

МЕТОД ПОКООРИНАТНОГО СПУСКА

Каретников Ф.В.

Бигаева Л.А., к.ф.-м.н., доцент
БФ УУНиТ, г.Бирск, РБ

Аннотация: Метод покоординатного спуска — это эффективный алгоритм оптимизации, основанный на последовательном поиске оптимальных значений переменных вдоль каждой координатной оси. Он широко применяется в различных областях, таких как математическое программирование, линейное программирование и нелинейное программирование. В статье рассматривается принцип работы метода, его применение.

Ключевые слова: численные методы, методы оптимизации, покоординатный спуск, алгоритм минимизации

Метод покоординатного спуска относится к группе прямых методов и основан на многократном применении алгоритмов многомерной оптимизации. Суть метода заключается в пошаговом приближении к точке минимума функции путём последовательных вариаций одной из координат при фиксированных значениях остальных. Рассмотрим алгоритм данного метода подробнее.

Пусть в n -мерном пространстве задана точка $X^{(0)}$ с координатами $x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}$, являющаяся точкой начального приближения к минимуму функции $f(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)})$. Зафиксируем все координаты, кроме первой — x_1 . Получим $f_1(x_1) = f(x_1, x_2, \dots, x_n)$ — функцию одной переменной. Для функции $f_1(x_1)$ решим задачу одномерной оптимизации и найдем значение $x_1^{(1)}$ — первую координату точки первого приближения к минимуму. Зафиксируем теперь все координаты, кроме x_2 , и решим задачу одномерной оптимизации для функции $f_2(x_2) = f(x_1^{(1)}, x_2, \dots, x_n^{(0)})$. Результатом решения будет $x_2^{(1)}$ — вторая

координата точки первого приближения к минимуму. Продолжая описанный процесс перебора переменных, получим все координаты $x_1^{(1)}, x_2^{(1)}, \dots, x_n^{(1)}$ точки первого приближения $X^{(1)}$. На этом первая итерация алгоритма завершается. Вторая и все последующие итерации алгоритма организуются аналогичным образом. Итерационный процесс завершается при выполнении условия близости точек, найденных на двух последовательных итерациях с номерами I и $I+1$:

$$\sqrt{\sum_{k=1}^n \left(x_k^{(i+1)} - x_k^{(i)} \right)^2} \leq \varepsilon,$$

где ε – малая величина, характеризующая точность расчетов [1, 2].

Таким образом, метод покоординатного спуска приводит задачу о нахождении наименьшего значения функции многих переменных к многократному решению одномерных задач оптимизации по каждому параметру [3].

Данный метод легко показать геометрически для случая функции двух переменных $z = f(x, y)$, описывающей некоторую поверхность в трехмерном пространстве. На рис. 1 нанесены линии уровня этой поверхности. Процесс оптимизации в этом случае проходит следующим образом. Точка M_0 описывает начальное приближение. Проводя спуск по координате x , попадем в точку M_1 . Далее, двигаясь параллельно оси ординат, придем в точку M_2 и т.д.

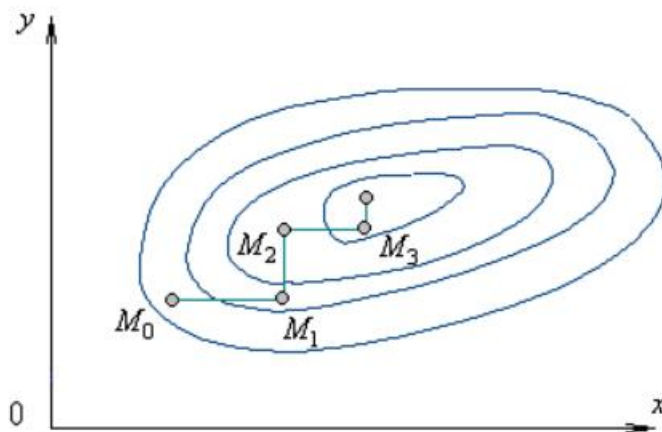


Рис.1. Линии уровня поверхности $z = f(x, y)$

Важным здесь является вопрос о сходимости данного процесса оптимизации. Иначе говоря, будет ли последовательность значений целевой функции сходиться к её наименьшему значению в данной области? Это зависит от вида функции и выбора начального приближения.

Для функции двух переменных очевидно, что метод неприменим в случае наличия изломов в линиях уровня. Поскольку поверхности с изломами встречаются в практике, то при использовании метода покоординатного спуска необходимо убедиться, что задача не имеет этого недостатка.

Для гладких функций при удачно выбранном начальном приближении (в некоторой окрестности минимума) процесс сходится к минимуму.

Рассмотрим решение конкретной задачи. Найдём точку минимума для функции $z = 3x^2 - 2xy + 4y^2 + x - y$.

Фиксируем $y = y_0$:

Находим производную по x : $\frac{\partial z}{\partial x} = 6x - 2y_0 + 1$

Приравняв производную к нулю, находим минимальный x : $x = \frac{2y_0 - 1}{6}$

Теперь находим y . Фиксируем $x = x_0$

$$z(x, y_0) = 3x_0^2 - 2x_0y + 4y^2 + x_0 - y$$

$$\frac{\partial z}{\partial y} = -2x_0 + 8y - 1$$

$$y = \frac{2x_0 + 1}{8}$$

Теперь будем чередовать минимизацию по переменным до сходимости:

Возьмём за начальное приближение $x_0 = 0, y_0 = 0$

i	x_i	y_i	z	усл ост
0	0	0	0	нет
1	-0,16667	0,125	-0,10417	нет
2	-0,125	0,083333	-0,11285	нет
3	-0,13889	0,09375	-0,11357	нет

4	-0,13542	0,090278	-0,11363	нет
5	-0,13657	0,091146	-0,11364	нет
6	-0,13628	0,090856	-0,11364	да

Т.к. в следующих итерациях изменения x и y становятся меньше, то результат сходится к точке $(-0,13628; 0,090856)$, отсюда следует что минимум функции равен: $Z = -0,11364$.

Реализуем алгоритм с помощью программного кода на языке программирования python:

```
import math
def coordinate_descent(func, x0=0, y0=0, epsilon=0.0001, step=1e-5,
max_iter=1000):
    x, y = x0, y0
    for iteration in range(max_iter):
        x_old, y_old = x, y

        def func_x(x_val):
            return func(x_val, y)
        x = minimize_brute_force(func_x, x, step)

        def func_y(y_val):
            return func(x, y_val)
        y = minimize_brute_force(func_y, y, step)

        if abs(x - x_old) < epsilon and abs(y - y_old) < epsilon:
            break

    return x, y, func(x, y)

def minimize_brute_force(func_1d, start, step, search_range=1.0):

    x_min = start
    min_val = func_1d(start)

    for offset in range(-int(search_range / step), int(search_range /
step) + 1):
        x_candidate = start + offset * step
        candidate_val = func_1d(x_candidate)
        if candidate_val < min_val:
            min_val = candidate_val
            x_min = x_candidate

    return x_min

def Z(x, y):
    return 3 * math.pow(x, 2) - 2 * x * y + 4 * math.pow(y, 2) + x - y

x_min, y_min, z_min = coordinate_descent(Z, x0=0, y0=0, epsilon=0.0001,
step=1e-5)
print(f"Минимум достигается в точке: x = {x_min:.5f}, y = {y_min:.5f}")
```

```
print(f"Значение функции в точке минимума: Z = {z_min:.10f}")
```

Результат работы программы:

```
Минимум достигается в точке: x = -0.13637, y = 0.09091  
Значение функции в точке минимума: Z = -0.1136363635  
PS C:\Users\dog23\OneDrive\Рабочий стол\sandboxes\test_code_pyth>
```

Алгоритм координатного спуска популярен ввиду простоты реализации, но в то же время свойство побуждает исследователей игнорировать их в пользу более сложных методов, которые устраняют очевидные недостатки этого метода.

Литература

1. Бахвалов, Н.С. Численные методы/ Н.С.Бахвалов, Н.П.Жидков, Г.М.Кобельков. – 3-е изд., доп. и перераб. – М.: БИНОМ. Лаборатория знаний, 2004. Литература.
2. Бигаева, Л. А. Курс лекций по численным методам : Учебное пособие для студентов физико-математического факультета / Л. А. Бигаева, И. И. Латыпов. – Бирск : Башкирский государственный университет, 2018. – 138 с. – EDN UMYXHX.
3. Зайцев В. В. Численные методы для физиков. Нелинейные уравнения и оптимизация: учебное пособие. – Самара, 2005г. – 86с.