

**Национальный исследовательский университет  
«Высшая школа экономики»  
Санкт-Петербургская школа физико-математических и  
компьютерных наук**

**Отчет к лабораторной работе № 3  
«Метод барьеров»**

**Выполнил:  
студент группы МОАД  
Сморчков Данил Дмитриевич**

## Оглавление

<b>I. Теория</b> .....	3
<b>II. Эксперимент 1а. Сходимость метода барьеров в зависимости от <math>\gamma</math> (скорость увеличения <math>tk</math>) и <math>\epsilon_{inner}</math> (точность для метода Ньютона)</b> .....	7
<b>III. Эксперимент 1б. Сходимость метода барьеров в зависимости от <math>n</math> (размерность задачи), <math>m</math> (размер выборки) и <math>\lambda</math> (коэффициент регуляризации)</b> .....	10
<b>Вывод</b> .....	14

# I. Теория

Модель LASSO является одной из стандартных моделей линейной регрессии. Имеется обучающая выборка  $((a_i, b_i))_{i=1}^m$ , где  $a_i \in \mathbb{R}^n$  – вектор признаков  $i$ -го объекта, а  $b_i \in \mathbb{R}$  – его регрессионное значение. Задача заключается в прогнозировании регрессионного значения  $b_{new}$  для нового объекта, представленного своим вектором признаков  $a_{new}$ .

В модели LASSO, как и в любой модели линейной регрессии, прогнозирование выполняется с помощью линейной комбинации компонент вектора  $a$  с некоторыми фиксированными коэффициентами  $x \in \mathbb{R}^n$ :

$$b(a) := \langle a, x \rangle$$

Далее, обозначим через  $A \in \mathbb{R}^{m \times n}$  матрицу объектов-признаков, а через  $b \in \mathbb{R}^m$  – вектор ответов. Тогда обучение модели (нахождение оптимальных параметров модели  $x$ ) происходит с помощью оптимизации следующей задачи:

$$\phi(x) := \frac{1}{2} \|Ax - b\|^2 + \lambda \|x\|_1 \rightarrow \min_{x \in \mathbb{R}^n}$$

где  $\lambda$  – коэффициент регуляризации.

Как легко заметить, данная задача негладкая из-за регуляризационного члена. Однако, выполнив эквивалентное преобразование задачи через надграфик, можно получить гладкую условную задачу:

$$\begin{cases} \frac{1}{2} \|Ax - b\|^2 + \lambda \langle 1_n, u \rangle \rightarrow \min_{x, u \in \mathbb{R}^n} \\ -u \leq x \leq u \end{cases}$$

где  $1_n := (1, \dots, 1) \in \mathbb{R}^n$  и неравенства  $-u \leq x \leq u$  обозначают обычные поэлементные неравенства между соответствующими компонентами.

Для решения этой условной задачи воспользуемся методом логарифмических барьеров. Основная идея метода – сведение исходной условной задачи к последовательности специальным образом построенных безусловных задач, решения которых сходятся к решению исходной условной задачи.

Далее определим вспомогательную функции  $f_t(x) := tf(x) + F(x)$ , здесь  $f(x)$  – начальная функция,  $F(x)$  – барьерная функция. Используя логарифмическую барьерную функцию, запишем получающуюся на каждом шаге безусловную задачу оптимизации:

$$f_t(x, u) = t \frac{1}{2} \|Ax - b\|^2 + t\lambda \langle 1_n, u \rangle - \langle 1_n, \log(u + x) + \log(u - x) \rangle \rightarrow \min_{x, u \in \mathbb{R}^n}$$

где функция  $\log(\dots)$  применяется к каждой координате.

Таким образом, находя оптимум для задачи выше, мы каждый раз будем приближать решение основной задачи  $(x_t, u_t)^* \xrightarrow{t \rightarrow \infty} (x, u)^*$

На каждом шаге (для конкретного  $t$ ) мы будем решать данную задачу с помощью метода Ньютона. А потому нам понадобятся градиент и гессиан функции  $f_t(x, u)$ .

Градиент выглядит следующим образом:

$$\nabla f_t(x, u) = \begin{pmatrix} \frac{\partial f_t(x, u)}{\partial x} \\ \frac{\partial f_t(x, u)}{\partial u} \end{pmatrix}$$

Далее распишем каждый из компонентов:

$$\begin{aligned} \frac{\partial f_t(x, u)}{\partial x} &= tA^T(Ax - b) - \left( \frac{1}{u+x} - \frac{1}{u-x} \right) \\ \frac{\partial f_t(x, u)}{\partial u} &= t\lambda 1_n - \left( \frac{1}{u+x} + \frac{1}{u-x} \right), \end{aligned}$$

где  $\frac{1}{(\dots)}$  – операция поэлементного деления.

Гессиан же имеет блочный вид:

$$\nabla^2 f_t(x, u) = \begin{pmatrix} \frac{\partial^2 f_t(x, u)}{\partial x^2} & \frac{\partial^2 f_t(x, u)}{\partial x \partial u} \\ \frac{\partial^2 f_t(x, u)}{\partial u \partial x} & \frac{\partial^2 f_t(x, u)}{\partial u^2} \end{pmatrix}$$

Распишем каждый компонент:

$$\begin{aligned} \frac{\partial^2 f_t(x, u)}{\partial x^2} &= tA^T A + \text{diag} \left( \frac{1}{(u+x)^2} + \frac{1}{(u-x)^2} \right) \\ \frac{\partial^2 f_t(x, u)}{\partial u^2} &= \text{diag} \left( \frac{1}{(u+x)^2} + \frac{1}{(u-x)^2} \right) \\ \frac{\partial^2 f_t(x, u)}{\partial u \partial x} &= \frac{\partial^2 f_t(x, u)}{\partial x \partial u} = \text{diag} \left( \frac{1}{(u+x)^2} - \frac{1}{(u-x)^2} \right) \end{aligned}$$

где  $\frac{1}{(\dots)^2}$  – операция поэлементного деления и возведения в квадрат,

$\text{diag}(a)$  – создание диагональной матрицы с вектором  $a$  на главной диагонали.

Теперь все готово для того, чтобы найти направление на каждой итерации метода Ньютона. Необходимо решить систему линейных уравнений:

$$\begin{pmatrix} \frac{\partial^2 f_t(x, u)}{\partial x^2} & \frac{\partial^2 f_t(x, u)}{\partial x \partial u} \\ \frac{\partial^2 f_t(x, u)}{\partial u \partial x} & \frac{\partial^2 f_t(x, u)}{\partial u^2} \end{pmatrix} \begin{pmatrix} d_k^x \\ d_k^u \end{pmatrix} = - \begin{pmatrix} \frac{\partial f_t(x, u)}{\partial x} \\ \frac{\partial f_t(x, u)}{\partial u} \end{pmatrix}$$

Соответственно,  $d_k = \begin{pmatrix} d_k^x \\ d_k^u \end{pmatrix}$  является направлением в методе Ньютона.

Так как я храню матрицу гессиана в разреженном формате, то сначала пытался использовать разные солверы в `scipy.sparse.linalg.spsolve`, однако на 8-12 итерациях (видимо, из-за больших  $t$ ) они вычисляли направление, которое приводило к сингулярному гессиану на следующей итерации. Потому я просто взял стандартный солвер `numpy.linalg.solve` (правда приходится переводить матрицу в плотный формат, и вся идея разреженных матриц сошла на нет).

И хотя решаемые подзадачи являются безусловными необходима некоторая аккуратность при поиске длины шага  $\alpha$  на каждой итерации метода Ньютона (внутреннего цикла). А именно, длина шага должна быть такой, чтобы  $y_l + \alpha d_l \in \text{int}(Q)$  (каждая следующая точка лежала во внутренности множества, задаваемого функциональными ограничениями типа неравенств):

$$g_i(y_l + \alpha d_l) < 0, \quad 1 \leq i \leq m$$

Если учесть, что функции  $g_1, \dots, g_m$  являются аффинными:  $g_i(x) = \langle q_i, x \rangle - s_i$ , то можно получить следующую систему неравенств:

$$\langle q_i, y_l + \alpha d_l \rangle < s_i, \quad 1 \leq i \leq m$$

Произведя некоторые алгебраические преобразования, можно получить следующий результат:

$$\alpha < \alpha_l^{\max}, \quad \text{где } \alpha_l^{\max} := \min_{i \in I_l} \frac{s_i - \langle q_i, y_l \rangle}{\langle q_i, d_l \rangle},$$

$$I_l := \{1 \leq i \leq m: \langle q_i, d_l \rangle > 0\}$$

В нашей задаче условия выглядят достаточно просто:

$$\begin{cases} x - u \leq 0 \\ -x - u \leq 0 \end{cases}$$

Так как оптимизируемый параметр выглядит как  $xu = \begin{pmatrix} x \\ u \end{pmatrix}$ , то для нахождения  $I_l$  можно взять маску от произведения матрицы на вектор ( $\text{mask}(x)$ ):

$$I_l = \text{mask} \left( \begin{pmatrix} I_n & -I_n \\ -I_n & -I_n \end{pmatrix} d_l > 0 \right)$$

То есть сначала идут все неравенства  $x_i - u_i \leq 0$ , а потом  $-x_i - u_i \leq 0$ . А все  $q_i$  задаются как раз матрицей из  $I_n$ -блоков.

Для окончательного получения длины шага нужно помнить, что мы используем метод Ньютона, который в случае квадратичной аппроксимации лучше всего работает с  $\alpha = 0$ , значит получаем:

$$\alpha_l^0 := \min(1, 0.99\alpha_l^{max})$$

Далее используется критерий Армихо, и если длина шага не подошла, то запускается backtracking.

Что касается стратегии увеличения параметра  $t_k$ , то в данном задании используется линейное увеличение:

$$t_{k+1} := \gamma t_k,$$

где  $\gamma > 1$  – гиперпараметр (обычно порядка 10-100)

Кажется, что в качестве начальной точки  $xu$  можно взять любой вектор, который удовлетворяет неравенствам. Например:

$$xu = \begin{pmatrix} 1 \\ \dots \\ 1 \\ 1.1 \\ \dots \\ 1.1 \end{pmatrix}$$

Во внутреннем цикле (методе Ньютона) критерий останова классический:

$$\|\nabla f_t(x_k, u_k)\|^2 \leq \varepsilon \|\nabla f_t(x_k^0, u_k^0)\|^2$$

Для внешнего цикла используется зазор двойственности, получающийся из из двойственной задачи Фенхеля к исходной постановке с регуляризатором:

$$\phi(x) - \phi^* \leq \frac{1}{2} \|Ax - b\|^2 + \lambda \|x\|_1 + \frac{1}{2} \|\mu\|^2 + \langle b, \mu \rangle := \eta(x, \mu),$$

$$\text{где } \mu(x) := \min \left\{ 1, \frac{\lambda}{\|A^T(Ax - b)\|_\infty} \right\} (Ax - b)$$

Такой выбор обеспечивает стремление зазора двойственности  $\eta(x, \mu(x))$  к нулю при  $x \rightarrow x^*$ , что позволяет использовать условие  $\eta(x, \mu(x)) < \epsilon$  в качестве критерия останова.

## II. Эксперимент 1а. Сходимость метода барьеров в зависимости от $\gamma$ (скорость увеличения $t_k$ ) и $\epsilon_{inner}$ (точность для метода Ньютона)

В данном эксперименте использовался датасет abalone ( $A \in \mathbb{R}^{4177 \times 8}$ ),  $\lambda = 0.001$ ,  $\gamma \in [5, 10, 25, 50, 100, 250, 500, 1000]$ ,  $\epsilon_{inner} \in [1e-1, 1e-2, 1e-5, 1e-8, 1e-10]$

Гарантируемая точность по зазору двойственности против числа итераций

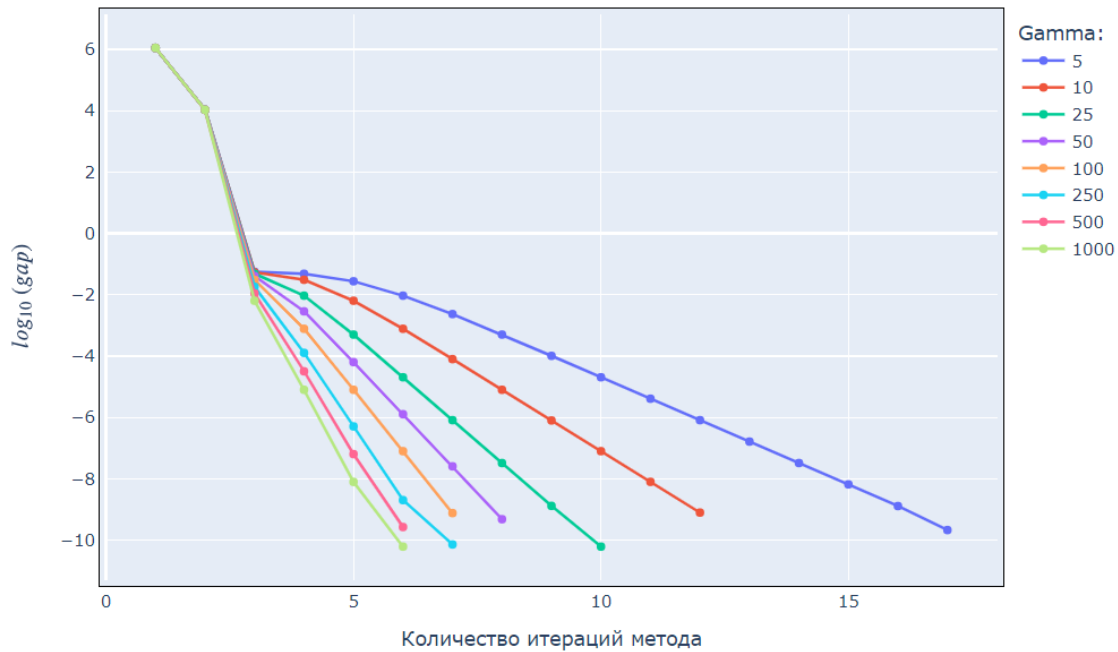


Рисунок 1 – Логарифм зазора двойственности против числа итераций для разных  $\gamma$

Гарантируемая точность по зазору двойственности против времени работы

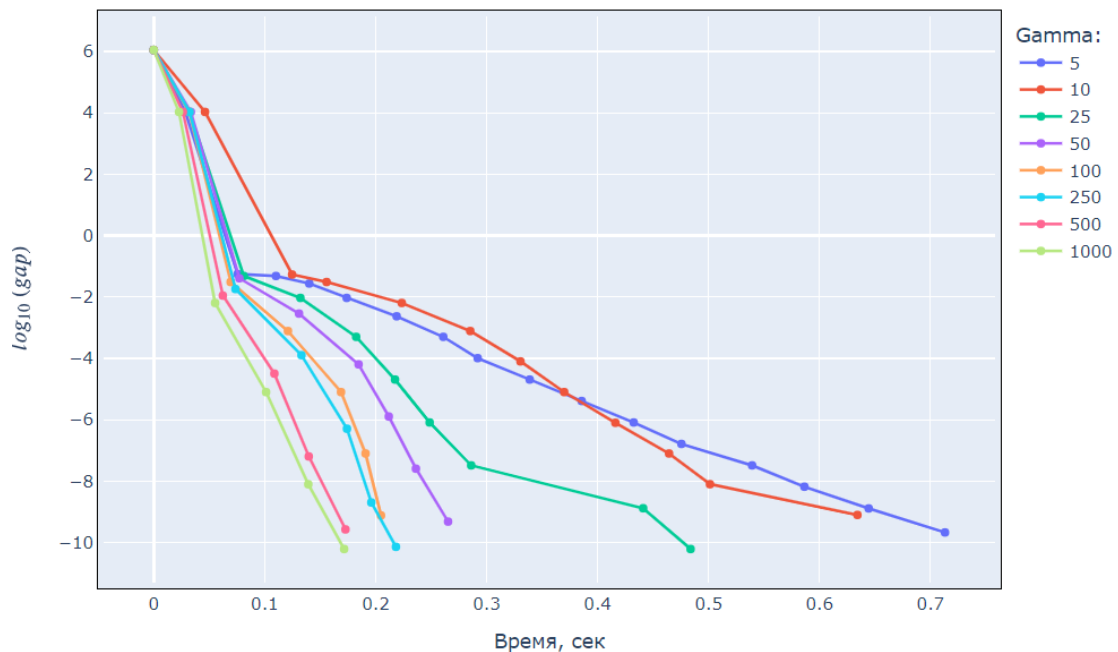


Рисунок 2 – Логарифм зазора двойственности против времени для разных  $\gamma$   
 Гарантируемая точность по зазору двойственности против числа итераций

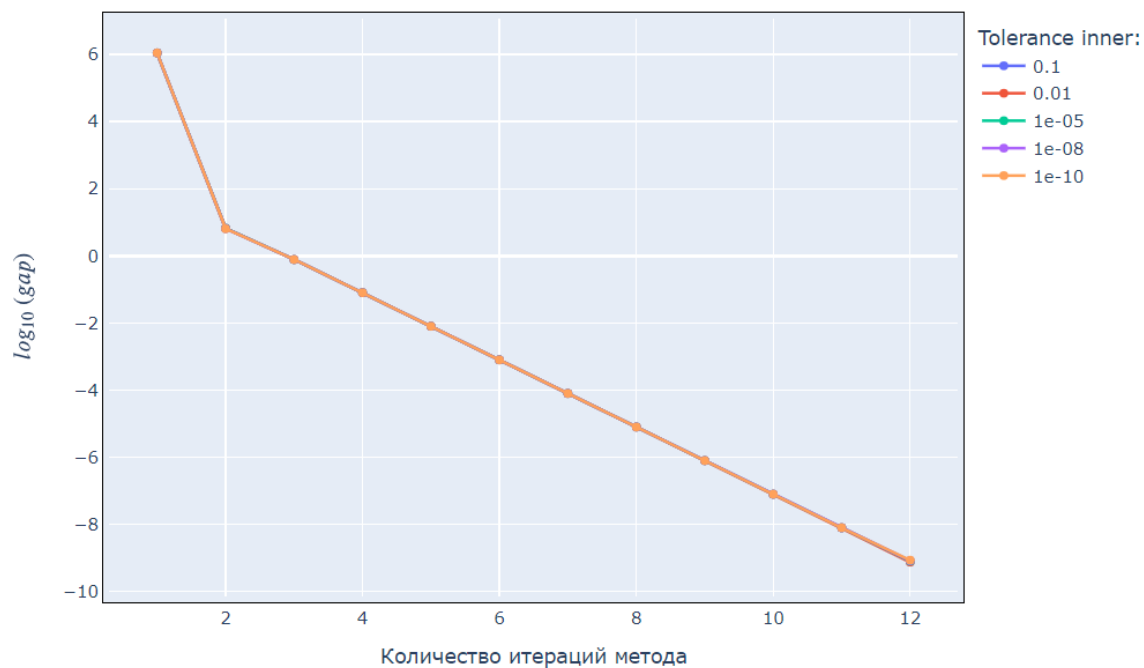


Рисунок 3 – Логарифм зазора двойственности против числа итераций для разных  $\epsilon_{inner}$

Гарантируемая точность по зазору двойственности против времени работы

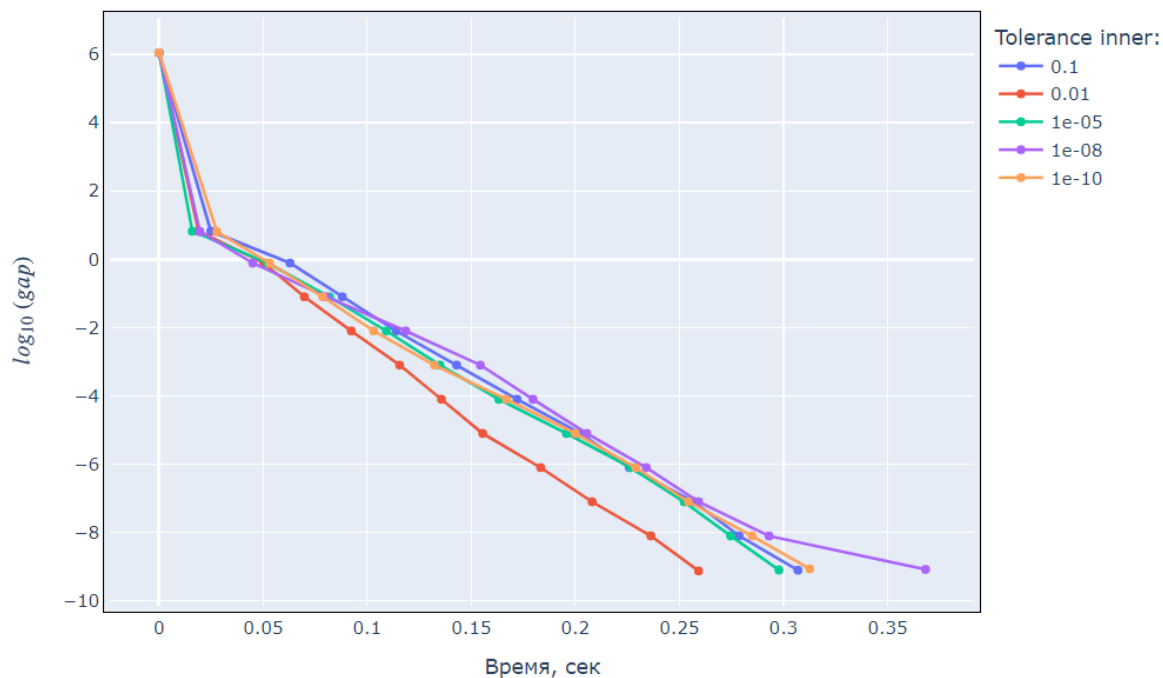


Рисунок 4 – Логарифм зазора двойственности против времени работы для разных  $\epsilon_{inner}$



Из Рисунков 1 и 2 видно, что скорость сходимости зависит от  $\gamma$  обратно пропорционально: чем больше  $\gamma$ , тем меньше скорость сходимости, притом, как по итерациям, так и по времени.

Из Рисунков 3 и 4 видно, что точность сходимости метода Ньютона не влияет на скорость в плане итераций метода, однако влияет на скорость в плане реального времени, однако эта зависимость не монотонна и слабо выражена в данном примере.

### III. Эксперимент 16. Сходимость метода барьеров в зависимости от $n$ (размерность задачи), $m$ (размер выборки) и $\lambda$ (коэффициент регуляризации)

В данном эксперименте использовались сгенерированные матрицы  $A$  и вектора коэффициентов  $b$ .

$$n \in [2, 5, 10, 50, 100, 500, 1000]$$

$$m \in [100, 500, 1000, 10000, 50000]$$

$$\lambda \in [1e-6, 1e-4, 1e-2, 1e-1, 1, 10, 100]$$

Гарантируемая точность по зазору двойственности против числа итераций

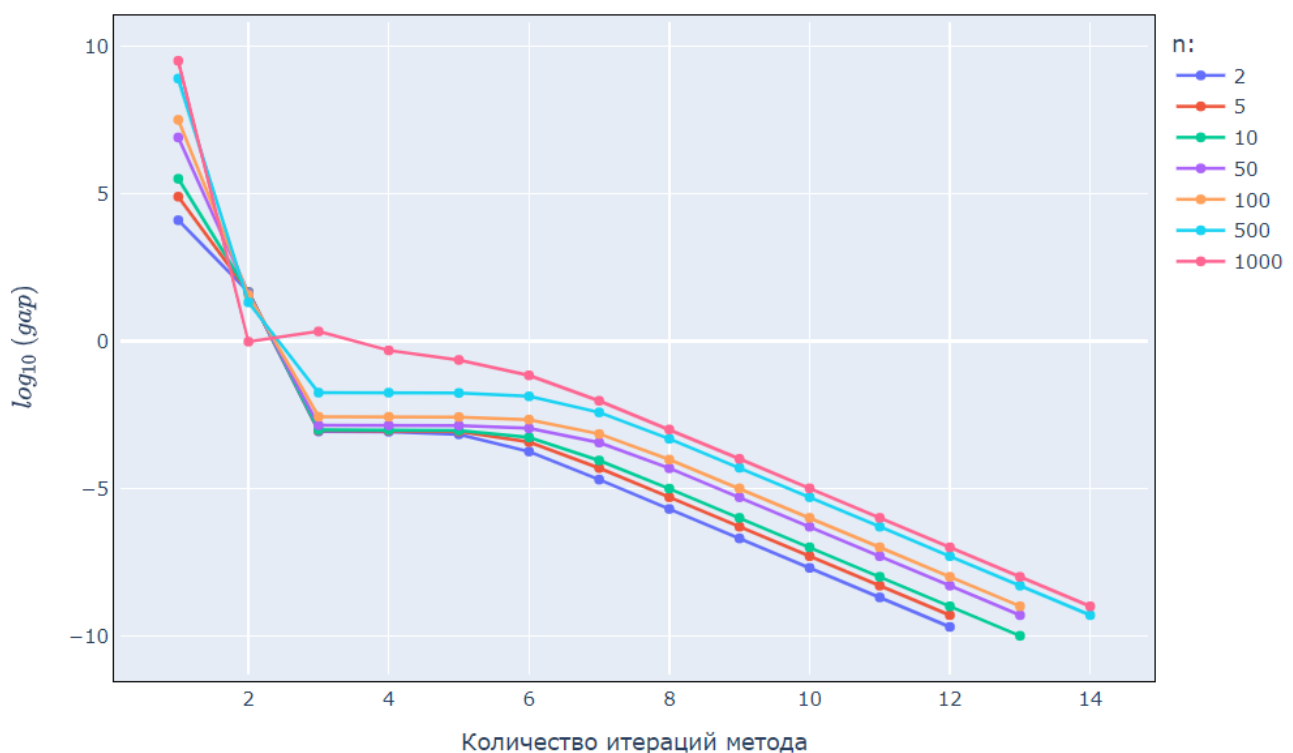


Рисунок 5 – Логарифм зазора двойственности против числа итераций для разных  $n$

### Гарантируемая точность по зазору двойственности против времени работы

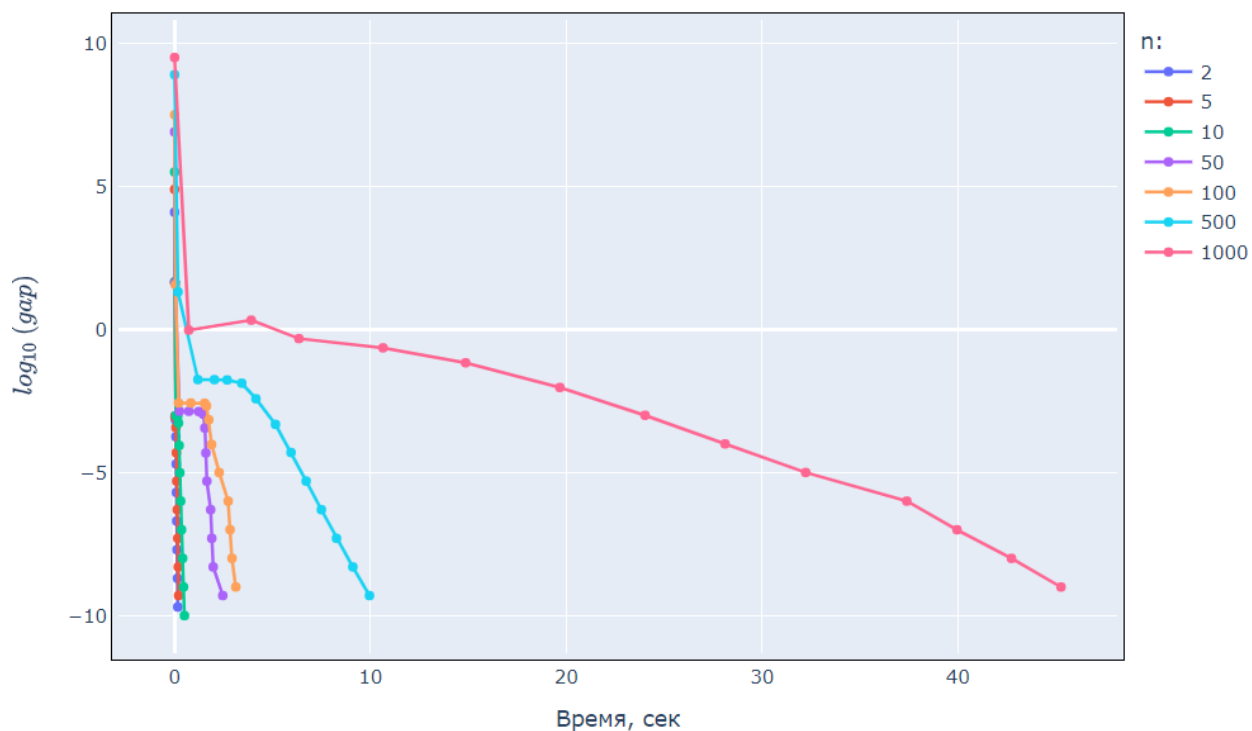


Рисунок 6 – Логарифм зазора двойственности против времени работы для разных  $n$

### Гарантируемая точность по зазору двойственности против числа итераций

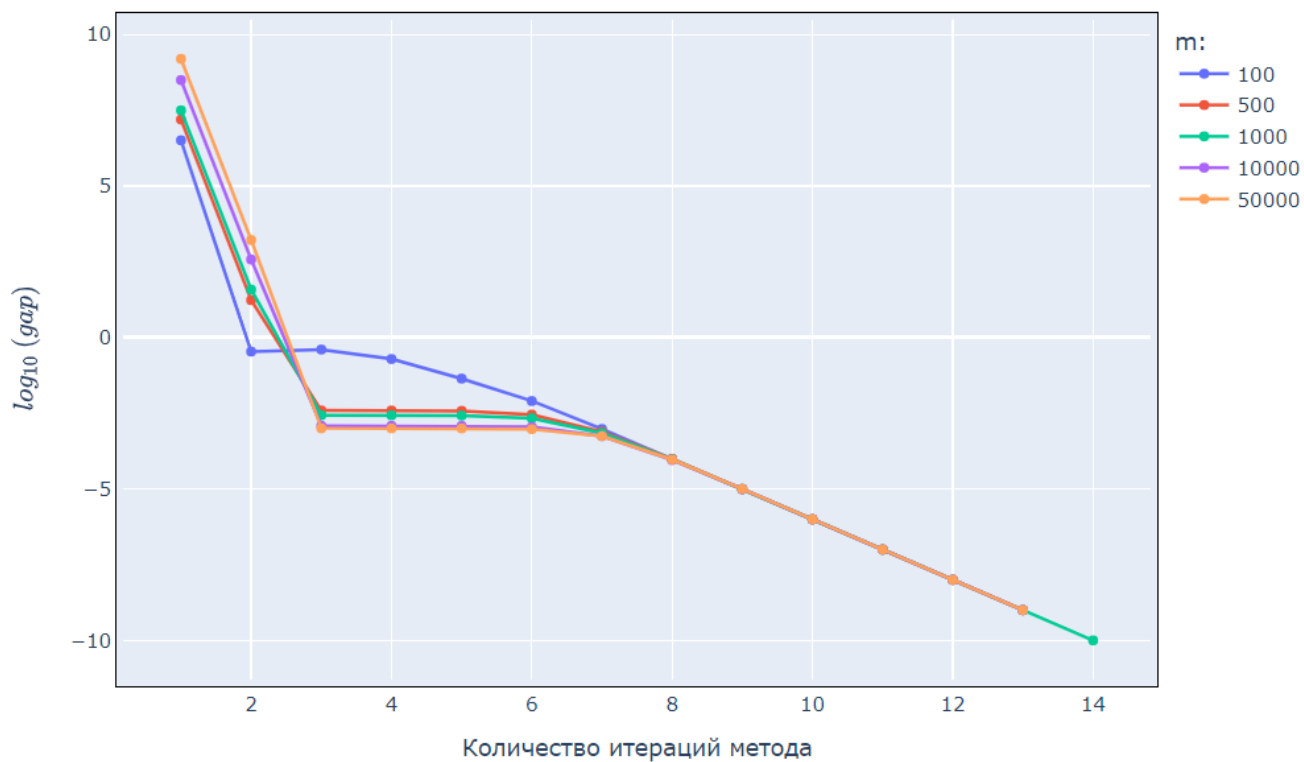


Рисунок 7 – Логарифм зазора двойственности против числа итераций для разных  $m$

### Гарантируемая точность по зазору двойственности против времени работы

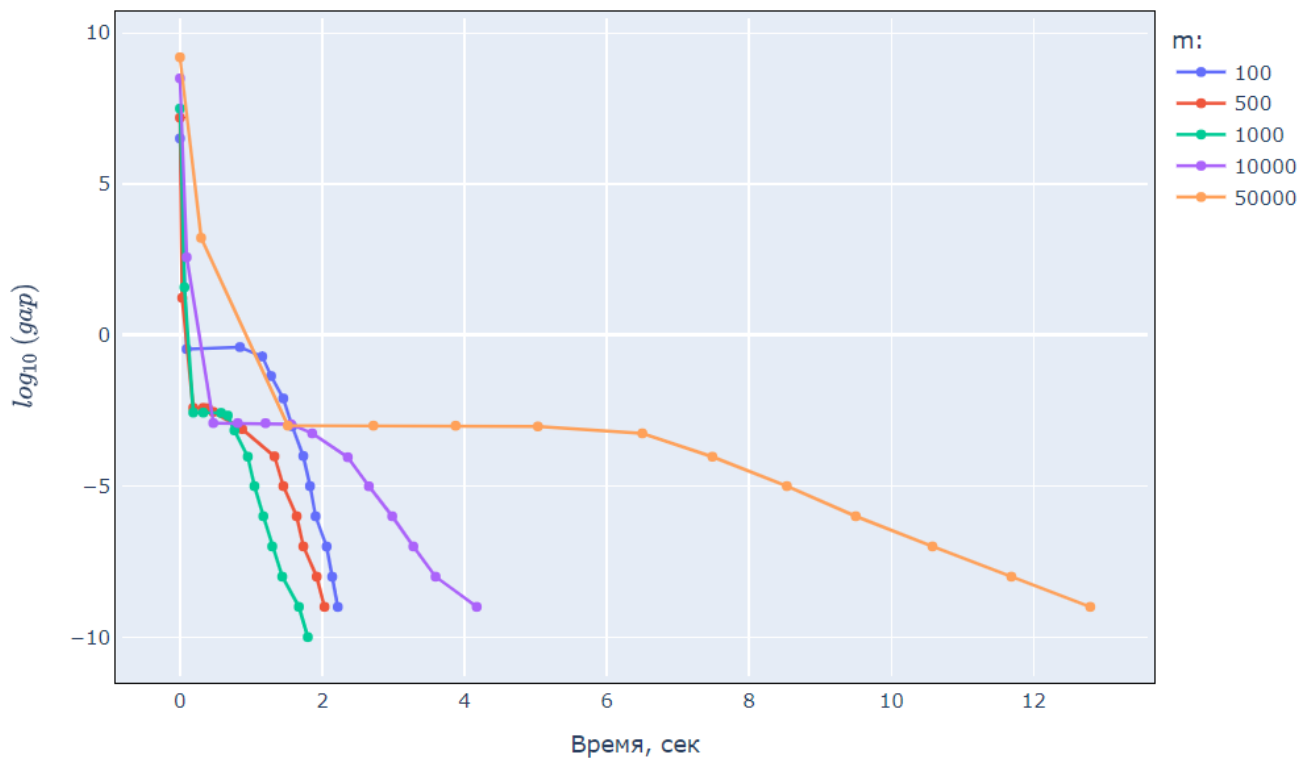


Рисунок 8 – Логарифм зазора двойственности против времени работы для разных  $m$

### Гарантируемая точность по зазору двойственности против числа итераций

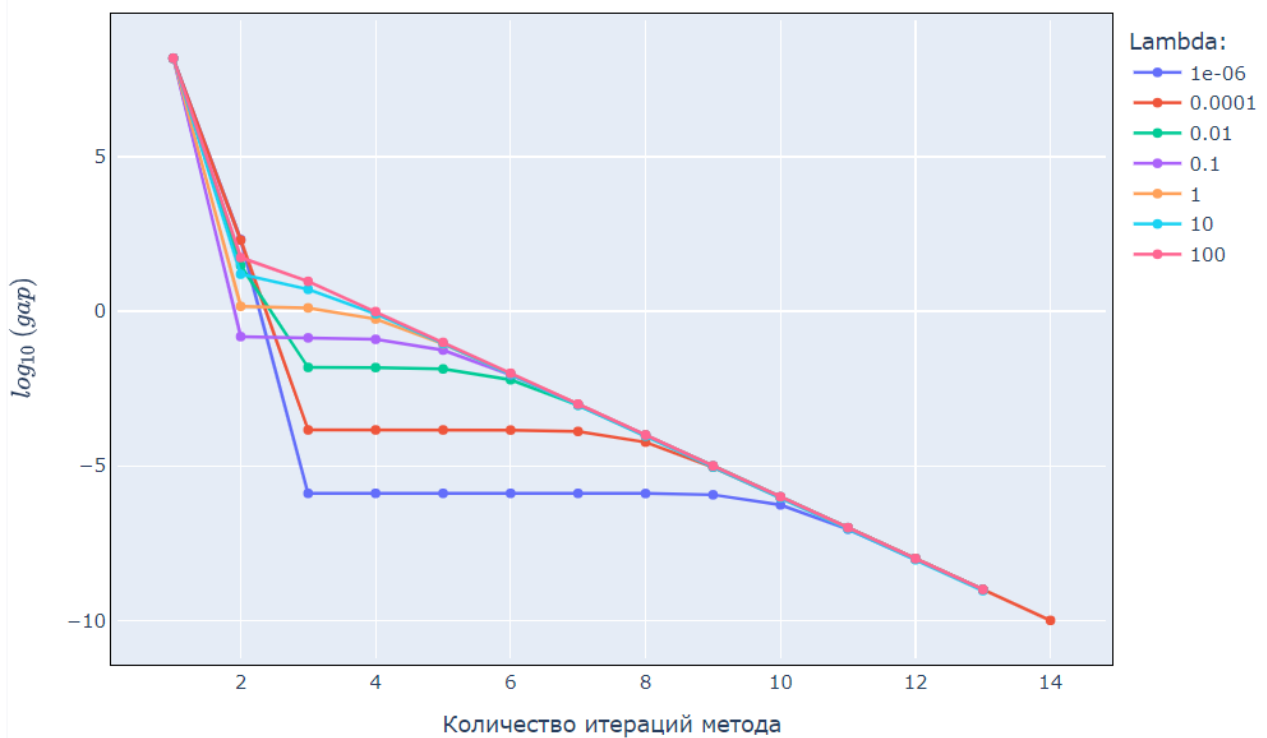


Рисунок 9 – Логарифм зазора двойственности против числа итераций для разных  $\lambda$

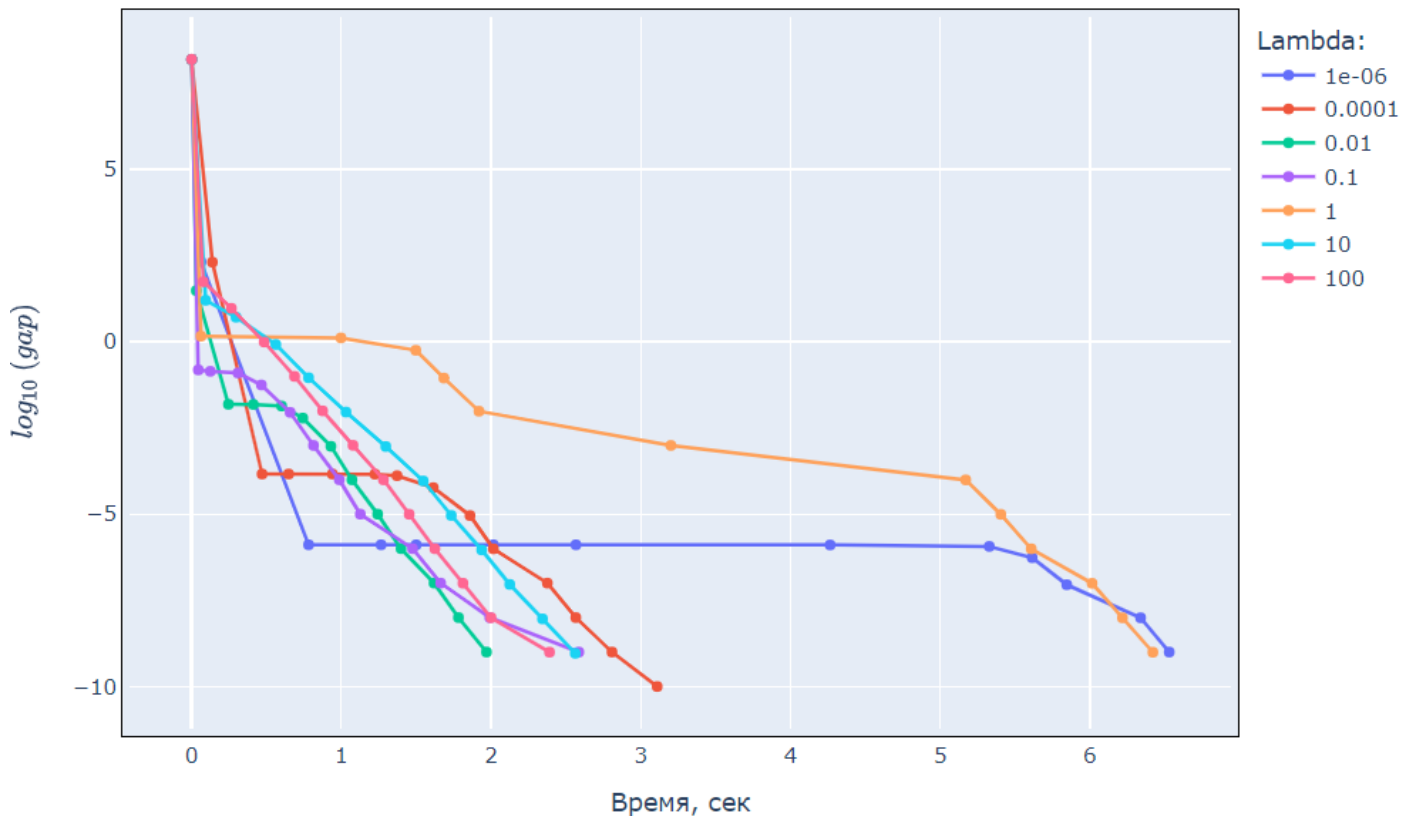


Рисунок 10 – Логарифм зазора двойственности против времени работы для разных  $\lambda$

Из Рисунков 5 и 6 видно, что скорость сходимости обратно зависит от размерности задачи  $n$ : чем больше  $n$ , тем больше итераций метода и времени работы алгоритма в целом.

Из Рисунков 7 и 8 видно, количество итераций практически не зависит от количества примеров, сходимость в этом плане практически не меняется. Однако ситуация другая, если рассматривать время работы, тут все ясно – чем больше  $m$ , тем дольше сходится метод барьеров, притом разница существенна.

Из Рисунков 9 и 10, видно, что метод барьеров при оптимизации LASSO-регрессии ведет себя по-разному в зависимости от коэффициента регуляризации  $\lambda$ , однако количество итераций до сходимости не меняется. Чего нельзя сказать о времени работы алгоритма – здесь вообще сложно что-то сказать: в целом скорость сходимости одинаковая, но есть два выброса из разных числовых диапазонов.

## **Вывод**

В данной работе была реализована модель LASSO-регрессии, а также метод логарифмических барьеров для решения данной задачи. Была приведена необходимая теория для данной задачи.

Проведены эксперименты для выявления чувствительности метода к основным настраиваемым параметрам.