Модел. и анализ информ. систем. Т.**14**, №**3** (2007) 50–52

УДК 519.987

Экономный алгоритм нахождения средних минимальных расстояний

Тимофеева Н.Е.
Ярославский государственный университет
150 000, Ярославль, Советская, 14
E-mail : ninaveta@gmail.com

получена 15 августа 2007

Аннотация

Пусть заданы $n+1$ строк $\boldsymbol{\xi_0}, \ldots, \boldsymbol{\xi_n}$ с символами из некоторого конечного алфавита. В работе предлагается алгоритм нахождения величин среднего значения k-го минимального расстояния между строками $\boldsymbol{\xi_0}, \ldots, \boldsymbol{\xi_s}$ для всех значений $s \leqslant n$. Трудоемкость алгоритма равна $\mathcal{O}(nm)$, где m – длина строк.

В работе [1] для нахождения оценок размерностей предлагается строить следующую величину:

$$r_n^{(k)} = \frac{1}{n+1} \sum_{j=0}^{n} \phi \left(\min_{i:i \neq j}{}^{(k)} \rho(\boldsymbol{\xi_i}, \boldsymbol{\xi_j}) \right), \tag{1}$$

где $\boldsymbol{\xi_0}, \ldots, \boldsymbol{\xi_n}$ – заданные точки метрического пространства Ω, ρ – заданная метрика, ϕ - заданная монотонная функция, а $\min^{(k)}\{x_1, \ldots, x_N\} = x_k$, если $x_1 \leq x_2 \leq \cdots \leq x_k \leq \cdots \leq x_N$.

В настоящей работе предлагается алгоритм нахождения величин $r_s^{(k)}$, для всех значений $s \leqslant n$ в пространстве последовательностей Ω. Трудоемкость алгоритма равна $\mathcal{O}(nm)$, где m – длина строк. Здесь параметр k предполагается небольшим (константа, не зависящая от n).

Задача нахождения величин $r_s^{(k)}$ для всех значений $s \leqslant n$ представляет интерес в связи с тем, что дисперсия случайной величины $r_n^{(k)}$ быстро убывает с ростом n [1], если $\boldsymbol{\xi_0}, \ldots, \boldsymbol{\xi_n}$ – независимые одинаково распределенные случайные величины, а смещение оценки $r_n^{(k)}$ не известно. Поэтому вычисление величин $r_s^{(k)}$ для $s \leqslant n$ позволяет заметить существенное смещение. Применение предлагаемого алгоритма для оценивания энтропии ($\phi(x) = -\log x$) [2] показало отсутствие значимого смещения в рассмотренных примерах.

1. Постановка задачи

Пусть заданы $n+1$ последовательностей $\boldsymbol{\xi_0} = (x_{01}, \ldots, x_{0m})$, $\ldots$, $\boldsymbol{\xi_n} = (x_{n1}, \ldots, x_{nm})$, где $x_{ij} \in \mathcal{A} = \{1, 2, \ldots, a\}$.

Будем обозначать лексикографический порядок на пространстве $\Omega = \mathcal{A}^m$ через $\boldsymbol{x} \preccurlyeq \boldsymbol{y}$.

Пусть на Ω задана метрика ρ, про которую будем предполагать, что она согласована с порядком, т.е.

$$\rho(\boldsymbol{x}, \boldsymbol{y}) \leqslant \rho(\boldsymbol{x}, \boldsymbol{z}) \quad \forall \boldsymbol{x} \preccurlyeq \boldsymbol{y} \preccurlyeq \boldsymbol{z}. \tag{2}$$

Требуется найти величины $r_s^{(l)}$, заданные по формуле (1), для всех $k < s \leqslant n$, $1 \leqslant l \leqslant k$, где k – заданное число.

2. Описание алгоритма

1) Выполним сортировку строк $\boldsymbol{\xi_0}, \ldots, \boldsymbol{\xi_n}$ в возрастающем порядке. Пусть

$$\boldsymbol{\eta_0} \preccurlyeq \ldots \preccurlyeq \boldsymbol{\eta_n}$$

– отсортированные строки, а σ – перестановка такая, что

$$\boldsymbol{\eta_{\sigma_i}} = \boldsymbol{\xi_i}, \quad i = 0, 1, \ldots, n.$$

Организуем строки $\boldsymbol{\eta_0}$, ..., $\boldsymbol{\eta_n}$ в двусвязанный список.

2) Для $j = 0, 1, \dots, n$ и $l = 1, 2, \dots, k$ находим величины

$$g_j^l = \min_{i \neq j : j-l \leqslant i \leqslant j+l}{}^{(k)} \rho(\boldsymbol{\eta_i}, \boldsymbol{\eta_j}). \tag{3}$$

Для $l = 1, 2, \dots, k$ полагаем

$$r_n^{(l)} = \frac{1}{n+1} \sum_{j=0}^{n} \phi(g_j^l). \tag{4}$$

3) Для $s = n, n-1, \dots, k$ делаем следующее:

(a) удаляем строку $\boldsymbol{\eta_{\sigma_s}}$ из двусвязанного списка;

(b) для $l = 1, 2, \dots, k$ и $j = \sigma_s - l, \dots, \sigma_s + l$ находим величины g_j^l по формуле (3);

(c) для $l = 1, 2, \dots, k$ пересчитываем величины $r_{s-1}^{(l)}$, заменяя в формуле (4) величины g_j^l для $j = \sigma_s - l, \dots, \sigma_s + l$.

Отметим, что нумерация по j на шаге 3b идет по двусвязанному списку.

Отметим также, что поскольку величины g_j^l находятся последовательно для $l = 1, 2, \dots, k$, то для нахождения каждой величины g_j^l по формуле (3) достаточно находить минимум только из двух расстояний.

3. Обоснование алгоритма

Корректность алгоритма следует из того, что по свойству (2)

$$\min_{i : i \neq j}{}^{(k)} \rho(\boldsymbol{\eta_i}, \boldsymbol{\eta_j}) = \min_{i \neq j : j-k \leqslant i \leqslant j+k}{}^{(k)} \rho(\boldsymbol{\eta_i}, \boldsymbol{\eta_j}).$$

Поэтому величины $r_n^{(k)}$, найденные по формулам (4) и (1), совпадают.

4. Трудоемкость алгоритма

Трудоемкость шага 1 равна $\mathcal{O}(nm)$.

На остальных шагах наиболее трудоемкой операцией является нахождение расстояния между двумя строками. Будем считать, что ее трудоемкость равна $\mathcal{O}(m)$.

Трудоемкость шага 2 равна $\mathcal{O}(k^2nm)$.

Трудоемкость шага 3 равна $\mathcal{O}(k^2nm)$.

Итак, трудоемкость алгоритма равна $\mathcal{O}(k^2nm)$.

Подчеркнем, что при небольших k (не зависящих от n) трудоемкость алгоритма равна $\mathcal{O}(nm)$.

5. Свойство согласованности

Нетрудно видеть, что свойство согласованности (2) выполняется для метрики

$$\rho_0(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{\min\{k : x_k \neq y_k\}}, \tag{5}$$

где $\boldsymbol{x} = (x_1, x_2, \dots)$, и $\boldsymbol{y} = (y_1, y_2, \dots)$.

Для второй метрики

$$\rho_1(\boldsymbol{x}, \boldsymbol{y}) = \sum_{k=1}^{m} \theta^{-k} |x_k - y_k|, \tag{6}$$

где $\theta > 1$, свойство (2) выполняется только при $\theta \geqslant a = |\mathcal{A}|$.

Список литературы

1. Майоров, В.В. Статистическая оценка обобщенных размерностей /В.В.Майоров, Е.А.Тимофеев //Мат. заметки. – 2002. –Т.71. №5. – С. 697 – 712.

2. Kaltchenko, A., Entropy Estimators with Almost Sure Convergence and an $O(n^{-1})$ Variance /A.Kaltchenko, En-hui Yang, N.Timofeeva //Information Theory Workshop. 2007. ITW '07. IEEE 2-6 Sept. 2007. – P. 644 – 649.

Fast algorithm for finding mean minimum distances

Timofeeva N.E.

Let $\boldsymbol{\xi_0}, \ldots, \boldsymbol{\xi_n}$ be strings drawn from some finite alphabet. In this paper we describe an algorithm for finding mean minimum distances between strings $\boldsymbol{\xi_0}, \ldots, \boldsymbol{\xi_s}$ for all $s \leqslant n$. The complexity of the algorithm is $\mathcal{O}(nm)$, where m is the length of strings.