ЖУРНАЛ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И МАТЕМАТИЧЕСКОЙ ФИЗИКИ, 2019, том 59, № 5, с. 895–904

УДК 519.16:519.85

РАНДОМИЗИРОВАННЫЕ АЛГОРИТМЫ ДЛЯ НЕКОТОРЫХ ТРУДНОРЕШАЕМЫХ ЗАДАЧ КЛАСТЕРИЗАЦИИ КОНЕЧНОГО МНОЖЕСТВА ТОЧЕК ЕВКЛИДОВА ПРОСТРАНСТВА[1)]

© 2019 г. А. В. Кельманов[1,2,*], А. В. Панасенко[1,2,], В. И. Хандеев[1,2,***]**

([1] 630090 Новосибирск, пр-т акад. Коптюга, 4, Ин-т матем. им. С.Л. Соболева СО РАН, Россия;

[2] 630090 Новосибирск, ул. Пирогова, 2, Новосибирский гос. ун-т, Россия)

**e-mail: kelm@math.nsc.ru;*

***e-mail: a.v.panasenko@math.nsc.ru;*

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

Поступила в редакцию 23.03.2018 г.
Переработанный вариант 11.01.2019 г.
Принята к публикации 11.01.2019 г.

Рассматриваются две NP-трудные в сильном смысле задачи кластеризации конечного множества точек евклидова пространства. Во входном множестве первой задачи требуется найти кластер (подмножество) заданной мощности, минимизирующий сумму квадратов расстояний между элементами этого кластера и его центроидом (геометрическим центром). Каждая точка вне этого кластера рассматривается как одноэлементный кластер. Во второй задаче требуется найти разбиение входного множества на два кластера, минимизирующее сумму по обоим кластерам взвешенных внутрикластерных сумм квадратов расстояний между элементами кластеров и их центрами. Центр одного из кластеров неизвестен и определяется как его центроид, а центр второго кластера задан в некоторой точке пространства (без ограничения общности этой точкой является начало координат). Весовыми множителями внутрикластерных сумм являются заданные мощности искомых кластеров. Для обеих задач предложены рандомизированные параметризованные алгоритмы. Для заданных верхних границ относительной ошибки и вероятности несрабатывания определено значение параметра, при котором алгоритмы находят приближенные решения задач за полиномиальное время. Это время линейно зависит как от размерности пространства, так и от мощности входного множества. Найдены условия, при которых оба алгоритма находят асимптотически точные решения и имеют трудоемкость, линейно зависящую от размерности пространства и квадратично – от мощности входного множества. Библ. 27.

Ключевые слова: разбиение, последовательность, евклидово пространство, минимум суммы квадратов расстояний, NP-трудность, приближенный алгоритм.

DOI: 10.1134/S0044466919050090

ВВЕДЕНИЕ

Объектом исследования работы являются две NP-трудные в сильном смысле квадратичные евклидовы задачи кластеризации конечного множества точек. Предмет исследования – вопросы алгоритмической аппроксимируемости этих задач. Цель исследования – построение быстрых приближенных рандомизированных алгоритмов, обеспечивающих решение задач за линейное время, а также выявление условий, при которых алгоритмы гарантируют асимптотически точное решение.

Обе рассматриваемые задачи и эффективные алгоритмы их решения важны для развития методов дискретной оптимизации. Непосредственно из формулировок задач (см. следующий раздел) видно, что они имеют прямое отношение к оптимизационным проблемам вычислительной математики, компьютерной геометрии, аппроксимации, статистики.

[1)]Работа выполнена при финансовой поддержке РНФ (проект 16-11-10041, разд. 3), а также РФФИ (проекты 18-31-00398, 19-01-00308, разд. 1 и 2). Программы фундаментальных научных исследований РАН (проект 0314-2019-0015, разд. 1 и 2) и программы ТОП-5/100 Министерства образования и науки (разд. 1 и 2).

Несмотря на то что обе задачи интенсивно исследуются в последние годы и для них построены (см. следующий раздел) эффективные алгоритмы с теоретическими гарантиями качества (точности и временной сложности), быстрые алгоритмы с линейной трудоемкостью до настоящего времени отсутствовали. Между тем быстрые эффективные алгоритмы с теоретическими гарантиями качества – необходимый и востребованный (особенно в последние годы) математический инструмент для решения большеразмерных задач (Big-scaling problems), возникающих, в частности, при решении проблем редактирования и очистки данных (Data editing, Data cleaning), интерпретации данных (Data mining), а также проблем машинного обучения (Machine learning) (см. следующий раздел).

Статья имеет следующую структуру. В разд. 1 приведены формулировка задач, их трактовка, а также известные алгоритмические результаты. В этом же разделе анонсирован полученный результат. Далее, в следующем разделе сформулированы утверждения, необходимые для установления свойств предлагаемых алгоритмов. Наконец, в разд. 3 приведена пошаговая запись алгоритмов и доказаны оценки их качества (точности, временной сложности, вероятности несрабатывания). Там же установлены условия асимптотической точности алгоритмов.

1. ФОРМУЛИРОВКА ЗАДАЧ И ИХ ТРАКТОВКА, ИЗВЕСТНЫЕ И ПОЛУЧЕННЫЕ РЕЗУЛЬТАТЫ

Всюду далее используются следующие обозначения: $\mathbb{R}$ – множество вещественных чисел, $\|\cdot\|$ – евклидова норма, $\langle\cdot,\cdot\rangle$ – скалярное произведение.

Рассматриваемые задачи имеют следующие формулировки.

Задача 1. Дано: множество $\mathcal{Y} = \{y_1, \ldots, y_N\}$ точек в евклидовом пространстве размерности d и натуральное число $M \le N$. Найти: подмножество $\mathcal{C} \subseteq \mathcal{Y}$ мощности M такое, что

$$f(\mathcal{C}) = \sum_{y\in\mathcal{C}} \|y - \overline{y}(\mathcal{C})\|^2 \to \min,$$

где

$$\overline{y}(\mathcal{C}) = \frac{1}{|\mathcal{C}|}\sum\nolimits_{y\in\mathcal{C}} y$$

есть центроид множества $\mathcal{C}$.

Задача 2. Дано: N-элементное множество $\mathcal{Y}$ точек в евклидовом пространстве размерности d и натуральное число $M \le N$. Найти: разбиение множества $\mathcal{Y}$ на два непустых кластера $\mathcal{C}$ и $\mathcal{Y}\backslash\mathcal{C}$ такое, что

$$g(\mathcal{C}) = |\mathcal{C}|\sum_{y\in\mathcal{C}} \|y - \overline{y}(\mathcal{C})\|^2 + |\mathcal{Y}\backslash\mathcal{C}|\sum_{y\in\mathcal{Y}\backslash\mathcal{C}} \|y\|^2 \to \min \tag{1.1}$$

при ограничении $|\mathcal{C}| = M$.

Задачу 1 можно трактовать как поиск во множестве $\mathcal{Y}$ подмножества $\mathcal{C}$ в виде сферического сгущения из M точек с минимальным суммарным квадратичным разбросом относительно их неизвестного центроида. Поскольку центроид одноточечного множества совпадает с единственной точкой этого множества, задачу можно также трактовать как разбиение множества Y на $N - M + 1$ кластеров, мощность одного из которых равна M, а мощность остальных $N - M$ кластеров равна 1.

В задаче 2 требуется найти 2-разбиение множества $\mathcal{Y}$, минимизирующее сумму мощностно взвешенных внутрикластерных суммарных квадратичных разбросов относительно заданного в начале координат центра одного кластера – $\mathcal{Y}\backslash\mathcal{C}$ – и относительно неизвестного центроида – $\overline{y}(\mathcal{C})$ – у другого кластера $\mathcal{C}$.

В прикладном плане обе рассматриваемые задачи можно трактовать, в частности, в виде проблем редактирования и очистки данных (Data editing, Data cleaning), интерпретации данных (Data mining), а также проблем машинного обучения (Machine learning) (см., например, [1]–[8] и цитированные там работы). Некоторые содержательные трактовки задач 1 и 2 можно найти в [9]–[18].

Задача 1 имеет следующую интерпретацию в терминах очистки или редактирования данных. Имеется таблица $\mathcal{Y}$, содержащая результаты $\{y_1, \ldots, y_N\}$ измерений d-мерного набора y числовых

информационно значимых характеристик семейства некоторых объектов. Некоторые объекты в этом семействе идентичны и имеют одинаковые характеристики; число M таких объектов известно. Остальные объекты имеют различные характеристики, не совпадающие с характеристиками идентичных объектов. В каждом результате измерения, представленном в таблице, имеется ошибка. При этом соответствие между объектами и элементами таблицы неизвестно. Требуется, используя критерий минимума суммы квадратов расстояний, найти подмножество $\mathcal{C}$ наборов, соответствующих идентичным объектам, и оценить по результатам измерений набор $\overline{y}(\mathcal{C})$ характеристик этих объектов (учитывая измерительные ошибки). Легко видеть, что оставшиеся одноэлементные кластеры можно трактовать как так называемые "выбросы", которые могут присутствовать в таблице с данными из-за возможных сбоев измерительного прибора.

В терминах анализа данных задачу 2 можно трактовать аналогичным образом. Имеется таблица $\mathcal{Y}$, содержащая результаты измерений совокупности объектов из двух групп $\mathcal{C}$ и $\mathcal{Y}\backslash\mathcal{C}$, включающих однородные или одинаковые (по некоторому набору характеристик) изделия. Первая группа $\mathcal{C}$ содержит M изделий, а вторая – $(N - M)$. Изделия из первой группы имеют неизвестные характеристики, а изделия из второй – заданные (в частности, можно считать, что значения всех характеристик равны нулю). Требуется, используя критерий (1), по результатам измерений разбить совокупность изделий на две группы (части) и оценить набор $\overline{y}(\mathcal{C})$ характеристик изделий из первой группы, учитывая, что данные получены с измерительной ошибкой. Поскольку $|\mathcal{Y}| = N$, $|\mathcal{C}| = M$ и $|\mathcal{Y}\backslash\mathcal{C}| = N - M$, а N и M заданы на входе задачи, нетрудно видеть, что задача 2 заключается в 2-разбиении данных по правилу Байеса при заданных априорных вероятностях

$$p = \frac{M}{N} \quad \text{и} \quad 1 - p = \frac{N - M}{N}$$

двух групп изделий.

Приведем доступные нам результаты (из опубликованных источников), полученные для каждой из задач.

Задача 1 известна также под названием M-Variance [19]. Сильная NP-трудность евклидова случая задачи доказана в [9]. В этой же работе установлено, что для общего случая задачи не существует полностью полиномиальной аппроксимационной схемы (FPTAS), если $P \neq NP$. Точные алгоритмы с трудоемкостью $\mathbb{O}(dN^{d+1})$ предложены в [19], [20]. Если размерность d пространства фиксирована (ограничена константой), то эти алгоритмы полиномиальны и их трудоемкость оценивается величиной $\mathbb{O}(N^{d+1})$.

В [10] предложен точный алгоритм для случая целочисленных входных данных. Трудоемкость алгоритма равна $\mathbb{O}(dN(2MB + 1)^d)$, где B – максимальное абсолютное значение координат точек входного множества. Если размерность пространства ограничена константой, то алгоритм псевдополиномиален и имеет трудоемкость $\mathbb{O}(N(MB)^d)$.

Для общего случая задачи 1 в [11] представлен 2-приближенный полиномиальный алгоритм с трудоемкостью $\mathbb{O}(dN^2)$. Полиномиальная приближенная схема (PTAS) построена в [21]. Время работы этой схемы $\mathbb{O}(dN^{2/\varepsilon+1}(9/\varepsilon)^{3/\varepsilon})$, где $\varepsilon > 0$ – относительная ошибка.

В [12] предложен алгоритм, который позволяет находить $(1 + \varepsilon)$-приближенное решение задачи за время $\mathbb{O}(dN^2(2\sqrt{d}M/\varepsilon + 2)^d)$ для заданного $\varepsilon \in (0,1)$. В случае фиксированной размерности d пространства алгоритм имеет трудоемкость $\mathbb{O}(N^2(M/\varepsilon)^d)$ и реализует схему FPTAS.

Улучшенный алгоритм, позволяющий находить приближенное решение задачи с относительной погрешностью ε за время

$$\mathbb{O}\left(dN^2\left(\sqrt{\frac{2d}{\varepsilon}} + 2\right)^d\right),$$

предложен в [13]. Этот алгоритм реализует схему FPTAS в случае фиксированной размерности d пространства, поскольку в этом случае имеет трудоемкость $\mathbb{O}(N^2(1/\varepsilon)^{d/2})$. В этой же работе была предложена другая улучшенная приближенная схема, имеющая трудоемкость

$$\mathbb{O}\left(\sqrt{d}N^2\left(\frac{\pi e}{2}\right)^{d/2}\left(\sqrt{\frac{2}{\varepsilon}}+2\right)^d\right).$$

Эта схема остается полиномиальной в случае, когда размерность d пространства есть величина $\mathbb{O}(\log N)$, т.е. в случае, когда размерность пространства – медленно растущая функция от мощности входного множества. В этом случае алгоритм реализует схему PTAS с трудоемкостью

$$\mathbb{O}\left(N^{C\left(1.05+\log\left(2+\sqrt{\frac{2}{\varepsilon}}\right)\right)}\right),$$

где C – положительная константа.

Следующие результаты были получены для задачи 2. Напомним, что задача 2 в постановочном плане близка к известной [22]–[25] задаче *Mini-Sum 2-clustering* (другое название – *Min-Sum All-Pairs 2-clustering*), в которой требуется найти разбиение, минимизирующее сумму

$$|\mathcal{C}|\sum_{y\in\mathcal{C}}\|y-\overline{y}(\mathcal{C})\|^2+|\mathcal{Y}\setminus\mathcal{C}|\sum_{y\in\mathcal{Y}\setminus\mathcal{C}}\|y-\overline{y}(\mathcal{Y}\setminus\mathcal{C})\|^2.$$

В задаче *Mini-Sum 2-clustering* центроиды обоих кластеров неизвестны, а в задаче 2 неизвестен только один. Эти задачи не эквивалентны. Сильная NP-трудность евклидовых случаев обеих задач была установлена в [14], [15]. Там же доказано, что для общего случая этих задач не существует схемы FPTAS, если $P \neq NP$.

Точный алгоритм для случая целочисленных координат входных точек задачи 2 предложен в [16]; время работы алгоритма – $\mathbb{O}(dN(2MB+1)^d)$, где B – максимальное абсолютное значение координат входных точек. Если размерность d пространства ограничена константой, то алгоритм псевдополиномиален и имеет трудоемкость $\mathbb{O}(N(MB)^d)$.

Приближенный эффективный алгоритм для общего случая задачи 2 представлен в [17]. Этот алгоритм находит 2-приближенное решение задачи за время $\mathbb{O}(dN^2)$.

В [18] предложен алгоритм, который находит приближенное решение задачи с относительной погрешностью ε за время

$$\mathbb{O}\left(dN^2(\sqrt{\frac{2d}{\varepsilon}}+2)^d\right).$$

В случае фиксированной размерности d пространства этот алгоритм реализует схему FPTAS, так как в этом случае его трудоемкость равна $\mathbb{O}(N^2(1/\varepsilon)^{d/2})$.

Кроме того, в [13] предложена модификация этого алгоритма с улучшенной трудоемкостью

$$\mathbb{O}\left(\sqrt{d}N^2\left(\frac{\pi e}{2}\right)^{d/2}\left(\sqrt{\frac{2}{\varepsilon}}+2\right)^d\right).$$

Модифицированный алгоритм реализует FPTAS с трудоемкостью $\mathbb{O}(N^2(1/\varepsilon)^{d/2})$ в случае фиксированной размерности d пространства и остается полиномиальным в случае, когда размерность пространства есть величина $\mathbb{O}(\log N)$. В последнем случае он реализует PTAS с трудоемкостью

$$\mathbb{O}\left(N^{C\left(1.05+\log\left(2+\sqrt{\frac{2}{\varepsilon}}\right)\right)}\right),$$

где C – положительная константа.

В настоящей работе для общего случая задач 1 и 2 предложены рандомизированные параметризованные алгоритмы. При условии $M \geq \beta N$, где $\beta \in (0,1)$ – некоторая константа, для заданных $\varepsilon > 0$ и $\gamma \in (0,1)$ алгоритмы находят $(1+\varepsilon)$-приближенные решения задач с вероятностью не менее $1-\gamma$ за время $\mathbb{O}(dN)$. Найдены условия асимптотической точности предложенных алгоритмов, то есть условия, при которых алгоритмы находят $(1+\varepsilon_N)$-приближенные решения задач за время $\mathbb{O}(dN^2)$ с вероятностью не менее $1-\gamma_N$, где $\varepsilon_N \to 0$ и $\gamma_N \to 0$ при $N \to \infty$.

2. ОСНОВЫ АЛГОРИТМОВ

Для обоснования алгоритмов нам потребуется несколько вспомогательных утверждений. Вероятностной основой алгоритмов являются следующие две леммы [26]. Первая основана на неравенстве Маркова, вторая – на неравенстве Чернова для вероятности больших уклонений.

Лемма 1. *Пусть $\mathcal{Z}$ – произвольное N-элементное множество точек из $\mathbb{R}^d$, $\mathcal{C} \subseteq \mathcal{Z}$, $|\mathcal{C}| = M$, а $\mathcal{T}$ – мультимножество, полученное k случайными независимыми выборками (с возвращением) по одному элементу из множества $\mathcal{Z}$. Пусть*

$$\overline{z}(\mathcal{C}) = \frac{1}{|\mathcal{C}|}\sum_{z\in\mathcal{C}} z \quad \text{и} \quad \overline{z}(\mathcal{T}\cap\mathcal{C}) = \frac{1}{|\mathcal{T}\cap\mathcal{C}|}\sum_{z\in\mathcal{T}\cap\mathcal{C}} z$$

суть центроиды множества $\mathcal{C}$ и мультимножества $\mathcal{T}\cap\mathcal{C}$ соответственно. Тогда для любого натурального $t \le k$ и произвольного $\delta \in (0,1)$ справедлива оценка

$$\Pr\left(\sum_{z\in\mathcal{C}}\|z-\overline{z}(\mathcal{T}\cap\mathcal{C})\|^2 \ge \left(1+\frac{1}{\delta t}\right)\sum_{z\in\mathcal{C}}\|z-\overline{z}(\mathcal{C})\|^2 \,\Big|\, |\mathcal{T}\cap\mathcal{C}| \ge t\right) \le \delta.$$

Лемма 2. *Пусть выполнены условия леммы* 1. *Тогда для произвольного $\nu \in (0,1)$ справедлива оценка*

$$\Pr\left(|\mathcal{T}\cap\mathcal{C}| \le (1-\nu)\frac{M}{N}k\right) \le e^{-\frac{\nu^2 Mk}{2N}}.$$

Доказательство следующей леммы приведено в [16].

Лемма 3. *Пусть*

$$S(\mathcal{C},x) = |\mathcal{C}|\sum_{y\in\mathcal{C}}\|y-x\|^2 + |\mathcal{Y}\setminus\mathcal{C}|\sum_{y\in\mathcal{Y}\setminus\mathcal{C}}\|y\|^2, \quad \mathcal{C}\subseteq\mathcal{Y}, \quad x\in\mathbb{R}^d.$$

Тогда справедливы следующие утверждения:

1) *для любого непустого множества $\mathcal{C}\subseteq\mathcal{Y}$ минимум функции $S(\mathcal{C},x)$ по $x\in\mathbb{R}^d$ достигается в точке $\overline{y}(\mathcal{C}) = \frac{1}{|\mathcal{C}|}\sum_{y\in\mathcal{C}} y$;*

2) *если $|\mathcal{C}| = M = \text{const}$, то для любой фиксированной точки $x\in\mathbb{R}^d$ минимум функции $S(\mathcal{C},x)$ по $\mathcal{C}\subseteq\mathcal{Y}$ достигается на множестве $\mathcal{B}^x$, состоящем из M точек множества $\mathcal{Y}$, имеющих наименьшие значения функции*

$$h^x(y) = (2M-N)\|y\|^2 - 2M\langle y,x\rangle, \quad y\in\mathcal{Y}. \tag{2.2}$$

3. РАНДОМИЗИРОВАННЫЕ АЛГОРИТМЫ

Общий подход к построению алгоритмов для рассматриваемых труднорешаемых задач – классический. А именно, для каждой из задач строится семейство допустимых решений (кластеров) и в найденном семействе выбирается наилучшее в смысле минимума целевой функции. При этом построение допустимых решений опирается на отличающиеся свойства оптимальных решений задач. В задаче 1 это известное (см., например, [11]) свойство состоит в том, что M точек оптимального решения являются точками множества $\mathcal{Y}$, ближайшими по расстоянию к неизвестному центроиду. В задаче 2 свойство оптимального решения устанавливается (см. [16]) утверждениями приведенной выше леммы 3.

Сформулируем алгоритм решения задачи 1.

Алгоритм $\mathcal{A}_1$

Вход: множество $\mathcal{Y}$, натуральное число M, натуральный параметр k.

Шаг 1. Сформируем мультимножество $\mathcal{T}$ точек с помощью k независимых случайных выборок (с возвращением) по одному элементу из множества $\mathcal{Y}$.

Шаг 2. Для каждого непустого мультиподмножества $\mathcal{H}$ мультимножества $\mathcal{T}$ вычислим центроид $\overline{y}(\mathcal{H})$ и сформируем подмножество $\mathcal{C}$ из M элементов множества $\mathcal{Y}$, ближайших (по расстоянию) к $\overline{y}(\mathcal{H})$. Вычислим и запомним значение функции $f(\mathcal{C})$.

Шаг 3. В семействе решений, найденных на шаге 2, выберем подмножество $\mathcal{C} = \mathcal{C}_{\mathcal{A}_1}$, для которого значение $f(\mathcal{C})$ минимально. Если оптимальных значений несколько, выберем любое из них.

Выход: множество $\mathcal{C}_{\mathcal{A}_1}$.

Алгоритм решения задачи 2 имеет сходную пошаговую запись. Основное отличие от алгоритма $\mathcal{A}_1$ состоит в построении допустимых решений задачи на шаге 2.

Алгоритм $\mathcal{A}_2$

Вход: множество $\mathcal{Y}$, натуральное число M, натуральный параметр k.

Шаг 1. Сформируем мультимножество $\mathcal{T}$ точек с помощью k независимых случайных выборок (с возвращением) по одному элементу из множества $\mathcal{Y}$.

Шаг 2. Для каждого непустого мультиподмножества $\mathcal{H}$ мультимножества $\mathcal{T}$ вычислим центроид $\overline{y}(\mathcal{H})$ и сформируем подмножество $\mathcal{C}$ из M элементов множества $\mathcal{Y}$ с наименьшими значениями функции $h^{\overline{y}(\mathcal{H})}(z)$, $z \in \mathcal{Y}$ (используя формулу (2.2) при $x = \overline{y}(\mathcal{H})$). Вычислим и запомним значение функции $g(\mathcal{C})$.

Шаг 3. В семействе решений, найденных на шаге 2, выберем подмножество $\mathcal{C} = \mathcal{C}_{\mathcal{A}_2}$, для которого значение $g(\mathcal{C})$ минимально. Если оптимальных значений несколько, выберем любое из них.

Выход: множество $\mathcal{C}_{\mathcal{A}_2}$.

Свойства алгоритмов устанавливает

Теорема 1. *Для произвольного вещественного* $\delta \in (0,1)$ *и натуральных* $t \le k$ *алгоритмы* $\mathcal{A}_1$ *и* $\mathcal{A}_2$ *находят* $\left(1+\frac{1}{\delta t}\right)$*-приближенные решения задач* 1 *и* 2 *за время* $\mathbb{O}(2^k d(k+N))$ *вероятностью не менее* $1-(\delta+\alpha)$*, где*

$$\alpha = \sum_{i=0}^{t-1} \binom{k}{i} \left(\frac{M}{N}\right)^i \left(1-\frac{M}{N}\right)^{k-i}.$$

Доказательство. Докажем оценку точности алгоритма $\mathcal{A}_1$. Пусть $\mathcal{C}_1^*$ – оптимальное решение задачи 1, а $\mathcal{C}_{\mathcal{A}_1}$ – множество, найденное алгоритмом $\mathcal{A}_1$.

Предположим, что мультимножество $\mathcal{T}$ сформировано так, что $|\mathcal{T} \cap \mathcal{C}_1^*| \ge 1$. В этом случае мультимножество $\mathcal{H} = \mathcal{T} \cap \mathcal{C}_1^*$ было рассмотрено на шаге 2 алгоритма. Пусть $\mathcal{C}$ – подмножество из M точек множества $\mathcal{Y}$, ближайших к $\overline{y}(\mathcal{T} \cap \mathcal{C}_1^*)$, построенное на этом шаге.

Из определения шага 3 следует неравенство

$$f(\mathcal{C}_{\mathcal{A}_1}) \le f(\mathcal{C}). \tag{3.3}$$

Кроме того, поскольку для произвольного конечного множества $\mathcal{Z} \subset \mathbb{R}^d$ минимум суммы $\sum_{y\in\mathcal{Z}} \|y-x\|^2$ по $x \in \mathbb{R}^d$ достигается в точке $x = \frac{1}{|\mathcal{Z}|}\sum_{z\in\mathcal{Z}} z$, для правой части (3.3) имеем

$$f(\mathcal{C}) = \sum_{y\in\mathcal{C}} \|y-\overline{y}(\mathcal{C})\|^2 \le \sum_{y\in\mathcal{C}} \|y-\overline{y}(\mathcal{H})\|^2. \tag{3.4}$$

Далее, так как по определению шага 2 множество $\mathcal{C}$ состоит из M точек, ближайших к $\overline{y}(\mathcal{H})$, справедлива оценка

$$\sum_{y\in\mathcal{C}} \|y-\overline{y}(\mathcal{H})\|^2 \le \sum_{y\in\mathcal{C}_1^*} \|y-\overline{y}(\mathcal{H})\|. \tag{3.5}$$

Объединяя (3.3)–(3.5), получаем, что при $|\mathcal{T} \cap \mathcal{C}_1^*| \geq 1$ справедлива цепочка неравенств

$$f(\mathcal{C}_{\mathcal{A}_1}) \leq f(\mathcal{C}) \leq \sum_{y \in \mathcal{C}} \|y - \bar{y}(\mathcal{H})\|^2 \leq \sum_{y \in \mathcal{C}_1^*} \|y - \bar{y}(\mathcal{H})\|^2. \tag{3.6}$$

Применяя лемму 1 к $\mathcal{Z} = \mathcal{Y}$ и $\mathcal{C} = \mathcal{C}_1^*$, получаем, что при $|\mathcal{T} \cap \mathcal{C}_1^*| \geq t$ с вероятностью не менее $1 - \delta$ выполнено неравенство

$$\sum_{y \in \mathcal{C}_1^*} \|y - \bar{y}(\mathcal{H})\|^2 < \left(1 + \frac{1}{\delta t}\right) \sum_{y \in \mathcal{C}_1^*} \left\|y - \bar{y}(\mathcal{C}_1^*)\right\|^2. \tag{3.7}$$

Тогда из (3.6), (3.7) следует, что при $|\mathcal{T} \cap \mathcal{C}_1^*| \geq t$ с вероятностью не менее $1 - \delta$ справедлива оценка

$$f(\mathcal{C}_{A_1}) < \left(1 + \frac{1}{\delta t}\right) \sum_{y \in \mathcal{C}_1^*} \left\|y - \bar{y}(\mathcal{C}_1^*)\right\|^2 = \left(1 + \frac{1}{\delta t}\right) f(\mathcal{C}_1^*).$$

В терминах условной вероятности эта оценка означает, что

$$\Pr\left(f(\mathcal{C}_{A_1}) < \left(1 + \frac{1}{\delta t}\right) f(\mathcal{C}_1^*) \,\middle|\, |\mathcal{T} \cap \mathcal{C}_1^*| \geq t\right) \geq 1 - \delta.$$

Поэтому, положив $\alpha = \Pr(|\mathcal{T} \cap \mathcal{C}_1^*| < t)$, для безусловной вероятности противоположного события, т.е. для вероятности несрабатывания алгоритма, найдем равенство

$$\Pr\left(f(\mathcal{C}_{\mathcal{A}_1}) \geq \left(1 + \frac{1}{\delta t}\right) f(\mathcal{C}_1^*)\right) = \Pr\left(f(\mathcal{C}_{\mathcal{A}_1}) \geq \left(1 + \frac{1}{\delta t}\right) f(\mathcal{C}_1^*) \text{ and } |\mathcal{T} \cap \mathcal{C}_1^*| \geq t\right) +$$

$$+ \Pr\left(f(\mathcal{C}_{\mathcal{A}_1}) \geq \left(1 + \frac{1}{\delta t}\right) f(\mathcal{C}_1^*) \text{ and } |\mathcal{T} \cap \mathcal{C}_1^*| < t\right) \leq$$

$$\leq \Pr\left(f(\mathcal{C}_{\mathcal{A}_1}) \geq \left(1 + \frac{1}{\delta t}\right) f(\mathcal{C}_1^*) \,\middle|\, |\mathcal{T} \cap \mathcal{C}_1^*| \geq t\right) + \Pr\left(|\mathcal{T} \cap \mathcal{C}_1^*| < t\right) \leq \delta + \alpha.$$

Отсюда следует, что вероятность срабатывания алгоритма $\mathcal{A}_1$ оценивается снизу величиной $1 - (\delta + \alpha)$.

Равенство

$$\alpha = \sum_{i=0}^{t-1} \binom{k}{i} \left(\frac{M}{N}\right)^i \left(1 - \frac{M}{N}\right)^{k-i}$$

следует из того, что формирование мультимножества $\mathcal{T}$ состоит из k независимых равновероятных испытаний, в которых “успех” – это событие “элемент, выбранный из $\mathcal{Y}$, лежит в $\mathcal{C}_1^*$”.

Доказательство оценки точности алгоритма $\mathcal{A}_2$ частично повторяет доказательство оценки точности алгоритма $\mathcal{A}_1$.

Пусть $\mathcal{C}_2^*$ – оптимальное решение задачи 2, а $\mathcal{C}_{\mathcal{A}_2}$ – решение, найденное алгоритмом $\mathcal{A}_2$.

Предположим, что в алгоритме $\mathcal{A}_2$ мультимножество $\mathcal{T}$ сформировано так, что $|\mathcal{T} \cap \mathcal{C}_2^*| \geq 1$. При этом условии пусть $\mathcal{C}$ – подмножество из M точек множества $\mathcal{Y}$ с наименьшими значениями функции $h^{\bar{y}(\mathcal{H})}(z)$, $z \in \mathcal{Y}$, построенное на шаге 2 для мультиподмножества $\mathcal{H} = \mathcal{T} \cap \mathcal{C}_2^*$.

Из определения шага 3 следует, что

$$g(\mathcal{C}_{\mathcal{A}_2}) \leq g(\mathcal{C}). \tag{3.8}$$

Справедливость следующего неравенства устанавливается с опорой на тот же факт, что и неравенство (3.4), установленное при доказательстве свойств алгоритма $\mathcal{A}_2$:

$$g(\mathcal{C}) = |\mathcal{C}| \sum_{y \in \mathcal{C}} \|y - \bar{y}(\mathcal{C})\|^2 + |Y \setminus \mathcal{C}| \sum_{y \in \mathcal{Y} \setminus \mathcal{C}} \|y\|^2 \leq |\mathcal{C}| \sum_{y \in \mathcal{C}} \|y - \bar{y}(\mathcal{H})\|^2 + |\mathcal{Y} \setminus \mathcal{C}| \sum_{y \in \mathcal{Y} \setminus \mathcal{C}} \|y\|^2. \tag{3.9}$$

Далее, в соответствии с определением шага 2 множество $\mathcal{C}$ состоит из M точек множества $\mathcal{Y}$ с наименьшими значениями $h^{\bar{y}(\mathcal{H})}(z)$, $z \in \mathcal{Y}$. Поэтому, применяя лемму 3, для правой части (3.9) получим оценку

$$|\mathcal{C}|\sum_{y\in\mathcal{C}}\|y-\bar{y}(\mathcal{H})\|^2 + |\mathcal{Y}\backslash\mathcal{C}|\sum_{y\in\mathcal{Y}\backslash\mathcal{C}}\|y\|^2 \le |\mathcal{C}_2^*|\sum_{y\in\mathcal{C}_2^*}\|y-\bar{y}(\mathcal{H})\|^2 + |\mathcal{Y}\backslash\mathcal{C}_2^*|\sum_{y\in\mathcal{Y}\backslash\mathcal{C}_2^*}\|y\|^2. \tag{3.10}$$

Наконец, как и при оценке свойств алгоритма $\mathcal{A}_1$, объединяя (3.8)–(3.10) и применяя лемму 1, получаем, что при $|\mathcal{T}\cap\mathcal{C}_2^*| \ge t$ с вероятностью не менее $1-\delta$ выполнено неравенство

$$g(\mathcal{C}_{\mathcal{A}_2}) < \left(1+\frac{1}{\delta t}\right)|\mathcal{C}_2^*|\sum_{y\in\mathcal{C}_2^*}\left\|y-\bar{y}(\mathcal{C}_2^*)\right\|^2 + |Y\backslash\mathcal{C}_2^*|\sum_{y\in\mathcal{Y}\backslash\mathcal{C}_2^*}\|y\|^2,$$

правая часть которого оценивается величиной $\left(1+\frac{1}{\delta t}\right)g(\mathcal{C}_2^*)$. Окончание доказательства аналогично окончанию доказательства свойств алгоритма $\mathcal{A}_1$.

Оценим временную сложность алгоритмов. Шаг 1 требует $\mathbb{O}(k)$ операций. Шаг 2 выполняется 2^k раз. Центроид каждого мультиподмножества $\mathcal{H}$ в обоих алгоритмах вычисляется за $\mathbb{O}(dk)$ операций. В алгоритме $\mathcal{A}_1$ вычисление расстояний от этого центроида до точек множества $\mathcal{Y}$ требует $\mathbb{O}(dN)$ операций, а M точек, ближайших к центроиду, выбираются за $\mathbb{O}(N)$ операций без сортировки (см., например, [27]). Аналогичным образом, в алгоритме $\mathcal{A}_2$ вычисление значений $h^{\bar{y}(\mathcal{H})}(z)$, $z \in \mathcal{Y}$, требует $\mathbb{O}(dN)$ операций, и нахождение M наименьших элементов из N осуществляется за $\mathbb{O}(N)$ операций без сортировки. Шаг 3 (выбор наименьшего элемента) требует не более $\mathbb{O}(2^k)$ операций.

Таким образом, временная сложность обоих алгоритмов есть величина $\mathbb{O}(2^k d(k+N))$.

Следующее следствие устанавливает условия, при которых оба алгоритма имеют линейную по d и по N временную сложность при заданных верхних границах относительной погрешности ε и вероятности γ несрабатывания.

Следствие 1. Пусть $M \ge \beta N$, где $\beta \in (0,1)$ – константа, $\varepsilon > 0$ и $\gamma \in (0,1)$. Тогда при фиксированном параметре

$$k = \max\left(\left\lceil\frac{2}{\beta}\left\lceil\frac{2}{\gamma\varepsilon}\right\rceil\right\rceil, \left\lceil\frac{8}{\beta}\ln\frac{2}{\gamma}\right\rceil\right)$$

алгоритмы $\mathcal{A}_1$ и $\mathcal{A}_2$ находят $(1+\varepsilon)$-приближенные решения задач 1 и 2 за время $\mathbb{O}(dN)$ с вероятностью не менее $1-\gamma$.

Доказательство. Докажем следствие для алгоритма $\mathcal{A}_1$ (доказательство для алгоритма $\mathcal{A}_2$ аналогично).

Пусть $\delta = \frac{\gamma}{2}$, $t = \left\lceil\frac{1}{\delta\varepsilon}\right\rceil = \left\lceil\frac{2}{\gamma\varepsilon}\right\rceil$. Заметим, что в этом случае $k \ge \frac{2t}{\beta}$ и $k \ge \frac{8}{\beta}\ln\frac{2}{\gamma}$. Применяя лемму 2 к $\nu = \frac{1}{2}$ и $\mathcal{C} = \mathcal{C}_1^*$, получаем, что

$$\Pr\left(|\mathcal{T}\cap\mathcal{C}_1^*| \le \frac{kM}{2N}\right) \le e^{-\frac{kM}{8N}}.$$

Тогда, в условиях следствия 1, выполнена следующая цепочка оценок:

$$\alpha = \Pr\left(|\mathcal{T}\cap\mathcal{C}_1^*| < t\right) \le \Pr\left(|\mathcal{T}\cap\mathcal{C}_1^*| < \frac{\beta k}{2}\right) \le \Pr\left(|\mathcal{T}\cap\mathcal{C}_1^*| \le \frac{kM}{2N}\right) \le$$
$$\le e^{-\frac{kM}{8N}} \le e^{-\frac{M}{\beta N}\ln\frac{2}{\gamma}} \le e^{-\ln\frac{2}{\gamma}} = \frac{\gamma}{2}.$$

Следовательно, по теореме 1 для заданного выше значения k алгоритм $\mathcal{A}_1$ находит решение задачи 1 с относительной погрешностью $\frac{1}{\delta t} = \left(\frac{\gamma}{2}\left\lceil\frac{2}{\gamma\varepsilon}\right\rceil\right)^{-1} \le \varepsilon$ за время $\mathbb{O}(2^k d(k+N))$ с вероятностью несрабатывания не более $\delta + \alpha \le \frac{\gamma}{2} + \frac{\gamma}{2} = \gamma$. Поскольку параметр k фиксирован, время работы алгоритма равно $\mathbb{O}(dN)$.

Условия асимптотической точности алгоритмов устанавливает

Теорема 2. *Пусть* $k = \lceil \log_2 N \rceil$ *и* $M \ge \beta N$, *где* $\beta \in (0,1)$ *– константа. Тогда алгоритмы* $\mathcal{A}_1$ *и* $\mathcal{A}_2$ *находят* $(1+\varepsilon_N)$*-приближенные решения задач* 1 *и* 2 *с вероятностью не менее* $1-\gamma_N$ *за время* $\mathbb{O}(dN^2)$, *где* $\varepsilon_N \xrightarrow[N\to\infty]{} 0$, $\gamma_N \xrightarrow[N\to\infty]{} 0$.

Доказательство. Оценка времени работы алгоритмов при условии $k = \lceil \log_2 N \rceil$ очевидна.

Оценим относительную погрешность и вероятность несрабатывания алгоритма $\mathcal{A}_1$. Пусть $\delta = (\log_2 N)^{-1/2}$, $t = \left\lceil \frac{kM}{2N} \right\rceil$ в условиях теоремы 1. Тогда относительная ошибка $\varepsilon_N = \frac{1}{\delta t} = (\log_2 N)^{1/2} / \left\lceil \frac{kM}{2N} \right\rceil$ ограничена сверху значением $\frac{2}{\beta}(\log_2 N)^{-1/2} \xrightarrow[N\to\infty]{} 0$.

Далее, применяя лемму 2 для $\nu = \frac{1}{2}$ и $\mathscr{C} = \mathscr{C}_1^*$, получаем

$$\Pr\left(|\mathscr{T} \cap \mathscr{C}_1^*| \le \frac{kM}{2N}\right) \le e^{-\frac{kM}{8N}}.$$

Следовательно,

$$\alpha = \Pr\left(|\mathscr{T} \cap \mathscr{C}_1^*| < t\right) \le \Pr\left(|\mathscr{T} \cap \mathscr{C}_1^*| \le \frac{kM}{2N}\right) \le e^{-\frac{kM}{8N}} \le e^{-\frac{\beta \log_2 N}{8}} = N^{-\frac{\beta}{8\ln 2}} \xrightarrow[N\to\infty]{} 0.$$

Таким образом, для вероятности γ_N несрабатывания алгоритма $\mathcal{A}_1$ получаем $\gamma_N = \delta + \alpha \xrightarrow[N\to\infty]{} 0$. Асимптотические свойства алгоритма $\mathcal{A}_2$ устанавливаются аналогично.

ЗАКЛЮЧЕНИЕ

В работе предложены сходные по построению рандомизированные параметризованные алгоритмы для двух неэквивалентных NP-трудных в сильном смысле квадратичных евклидовых задач кластеризации конечного множества точек. Для заданных верхних границ относительной ошибки и вероятности несрабатывания найдены значения параметра алгоритмов, при котором эти алгоритмы обеспечивают отыскание приближенных решений задач за время, линейно зависящее от размерности пространства и от мощности входного множества. Найдены условия, при которых оба алгоритма полиномиальны и асимптотически точны.

На наш взгляд, представленная в работе рандомизированная техника будет полезным математическим инструментом для построения быстрых алгоритмов решения близких по постановке задач. В свою очередь сами предложенные алгоритмы как инструменты могут применяться для получения быстрых решений прикладных проблем очистки и интерпретации данных, распознавания образов, машинного обучения и, конечно, проблем обработки (аппроксимации) большеразмерных данных.

СПИСОК ЛИТЕРАТУРЫ

1. *de Waal T., Pannekoek J., Scholtus S.* Handbook of Statistical Data Editing and Imputation. John Wiley and Sons, Inc. Hoboken, New Jersey, 2011. 456 p.
2. *Osborne J.W.* Best Practices in Data Cleaning: A Complete Guide to Everything You Need to Do Before and After Collecting Your Data. 1st Edition. SAGE Publication, Inc. Los Angeles, 2013.
3. *Greco L.* Robust Methods for Data Reduction Alessio Farcomeni. Chapman and Hall/CRC, 2015. 297 p.
4. *Bishop C.M.* Pattern Recognition and Machine Learning. New York: Springer Science + Business Media, LLC, 2006. 738 p.

5. *James G., Witten D., Hastie T., Tibshirani R.* An Introduction to Statistical Learning. New York: Springer Science + Business Media, LLC.2013. 426 p.
6. *Hastie T., Tibshirani R., Friedman J.* The Elements of Statistical Learning (2nd edition). Springer-Verlag, 2009. 763 p.
7. *Aggarwal C.C.* Data Mining: The Textbook. Springer International Publishing, 2015. 734 p.
8. *Goodfellow I., Bengio Y., Courville A.* Deep Learning (Adaptive Computation and Machine Learning series). The MIT Press, 2017. 800 p.
9. *Кельманов А.В., Пяткин А.В.* NP-полнота некоторых задач выбора подмножества векторов // Дискретн. анализ и исслед. опер. 2010. Т. 17. № 5. С. 37–45.
10. *Кельманов А.В., Романченко С.М.* Псевдополиномиальные алгоритмы для некоторых труднорешаемых задач поиска подмножества векторов и кластерного анализа // Автомат. и телемех. 2012. № 2. С. 156–162.
11. *Кельманов А.В., Романченко С.М.* Приближенный алгоритм решения одной задачи поиска подмножества векторов // Дискретн. анализ и исслед. опер. 2011. Т. 18. № 1. С. 61–69.
12. *Кельманов А.В., Романченко С.М.* FPTAS для одной задачи поиска подмножества векторов // Дискретн. анализ и исслед. опер. 2014. Т. 21. № 3. С. 41–52.
13. *Kel'manov A.V., Motkova A.V., Shenmaier V.V.* An approximation scheme for a weighted two-cluster partition problem // LNCS. 2018. V. 10716. P. 323–333.
14. *Кельманов А.В., Пяткин А.В.* NP-трудность некоторых квадратичных евклидовых задач 2-кластеризации // Докл. АН. 2015. Т. 464. № 5. С. 535–538.
15. *Кельманов А.В., Пяткин А.В.* О сложности некоторых квадратичных евклидовых задач 2-кластеризации // Ж. вычисл. матем. и матем. физ. 2016. Т. 56. № 3. С. 498–504.
16. *Кельманов А.В., Моткова А.В.* Точные псевдополиномиальные алгоритмы для задачи сбалансированной 2-кластеризации // Дискретн. анализ и исслед. опер. 2016. Т. 23. № 3. С. 21–34.
17. *Кельманов А.В., Моткова А.В.* Приближенный полиномиальный алгоритм для задачи взвешенной 2-кластеризации с ограничением на мощности кластеров // Ж. вычисл. матем. и матем. физ. 2018. Т. 58. № 1. С. 136–142.
18. *Kel'manov A.V., Motkova A.V.* A fully polynomial-time approximation scheme for a special case of a balanced 2-clustering problem // LNCS. 2016. V. 9869. P. 182–192.
19. *Aggarwal H., Imai N., Katoh N., Suri S.* Finding k points with minimum diameter and related problems // J. Algorithms. 1991. V. 12. № 1. P. 38–56.
20. *Шенмайер В.В.* Решение некоторых задач поиска подмножества векторов с использованием диаграмм Вороного // Дискретн. анализ и исслед. опер. 2016. Т. 23. № 4. С. 102–115.
21. *Шенмайер В.В.* Аппроксимационная схема для одной задачи поиска подмножества векторов // Дискретн. анализ и исслед. опер. 2012. Т. 19. № 2. С. 92–100.
22. *Sahni S., Gonzalez T.* P-Complete Approximation Problems // Journal of the ACM. 1976. V. 23. P. 555–566.
23. *Brucker P.* On the complexity of clustering problems // Lecture Notes in Economics and Mathematical Systems. 1978. V. 157. P. 45–54.
24. *Indyk P.* A sublinear time approximation scheme for clustering in metric space // Proc. of the 40th Ann. IEEE Symp. on Foundations of Computer Science (FOCS). 1999. P. 154–159.
25. *de la Vega F., Karpinski M., Kenyon C., Rabani Y.* Polynomial time approximation schemes for metric min-sum clustering // Electronic Colloquium on Computational Complexity (ECCC), Report № 25. 2002.
26. *Кельманов А.В., Хандеев В.И.* Рандомизированный алгоритм для одной задачи двухкластерного разбиения множества векторов // Ж. вычисл. матем. и матем. физ. 2015. Т. 55. № 2. С. 335–344.
27. *Wirth N.* Algorithms + Data Structures = Programs. Prentice Hall, New Jersey, 1976. 366 p.