Отчёт по лабораторной работе 4

Супонина Анастасия Павловна

Содержание

Список иллюстраций

Список таблиц

Элементы списка иллюстраций не найдены.

Цель работы

Ознакомиться со сложными алгоритмы, встроенными для решения систем линейных уравнений в Octave. Научиться решать в данной программе СЛАУ методом Гаусса, а также LU и LUP разложениями.

Теоретическая часть.

Алгоритм решения СЛАУ методом Гаусса подразделяется на два этапа.

– На *первом этапе* осуществляется так называемый прямой ход, когда путём элементарных преобразований над строками систему приводят к ступенчатой или треугольной форме, либо устанавливают, что система несовместна. А именно, среди элементов первого столбца матрицы выбирают ненулевой, перемещают его на крайнее верхнее положение перестановкой строк и вычитают получившуюся после перестановки первую строку из остальных строк, домножив её на величину, равную отношению первогоэлемента каждой из этих строк к первому элементу первой строки, обнуляя тем самымстолбец под ним. После того, как указанные преобразования были совершены, первую строку и первый столбец мысленно вычёркивают и продолжают пока не останется матрица нулевого размера. Если на какой-то из итераций среди элементов первого столбца не нашёлся ненулевой, то переходят к следующему столбцу и проделывают аналогичную операцию.

– На *втором этапе* осуществляется так называемый обратный ход, суть которого заключается в том, чтобы выразить все получившиеся базисные переменные через небазисные и построить фундаментальную систему решений, либо, если все переменные являются базисными, то выразить в численном виде единственное решение системы линейных уравнений. Эта процедура начинается с последнего уравнения, из которого выражают соответствующую базисную переменную (а она там всего одна) и подставляют в предыдущие уравнения, и так далее, поднимаясь наверх. Каждой строчке соответствует ровно одна базисная переменная, поэтому на каждом шаге, кроме последнего(самого верхнего), ситуация в точности повторяет случай последней строки.

LU-разложение

LU-разложение — это вид факторизации матриц для метода Гаусса. Цель состоит в том, чтобы записать матрицу A в виде:

$$A = LU$$

где L — нижняя треугольная матрица, а U — верхняя треугольная матрица. Эта факторизованная форма может быть использована для решения уравнения

$$Ax = b.$$

LU-разложение существует только в том случае, когда матрица A обратима, а все главные миноры матрицы A невырождены. Этот метод является одной из разновидностей метода Гаусса.

Проверить можено при помощи следующего выражения:

$$det(A) = det(LU) = det(L)det(U)$$

В Octave есть специальная функция для вычисления lu.

LUP-разложение

Если используются чередования строк, то матрица A умножается на матрицу перестановок, и разложение принимает форму

$$PA = LU.$$

При помощи данной формулы можно проверить получившееся разложение. Аналогично предыдущему разложению используем для него функцию lu

Задание.

1) Выполнить Метод Гаусса
2) Сделать левое деление матриц
3) Применить LU разложение
4) Применить LUP разложение

Выполнение работы

Метод Гаусса

Выделяю отдельные строки или столбцы из матрицы

```
>> B = [ 1 2 3 4 ; 0 -2 -4 6 ; 1 -1 0 0 ]
B =

   1   2   3   4
   0  -2  -4   6
   1  -1   0   0

>> B (2, 3)
ans = -4
>> B (1, :)
ans =

   1   2   3   4

>> B (:, 1)
ans =

   1
   0
   1
```

Строки и столбцы

Провожу простые преобразования, а именно домнажаю и вычитаю или складываю строки, для того чтобы получить треугольную матрицу

Вычисляю значения всех x и сверяю с значениями полученными при выполнении специальной функции для метода Гаусса в Octave (rref)

```
>>> B(3, :)= B(3, :) - B(1, :)
     ^
>> B(3,:)=B(3, :) - B(1, :)
B =

   1   2   3   4
   0  -2  -4   6
   0  -3  -3  -4

>> B(3,:)= B(3, :) - 1.5*(B(2, :))
B =

    1    2    3    4
    0   -2   -4    6
    0    0    3  -13

>> x3 = -13/3
x3 = -4.3333
>> x2 = 6 + 4
x2 = 10
>> x2 = (6 + 4*x3)/-2
x2 = 5.6667
>> x1 = 4 -3*x3 - 2*x2
x1 = 5.6667
>> rref(B)
ans =

   1.0000        0        0   5.6667
        0   1.0000        0   5.6667
        0        0   1.0000  -4.3333
```

Решение

Преобразовываю формат выводимых в матрице чисел

```
>> rref(B)
ans =

   1.0000        0        0   5.6667
        0   1.0000        0   5.6667
        0        0   1.0000  -4.3333

>> format long
>> rref(B)
ans =

   1.000000000000000                   0                   0   5.666666666666667
                   0   1.000000000000000                   0   5.666666666666666
                   0                   0   1.000000000000000  -4.333333333333333

>> format short
```

Изменение формата чисел

Левое деление матриц

Применяя оператор “/” для левого деления матриц, предварительно разделив матрицу B, на матрицу коэффициентов A и матрицу суммарных значений b. Получаю значения для всех x.

```
>> B = [ 1 2 3 4 ; 0 -2 -4 6 ; 1 -1 0 0 ]
B =

   1   2   3   4
   0  -2  -4   6
   1  -1   0   0

>> A = B(:,1:3)
A =

   1   2   3
   0  -2  -4
   1  -1   0

>> b = B (:,4)
b =

   4
   6
   0

>> A\b
ans =

   5.6667
   5.6667
  -4.3333
```

Левое деление матриц

LU - разложение

Используя функцию lu следующим образом:

$$[L, U] = lu(A)$$

получаю угловые матрицы L и U из матрицы A. Также провожу проверку при помощи формулы

$$det(A) = det(LU) = det(L)det(U)$$

```
>> A
A =

   1   2   3
   0  -2  -4
   1  -1   0

>> [L, U] = lu(A)
L =

   1.0000        0        0
        0   0.6667   1.0000
   1.0000   1.0000        0

U =

   1   2   3
   0  -3  -3
   0   0  -2

>> det(A)
ans = -6
>> det(L)*det(U)
ans = -6
```

LU - разложение

LUP - разложение

Используя функцию lu следующим образом:

$$[l, u, p] = lu(A)$$

получаю LUP - разложение матрицы А

```
>> [l, u, p] = lu(A)
l =

   1.0000        0        0
   1.0000   1.0000        0
        0   0.6667   1.0000

u =

   1   2   3
   0  -3  -3
   0   0  -2

p =

Permutation Matrix

   1   0   0
   0   0   1
   0   1   0
```

LUP - разложение

Выводы

В процессе выполнения работы, я узнала о новых функциях в Octave. А именно rref, используемая для метода Гаусса, а также lu, используемая для множество разных разложений матрицы, а в данной работе я научилась использовать её для LU и LUP разложений. Также ознакомилась с методом решения СЛАУ при помощи левого деления и решила такую задачу в среде программирования Octave.