Университет ИТМО

Факультет программной инженерии и компьютерной техники

# Лабораторная работа №5

# по «Методы Оптимизации»

Выполнил:

Студент группы P3207
Разинкин А.В.

Преподаватели:

Селина Е.Г.

г. Санкт-Петербург
2024

## Оглавление

## Условие

Решить задачу безусловной минимизации функции двух переменных
$f(x_1, x_2) = 4x_1^2 + 3x_2^2 + 16x_1 - 4x_2$ двумя градиентными методами:

1. Методом градиентного спуска
2. Методом наискорейшего спуска

Реализовать программу на одном из языков программирования и выполнить первые три итерации вручную.

## Первые три итерации методом градиентного спуска

В качестве начального приближения возьмем $(x_{10}, x_{20}) = (0, 0)$

$$grad(f) = \left(\frac{\delta f}{\delta x_1}; \frac{\delta f}{\delta x_2}\right) = (8x_1 + 16;\ 6x_2 - 4)$$

**<u>Итерация 1</u>**:

Посчитаем следующее приближение:

$$x_{11} = x_{10} - \lambda \frac{\delta f}{\delta x_1}$$

$$x_{21} = x_{20} - \lambda \frac{\delta f}{\delta x_2}$$

Подставим в исходную функцию:

$$f(x_{11}, x_{21}) = 4(x_{10} - \lambda \frac{\delta f}{\delta x_1})^2 + 3(x_{20} - \lambda \frac{\delta f}{\delta x_2})^2 + 16(x_{10} - \lambda \frac{\delta f}{\delta x_1}) - 4(x_{20} - \lambda \frac{\delta f}{\delta x_2})$$

Возьмем $\lambda = 0.1$:

$$x_{11} = 0 - 0.1(8 \cdot 0 + 16) = -1.6$$

$$x_{21} = 0 - 0.1(6 \cdot 0 - 4) = 0.4$$

$$f(x_{10}, x_{20}) = 0;\ f(x_{11}, x_{21}) = -16.48$$

$$f(x_{10}, x_{20}) > \ f(x_{11}, x_{21}) => \text{переход к следующей итерации}$$

**<u>Итерация 2</u>**:

Посчитаем следующее приближение:

$$x_{12} = x_{11} - \lambda \frac{\delta f}{\delta x_1}$$

$$x_{22} = x_{21} - \lambda \frac{\delta f}{\delta x_2}$$

Возьмем $\lambda = 0.1$:

$$x_{12} = 0 - 0.1(8 \cdot (-1.6) + 16) = -1.92$$

$$x_{22} = 0 - 0.1(6 \cdot 0.4 - 4) = 0.56$$

$$f(x_{11}, x_{21}) = -16.48;\ f(x_{12}, x_{22}) = -17.2736$$

$$f(x_{11}, x_{21}) > \ f(x_{12}, x_{22}) => \text{переход к следующей итерации}$$

**<u>Итерация 3</u>**:

Посчитаем следующее приближение:

$$x_{13} = x_{12} - \lambda \frac{\delta f}{\delta x_1}$$

$$x_{23} = x_{22} - \lambda \frac{\delta f}{\delta x_2}$$

Возьмем $\lambda = 0.1$:

$$x_{13} = 0 - 0.1(8 \cdot (-1.92) + 16) = -1.984$$

$$x_{23} = 0 - 0.1(6 \cdot 0.56 - 4) = 0.624$$
$$f(x_{12}, x_{22}) = -17.2736;\ f(x_{13}, x_{23}) = -17.326848$$
$$f(x_{12}, x_{22}) > f(x_{13}, x_{23}) => \text{переход к следующей итерации}$$

## Первые три итерации методом наискорейшего спуска

В качестве начального приближения возьмем $(x_{10}, x_{20}) = (0, 0)$

$$grad(f) = \left(\frac{\delta f}{\delta x_1}; \frac{\delta f}{\delta x_2}\right) = (8x_1 + 16;\ 6x_2 - 4)$$

**<u>Итерация 1</u>**:

Посчитаем следующее приближение:

$$x_{11} = x_{10} - \lambda \frac{\delta f}{\delta x_1}$$
$$x_{21} = x_{20} - \lambda \frac{\delta f}{\delta x_2}$$

Подставим в исходную функцию:

$$f(x_{11}, x_{21}) = 4(x_{10} - \lambda \frac{\delta f}{\delta x_1})^2 + 3(x_{20} - \lambda \frac{\delta f}{\delta x_2})^2 + 16(x_{10} - \lambda \frac{\delta f}{\delta x_1}) - 4(x_{20} - \lambda \frac{\delta f}{\delta x_2})$$
$$f'_\lambda(x_{11}, x_{21}) = -8x_{10}\frac{\delta f}{\delta x_1} + 8\lambda\left(\frac{\delta f}{\delta x_1}\right)^2 - 6x_{20}\frac{\delta f}{\delta x_2} + 6\lambda\left(\frac{\delta f}{\delta x_2}\right)^2 - 16\frac{\delta f}{\delta x_1} + 4\frac{\delta f}{\delta x_2}$$
$$f'_\lambda(x_{11}, x_{21}) = 0$$
$$\lambda = \frac{8x_{10}\frac{\delta f}{\delta x_1} + 6x_{20}\frac{\delta f}{\delta x_2} + 16\frac{\delta f}{\delta x_1} - 4\frac{\delta f}{\delta x_2}}{8\left(\frac{\delta f}{\delta x_1}\right)^2 + 6\left(\frac{\delta f}{\delta x_2}\right)^2}$$

$$\lambda = 0.12686567164179105$$
$$x_{11} = -2.029850746268657$$
$$x_{21} = 0.5074626865671642$$

**<u>Итерация 2</u>**:

Посчитаем следующее приближение:

$$x_{12} = x_{11} - \lambda \frac{\delta f}{\delta x_1}$$
$$x_{22} = x_{21} - \lambda \frac{\delta f}{\delta x_2}$$

Подставим в исходную функцию:

$$f(x_{12}, x_{22}) = 4(x_{11} - \lambda \frac{\delta f}{\delta x_1})^2 + 3(x_{21} - \lambda \frac{\delta f}{\delta x_2})^2 + 16(x_{11} - \lambda \frac{\delta f}{\delta x_1}) - 4(x_{21} - \lambda \frac{\delta f}{\delta x_2})$$

$$f'_\lambda(x_{12}, x_{22}) = -8x_{11}\frac{\delta f}{\delta x_1} + 8\lambda\left(\frac{\delta f}{\delta x_1}\right)^2 - 6x_{21}\frac{\delta f}{\delta x_2} + 6\lambda\left(\frac{\delta f}{\delta x_2}\right)^2 - 16\frac{\delta f}{\delta x_1} + 4\frac{\delta f}{\delta x_2}$$

$$f'_\lambda(x_{12}, x_{22}) = 0$$

$$\lambda = \frac{8x_{11}\frac{\delta f}{\delta x_1} + 6x_{21}\frac{\delta f}{\delta x_2} + 16\frac{\delta f}{\delta x_1} - 4\frac{\delta f}{\delta x_2}}{8\left(\frac{\delta f}{\delta x_1}\right)^2 + 6\left(\frac{\delta f}{\delta x_2}\right)^2}$$

$$\lambda = 0.1634615384615385$$

$$x_{11} = -1.9908151549942594$$

$$x_{21} = 0.6636050516647533$$

**<u>Итерация 3</u>**:

Посчитаем следующее приближение:

$$x_{13} = x_{12} - \lambda\frac{\delta f}{\delta x_1}$$

$$x_{23} = x_{22} - \lambda\frac{\delta f}{\delta x_2}$$

Подставим в исходную функцию:

$$f(x_{13}, x_{23}) = 4(x_{12} - \lambda\frac{\delta f}{\delta x_1})^2 + 3(x_{22} - \lambda\frac{\delta f}{\delta x_2})^2 + 16(x_{12} - \lambda\frac{\delta f}{\delta x_1}) - 4(x_{22} - \lambda\frac{\delta f}{\delta x_2})$$

$$f'_\lambda(x_{13}, x_{23}) = -8x_{12}\frac{\delta f}{\delta x_1} + 8\lambda\left(\frac{\delta f}{\delta x_1}\right)^2 - 6x_{22}\frac{\delta f}{\delta x_2} + 6\lambda\left(\frac{\delta f}{\delta x_2}\right)^2 - 16\frac{\delta f}{\delta x_1} + 4\frac{\delta f}{\delta x_2}$$

$$f'_\lambda(x_{13}, x_{23}) = 0$$

$$\lambda = \frac{8x_{12}\frac{\delta f}{\delta x_1} + 6x_{22}\frac{\delta f}{\delta x_2} + 16\frac{\delta f}{\delta x_1} - 4\frac{\delta f}{\delta x_2}}{8\left(\frac{\delta f}{\delta x_1}\right)^2 + 6\left(\frac{\delta f}{\delta x_2}\right)^2}$$

$$\lambda = 0.12686567164178675$$

$$x_{11} = -2.0001370872388913$$

$$x_{21} = 0.6659355347259112$$

## Листинг программы

Метод градиентного спуска:

```
def f(x1, x2):
    return 4 * x1 ** 2 + 3 * x2 ** 2 + 16 * x1 - 4 * x2


def gradient(x1, x2):
    return [8 * x1 + 16, 6 * x2 - 4]
```

```
def gradient_module(x1, x2):
    dx1, dx2 = gradient(x1, x2)
    return (dx1 ** 2 + dx2 ** 2) ** 0.5


x1 = 0
x2 = 0
step_value = 0.1
accuracy = 0.05


while True:
    f_current = f(x1, x2)
    if gradient_module(x1, x2) <= accuracy:
        break
    step = step_value
    while True:
        dx1, dx2 = gradient(x1, x2)
        x1_new = x1 - step * dx1
        x2_new = x2 - step * dx2
        f_new = f(x1_new, x2_new)
        print(f_current, x1_new, x2_new, f_new, step)
        if f_new < f_current:
            break
        step /= 2
    x1 = x1_new
    x2 = x2_new

print(x1, x2)
```

Метод наискорейшего спуска:

```
def f(x1, x2):
    return 4 * x1 ** 2 + 3 * x2 ** 2 + 16 * x1 - 4 * x2


def gradient(x1, x2):
    return [8 * x1 + 16, 6 * x2 - 4]


def gradient_module(x1, x2):
    dx1, dx2 = gradient(x1, x2)
    return (dx1 ** 2 + dx2 ** 2) ** 0.5


# f = 4(x1 - hdx1)^2 + 3(x2 - hdx2)^2 + 16(x1 - hdx1) - 4(x2 - hdx2)
# f' = -8dx1(x1 - hdx1) - 6dx2(x2 - hdx2) - 16dx1 + 4dx2 = -8x1dx1 +
8h(dx1)^2 - 6x2dx2 + 6h(dx2)^2 - 16dx1 + 4dx2
# f' = 0 => h = (8x1dx1 + 6x2dx2 + 16dx1 - 4dx2) / (8(dx1)^2 + 6(dx2)^2)
def calculate_step(x1, x2):
    dx1, dx2 = gradient(x1, x2)
    return (8 * x1 * dx1 + 6 * x2 * dx2 + 16 * dx1 - 4 * dx2) / (8 * dx1 ** 2
+ 6 * dx2 ** 2)


x1 = 0
x2 = 0
```

```
accuracy = 0.05


while True:
    if gradient_module(x1, x2) <= accuracy:
        break
    step = calculate_step(x1, x2)

    dx1, dx2 = gradient(x1, x2)
    x1 = x1 - step * dx1
    x2 = x2 - step * dx2


print(x1, x2)
```

## Вывод

В ходе выполнения данной лабораторной работы я ознакомился с градиентными методами: градиентного и наискорейшего спусков.