Stochastic Wasserstein Barycenters

Sebastian Claici [1] **Edward Chien** [1] **Justin Solomon** [1]

Abstract

We present a stochastic algorithm to compute the barycenter of a set of probability distributions under the Wasserstein metric from optimal transport. Unlike previous approaches, our method extends to continuous input distributions and allows the support of the barycenter to be adjusted in each iteration. We tackle the problem without regularization, allowing us to recover a much sharper output. We give examples where our algorithm recovers a more meaningful barycenter than previous work. Our method is versatile and can be extended to applications such as generating super samples from a given distribution and recovering blue noise approximations.

1. Introduction

Several scenarios in machine learning require summarizing a collection of probability distributions with shared structure but individual bias. For instance, multiple sensors might gather data from the same environment with different noise distributions; the samples they collect must be assembled into a single signal. As another example, a dataset might be split among multiple computers, each of which carries out MCMC Bayesian inference for a given model; the resulting "subset posterior" latent variable distributions must be reassembled into a single posterior for the entire dataset. In each case, the summarized whole can be better than the sum of its parts: noise in the input distributions cancels when averaging, while shared structure is reinforced.

The theory of *optimal transport* (OT) provides a promising and theoretically-justified approach to averaging distributions over a geometric domain. OT equips the space of measures with a distance metric known as the Wasserstein distance; the average, or *barycenter*, of a collection $\{\mu_j\}_{j=1}^N$ is then defined as a Fréchet mean minimizing the sum of squared Wasserstein distances to the input distributions (Agueh & Carlier, 2011). This mean is aware of the geometric structure of the underlying space. For example, the Wasserstein barycenter of two Dirac distributions δ_x and δ_y supported at points $x, y \in \mathbb{R}^n$ is a single Dirac delta at the center point $\delta_{(x+y)/2}$ rather than the bimodal superposition $\frac{1}{2}(\delta_x + \delta_y)$ obtained by averaging algebraically.

If the input distributions are discrete, then the Wasserstein barycenter is computable in polynomial time by solving a large linear program (Anderes et al., 2016). Adding entropic regularization yields elegant and efficient approximation algorithms (Genevay et al., 2016; Cuturi & Peyré, 2016; Cuturi & Doucet, 2014; Ye et al., 2017). These and other state-of-the-art methods typically suffer from any of a few drawbacks, mainly (1) poor behavior as regularization decreases, (2) required access to the distribution functions rather than sampling machinery, and/or (3) a fixed discretization on which the input or output distribution is supported, chosen without knowledge of the barycenter's structure.

Given sample access to N distributions μ_j, we propose an algorithm that iteratively refines an approximation to the true Wasserstein barycenter. The support of our barycenter is adjusted in each iteration, adapting to the geometry of the desired output. Unlike most existing OT algorithms, we tackle the problem without regularization, yielding a sharp result. Experiments show that the support of our barycenter is contained (to tolerance) within the support of the true barycenter even though we use stochastic optimization rather than computational geometry.

Contributions. We give a straightforward parallelizable stochastic algorithm to approximate and sample from the Wasserstein barycenter of a collection of distributions, which does not rely on regularization to make the problem tractable. We only employ samplers from the input distributions, and our technique is not restricted to input or output distributions supported on a fixed set of points. We verify convergence properties and showcase examples where our approach is inherently more suitable than competing approaches that require a fixed support.

[1] MIT, CSAIL, Cambridge, USA. Correspondence to: sclaici@mit.edu <
>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018. Copyright 2018 by the author(s).

2. Related Work

OT has made significant inroads in computation and machine learning; see (Lévy & Schwindt, 2017; Peyré & Cuturi, 2018; Solomon, 2018) for surveys. Although most algorithms we highlight approximate OT distances rather barycenters, they serve as potential starting points for barycenter computation.

Cuturi (2013) renewed interest in OT in machine learning through introduction of entropic regularization. The resulting Sinkhorn algorithm is compact and efficient; it has been extended to barycenter problems through gradient descent (Cuturi & Doucet, 2014) or iterative projection (Benamou et al., 2015). Improvements for structured instances enhance Sinkhorn's efficiency, e.g. via fast convolution (Solomon et al., 2015) or multiscale approximation (Schmitzer, 2016).

Our technique, however, is influenced more by *semidiscrete* methods, which compute OT distances to distributions supported on a finite set of points. Semidiscrete OT is equivalent to computing a power diagram (Aurenhammer, 1987; Aurenhammer et al., 1992), a weighted generalization of Voronoi diagrams. Algorithms by Mérigot (2011) in 2D and Lévy (2015) in 3D use computational geometry to extract gradients for the dual semidiscrete problem; Kitagawa et al. (2016a) accelerate convergence via a second-order Newton method. Similar to our technique, De Goes et al. (2012) move the support of a discrete approximation to a distribution to reduce Wasserstein distance.

Recent stochastic techniques target learning applications. Genevay et al. (2016) propose a scalable stochastic algorithm based on the dual of the entropically-regularized problem; they are among the first to consider the setting of sample-based access to distributions but rely on entropic regularization to smooth out the problem and approximate OT distances rather than barycenters. Staib et al. (2017) propose a stochastic barycenter algorithm from samples, but a finite, fixed set of support points must be provided a priori. Arjovsky et al. (2017) incorporate a coarse stochastic approximation of the 1-Wasserstein distance into a generative adversarial network (GAN); the 1-Wasserstein distance typically is not suitable for barycenter computation.

Further machine learning applications range from supervised learning to Bayesian inference. Schmitz et al. (2017) leverage OT theory for dictionary learning. Carrière et al. (2017) apply the Wasserstein distance to point cloud segmentation by developing a notion of distance on topological persistence diagrams. Courty et al. (2016) utilize the optimal transport plan for transfer learning on different domains. Srivastava et al. (2015a;b) use the Wasserstein barycenter to approximate the posterior distribution of a full dataset by the barycenter of the posteriors on smaller subsets; their method provably recovers the full posterior as the number of subsets increases.

3. Background and Preliminaries

Let (X, d) be a metric space, and let $\mathcal{P}(X)$ be the space of probability measures on X with finite second moment. Given two measures $\mu_1, \mu_2 \in \mathcal{P}(X)$, the squared 2-Wasserstein distance between μ_1 and μ_2 is given by

$$W_2^2(\mu_1, \mu_2) = \left(\inf_{\gamma \in \Gamma(\mu_1, \mu_2)} \int_{X \times X} d(\mathbf{x}, \mathbf{y})^2 \, \mathrm{d}\gamma(\mathbf{x}, \mathbf{y}) \right). \quad (1)$$

Here, $\Gamma(\mu_1, \mu_2) \subset \mathcal{P}(X \times X)$ is the set of measure couplings between μ_1 and μ_2:

$$\Gamma(\mu_1, \mu_2) = \{\gamma \in \mathcal{P}(X \times X) : (\pi_x)_{\#}\gamma = \mu_1, (\pi_y)_{\#}\gamma = \mu_2\},$$

where π_x and π_y are the two projections of $X \times X$, and the push-forward of a measure through a measurable map is defined as $f_{\#}\mu(A) = \mu(f^{-1}(A))$ for any set A in a σ-algebra of X.

For measures $\mu_1, \ldots, \mu_N$, we can define the Wasserstein barycenter as the minimizer of the functional

$$F[\nu] = \frac{1}{N} \sum_{j=1}^{N} W_2^2(\nu, \mu_j). \quad (2)$$

When the input measures are discrete distributions, (2) is a linear program solvable in polynomial time.

If at least one of the measures μ_j is absolutely continuous with respect to the Lebesgue measure, then (2) admits a unique minimizer μ^* (Agueh & Carlier, 2011; Santambrogio, 2015). However, μ^* will also be absolutely continuous, implying that computational systems typically can only find an inexact finite approximation.

We study a discretization of this problem. Suppose $\Sigma \subset X$ consists of m points $\{\mathbf{x}^i\}_{i=1}^m$, and define the functional

$$F[\Sigma] = \frac{1}{N} \sum_{j=1}^{N} W_2^2 \left(\frac{1}{m} \sum_{i=1}^{m} \delta_{\mathbf{x}^i}, \mu_j \right). \quad (3)$$

We define the main problem.

Problem 1 (Semidiscrete approximation)**.** *Find a minimizer of* $\Sigma \to F[\Sigma]$ *subject to the constraints* $\Sigma \subset X$, $|\Sigma| = m$.

Solving problem (1) for a single input measure is equivalent to finding the optimal m-point approximation to the input measure. We can use the solution as a set of supersamples from the input (Chen et al., 2010), or if the input distribution is a grayscale image, the solution yields a blue noise approximation to the image (De Goes et al., 2012).

4. Mathematical Formulation

The OT problem (1) admits an equivalent dual problem

$$\sup_{\phi \in L^1(X)} \int_X \phi(\mathbf{x}) \, \mathrm{d}\nu(\mathbf{x}) + \int_X \overline{\phi}(\mathbf{y}) \, \mathrm{d}\mu(\mathbf{y}), \tag{4}$$

where ϕ is the Kantorovich potential and $\overline{\phi}(\mathbf{x}) := \inf_{\mathbf{y} \in X} \{d(\mathbf{x}, \mathbf{y})^2 - \phi(\mathbf{y})\}$ is the c-transform of ϕ (Santambrogio, 2015; Villani, 2009).

Following Santambrogio (2015), if $\nu = \sum_{i=1}^m \frac{1}{m} \delta_{\mathbf{x}^i}$ is a finite measure supported on $\Sigma = \{\mathbf{x}^i\}_{i=1}^m$, then (4) becomes

$$\max_{\phi \in \mathbb{R}^m} \left\{ \sum_i \frac{1}{m} \phi^i + \int_X \overline{\phi}(\mathbf{y}) \, \mathrm{d}\mu(\mathbf{y}) \right\}, \tag{5}$$

where $\boldsymbol{\phi} = (\phi^1, \ldots, \phi^m)$. Note that the function $\phi \in L^1(X)$ is replaced with a finite-dimensional $\boldsymbol{\phi} \in \mathbb{R}^m$.

With this formula in mind, define

$$F_{\mathrm{OT}}[\phi, \Sigma; \mu] := \sum_i \frac{1}{m} \phi^i + \int_X \overline{\phi}(\mathbf{y}) \, \mathrm{d}\mu(\mathbf{y}). \tag{6}$$

Note that constant shifts in the ϕ^i do not change the value of F_{OT}. F_{OT} has a simple derivative with respect to the ϕ^i's:

$$\frac{\partial F_{\mathrm{OT}}}{\partial \phi^i} = \frac{1}{m} - \int_{V_\phi^i} \mathrm{d}\mu(\mathbf{y}) \tag{7}$$

where V_ϕ^i is the *power cell* of point $\mathbf{x}^i$:

$$V_\phi^i = \{x \in X : d(\mathbf{x}, \mathbf{x}^i)^2 - \phi^i \leq d(\mathbf{x}, \mathbf{x}^{i'})^2 - \phi^{i'}, \forall i'\}.$$

From here on we work with compact subsets of the Euclidean space $\mathbb{R}^D$ endowed with the Euclidean metric, $d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2$. To differentiate with respect to the $\mathbf{x}^i$'s, notice that the first term in equation (6) does not depend on the positions of the points. We rewrite the second term as

$$\sum_{i=1}^m \int_{V_\phi^i} (d(\mathbf{y}, \mathbf{x}^i)^2 - \phi^i) \, \mathrm{d}\mu(\mathbf{y}).$$

Using Reynolds' transport theorem to differentiate while accounting for boundary terms shows

$$\frac{\partial F_{\mathrm{OT}}}{\partial \mathbf{x}^i} = \mathbf{x}^i \int_{V_\phi^i} \mathrm{d}\mu(\mathbf{y}) - \int_{V_\phi^i} \mathbf{y} \, \mathrm{d}\mu(\mathbf{y}). \tag{8}$$

Equation (7) confirms the intuition that each cell contains as much mass as its associated source point. We will leverage (8) to design a fixed-point iteration that moves each point to the center of its power cell.

Each subproblem of (3) admits a different Kantorovich potential $\boldsymbol{\phi}_j = (\phi_j^1, \ldots, \phi_j^m)$, giving the following optimization functional

$$F\left[\{\boldsymbol{\phi}_j\}_{j=1}^N, \Sigma; \{\mu_j\}_{j=1}^N\right] = \frac{1}{N} \sum_{j=1}^N F_{\mathrm{OT}}[\boldsymbol{\phi}_j, \Sigma; \mu_j] \tag{9}$$

Define

$$a_j^i = \int_{V_{\phi_j^i}} \mathrm{d}\mu(\mathbf{y}) \qquad b_j^i = \frac{1}{a_j^i} \int_{V_{\phi_j^i}} \mathbf{y} \, \mathrm{d}\mu(\mathbf{y}).$$

With this notation in place, the partial derivatives are

$$\frac{\partial F}{\partial \phi_j^i} = \frac{1}{N} \left(\frac{1}{m} - a_j^i \right) \qquad \frac{\partial F}{\partial \mathbf{x}^i} = \frac{1}{N} \sum_{j=1}^N a_j^i \left(\mathbf{x}^i - b_j^i \right). \tag{10}$$

5. Optimization

With our optimization objective function in place, we now introduce our barycenter algorithm. To simplify nomenclature, from here on we refer to the dual potentials ϕ_j as weights on the generalized Voronoi diagram. Our overall strategy is an alternating optimization of F in (9):

- For fixed point positions, F is concave in the weights and is optimized using stochastic gradient ascent.
- For fixed weights, we apply a single fixed point iteration akin to Lloyd's algorithm (Lloyd, 1982).

5.1. Estimating Gradients

Each of a_j^i and b_j^i can be expressed as an expectation of a simple function with respect to the μ_j. We estimate these quantities by a simple Monte Carlo scheme.

In more detail, we can rewrite a_j^i and b_j^i as

$$a_j^i = \mathbb{E}_{y \sim \mu_j} \left[\mathbb{1}_{y \in V_{\phi_j}^i} \right] \qquad b_j^i = \mathbb{E}_{y \sim \mu_j} \left[y \cdot \mathbb{1}_{y \in V_{\phi_j}^i} \right].$$

Here, $\mathbb{1}$ indicates the indicator function of a set.

Since we have sample access to each μ_j, the expectations can be approximated by drawing K points independently $y_k \sim \mu_j$ and computing

$$\hat{a}_j^i = \frac{1}{K} \sum_{k=1}^K \mathbb{1}_{y_k \in V_{\phi_j}^i} \qquad \hat{b}_j^i = \frac{1}{K} \sum_{k=1}^K y_k \cdot \mathbb{1}_{y_k \in V_{\phi_j}^i}. \tag{11}$$

5.2. Concave Maximization

The first step in our alternating optimization maximizes F over the weights $\phi \in \mathbb{R}^m$ while the points $\mathbf{x}^i$ are fixed. We call this step of the algorithm an *ascent* step.

For a fixed set of points, the functional F is concave in the weights ϕ_j, since it is the dual of the convex semidiscrete transport problem. To solve for the weights, we perform gradient ascent using the formula in (10) where a_j^i is approximated using $\hat{a}_j^i$. Note that the gradient for a set of weights ϕ_j only requires computation of the density of a single measure μ_j, implying that the ascent steps can be decoupled across different measures.

Write $w^0 = \phi_j$ for the initial iterate. The simplest version of our algorithm updates

$$w^{k+1} = w^k + \alpha \frac{\partial F}{\partial \phi_j}[w^k].$$

The iterates converge when each point contains equal mass in its associated power cell.

F has a known Hessian as a function of the ϕ_j that can be used in Newton's algorithm (Kitagawa et al., 2016b). Computing the Hessian, however, is only possible with access to the density functions of the μ_j's as it requires computing a density of the measure on the boundary between two power cells. The boundary set is inherently lower dimensional than the problem space, and hence sample access to the μ_j is insufficient. Moreover, even had we access to the probability density functions, computing the Hessian would require the Delaunay triangulation of the point set, which is expensive in more than two dimensions.

In any event, choosing the step size α is important for convergence. Line search is difficult as we do not have access to true objective value at each iterate. Instead, we rely on Nesterov acceleration to improve performance (Nesterov, 1983). With acceleration, our iterates are

$$z^{k+1} = \beta z^k + \frac{\partial F}{\partial \phi_j}[w^k] \tag{12}$$

$$w^{k+1} = w^k + \alpha z^{k+1}. \tag{13}$$

where $w^k, z^k \in \mathbb{R}^m$. In our experiments, we use $\alpha = 10^{-3}$ and $\beta = 0.99$. Convergence of the accelerated gradient method can be shown when $\alpha = {}^1\!/\!_L$ where L is the Lipschitz constant of F; in §6, we give an estimate of this constant. Our convergence criterion for this step is $\|\nabla F\|_2^2 \le \epsilon$.

5.3. Fixed Point Iteration

The second step of our optimization is a fixed point iteration on the point positions. This step is similar to the point update in a k-means algorithm in that it snaps points to the centers of local cells, and we refer to it as a *snap* step.

Algorithm 1 Optimizing estimate of barycenter support

```
Require: Estimate of barycenter support $\Sigma = \{\mathbf{x}_i\}_{i=1}^m$
Ensure: Optimized barycenter support $\Sigma^*$ with lower cost.
for $t = 1, 2, \ldots, T$ do
    for $j = 1, 2, \ldots, J$ do
        $z^0 \leftarrow 0$                                   {Ascent on weights}
        $w^0 \leftarrow \phi_j$
        while $\left\| \frac{\partial F}{\partial \phi_j} \right\| > \epsilon$ do
            Compute $\hat{a}_j^i$ according to equation (11)
            $z^{k+1} = \beta z^k + \frac{\partial F}{\partial \phi_j}[w^k]$
            $w^{k+1} = w^k + \alpha z^{k+1}$
        end while
        $\phi_j \leftarrow w^{\mathrm{end}}$
    end for
    Compute $\hat{b}_j^i$ according to equation (11)
    for $\mathbf{x}_i \in S$ do
        $\mathbf{x}_i \leftarrow \frac{\sum_{j=1}^N \hat{a}_j^i \hat{b}_j^i}{\sum_{j=1}^N \hat{a}_j^i}$                     {Snap points}
    end for
end for
```

We set the second gradient in (10) to zero:

$$\frac{\partial F}{\partial \mathbf{x}^i} = 0 \qquad \Longrightarrow \qquad \frac{1}{N} \sum_{j=1}^{N} a_j^i (\mathbf{x}^i - b_j^i) = 0$$

which leads to the point update

$$\mathbf{x}^i = \frac{\sum_{j=1}^N a_j^i b_j^i}{\sum_{j=1}^N a_j^i}. \tag{14}$$

This suggests a fixed point iteration for the $\mathbf{x}^i$'s that can be decomposed into the following steps:

1. First find the barycenter of the power cells of each $\mathbf{x}^i$ with respect to each μ_j.
2. Then, average the points with weights given by the density of each measure in the cell.

If the concave maximization has converged appropriately, and uniform areas a_j^i have been achieved, then the update step becomes a uniform average over the barycenters b_j^i with respect to each measure.

5.4. Global and Local Strategies

The *ascent* and *snap* steps can be used to refine a configuration of points Σ. Once the iterates converge, we have an m-point approximation to the barycenter that can be used as an initialization for $m + 1$ point approximation in two ways. A new point $\mathbf{x}$ is sampled uniformly from X, and then we have a choice between (1) moving all points including the new one or (2) allowing only $\mathbf{x}$ to move.

These two approaches are codified in Algorithm 1 where the choice on the set S dictates which points move. The number of iterations of the outer loop is fixed beforehand. Typically, we see convergence in fewer than 20 steps, and empirically, we observe good performance even with $T = 1$. The two most natural choices for S are $S = \Sigma$ and $S = \{\mathbf{x}\}$. If the barycenter is absolutely continuous with respect to the underlying Lebesgue measure, these two strategies converge at the same rate asymptotically (Brancolini et al., 2009). The latter, however, can generate spurious samples that are not in the support of the barycenter. Note that optimizing the weights is regardless a global problem as moving or introducing just one point can change the volumes of the power cells of neighboring points.

Both algorithms are highly parallelizable, since (1) the gradient estimates are expectations computed using Monte Carlo integration and (2) the gradient step in the weights decouples across distributions.

6. Analysis

We justify the use of uniform finitely-supported measures, and then prove that our algorithm converges to a local minimum cost under mild assumptions.

We assume in this section that at least one of the distributions μ_j is absolutely continuous with respect to the Lebesgue measure, ensuring a unique Wasserstein barycenter.

6.1. Approximation Suitability

The simplest approach for absolutely continuous measures $\mu_j \in \mathcal{P}(X)$ is to sample p points from each of the J measures and solve for the true barycenter of the empirical distributions (Anderes et al., 2016). This approach likely approximates the barycenter as the number of samples increases, but requires solution of a linear program with $O(p^J)$ variables. As an alternative, Staib et al. (2017) propose a stochastic problem for approximating barycenters. They are able to prove a rate of convergence, but the support of their approximate barycenter is fixed to a finite set of points.

Our technique allows the support points to move during the optimization procedure, empirically allowing a better approximation of the barycenter with fewer points. The following theoretical result shows that the use of uniform measures supported on a finite set of points can approximate the barycenter arbitrarily well:

Theorem (Metric convergence, Kloeckner (2012); Brancolini et al. (2009)). *Suppose ν_m^* is a uniform measure supported on m points that minimizes $\frac{1}{N}\sum_{j=1}^N W_2(\nu_m^*, \mu_j)$, and let $\bar{\mu}$ denote the true barycenter of the measures $\{\mu_j\}_{j=1}^N$. Then $W_2(\nu_m^*, \bar{\mu}) \leq Cm^{-1/D}$ where C depends on the space X, the dimension D, and the metric $d(\cdot,\cdot)$.*

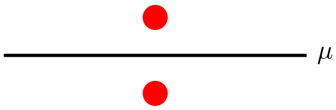

Figure 1. Non-existence of a set of weights. Let μ be the uniform measure on the line segment, and Σ be the two red points such that the line between them is orthogonal to the support of μ. There is no set of weights such that the mass of μ is split evenly between the two red points.

Note that this shows convergence in probability $\nu_m^* \rightharpoonup \bar{\mu}$ since the Wasserstein distance metrizes weak convergence (Villani, 2009). Brancolini et al. (2009) also show asymptotic equivalence of the local and global algorithms.

While we cannot guarantee that our method converges to ν_m^*, these properties indicate that the *global* minimizer of our objective provides an effective approximant to the true barycenter as the number of support points $m \to \infty$.

6.2. Algorithmic Properties

The functional F is concave in the weights ϕ_i^j with fixed point positions, and in fact usually strictly concave up to constant shifts. We can investigate the convergence properties of the gradient ascent step of the algorithm. We assume in the following section that the partial derivatives are obtained exactly, rather than approximated via sampling, so our results will hold true in the limit, as number of samples increases. We show first that the gradient of F is not necessarily Lipschitz continuous.

Counterexample. *Assume X is a compact subset of $\mathbb{R}^D$. There are measures $\mu \in \mathcal{P}(X)$ for which the gradient of F is not Lipschitz continuous. A set of weights that satisfies $\frac{\partial F}{\partial \phi} = 0$ may not exist, and if it does, it may not be unique.*

Construction. We provide a counterexample for $D = 1$. Let $X = [-1, 1]$ with the standard metric and $\mu = \delta_0$. Let $\Sigma = \{-1, 1\}$ be the fixed positions, and take $\phi_1 = \{-\epsilon, 0\}$ and $\phi_2 = \{\epsilon, 0\}$ for small ϵ. Then $\|\phi_1 - \phi_2\|_1 = 2\epsilon$, but $\|\nabla F_\phi[\phi_1] - \nabla F_\phi[\phi_2]\|_1 = 2$.

Non-existence is shown in Figure 1. To see non-uniqueness, take $\mu = \frac{1}{2}\delta_{-\epsilon} + \frac{1}{2}\delta_\epsilon$ with Σ as before. Any set of weights in $(-\epsilon, \epsilon)^2$ minimizes F_ϕ. □

For mildly behaved measures μ the gradient of F with respect to ϕ is Lipschitz continuous:

Lemma. *Assume X is a compact subset of $\mathbb{R}^D$, and μ is absolutely continuous with respect to the Lebesgue measure, with density function ρ. If the m points of Σ are distinct and*

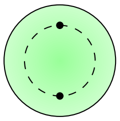

Figure 2. Non-unique minimizer on two points for the uniform measure defined on the unit disk. All antipodal points on the dashed circle at distance $2/\pi$ from the center are valid minimizers.

$\rho \le M$ almost everywhere for some constant M, then:

$$\|\nabla F_\phi[\phi_1] - \nabla F_\phi[\phi_2]\|_2 \le \sqrt{m}\frac{MS}{2L}\|\phi_1 - \phi_2\|_2.$$

where S denotes the surface area of $\partial\mathrm{conv}(X)$ and L denotes the minimum pairwise distance between points in Σ.

Proof. Consider the ith component of the gradient difference:

$$\left|\frac{\partial F_\phi}{\partial \phi^i}[\phi_1] - \frac{\partial F_\phi}{\partial \phi^i}[\phi_2]\right| = \left|\int_{V^i_{\phi_1}} \rho\,\mathrm{d}\lambda - \int_{V^i_{\phi_2}} \rho\,\mathrm{d}\lambda\right| \le \frac{S\|\phi_1 - \phi_2\|_2}{2L}M.$$

The second inequality follows as the area of a power cell is bounded by S and the faces of the cells change at a rate linear in $\|\phi_1 - \phi_2\|_2$. The rate is dependent on the distance between the points, so the constant L is required. The Lipschitz bound follows directly from considering all components of the gradient difference together. □

This lemma implies convergence for a step size that is the inverse of the Lipschitz constant. While the above requires absolute continuity of μ, we have found that our ascent steps and method often converge even when this is not satisfied (see Figures 4 and 6).

We may also show that our algorithm monotonically decreases $F[\Sigma]$ (defined in Equation (3)) after each pair of snap and then ascent steps for compact domain and absolutely continuous μ_j. For this purpose, recall that the transport cost for a map $T : X \to \Sigma$ sending measure μ_j to $\frac{1}{m}\sum_i \delta_{\mathbf{x}^i}$ is:

$$\int_X d(x, T(x))^2\, d\mu_j.$$

Fixing the power cells V^i_j after an ascent step, we may define $T_j(\Sigma)$ to be the transport cost for the map sending the power cells V^i_j to the point set Σ, and we may define $TC(\tilde\Sigma) = \frac{1}{N}\sum_j T_j$ to be the joint (average) transport cost. Letting $\tilde\Sigma = \{\mathbf{\tilde{x}^i}\}$ denote the new positions after a snap step, we may now show:

Lemma. *For $X \subset \mathbb{R}^D$ compact, and μ_j absolutely continuous with respect to the Lebesgue measure for all j:*

$$F[\tilde\Sigma] \le F[\Sigma].$$

Proof. By strong duality, we have the following equality for each j when the ϕ have been optimized after an ascent step:

$$F_{OT}[\phi, \Sigma; \mu_j] = W_2^2\left(\frac{1}{m}\sum_{i=1}^m \delta_{\mathbf{x}^i}, \mu_j\right).$$

This implies that $F[\Sigma] = TC(\Sigma)$ as W_2^2 is simply the optimal transport cost. We now argue that $TC(\tilde\Sigma) \le TC(\Sigma)$. We may split up the integrals for transport cost over the power cells corresponding to each ith point. We differentiate $\sum_{j=1}^N \int_{V^i_j} \|x - p\|^2\, d\mu_j$ with respect to p to find the point with lowest joint transport cost to the cells V^i_j. Setting this to 0 yields the following:

$$\sum_{j=1}^N a^i_j b^i_j - a^i_j p = 0$$

Note this is equivalent to the barycenter update step in Equation (14), and with convergence of the previous ascent step, we should have uniform a^i_j weights. This demonstrates that snapping to the uniform average of barycenters lowers TC, and we have that $F[\Sigma] = TC(\Sigma) \ge TC(\tilde\Sigma) \ge F[\tilde\Sigma]$. The last inequality follows as the next ascent step will find the optimal transport and decrease the transport cost. □

With joint transportation cost being non-negative, this implies that our objective function converges to a local minimum. This does not imply that our iterates converge, as there may not be a unique minimizing point configuration (see Figure 2). Empirically, our iterates converge in all of our test cases. We note also that our formula bears some resemblance to the mean-shift algorithm and to Lloyd's algorithm, both of which which are also known to converge under some assumptions (Li et al., 2007; Bottou & Bengio, 1995).

7. Experiments

We showcase the versatility of our method on several applications. We typically use between 16K and 256K samples per input distribution to approximate the power cell density and barycenter. The variance is due to different problem sizes and dimensionality of the input measures. We stop the gradient ascent step when $\|\nabla F\|_2^2 \le 10^{-6}$. The snap step empirically converges in under 20 iterations, and several of our examples use only one step.

7.1. Distributions with Sharp Features

Our algorithm is well-suited to problems where the input distributions have very sharp features. We test against the algorithms in (Staib et al., 2017) and (Solomon et al., 2015) on two test cases: ten uniform distributions over lines in

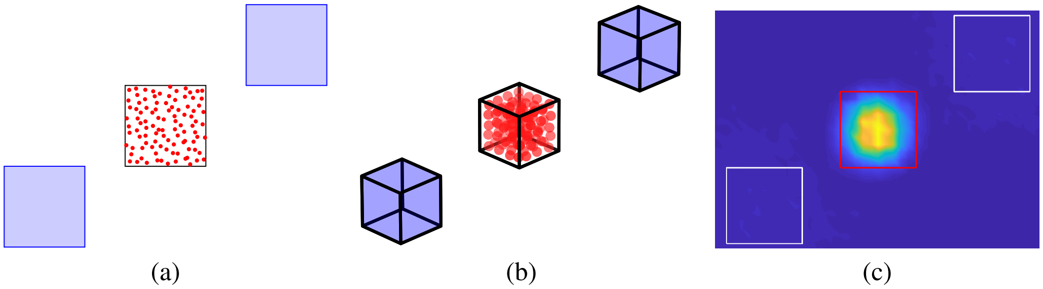

Figure 3. Barycenter when $N = 2$ tested on two uniform distributions over unit squares. (a) Our output: the input distributions are shown in blue, while the output barycenter points are shown in red, with the limits of the true barycenter in black. (b) A similar example in three dimensions. (c) The output barycenter of (Staib et al., 2017): note the output has non-zero measure outside the true barycenter.

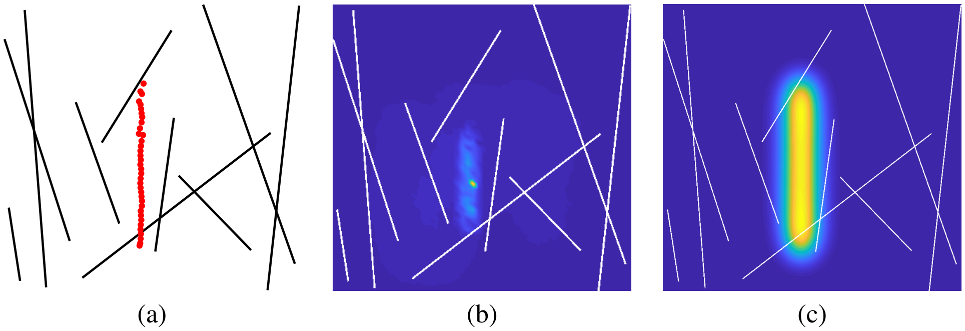

Figure 4. Barycenter of sharp featured distributions. (a) 50 points from our algorithm yields a barycenter supported on a line. (b) The barycenter from (Staib et al., 2017) using a grid of 20000 points. (c) Barycenter from (Solomon et al., 2015) using a regularizer value of $\gamma = 0.1$; smaller regularizers were numerically unstable.

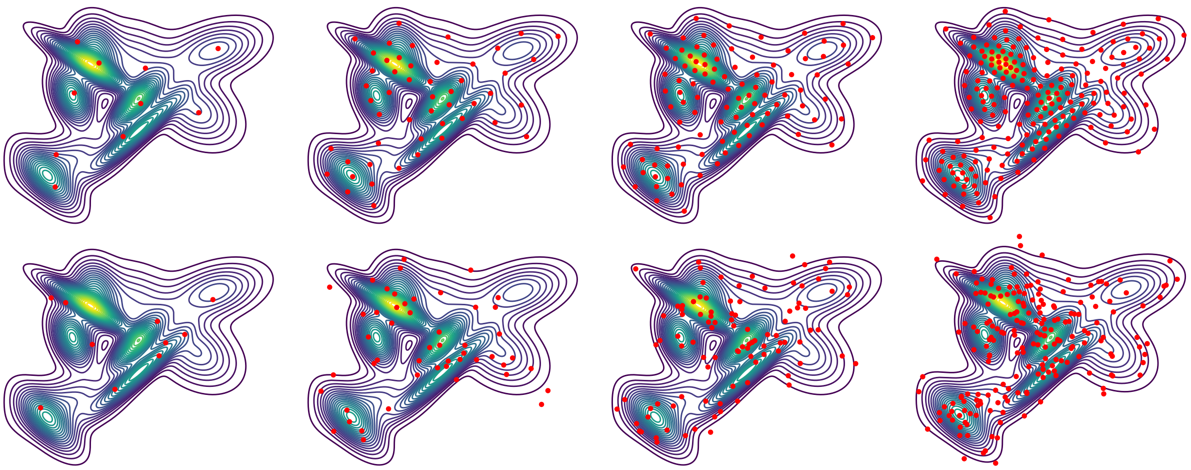

Figure 5. The n point approximation of a mixture of ten Gaussians. Top row: our method with 10, 50, 100, and 200 points. Bottom row: iid sampling with the same number of points.

the 2D plane (Figure 4), and 20 uniform distributions over ellipses (Figure 6).

The results of Figures 4 and 6 show that our barycenter is more sharply supported than the results of competing methods. Our output agrees with that of Solomon et al. (2015), but our results more closely match expected behavior. We strongly suspect that the true barycenter in Figure 4 is also a uniform measure on a line, while that in Figure 6 is a circle centered at the origin.

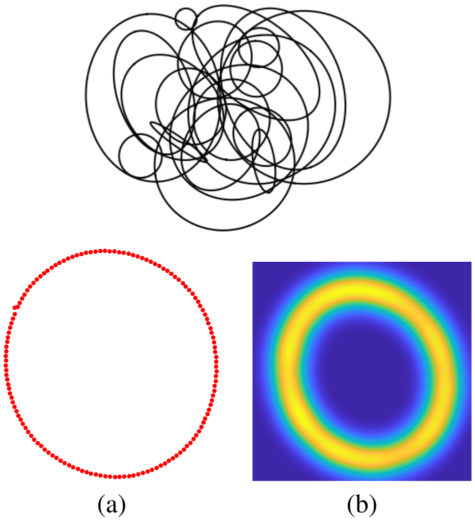

Figure 6. Barycenter of randomly generated ellipses. Top: plot showing 20 ellipses with randomly drawn center, semi-major and semi-minor axes, and skew. Bottom: (a) The output of our algorithm is a sharp distribution approximating a circle. (b) The output of (Solomon et al., 2015) with a regularizer value of $\gamma = 0.1$.

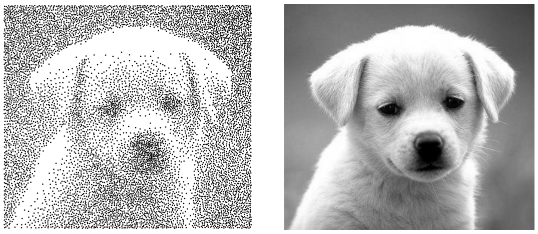

Figure 7. Blue noise sampling. Left: 10K samples from our algorithm. Right: Original image (approximately 90K pixels).

7.2. The Case $N = 2$

In the case of two input measures μ_1 and μ_2, we expect the barycenter to be McCann's interpolant (Agueh & Carlier, 2011; McCann, 1997):

$$\mu_{1/2} := \left(\frac{1}{2}\mathrm{id} + \frac{1}{2}T\right)_{\#} \mu_0 = \left(\frac{1}{2}\mathrm{id} + \frac{1}{2}T^*\right)_{\#} \mu_1$$

where T is the optimal map, and T^* is the inverse map, while $\#$ denotes the pushforward of a measure.

We test this on two uniform distributions on the unit square in Figure 3. The transport map in this case is transport of the entire distribution along a straight line. As expected from McCann's interpolant, we recover a uniform distribution on the unit square halfway between the two input distributions. We show our results alongside those of (Staib et al., 2017). Notice that their output barycenter is not uniform, and that it has non-zero measure outside the true barycenter.

7.3. The Case $N = 1$

The case $N = 1$ bears interest as well. There are instances when sampling iid from a distribution yields samples that do not approximate the underlying distribution accurately. We showcase two applications in generating super samples from distributions, as well as approximating grayscale images through blue noise.

7.3.1. Blue Noise

The term blue noise refers to an unstructured but even and isotropic distribution of points. It has been used in image dithering as it captures image intensity via local point density, without the need for varying point sizes as in halftoning.

De Goes et al. (2012) describe the link between optimal transport and blue noise generation. We recover a stochastic version of their algorithm by taking μ a discrete distribution over the image pixels proportional to intensity. As our method is more general, we observe performance loss, but the output is of comparable quality (Figure 7).

7.3.2. Super Samples

Our method can be adapted to generate super samples from complex distributions (Chen et al., 2010). Figure 5 details our results on a mixture of ten Gaussians. Our method better approximates the shape of the underlying distribution due to negative autocorrelations: points move away from oversampled regions. The points drawn iid from the mixture tend to oversample around the larger modes and do not approximate density contours as well.

8. Conclusion

We have proposed an algorithm for computing the Wasserstein barycenter of continuous measures using only samples from the input distributions. The algorithm decomposes into a concave maximization and a fixed point iteration similar to the mean-shift and k-means algorithms. Our algorithm is easy to implement and parallelize, and it does not rely on a fixed-support grid. This allows us to recover much sharper approximations to the barycenter than previous methods. Our algorithm is general and versatile enough to be applied to other problems beyond barycenter computation.

There are several avenues for future work. Solving the concave maximization problem is currently a bottleneck for our algorithm as we do not have access to the function value or the Hessian, but we believe multiscale methods can be adapted to our approach. The potential applications of this method extend beyond what was covered. One application we highlight is in developing coresets that minimize the distance to the empirical distribution on the input data.

Acknowledgements

The authors thank Fernando de Goes, Marco Cuturi, Gabriel Peyré, and Matthew Staib for input and early discussions. The authors acknowledge the generous support of Army Research Office grant W911NF-12-R0011 ("Smooth Modeling of Flows on Graphs"), from the MIT Research Support Committee, from the MIT–IBM Watson AI Lab, from the Skoltech–MIT Next Generation Program, and from an Amazon Research Award.

References

Agueh, M. and Carlier, G. Barycenters in the Wasserstein Space. *SIAM J. Math. Anal.*, 43(2):904–924, January 2011. ISSN 0036-1410. doi: 10.1137/100805741.

Anderes, E., Borgwardt, S., and Miller, J. Discrete Wasserstein barycenters: Optimal transport for discrete data. *Math Meth Oper Res*, 84(2):389–409, October 2016. ISSN 1432-2994, 1432-5217. doi: 10.1007/s00186-016-0549-x.

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein GAN. *arXiv:1701.07875*, 2017.

Aurenhammer, F. Power diagrams: properties, algorithms and applications. *SIAM Journal on Computing*, 16(1): 78–96, 1987.

Aurenhammer, F., Hoffmann, F., and Aronov, B. Minkowski-type theorems and least-squares partitioning. In *Proceedings of the Eighth Annual Symposium on Computational Geometry*, pp. 350–357. ACM, 1992.

Benamou, J., Carlier, G., Cuturi, M., Nenna, L., and Peyré, G. Iterative Bregman Projections for Regularized Transportation Problems. *SIAM J. Sci. Comput.*, 37 (2):A1111–A1138, January 2015. ISSN 1064-8275. doi: 10.1137/141000439.

Bottou, L. and Bengio, Y. Convergence properties of the k-means algorithms. In *Advances in Neural Information Processing Systems*, pp. 585–592, 1995.

Brancolini, A., Buttazzo, G., Santambrogio, F., and Stepanov, E. Long-term planning versus short-term planning in the asymptotical location problem. *ESAIM: Control, Optimisation and Calculus of Variations*, 15(3):509–524, 2009.

Carrière, M., Cuturi, M., and Oudot, S. Sliced wasserstein kernel for persistence diagrams. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pp. 664–673, 2017. URL `http://proceedings.mlr.press/v70/carriere17a.html`.

Chen, Y., Welling, M., and Smola, A. J. Super-samples from kernel herding. In *UAI 2010, Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence, Catalina Island, CA, USA, July 8-11, 2010*, pp. 109–116, 2010. URL `https://dslpitt.org/uai/displayArticleDetails.jsp?mmnu=1&smnu=2&article_id=2148&proceeding_id=26`.

Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. Optimal Transport for Domain Adaptation. *IEEE Trans. Pattern Anal. Mach. Intell.*, PP(99):1–1, 2016. ISSN 0162-8828. doi: 10.1109/TPAMI.2016.2615921.

Cuturi, M. Sinkhorn Distances: Lightspeed Computation of Optimal Transport. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 26*, pp. 2292–2300. Curran Associates, Inc., 2013.

Cuturi, M. and Doucet, A. Fast computation of Wasserstein barycenters. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pp. 685–693, 2014. URL `http://jmlr.org/proceedings/papers/v32/cuturi14.html`.

Cuturi, M. and Peyré, G. A Smoothed Dual Approach for Variational Wasserstein Problems. *SIAM J. Imaging Sci.*, 9(1):320–343, January 2016. doi: 10.1137/15M1032600.

De Goes, F., Breeden, K., Ostromoukhov, V., and Desbrun, M. Blue noise through optimal transport. *ACM Transactions on Graphics (TOG)*, 31(6):171, 2012.

Genevay, A., Cuturi, M., Peyré, G., and Bach, F. Stochastic Optimization for Large-scale Optimal Transport. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29*, pp. 3440–3448. Curran Associates, Inc., 2016.

Kitagawa, J., Mérigot, Q., and Thibert, B. Convergence of a Newton algorithm for semi-discrete optimal transport. *arXiv:1603.05579*, 2016a.

Kitagawa, J., Mérigot, Q., and Thibert, B. Convergence of a Newton algorithm for semi-discrete optimal transport. *arXiv:1603.05579 [cs, math]*, March 2016b.

Kloeckner, B. Approximation by finitely supported measures. *ESAIM Control Optim. Calc. Var.*, 18(2):343–359, 2012. ISSN 1292-8119.

Lévy, B. A numerical algorithm for L_2 semi-discrete optimal transport in 3d. *ESAIM: Mathematical Modelling and Numerical Analysis*, 49(6):1693–1715, 2015.

Lévy, B. and Schwindt, E. Notions of optimal transport theory and how to implement them on a computer. *arXiv:1710.02634*, 2017.

Li, X., Hu, Z., and Wu, F. A note on the convergence of the mean shift. *Pattern Recognition*, 40(6):1756–1762, 2007.

Lloyd, S. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

McCann, R. J. A convexity principle for interacting gases. *Advances in mathematics*, 128(1):153–179, 1997.

Mérigot, Q. A multiscale approach to optimal transport. In *Computer Graphics Forum*, volume 30, pp. 1583–1592. Wiley Online Library, 2011.

Nesterov, Y. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. In *Soviet Mathematics Doklady*, volume 27, pp. 372–376, 1983.

Peyré, G. and Cuturi, M. *Computational Optimal Transport*. Submitted, 2018.

Santambrogio, F. *Optimal Transport for Applied Mathematicians*, volume 87 of *Progress in Nonlinear Differential Equations and Their Applications*. Springer International Publishing, Cham, 2015. ISBN 978-3-319-20827-5 978-3-319-20828-2. doi: 10.1007/978-3-319-20828-2.

Schmitz, M. A., Heitz, M., Bonneel, N., Mboula, F. M. N., Coeurjolly, D., Cuturi, M., Peyré, G., and Starck, J. Wasserstein dictionary learning: Optimal transport-based unsupervised non-linear dictionary learning. *CoRR*, abs/1708.01955, 2017. URL http://arxiv.org/abs/1708.01955.

Schmitzer, B. A sparse multiscale algorithm for dense optimal transport. *Journal of Mathematical Imaging and Vision*, 56(2):238–259, 2016.

Solomon, J. *Optimal Transport on Discrete Domains*. AMS Short Course on Discrete Differential Geometry, 2018.

Solomon, J., de Goes, F., Peyré, G., Cuturi, M., Butscher, A., Nguyen, A., Du, T., and Guibas, L. Convolutional Wasserstein Distances: Efficient Optimal Transportation on Geometric Domains. *ACM Trans Graph*, 34(4):66:1–66:11, July 2015. ISSN 0730-0301. doi: 10.1145/2766963.

Srivastava, S., Cevher, V., Dinh, Q., and Dunson, D. WASP: Scalable Bayes via barycenters of subset posteriors. In Lebanon, G. and Vishwanathan, S. V. N. (eds.), *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, pp. 912–920, San Diego, California, USA, 09–12 May 2015a. PMLR. URL http://proceedings.mlr.press/v38/srivastava15.html.

Srivastava, S., Cevher, V., Tran-Dinh, Q., and Dunson, D. B. WASP: scalable bayes via barycenters of subset posteriors. 2015b. URL http://jmlr.org/proceedings/papers/v38/srivastava15.html.

Staib, M., Claici, S., Solomon, J. M., and Jegelka, S. Parallel streaming Wasserstein barycenters. In *Advances in Neural Information Processing Systems*, pp. 2644–2655, 2017.

Villani, C. *Optimal Transport: Old and New*. Number 338 in Grundlehren der mathematischen Wissenschaften. Springer, Berlin, 2009. ISBN 978-3-540-71049-3. OCLC: ocn244421231.

Ye, J., Wu, P., Wang, J. Z., and Li, J. Fast Discrete Distribution Clustering Using Wasserstein Barycenter With Sparse Support. *IEEE Trans. Signal Process.*, 65(9):2317–2332, May 2017. ISSN 1053-587X. doi: 10.1109/TSP.2017.2659647.