

# **Научное программирование**

**Отчет по лабораторной работе № 8**

Меньшов Иван Сергеевич НПМмд-02-21

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>5</b>
2.1	Собственные значения и собственные векторы . . . . .	5
2.2	Случайное блуждание . . . . .	6
<b>3</b>	<b>Вывод</b>	<b>8</b>

# List of Figures

2.1	Собственные значения и собственные векторы 01 . . . . .	5
2.2	Собственные значения и собственные векторы 02 . . . . .	6
2.3	Случайное блуждание 01 . . . . .	6
2.4	Случайное блуждание 02 . . . . .	7
2.5	Случайное блуждание 03 . . . . .	7

# 1 Цель работы

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

## 2 Выполнение лабораторной работы

### 2.1 Собственные значения и собственные векторы

Включим журналирование работы. После чего зададим матрицу A. 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 =

-0.23995 + 0.00000i    0.79195 + 0.00000i    0.79195 - 0.00000i
-0.91393 + 0.00000i   -0.45225 - 0.12259i   -0.45225 + 0.12259i
-0.32733 + 0.00000i   -0.23219 - 0.31519i   -0.23219 + 0.31519i

lambda =

Diagonal Matrix

 4.52510 + 0.00000i      0      0
      0    0.73745 + 0.88437i      0
      0      0      0    0.73745 - 0.88437i
```

Figure 2.1: Собственные значения и собственные векторы 01

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

```

>> 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
    0.14970         0         0
         0    8.47515         0
         0         0   28.37516

```

Figure 2.2: Собственные значения и собственные векторы 02

## 2.2 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. Задаем матрицу, начальные векторы, а затем находим соответствующие вероятности.

```

>> 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 0.5 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
    0.450000

>> T^5 * b
ans =
    0.50000
    0.00000
    0.00000
    0.00000
    0.50000

>> T^5 * c
ans =
    0.68750
    0.00000
    0.12500
    0.00000
    0.18750

>> T^5 * d
ans =
    0.37500
    0.12500
    0.00000
    0.12500
    0.37500

```

Figure 2.3: Случайное блуждание 01

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

```
>> 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 =

   -0.64840   -0.80111    0.43249
   -0.50463    0.26394   -0.81601
   -0.57002    0.53717    0.38351

lambda =

Diagonal Matrix

    1.00000         0         0
         0    0.21810         0
         0         0   -0.35810

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

    0.37631
    0.29287
    0.33082
```

Figure 2.4: Случайное блуждание 02

Таким образом,  $x = (0.37631 \ 0.29287 \ 0.33082)$ , является вектором равновесного состояния.

```
>> T^10 * x
ans =

    0.37631
    0.29287
    0.33082

>> T^50 * x
ans =

    0.37631
    0.29287
    0.33082

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

    2.2204e-16
    2.2204e-16
    1.6653e-16

>> diary off
```

Figure 2.5: Случайное блуждание 03

## 3 Вывод

В ходе выполнения данной работы я научился находить собственные значения и собственные векторы матрицы. Также научился работать с цепями Маркова и находить вектор равновесия.