arXiv:2304.00419v1 [cs.LG] 2 Apr 2023

M小批量 k-均值算法在 $O(d/\epsilon)$ 次迭代内终止

Gregory Schwartzman‡

摘要

我们要回答的问题是："在批次上的局部进展是否意味着在整个数据集上的全局进展，从而适用于小批量 k-均值算法？"具体而言，我们研究的小批量 k-均值算法仅当采样批次上的聚类质量改进低于某个阈值时才会终止。

乍看之下，这种算法似乎可能永远运行下去，但我们对上述问题给出了肯定回答，并证明：若批次大小为 $\tilde{\Omega}((d/\epsilon)^2)$，则该算法以高概率在 $O(d/\epsilon)$ 次迭代内终止，其中 d 是输入的维度，ϵ 是终止阈值参数。无论中心点如何初始化，这一结论都成立。使用 k-均值++ 初始化方案时，该算法可达到 $O(\log k)$ 的近似比（与全批量版本相同）。

最后，我们展示了这些结果适用于 scikit-learn（sklearn）Python 库中实现的 mini-batch k-means 算法。

1 引言

mini-batch k-means 算法 [Scu10] 是实践中最常用的聚类算法之一 [PVG+11]。然而，由于其具有随机性，若不显式限制算法的迭代次数，它似乎可能永远不会终止。我们证明，当批大小足够大时，仅使用"提前停止"条件——即当观察到某个批次上的局部进展低于某个阈值时终止算法——就能保证算法执行的迭代次数有一个与输入大小无关的上界。

问题陈述 我们考虑以下优化问题。给定一个大小为 n 的输入（数据集）$X = \{x_i\}_{i=1}^n \subseteq [0,1]^d$，其中包含 d 维实向量，以及一个参数 k。注意，假设 $X \subseteq [0,1]^d$ 在文献中很常见 [AMR11], ，其目的是简化记号（否则我们还必须为 X 的直径引入一个新参数）。我们的目标是找到一个由 k 个中心点（ $[0,1]^d$ 中的向量）组成的集合 $\mathcal{C}$，使以下目标函数最小化：

$$\frac{1}{n}\sum_{x\in X}\min_{c\in\mathcal{C}}\|c-x\|^2$$

通常， $1/n$ 因子不会出现，因为它不会影响优化目标；不过，在我们的情况下，将其定义为这样会很有用。

Lloyd 算法 解决上述问题最常用的方法是 Lloyd 算法（通常称为 k-均值算法） [Llo82]。该算法首先随机初始化一组 k中心点，然后执行以下两个步骤：(1) 将 X 中的每个点分配给距离其最近的中心点

‡日本先进科学技术研究所，
greg@jaist.ac.jp。

。(2) 将每个中心点更新为分配给它的点的均值。当没有点被重新分配到新的中心点时，算法终止。该算法在实践中的运行速度极快，但最坏情况下的运行时间为指数运行时间 [AV06,Vat11]。

小批次 k-均值算法 为了更新中心点，Lloyd 算法必须在每次迭代时遍历整个输入。当输入数据极其庞大时，这会带来很高的计算开销。为此，Sculley [提出了小批次 k-均值方法10]。该方法与 Lloyd 算法类似，但步骤 (1) 和 (2) 是在从 b 个元素中均匀随机地有放回抽取一个批次后执行的，而在步骤 (2) 中，中心点的更新方式略有不同。具体而言，每个中心点都会更新为其当前值与分配给它的点（批次中的点）均值的加权平均值。用于对这些值加权的参数称为学习率，其取值因中心点和迭代次数而异。在 Sculley 的原始论文中，没有类似 Lloyd 算法的停止条件；相反，算法只执行 t 次迭代，其中 t 是一个输入参数。

在实践中（例如 sklearn [PVG+11]），除了迭代次数的上限外，还有若干“提前停止”条件。当中心点位置的变化足够小时，或者连续多个批次的目标函数变化没有改善时，我们可以终止算法。值得注意的是，无论在理论上 [TM17,Scu10] 还是在实践中 [PVG+11]，学习率都会随时间趋于 0。也就是说，随着时间推移，中心点的移动幅度会越来越小，这保证了在极限情况下，大多数合理的提前停止条件最终都会得到满足。

我们的结果首次证明，采用提前停止条件的小批量 k-均值算法具有极快的终止保证。令人惊讶的是，我们不需要要求学习率趋于 0。

相关工作 小批量 k-均值算法最初由 [Scu10] 提出，作为在线 k-均值算法 [BB94]（此处批次大小为 1）的自然推广。据我们所知，只有一篇论文分析了小批量 k-均值算法的收敛速率 [TM17]。[TM17] 声称，在温和的假设下，该算法具有 $O(1/t)$ 的收敛速率。也就是说，经过 t 次迭代后，当前目标函数值与 Lloyd 算法某个局部最优解处的目标函数值之间，相差不超过一个加性 $O(1/t)$ 因子。然而，他们的渐近符号掩盖了依赖于输入规模的因子。考虑到这一点，我们得到的收敛速率为 $\Omega(n^2/t)$，这意味着该算法的执行时间上界至多为二次级。这是因为在第 t 次迭代时将学习率设为 $O(1/(n^2+t))$。我们的结果不保证收敛到任何局部极小值，但保证运行时间上界呈指数级缩短。

我们的结果 我们分析了上述描述的 mini-batch k-means 算法 [Scu10], ，该算法仅在采样批次的聚类质量提升小于某个阈值参数 ϵ 时才终止。也就是说，如果某个批次更新前后的聚类质量之差小于 ϵ，我们就终止算法。我们的停止条件与实践中使用的条件略有不同。在 sklearn 中，终止取决于簇中心的变化。我们在第 5 节证明，这一条件也符合我们的框架。

我们的主要目标是回答以下理论问题：“对于小批次 k-均值算法，即使学习率不趋于 0，局部进展（在批次上）是否意味着全局进展（在整个数据集上）？”直观地说，答案显然取决于算法使用的批大小。如果批次是整个数据集，那么该结论显然成立，并可得到终止

保证，迭代次数为 $O(d/\epsilon)$[1]。我们证明，当批大小超过某个阈值时，局部进展确实意味着全局进展，并且迭代次数的渐近界与批次为整个数据集时相同。我们给出以下若干结果：

我们先在第 3 节进行预热，证明当 $b = \tilde{\Omega}(kd^3\epsilon^{-2})$时，可以保证在 $O(d/\epsilon)$ 次迭代内终止[2]以高概率（with high probability）[3]。我们还需要额外假设：系统中的每个实数都可以用 $O(1)$ 位表示（例如，64 位浮点数）。上述界限不受簇中心初始化或更新方式的影响。也就是说，这个界限适用于任何中心更新规则，而不仅仅是上文所述的“标准”中心更新规则。我们的证明使用了初等工具，旨在为主要结果铺路。

在第 4 节中，我们证明，采用标准更新规则时，只需更小的批大小，就能达到相同的终止时间。具体来说，批大小为 $\Omega((d/\epsilon)^2\log(nkd/\epsilon)) = \tilde{\Omega}((d/\epsilon)^2)$，就足以保证在 $O(d/\epsilon)$ 次迭代内终止。无论中心点如何初始化，这一结论都成立，而且不需要对表示实数所需的位数作任何假设。我们的证明利用了这样一个事实：当学习率足够小时（但不必趋于 0），标准更新规则会为随机过程带来额外的稳定性。最后，在第 5 节中，我们证明，对于 sklearn 使用的提前停止条件（采用我们的学习率），主要结果同样成立。不过，这会导致批大小更大、终止速度更慢。具体而言，若 $b = \tilde{\Omega}((d/\epsilon)^3k)$，则我们以高概率在 $O((d/\epsilon)^{1.5}\sqrt{k})$ 次迭代内终止。

请注意，要使批大小合理，我们必须要求 $b \le n$，这意味着 $(d/\epsilon)^2\log(nkd/\epsilon) = O(n)$。因此，我们的结果仅适用于 k、d、ϵ 的某个取值范围。这是合理的，因为在实践中，$\epsilon = O(1)$ $d \ll n$ 的情况很常见，而其余参数的影响是对数级的。

解的质量 将 k-means++ 初始化方案应用于我们的结果后，我们在期望意义下得到与全批量算法相同的近似比 $O(\log k)$。k-means++ 的近似保证在初始化阶段就已得到保证（ [AV07] 中的定理 3.1），而初始化后执行 Lloyd 算法只会改善该解。我们证明，在整个执行过程中，全局目标函数以高概率持续下降，这意味着近似保证保持不变。

2 预备知识

本文始终使用有序元组而非集合，记作 $Y = (y_i)_{i\in[\ell]}$，其中 $[\ell] = \{1, \dots, \ell\}$。引用第 i 个元素时，我们写作 y_i 或 $Y[i]$。对元组使用集合记法会很有用，例如 $x \in Y \iff \exists i \in [\ell]$、$x = y_i$ 以及 $Y \subseteq Z \iff \forall i \in [\ell]$、$y_i \in Z$。求和时，我们常写作 $\sum_{x\in Y} g(x)$，这等价于 $\sum_{i=1}^{\ell} g(Y[i])$。

我们借用了 [KMN+04] 中的以下记号。对于每个 $x, y \in \mathbb{R}^d$ 令 $\Delta(x, y) = \|x - y\|^2$。对于每个有限元组 $S \subseteq \mathbb{R}^d$ 和向量 $x \in \mathbb{R}^d$，令 $\Delta(S, x) = \sum_{y\in S} \Delta(y, x)$。

[1]这是因为目标函数的最大值为 d（引理 1）。[2]本文中，波浪号记法表示省略了 n、k、d、ϵ中的对数因子。[3]通常取为某个常数 $p \ge 1$ 的 $1 - 1/n^p$。对于我们的情况，有 $p = 1$；不过，通过将批大小增加一个乘法常数因子，可以任意放大这一数值。

k-means 给定输入 $X = (x_i)_{i=1}^n \subseteq [0,1]^d$ 和参数 k。我们的目标是找到一个由 k 个中心点组成的元组 $\mathcal{C} \subseteq \mathbb{R}^d$，使以下目标函数最小化：

$$\frac{1}{n} \sum_{x \in X} \min_{C \in \mathcal{C}} \Delta(x, C)$$

对每个 $x \in X$，定义函数 f_x： $\mathbb{R}^{k \cdot d} \to \mathbb{R}$，其中 $f_x(\mathcal{C}) = \min_{C \in \mathcal{C}} \Delta(x, C)$。我们可以将 $\mathbb{R}^{k \cdot d}$ 视为由 k个 d维向量组成的元组集合。对于每个元组 $A = (a_i)_{i=1}^{\ell} \subseteq X$，我们还定义如下函数：

$$f_A(\mathcal{C}) = \frac{1}{\ell} \sum_{i=1}^{\ell} f_{a_i}(\mathcal{C})$$

注意， f_X 是我们原来的目标函数。下面给出一个有用的引理：

引理 1. 对于任意由 k 个中心点组成的元组 $\mathcal{C} \subset [0,1]^d$，有 $f_X(\mathcal{C}) \leq d$。

证明。由于 $X, \mathcal{C} \subset [0,1]^d$，有 $\forall x \in X, f_x(\mathcal{C}) \leq \max_{C \in \mathcal{C}} \Delta(x, C) \leq d$。因此$f_X(\mathcal{C})$ $= \frac{1}{n} \sum_{x \in X} f_x(\mathcal{C}) \leq \frac{1}{n} \cdot nd = d$。

□

下面给出以下著名定理：

定理 2（ [Hoe63]）。设 $Y_1, ..., Y_m$ 是相互独立的随机变量，满足 $\forall 1 \leq i \leq m, E[Y_i] = \mu$ 和 $Y_i \in [a_{min}, a_{max}]$。那么

$$Pr\left(\left| \frac{1}{m} \sum_{i=1}^{m} Y_k - \mu \right| \geq \delta \right) \leq 2e^{-2m\delta^2/(a_{max} - a_{min})^2}$$

定理 3（ [Jen06]）。设 ϕ 是凸函数， $y_1, \ldots, y_n$ 是其定义域内的数，权重为 $a_1, \ldots, a_n \in \mathbb{R}^+$。则有：

$$\phi\left(\frac{\sum_{i=1}^n a_i y_i}{\sum_{i=1}^n a_i} \right) \leq \frac{\sum_{i=1}^n a_i \phi(y_i)}{\sum_{i=1}^n a_i}$$

3 预热：一个简单的界

我们先证明一个简单的收敛保证，它不对中心点的更新方式作任何假设。这将为我们在第 4 节中的主要结果铺垫基础；在那里，我们将考察小批量 k-均值算法 [Scu10, PVG+11] 所使用的标准更新规则。

算法 我们分析一种通用的小批量 k-均值算法变体，该算法见算法 1。请注意，这是一类非常宽泛的算法（包括 [Scu10] 提出的广泛使用的算法）。我们所作的唯一假设是：

1. 中心点始终位于 $[0,1]^d$ （包围 X 的凸包）内。
2. 批次从 X 中均匀随机抽取，且允许重复。
3. 当更新中心点对采样批次的解质量没有显著提升时，算法终止。

第 (1) 项和第 (2) 项在理论和实践中都很常见 [Scu10, PVG+11, TM17]。第 (3) 项通常称为“提前停止”条件。提前停止条件在实践中被广泛使用（例如在 sklearn [PVG+11] 中），并且通常会设定迭代次数上限。不过，我们的提前停止条件与实践中使用的条件略有不同。我们将在第 5 节讨论这一差异。

乍看之下，保证无论以何种方式更新中心点都能终止，似乎有些奇怪。不过，如果更新过程退化了，它就不会取得任何进展，此时算法便会终止。

算法 1：通用小批次 k-means$_1$ $\mathcal{C}_1 \subseteq [0,1]^d$ 是一个初始中心元组$_2$ 对于 $i = 1$ 到 ∞
执行$_3$ 从 X 中均匀随机采样 b 个元素，$B_i = (y_1, \ldots, y_b)$（允许重复）$_4$ 更新 $\mathcal{C}_{i+1}$（使得 $\mathcal{C}_{i+1} \subseteq [0,1]^d$）$_5$ 如果 $f_{B_i}(\mathcal{C}_i) - f_{B_i}(\mathcal{C}_{i+1}) < \epsilon$ 则返回 $\mathcal{C}_{i+1}$

算法 1 的终止保证 为了界定此类通用算法的迭代次数，我们需要作出以下假设：系统中的每个实数都可以用 $q = O(1)$ 位表示。这意味着，每组 k 个中心点都可以用 qkd 位表示。因此，可能解的总数有上界 2^{qkd}。这将使我们能够证明，当批大小足够大时，采样批次可充当整个数据集的稀疏化器。具体而言，这意味着对于任意一个包含 k 个中心点的元组 $\mathcal{C}$，均有 $|f_{B_i}(\mathcal{C}) - f_X(\mathcal{C})| < \epsilon/4$。这说明，当批大小足够大时，只需采样一个批次并在该批次上执行 Lloyd 算法即可，无须执行小批次 k-means。尽管如此，这仍是展示我们通用方法并阐明我们在第 4 节中克服哪些挑战的良好起点；我们将在不影响运行时间的前提下，降低所需的批大小。

我们证明 该算法必须以高概率在前 $t = O(d/\epsilon)$ 次迭代内终止。

参数范围 首先，我们定义本节结果成立的参数值范围。回顾一下，n 是输入的大小，k 是中心点的数量，d 是维度，ϵ是终止阈值。在本节的其余部分，假设 $b = \Omega((d/\epsilon)^2(kd + \log(nt)))$。由于 $t = O(d/\epsilon)$，这意味着 $b = \tilde{\Omega}(kd^3\epsilon^{-2})$ 足以使我们的终止保证成立。

我们陈述以下有用的引理：当批大小足够大且 $\mathcal{C}$ 固定时（即与 B_i的选择无关），$f_B(\mathcal{C})$不会离 $f_X(\mathcal{C})$太远。

引理 4. 令 B 为从 X 中有放回地均匀随机选取的包含 b 个元素的元组。对于任意固定的 k个中心点 $\mathcal{C} \subseteq [0,1]^d$，有：$Pr[|f_B(\mathcal{C}) - f_X(\mathcal{C})| \geq \delta] \leq 2e^{-2b\delta^2/d^2}$。

证明。记作 $B = (y_1, \ldots, y_b)$，其中 y_i 是从 X 中有放回地均匀随机选取的随机元素。对于每个这样的 y_i，定义随机变量 $Z_i = f_{y_i}(\mathcal{C})$。对于任意固定的 $\mathcal{C}$，这些新的随机变量均为 IID。还可以得到 $\forall i \in [b], E[Z_i] = \frac{1}{n}\sum_{x \in X} f_x(\mathcal{C}) = f_X(\mathcal{C})$，以及 $f_B(\mathcal{C}) = \frac{1}{b}\sum_{x \in B} f_x(\mathcal{C}) = \frac{1}{b}\sum_{i=1}^{b} Z_i$。

应用 Hoeffding 界（定理 2），取参数 $m = b, \mu = f_X(\mathcal{C})$ $a_{max} - a_{min} \leq d$，我们得到：$Pr[|f_B(\mathcal{C}) - f_X(\mathcal{C})| \geq \delta] \leq 2e^{-2b\delta^2/d^2}$。 □

利用上述结果，我们可以证明每个 B_i 都是 X的稀疏化器。

引理 5. 以高概率，对每个 $i \in [t]$ 以及每组 k 个中心点 $\mathcal{C} \subset [0,1]^d$，都有$|f_{B_i}(\mathcal{C}) - f_X(\mathcal{C})| < \epsilon/4$。

证明。利用引理 4，令 $\delta = \epsilon/4$，并利用 $b = \Omega((d/\epsilon)^2(kd + \log(nt)))$ 这一事实，我们得到：$Pr[|f_B(\mathcal{C}) - f_X(\mathcal{C})| \geq \delta] \leq 2e^{-2b\delta^2/d^2} = 2^{-\Theta(b\delta^2/d^2)} = 2^{-\Omega(kd+\log(nt))}$。

对所有 t 次迭代和所有 2^{qkd} 中心配置应用并合界，可得概率上界为 $2^{-\Omega(kd+\log(nt))} \cdot 2^{qkd} \cdot t = O(1/n)$；其中，渐近符号中关于 b的常数取适当值。

□

下面的引理保证算法取得全局进展。

引理 6. 以高概率有 $\forall i \in [t], f_X(\mathcal{C}_i) - f_X(\mathcal{C}_{i+1}) \geq \epsilon/2$。

证明。记为（符号 $\pm x$ 表示我们加上并减去 x）：

$$f_X(\mathcal{C}_i) - f_X(\mathcal{C}_{i+1}) = f_X(\mathcal{C}_i) \pm f_{B_i}(\mathcal{C}_i) \pm f_{B_i}(\mathcal{C}_{i+1}) - f_X(\mathcal{C}_{i+1}) \geq \epsilon/2$$

根据引理 5，以高概率有 $f_X(\mathcal{C}_i) - f_{B_i}(\mathcal{C}_i) > -\epsilon/4$且 $f_{B_i}(\mathcal{C}_{i+1}) - f_X(\mathcal{C}_{i+1}) > -\epsilon/4$。最后，根据终止条件，有 $f_{B_i}(\mathcal{C}_i) - f_{B_i}(\mathcal{C}_{i+1}) \geq \epsilon$。证明完毕。

□

由于 f_X 有上界 d，因此当 $b = \Omega(kd^3\epsilon^{-2}\log(nd/\epsilon))$ 时，我们以高概率在 $O(d/\epsilon)$ 次迭代内终止。我们在本节陈述主要定理。

定理 7. 对于 $b = \tilde{\Omega}(kd^3\epsilon^{-2})$，算法 1 以高概率在 $O(d/\epsilon)$ 次迭代内终止。

减小批大小 请注意，本节使用的批大小约为第 4 节所需大小的 kd 倍。这个因子是为了对引理 5 中所有可能的 k 个中心点集合应用并合界所必需的。然而，在实际应用引理 5 时，我们只会在迭代 i 中对两个中心点应用它，并设置 $B = B_i$ 和 $\mathcal{C} = \mathcal{C}_i, \mathcal{C}_{i+1}$。一种更直接的方法是仅对 $\mathcal{C}_i$, $\mathcal{C}_{i+1}$ 应用引理 4，这样就能去掉额外的 kd 因子。当 $\mathcal{C} = \mathcal{C}_i$ 时，这种方法可行，因为 B_i 是在确定 $\mathcal{C}_i$ 后采样的；但对于 $\mathcal{C} = \mathcal{C}_{i+1}$ 则会失效，因为 $\mathcal{C}_{i+1}$ 可能依赖于 B_i。在下一节中，我们将展示如何利用学习率足够小这一事实来克服这个挑战。

4 主要结果

在本节中，我们将证明，使用标准中心更新规则时，可以显著改善对批大小的依赖关系。具体而言，我们将证明，大小为 $\tilde{\Omega}((d/\epsilon)^2)$ 的批次足以保证在 $O(d/\epsilon)$ 次迭代内终止。我们也不要求对表示一个实数所需的位数作任何假设。

预备知识部分 对于任意有限元组 $S \subset \mathbb{R}^d$，定义该元组的质心为 $cm(S) = \frac{1}{|S|}\sum_{x \in S} x$。对于任意元组 $S \subset \mathbb{R}^d$ 以及某个簇中心元组 $\mathcal{C} = (\mathcal{C}^\ell)_{\ell \in [k]}$，它意味着对 S 中点的一个划分 $(S^\ell)_{\ell \in [k]}$。具体而言，每个 S^ℓ 都包含 S 中距离 $\mathcal{C}^\ell$ 最近的点，并且每个点 S 都属于唯一的 $\mathcal{C}^\ell$（平局时任意打破）。我们陈述以下有用的观察：

观察 8。固定某个 $A \subseteq X$。设 $\mathcal{C}$ 为一个由 k 个中心点组成的元组，$S = (S^\ell)_{\ell \in [k]}$ 为由 $\mathcal{C}$ 诱导的 A 的划分，且 $\overline{S} = (\overline{S}^\ell)_{\ell \in [k]}$ 为 A的任意其他划分。则有 $\sum_{j=1}^k \Delta(S^j, \mathcal{C}^j) \leq \sum_{j=1}^k \Delta(\overline{S}^j, \mathcal{C}^j)$。

设 $\mathcal{C}_i^j$ 表示第 i次迭代开始时第 j个中心的所在位置。设 $(B_i^\ell)^{\ell\in[k]}$为由 $\mathcal{C}_i$ 诱导的 B_i 的划分，且 $(X_i^\ell)^{\ell\in[k]}$ 为由 $\mathcal{C}_i$诱导的 X 的划分。

我们分析中心点按如下方式更新时的算法 1： $\mathcal{C}_{i+1}^j = (1-\alpha_i^j)\mathcal{C}_i^j + \alpha_i^j cm(B_i^j)$，其中 α_i^j 是学习率。注意， B_i^j 可能为空，在这种情况下， $cm(B_i^j)$ 未定义；不过，学习率的选择保证了在这种情况下 $\alpha_i^j = 0$ （$\mathcal{C}_{i+1}^j = \mathcal{C}_i^j$）。请注意，不同中心点的学习率可能不同，也可能在迭代之间发生变化。在标准的小批次 k-均值算法 [Scu10、PVG+11] 中，学习率会随时间趋于 0。这保证了在大多数合理的停止条件下算法终止。

与之前一样，我们假设算法至少执行 t 次迭代，并且上界为t。我们证明，当批大小足够大时，即使学习率不趋于 0，也能保证终止。具体而言，我们设定 $\alpha_i^j = \sqrt{b_i^j/b}$，其中 $b_i^j = \left|B_i^j\right|$，并要求$b = \Omega((d/\epsilon)^2 \log(ndtk))$。

证明概要 在证明中，我们利用这样一个事实：足够小的学习率能够提高算法的稳定性，从而使我们能够使用比第 3 节小得多的批大小。定义辅助值 $\overline{\mathcal{C}}_{i+1}^j = (1-\alpha_i^j)\mathcal{C}_i^j + \alpha_i^j cm(X_i^j)$。如果在更新时使用整个数据集而非仅使用一个批次，那么这就是第 j 个中心在第 $i+1$ 步的取值。请注意，这只用于分析，不用于算法。

回想一下，在上一节中，我们要求较大的批大小，因为当 $B = B_i$ 且 $\mathcal{C} = \mathcal{C}_{i+1}$ 时，我们无法应用引理 4，这是因为 $\mathcal{C}_{i+1}$ 可能依赖于 B_i。为克服这一难题，我们使用 $\overline{\mathcal{C}}_{i+1}$ 而不是 $\mathcal{C}_{i+1}$。请注意，$\overline{\mathcal{C}}_{i+1}$ 只依赖于 $\mathcal{C}_i$、X，与 B_i 无关（也就是说，我们可以在采样 B_i之前固定其取值）。我们证明，对于所选的学习率，$\overline{\mathcal{C}}_{i+1}$、$\mathcal{C}_{i+1}$足够接近，这意味着 $f_X(\mathcal{C}_{i+1})$ $f_X(\overline{\mathcal{C}}_{i+1})$ 和 $f_{B_i}(\mathcal{C}_{i+1})$ $f_{B_i}(\overline{\mathcal{C}}_{i+1})$ 也足够接近。这使我们能够采用与引理 6 类似的证明，其中 $\overline{\mathcal{C}}_{i+1}$ 充当 $\mathcal{C}_{i+1}$的代理。接下来，我们将这一思路形式化。

首先，我们陈述以下有用的引理：

引理 9（[KMN+04]）。对于任意集合 $S \subseteq \mathbb{R}^d$ 和任意 $C \in \mathbb{R}^d$，都有 $\Delta(S,C) = \Delta(S, cm(S)) + |S|\,\Delta(C, cm(S))$。

引理 10。对于任意 $S \subseteq X$ 和 $C, C' \in [0,1]^d$，都有： $|\Delta(S,C') - \Delta(S,C)| \le 2\sqrt{d}\,|S|\,\|C - C'\|$。

证明。利用引理 9，我们得到 $\Delta(S,C) = \Delta(S, cm(S)) + |S|\,\Delta(cm(S)\ C)$，并且 $\Delta(S,C') = \Delta(S, cm(S)) + |S|\,\Delta(cm(S)\ C')$。因此，有 $|\Delta(S,C') - \Delta(S,C)| = |S| \cdot |\Delta(cm(S)\ C') - \Delta(cm(S)\ C)|$。注意，对于两个向量 $x, y \in \mathbb{R}^d$，都有 $\Delta(x,y) = (x-y)\cdot(x-y)$。接下来使用向量表示法，并对 $|\Delta(cm(S)\ C') - \Delta(cm(S)\ C)|$进行界定。

$$
\begin{aligned}
&\left|\Delta(cm(S), C') - \Delta(cm(S), C)\right| \\
&= \left|(cm(S) - C')\cdot(cm(S) - C') - (cm(S) - C)\cdot(cm(S) - C)\right| \\
&= \left|-2cm(S)\cdot C' + C'\cdot C' + 2cm(S)\cdot C - C\cdot C\right| \\
&= \left|2cm(S)\cdot(C - C') + (C' - C)\cdot(C' + C)\right| \\
&= \left|(C - C')\cdot(2cm(S) - (C' + C))\right| \\
&\le \|C - C'\|\|2cm(S) - (C' + C)\| \le 2\sqrt{d}\|C - C'\|
\end{aligned}
$$

其中，最后一步变换用到了柯西-施瓦茨不等式。 □

首先，我们证明，由于我们对学习率 $\mathcal{C}_{i+1}^j$的选择，$\overline{\mathcal{C}}_{i+1}^j$ 足够接近 e.

引理 11. 对于， $\forall i \in [t]$、 $j \in [k]$、 $\|\mathcal{C}_{i+1}^j - \overline{\mathcal{C}}_{i+1}^j\| \leq \frac{\epsilon}{10\sqrt{d}}$ 以高概率成立。

证明。注意到 $\mathcal{C}_{i+1}^j - \overline{\mathcal{C}}_{i+1}^j = \alpha_i^j(cm(B_i^j) - cm(X_i^j))$。固定某次迭代 i 和某个中心 j。为简化记号，记作： $X' = X_i^j, B' = B_i^j, b' = b_i^j, \alpha' = \alpha_i^j$。尽管 b' 是一个随机变量，在下文中我们将其视为固定值（本质上是以其取值为条件）。由于下文对 b' 的所有取值都成立，因此根据全概率公式，不加条件时也成立。

在证明的其余部分，我们假设 $b' > 0$ （如果 $b' = 0$ ，则结论显然成立）。记 $\{Y_\ell\}_{\ell=1}^{b'}$ 为在 B'中采样得到的点。注意，从 X 中随机采样的元素属于 B' ，当且仅当它属于 X'。由于批次中的元素是从X中均匀随机、有放回地采样的，以元素属于 B' 为条件，意味着该元素在 X'上均匀分布。因此，有 $\forall \ell, E[Y_\ell] = \frac{1}{|X'|}\sum_{x\in X'} x = cm(X')$ 和 $E[cm(B')] = \frac{1}{b'}\sum_{\ell=1}^{b'} E[Y_\ell] = cm(X')$。我们的目标是界定 $Pr[\|cm(B') - cm(X')\| \geq \frac{\epsilon}{10\alpha'\sqrt{d}}]$。注意，只需将每个坐标的偏差界定为 $\epsilon/(10\alpha' d)$，即可保证：

$$\|cm(B') - cm(X')\| = \sqrt{\sum_{\ell=1}^{d}(cm(B')[\ell] - cm(X')[\ell])^2} \leq \sqrt{\sum_{\ell=1}^{d}(\frac{\epsilon}{10\alpha' d})^2} = \frac{\epsilon}{10\alpha'\sqrt{d}}$$

注意，对于单个坐标 ℓ，我们可以使用参数为$\mu = cm(X')[\ell]$、 $a_{max} - a_{min} \leq 1$ 的 Hoeffding 界，并得到：

$$Pr[\left|cm(B')[\ell] - cm(X')[\ell]\right| \geq \frac{\epsilon}{10\alpha' d}] \leq 2 \cdot e^{-\frac{2b'\epsilon^2}{100(\alpha')^2 d^2}}$$

取并合界可得

$$\begin{aligned} &Pr[\|cm(B') - cm(X')\| \geq \frac{\epsilon}{10\alpha'\sqrt{d}}] \\ &\leq Pr[\exists \ell, \left|cm(B')[\ell] - cm(X')[\ell]\right| \geq \frac{\epsilon}{10\alpha' d}] \leq 2d \cdot e^{-\frac{2b'\epsilon^2}{100(\alpha')^2 d^2}} \end{aligned}$$

利用 $\alpha' = \sqrt{b'/b}$ 这一事实，以及 $b = \Omega((d/\epsilon)^2 \log(ntkd))$ 这一事实（对于适当的常数），我们得到上述结果为 $O(1/ntk)$。最后，对所有 t 次迭代以及每次迭代中的所有 k 个中心点取并合界，即可完成证明。 □

现在，我们使用上述引理来界定簇中心彼此接近时的目标函数。

引理 12。固定某个 $A \subseteq X$。以高概率有 $\forall i \in [t], \left|f_A(\overline{\mathcal{C}}_{i+1}) - f_A(\mathcal{C}_{i+1})\right| \leq \epsilon/5$

证明。设 $S = (S^\ell)_{\ell\in[k]}, \overline{S} = (\overline{S}^\ell)_{\ell\in[k]}$ 是由 $\mathcal{C}_{i+1}, \overline{\mathcal{C}}_{i+1}$ 在 A 上诱导的划分。展开该表达式：

$$\begin{aligned} &f_A(\overline{\mathcal{C}}_{i+1}) - f_A(\mathcal{C}_{i+1}) = \frac{1}{|A|}\sum_{j=1}^{k} \Delta(\overline{S}^j, \overline{\mathcal{C}}_{i+1}^j) - \Delta(S^j, \mathcal{C}_{i+1}^j) \\ &\leq \frac{1}{|A|}\sum_{j=1}^{k} \Delta(S^j, \overline{\mathcal{C}}_{i+1}^j) - \Delta(S^j, \mathcal{C}_{i+1}^j) \\ &\leq \frac{1}{|A|}\sum_{j=1}^{k} 2\sqrt{d}\left|S^j\right| \|\overline{\mathcal{C}}_{i+1}^j - \mathcal{C}_{i+1}^j\| \leq \frac{1}{|A|}\sum_{j=1}^{k}\left|S^j\right|\epsilon/5 = \epsilon/5 \end{aligned}$$

其中，第一个不等式由观察 8 得出，第二个不等式由引理 10 得出。最后，我们结合引理 11 和事实 $\sum_{j=1}^{k}\left|S^j\right| = |A|$ 得到结论。用同样的论证，我们还可以得到 $f_A(\mathcal{C}_{i+1}) - f_A(\overline{\mathcal{C}}_{i+1}) \leq \epsilon/5$，从而完成证明。□

从这里开始，我们的证明与第 3 节中的证明有些相似。下面给出一个有用的引理。

引理 13. 以高概率，对每个 $i \in [t]$，以下结论成立：

$$f_X(\overline{\mathcal{C}}_{i+1}) - f_X(\mathcal{C}_{i+1}) \geq -\epsilon/5 \tag{1}$$
$$f_{B_i}(\mathcal{C}_{i+1}) - f_{B_i}(\overline{\mathcal{C}}_{i+1}) \geq -\epsilon/5 \tag{2}$$
$$f_X(\mathcal{C}_i) - f_{B_i}(\mathcal{C}_i) \geq -\epsilon/5 \tag{3}$$
$$f_{B_i}(\overline{\mathcal{C}}_{i+1}) - f_X(\overline{\mathcal{C}}_{i+1}) \geq -\epsilon/5 \tag{4}$$

证明. 前两个不等式由引理 12 得出。后两个不等式由引理 4 得出，只需令 $\delta = \epsilon/5, B = B_i$：

$$Pr[|f_{B_i}(\mathcal{C}) - f_X(\mathcal{C})| \geq \delta] \leq 2e^{-2b\delta^2/d^2} = e^{-\Theta(b\epsilon^2/d^2)} = e^{-\Omega(\log(nt))} = O(1/nt)$$

其中，最后一个不等式源于 $b = \Omega((d/\epsilon)^2\log(nt))$ 这一事实（对某个适当的常数）。上述结论对 $\mathcal{C} = \mathcal{C}_i$ 或 $\mathcal{C} = \overline{\mathcal{C}}_{i+1}$ 均成立。对所有 t 次迭代应用并合界，即可得到所需结果。□

综合以上各项，我们希望对 $f_X(\mathcal{C}_i) - f_X(\mathcal{C}_{i+1})$ 给出下界。我们写出如下内容：

$$\begin{aligned}
&f_X(\mathcal{C}_i) - f_X(\mathcal{C}_{i+1}) = f_X(\mathcal{C}_i) \pm f_{B_i}(\mathcal{C}_i) - f_X(\mathcal{C}_{i+1}) \\
&\geq f_{B_i}(\mathcal{C}_i) - f_X(\mathcal{C}_{i+1}) - \epsilon/5 = f_{B_i}(\mathcal{C}_i) \pm f_{B_i}(\mathcal{C}_{i+1}) - f_X(\mathcal{C}_{i+1}) - \epsilon/5 \\
&\geq f_{B_i}(\mathcal{C}_{i+1}) - f_X(\mathcal{C}_{i+1}) + 4\epsilon/5 \\
&= f_{B_i}(\mathcal{C}_{i+1}) \pm f_{B_i}(\overline{\mathcal{C}}_{i+1}) \pm f_X(\overline{\mathcal{C}}_{i+1}) - f_X(\mathcal{C}_{i+1}) + 4\epsilon/5 \geq \epsilon/5
\end{aligned}$$

其中，第一个不等式由引理 13 中的不等式 (3) 得出（$f_X(\mathcal{C}_i) - f_{B_i}(\mathcal{C}_i) \geq -\epsilon/5$）；第二个不等式由算法的停止条件得出（$f_{B_i}(\mathcal{C}_i) - f_{B_i}(\mathcal{C}_{i+1}) > \epsilon$）；最后一个不等式则由引理 13 中其余的不等式得出。上述结论以高概率对算法的所有迭代均成立。

如第 3 节所述，我们得出，当 $b = \Omega((d/\epsilon)^2\log(knd/\epsilon))$ 时，算法以高概率在 $t = O(d/\epsilon)$ 次迭代内终止。下面给出我们的主要定理。

定理 14. 对于 $b = \tilde{\Omega}((d/\epsilon)^2)$ 和 $\alpha_i^j = \sqrt{b_i^j/b}$，采用中心更新 $\mathcal{C}_{i+1}^j = (1-\alpha_i^j)\mathcal{C}_i^j + \alpha_i^j cm(B_i^j)$ 的算法 1，以高概率在 $O(d/\epsilon)$ 次迭代内终止。

5 应用于 sklearn

在本节中，我们展示了我们的结果与 sklearn 算法实现的相关性。sklearn 中的主要差异在于学习率和停止条件。终止条件[4] 取决于迭代过程中中心点的移动，而不是 f_{B_i} 的值。

[4]该算法的确切参数直接从代码中提取（相关函数为 mini batchconvergence）：
https://github.com/scikit-learn/scikit-learn/blob/baf828ca1/sklearn/cluster/_kmeans.py#L150

具体而言，只要 $\sum_{j\in[k]}\Delta(\mathcal{C}_{i+1}^j,\mathcal{C}_i^j)\geq\epsilon$，我们就会继续迭代，其中 ϵ 是某个容差参数。学习率设为 $\alpha_i^j=\frac{b_i^j}{\sum_{\ell=1}^i b_\ell^j}$。粗略地说，这意味着 $\alpha_i^j\to 0$ 会随时间推移，并保证算法最终终止。

然而，对于我们的收敛保证，我们只要求 $\alpha_i^j=\sqrt{b_i^j/b}$，它不必随时间趋于 0。我们证明，采用我们的学习率和 sklearn 的终止条件时，第 4 节中的证明仍然能推出终止，但速度较慢，且需要更大的批大小。具体而言，如果批大小为 $\tilde{\Omega}(k(d/\epsilon)^3)$，则我们以高概率在 $O((d/\epsilon)^{1.5}\sqrt{k})$ 次迭代内终止。请注意，由于停止条件不同，第 3 节中的结果并未涵盖这一结果。

下面我们证明，只要 sklearn 中的终止条件不成立（$\sum_{j\in[k]}\Delta(\mathcal{C}_{i+1}^j,\mathcal{C}_i^j)\geq\epsilon$），对于适当的参数（$f_{B_i}(\mathcal{C}_i)-f_{B_i}(\mathcal{C}_{i+1})>\epsilon'$其中 $\epsilon'=\epsilon^{1.5}/\sqrt{kd}$），我们的停止条件也不成立。我们陈述以下有用的引理:

引理 15。设 x、$y\in\mathbb{R}^d$、$\alpha\in[0,1]$。则有 $\Delta(x,(1-\alpha)x+\alpha y)=\alpha^2\Delta(x,y)$。

证明。 $\Delta(x,(1-\alpha)x+\alpha y)=\|x-(1-\alpha)x+\alpha y\|^2=\|\alpha x-\alpha y\|^2=\alpha^2\Delta(x,y)$。 □

下面给出本节的主要引理:

引理 16。如果 $\sum_{j\in[k]}\Delta(\mathcal{C}_{i+1}^j,\mathcal{C}_i^j)>\epsilon$ 成立，则 $f_{B_i}(\mathcal{C}_i)-f_{B_i}(\mathcal{C}_{i+1})>\frac{\epsilon^{1.5}}{\sqrt{kd}}$。

证明。回顾可知，对于 $\alpha_i^j=\sqrt{b_i^j/b}$，$\mathcal{C}_{i+1}^j=(1-\alpha_i^j)\mathcal{C}_i^j+\alpha_i^j cm(B_i^j)$。因此，我们得到:

$$\epsilon<\sum_{j\in[k]}\Delta(\mathcal{C}_i^j,\mathcal{C}_{i+1}^j)\leq\sum_{j\in[k]}(\alpha_i^j)^2\Delta(\mathcal{C}_i^j,cm(B_i^j))=\sum_{j\in[k]}\frac{b_i^j}{b}\Delta(\mathcal{C}_i^j,cm(B_i^j)) \tag{5}$$

在推导过程中，我们使用了引理 15。固定任意一个 $j\in[k]$，可写出如下内容:

$$\begin{aligned}
&\Delta(B_i^j,\mathcal{C}_i^j)-\Delta(B_i^j,\mathcal{C}_{i+1}^j)\\
&=\Delta(B_i^j,cm(B_i^j))+b_i^j\Delta(\mathcal{C}_i^j,cm(B_i^j))-\Delta(B_i^j,cm(B_i^j))-b_i^j\Delta(\mathcal{C}_{i+1}^j,cm(B_i^j))\\
&=b_i^j(\Delta(\mathcal{C}_i^j,cm(B_i^j))-\Delta(\mathcal{C}_{i+1}^j,cm(B_i^j)))\\
&=b_i^j(\Delta(\mathcal{C}_i^j,cm(B_i^j))-\Delta((1-\alpha_i^j)\mathcal{C}_i^j+\alpha_i^j cm(B_i^j),cm(B_i^j)))\\
&=b_i^j(\Delta(\mathcal{C}_i^j,cm(B_i^j))-(1-\alpha_i^j)^2\Delta(\mathcal{C}_i^j,cm(B_i^j)))\\
&=(2\alpha_i^j-(\alpha_i^j)^2)b_i^j\Delta(\mathcal{C}_i^j,cm(B_i^j))\geq\alpha_i^j b_i^j\Delta(\mathcal{C}_i^j,cm(B_i^j))=
\end{aligned}$$

其中，在第一次变换中我们应用引理 9，在最后一次变换中我们使用以下事实： $\Delta(\mathcal{C}_{i+1}^j,cm(B_i^j))=(\alpha_i^j)^2\Delta(\mathcal{C}_i^j,cm(B_i^j))$，以及以下事实： $\forall,\alpha_i^j\in[0,1],2\alpha_i^j-(\alpha_i^j)^2\geq\alpha_i^j$。接下来，我们来界定 $f_{B_i}(\mathcal{C}_i)-f_{B_i}(\mathcal{C}_{i+1})$:

$$\begin{aligned}
f_{B_i}(\mathcal{C}_i)-f_{B_i}(\mathcal{C}_{i+1})&\geq\frac{1}{b}\sum_{j=1}^k(\Delta(B_i^j,\mathcal{C}_i^j)-\Delta(B_i^j,\mathcal{C}_{i+1}^j))\\
&\geq\sum_{j=1}^k\frac{\alpha_i^j b_i^j}{b}\Delta(\mathcal{C}_i^j,cm(B_i^j))=\sum_{j=1}^k\left(\frac{b_i^j}{b}\right)^{1.5}\Delta(\mathcal{C}_i^j,cm(B_i^j))
\end{aligned}$$

其中，第一个不等式源于观察 8，第二个不等式源于以下事实： $\forall j \in [k], \Delta(B_i^j, \mathcal{C}_i^j) - \Delta(B_i^j, \mathcal{C}_{i+1}^j) \geq \alpha_i^j b_i^j \Delta(\mathcal{C}_i^j, cm(B_i^j))$；最后一个等式则只需代入 $\alpha_i^j = \frac{b_i^j}{b}$ 并结合即可。我们通过应用詹森不等式完成证明，参数为： $\phi(x) = x^{1.5}$, $y_j = b_i^j/b$ 和 $a_j = \Delta(\mathcal{C}_i^j, cm(B_i^j))$，并结合不等式 (5)。

$$\sum_{j=1}^{k} \left(\frac{b_i^j}{b}\right)^{1.5} \Delta(\mathcal{C}_i^j, cm(B_i^j)) \geq \left(\sum_{j=1}^{k} \Delta(\mathcal{C}_i^j, cm(B_i^j))\right) \cdot \left(\frac{\sum_{j=1}^{k} \frac{b_i^j}{b} \Delta(\mathcal{C}_i^j, cm(B_i^j))}{\sum_{j=1}^{k} \Delta(\mathcal{C}_i^j, cm(B_i^j))}\right)^{1.5}$$
$$\geq \frac{\epsilon^{1.5}}{\sqrt{\sum_{j=1}^{k} \Delta(\mathcal{C}_i^j, cm(B_i^j))}} \geq \frac{\epsilon^{1.5}}{\sqrt{kd}}$$

□

最后，将 $\epsilon' = \frac{\epsilon^{1.5}}{\sqrt{kd}}$ 代入我们的界中，我们得出：如果 $b = \tilde{\Omega}(\epsilon^{-3} d^3 k)$，则迭代次数以高概率被 $O((d/\epsilon)^{1.5}\sqrt{k})$ 所界定。

致谢

作者谨向阅读本工作初稿的 Ami Paz、Uri Meir 和 Giovanni Viglietta 致谢。

本研究得到 JSPS KAKENHI 资助编号 JP21H05850、JP21K17703、JP21KK0204 的支持。

参考文献

[AMR11] David Arthur、Bodo Manthey 和 Heiko R¨oglin。《k-均值方法的平滑分析》。《ACM杂志》，58(5):19:1–19:31，2011年。[AV06] David Arthur 和 Sergei Vassilvitskii。k-均值方法有多慢？载于 SCG，第 144–153 页。ACM，2006年。[AV07] David Arthur 和 Sergei Vassilvitskii。k-均值：谨慎初始化的优势。载于 SODA，第 1027–1035 页。SIAM，2007年。[BB94] L´eon Bottou 和 Yoshua Bengio。k-均值算法的收敛性质。载于 NIPS，第 585–592 页。麻省理工学院出版社，1994年。[Hoe63] Wassily Hoeffding。有界随机变量之和的概率不等式。《美国统计学会会刊》，58(301):13–30，1963年。[Jen06] Johan Ludwig William Valdemar Jensen。论凸函数及均值之间的不等式。《数学学报》，30(1):175–193，1906年。[KMN+04] Tapas Kanungo、David M. Mount、Nathan S. Netanyahu、Christine D. Piatko、Ruth Silverman 和 Angela Y. Wu。用于 k-均值聚类的局部搜索近似算法。《计算几何》，28(2-3):89–112，2004年。[Llo82] Stuart P. Lloyd。PCM 中的最小二乘量化。《IEEE信息论汇刊》，28(2):129–136，1982年。

[PVG+11] F. Pedregosa、G. Varoquaux、A. Gramfort、V. Michel、B. Thirion、O. Grisel、M. Blondel、P. Prettenhofer、R. Weiss、V. Dubourg、J. Vanderplas、A. Passos、D. Cournapeau、M. Brucher、M. Perrot 和 E. Duchesnay。Scikit-learn：Python 机器学习。机器学习研究杂志，12:2825–2830，2011年。[Scu10] D. Sculley。网络规模的 k 均值聚类。载于 WWW，第 1177–1178 页。ACM，2010年。[TM17] Cheng Tang 和 Claire Monteleoni。随机 k 均值的收敛速率。载于 AIS-TATS，《机器学习研究论文集》第 54 卷，第 1495–1503 页。PMLR，2017年。[Vat11] Andrea Vattani。即使在平面上，k-均值也需要指数级多次迭代。《离散与计算几何》，45(4):596–616，2011年。