Numerical Methods

Report

Date of the exercise: **25/04/2019**

Exercise: **Determining eigenvalues and eigenvectors of matrix**

Group: 2, Team:

Subsection (names):

1. Mateusz Nowotnik

2. Dawid Tomala

1. Krylov method

The matrix we have chosen:

$$A = \begin{bmatrix} 3 & -1 & 7 \\ 5 & 0 & 5 \\ 6 & 12 & 6 \end{bmatrix}$$

Taking as the initial vector $y^{(0)} = 1{,}0{,}0^T$, yields

$y^{(0)}$	$y^{(1)} = Ay^{(0)}$	$y^{(2)} = Ay^{(1)}$	$y^{(3)} = Ay^{(2)}$
$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 3 \\ -1 \\ 7 \end{bmatrix}$	$\begin{bmatrix} 46 \\ 81 \\ 58 \end{bmatrix}$	$\begin{bmatrix} 891 \\ 650 \\ 1075 \end{bmatrix}$

From that we can create a system of linear equations

$$\begin{gathered} 46p_1 + 3p_2 + 1p_3 = -891 \\ 81p_1 - 1p_2 + 0p_3 = -650 \\ 58p_1 + 7p_2 + 0p_3 = -1075 \end{gathered}$$

Solving the system, we obtained the following characteristic polynomial coefficients:

$$p_1 = -9 \qquad p_2 = -79 \qquad p_3 = -240$$

The characteristic equation of matrix **A** is of the form:

$$\lambda^3 - 9\lambda^2 - 79\lambda - 240 = 0$$

Solving that polynomial, we obtained:

$$\lambda_1 \approx 15.22446$$

Calculating coefficients g_1, g_2 for λ_1:

$$\begin{gathered} g_2 = \lambda_1 + p_1 = 6.22446 \\ g_1 = \lambda_1 g_2 + p_2 = 15.76404 \end{gathered}$$

and setting $g_3 = 1$

Using the formula

$$x^{(i)} = g_1 y^{(0)} + g_2 y^{(1)} + g_3 y^{(2)} + \cdots + g_n y^{(n-1)}$$

Eigenvector $x^{(1)}$ was obtained.

$x^{(1)}$
$\begin{bmatrix} 15.7640 \\ 0 \\ 0 \end{bmatrix}$

2. Power method

For the program to work properly - a matrix should be 3x3 and its rank (number of linearly independent eigenvectors) equal to 3.

```cpp
#include <iostream>
using namespace std;

// initialize the initial vector
void initVec(int v, double y[1][3])
{
    switch(v) {
        case 1:
            y[0][0] = 1; y[0][1] = 0; y[0][2] = 0;
            break;
        case 2:
            y[0][0] = 0; y[0][1] = 1; y[0][2] = 0;
            break;
        case 3:
            y[0][0] = 0; y[0][1] = 0; y[0][2] = 1;
            break;
        default:
            cout << "Something's wrong!" << endl;
    }
}

int main()
{
    double A[3][3];
    double y[9][3] = {0};
    double eigenvalue, quot[3];
    bool oscQuot = true;
    int v = 0;

    cout << "3x3 matrix A:" << endl;


    for (int i = 0; i < 3; i++)
        for (int j = 0; j < 3; j++)
            cin >> A[i][j];
```

```

    while (oscQuot == true){
        // initial vector y(0)
        initVec(++v, y);

        // obtaining the iterations of vector 'y'
        for (int i = 1; i < 9; i++)
            for (int j = 0; j < 3; j++)
                for (int k = 0; k < 3; k++)
                y[i][j] += y[i-1][k]*A[k][j];

        // calculating the quotients
        quot[0] = y[8][0]/y[7][0];
        quot[1] = y[8][1]/y[7][1];
        quot[2] = y[8][2]/y[7][2];

        // changing the value of initial vector if the values of quotients are oscillating
        if (quot[0] != quot[2])
            oscQuot = false;
    }

    // calculating the eigenvalue as the arithmetic mean of all quotients
    eigenvalue = (quot[0] + quot[1] + quot[2])/3.0;

    /*
    // Krylov
    for (int i = 0; i < 3; i++) {
        cout << endl;
        for (int j = 0; j < 4; j++)
            cout << y[j][i] << " ";
    }
    cout << endl << endl;
    */

    cout << "\nHighest modulus eigenvalue: " << eigenvalue << endl;

    cout << "\nHighest corresponding eigenvector: " << endl;
    for (int i = 0; i < 3; i++)
        cout << y[8][i] << endl;

    cout << "\nAfter normalization: " << endl;
    cout << y[8][0]/ max(max(y[8][0],y[8][1]),y[8][2]) << endl;
    cout << y[8][1]/ max(max(y[8][0],y[8][1]),y[8][2]) << endl;
    cout << y[8][2]/ max(max(y[8][0],y[8][1]),y[8][2]) << endl;

    return 0;
}
```

Determining the highest-modulus eigenvalue and corresponding eigenvector of the given matrix:

$$\begin{bmatrix} 1 & 5 & 4 \\ 3 & 2 & 3 \\ 7 & 6 & 1 \end{bmatrix}$$

Input to the program:

```
1 5 4
3 2 3
7 6 1
```

The input/output from the console:

```
3x3 matrix A:
1 5 4
3 2 3
7 6 1

Highest modulus eigenvalue: 10.2919

Highest corresponding eigenvector:
4.14513e+007
4.94071e+007
3.37484e+007

After normalization:
0.838975
1
0.683068
```

Obtained highest modulus eigenvalue: **10.2919**

Obtained corresponding eigenvector:

$$\begin{bmatrix} 4.14513^{7} \\ 4.94071^{7} \\ 3.37484^{7} \end{bmatrix}$$