ECAI 2024
U. Endriss et al. (Eds.)

doi:10.3233/FAIA240701

Capacitated Online Clustering Algorithm

Shivam Gupta[a,*], **Shweta Jain**[a], **Narayanan C Krishnan**[b], **Ganesh Ghalme**[c] **and Nandyala Hemachandra**[d]

[a]IIT Ropar, India; [b]IIT Palakkad, India; [c]IIT Hyderabad, India; [d]IIT Bombay, India

Abstract. Clustering is a widely used unsupervised learning tool with applications in numerous real-world problems. Traditional clustering methods can result in highly skewed clusters where one cluster is notably larger than others, rendering them unsuitable for scenarios such as logistics and routing. In response, capacitated clustering approaches have emerged over the past decade. These approaches limit the number of data points each cluster can accommodate, thus resulting in more uniform cluster formations. In an online version of capacitated clustering, the algorithm must make an irrevocable decision for each incoming data point, determining whether to establish it as a new center or allocate it to existing centers. The goal is to minimize the count of opened centers while adhering to capacity constraints and achieving a satisfactory approximation of the clustering cost compared to the optimal solution. Although exploring online capacitated clustering remains uncharted, we are the first to propose a probabilistic Capacitated Online Clustering Algorithm (called COCA) for h-dimensional euclidean spaces. We theoretically bound the number of centers opened and provide constant cost approximation guarantees. Additionally, we conduct rigorous experiments to validate the computational efficacy of the proposed approaches.
Keywords: Unsupervised Learning, Capacitated Clustering, Online Algorithm, Doubling Trick, Coupon Collector Problem

1 Introduction

Clustering is a widely used tool in data mining and finds practical application in many real-world scenarios, including, but not limited to, automatic resume screening, detecting fraudulent claims, and targeted advertisements [45]. The past decade has witnessed various notable clustering methods such as k-means, k-medoid, k-median, and k-center [36, 30]. The fundamental principle that underlies these methods is partitioning the data points into k distinct groups (called clusters) such that data points within the same cluster are more similar than others. The similarity measurement involves different distance metrics, with each cluster represented by a center. The objective of cluster formation varies across these methods; for example, k-means seeks to minimize the sum of square distances (ℓ_2 norm) between the data points and their respective centers [1]. In contrast, k-median and k-center minimize the sum of absolute distances (ℓ_1 norm) and the maximum distance within a cluster, respectively [22].

However, these traditional methods do not impose restrictions on the sizes of the clusters, leading to clusters with arbitrary sizes. This lack of constraint can result in highly skewed clusters, where one cluster is significantly larger than the others with small sizes, thus hampering their applicability to real-world problems. For example, in logistics distribution (stores/garbage) or workforce team formulation, capacity constraints are defined by the number of customers (or employees) an individual salesperson (or manager) can serve. This poses management and productivity challenges [43, 23, 42]. To address the need for more uniform cluster sizes, researchers have delved into clustering with size constraints i.e., 'capacitated clustering' [15].

The capacitated clustering algorithms can be categorized based on data access and applications, dividing them into offline, streaming, and online environments. In the offline environment, all the data points are known in advance and are available in memory. This model provides the most flexibility in terms of data availability and finds application in fields such as group team formations, student project teams, facility location, and employee allocation [12, 46]. However, the scalability of these offline solutions is constrained by the size of the main memory. In contrast, streaming environments divide data into chunks that can easily fit into the memory. The performance of such algorithms is compared based on the number of passes performed over the complete data points [37]. Existing state-of-the-art (SOTA) approaches in capacitated streaming are reviewed in [49, 47].

A more stringent variation of these environments is online clustering (OC), where an endless stream of data points arrives over time. Due to limited memory, the algorithm must make an irrevocable decision about incorporating an incoming data point into existing clusters or opening it as a new center. Once a data point becomes a center, it remains so forever. Similarly, any data point previously seen cannot be chosen as the center when a new data point arrives [11, 13, 18, 20, 35, 6]. An important aspect to note in OC pertains to the absence of information regarding the ordering of arrival of points in the stream. As a result, the algorithm ends up opening more number of centers ($k_{\texttt{actual}}$) than the desired target ($k_{\texttt{target}}$), i.e., $k_{\texttt{actual}} \geq k_{\texttt{target}}$ to maintain good approximation guarantees on objective cost. In online capacitated clustering (OCC), all these constraints are imposed while adhering to a given capacity requirements. Note that $k_{\texttt{target}}$ and k are used interchangeably for ease of reading.

To understand the need for OCC, consider the dynamic landscape of wholesale distribution networks. In this scenario, retailers employ salespersons who navigate cities to promote products, offer discounts, and build relationships with consumers [43]. To enhance consumer retention, it becomes imperative to provide specialized salespersons. Efficient market coverage is achieved by clustering consumers (shopkeepers and direct customers) based on various features such as product consumption, order volume, and location [43]. The resulting clusters group similar consumers together for personalized marketing. However, a crucial limitation arises in the form of workload constraints, with each salesperson having a maximum

* Corresponding Author. Email: shivam.20csz0004@iitrpr.ac.in. Orcid ID(s) in author ordering: 0000-0003-0633-3917, 0000-0002-2666-9058, 0000-0002-6132-0310, 0000-0001-5049-4764, 0000-0003-2917-1551 respectively.
[1] referred to as objective or clustering cost interchangeably in literature.

capacity to maintain a healthy work-life balance and also offer quality service. Furthermore, the continuous influx of new consumers in a growing market makes handling such a vast network challenging. Traditional offline solutions face computational hurdles in adapting to these changes, emphasizing the need for online solutions. Our example acknowledges that decisions made in online solutions, such as salesperson and consumer allocations, are irrevocable. This permanence is vital as salespersons develop trust and liaisons with consumers over time, and making changes to assignments is impractical and potentially detrimental to established relationships.

This necessitates investigating online clustering solutions that incorporate capacity constraints. Given that this is a comparatively challenging and hard problem [23], only a few works are available in one [16, 14] and two-dimensional [17] space. These works specifically address the k-center objective and exploit the geometrical structural properties of one, two-dimensional spaces to devise deterministic algorithms. However, they are not directly extendible to higher dimensional spaces and alternative objective functions such as k-means or k-median, which focus on minimizing the distance between each data point assignment and its center. To this, we propose a probabilistic approach that handles capacity constraints and works well for any h-dimensional euclidean spaces similar to [35, 6] available in uncapacitated OC. This paper addresses the problem of minimizing the clustering objective while satisfying the capacity constraints in an online setting [2]. The challenge arises when there is an upper limit on the number of cluster centers that can be opened; either many data points are assigned to a single (or a few) cluster(s), resulting in skewed clustering and a violation of capacity constraints, or an inefficient assignment, leading to high objective costs. Our proposed algorithm (`COCA`), addresses this problem by randomized assignments. After a certain initial number of centers are created, with probability $1 - p_t$ each incoming data point is assigned to the closest available center with remaining capacity (see Algorithm 2) and with probability p_t is designated as a new center. Next, we summarize our **contributions**.

1. With careful choice of p_t, we establish an upper bound on the number of centers opened by `COCA`, that matches with that of the uncapacitated setting.

2. We provide a constant approximation guarantee for the objective cost compared to optimal offline capacitated clustering. These guarantee enhances existing bounds in an uncapacitated setting [35] by a logarithmic factor.

3. We estimate the challenging, a-priori unknown total number of data points using the doubling trick.

4. We establish an interesting connection between our framework and a well-known coupon collector problem to determine the initial number of centers to be opened.

5. Empirical evaluations demonstrate the comparable performance of `COCA` with existing `SOTA` in uncapacitated OC on variety of datasets.

Organization: Section 2 reviews the literature. Section 3 outlines the preliminaries needed for the paper. Section 4 and 5 present the proposed algorithms with Section 6 evaluating their efficacy experimentally. Finally, Section 7 concludes with potential future directions.

2 Related Work

The existing clustering literature encompasses various methods, ranging from hierarchical to centroid-based. This work focuses on centroid-based clustering due to its computational efficiency, scalability[3], and interpretability[4]. We now review various `SOTA` approaches to approximate the capacitated clustering problem (CCP) available in different environments based on data access and applicability.

Offline CCP: The first attempt in offline CCP is by [41]. The authors proposed a heuristic that modifies uncapacitated clustering by validating capacity constraints before assignment. Building on this work, [38] extended the heuristic method. Later, [7] proposed a more effective and exact solution using cutting plane algorithms. The complete list of offline capacitated clustering approaches is available in [24, 44]. Notably, the best approximation factor for the k-means/median objective in euclidean spaces is $(1 + \epsilon)$ [12] where $\epsilon > 0$, and for the k-center method, it is two [25].

Streaming CCP: An initial attempt to achieve uniform (almost equal-sized) clustering is by [5]. The algorithm they propose requires three passes over the data stream. Later, [21] improves the work and proposes a single-pass algorithm. However, their algorithm is not directly applicable to the online environment as it involves generating coresets first and then obtaining the final assignment. In contrast, in the online environment, decisions must be made as soon as the data point arrives.

Online CCP: Recent investigations into k-center problem have explored the one-dimensional case [16, 14]. In these works, each cluster is a closed interval with no restriction on the cluster's diameter. Whenever a data point falls in a specific interval, that interval opens as a cluster for future data points. The goal is to minimize the sum of the diameter of clusters while accommodating all data points. Extending this concept to two-dimensional space involves replacing intervals with squares [17]. The algorithm initiates a new cluster whenever a point falls within an unopened square-grid cell, and the goal is to reduce the sum of the area of the opened clusters. However, the study of k-means or k-median objective, especially in higher dimensions, remains an open problem, a concern we tackle in this paper.

Deep and Contrastive clustering: Deep clustering methods require model training with data before responding to online queries [28, 29, 48]. In contrast, our setting is much stricter, and mini-batches of samples for training may not be available. Works that employ contrastive clustering also face similar limitations [33, 34].

Other works: Recent studies have revealed that the clusters stemming from the above algorithms may not exhibit a sufficient representation of different protected groups (say gender) within each cluster. An attempt to tackle such demographic bias in offline CCP is by [32]. The authors impose additional constraints using the concept of `Balance` which requires each protected group value to have approximately equal representation in every cluster [10]. Similarly, works in student topic grouping problems devise knapsack-based reduction or fair coresets to achieve the maximum possible `Balance`. However, note that CCP is a more generalized framework where the goal is to constrain the total number of data points (n) that can be assigned to each cluster (say c), which may not necessarily result in maximally (or perfectly) balanced clusters (unless $c = n/k$). On the contrary, `Balance` focuses on the ratio of the largest and smallest protected groups within each cluster. Although a prior work reformulates the `Balance` concept by utilizing linear programming to set upper bounds on the number of points from each group [2], this extension does not directly apply to online settings and is limited to k-center. A few studies also examine online facility location [3, 4]. However, these works differ slightly from ours as they either focus on assignment problems without addressing facility location or leverage the benefits of multiple expert advice, which differs from online CCP.

[2] With unrestricted capacity constraints, our problem reduces to the problem of uncapacitated online clustering.

[3] In terms of dataset size and dimensionality.

[4] In terms of visualization and interpretation.

3 Preliminaries

Let $X \subseteq \mathbb{R}^h$ be an endless stream of data points with x_t being the point arriving at time t. Each data point $x_t \in X$ is articulated using h dimensional real-valued features. We assume that these points are embedded in metric space with $d : X \times X \to \mathbb{R}^+$ measuring the dissimilarity between any two data points. Then, the goal of any centroid-based clustering algorithm is to partition data points into clustering $\mathcal{C} = (C, \phi)$. The clustering produces k disjoint subsets ($[k] = \{1, \ldots, k\}$) with centers $C = \{c_j\}_{j=1}^{k}$ using an assignment function $\phi : X \to C$ that maps each point to corresponding cluster center. A vanilla algorithm produces a clustering which minimizes the following cost:

Definition 1 (Objective Cost)**.** *For metric space $(X, d(\cdot))$ with p-norm, the cost is defined as $W = \left(\sum_{x_t \in X} d(x_t, \phi(x_t))^p\right)^{\frac{1}{p}}$.*

Different values of norm p leads to different objective functions, such as, $p = 1$ for k-median, $p = 2$ for k-means, and $p = \infty$ for k-center. Also, let W^*, ϕ^* represent the optimal (offline/online) objective cost and the optimal assignment function, respectively. We now define the capacity constraint mapping $\boldsymbol{\gamma} : C \to [0, 1]$, representing the capacity on fraction of points each cluster accommodates. We consider the following assumptions on capacity constraints:

- Capacities are same across all clusters i.e., $\boldsymbol{\gamma}(c_j) = \gamma,\ \forall j \in [k]$. This ensures equal treatment among clusters, avoiding any favouritism. Furthermore, in the online setting, imposing capacity constraints at the cluster level is infeasible due to the dynamic nature of the number of opened centers.
- With n as the total data points that the algorithm eventually sees, we adopt an assumption that γ is a multiple of n/k, i.e., $\gamma = \frac{\alpha n}{k}$. Here α indicates the permissible degree of skewness among clusters and belongs to $[1, k]$. When $\alpha = 1$, it results in perfect uniform cluster sizes, while $\alpha = k$ indicates an uncapacitated clustering problem.

We now provide a brief overview of the uncapacitated OC methods [35, 6]. We begin with the initial work, as described by [35]. The fully online algorithm initiates by selecting the first $(k + 1)$ data points as the initial set of centers to estimate the lower bound on objective cost (w^*). The heuristic is based on the idea that clustering $(k + 1)$ data points should put at least two points together. Subsequently, for the remaining data points, the algorithm determines whether to assign each point to the nearest center or if the data point's distance incurs a high assignment cost $d(x_t, c)$. This assessment is quantified using a probability that depends on the ratio of the assignment cost to the center opening cost (f_r). To prevent excessive points from being opened as centers, value of f_r for round r doubles when the center count exceeds a predefined threshold. While [6] improves this method, the algorithm now makes delayed decisions. This implies that if the current data point needs to be opened as a center, it is not opened immediately but deferred to a later time. Although this delayed approach contributes to improved objective cost approximation by a logarithmic factor, the current focus of the study is on immediate assignment or opening of data points, as necessitated by the need in the running example in Section 1, i.e., each new consumer must be promptly assigned to a salesperson to ensure the seamless operation of the business and timely product deliveries. A delayed response from the wholesaler could result in a shift to alternate avenues. Consequently, we build upon the algorithm presented in [35] by extending it to capacitated clustering. Our approach introduces several modifications that result in substantial enhancements over the conventional online clustering problem (and subsequently to online CCP).

- Through experimental observation, we have noted that an initial selection of $(k + 1)$ data points for estimation of lower bound on optimal cost can potentially result in a higher likelihood of opening more centers in future. It is primarily due to bad cost estimation that the algorithm relies on. Instead, we propose a selection criterion based on the non-uniform coupon collector problem.
- In [35], algorithm estimates the total data points using the current point count observed, achieving $O(\log n)$ cost approximation. Our approach improves this by updating the estimation with the doubling trick, providing improved constant cost approximation.

We now formally restate **non-uniform coupon collector problem** with replacement [19]:

Claim 1. *Given ℓ distinct coupon types, the expected number of coupons required to obtain at least one coupon from each type is denoted as $\mathcal{H}_\ell$, and it is calculated as follows: $\mathcal{H}_\ell = \sum_{a=1}^{\ell} (-1)^{a-1} \sum_{1 \le j_1, \ldots, j_a \le k} \frac{1}{p(j_1) + \ldots + p(j_a)}$ where $p(i)$ is the probability of obtaining a coupon of type i.*

We aim to determine the expected number of data points be opened as a center to ensure representation from each center in the optimal capacitated clustering. As this will result in a better approximation of w^*, which the fully online algorithm will require. To achieve this, we employ a non-uniform coupon collector problem as follows: consider coupons to be the data points and each coupon type to be the centers in the offline optimal clustering (i.e., $\ell = k$ in our case). However, the main challenge lies in computing the probabilities $p(i)$, representing the probability of a data point belonging to cluster i. When the capacities are uniform, i.e. n/k with $\alpha = 1$ and data points are coming uniform at random from any cluster, it becomes evident that we obtain $p(i) = 1/k\ \forall i \in [k]$ thus leading to $H_k = k \log k$. Further, since $p(i)'s$ are not known to us, we restrict the value of H_k to be $k \log k$, and therefore, the total number of centers opened by `COCA` remains of the same order as that of by [35]. It must be noted that, in order to satisfy Claim 1, however, H_k may reach the value of n (when differences in $p(i)$'s are arbitrarily high), which is again consistent with the literature as shown by [40] that even with knowledge of n, any algorithm would inevitably open $\Omega(n)$ centers in the worst case ordering of data points. Note that our theoretical proofs hold and remain unaffected by choice of $p(i)$'s and the value of H_k. It's just that if prior information about sampling probabilities is known, one can leverage Claim 1 to obtain a better estimate of initial centers (and w^*).

4 Capacitated Semi-Online Clustering Algorithm (CSCA)

We first begin by looking into semi-online clustering wherein the total number of points (n) and lower-bound on optimal cost (w^*) is known. Note that most restrictions in fully online clustering (i.e., when both these n, w^* are unknown) apply to semi-online clustering. This means that for each data point, the algorithm must make an irrevocable decision of either assigning it to the existing centers or open it as a new center.

The algorithm, referred to as the Capacitated Semi-online Clustering Algorithm (`CSCA`), is described in Algorithm 1. `CSCA` begins by opening the first data point as the center. Since we know w^*, we do not use H_k. Subsequently, for the remaining data points, with probability based on distance to closest center with remaining capacity a data point is opened as a center otherwise the algorithm assigns it to the nearest vacant center. We now look into `CSCA`'s theoretical guarantees.

Algorithm 1 Capacitated Semi-online Clustering Algorithm

Input: X, n, w^* and capacity constraint γ
Output: Centers C and assignment function $\hat{\phi}$.

Initialize $\Gamma \leftarrow \varnothing$.
Open first point as center (c_1) and set $\Gamma(c_1) = \gamma$.
Initialize $\phi \leftarrow \varnothing$, $r \leftarrow 1$, $q_r \leftarrow 0$.

Initialize center opening cost $f_r = w^* \alpha / k \log(n)$
for remaining $x_t \in X$ **do**
  $c \leftarrow \text{argmin}_{c \in C:\Gamma(c)>0}\, d(x_t, c)$
  with probability $p_t = \min\left(d(x_t, c)/f_r, 1\right)$
    $C \leftarrow C \cup \{x_t\}$; $\phi(x_t) = x_t$; $\Gamma(x_t) = \gamma$; $q_r \leftarrow q_r + 1$
  otherwise with $1 - p_t$
    $\phi(x_t)$=c; $\Gamma(c) = \Gamma(c) - 1$

  if $q_r \geq \frac{3k}{\alpha}(1 + \log n)$ **then**
    $r \leftarrow r + 1$; $q_r \leftarrow 0$; $f_r \leftarrow 2 \cdot f_{r-1}$
  end if
end for
return (C, ϕ).

4.1 Theoretical Results

We now first look into the expected number of centers opened by Algorithm 1, and subsequently, cost approximation bounds. To this, let us denote optimal clustering as $\mathcal{C}$ with corresponding clusters $\{\mathcal{C}_1^*, \ldots, \mathcal{C}_k^*\}$ and assignment function ϕ^*. We omit the p-norm factor from the distance function in the proofs for ease of reading. However, proofs hold for all finite values of p and also for special cases: 1 (k-median) and 2 (k-means).

Theorem 2. *Let C be a set of cluster centers opened by Algorithm 1. Then,* $\mathbb{E}[|C|] = O\left(\frac{k}{\alpha}\log(n)\log\left(\frac{W^*}{w^*}\right)\right)$.

Proof. Let W_i^* be the optimal capacitated clustering cost of cluster i and is given by $W_i^* = \sum_{x \in \mathcal{C}_i^*} d(x, \phi^*(x))$. So the total optimal cost is $W^* = \sum_{i=1}^k W_i^*$. Further, let A_i^* denote the average distance from points in the i^{th} optimal cluster to its center and is computed as $A_i^* = \frac{1}{|\mathcal{C}_i^*|}\sum_{x \in \mathcal{C}_i^*} d(x, \phi^*(x)) = \frac{W_i^*}{|\mathcal{C}_i^*|}$.

Now, our primary goal is to bound the number of centers opened. We have k optimal clusters, and as the arrival of points is unknown in the online setup, we end up opening more centers in each cluster as an estimation of the optimal center. Let us now divide the k optimal clusters into different rings motivated from [8, 9, 39]. The broader idea is to compute the expected number of centers that we end up opening in each of these rings. The 0^{th} ring is denoted by $\mathcal{C}_{i,0}^*$=$\{x \in \mathcal{C}_i^* : d(x, \phi^*(x)) \leq A_i^*\}$. The subsequent rings, from 1 to τ, are given by $\mathcal{C}_{i,\tau}^*$=$\{x \in \mathcal{C}_i^* : 2^{\tau-1}A_i^* < d(x, \phi^*(x)) \leq 2^\tau A_i^*\}$. Note that a cluster $\mathcal{C}_i^*$ will be divided into $(1 + \log n)$ rings, as all rings after $\log n$ will be essentially empty. Let r' be the first round when the center opening cost $f_{r'}$ becomes some fraction of W^* such that, $f_{r'} \geq \frac{2^4 W^* \alpha}{k \log n}$. Now, we bound the expected number of centers in two separate parts, i.e., before round r' and second during and after round r'. Let us first begin with the former,

Case 1: By the definition of r', we have $f_{r'-1} < \frac{2^4 W^* \alpha}{k \log n}$. Further, since the center opening cost becomes twice at every round, we have, $f_{r'-1} = 2^{r'-1} f_1$. Substituting the value of $f_1 = \frac{w^* \alpha}{k \log(n)}$, we get, $r' \leq \log\left(\frac{W^*}{w^*}\right) + 5$. Therefore, before round r', the number of centers opened by the algorithm is,

$$\mathbb{E}(|C|_{\text{before } r'}) = O\left(\frac{3k}{\alpha}(1 + \log n)\log\left(\frac{W^*}{w^*}\right)\right) \tag{1}$$

Case 2: Now, let's look into computing the number of centers opened during and after round r' in each of these rings. To avoid getting struck due to not knowing order of arrival of data points, we will loosely estimate the expected number of centers present in any ring during or after round r'. To this, we divide the bounds into three subparts-

Case 2(a): First, we estimate the number of new centers that will open for the first time in each ring. Let's denote these centers as K_τ^1. Since there are a total of $(1 + \log n)$ rings in each cluster, therefore the total number of such centers are $\sum_k \sum_\tau 1 = k(1 + \log n)$.

Case 2(b): Next, suppose there is a data point x that arrives and the closest center to x has already reached its capacity; in such a case, the data point will continue searching for the next closest center in any of the rings in increasing order of distance. There are two possibilities: either data point x will find a vacant center or its likelihood of becoming a center increases as it delves further into the chain if the next closest center is too far away (handled in next case). Let's denote the extra number of centers that need to opened up in any ring $\boldsymbol{\tau}$ due to exhausting of capacity of first centers of Case 2a, be denoted by K_τ^c. It is important to note that once a center K_τ^1 fills up and we need to open second center within the ring, then atleast $\frac{\alpha n}{k}$ data points have already arrived and been assigned. Therefore, the total number of such K_τ^c centers over all rings and clusters is upper bounded by k/α i.e., $\sum K_\tau^c \leq k/\alpha$.

Case 2(c): Now since there were two possibilities: either data point x will find a vacant center or its likelihood of becoming a center increases as it delves further into the chain if the next closest center is too far away. Now, the remaining task is to bound the probabilistically opened centers in each ring apart from the centers opened in previous two subcases. To do this according to Algorithm 1, if a data point x is the initial center opened within any ring, then the probability of subsequent point x' from the same ring opening as a center is defined and bounded using the properties of rings as follows:

$$\frac{d(x, x')}{f_{r'}} \leq \frac{d(x, \phi^*(x)) + d(x', \phi^*(x))}{f_{r'}} \leq 2 \cdot 2^\tau \frac{A_i^*}{f_{r'}}$$

($\because$ Using triangular inequality and ring property)

So, the expected number of centers that will open in any ring over all rounds $r \geq r'$ is $\sum_{r \geq r'} \frac{2 \cdot 2^\tau A_i^*}{f_r} |\mathcal{C}_{i,\tau,r}^*|$. Summing these probabilistic centers over all rings and using $f_r \geq f_{r'}$ we obtain the number of centers opened for estimating one optimal cluster center as,

$$\begin{aligned} K_{\boldsymbol{\tau}}^p &= \sum_{\tau \geq 0}\left(\sum_{r \geq r'} \frac{2 \cdot 2^\tau A_i^*}{f_r}|\mathcal{C}_{i,\tau,r}^*|\right) \leq \sum_{\tau \geq 0} \frac{2 \cdot 2^\tau A_i^*}{f_{r'}} \sum_{r \geq r'} |\mathcal{C}_{i,\tau,r}^*| \\ &\leq \sum_{\tau \geq 0} \frac{2 \cdot 2^\tau A_i^*}{f_{r'}}|\mathcal{C}_{i,\tau}^*| \leq \frac{2A_i^*|\mathcal{C}_{i,0}^*|}{f_{r'}} + \frac{4}{f_{r'}} \sum_{\tau \geq 1} \sum_{x \in \mathcal{C}_{i,\tau}^*} 2^{\tau-1} A_i^* \\ &\leq \frac{6W_i^*}{f_{r'}} \end{aligned}$$

(Using $W_i^* = A_i^*|\mathcal{C}_i^*|$ and $2^{\tau-1}A_i^* \leq d(x, \phi^*(x))$)

Summing this up for all k cluster centers and considering the estimate of $f_{r'} \geq \frac{16W^*\alpha}{k \log n}$ we get,

$$K_k^p \leq \frac{6W^*}{f_{r'}} \leq \frac{6k \log n}{16\alpha} \tag{2}$$

Therefore, number of total centers opened in Case 2 are as follows

$$\mathbb{E}[|C|_{\text{during and after } r'}] = O\left(K^1_{\boldsymbol{\tau}} + \sum_{\boldsymbol{\tau},k} K^c_{\boldsymbol{\tau}} + K^p_k\right)$$

$$= O\left(k(1+\log n) + \frac{k}{\alpha} + \frac{k\, log(n)}{\alpha}\right) = O\left(\frac{k}{\alpha} log(n)\right) \quad (3)$$

Thus, combining Equation 1 and 3, completes the proof, resulting in the total expected number of centers opened by CSCA as $O\left(\frac{k}{\alpha}\log(n)\log\left(\frac{W^*}{w^*}\right)\right)$. Note that for the unbounded capacity case, when $\alpha = k$, our bounds in semi-online algorithm CSCA match with that of Liberty et al [35].

□

Theorem 3. *Let $W_{\texttt{CSCA}}$ represent the cost of the semi-online capacitated cost and W^* denote the optimal offline capacitated cost. Then, $\mathbb{E}[W_{\texttt{CSCA}}] = O(W^*)$.*

Proof. To approximate the cost guarantees, our primary focus is on bounding the assignments in line 10 in CSCA. In all other assignments, data points are centers themselves, resulting in zero cost. However, after the opening of these initial centers, the data points have two possibilities of getting assigned. Firstly, they may be assigned to one of the centers within the same ring as the data point's optimal ring. Secondly, suppose the center within the same ring is already occupied; in that case, points may be assigned to a center located in a different ring within the same cluster or in a ring belonging to a different cluster. We first bound the latter as follows:

Case 1: Note that the cost of points (say $x_t \in X$) going to rings other than the optimal one incurs a cost equal to the distance to the assigned center from set C. We will use the Lemma 1 of [35] to bound these costs, i.e., $\mathbb{E}(d(x_t, C))$ and restate the lemma below:

Lemma 4 ([35])**.** *Given a sequence of n independent experiments, each of which succeeds with probability atleast $\min(A_i/B, 1)$ where $B \geq 0$ and $A_i \geq 0\ \forall i \in [n]$. Let t be the (random) number of sequential unsuccessful experiments, then, $\mathbb{E}(\sum_{i=0}^{t} A_i) \leq B$.*

Now, before we delve into using the above lemma, let us first understand the mapping between our problem and the technical lemma. In CSCA probability of event (center opening) is atleast $\min(d(x_t, C)/f_{r'}, 1)$ where x_t is any data point and C is set of existing vacant centers. On using the fact that given R as the last round, $f_R > f_{r'}$ for any round $r' < R$, the denominator can be made constant as in the lemma. Now, each independent unsuccessful experiment represents the assignment of one data point, and we can use the lemma to bound the expected value of the sum of A_i's ($d(x_i, C)$'s in our problem) by $B(= f_R)$. Note that once the event gets successful, i.e., the center in any ring gets opened, we will bound the cost of assignments to the opened center in the next case, but here we look into the scenario once this center gets filled up. In such a situation, assignments are again upper bounded by $O(f_R)$ along similar lines.

Case 2: We will next bound the cost of all data points that are allocated within the ring. Now, after any point x is opened as center, then cost of subsequent point x' is given by $d(x, x') \leq d(x, \phi^*(x)) + d(x', \phi^*(x)) \leq 2 \cdot 2^{\tau} A^*_i$ (Using triangular inequality and ring property). Now, using Equation 2, the total cost over all such x' is given as $\sum_{x'} d(x, x') \leq 6W^*$. It is important to note that unlike the uncapacitated case in [35], once a center gets opened in the ring, its capacity can eventually get exhausted, and then one returns to Case 1 and needs to wait until the next center is opened within the ring and once a new center opens up, which is already accounted for by Case 2. Therefore, the total expected cost by combining both cases over all rings is given as follows:

$$O(f_R k \log n + W^*) \quad (4)$$

Therefore, we must find our case's expected value of f_R. To this, let us consider some round r' in CSCA such that,

$$f_{r'} \geq \frac{16W^*\alpha}{k\log n} \geq \frac{16W^*\alpha}{k(1+\log n)} \quad (5)$$

Using Equation 2 (i.e., $6W^*/f'_r$), Equation 5 and Markov inequality, the probability of opening more than $\frac{3k}{\alpha}(1+\log n)$ centers is $\frac{1}{8}$ and thus, CSCA concluding at round r' is equal to $\frac{7}{8}$. Now, let b be probability that CSCA terminates before round r', then,

$$\mathbb{E}[f_R] \leq b f_{r'-1} + (1-b)\sum_{r=r'}^{\infty} f_r \left(\frac{7}{8}\right)\left(\frac{1}{8}\right)^{(r-r')}$$

$$\leq b f_{r'} + \frac{7}{8}(1-b)\sum_{i=0}^{\infty} f_{r'+i}\left(\frac{1}{8}\right)^i < O(f_{r'})$$

(Using $f_{r'+i} = 2^i f_{r'}$ and $\frac{1}{8} < 1$)

$$\implies \mathbb{E}[W_{\texttt{CSCA}}] = O(f_R k \log n + W^*) = O(W^*).$$

□

5 Capacitated Online Clustering Algorithm (COCA)

Now, we delve into a fully online setup in which n is unknown, and the algorithm needs to compute the lower-bound w^* without any prior knowledge of n. To approximate w^*, the algorithm leverages the insight that once H_k ($\geq k$) data points are opened as center, a more accurate estimation of w^* can be obtained by performing clustering on these H_k data points (see Claim 1). Further, for monitoring the estimated value of n, the method utilizes a doubling technique in lines 16 to 17, wherein the estimate is doubled once it is achieved. The code is outlined in Algorithm 2, with the highlighted portion illustrating the variations compared to the semi-online setup.

5.1 Theoretical Results

Theorem 5. *If C is set of centers opened by COCA, then, $\mathbb{E}[|C|] = O\left(H_k + \frac{k}{\alpha}\log(n)\log(n\delta)\right)$ where $\delta = \frac{\max_{x,x'} d(x,x')}{\min_{x,x':x\neq x'} d(x,x')}$.*

Proof. The proof will follow similarly to Theorem 2 except for the fact that in each round instead of opening $\frac{3k}{\alpha}(1 + log\ n)$ centers, we are opening $\frac{3k}{\alpha}(1 + log\ n_r) \leq \frac{3k}{\alpha}(1 + log\ n)$ centers for all r except the last round. Even for the last round $n_r \leq 2n$. Therefore, we can simply substitute the value of w^* and W^*. Now, as Algorithm 2 computes w^* as capacitated cost using [41] on H_k points. So, $w^* \geq \min_{x,x'\in X: x\neq x'} d(x, x')$. Similarly, the optimal capacitated cost $W^* \leq n \max_{x,x'\in X} d(x, x')$ (as the maximum distance from any center (data point) to other points is bounded by the maximum pairwise distance). Substituting these values and adding initial H_k centers completes the proof. □

Theorem 5 indicates that the number of centers can be negatively affected by the presence of the term H_k. However, our experiments demonstrate that selecting the initial H_k data points as center, rather than $(k+1)$ (as done in [35]), actually contributes to opening overall fewer centers because it results in a better estimate of w^*. Further,

Algorithm 2 Fully online COCA

Input: X and capacity constraint γ
Output: Centers C and assignment function $\hat{\phi}$

Initialize $\Gamma \leftarrow \varnothing$ {stores vacant capacity of center}.
Open first H_k points as centers. (Claim 1) and $\forall j \in [H_k]$ set $\Gamma(c_j) = \gamma$
Initialize $\phi \leftarrow \varnothing$, $r \leftarrow 1$, $n_r \leftarrow H_k$, $q_r \leftarrow H_k$, idx $\leftarrow H_k$.
$w^* \leftarrow$ objective cost on H_k using Mulvey et al [41].
Initialize center opening cost $f_r = (w^* \cdot \alpha)/(k \cdot \log n_r)$
for remaining $x_t \in X$ **do**
  $c \leftarrow \text{argmin}_{c \in C: \Gamma(c) > 0} d(x_t, c)$
  with probability p_t = $\min\left(d(x_t, c)/f_r, 1\right)$
    $C \leftarrow C \cup \{x_t\}$; $\phi(x_t) = x_t$; $\Gamma(x_t) = \gamma$; $q_r \leftarrow q_r + 1$
  otherwise with $1 - p_t$
    $\phi(x_t)$=c; $\Gamma(c) = \Gamma(c) - 1$
  if $q_r \geq \frac{3k}{\alpha}(1 + \log n_r)$ **then**
    $r \leftarrow r + 1$; $q_r \leftarrow 0$; $f_r \leftarrow 2 \cdot f_{r-1}$
  end if
  idx $\leftarrow$ idx $+ 1$
  if idx $\geq n_r$ **then**
    $n_r \leftarrow 2n_r$
  end if
end for
return (C, ϕ).

when $k \geq 2$ and n is unknown, [40] shows that at least $\Theta(\log n)$ centers for random ordering are needed to achieve constant cost approximation. Our upper bound in the online capacitated setting ($\alpha = k$) aligns with lower bounds in the uncapacitated setting.

Theorem 6. *Let $W_{\texttt{COCA}}$ be the cost of online Algorithm 2 and W^* be the optimal offline capacitated cost. Then, $\mathbb{E}[W_{\texttt{COCA}}] = O\left(W^*\right)$.*

Proof. We begin with Equation 4 given in Theorem 3 i.e, $\mathbb{E}[W_{\texttt{COCA}}] = O(f_R k \log n + W^*)$. Thus, we need to estimate the value of f_R at the last round R. Let us consider any round r such that $f_r \geq \frac{16W^*\alpha}{k\log(n_r)}$. Then, number of centers opened in round r is given as $q_r \leq \frac{k}{\alpha}(1 + \log(n_r)) + q'_r$. Here, we pessimistically count one (first) centers in each ring up to round r and q'_r is the number of centers opened in rings after opening former $\lceil k/\alpha \rceil$ centers. In order to have more rounds than r, COCA needs $q'_r \geq \frac{2k}{\alpha}(1 + \log(n_r))$. We will now compute the probability that COCA terminates by round r. Applying Markov inequality by using the above information along with $\mathbb{E}(q'_r) \leq 6W^*/f_r$ from Equation 2, we get the probability of reaching the next round as at most $3/16$. Thus, if b is the probability that COCA terminates before round r. We have,

$$\begin{aligned}
\mathbb{E}(f_R) &= bf_{r-1} + (1-b)\sum_{r'=r}^{\infty} f_r \left(\frac{13}{16}\right)\left(\frac{3}{16}\right)^{r'-r} \\
&< bf_r + f_r(1-b)\left(\frac{13}{16}\right)\sum_{i=0}^{\infty} 2^i \left(\frac{3}{16}\right)^i \\
&= O(f_r) = O\left(\frac{16\alpha W^*}{k\log(n_r)}\right). \\
&\qquad \text{(Using } f_{r-1} = 2f_r \text{ and } b \leq 1)
\end{aligned}$$

On substituting this back, we get

$$\mathbb{E}[W_{\texttt{COCA}}] = O(f_R k \log(n) + W^*) = O\left(\frac{16W^*\alpha k \log n}{k \log n_r} + W^*\right)$$

Now since with high probability the algorithm will terminate at r^{th} round and from doubling trick, we can say, $n_r \geq n$. So, $\mathbb{E}(W_{\texttt{COCA}}) = O\left(W^*\alpha + W^*\right) = O(W^*)$. This completes the proof. The derived bounds exhibit a substantial reduction by a logarithmic factor compared to [35]. Note that, due to capacity constraints in an online setup, there may be some misassignments compared to the offline method. However, these disruptions will be minimal owing to constant cost bounds. Additionally, all results hold for any scalar distance metric and for higher dimensions, manhattan or fractional norms [1] are sometime preferred over euclidean.

□

6 Experimental Results and Discussion

We will now validate our approach against SOTA on following datasets motivated by clustering literature [26]:

- **Synthetic1d**: consist of 1000 points sampled each from $\{\mathcal{N}_i(\mu = 1 + a \cdot i, \sigma = 2)\}_{i=1}^{k}$, where a = 7 in well separable (s) and a = 5 in partially overlapping (o) clusters.
- **Synthetic2d**: consist of 1000 points sampled each from $\{\mathcal{N}_i(\mu = 1 + a \cdot i, \Sigma = I_{2\times 2})\}_{i=1}^{k}$, where a as above and I is identity matrix. Synthetic2d is shortened to Syn2d.
- **Adult**: A public 1994 US census data[5] of $32K$ people with five features provided in Appendix A [27].
- **Bank**: Portuguese marketing data[6] of $41K$ records regarding six client features (see Appendix A [27]).
- **Diabetes**: US Medical records over ten years with $100K$ instances over two features[7].

We evaluate the performance of COCA against the following:

- **Uncapacitated Online k-means** We call fully online algorithm as LIB for comparison with COCA (see prelims for details and Appendix B for pseudo-code [27]). A heuristic approach is also provided by the authors in which they initially open $(k+1)$ data points as centers and compute w^*, by taking half sum of ten closest neighbours instead of the pairwise minimum distance between $(k+1)$ data points. Further, they drop logarithmic factors by setting $q_r \geq k$ and increasing f_r by ten times instead of doubling it. We denote this as $\texttt{LIB}_\texttt{H}$ [35]. Note that while $\texttt{LIB}_\texttt{H}$ outperforms LIB but, it lacks theoretical results to support it.
- **Capacitated Online Clustering Heuristic** (COCH): Motivated from $\texttt{LIB}_\texttt{H}$, we also use heuristic with a selection of $(k+1)$ initial points as centers, setting $q_r \geq k$ and updating f_r by ten times in COCA to enable comparison of COCH with $\texttt{LIB}_\texttt{H}$ (Code in Appendix B [27]).
- **Offline Capacitated Clustering** (CAP): Assigns data points to closest vacant centers and performs mean (or median) center updates as in heuristic [41]. We focus on the k-means version in the main paper and defer the comparison to k-median in Appendix C [27].

Experimental Setup: All experiments are performed on Intel Xeon with 280GB RAM, and Python 3.6. We report mean and standard deviation over ten independent runs and seed from set $\{0, 100, \ldots, 900\}$. Notably, the capacity parameter (α) is such that $\alpha \geq 1$, with $\alpha = 1$ representing the most restrictive scenario, i.e., having uniform capacities. Consequently, we showcase the efficacy of our algorithms under uniform capacity in the main paper. However, we observe similar findings on other α values and defer the results to Appendix C [27] due to space constraints. The code[8] and Appendix [27] are public.

[5] archive.ics.uci.edu/ml/datasets/Adult
[6] archive.ics.uci.edu/ml/datasets/Bank+Marketing
[7] archive.ics.uci.edu/dataset/116/us+census+data+1990
[8] https://github.com/shivi98g/Capacitated-Online-Clustering-Algorithm

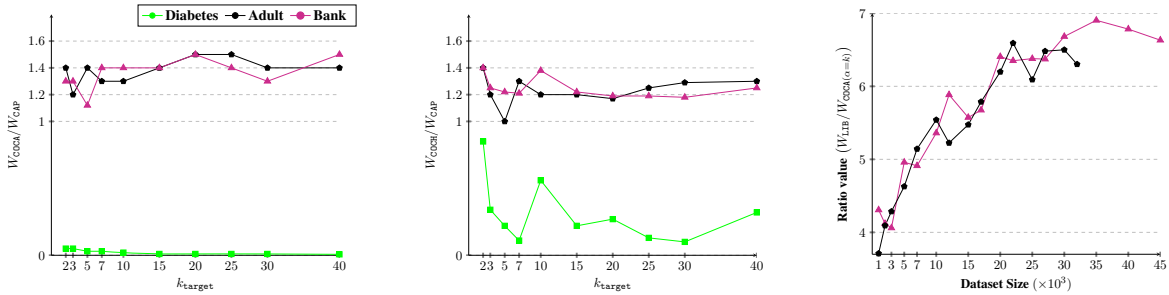

Figure 1: (Left to Right): (a) Constant cost approximation of COCA to CAP kmeans. (b) COCH to CAP kmeans (c) Logarithmic trend in cost ratio of LIB to COCA with $\alpha = k$. Note: Input to online methods is $k_{\texttt{target}}$ but the cost for both methods is computed on the resulting centers ($k_{\texttt{actual}}$).

Metric: We re-introduce: $k_{\texttt{target}}$, $k_{\texttt{actual}}$. The former is the input for any online algorithm, while the latter represents the count of final centers opened. Also, we compare COCA's performance on cost. It involves comparing the cost of online solutions when executed for $k_{\texttt{target}}$ as input (resulting in $k_{\texttt{actual}}$ centers) v/s their offline counterparts when executed and compared on $k_{\texttt{actual}}$ centers. **An important note** that since LIB and the proposed COCA have theoretical guarantees and should thus be compared. In contrast, $\texttt{LIB}_{\texttt{H}}$ and proposed COCH are heuristic approaches that warrant comparison.

Analysis on Number of Centers Opened: We compare the centers opened by COCA, COCH with SOTA. Results for $k_{\texttt{target}}$ of 2, 10, and 30 are in Tables 1, 2, and 3. Results for other targets are provided in Appendix C [27]. Notably, for a lower target of 2, COCA needs more centers to meet capacity constraints. Conversely, as the target increases, LIB's performance degrades considerably validating the opening of H_k number of initial centers instead of opening only $(k+1)$ points. For heuristics, the gap between $k_{\texttt{target}}$, $k_{\texttt{actual}}$ is tolerable for lower targets and slightly high in higher targets, considering the rising uncertainty of the arrival order of points.

Dataset	LIB	COCA	$\texttt{LIB}_{\texttt{H}}$	COCH
Adult	292.9±20.79	541.7 ± 79.80	9.0±0.0	9.0±0.0
Bank	311.9±33.29	544.3 ± 82.69	9.0±0.0	9.0±0.0
Diabetes	109.5±15.18	143.5 ± 16.55	8.6±0.79	9.0±0.0
Syn2d-(s)	134.2±45.52	113.9 ± 19.32	7.0±1.54	8.5±0.67
Syn2d-(o)	142.1±72.06	137.1 ± 21.39	7.0±0.44	7.8±0.60
Syn1d-(s)	104.8±98.75	57.6 ± 8.18	6.8±1.32	6.2±0.75
Syn1d-(o)	66.6±34.67	61.9 ± 7.54	7.8±1.32	6.6±0.49

Table 1: $k_{\texttt{actual}}$ on various SOTA methods when $k_{\texttt{target}}$ is 2.

Dataset	LIB	COCA	$\texttt{LIB}_{\texttt{H}}$	COCH
Adult	1857.4±206.12	1735.8 ± 202.95	36.5±2.57	41.0±0.44
Bank	1969.5±205.36	1751.4 ± 191.69	37.3±3.00	41.3±0.64
Diabetes	246.3±20.34	189.0 ± 16.05	31.1±0.3	33.9±2.7
Syn2d-(s)	1357.8±358.48	619.3 ± 90.66	29.7±4.10	32.5±3.90
Syn2d-(o)	1357.8±358.48	675.6 ± 99.39	31.0±3.34	33.8±4.77
Syn1d-(s)	938.4±560.99	252.1 ± 44.50	28.7±2.9	31.0±0.44
Syn1d-(o)	910.4±485.25	239.1 ± 34.03	28.7±2.45	31.1±0.3

Table 2: $k_{\texttt{actual}}$ on SOTA methods when $k_{\texttt{target}}$ is 10.

Dataset	LIB	COCA	$\texttt{LIB}_{\texttt{H}}$	COCH
Adult	7136.4±1617.56	3071.1 ± 251.07	114.7±8.1	118.7±4.00
Bank	6756.1±834.40	3485.0 ± 335.15	116.2±6.24	122.9 ± 1.58
Diabetes	271.5±1.74	252.7 ± 12.46	198.7±75.14	93.6±2.42
Syn2d-(s)	6728.9±1585.75	1479.2 ± 201.29	116.8±7.39	94.2±3.89
Syn2d-(o)	6728.9±1585.75	1775.5 ± 222.62	120.2±5.05	97.0±4.63
Syn1d-(s)	4489.2±1692.76	601.4 ± 50.25	116.8±7.39	91.9±0.83
Syn1d-(o)	4055.8±1609.87	567.1 ± 44.78	112.5±8.64	92.5±1.69

Table 3: $k_{\texttt{actual}}$ on SOTA methods when $k_{\texttt{target}}$ is 30.

Analysis on Clustering Cost: Let $W_{\texttt{COCA}}$ and $W_{\texttt{CAP}}$ be costs achieved by COCA and CAP respectively. The results for COCA v/s CAP are depicted in Figure 1 (a). We can observe a nearly constant cost approximation, which validates our theoretical findings. A similar trend is noted for COCH (Figure 1b), LIB, $\texttt{LIB}_{\texttt{H}}$ (Appendix C [27]). Also, we compare online to offline uncapacitated costs, reporting a constant factor approximation (Appendix C [27]). Note that the ratio is low in the Diabetes dataset as it suffers from local optima [26], leading to higher value of offline cost.

Dataset	LIB	COCA	$\texttt{LIB}_{\texttt{H}}$	COCH
Adult	4657.4 ± 1060.78	2566.2 ± 256.22	77.1 ± 5.37	81.0 ± 0.0
Bank	4469.8 ± 544.19	2863.4 ± 312.30	80.3 ± 2.09	82.4 ± 2.10
Diabetes	268.8 ± 1.4	211.1 ± 13.07	71.5 ± 7.81	66.0 ± 6.55

Table 4: $k_{\texttt{actual}}$ on SOTA uncapacitated methods when $k_{\texttt{target}}$ is 20 and α is $k_{\texttt{target}}$ for COCA, COCH.

Ablation Study on Variance: Table 1, 2, and 3 exhibit that proposed COCA and COCH demonstrate significantly lower deviations than LIB, $\texttt{LIB}_{\texttt{H}}$. This is attributed to the doubling trick instead of increasing the estimate of the number of points in a linear fashion. Reduced variance helps online algorithms avoid opening more centers than the target.

Reduction to Uncapacitated Problem: We set $\alpha = k$ (unrestricted) in our algorithms and assess their performance on $k_{\texttt{actual}}$ and cost. The results on the number of centers resemble the uniform capacities but significantly better than LIB (see Table 4 and Appendix C [27]) supporting choice to choose H_k initial centers instead of $k+1$. Noteworthy are the findings in the cost comparison of COCA, and LIB (see Figure 1c), which confirm a logarithmic reduction by use of the doubling trick.

7 Conclusion

This work extends the probabilistic algorithm available in uncapacitated to capacitated OC. Our algorithm is the first online algorithm to tackle capacity constraints in h-dimensional space for k-means or k-median. We introduce two novel changes to existing online uncapacitated clustering: First, we determine the initial number of centers to be opened by the algorithm to get a better representation, and second, we employ a doubling trick to estimate the total number of points. These changes result in fewer centers opening while achieving constant cost approximation to the optimal clustering problem. An immediate future direction involves extending the work in the presence of noisy data. Another interesting problem is the extension to group fair assignments [26] or centers [31]. Also, since capacity constraints in online streaming can result in different assignments compared to offline counterparts, focusing on minimizing such reassignments is interesting.

Acknowledgment

We would like to thank PMRF (ID:2901481) and the Science and Engineering Research Board (Anusandhan National Research Foundation), Government of India, under grant number CRG/2022/007927.

References

[1] C. C. Aggarwal, A. Hinneburg, and D. A. Keim. On the surprising behavior of distance metrics in high dimensional space. In *Database theory—ICDT 2001: 8th international conference London, UK, January 4–6, 2001 proceedings 8*, pages 420–434. Springer, 2001.

[2] S. Ahmadian, A. Epasto, R. Kumar, and M. Mahdian. Clustering without over-representation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 267–275, 2019.

[3] A. R. Ahmed, M. S. Rahman, and S. Kobourov. Online facility assignment. *Theoretical Computer Science*, 806:455–467, 2020.

[4] M. Almanza, F. Chierichetti, S. Lattanzi, A. Panconesi, and G. Re. Online facility location with multiple advice. *Advances in Neural Information Processing Systems*, 34:4661–4673, 2021.

[5] M. H. Bateni, A. Bhaskara, S. Lattanzi, and V. Mirrokni. Distributed balanced clustering via mapping coresets. *Advances in Neural Information Processing Systems*, 27, 2014.

[6] A. Bhaskara and A. K. Ruwanpathirana. Robust algorithms for online k-means clustering. In *Algorithmic Learning Theory*, pages 148–173. PMLR, 2020.

[7] M. Boccia, A. Sforza, C. Sterle, and I. Vasilyev. A cut and branch approach for the capacitated p-median problem based on fenchel cutting planes. *Journal of mathematical modelling and algorithms*, 7:43–58, 2008.

[8] M. Charikar, L. O'Callaghan, and R. Panigrahy. Better streaming algorithms for clustering problems. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, pages 30–39, 2003.

[9] 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.

[10] F. Chierichetti, R. Kumar, S. Lattanzi, and S. Vassilvitskii. Fair clustering through fairlets. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 5036–5044, 2017.

[11] A. Choromanska and C. Monteleoni. Online clustering with experts. In *Artificial Intelligence and Statistics*, pages 227–235. PMLR, 2012.

[12] V. Cohen-Addad and J. Li. On the fixed-parameter tractability of capacitated clustering. In *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, volume 132, pages 41–1. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2019.

[13] V. Cohen-Addad, B. Guedj, V. Kanade, and G. Rom. Online k-means clustering. In *International Conference on Artificial Intelligence and Statistics*, pages 1126–1134. PMLR, 2021.

[14] J. Csirik, L. Epstein, C. Imreh, and A. Levin. Online clustering with variable sized clusters. *Algorithmica*, 65(2):251–274, 2013.

[15] D. Dinler and M. K. Tural. A survey of constrained clustering. In *Unsupervised learning algorithms*, pages 207–235. Springer, 2016.

[16] G. Divéki. Online clustering on the line with square cost variable sized clusters. *Acta Cybernetica*, 21(1):75–88, 2013.

[17] G. Divéki and C. Imreh. An online 2-dimensional clustering problem with variable sized clusters. *Optimization and Engineering*, 14(4):575–593, 2013.

[18] G. Divéki and C. Imreh. Grid based online algorithms for clustering problems. In *2014 IEEE 15th International Symposium on Computational Intelligence and Informatics (CINTI)*, pages 159–162. IEEE, 2014.

[19] A. V. Doumas and V. G. Papanicolaou. The coupon collector's problem revisited: asymptotics of the variance. *Advances in Applied Probability*, 44(1):166–195, 2012.

[20] A. Dzhoha and I. Rozora. Multi-armed bandit problem with online clustering as side information. *Journal of Computational and Applied Mathematics*, 427:115132, 2023.

[21] H. Esfandiari, V. Mirrokni, and P. Zhong. Brief announcement: Streaming balanced clustering. In *Proceedings of the 35th ACM Symposium on Parallelism in Algorithms and Architectures*, pages 311–314, 2023.

[22] A. E. Ezugwu, A. M. Ikotun, O. O. Oyelade, L. Abualigah, J. O. Agushaka, C. I. Eke, and A. A. Akinyelu. A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects. *Engineering Applications of Artificial Intelligence*, 110:104743, 2022.

[23] S. Geetha, G. Poonthalir, and P. Vanathi. Improved k-means algorithm for capacitated clustering problem. *INFOCOMP Journal of Computer Science*, 8(4):52–59, 2009.

[24] M. Gnägi and P. Baumann. A matheuristic for large-scale capacitated clustering. *Computers & operations research*, 132:105304, 2021.

[25] D. Goyal and R. Jaiswal. Tight fpt approximation for constrained k-center and k-supplier. *Theoretical Computer Science*, 940:190–208, 2023.

[26] S. Gupta, G. Ghalme, N. C. Krishnan, and S. Jain. Efficient algorithms for fair clustering with a new notion of fairness. *Data Mining and Knowledge Discovery*, pages 1–39, 2023.

[27] S. Gupta, S. Jain, N. C. Krishnan, G. Ghalme, and N. Hemachandra. Appendix: Capacitated online clustering algorithm, Aug. 2024. URL https://doi.org/10.5281/zenodo.13370870.

[28] W. Hu, T. Miyato, S. Tokui, E. Matsumoto, and M. Sugiyama. Learning discrete representations via information maximizing self-augmented training. In *International conference on machine learning*, pages 1558–1567. PMLR, 2017.

[29] J. Huang, S. Gong, and X. Zhu. Deep semantic clustering by partition confidence maximisation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8849–8858, 2020.

[30] J. Huang, Q. Feng, Z. Huang, J. Xu, and J. Wang. Fls: A new local search algorithm for k-means with smaller search space. In *31st International Joint Conference on Artificial Intelligence, IJCAI 2022*, pages 3092–3098. International Joint Conferences on Artificial Intelligence, 2022.

[31] M. Kleindessner, P. Awasthi, and J. Morgenstern. Fair k-center clustering for data summarization. In *International Conference on Machine Learning*, pages 3448–3457. PMLR, 2019.

[32] T. Le Quy, A. Roy, G. Friege, and E. Ntoutsi. Fair-capacitated clustering. In *EDM*, 2021.

[33] Y. Li, P. Hu, Z. Liu, D. Peng, J. T. Zhou, and X. Peng. Contrastive clustering. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 8547–8555, 2021.

[34] Y. Li, M. Yang, D. Peng, T. Li, J. Huang, and X. Peng. Twin contrastive learning for online clustering. *International Journal of Computer Vision*, 130(9):2205–2221, 2022.

[35] E. Liberty, R. Sriharsha, and M. Sviridenko. An algorithm for online k-means clustering. In *2016 Proceedings of the eighteenth workshop on algorithm engineering and experiments*, pages 81–89. SIAM, 2016.

[36] W. Lin, Z. He, and M. Xiao. Balanced clustering: a uniform model and fast algorithm. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 2987–2993, 2019.

[37] J. S. Low, Z. Ghafoori, and C. Leckie. Online k-means clustering with lightweight coresets. In *AI 2019: Advances in Artificial Intelligence: 32nd Australasian Joint Conference, Adelaide, SA, Australia, December 2–5, 2019, Proceedings 32*, pages 191–202. Springer, 2019.

[38] F. Mai, M. J. Fry, and J. W. Ohlmann. Model-based capacitated clustering with posterior regularization. *European journal of operational research*, 271(2):594–605, 2018.

[39] A. Meyerson. Online facility location. In *Proceedings 42nd IEEE Symposium on Foundations of Computer Science*, pages 426–431. IEEE, 2001.

[40] M. Moshkovitz. Unexpected effects of online no-substitution k-means clustering. In *Algorithmic Learning Theory*, pages 892–930, 2021.

[41] J. M. Mulvey and M. P. Beck. Solving capacitated clustering problems. *European Journal of Operational Research*, 18(3):339–348, 1984.

[42] M. Negreiros and A. Palhano. The capacitated centred clustering problem. *Computers & operations research*, 33(6):1639–1663, 2006.

[43] M. J. Negreiros, N. Maculan, A. W. Palhano, A. E. Muritiba, and P. L. Batista. Capacitated clustering models to real-life applications. In F. P. G. Marquez, editor, *Operations Management and Management Science*, chapter 8. IntechOpen, Rijeka, 2022. doi: 10.5772/intechopen.1000213. URL https://doi.org/10.5772/intechopen.1000213.

[44] M. K. Ng. A note on constrained k-means algorithms. *Pattern Recognition*, 33(3):515–519, 2000.

[45] G. J. Oyewole and G. A. Thopil. Data clustering: Application and trends. *Artificial Intelligence Review*, 56(7):6439–6475, 2023.

[46] T. L. Quy, A. Roy, G. Friege, and E. Ntoutsi. Fair-capacitated clustering. *International Educational Data Mining Society*, 2021.

[47] J. A. Silva, E. R. Faria, R. C. Barros, E. R. Hruschka, A. C. d. Carvalho, and J. Gama. Data stream clustering: A survey. *ACM Computing Surveys (CSUR)*, 46(1):1–31, 2013.

[48] P. Zeng, Y. Li, P. Hu, D. Peng, J. Lv, and X. Peng. Deep fair clustering via maximizing and minimizing mutual information: Theory, algorithm and metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23986–23995, 2023.

[49] A. Zubaroğlu and V. Atalay. Data stream clustering: a review. *Artificial Intelligence Review*, 54(2):1201–1236, 2021.