Приближённый полиномиальный алгоритм для одной задачи разбиения последовательности

Кельманов А. В., Хамидуллин С. А.

Институт математики им. С. Л. Соболева СО РАН, Новосибирск

10-я Международная конференция «Интеллектуализация обработки информации» (ИОИ-2014)

о. Крит, 4–11 октября, 2014 г.

Введение

Предмет исследования —

одна из труднорешаемых задач дискретной оптимизации.

Цель исследования —

обоснование приближенного полиномиального алгоритма для её решения.

Мотивация исследования —

отсутствие каких-либо эффективных алгоритмов с гарантированными оценками точности для решения этой задачи.

Задача 1-MSSC-S-NF

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

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

$$F(\mathcal{M}) = \sum_{j \in \mathcal{M}} \|y_j - \overline{y}(\mathcal{M})\|^2 + \sum_{i \in \mathcal{N} \setminus \mathcal{M}} \|y_i\|^2 \to \min, \qquad (1)$$

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

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

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

Имеется

таблица, содержащая упорядоченные по времени результаты многократных измерений набора числовых информационно значимых характеристик некоторого объекта, который может находиться в пассивном и активном состоянии.

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.

Требуется:

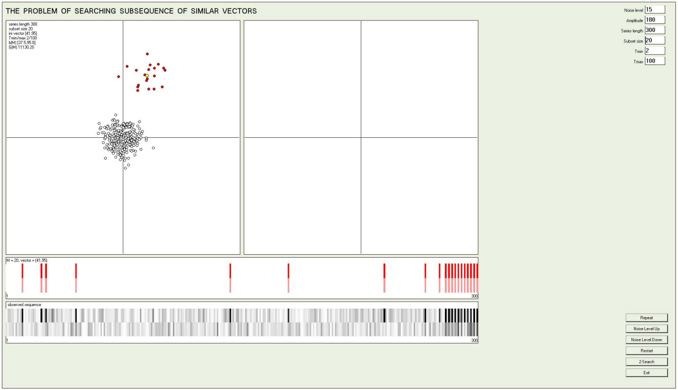
1. разбить таблицу на подмножества наборов, соответствующих пассивному и активному состояниям объекта, используя критерий минимума суммы квадратов расстояний,
2. оценить по результатам измерения наборы характеристик объекта в активном состоянии (учитывая, что данные содержат ошибку измерения).

Замечание

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

Пример. Задача 1-MSSC-S-NF

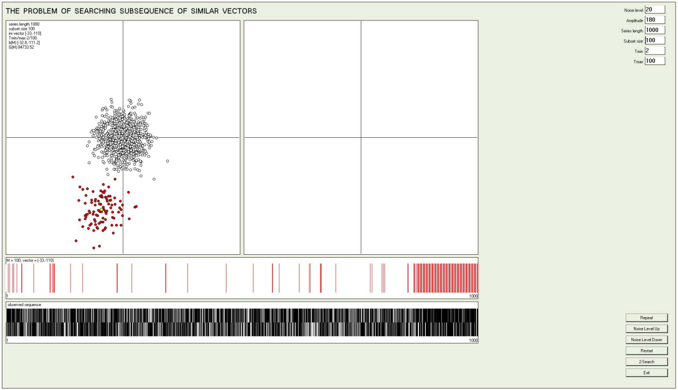
300 результатов измерений характеристик объекта, изображенные на плоскости и в виде последовательности.
20 раз были измерены характеристики объекта в активном состоянии и 280 — в пассивном.



Задача J-MSSC-S-NF

Пример. Задача 1-MSSC-S-NF

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



Задача 1-MSSC-S-NF

Известные результаты

1. Задача NP-трудна в сильном смысле. Поэтому для этой задачи не существует ни точного полиномиального, ни точного псевдополиномиального алгоритмов, если P$\neq$NP (Кельманов, Пяткин, 2009).
2. Параметрический вариант — задача 1-MSSC-S-NF($T_{\min}$, $T_{\max}$) (Кельманов, Пяткин, 2013)

- NP-трудна в сильном смысле, когда $T_{\min} < T_{\max}$;
- разрешима за полиномиальное время при $T_{\min} = T_{\max}$.

3. Какие-либо полиномиальные алгоритмы с оценками точности до настоящего времени отсутствовали.

Задача 1-MSSC-S-NF

Новый результат

Обоснован 2-приближенный полиномиальный алгоритм (Кельманов, Хамидуллин, 2014).

Задача 1-MSSC-S-NF. 2-приближенный алгоритм

Суть подхода

1. Заменим решение исходной задачи 1-MSSC-S-NF решением более простой (вспомогательной) задачи.

2. Построим точный полиномиальный алгоритм ее решения.

3. Оценим точность такой замены (приближения).

Подход опирается на записи целевой функции задачи 1-MSSC-S-NF в виде

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

$$= \sum_{j\in\mathcal{N}} \|y_j\|^2 - \sum_{i\in\mathcal{M}} \left(2(y_i, \overline{y}(\mathcal{M})) - \|\overline{y}(\mathcal{M})\|^2\right).$$

Задача 1-MSSC-S-NF. Вспомогательная задача

Задача 1

Дано: последовательность $\mathcal{Y} = (y_1, \ldots, y_N)$ векторов из $\mathbb{R}^q$, вектор $b \in \mathbb{R}^q$, натуральные числа $T_{\min}$ и $T_{\max}$.

Найти: набор $\mathcal{M} = \{n_1, \ldots, n_M\} \subseteq \mathcal{N}$ номеров элементов $\mathcal{Y}$ такой, что

$$G(\mathcal{M}) = \sum_{n \in \mathcal{M}} g(n) \to \max, \tag{3}$$

где

$$g(n) = 2(y_n, b) - \|b\|^2, \ n \in \mathcal{N}, \tag{4}$$

при ограничениях (2) на элементы набора $\mathcal{M}$.

Алгоритм решения вспомогательной задачи

Лемма 1

Пусть $M \le \lfloor (N-1)/T_{\min} \rfloor + 1$. *Тогда оптимальное значение* $G_{\max}$ *целевой функции задачи 1 находится по формуле*

$$G_{\max} = \max_{n \in \mathcal{N}} G_n, \tag{5}$$

а значения функции G_n *вычисляются по следующим рекуррентным формулам*

$$G_n = \begin{cases} g(n), & \text{если } n \in \{1, \dots, T_{\min}\}; \\ g(n) + \max\{0, \max\limits_{j \in \gamma(n)} G_j\}, & \text{если } n \in \{T_{\min}+1, \dots, N\}, \end{cases} \tag{6}$$

где множество γ *задается формулой*

$$\gamma(n) = \{j \mid \max\{1, n - T_{\max}\} \le j \le n - T_{\min}\}.$$

Формулы леммы реализуют схему динамического программирования.

Для отыскания оптимального набора определим функцию $I(n) : \mathcal{N} \to \{0\} \cup \mathcal{N}$ по следующей формуле

$$I(n) = \begin{cases} 0, & \text{если } n \in \{1, \dots, T_{\min}\}, \\ 0, & \text{если } n \in \{T_{\min}+1, \dots, N\} \text{ и } \max\limits_{j \in \gamma(n)} G_j < 0, \\ \arg\max\limits_{j \in \gamma(n)} G_j, & \text{если } n \in \{T_{\min}+1, \dots, N\} \text{ и } \max\limits_{j \in \gamma(n)} G_j \geq 0. \end{cases} \tag{7}$$

Формулы для отыскания оптимального набора

$$\widehat{\mathcal{M}} = (\widehat{n}_1, \dots, \widehat{n}_{\widehat{M}})$$

устанавливает

Следствие 1

Пусть выполнены условия леммы 1, G_n и $I(n)$, $n \in \mathcal{N}$, — *функции, заданные формулами* (6) и (7), *соответственно. Тогда последняя компонента оптимального набора вычисляется по формуле*

$$\widehat{n}_{\widehat{M}} = \arg\max_{n \in \mathcal{N}} G_n. \tag{8}$$

Если $I(\widehat{n}_{\widehat{M}}) > 0$, *то оставшиеся компоненты этого набора находятся по следующим рекуррентным формулам*

$$\widehat{n}_{m-1} = I(\widehat{n}_m), \;\; m = \widehat{M}, \widehat{M} - 1, \ldots \tag{9}$$

Если на некотором шаге m *вычислений по формуле* (9) *имеет место равенство* $I(\widehat{n}_m) = 0$, *то* $m = 1$ *и найдены все компоненты оптимального набора.*

Алгоритм $\mathcal{A}_1$

Входами алгоритма являются $\mathcal{Y}$, b, $T_{\min}$ и $T_{\max}$.

Шаг 1. Вычислим значения $g(n)$, $n \in \mathcal{N}$, по формуле (4).

Шаг 2. Используя формулы (6) и (7), вычислим значения G_n и $I(n)$ для каждого $n \in \mathcal{N}$. Найдем значение $G_{\max}$ максимума целевой функции G по формуле (5).

Шаг 3. Вычислим элементы оптимального набора $\widehat{\mathcal{M}} = \{\widehat{n}_1, \ldots, \widehat{n}_{\widehat{M}}\}$ по формулам (8) и (9); выход.

Выходом алгоритма являются значения, зависящие от b, т.е. $G_{\max} = G_{\max}(b)$, $\widehat{\mathcal{M}} = \widehat{\mathcal{M}}(b) = \{\widehat{n}_1(b), \ldots, \widehat{n}_{\widehat{M}}(b)\}$ и $\widehat{M} = \widehat{M}(b)$.

Теорема 1

Алгоритм $\mathcal{A}_1$ *находит оптимальное решение задачи* 1 *за время* $\mathcal{O}(N(T_{\max} - T_{\min} + q))$.

Доказательство. Оптимальность решения следует из леммы 1.

На **первом шаге** алгоритма требуется $\mathcal{O}(Nq)$ операций.

Трудоемкость **второго шага**, выполняемого для каждого $n = 1, \ldots, N$, определяется мощностью множеств $\mathcal{N}$ и $\gamma(n)$, входящих в определение функции (6). Поэтому трудоемкость шага 2 есть величина $\mathcal{O}(N(T_{\max} - T_{\min} + 1))$.

Из формул (5), (8) и (9) видно, что на **шаге 3** требуется не более $\mathcal{O}(M_{\max})$ операций, что не превышает $\mathcal{O}(N)$, так как $M_{\max} \leq N$.

Просуммировав затраты на всех шагах, получим оценку временной сложности, приведенную в формулировке теоремы.

Замечание 1

Из ограничений (2) *следует двойное неравенство* $0 \le T_{\max} - T_{\min} \le N - 1$. *Поэтому алгоритм* $\mathcal{A}_1$ *полиномиален по* N *и по* q, *а его сложность в общем случае можно оценить как* $\mathcal{O}(N(N+q))$ и как $\mathcal{O}(Nq)$ в частном случае, когда $T_{\max} = T_{\min}$.

Алгоритм $\mathcal{A}$

Входами алгоритма являются $\mathcal{Y}$, $T_{\min}$, $T_{\max}$ и M.
Шаг 1. Положим $i = 0, M_A = 0, \mathcal{M}_A = \emptyset, H = -\infty$.
Шаг 2. $i := i + 1$; положим $b = y_i$.
Шаг 3. Для фиксированного вектора $b \in \mathcal{Y}$ найдем оптимальное решение $\widehat{\mathcal{M}}(b)$ и значение $G_{\max}(b)$ целевой функции задачи 1 с помощью алгоритма $\mathcal{A}_1$.
Шаг 4. Если $H < G_{\max}(b)$, то положим $b_A = b$, $H = G_{\max}(b)$, $M_A = \widehat{M}(b)$, $\mathcal{M}_A = \widehat{\mathcal{M}}(b)$.
Шаг 5. Если $i < N$, то переходим на Шаг 2, иначе — к следующему шагу.
Шаг 6. Вычислим вектор $\overline{y}(\mathcal{M}_A) = (1/M_A) \sum_{n \in \mathcal{M}_A} y_n$ и значение $F(\mathcal{M}_A)$ целевой функции по формуле (1); положим $G_{\max}(b_A) = H$; выход.
Выходом алгоритма (решением задачи) объявляем набор $\mathcal{M}_A$, значение $F(\mathcal{M}_A)$, а также векторы $\overline{y}(\mathcal{M}_A)$ и b_A. Если максимуму $G_{\max}(b_A)$ соответствует несколько наборов $\widehat{\mathcal{M}}_A$, то выбираем любой из них.

Алгоритм $\mathcal{A}$

Суть алгоритма $\mathcal{A}$ состоит в решении задачи 1 с помощью алгоритма $\mathcal{A}_1$ для каждого вектора последовательности $\mathcal{Y}$ и последующего выбора из найденных решений (наборов) наилучшего набора $\widehat{\mathcal{M}}_A$, которому соответствует наибольшее значение $G_{\max}(b_A)$ целевой функции задачи 1.

Приведем вспомогательное утверждение.

Лемма 2

Пусть $\mathcal{Z}$ — непустое конечное множество векторов из $\mathbb{R}^q$, а $\overline{z}(\mathcal{Z}) = \frac{1}{\mathcal{Z}} \sum_{z \in \mathcal{Z}} z$. Тогда, если вектор $x \in \mathbb{R}^q$ удовлетворяет условиям

$$\|x - \overline{z}\| \leq \|z - \overline{z}\|, \qquad \forall z \in \mathcal{Z},$$

то имеет место неравенство

$$\sum_{z \in \mathcal{Z}} \|z - x\|^2 \leq 2 \sum_{z \in \mathcal{Z}} \|z - \overline{z}\|^2.$$

Теорема 2

Алгоритм $\mathcal{A}$ находит 2-приближенное решение задачи 1-MSSC-S-NF за время $\mathcal{O}(N^2(T_{\max} - T_{\min} + q))$. Оценка 2 точности алгоритма достижима.

Доказательство теоремы 2

Пусть $\mathcal{M}^*$ — оптимальное решение задачи 1-MSSC-S-NF,

$\mathcal{C}^* = \{y_n | n \in \mathcal{M}^*\}$ — подмножество векторов из $\mathcal{Y}$, соответствующих оптимальному набору $\mathcal{M}^*$,

$\bar{y}(\mathcal{M}^*) = \frac{1}{|\mathcal{M}^*|} \sum_{n \in \mathcal{M}^*} y_n$ — центр подмножества $\mathcal{C}^* \subseteq \mathcal{Y}$,

$u = \arg\min_{y \in \mathcal{C}^*} \|y - \bar{y}(\mathcal{M}^*)\|$ — вектор, ближайший к центру подмножества $\mathcal{C}^*$.

Согласно пошаговой записи алгоритм находит вектор

$$b_A = \arg\max_{b \in \mathcal{Y}} G_{\max}(b) \tag{10}$$

из множества $\mathcal{Y}$, набор $\mathcal{M}_A = \widehat{\mathcal{M}}(b_A) = \{\hat{n}_1(b_A), \ldots, \hat{n}_{\widehat{M}}(b_A)\}$, вектор $\overline{y}(\mathcal{M}_A)$, значение $F(\mathcal{M}_A)$ целевой функции задачи 1-MSSC-S-NF, а также максимум

$$G_{\max}(b_A) = \sum_{n \in \mathcal{M}_A} \{2(y_n, b_A) - \|b_A\|^2\} = \max_{b \in \mathcal{Y}} \max_{\mathcal{M}} \sum_{n \in \mathcal{M}} \{2(y_n, b) - \|b\|^2\} \tag{11}$$

функции $G_{\max}(b)$, $b \in \mathcal{Y}$.

Справедливость утверждения теоремы вытекает из следующей цепочки равенств и неравенств

$$F(\mathcal{M}_A) = \sum_{n\in\mathcal{M}_A} \|y_n - \bar{y}(\mathcal{M}_A)\|^2 + \sum_{n\in\mathcal{N}\setminus\mathcal{M}_A} \|y_n\|^2$$
$$\leq_{(1)} \sum_{n\in\mathcal{M}_A} \|y_n - b_A\|^2 + \sum_{n\in\mathcal{N}\setminus\mathcal{M}_A} \|y_n\|^2$$
$$= \sum_{n\in\mathcal{N}} \|y_n\|^2 - \sum_{n\in\mathcal{M}_A} \{2(y_n, b_A) - \|b_A\|^2\} =$$

Неравенство 1 справедливо, так как для любого конечного множества $\mathcal{Z}$ векторов из $\mathbb{R}^q$ (в частности, для $\mathcal{Z} = \{y_n | n \in \mathcal{M}_A\}$) минимум суммы квадратов $\sum\limits_{z\in\mathcal{Z}} \|z - c\|^2$ по c достигается в точке $c = \bar{z}(\mathcal{Z}) = \frac{1}{|\mathcal{Z}|}\sum_{z\in\mathcal{Z}} z$ (т.е. в точке $c = \bar{y}(\mathcal{M}_A)$).

$$
\begin{aligned}
&=_{(2)} \sum_{n\in\mathcal{N}} \|y_n\|^2 - \max_{b\in\mathcal{Y}} \max_{\mathcal{M}} \sum_{n\in\mathcal{M}} \{2(y_n, b) - \|b\|^2\} \\
&\qquad =_{(3)} \min_{b\in\mathcal{Y}} \min_{\mathcal{M}} \Big(\sum_{n\in\mathcal{N}} \|y_n\|^2 - \sum_{n\in\mathcal{M}} \{2(y_n, b) - \|b\|^2\} \Big) \\
&\qquad\qquad = \min_{b\in\mathcal{Y}} \min_{\mathcal{M}} \Big(\sum_{n\in\mathcal{M}} \|y_n - b\|^2 + \sum_{n\in\mathcal{N}\setminus\mathcal{M}} \|y_n\|^2 \Big) \leq
\end{aligned}
$$

Равенство 2 следует из теоремы 1 и формул (10), (11).

Равенство 3 справедливо, т.к. $\sum_{n\in\mathcal{N}} \|y_n\|^2$ — константа.

$$
\begin{aligned}
&\leq_{(4)} \sum_{n\in\mathcal{M}^*} \|y_n - u\|^2 + \sum_{n\in\mathcal{N}\setminus\mathcal{M}^*} \|y_n\|^2 \\
&\qquad \leq_{(5)} 2 \sum_{n\in\mathcal{M}^*} \|y_n - \bar{y}(\mathcal{M}^*)\|^2 + \sum_{n\in\mathcal{N}\setminus\mathcal{M}^*} \|y_n\|^2 \\
&\quad \leq 2\Big(\sum_{n\in\mathcal{M}^*} \|y_n - \bar{y}(\mathcal{M}^*)\|^2 + \sum_{n\in\mathcal{N}\setminus\mathcal{M}^*} \|y_n\|^2\Big) = 2F(\mathcal{M}^*). \qquad (12)
\end{aligned}
$$

Неравенство 4 следует из того, что подмножество $\mathcal{M}^*$ и вектор $\mathbf{u} \in \mathcal{Y}$ — допустимое решение задачи $\min_{b\in\mathcal{Y}} \min_{\mathcal{M}}(\cdot)$.

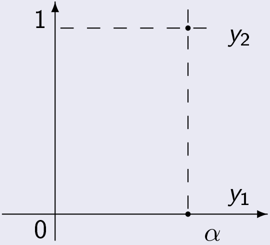
Справедливость неравенства 5 вытекает из леммы 2.

Таким образом, из (12) следует, что $\frac{F(\mathcal{M}_A)}{F(\mathcal{M}^*)} \leq 2$, т.е. алгоритм $\mathcal{A}$ находит 2-приближенное решение.

Покажем, что оценка 2 точности алгоритма достижима.

Для этого, приведем пример, показывающий существование таких входных данных задачи, для которых отношение $F(\mathcal{M}_A)/F(\mathcal{M}^*)$ может быть сколь угодно близко к 2 и равно 2.

Пример достижимости



Пусть $q = 2$, $N = 2$, $T_{\min} = 1$, $T_{\max} = 2$, $\mathcal{Y} = (y_1, y_2)$, где $y_1 = (0, \alpha)$, $y_2 = (1, \alpha)$.

Тогда если $0 < \alpha < 1$, то $\mathcal{M}_A = \{2\}$, $\mathcal{M}^* = \{1, 2\}$, $F(\mathcal{M}_A) = \alpha^2$, $F(\mathcal{M}^*) = 1/2$. Таким образом, отношение $F(\mathcal{M}_A)/F(\mathcal{M}^*) = 2\alpha^2$ может быть сколь угодно близко к 2 при $\alpha \to 1$.

Если же $\alpha = 1$, то имеем два алгоритмических решения: либо $\mathcal{M}_A = \{1, 2\}$, либо $\mathcal{M}_A = \{2\}$. При этом для второго решения $F(\mathcal{M}_A) = 1$; $\mathcal{M}^* = \{1, 2\}$, $F(\mathcal{M}^*) = 1/2$ и $F(\mathcal{M}_A)/F(\mathcal{M}^*) = 2$, т.е. оценка точности алгоритма достижима.

Оценим временную сложность алгоритма. Время вычислений определяется трудоемкостью шага 3. На этом шаге N раз решается вспомогательная задача 1 с помощью алгоритма $\mathcal{A}_1$, трудоемкость которого оценена в теореме 1. Отсюда следует оценка сложности. Теорема 2 доказана.

Замечание 2

В соответствии с замечанием 1 алгоритм $\mathcal{A}$ полиномиален по N и по q, а его сложность можно оценить как $\mathcal{O}(N^2(N+q))$.

Заключение

Обоснован 2-приближенный полиномиальный алгоритм для решения NP-трудной в сильном смысле задачи разбиения последовательности на два кластера, которая индуцируется, в частности, оптимизационной моделью одной из актуальных проблем анализа данных.

Рассмотренная задача относится к числу практически неизученных в алгоритмическом плане. Поэтому исследование вопросов её аппроксимируемости, а также обоснование алгоритмов другого типа (асимптотически точных, рандомизированных и др.) для её решения представляется делом ближайшей перспективы.

Спасибо за внимание!