ДОКЛАДЫ АКАДЕМИИ НАУК, *2019*, *том 487*, *№ 2*, *с. 126–129*

МАТЕМАТИКА

УДК 519.16+519.85

О ПОЛИНОМИАЛЬНОЙ РАЗРЕШИМОСТИ ОДНОЙ КВАДРАТИЧНОЙ ЕВКЛИДОВОЙ ЗАДАЧИ КЛАСТЕРИЗАЦИИ В ОДНОМЕРНОМ СЛУЧАЕ

А. В. Кельманов[1,2,*], В. И. Хандеев[1,2,]**

Представлено академиком РАН Ю.И. Журавлевым 28.02.2019 г.

Поступило 06.03.2019 г.

Рассматривается задача разбиения конечного множества точек евклидова пространства на кластеры. В этой задаче требуется минимизировать сумму по всем кластерам внутрикластерных сумм квадратов расстояний от элементов кластеров до их центров. Центры некоторых кластеров заданы на входе, а центры других кластеров определяются как центроиды (геометрические центры). Известно, что в общем случае задача NP-трудна в сильном смысле. Мы доказываем, что существует точный полиномиальный алгоритм для одномерного случая задачи.

Ключевые слова: кластеризация по критерию минимума суммы квадратов, евклидово пространство, NP-трудная задача, одномерный случай, полиномиальная разрешимость.

DOI: https://doi.org/10.31857/S0869-56524872126-129

Предметом исследования этой работы является одна из труднорешаемых задач разбиения конечного множества точек евклидова пространства. Цель исследования — выяснение вопроса о статусе вычислительной сложности одномерного случая задачи. Наше исследование мотивировано открытостью указанного математического вопроса, а также важностью задачи для приложений, в частности, для прикладных проблем анализа данных (Data analysis), интерпретации данных (Data mining), распознавания образов (Pattern recognition), машинного обучения (Machine learning) и обработки большеразмерных данных (Big data processing).

1. ФОРМУЛИРОВКА ЗАДАЧИ, ЕЁ ИСТОКИ И СВЯЗАННЫЕ С НЕЙ ПРОБЛЕМЫ

В известной кластеризационной задаче *K-Means* дано N-элементное множество $\mathcal{Y}$ точек в евклидовом пространстве размерности d и натуральное число K. Требуется найти разбиение входного множества $\mathcal{Y}$ на непустые непересекающиеся кластеры $\mathcal{C}_1, ..., \mathcal{C}_K$, минимизирующее сумму

$$\sum_{k=1}^{K} \sum_{y \in \mathcal{C}_k} \|y - \bar{y}(\mathcal{C}_k)\|^2,$$

где $\bar{y}(\mathcal{C}_k) = \frac{1}{|\mathcal{C}_k|} \sum_{y \in \mathcal{C}_k} y$ — центроид k-го кластера.

Другое распространённое название задачи *K-Means* — MSSS (Minimum Sum-of-Squares Clustering), т.е. кластеризация по критерию минимума суммы квадратов. В статистике эта задача известна с прошлого века и связана с именем Фишера (см., например, [1, 2]). На практике (в самых разнообразных приложениях) она возникает в ситуациях, когда имеется гипотеза о том, что имеющаяся совокупность $\mathcal{Y}$ выборочных (входных) данных содержит K однородных кластеров (подмножеств) $\mathcal{C}_1, ..., \mathcal{C}_K$, в которых точки разбросаны относительно соответствующих неизвестных средних значений $\bar{y}(\mathcal{C}_1), ..., \bar{y}(\mathcal{C}_K)$. Однако соответствие данных и кластеров неизвестно. Очевидно, что в этой ситуации для корректного применения классических статистических методов (проверки гипотез и оценивания) к обработке выборочных данных требуется сначала разбить их на однородные группы (кластеры). Эта ситуация типична, в частности, для отмеченных выше прикладных проблем.

Сильная NP-трудность задачи *K-Means* установлена относительно недавно [3]. Полиномиальная разрешимость этой задачи на числовой прямой доказана ещё в прошлом веке в [4]. В этой работе представлен алгоритм, реализующий схему динамиче-

[1] *Институт математики им. С.Л. Соболева Сибирского отделения Российской Академии наук, Новосибирск*

[2] *Новосибирский национальный исследовательский государственный университет*

**E-mail: kelm@math.nsc.ru*

***E-mail: khandeev@math.nsc.ru*

ского программирования. Он позволяет находить точное решение задачи в одномерном случае за время $\mathcal{O}(KN^2)$. Этот алгоритм опирается на точный полиномиальный алгоритм решения известной задачи о ближайшем соседе (Nearest neighbor search problem) [5]. В последние годы для одномерной задачи *K-Means* построен ряд точных алгоритмов, имеющих улучшенные показатели быстродействия. Обзор ускоренных алгоритмов и их свойств можно найти в [6].

Объектом нашего внимания является близкая по постановке к задаче *K-Means* слабоизученная

З а д а ч а 1 (*K-Means and Given J-Centers*). Д а н о: N-элементное множество $\mathcal{Y}$ точек в евклидовом пространстве размерности d, натуральное число K и набор $\{c_1, ..., c_J\}$ точек. Н а й т и: разбиение множества $\mathcal{Y}$ на $K+J$ непустых кластеров $\mathcal{C}_1, ..., \mathcal{C}_K, \mathcal{D}_1, ..., \mathcal{D}_J$ такое, что

$$F = \sum_{k=1}^{K} \sum_{y \in \mathcal{C}_k} \|y - \bar{y}(\mathcal{C}_k)\|^2 + \sum_{j=1}^{J} \sum_{y \in \mathcal{D}_j} \|y - c_j\|^2 \to \min,$$

где $\bar{y}(\mathcal{C}_k)$ — центроид k-го кластера.

Эту задачу можно рассматривать как модификацию задачи *K-Means*. С другой стороны, введённые обозначения позволяют назвать задачу 1 как *K-Means and Given J-Centers*.

В отличие от задачи *K-Means*, задача 1 моделирует прикладную проблему, в которой у части кластеров, а именно у $\mathcal{D}_1, ..., \mathcal{D}_J$, соответствующие центры $c_1, ..., c_J$ квадратичного разброса данных известны заранее (заданы). Эта прикладная проблема также типична для анализа и интерпретации данных, распознавания образов и машинного обучения. В частности, двухкластерный вариант задачи — 1-*Mean and Given* 1-*Center* — связан с решением прикладной проблемы обработки сигналов, а именно с проблемой совместного обнаружения квазипериодически повторяющегося импульса неизвестной формы и оценивания этой формы в условиях гауссовского шума с нулевым средним [7–9]. В этом двухкластерном варианте задачи нулевое среднее соответствует кластеру с заданным в начале координат центром. Первое сообщение об этом двухкластерном варианте задачи 1, по-видимому, было сделано в [7]. Заметим, что более простые экстремальные задачи, которые индуцируются прикладными проблемами помехоустойчивого обнаружения и различения импульсов заданных форм, характерны, в частности, для радиолокации, электронной разведки, гидроакустики, геофизики, технической и медицинской диагностики, космического мониторинга (см., например, [10–12]).

Сильная NP-трудность задачи 1 установлена в [13–15]. Вопрос о разрешимости этой задачи на числовой прямой до последнего времени оставался открытым.

Основной результат настоящей работы — доказательство полиномиальной разрешимости задачи 1 в одномерном случае.

2. ВСПОМОГАТЕЛЬНЫЕ УТВЕРЖДЕНИЯ

Наше доказательство опирается на несколько приведённых ниже вспомогательных утверждений, которые раскрывают структуру оптимального решения задачи 1. Для краткости мы приводим эти утверждения без доказательства, ограничиваясь изложением их идей.

Обозначим через $\mathcal{C}_1^*, ..., \mathcal{C}_K^*, \mathcal{D}_1^*, ..., \mathcal{D}_J^*$ оптимальные кластеры в задаче 1.

Л е м м а 1. *Пусть $d = 1$ в задаче* 1. *Тогда если $c_m < c_\ell$, где $1 \le m \le J, 1 \le \ell \le J$, то для любых $x \in \mathcal{D}_m^*$ и $z \in \mathcal{D}_\ell^*$ выполнено неравенство $x \le z$.*

Л е м м а 2. *Пусть $d = 1$ в задаче* 1. *Тогда*:

1) *если $\bar{y}(\mathcal{C}_m^*) < c_\ell$, где $1 \le m \le K$, $1 \le \ell \le J$, то для любых $x \in \mathcal{C}_m^*$ и $z \in \mathcal{D}_\ell^*$ справедливо неравенство $x \le z$;*

2) *если $\bar{y}(\mathcal{C}_m^*) > c_\ell$, где $1 \le m \le K$, $1 \le \ell \le J$, то для любых $x \in \mathcal{C}_m^*$ и $z \in \mathcal{D}_\ell^*$ выполнено неравенство $x \ge z$.*

Л е м м а 3. *Пусть $d = 1$ в задаче* 1. *Тогда если $\bar{y}(\mathcal{C}_m^*) < \bar{y}(\mathcal{C}_\ell^*)$, где $1 \le m \le K$, $1 \le \ell \le K$, то для любых $x \in \mathcal{C}_m^*$ и $z \in \mathcal{C}_\ell^*$ справедливо неравенство $x \le z$.*

Доказательство лемм 1–3 проводится методом от противного с применением равенства

$$(x - c_m)^2 + (z - c_\ell)^2 = \\ = 2(x - z)(c_\ell - c_m) + (z - c_m)^2 + (x - c_\ell)^2,$$

справедливость которого следует из известной формулы для суммы квадратов диагоналей трапеции.

Л е м м а 4. *Если $d = 1$ в задаче* 1, *то для каждого $k \in \{1, 2, ..., K\}$ и каждого $j \in \{1, 2, ..., J\}$ справедливо $\bar{y}(\mathcal{C}_k^*) \ne c_j$.*

Л е м м а 5. *Если $d = 1$ в задаче* 1, *то для каждых $k, j \in \{1, 2, ..., K\}$ таких, что $k \ne j$, справедливо $\bar{y}(\mathcal{C}_k^*) \ne \bar{y}(\mathcal{C}_j^*)$.*

Доказательство этих лемм также проводится методом от противного.

Леммы 1–5 устанавливают взаимное расположение на числовой прямой оптимальных кластеров $\mathcal{D}_1^*, ..., \mathcal{D}_J^*$ с заданными центрами и кластеров

$\mathcal{C}_1^*, \ldots, \mathcal{C}_K^*$ с неизвестными центрами. Эти леммы служат базой для доказательства следующего утверждения.

Теорема 1. *Пусть* $d = 1$ *в задаче* 1 *и, кроме того, точки* $y_1, \ldots, y_N$ *входного множества* $\mathcal{Y}$, *а также точки* $c_1, \ldots, c_J$ *упорядочены так, что*

$$y_1 < \ldots < y_N,$$
$$c_1 < \ldots < c_J.$$

Тогда оптимальному разбиению $\mathcal{Y}$ *на кластеры* $\mathcal{C}_1^*, \ldots, \mathcal{C}_K^*, \mathcal{D}_1^*, \ldots, \mathcal{D}_J^*$ *соответствует разбиение последовательности* 1, 2, ..., N *натуральных чисел на непересекающиеся целочисленные отрезки*.

3. ПОЛИНОМИАЛЬНАЯ РАЗРЕШИМОСТЬ ЗАДАЧИ В ОДНОМЕРНОМ СЛУЧАЕ

Следующая теорема является основным результатом работы.

Теорема 2. *Существует полиномиальный алгоритм, который при* $d = 1$ *находит оптимальное решение задачи* 1 *за время* $\mathcal{O}(KJN^2)$.

Доказательство теоремы проводится конструктивно, а именно: мы даём обоснование алгоритма, который реализует схему динамического программирования и позволяет находить точное решение задачи за полиномиальное время.

Идея доказательства состоит в следующем. Без ограничения общности мы полагаем, что точки $y_1, \ldots, y_N$ входного множества $\mathcal{Y}$, а также точки $c_1, \ldots, c_J$ упорядочены, как в теореме 1.

Пусть $\mathcal{Y}_{s,t} = \{y_s, \ldots, y_t\}$, где $1 \le s \le t \le N$, — подмножество из $t - s + 1$ точек входного множества $\mathcal{Y}$ с номерами от s до t.

Положим

$$f_{s,t}^j = \sum_{i=s}^{t} (y_i - c_j)^2, \quad j = 1, 2, \ldots, J,$$

$$f_{s,t} = \sum_{i=s}^{t} (y_i - \bar{y}(\mathcal{Y}_{s,t}))^2,$$

где $\bar{y}(\mathcal{Y}_{s,t})$ — центроид подмножества $\mathcal{Y}_{s,t}$.

Мы доказываем, что оптимальное значение F^* целевой функции задачи 1 находится по формуле

$$F^* = F_{K,J}(N),$$

а значения функции

$$F_{k,j}(n), \quad k = -1, 0, 1, \ldots, K, \quad j = -1, 0, 1, \ldots, J, \quad n = 0, 1, \ldots, N,$$

вычисляются по нижеприведённым рекуррентным формулам. При этом формула

$$F_{k,j}(n) = \begin{cases} 0, & n = k = j = 0; \\ +\infty, & n = 0;\ k = 0, 1, \ldots, K; \\ & j = 0, 1, \ldots, J;\ k + j \ne 0; \\ +\infty, & k = -1;\ j = -1, 0, \ldots, J; \\ & n = 0, 1, \ldots, N; \\ +\infty, & j = -1;\ k = -1, 0, \ldots, K; \\ & n = 0, 1, \ldots, N; \end{cases} \qquad (1)$$

задаёт начальные и граничные условия для последующих вычислений. Основная формула

$$F_{k,j}(n) = \min\left\{ \min_{i=1}^{n} \{F_{k-1,j}(i-1) + f_{i,n}\}, \min_{i=1}^{n} \{F_{k,j-1}(i-1) + f_{i,n}^j\} \right\},$$

$$k = 0, 1, \ldots, K;\ j = 0, 1, \ldots, J;\ n = 1, 2, \ldots, N, \quad (2)$$

определяет рекурсию. В целом формулы (1), (2) реализуют прямой ход алгоритма.

Далее мы доказываем, что оптимальные кластеры $\mathcal{C}_1^*, \ldots, \mathcal{C}_K^*, \mathcal{D}_1^*, \ldots, \mathcal{D}_J^*$ находятся с помощью следующего рекуррентного правила, которое реализуется на обратном ходе алгоритма. Пошаговая запись правила:

Шаг 0. $k := K$, $j := J$, $n := N$.

Шаг 1. Если

$$\min_{i=1}^{n} (F_{k-1,j}(i-1) + f_{i,n}) \le \min_{i=1}^{n} (F_{k,j-1}(i-1) + f_{i,n}^j),$$

то

$$\mathcal{C}_k^* = \{y_{i^*}, y_{i^*+1}, \ldots, y_n\},$$

где

$$i^* = \arg\min_{i=1}^{n} (F_{k-1,j}(i-1) + f_{i,n}),$$

выполнить $k := k - 1$; $n := i^* - 1$.

Если же

$$\min_{i=1}^{n} (F_{k-1,j}(i-1) + f_{i,n}) > \min_{i=1}^{n} (F_{k,j-1}(i-1) + f_{i,n}^j),$$

то

$$\mathcal{D}_j^* = \{y_{i^*}, y_{i^*+1}, \ldots, y_n\},$$

где

$$i^* = \arg\min_{i=1}^{n} (F_{k,j-1}(i-1) + f_{i,n}^j),$$

выполнить $j := j - 1$; $n := i^* - 1$.

Ш а г 2. Если $k > 0$ или $j > 0$, то переход на шаг 1, иначе — конец вычислений.

Справедливость этого правила доказывается по индукции.

Наконец мы доказываем, что время работы алгоритма есть величина $\mathcal{O}(KJN^2)$, т.е. алгоритм полиномиален. Справедливость этой оценки времени работы алгоритма следует из того, что вычисления по формуле (2) производятся $\mathcal{O}(KJN)$ раз, причём для каждого вычисления требуется $\mathcal{O}(N)$ операций.

Таким образом, NP-трудная в сильном смысле задача 1 в одномерном случае разрешима за полиномиальное время. Построение приближённых эффективных алгоритмов с гарантированными оценками точности для общего случая представляется делом ближайшей перспективы.

Источники финансирования. Разделы 2, 3 выполнены при финансовой поддержке РФФИ, проекты 19–01–00308 и 18–31–00398, остальные разделы — при поддержке программы ФНИ РАН, проект 0314-2019-0015.

СПИСОК ЛИТЕРАТУРЫ

1. *Fisher R.A.* Statistical Methods and Scientific Inference. N.Y.: Hafner, 1956.
2. *MacQueen J.B.* Some Methods for Classification and Analysis of Multivariate Observations. In: Proc. 5th Berkeley Symp. on Mathematical Statistics and Probability. Berkeley: Univ. of California Press, 1967. V. 1. P. 281–297.
3. *Aloise D.*, *Deshpande A.*, *Hansen P.*, *Popat P.* NP-hardness of Euclidean Sum-of-Squares Clustering // Mach. Learn. 2009. V. 75. № 2. P. 245–248.
4. *Rao M.* Cluster Analysis and Mathematical Programming // J. Amer. Stat. Assoc. 1971. V. 66. P. 622–626.
5. *Bellman R.* Dynamic Programming. Princeton: Princeton Univ. Press, 1957.
6. *Grønlund A.*, *Larsen K.G.*, *Mathiasen A.*, *Nielsen J.S.*, *Schneider S.*, *Song M.* Fast Exact k-Means, k-Medians and Bregman Divergence Clustering in 1D. CoRR arXiv:1701.07204 (2017).
7. *Кельманов А.В.*, *Хамидуллин С.А.*, *Кельманова М.А.* Совместное обнаружение и оценивание повторяющегося фрагмента в зашумленной числовой последовательности при заданном числе повторов // Тез. докл. Рос. конф. “Дискретный анализ и исследование операций” (ДАИО-4). Новосибирск: Изд-во ИМ СО РАН, 2004. С. 185.
8. *Gimadi E.Kh.*, *Kel’manov A.V.*, *Kel’manova M.A.*, *Khamidullin S.A.* A Posteriori Detection of a Quasi Periodic Fragment in Numerical Sequences with Given Number of Recurrences // Sib. J. Industrial Math. 2006. V. 9. № 1 (25). P. 55–74.
9. *Gimadi E.Kh.*, *Kel’manov A.V.*, *Kel’manova M.A.*, *Khamidullin S.A.* A Posteriori Detecting a Quasiperiodic Fragment in a Numerical Sequence // Pattern Recogn. and Image Anal. 2008. V. 18. № 1. P. 30–42.
10. *Kel’manov A.V.*, *Khamidullin S.A.* Posterior Detection of a Given Number of Identical Subsequences in a Guasi-Periodic Sequence // Comput. Math. Math. Phys. 2001. V. 41. № 5. P. 762–774.
11. *Kel’manov A.V.*, *Jeon B.* A Posteriori Joint Detection and Discrimination of Pulses in a Quasiperiodic Pulse Train // IEEE Trans. Signal Processing. 2004. V. 52. № 3. P. 645–656.
12. *Carter J.A.*, *Agol E.*, *et al.* Kepler-36: A Pair of Planets with Neighboring Orbits and Dissimilar Densities // Science. 2012. V. 337. № 6094. P. 556–559.
13. *Кельманов А.В.*, *Пяткин И.В.* // ДАН. 2009. Т. 471. № 5. С. 590–592.
14. *Kel’manov A.V.*, *Pyatkin A.V.* On a Version of the Problem of Choosing a Vector Subset // J. Appl. Ind. Math. 2009. V. 3. № 4. P. 447–455.
15. *Kel’manov A.V.*, *Pyatkin A.V.* Complexity of Certain Problems of Searching for Subsets of Vectors and Cluster Analysis // Comput. Math. Math. Phys. 2009. V. 49. № 11. P. 1966–1971.

ON POLYNOMIAL SOLVABILITY OF ONE QUADRATIC EUCLIDEAN CLUSTERING PROBLEM ON A LINE

A. V. Kel’manov[1,2], V. I. Khandeev[1,2]

[1] *Sobolev Institute of Mathematics*, *Novosibirsk*, *Russian Federation*
[2] *Novosibirsk State University*, *Novosibirsk*, *Russian Federation*

Presented by Academician of the RAS Yu.I. Zhuravlev February 28, 2019

Received March 6, 2019

We consider the problem of partitioning a finite set of points in Euclidean space into clusters so as to minimize the sum over all clusters of the intracluster sums of the squared distances between clusters elements and their centers. The centers of some clusters are given as an input, while the other centers are defined as centroids (geometrical centers). It is known that the general case of the problem is strongly NP-hard. We show that there exists an exact polynomial algorithm for the one-dimensional case of the problem.

Keywords: minimum sum-of-squares clustering, Euclidean space, NP-hard problem, one-dimensional case, polynomial solvability.