Национальный исследовтельский университет ИТМО
Факультет информационных технологий и программирования
Прикладная математика и информатика

Методы оптимизации

Отчёт по лабораторной работе №1

Работу выполнили:
Плешанов Павел M3236
Чистяков Александр M3237
Королев Пётр M3236

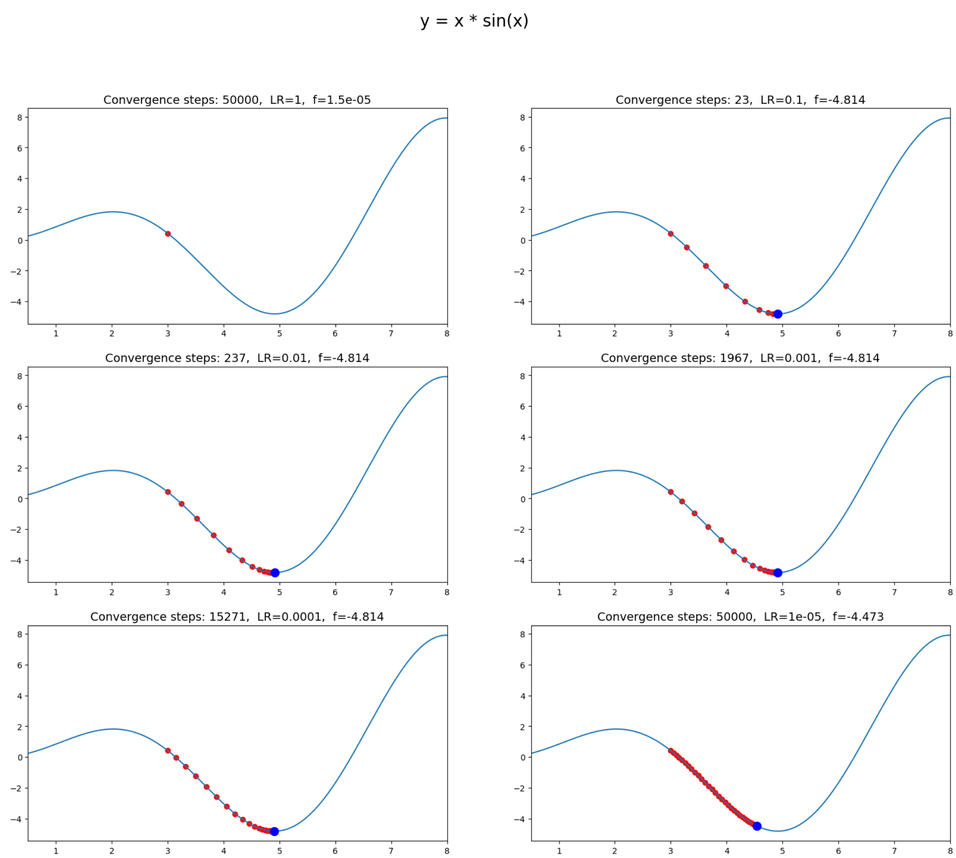
Санкт-Петербург
2021

1. Градиентный спуск с постоянным шагом (learning rate)

$$X = X - lr * \frac{d}{dX} f(X)$$

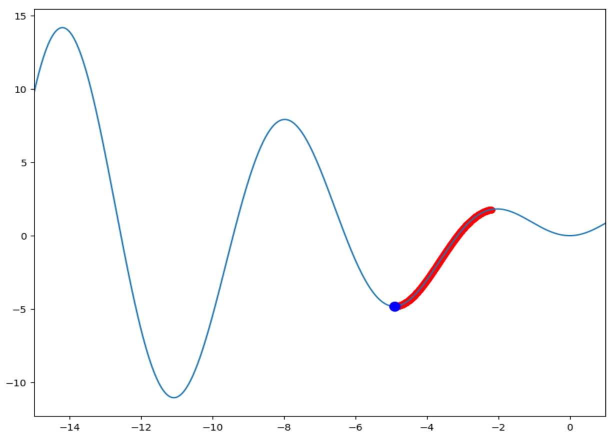
$$\begin{aligned} &Where, \\ &X \quad\quad = \; input \\ &F(X) \; = \; output \;\; based\; on\; X \\ &lr \quad\quad = \; learning\; rate \end{aligned}$$

X_0 = 3
Отчётливо видно, что lr
стоит выбирать определённым образом.
Не очень маленький (картинка 6), иначе будет недообучение и мы не достигнем минимума.
Также не очень большой (численная стабильность будет очень плохой).

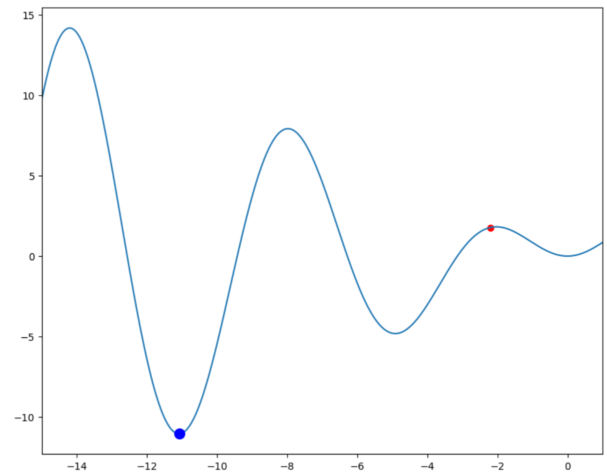


2. Реализуйте метод одномерного поиска (**метод дихотомии**, метод Фибоначчи, метод золотого сечения) и градиентный спуск на его основе.

total steps = 2558,
method=DefaultMethod f(x_min) = -4.81
x_min = -4.91



total steps = 3
method=DichotomyMethod f(x_min) = -11.04
x_min = -11.08



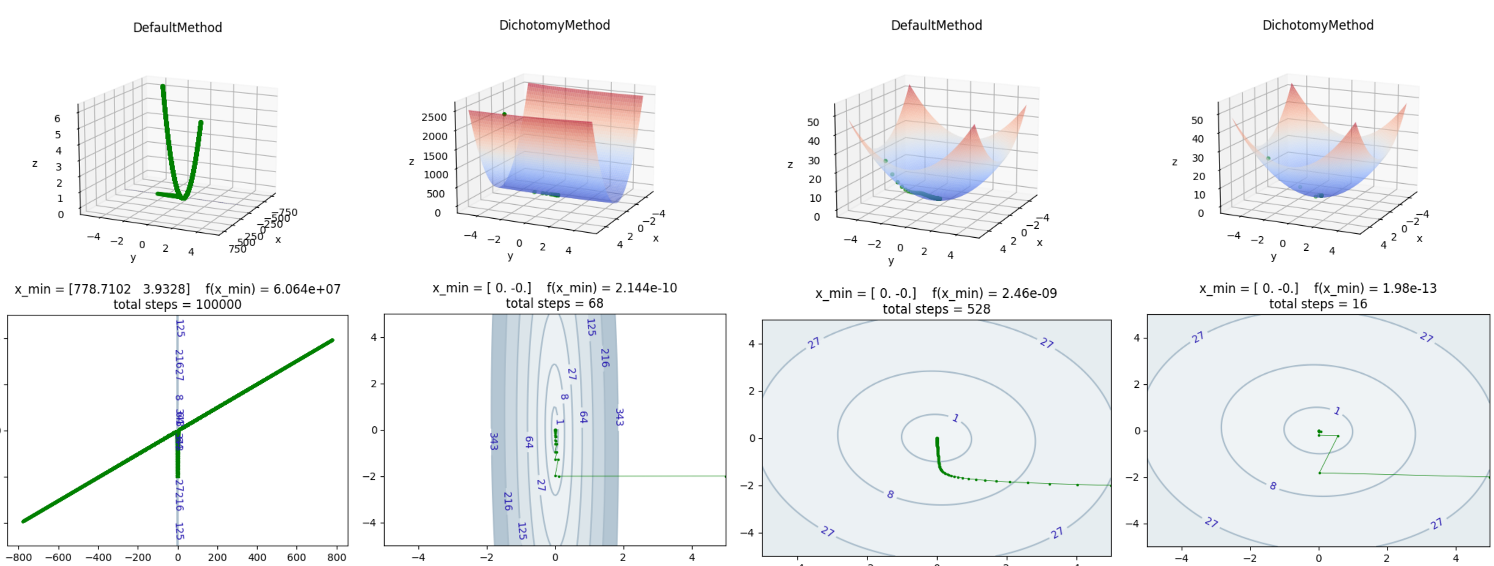
Метод дихотомии в данных начальных условиях отработал намного лучше (строго говоря сошёлся быстрее + более глубокий минимум).

3. Исследуйте влияние нормализации (scaling) на сходимость на примере масштабирования осей плохо обусловленной функции

Итого можно наблюдать, что оба метода линейного поиска проявили себя в разы лучше на нормализованных данных.
За кадром проводились эксперименты и оказалось, что с нормализацией можно выставлять больший learning rate, при котором метод будет сходиться => экономия вычислительных ресурсов и времени

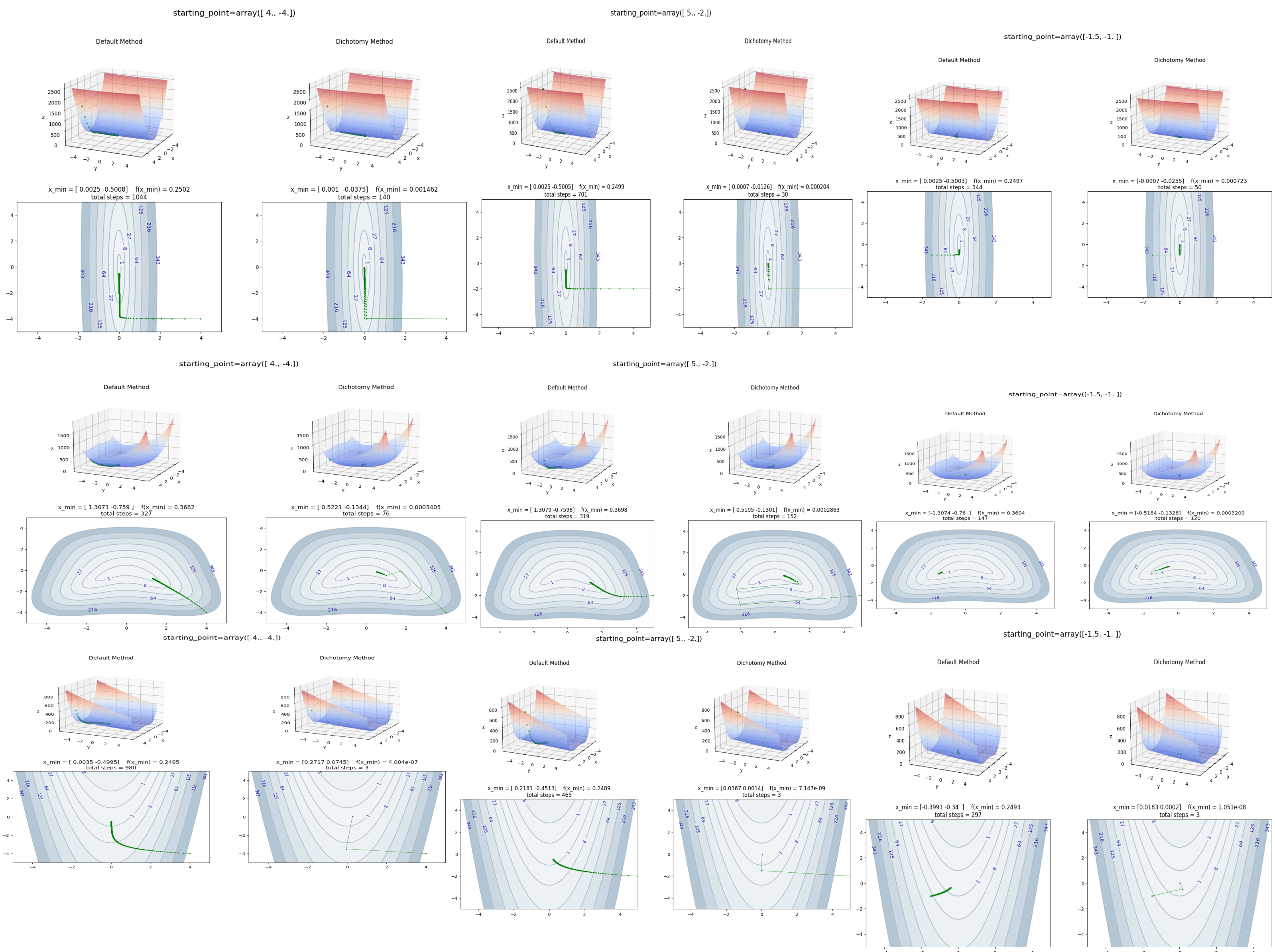
$$100 * x^2 + x * y + y^2 \quad => \quad 100 * (x/10)^2 + (x/10) * y + y^2$$

Unscaled **Scaled**



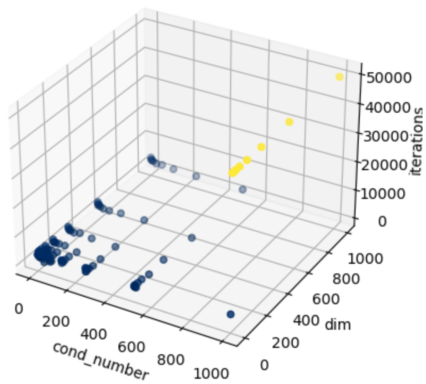
3, 4. Для каждой функции:
(a) исследуйте сходимость градиентного спуска с постоянным шагом, сравните полученные результаты для выбранных функций;
(b) сравните эффективность градиентного спуска с использованием одномерного поиска с точки зрения количества вычислений минимизируемой функции и ее градиентов;
(c) исследуйте работу методов в зависимости от выбора начальной точки;
~~(d) исследуйте влияние нормализации (scaling) на сходимость на примере масштабирования осей плохо обусловленной функции;~~
(e) в каждом случае нарисуйте графики с линиями уровня и траекториями методов;

После завершения этой серии опытов выяснилось, что метод градиентного спуска с дихотомией показывает лучшую сходимость (количество итераций уменьшается в 100 и более раз). Это логично, ведь на каждом шаге мы подбираем лучший learning rate => алгоритм может более "нагло" скатываться в локальный минимум.



$$fn_1 = 100 * x^2 + x * y + y^2 \quad fn_2 = (x^2 + 2 * y)^2 + y^4 \quad fn_3 = (x^2 - y)^2$$

5, 6, 7 (реализовано в коде)



8.

Пусть решается задача минимизации

$$\min_x f(x),$$

уже имеется приближение решения задачи x_k и пусть каким-либо методом мы нашли направление p_k, в котором будем искать новое приближение решения x_{k+1}. Тогда $x_{k+1} = x_k + \alpha_k p_k$, где α_k удовлетворяет условиям Вольфе:

$$f(x_k + \alpha_k p_k) \leq f(x_k) + c_1 \alpha_k \nabla f_k^T p_k,$$

$$\nabla f(x_k + \alpha_k p_k)^T p_k \geq c_2 \nabla f_k^T p_k$$

Константы выбираются следующим образом: $0 < c_1 < c_2 < 1$. Обычно константа c_1 выбирается достаточно маленькой (в окрестности 0), что означает, что функция после совершения шага должна уменьшиться, в то время как c_2 выбирается значительно большей (в окрестности 1), что, в свою очередь, означает, что проекция градиента в новом приближении должна либо изменить направление, либо уменьшиться.