Variational Wasserstein Clustering

Liang Mi[1], Wen Zhang[1], Xianfeng Gu[2], and Yalin Wang[1]

[1]Arizona State University, Tempe, USA [2]Stony Brook University, Stony Brook, USA
{liangmi,wzhan139,ylwang}@asu.edu gu@cs.stonybrook.edu

Abstract. We propose a new clustering method based on optimal transportation. We discuss the connection between optimal transportation and k-means clustering, solve optimal transportation with the variational principle, and investigate the use of power diagrams as transportation plans for aggregating arbitrary domains into a fixed number of clusters. We drive cluster centroids through the target domain while maintaining the minimum clustering energy by adjusting the power diagram. Thus, we simultaneously pursue clustering and the Wasserstein distance between the centroids and the target domain, resulting in a measure-preserving mapping. We demonstrate the use of our method in domain adaptation, remeshing, and learning representations on synthetic and real data.

Keywords: clustering, discrete distribution, k-means, measure preserving, optimal transportation, Wasserstein distance

1 Introduction

Aggregating distributional data into clusters has ubiquitous applications in computer vision and machine learning. A continuous example is unsupervised image categorization and retrieval where similar images reside close to each other in the image space or the descriptor space and they are clustered together and form a specific category. A discrete example is document or speech analysis where words and sentences that have similar meanings are often grouped together. k-means [1,2] is one of the most famous clustering algorithms, which aims to partition empirical observations into k clusters in which each observation has the closest distance to the *mean* of its own cluster. It was originally developed for solving quantization problems in signal processing and in early 2000s researchers have discovered its connection to another classic problem optimal transportation which seeks a transportation plan that minimizes the transportation cost between probability measures [3].

The optimal transportation (OT) problem has received great attention since its very birth. Numerous applications such as color transfer and shape retrieval have benefited from solving OT between probability distributions. Furthermore, by regarding the minimum transportation cost – *the Wasserstein distance* – as a metric, researchers have been able to compute the barycenter [4] of multiple distributions, e.g. [5,6], for various applications. Most researchers regard OT as finding the optimal coupling of the two probabilities and thus each sample can be

mapped to multiple places. It is often called Kantorovich's OT. Along with this direction, several works have shown their high performances in clustering distributional data via optimal transportation, .e.g. [7,6,8]. On the other hand, some researchers regard OT as a measure-preserving mapping between distributions and thus a sample cannot be split. It is called Monge-Brenier's OT.

In this paper, we propose a clustering method from Monge-Brenier's approach. Our method is based on Gu *et al.* [9] who provided a variational solution to Monge-Brenier OT problem. We call it *variational optimal transportation* and name our method *variational Wasserstein clustering*. We leverage the connection between the *Wasserstein distance* and the clustering error function, and simultaneously pursue the Wasserstein distance and the k-means clustering by using a power Voronoi diagram. Given the empirical observations of a target probability distribution, we start from a sparse discrete measure as the initial condition of the centroids and alternatively update the partition and update the centroids while maintaining an optimal transportation plan. From a computational point of view, our method is solving a special case of the *Wasserstein barycenter* problem [4,5] when the target is a univariate measure. Such a problem is also called the *Wasserstein means* problem [8]. We demonstrate the applications of our method to three different tasks – domain adaptation, remeshing, and representation learning. In domain adaptation on synthetic data, we achieve competitive results with D2 [7] and JDOT [10], two methods from Kantorovich's OT. The advantages of our approach over those based on Kantorovich's formulation are that (1) it is a local diffeomorphism; (2) it does not require pre-calculated pairwise distances; and (3) it avoids searching in the product space and thus dramatically reduces the number of parameters.

The rest of the paper is organized as follows. In Section 2 and 3, we provide the related work and preliminaries on optimal transportation and k-means clustering. In Section 4, we present the variational principle for solving optimal transportation. In Section 5, we introduce our formulation of the k-means clustering problem under variational Wasserstein distances. In Section 6, we show the experiments and results from our method on different tasks. Finally, we conclude our work in Section 7 with future directions.

2 Related Work

2.1 Optimal Transportation

The optimal transportation (OT) problem was originally raised by Monge [11] in the 18th century, which sought a transportation plan for matching distributional data with the minimum cost. In 1941, Kantorovich [12] introduced a relaxed version and proved its existence and uniqueness. Kantorovich also provided an optimization approach based on linear programming, which has become the dominant direction. Traditional ways of solving the Kantorovich's OT problem rely on pre-defined pairwise transportation costs between measure points, e.g. [13], while recently researchers have developed fast approximations that incorporate computing the costs within their frameworks, e.g. [6].

Meanwhile, another line of research followed Monge's OT and had a breakthrough in 1987 when Brenier [14] discovered the intrinsic connection between optimal transportation and convex geometry. Following Brenier's theory, Mérigot [15], Gu *et al.* [9], and Lévy [16] developed their solutions to Monge's OT problem. Mérigot and Lévy's OT formulations are non-convex and they leverage damped Newton and quasi-Newton respectively to solve them. Gu *et al.* proposed a convex formulation of OT particularly for convex domains where pure Newton's method works and then provided a variational method to solve it.

2.2 Wasserstein Metrics

The Wasserstein distance is the minimum cost induced by the optimal transportation plan. It satisfies all metric axioms and thus is often borrowed for measuring the similarity between probability distributions. The transportation cost generally comes from the product of the geodesic distance between two sample points and their measures. We refer to p–Wasserstein distances to specify the exponent p when calculating the geodesic [17]. The 1–Wasserstein distance or earth mover's distance (EMD) has received great attention in image and shape comparison [18,19]. Along with the rising of deep learning in numerous areas, 1–Wasserstein distances have been adopted in many ways for designing loss functions for its superiority over other measures [20,21,22,23]. The 2–Wasserstein distance, although requiring more computation, are also popular in image and geometry processing thanks to its geometric properties such as barycenters [4,6]. In this paper, we focus on 2–Wasserstein distances.

2.3 K-means Clustering

The K-means clustering method goes back to Lloyd [1] and Forgy [2]. Its connections to the $1, 2$-Wasserstein metrics were leveraged in [8] and [24], respectively. The essential idea is to use a sparse discrete point set to cluster denser or continuous distributional data with respect to the Wasserstein distance between the original data and the sparse representation, which is equivalent to finding a Wasserstein barycenter of a single distribution [5]. A few other works have also contributed to this problem by proposing fast optimization methods, e.g. [7].

In this paper, we approach the k-means problem from the perspective of optimal transportation in the variational principle. Because we leverage power Voronoi diagrams to compute optimal transportation, we simultaneously pursue the Wasserstein distance and k-means clustering. We compare our method with others through empirical experiments and demonstrate its applications in different fields of computer vision and machine learning research.

3 Preliminaries

We first introduce the optimal transportation (OT) problem and then show its connection to k-means clustering. We use X and Y to represent two Borel probability measures and M their compact embedding space.

3.1 Optimal Transportation

Suppose $\mathcal{P}(M)$ is the space of all Borel probability measures on M. Without losing generality, suppose $X(x,\mu)$ and $Y(y,\nu)$ are two such measures, i.e. X, $Y \in \mathcal{P}(M)$. Then, we have $1 = \int_M \mu(x)dx = \int_M \nu(y)dy$, with the supports $\Omega_X = \{x\} = \{m \in M \mid \mu(m) > 0\}$ and $\Omega_Y = \{y\} = \{m \in M \mid \nu(m) > 0\}$. We call a mapping $T : X(x,\mu) \to Y(y,\nu)$ a measure-preserving one if the measure of any subset B of Y is equal to the measure of the origin of B in X, which means $\mu(T^{-1}(B)) = \nu(B),\ \forall B \subset Y$.

We can regard T as the *coupling* $\pi(x,y)$ of the two measures, each being a corresponding *marginal* $\mu = \pi(\cdot,y)$, $\nu = \pi(x,\cdot)$. Then, all the couplings are the probability measures in the product space, $\pi \in \prod(M \times M)$. Given a transportation cost $c : M \times M \to \mathbb{R}^+$ — usually the geodesic distance to the power of p, $c(x,y) = d(x,y)^p$ — the problem of optimal transportation is to find the mapping $\pi_{opt} : x \to y$ that minimizes the total cost,

$$W_p(\mu,\nu) \stackrel{\text{def}}{=} \left(\inf_{\pi \in \prod(\mu,\nu)} \int_{M \times M} c(x,y) d\pi(x,y) \right)^{1/p}, \tag{1}$$

where p indicates the power. We call the minimum total cost the $p-$*Wasserstein distance*. Since we address Monge's OT in which mass cannot be split, we have the restriction that $d\pi(x,y) = d\pi_T(x,y) \equiv d\mu(x)\delta[y = T(x)]$, inferring that

$$\pi_{T\,opt} = T_{opt} = \arg\min_T \int_M c(x, T(x)) d\mu(x). \tag{2}$$

In this paper, we follow Eq. (2). The details of the optimal transportation problem and the properties of the Wasserstein distance can be found in [25,23]. For simplicity, we use π to denote the optimal transportation map.

3.2 K-means Clustering

Given the empirical observations $\{(x_i,\mu_i)\}$ of a probability distribution $X(x,\mu)$, the k-means clustering problem seeks to assign a cluster centroid (or prototype) $y_j = y(x_i)$ with label $j = 1,...,k$ to each empirical sample x_i in such a way that the error function (3) reaches its minimum and meanwhile the measure of each cluster is preserved, i.e. $\nu_j = \sum_{y_j = y(x_i)} \mu_i$. It is equivalent to finding a partition $V = \{(V_j, y_j)\}$ of the embedding space M. If M is convex, then so is V_j.

$$\arg\min_y \sum_{x_i} \mu_i d(x_i, y(x_i))^p \ \equiv\ \arg\min_V \sum_{j=1}^{K} \sum_{x_i \in V_j} \mu_i d(x_i, y(V_j))^p. \tag{3}$$

Such a clustering problem (3), when ν is fixed, is equivalent to Monge's OT problem (2) when the support of y is sparse and not fixed because π and V induce each other, i.e. $\pi \Leftrightarrow V$. Therefore, the solution to Eq. (3) comes from the optimization in the search space $\mathcal{P}(\pi, y)$. Note that when ν is not fixed such a problem becomes *the Wasserstein barycenter* problem as finding a minimum in $\mathcal{P}(\pi, y, \nu)$, studied in [4,5,7].

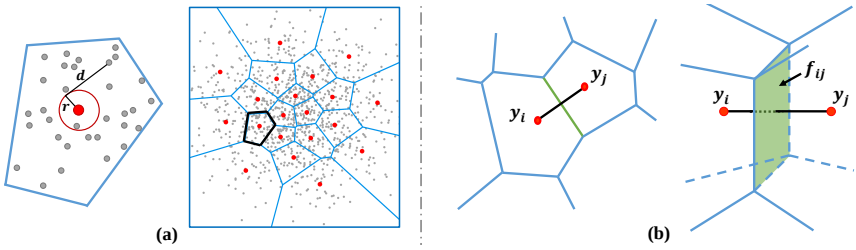

Fig. 1. (a) Power Voronoi diagram. Red dots are centroids of the Voronoi cells, or clusters. The Power distances has an offset depending on the weight of the cell. (b) Intersection of adjacent cells in 2D and 3D for computing Hessian.

4 Variational Optimal Transportation

We present the variational principle for solving the optimal transportation problem. Given a metric space M, a Borel probability measure $X(x, \mu)$, and its compact support $\Omega = supp\ \mu = \{x \in M \mid \mu(x) > 0\}$, we consider a sparsely supported point set with Dirac measure $Y(y, \nu) = \{(y_j, \nu_j > 0)\}$, $j = 1, ..., k$. (Strictly speaking, the empirical measure $X(x, \mu)$ is also a set of Dirac measures but in this paper we refer to X as the empirical measure and Y as the Dirac measure for clarity.) Our goal is to find an optimal transportation plan or map (OT-map), $\pi : x \to y$, with the *push-forward* measure $\pi_{\#}\mu = \nu$. This is *semi-discrete OT*.

We introduce a vector $\boldsymbol{h} = (h_1, ..., h_k)^T$, a hyperplane on M, $\gamma_j(\boldsymbol{h}) : \langle m, y_j \rangle + h_j = 0$, and a piecewise linear function:

$$\theta_{\boldsymbol{h}}(x) = \max\{\langle x, y_j \rangle + h_j\},\ j = 1, ..., k.$$

Theorem 1. *(Alexandrov [26]) Suppose Ω is a compact convex polytope with non-empty interior in $\mathbb{R}^n$ and $y_1, ..., y_k \subset \mathbb{R}^n$ are k distinct points and $\nu_1, ..., \nu_k > 0$ so that $\sum_{j=1}^{k} \nu_j = vol(\Omega)$. There exists a unique vector $\boldsymbol{h} = (h_1, ..., h_k)^T \in \mathbb{R}^k$ up to a translation factor $(c, ..., c)^T$ such that the piecewise linear convex function $\theta_{\boldsymbol{h}}(x) = \max\{\langle x, y_j \rangle + h_j\}$ satisfies $vol(x \in \Omega \mid \nabla\theta_{\boldsymbol{h}}(x) = y_j) = \nu_j$.*

Furthermore, Brenier [14] proved that the gradient map $\nabla\theta$ provides the solution to Monge's OT problem, that is, $\nabla\theta_{\boldsymbol{h}}$ minimizes the transportation cost $\int_\Omega \|x - \theta_{\boldsymbol{h}}(x)\|^2$. Therefore, given X and Y, $\boldsymbol{h}$ by itself induces OT.

From [27], we know that a convex subdivision associated to a piecewise-linear convex function $u_h(x)$ on $\mathbb{R}^n$ equals a *power Voronoi diagram*, or *power diagram*. A typical power diagram on $M \subset \mathbb{R}^n$ can be represented as:

$$V_j \stackrel{\text{def}}{=} \{m \in M \mid \|m - y_j\|^2 - r_j^2 \leqslant \|m - y_i\|^2 - r_i^2\},\ \forall j \neq i.$$

Then, a simple calculation gives us

$$m \cdot y_j - \frac{1}{2}(y_j \cdot y_j + r_j^2) \leqslant m \cdot y_i - \frac{1}{2}(y_i \cdot y_i + r_i^2),$$

Algorithm 1: Variational optimal transportation

Function `Variational-OT`($X(x,\mu)$, $Y(y,v)$, ϵ)
  $\boldsymbol{h} \leftarrow \mathbf{0}$.
  repeat
    Update power diagram V with $(y, \boldsymbol{h})$.
    Compute cell weight $w(\boldsymbol{h}) = \{\sum_{m \in V_j} \mu(m)\}$.
    Compute gradient $\nabla E(\boldsymbol{h})$ and Hessian H using Equation (5) and (6).
    $\boldsymbol{h} \leftarrow \boldsymbol{h} - \lambda H^{-1} \nabla E(\boldsymbol{h})$. // Update the minimizer $\boldsymbol{h}$ according to (7)
  until $|\nabla E(\boldsymbol{h})| < \epsilon$.
  return $V, \boldsymbol{h}$.
end

where $m \cdot y_j = \langle m, y_j \rangle$ and w_j represents the offset of the *power distance* as shown in Fig. 1 (a). On the other hand, the graph of the hyperplane $\pi_j(\boldsymbol{h})$ is

$$U_i \overset{\text{def}}{=} \{m \in M \mid \langle m, y_j \rangle - h_j \geqslant \langle m, y_i \rangle - h_i\}, \ \forall j \neq i.$$

Thus, we obtain the numerical representation $h_j = -\frac{|y_j|^2 - r_j^2}{2}$.

We substitute $M(m)$ with the measure $X(x)$. In our formulation, Brenier's gradient map $\nabla \theta_{\boldsymbol{h}} : V_j(\boldsymbol{h}) \to y_j$ "transports" each $V_j(\boldsymbol{h})$ to a specific point y_j. The total mass of $V_j(\boldsymbol{h})$ is denoted as: $w_j(\boldsymbol{h}) = \sum_{x \in V_j(\boldsymbol{h})} \mu(x)$.

Now, we introduce an energy function

$$\begin{aligned} E(\boldsymbol{h}) &\overset{\text{def}}{=} \int_{\Omega} \theta_{\boldsymbol{h}}(x)\mu(x)dx - \sum_{j=1}^{k} \nu_i h_j \\ &\equiv \int^{\boldsymbol{h}} \sum_{j=1}^{k} w_j(\xi)d\xi - \sum_{j=1}^{k} \nu_i h_j. \end{aligned} \tag{4}$$

E is differentiable w.r.t. $\boldsymbol{h}$ [9]. Its gradient and Hessian are then given by

$$\nabla E(\boldsymbol{h}) = (w_1(\boldsymbol{h}) - \nu_1, ..., w_k(\boldsymbol{h}) - \nu_k)^T, \tag{5}$$

$$H = \frac{\partial^2 E(\boldsymbol{h})}{\partial h_i \partial h_j} = \begin{cases} \sum_l \dfrac{\int_{f_{il}} \mu(x)dx}{\|y_l - y_i\|}, & i = j, \ \forall l, s.t. \ f_{il} \neq \emptyset, \\ -\dfrac{\int_{f_{ij}} \mu(x)dx}{\|y_j - y_i\|}, & i \neq j, \ f_{ij} \neq \emptyset, \\ 0, & i \neq j, \ f_{ij} = \emptyset, \end{cases} \tag{6}$$

where $\|\cdot\|$ is the $L1$–norm and $\int_{f_{ij}} \mu(x)dx = \text{vol}(f_{ij})$ is the volume of the intersection f_{ij} between two adjacent cells. Fig. 1 (b) illustrates the geometric

Algorithm 2: Iterative measure-preserving mapping

Function `Iterative-Measure-Preserving-Mapping`$(X(x,\mu), Y(y,\nu))$
  repeat
    $V(\boldsymbol{h}) \leftarrow$ Variational-OT(x,μ,y,ν). // 1. Update Voronoi partition
    $y_j \leftarrow \sum_{x\in V_j} \mu_i x_i / \sum_{x\in V_j} \mu_i$. // 2. Update y
  until y converges.
  return y, V.
end

relation. The Hessian H is positive semi-definite with only constant functions spanned by a vector $(1, ..., 1)^T$ in its null space. Thus, E is strictly convex in $\boldsymbol{h}$. By Newton's method, we solve a linear system,

$$H\delta\boldsymbol{h} = \nabla E(\boldsymbol{h}), \tag{7}$$

and update $\boldsymbol{h}^{(t+1)} \leftarrow \boldsymbol{h}^{(t)} + \delta\boldsymbol{h}^{(t)}$. The energy E (4) is motivated by Theorem 1 which seeks a solution to $vol(x \in \Omega \mid \nabla\theta_{\boldsymbol{h}}(x) = y_j) = \nu_j$. Move the right-hand side to left and take the integral over $\boldsymbol{h}$ then it becomes E (4). Thus, minimizing (4) when the gradient approaches $\mathbf{0}$ gives the solution. We show the complete algorithm for obtaining the OT-Map $\pi : X \to Y$ in Alg. 1.

5 Variational Wasserstein Clustering

We now introduce in detail our method to solve clustering problems through variational optimal transportation. We name it *variational Wasserstein clustering* (VWC). We focus on the semi-discrete clustering problem which is to find a set of discrete sparse centroids to best represent a continuous probability measure, or its discrete empirical representation. Suppose M is a metric space and we embody in it an empirical measure $X(x,\mu)$. Our goal is to find such a sparse measure $Y(y,\nu)$ that minimizes Eq. (3).

We begin with an assumption that the distributional data are embedded in the same Euclidean space $M = \mathbb{R}^n$, i.e. $X, Y \in \mathcal{P}(M)$. We observe that if ν is fixed then Eq. (2) and Eq. (3) are mathematically equivalent. Thus, the computational approaches to these problems could also coincide. Because the space is convex, each cluster is eventually a Voronoi cell and the resulting partition $V = \{(V_j, y_j)\}$ is actually a power Voronoi diagram where we have $\|x - y_j\|^2 - r_j^2 \leq \|x - y_i\|^2 - r_i^2$, $x \in V_j$, $\forall j \neq i$ and r is associated with the total mass of each cell. Such a diagram is also the solution to Monge's OT problem between X and Y. From the previous section, we know that if X and Y are fixed the power diagram is entirely determined by the minimizer $\boldsymbol{h}$. Thus, assuming ν is fixed and y is allowed to move freely in M, we reformulate Eq. (3) to

$$f(\boldsymbol{h}, y) = \sum_{j=1}^{K} \sum_{x_i \in V_j(\boldsymbol{h})} \mu_i \|x_i - y_j\|^2, \tag{8}$$

Algorithm 3: Variational Wasserstein clustering

Input : Empirical measures $X_M(x, \mu)$ and $Y_N(y, \nu)$
Output: Measure-preserving Map $\pi : X \to Y$ represented as (y, V).
begin
  $\nu \leftarrow$ Sampling-known-distribution. // Initialization.
  Harmonic-mapping: $M, N \to \mathbb{R}^n$ or $\mathbb{D}^n$. // Unify domains.
  $y, V \leftarrow$ Iterative-Measure-Preserving-Mapping(x, μ, y, ν).
end
return y, V.

where every V_j is a power Voronoi cell.

The solution to Eq. (8) can be achieved by iteratively updating $\boldsymbol{h}$ and y. While we can use Alg. 1 to compute $\boldsymbol{h}$, updating y can follow the rule:

$$y_j^{(t+1)} \leftarrow \sum \mu_i x_i^{(t)} \Big/ \sum \mu_i,\ x_i^{(t)} \in V_j. \tag{9}$$

Since the first step preserves the measure and the second step updates the measure, we call such a mapping an *iterative measure-preserving mapping*. Our algorithm repeatedly updates the partition of the space by variational-OT and computes the new centroids until convergence, as shown in Alg. 2. Furthermore, because each step reduces the total cost (8), we have the following propositions.

Proposition 1. *Alg. 2 monotonically minimizes the object function (8).*

Proof. It is sufficient for us to show that for any $t \geq 0$, we have

$$f(\boldsymbol{h}^{(t+1)}, y^{(t+1)}) \leq f(\boldsymbol{h}^{(t)}, y^{(t)}). \tag{10}$$

The above inequality is indeed true since $f(\boldsymbol{h}^{(t+1)}, y^{(t)}) \leq f(\boldsymbol{h}^{(t)}, y^{(t)})$ according to the convexity of our OT formulation, and $f(\boldsymbol{h}^{(t+1)}, y^{(t+1)}) \leq f(\boldsymbol{h}^{(t+1)}, y^{(t)})$ for the updating process itself minimizes the mean squared error. □

Corollary 1. *Alg. 2 converges in a finite number of iterations.*

Proof. We borrow the proof for k-means. Given N empirical samples and a fixed number k, there are k^N ways of clustering. At each iteration, Alg. 2 produces a new clustering rule only based on the previous one. The new rule induces a lower cost if it is different than the previous one, or the same cost if it is the same as the previous one. Since the domain is a finite set, the iteration must eventually enter a cycle whose length cannot be greater than 1 because otherwise it violates the fact of the monotonically declining cost. Therefore, the cycle has the length of 1 in which case the Alg. 2 converges in a finite number of iterations. □

Corollary 2. *Alg. 2 produces a unique (local) solution to Eq. (8).*

Proof. The initial condition, y the centroid positions, is determined. Each step of Alg. 2 yields a unique outcome, whether updating $\boldsymbol{h}$ by variational OT or updating y by weighted averaging. Thus, Alg. 2 produces a unique outcome. □

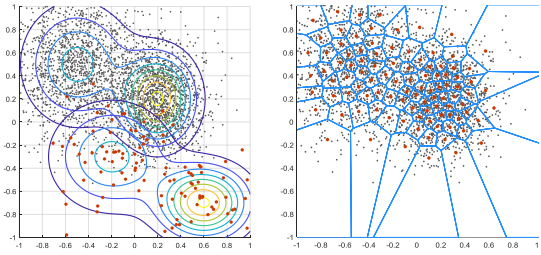

Fig. 2. Given the source domain (red dots) and target domain (grey dots), the distribution of the source samples are driven into the target domain and form a power Voronoi diagram.

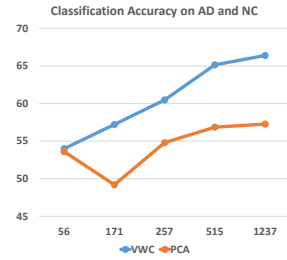

Fig. 3. Classification accuracies of VWC and PCA on AD and NC w.r.t. number of centroids.

Now, we introduce the concept of variational Wasserstein clustering. For a subset $M \subset \mathbb{R}^n$, let $\mathcal{P}(M)$ be the space of all Borel probability measures. Suppose $X(x, \mu) \in \mathcal{P}(M)$ is an existing one and we are to aggregate it into k clusters represented by another measure $Y(y, \nu) \in \mathcal{P}(M)$ and assignment $y_j = \pi(x)$, $j = 1, ..., k$. Thus, we have $\pi \in \mathcal{P}(M \times M)$. Given ν fixed, our goal is to find such a combination of Y and π that minimize the object function:

$$Y_{y,\nu} = \underset{\substack{Y \in P(M) \\ \pi \in P(M \times M)}}{argmin} \sum_{j=1}^{k} \sum_{y_j = \pi(x_i)} \mu_i \|x_i - y_j\|^2, \ s.t. \ \nu_j = \sum_{y_j = \pi(x_i)} \mu_i. \tag{11}$$

Eq. (11) is not convex w.r.t. y as discussed in [5]. We thus solve it by iteratively updating π and y. When updating π, since y is fixed, Eq. (11) becomes an optimal transportation problem. Therefore, solving Eq. (11) is equivalent to approaching the infimum of the 2-Wasserstein distance between X and Y:

$$\inf_{\substack{Y \in P(M) \\ \pi \in P(M \times M)}} \sum_{j=1}^{k} \sum_{y_j = \pi(x_i)} \mu_i \|x_i - y_j\|^2 = \inf_{Y \in P(M)} W_2^2(X, Y). \tag{12}$$

Assuming the domain is convex, we can apply iterative measure-preserving mapping (Alg. 2) to obtain y and h which induces π. In case that X and Y are not in the same domain i.e. $Y(y, \nu) \in P(N)$, $N \subset \mathbb{R}^n$, $N \neq M$, or the domain is not necessarily convex, we leverage *harmonic mapping* [28,29] to map them to a convex canonical space. We wrap up our complete algorithm in Alg. 3. Fig. 2 illustrates a clustering result. Given a source Gaussian mixture (red dots) and a target Gaussian mixture (grey dots), we cluster the target domain with the source samples. Every sample has the same mass in each domain for simplicity. Thus, we obtain an unweighted Voronoi diagram. In the next section, we will show examples that involve different mass. We implement our algorithm in C/C++ and adopt Voro++ [30] to compute Voronoi diagrams. The code is available at https://github.com/icemiliang/vot.

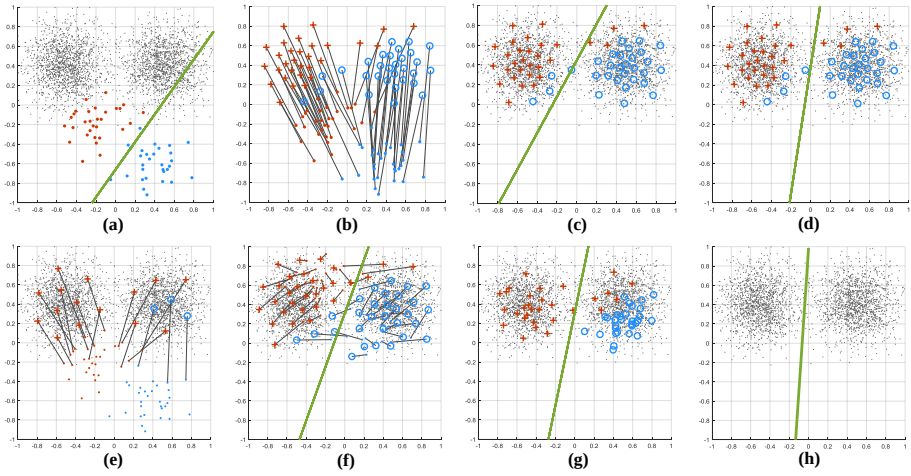

Fig. 4. SVM RBF boundaries for domain adaptation. (a) Target domain in gray dots and source domain of two classes in red and blue dots; (b) Mapping of centroids by using VWC; (c,d) Boundaries from VWC with linear and RBF kernels; (e) k-means++ [32] fails to produce a model; (f) After recentering source and target domains, k-means++ yields acceptable boundary; (g) D2 [7]; (h) JDOT [10], final centroids not available.

6 Applications

While the k-means clustering problem is ubiquitous in numerous tasks in computer vision and machine learning, we present the use of our method in approaching domain adaptation, remeshing, and representation learning.

6.1 Domain Adaptation on Synthetic Data

Domain adaptation plays a fundamental role in knowledge transfer and has benefited many different fields such as scene understanding and image style transfer. Several works have coped with domain adaptation by transforming distributions in order to close their gap with respect to a measure. In recent years, Courty *et al.* [31] took the first steps in applying optimal transportation to domain adaptation. Here we revisit this idea and provide our own solution to *unsupervised many-to-one domain adaptation* based on variational Wasserstein clustering.

Consider a two-class classification problem in the 2D Euclidean space. The source domain consists of two independent Gaussian distributions sampled by red and blue dots as shown in Fig. 4 (a). Each class has 30 samples. The target domain has two other independent Gaussian distributions with different means and variances, each having 1500 samples. They are represented by denser gray dots to emulate the source domain after an unknown transformation.

We adopt support vector machine (SVM) with linear and radial basis function (RBF) kernels for classification. The kernel scale for RBF is 5. One can notice

that directly applying the RBF classifier learned from the source domain to the target domain provides a poor classification result (59.80%). While Fig. 4 (b) shows the final positions of the samples from the source domain by VWC, (c) and (d) show the decision boundaries from SVMs with a linear kernel and an RBF kernel, respectively. In (e) and (f) we show the results from the classic k-means++ method [32]. In (e) k-means++ fails to cluster the unlabeled samples into the original source domain and produces an extremely biased model that has 50% of accuracy. Only after we recenter the source and the target domains yields k-means++ better results as shown in (f).

For more comparison, we test two other methods – D2 [7] and JDOT [10]. The final source positions from D2 are shown in (g). Because D2 solves the general barycenter problem and also updates the weights of the source samples, it converges as soon as it can find them some positions when the weights can also satisfy the minimum clustering loss. Thus, in (g), most of the source samples dive into the right, closer density, leaving those moving to the left with larger weights. We show the decision boundary obtained from JDOT [10] in (h). JDOT does not update the centroids, so we only show its decision boundary. In this experiment, both our method for Monge's OT and the methods [10,7] for Kantorovich's OT can effectively transfer knowledge between different domains, while the traditional method [32] can only work after a prior knowledge between the two domains, e.g. a linear offset. Detailed performance is reported in Tab. 1.

6.2 Deforming Triangle Meshes

Triangle meshes is a dominant approximation of surfaces. Refining triangle meshes to best represent surfaces have been studied for decades, including [33,34,35]. Given limited storage, we prefer to use denser and smaller triangles to represent the areas with relatively complicated geometry and sparser and larger triangles for flat regions. We follow this direction and propose to use our method to solve this problem. The idea is to drive the vertices toward high-curvature regions.

We consider a surface $\mathbb{S}^2$ approximated by a triangle mesh $T_{\mathbb{S}^2}(v)$. To drive the vertices to high-curvature positions, our general idea is to reduce the areas of the triangles in there and increase them in those locations of low curvature, producing a new triangulation $T'_{\mathbb{S}^2}(v)$ on the surface. To avoid computing the geodesic on the surface, we first map the surface to a *unit disk* $\phi : \mathbb{S}^2 \to \mathbb{D}^2 \subset \mathbb{R}^2$

Table 1. Classification Accuracy for Domain Adaptation on Synthetic Data

	k-means++ [32]*	k-means++^{r}		D2 [7]		JDOT [10]		VWC	
Kernel	Linear/RBF	Linear	RBF	Linear	RBF	Linear	RBF	Linear	RBF
Acc.	50.00	97.88	99.12	95.85	99.25	99.03	99.23	98.56	99.31
Sen.	100.00	98.13	98.93	99.80	99.07	98.13	99.60	98.00	99.07
Spe.	0.00	97.53	99.27	91.73	99.40	99.93	98.87	99.07	99.53

*: extremely biased model labeling all samples with same class; ^{r}: after recenterd.

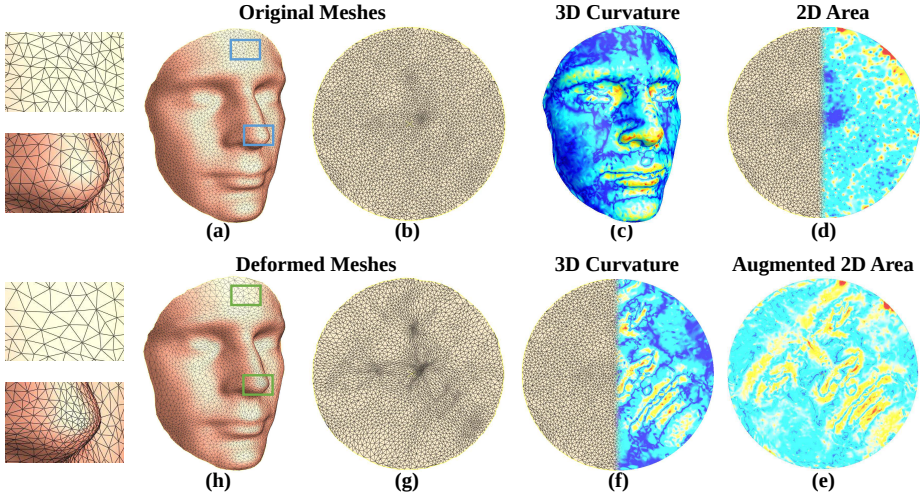

Fig. 5. Redistribute triangulation based on curvature. Original mesh (a) is mapped to a unit disk (b). Mean curvature on the 3D mesh (c) is copied to the disk (f). Design an "augmented" measure μ (e) on the disk by incorporating curvature C into 2D vertex area A (d), e.g. $\mu = 0.4A + 0.6C$. A vertex y with a large curvature C, in order to maintain its original measure A, will shrink its own cluster. As a result vertices collapse in high-curvature regions (g). Mesh will be pulled back to 3D (h) by inverse mapping.

and equip it with the Euclidean metric. We drop the superscripts 2 for simplicity. To clarify notations, we use $T_{\mathbb{S}}(v)$ to represent the original triangulation on surface $\mathbb{S}$; $T_{\mathbb{D}}(v)$ to represent its counterpart on $\mathbb{D}$ after harmonic mapping; $T'_{\mathbb{D}}(v)$ for the target triangulation on $\mathbb{D}$ and $T'_{\mathbb{S}}(v)$ on $\mathbb{S}$. Fig. 5 (a) and (b) illustrate the triangulation before and after the harmonic mapping. Our goal is to rearrange the triangulation on $\mathbb{D}$ and then the following composition gives the desired triangulation on the surface:

$$T_{\mathbb{S}}(v) \xrightarrow{\phi} T_{\mathbb{D}}(v) \xrightarrow{\pi} T'_{\mathbb{D}}(v) \xrightarrow{\phi^{-1}} T'_{\mathbb{S}}(v).$$

π is where we apply our method.

Suppose we have an original triangulation $T_{sub,\mathbb{S}}(v)$ and an initial downsampled version $T_{\mathbb{S}}(v)$ and we map them to $T_{sub,\mathbb{D}}(v)$ and $T_{\mathbb{D}}(v)$, respectively. The vertex area $A_{\mathbb{D}} : v \to a$ on $\mathbb{D}$ is the source (Dirac) measure. We compute the (square root of absolute) mean curvature $C_{sub,\mathbb{S}} : v_{sub} \to c_{sub}$ on $\mathbb{S}$ and the area $A_{sub,\mathbb{D}} : v_{sub} \to a_{sub}$ on $\mathbb{D}$. After normalizing a and c, a weighted summation gives us the target measure, $\mu_{sub,\mathbb{D}} = (1-\lambda)\ a_{sub,\mathbb{D}} + \lambda\ c_{sub,\mathbb{D}}$. We start from the source measure (v, a) and cluster the target measure (v_{sub}, μ_{sub}). The intuition is the following. If $\lambda = 0$, $\mu_{i,sub} = a_{i,sub}$ everywhere, then a simple unweighted Voronoi diagram which is the dual of $T_{\mathbb{D}}(v)$ would satisfy Eq. (12). As λ increases, the clusters $V_j(v_j, a_j)$ in the high-curvature $(c_{sub,\mathbb{D}})$ locations will require smaller areas $(a_{sub,\mathbb{D}})$ to satisfy $a_j = \sum_{v_{i,sub} \in V_j} \mu_{i,sub}$.

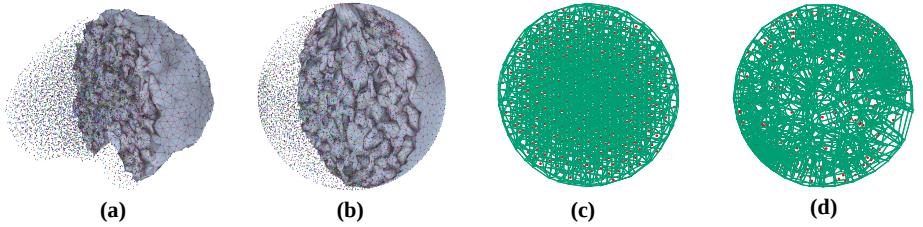

Fig. 6. Brain images are projected to tetrahedral meshes (a) that are generated from brain surfaces. Meshes are deformed into unit balls (b) via harmonic mapping. A sparse uniform distribution inside the ball (c) is initialized and shown with the initial Voronoi diagram. We start from (c) as the initial centroids and cluster (b) as the empirical measure by using our proposed method. (d) shows the resulting centroids and diagram.

We apply our method on a human face for validation and show the result in Fig. 5. On the left half, we show the comparison before and after the remeshing. The tip of the nose has more triangles due to the high curvature while the forehead becomes sparser because it is relatively flatter. The right half of Fig. 5 shows different measures that we compute for moving the vertices. (c) shows the mean curvature on the 3D surface. We map the triangulation with the curvature onto the planar disk space (f). (d) illustrates the vertex area of the planar triangulation and (e) is the weighted combination of 3D curvature and 2D area. Finally, we regard area (d) as the source domain and the "augmented" area (e) as the target domain and apply our method to obtain the new arrangement (g) of the vertices on the disk space. After that, we pull it back to the 3D surface (h). As a result, vertices are attracted into high-curvature regions. Note the boundaries of the deformed meshes (g,h) have changed after the clustering. We could restrict the boundary vertices to move *on* the unit circle if necessary. Rebuilding a Delaunay triangulation from the new vertices is also an optional step after.

6.3 Learning Representations of Brain Images

Millions of voxels contained in a 3D brain image bring efficiency issues for computer-aided diagnoses. A good learning technique can extract a better representation of the original data in the sense that it reduces the dimensionality and/or enhances important information for further processes. In this section, we address learning representations of brain images from a perspective of Wasserstein clustering and verify our algorithm on magnetic resonance imaging (MRI).

In the high level, given a brain image $X(x, \mu)$ where x represents the voxels and μ their intensities, we aim to cluster it with a known sparse distribution $Y(y, \nu)$. We consider that each brain image is a submanifold in the 3D Euclidean space, $M \subset \mathbb{R}^3$. To prepare the data, for each image, we first remove the skull and extract the surface of the brain volume by using Freesurfer [36], and use Tetgen [37] to create a tetrahedral mesh from the surface. Then, we project the

image onto the tetrahedral mesh by finding the nearest neighbors and perform harmonic mapping to deform it into a *unit ball* as shown in Fig. 6 (a) and (b).

Now, following Alg. 3, we set a discrete uniform distribution sparsely supported in the unit ball, $Y(y, \nu) \sim U_{\mathbb{D}^3}(-1, 1)$ as shown in Fig. 6 (c). Starting from this, we learn such a new y that the representation mapping $\pi : x \to y$ has the minimum cost (12). Thus, we can think of this process as a non-parametric mapping from the input to a latent space $\mathcal{P}(y)$ of dimension $k \times n \ll |x|$ where k is the number of clusters and n specifies the dimension of the original embedding space, e.g. 3 for brain images. Fig. 6 (d) shows the resulting centroids and the corresponding power diagram. We compare our method with principle component analysis (PCA) to show its capacity in dimensionality reduction. We apply both methods on 100 MRI images with 50 of them labeled Alzheimer's disease (AD) and 50 labeled normal control (NC). After obtaining the low-dimensional features, we directly apply a linear SVM classifier on them for 5-fold cross-validation. The plots in Fig. 3 show the superiority of our method. It is well known that people with AD suffer brain atrophy resulting in a group-wise shift in the images [38]. The result shows the potential of VWC in embedding the brain image in low-dimensional spaces. We could further incorporate prior knowledge such as regions-of-interest into VWC by hand-engineering initial centroids.

7 Discussion

Optimal transportation has gained increasing popularity in recent years thanks to its robustness and scalability in many areas. In this paper, we have discussed its connection to k-means clustering. Built upon variational optimal transportation, we have proposed a clustering technique by solving iterative measure-preserving mapping and demonstrated its applications to domain adaptation, remeshing, and learning representations.

One limitation of our method at this point is computing a high-dimensional Voronoi diagram. It requires complicated geometry processing which causes efficiency and memory issues. A workaround of this problem is to use gradient descent for variational optimal transportation because the only thing we need from the diagram is the intersections of adjacent convex hulls for computing the Hessian. The assignment of each empirical observation obtained from the diagram can be alternatively determined by nearest search algorithms. This is beyond the scope of this paper but it could lead to more real-world applications.

The use of our method for remeshing could be extended to the general feature redistribution problem on a compact 2–manifold. Future work could also include adding regularization to the centroid updating process to expand its applicability to specific tasks in computer vision and machine learning. The extension of our formulation of Wasserstein means to barycenters is worth further study.

Acknowledgemants The research is partially supported by National Institutes of Health (R21AG043760, RF1AG051710, and R01EB025032), and National Science Foundation (DMS-1413417 and IIS-1421165).

References

1. Lloyd, S.: Least squares quantization in pcm. IEEE transactions on information theory **28**(2) (1982) 129–137
2. Forgy, E.W.: Cluster analysis of multivariate data: efficiency versus interpretability of classifications. biometrics **21** (1965) 768–769
3. Graf, S., Luschgy, H.: Foundations of quantization for probability distributions. Springer (2007)
4. Agueh, M., Carlier, G.: Barycenters in the Wasserstein space. SIAM Journal on Mathematical Analysis **43**(2) (2011) 904–924
5. Cuturi, M., Doucet, A.: Fast computation of wasserstein barycenters. In: International Conference on Machine Learning. (2014) 685–693
6. Solomon, J., De Goes, F., Peyré, G., Cuturi, M., Butscher, A., Nguyen, A., Du, T., Guibas, L.: Convolutional wasserstein distances: Efficient optimal transportation on geometric domains. ACM Transactions on Graphics (TOG) **34**(4) (2015) 66
7. Ye, J., Wu, P., Wang, J.Z., Li, J.: Fast discrete distribution clustering using Wasserstein barycenter with sparse support. IEEE Transactions on Signal Processing **65**(9) (2017) 2317–2332
8. Ho, N., Nguyen, X., Yurochkin, M., Bui, H.H., Huynh, V., Phung, D.: Multilevel clustering via wasserstein means. arXiv preprint arXiv:1706.03883 (2017)
9. Gu, X., Luo, F., Sun, J., Yau, S.T.: Variational principles for minkowski type problems, discrete optimal transport, and discrete monge-ampere equations. arXiv preprint arXiv:1302.5472 (2013)
10. Courty, N., Flamary, R., Habrard, A., Rakotomamonjy, A.: Joint distribution optimal transportation for domain adaptation. In: Advances in Neural Information Processing Systems. (2017) 3733–3742
11. Monge, G.: Mémoire sur la théorie des déblais et des remblais. Histoire de l'Académie Royale des Sciences de Paris (1781)
12. Kantorovich, L.V.: On the translocation of masses. In: Dokl. Akad. Nauk SSSR. Volume 37. (1942) 199–201
13. Cuturi, M.: Sinkhorn distances: Lightspeed computation of optimal transport. In: Advances in neural information processing systems. (2013) 2292–2300
14. Brenier, Y.: Polar factorization and monotone rearrangement of vector-valued functions. Communications on pure and applied mathematics **44**(4) (1991) 375–417
15. Mérigot, Q.: A multiscale approach to optimal transport. In: Computer Graphics Forum. Volume 30., Wiley Online Library (2011) 1583–1592
16. Lévy, B.: A numerical algorithm for l2 semi-discrete optimal transport in 3d. ESAIM: Mathematical Modelling and Numerical Analysis **49**(6) (2015) 1693–1715
17. Givens, C.R., Shortt, R.M., et al.: A class of wasserstein metrics for probability distributions. The Michigan Mathematical Journal **31**(2) (1984) 231–240
18. Rubner, Y., Tomasi, C., Guibas, L.J.: The earth mover's distance as a metric for image retrieval. International journal of computer vision **40**(2) (2000) 99–121
19. Ling, H., Okada, K.: An efficient earth mover's distance algorithm for robust histogram comparison. IEEE transactions on pattern analysis and machine intelligence **29**(5) (2007) 840–853
20. Lee, K., Xu, W., Fan, F., Tu, Z.: Wasserstein introspective neural networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (June 2018)

21. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: International Conference on Machine Learning. (2017) 214–223
22. Frogner, C., Zhang, C., Mobahi, H., Araya, M., Poggio, T.A.: Learning with a wasserstein loss. In: Advances in Neural Information Processing Systems. (2015) 2053–2061
23. Gibbs, A.L., Su, F.E.: On choosing and bounding probability metrics. International statistical review **70**(3) (2002) 419–435
24. Applegate, D., Dasu, T., Krishnan, S., Urbanek, S.: Unsupervised clustering of multidimensional distributions using earth mover distance. In: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM (2011) 636–644
25. Villani, C.: Topics in optimal transportation. Number 58. American Mathematical Soc. (2003)
26. Alexandrov, A.D.: Convex polyhedra. Springer Science & Business Media (2005)
27. Aurenhammer, F.: Power diagrams: properties, algorithms and applications. SIAM Journal on Computing **16**(1) (1987) 78–96
28. Gu, X.D., Yau, S.T.: Computational conformal geometry. International Press Somerville, Mass, USA (2008)
29. Wang, Y., Gu, X., Chan, T.F., Thompson, P.M., Yau, S.T.: Volumetric harmonic brain mapping. In: Biomedical Imaging: Nano to Macro, 2004. IEEE International Symposium on, IEEE (2004) 1275–1278
30. Rycroft, C.: Voro++: A three-dimensional voronoi cell library in c++. (2009)
31. Courty, N., Flamary, R., Tuia, D.: Domain adaptation with regularized optimal transport. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer (2014) 274–289
32. Arthur, D., Vassilvitskii, S.: k-means++: The advantages of careful seeding. In: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics (2007) 1027–1035
33. Shewchuk, J.R.: Delaunay refinement algorithms for triangular mesh generation. Computational geometry **22**(1-3) (2002) 21–74
34. Fabri, A., Pion, S.: Cgal: The computational geometry algorithms library. In: Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems, ACM (2009) 538–539
35. Goes, F.d., Memari, P., Mullen, P., Desbrun, M.: Weighted triangulations for geometry processing. ACM Transactions on Graphics (TOG) **33**(3) (2014) 28
36. Fischl, B.: Freesurfer. Neuroimage **62**(2) (2012) 774–781
37. Si, H., TetGen, A.: A quality tetrahedral mesh generator and three-dimensional delaunay triangulator. Weierstrass Institute for Applied Analysis and Stochastic, Berlin, Germany (2006) 81
38. Fox, N.C., Freeborough, P.A.: Brain atrophy progression measured from registered serial mri: validation and application to alzheimer's disease. Journal of Magnetic Resonance Imaging **7**(6) (1997) 1069–1075