

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ М. В. ЛОМОНОСОВА
ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И КИБЕРНЕТИКИ

**Компьютерный практикум по курсе
«ВВЕДЕНИЕ В ЧИСЛЕННЫЕ МЕТОДЫ»**

**ЗАДАНИЕ №1
Подвариант №1**

РЕШЕНИЕ СИСТЕМ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ
МЕТОДОМ ГАУССА И МЕТОДОМ ГАУССА С ВЫБОРОМ ГЛАВНОГО
ЭЛЕМЕНТА

ОТЧЕТ
о выполненном задании
студента 202 учебной группы факультета ВМК МГУ
Кузнецова Михаила Константиновича

Москва
2020

Содержание

Цель работы	2
Постановка задачи	2
Задачи практической работы	2
Метод Гаусса	3
Метод Гаусса с выбором главного элемента	4
Описание программы	4
Код программы	5
Тестирование программы	7
Выводы	9

Цель работы

Изучить классический метод Гаусса (а также модифицированный метод Гаусса), применяемый для решения системы линейных алгебраических уравнений.

Постановка задачи

Дана система уравнений $Ax = f$ порядка $n \times n$ с невырожденной матрицей A . Написать программу, решающую систему линейных алгебраических уравнений заданного пользователем размера (n - параметр программы) методом Гаусса и методом Гаусса с выбором главного элемента. Предусмотреть возможность задания элементов матрицы системы и ее правой части как во входном файле данных, так и путем задания специальных формул.

Задачи практической работы

1. Решить заданную СЛАУ методом Гаусса и методом Гаусса с выбором главного элемента;
2. Вычислить определитель матрицы $\det(A)$;
3. Вычислить обратную матрицу A^{-1} ;
4. Определить число обусловленности $\|A\| * \|A^{-1}\|$
5. Исследовать вопрос вычислительной устойчивости метода Гаусса (при больших значениях параметра n);
6. Правильность решения СЛАУ подтвердить системой тестов

Метод Гаусса

Метод Гаусса решения уравнения $Ax = f$ или $\sum_{j=1}^n a_{ij}x_j = f_i; i = 1, 2, \dots, n$ с невырожденной матрицей A , удобно разделить на два этапа. На первом этапе (прямой ход) система приводится к треугольному виду. На втором этапе (обратный ход) осуществляется последовательное отыскание неизвестных $x_1, \dots, x_n$ из этой треугольной системы.

1. (прямой ход) Будем считать, что элемент a_{11} , назовем его ведущим, отличен от 0 (если нет, поменяем местами 1 и i уравнения, где i такое, что $a_{ii} \neq 0$, которое всегда найдется, так как иначе $\det(A) = 0$). Разделим все члены первого уравнения на a_{11} . Из каждого следующего i -ого уравнения вычтем первое, умноженное на a_{i1} , после чего x_1 останется только в первом уравнении. Выделим из системы "укороченную" систему, содержащую $(n-1)$ уравнение со 2 до n -ого. Продолжая процесс исключения, после $(n-1)$ шага редуцируем исходную систему к виду

$$\begin{aligned}x_1 + c_{12}x_2 + c_{13}x_3 + \dots + c_{1n}x_n &= y_1 \\x_2 + c_{23}x_3 + \dots + c_{2n}x_n &= y_2 \\&\dots \\x_{n-1} + c_{n-1,n}x_n &= y_{n-1} \\x_n &= y_n\end{aligned}$$

или в матричной форме $Cx = y$, где C является верхней треугольной матрицей с единицами на главной диагонали:

$$C = \begin{bmatrix} 1 & c_{12} & c_{13} & \dots & c_{1n} \\ 0 & 1 & c_{23} & \dots & c_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}$$

2. (обратный ход) Последовательно определим неизвестные из системы в обратном порядке:

$$\begin{aligned}x_n &= y_n \\x_{n-1} &= y_{n-1} - c_{n-1,n}x_n \\x_{n-2} &= y_{n-2} - c_{n-2,n-1}x_{n-1} - c_{n-2,n}x_n \\x_1 &= y_1 - c_{12}x_2 - c_{13}x_3 - \dots - c_{1n}x_n\end{aligned}$$

Метод Гаусса с выбором главного элемента

Производятся те же действия, что и в методе Гаусса, но на каждом шаге берется не просто, элемент, отличный от нуля, а наибольший по модулю в строке. Потом переставляются соответствующие столбцы матрицы и переменные. В конце переменные обратно восстанавливаются.

Описание программы

- Вводится откуда брать данные: 1- с файла, 2- по уравнению
- Вводится какой метод использовать: 1- без главных компонент, 2- с гл.к.
- Вводится что получить: 1- $\det(A)$, 2- A_{-1}
- По окончании работы вводится 0 и выводятся решения

Код программы

```
import numpy as np

def produce_A_f_eq(n=50, m=15):
    A = np.zeros((n, n), dtype=np.float32)
    f = np.zeros((n, 1), dtype=np.float32)
    for i in np.arange(n):
        for j in np.arange(n):
            if i != j:
                A[i][j] = (i + j + 2) / (m + n)
            else:
                A[i][j] = n + m ** 2 + (j + 1) / m + (i + 1) / n
    for i in np.arange(n):
        f[i, 0] = m * n - (i + 1) ** 3
    Af = np.concatenate((A, f), axis=1)
    return A, f, Af

def produce_A_f_path(path):
    Af = np.loadtxt(path)
    return Af[:, :-1], Af[:, -1:], Af

def gg(Af):
    a = cp.deepcopy(Af)
    A_1 = np.identity(A.shape[0]);

    #прямой ход
    for bcl in range(a.shape[0]):
        si = 0
        for i in range(bcl, a.shape[0]):
            if (a[i][bcl] != 0):
                si = i
                break

        A_1[bcl, :], A_1[si, :] = cp.deepcopy(A_1[si, :]), cp.deepcopy(A_1[bcl, :])
        A_1[bcl, :] /= a[bcl][bcl]

        a[bcl, :], a[si, :] = cp.deepcopy(a[si, :]), cp.deepcopy(a[bcl, :])
        a[bcl, :] /= a[bcl][bcl]

        for i in range(bcl+1, a.shape[0]):
            A_1[i, :] -= a[i][bcl]*A_1[bcl, :]
            a[i, :] -= a[i][bcl]*a[bcl, :]

    #обратный ход
    for i in range(a.shape[0]-2, -1, -1):
        for j in range(i+1, a.shape[0]):
            A_1[i, :] -= a[i][j]*A_1[j, :]
            a[i, :] -= a[i][j]*a[j, :]

    X_gg = a[:, -1]
    return X_gg, A_1

def gm(Af):
    a = cp.deepcopy(Af)
    detA = 1
    ind_x = np.array(range(A.shape[0]))

    #прямой ход
    for bcl in range(a.shape[0]):
        si = bcl
        maxv = a[bcl][bcl]
        for i in range(bcl + 1, a.shape[0]):
            if (a[bcl][i] > maxv and a[bcl][i] != 0):
                maxv = a[bcl][i]
                si = i

        ind_x[bcl], ind_x[si] = cp.deepcopy(ind_x[si]), cp.deepcopy(ind_x[bcl])
        a[:, bcl], a[:, si] = cp.deepcopy(a[:, si]), cp.deepcopy(a[:, bcl])
```

```

        if si != bcl:
            detA = -detA
        detA *= a[bcl][bcl]
        a[bcl,:] /= a[bcl][bcl]

    for i in range(bcl+1, a.shape[0]):
        a[i,:] -= a[i][bcl]*a[bcl,:]

#обратный ход
for i in range(a.shape[0]-2, -1, -1):
    for j in range(i+1, a.shape[0]):
        a[i,:] -= a[i][j]*a[j,:]

X_gm = a[:,-1]
_X_gm = cp.deepcopy(X_gm)
for i in range(X_gm.shape[0]):
    _X_gm[ind_x[i]] = X_gm[i]
X_gm = _X_gm
return X_gm, detA

#start of the programm
detA = np.nan
A_1 = np.nan

inc = int(input())
if inc == 1:
    A, f, Af = produce_A_f_path(input())
elif inc == 2:
    A, f, Af = produce_A_f_eq(n=int(input()), m=int(input()))
inc = int(input())
if inc == 1:
    x, A_1 = gg(Af)
elif inc == 2:
    x, detA = gm(Af)
inc = int(input())
while inc != 0:
    if inc == 1:
        if detA == np.nan:
            _, detA = gm(Af)
        print(detA)
    elif inc == 2:
        if A_1 == np.nan:
            _, A_1 = gg(Af)
        print(A_1)
    inc = int(input())
print(x)

```

Тестирование программы

Приложение 1. Вариант 13

1. Матрица $A = \left[\begin{array}{cccc|c} 3 & -2 & 2 & -2 & 8 \\ 2 & -1 & 2 & 0 & 4 \\ 2 & 1 & 4 & 8 & -1 \\ 1 & 3 & -6 & 2 & 3 \end{array} \right]$

$$|A| = 24$$

$$\text{Обратная матрица } A^{-1} = \left[\begin{array}{cccc} -0.5 & 1.333333 & -0.166666 & 0.166666 \\ -5 & 8.666666 & -1.333333 & 0.333333 \\ -2 & 3.5 & -0.5 & 0 \\ 1.75 & -3.166666 & 0.583333 & -0.083333 \end{array} \right]$$

$$\text{Число обусловленности } \|A\| * \|A^{-1}\| = 233.3$$

Решение системы методом Гаусса: (2, -3, -1.5, 0.5)

Решение системы методом Гаусса: (2, -3, -1.5, 0.5)

Решение системы numru : (2, -3, -1.5, 0.5)

2. Матрица $A = \left[\begin{array}{cccc|c} 2 & 3 & 1 & 2 & 4 \\ 4 & 3 & 1 & 1 & 5 \\ 1 & -7 & -1 & -2 & 7 \\ 2 & 5 & 1 & 1 & 1 \end{array} \right]$

$$|A| = 1.999999$$

$$\text{Обратная матрица } A^{-1} = \left[\begin{array}{cccc} -1 & 2 & -1 & -2 \\ -1 & 1.5 & -1 & -1.5 \\ 8 & -14.5 & 9 & 16.5 \\ -1 & 3 & -2 & -4 \end{array} \right]$$

$$\text{Число обусловленности } \|A\| * \|A^{-1}\| = 432$$

Решение системы методом Гаусса: (-3, -5, 39, -7)

Решение системы методом Гаусса с выбором главного элемента: (-3, -5, 39, -7)

Решение системы numru : (-3, -5, 39, -7)

3. Матрица $A = \left[\begin{array}{cccc|c} 1 & -1 & 1 & -1 & 0 \\ 4 & -1 & 0 & -1 & 0 \\ 2 & 1 & -2 & 1 & 0 \\ 5 & 1 & 0 & -4 & 0 \end{array} \right]$

$$|A| = 0$$

Обратная матрица A^{-1} не существует

Число обусловленности $\|A\| * \|A^{-1}\|$ не определено

Решение системы методом Гаусса: (0, 0, 0, 0)

Решение системы методом Гаусса с выбором главного элемента: (0, 0, 0, 0)

Решение системы numru : (0, 0, 0, 0)

Приложение 2. Пример 1. Вариант 4

$$A_{ij} = \begin{cases} \frac{i+j}{m+n}, & i \neq j \\ n + m^2 + \frac{j}{m} + \frac{i}{n}, & i = j \end{cases}, \text{ где } i, j = 1, \dots, n$$

$$b_i = m \cdot n - i^3$$

$$n = 50, m = 15$$

1. Решение системы методом Гаусса: [14.579697 , 14.820286 , 15.017169 , 15.14857 , 15.192737 , 15.127925 , 14.932418 , 14.584508 , 14.062514 , 13.344768 , 12.409616 , 11.235428 , 9.800586 , 8.083496 , 6.0625796 , 3.7162647 , 1.0230141, -2.0387042, -5.4904013, -9.353563 , -13.64967 , -18.400167 , -23.626526 , -29.350159 , -35.59247 , -42.374817 , -49.7186 , -57.645206 , -66.175865 , -75.33203 , -85.13498 , -95.60591 , -106.76616 , -118.63695 , -131.23946 , -144.5951 , -158.72484 , -173.65 , -189.39175 , -205.9711 , -223.40935 , -241.72754 , -260.94696 , -281.0882 , -302.1727 , -324.22153 , -347.25543 , -371.29575 , -396.36328 , -422.47928]

Решение системы методом Гаусса с выбором главного элемента: также как и в обычном методе

Решение системы numru : [14.579697 , 14.820287 , 15.01717 , 15.148572 , 15.192738 , 15.127926 , 14.932418 , 14.58451 , 14.062515 , 13.344769 , 12.409616 , 11.235429 , 9.800589 , 8.083498 , 6.062579 , 3.716266 , 1.0230145, -2.0387027, -5.4903965, -9.3535595, -13.649668 , -18.400179 , -23.626535 , -29.35016 , -35.59246 , -42.37482 , -49.718613 , -57.645195 , -66.1759 , -75.332054 , -85.13496 , -95.605896 , -106.766136 , -118.636925 , -131.2395 , -144.59508 , -158.72485 , -173.65002 , -189.39174 , -205.97115 , -223.4094 , -241.72758 , -260.94684 , -281.0882 , -302.17273 , -324.2215 , -347.25552 , -371.2958 , -396.36337 , -422.4792]

Выводы

В ходе работы рассмотрен метод Гаусса, а также метод Гаусса с выбором главного элемента. Реализован алгоритм поиска определителя матрицы, обратной матрицы, числа обусловленности матрицы, а также решения системы линейных алгебраических уравнений.

На продемонстрированных примерах метод Гаусса с и без выбора главного элемента дает одинаковые результаты.