# *Лабораторна робота № 6*

## Обчислення власних значень і власних векторів матриць

***Мета роботи:*** отримання практичних навичок в чисельному розрахунку власних значень і власних векторів матриць на основі використання характеристичних рівнянь і ортогональних перетворень і реалізація описаних алгоритмів у пакеті *Mathematica*.

***Короткі теоретичні відомості***

В роботі передбачено вивчення прямих і ітераційних методів обчислення власних значень і власних векторів матриці. До першої групи методів належать методи Фаддєєва–Левер'є і Крилова, що базуються на обчисленні коренів характеристичного рівняння матриці, до другої групи відносяться варіанти QR-алгоритму, які використовують ортогональні подібні перетворення за допомогою матриць обернення Гівенса чи матриць відбиття Хаусхолдера і зводять структуру вхідної матриці до блочно-діагональної форми з блоками першого і другого порядку.

### 6.1. Методи характеристичного рівняння матриці

Власними значеннями матриці називають такі скалярні величини, які є коренями рівняння:

$$Ax = \lambda x,$$

або

$$(A - \lambda E)x = 0. \tag{6.1}$$

Множина всіх власних значень матриці називається її спектром. Кожному власному значенню матриці відповідає свій власний вектор. Якщо деякий ненульовий вектор $x = x_\lambda \neq 0$ задовольняє умові (6.1), то він є власним вектором матриці $A$. Відносно власних значень можна побудувати характеристичне рівняння матриці $A$ у вигляді полінома степеня $n$ $p(\lambda)$. Ров'язок цього рівняння визначає множину всіх власних значень матриці.

Для обчислення коефіцієнтів характеристичного рівняння крім прямого розкриття самого визначника існує декілька методів, серед яких виділяється *метод Фадєєва–Левер'є*, за допомогою якого обчислюються:

- коефіцієнти характеристичного полінома

$$\det(\lambda E - A) = \lambda^n + b_{n-1}\lambda^{n-1} \ldots + b_1\lambda^1 + b_0; \tag{6.2}$$

- обернена матриця

$$A^{-1} = \frac{K_0}{b_0}; \tag{6.3}$$

- резольвента матриці

$$(\lambda E - A)^{-1} = \frac{K_{n-1}\lambda^{n-1} + K_{n-2}\lambda^{n-2} + \ldots + K_1\lambda^1 + K_0}{\det(\lambda E - A)}. \tag{6.4}$$

Метод Фаддєєва-Левер'є базується на результатах обчислення слідів матриці $A$ і добутків матриць $A \cdot K_{n-1}$, де $K_{n-1}$ – коефіцієнти чисельника резольвенти матриці (6.4), які визначаються ітераційно (спочатку коефіцієнт чисельника $K_{n-i}$, потім коефіцієнт знаменника $b_i$) відповідно до наведеної нижче процедури, де $E$ – одинична матриця, $\mathrm{Tr}(A) = a_{11} + a_{22} + \ldots a_{nn}$ – слід матриці.

$$
\begin{array}{ll}
K_{n-1} = E; & b_{n-1} = -\,\mathrm{Tr}(A\cdot K_{n-1}); \\
K_{n-2} = A\cdot K_{n-1} + b_{n-1}\cdot E; & b_{n-2} = -1/2\ \mathrm{Tr}(A\cdot K_{n-2}); \\
\text{-----------------------------------} & \text{-----------------------------------} \\
K_{n-k} = A\cdot K_{n-k+1} + b_{n-k+1}\cdot E; & b_{n-k} = -1/k\ \mathrm{Tr}(A\cdot K_{n-k}); \\
K_0 = A\cdot K_1 + b_1\cdot E; & b_0 = -1/n\ \mathrm{Tr}(A\cdot K_0);
\end{array}
\tag{6.5}
$$

**Приклад 6.1**

Побудувати характеристичне рівняння за допомогою метода Фадєєва–Левер'є і знайти власні числа матриці $A$={{5,3,1},{3,5,3},{1,3,5}};

```
A={{35,3,1.3},{3,6,3},{1,3,5}};
n=Length[A];
b[n]=-Tr[A];b[n+1]=1;K[n]=IdentityMatrix[n];
Print["b[",n-1,"]=  ",b[n-1],"   K[",n-1,"]=  ",K[n-1]];
Do[K[n-i]=A.K[n-i+1]+b[n-i+1]*IdentityMatrix[n];
 b[n-i]=-1/(i+1)*Tr[A.K[n-i]];
 Print["b[",n-i-1,"]=    ",b[n-i],"      K[",n-i-1,"]=    ",K[n-i]],{i,1,n-1}]

b[2]=   -15    K[2]=   {{1,0,0},{0,1,0},{0,0,1}}
b[1] =   56    K[1]=   {{-10,3,1},{3,-10,3},{1,3,-10}}
b[0]=   -48    K[0]=   {{16,-12,4},{-12,24,-12},{4,-12,16}}
```

Згідно з отриманими коефіцієнтами характеристичне рівняння має вид:

$$-48 + 56\,\lambda - 15\,\lambda^2 + \lambda^3.$$

Власні числа матриці розрахуємо на основі вбудованої функції пакету *Mathematica* `NSolve`:

```
NSolve [-48+56 x -15 x^2 +x^3 ==0,x]
```

{{x→1.228},{x→4.},{x→9.772}}

Те ж характеристичне рівняння можна отримати, скориставшись стандартною функцією пакету:

```
U=CharacteristicPolynomial[A,x]
```

48-56 x+15 $x^2$-$x^3$

Власні значення матриці можуть бути обчислені за допомогою засобів *Mathematica*:

```
Eigenvalues[N[A]]

{9.772,4.,1.228}
```

Крім методу Фадєєва–Левер'є досить поширеним є метод Крилова, який базується на теоремі Келлі–Гамільтона про те, що довільна матриця $A$ задовольняє своєму характеристичному рівнянню:

$$A^n + b_{n-1}A^{n-1} + b_{n-2}A^{n-2} + \ldots + b_1 A + b_0 = 0. \tag{6.6}$$

Якщо вираз (6.6) помножити на довільний вектор $x$, можна отримати такий запис:

$$x^{(n)} = A^n x = -b_{n-1}x^{(n-2)} - b_{n-2}x^{(n-2)} - \ldots - b_1 x^{(1)} - b_0, \tag{6.7}$$

де вектори $x^{(i)}$ обчислюються за формулами

$$x^{(1)} = Ax;\ x^{(2)} = Ax^{(1)} = A^2 x;\ x^{(i)} = A^i x. \tag{6.8}$$

або у скалярному виді:

$$x_k^{(i)} = \sum_{j=1}^{i} a_{kj} x_j^{(i-1)}. \tag{6.9}$$

На основі матрично-векторного рівняння (6.7) формуємо систему лінійних рівнянь для коефіцієнтів характеристичного поліному матриці:

$$\begin{pmatrix} x_1^{(n-1)} & x_1^{(n-2)} & \ldots & x_1^{(0)} \\ x_2^{(n-1)} & x_2^{(n-2)} & \ldots & x_2^{(0)} \\ \ldots & \ldots & \ldots & \ldots \\ x_n^{(n-1)} & x_n^{(n-2)} & \ldots & x_n^{(0)} \end{pmatrix} \begin{pmatrix} b_{n-1} \\ b_{n-2} \\ \ldots \\ b_0 \end{pmatrix} = \begin{pmatrix} -x_1^{(n)} \\ -x_2^{(n)} \\ \ldots \\ -x_n^{(n)} \end{pmatrix}. \tag{6.10}$$

Якщо система (6.10) буде виродженою, необхідно замінити початковий вектор $x$, який для зручності звичайно обирають як $x = [1,0,..,0]^T$.

**Приклад 6.2**

Користуючись методом Крилова, побудувати характеристичне рівняння матриці з прикладу 6.1. Вибираючи початковий вектор $x^{(0)}=\{1,0,0\}$, знаходимо необхідні вектори (6.9) і будуємо та розв’язуємо систему рівнянь (6.10):

```
Clear[x,b];
x[0]={1,0,0};
A={{5,3,1},{3,5,3},{1,3,5}};
n=Length[A];
Do[x[j]= A.x[j-1],{j,n}]
Array[x,n]

{{5,3,1},{35,33,19},{293,327,229}}

K={x[2],x[1],x[0]}  (* Формування матриці системи рівнянь *)

{{35,33,19},{5,3,1},{1,0,0}}

p = -x[n]   (* Формування правої частини системи рівнянь *)

{-293,-327,-229}

b=LinearSolve[Transpose[K],p]   (* Обчислення коефіцієнтів характеристичного рівняння *)

{-15,56,-48}

Pol[x_]=Sum[x^i*b[[n-i]],{i,0,n-1}]+x^n;
Pol[x]

48-56x+15x²-x³
```

Таким чином, отримуємо вектор коефіцієнтів b={-15,56,-48}, який збігається з результатом, отриманим у прикладі 6.1. Такі ж самі власні значення матриць, а також власні вектори можуть бути обчислені за допомогою засобів *Mathematica*:

```
{values,vectors}=Eigensystem[N[A]];
DiagonalMatrix[values]//MatrixForm
```

$$\begin{pmatrix} 9.772 & 0. & 0. \\ 0. & 4. & 0. \\ 0. & 0. & 1.228 \end{pmatrix}$$

```
vectors//MatrixForm
```

$$\begin{pmatrix} -0.528451 & -0.664439 & -0.528451 \\ -0.707107 & -1.63758\times 10^{-15} & 0.707107 \\ 0.469829 & -0.747342 & 0.469829 \end{pmatrix}$$

### 6.2. Методи ортогональних перетворень

Для матриці розмірності $n$ характеристичний поліном має такий же порядок. Оскільки аналітичний розв'язок існує лише для алгебраїчних рівнянь до четвертого порядку включно, для матриць великих розмірностей використовують ітераційні методи.

#### 6.2.1. QR–алгоритм

QR-алгоритм базується на перетворенні подібності матриці $A$ таким чином, щоб власні значення матриці, отриманої внаслідок перетворення, обчислювалися простіше, ніж для початкової матриці. Найпростіше знаходити власні значення трикутної матриці, для якої:

$$\det(\lambda E - A) = (\lambda - a_{11}) \cdot (\lambda - a_{22}) \cdot \ldots \cdot (\lambda - a_{nn}),$$

при цьому власні значення дорівнюють діагональним елементам матриці $\lambda_i = a_{ii}$, $i = 1,2,..,n$.

**Приклад 6.3**

Знайти власні значення матриці, розглянутої в прикладах 6.1-6.2 за допомогою QR-алгоритму.

Побудуємо ітераційний процес, що базується на QR-перетвореннях. Кількість ітерацій можна визначити, виходячи з розмірності матриці або накладаючи обмеження на значення елементів, що знаходяться нижче головної діагоналі (наприклад, всі $a_{ij} < 0.01$, якщо $i > j$, $i, j = 1,2,\ldots,n$).

```
A={{5,3,1},{3,5,3},{1,3,5}};
A1[1]=A;
n=10;
Do[{Q[i],R[i]}=QRDecomposition[N[A1[i]]];
A1[i+1]=R[i].Transpose[Q[i]],{i,n}];
Print[A1[n+1]]

{{9.772,0.00101993,3.95546×10-9},
{0.00101993,4.,0.0000310253},{3.95546×10-9,0.0000310253,1.228}}

MatrixForm[A1[11]
```

$$\begin{pmatrix} 9.772 & 0.00101993 & 3.95546\times 10^{-9} \\ 0.00101993 & 4. & 0.0000310253 \\ 3.95546\times 10^{-9} & 0.0000310253 & 1.228 \end{pmatrix}$$

На останній десятій ітерації отримали матрицю, на головній діагоналі у якої розміщені всі власні значення.

### 6.2.2. Декомпозиція Хаусхолдера

Існує також різновид $QR$–алгоритму, в якому для отримання трикутної матриці $R$ використовується не матриці обертання Гівенса, а метод відбиття *Хаусхолдера*. В першому випадку подібними перетворюваннями обнуляються послідовно елементи матриці по стовпчикам, а в другому – всі елементи стовпчика одночасно.

Для окремого випадку симетричних матриць перетворення Хаусхолдера використовується для приведення таких матриць до тридіагональної (а не трикутної) форми. Тоді відпадає необхідність генерації послідовності подібних матриць, як в $QR$–алгоритмі, оскільки власні значення тридіагональної матриці визначаються безпосередньо, наприклад, знаходженням коренів поліномів Штурма, які мають наступний вигляд:

$$f_i(\lambda) = (a_{ii} - \lambda) f_{i-1}(\lambda) - d_i^2 f_{i-2}(\lambda),$$

де $f_0(\lambda) = 1$, $f_1(\lambda) = a_{11} - \lambda$; $a_{ii}$ і $d_i = d_{ii\text{-}1} = d_{i\text{-}1,i}$ — діагональні і недіагональні елементи матриці.

**Приклад 6.4**

Отримати власні значення матриці за допомогою декомпозиції Хаусхолдера:

```
A={{5,3,1},{3,5,3},{1,3,5}};
{Q,R}=SchurDecomposition[N[A]]; Map[MatrixForm,{N[Q],N[R]}]
```

$$\left\{\begin{pmatrix} -0.528451 & -0.707107 & 0.469829 \\ -0.664439 & 2.52402\times10^{-16} & -0.747342 \\ -0.528451 & 0.707107 & 0.469829 \end{pmatrix}, \begin{pmatrix} 9.772 & 1.04083\times10^{-17} & 4.68978\times10^{-16} \\ 0. & 4. & 3.09697\times10^{-16} \\ 0. & 0. & 1.228 \end{pmatrix}\right\}$$

В такому випадку матриця $R$ буде містити всі власні значення початкової матриці $A$:

```
Chop[R]//MatrixForm
```

$$\begin{pmatrix} 9.772 & 0 & 0 \\ 0 & 4. & 0 \\ 0 & 0 & 1.228 \end{pmatrix}$$

```
Tr[R,List]

{9.772, 4., 1.228}
```

## 6.3. Обчислення окремих власних значень

Для оцінки окремих власних значень матриці можна використовувати теорему Гершгоріна, яка стверджує, що матриця $A$ порядку $n\times n$ має $n$ власних значень, кожне з яких лежить в межах кола:

$$|z - a_{ii}| \le r_i, \quad r_i = \sum_{\substack{j=1 \\ j\ne i}}^{n} |a_{ij}|. \qquad (6.11)$$

Якщо необхідно оцінити лише деякі з власних значень (наприклад, $\lambda_{max}$ або $\lambda_{min}$), то простіше використовувати степеневий метод для формування послідовності векторів:

$$z_{k+1} = A \cdot z_k = A^{k+1} \cdot z_0,\ k = 1,2,3\dots \qquad (6.12)$$

Припустимо, що матриця $A$ має $n$ лінійно незалежних власних векторів і максимальне за величиною власне значення $\lambda_{max} = \lambda_1$ таке, що $|\lambda_1| > |\lambda_2| > \dots > |\lambda_n|$. Коли розкласти деякий початковий вектор $z_0$ в базисі власних векторів матриці:

$$z_0 = \alpha_1 x_1 + \alpha_2 x_2 + \dots + \alpha_n x_n,$$

то

$$z_k = \sum_{j=1}^{n} \lambda_j^k \alpha_j x_j = \lambda_1^k \left( \alpha_1 x_1 + \sum_{j=2}^{n} \left( \frac{\lambda_j}{\lambda_1} \right)^k \alpha_j x_j \right), \quad k = 0,1,2,.... \qquad (6.13)$$

Оскільки $|\lambda_j / \lambda_1| < 1$ для $j > 2$, то напрямок вектора $z_k$ прямує до напрямку власного вектора $x_1$, якщо тільки $a_1 \neq 0$. Для підвищення стійкості обчислень проводять масштабування послідовності векторів $z_k$, яке найпростіше здійснити, якщо перейти до послідовності $y_k$ шляхом нормування векторів $z_k$ за значеннями їх найбільших елементів $z_{ki}$, тобто замість виразу (6.13) використовують співвідношення:

$$y_k = \frac{z_k}{\max z_{ki}}, \quad z_{k+1} = A \cdot y_k, \quad k = 0,1,2,.... \qquad (6.14)$$

при цьому

$$z_{ki} \Rightarrow \lambda_1, \qquad (6.15)$$

і похибка визначення найбільшого власного значення прямує до нуля як $(\lambda_2/\lambda_1)^k$. Коли степеневий метод застосувати до оберненої матриці $A^{-1}$, то аналогічно можна оцінити величину найменшого власного значення $\lambda_{min}$, якщо виконується умова

$$|\lambda_n| < |\lambda_{n-1}|.$$

**Приклад 6.5.**

Знайти найбільше власне значення матриці з прикладів 6.1-6.4. Спочатку, вибираючи вектор z[1]={1,1,1,1,1} згідно з формулою (6.15) обчислюємо послідовність векторів $z_{k+1}$:

```
A={{5,3,1},{3,5,3},{1,3,5}};
n=Length[A];z[1]={1,1,1};
Do[k=N[z[i][[Ordering[Abs[z[i]],-1]]]];z[i+1]=A.z[i]/k[[1]];
Print["k[",i,"]=  ",k[[1]],"   z[",i,"]=  ",z[i]],{i,n^2}]
```

```
k[1]=  1.         z[1]=  {1, 1, 1}
k[2]=  11.        z[2]=  {9., 11., 9.}
k[3]=  9.90909    z[3]=  {7.90909, 9.90909, 7.90909}
k[4]=  9.78899    z[4]=  {7.78899, 9.78899, 7.78899}
k[5]=  9.77413    z[5]=  {7.77413, 9.77413, 7.77413}
k[6]=  9.77227    z[6]=  {7.77227, 9.77227, 7.77227}
k[7]=  9.77204    z[7]=  {7.77204, 9.77204, 7.77204}
k[8]=  9.77201    z[8]=  {7.77201, 9.77201, 7.77201}
k[9]=  9.772      z[9]=  {7.772, 9.772, 7.772}
```

Оскільки матриця симетрична, то можна знайти наступне після найбільшого власне її значення. Для цього необхідно побудувати нову матрицю $A_1$ на основі початкової матриці $A$, яка має найбільше власне значення $\lambda_1 = \lambda_{max}$:

$$A_1 = A - l_{max} V V^T,$$

де $V$ – власний вектор матриці.

```
A={{5,3,1},{3,5,3},{1,3,5}};
n=Length[A];
V = Eigenvectors[N[A]]

{{-0.528451,-0.664439,-0.528451},{-0.707107,-1.63758×10^-15,
0.707107},{0.469829,-0.747342,0.469829}}

lmax= 9.772;
A1=A-lmax*V[[1]]*V[[1]]

{{2.27107, 0.271067, -1.72893}, {-1.31413, 0.685865, -1.31413},
{-1.72893, 0.271067, 2.27107}}

MatrixForm[A1]
```

$$\begin{pmatrix} 2.27107 & 0.271068 & -1.72893 \\ -1.31414 & 0.685863 & -1.31414 \\ -1.72893 & 0.271068 & 2.27107 \end{pmatrix}$$

```
z[1]={0,0,1};
n=3;
Do[k=N[z[i][[Ordering[Abs[z[i]],-1]]]];z[i+1]=A1.z[i]/k[[1]];
Print["k[",i,"]=  ",k[[1]],"   z[",i,"]=  ",z[i]],{i,n^2}]

k[ 1 ]=   1.     z[ 1 ]=   {0,0,1}
k[ 2 ]=   2.27107    z[ 2 ]=   {-1.72893,-1.31414,2.27107}
k[ 3 ]=   -3.61471    z[ 3 ]=   {-3.61471,-0.710573,3.43043}
k[ 4 ]=   3.96514    z[ 4 ]=   {3.96514,0.0678274,-3.83093}
k[ 5 ]=   3.94612    z[ 5 ]=   {3.94612,-0.0327485,-3.91849}
k[ 6 ]=   -3.98635    z[ 6 ]=   {3.98565,-0.0148913,-3.98635}
k[ 7 ]=   4.00071    z[ 7 ]=   {-3.99859,0.00232999,4.00071}
k[ 8 ]=   3.99924    z[ 8 ]=   {-3.99864,-0.000297232,3.99924}
k[ 9 ]=   3.99972    z[ 9 ]=   {-3.99968,-0.000249167,3.99972}
```

Отриманий результат досить близький до значення 4., отриманому в попередніх прикладах.

**Приклад 6.6**

Знайти найменше власне значення матриці з прикладів 6.1-6.5.

```
A={{5,3,1},{3,5,3},{1,3,5}};
A2=Inverse[A]

{{0.333333, -0.25, 0.0833333}, {-0.25, 0.5, -0.25}, {0.0833333,
-0.25, 0.333333}}

z[1]={0,0,1};
n=3;
Do[k=N[z[i][[Ordering[Abs[z[i]],-1]]]];
z[i+1]=A2.z[i]/k[[1]],{i,n^2}];
k
```

{0.814334}

Найменше власне значення

Lmin=1/0.814334=1.228,

що відповідає результату, який отримано в прикладах 6.1-6.4.

### 6.4. Обчислення власних значень матриці в пакеті Mathematica

В пакеті Mathematica реалізовано метод характеристичного рівняння матриці (6.2), який для початкової матриці *A* запускається такими командами:

Eigenvalues[A] – для обчислення власних значень;

Eigenvectors[A] – для обчислення власних векторів:

Eigensystem[A] – для одночасного визначення власних значень і векторів матриці.

Для обчислень коренів характеристичного рівняння чисельними методами слід замість посилання на матрицю A вказати N[A]. Ці оператори базуються на обчисленні коренів характеристичного рівняння матриці, яке також можливо отримати за допомогою оператора CharacteristicPolynomial[A,x].

**Порядок виконання роботи**

1. Використати матрицю коефіцієнтів з лабораторної роботи № 4.
2. За допомогою методів, що потребують застосування характеристичного рівняння матриці, визначити всі власні значення матриці, користуючись засобами пакету щодо розв'язку нелінійних або систем лінійних рівнянь. Непарні номери варіантів здійснюють пошук методом Фаддєєва–Левер'є, парні – методом Крилова.
3. Порівняти отримані результати з власними значеннями, обчисленими за допомогою функції Eigenvalues[A]. За допомогою функції Eigenvectors[A] обчислити власні вектори матриці.
4. Використовуючи ітераційний метод, що базується на QR-перетворенні, отримати первісну декомпозицію матриці *A* на складові Q і R. Побудувати ітераційний процес, задаючись обмеженням на значення елементів, що знаходяться нижче головної діагоналі на рівні 0.05. Порівняти отримані результати з визначеними в п. 2. Для непарних номерів варіантів використати QR-декомпозиції, для парних – декомпозицію Хаусхолдера
5. Визначити кількість ітерацій, яка необхідна для досягнення розбіжності між отриманими в п.3 і п.4 значеннями на рівні 0.01.
6. Використовуючи степеневий метод, обчислити максимальне і мінімальне власні значення матриці *A*. Для непарних номерів варіантів – максимальне, для парних – мінімальне власні значення матриці *A*. Виключити отримане значення шляхом перетворення матриці і визначити наступне власне значення. Оцінити похибку.
7. Скласти звіт за отриманими результатами, навести у ньому математичні формули використаних методів по кожному пункту завдання, дати оцінку порівняльної точності отриманих рішень різними методами і кількості виконаних ітерацій.