UC San Diego

Embedding Learning by Optimal Transport

Presenter: An Yan, Fanbo Xiang, Yiming Zhang

March 05, 2019

Outline

- Wasserstein Distance
  - Optimal Transport
  - Exact Algorithm
- Learning Wasserstein Embeddings
- Entropic Transport
  - Entropic Regularization
  - Sinkhorn Divergence
- Learning Entropic Wasserstein Embeddings

Review: Optimal Transport

Discrete Kantorovish formulation(Earth mover's distance)

Discrete distributions $\mathbf{a} \in \mathbb{R}_+^n$, $\mathbf{b} \in \mathbb{R}_+^m$. Cost matrix $\mathbf{C} \in \mathbb{R}_+^{n\times m}$. $\mathbf{C}_{i,j}$ denotes the unit cost of transporting mass from ith point in $\mathbf{a}$ to jth point in $\mathbf{b}$.

$$\mathbf{U}(a, b) = \{\mathbf{P} \in \mathbb{R}_+^{n\times m} : \mathbf{P}\mathbb{1}_m = \mathbf{a}, \mathbf{P}^T\mathbb{1}_n = \mathbf{b}\}$$

$\mathbf{P}_{i,j}$ denotes how much mass from ith point in $\mathbf{a}$ is transported to the jth point in $\mathbf{b}$. $\mathbf{U}(a, b)$ is all valid transport plans. $\mathbf{P}$ is known as a coupling matrix.

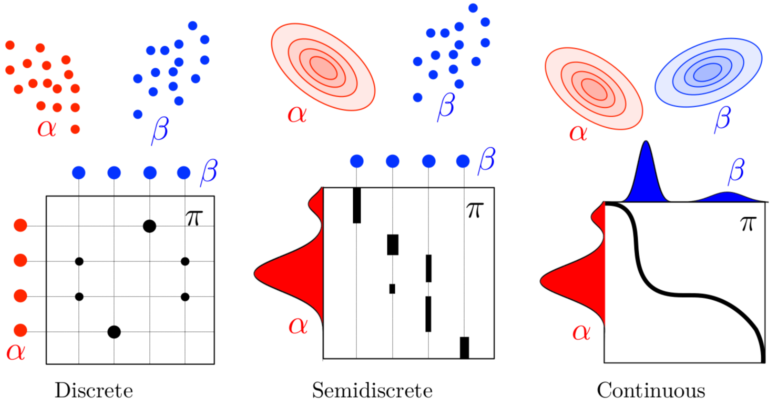
(Discrete) Optimal transport

A transport plan is optimal if it has the lowest cost.

$$L_{\mathbf{C}}(\mathbf{a}, \mathbf{b}) = \min_{\mathbf{P}\in\mathbf{U}(\mathbf{a},\mathbf{b})} \sum_{i,j} \mathbf{C}_{i,j}\mathbf{P}_{i,j}$$

Review: Optimal Transport

Moving mass from 1 distribution to the other.



Review: Optimal Transport

General formulation

$$\mathcal{L}_C(\alpha, \beta) = \min_{\pi \in \mathcal{U}(\alpha,\beta)} \int_{\mathcal{X}\times\mathcal{Y}} c(x, y) d\pi(x, y)$$

Probabilistic interpretation

$$\mathcal{L}_C(\alpha, \beta) = \min_{X,Y} \{\mathbb{E}(c(X, Y)) : X \sim \alpha, Y \sim \beta\}$$

Intuition

Optimal transport gives a distance measure between probability distributions.

Wasserstein Distance

A special case of optimal transport. "A natural way to lift ground distance to distribution distance."

Definition

Let $P_p(\Omega)$ be the set of Borel probability measures with finite pth moment defined on a given metric space (Ω, d). The p-Wasserstein metric W_p, for $p \geq 1$, on $P_p(\Omega)$ between distribution μ and ν, is defined as

$$W_p(\mu, \nu) = \Big(\min_{\gamma \in \mathcal{U}(\mu,\nu)} \int_{\Omega \times \Omega} d^p(x, y) d\gamma(x, y) \Big)^{\frac{1}{p}}$$

1-Wasserstein Distance

Primal Problem

$$KP(\mu,\nu) = \min_{\gamma} \int_{\Omega\times\Omega} d(x,y)d\gamma(x,y)$$

$$s.t. \quad \int_Y d\gamma(x,y) = p(x), \int_X d\gamma(x,y) = q(y)$$

$$\gamma(x,y) \geq 0$$

Kantorovich-Rubinstein theorem

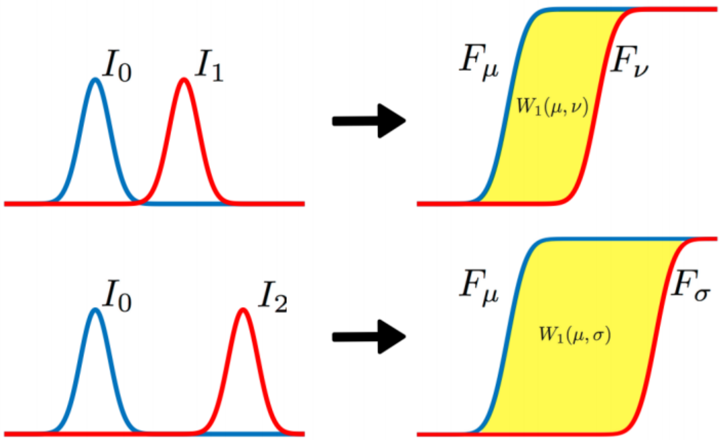
$$DP(\mu,\nu) = \max_{\phi\in Lip_1(X)} \int_X \phi(x)p(x)dx - \int_X \phi(x)q(x)dx$$

$$DP(\mu,\nu) = \max_{\phi\in Lip_1(X)} \mathbb{E}_p\phi(x) - \mathbb{E}_q\phi(x)$$

$$Lip_1(X) = \{\phi : |\phi(x) - \phi(y)| \leq d(x,y)\}, \forall x,y \in X$$

1-Wasserstein Distance

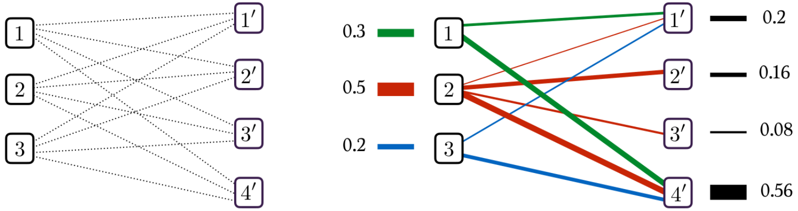
1-D: area between CDF.



Algorithm for Optimal Transport

Discrete problem: linear programming

Can be formulated as a minimum cost maximum flow problem.



If the distributions are uniform with the same number of elements. The problem further reduces to a minimum cost bipartite matching.

Any Questions?

Drawbacks of other distances

Let $X \sim P$ and $Y \sim Q$ and let the densities be p and q. Assume $X, Y \in \mathbb{R}^d$

Other distance functions

- Total Variation: $\sup_A |P(A) - Q(A)| = \frac{1}{2} \int |p - q|$
- Hellinger: $\sqrt{\int (\sqrt{p} - \sqrt{q})^2}$
- L_2: $\int (p - q)^2$

Drawbacks of other distances

Drawbacks

- Provide no information about why the distributions differ
- Problematic when comparing discrete to continuous
  - e.g. uniform P on $[0, 1]$ and uniform Q on $\{0, 1/N, 2/N, ..., 1\}$
- Ignore the underlying geometry of the space

Drawbacks of other distances

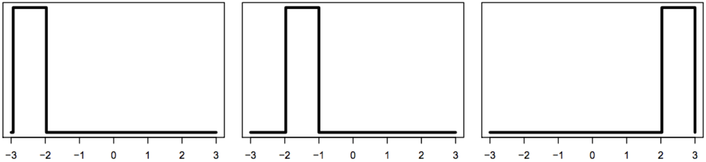


Figure: Three densities p1, p2, p3. Each pair has the same distance in L1, L2, Hellinger etc. But in Wasserstein distance, p1 and p2 are close.

Drawbacks of other distances

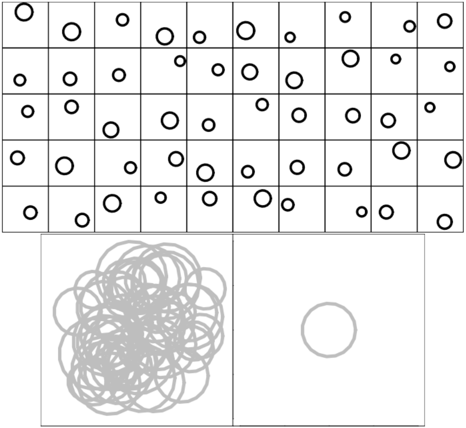


Figure: Top: Some random circles. Bottom left: Euclidean average of the circles. Bottom right: Wasserstein barycenter.

Drawbacks of other distances

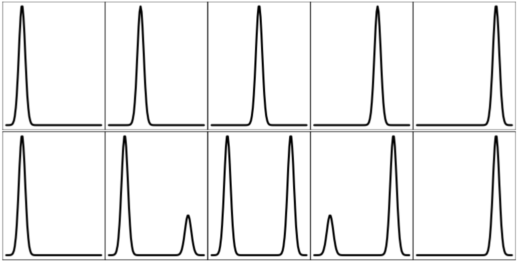


Figure: Top row: Geodesic path from P0 to P1. Bottom row: Euclidean path from P0 to P1.

Learning Wasserstein Embeddings

Motivation

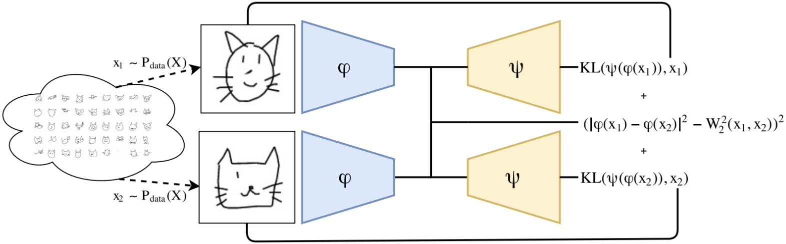
- Solving LP for computing Wasserstein distance between discrete distributions (histograms) is super cubic in complexity
- Some approximation techniques
  - slicing techniques
  - entropic regularization
  - stochastic optimization
- However, computing pairwise Wasserstein distances between a huge number of large distributions (e.g. image collection) or optimization problems with a lot of Wasserstein distances (e.g. barycenters) is still intractable.

Learning Wasserstein Embeddings

Idea

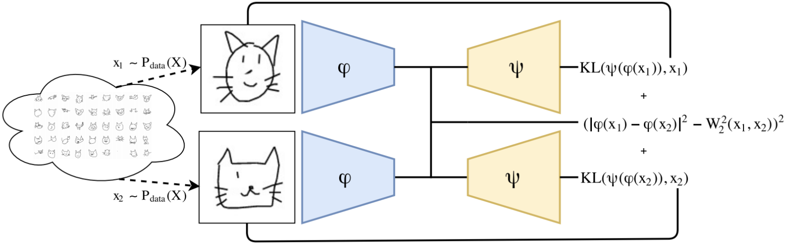
- Learn an embedding where Wasserstein distance is reproduced by Euclidean norm
- Once the embedding is found, computing distances or solving problems related to Wasserstein distances can be conducted extremely fast
- Simultaneously learn the inverse mapping to improve performance and allow interpretations of the results

Deep Wasserstein Embedding



- Pre-computed dataset consists of pair of histograms $\{x_i^1, x_i^2\}_{i \in 1,\dots,n}$ of dimensionality d and their corresponding W_2 distances $\{y_i = W_2^2(x_i^1, x_i^2)\}_{i \in 1,\dots,n}$
- Siasame architecture + Decoder

Deep Wasserstein Embedding



- Global objective function

$$\min_{\phi,\psi} \sum_i \left\| \left\| \phi(x_i^1) - \phi(x_i^2) \right\|^2 - y_i \right\|^2 + \lambda \sum_i KL(\psi(\phi(x_i^1)), x_i^1) + KL(\psi(\phi(x_i^2)), x_i^2)$$

Deep Wasserstein Embedding

Decoder eases the learning

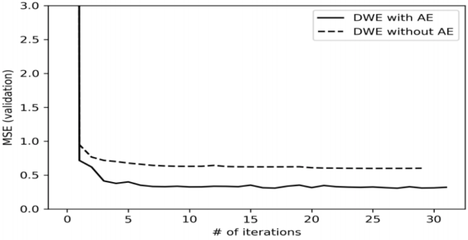


Figure: W_2^2 validation MSE along the number of epochs for the MNIST dataset (DWE).

Wasserstein Barycenters

Idea

- An analogy with barycenters in a Euclidean space

$$\bar{x} = \arg\min_{x} \sum_{i} \alpha_i W(x, x_i) \approx \psi(\sum_{i} \alpha_i \phi(x_i))$$

Principal Geodesic Analysis

Idea

- Generalization of PCA
  - Find approximated Fréchet mean $\bar{x} = \sum_i^N \phi(x_i)$ and subtract it to all samples
  - Build $V_k = span(v_1, ..., v_k)$ recusively

$$v_1 = \mathbf{argmax}_{|v|=1} \sum_{i=1}^{n} (v.\phi(x_i))^2$$

$$v_k = \mathbf{argmax}_{|v|=1} \sum_{i=1}^{n} \left((v.\phi(x_i))^2 + \sum_{j=1}^{k-1} (v_j.\phi(x_i))^2 \right)$$

Numerical Experiments

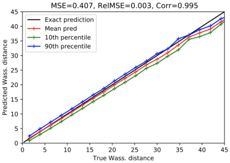
MNIST dataset

- MNIST: contains 28×28 images from 10 digit classes
- Dataset used: 1 million pairs from 60000 samples with exact Wasserstein distances

Numerical Experiments

MNIST dataset

- Computational performance



Method	W_2^2/sec
LP network flow (1 CPU)	192
DWE Indep. (1 CPU)	3 633
DWE Pairwise (1 CPU)	213 384
DWE Indep. (GPU)	233 981
DWE Pairwise (GPU)	10 477 901

Figure 2: Prediction performance on the MNIST dataset. (Figure) The test performance are as follows: MSE=0.41, Relative MSE=0.003 and Correlation=0.995. (Table) Computational performance of W_2^2 and DWE given as average number of W_2^2 computation per seconds for different configurations.

- Interpretation: better suited for mining large scale datasets and online applications

Numerical Experiments

MNIST dataset

- Wasserstein Barycenter
  - Computed with uniform weights from 1000 samples per class

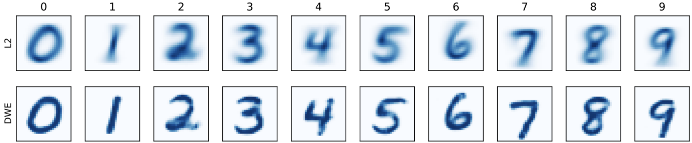


Figure 3: Barycenter estimation on each class of the MNIST dataset for squared Euclidean distance (L2) and Deep Wasserstein Embedding (DWE).

Numerical Experiments

MNIST dataset

- Principal Geodesic Analysis

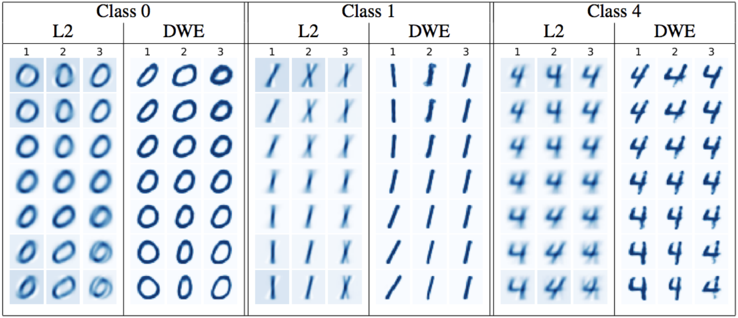


Figure 4: Principal Geodesic Analysis for classes 0,1 and 4 from the MNIST dataset for squared Euclidean distance (L2) and Deep Wasserstein Embedding (DWE). For each class and method we show the variation from the barycenter along one of the first 3 principal modes of variation.

Numerical Experiments

Google Doodle Dataset

- Google Doodle: crowd sourced dataset of 50 million drawings
- Dataset used: Three classes, Cat, Crab, and Face, rendered into 28x28 grayscale images. Draw 1 million pairs and compute exact Wasserstein distances

Numerical Experiments

Google Doodle Dataset

- Computational performance

Learn \ Test	CAT	CRAB	FACE	MNIST
CAT	**1.491**	*1.818*	1.927	12.525
CRAB	*2.679*	**0.918**	3.510	11.750
FACE	*4.884*	4.843	**1.313**	52.994
MNIST	9.776	6.689	*4.387*	**0.407**

(a) MSE

Learn \ Test	CAT	CRAB	FACE	MNIST
CAT	**0.004**	0.007	0.011	0.082
CRAB	0.009	**0.004**	0.018	0.075
FACE	0.018	0.024	**0.008**	0.329
MNIST	0.028	0.030	0.026	**0.003**

(b) Relative MSE

Table 1: Cross performance between the DWE embedding learned on each datasets. On each row, we observe the MSE (table a) and relative MSE (table b) on the test set of each dataset given a DWL (Cat, Crab, Faces and MNIST).

Numerical Experiments

Google Doodle Dataset

- Interpolation
  - LP solver: 20 sec/interp, noisy
  - Regularized Wasserstein barycenter: 4 sec/interp, smooth, loosing details
  - DWE: 4 ms/interp, smooth, looses some details

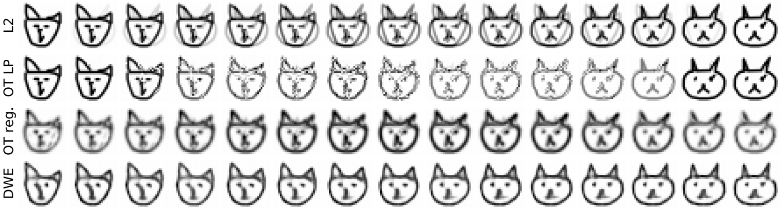


Figure 5: Comparison of the interpolation with L2 Euclidean distance (top), LP Wasserstein interpolation (top middle) regularized Wasserstein Barycenter (down middle) and DWE (down).

Numerical Experiments

Google Doodle Dataset

- Interpolation (more results)

Figure 8: Interpolation between four samples of each datasets using DWE. (left) cat dataset, (center) Crab dataset (right) Face dataset.

Numerical Experiments

Google Doodle Dataset

- Nearest neighbor walk

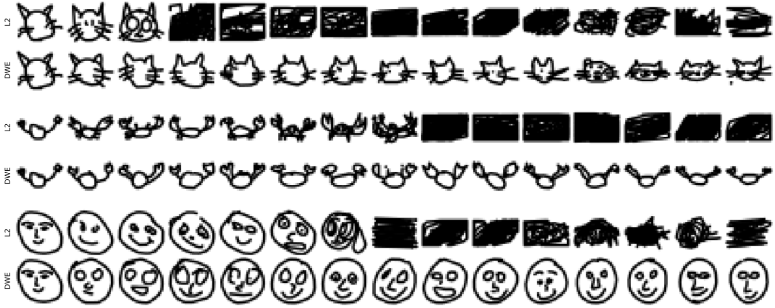


Figure 10: Nearest neighbor walk along the 3 datasets when using L2 or DWE for specifying the neighborhood. (up) Cat dataset, (middle) Crab dataset (down) Face dataset.

Entropic Regularization

Kantorovish formulation

$$U(a,b) = \{\mathbf{P} \in \mathbb{R}_+^{n\times m} : \mathbf{P}\mathbb{1}_m = \mathbf{a}, \mathbf{P}^T\mathbb{1}_n = \mathbf{b}\}$$

$\mathbf{P}_{i,j}$ denotes how much mass from ith point in $\mathbf{a}$ is transported to the jth point in $\mathbf{b}$. $U(a,b)$ is all valid transport plans. $\mathbf{P}$ is known as a coupling matrix.

Entropy

Discrete entropy of a coupling matrix $\mathbf{P}$:

$$\mathbf{H}(\mathbf{P}) := -\sum_{i,j} \mathbf{P}_{i,j}(\log(\mathbf{P}_{i,j}) - 1)$$

$\mathbf{H}(\mathbf{P}) = -\infty$ if any entry of $\mathbf{P}$ is negative or 0.

Entropic Regularization

property

H is 1-strongly concave:

$$\forall x, y, (\nabla f(x) - \nabla f(y))^T (x - y) \leq ||x - y||_2^2$$

$$\forall x, -Hf(x) - I \text{ is positive semidefinite}$$

Motivation

Larger $\mathbf{H}(\mathbf{P}) \to$ distribution more uniform.
We can use **H** to regularize optimal transport.

$$L_{\mathbf{c}}(\mathbf{a}, \mathbf{b}) = \min_{\mathbf{P} \in U(\mathbf{a}, \mathbf{b})} \langle \mathbf{P}, \mathbf{C} \rangle$$

$$L_{\mathbf{c}}^{\epsilon}(\mathbf{a}, \mathbf{b}) = \min_{\mathbf{P} \in U(\mathbf{a}, \mathbf{b})} \langle \mathbf{P}, \mathbf{C} \rangle - \epsilon \mathbf{H}(\mathbf{P})$$

Entropic Regularization

$$L_{\mathbf{c}}^{\epsilon}(\mathbf{a}, \mathbf{b}) = \min_{\mathbf{P} \in U(\mathbf{a}, \mathbf{b})} \langle \mathbf{P}, \mathbf{C} \rangle - \epsilon \mathbf{H}(\mathbf{P})$$

$L_{\mathbf{c}}^{\epsilon}(\mathbf{a}, \mathbf{b})$ is known as the **Sinkhorn divergence**.

Properties

1. There exists unique solution $\mathbf{P}_\epsilon$.
2. When $\epsilon \to 0$, $\mathbf{P}_\epsilon \to \mathbf{P}$.
3. When $\epsilon \to \infty$, $\mathbf{P}_\epsilon \to \mathbf{a}\mathbf{b}^T$ (uniform distribution).

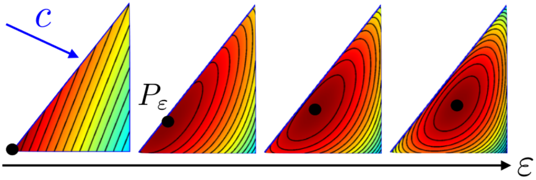


Figure 4.1: Impact of ε on the optimization of a linear function on the simplex, solving $\mathbf{P}_\varepsilon = \operatorname{argmin}_{\mathbf{P} \in \Sigma_3} \langle \mathbf{C}, \mathbf{P} \rangle - \varepsilon \mathbf{H}(\mathbf{P})$ for a varying ε.

Entropic Regularization

Proposition (4.3)

Solution to the discrete entropic optimal transport problem

$$L_{\mathbf{C}}^{\epsilon}(\mathbf{a}, \mathbf{b}) = \min_{\mathbf{P} \in U(\mathbf{a}, \mathbf{b})} \langle \mathbf{P}, \mathbf{C} \rangle - \epsilon \mathbf{H}(\mathbf{P})$$

is unique and has the form

$$\forall (i, j) \in [n] \times [m], \mathbf{P}_{i,j} = \mathbf{u}_i \mathbf{K}_{i,j} \mathbf{v}_j$$

or equivalently,

$$\mathbf{P} = \mathrm{diag}(\mathbf{u}) \mathbf{K} \mathrm{diag}(\mathbf{v})$$

where

$$\mathbf{K}_{i,j} = e^{-\mathbf{C}_{i,j}/\epsilon}, (\mathbf{u}, \mathbf{v}) \in \mathbb{R}_+^n \times \mathbb{R}_+^m$$

Entropic Regularization

Sinkhorn iterations

$$\mathbf{P} = \mathrm{diag}(\mathbf{u})\mathbf{K}\mathrm{diag}(\mathbf{v})$$

Adding constraints $\mathbf{P}\mathbb{1}_m = \mathbf{a}, \mathbf{P}^T\mathbb{1}_n = \mathbf{b}$,

$$\mathbf{u} \odot (\mathbf{K}\mathbf{v}) = \mathbf{a}, \mathbf{v} \odot (\mathbf{K}^T\mathbf{u}) = \mathbf{b}$$

This problem is known as "matrix scaling" and can be solved iteratively:

$$\mathbf{u}^{(l+1)} = \frac{\mathbf{a}}{\mathbf{K}\mathbf{v}^{(l)}}, \mathbf{v}^{(l+1)} = \frac{\mathbf{b}}{\mathbf{K}^T\mathbf{u}^{(l+1)}}$$

Note: this algorithm converges but possibly to different values for different initialization, since $(\lambda\mathbf{u}, \mathbf{v}/\lambda)$ is also a solution.

Entropic Regularization

Complexity

Let $n = m$ for simplicity, to achieve approximate transport plan $\hat{\mathbf{P}} \in U(\mathbf{a}, \mathbf{b})$ with $\langle \hat{\mathbf{P}}, \mathbf{C} \rangle \leq L_{\mathbf{C}}(\mathbf{a}, \mathbf{b}) + \tau$, the time complexity is

$$O(n^2 \log n \tau^{-3})$$

Remarks

The Sinkhorn iteration approximates optimal transport. Given enough time, it can give arbitrarily close approximations.

Any Questions?

"Size" of Wasserstein space

Question

How well can we embed other spaces into Wasserstein spaces?

Universality

A space is universal if it can embed any **finite** dimensional metric space with $O(1)$ distortion.

$$W_1(l^1) \text{ is universal. (Bourgain, 1986)}$$

l_1 is the sequence space consisting of sequences (x_n) s.t.

$$\sum_n |x_n| < \infty$$

Or intuitively, the infinite dimensional vector space with finite sum.

“Size” of Wasserstein space

Open Problem

Is $W_1(\mathbb{R}^k)$ universal for some k?

Snowflake Universality

The $\theta-$snowflake of a metric space (Y, d_Y) is (Y, d_Y^θ).

$$c_{W_p(\mathbb{R}^3)}(X, d_X^{\frac{1}{p}}) = 1$$

However,

"Size" of Wasserstein space

Open Problem

Is $W_1(\mathbb{R}^k)$ universal for some k?

Snowflake Universality

The $\theta-$snowflake of a metric space (Y, d_Y) is (Y, d_Y^θ).

$$c_{W_p(\mathbb{R}^3)}(X, d_X^{\frac{1}{p}}) = 1$$

However, only for $p \in (1, \infty)$

Open Problem

Does it hold for $p = 1$?

“Size” of Wasserstein space

Question

How well can Wasserstein space embed into other spaces?

Result

Embedding $W_2(\mathbb{R}^3)$ into L^1 will incur $\Omega(\sqrt{\log n})$ distortion.

Intuitively, it is hard to faithfully embed Wasserstein space into some very large spaces. (Open problem: is this bound tight?)

For more open problems see: *Snowflake Universality Of Wasserstein Spaces* by Andoni, Naor and Neiman

Learning Entropic Wasserstein Embeddings

Motivations

- Embedding in Euclidean space
  - Use distances and angles between vectors to encode the levels of association.
  - Bourgain's theorem

$$(X, d) \xrightarrow{O(log n)} \ell_p^{O(log^2 n)}$$

  - L_p distances ignore the geometry of the distributions.

Learning Entropic Wasserstein Embeddings

Motivations

- Wasserstein space
  - A 'large' space: Many spaces can embed into Wasserstein spaces with low distortion, while the converse may not hold
  - A 'universal' space: Can embed arbitrary metrics on finite spaces. e.g. $\mathcal{W}_1(l^1)$
- Use Sinkhorn iteration to approximate Wasserstein distance.
  - Efficient computation.

Learning Entropic Wasserstein Embeddings

Motivations

- What can we embed in theory?
  - Metric spaces $\mathcal{A}$ and $\mathcal{B}$, map $\phi : \mathcal{A} \to \mathcal{B}$ is embedding of $\mathcal{A}$ into $\mathcal{B}$
  - $Ld_{\mathcal{A}}(u, v) \leq d_{\mathcal{B}}(\phi(u), \phi(v)) \leq CLd_{\mathcal{A}}(u, v), \forall u, v \in \mathcal{A}$
  - The distortion of the embedding ϕ is the smallest C such that the equation holds.
  - Can characterize how "large" a space is (its representational capacity) by the spaces that embed into it with low distortion.

Learning Entropic Wasserstein Embeddings

The learning task

- Objects $\mathcal{C}$: words, images, nodes
- target relationships r: $\{(u^{(i)}, v^{(i)}, r(u^{(i)}, v^{(i)}))\}$
- Our goal is to find a map $\phi : \mathcal{C} \to \mathcal{W}_p(x)$ such that the relationship r(u,v) can be recovered from the Wasserstein distance between $\phi(u)$ and $\phi(v)$, for u, v $\in \mathcal{C}$

Learning Entropic Wasserstein Embeddings

Representation

- Discrete distributions on finite sets of points in $\mathbb{R}^n$

$$\mu = \sum_{i=1}^{M} u_i \delta_x^{(i)}, \nu = \sum_{i=1}^{M} v_i \delta_y^{(i)} \to \mathcal{W}_p(\mu, \nu)$$

$$\sum_{i=1}^{M} u_i = \sum_{i=1}^{M} v_i = 1, u_i, v_i \geq 0, \forall i \in \{1, ..., M\}$$

- Fix weights and only learn positions.

Learning Entropic Wasserstein Embeddings

Optimization

- replace $\mathcal{W}_p$ with the Sinkhorn divergence $\mathcal{W}_p^\lambda$
- Try to find

$$\phi_* = \underset{\phi \in \mathcal{H}}{\arg\min} \frac{1}{N} \sum_i^N \mathcal{L}(\mathcal{W}_p^\lambda(\phi(u^{(i)}), \phi(v^{(i)})), r^{(i)})$$

Empirical Study

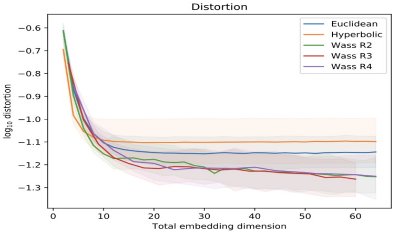
Embedding complex networks

- Input space $\mathcal{C}$: collection of vertices for each network
- To learn a map ϕ such that $\mathcal{W}_1(\phi(u), \phi(v))$ matches the shortest path distance between vertices u and v.
- Minimize mean distortion

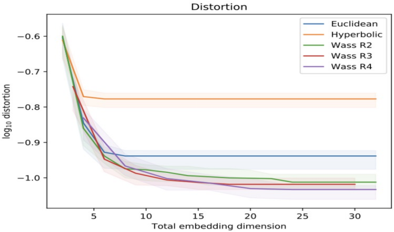
$$\phi_* = \arg\min_{\phi} \frac{1}{\binom{n}{2}} \sum_{j>i} \frac{|\mathcal{W}_1^{\lambda}(\phi(u^{(i)}), \phi(v^{(i)})) - d_c(v_i, v_j)|}{d_c(v_i, v_j)}$$

Empirical Study

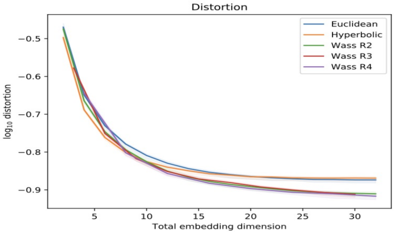
Embedding performance on random networks



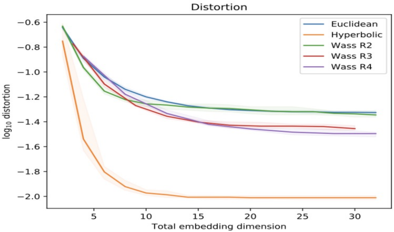
(a) Random scale-free networks.



(b) Random small-world networks.



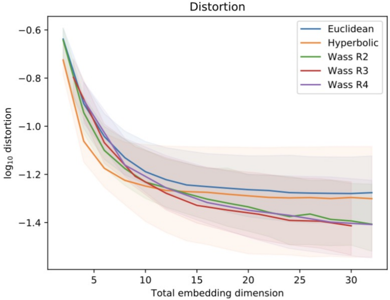
(c) Random community-structured networks.



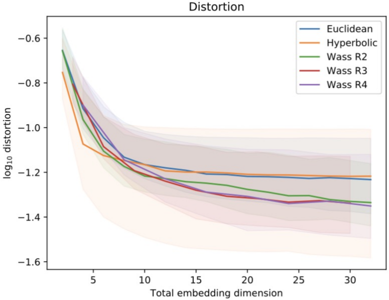
(d) Random trees.

Empirical Study

Embedding performance on real networks



(a) arXiv co-authorship.



(b) Amazon product co-purchases.

Empirical Study

Word2Cloud: Wasserstein word embeddings

- sentence $s = (x_0, x_1, ...x_n)$, word x_i
- Use a Siamese network to learn word embeddings

$$\phi_* = \arg\min_{\phi} \sum r[\mathcal{W}_1^\lambda(\phi(x_i), \phi(x_j))]^2 + (1-r)[m - W_1^\lambda(\phi(x_i), \phi(x_j))]^2$$

where r=1 for related embeddings and r=0 for unrelated ones.

Empirical Study

Word2Cloud

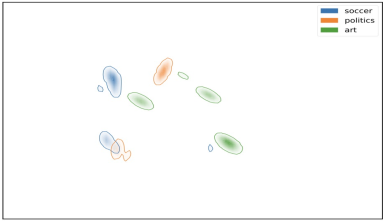
- Nearest Neighbors

$\mathcal{W}_1^\lambda(\mathbb{R}^2)$	one:	f, two, i, after, four
	united:	series, professional, team, east, central
	algebra:	skin, specified, equation, hilbert, reducing
$\mathcal{W}_1^\lambda(\mathbb{R}^3)$	one:	two, three, s, four, after
	united:	kingdom, australia, official, justice, officially
	algebra:	binary, distributions, reviews, ear, combination
$\mathcal{W}_1^\lambda(\mathbb{R}^4)$	one:	six, eight, zero, two, three
	united:	army, union, era, treaty, federal
	algebra:	tables, transform, equations, infinite, differential

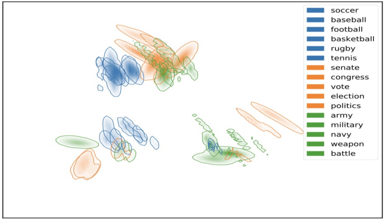
Table 1: Change in the 5-nearest neighbors when increasing dimensionality of each point cloud with fixed total length of representation.

Empirical Study

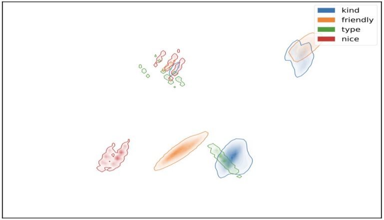
Word2Cloud: Visualization



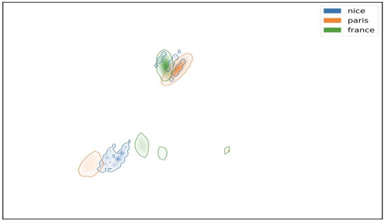
(a) Densities of three embedded words.



(b) Class separation.



(c) Word with multiple meanings: `kind`.



(d) Explaining a failed association: `nice`.