关于解决 k 中心聚类的方法的调查

姓名：王瀚橙　　学号：55161125
（2016 级本科生 11 班）

摘要：k 中心问题在过去的四十年中不断发展，其研究思路也随着数据的急剧增长不断变化。本文整理了近四十年来关于 k 中心问题的研究成果，分类比较了 MapReduce 和 Streaming 场景下 k 中心算法之间的引证关系与改进，以期对了解 k 中心聚类发展脉络有所助益。

1 引言

聚类是计算机科学中的核心问题之一，在数据挖掘[1]、异常检测[2]等诸多领域有着广泛的应用。k 中心聚类作为聚类问题的一个分支，因其在设备选址[3]问题中的广泛应用而受到大量关注。在实际生活中，从网络中寻找设备的最佳位置的需求有很多，比如工厂选址、人员安排等等。合理的设备选址可降低服务成本，提高系统效率，因而受到了广泛的研究。图一展示了在谷歌学术上搜索关键词“k center”获得的馆藏数量在过去四十年间的变化。在过去的十年里，关于 k 中心问题的研究成果超过了先前研究的总和，达到总计一万四千多件。虽然我们可以使用更加合适的关键词、更加科学的统计方法来搜索，但仅通过下面的折线我们可以直观感受到 k 中心问题研究的热度。

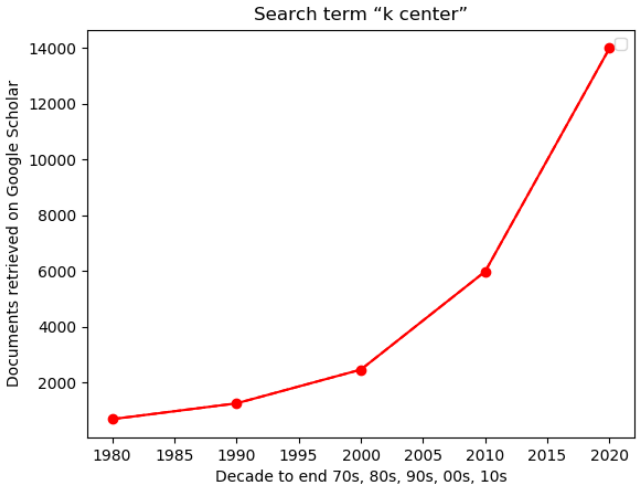

图 1 在谷歌学术搜索“k center”获得的馆藏数目变化

虽然 k 中心问题受到了人们广泛的研究，并开发出了许多算法来解决这个问题，但针对该问题设计既复杂度低又近似度高的解决方案依然十分困难。

我认为造成此困境的原因有二：

一方面 k 中心问题需要在集合 S 中找到一个大小为 k 的子集，并最小化 S 中任意点到这个子集的距离，这本身就是十分复杂的。前人早已证明这是一个 NP Hard 的问题，不太可能在多项式时间内获得精确解。此外，随着现实生活需求的改进以及现实世界固有的复杂性，k 中心问题也出现了很多变式，比如为了应对数据集中的噪音点而引入的可以忽略集合 S 中 z 个离群值的 k 中心问题[3]无疑使问题更加复杂。

另一方面，随着海量数据的产生，先前很多即使复杂度较低的算法也变得不再适用，这直接影响了 k 中心问题的研究思路，越来越多针对大数据的分布式计算模型例如 MapReduce、Streaming 走入人们的视野，如何开发更少轮次、近似度更高、占用内存更少的算法成为了研究的焦点。

我整理了四十年以来关于 k 中心算法的研究成果，以期捋顺 k 中心问题的发展脉络，启发未来 k 中心问题的研究思路。我将 k 中心问题的研究历史粗略分为从上世纪八十年代到本世纪初的早期阶段和本世纪初至今的大数据分布式阶段。两阶段之间并没有明显的界限，但不同时期的主流策略不同，在下文中我将按照时间顺序介绍每个阶段的研究成果。

本文的主要内容和贡献有三：

1） 搜集整理了过去四十年间关于 k 中心问题的主要研究成果，按照时间顺序描述了 k 中心问题的发展历史。

2） 分类比较了近期海量数据背景下，使用 MapReduce、Streaming 解决 k 中心问题的算法之间的优劣与改进。

3） 将人们在大数据聚类的成果进行分类总结，分析未来研究的发展趋势。

文章的剩余部分安排如下。第二节回顾数据量较小的情况下人们在 k 中心问题上所作的研究。第三节分类对比大数据背景下的分布式算法之间的优劣与不足。第四节分析未来发展的趋势。

2 k 中心问题的早期研究成果

早期的 k 中心问题并不考虑舍弃离群点的问题，k 中心问题可以理解为使用最小的集合覆盖所有的点，这是一种最小集合覆盖问题，可以使用线性规划的方法来解决。早在上世纪七十年代，MINIEKA[4]首先使用线性规划来解决 k 中心问题。其后，ELLOUMI 等人[5]提出了更高效的线性规划解法。ELLOUMI 通过求解$O(\log(MN))$的线性规划程序，在多项式时间内解决从 M 个设备中选取 k 中心分配给 N 个客户，并最小化 N 个客户距离其设备的距离这一问题的近似解。虽然在今天看来这个复杂度根本无法接受，但在当时该算法第一次解决了 N=M=1817 数量级的问题。时至今日，依然有使用线性规划解决 k 中心问题的研究，Chakrabarty 等人[6]提出了一种可以解决带有离群值的 k 中心问题的二近似解法，但是该算法需要考虑一个复杂的线性规划模型。

此外，贪婪算法也被应用于 k 中心问题的求解。早在 80 年代，Gonzalez 等人[7]以及 HOCHBAUM 等人[8]均使用贪心法解决 k 中心问题，他们均针对三角不等式的 k 中心问题提出了二近似的算法。前者针对k中心问题开发了一种$O(k\,|S|)$复杂度的二近似顺序算法，该算法后被应用于分布式解决 k 中心问题[9]。Gonzalez 在该论文中表明，对于固定的$\varepsilon > 0$，在基本度量空间中，除非$P = NP$，否则不可能获得近似因子$2 - \varepsilon$。HOCHBAUM 可以在$O(|E|\log|E|)$的时间复杂度中获得最优解两倍以内的近似算法，该算法作为众所周知的经典算法后来被用于 Ding 等人[10]的分布式解法研究论文中的 baseline。

此外，搜索类的方法也被应用于解决此类问题。Mladenovic 等人[11]使用了禁忌搜索的方法，Gupta 等人[12]使用局部搜索的方法提出了$O(\log n)$近似的 k 中心算法。

表 1 k 中心问题早期解法整理

	1980 前	2000 前	2000 后
线性规划	[4]		[5][6]
贪婪算法		[8][7]	
搜索算法			[11][12]

表一根据上文的分类按照时间顺序整理了这些算法，Mihelic 等人[13]的研究表明贪婪算法在当时的实际表现并不突出，但是因为其他的算法难以应对如今海量数据分布式解法并行化的需求，反而 Gonzalez 等人[7]的贪婪算法后来被广泛应用于现如今的分布式场景中。

3 大数据时代 k 中心问题的分布式解法

本世纪初，出现了越来越多考虑离群点的 k 中心问题。k 中心问题需要最小化点集 S 中任意点到中心的距离，因而非常容易受到几个离群点的影响，表现在现实生活中就是为了迁就几个偏远的客户而影响设施位置的选择。然而，现实生活中，离群点是许多数据集中固有的存在，这些数据可能是收集时人为造成的，也可能是现实存在的噪音。为了解决这个问题，k中心使用了一种考虑了离群点的变式：当我们计算目标函数时，最多可以舍弃z个点，其中z是用户自己输入的参数。此外，为应对海量的数据，Streaming 和 MapReduce 的算法受到人们的关注。

作为带有异常值的 Streaming 场景下 k 聚类的知名研究者，Charikar 等人[3]首先提出了带有异常值的 k 中心问题的三逼近算法，他们提供了一种$O(k\,|S|^2\log|S|)$复杂度的三近似算法。并且他们证明，对于固定的$\varepsilon > 0$，在基本度量空间中，除非$P = NP$，否则不可能获得近似因子$3 - \varepsilon$。这个著名的算法被广泛引用，其后很多分布式解法都使用了这个算法。随后，Charikar 等人[14]又表明带有离群值的优化是十分复杂的，但如果我们放宽限制，被允许忽略更多的异常值，可极大降低计算成本。这种松弛的思想后来被广泛应用到此类问题的改进，比如 Guo 等人[15]为(k, z)中心问题提出了$a\big(24(1+\varepsilon),(1+\varepsilon)\big)$四轮近似解。之后，随着数据量的加大，增量聚类被应用于此领域，Charikar 等人[16]又提出了高效

的一遍 Streaming 算法，该算法仅仅需要$O(k)$的工作内存，就可以确定性的计算 8 近似、概率性的计算 5.43 近似的结果。受到增量聚类的启发，McCutchen 等人[17]使用少量内存维护当前已经选择出来的中心点集，提出了带有离群点的 4 近似的 Streaming 算法，该算法仅需要$O(kz\varepsilon^{-1})$工作内存，适应于离群值数目不多的情况。之后，Matteo 等人[9]基于 Charikar 的一遍 Streaming 算法，使用更大的核集这一思路，改进出了确定的一遍，$3+\varepsilon$近似有 z 离群点的k中心的 Streaming 算法，空间复杂度为$O\left((k+z)(96/\varepsilon)^D\right)$。

与 Streaming 框架对应的另一框架是 MapReduce，Ene 等人[18]最早展开了 MapReduce 上 k 中心问题的研究，虽然他的工作没有考虑离群点的问题，但他们提出了 10 近似的随机采样算法，可是正如 Ene 所说，随机采样造成该算法的实际表现并不好。Malkomes 等人[19]基于抛弃了先前的随机采样的方法，使用可组合核集[20]改进了之前的方法，针对 k 中心问题提出了 4 近似的算法，并且他们提出了带有离群值的 13 近似算法。Matteo 等人[9]也是基于可组合核集的思路，他们在 Malkomes 研究的基础上，使用更大的核集，针对不带有异常值的 k 中心问题提出了 2 轮 2 近似算法。针对带有异常值的问题，他们提出了空间复杂度为$O\left(\sqrt{|S|(k+z)}(24/\varepsilon)^D\right)$的确定性 2 轮 3 近似算法以及空间复杂度为$O\left(\left(\sqrt{|S|(k+\log|S|)}+z\right)(24/\varepsilon)^D\right)$的概率保证的 2 轮 3 近似算法。

事实上核集的应用很早就有，早在 2002 年，Badoiu 等人[21]就使用核集来解决 k 中心问题的解法，但是他的核集构造十分复杂。近年来，核集已经成为解决大数据集上的优化问题的重要工具，可能成为开发 MapReduce 和 Streaming 框架上高效算法的关键要素。并且现在出现了将核集应用于 MapReduce 等分布式框架解决其他聚类问题的应用。Pietracaprina 等人[22]将这一思路迁移应用到了鲁棒中心 RMC 和鲁棒背包中心 RKC 问题之中。未来可能有更多针对基于中心的聚类的类似算法出现。

总而言之，k 中心问题的分布式解法趋向于使用更少的轮次，更少的内存实现更高的近似度。我们将近 20 年的分布式解决 k 中心问题的成果总结如下表二所示：

表 2 k 中心问题分布式解法整理

	不带有异常值	带有异常值
Streaming	[17]	[17][9][3,14,16]
MapReduce	[18][9][21]	[19][9][10][15]

4 总结

为了加快聚类算法的运行效率，Aggarwal 等人[23]认为过去主要从以下三个方面来提高算法效率：

1） 开发一遍算法，比如上文提到的 Charikar 等人[16]的一遍 Streaming 算法，该算法仅仅需要$O(k)$的工作内存，就可以确定性的计算 8 近似、概率性的计算 5.43 近似的结果，相比其他多轮次的算法极大提高了运行效率。

2） 通过随机化的手段，采样或者核集等方法，降低数据量。

3） 通过使用分布式并行化的计算，加快计算速度。

回顾过去四十年中的发展，应对爆炸性增长的数据量，未来的算法将会综合应用这三类思想，以期计算速度获得进一步提升。

参考文献

[1] Hennig C. Handbook of Cluster Analysis[M]. Chapman and Hall/CRC, 2015.

[2] Chandola V, Banerjee A, Kumar V. Anomaly detection: A survey[J]. ACM computing surveys (CSUR), ACM New York, NY, USA, 2009, 41(3): 1–58.

[3] Charikar M, Khuller S, Mount D M等. Algorithms for facility location problems with outliers[R]. 2001.

[4] Minieka E. The m-center problem[J]. Siam Review, SIAM, 1970, 12(1): 138–139.

[5] Elloumi S, Labbé M, Pochet Y. A new formulation and resolution method for the p-center problem[J]. INFORMS Journal on Computing, INFORMS, 2004, 16(1): 84–94.

[6] Chakrabarty D, Goyal P, Krishnaswamy R. The Non-Uniform k-Center Problem[C]//43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016. 2016: 67:1–67:15.

[7] Gonzalez T F. Clustering to minimize the maximum intercluster distance[J]. Theoretical Computer Science, Elsevier, 1985, 38: 293–306.

[8] Hochbaum D S, Shmoys D B. A best possible heuristic for the k-center problem[J]. Mathematics of operations research, INFORMS, 1985, 10(2): 180–184.

[9] Ceccarello M, Pietracaprina A, Pucci G. Solving k-center clustering (with outliers) in MapReduce and streaming, almost as accurately as sequentially[J]. Proceedings of the VLDB Endowment, 2019, 12(7): 766–778.

[10] Ding H, Yu H, Wang Z. Greedy Strategy Works for k-Center Clustering with Outliers and Coreset Construction[C]//27th Annual European Symposium on Algorithms (ESA 2019). 2019.

[11] Mladenović N, Labbé M, Hansen P. Solving the p-center problem with tabu search and variable neighborhood search[J]. Networks: An International Journal, Wiley Online Library, 2003, 42(1): 48–64.

[12] Gupta A, Tangwongsan K. Simpler analyses of local search algorithms for facility location[J]. arXiv preprint arXiv:0809.2554, 2008.

[13] Robič B, Mihelič J. Solving the k-center problem efficiently with a dominating set algorithm[J]. Journal of computing and information technology, SRCE-Sveučilišni računski centar, 2005, 13(3): 225–234.

[14] Charikar M, O'Callaghan L, Panigrahy R. Better streaming algorithms for clustering problems[C]//Proceedings of the thirty-fifth annual ACM symposium on Theory of computing. 2003: 30–39.

[15] Guo X, Li S. Distributed k-Clustering for Data with Heavy Noise[J]. Advances in Neural Information Processing Systems, 2018, 2018-Decem: 7838–7846.

[16] Charikar M, Chekuri C, Feder T等. Incremental clustering and dynamic information retrieval[J]. SIAM Journal on Computing, SIAM, 2004, 33(6): 1417–1440.

[17] Matthew McCutchen R, Khuller S. Streaming Algorithms for k-Center Clustering with Outliers and with Anonymity[G]//Approximation, Randomization and Combinatorial Optimization.

Algorithms and Techniques. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, 5171 LNCS: 165–178.

[18] Ene A, Im S, Moseley B. Fast clustering using MapReduce[C]//Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD ’11. New York, New York, USA: ACM Press, 2011: 681.

[19] Malkomes G, Kusner M J, Chen W等. Fast distributed k-center clustering with outliers on massive data[R]. 2015, 2015-Janua.

[20] Indyk P, Mahabadi S, Mahdian M等. Composable core-sets for diversity and coverage maximization[C]//Proceedings of the 33rd ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems. 2014: 100–108.

[21] Bādoiu M, Har-Peled S, Indyk P. Approximate clustering via core-sets[C]//Proceedings of the thiry-fourth annual ACM symposium on Theory of computing - STOC ’02. New York, New York, USA: ACM Press, 2002: 250.

[22] Pietracaprina A, Pucci G, Soldà F. Coreset-based Strategies for Robust Center-type Problems[J]. arXiv preprint arXiv:2002.07463, 2020.

[23] Aggarwal C C. Data classification: algorithms and applications[M]. CRC press, 2014.