DOI: https://doi.org/10.26896/1028-6861-2024-90-5-79-87

СПУСК ПО УЗЛОВЫМ ПРЯМЫМ И СИМПЛЕКС-АЛГОРИТМ — ДВА ВАРИАНТА РЕГРЕССИОННОГО АНАЛИЗА НА ОСНОВЕ МЕТОДА НАИМЕНЬШИХ МОДУЛЕЙ

© **Олег Александрович Голованов[1,2], Александр Николаевич Тырсин[1,3]***

[1] Уральский федеральный университет имени первого Президента России Б. Н. Ельцина, Россия, 620002, Екатеринбург, ул. Мира, д. 19.
[2] Институт экономики Уральского отделения РАН, Россия, 620014, Екатеринбург, ул. Московская, д. 29
[3] Научно-инженерный центр «Надежность и ресурс больших систем и машин» Уральского отделения РАН, Россия, 620049, Екатеринбург, ул. Студенческая, д. 54а; *e-mail: at2001@yandex.ru

Статья поступила 3 июля 2023 г. Поступила после доработки 11 августа 2023 г. Принята к публикации 30 августа 2023 г.

Проведен сравнительный анализ вычислительной сложности точных алгоритмов оценивания линейных регрессионных уравнений методом наименьших модулей. Цель работы — сравнение вычислительной эффективности точных алгоритмов спуска по узловым прямым и алгоритмов, основанных на решении задачи линейного программирования. Для этого рассмотрены алгоритм градиентного спуска по узловым прямым и алгоритмы решения эквивалентной прямой и двойственной задач линейного программирования с использованием симплекс-метода. Выполнена оценка вычислительной сложности алгоритмов реализации метода наименьших модулей с помощью решения прямой и двойственной задач линейного программирования. Также с помощью метода статистических испытаний Монте-Карло проведено сравнение среднего времени определения коэффициентов регрессии с помощью решения прямой и двойственной задач линейного программирования со средним временем градиентного спуска по узловым прямым. Установлено, что оба варианта значительно уступают градиентному спуску по узловым прямым как в плане вычислительной сложности алгоритмов, так и по времени вычисления. При этом выигрыш алгоритма спуска по узловым прямым растет с увеличением объема выборки, достигая сотни и более раз.

Ключевые слова: метод наименьших модулей; линейная регрессия; симплекс-алгоритм; узловая прямая; вычислительная эффективность.

DESCENT ALONG NODAL STRAIGHT LINES AND SIMPLEX ALGORITHM: TWO VARIANTS OF REGRESSION ANALYSIS BASED ON THE LEAST ABSOLUTE DEVIATION METHOD

© **Oleg A. Golovanov,[1,2] Alexander N. Tyrsin[1,3]***

[1] The First President of Russia B. N. Yeltsin Ural Federal University, 19, ul. Mira, Yekaterinburg, 620002, Russia.
[2] Institute of Economics, Ural Branch of RAS, 29, ul. Moskovskaya, Yekaterinburg, 620014, Russia.
[3] Science and Engineering Center “Reliability and Resource of Large Systems and Machines”, Ural Branch of RAS, 54a, ul. Studencheskaya, Yekaterinburg, 620049, Russia; *e-mail: at2001@yandex.ru

Received July 3, 2023. Revised August 11, 2023. Accepted August 30, 2023.

A comparative analysis of the computational complexity of exact algorithms for estimating linear regression equations was conducted using the least absolute deviation method. The goal of the study is to compare the computational efficiency of exact algorithms for descent along nodal lines and algorithms based on solving linear programming problems. For this purpose, the algorithm of gradient descent along nodal lines and algorithms for solving the equivalent primal and dual linear programming problems using the simplex method were considered. The computational complexity of algorithms for implementing the method of least modules in solving direct and dual linear programming problems was estimated. A comparison between the average time for determining the regression coefficients using the primal and dual linear programming problems and the average time for gradient descent along nodal lines was conducted using the Monte Carlo method of statistical experiments. It is shown that both options are significantly inferior behind gradient descent along nodal lines, both in terms of the computational complexity of the algorithms and in terms of computation time, and this advantage increases with the sample size, reaching hundred times or more.

Keywords: least absolute deviations method; linear regression; simplex algorithm; nodal straight line; computational efficiency.

Введение

Метод наименьших модулей (МНМ) представляет собой одну из наиболее распространенных альтернатив методу наименьших квадратов (МНК) в регрессионном анализе [1 – 3]. Его свойства приведены во многих работах [4 – 7]. Метод позволяет получать устойчивые оценки коэффициентов, когда плотность вероятности случайных ошибок имеет более вытянутые хвосты по сравнению с нормальным распределением [8].

Применимость МНМ, как и МНК, качественно обоснована в регрессионном анализе. Известно, что МНМ обеспечивает максимум функции правдоподобия, т.е. он наиболее эффективен, если ошибки измерений распределены по закону Лапласа [9]. Пусть случайная ошибка измерения, как и для оптимального применения МНК, распределена по нормальному закону. В условиях стохастической неоднородности данных естественна ситуация, когда среднеквадратическое отклонение ошибок является случайной величиной с математическим ожиданием, соответствующим некоторым «средним» окружающим условиям. В этом случае, как показано в [1, с. 26 – 32], распределение ошибок станет лапласовым.

Задача оценки множественной линейной регрессионной зависимости с помощью МНМ выглядит следующим образом [1]:

$$Q(\mathbf{a}) = \sum_{i=1}^{n} \left| y_i - \sum_{j=1}^{m} a_j x_{ij} \right| \to \min_{a \in \mathbb{R}^m}, \qquad (1)$$

где

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{pmatrix}, \mathbf{X} = \{x_{ij}\}_{n \times m} = \begin{pmatrix} 1 & x_{12} & \dots & x_{1m} \\ 1 & x_{22} & \dots & x_{2m} \\ \dots & \dots & \dots & \dots \\ 1 & x_{n2} & \dots & x_{nm} \end{pmatrix} =$$

$$= \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \dots \\ \mathbf{x}_n \end{pmatrix}, \mathbf{a} = \begin{pmatrix} a_1 \\ a_2 \\ \dots \\ a_m \end{pmatrix}.$$

Здесь $\mathbf{y}$ — вектор значений зависимой переменной Y; $\mathbf{X}$ — матрица значений независимых переменных X_2, ..., X_m; $\mathbf{a}$ — вектор оценок неизвестных коэффициентов α_j уравнения регрессии $Y = \alpha_1 + \alpha_2 X_2 + \dots + \alpha_m X_m + \varepsilon$, ε — ненаблюдаемые случайные ошибки.

Целевая функция $Q(\mathbf{a})$ является непрерывной, выпуклой и ограниченной снизу, поэтому всегда имеет единственный глобальный и одновременно локальный минимум, который и является решением задачи (1) [10]. Известен ряд как точных, так и приближенных алгоритмов решения задачи (1). Обзоры этих алгоритмов приведены, например, в [5, 11, 12].

Точным назовем тот алгоритм, который позволяет за конечное число шагов определить глобальный минимум целевой функции $Q(\mathbf{a})$. Поскольку вычислительные операции осуществляются с погрешностями, а техника безошибочных вычислений очень трудоемка, то в качестве точного решения примем то, которое вычисляется с минимальными вычислительными погрешностями. На сегодняшний день к точным решениям можно отнести алгоритм полного перебора узловых точек, алгоритмы спуска по узловым прямым [12, 13] и алгоритмы, основанные на решении задачи линейного программирования (ЛП) [14 – 17].

Наличие различных приближенных алгоритмов объясняется тем, что точные алгоритмы реализации МНМ значительно проигрывали МНК в быстродействии. Однако в последние годы наблюдается прогресс в данном вопросе.

Отметим, что оценка вычислительной эффективности решения задачи (1) на основе симплекс-метода пока не проводилась. В [12] приведены лишь известные в литературе верхняя и нижняя оценки трудоемкости симплекс-метода в общем случае. Поскольку (1) приводит к частному виду задачи ЛП, то представляется оправданным провести уточнение вычислительной эффективности реализации МНМ с помощью симплекс-метода. Алгоритм полного перебора узловых точек очень трудоемок и малопригоден для решения практических задач. Поэтому цель данной работы — проведение сравнительного анализа вычислительной эффективности точных алгоритмов спуска по узловым прямым и алгоритмов, основанных на решении задачи ЛП.

Методы исследования

Из вариантов спуска по узловым прямым наиболее быстрым является алгоритм градиентного спуска [13, 18]. Здесь вместо вычисления значений целевой функции находят ее производную по направлению в окрестности узловых точек. Различные модификации алгоритмов, основанных на решении задачи ЛП (см., например, [19, 20]), имеют вычислительную эффективность, сопоставимую с симплекс-методом. Поэтому проведем сравнение вычислительной эффективности и точности решения задачи (1) при помощи градиентного спуска и алгоритмами решения эквивалентной прямой и двойственной задач ЛП с использованием симплекс-метода.

В [13] приведено описание алгоритма градиентного спуска, включающее оценку его вычислительной сложности. Сформируем эквивалент-

ную (1) задачу ЛП, для этого представим каждую невязку в виде

$$0 \le \left| y_i - \sum_{j=1}^{m} a_j x_{ij} \right| \le z_i,\ i = 1, 2, ..., n.$$

Отсюда получим систему

$$\begin{cases} z_i + \sum\limits_{j=1}^{m} a_j x_{ij} \ge y_i, i = 1, 2, \ldots, n, \\ z_i - \sum\limits_{j=1}^{m} a_j x_{ij} \ge -y_i, i = 1, 2, \ldots, n, \end{cases}$$

из которой, обозначив $a_j = a_j^{(1)} - a_j^{(2)}$, сформируем прямую задачу ЛП [17, 21] как

$$\begin{cases} \sum\limits_{i=1}^{n} z_i \to \min\limits_{a_j^{(k)}, z_i \in \mathbb{R}}, \\ z_i + \sum\limits_{j=1}^{m} (a_j^{(1)} - a_j^{(2)}) x_{ij} \ge y_i, i = 1, 2, \ldots, n, \\ z_i - \sum\limits_{j=1}^{m} (a_j^{(1)} - a_j^{(2)}) x_{ij} \ge -y_i, i = 1, 2, \ldots, n, \\ a_j^{(k)}, z_i \ge 0, k = 1, 2, \end{cases}$$

или в матричном виде

$$\begin{cases} \mathbf{b}^{\mathrm{T}} \widetilde{\mathbf{y}} \to \min, \\ \mathbf{A} \widetilde{\mathbf{y}} \ge C, \\ \widetilde{\mathbf{y}} \ge 0, \end{cases} \qquad (2)$$

где $\mathbf{b}^{\mathrm{T}} = (\overbrace{1, 1, \ldots, 1}^{n}, \overbrace{0, \ldots, 0}^{2m})$ — вектор размерностью $1 \times (n + 2m)$; $\widetilde{\mathbf{y}}$ — вектор значений целевой функции размерностью $(n + 2m) \times 1$; $\mathbf{C}$ — вектор правой части ограничений размерностью $2n \times 1$; $\mathbf{A}$ — матрица коэффициентов небазисных переменных размерностью $2n \times (n + 2m)$. Симплекс-таблица прямой задачи ЛП приведена в табл. 1.

Очевидно, что в связи с наличием отрицательных значений в правой части ограничений симплекс-таблица прямой задачи ЛП (2) не будет являться допустимой. Чтобы это исправить, можно использовать стандартные преобразования, основанные на изменении базиса на элементы, соответствующие строкам с отрицательными ограничениями и столбцам с отрицательными элементами. Однако подобные преобразования могут привести к значительному росту вычислительной сложности, а следовательно, и времени вычисления или вовсе к зацикливанию.

Учитывая вид задачи (2), можно строго ограничить число итераций, необходимых для приведения табл. 1 к допустимому виду. Для этого в качестве разрешающих элементов выбираются первая строка сверху с отрицательным ограничением и первый столбец слева с отрицательным элементом. В силу (2) отрицательный элемент в выбранном столбце будет равен – 1, что ограничит число итераций для получения допустимого опорного плана числом n.

Аналогичным образом можно сформировать двойственную задачу ЛП

$$\begin{cases} \mathbf{C}^{\mathrm{T}} \widetilde{\mathbf{x}} \to \max, \\ \mathbf{A}^{\mathrm{T}} \widetilde{\mathbf{x}} \le \mathbf{b}, \\ \widetilde{\mathbf{x}} \ge 0, \end{cases} \qquad (3)$$

где $\widetilde{\mathbf{x}}$ — вектор размерностью $2n \times 1$.

При проведении вычислительных экспериментов в силу большого объема анализируемых выборок, а как следствие, и симплекс-таблиц начали происходить зацикливания поиска решений. В таком случае при наличии нескольких совпадающих оптимальных путей решения один из путей может приводить к самому себе. Чтобы этого избежать, необходимо провести оптимиза-

Таблица 1. Симплекс-таблица прямой задачи ЛП без единичной матрицы базисных переменных
Table 1. Simplex-table of a direct linear programming problem without a unit matrix of basic variables

Базис	b_1	b_2	...	b_n	b_{n+1}	b_{n+2}	...	b_{n+2m-1}	b_{n+2m}	**C**
b_{n+2m+1}	z_1	0	...	0	$a_1^{(1)}x_{11}$	$-a_1^{(2)}x_{11}$	...	$a_m^{(1)}x_{1m}$	$-a_m^{(2)}x_{1m}$	y_1
b_{n+2m+2}	0	z_2	...	0	$a_1^{(1)}x_{21}$	$-a_1^{(2)}x_{21}$	...	$a_m^{(1)}x_{2m}$	$-a_m^{(2)}x_{2m}$	y_2
...	...	...	...	...	...	...	...	...	...	...
$b_{2n+2m-1}$	0	0	...	z_n	$a_1^{(1)}x_{n1}$	$-a_1^{(2)}x_{n1}$	...	$a_m^{(1)}x_{nm}$	$-a_m^{(2)}x_{nm}$	y_n
b_{2n+2m}	z_1	0	...	0	$-a_1^{(1)}x_{11}$	$a_1^{(2)}x_{11}$	...	$-a_m^{(1)}x_{1m}$	$a_m^{(2)}x_{1m}$	$-y_1$
$b_{2n+2m+1}$	0	z_2	...	0	$-a_1^{(1)}x_{21}$	$a_1^{(2)}x_{21}$	...	$-a_m^{(1)}x_{2m}$	$a_m^{(2)}x_{2m}$	$-y_2$
...	...	...	...	...	...	...	...	...	...	...
b_{3n+2m}	0	0	...	z_n	$-a_1^{(1)}x_{n1}$	$-a_1^{(2)}x_{n1}$	...	$-a_m^{(1)}x_{nm}$	$a_m^{(2)}x_{nm}$	$-y_n$
$F(X)$	c_1	c_2	...	c_n	c_{n+1}	c_{n+2}	...	c_{n+2m-1}	c_{n+2m}	F_0

Примечание. F_0 — искомый экстремум; $c_1, c_2, ..., c_{n+2m}$ — значения строки функционала.

цию принципа выбора разрешающих строк и столбцов в случае наличия нескольких предполагаемых разрешающих элементов.

При выборе разрешающей строки в случае совпадения значений минимального положительного симплексного отношения B предполагаемых разрешающих элементов к соответствующей правой части ограничений было использовано правило Креко [22, с. 42]. Оно заключается в выборе такой строки, в которой раньше встретится наименьшее частное отношения элементов строк к предполагаемым разрешающим элементам. Принцип выбора столбца при совпадении оптимальных элементов заключается в поиске наибольшего положительного значения, полученного при делении правой части ограничений на предполагаемые разрешающие элементы.

Таким образом, текущий вариант симплекс-метода можно считать модифицированным в рамках исследуемой задачи.

Опишем принцип алгоритма градиентного спуска. Для этого введем несколько обозначений [13]. Пусть $\mathbf{\Omega}$: $\{\Omega_1, ..., \Omega_n\}$ будет множеством всех гиперплоскостей вида

$$\Omega_i = \Omega(\mathbf{a}, \mathbf{x}_i, y_i) = y_i - \langle \mathbf{a}, \mathbf{x}_i \rangle = 0 \ \ (i = 1, ..., n).$$

Тогда узловая точка будет представлять собой точку пересечения m независимых гиперплоскостей

$$\mathbf{u} = \bigcap_{s \in M} \Omega_s, \ M = \{k_1, ..., k_m\},$$
$$k_1 < k_2 < ... < k_m, \ k_l \in \{1, ..., n\}.$$

Обозначим U множество всех узловых точек. Назовем узловой такую прямую, которая будет являться пересечением $(m - 1)$ независимых гиперплоскостей:

$$l_{(k_1,...,k_{m-1})} : \cap\Omega_i, \ i \in \{k_1, ..., k_{m-1}\}, \ k_l \in \{1, ..., n\}.$$

Алгоритм заключается в спуске по узловым прямым за конечное число переходов к точке, в которой будет найдено точное решение задачи (1) [10]. В качестве первого приближения берут случайную узловую точку $\mathbf{u}^{(0)}$, получающуюся путем решения системы линейных алгебраических уравнений порядка m. Далее, поочередно убирая наблюдения из системы, проверяют все проходящие через точку узловые прямые и находят такую, которая приведет к меньшему относительно текущего значению целевой функции (1). Алгоритм продолжает проверку до тех пор, пока не найдется точка, дальнейший спуск из которой будет приводить только к увеличению значения целевой функции.

Отличием от обычного спуска является способ проверки прилежащих узловых прямых, где нужно было вычислить значения целевой функции в каждой точке на прямой, что увеличивало вычислительную сложность алгоритма. В градиентном спуске за счет использования производных по направлению значение целевой функции нужно вычислить лишь один раз. В каждой узловой точке $\mathbf{u}^{(*)} = (u_1^{(*)}, ..., u_m^{(*)})$ по направлению прямой находится производная, равная сумме n слагаемых [13]:

$$\frac{\partial Q(\mathbf{u}^{(*)})}{\partial \mathbf{l}_{(k_1,...,k_{m-1})}} =$$

$$= \sum_{i=1}^{n} (c_1 + x_{i2}c_2 + ... + x_{im}c_m) \operatorname{sign}\left(\sum_{j=1}^{m} u_j^{(*)} x_{ij} - y_i \right),$$

где $\mathbf{l}_{(k_1,...,k_{m-1})} = (c_1, \ c_2, \ ..., \ c_m)$ — направляющий вектор узловой прямой $l_{(k_1,...,k_{m-1})}$.

Если производная по направлениям слева $\left.\frac{\partial Q(\mathbf{u}^{(*)})}{\partial \mathbf{l}_{(k_1,...,k_{m-1})}}\right|_{\mathbf{u}_-^{(*)}}$ и справа $\left.\frac{\partial Q(\mathbf{u}^{(*)})}{\partial \mathbf{l}_{(k_1,...,k_{m-1})}}\right|_{\mathbf{u}_+^{(*)}}$ меняет знак, то для данной прямой достигнут экстремум, где целевая функция имеет свое минимальное значение.

Вычислительная сложность алгоритма составляет $O(n \ln^2 n \, m^2) < O(n^{1,4} m^2)$ [13].

Обсуждение результатов

Эксперименты для оценки точности и определения вычислительной сложности алгоритмов проводили на суперкомпьютере «Уран» ИММ УрО РАН, время их работы оценивали на ноутбуке Lenovo Legion 7 16ACHg6 с восьмиядерным процессором Ryzen 7 5800H. Алгоритмы реализованы при помощи языка программирования C++ в среде Microsoft Visual Studio 2019. Для генерации загрязненных случайных ошибок ε использовали модель Тьюки – Хьюбера [23, 24] с вероятностью засорения $\gamma = 0{,}1$:

$$F_\gamma(x) = (1 - \gamma)F(x) + \gamma f F_H(x),$$

где $F(x)$ — это функция распределения случайных ошибок, обладающая необходимыми «хорошими» признаками; $F_H(x)$ — функция распределения засорений.

Были сгенерированы выборка со стандартным нормальным распределением и выборки распределения Гаусса с ненулевым математическим ожиданием, а также двустороннее и одностороннее распределения Коши. Выбор вероятности засорения $\gamma = 0{,}1$ обусловлен тем, что при статистическом моделировании случайных ошибок у выборок данных наблюдался разброс фактических частот засорения от нескольких сотых

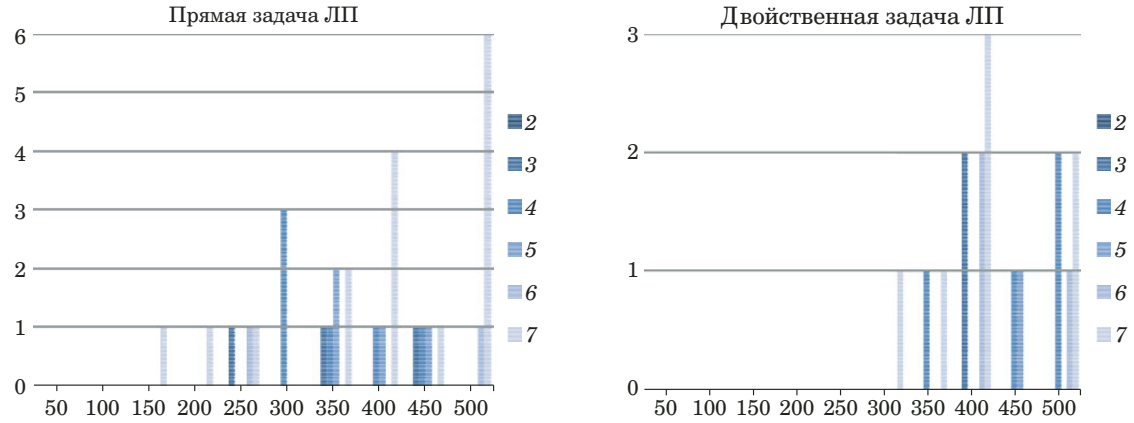

Рис. 1. Число отклонений решения прямой и двойственной задач ЛП при помощи симплекс-метода от решения градиентным спуском по узловым прямым для $n = 50, 100, ..., 500$ и $m = 2, 3, ..., 7$

Fig. 1. The number of deviations of the solution of a direct and dual linear programming problems using the simplex method from the solution by gradient descent along nodal straight lines for $n = 50, 100, ..., 500$ and $m = 2, 3, ..., 7$

до 0,15 – 0,2. Ситуации с бóльшим засорением случайных ошибок на практике маловероятны.

Оценим точность алгоритмов на основе сравнения решений 1000 вычислительных экспериментов для четырех вариантов генерации случайных ошибок. Сравним результаты, полученные при помощи симплекс-метода, с результатами алгоритма градиентного спуска. Согласно рис. 1, наибольшее число отклонений достигнуто для решения прямой задачи ЛП при $n = 500$, $m = 7$ и для двойственной задачи при $n = 400$, $m = 7$. При 4000 экспериментах это составляет 0,15 и 0,08 % соответственно, т.е. пренебрежимо мало, что свидетельствует о высокой точности алгоритмов.

Полученные результаты противоречат выводам, приведенным ранее в [12], где уже при 32 наблюдениях в симплекс-методе наблюдалось 5,5 % отклонений от точного решения. Здесь был применен модифицированный симплекс-метод, противодействующий появлению зацикливаний и приводящий к решению за конечное число шагов оптимальным путем. Автор [12] использовал внутреннюю функцию языка программирования R, которая из-за отсутствия подобной защиты и более строгого ограничения числа знаков после запятой могла приводить решение задачи (1) к отклонению от точного значения.

Для определения вычислительной сложности алгоритма симплекс-метода при решении прямой и двойственных задач в первую очередь необходимо провести проверку на однородность числа итераций, необходимых для нахождения решения при помощи симплекс-метода для одного вычислительного эксперимента и четырех вариантов генерации случайных ошибок. Воспользуемся t-критерием Стьюдента и U-критерием Манна – Уитни для независимых выборок, для оценки уравнений регрессии отдельных подвыборок применим тест Чоу [25]. Это позволит оценить статистическую значимость различий самих выборок, а также поведение алгоритма в зависимости от появления различных возмущений.

Согласно результатам (табл. 2) вероятность принятия нулевой гипотезы об однородности числа итераций значительно больше 0,05, и она не отвергается. Текущие результаты можно считать исчерпывающими, так как однородность числа итераций второй и третьей выборок исходит из однородности первой и второй, а также первой и третьей выборок. Аналогичным способом можно получить и оставшиеся отношения. Это позволит использовать среднее значение числа итераций каждой из генераций и тем са-

Таблица 2. Оценка однородности числа итераций алгоритмов на основе симплекс-метода при разных распределениях случайных ошибок

Table 2. Estimation of the homogeneity of the number of iterations of algorithms based on the simplex method for different distributions of random errors

Метод Стьюдента		Метод Манна – Уитни		Метод Чоу	
Выборка	G_1	Выборка	G_1	Выборка	G_1
Прямая задача ЛП					
G_2	0,97	G_2	0,93	G_2	0,86
G_3	0,90	G_3	0,91	G_3	0,82
G_4	0,90	G_4	0,87	G_4	0,90
Двойственная задача ЛП					
G_2	0,85	G_2	0,89	G_2	0,99
G_3	0,92	G_3	0,95	G_3	0,97
G_4	0,92	G_4	0,85	G_4	0,98

Примечание. G_i — варианты генерации выборки случайных ошибок: G_1 — стандартное нормальное распределение; G_2 — распределение Гаусса с ненулевым математическим ожиданием; G_3 и G_4 — двустороннее и одностороннее распределение Коши.

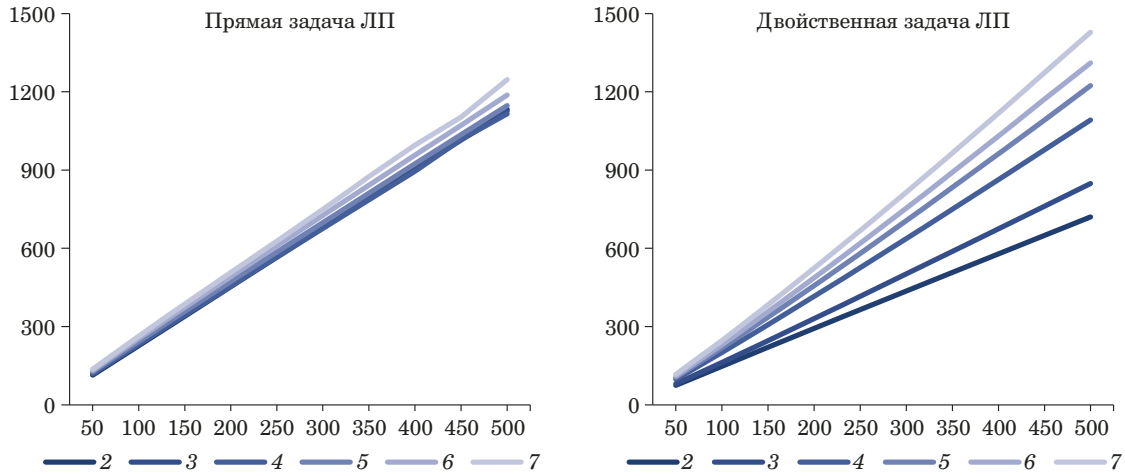

Рис. 2. Зависимость числа итераций решения прямой и двойственной задач ЛП при помощи симплекс-метода для $n = 50, 100, ..., 500$ и $m = 2, 3, ..., 7$

Fig. 2. Dependence of the number of iterations in solving a direct and dual linear programming problem using the simplex method for $n = 50, 100, ..., 500$ and $m = 2, 3, ..., 7$

мым в четыре раза расширит количество оценок при помощи метода Монте-Карло.

Из рис. 2 видно, что в отличие от двойственной задачи ЛП прямая задача менее чувствительна к увеличению числа коэффициентов m. В результате, несмотря на преимущество решения двойственной задачи в начале, с ростом m она начинает все сильнее проигрывать. Кроме того, если не учитывать число итераций, необходимых для приведения опорного плана к допустимому виду, то для решения прямой задачи ЛП нужно меньше итераций, чем для решения двойственной задачи. Следовательно, ее первое приближение находится ближе к конечному решению, что изначально нивелируется необходимостью приведения симплекс-таблицы к допустимому виду, но снова проявляется при увеличении m. Так, при $m = 7$, $n = 500$ на решение двойственной задачи пришлось на 200 итераций больше, чем для решения прямой.

Утверждение. Средняя вычислительная сложность решения прямой и двойственной задачи ЛП при помощи симплекс-метода равна $O(n^{3,2}m^{0,2})$ и $O(n^3m^{0,5})$ соответственно.

Доказательство. Определим вычислительную сложность одной итерации решения задачи ЛП при помощи симплекс-метода. Число логических операций, необходимых для нахождения разрешающего элемента, будет равно сумме строк и столбцов симплекс-таблицы и составлять $(5n + 2m + 1)$. Исходя из найденного элемента, происходит преобразование матрицы методом Гаусса – Жордана, а именно:

деление разрешающей строки на разрешающий элемент — $(3n + 2m + 1)$ операций;

преобразование оставшихся элементов матрицы — $(4n – 2)(3n + 2m + 1)$ операций;

вычисление симплексного отношения — $2n$ операций;

определение значений строки функционала — $(4n + 1)(3n + 2m) + 4n$ операций.

Таким образом, общая сложность вычислений для одной итерации будет составлять порядка $(24n^2 + 16mn + 2m + 15n)$ операций или $O(n^2)$ с учетом $n > m$.

Для определения общей вычислительной сложности алгоритмов нужно найти зависимость числа итераций от n и m исходя из рис. 2. Для прямой задачи ЛП получим $n^{1,2}m^{0,1}$ с коэффициентом детерминации $R^2 = 0{,}99$ и для двойственной задачи ЛП — $nm^{0,5}$ с $R^2 = 0{,}99$. Следовательно, средняя вычислительная сложность алгоритмов будет равна произведению сложности одной итерации на найденную зависимость. Для прямой задачи ЛП она будет составлять $O(n^{3,2}m^{0,1}))$, а для двойственной задачи — $O(n^3m^{0,5})$.

Однако стоит учитывать, что оценка вычислительной сложности в нотации *BigO* — это мера, отражающая в большей степени узкие места алгоритмов, которая не рассчитана на их всесторонний анализ. Поэтому ее следует рассматривать в совокупности с другими критериями и оценками. Сравним среднее время определения коэффициентов регрессии в (1) с помощью решения прямой и двойственной задачи ЛП со средним временем градиентного спуска по узловым прямым для 100 экспериментов. Из рис. 3 видно, что градиентный спуск работает быстрее как минимум в 10 раз, причем этот выигрыш растет с увеличением n (например, при $n = 500$ выигрыш в быстродействии достигает 2140 и 670 раз для решения прямой и двойственной задач ЛП соответственно). Средний выигрыш по вычислительной сложности градиентного спуска по узловым

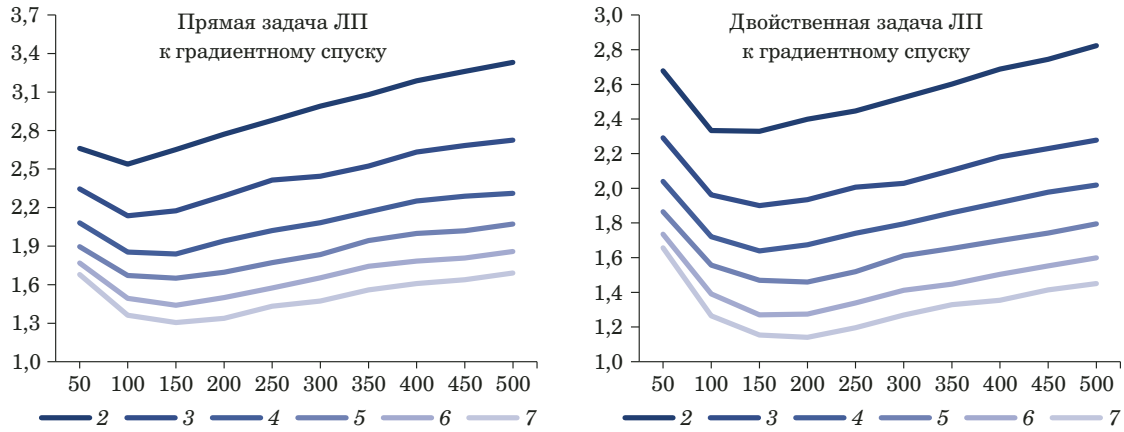

Рис. 3. Десятичный логарифм от деления среднего времени определения коэффициентов линейной регрессии путем решения прямой и двойственной задач ЛП на среднее время градиентного спуска для $n = 50, 100, ..., 500$ и $m = 2, 3, ..., 7$

Fig. 3. The decimal logarithm of dividing the average time of determining regression coefficients by solving a direct and dual linear programming problem by the average gradient descent time for $n = 50, 100, ..., 500$ and $m = 2, 3, ..., 7$

прямым относительно решения прямой и двойственной задач ЛП составляет соответственно $O(n^{1,8}m^{-1,8})$ и $O(n^{1,6}m^{-1,5})$.

Наличие локальных минимумов у графиков связано с двумя факторами. Во-первых, темп роста времени вычисления симплекс-метода относительно объема выборки n постепенно возрастает до достижения некоторого максимума, после чего начинает уменьшаться. Так, для решения прямой задачи ЛП этот максимум достигается при $n = 150 – 200$, а для двойственной задачи — при $n = 200 – 250$. Во-вторых, темп роста времени вычисления для градиентного спуска с ростом n монотонно замедляется, стремясь к единице.

Отрицательные степени у $O(m^{-1,8})$ и $O(m^{-1,5})$ свидетельствуют о том, что при фиксированном n с ростом порядка модели m эффективность градиентного спуска снижается по сравнению с решениями прямой и двойственной задач ЛП.

Меньшая степень для m у $O(n^{3,2}m^{0,1})$ по сравнению с $O(n^3m^{0,5})$ говорит о том, что увеличение порядка модели m в меньшей степени влияет на рост вычислительной сложности прямой задачи ЛП по сравнению с двойственной задачей.

Возникает вопрос, почему при числе итераций прямой задачи ЛП, меньшем, чем для двойственной (см. рис. 2), она сильнее проигрывает по быстродействию градиентному спуску с ростом n? Это связано с ограничениями правой части, которые для двойственной задачи равны **b**, а значит, представляют собой набор нулей и единиц. Поэтому при вычислении значений строки функционала используется меньше вычислительных ресурсов, чем для прямой задачи ЛП, так как умножение на ноль или единицу либо сохраняет значение сомножителя, либо его обнуляет. Однако этот проигрыш в быстродействии не столь значителен, например, для $n = 500$ прямая задача в среднем вычислялась медленнее в 3,2 раза при $m = 2$ и в 1,7 раза при $m = 7$. А при $n \le 50$ различия в быстродействии становятся менее 10 %. Поэтому в большинстве практических ситуаций различия в трудоемкости прямой и двойственной задач ЛП не существенны.

Заключение

Выполнена оценка вычислительной сложности алгоритмов реализации МНМ с помощью решения прямой и двойственной задач ЛП. Установлено, что оба варианта значительно проигрывают градиентному спуску по узловым прямым как в плане вычислительной сложности алгоритмов, так и по времени вычисления. Причем выигрыш градиентного спуска растет с увеличением n — в сотни и более раз. Варьирование типа распределения случайных ошибок не оказывает значительного влияния на вычислительную трудоемкость. Решения прямой и двойственной задач ЛП для выборок объема менее 50 наблюдений по быстродействию практически не различаются, с увеличением объема выборки решение двойственной задачи постепенно выигрывает по быстродействию в 2 – 4 раза.

Конфликт интересов

Авторы заявляют об отсутствии конфликта интересов.

ЛИТЕРАТУРА

1. **Мудров В. И., Кушко В. Л.** Методы обработки измерений. Квазиправдоподобные оценки. — М.: Радио и связь, 1983. — 304 с.
2. **Орлов А. И.** Многообразие моделей регрессионного анализа (обобщающая статья) / Заводская лаборатория. Диагностика материалов. 2018. Т. 84. № 5. С. 63 – 73. DOI: 10.26896/1028-6861-2018-84-5-63-73

3. **Нелюбин А. П., Подиновский В. В.** Аппроксимация таблично заданных функций: многокритериальный подход / Журнал вычислительной математики и математической физики. 2023. Т. 63. № 5. С. 717 – 730. DOI: 10.31857/S0044466923050174
4. **Basset G., Koenker R.** Asymptotic theory of least absolute error regression / Journal of the American Statistical Association. 1978. Vol. 73. N 363. P. 618 – 622.
5. **Birkes D., Dodge Y.** Alternative Methods of Regression. — John Wiley & Sons, 1993. — 239 p.
6. **Болдин М. В., Симонова Г. И., Тюрин Ю. Н.** Знаковый статистический анализ линейных моделей. — М.: Наука. Физматлит, 1997. — 288 с.
7. **Wei Xue, Wensheng Zhang, Gaohang Yu.** Least absolute deviations learning of multiple tasks / Journal of Industrial & Management Optimization. 2018. N 14(2). P. 719 – 729. DOI: 10.3934/jimo.2017071
8. **Вучков И., Бояджиева Л., Солаков Е.** Прикладной линейный регрессионный анализ / Пер. с болг. — М.: Финансы и статистика, 1987. — 239 с.
9. **Авдюшев В. А., Мезенцева А. Д.** Метод наименьших модулей и его эффективность при обработке измерений с ошибками различного распределения / Известия вузов. Физика. 2012. Т. 55. № 10-2. С. 68 – 76.
10. **Тырсин А. Н., Азарян А. А.** Точное оценивание линейных регрессионных моделей методом наименьших модулей на основе спуска по узловым прямым / Вестник ЮУрГУ. Серия «Математика. Механика. Физика». 2018. Т. 10. № 2. С. 47 – 56. DOI: 10.14529/mmph180205
11. **Bloomfield P., Steiger W. L.** Least Absolute Deviations: Theory, Applications, and Algorithms. — Boston – Basel – Stuttgart: Birkhauser, 1983. — 349 p.
12. **Азарян А. А.** Быстрые алгоритмы моделирования многомерных линейных регрессионных зависимостей на основе метода наименьших модулей: дис. ... канд. физ.-мат. наук. — Екатеринбург, 2018.
13. **Тырсин А. Н.** Алгоритмы спуска по узловым прямым в задаче оценивания регрессионных уравнений методом наименьших модулей / Заводская лаборатория. Диагностика материалов. 2021. Т. 87. № 5. С. 68 – 75. DOI: 10.26896/1028-6861-2021-87-5-68-75
14. **Barrodale I., Roberts F. D. K.** An improved algorithm for discrete L1 linear approximation / SIAM Journal on Numerical Analysis. 1973. Vol. 10. P. 839 – 848.
15. **Narula S. C., Wellington J. F.** Algorithm AS108: Multiple linear regression with minimum sum of absolute errors / Applied Statistics. 1977. Vol. 26. P. 106 – 111.
16. **Armstrong R. D., Kung D. S.** Algorithm AS132: Least absolute value estimates for a simple linear regression problem / Applied Statistics. 1978. Vol. 27. P. 363 – 366.
17. **Панюков А. В., Мезал Я. А.** Параметрическая идентификация квазилинейного разностного уравнения / Вестник Южно-Уральского государственного университета. Серия: Математика. Механика. Физика. 2019. Т. 11. № 4. С. 32 – 38. DOI: 10.14529/mmph190404
18. **Голованов О. А., Тырсин А. Н.** Регрессионный анализ данных на основе метода наименьших модулей в динамических задачах оценивания / Заводская лаборатория. Диагностика материалов. 2023. Т. 89. № 5. С. 71 – 80. DOI: 10.26896/1028-6861-2023-89-5-71-80
19. **Wesolowsky G. O.** A new descent algorithm for the least absolute value regression problem / Communications in Statistics, Simulation and Computation. 1981. Vol. 10. N 5. P. 479 – 491. DOI: 10.1080/03610918108812224
20. **Hawley R. W., Gallagher Jr. N. C.** On Edgeworth's method for minimum absolute error linear regression / IEEE Transactions on Signal Processing. 1994. Vol. 42. N 8. P. 2045 – 2054. DOI: 10.1109/78.301827
21. **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.
22. **Богданова Е. Л., Соловейчик К. А., Аркина К. Г.** Оптимизация в проектном менеджменте. Линейное программирование. — СПб.: Университет ИТМО, 2017. — 165 с. https://books.ifmo.ru/file/pdf/2252.pdf
23. **Tukey J. W.** A Survey of Sampling from Contaminated Distribution / Contributions to Probability and Statistics. — Stanford: Stanford Univ. Press, 1960. P. 443 – 485.
24. **Хьюбер П.** Робастность в статистике / Пер. с англ. — М.: Мир, 1984. — 304 с.
25. **Chow G. C.** Tests of equality between sets of coefficients in two linear regressions / Econometrica. 1960. Vol. 28. N 3. P. 591 – 605. DOI: 10.2307/1910133

REFERENCES

1. **Mudrov V. I., Kushko V. L.** Measurement processing methods. Quasi-plausible estimates. — Moscow: Radio i svyaz, 1983. — 304 p. [in Russian].
2. **Orlov A. I.** Diversity of the models for regression analysis (generalizing article) / Industr. Lab. Mater. Diagn. 2018. Vol. 84. N 5. P. 63 – 73 [in Russian]. DOI: 10.26896/1028-6861-2018-84-5-63-73
3. **Nelyubin A. P., Podinovskiy V. V.** Approximation of tabular given functions: multicriteria approach / Computational Mathematics and Mathematical Physics. 2023. Vol. 63. N 5. P. 739 – 742. DOI: 10.1134/S0965542523050147
4. **Basset G., Koenker R.** Asymptotic theory of least absolute error regression / Journal of the American Statistical Association. 1978. Vol. 73. N 363. P. 618 – 622.
5. **Birkes D., Dodge Y.** Alternative Methods of Regression. — John Wiley & Sons, 1993. — 239 p.
6. **Boldin M. V., Simonova G. I., Tyurin Yu. N.** Sign statistical analysis of linear models. — Moscow: Nauka. Fizmatlit, 1997. — 288 p. [in Russian].
7. **Wei Xue, Wensheng Zhang, Gaohang Yu.** Least absolute deviations learning of multiple tasks / Journal of Industrial & Management Optimization. 2018. N 14(2). P. 719 – 729. DOI: 10.3934/jimo.2017071
8. **Vuchkov I., Boyadzhieva L., Solakov E.** Applied linear regression analysis. — Moscow: Finansy i statistika, 1987. — 239 p. [Russian translation].
9. **Avdyushev V. A., Mezentseva A. D.** The method of least modules and its effectiveness in processing measurements with errors of various distributions / Izv. Vuzov. Fizika. 2012. Vol. 55. N 10 – 2. P. 68 – 76 [in Russian].
10. **Tyrsin A. N., Azaryan A. A.** Exact evaluation of linear regression models by the least absolute deviations method based on the descent through the nodal straight lines / Vestn. Yuzh.-Ural. Gos. Univ. Ser. Matem. Mekh. Fizika. 2018. Vol. 10. No. 2. P. 47 – 56 [in Russian]. DOI: 10.14529/mmph180205
11. **Bloomfield P., Steiger W. L.** Least Absolute Deviations: Theory, Applications, and Algorithms. — Boston – Basel – Stuttgart: Birkhauser, 1983. — 349 p.
12. **Azaryan A. A.** Fast algorithms for modeling multivariate linear regression dependencies based on the least modulus method. Candidate's thesis. — Yekaterinburg, 2018 [in Russian].
13. **Tyrsin A. N.** Algorithms for descending along nodal lines in the problem of estimating regression equations by the method of least modules / Industr. Lab. Mater. Diagn. 2021. Vol. 87. N 5. P. 68 – 75 [in Russian]. DOI: 10.26896/1028-6861-2021-87-5-68-75
14. **Barrodale I., Roberts F. D. K.** An improved algorithm for discrete L1 linear approximation / SIAM Journal on Numerical Analysis. 1973. Vol. 10. P. 839 – 848.
15. **Narula S. C., Wellington J. F.** Algorithm AS108: Multiple linear regression with minimum sum of absolute errors / Applied Statistics. 1977. Vol. 26. P. 106 – 111.
16. **Armstrong R. D., Kung D. S.** Algorithm AS132: Least absolute value estimates for a simple linear regression problem / Applied Statistics. 1978. Vol. 27. P. 363 – 366.
17. **Panyukov A. V., Mezal Ya. A.** Parametric identification of quasilinear difference equation / Vestn. Yuzh.-Ural. Gos. Univ.

Ser. Matem. Mekh. Fizika. 2019. Vol. 11. N 4. P. 32 – 38 [in Russian]. DOI 10. 14529/mmph190404
18. **Golovanov O. A., Tyrsin A. N.** Regression analysis of data based on the method of least absolute deviations in dynamic estimation problems / Industr. Lab. Mater. Diagn. 2023. Vol. 89. N 5. P. 71 – 80 [in Russian].
DOI: 10.26896/1028-6861-2023-89-5-71-80
19. **Wesolowsky G. O.** A new descent algorithm for the least absolute value regression problem / Communications in Statistics, Simulation and Computation. 1981. Vol. 10. N 5. P. 479 – 491. DOI: 10.1080/03610918108812224
20. **Hawley R. W., Gallagher Jr. N. C.** On Edgeworth's method for minimum absolute error linear regression / IEEE Transactions on Signal Processing. 1994. Vol. 42. N 8. P. 2045 – 2054. DOI: 10.1109/78.301827
21. **Tyrsin A. N., Maksimov K. Ye.** Estimation of linear regression equations using the method of least modules / Industr. Lab. Mater. Diagn. 2012. Vol. 78. N 7. P. 65 – 71 [in Russian].
22. **Bogdanova Ye. L., Soloveychik K. A., Arkina K. G.** Optimization in project management: linear programming. — St. Petersburg: ITMO University, 2017. — 165 p. [in Russian] https://books.ifmo.ru/file/pdf/2252.pdf
23. **Tukey J. W.** A Survey of Sampling from Contaminated Distribution / Contributions to Probability and Statistics. — Stanford: Stanford Univ. Press, 1960. P. 443 – 485.
24. **Huber P.** Robust Statistics. — John Wiley & Sons, 1981. — 320 p.
25. **Chow G. C.** Tests of equality between sets of coefficients in two linear regressions / Econometrica. 1960. Vol. 28. N 3. P. 591 – 605. DOI: 10.2307/1910133