**Федеральное государственное образовательное**

**бюджетное учреждение**

**высшего образования**

**«ФИНАНСОВЫЙ УНИВЕРСИТЕТ**

**ПРИ ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ**

**ФЕДЕРАЦИИ»**

**(Финансовый университет)**

**Факультет**

**информационных технологий и анализа больших данных**

**Направление «Прикладная математика и информатика»**

**Домашнее задание № 5**

**«Метод внутренней точки»**

Студенты группы ПМ19-3

Караваев Артем Евгеньевич

Лазаренко Владлена Владимировна

Минаков Артем Дмитриевич

Пластун Екатерина Сергеевна

Голомысов Даниил Владиславович

Руководитель:

Аксенов Дмитрий Андреевич

**Москва 2022**

Оглавление.

1. Математическая модель (постановка задачи)
2. Алгоритмы
   1. Алгоритм 1
      1. Описание входных данных
      2. Описание алгоритма решения
      3. Описание выходных данных
   2. Алгоритм 2
      1. Описание входных данных
      2. Описание алгоритма решения
      3. Описание выходных данных
   3. Алгоритм 3
      1. Описание входных данных
      2. Описание алгоритма решения
      3. Описание выходных данных
3. Варианты использования системы
   1. ВИ 1
   2. ВИ 2
4. Архитектура решения
   1. Функции считывания информации
   2. Функции обработки информации
   3. Функции вывода информации
5. Тестирование
6. Выводы и заключение
7. **Постановка задачи (физическая модель)**

К нам пришел заказчик с новым проектом по созданию ресторана правильного питания. Он хочет создать меню, в состав которого будет входить идеальные пропорции белка, протеина и клетчатки. Наша задача будет состоять в том, чтобы помочь ему минимизировать затраты на открытие ресторана.

1. **Математическая модель и алгоритмы**

Он знает, что при открытии ресторана ему нужно сделать закуп смеси с нужными питательными веществами, а также он понимает, что для наиболее благоприятного для здоровья эффекта, нужно в блюде: не менее 19 единиц протеина a1, 18 единиц белка a2 и не менее 21 единиц клетчатки a3. Для первоначального открытия он хочет закупить два вида смеси.

Содержание количества единиц питательных веществ в одном килограмме каждого вида смеси и стоимость приведены в таблице ниже:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Кол-во питательных в-в в 1 кг. | А1 | А2 | А3 | **Цена 1 кг ден. ед.** |
| Смесь 1 | 5 | 2 | 2 | 10 |
| Смесь 2 | 2 | 4 | 6 | 15 |

Пусть x1 – это количество единиц питательных веществ смеси 1, а x2 – смеси 2 вида. Стоимость 1 кг. x1 – 10 денежных ед., x2 – 15. Тогда суммарная стоимость составляет: 10x1 + 15x2.

Полученные ограничения:

x1, x2>=0

5x1 +2x2 >=19

2x2 + 4x2>=18

2x1+6x2>=21

Необходимо найти минимум функции f(x1,x2) при заданных ограничениях типа неравенства, а также найти значения x1, x2.

* 1. **Алгоритм 1**

**Решение задачи оптимизации для функции с ограничениями типа равенства методом Ньютона** (способ решения двойственной задачи).

* + 1. **Описание входных данных**

**Обязательные параметры:**

а) Функция в явном виде;

б) Ограничения типа равенства в явном виде;

в) Координаты начальной точки (начального приближения);

* + 1. **Описание алгоритма решения**

1. Ввести функцию в явном виде (пример: ), ограничения типа равенств в явном виде (, например, ), координаты начальной точки .

2. Составить функцию Лагранжа:

), где  - множители Лагранжа.

3. С помощью уже реализованного раннее метода Ньютона находятся новые значения неизвестных x и lambda.

* + 1. **Описание выходных данных**

а) координаты точки экстремума;

* 1. **Алгоритм 2**

**Решение задачи оптимизации для функции с ограничениями типа неравенства методом логарифмических барьеров (прямой метод внутренней точки).**

* + 1. **Описание входных данных**

**Обязательные параметры:**

а) Функция в явном виде;

б) Ограничения типа равенств и неравенств в явном виде;

в) Координаты начальной точки (начального приближения);

**Необязательные параметры:**

а) точность оптимизации (по умолчанию 10^(-6));

* + 1. **Описание алгоритма решения**

1. Ввести функцию в явном виде (пример: ), ограничения типа неравенств в явном виде, координаты начальной точки .

2. Задаем параметры τ и v. (пример: τ =1, v =10)

3. Вводим новую функцию:

Изображение выглядит как текст, часы, датчик

Автоматически созданное описание

4. С помощью метода Ньютона находим новую точку.

5. Если ) < заданной точности, конец цикла.

Иначе: τ= τ \*v. Повтор пунктов 3–5 с новым τ.

**Описание выходных данных**

а) координаты точки экстремума;

* 1. **Алгоритм 3**

**Решение задачи оптимизации для функции с ограничениями типа неравенства прямо-двойственным методом внутренней точки.**

* + 1. **Описание входных данных**

**Обязательные параметры:**

а) Функция в явном виде;

б) Ограничения типа равенств и неравенств в явном виде;

в) Координаты начальной точки (начального приближения);

**Необязательные параметры:**

а) точность оптимизации;

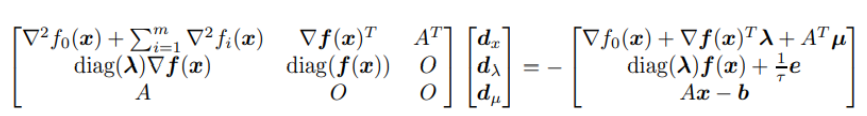
* + 1. **Описание алгоритма решения**

1. Ввести функцию в явном виде (пример: ), ограничения типа равенств в явном виде, координаты начальной точки , точность.
2. n – количество переменных; m – количество ограничений типа неравенства; p – количество ограничений типа равенство

Задается:

* положительный вектор лямбда;
* произвольный вектор mu
* τ, v

1. Решаем СЛАУ:



1. По правой матрице методом перебора всех возможных значений и при условии alpha> = 0, находим такое значение alpha, при котором норма матрицы минимальна.
2. Если для нового набора (x, λ, µ) выполнено условие  и , то алгоритм заканчивает работу;
3. Иначе:

τ =  и переход к шагу 3.

* + 1. **Описание выходных данных**

а) координаты точки экстремума;

1. **Варианты использования системы**

Предполагается, что пользователь будет предлагать программе на вход файл функцию для оптимизации, а также ограничения в виде равенств или неравенств. Функция сама будет производить необходимые вычисления и возвращать координаты точки Х и значение У в ней.

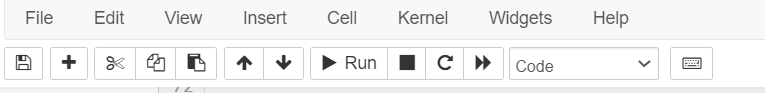
В соответствии с нужным алгоритмом оптимизации пользователь будет выбирать нужную ему функцию среди трех, так как они будут иметь говорящие названия.

* 1. **ВИ** 
     1. **Ввод данных.**
        1. **Ввод переменных Х и У.**

Вы можете присвоить функцию для оптимизации переменной example1, а ограничение переменной limit1, после чего нажмите «Run». Пример:

Изображение выглядит как текст

Автоматически созданное описание





В следующей ячейке присвойте x, y функцию с нужным Вам методом (Newton, logBarMethod, inequality). Важно: в скобках указать переменные, созданные выше example1, limit1, а также не забудьте ввести координаты начальной точки, как это показано ниже:







После ввода переменных пользователь получит результаты работы функции.

* + - 1. **Вывод полученных результатов.**

После запуска программы пользователь увидит следующее:

****

Первое значение – найденные координаты точки х.

Второе – найденные значение y в точке х.

1. **Архитектура решения**
   1. **Решение задачи оптимизации для функции с ограничениями типа равенства методом Ньютона (способ решения двойственной задачи).**

def Newton(func: str, equality: list, x0: tuple, tol=5):

"""

Parameters

----------

func : str

Функция оптимизации.

equality : list

Список заданных линейных ограничений.

x0 : tuple

Начальная точка.

tol : int, default=5

Кол-во цифр после запятой после округления.

Returns

----------

res['x'], res['fun']:

Найденная точка и значение функции в ней.

"""

* 1. **Решение задачи оптимизации для функции с ограничениями типа неравенства методом логарифмических барьеров (прямой метод внутренней точки).**

def logBarMethod(func: str, restrictions: list, start\_point: tuple = tuple(), accuracy:float = 10\*\*(-6), max\_steps: int=500):

'''

Решение задачи оптимизации для функции методом логарифмических барьеров.

Parameters

----------

func : str

Функция оптимизации.

equality : list

Список заданных линейных ограничений.

x0 : tuple

Начальная точка.

tol : int, default=5

Кол-во цифр после запятой после округления.

Returns

----------

np.array(next\_point), res\_new:

Найденная точка и значение функции в ней.

'''

* 1. **Решение задачи оптимизации для функции с ограничениями типа неравенства прямо-двойственным методом внутренней точки.**

def inequality(func, x0, us, a, b, tol=10\*\*-5):

"""

Parameters

----------

func : str

Функция для оптимизации.

x0 : list

Начальная точка.

us : list

Список заданных линейных ограничений.

Варианты ввода: '4\*x - 4 = 0' или '4\*x - 4 < 0'

a : list

Список коэффициентов для переменных в ограничении

типа равенства.

b : list

Список свободных членов для ограничения типа равенства.

tol : float, default=10\*\*-5

Критерий остановы.

Returns

----------

point, y:

Найденная точка и значение функции в ней.

"""

* 1. **Вспомогательные функции для решения.**

def get\_pain(first\_line, second\_line, third\_line, A):

"""

Вспомогательная функция для записи матрицы.

Parameters

----------

first\_line : list

Первая строка матрицы.

second\_line : list

Вторая строка.

third\_line : list

Третья строка.

A : list

Список коэффициентов для переменных в ограничении

типа равенства.

Returns

----------

pain: list

Матрица.

"""

def get\_first\_line(F0, Fi, Y, A, n, us\_n):

"""

Вычисляет первую строку для матрицы.

Returns

----------

first\_line: list

Первая строка матрицы.

"""

def get\_second\_line(diag\_lam, Fi, Y, A, m, n, p):

"""

Вычисляет вторую строку для матрицы.

Returns

----------

second\_line: list

Вторая строка матрицы.

"""

def search\_point(x0, lmbd, mu, var,

left\_pain, right\_pain,

n, m, p, Fi, func, tol, A):

"""

Выполняет поиск точки для заданных условий.

Returns

----------

point: np.array

Найденная точка.

"""

1. **Тестирование**

Таблица 1. Результаты тестирования алгоритма 1 на заданных функциях.

|  |  |  |
| --- | --- | --- |
| **Входные данные** | **Алгоритм 1** | **Скорость выполнения** |
| y=x1\*\*3+x2\*\*2+2\*x1\*x2  Limit = 2\*x1+x2-2=4 | x=[ 1.15472, -0.30945]  y =0.92080 | 0.026183366 |
| y=x1\*\*4+x2\*\*4-2\*\*x1-2\*x2\*\*2  Limit= x1\*\*2-x2\*\*2=7 | x=[ 0.82809  , 0.82809]  y = -2.20634 | 0.009831667 |
| у= -x1\*\*2+6\*x1+x2  Limit=2\*x1+3\*x2=24; 3\*x1+2\*x2=24 | x=[ 0, 0]  y = 0 | 0.005002737 |
| y=(x1-3)\*\*2+(x2-4)\*\*2  Limit=-3\*x1-2\*x2-6=1; 3\*x1+2\*x2+6=0 | x=[ 0, 0]  y = 25 | 0.009435415 |
| у=29\*x1\*\*2-54\*x1\*x2+26\*x2\*\*2  Limit= 3\*x1-4\*x2=1 | x=[ 0, 0]  y = 0 | 0.036577939 |

Таблица 2. Результаты тестирования алгоритмов 2 и 3 на заданных функциях с ограничениями типа неравенств.

|  |  |  |
| --- | --- | --- |
| **Входные данные** | **Алгоритм 2** | **Алгоритм 3** |
| y=x1\*\*3+x2\*\*2+2\*x1\*x2  Limit = 2\*x1+x2-2 <=4 | x=[111111/400000, -111111/200000]  y = 1371737997260631/64 | x=[ 0.25, 0.5]  y = 0.015625 |
| y=x1\*\*4+x2\*\*4-2\*\*x1-2\*x2\*\*2  Limit= x1\*\*2-x2\*\*2 >=7 | x=[ 0, 0]  y = -1 | - |
| у= -x1\*\*2+6\*x1+x2  Limit=2\*x1+3\*x2>=24; 3\*x1+2\*x2<=24 | x=[ 0, 0]  y = 0 | x=[ 9/4,-3.875]  y = 4.5625 |
| y=(x1-3)\*\*2+(x2-4)\*\*2  Limit=-3\*x1-2\*x2-6<=1; 3\*x1+2\*x2+6<=0 | x=[1250.23509339220, 1800.22339559479]  y = 4782013.86507113 | x=[ 1.08  , 2.72]  y = 5.3248 |
| у=29\*x1\*\*2-54\*x1\*x2+26\*x2\*\*2  Limit= 3\*x1-4\*x2<=1 | x=[-248.822837497512,-248.709977080431]  y = 61969.300133 | x=[ 0.6, 0.7]  y = 0.5 |

Таблица 3. Время выполнения в секундах.

|  |  |  |
| --- | --- | --- |
| **Входные данные** | **Алгоритм 2** | **Алгоритм 3** |
| y=x1\*\*3+x2\*\*2+2\*x1\*x2  Limit = 2\*x1+x2-2 <=4 | 0.0031132698 | 0.0080983638 |
| y=x1\*\*4+x2\*\*4-2\*\*x1-2\*x2\*\*2  Limit= x1\*\*2-x2\*\*2 >=7 | 0.0031132698 | - |
| у= -x1\*\*2+6\*x1+x2  Limit=2\*x1+3\*x2>=24; 3\*x1+2\*x2<=24 | 0.0047075752 | 0.0080983638 |
| y=(x1-3)\*\*2+(x2-4)\*\*2  Limit=-3\*x1-2\*x2-6<=1; 3\*x1+2\*x2+6<=0 | 7.219464778 | 0.019061326 |
| у=29\*x1\*\*2-54\*x1\*x2+26\*x2\*\*2  Limit= 3\*x1-4\*x2<=1 | 6.445163726 | 0.036577954 |

По результатам вычисления скорости, самый эффективный метод для использования – метод Ньютона с ограничениями типа равенств.

С ограничениями типа неравенств – быстрее работает прямо-двойственный метод внутренней точки.

1. **Заключение**

По результатам сравнительного анализа методов, оптимальный для заказчика, оказался метод логарифмических барьеров. Его результаты:



X = (0,0)

Y = 0

Так как при запуске прямо-двойственный метод внутренней точки выдал ошибку, а в метод Ньютона не можем внести ограничения типа неравенств.