Отчёт по лабораторной работе №4

СЛАУ в Octave

Сырцов Александр Юрьевич

Содержание

List of Tables

List of Figures

1 Цель работы

Ознакомится со способами решения СЛАУ инструментами языка Octave: метод Гаусса, LU-decomposition в методе Гаусса и поиск обратной матрицы.

2 Задание

Проделать шаги, указанные в методических материалах и подготовить отчёт.

3 Выполнение лабораторной работы

1. Создаём расширенную матрицу B, с которой мы будем проделывать преобразования (рис. -fig. 3.1)

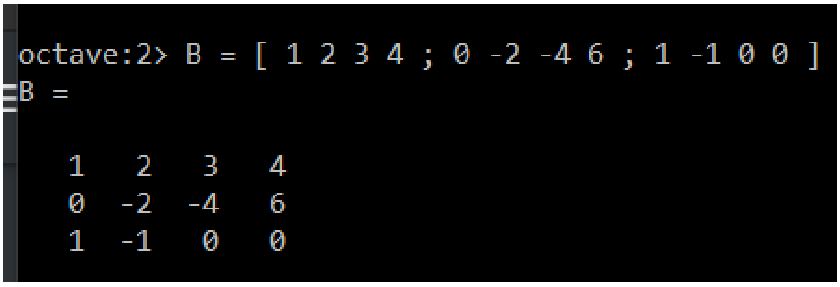


Figure 3.1: Матрица B

2. Пробуем разные подходы обращения к элементам матрицы (рис. -fig. 3.2):

    1. обращение к индексу на второй строке в третьем столбце.
    2. обращение к элементу 8 работает аналогичным образом, так как научные языки программирования поддерживают возможность использования матриц в качестве таблиц с упорядоченными по столбцам элементами.
    3. обращаемся к первой строке, используя векторный оператор ":" на месте номера столбцов.

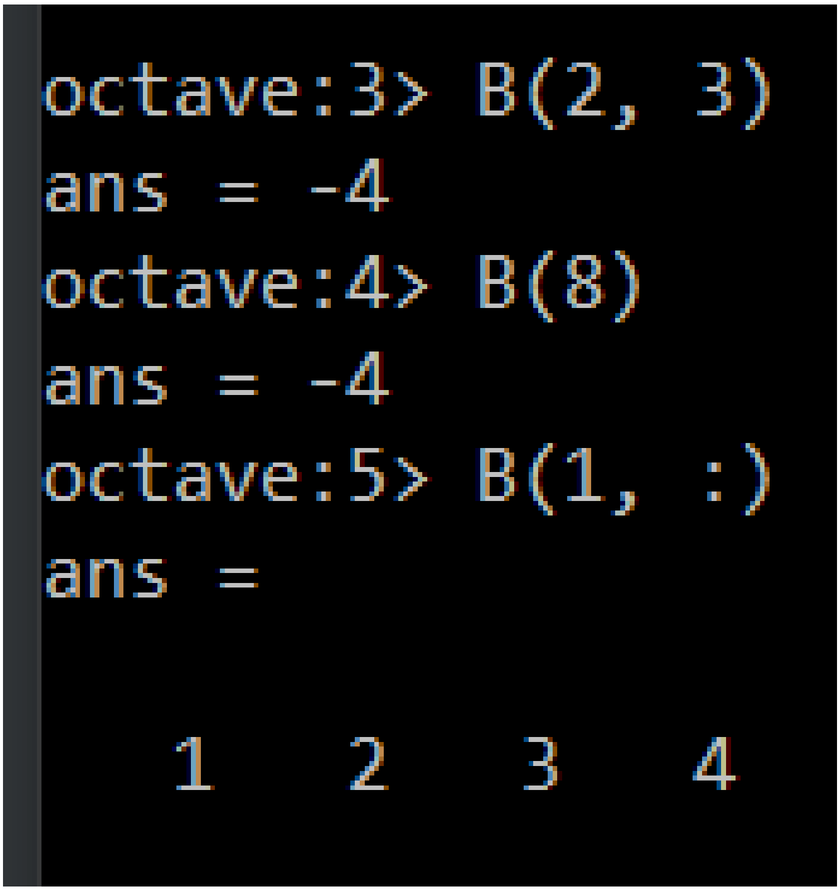


Figure 3.2: Обращение к элементам

3. По методу Гаусса в явном виде приводим дополненную матрицу к треугольному виду (рис. -fig. 3.3):

    1. из строки 3 вычитаем строку 1.
    2. из строки 3 вычитаем строку 2.

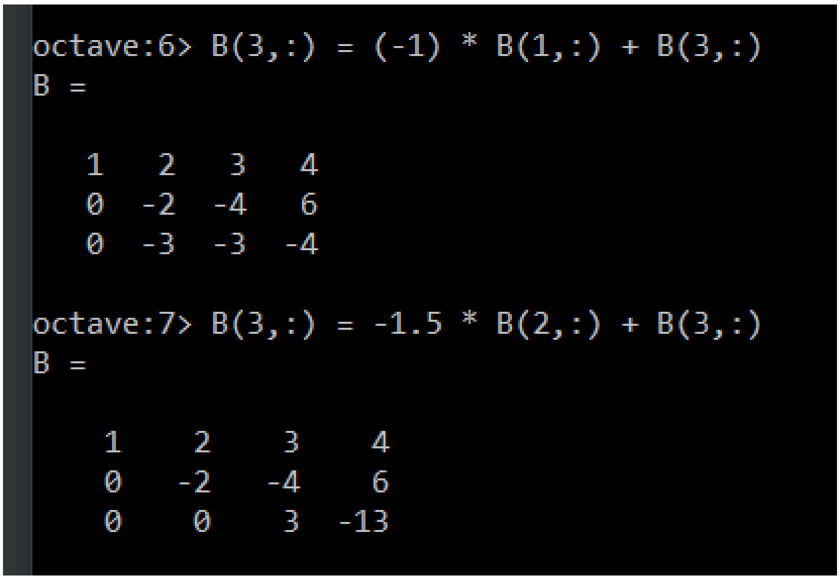


Figure 3.3: Приведение матрицы B к треугольноому виду

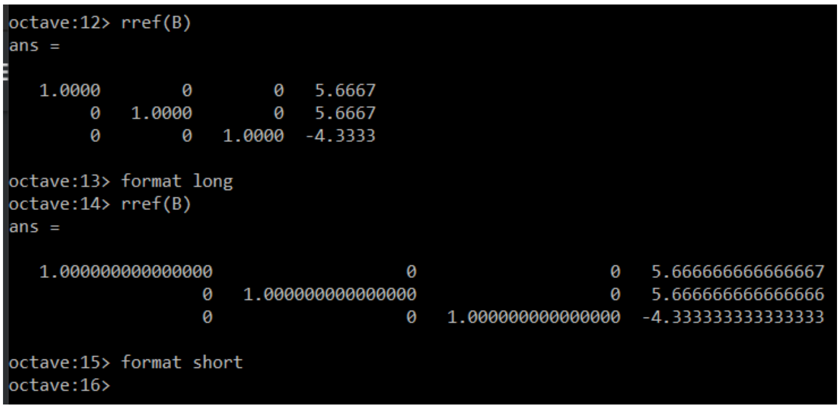
Так как у нас не получилось обнулить строки, мы можем спокойно переходить к следующему шагу решения, а именно: сокращаем третью строку на 3 и подставляем значение во вторую строку и тд. таким образом получаем вектор-решение

$$\vec{v} = \begin{pmatrix} \frac{17}{3} \\ \frac{17}{3} \\ \frac{-13}{3} \end{pmatrix}.$$

4. Эквивалентное решение можно получить стандартными средствами языка, используя функцию `rref()`. Выводиться дополненная единичная матрица (рис. -fig. 3.4)

```
format long % отображаем не менее 15 символов после запятой
```

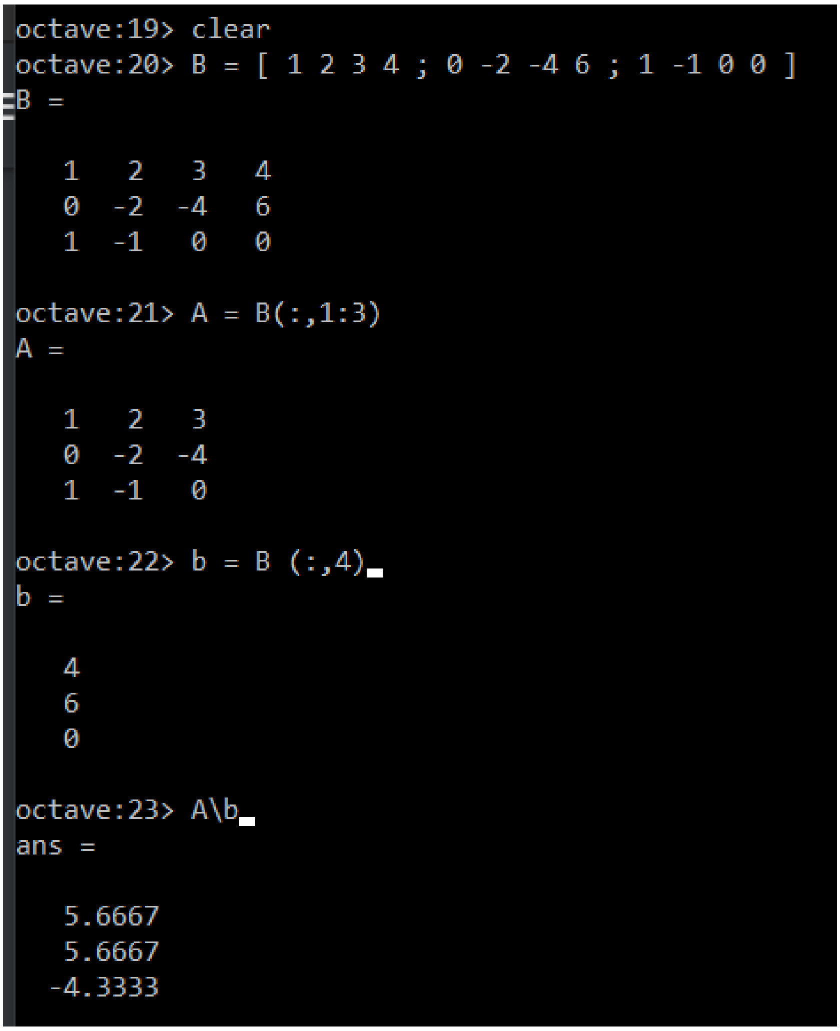
```
format short % отображаем не более 5 символов после запятой
```



```
octave:12> rref(B)
ans =

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

octave:13> format long
octave:14> rref(B)
ans =

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

octave:15> format short
octave:16>
```

Figure 3.4: Решение стандартной функцией rref()

5. Теперь решаю СЛАУ методом левостороннего деления, или методом нахождения обратной матрицы. метод заключается в нахождении матрицы обратной к матрице системы и левостороннему домножению обеих частей матречного уравнения на неё. К счастью все эти действия сводятся к операции левостороннего деления, которая самостоятельно проводит необходимые действия и выводит в ответ значение вектора неизвестных, то есть вектор-решение системы (рис. -fig. 3.5)



```
octave:19> clear
octave:20> 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

octave:21> A = B(:,1:3)
A =

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

octave:22> b = B (:,4)
b =

   4
   6
   0

octave:23> A\b
ans =

   5.6667
   5.6667
  -4.3333
```

Figure 3.5: Решение методом левостороннего деления

6. LU-декомпозиция – метод используемый в методе Гаусса. L – lower triange matrix, U – upper triangle matrix. Раскладывая матрицу системы таким образом достаточно решить следующие выражения прямой и обратной подстановкой соответственно.

$$U * \vec{x} = \vec{y}$$

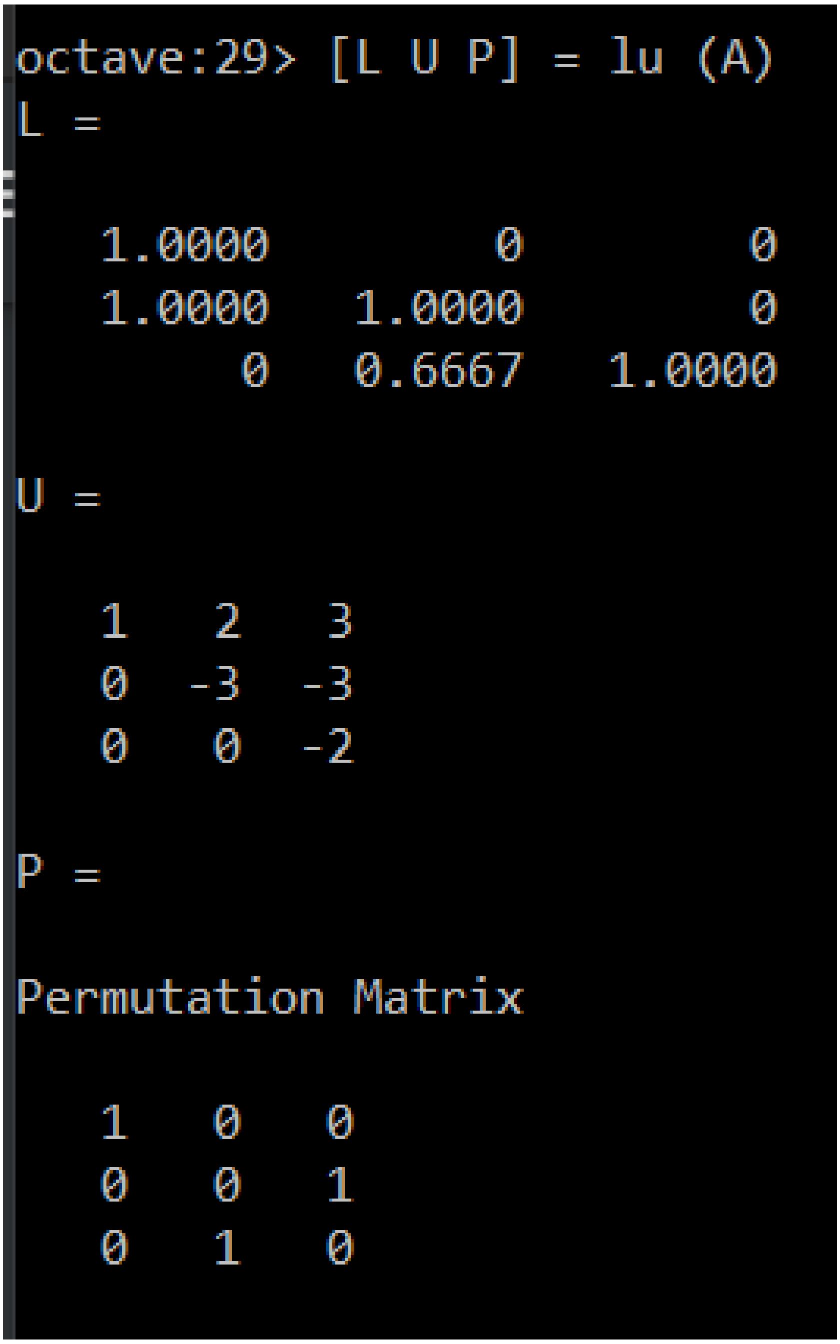
$$L * \vec{y} = \vec{b}$$

Реализация разложения матрицы в языке Octave выглядит просто и не требеут особых действий: достаточно применить к конвектору (объекту вектору) с переменными `L` и `U` инициализацию функцией `lu()` (рис. -fig. 3.6). На этом же рисунке деманстрирую свойства получившихся матриц. Без наличия матрицы перестановок Octave записывает матрицу L без перестановок строк.

```
octave:24> A
A =

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

octave:25> [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

octave:26> L*U
ans =

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

octave:27> det(A)
ans = -6
octave:28> det(L*U)
ans = -6
```

Figure 3.6: LU-decomposition

7. Аналогичным образом находим LUP-декомпозицию. Различие методов в наличии матрицы перестановок P. (рис. -fig. 3.7).



```
octave:29> [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
```

Figure 3.7: LUP-decomposition

Последним шагом проверяю свойства разложения (рис. -fig. 3.8).

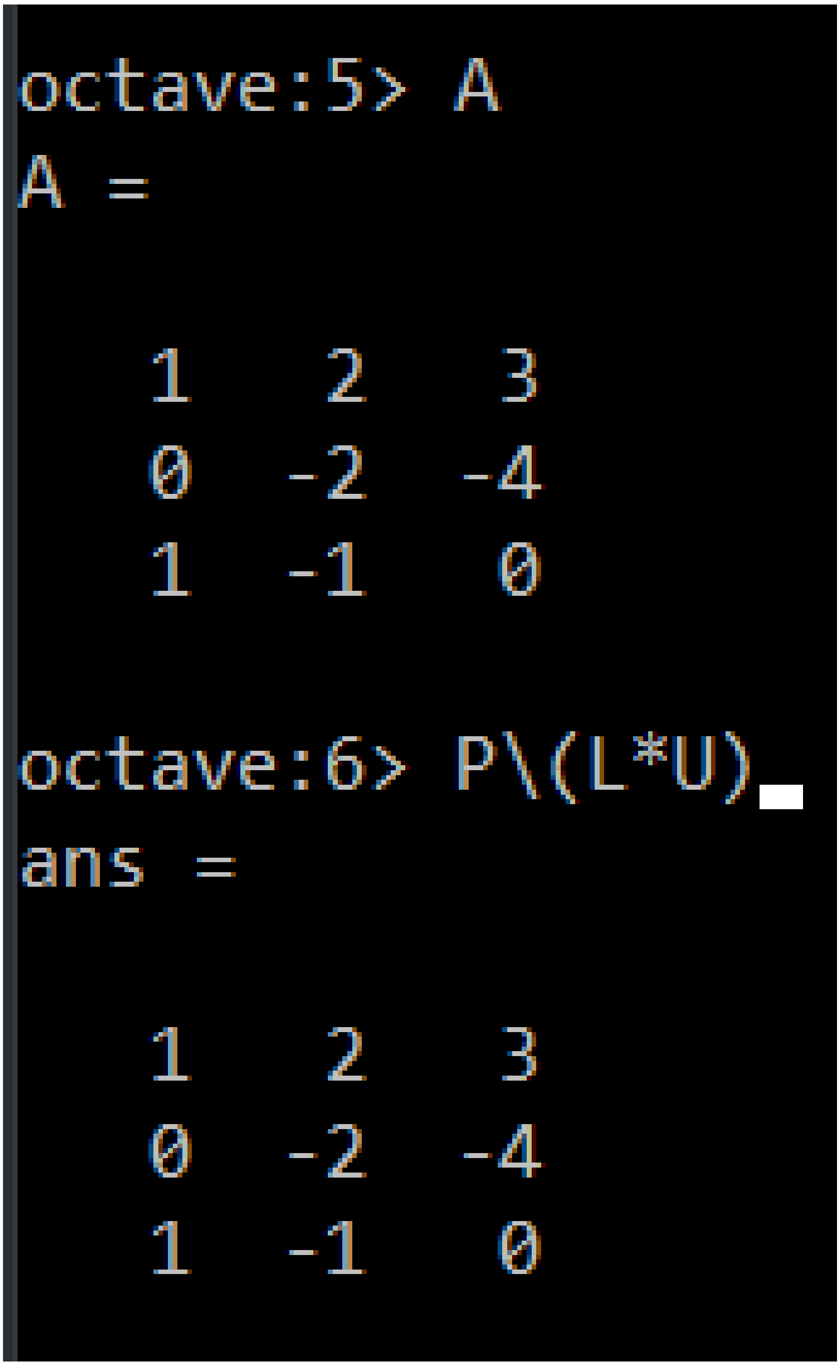


Figure 3.8: Проверка LUP-декомпозиции

4 Выводы

Мне удалось применить основные приёмы решения СЛАУ в языке Octave, представленных в лабораторной работе.