Московский государственный университет имени М.В.Ломоносова
Факультет вычислительной математики и кибернетики
Кафедра математических методов прогнозирования

Отчет по теме
Выпуклая негладкая, условная и структурная оптимизация

Выполнил студент 517 группы
Камалов Руслан Рамилевич

Москва
2016

1 Метод барьеров

1.1 Поиск направления спуска

Вспомогательная функция, минимизируемая методом барьеров в задаче 1 имеет следующий вид

$$\phi_t(w^+, w^-) = \frac{\tau}{2n}\|X(w^+ - w^-) - y\|_2^2 + \tau\lambda\vec{1}^T(w^+ + w^-) - \sum_{i=1}^{d}\log(w_i^+) - \sum_{i=1}^{d}\log(w_i^-)$$

Выпишем систему линейных уравнений, задающие направление спуска (p^+, p^-). Для этого возьмем вторую производную функции $\phi_t(w^+, w^-)$ по вектору $w = (w^+, w^-)$. Для начала выпишем производные по каждой из двух составляющих вектора w : w^+ и w^-

$$\nabla^2_{w^+}\phi_t = \frac{\tau}{n}X^TX + d_+ = \tau A + d_+, \quad d_+ = diag\left(\frac{1}{(w_i^+)^2}\right)$$

$$\nabla^2_{w^-}\phi_t = \frac{\tau}{n}X^TX + d_- = \tau A + d_-, \quad d_- = diag\left(\frac{1}{(w_i^-)^2}\right)$$

$$\nabla^2_{w^+,w^-}\phi_t = \nabla^2_{w^-,w^+}\phi_t = -\frac{\tau}{n}X^TX = -\tau A$$

$$\nabla_{w^+}\phi_t = \frac{\tau}{n}X^T(X(w^+ - w^-) - y) - (\frac{\vec{1}}{w_i^+})$$

$$\nabla_{w^-}\phi_t = -\frac{\tau}{n}X^T(X(w^+ - w^-) - y) - (\frac{\vec{1}}{w_i^-})$$

Отсюда система для направления спуска имеет вид :

$$\begin{bmatrix} \tau A + d_+ & -\tau A \\ -\tau A & \tau A + d_- \end{bmatrix}\begin{bmatrix} p^+ \\ p^- \end{bmatrix} = \begin{bmatrix} -\nabla_{w^+}\phi_t \\ -\nabla_{w^-}\phi_t \end{bmatrix}$$

Прибавим ко второй "строке"матрицы системы первую :

$$\begin{bmatrix} \tau A + d_+ & -\tau A \\ d_+ & d_- \end{bmatrix}\begin{bmatrix} p^+ \\ p^- \end{bmatrix} = \begin{bmatrix} -\nabla_{w^+}\phi_t \\ -\nabla_{w^-}\phi_t - \nabla_{w^+}\phi_t \end{bmatrix}$$

Выразим из второго блока уравнений p^- через p^+, получим

$$p^- = -d_-^{-1}(\nabla_{w^-}\phi_t + \nabla_{w^+}\phi_t + d_+^{-1}p^+) = -d_-^{-1}(-(\frac{\vec{1}}{w_i^+}) - (\frac{\vec{1}}{w_i^-}) + d_+^{-1}p^+) \quad (1)$$

подставим выражение для p^- в первый блок уравнений, получим систему уравнений на p^+

$$(\tau A(I + d^+ d_-^{-1}) + d^+)p^+ = -\nabla_{w^+}\phi_t - \tau A d_-^{-1}(2\tau\lambda\vec{1} - ((\frac{\vec{1}}{w_i^+}) + (\frac{\vec{1}}{w_i^-}))) \quad (2)$$

В итоге система размером $2d \times 2d$ сведена к системе размером $d \times d$. Для построения матрицы этой системы умножение на диагональную матрицу $I + d^+ d_-^{-1}$, которое имеет линейную сложность $O(d)$, время для построения вектора правой части сопоставимо с временем умножения матрицы $A \in R^{d \times d}$ на вектор, то есть $O(d^2)$. Матрица $A(I + d^+ d_-^{-1}) + d^+$ не является матрицей специального вида, поэтому для решения системы использовался стандартный метод решения СЛАУ из numpy. После получения вектора p^+, p^- вычисляется из соотношения (1). Матрица A вычисляется один раз.

Достоинством такого метода является сокращение размеров реально решаемой системы с $2d \times 2d$ до $d \times d$. Быстрое вычисление матрицы и вектора правой части реальнорешаемой системы (2), за счет хранения матрицы $A \in R^{d \times d}$

Недостатком такого метода можно назвать хранение матрицы A, если размерность признакового пространства велика, то понадобиться много памяти на ее хранение. Так же, возможно, есть более эффективный метод для решения системы (2).

1.2 Выбор длины шага

После того, как направление спуска (p^+, p^-) найдено, необходимо выбрать длину шага α. Она выбирается исходя из условий:

1. $w^+ + \alpha p^+ \succeq 0, \quad w^- + \alpha p^- \succeq 0$
2. оптимальность длины шага для метода Ньютона : $\alpha = \min(1, \alpha)$
3. условие Армихо

Сначала ведется поиск α, удовлетворяющей первым двум условиям, далее производится деление α до выполнения условия

$$\phi_t(w^+ + \alpha p^+, w^- + \alpha p^-) \leq \phi_t(w^+, w^-) + c_1\alpha(\nabla\phi)^T p, \quad p = (p^+, p^-)$$

Перепишем это условие в виде

$$\alpha\left((p^{+}+p^{-})^{T}A(w^{+}+w^{-})-(p^{+}+p^{-})^{T}(\frac{1}{n}X^{T})y\right)+\alpha^{2}\frac{1}{2n}\|X(p^{+}-p^{-})\|^{2}+$$
$$\alpha\vec{1}(p^{+}+p^{-})-\frac{\sum_{i=1}^{d}\log(1+\alpha\frac{p^{+}}{w^{+}})-\sum_{i=1}^{d}\log(1+\alpha\frac{p^{-}}{w^{-}})}{\tau}-c_{1}\alpha(\nabla\phi)^{T}p\leq 0 \tag{3}$$

Отсюда видно, что перед началом поиска α можно предварительно вычислить значения выражений:

$$(p^{+}+p^{-})^{T}A(w^{+}+w^{-})-(p^{+}+p^{-})^{T}(\frac{1}{n}X^{T})y,\quad \frac{1}{2n}\|X(p^{+}-p^{-})\|^{2},$$
$$\vec{1}(p^{+}+p^{-}),\quad (\nabla\phi)^{T}p,$$

которые не изменяются по ходу поиска, что ускорит работу алгоритма. Схема подбора длины шага:

1. $\bar{\alpha}=\min(\min(-\frac{w_i^{+}}{p_i^{+}}),\min(-\frac{w_i^{-}}{p_i^{-}}))$
2. $\alpha=\min(1,\bar{\alpha})$
3. поиск α удовлетворяющий условию (3)

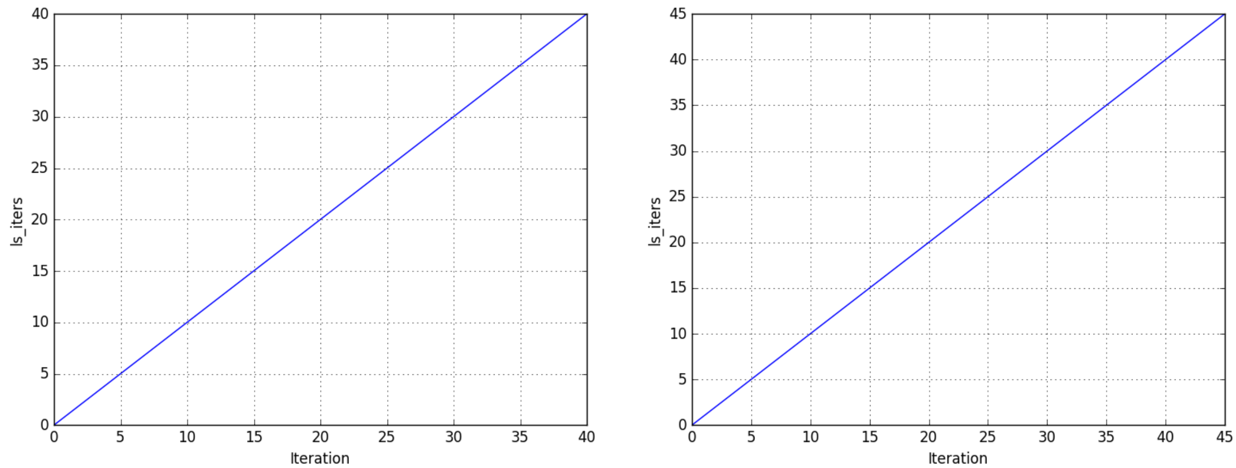
Начальная точка в методе барьеров должна лежать строго внутри множества ограничений, "достаточно далеко"от границы.

2 Сравнение реализованных методов

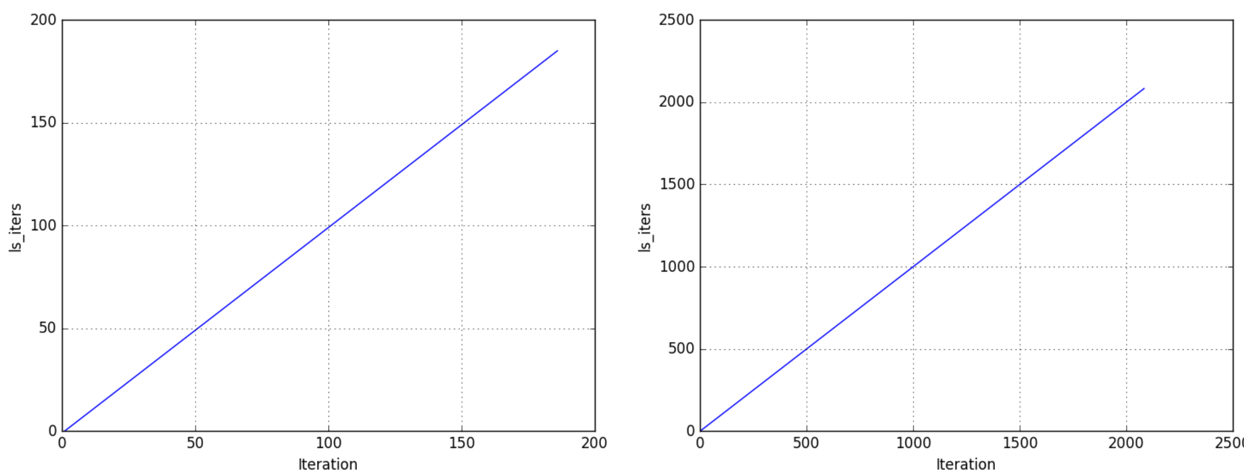
2.1 Число итераций одномерного поиска проксимального метода

Для эксперимента были выбраны два датасета с размерами:
Sonar $X\in R^{208\times 60}$ и Colon cancer cancer $X\in R^{62\times 2000}$. Ниже приведены графики числа итераций одномерного поиска от общего числа итераций проксимального метода с точностью 1e-2 и 1e-8 для каждого из датасетов.

SONAR SCALE



COLON CANCER

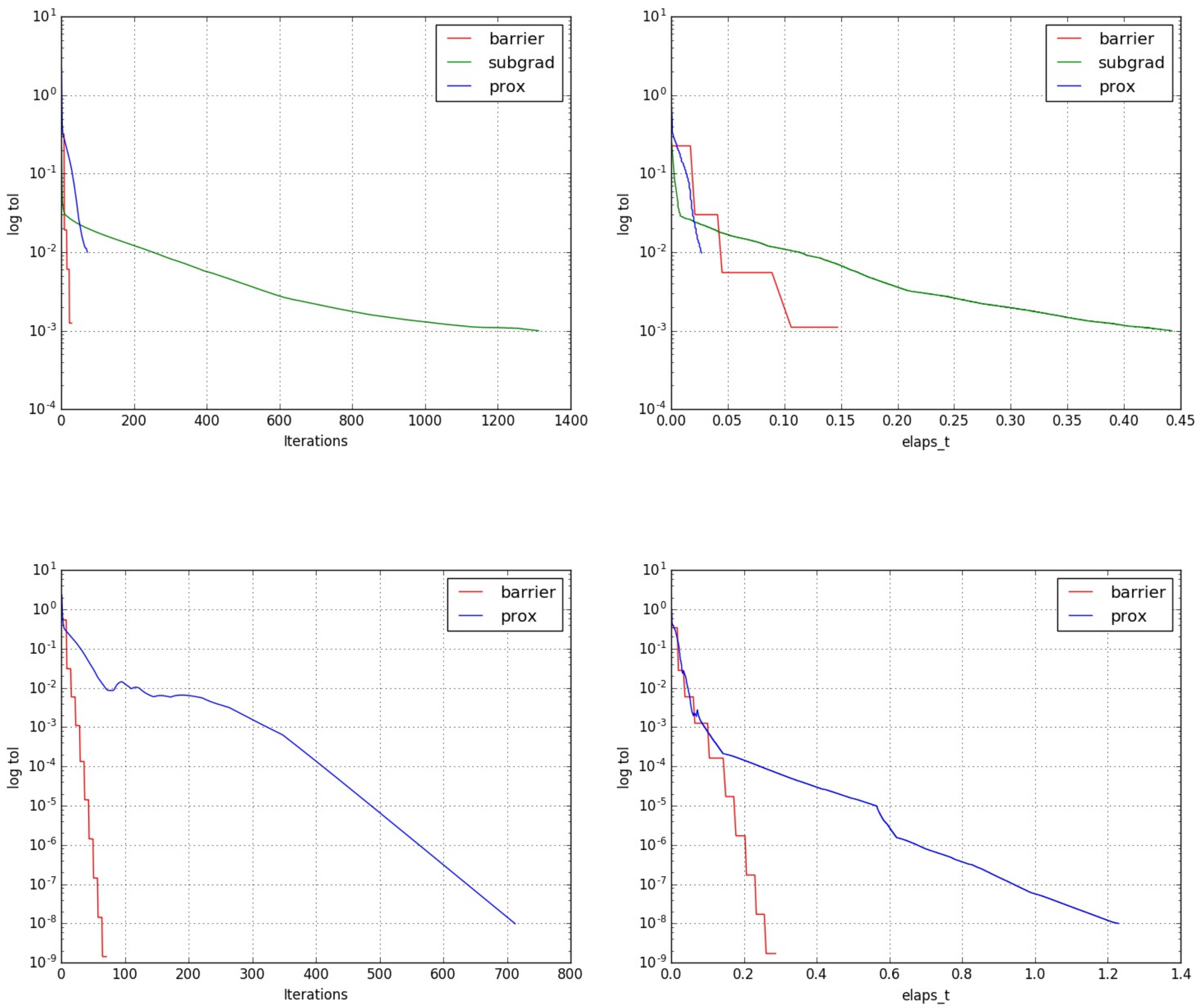


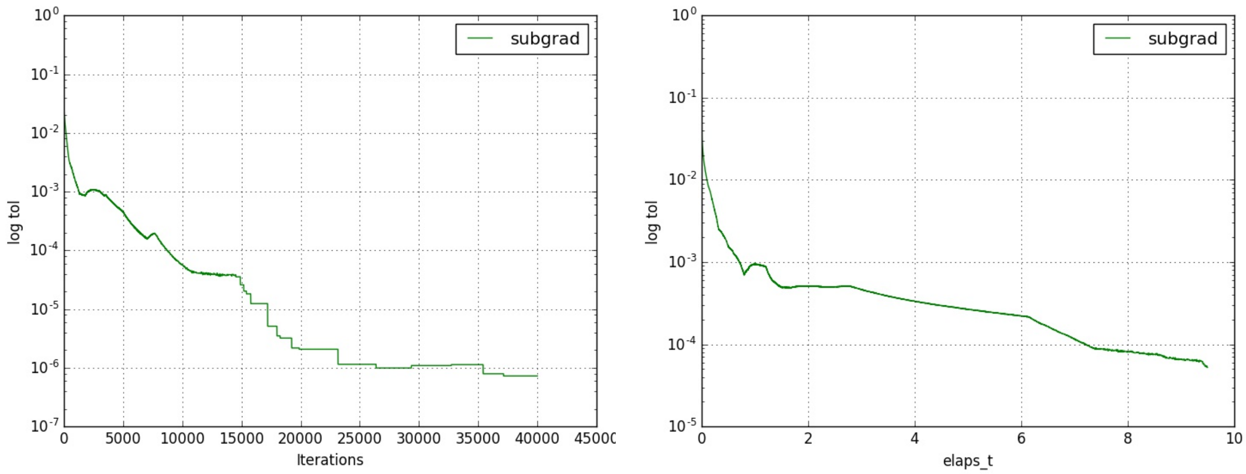
На выбранных датасетах на каждой итерации требовался один шаг одномерного поиска. Судя по этим результатам число тераций одномерного поиска в проксимальном методе не превышает двух.

2.2 Сравнение реализованных методов

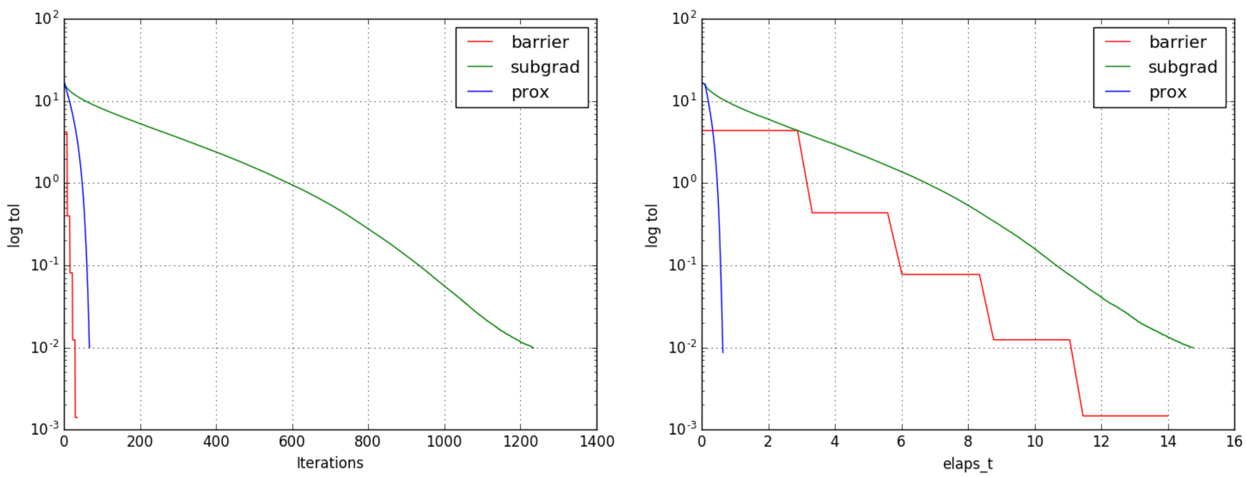
Ниже представлены результаты работы трех реализованных алгоритмов.

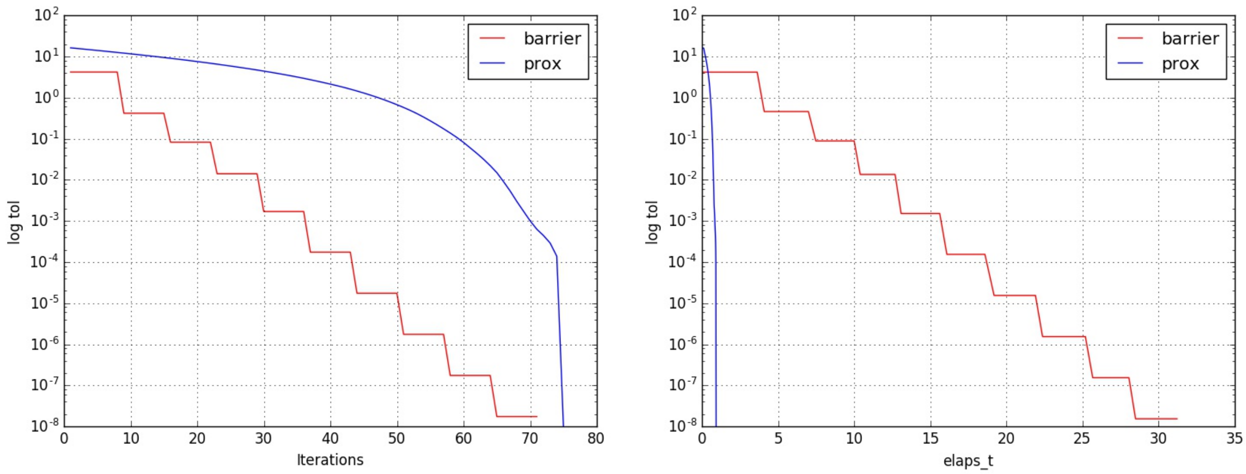
SONAR SCALE





COLON CANCER





На втором датасете субградиентный метод за разумное время не сходился.

Исходя из графиков сходимости методов, можно сделать выводы:

1. в задачах малой размерности низкой точности быстрее достигает проксимальный метод, высокой точности быстрее достигает метод барьеров

2. в задачах большой размерности быстрее работает проксимальный метод