|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

|  |  |
| --- | --- |
| ФАКУЛЬТЕТ | «Информатика и системы управления» |

|  |  |
| --- | --- |
| КАФЕДРА | «Теоретическая информатика и компьютерные технологии» |

**Отчет по лабораторной работе № 2**

**«Реализация метода Гаусса c перестановками»**

***по курсу***

***«Численные методы линейной алгебры»***

Выполнил:

Студент группы ИУ9-72Б

Караник А.А.

Проверил:

Посевин Д.П.

1. *г.*

## Цель работы

Реализовать три варианта метода Гаусса с перестановками и научиться оценивать погрешность решения системы линейных уравнений для матриц произвольной размерности.

## Постановка задачи

• Реализовать метод Гаусса с перестановками по столбцам, по строкам, по столбцам и строкам одновременно для действительных квадратных матриц произвольной размерности n.

• Для проверки работоспособности алгоритмов необходимо использовать алгоритм тестирования, который заключался в том, что мы заведомо определяем значения координат вектора x, данный вектор является решением уравнения A•х = b; вычисляем b путем прямого перемножения матрицы A на вектор x и далее производим поиск решения уравнения A•х = b тем или иным методом Гаусса, получая xчисл, после чего производим сравнение полученного xчисл c заданным x, а также решением xбибл , полученным с использованием сторонней библиотеки выбранной студеном. При этом сравнение производится по Евклидовой норме разности вектора x-xчисл и x-xбибл.

• На защите лабораторной работы студент должен показать умение оценивать погрешность вычислений в зависимости от выполнения условия диагонального преобладания матрицы, умение сравнивать погрешности вычислений полученных методом Гаусса с перестановками по столбцам, по строкам, по столбцам и строкам одновременно. Понимать связь теории с практикой.

• Результат работы должен быть представлен в виде графиков зависимости абсолютной погрешности вычислений классическим методом Гаусса, методом Гаусса с перестановками по строкам, методом Гаусса с перестановками по столбцам, методом Гаусса с перестановками по столбцам и строкам, библиотечным методом от степени диагонального преобладания. Все графики должны быть построены на одной координатной плоскости. Напомним, что погрешность вычисления вектора x системы линейных алгебраических уравнений A•x=b тем или иным способом рассчитывается по Евклидовой норме разности точного решения и решения полученного соответствующим методом. Степень диагонального преобладания вычисляется, как максимальная разность по i между модулем диагонального элемента и суммы модулей вне диагональных элементов. Очевидно, что если значение степени диагонального преобладания положительна, то условие диагонального преобладания выполняется, в противном случае — не выполняется. Поэтому график должен быть построен как для отрицательных значений степени диагонального преобладания, так и для положительных.

## Реализация

Исходный код программы:

using Random

using LinearAlgebra

using Plots

using Distributions

function gauss\_classic(A, f)

    A\_aug = hcat(A, f)

    n = size(A, 1)

    for i in 1:n

        pivot = A\_aug[i, i]

        A\_aug[i, :] /= pivot

        for j in i+1:n

            factor = A\_aug[j, i]

            A\_aug[j, :] -= factor \* A\_aug[i, :]

        end

    end

    x = zeros(n)

    for i in n:-1:1

        x[i] = A\_aug[i, end] - A\_aug[i, i+1:end-1]' \* x[i+1:end]

    end

    return x

end

function gauss\_by\_rows(A, f)

    n = size(A, 1)

    A\_aug = [A f]

    row\_perm = collect(1:n)

    for k in 1:n-1

        max\_col = argmax(abs.(A\_aug[k, k:n])) + k - 1

        if max\_col != k

            A\_aug[:, [k, max\_col]] = A\_aug[:, [max\_col, k]]

            row\_perm[[k, max\_col]] = row\_perm[[max\_col, k]]

        end

        for i in k+1:n

            factor = A\_aug[i, k] / A\_aug[k, k]

            A\_aug[i, k:end] -= factor \* A\_aug[k, k:end]

        end

    end

    x = zeros(n)

    for j in n:-1:1

        x[j] = (A\_aug[j, end] - A\_aug[j, j+1:end-1]' \* x[j+1:end]) / A\_aug[j, j]

    end

    return x[row\_perm]

end

function gauss\_by\_columns(A, f)

    n = size(A, 1)

    A\_aug = [A f]

    for k in 1:n-1

        max\_row = argmax(abs.(A\_aug[k:n, k])) + k - 1

        if max\_row != k

            A\_aug[[k, max\_row], :] = A\_aug[[max\_row, k], :]

        end

        for i in k+1:n

            factor = A\_aug[i, k] / A\_aug[k, k]

            A\_aug[i, k:end] -= factor \* A\_aug[k, k:end]

        end

    end

    x = zeros(n)

    for j in n:-1:1

        x[j] = (A\_aug[j, end] - A\_aug[j, j+1:end-1]' \* x[j+1:end]) / A\_aug[j, j]

    end

    return x

end

function gauss\_combined(A, f)

    n = size(A, 1)

    A\_aug = [A f]

    row\_perm = collect(1:n)

    col\_perm = collect(1:n)

    for k in 1:n-1

        max\_val, max\_index = findmax(abs.(A\_aug[k:n, k:n]))

        max\_row, max\_col = max\_index[1] + k - 1, max\_index[2] + k - 1

        if max\_row != k

            A\_aug[[k, max\_row], :] = A\_aug[[max\_row, k], :]

            row\_perm[[k, max\_row]] = row\_perm[[max\_row, k]]

        end

        if max\_col != k

            A\_aug[:, [k, max\_col]] = A\_aug[:, [max\_col, k]]

            col\_perm[[k, max\_col]] = col\_perm[[max\_col, k]]

        end

        for i in k+1:n

            factor = A\_aug[i, k] / A\_aug[k, k]

            A\_aug[i, k:end] -= factor \* A\_aug[k, k:end]

        end

    end

    x = zeros(n)

    for j in n:-1:1

        x[j] = (A\_aug[j, end] - A\_aug[j, j+1:end-1]' \* x[j+1:end]) / A\_aug[j, j]

    end

    result = zeros(n)

    for i in 1:n

        result[col\_perm[i]] = x[i]

    end

    return result

end

function library(A, b)

    return A \ b

end

function euclidean\_norm(vec)

    return sqrt(sum(element^2 for element in vec))

end

function mulvec(A, vector)

    result = Float64[]

    for i in 1:size(A, 1)

        element = 0.0

        for j in 1:length(vector)

            element += A[i, j] \* vector[j]

        end

        push!(result, element)

    end

    return result

end

function generate\_matrix(l, r, n)

    return rand(Uniform(l,r), n, n)

end

function increase\_diag\_elements(A, diag)

    n = size(A, 1)

    for i in 1:n

        A[i, i] += diag \* sum(abs(A[i, j]) for j in 1:n if j != i)

    end

    return A

end

function diag\_dominance(A)

    return maximum(abs(A[i, i]) - sum(abs(A[i, j]) for j in 1:size(A, 2) if j != i) for i in 1:size(A, 1))

end

function calculate(method, A::Array{Float64}, x::Array{Float64})

    b = mulvec(A, x)

    x\_calc = method(A, b)

    return euclidean\_norm(x .- x\_calc)

end

n = 100

diag = Float64[]

y\_gauss = Float64[]

y\_gauss\_row = Float64[]

y\_gauss\_columns = Float64[]

y\_gauss\_combined = Float64[]

y\_library = Float64[]

coefs = [i \* 0.2 for i in 1:2:10]

for c in coefs

    A = generate\_matrix(-10.0, 10.0, n)

    A = increase\_diag\_elements(A, c)

    x = rand(Uniform(-10.0, 10.0), n)

    push!(diag, diag\_dominance(A))

    push!(y\_gauss, calculate(gauss\_classic, A, x))

    push!(y\_gauss\_row, calculate(gauss\_by\_rows, A, x))

    push!(y\_gauss\_columns, calculate(gauss\_by\_columns, A, x))

    push!(y\_gauss\_combined, calculate(gauss\_combined, A, x))

    push!(y\_library, calculate(library, A, x))

end

p = plot(diag,

    [y\_gauss, y\_gauss\_row, y\_gauss\_columns, y\_gauss\_combined, y\_library],

    label=["gauss\_classic" "gauss\_by\_rows" "gauss\_by\_columns" "gauss\_combined" "library"],

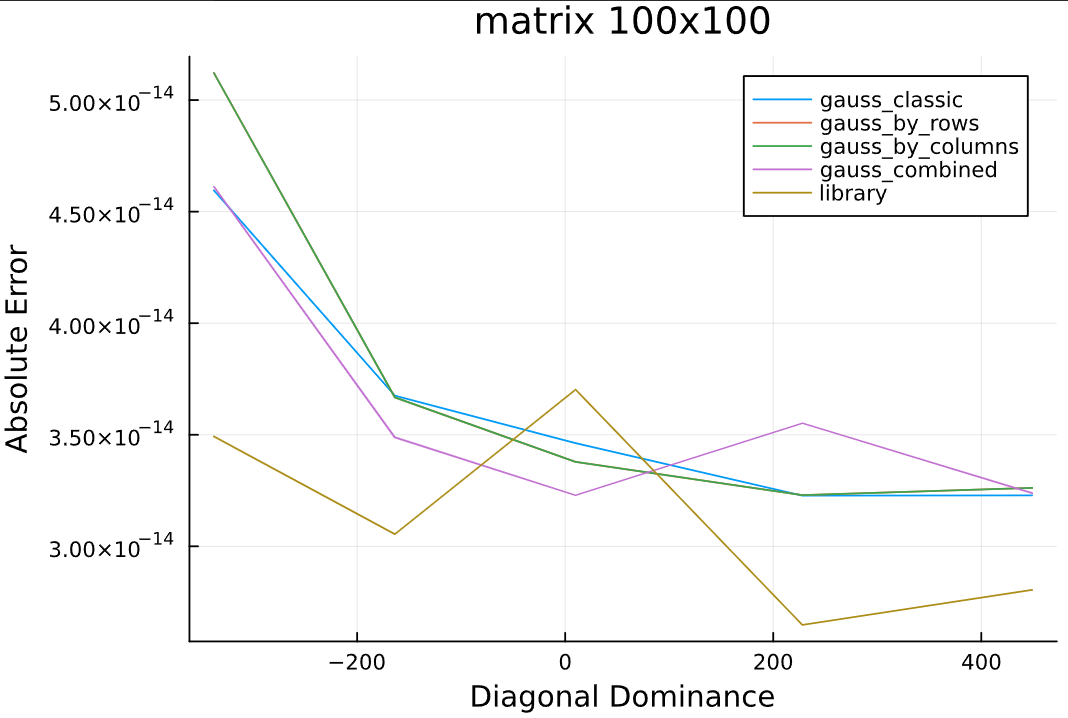
    title=("matrix $(n)x$(n)"),

    xlabel=("Diagonal Dominance"),

    ylabel=("Absolute Error"))

display(p)

Результат работы:



## Вывод

В ходе выполнения лабораторной работы были реализованы различные варианты метода Гаусса для решения систем линейных уравнений. Кроме того, был разработан и применен алгоритм тестирования для проверки корректности реализации. Тестирование состояло в сравнении численного решения системы с заведомо известным точным решением. Также результат работы был представлен в виде графиков зависимости абсолютной погрешности от степени диагонального преобладания.