers in $\mathbb{R}$

Consider the collection of numbers $C = \{2, 3, 4, 6, 7, 12\}$.

QUESTION: Which is the *Élément typique* of C?

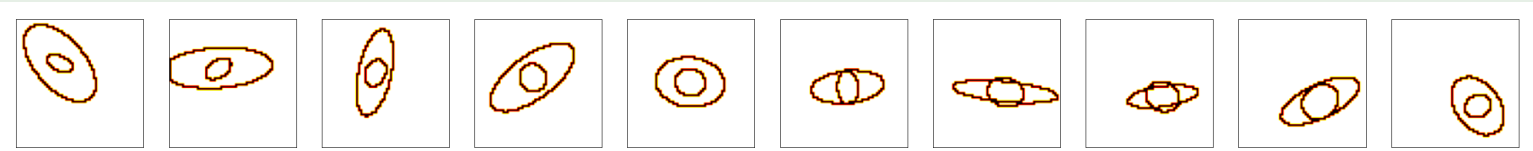
Mean: $\rho^* = \arg\min_{\rho \in \mathbb{R}} \frac{1}{2} \sum_{c \in C} ||\rho - c||_2^2 \quad \rightarrow \quad \rho^* = \frac{\sum_{c \in C} c}{|C|} \approx 5.66$

Median: $\rho^* = \arg\min_{\rho \in \mathbb{R}} \frac{1}{2} \sum_{c \in C} ||\rho - c||_2 \quad \rightarrow \quad \rho^* \in [4 \dots 6]$ (...many solutions!)

QUESTION: How generalize *mean* and *median* to distributions in $\mathbb{R}^q$?

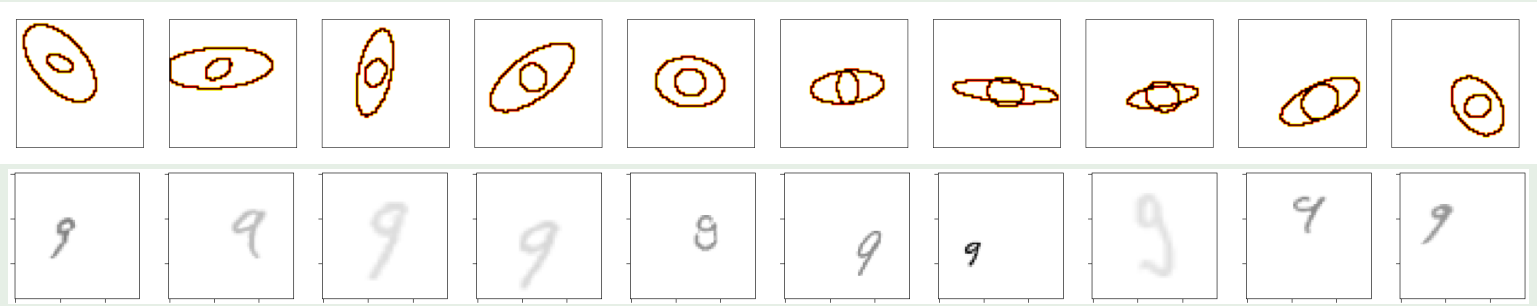
Mean and Medians of distributions

Example 2: Images (Double ellipses and MNIST digits)



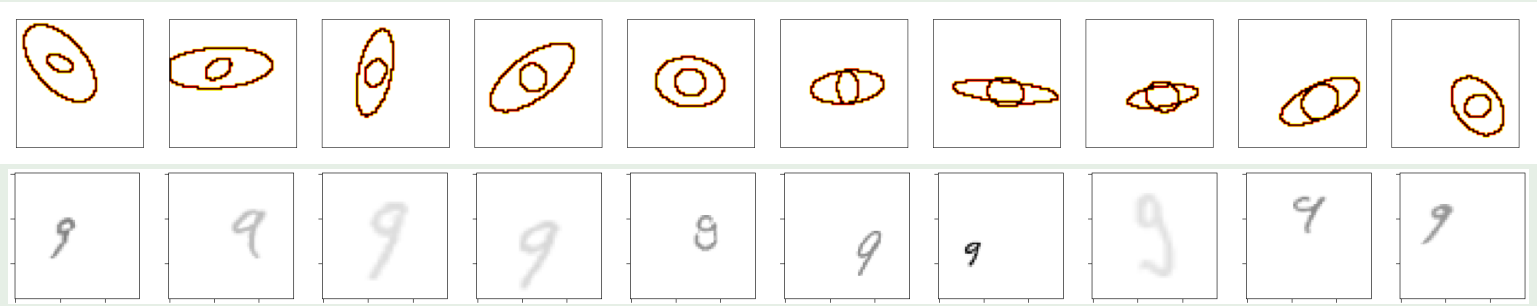
Mean and Medians of distributions

Example 2: Images (Double ellipses and MNIST digits)



Mean and Medians of distributions

Example 2: Images (Double ellipses and MNIST digits)



Definition 1 (Fréchet mean)

Let $\mu_1, \ldots, \mu_k \in \mathcal{X}$ with $(\mathcal{X}, \text{distance})$ a metric space, $k \geq 2$, and $\lambda_1, \ldots, \lambda_k \geq 0$, such that $\sum_{i=1}^{k} \lambda_i = 1$. For any $\alpha \geq 1$, a Fréchet mean of order α is any optimal solution to the following problem:

$$\rho^* = \arg\min \left\{ \sum_{i=1}^{k} \lambda_i \text{distance}^{\alpha}(\rho, \mu_i) \mid \rho \in \mathcal{X} \right\}$$

When $\alpha = 2$, we get a **(weighted) Fréchet barycenter**.
When $\alpha = 1$, we get a **(weighted) Fréchet median**.

Mean and Medians of distributions

Definition 2 (Wasserstein Means)

Let $\mathcal{P}_2(\mathbb{R}^q)$ be the space of probability measures on $\mathbb{R}^q$ with finite second moments equipped with a Wasserstein distance of order α.

Mean and Medians of distributions

Definition 2 (Wasserstein Means)

Let $\mathcal{P}_2(\mathbb{R}^q)$ be the space of probability measures on $\mathbb{R}^q$ with finite second moments equipped with a Wasserstein distance of order α. Given $k \geq 2$ probability measures $\mu_1, \ldots, \mu_k \in \mathcal{P}_2(\mathbb{R}^q)$, a Wasserstein mean of $\mu_1, \ldots, \mu_k$ is any optimal solution to

$$\rho^* = \arg\min \left\{ \sum_{i=1,\ldots,k} \lambda_i W^\alpha(\rho, \mu_i) \mid \rho \in \mathcal{P}_2(\mathbb{R}^q) \right\}$$

where $\lambda_1, \ldots, \lambda_k \geq 0$, such that $\sum_{i=1}^k \lambda_i = 1$.

When $\alpha = 2$, we get a **(weighted) Wasserstein barycenter**.
When $\alpha = 1$, we get a **(weighted) Wasserstein median**.

Mean and Medians of distributions

Definition 2 (Wasserstein Means)

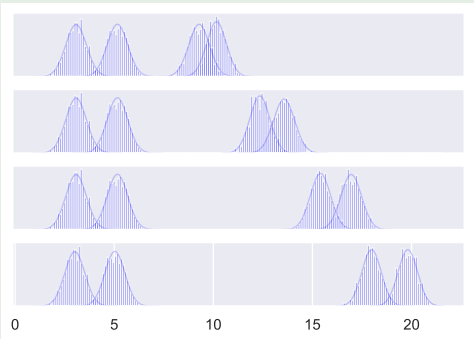
Let $\mathcal{P}_2(\mathbb{R}^q)$ be the space of probability measures on $\mathbb{R}^q$ with finite second moments equipped with a Wasserstein distance of order α. Given $k \geq 2$ probability measures $\mu_1, \ldots, \mu_k \in \mathcal{P}_2(\mathbb{R}^q)$, a Wasserstein mean of $\mu_1, \ldots, \mu_k$ is any optimal solution to

$$\rho^* = \arg\min \left\{ \sum_{i=1,\ldots,k} \lambda_i W^\alpha(\rho, \mu_i) \mid \rho \in \mathcal{P}_2(\mathbb{R}^q) \right\}$$

where $\lambda_1, \ldots, \lambda_k \geq 0$, such that $\sum_{i=1}^k \lambda_i = 1$.

When $\alpha = 2$, we get a **(weighted) Wasserstein barycenter**.
When $\alpha = 1$, we get a **(weighted) Wasserstein median**.

Example 3: Distributions in $\mathcal{P}_2(\mathbb{R})$ `https://github.com/stegua/ot1d`



Mean and Medians of distributions

Definition 2 (Wasserstein Means)

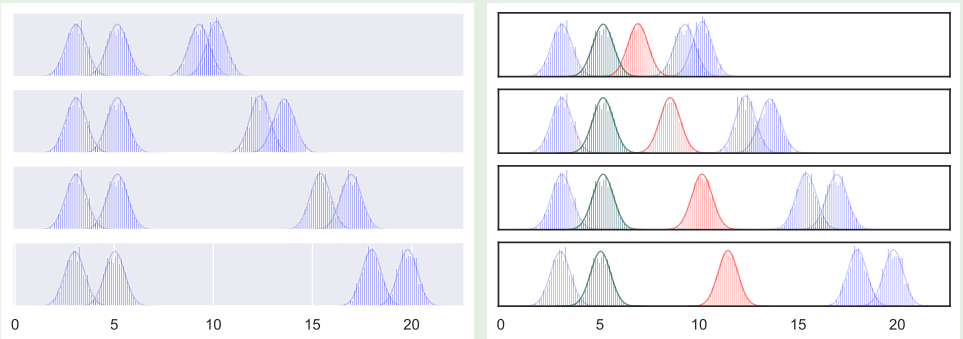
Let $\mathcal{P}_2(\mathbb{R}^q)$ be the space of probability measures on $\mathbb{R}^q$ with finite second moments equipped with a Wasserstein distance of order α. Given $k \geq 2$ probability measures $\mu_1, \ldots, \mu_k \in \mathcal{P}_2(\mathbb{R}^q)$, a Wasserstein mean of $\mu_1, \ldots, \mu_k$ is any optimal solution to

$$\rho^* = \arg\min \left\{ \sum_{i=1,\ldots,k} \lambda_i W^\alpha(\rho, \mu_i) \mid \rho \in \mathcal{P}_2(\mathbb{R}^q) \right\}$$

where $\lambda_1, \ldots, \lambda_k \geq 0$, such that $\sum_{i=1}^k \lambda_i = 1$.

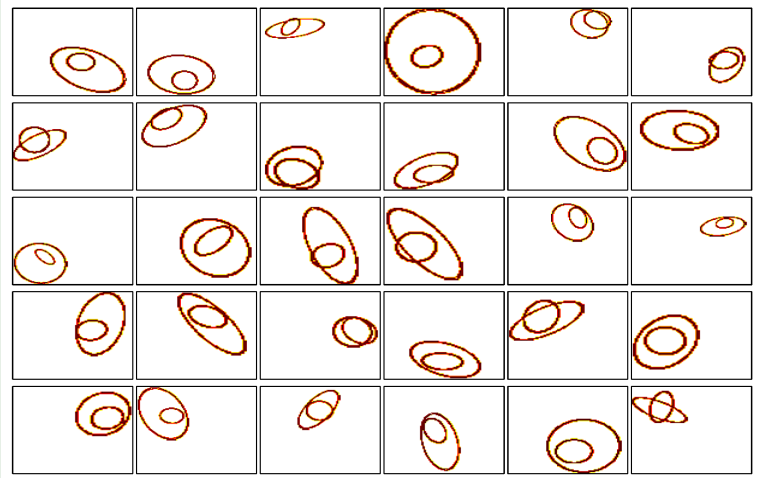
When $\alpha = 2$, we get a **(weighted) Wasserstein barycenter**.
When $\alpha = 1$, we get a **(weighted) Wasserstein median**.

Example 3: Distributions in $\mathcal{P}_2(\mathbb{R})$ `https://github.com/stegua/ot1d`



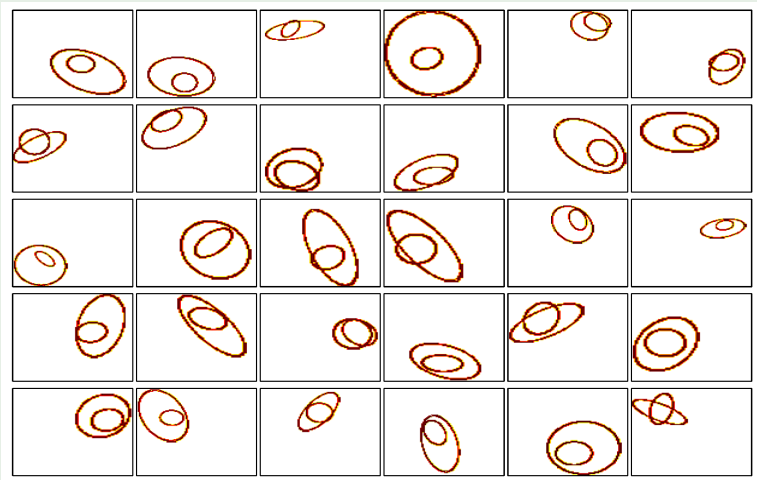
Barycenters of images [Cuturi and Doucet, 2014]

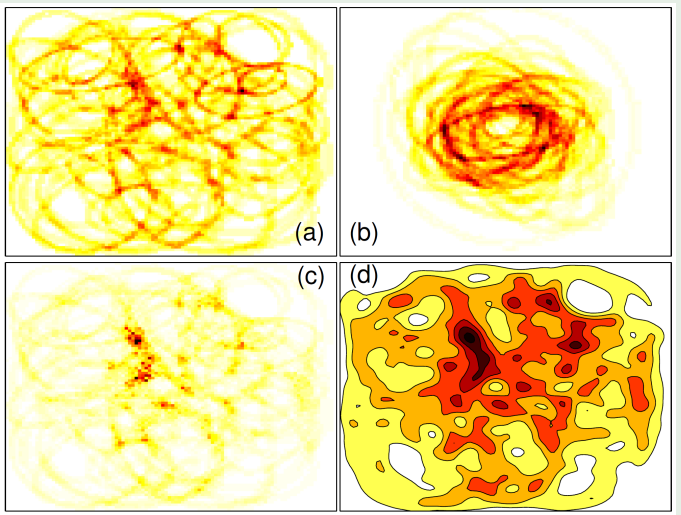
Example 4: Distributions in $\mathcal{P}_2(\mathbb{R}^2)$



Barycenters of images [Cuturi and Doucet, 2014]

Example 4: Distributions in $\mathcal{P}_2(\mathbb{R}^2)$





(a) Euclidean Barycenter

(b) Reentered Euclidean Barycenter

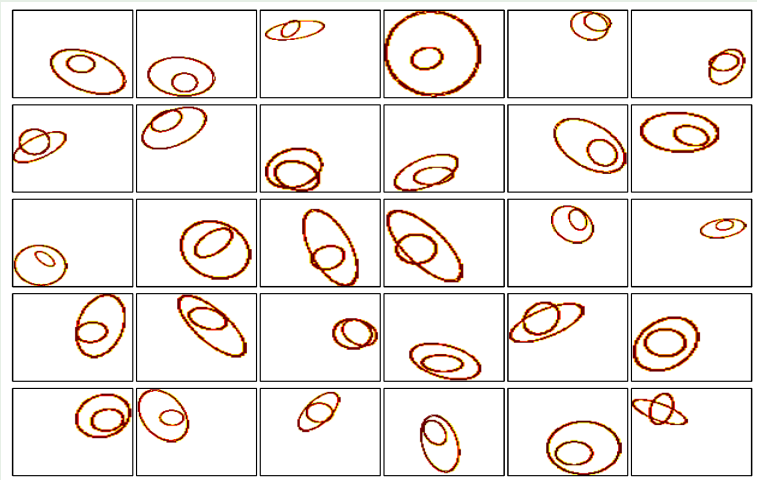
(c) Jeffrey Centroid

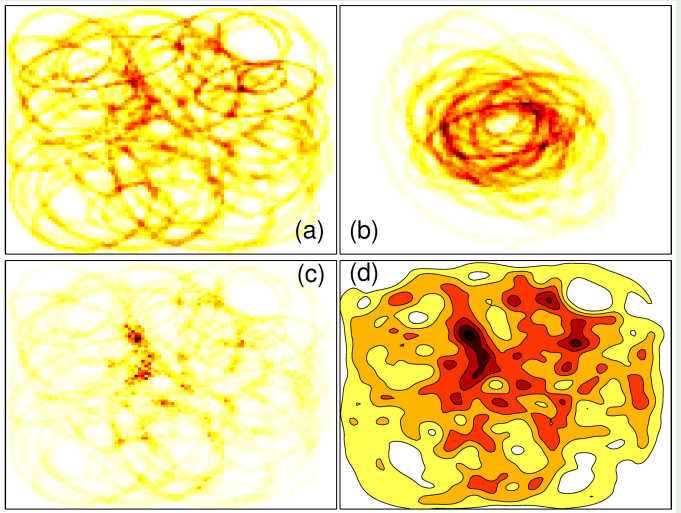
(d) Gaussian kernel-based

(e) Wasserstein Barycenter

Barycenters of images [Cuturi and Doucet, 2014]

Example 4: Distributions in $\mathcal{P}_2(\mathbb{R}^2)$





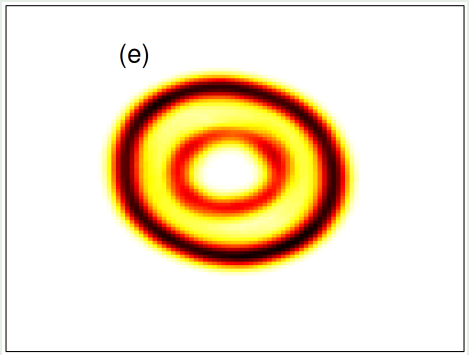
(a) Euclidean Barycenter

(b) Reentered Euclidean Barycenter

(c) Jeffrey Centroid

(d) Gaussian kernel-based

(e) Wasserstein Barycenter



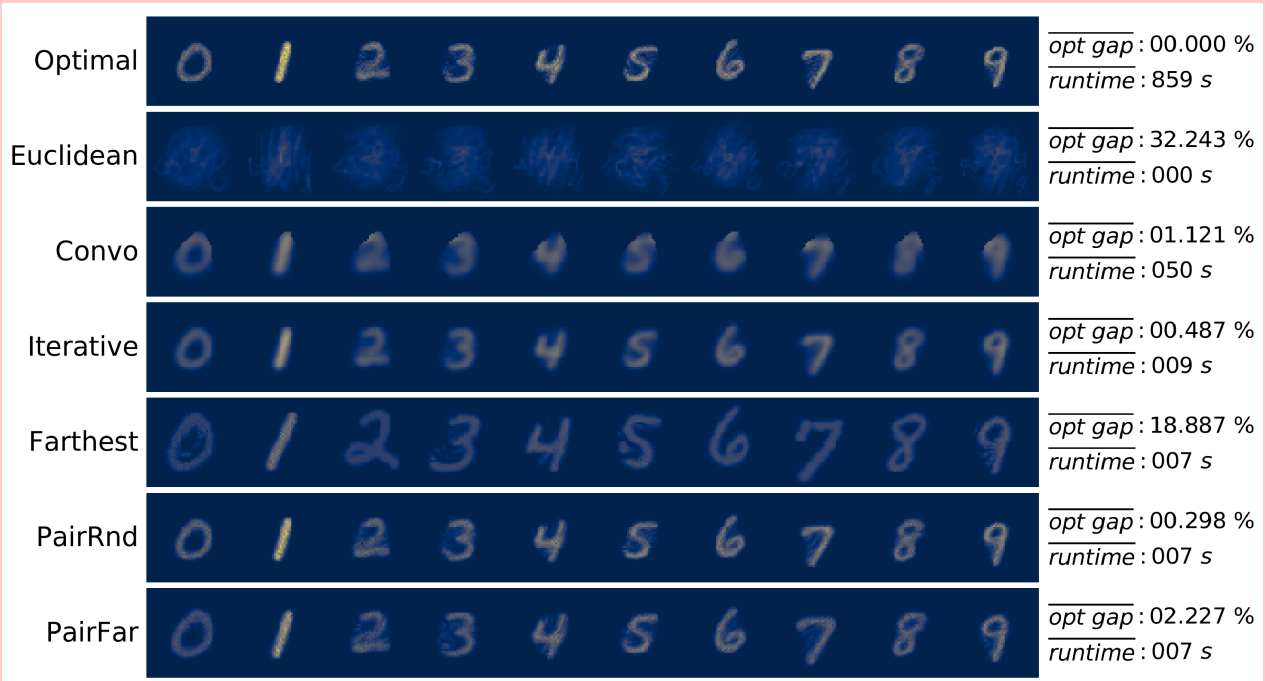
Cuturi, M. and Doucet, A., 2014, June. Fast computation of Wasserstein barycenters. In International Conference on Machine Learning (ICML), pp. 685–693.

This Talk: Primal Heuristic for Wasserstein Barycenters

In this talk, we present a **Primal Heuristic for Wasserstein Barycenters**, which is based on the exact solution of LP subproblems to compute pairwise distances among samples of a collection of distributions (e.g., images).

This Talk: Primal Heuristic for Wasserstein Barycenters

In this talk, we present a **Primal Heuristic for Wasserstein Barycenters**, which is based on the exact solution of LP subproblems to compute pairwise distances among samples of a collection of distributions (e.g., images). *Results spoiler*:



Bouchet, P.Y., G., S. and Rousseau, L.M., 2020, September. Primal heuristics for Wasserstein Barycenters. In proc of Constraint Programming, Artificial Intelligence, and Operations Research (CPAIOR2020), pp. 239–255.

Section 2

Outline

Discrete Probability measures

We define the Dirac δ-measure centered at $x \in X \subseteq \mathbb{R}^q$ via

$$\delta_x(A) = \begin{cases} 1 & \text{if } x \in A \subseteq X \\ 0 & \text{if } x \notin A \subseteq X. \end{cases}$$

Discrete Probability measures

We define the Dirac δ-measure centered at $x \in X \subseteq \mathbb{R}^q$ via

$$\delta_x(A) = \begin{cases} 1 & \text{if } x \in A \subseteq X \\ 0 & \text{if } x \notin A \subseteq X. \end{cases}$$

Let μ and ν be two discrete probability measures defined on $X, Y \subseteq \mathbb{R}^q$

$$\mu(A) = \sum_{i=1}^{m} \mu_i \delta_{x_i}(A) \qquad \text{and} \qquad \nu(A) = \sum_{j=1}^{n} \nu_j \delta_{y_j}(A)$$

where $\sum_{i=1}^{m} \mu_i = 1$ and $\sum_{i=1}^{n} \nu_i = 1$.

Discrete Probability measures

We define the Dirac δ-measure centered at $x \in X \subseteq \mathbb{R}^q$ via

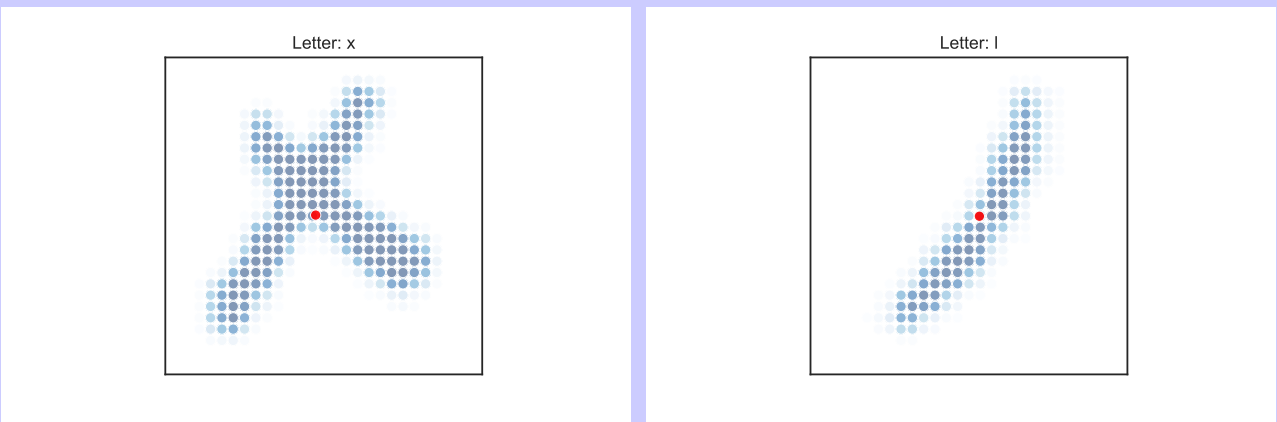
$$\delta_x(A) = \begin{cases} 1 & \text{if } x \in A \subseteq X \\ 0 & \text{if } x \notin A \subseteq X. \end{cases}$$

Let μ and ν be two discrete probability measures defined on $X, Y \subseteq \mathbb{R}^q$

$$\mu(A) = \sum_{i=1}^{m} \mu_i \delta_{x_i}(A) \qquad \text{and} \qquad \nu(A) = \sum_{j=1}^{n} \nu_j \delta_{y_j}(A)$$

where $\sum_{i=1}^{m} \mu_i = 1$ and $\sum_{i=1}^{n} \nu_i = 1$.

Example: Grey scale images are discrete measures: $x_i = (x_{i1}, x_{i2})$ is the pixel position, and μ_i is the pixel intensity at x_i. Two letters from E-MNIST dataset:



Optimal Transport: Probabilistic Interpretation

Problem: Find the Optimal (Mass) Transport from μ to ν, where moving a unit mass from x to y costs $c(x, y)$ and

- μ and ν are two probability measures on X and Y
- $c : X \times Y \to \mathbb{R}_+$ is a cost function

Optimal Transport: Probabilistic Interpretation

Problem: Find the Optimal (Mass) Transport from μ to ν, where moving a unit mass from x to y costs $c(x, y)$ and

- μ and ν are two probability measures on X and Y
- $c : X \times Y \to \mathbb{R}_+$ is a cost function

Definition 3 (Kantorovich-Rubinstein Functional)

The *Kantorovich-Rubinstein functional (I)* between μ and ν is defined as

$$\mathcal{W}_c(\mu, \nu) = \inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times Y} c(x, y)\pi(dxdy)$$

where $\Pi(\mu, \nu)$ is the set of all probability measures on $X \times Y$ that have marginals μ and ν.

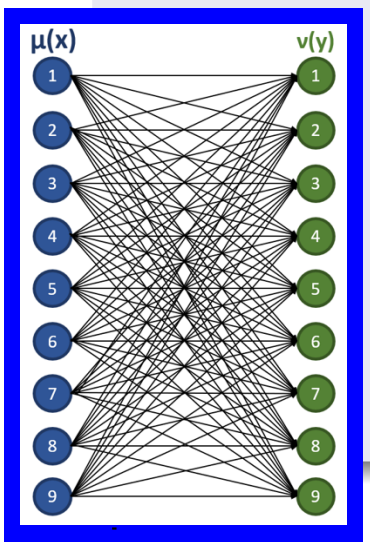
Discrete Kantorovich-Rubinstein is a Linear Program

Let $X = \{x_1, \dots, x_n\}$ and $Y = \{y_1, \dots, y_m\}$ be two discrete spaces.

Definition 4 (Kantorovich-Rubinstein Functional (II))

The Kantorovich-Rubinstein functional in the discrete setting is the following Linear Programming problem

$$
\begin{aligned}
(OT) \quad \mathcal{W}_c(\mu,\nu) = \min \quad & \sum_{x\in X}\sum_{y\in Y} c(x,y)\pi(x,y) \\
\text{s.t.} \quad & \sum_{y\in Y} \pi(x,y) = \mu(x) && \forall x \in X \\
& \sum_{x\in X} \pi(x,y) = \nu(y) && \forall y \in Y \\
& \pi(x,y) \geq 0.
\end{aligned}
$$



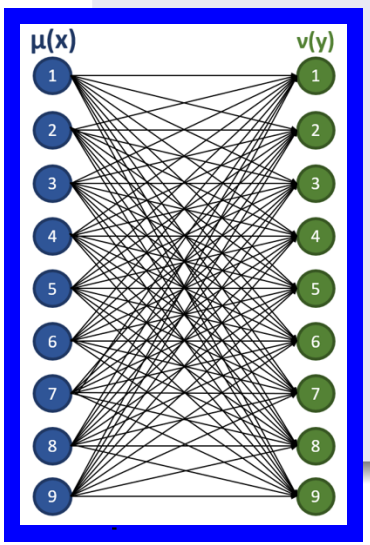
Discrete Kantorovich-Rubinstein is a Linear Program

Let $X = \{x_1, \dots, x_n\}$ and $Y = \{y_1, \dots, y_m\}$ be two discrete spaces.

Definition 4 (Kantorovich-Rubinstein Functional (II))

The Kantorovich-Rubinstein functional in the discrete setting is the following Linear Programming problem

$$
\begin{aligned}
(OT) \quad \mathcal{W}_c(\mu,\nu) = \min \quad & \sum_{x\in X}\sum_{y\in Y} c(x,y)\pi(x,y) \\
\text{s.t.} \quad & \sum_{y\in Y} \pi(x,y) = \mu(x) && \forall x \in X \\
& \sum_{x\in X} \pi(x,y) = \nu(y) && \forall y \in Y \\
& \pi(x,y) \geq 0.
\end{aligned}
$$



Discrete Kantorovich-Rubinstein is a Linear Program

Let $X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_m\}$ be two discrete spaces.

Definition 4 (Kantorovich-Rubinstein Functional (II))

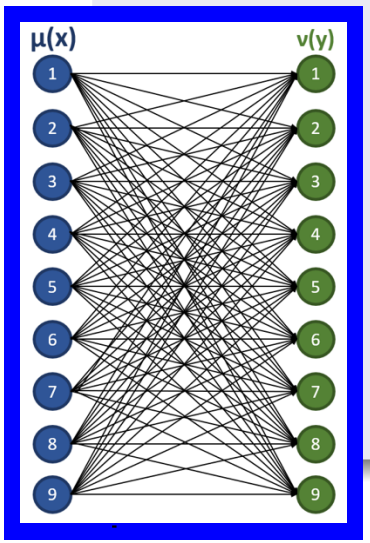
The Kantorovich-Rubinstein functional in the discrete setting is the following Linear Programming problem

$$(OT) \quad \mathcal{W}_c(\mu,\nu) = \min \sum_{x\in X}\sum_{y\in Y} c(x,y)\pi(x,y)$$

$$\text{s.t.} \quad \sum_{y\in Y} \pi(x,y) = \mu(x) \qquad \forall x \in X$$

$$\sum_{x\in X} \pi(x,y) = \nu(y) \qquad \forall y \in Y$$

$$\pi(x,y) \geq 0.$$



Definition 5 (Kantorovich-Wasserstein metric)

When $X = Y$ and $c(x,y) = d^\alpha(x,y)$, where d is a ground distance on X, the Kantorovich-Wasserstein metric of order $\alpha > 0$ is

$$W_\alpha(\mu,\nu) := \mathcal{W}_{d^\alpha}(\mu,\nu)^{\min(\frac{1}{\alpha},1)}.$$

Optimal Transport: From Monge to ...

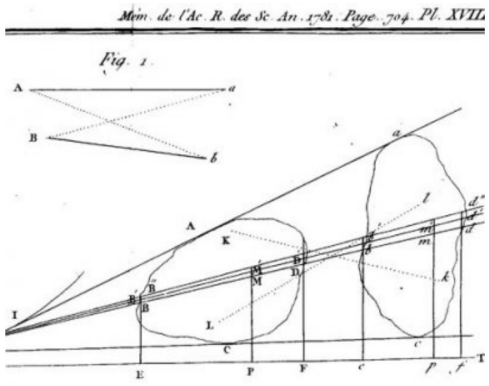
Optimal Transport originates from the work of Monge on finding the minimum cost plan for transporting a distribution of *molécules* from its origins to a given target distribution (1781).

MÉMOIRE
SUR LA
THÉORIE DES DÉBLAIS
ET DES REMBLAIS.

Par M. MONGE.

LORSQU'ON doit transporter des terres d'un lieu dans un autre, on a coutume de donner le nom de *Déblai* au volume des terres que l'on doit transporter, & le nom de *Remblai* à l'espace qu'elles doivent occuper après le transport.

Le prix du transport d'une molécule étant, toutes choses d'ailleurs égales, proportionnel à son poids & à l'espace qu'on lui fait parcourir, & par conséquent le prix du transport total devant être proportionnel à la somme des produits des molécules multipliées chacune par l'espace parcouru, il s'ensuit que le déblai & le remblai étant donnés de figure & de position, il n'est pas indifférent que telle molécule du déblai soit transportée dans tel ou tel autre endroit du remblai, mais qu'il y a une certaine distribution à faire des molécules du premier dans le second, d'après laquelle la somme de ces produits sera la moindre possible, & le prix du transport total sera un *minimum*.



Optimal Transport: From Monge to ...

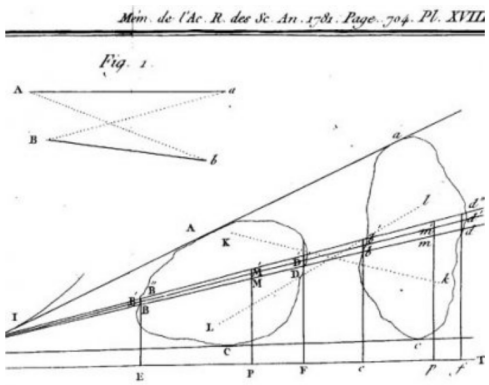
Optimal Transport originates from the work of Monge on finding the minimum cost plan for transporting a distribution of *molécules* from its origins to a given target distribution (1781).

MÉMOIRE
SUR LA
THÉORIE DES DÉBLAIS
ET DES REMBLAIS.

Par M. MONGE.

LORSQU'ON doit transporter des terres d'un lieu dans un autre, on a coutume de donner le nom de *Déblai* au volume des terres que l'on doit transporter, & le nom de *Remblai* à l'espace qu'elles doivent occuper après le transport.

Le prix du transport d'une molécule étant, toutes choses d'ailleurs égales, proportionnel à son poids & à l'espace qu'on lui fait parcourir, & par conséquent le prix du transport total devant être proportionnel à la somme des produits des molécules multipliées chacune par l'espace parcouru, il s'ensuit que le déblai & le remblai étant donnés de figure & de position, il n'est pas indifférent que telle molécule du déblai soit transportée dans tel ou tel autre endroit du remblai, mais qu'il y a une certaine distribution à faire des molécules du premier dans le second, d'après laquelle la somme de ces produits sera la moindre possible, & le prix du transport total sera un *minimum*.



In Combinatorial Optimization terms, the original Monge's problem is an Assignment Problem, which can be solved in strongly polynomial time with flow algorithms: $O(nm + n^2 \log(n))$ on general graphs and in $O(n^3)$ on bipartite complete graphs.

Ahuja, Magnanti, Orlin. Network flows: Theory, Algorithms, and Applications, 1988

Optimal Transport: From Monge to Kantorovich

An important contribution to Optimal Transport is due to Kantorovich, *Mathematical methods in the organization and planning of production*, (1939). Apart from the Duality theory of Linear Programming, Kantorovich is the father of the Transport Metric, a.k.a. the *Wasserstein Metric*

Figure 1. L. V. Kantorovich in his youth.

Long History of the Monge-Kantorovich Transportation Problem

(Marking the centennial of L.V. Kantorovich's birth!)

A. M. Vershik

Leonid Vital'evich Kantorovich (1912–1986) was one of the great mathematicians and economists of the twentieth century.

In 2012, the centenary of his birth was marked in St. Petersburg. Short histories were presented describing some of the main parts of his legacy, which continue in importance today: duality in linear programming, the so-called "Monge-Kantorovich transportation problem," and the "Kantorovich metric." Note that 2012 was also the 70th anniversary of the publication of his historic paper on the transport metric. The present article offers a somewhat expanded version of my talk on that occasion.

L. V. Kantorovich the Person

We remember Leonid Vital'evich Kantorovich for his massive contributions to foundations of mathematics, computational mathematics, and other areas, and in particular as one of the founders of mathematical economics.

He began as a child prodigy, entering Leningrad University at the age of 14. His first paper on descriptive set theory, which

reception was quite different. The ideas of Kantorovich, and Kantorovich himself, were not appreciated in Soviet Russia. For a long time—until the end of the 1950s—his ideas on mathematical economics were considered in official circles as anti-Marxist; consequently it was prohibited and even dangerous to study and develop them. Such a story, carried much farther, is well-known in Soviet biology ("lysenkovshchina").

So, between 1947 and the late 1950s Kantorovich never mentioned his ideas about mathematical economics and linear programming. My colleagues and I heard his lectures on functional analysis (later published as a book with G. P. Akilov) but knew nothing of his economically related work. He lectured openly on the subject only after the beginning of Khrushchev's "ottepel'" ("thaw"), the liberalization of 1957–1958.

L. V. Kantorovich (LV hereinafter) has left a rich record of the defence of scientific truth. A majority of the Soviet economists of the generation of the 1960s and 1970s were pupils of LV, and many mathematicians (including the author) considered themselves his pupils.

Optimal Transport: From Monge to Kantorovich

An important contribution to Optimal Transport is due to Kantorovich, *Mathematical methods in the organization and planning of production*, (1939). Apart from the Duality theory of Linear Programming, Kantorovich is the father of the Transport Metric, a.k.a. the *Wasserstein Metric*

Figure 1. L. V. Kantorovich in his youth.

Long History of the Monge-Kantorovich Transportation Problem

(Marking the centennial of L.V. Kantorovich's birth!)

A. M. Vershik

Leonid Vital'evich Kantorovich (1912–1986) was one of the great mathematicians and economists of the twentieth century.

In 2012, the centenary of his birth was marked in St. Petersburg. Short histories were presented describing some of the main parts of his legacy, which continue in importance today: duality in linear programming, the so-called "Monge-Kantorovich transportation problem," and the "Kantorovich metric." Note that 2012 was also the 70th anniversary of the publication of his historic paper on the transport metric. The present article offers a somewhat expanded version of my talk on that occasion.

L. V. Kantorovich the Person

We remember Leonid Vital'evich Kantorovich for his massive contributions to foundations of mathematics, computational mathematics, and other areas, and in particular as one of the founders of mathematical economics.

He began as a child prodigy, entering Leningrad University at the age of 14. His first paper on descriptive set theory, which

reception was quite different. The ideas of Kantorovich, and Kantorovich himself, were not appreciated in Soviet Russia. For a long time—until the end of the 1950s—his ideas on mathematical economics were considered in official circles as anti-Marxist; consequently it was prohibited and even dangerous to study and develop them. Such a story, carried much farther, is well-known in Soviet biology ("lysenkovshchina").

So, between 1947 and the late 1950s Kantorovich never mentioned his ideas about mathematical economics and linear programming. My colleagues and I heard his lectures on functional analysis (later published as a book with G. P. Akilov) but knew nothing of his economically related work. He lectured openly on the subject only after the beginning of Khrushchev's "ottepel'" ("thaw"), the liberalization of 1957–1958.

L. V. Kantorovich (LV hereinafter) has left a rich record of the defence of scientific truth. A majority of the Soviet economists of the generation of the 1960s and 1970s were pupils of LV, and many mathematicians (including the author) considered themselves his pupils.

The Kantorovich reformulation of Monge's problem gives a Transportation Problem: the complexity raises to $O(n\log(n)(m + n\log(n)))$ on general graphs, and to $O(n^3\log(n))$ for bipartite complete graphs.

Orlin. A faster strongly polynomial minimum cost flow algorithm.
Operations research 41(2) 338-350 (1993)

Optimal Transport: State-of-the-art (?)

- We compute Wasserstein Distance using a Parallel Network Simplex algorithm (the optimal solution is sparse).

Optimal Transport: State-of-the-art (?)

- We compute Wasserstein Distance using a Parallel Network Simplex algorithm (the optimal solution is sparse).

- Modern Computational Optimal Transport for Machine Learning has focused on entropic regularized costs ($c_{ij}x_{ij} + \epsilon x_{ij}\log(x_{ij})$), which add a regularization parameter depending on a parameter ϵ. For $\epsilon \to 0$, we recover the standard Optimal Transport, (in theory).

Optimal Transport: State-of-the-art (?)

- We compute Wasserstein Distance using a Parallel Network Simplex algorithm (the optimal solution is sparse).

- Modern Computational Optimal Transport for Machine Learning has focused on entropic regularized costs ($c_{ij}x_{ij} + \epsilon x_{ij}\log(x_{ij})$), which add a regularization parameter depending on a parameter ϵ. For $\epsilon \to 0$, we recover the standard Optimal Transport, (in theory).
- Using the Sinkhorn algorithm, the regularized entropic version can compute the optimal distance in time $O(\frac{n^2}{\epsilon^2})$ [AWR17],

Optimal Transport: State-of-the-art (?)

- We compute Wasserstein Distance using a Parallel Network Simplex algorithm (the optimal solution is sparse).

- Modern Computational Optimal Transport for Machine Learning has focused on entropic regularized costs ($c_{ij}x_{ij} + \epsilon x_{ij}\log(x_{ij})$), which add a regularization parameter depending on a parameter ϵ. For $\epsilon \to 0$, we recover the standard Optimal Transport, (in theory).
- Using the Sinkhorn algorithm, the regularized entropic version can compute the optimal distance in time $O(\frac{n^2}{\epsilon^2})$ [AWR17], but we do not get the optimal transport plan (!)

Optimal Transport: State-of-the-art (?)

- We compute Wasserstein Distance using a Parallel Network Simplex algorithm (the optimal solution is sparse).

- Modern Computational Optimal Transport for Machine Learning has focused on entropic regularized costs ($c_{ij}x_{ij} + \epsilon x_{ij}\log(x_{ij})$), which add a regularization parameter depending on a parameter ϵ. For $\epsilon \to 0$, we recover the standard Optimal Transport, (in theory).
- Using the Sinkhorn algorithm, the regularized entropic version can compute the optimal distance in time $O(\frac{n^2}{\epsilon^2})$ [AWR17], but we do not get the optimal transport plan (!)
- The Sinkhorn algorithm is highly parallelizable,

Optimal Transport: State-of-the-art (?)

- We compute Wasserstein Distance using a Parallel Network Simplex algorithm (the optimal solution is sparse).

- Modern Computational Optimal Transport for Machine Learning has focused on entropic regularized costs ($c_{ij}x_{ij} + \epsilon x_{ij}\log(x_{ij})$), which add a regularization parameter depending on a parameter ϵ. For $\epsilon \to 0$, we recover the standard Optimal Transport, (in theory).
- Using the Sinkhorn algorithm, the regularized entropic version can compute the optimal distance in time $O(\frac{n^2}{\epsilon^2})$ [AWR17], but we do not get the optimal transport plan (!)
- The Sinkhorn algorithm is highly parallelizable, but suffers from stability numerical issues.

Optimal Transport: State-of-the-art (?)

- We compute Wasserstein Distance using a Parallel Network Simplex algorithm (the optimal solution is sparse).

- Modern Computational Optimal Transport for Machine Learning has focused on entropic regularized costs ($c_{ij}x_{ij} + \epsilon x_{ij}\log(x_{ij})$), which add a regularization parameter depending on a parameter ϵ. For $\epsilon \to 0$, we recover the standard Optimal Transport, (in theory).
- Using the Sinkhorn algorithm, the regularized entropic version can compute the optimal distance in time $O(\frac{n^2}{\epsilon^2})$ [AWR17], but we do not get the optimal transport plan (!)
- The Sinkhorn algorithm is highly parallelizable, but suffers from stability numerical issues.
- The Sinkhorn algorithm computes a smooth plan, with a less concentrate optimal plan.

Parallel Network Simplex (IPAM Workshop 2021)

			Average runtime in seconds				
Size	Meth.	Num. Vars	Total	(speedup)	Master	Pricing	(speedup)
32×32	Full	1048572	0.29	(×3.8)			
	CPU	3044	0.33	(×4.4)	0.02	0.31	(×4.9)
	GPU	3036	**0.08**		0.01	0.06	
	Sinkhorn		0.67				
64×64	Full	16777212	13.7	(×11.2)			
	CPU	12240	4.4	(×3.6)	0.8	3.6	(×6.6)
	GPU	12266	**1.2**		0.7	0.5	
	Sinkhorn		3.5				
128×128	Full (∗)	268435452	1542.3	(×56)			
	CPU	49361	64.2	(×2.3)	23.2	41.0	(×6.4)
	GPU	49277	27.6		21.1	6.4	
	Sinkhorn		**21.7**				

- **Full**: Complete bipartite model with CPU Parallel Simplex
- **CPU**: Column generation with CPU Parallel Simplex
- **GPU**: Column generation with GPU Parallel Simplex (CUDA kernel)
- Sinkhorn: Stabilized Sinkhorn algorithm (C++), CPU Parallel (Schmitzer, 2016)

NOTE: Speedup refers only to GPU vs. Full and CPU. Averages over 90 instances per row.

Interpolation

Lemma 6 (Interpolation between two measures (Chap. 7, in [PC+19]))

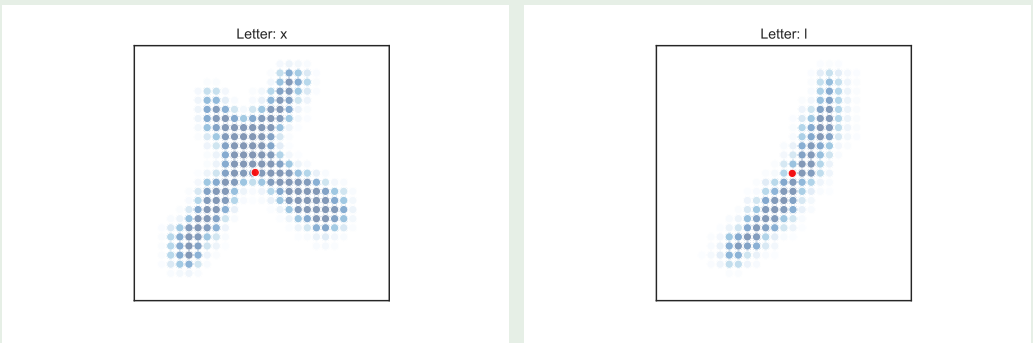
Given (i) two weights $(\lambda_1, \lambda_2) \in \mathbb{R}_+$ *satisfying* $\lambda_1 + \lambda_2 = 1$, *(ii) two discrete measures* μ *and* ν *defined on a grid of points* X*:*

$$\mu = \sum_{i=1}^{n} \mu_i \delta_{x_i} \quad \text{and} \quad \nu = \sum_{j=1}^{n} \nu_j \delta_{y_j},$$

and (iii) an optimal transportation plan π^* *minimizing the functional* $W_2^2(\mu, \nu)$, *that is, an optimal solution of Problem (OT) with* $c(x_i, y_j) = ||x_i - y_j||_2^2$, *the interpolated average measure* ρ *between* μ *and* ν *is*

$$\rho = f(\mu, \nu, \lambda_1, \lambda_2) := \sum_{i=1}^{n} \sum_{j=1}^{n} \pi_{ij}^* \delta_{(\lambda_1 x_i + \lambda_2 y_j)}. \tag{1}$$

Example 5: Interpolation of two letters (DEMO VIDEO)



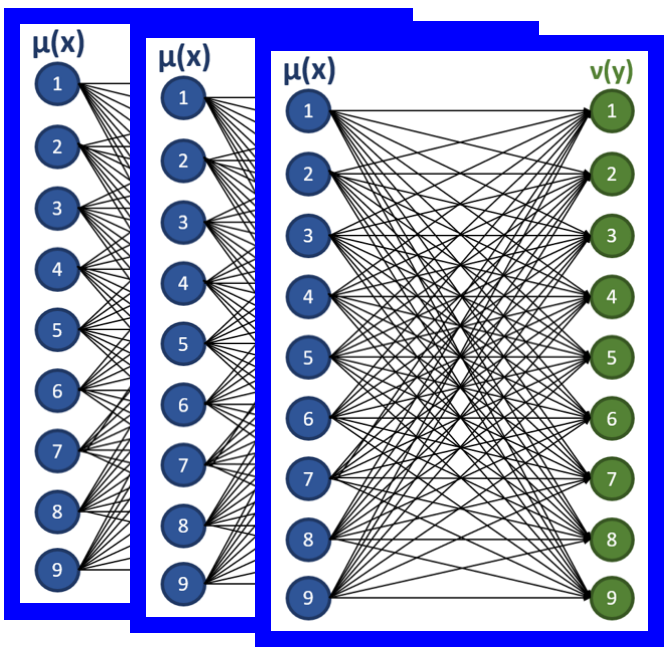
Section 3

Outline

Wasserstein Barycenter via LP [CPAIOR, 2019]

If μ_k are discrete probability measures and μ_{ik} is the i-th element of the measure μ_k, if we use the cost $c(x_i, y_j) = ||x_i - y_j||_2^2$ in the objective function, and if we fix a set of possible locations y_j for the support points of the barycenter ρ, then Wasserstein Barycenter problem is equivalent to the following Linear Program:

$$\mathcal{B}(\boldsymbol{\mu}, \boldsymbol{\lambda}) = \min \quad \sum_{k=1}^{m} \lambda_k \left(\sum_{i=1}^{n} \sum_{j=1}^{n} ||x_i - y_j||_2^2 \ \pi_{ijk} \right) \tag{2}$$



...

$$\text{s.t.} \quad \sum_{j=1}^{n} \pi_{ijk} = \mu_{ik} \qquad i = 1, \ldots, n, k = 1, \ldots, m \tag{3}$$

$$\sum_{i=1}^{n} \pi_{ijk} = \rho_j \qquad j = 1, \ldots, n, k = 1, \ldots, m \tag{4}$$

$$\sum_{j=1}^{n} \rho_j = 1 \tag{5}$$

$$\pi_{ij_k} \geq 0, \rho_j \geq 0, \ \ i, j = 1, \ldots, n, k = 1, \ldots, m \tag{6}$$

REMARK: We can solve this LP using an Interior Point Algorithm to a given accuracy.

Our Sequential Heuristic

Given the input measures with a fixed order $\mu_1, \ldots, \mu_k$, our heuristic computes $\bar{\rho}^{IH}$ by solving the following recursion:

$$\Theta^{(i)} = \begin{cases} \mu_1 & \text{if } i = 1, \\ f\left(\mu_i, \Theta^{(i-1)}, \frac{1}{i}, \frac{i-1}{i}\right) & \text{if } i > 1, \end{cases} \tag{7}$$

$$\bar{\rho}^{IH} = \Theta^{(i)}. \tag{8}$$

We call this heuristic the **Iterative Heuristic**.

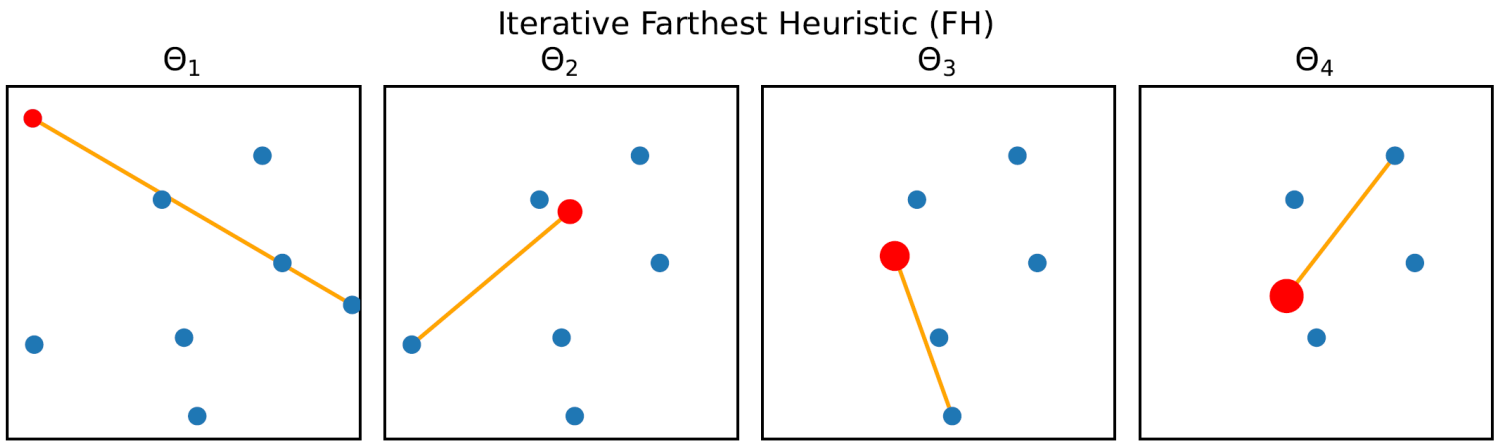
Our Sequential Heuristic

Given the input measures with a fixed order $\mu_1, \ldots, \mu_k$, our heuristic computes $\bar{\rho}^{IH}$ by solving the following recursion:

$$\Theta^{(i)} = \begin{cases} \mu_1 & \text{if } i = 1, \\ f\left(\mu_i, \Theta^{(i-1)}, \frac{1}{i}, \frac{i-1}{i}\right) & \text{if } i > 1, \end{cases} \tag{7}$$

$$\bar{\rho}^{IH} = \Theta^{(i)}. \tag{8}$$

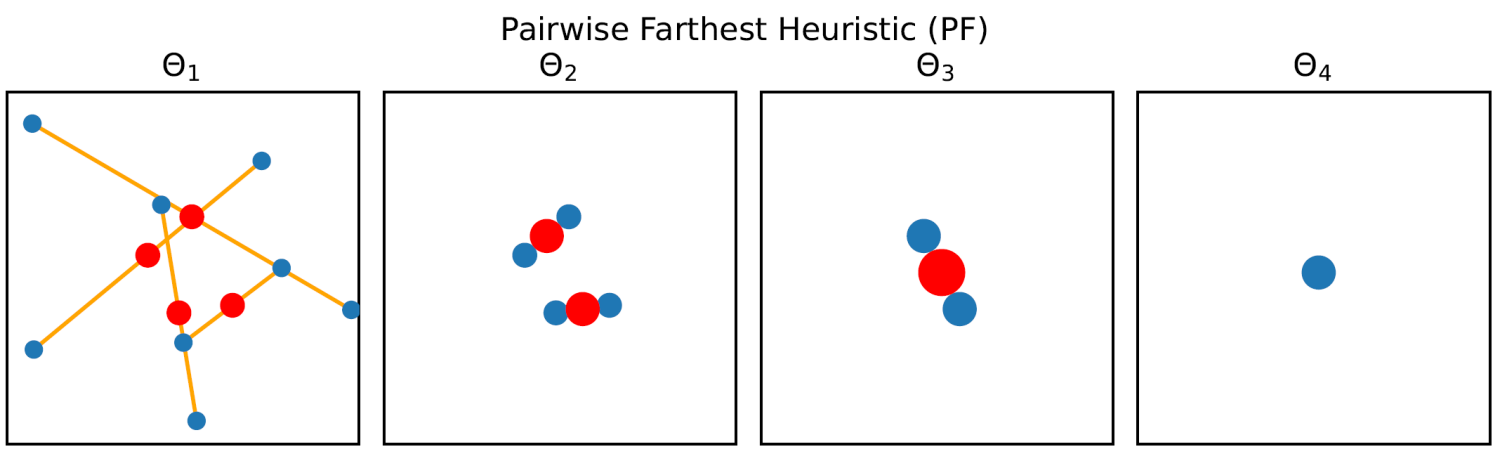
We call this heuristic the **Iterative Heuristic**. Indeed, this heuristic is order sensitive.



Our Pairwise Heuristic

We begin with a vector $\mathbf{\Theta}^{(0)}$ that is equal to the input sequence of measures. Then, at each iteration i, we compute the barycenter of every consecutive pair of measures $\Theta_{2j-1}^{(i-1)}$ and $\Theta_{2j}^{(i-1)}$ with weights equal to $\frac{1}{2}$, and we get a new vector $\mathbf{\Theta}^{(i)}$ of size $q_i = \frac{m}{2^i}$. More formally, the **Pairwise Heuristic** is defined by the following procedure:

$$
\begin{aligned}
\mathbf{\Theta}^{(0)} &= \{\mu_1, \ldots, \mu_k\}, \boldsymbol{\lambda}^{(0)} = \{1, \ldots, 1\} \\
\mathbf{\Theta}^{(i)} &= \left\{ f\left(\Theta_{2j-1}^{(i-1)}, \Theta_{2j}^{(i-1)}, \frac{\lambda_{2j-1}^{(i-1)}}{\lambda_{2j-1}^{(i-1)} + \lambda_{2j}^{(i-1)}}, \frac{\lambda_{2j}^{(i-1)}}{\lambda_{2j-1}^{(i-1)} + \lambda_{2j}^{(i-1)}} \right) \right\}, \\
&\qquad\qquad j = 1, \ldots, q_i \\
\boldsymbol{\lambda}^{(i)} &= \{\lambda_{2j-1}^{(i-1)} + \lambda_{2j}^{(i-1)}\}, \quad j = 1, \ldots, q_i \\
\bar{\rho}^{PR} &= \mathbf{\Theta}^{(h)}
\end{aligned}
$$



Section 4

Primal Heuristics [CPAIOR2020]

Comparison with Optimal solution (via Gurobi), Euclidean barycenters, and Convolutional Wasserstein Barycenters (using POT [FCG+21]), with the following heuristics:

1. Iterative Random
2. Iterative Farthest
3. Pairwise Random
4. Pairwise Farthest

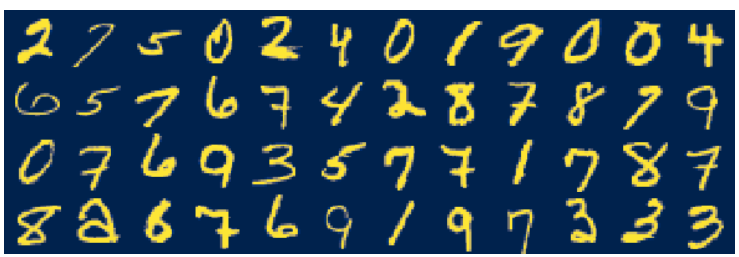
Primal Heuristics [CPAIOR2020]

Comparison with Optimal solution (via Gurobi), Euclidean barycenters, and Convolutional Wasserstein Barycenters (using POT [FCG+21]), with the following heuristics:

1. Iterative Random
2. Iterative Farthest
3. Pairwise Random
4. Pairwise Farthest

We tested the following dataset of images:

1. The MNIST handwritten digit dataset [LCB]
2. The Fashion MNIST dataset [XRV17].
3. Rescaled and translated images from the MNIST, as proposed in [SDGP+15]



The MNIST handwritten digit dataset [LCB]

The MNIST handwritten digit dataset [LCB]

$\overline{opt\ gap}$: 00.000 %
$\overline{runtime}$: 07057 *s*

$\overline{opt\ gap}$: 15.119 %
$\overline{runtime}$: 00001 *s*

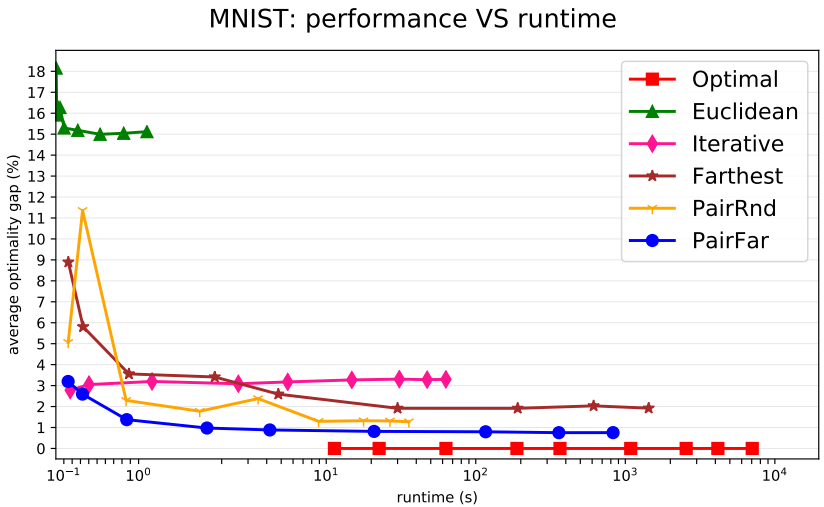
$\overline{opt\ gap}$: 03.298 %
$\overline{runtime}$: 00063 *s*

$\overline{opt\ gap}$: 01.924 %
$\overline{runtime}$: 01432 *s*

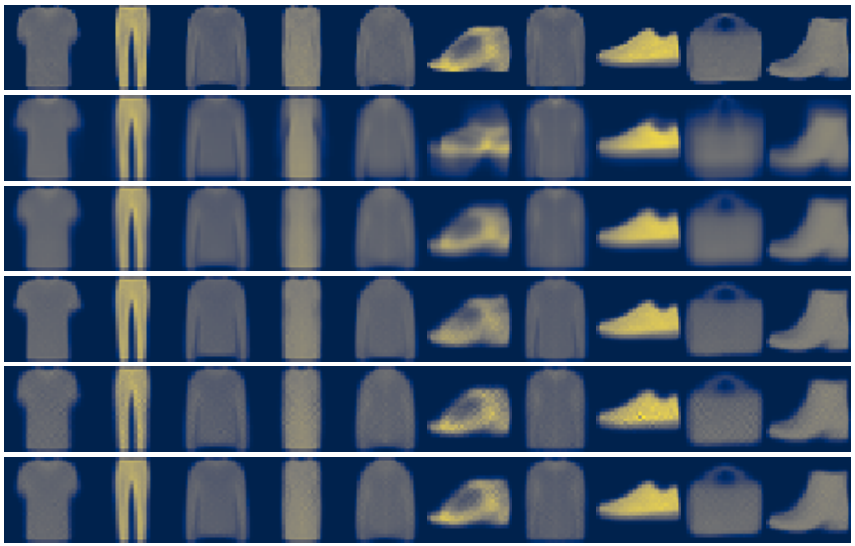
$\overline{opt\ gap}$: 01.275 %
$\overline{runtime}$: 00035 *s*

$\overline{opt\ gap}$: 00.755 %
$\overline{runtime}$: 00828 *s*

Scaling with the number of input images $k \in \{10, 15, 20, 50, 75, 100, 150, 200, 250\}$



The Fashion-MNIST by Zalando



$\overline{opt\ gap}$: 00.000 %
$\overline{runtime}$: 11360 *s*

$\overline{opt\ gap}$: 07.454 %
$\overline{runtime}$: 00001 *s*

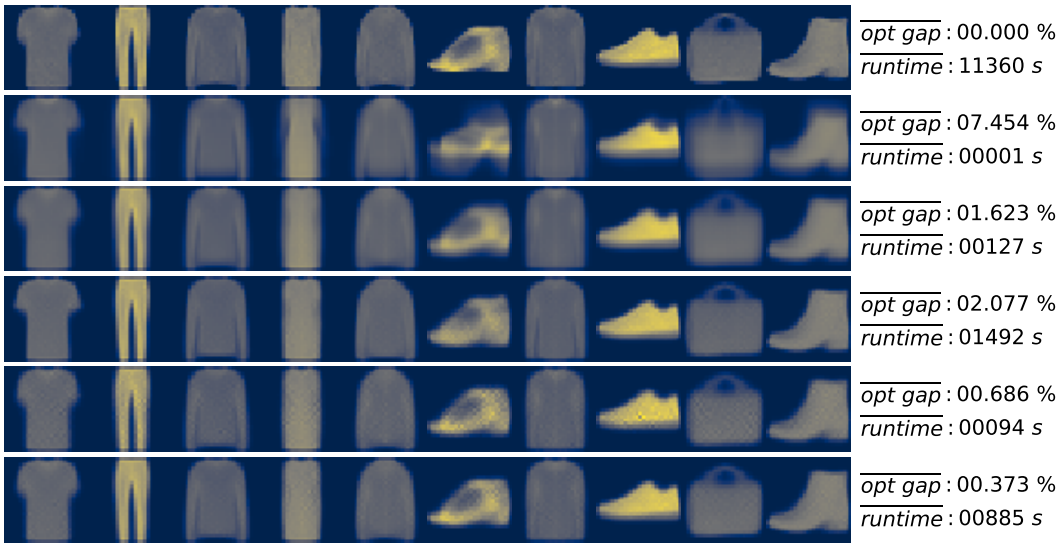
$\overline{opt\ gap}$: 01.623 %
$\overline{runtime}$: 00127 *s*

$\overline{opt\ gap}$: 02.077 %
$\overline{runtime}$: 01492 *s*

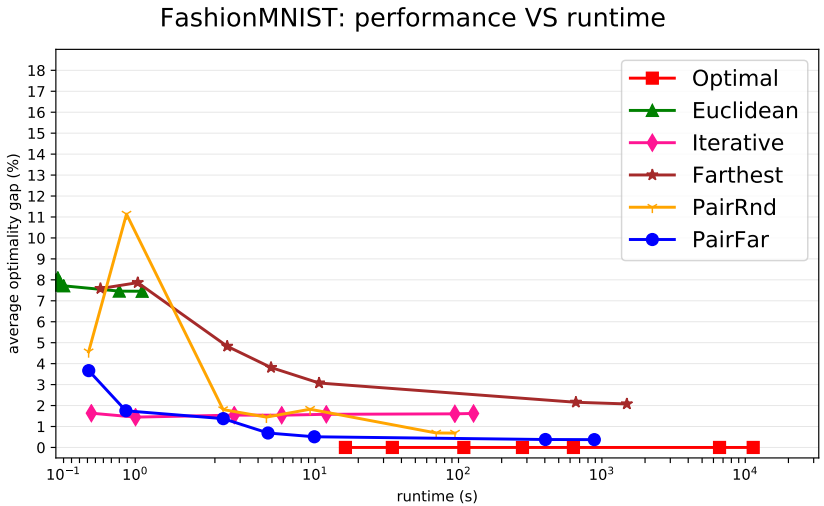
$\overline{opt\ gap}$: 00.686 %
$\overline{runtime}$: 00094 *s*

$\overline{opt\ gap}$: 00.373 %
$\overline{runtime}$: 00885 *s*

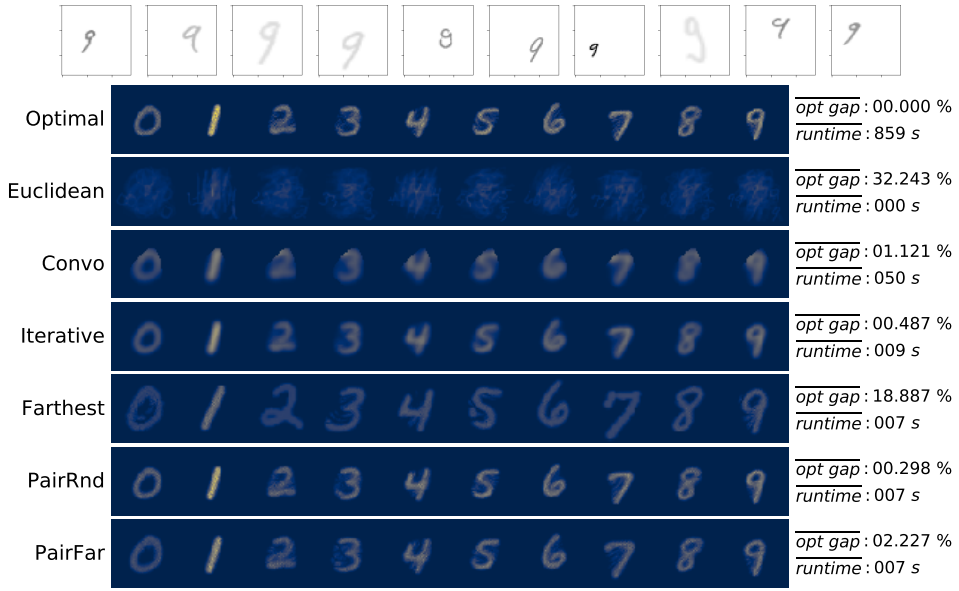
The Fashion-MNIST by Zalando



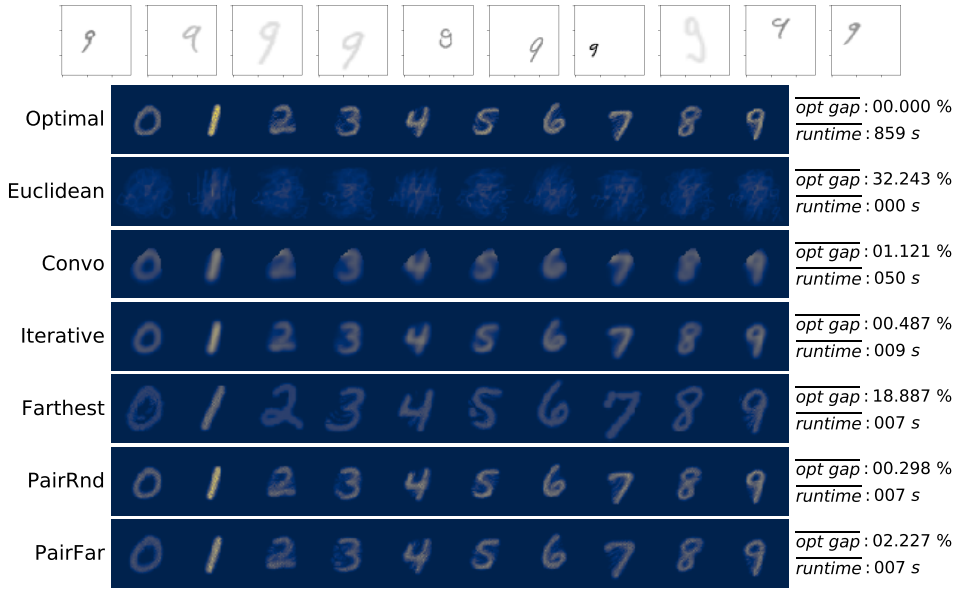
Scaling with the number of input images $k \in \{10, 15, 20, 50, 75, 100, 150, 200, 250\}$



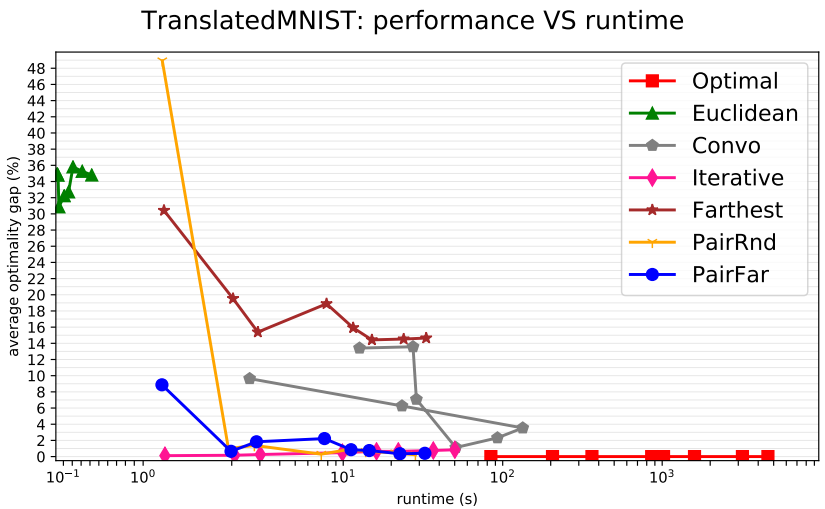
The Translated-MNIST dataset



The Translated-MNIST dataset

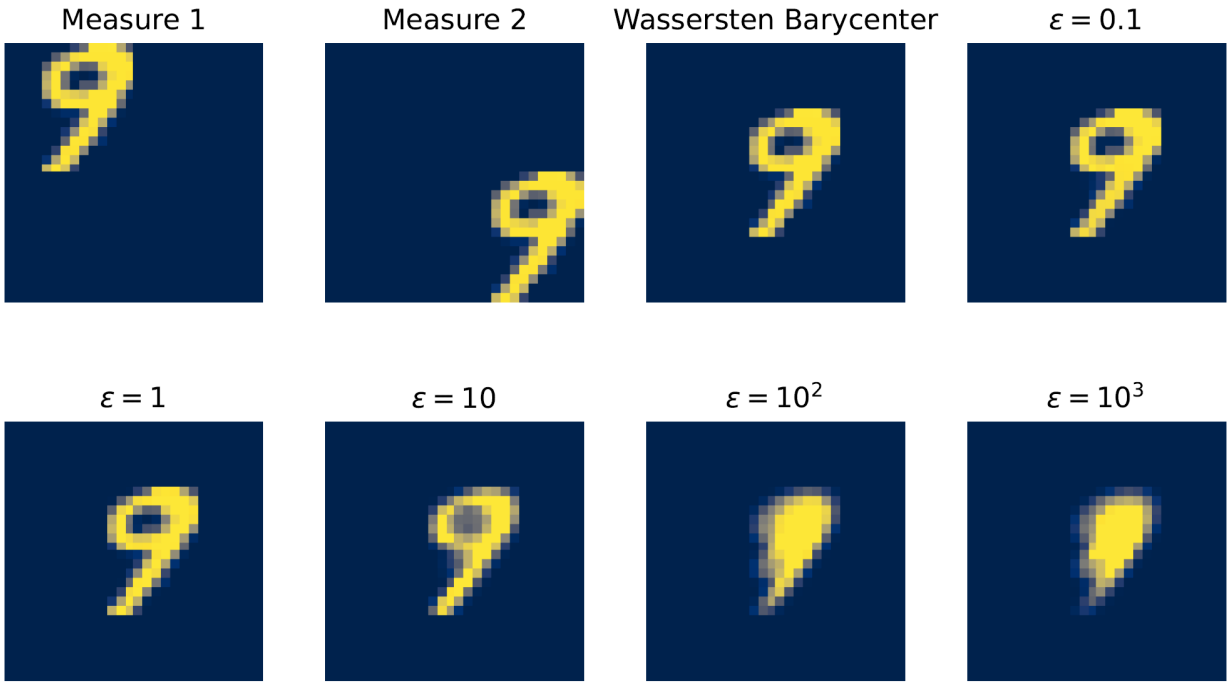


Scaling with the number of input images $k \in \{10, 15, 20, 50, 75, 100, 150, 200, 250\}$



The numerical issues with Convolutional Barycenters

Entropic-regularized Wasserstein barycenters [FCG+21, SDGP+15], for different values of the regularization parameter ϵ, compared with the exact barycenter.



REMARK: To find a single value of the regularization parameter ϵ that does not raise numerical issues on all the images of a given dataset is a challenge.

Section 5

Comparison with MAAIPM

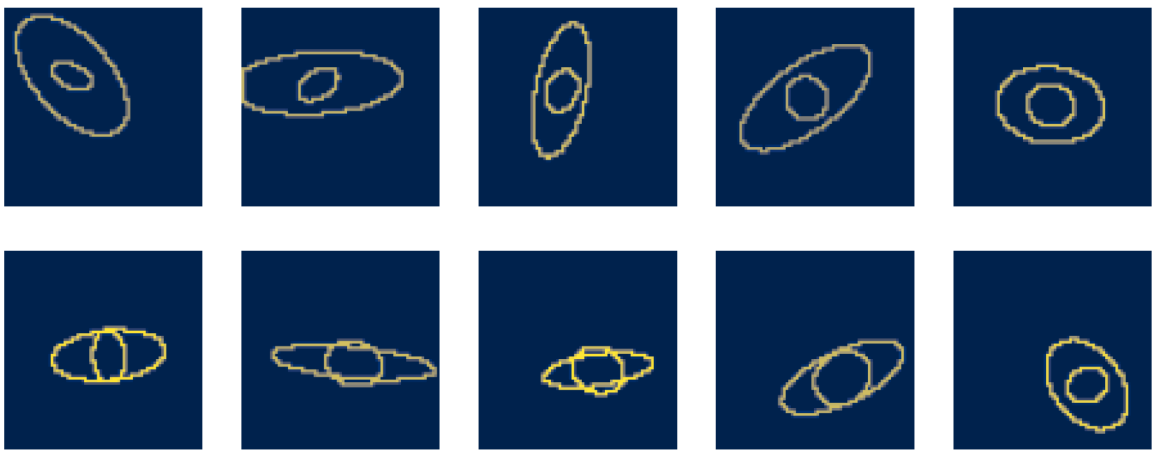
Recently, we compared with another approach based on a custom implementation of an Interior Point Method (IPM), called the **Matrix-based Adaptive Alternating Interior-Point Method (MAAIPM)**.

MAAIPM uses a Mehrotra's Predictor-Corrector algorithm. By exploiting the block-diagonal structure of the matrices that arise in the barycenter problem they can solve the problem with $O(kn^3)$ operations.

Comparison with MAAIPM

Recently, we compared with another approach based on a custom implementation of an Interior Point Method (IPM), called the **Matrix-based Adaptive Alternating Interior-Point Method (MAAIPM)**.

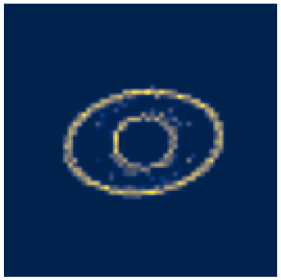
MAAIPM uses a Mehrotra's Predictor-Corrector algorithm. By exploiting the block-diagonal structure of the matrices that arise in the barycenter problem they can solve the problem with $O(kn^3)$ operations.



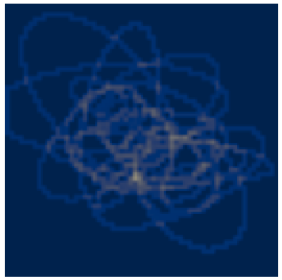
Ge, D., Wang, H., Xiong, Z. and Ye, Y., 2019. Interior-point methods strike back: Solving the Wasserstein barycenter problem. In: Advances in Neural Information Processing Systems. Vol. 32. 2019. arXiv preprint arXiv:1905.12895.

Nested-ellipses dataset: results

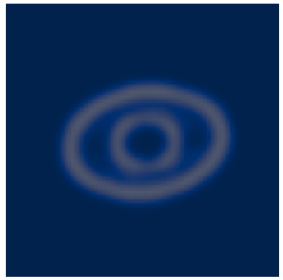
Wasserstein
Ran in 153.95 s
Opt. gap = 0.0 %



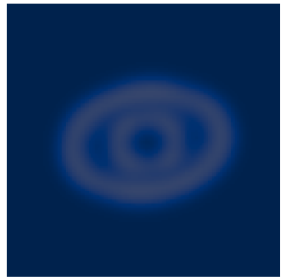
Euclidean
Ran in 0.04 s
Opt. gap = 32.035 %



Entropic $\varepsilon = 3.5$
Ran in 40.65 s
Opt. gap = 0.893 %

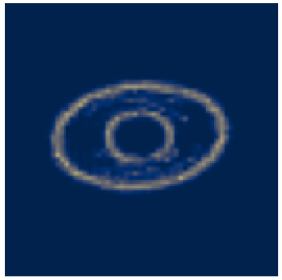


Entropic $\varepsilon = 10$
Ran in 8.4 s
Opt. gap = 2.549 %

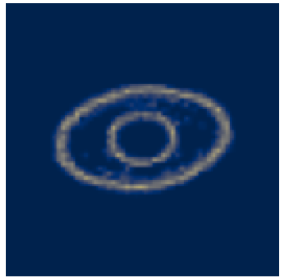


MAAIPM
Ran in 137.87 s
Opt. gap = 0.058 %

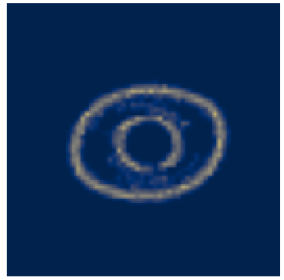
Farthest H
Ran in 4.23 s
Opt. gap = 4.077 %



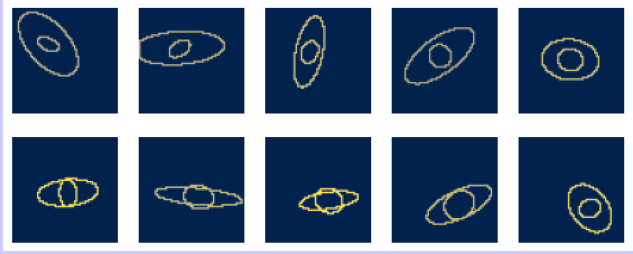
PairRand
Ran in 5.47 s
Opt. gap = 2.082 %



PairFarthest
Ran in 4.33 s
Opt. gap = 0.459 %



Input ellipses:



Conclusions

- Wasserstein Barycenters raises interesting computational challenges

Conclusions

- Wasserstein Barycenters raises interesting computational challenges
- The **block structure** of the LP formulation should be exploited: in our work, it is used to design primal heuristics

Conclusions

- Wasserstein Barycenters raises interesting computational challenges
- The **block structure** of the LP formulation should be exploited: in our work, it is used to design primal heuristics
- We are working on a new parallel implementation of the primal heuristics that uses **different ordering criteria**

Conclusions

- Wasserstein Barycenters raises interesting computational challenges
- The **block structure** of the LP formulation should be exploited: in our work, it is used to design primal heuristics
- We are working on a new parallel implementation of the primal heuristics that uses **different ordering criteria**
- **OPEN QUESTION:** Does exist an ordering of the input measures that guarantees a worst-case optimality gap?

Conclusions

- Wasserstein Barycenters raises interesting computational challenges
- The **block structure** of the LP formulation should be exploited: in our work, it is used to design primal heuristics
- We are working on a new parallel implementation of the primal heuristics that uses **different ordering criteria**
- **OPEN QUESTION:** Does exist an ordering of the input measures that guarantees a worst-case optimality gap?

MARKETING: A version of the CPU Parallel Simplex algorithm, customized for Spatial Statistics applications, is available at:

eurostat

`https://github.com/eurostat/Spatial-KWD`

Questions

Jason Altschuler, Jonathan Weed, and Philippe Rigollet.

Near-linear time approximation algorithms for optimal transport via sinkhorn iteration.

arXiv preprint arXiv:1705.09634, 2017.

Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z. Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, Léo Gautheron, Nathalie T.H. Gayraud, Hicham Janati, Alain Rakotomamonjy, Ievgen Redko, Antoine Rolet, Antony Schutz, Vivien Seguy, Danica J. Sutherland, Romain Tavenard, Alexander Tong, and Titouan Vayer.

Pot: Python optimal transport.

Journal of Machine Learning Research, 22(78):1–8, 2021.

Yann LeCun, Corinna Cortes, and Christopher J.C. Burges.

MNIST dataset.

Accessed: 2019-12-03, `http://yann.lecun.com/exdb/mnist/`.

Gabriel Peyré, Marco Cuturi, et al.

Computational optimal transport.

Foundations and Trends® in Machine Learning, 11(5-6):355–607, 2019.

Justin Solomon, Fernando De Goes, Gabriel Peyré, Marco Cuturi, Adrian Butscher, Andy Nguyen, Tao Du, and Leonidas Guibas.

Convolutional Wasserstein distances: Efficient Optimal Transportation on geometric domains.

ACM Transactions on Graphics, 34(4):66, 2015.

Han Xiao, Kashif Rasul, and Roland Vollgraf.

Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms.

arXiv preprint arXiv:1708.07747, 2017.