Прямые методы решения линейных систем. Вариант 11.

Содержание

Ввод данных

Введем расширенную матрицу системы

```
M = [6.687233,0.80267,-2.06459,0;
    0.80267,5.07816,0.48037,1;
    -2.06459,0.48037,4.02934,0]
```

```
M = 3×4
    6.6872    0.8027   -2.0646         0
    0.8027    5.0782    0.4804    1.0000
   -2.0646    0.4804    4.0293         0
```

Решение системы методом Гаусса

Алгоритм

А) *Прямой ход:* исходная система сводится к эквивалентной системе с верхнетреугольной матрицей по формулам:

- $a_{kj}^{(k)} = \dfrac{a_{kj}^{(k-1)}}{a_{kk}^{(k-1)}},\ a_{kk}^{(k-1)} \neq 0,\ j = k, \ldots, n+1$
- $a_{ij}^{(k)} = a_{ij}^{(k-1)} - a_{kj}^{(k)} a_{ik}^{(k-1)},\ \ i = k, \ldots, n,\ \ j = k, \ldots, n+1$
- $k = 1, \ldots, n$

В) *Обратный ход:* вычисляется решение системы по формулам:

- $x_i = a_{in+1}^{(i)} + \sum\limits_{j=n+1}^{n} a_{ij}^{(i)}\ x_j,\ \ i = n, \ldots, 1$

Замечание: Выбор ведущего элемента осуществляется по столбцу

Реализация алгоритма

Реализация алгоритма содержится в функции Gauss.m

Результат работы

Рассмотрим полученный вектор

```
x = Gauss(M)
```

```
x = 3×1
   -0.0386
    0.2072
   -0.0445
```

Сравним его с результатом работы стандартной функции

```
y=M(:,1:size(M,1))\M(:,size(M,1)+1)
```

```
y = 3×1
   -0.0386
    0.2072
   -0.0445
```

LU - разложение матрицы

Алгоритм

Воспользуемся следующим алгоритмом:

Для шагов $i = 1 : n$ поочередно выполним:

1. $$l_{ij} = a_{ji} - \sum_{k=1}^{i-1} l_{jk} u_{ki}, \ \ j = i : n$$
2. $$u_{i,j} = \frac{a_{ij} - \sum_{k=1}^{i-1} l_{ik} u_{kj}}{l_{ii}}, \ j = i : n$$

Реализация алгоритма

Реализация алгоритма содержится в функции LU_Dec.m

Результат работы

Получим следующее разложение:

```
[myL,myU]=LU_Dec(M)
```

```
myL = 3×3
    6.6872         0         0
    0.8027    4.9818         0
```

```
   -2.0646    0.7282    3.2855
myU = 3×3
    1.0000    0.1200   -0.3087
         0    1.0000    0.1462
         0         0    1.0000
```

Проверим его правильность, посчитав разность LU - A

```
myL*myU - M(:,1:size(M,1))
```

```
ans = 3×3
     0     0     0
     0     0     0
     0     0     0
```

Поиск обратной матрицы

Алгоритм

Для нахождения обратной матрицы решим следующий набор систем, используя полученное выше LU разложение матрицы:

1. $Ly = e_i,\ e_i = (0, \ldots, 1_i, \ldots)^T$
2. $U(A^{-1})_i = y,\ \forall i = 1 : n$

Реализация алгоритма

Реализация находится в файле inverseMatrix.m

Результат работы

```
invA = inverseMatrix(myL,myU)
```

```
invA = 3×3
    0.1848   -0.0386    0.0993
   -0.0386    0.2072   -0.0445
    0.0993   -0.0445    0.3044
```

Для проверки воспользуемся встроенным методом поиска обратной матрицы

```
inv(M(:,1:size(M,1)))
```

```
ans = 3×3
    0.1848   -0.0386    0.0993
   -0.0386    0.2072   -0.0445
    0.0993   -0.0445    0.3044
```

Число обусловленности матрицы

Найдем число обусловленности матрицы по следующей формуле:

$$cond(A) = \|A\|\|A^{-1}\|$$

Подставив полученные ранее значения, получим

```
condA = norm(M(:,1:size(M,1)))*norm(invA)
```

```
condA = 3.0000
```