FPTAS для квадратичных евклидовых задач 2-кластеризации множества и последовательности

Кельманов А. В., Хамидуллин С. А., Хандеев В. И.

Институт математики им. С. Л. Соболева СО РАН,
Новосибирский государственный университет,
Новосибирск

17-я Всероссийская конференция
«Математические методы распознавания образов»

г. Светлогорск, 19–25 сентября 2015 г.

Введение

Предмет исследования —

некоторые NP-трудные в сильном смысле задачи разбиения множества и последовательности.

Цель исследования —

обоснование вполне полиномиальной приближённой схемы (FPTAS) для специальных случаев этих задач (когда размерность пространства фиксирована).

Мотивация исследования:

1) отсутствие FPTAS для общего случая этих задач (если $P \neq NP$);
2) поиск подклассов задач, для которых такая схема существует.

Формулировка задачи (разбиение последовательности)

Minimum Sum-of-Squares Clustering problem on sequence with given center of one cluster and cluster cardinalities

Дано: последовательность $\mathcal{Y} = (y_1, \ldots, y_N)$ точек из $\mathbb{R}^q$, натуральные числа $T_{\min}$, $T_{\max}$ и $M > 1$.

Найти: подмножество $\mathcal{M} = \{n_1, \ldots, n_M\} \subseteq \mathcal{N}$ номеров элементов последовательности $\mathcal{Y}$ такое, что

$$F(\mathcal{M}) = \sum_{j \in \mathcal{M}} \|y_j - \overline{y}(\mathcal{M})\|^2 + \sum_{i \in \mathcal{N} \setminus \mathcal{M}} \|y_i\|^2 \to \min,$$

где $\overline{y}(\mathcal{M}) = \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} y_i$, при ограничениях

$$1 \leq T_{\min} \leq n_m - n_{m-1} \leq T_{\max} \leq N, \ m = 2, \ldots, M,$$

на элементы искомого набора $\mathcal{M}$.

Далее для краткости эту задачу будем называть задачей 1.

Формулировка задачи (разбиение множества)

Minimum Sum-of-Squares Clustering problem with given center of one cluster and cluster cardinalities

Дано: множество $\mathcal{Y} = (y_1, \ldots, y_N)$ точек из $\mathbb{R}^q$ и натуральное число $M > 1$.

Найти: подмножество $\mathcal{C} \subseteq \mathcal{Y}$ мощности M такое, что

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

где $\overline{y}(\mathcal{C}) = \frac{1}{|\mathcal{C}|} \sum_{i \in \mathcal{C}} y$ — центроид кластера $\mathcal{C}$.

Далее для краткости эту задачу будем называть задачей 2.

Содержательная трактовка

Области приложений (истоки задач)

Математическая статистика, теория приближения, комбинаторная геометрия, анализ данных и распознавание образов.

Содержательная проблема из области анализа данных (Задача 1)

Имеется таблица, содержащая **упорядоченные по времени** результаты многократных измерений набора числовых информационно значимых характеристик некоторого **объекта**, который может находиться двух состояниях: **пассивном** и **активном**.

Содержательная трактовка

Содержательная проблема из области анализа данных (Задача 1)

Предполагается, что:

1) в пассивном состоянии все числовые характеристики из набора равны нулю, а в любом активном — значение хотя бы одной характеристики не равно нулю;

2) в каждом результате измерения, представленном в таблице, имеется ошибка;

3) соответствие элементов таблицы какому-либо состоянию объекта неизвестно;

4) временной интервал между двумя последовательными активными состояниями объекта ограничен сверху и снизу некоторыми константами.

Содержательная трактовка

Содержательная проблема из области анализа данных (Задача 1)

Требуется:

1) разбить таблицу на подмножества наборов, соответствующих пассивному и активному состояниям объекта, используя критерий минимума суммы квадратов расстояний;

2) оценить по результатам измерения наборы характеристик объекта в активном состоянии (учитывая, что данные содержат ошибку измерения).

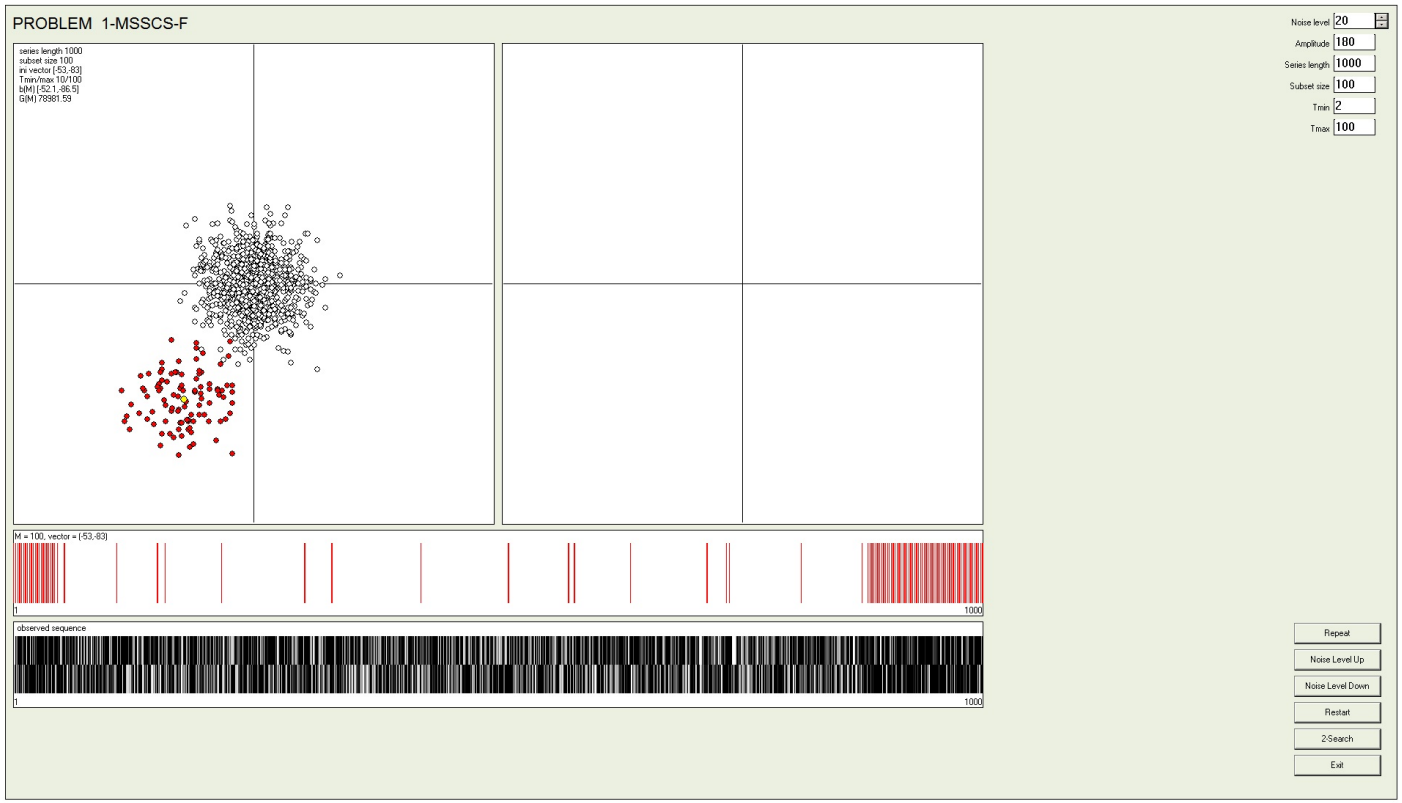
Замечание

Критерий минимума суммы квадратов расстояний обусловлен оптимизационной (аппроксимационной) моделью помехоустойчивого анализа данных.

Содержательная трактовка

Пример

1000 результатов измерений характеристик объекта, изображённые на плоскости и в виде последовательности.
100 раз были измерены характеристики объекта в активном состоянии и 900 — в пассивном.



Известные результаты

Известные результаты (Задача 1)

1. Задача NP-трудна в сильном смысле. Поэтому для этой задачи не существует ни точного полиномиального, ни точного псевдополиномиального алгоритмов, если P≠NP (Кельманов, Пяткин, 2013).

2. Параметрический вариант задачи 1 (когда $T_{\min}, T_{\max}$ — параметры) (Кельманов, Пяткин, 2013):
(1) NP-трудна в сильном смысле, когда $T_{\min} < T_{\max}$;
(2) разрешима за полиномиальное время при $T_{\min} = T_{\max}$.

3. Предложен 2-приближённый полиномиальный алгоритм, временная сложность которого есть величина $\mathcal{O}(N^2(MN + q))$ (Кельманов, Хамидуллин, 2013).

4. Для случая, когда компоненты точек целочисленны, а размерность q пространства фиксирована, обоснован точный псевдополиномиальный алгоритм, который находит решение задачи за время $\mathcal{O}(MN^2(MD)^q)$ (Кельманов, Хамидуллин, Хандеев, 2015).

Известные результаты

Известные результаты (Задача 2)

1. Задача NP-трудна в сильном смысле (Кельманов А. В., Пяткин А. В. 2008).

2. 2-приближённый алгоритм с временной сложностью $\mathcal{O}(qN^2)$ (Долгушев А. В., Кельманов А. В. 2011).

3. Схема PTAS, временная сложность которой $\mathcal{O}(qN^{2/\varepsilon+1}(9/\varepsilon)^{3/\varepsilon})$, где ε — относительная погрешность (Долгушев А. В., Кельманов А. В., Шенмайер В. В. 2012).

4. Рандомизированный алгоритм, позволяющий для установленного значения параметра при фиксированных ε и γ находить $(1+\varepsilon)$-приближённое решение с вероятностью 1-γ за время $\mathcal{O}(qN)$ (Кельманов А. В., Хандеев В. И. 2013);
найдены условия, при которых алгоритм асимптотически точен и имеет временную сложность $\mathcal{O}(qN^2)$ (Кельманов А. В., Хандеев В. И. 2014).

Известные результаты (Задача 2)

5. Установлено, что задача разрешима за время $\mathcal{O}(q^2 N^{2q})$, полиномиальное в случае, когда размерность q пространства фиксирована;
предложен точный псевдополиномиальный алгоритм для случая, когда компоненты векторов целочисленны, а размерность пространства фиксирована; временная сложность алгоритма есть величина $\mathcal{O}(N(MD)^q)$; здесь D — максимальное абсолютное значение координат векторов входного множества (Кельманов, Хандеев, 2014).

Новые результаты

Новый результат настоящей работы (Задача 1)

Предложен приближённый алгоритм, имеющий временную сложность $\mathcal{O}(N^2(M(T_{\max} - T_{\min} + 1) + q)(\sqrt{\frac{2q}{\varepsilon}} + 1)^q)$, где ε — относительная погрешность; в случае фиксированной размерности пространства алгоритм имеет трудоёмкость $\mathcal{O}(MN^3(1/\varepsilon)^{q/2})$ и реализует схему FPTAS.

Новый результат настоящей работы (Задача 2)

Предложен приближённый алгоритм, имеющий временную сложность $\mathcal{O}(qN^2(\sqrt{\frac{2q}{\varepsilon}} + 1)^q)$, где ε — относительная погрешность; в случае фиксированной размерности пространства алгоритм имеет трудоёмкость $\mathcal{O}(N^2(1/\varepsilon)^{q/2})$ и реализует схему FPTAS.

Задача 1. Суть подхода

Суть подхода к алгоритмическому решению

1. Для каждой точки входного множества строится область (куб) так, что одна из этих областей гарантировано включает центр искомого подмножества.

2. По заданной на входе желаемой относительной погрешности решения строится сетка (решётка), дискретизирующая куб с равномерным по всем координатам шагом.

3. Для каждого узла решётки с помощью схемы динамического программирования решается задача максимизации вспомогательной целевой функции и строится набор номеров элементов последовательности, доставляющий максимум этой функции. Сформированный набор объявляется претендентом на решение.

4. В качестве окончательного решения выбирается то подмножество-претендент, которое доставляет наименьшее значение целевой функции.

Вспомогательная задача

Положим

$$Q(\mathcal{M}, x) = \sum_{n \in \mathcal{M}} \|y_n - x\|^2 + \sum_{n \in \mathcal{N} \setminus \mathcal{M}} \|y_n\|^2, \qquad (1)$$

где $\mathcal{M} \subseteq \mathcal{N}$, $x \in \mathbb{R}^q$, а y_n — элементы последовательности $\mathcal{Y}$.

Лемма 1

1. При любом фиксированном подмножестве $\mathcal{M} \subseteq \mathcal{N}$ минимум целевлй функции (1) доставляется точкой $x = \overline{y}(\mathcal{M})$ и равен $F(\mathcal{M})$.

2. При любой фиксированной точке $x \in \mathbb{R}^q$ минимум целевой функции (1) достигается на наборе элементов, для которых сумма проекций на луч из начала координат в точку x максимальна.

Вспомогательная задача

Для произвольной фиксированной точки $x \in \mathbb{R}^q$ положим

$$g^x(n) = \langle y_n, x \rangle, \quad n \in \mathcal{N},$$

где y_n — n-й элемент входной последовательности $\mathcal{Y}$, и

$$G^x(\mathcal{M}) = \sum_{n \in \mathcal{M}} g^x(n), \quad \mathcal{M} \subseteq \mathcal{N}. \tag{2}$$

Задача 3

Дано: последовательность $\mathcal{Y} = (y_1, \ldots, y_N)$ точек из $\mathbb{R}^q$, точка $x \in \mathbb{R}^q$, натуральные числа $T_{\min}$, $T_{\max}$ и $M > 1$.
Найти: подмножество $\mathcal{M} = \{n_1, \ldots, n_M\} \subseteq \mathcal{N}$ номеров элементов последовательности $\mathcal{Y}$, доставляющее максимум целевой функции (2) при ограничениях

$$1 \leq T_{\min} \leq n_m - n_{m-1} \leq T_{\max} \leq N, \ m = 2, \ldots, M.$$

Для решения этой задачи обоснована следующая схема динамического программирования.

Лемма 2

Для любого натурального $M > 1$ такого, что $(M-1)T_{\min} \leq N-1$, и для произвольной точки $x \in \mathbb{R}^q$ оптимальное значение $G^x_{\max} = \max_{\mathcal{M}} G^x(\mathcal{M})$ целевой функции задачи 3 находится по формуле

$$G^x_{\max} = \max_{n \in \omega_M} G^x_M(n), \tag{3}$$

а значения функции $G^x_M(n)$, $n \in \omega_M$, вычисляются по следующим рекуррентным формулам

$$G^x_m(n) = \begin{cases} g^x(n), & \text{если } n \in \omega_1, \\ & m = 1; \\ g^x(n) + \max\limits_{j \in \gamma^-_{m-1}(n)} G^x_{m-1}(j), & \text{если } n \in \omega_m, \\ & m = 2, \ldots, M, \end{cases} \tag{4}$$

где множества ω_m и $\gamma^-_{m-1}(n)$ задаются следующими формулами:

Вспомогательная задача. Алгоритм решения

Лемма 2

$$\omega_m = \left\{ n \,|\, 1 + (m-1)\,T_{\min} \le n \le N - (M-m)\,T_{\min} \right\},\ m = 1, \ldots, M,$$

$$\gamma^{-}_{m-1}(n) = \left\{ j \,|\, \max\{1 + (m-2)\,T_{\min},\, n - T_{\max}\} \le j \le n - T_{\min} \right\},$$
$$n \in \omega_m,\ m = 2, \ldots, M.$$

Следствие 1

Элементы $n^x_1, \ldots, n^x_M$ оптимального набора $\mathcal{M}^x = \arg\max_{\mathcal{M}} G^x(\mathcal{M})$ находятся по следующим рекуррентным формулам:

$$n^x_M = \arg \max_{n \in \omega_M} G^x_M(n), \tag{5}$$

$$n^x_{m-1} = \arg \max_{n \in \gamma^{-}_{m}(n^x_m)} G^x_m(n), \quad m = M, M-1, \ldots, 2. \tag{6}$$

Вспомогательная задача. Алгоритм решения

Алгоритм $\mathcal{A}$

Входами алгоритма являются $\mathcal{Y}$, x, $T_{\min}$, $T_{\max}$ и M.
Шаг 1. Вычислим значения $g^x(n)$, $n \in \mathcal{N}$.
Шаг 2. Используя рекуррентные формулы (4), вычислим значения $G_m^x(n)$ для каждого $n \in \omega_m$ и $m = 1, \ldots, M$.
Шаг 3. Найдём значение $G_{\max}^x$ максимума целевой функции G^x по формуле (3) и оптимальный набор $\mathcal{M}^x = (n_1^x, \ldots, n_M^x)$ по формулам (5), (6); выход.

Теорема 1

Алгоритм $\mathcal{A}$ находит оптимальное решение задачи 3 за время $\mathcal{O}(N(M(T_{\max} - T_{\min} + 1) + q))$.

Замечание

В оценке временной сложности алгоритма $\mathcal{A}$ множитель $(T_{\max} - T_{\min} + 1)$ не превосходит N. Поэтому алгоритм полиномиален по N и по q, а его сложность можно оценить как $O(N(MN + q))$.

Задача 1. Алгоритм решения

Лемма 3

Пусть $\mathcal{M}^*$ — оптимальное решение задачи 1, $x \in \mathbb{R}^q$ — произвольная точка, $\mathcal{M}^x$ — оптимальное решение задачи 3, а $t = \arg\min_{y \in \{y_i \mid i \in \mathcal{M}^*\}} \|y - \overline{y}(\mathcal{M}^*)\|$ — точка из множества $\{y_i \mid i \in \mathcal{M}^*\}$, ближайшая к центроиду этого множества.

Тогда для того чтобы при фиксированном $\varepsilon > 0$ множество $\mathcal{M}^x$ было $(1+\varepsilon)$-приближённым решением задачи 1, достаточно, чтобы точка x удовлетворяла неравенству

$$\|x - \overline{y}(\mathcal{M}^*)\|^2 \leq \frac{\varepsilon}{2M} F(\mathcal{M}^t), \tag{7}$$

где $\mathcal{M}^t$ — оптимальное решение задачи 3 при $x = t$.

Замечание

Лемма 3 показывает, насколько близко должна быть точка x к оптимальному центроиду, чтобы условно-оптимальное решение $\mathcal{M}^x$ гарантировало получение $(1+\varepsilon)$-приближённого решения задачи 1.

Задача 1. Алгоритм решения

Лемма 4

Пусть $\mathcal{M}^*$ — оптимальное решение задачи 1, а $t = \arg\min_{y \in \{y_i \mid i \in \mathcal{M}^*\}} \|y - \overline{y}(\mathcal{M}^*)\|$ — точка из множества $\{y_i \mid i \in \mathcal{M}^*\}$, ближайшая к центроиду этого множества.

Тогда для точки $t = \arg\min_{y \in \{y_i \mid i \in \mathcal{M}^*\}} \|y - \overline{y}(\mathcal{M}^*)\|$ справедлива оценка

$$\|t - \overline{y}(\mathcal{M}^*)\|^2 \leq \frac{1}{M} F(\mathcal{M}^t), \tag{8}$$

где $\mathcal{M}^t$ — оптимальное решение задачи 3 при $x = t$.

Замечание

Лемма 4 даёт оценку сверху на расстояние от оптимального центроида до ближайшей к нему точки из входного множества.

Задача 1. Алгоритм решения

Для произвольной точки $y \in \mathbb{R}^q$ и положительных чисел h, H определим множество точек

$$\mathcal{D}(y,h,H) = \{d \mid d = y + h(j_1,\ldots,j_q),\ j_i \in \mathbb{Z},\ |h\cdot j_i| \le H,\ i = 1,\ldots,q\}.$$

Замечания

1. Имеет место оценка

$$|\mathcal{D}(y,h,H)| \le (2\lfloor\frac{H}{h}\rfloor + 1)^q \le (2\frac{H}{h} + 1)^q.$$

2. Для любого $x \in \mathbb{R}^q$ такого, что $\|y - x\| \le H$, расстояние до ближайшей точки из множества $\mathcal{D}(y,h,H)$ не превосходит $\frac{h\sqrt{q}}{2}$.

Для произвольных $\varepsilon > 0$ и $y \in \mathcal{Y}$ положим

$$h = \sqrt{\frac{2\varepsilon}{qM}F(\mathcal{M}^y)}, \qquad H = \sqrt{\frac{1}{M}F(\mathcal{M}^y)}.$$

Задача 1. Алгоритм решения

Алгоритм $\mathcal{A}_1$

Вход алгоритма: множество $\mathcal{Y}$, числа $T_{\min}$, $T_{\max}$, M и ε.

Для каждой точки $y \in \mathcal{Y}$ выполним шаги 1–5.

Шаг 1. С помощью алгоритма $\mathcal{A}$ найдём оптимальное решение $\mathcal{M}^y$ задачи 3 при $x = y$.

Шаг 2. Вычислим $F(\mathcal{M}^y)$, h и H.

Шаг 3. Если $F(\mathcal{M}^y) = 0$, то множество $\mathcal{M}^y$ объявим результатом работы алгоритма; выход. Иначе переходим к следующему шагу.

Шаг 4. Построим решётку $\mathcal{D}(y, h, H)$.

Шаг 5. Для каждой точки d решётки $\mathcal{D}(y, h, H)$ построим оптимальное решение $\mathcal{M}^d$ задачи 3 при $x = d$ с помощью алгоритма $\mathcal{A}$ и вычислим значение $F(\mathcal{M}^d)$.

Шаг 6. В семействе $\{\mathcal{M}^d \mid d \in \mathcal{D}(y, h, H),\ y \in \mathcal{Y}\}$ множеств в качестве решения выберем то множество $\mathcal{M}^d$, для которого значение $F(\mathcal{M}^d)$ минимально.

Выход.

Задача 1. Алгоритм решения

Теорема 2

Для любого фиксированного $\varepsilon > 0$ алгоритм $\mathcal{A}_1$ находит $(1+\varepsilon)$-приближённое решение задачи 1 за время $\mathcal{O}(N^2(M(T_{\max} - T_{\min} + 1) + q)(\sqrt{\frac{2q}{\varepsilon}} + 1)^q)$.

Замечание

Если размерность q пространства фиксирована, то трудоёмкость $\mathcal{O}(N^2(M(T_{\max} - T_{\min} + 1) + q)(\sqrt{\frac{2q}{\varepsilon}} + 1)^q)$ алгоритма оценивается величиной $\mathcal{O}(MN^3(1/\varepsilon)^{q/2})$, так как

$$(\sqrt{\frac{2q}{\varepsilon}} + 1)^q \leq 2^q(\sqrt{\frac{2q}{\varepsilon}})^q = 2^{3q/2}q^{q/2}(1/\varepsilon)^{q/2} = \mathcal{O}((1/\varepsilon)^{q/2}).$$

Таким образом, в указанном случае алгоритм $\mathcal{A}_1$ реализует схему FPTAS.

Задача 2. Алгоритм решения

Для произвольной точки $y \in \mathcal{Y}$ обозначим через $\mathcal{C}^x$ множество из M элементов множества $\mathcal{Y}$, имеющих наибольшие проекции на луч из начала координат в точку x.

Для произвольных $\varepsilon > 0$ и $y \in \mathcal{Y}$ положим

$$h = \sqrt{\frac{2\varepsilon}{qM} S(\mathcal{C}^y)}, \qquad H = \sqrt{\frac{1}{M} S(\mathcal{C}^y)}.$$

Задача 2. Алгоритм решения

Алгоритм $\mathcal{A}_2$

Вход алгоритма: множество $\mathcal{Y}$, числа M и ε.

Для каждого вектора $y \in \mathcal{Y}$ выполним шаги 1–5.

Шаг 1. Построим множество $\mathcal{C}^y$.

Шаг 2. Вычислим $S(\mathcal{C}^y)$, h и H .

Шаг 3. Если $S(\mathcal{C}^y) = 0$, то множество $\mathcal{C}^y$ объявим результатом работы алгоритма. Иначе переходим к следующему шагу.

Шаг 4. Построим множество $\mathcal{D}(y, h, H)$.

Шаг 5. Для каждого вектора $d \in \mathcal{D}(y, h, H)$ построим множество $\mathcal{C}^d$ и вычислим значение $S(\mathcal{C}^d)$.

Шаг 6. В качестве решения выберем множество $\mathcal{C}^d$, для которого значение $S(\mathcal{C}^d)$ минимально.

Выход.

Задача 2. Алгоритм решения

Теорема 3

Для любого фиксированного $\varepsilon > 0$ алгоритм $\mathcal{A}_2$ находит $(1+\varepsilon)$-приближённое решение задачи 2 за время $\mathcal{O}(qN^2(\sqrt{\frac{2q}{\varepsilon}}+1)^q)$.

Замечание

Если размерность q пространства фиксирована, то трудоёмкость $\mathcal{O}(qN^2(\sqrt{\frac{2q}{\varepsilon}}+1)^q)$ алгоритма оценивается величиной $\mathcal{O}(N^2(1/\varepsilon)^{q/2})$, так как

$$(\sqrt{\frac{2q}{\varepsilon}}+1)^q = (2q)^{q/2}(\frac{1}{\sqrt{\varepsilon}}+\frac{1}{\sqrt{2q}})^q \le (2q)^{q/2}2^q(\frac{1}{\sqrt{\varepsilon}})^q$$
$$= 2^{3q/2}q^{q/2}(1/\varepsilon)^{q/2} = \mathcal{O}((1/\varepsilon)^{q/2}).$$

Таким образом, в указанном случае алгоритм $\mathcal{A}_2$ реализует схему FPTAS.

Заключение

Обоснована схема FPTAS для специальных случаев NP-трудных в сильном смысле задач разбиения конечной последовательности и конечного множества точек евклидова пространства на два кластера.

Рассмотренный случай задач, предполагающий фиксированность размерности пространства, актуален для многих приложений, в которых цифровой ввод данных и их последующая обработка компьютерными системами является неотъемлемым элементом.

Важными направлениями дальнейших исследований являются обоснование алгоритмов другого типа (асимптотически точных, рандомизированных) для решения задачи разбиения последовательности, а также построение алгоритмов с оценками точности для обобщения задач на случай нескольких кластеров.

Спасибо за внимание!