Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий
институт

Кафедра «Информатика»
кафедра

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ № 10

Метод покоординатного спуска
Тема

Преподаватель			В. В. Тынченко
		Подпись, дата	Инициалы, Фамилия
Студент	КИ19-17/1Б, №031939174		А. К. Никитин
	Номер группы, зачетной книжки	Подпись, дата	Инициалы, Фамилия

Красноярск 2021

1 Постановка задачи

Разработать программу, реализующую метод покоординатного спуска.

Найти безусловный экстремум функции, выбранной в соответствии с заданием, с использованием разработанной программы.

Функция: $(x_2 + x_1 - 1)^2 + 2(x_1 - 2)^2 \to min$

2 Описание метода

Стратегия решения задачи состоит в построении последовательности точек {x k}, k = 0,1,..., таких, что $f(x_{k+1}) < f(x_k),$ k = 0,1,.... Точки последовательности {x k} вычисляются по правилу

$$x_{k+1} = x_k - \alpha_k f'(x_k), \quad k = 0,1,\ldots$$

где точка x0 задается пользователем; величина шага α_k определяется для каждого значения k из условия

$$\varphi(\alpha_k) = f(x_k - \alpha_k f'(x_k)) \to \min_{\alpha_k}$$

Решение задачи (3) может осуществляться с использованием необходимого условия минимума $\frac{d\varphi}{d\alpha_k} = 0$ с последующей проверкой достаточного условия минимума $\frac{d^2\varphi}{d{\alpha_k}^2} > 0$. Такой путь может быть использован либо при достаточно простой минимизируемой функции $\varphi(\alpha_k)$, либо при предварительной аппроксимации достаточно сложной функции $\varphi(\alpha_k) = f(x_k - \alpha_k f'(x_k))$ полиномом P (α_k) (как правило, второй или третьей степени), и тогда условие $\frac{d\varphi}{d\alpha_k} = 0$ замещается условием $\frac{dP}{d\alpha_k} = 0$, а условие $\frac{d^2\varphi}{d{\alpha_k}^2} > 0$ условием $\frac{d^2P}{d{\alpha_k}^2} > 0$.

Построение последовательности $\{x_k\}$, $k = 0,1,...$, заканчивается в точке x_k, для которой $\|f'(x_k)\| < \varepsilon_1$, где ε_1 - заданное число, или, если $k \geq M$, M - предельное число итераций, или при двукратном одновременном выполнении неравенств $\|x_{k+1} - x_k\| < \varepsilon_2$, $|f(x_{k+1}) - f(x_k)| < \varepsilon_2$, где ε_2 - малое положительное число. Вопрос о том, может ли точка x k рассматриваться как найденное приближение искомой точки локального минимума x*, решается путем дополнительного исследования.

3 Исходные тексты программ

На листинге 1 представлен код программы, реализующий задание.

Листинг 1 – Метод градиентного спуска с постоянным шагом

```
from const import func, asses_func
import numpy as np

def constant_gradient_descent_method(f, x0, epsilon=0.1, epsilon1=0.1,
epsilon2=0.1, initial_step=0.5, M=100):
    x_list = [np.array(x0).astype(float)]
    k = 0
    while k < M:
        step = initial_step

        gradient = f.gradient_value(x_list[k])
        if np.linalg.norm(gradient) < epsilon1:
            return x_list[k], k

        x_list.append(x_list[k] - step * gradient)
        while not (f.calc(x_list[k + 1]) - f.calc(x_list[k]) < 0 or
                   f.calc(x_list[k + 1]) - f.calc(x_list[k]) < -epsilon *
gradient.dot(gradient)):
            step /= 2
            x_list[k + 1] = x_list[k] - step * gradient

        if np.linalg.norm(x_list[k + 1] - x_list[k]) < epsilon2 \
                and abs(f.calc(x_list[k + 1]) - f.calc(x_list[k])) < epsilon2 \
                and len(x_list) > 2 \
                and np.linalg.norm(x_list[k] - x_list[k - 1]) < epsilon2 \
```

```
                and abs(f.calc(x_list[k]) - f.calc(x_list[k - 1])) < epsilon2:
            return x_list[k + 1], k

        k += 1

    return x_list[-1], k


if __name__ == '__main__':
    print(constant_gradient_descent_method(asses_func, [-10, 10], epsilon1=0.1,
epsilon2=0.15, initial_step=0.5, M=20))
```

4 Исследование влияния параметров метода на точность и скорость нахождения решения

В качестве гиперпараметров по умолчанию использовались следующие значения:

а) x0 = (-10, 10);

б) $\varepsilon = 0.1$;

в) $\varepsilon_1 = 0.1$;

г) $\varepsilon_2 = 0.1$;

д) $t_0 = 0.5$;

е) M = 100.

Из рисунка 1 что параметр ε при данных гиперпараметрах никак не влияет на результат.

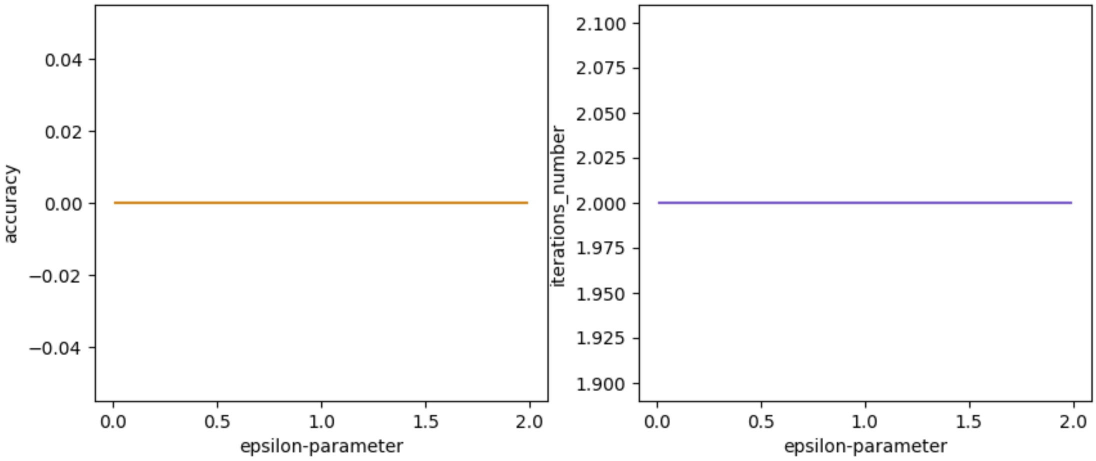

Рисунок 1 – Влияние параметра ε на точность и производительность

Из рисунка 2 можно заключить, что, в целом, чем меньше параметр ε_1, тем точнее решение, однако тем больше шагов необходимо алгоритму для нахождения ответа.

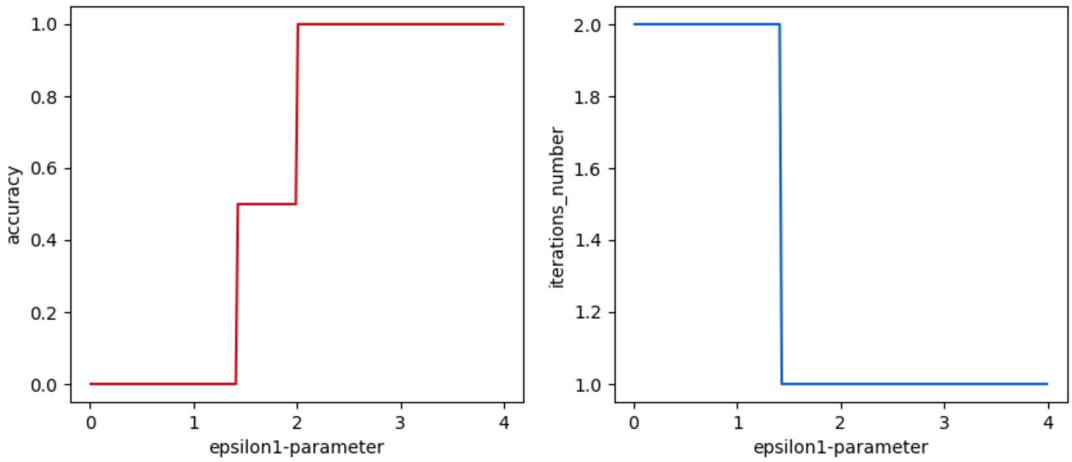

Рисунок 2 - Влияние параметра ε_1 на точность и производительность

Из рисунка 3 понятно, что параметр ε_2 никак не влияет на результаты работы алгоритма.

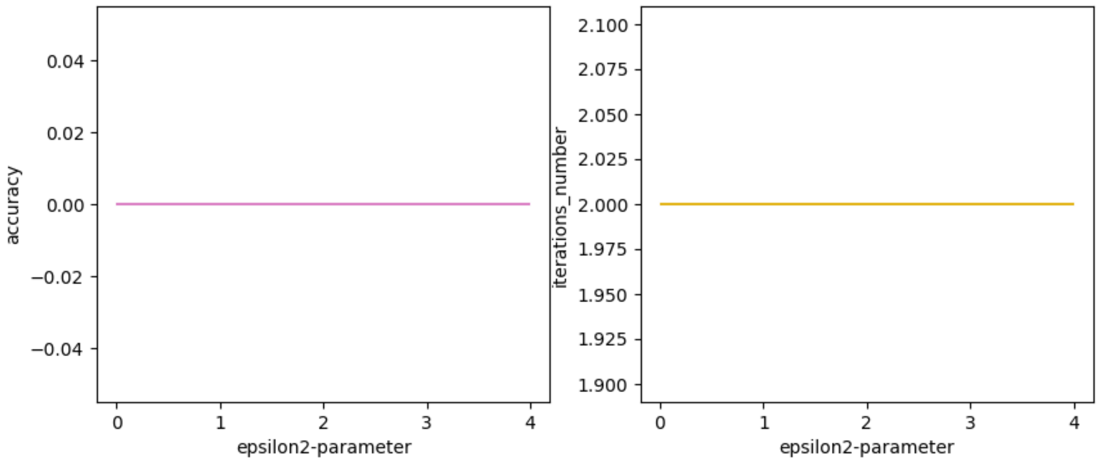

Рисунок 3 - Влияние параметра ε_2 на точность и производительность

Из рисунка 4 можно заключить, что параметр начального шага лучше не делать слишком небольшим. Однако стоит учитывать, что, хоть количество итераций алгоритма не увеличилось, время выполнения программы значительно возрастает с увеличением шага.

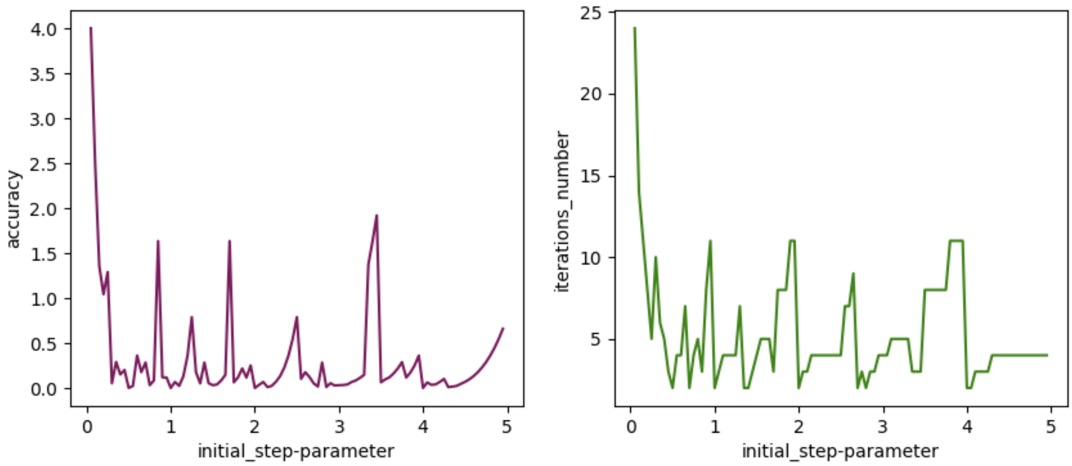

Рисунок 4 – Влияние параметра начального шага на точность и производительность

Из рисунка 5 можно заключить, что параметр максимального количества шагов лучше не делать слишком небольшим, и что чем он больше, тем точнее, но менее производительнее, программа.

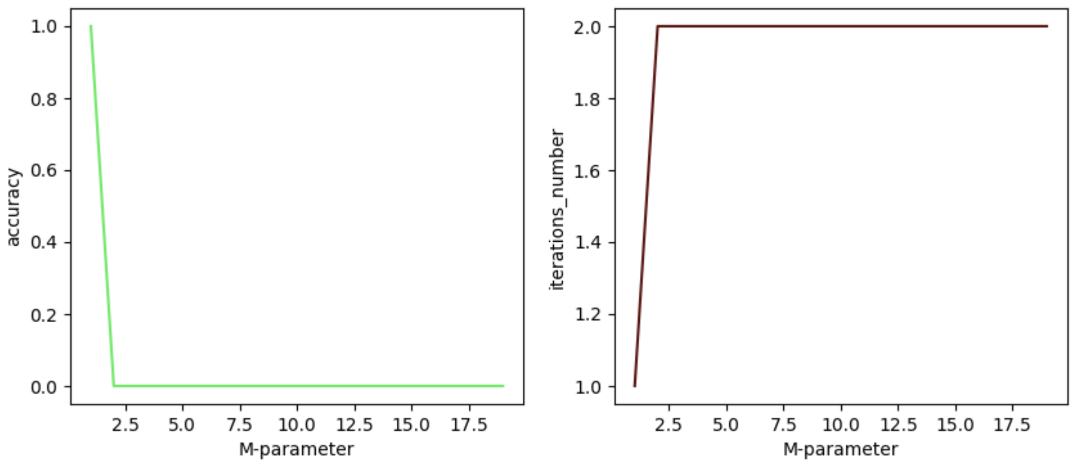

Рисунок 5 - Влияние параметра максимального количества шагов на точность и производительность

5 Вывод

В результате данной работы был реализован и проанализирован метод координатного спуска для поиска локального минимума многомерной функции. Также были проанализированы гиперпараметры метода и их влияние на работу функции.