Лабораторная работа №4

Системы линейных уравнений

Демидова Екатерина Алексеевна

Содержание

Список иллюстраций

1 Цель работы

Научиться решать системы линейных уравнениий с помощью системы для математических вычислений Octave.

2 Задание

- Решить СЛАУ с помощью Метода Гаусса
- Решить СЛАУ, применив левое деление
- Найти LU-разложение
- Найти LUP-разложение

3 Теоретическое введение

Система линейных уравнений в общем виде записывается следующим образом:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m \end{cases}$$

где a_{ij} - коэффициенты при неизвестных переменных x_i, x_j и т.д., b_i - свободные члены.

Матрица A называется основной матрицей системы, b — столбцом свободных членов. Алгоритм решения СЛАУ методом Гаусса подразделяется на два этапа [1]. – На первом этапе осуществляется так называемый прямой ход, когда путём элементар- ных 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 по- ложение перестановкой строк и вычитают получившуюся после перестановки первую строку из остальных строк, домножив её на величину, равную отношению первого элемента каждой из этих строк к первому элементу первой строки, обнуляя тем самым столбец 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 соответ- ствует ровно одна базисная переменная, поэтому на каждом шаге, кроме последнего (самого верхнего), ситуация в точности повторяет случай последней строки.

Приведём пример использования Octave[2]:

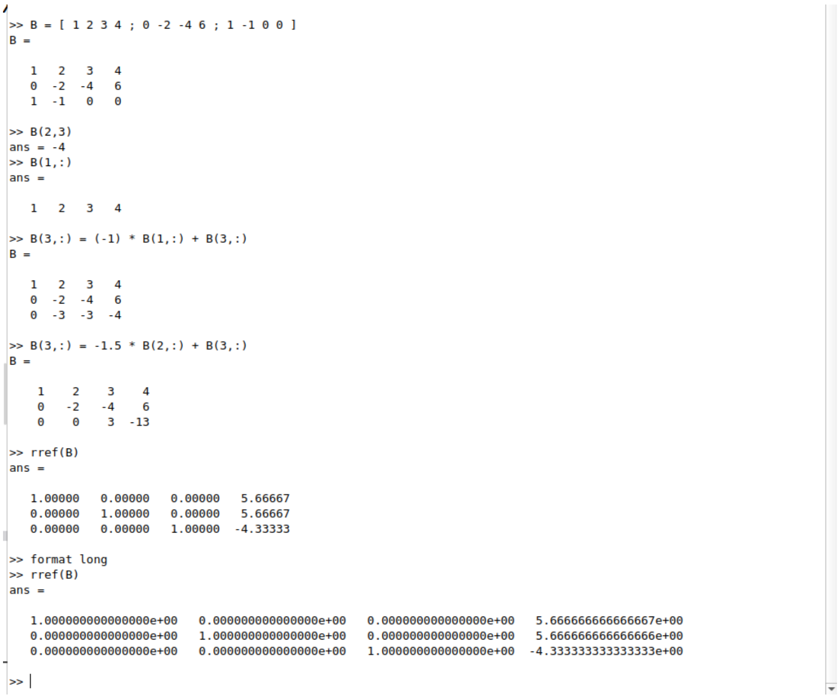
Решение систем уравнений с помощью операций линейной алгебры над векторами и матрицами.

```
b = [4; 9; 2] # Column vector
A = [ 3 4 5;
      1 3 1;
      3 5 9 ]
x = A \ b     # Solve the system Ax = b
```

Кроме того, в Octave есть функция для LU- и LUP-разложений: `lu(A)`

4 Выполнение лабораторной работы

Для данной в задании системы линейных уравнений построим расширенную матрицу и передадим её в переменную B. Затем попробуем получить конкретный элемент матрицы и строку. С помощью этой функции Octave сможем реализовать явно метод Гаусса. Сначала добавим к третьей строке первую строку, умноженную на -1, далее добавим к третьей строке вторую строку, умноженную на -1.5. Матрица теперь имеет треугольный вид. Получим ответ, выразив сначала базисную переменную в последней строке, поделив четвертый элемент на 3, затем подставив полученное значение в предпоследнюю строку и так далее. также решим эту СЛАУ с помощью функции `rref(B)`. Чтобы увеличить количество отображаемых знаков после запятой используем функцию `format long`. (рис. [4.1])



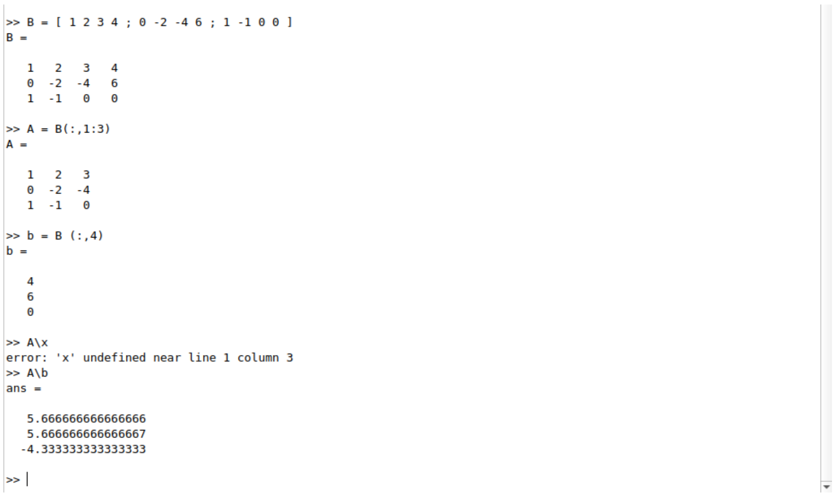
```
>> B = [ 1 2 3 4 ; 0 -2 -4 6 ; 1 -1 0 0 ]
B =

   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(3,:) = (-1) * B(1,:) + B(3,:)
B =

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

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

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

>> rref(B)
ans =

   1.00000   0.00000   0.00000   5.66667
   0.00000   1.00000   0.00000   5.66667
   0.00000   0.00000   1.00000  -4.33333

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

   1.000000000000000e+00   0.000000000000000e+00   0.000000000000000e+00   5.666666666666667e+00
   0.000000000000000e+00   1.000000000000000e+00   0.000000000000000e+00   5.666666666666666e+00
   0.000000000000000e+00   0.000000000000000e+00   1.000000000000000e+00  -4.333333333333333e+00

>> |
```

Рис. 4.1: Метод Гаусса

строенная операция для решения линейных систем вида

$$Ax = b$$

в Octave называется левым делением и записывается как A.⊠ Это концептуально эквивалентно выражению $A^{-1}b$. Снова введём расширенную матрицу, выделим из неё квадратную матрицу A и вектор b, а затем найдём вектор x с помощью левого деления (рис. [4.2])



```
>> 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\x
error: 'x' undefined near line 1 column 3
>> A\b
ans =

   5.666666666666666
   5.666666666666667
  -4.333333333333333

>> |
```

Рис. 4.2: Левое деление

С помощью функции lu() d Octave распишем LU-разложение матрицы A (рис. [4.3]):

```
>> A
A =

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

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

   1.000000000000000e+00   0.000000000000000e+00   0.000000000000000e+00
   0.000000000000000e+00   6.666666666666666e-01   1.000000000000000e+00
   1.000000000000000e+00   1.000000000000000e+00   0.000000000000000e+00

U =

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

>> L*U = A
parse error:

  invalid left hand side of assignment

>>> L*U = A
        ^

>> L*U == A
ans =

  1  1  1
  1  1  1
  1  1  1

>> |
```

Рис. 4.3: LU-разложение

С помощью функции lu() d Octave распишем LUP-разложение матрицы A (рис. [4.4])

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

   1.000000000000000e+00   0.000000000000000e+00   0.000000000000000e+00
   1.000000000000000e+00   1.000000000000000e+00   0.000000000000000e+00
   0.000000000000000e+00   6.666666666666666e-01   1.000000000000000e+00

U =

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

P =

Permutation Matrix

   1   0   0
   0   0   1
   0   1   0

>> P*A==L*U
ans =

  1  1  1
  1  1  1
  1  1  1

>> |
```

Рис. 4.4: LUP-разложение

5 Выводы

В результате выполнения работы научились решать системы линейных уравнениий с помощью системы для математических вычислений Octave.

Список литературы

1. Метод Гаусса [Электронный ресурс]. Wikimedia Foundation, Inc., 2023. URL: https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D0%93%D0%B0%D1%83%D1%81%D1%81%D0%B0.
2. GNU Octave Documentation [Электронный ресурс]. Free Software Foundation, 2023. URL: https://docs.octave.org/latest/.