Coresets for Fuzzy K-Means with Applications

Johannes Blömer
Department of Computer Science, Paderborn University, Paderborn, Germany
bloemer@upb.de

Sascha Brauer
Department of Computer Science, Paderborn University, Paderborn, Germany
sascha.brauer@upb.de

Kathrin Bujna
Department of Computer Science, Paderborn University, Paderborn, Germany
kathrin.bujna@upb.de

Abstract

The fuzzy K-means problem is a popular generalization of the well-known K-means problem to soft clusterings. We present the first coresets for fuzzy K-means with size linear in the dimension, polynomial in the number of clusters, and poly-logarithmic in the number of points. We show that these coresets can be employed in the computation of a $(1 + \epsilon)$-approximation for fuzzy K-means, improving previously presented results. We further show that our coresets can be maintained in an insertion-only streaming setting, where data points arrive one-by-one.

2012 ACM Subject Classification Theory of computation → Unsupervised learning and clustering

Keywords and phrases clustering, fuzzy k-means, coresets, approximation algorithms, streaming

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2018.46

Related Version A full version of the paper is available at `https://arxiv.org/abs/1612.07516`.

Funding This work was partially supported by the German Research Foundation (DFG) under grant BL 314/8-1.

1 Introduction

Clustering is a widely used technique in unsupervised machine learning. The goal is to divide some set of objects into groups, the so-called clusters, such that objects in the same cluster are more similar to each other than to objects in other clusters. Nowadays, clustering is ubiquitous in many research areas, such as data mining, image and video analysis, information retrieval, and bioinformatics. The most common approach are hard clusterings, where the input is partitioned into a given number of clusters, i.e. each point belongs to exactly one of the clusters. The K-means problem is the most well-known hard clustering problem. It has been studied extensively from practical and theoretic points of view. However, in some applications it is beneficial to be less decisive and allow points to belong to more than one cluster. This idea leads to so-called *soft clusterings*. In the following, we study a popular soft clustering problem, the *fuzzy K-means* problem.

The fuzzy K-means objective function goes back to work by Dunn and Bezdek et al. [4, 10]. Today, it has found numerous practical applications, for example in data mining [19], image segmentation [27], and biological data analysis [9]. Practical applications generally use the fuzzy K-means algorithm, an iterative relocation scheme similar to Lloyd's algorithm [25] for

K-means, to tackle the problem. The fuzzy K-means algorithm has been proven to converge to a local minimum or a saddle point of the objective function [4, 5]. Distinguishing whether the fuzzy K-means algorithm has reached a local minimum or a saddle point is a problem which got some attention on its own [20, 24]. Moreover, it is known that the algorithm converges locally, i.e. started sufficiently close to a minimizer, the iteration sequence converges to that particular minimizer [17]. However, from a theoretician's point of view this algorithm has the major downside that stationary points of the objective function can be arbitrarily worse than a globally optimal solution [6]. Currently, the only paper on algorithms with approximation guarantees for the fuzzy K-means problem is [6], where the authors present a PTAS assuming a constant number of clusters.

Clustering is usually applied when huge amounts of data need to be processed. This has sparked significant interest in researching clustering in a streaming model, where the data does not fit into memory. A lot of research has been done on this setting for K-means. In a single pass setting, where we are only allowed to read the data set once, the K-means objective function can be approximated up to a constant factor, by choosing $\mathcal{O}(K \log(K))$ means, instead of K [1]. This has been improved to an algorithm computing exactly K means but still maintaining a constant factor approximation [7, 28]. There, the authors considered a streaming setting where points arrive one-by-one and they are allowed to use $\mathcal{O}(K \log(N))$ memory, where N is the total number of points.

The goal of a coreset is to find a small representation of a large data set, retaining the characteristics of the original data. Coresets have emerged as a key technique to tackle the streaming model. The idea is to treat the computation of the coreset as an online problem where points arrive in some kind of stream. If, after having read the whole stream, the computed coreset is small enough to fit into memory, then standard algorithms can be used to solve the problem almost optimally for the points in the stream. Usually, the algorithm does not know the size of the stream beforehand and hence, always maintains a coreset of the points seen so far.

The first coreset construction for K-means is due to Har-Peled and Mazumdar, and is of size $\mathcal{O}(\log(N))$ [16]. They also showed how to maintain a coreset, with size poly-logarithmic in N, of a data stream, by combining their notion of a coreset with the merge-and-reduce technique by Bentley and Saxe [3]. This construction was improved to a coreset with size independent of N [15]. Feldman and Langberg presented a general framework computing coresets for a large class of hard clustering problems with size independent of N [12]. Later, Feldman et al. presented coresets with size independent of N and D by using a construction based on low-rank approximation [14]. Furthermore, they generalize Har-Peled and Mazumdar's application of the merge-and-reduce technique, showing how coresets with certain properties can be maintained in a streaming setting. The results of this paper are based on Chen's sampling based construction, which yields coresets with size poly-logarithmic in N, K, and D [8]. Applying the merge-and-reduce technique, Chen's coresets can also be used to maintain a poly-logarithmic sized coreset of a data stream.

There has been some work on applying the fuzzy K-means algorithm to large data sets. Hore et al. [21] presented a single pass variant of the algorithm, which processes the data chunk-wise. This idea was refined and extended to a single pass and online kernel fuzzy K-means algorithm [18]. However, these are still variants of the fuzzy K-means algorithm, hence provide no guarantees for the quality of solutions. So far, no coreset constructions have been presented for the fuzzy K-means problem, and the literature is not rich on coreset constructions for soft clustering problems, in general. There is a construction for the problem of estimating mixtures of semi-spherical Gaussians which yields coresets with size independent of N [11]. This result was generalized to a large class of hard and soft clustering problems based on μ-similar Bregman divergences [26].

1.1 Our Result

We prove the existence of small coresets for the fuzzy K-means problem. In Section 3, we show that, by adjusting some parameters of Chen's construction [8], we obtain a coreset for the fuzzy K-means problem with size still poly-logarithmic in N. Our proof technique is a non-trivial combination of the notion of negligible fuzzy clusters [6] and weak coresets [13]. This results in a general weak-to-strong lemma (cf. Lemma 7), which states that weak coresets for the fuzzy K-means problem fulfilling certain conditions are already strong coresets. Afterwards, we argue that our adaptation of Chen's algorithm yields a weak coreset satisfying all conditions of the weak-to-strong theorem (a comprehensive proof can be found in the full version). In Section 4, we substantiate the usefulness of our result by presenting two applications of coresets for fuzzy K-means. First, we improve the analysis of a previously presented [6] PTAS for fuzzy K-means, removing the dependency on the weights of the data points from the runtime. Running this algorithm on our coreset instead of the original input improves upon the runtime of previously known $(1+\epsilon)$-approximation schemes. The improvement lies in the exponential term, which we reduce from $N^{\mathcal{O}(\mathrm{poly}(K,1/\epsilon))}$ to $\log(N)^{\mathcal{O}(\mathrm{poly}(K,1/\epsilon))}$, while maintaining non-exponential dependence on D. Second, we argue that an application of the merge-and-reduce technique enables us to maintain a fuzzy K-means coreset in a streaming model, where points arrive one-by-one.

2 Preliminaries

Let $X \subset \mathbb{R}^D$ be a set of points in D-dimensional space and $w : X \to \mathbb{N}$ be an integer weight function on the points. Using integer weights eases the notation of our exposition. We later argue how our results generalize to rational weights. Unweighted data sets are denoted by using the weight function $\mathbf{1}$ mapping every input to 1. We call $w(X) = \sum_{x\in X} w(x)$ the total weight of X and denote the maximum and minimum weights by $w_{\max}(X) = \max_{x\in X} w(x)$ and $w_{\min}(X) = \min_{x\in X} w(x)$.

▶ **Definition 1** (Fuzzy K-means)**.** Let $m \in \mathbb{R}_{>1}$ and $K \in \mathbb{N}$. The *fuzzy K-means problem* is to find a set of means $M = \{\mu_k\}_{k\in[K]} \subset \mathbb{R}^D$ and a membership function $r : X \times [K] \to [0,1]$ minimizing

$$\phi(X,w,M,r) = \sum_{x\in X} w(x) \sum_{k\in[K]} r(x,k)^m \left\|x-\mu_k\right\|^2$$

subject to

$$\forall x \in X : \sum_{k\in[K]} r(x,k) = 1 \ .$$

The parameter m is called fuzzifier. It determines the softness of an optimal clustering and is not subject to optimization, since the cost of any solution can always be decreased by increasing m. In the case $m = 1$, the cost can not be decreased by assigning membership of a point to any mean except its closest. Consequently, optimal solutions of the fuzzy K-means problem for $m = 1$ coincide with optimal solutions for the K-means problem on the same instance. Hence, in the following we always assume m to be some constant larger than 1.

Similar to the classic K-means problem, it is easy to optimize means or memberships of fuzzy K-means, assuming the other part of the solution is fixed [4]. This means, given some set of means M we call a respective optimal membership function r^*_M induced by M and set $\phi(X,w,M) := \phi(X,w,M,r^*_M)$. Analogously, given some membership function r we call a respective optimal set of means M^*_r induced by r and set $\phi(X,w,r) := \phi(X,w,M^*_r,r)$. Finally, given some optimal solution M^*, r^* we denote $\phi^{opt}(X,w) := \phi(X,w,M^*,r^*)$.

2.1 Fuzzy Clusters

Recall that, in a soft-clustering, there is no partitioning of the input points. Instead, we describe the k^{th} cluster of a fuzzy clustering as a vector of the fractions of points assigned to it by the membership function. We denote the size (or the total weight) of the k^{th} cluster by $r(X, w, k) = \sum_{x \in X} w(x) r(x, k)^m$. Given a set of means M, we denote the cost of the k^{th} cluster by $\phi_k(X, w, M, r) = \sum_{x \in X} w(x) r(x, k)^m \|x - \mu_k\|^2$.

2.2 K-Means Notation

We denote the distance of a point to a set of means M by $d(x, M) = \min_{\mu \in M}\{\|x - \mu\|\}$ and the K-means cost by $\mathrm{km}(X, w, M) = \sum_{x \in X} w(x) d(x, M)^2$. Let $C \subseteq X$ be some cluster, then $\mathrm{km}(C, w) = \sum_{x \in C} w(x) \|x - \mu_w(C)\|^2$, where $\mu_w(C) = \sum_{x \in C} w(x) x / w(C)$.

3 Coresets for Fuzzy K-Means

A coreset is a representation of a data set that preserves properties of the original data set [16]. Formally, we require the cost of a set of means with respect to the coreset to be close to the cost the same set of means incurs on the original data.

▶ **Definition 2** (Coreset)**.** Let $\epsilon \in (0, 1)$. A set $S \subset \mathbb{R}^D$ together with a weight function $w_S : S \to \mathbb{N}$ is called an ϵ-coreset of (X, w) for the fuzzy K-means problem if

$$\forall M \subset \mathbb{R}^D, |M| \leq K : \ \phi(S, w_S, M) \in [1 \pm \epsilon] \phi(X, w, M) \ , \tag{1}$$

We sometimes refer to a coreset as a *strong coreset*.

In the following, we show how to construct coresets for the fuzzy K-means problem with high probability. To this end, our proof consists of two independent steps. First, we show that it is sufficient to construct a so-called weak coreset [13] for the fuzzy K-means problem fulfilling certain properties. Second, we present an adaptation of Chen's coreset construction for K-means [8] which computes weaks coresets with the desired properties, with high probability.

▶ **Theorem 3.** *There is an algorithm that, given a set $X \subset \mathbb{R}^D$, $K \in \mathbb{N}$, $\delta \in (0, 1)$, and $\epsilon \in (0, 1)$, computes an ϵ-coreset (S, w_S), with $S \subseteq X$ and $w_S : S \to \mathbb{N}$, of $(X, \mathbf{1})$ for the fuzzy K-means problem, with probability at least $1 - \delta$, such that*

$$|S| \in \mathcal{O}\left(\log(N) \log(\log(N))^2 \epsilon^{-3} D K^{4m-1} \log(\delta^{-1})\right) \ .$$

The algorithms' runtime is $\mathcal{O}(NDK \log(\delta^{-1}) + |S|)$.

This result trivially generalizes to integer weighted data sets, by treating each point $x \in X$ as $w(x)$ copies of the same point. However, in that case we have to replace each occurrence on N in the runtime of the algorithm and the size of the coreset by $w(X)$. For rational weights, we normalize the weight function. This incurs an additional multiplicative factor of $w_{\max}(X)/w_{\min}(X)$ to each occurrence of N.

3.1 From Weak to Strong Coresets

Weak coresets are a relaxation of the previously introduced (strong) coresets. Consider a set of points together with a weight function and a set of solutions. This forms a weak coreset if the set of solutions contains a solution close to the optimum and the coreset property (1) is satisfied for all solutions from the solution set.

▶ **Definition 4** (Weak Coresets). A set $S \subset \mathbb{R}^D$ together with a weight function $w_S : S \to \mathbb{N}$ and a set of solutions $\Theta \subseteq \{\theta \mid \theta \subset \mathbb{R}^D, |\theta| \leq K\}$ is called a weak ϵ-coreset of (X, w) for the fuzzy K-means problem if

$$\exists M \in \Theta : \ \phi(S, w_S, M) \leq (1+\epsilon) \cdot \phi^{opt}(X, w) \text{ and}$$
$$\forall M \in \Theta : \ \phi(S, w_S, M) \in [1 \pm \epsilon]\phi(X, w, M) \ .$$

In contrast to the definition of weak coresets for the K-means problem [13], we consider elements M of a given set of solutions Θ instead of subsets of a set of candidate means. This is just a slight generalization which allows us to characterize solutions more precisely.

One difficulty when analysing the fuzzy K-means objective function is that, in optimal solutions, clusters are never empty. Consider a set of means, where there exists a mean which is far away from every point. In an optimal hard clustering, this mean's cluster is empty and we can safely ignore it in the analysis. For fuzzy K-means, this is not the case. In an optimal solution, every point has a non-trivial membership to this mean, thus it cannot be ignored (or removed from the solution) without increasing the cost. Bounding the cost of means with small membership mass proves to be rather difficult. A central concept we use to control the cost of such means are fuzzy clusters which are almost empty, or negligible.

▶ **Definition 5** (negligible). Let $M \subset \mathbb{R}^D$ with $|M| \leq K$. We say the k^{th} cluster of a membership function $r : X \times [|M|] \to [0, 1]$ is (K, ϵ)-negligible if

$$\forall x \in X : \ r(x, k) \leq \frac{\epsilon}{4mK^2} \ .$$

In the following, we omit the parameters (K, ϵ) if they are clear from context.

We cannot preclude the possibility that an optimal fuzzy K-means clustering contains a negligible cluster. However, we can circumvent negligible clusters altogether, by observing that we can remove a mean inducing a negligible cluster without increasing the cost significantly.

▶ **Theorem 6** ([6]). *Let $M \subset \mathbb{R}^D$ with $|M| \leq K$ and $\epsilon \in (0, 1)$. There exists a set of means $M' \subseteq M$ with*

$$\phi(X, w, M') \leq (1+\epsilon)\phi(X, w, M) \ ,$$

such that the optimal membership function with respect to M' contain no negligible clusters.

Given some set of means, the optimal memberships of a point depend only on the location of the point relative to the means and not on its weight or any other points in the data set [4]. This means that negligible clusters are, in some sense, transitive. That is: If a cluster induced by some set of means is negligible, then it is also negligible with respect to any subset of X and the same set of means. Using this observation we can prove our key weak-to-strong result.

▶ **Lemma 7** (weak-to-strong). *Let $\epsilon \in (0, 1)$ and*

$$\Theta_{(K,\epsilon)}(X) := \left\{ M \subset \mathbb{R}^D \,\middle|\, \begin{array}{c} |M| \leq K \text{ and } M \text{ induces no negligible} \\ \text{cluster with respect to } X \end{array} \right\} \ .$$

If $S \subseteq X$ and $w_S : S \to \mathbb{N}$, such that $(S, w_S, \Theta_{(K,\epsilon)}(X))$ is weak ϵ-coreset of (X, w) for the fuzzy K-means problem, then (S, w_S) is a strong (3ϵ)-coreset of (X, w) for the fuzzy K-means problem.

Proof. We need to verify that the coreset property (1) holds for all solutions $M \subset \mathbb{R}^D$ with $|M| \leq K$. Since $(S, w_S, \Theta_{(K,\epsilon)}(X))$ is a weak ϵ-coreset we only have to show this for all $M \notin \Theta_{(K,\epsilon)}(X)$. From Theorem 6, we know that there exists $M' \in \Theta_{(K,\epsilon)}(X)$, $M' \subseteq M$ with $\phi(X, w, M') \leq (1+\epsilon)\phi(X, w, M)$.

We obtain the upper bound by observing that

$$\begin{aligned}
\phi(S, w_S, M) &\leq \phi(S, w_S, M') && (M' \subseteq M)\\
&\leq (1+\epsilon)\phi(X, w, M') && (\text{weak coreset property})\\
&\leq (1+\epsilon)^2\phi(X, w, M) && (\text{choice of } M')\\
&\leq (1+3\epsilon)\phi(X, w, M)\ . && (\epsilon \in (0,1))
\end{aligned}$$

The lower bound is slightly more involved. Again, from Theorem 6, we obtain that there exists $M'_S \in \Theta_{(K,\epsilon)}(S)$, $M'_S \subseteq M$ with $\phi(S, w_S, M'_S) \leq (1+\epsilon)\phi(S, w_S, M)$. Recall that, for each point, the membership induced by some set of means only depends on the point itself and the given set of means. In particular, this membership does not depend on the weight of the point, nor on other data points. Hence, if there is no point in X such that the induced membership with respect to some mean $\mu_k \in M$ is larger than some constant, then there is no point in $S \subseteq X$, such that the induced membership to $\mu_k \in M$ is larger than this constant. Since $M' \in \Theta_{(K,\epsilon)}(X)$, it holds that all means in $M \setminus M'$ induce negligible clusters on S and thus $M'_S \subseteq M'$. We conclude

$$\begin{aligned}
\phi(S, w_S, M) &\geq \frac{1}{1+\epsilon}\phi(S, w_S, M'_S) && (\text{choice of } M'_S)\\
&\geq \frac{1}{1+\epsilon}\phi(S, w_S, M') && (M'_S \subseteq M')\\
&\geq \frac{1-\epsilon}{1+\epsilon}\phi(X, w, M') && (\text{weak coreset property})\\
&\geq \frac{1-\epsilon}{1+\epsilon}\phi(X, w, M) && (M' \subseteq M)\\
&\geq (1-3\epsilon)\phi(X, w, M)\ . && (\epsilon \geq 0)
\end{aligned}$$

◀

3.2 Weak Coresets for Solutions with Non-Negligible Clusters

In the following, we explain how to adapt Chen's coreset construction for the K-means problem [8] to construct a set $S \subseteq X$ and weight function $w_S : S \to \mathbb{N}$ such that $(S, w_S, \Theta_{(K,\epsilon)}(X))$ is a weak ϵ-coreset of $(X, \mathbf{1})$ for the fuzzy K-means problem. Applying Lemma 7 to this construction yields Theorem 3. We give a high-level description of Chen's algorithm. In the first step, we compute an (α, β)-bicriteria approximation of the K-means problem with respect to X, i.e. a set M approximating an optimal K-means solution within factor α and with $|M| \leq \beta K$, such that $\alpha, \beta \in \mathcal{O}(1)$.

In the second step, the input points are partitioned based on concentric balls around the means of the bicriteria approximation with exponentially increasing radii. By $X_{i,j}$ we denote the intersection of X with the j^{th} annulus around the i^{th} mean. Then, we sample points from each $X_{i,j}$ uniformly and independently at random. Finally, each point sampled from $X_{i,j}$ is evenly weighted, such that the sum of these weights is equal to the number of original data points in $X_{i,j}$. These sampled points together with the weights form the coreset.

There is no natural adaptation of the first step to fuzzy K-means since, so far, there exists no bicriteria approximation algorithm for the fuzzy K-means problem with constant α and β. However, we know that the K-means cost of all sets of means M is no larger than $|M|^{m-1}$ times the fuzzy K-means cost of M [6]. Hence, an (α, β)-bicriteria approximation for the K-means problem is an $(\alpha \cdot (\beta K)^{m-1}, \beta)$-bicriteria approximation for the fuzzy K-means problem on the same instance. We can counteract this very coarse bound on the cost in the second step by sampling roughly a factor of $K^{\mathcal{O}(m)}$ more points than the original algorithm.

▶ **Lemma 8.** *The algorithm described in the previous paragraph computes $S \subseteq X$ and $w_S : S \to \mathbb{N}$ such that $(S, w_S, \Theta_{(K,\epsilon)}(X))$ is a weak ϵ-coreset of $(X, \mathbf{1})$ for the fuzzy K-means problem, with high probability.*

Proof Sketch. Let $M \in \Theta_{(K,\epsilon)}(X)$ be a set of means inducing no negligible clusters. We consider large balls around each mean of the bicriteria-approximation. As in Chen's original proof, we establish the coreset property for the case where at least one mean of a given solution is outside of these balls and the case where all means are contained in the union of these balls.

For the first case, assume that M contains at least one mean, say μ_k, outside of (sufficiently large) balls around the means of the bicriteria approximation. Since μ_k has a non negligible portion of the membership of at least one point from which it is far away, we can bound the cost of M from below. This lower bound is significantly larger than the distances of data points to their respective representative in the coreset. Using this, we can easily verify the coreset property with respect to M.

For the second case, assume that all means of M lie in the union of these balls. In this case, we do not need to use that clusters induce non-negligible memberships. Instead, we can basically follow the arguments of Chen's original proof. However, the cost estimations are more technically involved due to the difficult structure of the fuzzy K-means objective function. A detailed exposition of our proof can be found in the full version.

The size of the coreset and the runtime of the algorithm are as claimed in Theorem 3. ◀

4 Applications

In the following, we present two applications of our coresets for fuzzy K-means. In general, our coresets can be plugged in before any application of an algorithm that tries to solve fuzzy K-means and can handle weighted data sets. If the applied algorithm's runtime does not depend on the actual weights, then this leads to a significant reduction in runtime. We show that this yields a faster PTAS for fuzzy K-means than the ones presented before [6]. Furthermore, we argue that our coresets can be maintained in an insertion-only streaming setting.

4.1 Speeding up Aproximation

We start by presenting an improved analysis of a simple sampling-based PTAS for the fuzzy K-means problem. Our analysis exploits that the algorithm can ignore the weights of the data points and still obtain an approximation guarantee of $(1 + \epsilon)$ for the weighted problem. This means, that the algorithm's runtime is independent of the weights, and thus can be significantly reduced by applying it to a coreset instead of the original data. The first ingredient is the following, previosuly presented, soft-to-hard lemma.

Algorithm 1: Derandomized Sampling.

Input: $X \subset \mathbb{R}^D$, $K \in \mathbb{N}$, $\epsilon \in (0,1)$

1 $\mathcal{T} \leftarrow \{\mu_{\mathbf{1}}(S) \mid S \subseteq X, |S| = \frac{64K}{\epsilon}\}$

/* S as multisets - Points can occur multiple times in each S and are counted with multiplicity. */

2 $M \leftarrow \arg\min_{T \subseteq \mathcal{T}, |T|=K}\{\phi(X, w, T)\}$

3 **return** M

▶ **Lemma 9** ([6]). *Let $\epsilon \in (0,1)$, $r : X \times [K] \to [0,1]$ be a membership function and let M_r^* be a set of means induced by r.*

If $\forall k \in [K] : r(X,w,k) \geq 16Kw_{\max}(X)/\epsilon$, then there exist pairwise disjoint sets $C_1, \ldots, C_K \subseteq X$ such that for all $k \in [K]$

$$w(C_k) \geq \frac{r(X,w,k)}{2} ,$$
$$\|\mu_w(C_k) - \mu_k\|^2 \leq \frac{\epsilon}{r(X,w,k)}\phi_k(X,w,M_r^*,r), \text{ and}$$
$$\mathrm{km}(C_k) \leq 4K \cdot \phi_k(X,w,M_r^*,r) .$$

We combine this with a classic concentration bound by Inaba et al.

▶ **Lemma 10** ([22]). *Let $P \subset \mathbb{R}^D$, $n \in \mathbb{N}$, $\delta \in (0,1)$, and let S be a set of n points drawn uniformly at random from P. Then we have*

$$\Pr\left(\|\mu_{\mathbf{1}}(S) - \mu_{\mathbf{1}}(P)\|^2 \leq \frac{1}{\delta n}\frac{\mathrm{km}(P,\mathbf{1})}{|P|}\right) \geq 1 - \delta .$$

▶ **Corollary 11.** *Let $X \subset \mathbb{R}^D$, $w : X \to \mathbb{N}$, $K \in \mathbb{N}$, $\epsilon \in (0,1)$, and let $C_1, \ldots, C_K \subseteq X$ be non-empty subsets of X. There exist K multisets $S_1, \ldots, S_K \subseteq X$, such that*

$$\forall k \in [K] : |S_k| = \frac{2}{\epsilon} \text{ and } \|\mu_{\mathbf{1}}(S_k) - \mu_w(C_k)\|^2 \leq \epsilon\frac{\mathrm{km}(C_k,w)}{w(C_k)} .$$

We can find means of subsets obtained from applying the soft-to-hard lemma to the clusters of an optimal fuzzy K-means solution by derandomizing Inaba's sampling technique.

▶ **Theorem 12.** *Algorithm 1 computes $M \subset \mathbb{R}^D$ with $|M| = K$, such that*

$$\phi(X,w,M) \leq (1+\epsilon)\phi^{opt}(X,w)$$

in time $DN^{\mathcal{O}(K^2/\epsilon)}$.

Proof. We analyse the result M of Algorithm 1. Let M^*, r^* be an optimal solution to the fuzzy K-means problem on X, w. Let X_c be a modified point set, which contains c copies of every point $x \in X$, where

$$c = \left\lceil \frac{\gamma K w_{\max}(X)}{\epsilon \min_{k\in[K]} r^*(X,w,k)} \right\rceil ,$$

for some large enough constant γ. For all sets of means M and all membership functions r, we have $\phi(X_c, w, M, r) = c \cdot \phi(X, w, M, r)$. Thus, M^* and r^* (where $r^*(y,k) = r^*(x,k)$ for

all $k \in [K]$ and $x \in X, y \in X_c$ with $x = y$) are also optimal for the modified instance X_c. Observe, that for all $k \in [K]$ we have

$$r^*(X_c, w, k) \geq \sum_{x \in X} \frac{\gamma K w_{\max}(X)}{\epsilon \min_{k \in [K]} r^*(X, w, k)} w(x) r^*(x, k)^m \geq \frac{\gamma K w_{\max}(X)}{\epsilon} \geq \frac{64 K w_{\max}(X)}{\epsilon} .$$

Observe, that M^* is a set of means induced by r^*. Hence, by applying Lemma 9 with respect to X_c, w, r^*, and $\epsilon/4$ we obtain that there exist disjoint sets $C_1, \ldots, C_K \subseteq X_c$ such that for all $k \in [K]$ we have

$$w(C_k) \geq \frac{r^*(X_c, w, k)}{2} , \tag{2}$$

$$\|\mu_w(C_k) - \mu_k^*\|^2 \leq \frac{\epsilon}{4 r^*(X_c, w, k)} \phi_k(X_c, w, M^*, r^*) \text{ , and} \tag{3}$$

$$\mathrm{km}(C_k, w) \leq 4K \cdot \phi_k(X_c, w, M^*, r^*) . \tag{4}$$

Next, we apply Corollary 11 to X_c, w, K, $\epsilon/(32K)$, and $C_1, \ldots, C_K$. We obtain that there exist $S_1, \ldots, S_K \subseteq X_c$ such that for all $k \in [K]$ we have $|S_k| = 64K/\epsilon$ and

$$\|\mu_{\mathbf{1}}(S_k) - \mu_w(C_k)\|^2 \leq \epsilon/(32K) \, \mathrm{km}(C_k, w)/w(C_k) . \tag{5}$$

Since X_c consists of copies of points from X, we conclude that $S_1, \ldots, S_K \subseteq X$, if we treat the S_k as multisets, i.e. allow the same point to appear multiple times in the same set. Hence, by choice of M, as made by Algorithm 1, we have $\phi(X, w, M) \leq \phi(X, w, \{\mu_{\mathbf{1}}(S_k)\}_{k \in [K]})$. Plugging all this together, we can bound the cost of M as follows

$$\begin{aligned}
\phi(X, w, M) &\leq \phi(X, w, \{\mu_{\mathbf{1}}(S_k)\}_{k \in [K]}) = \frac{1}{c} \phi(X_c, w, \{\mu_{\mathbf{1}}(S_k)\}_{k \in [K]}) \\
&\leq \frac{1}{c} \phi(X_c, w, \{\mu_{\mathbf{1}}(S_k)\}_{k \in [K]}, r^*) = \frac{1}{c} \sum_{x \in X_c} \sum_{k \in [K]} w(x) r^*(x, k)^m \|x - \mu_{\mathbf{1}}(S_k)\|^2 \\
&\leq \phi(X, w, r^*) + \frac{2}{c} \sum_{x \in X_c} \sum_{k \in [K]} w(x) r^*(x, k)^m \|\mu_k^* - \mu_w(C_k)\|^2 \\
&\qquad + \frac{2}{c} \sum_{x \in X_c} \sum_{k \in [K]} w(x) r^*(x, k)^m \|\mu_w(C_k) - \mu_{\mathbf{1}}(S_k)\|^2 \\
&\qquad\qquad \text{(by 2-approximate triangle inequality)} \\
&\leq \phi^{opt}(X, w) + \frac{\epsilon}{2c} \sum_{k \in [K]} \phi_k(X_c, w, M^*, r^*) \qquad \text{(by (3))} \\
&\qquad + \frac{\epsilon}{c16K} \sum_{k \in [K]} \frac{\mathrm{km}(C_k, w)}{w(C_k)} \sum_{x \in X_c} w(x) r^*(x, k)^m \qquad \text{(by (5))} \\
&\leq (1 + \epsilon/2) \phi^{opt}(X, w) + \frac{\epsilon}{2c} \sum_{k \in [K]} \phi_k(X_c, w, M^*, r^*) \qquad \text{(by (2) and (4))} \\
&= (1 + \epsilon) \phi^{opt}(X, w) .
\end{aligned}$$

Bounding the runtime of Algorithm 1 is straightforward. We have to evaluate the cost of $|\mathcal{T}|^K$ different fuzzy K-means solution, each evaluation costing $\mathcal{O}(NDK)$. Hence, the total runtime is bounded by $\mathcal{O}(NDK\,|\mathcal{T}|^K) = \mathcal{O}(NDK(N^{64K/\epsilon})^K) = DN^{\mathcal{O}(K^2/\epsilon)}$. ◀

Recall, that the runtime of Algorithm 1 is independent of point weights. Hence, we obtain a more efficient algorithm by first computing a coreset using Theorem 3 and then applying Algorithm 1 to this coreset instead of the original data set. In the following, we formally only state an unweighted version of our result.

▶ **Corollary 13.** *There exists an algorithm which, given $X \subset \mathbb{R}^D$, $K \in \mathbb{N}$, and $\epsilon \in (0,1)$, computes a set $M \subset \mathbb{R}^D$ with $|M| = K$, such that with constant probability*

$$\phi(X, \mathbf{1}, M) \leq (1+\epsilon)\phi^{opt}(X, \mathbf{1})$$

in time $\mathcal{O}(NDK) + (\log(N)D)^{\mathcal{O}(K^2/\epsilon \log(K/\epsilon))}$.

Proof. Given X, K, and ϵ, apply Theorem 3 (with $\epsilon/3$) to obtain, with constant probability, an $\epsilon/3$-coreset (S, w_S) of $(X, \mathbf{1})$. Let M be the output of Algorithm 1 given S, w_S, and $\epsilon/3$ and let M_X^* be an optimal set of means with respect to X. We obtain

$$\begin{aligned}\phi(S, w_S, M) &\leq (1+\epsilon/3)\phi^{opt}(S, w_S) \leq (1+\epsilon/3)\phi(S, w_S, M_X^*) \\ &\leq (1+\epsilon/3)^2\phi^{opt}(X, \mathbf{1}) \leq (1+\epsilon)\phi^{opt}(X, \mathbf{1}) \ .\end{aligned}$$

The overall runtime is $\mathcal{O}(NDK) + D(|S|)^{\mathcal{O}(K^2/\epsilon)} = \mathcal{O}(NDK) + (\log(N)D)^{\mathcal{O}(K^2/\epsilon \log(K/\epsilon))}$. ◀

The algorithm from Corollary 13 can also be applied to weighted data sets. However, its runtime is not independent of these weights. We argued that the runtime of the PTAS from Theorem 12 is independent of any weights, but this is not true for the coreset construction. Hence, weight functions have an impact on the runtime as discussed in Section 3 in regard to the coreset construction.

Nonetheless, our algorithm has significant advantages over previously presented $(1+\epsilon)$-approximation algorithms for fuzzy K-means. The runtimes of all algorithms presented in [6] have an exponential dependency on the dimension D or contain a term $N^{\mathcal{O}(\text{poly}(K, 1/\epsilon))}$. Our result constitutes the first algorithm with a non-exponential dependence on D whose only exponential term is of the form $\log(N)^{\mathcal{O}(\text{poly}(K, 1/\epsilon))}$.

Strictly speaking, applying Algorithm 1 directly to X is faster if $D \in \Omega(N)$. However, in that case we can apply the lemma of Johnson and Lindenstrauss [23] to replace D by $\log(N)/\epsilon^2$

4.2 Streaming Model

We give a brief overview of the method to maintain coresets in a streaming model presented in [14]. It is an improved version of the techniques previously used by [8] and [16]. The central observation is that the union of coresets of two input data sets is a coreset of the union of the data sets. Whenever a sufficient (depending on the coreset construction) number of points has arrived in the stream, we compute a coreset of these points. After two coresets have been computed, we merge them into a larger coreset of all points that have arrived, so far. Following two of these merge operations, we merge the two larger coresets into one even larger one. This continues in the fashion of a binary tree. Since our coresets for fuzzy K-means fulfil all requirements to apply this approach, it can also be used to maintain fuzzy K-means coresets in the streaming model.

▶ **Theorem 14.** *Given N data points in a stream (one-by-one) and $\epsilon \in (0,1)$ one can maintain, with high probability, an ϵ-coreset for the fuzzy K-means problem, of the points seen so far, using $\mathcal{O}(DK^{4m-1} \cdot \text{polylog}(N/\epsilon))$ memory. Arriving data points cause an update with an amortized runtime of $\mathcal{O}(DK \cdot \text{polylog}(NDK/\epsilon))$.*

5 Discussion and Outlook

We proved that a parameter tuned version of Chen's construction yields the first coresets for the fuzzy K-means problem. While there are a plethora of coreset constructions for K-means, Chen's construction is the best purely sampling based approach. More efficient techniques, for example ϵ-nets [15] or subspace approaches like low-rank approximation [14], heavily rely on the partitioning of the input set that a K-means solution induces. So far, we have not found a way to apply these to the, already notoriously hard to analyse, fuzzy K-means objective function. This is because the membership function essentially introduces an unknown weighting on the points. Hence, when the data set is partitioned or projected into some subspace without respecting this weighting, we introduce a factor $K^{\mathcal{O}(1)}$ to the cost estimation. It has proven difficult to control these additional factors. Partly for these reasons, there is still a large number of open questions regarding fuzzy K-means.

In this paper, we almost match the asymptotic runtime of the fastest $(1+\epsilon)$-approximation algorithms for K-means. However, even assuming constant K, our algorithms lack practicality due to the large constants hidden in the $\mathcal{O}$. Hence, this raises interesting follow-up questions. Is there an efficient approximation algorithm for fuzzy K-means with a constant approximation factor? What can be done in terms of bicriteria algorithms, i.e. if we are allowed to chose more than K means? In regard to the complexity of fuzzy K-means it is interesting to examine whether one can show that there is no true PTAS (polynomial runtime in N, D, and K) for fuzzy K-means, as it was shown for K-means [2]. Finally, can we relate the hardness of fuzzy K-means directly to K-means?

References

1 N. Ailon, R. Jaiswal, and C. Monteleoni. Streaming k-means approximation. In *Advances in Neural Information Processing Systems 22*, pages 10–18. Curran Associates, Inc., 2009.

2 P. Awasthi, M. Charikar, R. Krishnaswamy, and A. K. Sinop. The Hardness of Approximation of Euclidean k-Means. In *31st International Symposium on Computational Geometry*, pages 754–767, 2015. doi:10.4230/LIPIcs.SOCG.2015.754.

3 J. L. Bentley and J. B. Saxe. Decomposable searching problems I. Static-to-dynamic transformation. *Journal of Algorithms*, 1(4):301–358, 1980.

4 J. C. Bezdek, R. Ehrlich, and W. Full. FCM: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, 10(2):191–203, 1984.

5 J. C. Bezdek, R. J. Hathaway, M. J. Sabin, and W. T. Tucker. Convergence theory for fuzzy c-means: Counterexamples and repairs. *IEEE Transactions on Systems, Man and Cybernetics*, 17(5):873–877, 1987. doi:10.1109/TSMC.1987.6499296.

6 J. Blömer, S. Brauer, and K. Bujna. A Theoretical Analysis of the Fuzzy K-Means Problem. In *2016 IEEE 16th International Conference on Data Mining*, pages 805–810, 2016. doi:10.1109/ICDM.2016.0094.

7 V. Braverman, A. Meyerson, R. Ostrovsky, A. Roytman, M. Shindler, and B. Tagiku. Streaming K-means on Well-clusterable Data. In *Proceedings of the Twenty-second Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 26–40, 2011.

8 K. Chen. On Coresets for K-Median and K-Means Clustering in Metric and Euclidean Spaces and Their Applications. *SIAM Journal on Computing*, 39(3):923–947, 2009. doi:10.1137/070699007.

9 D. Dembélé and P. Kastner. Fuzzy C-means method for clustering microarray data. *Bioinformatics*, 19(8):973–980, 2003. doi:10.1093/bioinformatics/btg119.

10 J. C. Dunn. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics*, 3(3):32–57, 1973. doi:10.1080/01969727308546046.

11 D. Feldman, M. Faulkner, and A. Krause. Scalable Training of Mixture Models via Coresets. In *Advances in Neural Information Processing Systems 24*, pages 2142–2150. Curran Associates, Inc., 2011.

12 D. Feldman and M. Langberg. A Unified Framework for Approximating and Clustering Data. In *Proceedings of the Forty-third Annual ACM Symposium on Theory of Computing*, pages 569–578, 2011. doi:10.1145/1993636.1993712.

13 D. Feldman, M. Monemizadeh, and C. Sohler. A PTAS for K-means Clustering Based on Weak Coresets. In *Proceedings of the Twenty-third Annual Symposium on Computational Geometry*, pages 11–18, 2007. doi:10.1145/1247069.1247072.

14 D. Feldman, M. Schmidt, and C. Sohler. Turning Big Data into Tiny Data: Constant-size Coresets for K-means, PCA and Projective Clustering. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1434–1453, 2013.

15 S. Har-Peled and A. Kushal. Smaller Coresets for K-median and K-means Clustering. In *Proceedings of the Twenty-first Annual Symposium on Computational Geometry*, pages 126–134, 2005. doi:10.1007/s00454-006-1271-x.

16 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. doi:10.1145/1007352.1007400.

17 R. J. Hathaway and J. C. Bezdek. Local convergence of the fuzzy c-Means algorithms. *Pattern Recognition*, 19(6):477–480, 1986.

18 T. C. Havens, J. C. Bezdek, C. Leckie, L. O. Hall, and M. Palaniswami. Fuzzy c-Means Algorithms for Very Large Data. *IEEE Transactions on Fuzzy Systems*, 20(6):1130–1146, 2012. doi:10.1109/TFUZZ.2012.2201485.

19 K. Hirota and W. Pedrycz. Fuzzy computing for data mining. *Proceedings of the IEEE*, 87(9):1575–1600, 1999. doi:10.1109/5.784240.

20 F. Hoppner and F. Klawonn. A contribution to convergence theory of fuzzy c-means and derivatives. *IEEE Transactions on Fuzzy Systems*, 11(5):682–694, 2003. doi:10.1109/TFUZZ.2003.817858.

21 P. Hore, L. O. Hall, and D. B. Goldgof. Single Pass Fuzzy C Means. In *2007 IEEE International Fuzzy Systems Conference*, pages 1–7, 2007. doi:10.1109/FUZZY.2007.4295372.

22 M. Inaba, N. Katoh, and H. Imai. Applications of Weighted Voronoi Diagrams and Randomization to Variance-based K-clustering. In *Proceedings of the Tenth Annual Symposium on Computational Geometry*, pages 332–339, 1994. doi:10.1145/177424.178042.

23 W. B. Johnson and J. Lindenstrauss. Extensions of Lipschitz maps into a Hilbert space. *Contemporary Mathematics*, 26, 1984.

24 T. Kim, J. C. Bezdek, and R. J. Hathaway. Optimality tests for fixed points of the fuzzy c-means algorithm. *Pattern Recognition*, 21(6):651–663, 1988. doi:10.1016/0031-3203(88)90037-4.

25 S. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982. doi:10.1109/tit.1982.1056489.

26 Mario Lucic, Olivier Bachem, and Andreas Krause. Strong Coresets for Hard and Soft Bregman Clustering with Applications to Exponential Family Mixtures. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 1–9, 2016.

27 M. R. Rezaee, P. M. J. van der Zwet, B. P. F. Lelieveldt, R. J. van der Geest, and J. H. C. Reiber. A multiresolution image segmentation technique based on pyramidal segmentation and fuzzy clustering. *IEEE Transactions on Image Processing*, 9(7):1238–1248, 2000. doi:10.1109/83.847836.

28 M. Shindler, A. Wong, and A. Meyerson. Fast and accurate k-means for large datasets. In *Advances in neural information processing systems*, pages 2375–2383, 2011.