Approximate Algorithms for k-Sparse Wasserstein Barycenter with Outliers

Qingyuan Yang, **Hu Ding**[*]

School of Computer Science and Technolog
University of Science and Technology of China
yangqingyuan@mail.ustc.edu.cn, huding@ustc.edu.cn

Abstract

Wasserstein Barycenter (WB) is one of the most fundamental optimization problems in optimal transportation. Given a set of distributions, the goal of WB is to find a new distribution that minimizes the average Wasserstein distance to them. The problem becomes even harder if we restrict the solution to be "k-sparse". In this paper, we study the k-sparse WB problem in the presence of outliers, which is a more practical setting since real-world data often contains noise. Existing WB algorithms cannot be directly extended to handle the case with outliers, and thus it is urgently needed to develop some novel ideas. First, we investigate the relation between k-sparse WB with outliers and the clustering (with outliers) problems. In particular, we propose a clustering based LP method that yields constant approximation factor for the k-sparse WB with outliers problem. Further, we utilize the coreset technique to achieve the $(1+\epsilon)$-approximation factor for any $\epsilon > 0$, if the dimensionality is not high. Finally, we conduct the experiments for our proposed algorithms and illustrate their efficiencies in practice.

1 Introduction

Let $P = \{p_1, p_2, \cdots, p_{n_1}\}$ and $Q = \{q_1, q_2, \cdots, q_{n_2}\}$ be two sets of weighted points in the Euclidean space $\mathbb{R}^d$, where we use $w_P(\cdot)$ (*resp.,* $w_Q(\cdot)$) to denote the non-negative weight function for each point of P (*resp.,* Q); we also assume that P and Q have the same total weight, *i.e.,* $\sum_{i=1}^{n_1} w_P(p_i) = \sum_{j=1}^{n_2} w_Q(q_j) = n > 0$. For any $l \geq 1$, the seminal **Wasserstein Distance** [Rubner *et al.*, 2000] is to measure their minimum transportation cost:

$$\mathcal{W}(P,Q) = \min_{F} \Big(\sum_{i=1}^{n_1} \sum_{j=1}^{n_2} f_{ij} ||p_i - q_j||^l \Big)^{\frac{1}{l}}, \tag{1}$$

where $||\cdot||$ is the Euclidean distance, and the flow set $F = \{f_{ij} \mid 1 \leq i \leq n_1, 1 \leq j \leq n_2\}$ from P to Q should satisfy: $\sum_{i=1}^{n_1} f_{ij} = w_Q(q_j)$ for any j, and $\sum_{j=1}^{n_2} f_{ij} = w_P(p_i)$ for any i. We usually set $l = 1$ or 2 for most practical applications. Also note that "$||p_i - q_j||^l$" in (1) can be replaced by other distance function if the input P and Q are in some other metric space rather than the Euclidean space. The Wasserstein distance is one of the most fundamental topics in mathematics in past decades, and recently finds a broad range of applications in machine learning, such as image retrieval [Rubner *et al.*, 2000], generative model [Arjovsky *et al.*, 2017], and robust optimization [Kuhn *et al.*, 2019].

In this paper, we focus on an important optimization problem from Wasserstein distance, **k-sparse Wasserstein Barycenter (WB)** [Borgwardt and Patterson, 2021]. Suppose the input contains $m \geq 1$ weighted point sets $P_1, P_2, \cdots, P_m$, the problem of k-sparse WB for a given $k \in \mathbb{Z}^+$ is to construct a new weighted point set S with the support size $|\mathtt{supp}(S)| = k$, such that

$$\frac{1}{m} \sum_{j=1}^{m} \mathcal{W}^l(S, P_j) \tag{2}$$

is minimized. The Wasserstein barycenter [Agueh and Carlier, 2011] is a natural representation for the average of a set of given distributions. Recently, it has been applied to a number of real-world problems, such as medical imaging [Gramfort *et al.*, 2015], Bayesian learning [Srivastava *et al.*, 2018], clustering [Ho *et al.*, 2017], and natural language processing [Singh *et al.*, 2020]. In practice, we often prefer a simple representation of the barycenter with low complexity, and thus we have the sparsity requirement "$|\mathtt{supp}(S)| = k$".

Note that it is quite challenging to achieve a high-quality solution with theoretical guarantee for the objective (2); this is mainly due to two aspects, where one is from the inherent hardness for computing the Wasserstein distance, and the other reason is due to the restriction of "$|\mathtt{supp}(S)| = k$" which is a troublesome combinatorial constraint for the optimization. For example, [Borgwardt and Patterson, 2021] proved that the k-sparse WB problem is NP-hard, even if $m = 3$ and the dimensionality $d = 2$; on the other hand, if the k-sparse restriction is removed, one can compute Wasserstein barycenter in polynomial time in low dimensional space [Altschuler and Boix-Adserà, 2021].

We consider an even harder but also more practical variant of the k-sparse WB problem called "**k-sparse WB with outliers**". Roughly speaking, we allow a certain fraction of

[*] Corresponding author.

outliers for the mapping from each P_j to S in (2) (the formal definition is shown in Section 2). The motivation of allowing outliers is very natural in practical scenarios. Suppose we want to compute the WB for a set of images [Cuturi and Doucet, 2014]; it is common that the input images may contain noises and/or some irrelevant objects. To achieve a more robust solution, it is better to compute a barycenter that matches each input image partially (the un-matched part can be viewed as outliers).

Actually, the study on Wasserstein distance with outliers has already attracted attentions recently in machine learning [Chapel *et al.*, 2020; Mukherjee *et al.*, 2021; Le *et al.*, 2021; Nietert *et al.*, 2022]. However, the existing algorithms for computing WB with outliers are still quite limited, to the best of our knowledge. Though several robust models and algorithms for WB have been studied [Cazelles *et al.*, 2021; Le *et al.*, 2021], their methods do not explicitly handle outliers. In the current article, we consider developing approximate algorithms for k-sparse WB with outliers. For the sake of simplicity, we fix $l = 2$ for the Wasserstein distance in (1); actually our results can be easily extended for any $l \geq 1$. Our main ideas rely on the vanilla clustering algorithms. Suppose we have an α-approximation algorithm $\mathcal{A}$ for k-means clustering (*e.g.,* the algorithms from [Kanungo *et al.*, 2002; Arthur and Vassilvitskii, 2007]) and a β-approximation algorithm $\mathcal{B}$ for k-means clustering with outliers (*e.g.,* the algorithms from [Friggstad *et al.*, 2019; Gupta *et al.*, 2017]), where $\alpha, \beta \geq 1$. **Our contributions are as follows.**

- Our first contribution is to illustrate the relation between k-sparse WB with outliers and the problems of k-means clustering and k-means clustering with outliers. We are aware that the relation between the vanilla WB (without outliers) problem and k-means clustering has been discussed before [Cuturi and Doucet, 2014]. However, it is much more challenging to analyze the case with outliers; for example, even one single outlier can seriously destroy the clustering performance. In particular, we need to develop some significantly new insight to investigate the influence of outliers in the context of the complicated Wasserstein flows. We show that one can achieve an $O(\alpha)$-approximate solution by utilizing the algorithm $\mathcal{A}$, but the returned barycenter has the support size larger than k. If we want to keep the support size being equal to k, we can take advantage of $\mathcal{B}$ instead and achieve an $O(\beta)$-approximate solution.
- We further study the problem in low-dimensional space (*e.g.,* computing the WB for a set of 2D images). Our idea follows the aforementioned clustering method, but in a more sophisticated manner. We utilize the low-dimensional coreset technique [Har-Peled and Mazumdar, 2004] to generate a set of "anchor" points, and then build a set of non-uniform grids surrounding them. For any $\epsilon > 0$, if we relax the "k-sparse" requirement, we can compute a WB that achieves a $(1 + \epsilon)$ approximation factor based on those grids. Moreover, our result can be generalized to any metric space with constant doubling dimension (*e.g.,* the input distributions could have small intrinsic dimension even in high-dimensional space [Roweis and Saul, 2000]).

Remark 1. *The algorithms $\mathcal{A}$ and $\mathcal{B}$ can be bi-criteria approximation algorithms, that is, they can return more than k cluster centers (if we relax the k-sparse requirement). The benefit of using bi-criteria approximation algorithms is that they often achieve lower α and β. For example, the popular k-means++ algorithm yields an $O(\log k)$ approximation factor [Arthur and Vassilvitskii, 2007]; but if we run the k-means++ seeding procedure more than k steps (say λk with some constant integer $\lambda > 1$), the approximation factor can be reduced to be $O(1)$ [Aggarwal* et al.*, 2009].*

1.1 Related Works

Wasserstein distance. The research on computing the Wasserstein distance (1) has gained a great amount of attentions in theory and various practical applications. It is easy to see that (1) can be viewed as a min-cost flow problem that one can apply the network simplex algorithm to solve it [Ahuja *et al.*, 1988]. In machine learning community, [Cuturi, 2013] proposed a new variant called "Sinkhorn distance" that can be computed much faster than the original Wasserstein distance. Following Cuturi's work, [Altschuler *et al.*, 2017] proposed a nearly-linear time Wasserstein distance algorithm. [Genevay *et al.*, 2016] proposed a stochastic algorithm for solving large-scale optimal transportation. Some recent improvements also include [Dvurechensky *et al.*, 2018; Lin *et al.*, 2019]. As mentioned before, several works on the Wasserstein distance with outliers problem and its applications (*e.g.,* outlier detection and shape matching) were also proposed recently [Chapel *et al.*, 2020; Mukherjee *et al.*, 2021; Le *et al.*, 2021; Nietert *et al.*, 2022].

Wasserstein barycenter. The study of Wasserstein barycenter mainly focuses on two different types. One is called "**fixed-support WB**", where the barycenter S in (2) has a fixed support (the support size can be very large) and the task is to determine the weight distribution over the support such that the average Wasserstein distance to the given m input distributions is minimized. The other one is called "**free-support WB**" where the barycenter S can have the support that locates anywhere in the space. The former problem is relatively easier to solve, since the weight distribution can be obtained by computing a linear programming (LP) [Auricchio *et al.*, 2019]. A number of efficient algorithms for fixed-support WB have been proposed. For example, [Claici *et al.*, 2018] presented a stochastic algorithm for WB; [Ge *et al.*, 2019] developed a novel interior-point method by removing redundant constraints for the LP; [Lin *et al.*, 2020] provided a fast iterative Bregman projection algorithm.

On the other hand, the free-support WB problem is more challenging. Recently, [Altschuler and Boix-Adsera, 2022] showed that it is NP-hard to compute even a WB with ϵ additive error in Euclidean space (if d is not constant); only for low-dimensional space, one can obtain the optimal WB in polynomial time [Altschuler and Boix-Adserà, 2021]. [Borgwardt, 2022] provided a 2-approximate WB in Euclidean space. But those algorithms cannot guarantee small support for the obtained barycenter (*e.g.,* the support can be as large as $O(\sum_{j=1}^{m} |\mathtt{supp}(P_j)|)$).

If we further require $|\texttt{supp}(S)| = k$, the problem can be much more challenging; as mentioned before the k-sparse WB problem is NP hard even for $m = 3$ and $d = 2$ [Borgwardt and Patterson, 2021]. Most existing algorithms for k-sparse WB rely on the idea of alternating minimization, that is, they iteratively update the location of the k sparse support of S and the weight distribution, until the solution converges to some local optimum [Cuturi and Doucet, 2014; Ye *et al.*, 2017; Claici *et al.*, 2018; Ge *et al.*, 2019].

2 Preliminaries

To formally define the k-sparse WB with outliers problem, we should provide the definition for Wasserstein distance with outliers first (the vanilla Wasserstein distance (1) is the special case with zero outlier). For any weighted point set $P \subseteq \mathbb{R}^d$, we use $w_P(p)$ and $w_P(S)$ to denote the non-negative weight of a point $p \in P$ and the total weight of a subset $S \subset P$, respectively. We also define the relation "$\preceq$" between two point sets P and P': $P' \preceq P$ if $\texttt{supp}(P') = \texttt{supp}(P)$ and $w_{P'}(p) \leq w_P(p)$ for any $p \in \texttt{supp}(P)$. Further, we define the set $\mathbb{P}^P_{-z} = \{P' \subset \mathbb{R}^d \mid P' \preceq P$ and $w_{P'}(P') = w_P(P) - z\}$ for any given non-negative value $z \leq w_P(P)$.

Definition 1 (Wasserstein distance with z outliers)**.** *Let $n > z \geq 0$. Suppose $P = \{p_1, p_2, \cdots, p_{n_1}\}$ and $Q = \{q_1, q_2, \cdots, q_{n_2}\}$ are two sets of weighted points in $\mathbb{R}^d$; also $\sum_{i=1}^{n_1} w_P(p_i) = n$ and $\sum_{j=1}^{n_2} w_Q(q_j) = n - z$. The* ***Wasserstein distance with** z **outliers*** *from P to Q is*

$$\mathcal{W}_{-z}(P, Q) = \min_{P' \in \mathbb{P}^P_{-z}} \mathcal{W}(P', Q). \tag{3}$$

*The set $P^Q = \arg\min_{P' \in \mathbb{P}^P_{-z}} \mathcal{W}(P', Q)$ is called "**the inliers of** P **induced by** Q".*

Remark 2. *The objective function (3) was similarly defined as "unbalanced optimal transport" before [Chapel* et al.*, 2020; Pham* et al.*, 2020]. This definition is similar with the "trim" idea widely used in robust statistics [Rousseeuw and Leroy, 1987]. Also our definition for the function $\mathcal{W}_{-z}(P, Q)$ is one-side,* i.e., *only P contains outliers; actually, a more general definition can be two-side that both P and Q may contain outliers. We refer the reader to the recent articles [Mukherjee* et al.*, 2021; Nietert* et al.*, 2022] for more detailed discussion. Due to the space limit, we only present our results of one-side here, and leave the results for two-side (which can be easily extended from the one-side results) to our full paper [Yang and Ding, 2024].*

Computing $\mathcal{W}_{-z}(P, Q)$. Obviously, the total flow $\sum_{i=1}^{n_1} \sum_{j=1}^{n_2} f_{ij} = n - z$ in Definition 1. So the missing flows with the total weight z can be viewed as the outliers. Actually the problem of Wasserstein distance with z outliers can be easily reduced to the vanilla Wasserstein distance problem (1) via a "dummy point" idea that was studied in [Chapel *et al.*, 2020; Ding *et al.*, 2023] before. We add a dummy point q_* to Q with the weight equal to z; also we force the "distance" between q_* and each p_i to be 0. Note that we cannot find such a real point q_* in the space, where in reality we just need to set all the entries corresponding to the line of q_* to be 0 in the $n_1 \times (n_2 + 1)$ distance matrix. Then we can run any off-the-shelf Wasserstein distance algorithm, *e.g.,* the simplex network algorithm [Ahuja *et al.*, 1988] or the sinkhorn distance algorithm [Cuturi, 2013], to compute the solution. Intuitively, the dummy point q_* absorbs the furthest z outliers from P.

Claim 1. *Computing $\mathcal{W}_{-z}(P, Q)$ is equivalent to computing $\mathcal{W}(P, Q \cup \{q_*\})$.*

Definition 2 (k-sparse WB with z outliers)**.** *Let $n > z \geq 0$. Suppose the input $\mathbb{P}$ contains $m \geq 1$ weighted point sets $P_1, P_2, \cdots, P_m$ where each P_j has the total weight n. Then the problem of k-sparse Wasserstein Barycenter with z outliers for a given $k \in \mathbb{Z}^+$ is to construct a new weighted point set S with total weight $n - z$ and the size $|\texttt{supp}(S)| = k$, such that*

$$\texttt{Cost}_{-z}(\mathbb{P}, S) = \frac{1}{m} \sum_{j=1}^{m} \mathcal{W}^2_{-z}(P_j, S) \tag{4}$$

is minimized. Throughout this paper, we always use "$S_{\texttt{opt}}$" to denote the optimal solution of (4).

Remark 3. ***(Fixed-support WB with outliers)*** *As mentioned in Section 1.1, if we remove the "k-sparse" requirement and let the support of S be fixed to a given set G (|G| can be larger than k), the problem of WB can be solved by a linear programming [Ge* et al.*, 2019; Lin* et al.*, 2020]. Through the idea of Claim 1, the fixed-support WB with outliers problem can be also solved by using LP. Namely, we add a dummy point g_* to G and set its weight to be z; then the problem is exactly equivalent to the vanilla fixed-support WB problem on $G \cup \{g_*\}$. For completeness, we provide the detailed formulation in our full paper [Yang and Ding, 2024].*

Relation to k-means clustering with z outliers. To better illustrate our algorithms for k-sparse WB with outliers, we need to elaborate on its relation to k-means clustering with outliers first. Given a set P of weighted points in $\mathbb{R}^d$ with the total weight n, the goal of the vanilla k-means is to find k cluster centers $C = \{c_1, c_2, \cdots, c_k\}$, such that each point of P is assigned to its nearest cluster center and the total weighted squared distances $\mathcal{S}(P, C) = \sum_{p \in P} w_P(p) \cdot \min_{1 \leq s \leq k} ||p - c_s||^2$ is minimized. If we allow to discard z outliers, the goal becomes to find not only the k cluster centers, but also a set $P' \in \mathbb{P}^P_{-z}$, such that $\mathcal{S}(P', C)$ is minimized. We denote this optimal cost as $\texttt{Mean}^k_{-z}(P)$, and let $\texttt{C}^k_{-z}(P)$ denote the set of optimal cluster centers $\{c_1, c_2, \cdots, c_k\}$ with the weight $w_{\texttt{C}^k_{-z}}(c_s) =$ the total weight of the s-th cluster, $1 \leq s \leq k$. If $z = 0$, we use $\texttt{Mean}^k(P)$ and $\texttt{C}^k(P)$ for simplicity.

(1) First, we consider the basic case $m = 1$ for k-sparse WB with z outliers. It is easy to see that it is equivalent to the k-means clustering with z outliers on P_1.

Claim 2. *Suppose $m = 1$. The set $\texttt{C}^k_{-z}(P_1)$ forms the optimal solution for k-sparse WB with z outliers. Namely, for any $|\texttt{supp}(Q)| = k$, $\texttt{Mean}^k_{-z}(P_1) = \mathcal{W}^2_{-z}(P_1, \texttt{C}^k_{-z}(P_1)) \leq \mathcal{W}^2_{-z}(P_1, Q)$.*

(2) Then we consider the general case $m \geq 2$. From Claim 2, we know that the barycenter actually induces k

clusters and z outliers on P_1 for the case $m = 1$. When $m \geq 2$, the k points of the barycenter also induce k clusters on $\cup_{j=1}^{m} P_j$ but with additional constraint.

Claim 3. *The optimal solution of k-sparse WB with z outliers is equivalent to solving the k-means clustering with mz outliers on $\cup_{j=1}^{m} P_j$ with the following constraint: for each obtained cluster U_s, $1 \leq s \leq k$, $w_{U_s}(U_s \cap P_1) = w_{U_s}(U_s \cap P_2) = \cdots = w_{U_s}(U_s \cap P_m)$.*

Actually the above constrained k-means clustering with outliers problem is a special *fairness clustering with outliers* problem [Bera *et al.*, 2019]: suppose each P_j has a unique color, and we require that each color only takes $\frac{1}{m}$ of the total weight in each cluster C_s. Though a number of algorithms have been proposed for fairness clustering, the study on the case with outliers is still quite limited, to the best of our knowledge.

3 Our Clustering Based LP Algorithm

In this section we propose a clustering based LP algorithm for solving the k-sparse WB with outliers problem, where our main idea is inspired by the observations of Claim 2 and Claim 3. Though the algorithm is simple, the analysis on the quality is the major challenge since the inliers and outliers are mixed without any prior knowledge.

The clustering based LP algorithm. Let $\mathbb{P} = \{P_1, P_2, \cdots, P_m\}$ be an instance of the k-sparse WB with outliers problem as Definition 2. Assume $w_{\texttt{min}} = \min_{1 \leq j \leq m, p \in P_j} w_{P_j}(p)$, and denote by $\hat{z} = \lceil z / w_{\texttt{min}} \rceil$ for convenience. Our algorithm has the following three steps.

(1) We first run an α-approximate $(k + \hat{z})$-means clustering algorithm $\mathcal{A}$ on each P_j and obtain the set T_j of its $\lambda(k + \hat{z})$ cluster centers with some integer $\lambda \geq 1$ (as discussed in Remark 1, we can run a bi-criteria approximation algorithm).

(2) Then, for each T_j we consider the following fixed-support WB with outliers problem (as described in Remark 3): the support of the barycenter is fixed to be the $O(k + \hat{z})$ points of T_j, and compute the optimal weight distribution over T_j via LP.

(3) Let $\tilde{T}_1, \cdots, \tilde{T}_m$ be the obtained m candidate WBs from Step (2). We return the best one, say $\tilde{T}_{j_0}$, which has the smallest cost over the m candidates with respect to the cost (4).

The theoretical quality guarantee of $\tilde{T}_{j_0}$ is given in Theorem 1. Since $|\texttt{supp}(\tilde{T}_{j_0})| = O(k + \hat{z})$ that violates the k-sparse requirement, we can replace the algorithm $\mathcal{A}$ by a β-approximate k-means clustering with z outliers algorithm $\mathcal{B}$ in the above method. Then each T_j should have the support size exactly equal to k for $j = 1, 2, \cdots, m$. We still select the best candidate WB $\tilde{T}_{j_0}$ by the same manner, and return it as the solution for k-sparse WB with outliers. The improved result is shown in Theorem 2.

Theorem 1. *Our clustering based LP Algorithm returns a solution $\tilde{T}_{j_0}$ for k-sparse WB with outliers and achieves the following quality guarantee:*

$$\texttt{Cost}_{-z}(\mathbb{P}, \tilde{T}_{j_0}) \leq (2 + \sqrt{\alpha})^2 \cdot \texttt{Cost}_{-z}(\mathbb{P}, S_{\texttt{opt}}). \tag{5}$$

We have multiple choices for the algorithm $\mathcal{A}$. For example, we can run the $(9 + \epsilon)$-approximate local search algorithm [Kanungo *et al.*, 2002] (but its running time is super linear since there are too many swap combinations that should be tested in the local search procedure). We also can run the k-means++ based algorithms [Aggarwal *et al.*, 2009; Lattanzi and Sohler, 2019] to achieve an $O(1)$-approximation as discussed in Remark 1. So the approximation factor in Theorem 1 can be $O(1)$ as well.

To prove Theorem 1, we need several key lemmas. Lemma 1 shows that each obtained T_j can approximately represent the corresponding P_j, even in the presence of outliers. Lemma 2 further shows that the set $T_j^{S_{\texttt{opt}}}$, *i.e.,* the inliers of T_j induced by $S_{\texttt{opt}}$ (see Definition 1), should yield an upper bound for the total cost where the bound is determined by the distance between P_j and $S_{\texttt{opt}}$. Also note that we cannot obtain $T_j^{S_{\texttt{opt}}}$ in reality since the optimal solution $S_{\texttt{opt}}$ is always unknown to us; in fact we only use $T_j^{S_{\texttt{opt}}}$ in our analysis for bridging the gap between $\tilde{T}_{j_0}$ and $S_{\texttt{opt}}$. Through Lemma 2 we can prove that the selected best candidate $\tilde{T}_{j_0}$ yields the desired quality guarantee.

Lemma 1. *For each $1 \leq j \leq m$, suppose the obtained cluster centers from $\mathcal{A}$ is $T_j = \{t_1, t_2, \cdots, t_{\lambda(k+\hat{z})}\}$; also each weight $w_{T_j}(t_s)$ = the total weight of the s-th cluster. Then $\mathcal{W}_{-z}(T_j, S_{\texttt{opt}}) \leq (1 + \sqrt{\alpha}) \mathcal{W}_{-z}(P_j, S_{\texttt{opt}})$.*

Proof. First, we consider the relationship between k-means clustering with z outliers and $(k + \hat{z})$-means clustering. Intuitively, we can regard the result of k-means clustering with z outliers as a special solution for the $(k + \hat{z})$-means clustering, where each outlier actually is a cluster of single point. We then have the following claim (due to the space limit, the proof is shown in the full paper [Yang and Ding, 2024]).

Claim 4. $\texttt{Mean}^{k+\hat{z}}(P_j) \leq \texttt{Mean}^{k}_{-z}(P_j)$.

From Claim 2 we know that $\texttt{Mean}^{k}_{-z}(P_j) \leq \mathcal{W}^2_{-z}(P_j, S_{\texttt{opt}})$. Also, because T_j is obtained from the α-approximate algorithm $\mathcal{A}$, we have

$$\begin{aligned} \mathcal{W}(T_j, P_j)^2 &\leq \alpha \texttt{Mean}^{k+\hat{z}}(P_j) \\ &\leq \alpha \texttt{Mean}^{k}_{-z}(P_j) \leq \alpha \mathcal{W}^2_{-z}(P_j, S_{\texttt{opt}}). \end{aligned} \tag{6}$$

According to Definition 1, we know that $\mathcal{W}_{-z}(T_j, S_{\texttt{opt}}) = \mathcal{W}(T_j^{S_{\texttt{opt}}}, S_{\texttt{opt}})$, where $T_j^{S_{\texttt{opt}}}$ is the inliers of T_j induced by $S_{\texttt{opt}}$. Note $P_j^{S_{\texttt{opt}}}$ is a set with total weight $= n - z$, so we have $\mathcal{W}(T_j^{S_{\texttt{opt}}}, S_{\texttt{opt}}) \leq \mathcal{W}(T_j^{P_j^{S_{\texttt{opt}}}}, S_{\texttt{opt}})$. Thus,

$$\mathcal{W}_{-z}(T_j, S_{\texttt{opt}}) \leq \mathcal{W}(T_j^{P_j^{S_{\texttt{opt}}}}, S_{\texttt{opt}}). \tag{7}$$

We also have the following bound

$$\begin{aligned}
& \mathcal{W}(T_j^{P_j^{S_{\mathtt{opt}}}}, S_{\mathtt{opt}}) \\
\leq\ & \mathcal{W}(T_j^{P_j^{S_{\mathtt{opt}}}}, P_j^{S_{\mathtt{opt}}}) + \mathcal{W}(P_j^{S_{\mathtt{opt}}}, S_{\mathtt{opt}}) \\
=\ & \mathcal{W}_{-z}(T_j, P_j^{S_{\mathtt{opt}}}) + \mathcal{W}_{-z}(P_j, S_{\mathtt{opt}}) \\
\leq\ & \mathcal{W}(T_j, P_j) + \mathcal{W}_{-z}(P_j, S_{\mathtt{opt}}) \\
\leq\ & (1+\sqrt{\alpha})\mathcal{W}_{-z}(P_j, S_{\mathtt{opt}}), \qquad (8)
\end{aligned}$$

where the first inequality follows from the triangle inequality of Wasserstein Distance, and the last inequality follows from (6). Finally, we complete the proof by combining (7) and (8). □

Lemma 2. *For any* $1 \leq j \leq m$, $\mathtt{Cost}_{-z}(\mathbb{P}, T_j^{S_{\mathtt{opt}}}) \leq (2 + \sqrt{\alpha})\mathtt{Cost}_{-z}(\mathbb{P}, S_{\mathtt{opt}}) + (2 + 3\sqrt{\alpha} + \alpha)\mathcal{W}_{-z}^2(P_j, S_{\mathtt{opt}})$.

Proof. For any $1 \leq j_1 \leq m$, we have

$$\begin{aligned}
& \mathcal{W}_{-z}(P_{j_1}, T_j^{S_{\mathtt{opt}}}) \leq \mathcal{W}(P_{j_1}^{S_{\mathtt{opt}}}, T_j^{S_{\mathtt{opt}}}) \\
\leq\ & \mathcal{W}(P_{j_1}^{S_{\mathtt{opt}}}, S_{\mathtt{opt}}) + \mathcal{W}(S_{\mathtt{opt}}, T_j^{S_{\mathtt{opt}}}) \\
=\ & \mathcal{W}_{-z}(P_{j_1}, S_{\mathtt{opt}}) + \mathcal{W}_{-z}(T_j, S_{\mathtt{opt}}) \\
\leq\ & \mathcal{W}_{-z}(P_{j_1}, S_{\mathtt{opt}}) + (1+\sqrt{\alpha})\mathcal{W}_{-z}(P_j, S_{\mathtt{opt}}), \quad (9)
\end{aligned}$$

where the second inequality follows from the triangle inequality of Wasserstein distance and the third inequality follows from Lemma 1. Then we obtain the following bound by using (9):

$$\begin{aligned}
& \mathtt{Cost}_{-z}(\mathbb{P}, T_j^{S_{\mathtt{opt}}}) \\
\leq\ & \frac{1}{m}\sum\nolimits_{j_1=1}^{m}\Big(\mathcal{W}_{-z}(P_{j_1}, S_{\mathtt{opt}}) \\
& + (1+\sqrt{\alpha})\mathcal{W}_{-z}(P_j, S_{\mathtt{opt}})\Big)^2 \\
\leq\ & \frac{1}{m}\sum\nolimits_{j_1=1}^{m}(2+\sqrt{\alpha})\mathcal{W}_{-z}^2(P_{j_1}, S_{\mathtt{opt}}) \\
& + (2 + 3\sqrt{\alpha} + \alpha)\mathcal{W}_{-z}^2(P_j, S_{\mathtt{opt}}) \\
=\ & (2+\sqrt{\alpha})\mathtt{Cost}_{-z}(\mathbb{P}, S_{\mathtt{opt}}) \\
& + (2 + 3\sqrt{\alpha} + \alpha)\mathcal{W}_{-z}^2(P_j, S_{\mathtt{opt}}), \qquad (10)
\end{aligned}$$

where the second inequality follows from the fact that $(a + \delta b)^2 \leq (1+\delta)a^2 + (\delta^2 + \delta)b^2$ for any numbers a, b, and δ. □

Proof. [**of Theorem 1**] Because $\tilde{T}_j$ is the optimal weight distribution over T_j, we have $\mathtt{Cost}_{-z}(\mathbb{P}, \tilde{T}_j) \leq \mathtt{Cost}_{-z}(\mathbb{P}, T_j^{S_{\mathtt{opt}}})$. For the best candidate $\tilde{T}_{j_0}$, we have

$$\begin{aligned}
& \mathtt{Cost}_{-z}(\mathbb{P}, \tilde{T}_{j_0}) \leq \min\nolimits_{1\leq j\leq m}\mathtt{Cost}_{-z}(\mathbb{P}, T_j^{S_{\mathtt{opt}}}) \\
\leq\ & (2+\sqrt{\alpha})\ \mathtt{Cost}_{-z}(\mathbb{P}, S_{\mathtt{opt}}) \\
& + (2 + 3\sqrt{\alpha} + \alpha)\min\nolimits_{1\leq j\leq m}\mathcal{W}_{-z}^2(P_j, S_{\mathtt{opt}}) \quad (11)
\end{aligned}$$

based on Lemma 2. Also it is easy to know $\min_{1\leq j\leq m}\mathcal{W}_{-z}^2(P_j, S_{\mathtt{opt}}) \leq \frac{1}{m}\sum_{j=1}^{m}\mathcal{W}_{-z}^2(P_j, S_{\mathtt{opt}}) = \mathtt{Cost}_{-z}(\mathbb{P}, S_{\mathtt{opt}})$, so (11) implies $\mathtt{Cost}_{-z}(\mathbb{P}, \tilde{T}_{j_0}) \leq (2 + \sqrt{\alpha})^2\mathtt{Cost}_{-z}(\mathbb{P}, S_{\mathtt{opt}})$ which completes the proof. □

Theorem 2. *If we run the* β*-approximate* k*-means clustering with* z *outliers algorithm* $\mathcal{B}$ *instead of* $\mathcal{A}$*, and compute the optimal weight distribution with* $2z$ *outliers over* T_j *in the clustering based LP algorithm, we have*

$$\mathtt{Cost}_{-2z}(\mathbb{P}, \tilde{T}_{j_0}) \leq (2 + \sqrt{\beta})^2 \cdot \mathtt{Cost}_{-z}(\mathbb{P}, S_{\mathtt{opt}}). \qquad (12)$$

Comparing with Theorem 1, it is guaranteed that the output $\tilde{T}_{j_0}$ in Theorem 2 is exactly k-sparse, with only a violation on the size of outliers. The total weight of discarded outliers is increased to $2z$; but usually z is a value much smaller than n and thus we believe this influence is acceptable in practice. For the algorithm $\mathcal{B}$, we can use the $O(1)$-approximate algorithms [Friggstad *et al.*, 2019; Krishnaswamy *et al.*, 2018] (the algorithm of [Friggstad *et al.*, 2019] slightly violates the number of returned cluster centers to be $(1+\epsilon)k$ with an arbitrarily small value $\epsilon > 0$). In practice, we can also use some faster algorithms like [Chawla and Gionis, 2013; Gupta *et al.*, 2017] (though their theoretical guarantees are weaker than [Friggstad *et al.*, 2019; Krishnaswamy *et al.*, 2018]).

The proof of Theorem 2 is similar with that of Theorem 1, but the only major challenge is that we have to provide a more complicated version for Lemma 1 (which is Lemma 3 below). For each $1 \leq j \leq m$, let the obtained cluster centers from $\mathcal{B}$ be $T_j = \{t_1, t_2, \cdots, t_k\}$; the key difficult problem is that T_j and $S_{\mathtt{opt}}$ may induce different inliers on P_j. To resolve this issue, we should make a deep analysis on the distribution of T_j and prove the existence of a set $T_j' \preceq T_j$ who can play the same role as T_j in Lemma 1. Due to the space limit, we leave the proofs of Lemma 3 and Lemma 4 to our full paper.

Lemma 3. *There exists a weighted point set* T_j' *satisfying* $\mathcal{W}_{-z}(S_{\mathtt{opt}}, T_j') \leq (1 + \sqrt{\beta})\mathcal{W}_{-z}(P_j, S_{\mathtt{opt}})$*, where* $\mathtt{supp}(T_j') = \mathtt{supp}(T_j)$ *and* $w_{T_j'}(T_j') = n - 2z$.

Lemma 4. *For any* $1 \leq j \leq m$*, the weighted point set* T_j' *in Lemma 3 also satisfies* $\mathtt{Cost}_{-2z}(\mathbb{P}, T_j') \leq (2 + \sqrt{\beta})\mathtt{Cost}_{-z}(\mathbb{P}, S_{\mathtt{opt}}) + (2 + 3\sqrt{\beta} + \beta)\mathcal{W}_{-z}^2(P_j, S_{\mathtt{opt}})$.

Proof. [**of Theorem 2**] Because of $\tilde{T}_j$ is the optimal weight distribution over T_j, so we have $\mathtt{Cost}_{-2z}(\mathbb{P}, \tilde{T}_j) \leq \mathtt{Cost}_{-2z}(\mathbb{P}, T_j')$. For the best candidate $\tilde{T}_{j_0}$, we have

$$\begin{aligned}
& \mathtt{Cost}_{-2z}(\mathbb{P}, \tilde{T}_{j_0}) \leq \min\nolimits_{1\leq j\leq m}\mathtt{Cost}_{-2z}(\mathbb{P}, T_j') \\
\leq\ & (2+\sqrt{\beta})\mathtt{Cost}_{-z}(\mathbb{P}, S_{\mathtt{opt}}) \\
& + (2 + 3\sqrt{\beta} + \beta)\min\nolimits_{1\leq j\leq m}\mathcal{W}_{-z}^2(P_j, S_{\mathtt{opt}}) \\
\leq\ & (2+\sqrt{\beta})^2\mathtt{Cost}_{-z}(\mathbb{P}, S_{\mathtt{opt}}), \qquad (13)
\end{aligned}$$

where the second inequality follows from Lemme 4 and the last inequality follows from $\min_{1\leq j\leq m}\mathcal{W}_{-z}^2(P_j, S_{\mathtt{opt}}) \leq \mathtt{Cost}_{-z}(\mathbb{P}, S_{\mathtt{opt}})$. □

Analysis on running time. Let Γ_1 be the time complexity of $\mathcal{A}$ or $\mathcal{B}$ on each input set P_j, and let Γ_2 be the time complexity for solving the fixed-support WB for each $\tilde{T}_j$ as described in Remark 3. Then the total time complexity of our algorithm is $m(\Gamma_1 + \Gamma_2)$. Usually Γ_1 can be linear in n if using the previous clustering algorithms (e.g., [Aggarwal *et al.*, 2009]) ; the complexity Γ_2 can be $\tilde{O}(mn^{7/3}\epsilon^{-4/3})$ if using the recent algorithm [Lin *et al.*, 2020] ($\epsilon > 0$ is a prespecified small error).

4 Improvement in Low-Dimensional Space

We further consider a common case that the dimensionality d is small (*e.g.,* the input $\mathbb{P}$ is a set of 2D images). In the previous section, we show that the quality of our solution heavily depends on the clustering performance of the algorithm $\mathcal{A}$ or $\mathcal{B}$. So a natural question is

Can we remove the dependence on the factors α and β in our quality guarantee?

We answer this question in the affirmative. Our main idea is to generate the support through a more sophisticated approach. Our algorithm contains the following two steps.

(1) Generate the anchor points. We still run the α-approximate $(k+\hat{z})$-means clustering algorithm $\mathcal{A}$ on each P_j and obtain the set $T_j = \{t_1, \cdots, t_{\lambda(k+\hat{z})}\}$ of its cluster centers. To have a higher-quality fixed support for replacing T_j, we utilize the low-dimensional coreset[1] technique [Har-Peled and Mazumdar, 2004] to generate a set of "anchor" points $\hat{T}_j$ as follows. Denote by $\mathbb{B}(c, r)$ the ball centered at point c with radius $r \geq 0$. We fix a j, and let $\bar{r}_j = \sqrt{\mathcal{W}(P_j, T_j)/n}$ and $\epsilon_1 > 0$. Then for $s = 1, 2, \cdots, \lambda(k+\hat{z})$, we partition the ball $\mathbb{B}(t_s, n\bar{r}_j)$ into $\lceil \log n \rceil + 1$ layers: $T_{j,s,0} = \mathbb{B}(t_s, \bar{r}_j)$ and $T_{j,s,h} = \mathbb{B}(t_s, \bar{r}_j 2^h) \backslash \mathbb{B}(t_s, \bar{r}_j 2^{h-1})$ for $h = 1, \cdots, \lceil \log(n) \rceil$. For each h-th layer, we can build a grid with the side length $\bar{r}_j \epsilon_1 2^{h-1}/\sqrt{\alpha d}$; each point of $P_j \cap T_{j,s,h}$ is assigned to its nearest grid point, and each grid point has the weight equal to the total weight of the points assigned to it. Finally, we have the set $\hat{T}_j$ which contains all the weighted grid points of the $\lceil \log n \rceil + 1$ layers. The size $|\hat{T}_j| = O\big((k+\hat{z})\log(n)\alpha^{d/2}/\epsilon_1^d\big)$. We call the union set $\cup_{j=1}^m \hat{T}_j$ as the "anchor points" (which is actually the coreset in [Har-Peled and Mazumdar, 2004]).

(2) Construct the support. Without loss of generality, we assume that the minimum and maximum pairwise distances of $\cup_{j=1}^m P_j$ are 1 and Δ, respectively. For each anchor point $q \in \cup_{j=1}^m \hat{T}_j$, we draw $\lceil \log \Delta \rceil + 1$ concentric balls $\mathbb{B}(q, 2^h)$, $h = 0, 1, \cdots, \lceil \log \Delta \rceil$. Let $\epsilon_2 > 0$. Inside each ball $\mathbb{B}(q, 2^h)$, we build a grid of side length $\epsilon_2 2^{h-1}/\sqrt{d}$. We denote the union set of the $\lceil \log \Delta \rceil + 1$ grids as G_q, and denote $\bar{G} = \bigcup_{q \in \cup_{j=1}^m \hat{T}_j} G_q$.

Finally, we solve the fixed-support WB with outliers problem by using the LP method on $\bar{G}$ instead of the T_js. The obtained solution is denoted by $\tilde{G}$.

Theorem 3. *Our Algorithm returns a solution $\tilde{G}$ that has the following quality guarantee by setting $\epsilon_1 = \epsilon_2 = \epsilon/16$:*

$$\texttt{Cost}_{-z}(\mathbb{P}, \tilde{G}) \leq (1+\epsilon) \cdot \texttt{Cost}_{-z}(\mathbb{P}, S_{\texttt{opt}}). \tag{14}$$

Before proving Theorem 3, we provide the following two key lemmas first. Lemma 5 shows that each obtained $\hat{T}_j$ can efficiently preserve the Wasserstein distance error for P_j within any arbitrarily small bound, even in the presence of outliers. Lemma 6 further shows that the instance $\mathbb{T} = \{\hat{T}_1, \hat{T}_2, \cdots, \hat{T}_m\}$ yields a barycenter on the fix support $\bar{G}$ which can approximately represent the barycenter on the input instance $\mathbb{P}$. The detailed proofs are placed to our full paper [Yang and Ding, 2024].

Lemma 5. *For each $1 \leq j \leq m$, we have*

$$\mathcal{W}(P_j, \hat{T}_j) \leq \sqrt{1.25}\epsilon_1 \mathcal{W}_{-z}(P_j, S_{\texttt{opt}}). \tag{15}$$

Lemma 6. *Let $\mathbb{T} = \{\hat{T}_1, \hat{T}_2, \cdots, \hat{T}_m\}$ be a new instance of k-sparse WB with outliers, then we have*

$$\texttt{Cost}_{-z}(\mathbb{T}, \tilde{G}) \leq (1+\epsilon_2)^2(1+\sqrt{1.25}\epsilon_1)^2\texttt{Cost}_{-z}(\mathbb{P}, S_{\texttt{opt}}). \tag{16}$$

Proof. [**of Theorem 3**] We can combine Lemma 5 and Lemma 6 to complete the proof. Let $\delta > 0$ be a parameter that will be determined later. First, we have

$$\begin{aligned}
&\texttt{Cost}_{-z}(\mathbb{P}, \tilde{G}) = \frac{1}{m}\sum\nolimits_{j=1}^m \mathcal{W}_{-z}^2(P_j, \tilde{G}) \\
\leq\ & \frac{1}{m}\sum\nolimits_{j=1}^m \Big(\mathcal{W}(P_j, \hat{T}_j) + \mathcal{W}_{-z}(\hat{T}_j, \tilde{G})\Big)^2 \\
\leq\ & \frac{1}{m}\sum\nolimits_{j=1}^m \Big((1+\delta)\, \mathcal{W}^2(P_j, \hat{T}_j) \\
&+ (1+\frac{1}{\delta})\, \mathcal{W}_{-z}^2(\hat{T}_j, \tilde{G})\Big),
\end{aligned} \tag{17}$$

where the second inequality follows from the generalized triangle inequality for any real numbers $\delta > 0$, a, and b: $(a+b)^2 \leq (1+\delta)a^2 + (1+\frac{1}{\delta})b^2$. Then from (17) we have $\texttt{Cost}_{-z}(\mathbb{P}, \tilde{G}) \leq (1+\delta)\frac{1}{m}\sum_{j=1}^m \mathcal{W}_{-z}^2(P_j, S_{\texttt{opt}}) + (1+\frac{1}{\delta})\texttt{Cost}_{-z}(\mathbb{T}, \tilde{G}) \leq ((1+\delta)1.25\epsilon_1^2 + (1+\frac{1}{\delta})(1+\epsilon_2)^2(1+\sqrt{1.25}\epsilon_1)^2)\texttt{Cost}_{-z}(\mathbb{P}, S_{\texttt{opt}})$, where the second inequality follows from Lemma 5 and Lemma 6. Finally we choose $\delta = 1/(\sqrt{1.25}\epsilon_1(1+\epsilon_2)(1+\sqrt{1.25}\epsilon_1))$, then we have $\texttt{Cost}_{-z}(\mathbb{P}, \tilde{G}) \leq (\sqrt{1.25}\epsilon_1 + (1+\epsilon_2)(1+\sqrt{1.25}\epsilon_1))^2\texttt{Cost}_{-z}(\mathbb{P}, S_{\texttt{opt}})$. By setting $\epsilon_1 = \epsilon_2 = \epsilon/16$ we obtain Theorem 3. □

Running time analysis. The running time is similar with the complexity of Section 3, where the only difference is adding the time complexity for building the anchor points and constructing the support in our algorithm. Note that the complexity of [Har-Peled and Mazumdar, 2004] is linear in the input size n for each P_j. The total complexity for this extra part is $\tilde{O}\Big(\log(\Delta)(k+\hat{z})\alpha^{d/2}\epsilon^{-2d} + n\Big)$.

Extension in doubling metric. Our result can be easily extended to the more general case in doubling metric. Informally speaking, the "doubling dimension" measures the intrinsic dimension of data (Euclidean dimension is one kind of special doubling dimension) [Gupta *et al.*, 2003]. We show the extension with details in our full paper.

[1] Coreset is an algorithmic technique for representing large-scale data, which has been widely used for the optimization problems like clustering and regression [Feldman, 2020].

5 Experiments

In this section, we illustrate the practical performance of our algorithms and study the significance of considering outliers for WB. Our experiments contain three parts. Firstly, we conduct the experiments on synthetic datasets, where the positions of the barycenter supports are predefined, allowing us to compute the exact optimal objective value for measuring the approximation ratio of our algorithm. Secondly, we compare our algorithms with several baselines on real-world datasets. Finally, we provide the visualized results on the MNIST dataset [LeCun *et al.*, 2010]. Some omitted experimental results are placed to our full paper [Yang and Ding, 2024].

Datasets. In our synthetic datasets, we set the supports size $k \in [10, 40]$ and the dimensionality $d \in [10, 40]$; each instance comprises $m \in [2, 10]$ different distributions, where each distribution consists of $n = 20,000$ points. The true barycenter supports are uniformly sampled within a hypercube with a side length of 10, and random weights are assigned to each center point. The points are randomly generated within Gaussian balls around the centers based on the assigned weights. We introduce outliers by uniformly sampling z points for each distribution within the cube, with z ranging from 0 to $0.15 \times n$.

We also select three widely-used datasets from the UCI repository [Dua and Graff, 2017]: **Bank** [Moro *et al.*, 2012] ($4,521$ points in $\mathbb{R}^3$) represents the individual telephone calls during a marketing campaign, which contains the information of the customers. We have $m = 3$ distributions categorized based on marital status. **Credit card** [Yeh, 2016] ($30,000$ points in $\mathbb{R}^{14}$) includes the information about the credit card holders. We partitioned the data into $m = 9$ distributions based on marriage and education. **Adult** [Becker and Kohavi, 1996] ($32,561$ points in $\mathbb{R}^5$) represents the individual information from the 1994 U.S. Census. We partitioned it into $m = 10$ distributions based on sex and race. Finally, 5% random noise are added to each dataset as outliers.

Baselines and our implementation. It is worth noting that there is no method that explicitly addresses k-sparse WB with outliers or fair clustering with outliers, to the best of our knowledge. We employ three baselines. First, following Remark 3, we consider the fixed-support WB with outliers algorithm, utilizing k random centers as support, and compute the optimal weight distribution via LP (denoted as "Random_$\mathcal{O}$"). The other two baselines include a fair clustering algorithm that does not consider outliers (denoted as "FC_$\mathcal{O}$") [Bera *et al.*, 2019], and a non-fair clustering method considering outliers "k-means- -_$\mathcal{O}$" [Chawla and Gionis, 2013]. For the FC algorithm, we identify the farthest points in each class as outliers; for k-means- -, after obtaining the support positions, a new fair clustering solution can be obtained through LP. Additionally, we also test their three "plain" versions that do not discard outliers, aiming to study the significance of considering outliers (denoted as "Random", "FC", "k-means- -", respectively).

In our implementation, we use the k-means++[Arthur and Vassilvitskii, 2007] as Algorithm $\mathcal{A}$ and k-means- -[Chawla and Gionis, 2013] as Algorithm $\mathcal{B}$; we also employ the LP solver [Gurobi Optimization, LLC, 2023] as the subroutine for solving fixed-support WB with outliers. To ensure a fair comparison, although Theorem 2 suggests removing $2z$ outliers, we remove only z outliers in reality. Also, to keep k-sparsity for the result returned by $\mathcal{A}$, we only retain the top k centers with the largest cluster sizes. We use "Our_$\mathcal{A}$" and "Our_$\mathcal{B}$" to denote them.

Results on synthetic datasets. We compute the optimal cost $\texttt{Cost}_{\texttt{opt}}$ for the WB problem by using the pre-specified barycenter support. Subsequently, we execute our algorithm under various parameters to obtain the $\texttt{Cost}$ and calculate the approximation ratio, defined as $\frac{\texttt{Cost}}{\texttt{Cost}_{\texttt{opt}}}$. Part of the results obtained by Algorithm $\mathcal{A}$ is presented in Table 1. We can see that our algorithm consistently achieves favorable approximation ratios across different dimensions and outlier proportions, where more than 70% of them are less than 1.5.

		Proportion of Outliers z/n						
d	k	0	0.025	0.05	0.075	0.1	0.125	0.15
10	10	1.321	1.380	1.477	1.651	1.547	1.452	1.493
	20	1.346	1.326	1.395	1.435	1.475	1.497	1.527
	30	1.370	1.375	1.397	1.434	1.476	1.496	1.558
	40	1.367	1.380	1.413	1.450	1.490	1.498	1.554
20	10	1.332	1.412	1.695	1.714	1.746	1.353	1.399
	20	1.349	1.459	1.789	1.423	1.429	1.455	1.485
	30	1.373	1.468	1.412	1.441	1.485	1.497	1.538
	40	1.386	1.422	1.420	1.495	1.520	1.575	1.602

Table 1: The approximation ratios of our algorithm for $m = 10$.

Results on real datasets. The results are illustrated in Figure 1. As can be seen, even with only 5% outliers, the plain versions of the three baselines take almost double costs than their counterparts who consider outliers. Moreover, our algorithms demonstrate even lower costs across all the datasets with different values of k.

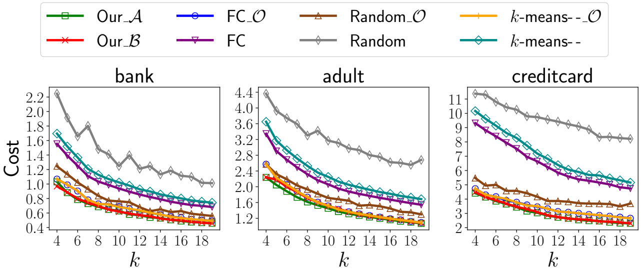

Figure 1: The obtained costs on real datasets.

Visualized results. In Figure 2 and Figure 3, we show the 40-sparse barycenters obtained by Our_$\mathcal{A}$ and Our_$\mathcal{B}$ for digit 0-9 in the MNIST dataset, with 2% of outliers removed from each digit. It is evident that the obtained set of 40 points effectively captures the distinctive features for each digit.

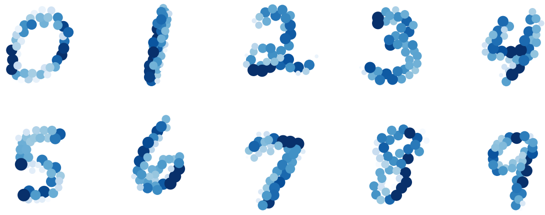

Figure 2: k-sparse WB obtained by Our_$\mathcal{A}$ for $k = 40$.

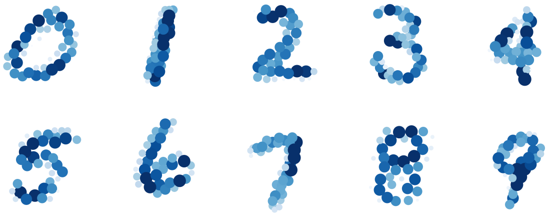

Figure 3: k-sparse WB obtained by Our_$\mathcal{B}$ for $k = 40$.

6 Conclusions

In this paper, we study the problem of k-sparse WB with outliers and present several efficient approximate algorithms with theoretical quality guarantees. Some omitted proofs are placed to our full paper [Yang and Ding, 2024]. Following this work, there are several interesting problems deserved to study in future. For example, inspired by the local search method for designing the PTAS algorithm for ordinary k-means clustering with outliers [Friggstad *et al.*, 2019], an interesting theoretical question is that whether we can also apply it to achieve a PTAS for k-sparse WB with outliers in low-dimensional space. Our current paper focuses more on the theoretical quality for computing WB with outliers, so an ignored question is that how to improve the time complexity (e.g., using some randomization techniques). From the perspective of applications, it is deserved to consider that using WB with outlier to handle some complex data fusion problems [Cheng *et al.*, 2021; Mao *et al.*, 2018; Buchin *et al.*, 2019].

Acknowledgments

The research of this work was supported in part by the National Key Research and Development Program of China 2021YFA1000900, the National Natural Science Foundation of China 62272432, and the Natural Science Foundation of Anhui Province 2208085MF163.

References

[Aggarwal *et al.*, 2009] Ankit Aggarwal, Amit Deshpande, and Ravi Kannan. Adaptive sampling for k-means clustering. In *International Workshop on Approximation Algorithms for Combinatorial Optimization*, pages 15–28. Springer, 2009.

[Agueh and Carlier, 2011] Martial Agueh and Guillaume Carlier. Barycenters in the wasserstein space. *SIAM Journal on Mathematical Analysis*, 43(2):904–924, 2011.

[Ahuja *et al.*, 1988] Ravindra K Ahuja, Thomas L Magnanti, and James B Orlin. Network flows. 1988.

[Altschuler and Boix-Adserà, 2021] Jason M. Altschuler and Enric Boix-Adserà. Wasserstein barycenters can be computed in polynomial time in fixed dimension. *J. Mach. Learn. Res.*, 22:44:1–44:19, 2021.

[Altschuler and Boix-Adsera, 2022] Jason M Altschuler and Enric Boix-Adsera. Wasserstein barycenters are np-hard to compute. *SIAM Journal on Mathematics of Data Science*, 4(1):179–203, 2022.

[Altschuler *et al.*, 2017] Jason Altschuler, Jonathan Niles-Weed, and Philippe Rigollet. Near-linear time approximation algorithms for optimal transport via sinkhorn iteration. *Advances in neural information processing systems*, 30:1964–1974, 2017.

[Arjovsky *et al.*, 2017] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.

[Arthur and Vassilvitskii, 2007] David Arthur and Sergei Vassilvitskii. k-means++: the advantages of careful seeding. In *SODA '07*, pages 1027–1035, 2007.

[Auricchio *et al.*, 2019] Gennaro Auricchio, Federico Bassetti, Stefano Gualandi, and Marco Veneroni. Computing wasserstein barycenters via linear programming. In *Integration of Constraint Programming, Artificial Intelligence, and Operations Research*, pages 355–363. Springer, 2019.

[Becker and Kohavi, 1996] Barry Becker and Ronny Kohavi. Adult. UCI Machine Learning Repository, 1996. DOI: https://doi.org/10.24432/C5XW20.

[Bera *et al.*, 2019] Suman Bera, Deeparnab Chakrabarty, Nicolas Flores, and Maryam Negahbani. Fair algorithms for clustering. *Advances in Neural Information Processing Systems*, 32:4955–4966, 2019.

[Borgwardt and Patterson, 2021] Steffen Borgwardt and Stephan Patterson. On the computational complexity of finding a sparse wasserstein barycenter. *J. Comb. Optim.*, 41(3):736–761, 2021.

[Borgwardt, 2022] Steffen Borgwardt. An lp-based, strongly-polynomial 2-approximation algorithm for sparse wasserstein barycenters. *Operational Research*, 22(2):1511–1551, 2022.

[Buchin *et al.*, 2019] Kevin Buchin, Anne Driemel, Natasja van de L'Isle, and André Nusser. klcluster: Center-based clustering of trajectories. In Farnoush Banaei Kashani, Goce Trajcevski, Ralf Hartmut Güting, Lars Kulik, and Shawn D. Newsam, editors, *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, SIGSPATIAL 2019, Chicago, IL, USA, November 5-8, 2019*, pages 496–499. ACM, 2019.

[Cazelles *et al.*, 2021] Elsa Cazelles, Felipe Tobar, and Joaquin Fontbona. A novel notion of barycenter for probability distributions based on optimal weak mass transport. *Advances in Neural Information Processing Systems*, 34:13575–13586, 2021.

[Chapel *et al.*, 2020] Laetitia Chapel, Mokhtar Z Alaya, and Gilles Gasso. Partial optimal tranport with applications on positive-unlabeled learning. *Advances in Neural Information Processing Systems*, 33:2903–2913, 2020.

[Chawla and Gionis, 2013] Sanjay Chawla and Aristides Gionis. k-means–: A unified approach to clustering and outlier detection. In *Proceedings of the 2013 SIAM international conference on data mining*, pages 189–197. SIAM, 2013.

[Cheng *et al.*, 2021] Kevin C. Cheng, Shuchin Aeron, Michael C. Hughes, and Eric L. Miller. Dynamical wasserstein barycenters for time-series modeling. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 27991–28003, 2021.

[Claici *et al.*, 2018] Sebastian Claici, Edward Chien, and Justin Solomon. Stochastic wasserstein barycenters. In *International Conference on Machine Learning*, pages 999–1008. PMLR, 2018.

[Cuturi and Doucet, 2014] Marco Cuturi and Arnaud Doucet. Fast computation of wasserstein barycenters. In *International conference on machine learning*, pages 685–693. PMLR, 2014.

[Cuturi, 2013] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

[Ding *et al.*, 2023] Hu Ding, Wenjie Liu, and Mingquan Ye. A data-dependent approach for high-dimensional (robust) wasserstein alignment. *ACM Journal of Experimental Algorithmics*, 28:1–32, 2023.

[Dua and Graff, 2017] Dheeru Dua and Casey Graff. Uci machine learning repository, 2017.

[Dvurechensky *et al.*, 2018] Pavel Dvurechensky, Alexander Gasnikov, and Alexey Kroshnin. Computational optimal transport: Complexity by accelerated gradient descent is better than by sinkhorn's algorithm. In *International conference on machine learning*, pages 1367–1376. PMLR, 2018.

[Feldman, 2020] Dan Feldman. Core-sets: Updated survey. *Sampling techniques for supervised or unsupervised tasks*, pages 23–44, 2020.

[Friggstad *et al.*, 2019] Zachary Friggstad, Kamyar Khodamoradi, Mohsen Rezapour, and Mohammad R Salavatipour. Approximation schemes for clustering with outliers. *ACM Transactions on Algorithms (TALG)*, 15(2):1–26, 2019.

[Ge *et al.*, 2019] Dongdong Ge, Haoyue Wang, Zikai Xiong, and Yinyu Ye. Interior-point methods strike back: Solving the wasserstein barycenter problem. *Advances in Neural Information Processing Systems*, 32, 2019.

[Genevay *et al.*, 2016] Aude Genevay, Marco Cuturi, Gabriel Peyré, and Francis Bach. Stochastic optimization for large-scale optimal transport. *Advances in neural information processing systems*, 29, 2016.

[Gramfort *et al.*, 2015] Alexandre Gramfort, Gabriel Peyré, and Marco Cuturi. Fast optimal transport averaging of neuroimaging data. In *Information Processing in Medical Imaging: 24th International Conference*, pages 261–272. Springer, 2015.

[Gupta *et al.*, 2003] Anupam Gupta, Robert Krauthgamer, and James R Lee. Bounded geometries, fractals, and low-distortion embeddings. In *44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.*, pages 534–543. IEEE, 2003.

[Gupta *et al.*, 2017] Shalmoli Gupta, Ravi Kumar, Kefu Lu, Benjamin Moseley, and Sergei Vassilvitskii. Local search methods for k-means with outliers. *Proceedings of the VLDB Endowment*, 10(7):757–768, 2017.

[Gurobi Optimization, LLC, 2023] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2023.

[Har-Peled and Mazumdar, 2004] Sariel Har-Peled and Soham Mazumdar. On coresets for k-means and k-median clustering. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 291–300, 2004.

[Ho *et al.*, 2017] Nhat Ho, XuanLong Nguyen, Mikhail Yurochkin, Hung Hai Bui, Viet Huynh, and Dinh Phung. Multilevel clustering via wasserstein means. In *International conference on machine learning*, pages 1501–1509. PMLR, 2017.

[Kanungo *et al.*, 2002] Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, and Angela Y Wu. A local search approximation algorithm for k-means clustering. In *Proceedings of the eighteenth annual symposium on Computational geometry*, pages 10–18, 2002.

[Krishnaswamy *et al.*, 2018] Ravishankar Krishnaswamy, Shi Li, and Sai Sandeep. Constant approximation for k-median and k-means with outliers via iterative rounding. In *Proceedings of the 50th annual ACM SIGACT symposium on theory of computing*, pages 646–659, 2018.

[Kuhn *et al.*, 2019] Daniel Kuhn, Peyman Mohajerin Esfahani, Viet Anh Nguyen, and Soroosh Shafieezadeh-Abadeh. Wasserstein distributionally robust optimization: Theory and applications in machine learning. In *Operations research & management science in the age of analytics*, pages 130–166. Informs, 2019.

[Lattanzi and Sohler, 2019] Silvio Lattanzi and Christian Sohler. A better k-means++ algorithm via local search. In *International Conference on Machine Learning*, pages 3662–3671. PMLR, 2019.

[Le *et al.*, 2021] Khang Le, Huy Nguyen, Quang M Nguyen, Tung Pham, Hung Bui, and Nhat Ho. On robust optimal transport: computational complexity and barycenter computation. *Advances in Neural Information Processing Systems*, 34:21947–21959, 2021.

[LeCun *et al.*, 2010] Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

[Lin *et al.*, 2019] Tianyi Lin, Nhat Ho, and Michael Jordan. On efficient optimal transport: An analysis of greedy and accelerated mirror descent algorithms. In *International Conference on Machine Learning*, pages 3982–3991. PMLR, 2019.

[Lin *et al.*, 2020] Tianyi Lin, Nhat Ho, Xi Chen, Marco Cuturi, and Michael Jordan. Fixed-support wasserstein barycenters: Computational hardness and fast algorithm. *Advances in Neural Information Processing Systems*, 33:5368–5380, 2020.

[Mao *et al.*, 2018] Jiali Mao, Qiuge Song, Cheqing Jin, Zhigang Zhang, and Aoying Zhou. Online clustering of streaming trajectories. *Frontiers Comput. Sci.*, 12(2):245–263, 2018.

[Moro *et al.*, 2012] S. Moro, P. Rita, and P. Cortez. Bank Marketing. UCI Machine Learning Repository, 2012. DOI: https://doi.org/10.24432/C5K306.

[Mukherjee *et al.*, 2021] Debarghya Mukherjee, Aritra Guha, Justin M Solomon, Yuekai Sun, and Mikhail Yurochkin. Outlier-robust optimal transport. In *International Conference on Machine Learning*, pages 7850–7860. PMLR, 2021.

[Nietert *et al.*, 2022] Sloan Nietert, Ziv Goldfeld, and Rachel Cummings. Outlier-robust optimal transport: Duality, structure, and statistical analysis. In *International Conference on Artificial Intelligence and Statistics*, pages 11691–11719. PMLR, 2022.

[Pham *et al.*, 2020] Khiem Pham, Khang Le, Nhat Ho, Tung Pham, and Hung Bui. On unbalanced optimal transport: An analysis of sinkhorn algorithm. In *International Conference on Machine Learning*, pages 7673–7682. PMLR, 2020.

[Rousseeuw and Leroy, 1987] Peter J. Rousseeuw and Annick Leroy. *Robust Regression and Outlier Detection*. Wiley Series in Probability and Statistics. Wiley, 1987.

[Roweis and Saul, 2000] Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326, 2000.

[Rubner *et al.*, 2000] Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99, 2000.

[Singh *et al.*, 2020] Sidak Pal Singh, Andreas Hug, Aymeric Dieuleveut, and Martin Jaggi. Context mover's distance & barycenters: Optimal transport of contexts for building representations. In *International Conference on Artificial Intelligence and Statistics*, pages 3437–3449. PMLR, 2020.

[Srivastava *et al.*, 2018] Sanvesh Srivastava, Cheng Li, and David B Dunson. Scalable bayes via barycenter in wasserstein space. *The Journal of Machine Learning Research*, 19(1):312–346, 2018.

[Yang and Ding, 2024] Qingyuan Yang and Hu Ding. Approximate algorithms for k-sparse wasserstein barycenter with outliers. *CoRR*, abs/2404.13401, 2024.

[Ye *et al.*, 2017] Jianbo Ye, Panruo Wu, James Z Wang, and Jia Li. Fast discrete distribution clustering using wasserstein barycenter with sparse support. *IEEE Transactions on Signal Processing*, 65(9):2317–2332, 2017.

[Yeh, 2016] I-Cheng Yeh. default of credit card clients. UCI Machine Learning Repository, 2016. DOI: https://doi.org/10.24432/C55S3H.