Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет Программной инженерии и компьютерной техники

Отчёт по лабораторной работе №4

Методы оптимизации

Вариант № 9

Выполнил: студент группы P3214

Силинцев В.В.

Преподаватель: Селина Е.Г.

Содержание

[Цель работы 3](#__RefHeading___Toc520_1614521587)

[Задание 4](#__RefHeading___Toc522_1614521587)

[Ручные расчеты 5](#__RefHeading___Toc522_1614521587_%25D0%2)

[Метод покоординатного спуска 5](#__RefHeading___Toc1093_2157254880)

[Метод градиентного спуска 8](#__RefHeading___Toc1093_2157254880_%25D0%)

[Метод наискорейшего спуска 10](#__RefHeading___Toc1093_2157254880_%25D01)

[Код программы 13](#__RefHeading___Toc524_1614521587_%25D0%2)

[Результат работы программы 14](#__RefHeading___Toc524_1614521587_%25D0%1)

[Заключение 15](#__RefHeading___Toc524_1614521587)

# **Цель работы**

Изучить методы покоординатного спуска, градиентного спуска и наискорейшего спуска. Реализовать их программно.

# **Задание**

Найти экстремум функции методами покоординатного спуска, градиентного спуска и наискорейшего спуска. Три итерации каждого метода выполнить вручную + написать программу на одном из языков программирования. .

* Функция для нахождения экстремума: .
* Начальная точка: .
* Необходимая точность вычислений: .

# **Ручные расчеты**

## Метод покоординатного спуска

Алгоритм поиска экстремума:

* Выбираем начальную точку для поиска .
* Фиксируем значения и проводим минимизацию по переменной . Получаем .
* Аналогично повторяем предыдущий пункт для всех , используя уже полученные значения.
* Далее в качестве начальной точки берём полученную и переходим к пункту 1.
* Продолжаем данный алгоритм, пока не будет выполнено или .

Поиск методом покоординатного спуска (3 итерации):

Первая итерация:

1. Начальная точка .
2. Фиксируем . Тогда .
3. Для поиска минимума найдем производную . Тогда .
4. Находим значения и . Так как , то фиксируем значение . Тогда .
5. Для поиска минимума найдем производную . Тогда .
6. Находим значения и . Так как , то фиксируем значение .

Вторая итерация:

1. Начальная точка .
2. Фиксируем . Тогда .
3. Для поиска минимума найдем производную . Тогда .
4. Находим значения и . Так как , то фиксируем значение . Тогда .
5. Для поиска минимума найдем производную . Тогда .
6. Находим значения и . Так как , то фиксируем значение .

Третья итерация:

1. Начальная точка .
2. Фиксируем . Тогда .
3. Для поиска минимума найдем производную . Тогда .
4. Находим значения и . Так как , то фиксируем значение . Тогда .
5. Для поиска минимума найдем производную . Тогда .
6. Находим значения и . Так как , то фиксируем значение .

Итоговые значения: , , .

## Метод градиентного спуска

Алгоритм поиска экстремума:

* Выбираем начальную точку для поиска , .
* Находим градиент функции в выбранной точке.
* Вычисляем .
* Если , то уменьшаем шаг .
* Переходим к пункту 1.
* Продолжаем данный алгоритм, пока не будет выполнено .

Поиск методом градиентного спуска (3 итерации):

Первая итерация:

1. Начальная точка . В качестве шага возьмем .
2. Находим градиент функции . Найдем производные: , . Тогда . Найдем градиент функции в выбранной точке: .
3. Вычислим .
4. , , .

Вторая итерация:

1. Начальная точка . В качестве шага возьмем .
2. Найдем градиент функции в выбранной точке: .
3. Вычислим .
4. , , .

Третья итерация:

1. Начальная точка . В качестве шага возьмем .
2. Найдем градиент функции в выбранной точке: .
3. Вычислим .
4. , , .

Итоговые значения: , , .

## Метод наискорейшего спуска

Алгоритм поиска экстремума:

* Выбираем начальную точку для поиска , .
* Находим градиент функции в выбранной точке.
* Вычисляем .
* Находим такое , при котором минимальна.
* Переходим к пункту 1.
* Продолжаем данный алгоритм, пока не будет выполнено .

Поиск методом наискорейшего спуска (3 итерации):

Первая итерация:

1. Начальная точка .
2. Находим градиент функции . Найдем производные: , . Тогда . Найдем градиент функции в выбранной точке: .
3. Вычислим .
4. Найдем минимум функции . Найдем производную: . Тогда . Найдем вторую производную: , следовательно – точка минимума.

Вторая итерация:

1. Начальная точка .
2. Найдем градиент функции в выбранной точке: .
3. Вычислим .
4. Найдем минимум функции . Найдем производную: . Тогда . Найдем вторую производную: , следовательно – точка минимума.

Третья итерация:

1. Начальная точка .
2. Найдем градиент функции в выбранной точке: .
3. Вычислим .
4. Найдем минимум функции . Найдем производную: . Тогда . Найдем вторую производную: , следовательно – точка минимума.

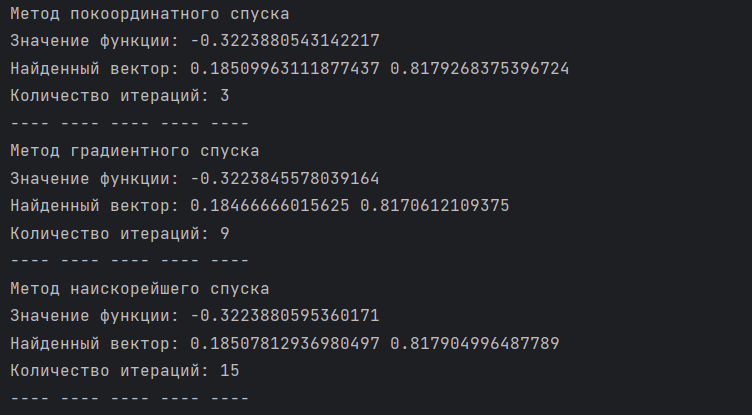
Итоговые значения: , , .

# **Код программы**

Полный исходный код приложения:

[https://github.com/vvlaads/vvlaads/tree/master/Optimization%20methods/Lab4](https://github.com/vvlaads/vvlaads/tree/master/Optimization methods/Lab4)

# **Результат работы программы**

Рисунок 1: Результат работы программы

# **Заключение**

В ходе этой работы я изучил методы покоординатного спуска, градиентного спуска и наискорейшего спуска и реализовал их программно..