## ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА

УДК 519.237.5:519.24 doi: 10.18101/2304-5728-2017-4-21-32

## ТОЧНЫЕ АЛГОРИТМЫ РЕАЛИЗАЦИИ МЕТОДА НАИМЕНЬШИХ МОДУЛЕЙ НА ОСНОВЕ СПУСКА ПО УЗЛОВЫМ ПРЯМЫМ<sup>1</sup>

## © Тырсин Александр Николаевич

доктор технических наук, заведующий кафедрой прикладной математики Уральского энергетического института, Уральский федеральный университет им. первого Президента России Б. Н. Ельцина Россия, 620002, Екатеринбург, ул. Мира, 19 E-mail: at2001@yandex.ru

## © Азарян Алексан Артурович

аспирант Уральский энергетический институт, Уральский федеральный университет им. первого Президента России Б. Н. Ельцина Россия, 620002, Екатеринбург, ул. Мира, 19 E-mail: a.a.azaryan@gmail.com

Описаны алгоритмы точного решения задачи оценивания параметров линейных регрессионных моделей методом наименьших модулей. Они основаны на спуске по узловым прямым. Предложенные алгоритмы включают в себя обычный спуск, спуск с использованием разреженных матриц и с учетом направления спуска. Данные алгоритмы значительно выигрывают по сравнению с известным переборным алгоритмом и могут эффективно использоваться на практике. Получена оценка вычислительной сложности алгоритма спуска по узловым прямым. Приведена схема алгоритма. Методом статистических испытаний Монте – Карло был проведен сравнительный анализ предложенного алгоритма на основе спуска по узловым прямым и приближенного алгоритма Вейсфельда. Описан пример практической реализации предложенных алгоритмов.

*Ключевые слова*: метод наименьших модулей; линейная регрессионная модель; алгоритм; узловая точка; узловая прямая; гиперплоскость; вычислительная сложность.

### Введение

Одной из распространенных математических моделей является линейная регрессионная зависимость. Эта модель имеет вид

$$\mathbf{y} = \mathbf{X}\mathbf{a} + \boldsymbol{\varepsilon} \tag{1}$$

<sup>&</sup>lt;sup>1</sup> Работа выполнена при финансовой поддержке гранта РФФИ, проект № 17-01-00315а.

где 
$$\mathbf{X} = \{x_{ij}\}_{n \times m} = \begin{pmatrix} 1 & x_{12} & \dots & x_{1m} \\ 1 & x_{22} & \dots & x_{2m} \\ \dots & \dots & \dots & \dots \\ 1 & x_{n2} & \dots & x_{nm} \end{pmatrix} = \begin{pmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \dots \\ \mathbf{X}_n \end{pmatrix}$$
 – заданная матрица;  
 $\mathbf{a} = \begin{pmatrix} a_1 \\ a_2 \\ \dots \\ a_m \end{pmatrix} \in \mathbf{R}^m, \ \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{pmatrix} \in \mathbf{R}^n, \ \mathbf{\varepsilon} = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \dots \\ \varepsilon_n \end{pmatrix} \in \mathbf{R}^n$  – векторы коэффициентов

регрессии, измерений и случайных ошибок, соответственно.

Среди методов оценивания параметров модели (1) наибольшее распространение получил метод наименьших квадратов (МНК). Его использование требует выполнения ряда предпосылок, называемых условиями Гаусса-Маркова [1]. При их выполнении МНК-оценки параметров модели (1) являются состоятельными и несмещенными. Кроме того, если случайные ошибки имеют нормальный закон распределения, то МНК-оценки становятся эффективными. Однако использование МНК при нарушении условий Гаусса-Маркова может привести к значительным ошибкам при оценивании коэффициентов, а в случае присутствия в измерениях больших выбросов даже к несостоятельности оценок.

С целью обеспечения устойчивости оценок относительно отклонений случайных ошибок от гауссовой модели разработан ряд статистических методов. Данные методы основаны на более общих предположениях относительно случайных ошибок. Одним из таких методов является метод наименьших модулей (МНМ) [2]. МНМ для модели (1) имеет вид

$$Q(\mathbf{a}) = \sum_{i=1}^{n} \left| y_i - \left\langle \mathbf{a}, \mathbf{x}_i \right\rangle \right| \to \min_{\mathbf{a} \in \mathbf{R}^m}.$$
 (2)

Несмотря на то, что, как указано в [2], МНМ появился примерно на полстолетия раньше МНК, методы решения задачи (2) разработаны в недостаточной степени. Особенно это касается проблемы получения точного определения вектора **a** при значительном объеме многомерных экспериментальных данных. Данная статья посвящена описанию и анализу вычислительных затрат нового подхода реализации МНМ на основе спуска по узловым прямым.

### 1. Постановка задачи

Целевая функция  $Q(\mathbf{a})$  задачи (2) является выпуклой. Поскольку во всех точках, где у функции  $Q(\mathbf{a})$  существует производная, она линейна, то, следовательно, ее локальные минимумы могут быть только в особых точках. Известно несколько приближенных и точных методов решения задачи (2).

В настоящее время в целом удовлетворительно решена только проблема приближенного решения (2). Здесь можно отметить метод вариацион*А. Н. Тырсин, А. А. Азарян.* Точные алгоритмы реализации метода наименьших модулей на основе спуска по узловым прямым

но-взвешенных квадратических приближений (называемый также алгоритмом Вейсфельда) [3–5], подход, основанный на сведении данной проблемы к задаче линейного программирования и ее приближенное решение методом внутренней точки [6], численные методы спуска нулевого порядка [7]. Недостатком этих методов в первую очередь является неточное решение, при относительной простоте целевой функции. Кроме того, не ясен вопрос качества получаемых приближенных решений.

Кусочно-линейный вид целевой функции позволяет рассчитывать на наличие быстрых и точных методов решения задачи (2). К точным методам относится сведение (2) к эквивалентной задаче линейного программирования [4, 8] и ее решение при помощи широко известного симплексметода и переборный алгоритм, описанный в [7].

Использование симплекс-метода ограничено задачами малой размерности. Это вызвано накоплением погрешностей из-за ошибок округлений и требованием чрезмерно большого выделения памяти.

Второй точный метод [7] основан на переборе всех узловых точек, в которых не существует производная функции  $Q(\mathbf{a})$  ни по одному из возможных направлений пространства  $\mathbf{R}^m$ . Поскольку функция  $Q(\mathbf{a})$  является выпуклой, то она имеет единственный минимум. Если минимум одновременно достигается в нескольких особых точек  $\mathbf{a}^1, \mathbf{a}^2, ..., \mathbf{a}^l$ , то областью минимума будет являться выпуклое множество [9]

$$A = \{\mathbf{a} : \mathbf{a} = \sum_{k=1}^{l} \lambda_k^2 \mathbf{a}^k, \sum_{k=1}^{l} \lambda_k^2 = 1\}.$$

Введем гиперплоскости  $\Omega_i = \Omega(\mathbf{a}, \mathbf{x}_i, y_i)$  в виде уравнений

$$v_i - \langle \mathbf{a}, \mathbf{x}_i \rangle = 0, \ (i = 1, 2, \dots, n).$$
(3)

Зададим также узловые точки пересечения гиперплоскостей (3)

$$\mathbf{u}_{(k_1,\dots,k_m)} = \bigcap_{i=k_1}^{k_m} \Omega_i, \ 1 \le k_1 < k_2 < \dots < k_m \le n \ , \ k_s \in N \ .$$
(4)

Обозначим U – множество всех узловых точек (4). В [5] показано, что минимум целевой функции  $Q(\mathbf{a})$  принадлежит множеству U. Поэтому решение  $\mathbf{a}^*$  задачи (2) может быть получено путем перебора всех узловых точек и выбора в качестве решения той, которая обеспечивает минимум целевой функции  $Q(\mathbf{a})$ .

Переборный алгоритм требует решения  $C_n^m$  систем линейных уравнений порядка *m*. Вычислительные погрешности здесь незначительны и не накапливаются. Однако с ростом *n* и *m* наблюдается экспоненциальный рост вычислительных затрат. Фактически практическое применение переборного алгоритма ограничено объемом выборки *n* < 200 и числом коэффициентов регрессии *m* < 4. Однако отсутствие эффекта накопления вычислительных погрешностей делает возможным использование данного метода при условии ограничения числа перебираемых узловых точек.

Целью статьи является описание эффективного точного вычислитель-

ного метода решения задачи (2), осуществляющего неполный перебор особых точек и его сравнительный анализ с приближенными методами.

### 2. Алгоритмы реализации спуска по узловым прямым

**Обычный спуск.** Данный алгоритм точного решения задачи (2) основан на спуске к точному решению, двигаясь вдоль узловых прямых  $l_{(k_1,...,k_{m-1})}$ , каждая из которых является пересечением (m-1) различных гиперплоскостей  $\Omega_i$ :

$$l_{(k_1,...,k_{m-1})} : \bigcap_{i=k_1}^{k_{m-1}} \Omega_i \ , \ 1 \le k_1 < k_2 < \ldots < k_{m-1} \le n \ , \ k_s \in N$$

В качестве начального приближения берется узловая точка  $\mathbf{u}_{(k_1,...,k_m)}$ , являющаяся пересечением *m* произвольных различных гиперплоскостей  $\Omega_{k_1},...,\Omega_{k_m}$ . Исключив одну из гиперплоскостей, получим узловую прямую  $l_{(k_1,...,k_{m-1})}$ . Через любую узловую точку проходит *m* узловых прямых. Выберем из этих узловых прямых ту, вдоль которой целевая функция достигает наименьшего значения, которое всегда будет достигаться в одной из узловых точек. Найдя эту узловую точку, продолжим движение из нее по тому же принципу. В результате будет найдена узловая точка  $\mathbf{u}_{(k_1,...,k_m)}^* = \mathbf{a}^*$ , спуск из которой невозможен. Эта узловая точка будет являться точным решением задачи (2). В [10] доказано, что алгоритм спуска вдоль узловых прямых для нахождения решения задачи (2), сходится к точному решению за конечное число шагов.

Поясним этот алгоритм, используя рис. 1 (для случая m = 2).

В качестве начальной точки выберем точку Н. Через нее проходят прямые I и II. Сначала рассмотрим прямую I. Среди узловых точек, которые лежат на этой прямой, выбираем ту, в которой достигается минимальное значение целевой функции (таким образом, происходит спуск по прямой). Предположим, что такой точкой является точка С. Рассмотрим прямую II. Спускаясь по прямой II, находим на ней точку, в которой достигается минимальное значение. Пусть, например, эта точка L. Далее, сравнивая значения целевой функции в точках C и L, выбираем ту, в которой достигается минимальное значение. Пусть, например, эта точка С. Через нее помимо прямой I проходит прямая III. На этой прямой находим очередную точку, в которой достигается минимальное значение целевой функции. Далее, сравнивая значения целевой функции в точках С и D, выбираем ту, в которой достигается минимальное значение. Пусть такой точкой является D. Допустим, что на прямой IV, проходящей через D, точкой минимума является сама точка D, тогда в качестве решения выбираем точку D, в которой достигается минимум целевой функции (если бы нашлась другая точка минимума на этой прямой, то из нее продолжили бы дальнейший спуск).

*А. Н. Тырсин, А. А. Азарян.* Точные алгоритмы реализации метода наименьших модулей на основе спуска по узловым прямым



Рис 1. Спуск по узловым прямым

Отметим, что при вычислительные затраты данного алгоритма можно значительно сократить, если использовать информацию о предыдущей найденной узловой точке.

Спуск с использованием разреженных матриц. Двигаясь вдоль прямой  $l_{(k_1,...,k_{m-1})}$ , для нахождения узловых точек, принадлежащих этой прямой, нужно для каждой точки решать систему линейных алгебраических уравнений (СЛАУ) порядка *m* 

$$\begin{cases} a_{1} + a_{2}x_{k_{1},2} + a_{3}x_{k_{1},3} + \dots + a_{m}x_{k_{1},m} = y_{k_{1}}, \\ a_{1} + a_{2}x_{k_{2},2} + a_{3}x_{k_{2},3} + \dots + a_{m}x_{k_{2},m} = y_{k_{2}}, \\ \dots \\ a_{1} + a_{2}x_{k_{m-1},2} + a_{3}x_{k_{m-1},3} + \dots + a_{m}x_{k_{m-1},m} = y_{k_{m-1}}, \\ a_{1} + a_{2}x_{i,2} + a_{3}x_{i,3} + \dots + a_{m}x_{i,m} = y_{i}, \\ 1 \le k_{1} < k_{2} < \dots < k_{m-1} \le n, \ i \in \{1, 2, \dots, n\}, \ i \notin \{k_{1}, k_{2}, \dots, k_{m-1}\}. \end{cases}$$

$$(5)$$

Очевидно, что СЛАУ двух различных узловых точек, принадлежащих этой прямой одной прямой отличаются лишь одним (последним) уравнением. Следовательно, вычислительная эффективность алгоритма спуска существенно повысится, если для нахождения узловых точек, которые лежат на прямой  $l_{(k_1,...,k_{m-1})}$ , первые (m-1) строк расширенной матрицы, соответствующей СЛАУ (5) предварительно преобразуем с помощью элементарных преобразований к ступенчатому виду.

Расширенная матрица СЛУ прямой  $l_{(k_1,...,k_{m-1})}$  имеет вид

где

$$\mathbf{A}_{(k_1,\ldots,k_{m-1})} = \begin{pmatrix} 1 & x_{k_1,2} & x_{k_1,3} & \cdots & x_{k_1,m-1} & x_{k_1,m} & y_{k_1} \\ 1 & x_{k_2,2} & x_{k_2,3} & \cdots & x_{k_2,m-1} & x_{k_2,m} & y_{k_2} \\ 1 & x_{k_3,2} & x_{k_3,3} & \cdots & x_{k_3,m-1} & x_{k_3,m} & y_{k_3} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & x_{k_{m-1},2} & x_{k_{m-1},3} & \cdots & x_{k_{m-1},m-1} & x_{k_{m-1},m} & y_{k_{m-1}} \end{pmatrix}.$$

Применив алгоритм прямого хода метода Гаусса, преобразуем матрицу  $\mathbf{A}_{(k_1,...,k_{m-1})}$  к ступенчатому виду

$$\mathbf{A}'_{(k_1,\dots,k_{m-1})} = \begin{pmatrix} 1 & x_{k_1,2} & x_{k_1,3} & \dots & x_{k_1,m-1} & x_{k_1,m} & y_{k_1} \\ 0 & 1 & x'_{k_2,3} & \dots & x'_{k_2,m-1} & x'_{k_2,m} & y'_{k_2} \\ 0 & 0 & 1 & \dots & x'_{k_3,m-1} & x'_{k_3,m} & y'_{k_3} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 & x'_{k_{m-1},m} & y'_{k_{m-1}} \end{pmatrix}$$

Используя ступенчатую матрицу  $\mathbf{A}'_{(k_1,...,k_{m-1})}$  можно значительно сократить вычислительные затраты на нахождение всех узловых точек, лежащих на прямой  $l_{(k_1,...,k_{m-1})}$ . Действительно для каждой искомой узловой точки имеем расширенную матрицу

$$\mathbf{A}_{(k_{1},\dots,k_{m-1},i)} = \begin{pmatrix} 1 & x_{k_{1},2} & x_{k_{1},3} & \dots & x_{k_{1},m-1} & x_{k_{1},m} & y_{k_{1}} \\ 0 & 1 & x'_{k_{2},3} & \dots & x'_{k_{2},m-1} & x'_{k_{2},m} & y'_{k_{2}} \\ 0 & 0 & 1 & \dots & x'_{k_{3},m-1} & x'_{k_{3},m} & y'_{k_{3}} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 & x'_{k_{m-1},m} & y'_{k_{m-1}} \\ 1 & x_{i,2} & x_{i,3} & \dots & x_{i,m-1} & x_{i,m} & y_{i} \end{pmatrix}.$$
(6)

Варьируя номер i в (6), найдем все узловые точки, лежащие на прямой  $l_{(k_1,...,k_{m-1})}$ .

Вычислительная эффективность алгоритма спуска можно повысить, если учитывать направление спуска.

Спуск с использованием разреженных матриц и с учетом направления спуска. Используя ступенчатую матрицу  $\mathbf{A}'_{(k_1,...,k_{m-1})}$ , и решив СЛАУ, соответствующую расширенной матрице (6), находим значение *m*-го коэффициента  $a_m^{(k_1,k_2,...,k_{m-1},i)}$ ,  $1 \le k_1 < k_2 < ... < k_{m-1} \le n$ ,  $i \in \{1, 2, ..., n\}$ ,  $i \notin \{k_1, k_2, ..., k_{m-1}\}$ , для каждой узловой точки. После чего по возрастанию  $a_m^{(k_1,k_2,...,k_{m-1},i)}$  упорядочиваем все узловые точки, которые лежат на прямой  $l_{(k_1,...,k_{m-1})}$ , и выполняем описанный выше алгоритм спуска, но с учетом направления. Если при непосредственном переходе от одной узловой точки к другой значение целевой функции увеличивается, то в этом направлении значение целевой функции будет увеличиваться во всех узловых

*А. Н. Тырсин, А. А. Азарян.* Точные алгоритмы реализации метода наименьших модулей на основе спуска по узловым прямым

точках (вытекает из выпуклости целевой функции). Назовем такое направление «плохим». Для осуществления спуска, до вычисления значения целевой функции в очередной узловой точке рассматриваем направление спуска. Если оно «плохое», то переходим к следующей точке, не вычисляя в данной узловой точке, ни значение целевой функции, ни значения коэффициентов  $a_j^{(k_1,k_2,...k_{m-1},i)}$ , j = 1, 2, ..., m-1.

## 3. Анализ вычислительных затрат спуска по узловым прямым

Оценим вычислительные затраты спуска с использованием разреженных матриц и с учетом направления спуска. Воспользуемся методом статистических испытаний Монте–Карло [11]. В табл. 1 для 1000 испытаний приведены средние значения общего количества рассмотренных узловых точек L и узловых прямых P (переходов с одной прямой на другую) в алгоритме спуска с использованием разреженных матриц и с учетом направления спуска для некоторых значений n и m. Случайные ошибки  $\varepsilon$ имели стандартное нормальное распределение.

Таблица 1. Среднее число рассмотренных узловых точек и прямых в алгоритме спуске

п	Кол	ичеств	о узлов	ых точ	ек L	Количество узловых прямых Р					
	<i>m</i> =3	m=4	<i>m</i> =5	<i>m</i> =6	m=7	<i>m</i> =3	m=4	m=5	<i>m</i> =6	m=7	
30	56	91	127	168	211	5,2	6,6	7,7	8,8	9,6	
50	85	136	197	262	334	6,1	8,0	9,8	11,3	12,7	
100	148	235	348	459	604	7,3	9,9	12,3	14,3	16,8	
150	214	337	490	655	847	7,9	10,8	13,7	16,3	19,0	
300	397	618	895	1205	1529	9,1	12,7	16,1	19,6	22,9	
500	644	1002	1441	1905	2477	9,8	13,9	18,0	21,8	26,0	
700	881	1417	2008	2665	3390	10,4	15,0	19,1	23,5	28,1	
1000	1257	1946	2771	3676	4701	11,0	15,7	20,3	25,0	30,1	
1500	1855	2928	4043	5413	6973	11,8	16,8	21,7	26,9	32,4	
2000	2453	3867	5363	7215	9204	12,2	17,6	22,8	28,2	33,8	

Оценим средние количества рассмотренных узловых точек L и узловых прямых P. Анализ полученных результатов для различных n и m показал, что  $L \sim O(mn)$ ,  $P \sim O(m \ln n)$ .

**Теорема 1.** Алгоритм спуска по узловым прямым имеет вычислительную сложность  $W = O(m^2n^2 + m^4n\ln n + m^2n\ln^2 n)$ .

Доказательство. Оценим вычислительную сложность всех основных функций, заложенных в базисе алгоритма спуска по узловым прямым. Его схема приведена на рис. 2.



Рис. 2. Схема спуска по узловым прямым

Функция trapezoidalMatrix расширенную матрицу размера  $(m-1) \times (m+1)$  приводит к трапециевидному виду за (m-1) итерацию. В ходе *i*-й итерации выполняется (m+2-i)(2m-1-2i) операций умножения и вычитания. Поэтому общее количество выполненных операций

$$I = \sum_{i=1}^{m-1} (m+2-i)(2m-1-2i) = 2\sum_{i=1}^{m-1} (m+2-i)^2 - 5\sum_{i=1}^{m-1} (m+2-i) =$$
$$= [j = m+2-i] = 2\sum_{j=3}^{m+1} j^2 - 5\sum_{j=3}^{m+1} j.$$

2017/4

*А. Н. Тырсин, А. А. Азарян.* Точные алгоритмы реализации метода наименьших модулей на основе спуска по узловым прямым

Известно, что 
$$\sum_{j=1}^{m} j^2 = \frac{m(m+1)(2m+1)}{6}$$
 и  $\sum_{j=1}^{m} j = \frac{m(m+1)}{2}$ . Значит  
 $I = 2 \cdot \left(\frac{(m+1)(m+2)(2m+3)}{6} - 5\right) - 5 \cdot \left(\frac{(m+1)(m+2)}{2} - 3\right) = O(m^3).$ 

Функция getMCoeff к трапециевидной матрице размера  $(m-1) \times (m+1)$  добавляет строку и вычисляет *m*-й коэффициент. Здесь выполняется (m-1) итераций. В ходе *i*-й итерации выполняется 2(m+1-i) операций умножения и вычитания (кроме первой итерации, во время которой выполняется не 2m, а *m* операций умножения и вычитания). Поэтому, общее количество выполненных операций будет равно

$$2\sum_{i=1}^{m-1}(m+1-i)+m-1=m^2-1=O(m^2).$$

Функция оbj функция вычисляет и возвращает значение целевой функции  $Q(\mathbf{a})$ . Очевидно, что функция obj имеет O(mn) вычислительную сложность.

Функция sort реализует сортировку полученного массива. Известно, что сортировка Хоара в среднем имеет для входного массива из n элементов  $O(n \ln n)$  вычислительную сложность [12].

Функция descent, имея матрицу вида (6) и *m*-й коэффициент, вычисляет остальные коэффициенты и вызывает функцию obj. Данные действия выполняются циклически (количество итераций примерно равно

 $L/(m \cdot P) = O(mn)/(m \cdot )(m \ln n)) = O(n/(m \ln n)).$ 

Для вычисления остальных коэффициентов выполняется (m-1) итераций. В ходе *i*-й итерации выполняется 2i операций умножения и вычитания. Следовательно, общее количество выполненных операций

$$2\sum_{i=1}^{m-1} i = m^2 - m = O(m^2).$$

Т.к. функция obj имеет вычислительную сложность O(mn), то функ-

ция descent имеет  $O(n \cdot (mn + m^2)/m \ln n)$  вычислительную сложность.

Теперь из схемы работы алгоритма получим, что спуск по узловым прямым имеет вычислительную сложность

$$O\left\{m\ln n \cdot \left[m \cdot \left(m^{3} + (n-m) \cdot m^{2} + (n-m) \cdot \ln(n-m) + \frac{n \cdot (m^{2} + mn)}{m\ln n}\right)\right]\right\} = O(m^{2}n^{2} + m^{4}n\ln n + m^{2}n\ln^{2}n + m^{3}n) = O(m^{2}n^{2} + m^{4}n\ln n + m^{2}n\ln^{2}n).$$
  
Teopema dokasaha.

Отметим, что для  $n > \max(\ln^2 n; m^2 \ln n)$  вычислительная сложность спуска по узловым прямым  $W = O(m^2 n^2)$ .

# 4. Сравнительный анализ алгоритмов спуска по узловым прямым с переборным алгоритмом и алгоритмом Вейсфельда

Также воспользуемся методом Монте–Карло. В каждом случае используем по 1000 испытаний.

Переборный алгоритм имеет очень высокую вычислительную трудоемкость. Для иллюстрации эффективности предложенного подхода в табл. 2 указаны отношения средней продолжительности работы различных алгоритмов спуска по узловым точкам к средней продолжительности работы переборного алгоритма для количества параметров m = 5. Результаты свидетельствуют о существенном выигрыше по быстродействию алгоритмов спуска.

### Таблица 2

Отношение средней продолжительности работы различных алгоритмо
спуска по узловым точкам к средней продолжительности работы пер
борного алгоритм

Объем выборки <i>n</i>	Алгоритмы спуска по узловым точкам						
		Спуск с исполь-	Спуск с использованием				
	спуск	зованием разре-	разреженных матриц и с				
		женных матриц	учетом направления спуска				
30	111,6	138,9	284,0				
100	10352,0	14725,8	42073,5				
300	602404,4	911002,5	3077299,4				

Сравним метод спуска по узловым прямым с приближенным алгоритмом Вейсфельда (вычислялся с точностью до  $10^{-5}$ ). Результаты сравнения приведены в табл. 3. Здесь обозначено:  $s_v(n,m)$  — среднее квадратическое отклонение вектора  $\hat{\mathbf{a}}_v$  выборочных оценок параметров модели множественной линейной регрессии относительно вектора  $\mathbf{a}$  теоретических значений параметров для алгоритма Вейсфельда,  $s_u(n,m)$  — среднее квадратическое отклонение вектора  $\hat{\mathbf{a}}_u$  выборочных оценок параметров модели множественной линейной регрессии относительно вектора  $\mathbf{a}$ теоретических значений параметров для алгоритма спуска по узловым прямым,  $t_v(n,m)$  — среднее время вычислений для алгоритма спуска по узловым прямым.

Таблица 3

Результаты сравнения алгоритмов Вейсфельда и спуска по узловым прямым по точности и быстродействию

п	s(n,	m) = s	$v_v(n,m)$	$)/s_u(n$	(m,m)	$t(n,m) = t_v(n,m) / t_u(n,m)$				
	<i>m</i> =3	<i>m</i> =4	<i>m</i> =5	<i>m</i> =6	<b>m</b> =7	<i>m</i> =3	<i>m</i> =4	<i>m</i> =5	<i>m</i> =6	<i>m</i> =7
32	1,03	1,02	1,02	1,00	1,00	5,85	4,43	4,36	3,38	2,16
64	1,02	0,98	1,01	1,00	0,98	6,66	6,49	4,56	3,44	2,23
128	1,01	1,02	1,05	1,02	0,99	8,22	6,72	4,65	3,60	2,32

*А. Н. Тырсин, А. А. Азарян.* Точные алгоритмы реализации метода наименьших модулей на основе спуска по узловым прямым

256	1,00	0,99	0,99	0,98	1,00	13,46	10,58	7,58	4,49	2,91
512	1,03	1,10	1,04	1,02	1,04	25,84	12,79	10,32	7,31	3,82
1024	1,02	1,04	1,04	1,00	1,08	47,66	31,06	20,11	10,83	7,61

Согласно приведенным в табл. 3 результатам при примерно одинаковой точности оценок алгоритм спуска по узловым прямым превосходит по быстродействию приближенный алгоритм Вейсфельда. Это значит, что предлагаемый точный алгоритм работает быстрее (t(n,m)>1) приближенного алгоритма Вейсфельда. Примерная близость значений отношения s(n,m) к единице означает, что алгоритм Вейсфельда стабильно сходится к точному решению с требуемой точностью.

### Заключение

Предложены точные алгоритмы реализации метода наименьших модулей при оценивании параметров линейных регрессионных моделей, основанные на спуске по узловым прямым.

Данные алгоритмы позволяют значительно снизить вычислительные затраты при точной реализации метода наименьших модулей. Выигрыш достигается за счет того, что вместо перебора узловых точек осуществляется спуск по ним. Вычислительная сложность спуска по узловым прямым позволяет на практике реализовать точное оценивание методом наименьших модулей для любых размеров экспериментальных данных. При этом, предложенный точный метод превзошел по быстродействию известный приближенный алгоритм Вейсфельда.

### Литература

1. Айвазян С. А., Енюков И. С., Мешалкин Л. Д. Прикладная статистика: Исследование зависимостей. М.: Финансы и статистика, 1985. 488 с.

2. Bloomfield P., Steiger W. L. Least absolute seviations: theory, applications, and algorithms. Boston-Basel-Stuttgart: Birkhauser, 1983. 349 p.

3. Weiszfeld E. On the point for which the sum of the distances to *n* given points is minimum // Annals of Operations Research. 2008. V. 167,  $N_{0}$  1. P. 7–41. Translated from the French original [Tohoku Mathematics Journal. 1937. V. 43. P. 355–386] and annotated by Frank Plastria.

4. Мудров В. И., Кушко В. Л. Методы обработки измерений: квазиправдоподобные оценки. М.: Радио и связь, 1983. 304 с.

5. Акимов П. А., Матасов А. И. Уровни неоптимальности алгоритма Вейсфельда в методе наименьших модулей // Автоматика и телемеханика. 2010. № 2. С. 4–16.

6. Boyd S., Vandenberghe L. Convex optimization. Cambridge: Cambridge University Press, 2004. 730 p.

7. Тырсин А. Н., Максимов К. Е. Оценивание линейных регрессионных уравнений с помощью метода наименьших модулей // Заводская лаборатория. Диагностика материалов. 2012. Т. 78, № 7. С. 65–71.

8. Зуховицкий С. И., Авдеева Л. И. Линейное и выпуклое программи-

рование. 2-е изд., перераб. и доп. М.: Наука. Физматлит, 1967. 460 с.

9. Рокафеллар Р. Выпуклый анализ: пер. с англ. М.: Мир, 1973. 470 с.

10. Азарян А. А., Тырсин А. Н. Эффективные алгоритмы оценивания линейных регрессионных моделей на основе метода наименьших модулей // Анализ, моделирование, управление, развитие социально-экономических систем: сборник научных трудов XI Международной школы-симпозиума, Симферополь-Судак, 14–27 сентября 2017. Симферополь: ИП Корниенко А. А., 2017. С. 11–16.

11. Михайлов Г. А., Войтишек А. В. Численное статистическое моделирование. Методы Монте-Карло. М.: Академия, 2006. 368 с.

12.Кормен Т. Х. Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы: построение и анализ: пер. с англ. 3-е изд. М.: Вильямс, 2013. 1328 с.

## EXACT ALGORITHMS FOR IMPLEMENTATION OF THE LEAST ABSOLUTE DEVIATIONS METHOD BASED ON THE DESCENT THROUGH THE NODAL STRAIGHT LINES

### Aleksandr N. Tyrsin

Dr. Sci. (Engineering), Head of Department "Applied Mathematics", Ural Federal University named after the First President of Russia B. N. Yeltsin 19 Mira St., Yekaterinburg 620002, Russia

## Aleksan A. Azaryan

Research Assistant,

Ural Federal University named after the First President of Russia B. N. Yeltsin 19 Mira St., Yekaterinburg 620002, Russia

Algorithms for the exact solution of the problem of estimating the parameters of linear regression models by the least absolute deviations method are described. They are based on the descent through the nodal straight lines. The proposed algorithms include common descent, descent with the use of sparse matrices and descend with the use of sparse matrices and taking into consideration its directions. These algorithms significantly outperform the best-known brute-force search and can be effectively used in practice. The computational complexity of the descent algorithm for nodal straight lines is assessed. The scheme of the algorithm is provided. We have carried out a comparative analysis of the proposed algorithm based on descent through the nodal straight lines and Waisfeld approximate algorithm by Monte Carlo method of statistical testing. The examples of practical implementation of the proposed algorithms are described.

*Keywords*: the least absolute deviations method; linear regression model; algorithm; nodal point; nodal straight line; hyperplane; computational complexity.