Решатель уравнений методом Гаусса

Калюжин Александр Сергеевич, группа 3640103/00401

Преподаватель: Калюжнюк Александр Всеволодович

Санкт-Петербургский Политехнический Университет Петра Великого
Институт Прикладной Математики и Механики

2020 г.

Введение

Проектная работа посвящена созданию простого оконного приложения для решателя СЛАУ размера nxn с постоянными коэффициентами методом Гаусса с выделением главного элемента.
Программа была составлена на языке Python 3, т.к. в этом языке программирования имеются готовые библиотеки для подобного класса задач. Для реализации оконного интерфейса использовалась библиотека Tkinter.

Классический метод Гаусса заключается в том, что расширенную матрицу СЛАУ приводят путем линейных преобразований — сложения или вычитания строк с умножением их на константу — к верхнетреугольному виду с нормализованным первым в строке ненулевым элементом (прямой ход метода Гаусса), а затем, перейдя от матричного формализма обратно к формализму СЛАУ, последовательно выражают переменные снизу вверх (обратный ход метода Гаусса), получая столбец решений.

Теория. Классический метод Гаусса

Пусть исходная система выглядит так:

$$\begin{cases} a_{11}x_1 + ... + a_{1n}x_n = b_1 \\ \vdots \\ a_{n1}x_1 + ... + a_{nn}x_n = b_n \end{cases}$$

тогда ее матрица и столбец свободных членов:

$$A = \begin{pmatrix} a_{11} & ... & a_{1n} \\ . & & \\ & . & \\ & & . \\ a_{n1} & ... & a_{nn} \end{pmatrix} \qquad B = \begin{pmatrix} b_1 \\ . \\ . \\ . \\ b_n \end{pmatrix}$$

Расширенной матрицей будет матрица размером nx(n+1), получающаяся при совмещении матриц A и B

Теория. Классический метод Гаусса

Затем, производя над расширенной матрицей линейные преобразования, получаем ее верхнетреугольную форму (прямой ход метода):

$$\left(\begin{array}{cccc|c} 1 & \hat{a}_{12} & ... & \hat{a}_{1n} & \hat{b}_1 \\ 0 & 1 & ... & \hat{a}_{2n} & \hat{b}_2 \\ & & . & & . \\ & & . & & . \\ & & . & & . \\ 0 & 0 & ... & \hat{a}_{nn} & \hat{b}_n \end{array}\right)$$

Эта матрица эквивалентна СЛАУ вида:

$$\begin{cases} x_1 + \hat{a}_{12}x_2 + \hat{a}_{13}x_3 + \ldots + \hat{a}_{1n}x_n = \hat{b}_1 \\ x_2 + \hat{a}_{21}x_2 + \ldots + \hat{a}_{2n}x_n = \hat{b}_2 \\ \vdots \\ \hat{x}_n = \hat{b}_n \end{cases}$$

из нее без труда методом последовательных подстановок можно получить столбец решений первоначальной СЛАУ (обратный ход метода).
Для своей реализации метод требует $O(n^3)$ операций.

Мотивировка

Однако при программировании такого метода как метод Гаусса важно помнить, что при приведении матрицы к верхнедиагональному виду в линейных преобразованиях возможно случайное деление на ноль при встрече нулевых коэффициентов в случае, например, разреженной матрицы или на очень малое число при изначальной плохой обусловленности матрицы. В лучшем случае это приведет к неправильному результату, в худшем - в возвращению программой ошибки или вызову исключения (деление на ноль).
Чтобы избежать такой ситуации и не утруждать себя дополнительным прописыванием исключений можно модифицировать стандартный метод Гаусса до метода Гаусса с выделением главного элемента.

Теория. Выделение главного элемента

Метода Гаусса с выделением главного элемента — это модификация классического метода Гаусса, особенностью которой является введение такой перестановки уравнений (строк матрицы) на каждом шаге алгоритма, при которой на k-ом шаге ведущим элементом, т.е. элементом для которого вычисляется следующий постоянный коэффициент, оказывается наибольший по модулю элемент k-го столбца, называемый главным.

Понятно, что использование такого метода повышает количество необходимых для решения СЛАУ операций, но взамен это уберегает от лишних программных ошибок и предотвращает вероятную при таком стечении обстоятельств потерю времени.

Выделение главного элемента. Реализация алгоритма

В программном коде выделение главного элемента реализуется посредством добавления цикла, который проводит проверку на то, является ли данный элемент в k-ом столбце максимальным по модулю среди всех элементов k-го столбца.
Если является — строка остается на месте и с ней производятся необходимые вычисления по классическому алгоритму, если нет - то ищется строка с максимальным по модулю k-ым элементом из всех k-ых элементов строк, строки меняются местами и со строкой с найденным главным элементом производятся вычисления по алгоритму классического метода Гаусса.

Выделение главного элемента. Реализация алгоритма

Перед непосредственным вычислением столбца решений СЛАУ на вход программы подается число N, соотвествующее размерности системы, затем программа генерирует матрицы A и B, заполненные случайными целыми числами от 0 до 9. Для работы с матрицами была использована библиотека Numpy. Участки кода, в которых описан основной алгоритм вычисления стобца решений, был реализован в полностью функциональном стиле. Основная работа програмы заключалась в вызове созданных функций в той же области, где принимается на вход размерность матрицы.

Результаты

Написанная программа продемонстрировала отличную точность вычислений (производилось сравнение с уже существующими онлайн-решателями СЛАУ) и хорошую скорость работы, причем при задании систем с большими размерностями ($N > 30$) программа работала слегка быстрее, чем некоторые существующие онлайн-решатели СЛАУ.
Все это указывает на то, что реализованный алгоритм весьма эффективен и может применяться при реальных вычислениях.

Спасибо за внимание!