Задача на собственные значения

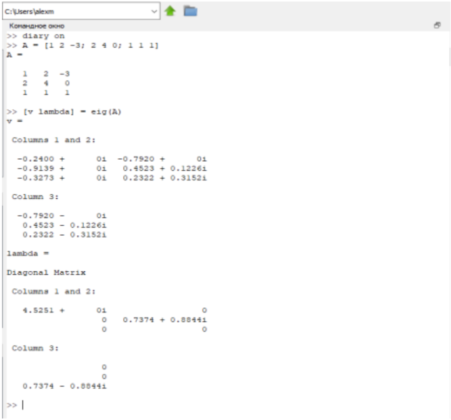
Милёхин Александр НПМмд-02-21

Цель работы

Научиться находить в Octave собственные значения и собственные векторы матрицы, а также научиться предсказывать вероятность состояния системы.

Собственные значения и собственные векторы

Включим журналирование работы. После чего зададим матрицу А. Для нахождения собственных значений и собственных векторов матрицы используем команду eig с двумя выходными аргументами.




```
>> diary on
>> A = [1 2 -3; 2 4 0; 1 1 1]
A =

   1   2  -3
   2   4   0
   1   1   1

>> [v lambda] = eig(A)
v =

 Columns 1 and 2:

  -0.2400 +      0i  -0.7920 +      0i
  -0.9139 +      0i   0.4523 + 0.1226i
  -0.3273 +      0i   0.2322 + 0.3152i

 Column 3:

  -0.7920 -      0i
   0.4523 - 0.1226i
   0.2322 - 0.3152i

lambda =

Diagonal Matrix

 Columns 1 and 2:

   4.5251 +      0i                  0
                  0   0.7374 + 0.8844i
                  0                  0

 Column 3:

                  0
                  0
   0.7374 - 0.8844i

>> |
```

Figure 1: Собственные значения и векторы матрицы

Собственные значения и собственные векторы

Для того, чтобы получить матрицу с действительными собственными значениями, создадим симмитричную матрицу путём умножения исходной матрицы на транспонированную. И повторим шаги, проделанные ранее.

```
C:\Users\alexm
Командное окно
>> C = A' * A
C =

    6   11   -2
   11   21   -5
   -2   -5   10

>> [v lambda] = eig(C)
v =

   0.876137   0.188733  -0.443581
  -0.477715   0.216620  -0.851390
  -0.064597   0.957839   0.279949

lambda =

Diagonal Matrix

    0.1497         0         0
         0    8.4751         0
         0         0   28.3752

>>
```

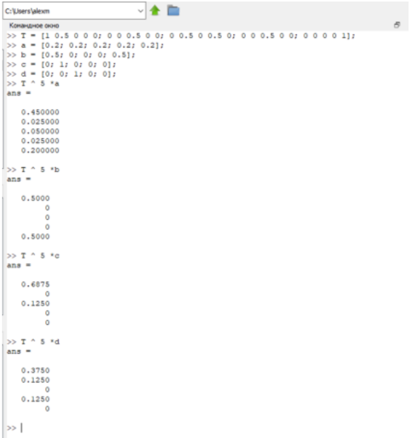
Figure 2: Действительные собственные значения

Случайное блуждание

На курсе “Теория случайных процессов” мы дополнительно ознакомились с цепями Маркова. Наша задача - предсказать вероятности состояния системы. Для примера случайного блуждания найдем вектор вероятности после 5 шагов для каждого начального вектора.

Случайное блуждание

Покажем, как мы задаем матрицу, начальные векторы, а затем находим соответствующие вероятности.



```
C:\Users\alexm
Командное окно
>> T = [1 0.5 0 0 0; 0 0 0.5 0 0; 0 0.5 0 0.5 0; 0 0 0.5 0 0; 0 0 0 0 1];
>> a = [0.2; 0.2; 0.2; 0.2; 0.2];
>> b = [0.5; 0; 0; 0; 0.5];
>> c = [0; 1; 0; 0; 0];
>> d = [0; 0; 1; 0; 0];
>> T ^ 5 *a
ans =

   0.450000
   0.025000
   0.050000
   0.025000
   0.200000

>> T ^ 5 *b
ans =

   0.5000
        0
        0
        0
   0.5000

>> T ^ 5 *c
ans =

   0.6875
        0
   0.1250
        0
        0

>> T ^ 5 *d
ans =

   0.3750
   0.1250
        0
   0.1250
        0

>> |
```

Figure 3: Нахождение вероятностей

Случайное блуждание

Теперь найдём вектор равновесного состояния для цепи Маркова с переходной матрицей.

```
C:\Users\alexm
Командное окно
>> T = [0.48 0.51 0.14; 0.29 0.04 0.52; 0.23 0.45 0.34]
T =

   0.480000   0.510000   0.140000
   0.290000   0.040000   0.520000
   0.230000   0.450000   0.340000

>> [v lambda] = eig(T)
v =

  -0.6484  -0.8011   0.4325
  -0.5046   0.2639  -0.8160
  -0.5700   0.5372   0.3835

lambda =

Diagonal Matrix

   1.0000        0        0
        0   0.2181        0
        0        0  -0.3581

>> x = v(:,1)/sum(v(:,1))
x =

   0.3763
   0.2929
   0.3308

>> |
```

Figure 4: Вектор равновесного состояния

Случайное блуждание

Таким образом, x = (0.37631 0.29287 0.33082), является вектором равновесного состояния. Проверим это.

```
C:\Users\alexm
Командное окно
>> T ^ 10 *x
ans =

   0.3763
   0.2929
   0.3308

>> T ^ 50 *x
ans =

   0.3763
   0.2929
   0.3308

>> T ^ 50 * x - T ^ 10 * x
ans =

   4.4409e-16
   2.7756e-16
   3.8858e-16

>> diary off
```

Figure 5: Проверка вектора равновесия

Результат лабораторной работы

Я научился в Octave находить собственные значения и собственные векторы матрицы. Также научился работать с цепями Маркова и находить вектор равновесия.

Спасибо за внимание