2.13. EIGENVALUES AND EIGENVECTORS OF MATRICES

Let's consider the matrix $A_{n\times n}$ in an *n*-dimensional real space R^n of vectors $x=(x_1, x_2, \ldots, x_n)^T$.

Definition. The *eigenvector* *x* of the matrix *A* is a nonzero vector ($x\neq\vartheta$) satisfying the equality:

$$Ax=\lambda x,$$

where λ is the *eigenvalue* of the matrix *A* corresponding to the eigenvector *x* under consideration.

The eigenvalues of the matrix *A* with real elements can be real different, real multiple, complex pairwise-conjugate, and complex multiple.

The classical analytical method for finding eigenvalues and eigenvectors is known and consists of the following. For the homogeneous SLAE $(A-\lambda E)x=\vartheta$ nonzero solutions ($x\neq\vartheta$, these are the solutions that are found) take place only if $\det(A-\lambda E)=0$.

Definition. The equation $\det(A-\lambda E)=0$ is called the *characteristic equation*, and the expression in the left part is called the *characteristic polynomial*.

Then, using any method, the solutions $\lambda_1,\lambda_2,\ldots,\lambda_n$ of the obtained *n*-th power algebraic equation are found (suppose that they are real and different); after that, linearly independent eigenvectors $x^j, j=1,\ldots,n$ corresponding to eigenvalues $\lambda_j, j=1,\ldots,n$ are obtained by solving for various eigenvalues $\lambda_j, j=1,\ldots,n$ a homogeneous SLAE $(A-\lambda_j E)x^j=\vartheta, j=1,\ldots,n$. Usually, the obtained eigenvectors $x^j, j=1,\ldots,n$ are then normalized for unambiguity.

Pairwise different eigenvalues correspond to linearly independent eigenvectors; the *k*-fold root of the characteristic equation constructed for a given matrix $A_{n\times n}$ corresponds to no more than *k* linearly independent eigenvectors.

Example. Analytically calculate eigenvalues and eigenvectors of the matrix $A=\begin{pmatrix}1 & 2\\ 2 & -2\end{pmatrix}$.

Let's write the characteristic equation $\det(A-\lambda E)=0$:

$$\det(A-\lambda E)=\det\left[\begin{pmatrix}1 & 2\\ 2 & -2\end{pmatrix}-\begin{pmatrix}\lambda & 0\\ 0 & \lambda\end{pmatrix}\right]=\det\begin{pmatrix}1-\lambda & 2\\ 2 & -2-\lambda\end{pmatrix}=0,$$

$$(1-\lambda)(-2-\lambda)-2\cdot 2=0,$$

$$\lambda^2+\lambda-6=0,$$

$$\lambda_1=2,\lambda_2=-3.$$

The eigenvalues $\lambda_1=2,\lambda_2=-3$ correspond to the following normalized (according to the Euclidean norm) eigenvectors (solutions of homogeneous SLAEs $(A-\lambda_j E)x^j=\vartheta, j=1,\ldots,n$):

$$\begin{pmatrix}1-\lambda_1 & 2\\ 2 & -2-\lambda_1\end{pmatrix}\begin{pmatrix}x_1^1\\ x_2^1\end{pmatrix}=\begin{pmatrix}-1 & 2\\ 2 & -4\end{pmatrix}\begin{pmatrix}x_1^1\\ x_2^1\end{pmatrix}=\begin{pmatrix}0\\ 0\end{pmatrix},\ -x_1^1+2x_2^1=0,\ \begin{pmatrix}x_1^1\\ x_2^1\end{pmatrix}=\begin{pmatrix}\frac{2}{\sqrt{5}}\\ \frac{1}{\sqrt{5}}\end{pmatrix};$$

$$\begin{pmatrix}1-\lambda_2 & 2\\ 2 & -2-\lambda_2\end{pmatrix}\begin{pmatrix}x_1^2\\ x_2^2\end{pmatrix}=\begin{pmatrix}4 & 2\\ 2 & 1\end{pmatrix}\begin{pmatrix}x_1^2\\ x_2^2\end{pmatrix}=\begin{pmatrix}0\\ 0\end{pmatrix},\ 2x_1^2+x_2^2=0,\ \begin{pmatrix}x_1^2\\ x_2^2\end{pmatrix}=\begin{pmatrix}-\frac{1}{\sqrt{5}}\\ \frac{2}{\sqrt{5}}\end{pmatrix}.$$

It is easy to see that the scalar product of the obtained vectors $x^1\cdot x^2=\frac{2}{\sqrt{5}}\cdot\frac{1}{\sqrt{5}}+\frac{-1}{\sqrt{5}}\cdot\frac{2}{\sqrt{5}}=0$, i.e. the eigenvectors are orthogonal.

If the amount of linearly independent eigenvectors of the matrix $A_{n\times n}$ coincides with the dimension of space R^n, then they can be taken as a new basis in which the matrix $A_{n\times n}$ takes a diagonal form $\Lambda=U^{-1}\cdot A\cdot U$, on the main diagonal of which there are the eigenvalues, and the columns of the transformation matrix U are the eigenvectors of the matrix A.

Definition. The matrices Λ and A satisfying the stated equality $\Lambda=U^{-1}\cdot A\cdot U$ are called *similar*.

The eigenvalues and the determinant of similar matrices Λ and A coincide. The eigenvectors of the matrix A are related to the eigenvectors of the matrix Λ by the relation $x=\Lambda y$.

The symmetric matrix A ($A = A^T$) has a full spectrum $\lambda_j, j = 1,\ldots,n$ of real eigenvalues; exactly k linearly independent eigenvectors correspond to the k-fold root of the characteristic equation of the symmetric matrix.

So, the symmetric matrix A always has exactly n orthogonal eigenvectors; by taking them as a new basis (i.e. constructing the transformation matrix U, taking the coordinate columns of the eigenvectors as its columns), we can transform the symmetric matrix to the diagonal form with the help of the transformation $\Lambda = U^{-1} \cdot A \cdot U$.

Definition. The matrix U for which $U^{-1} = U^T$ is true is called *orthogonal*.

All eigenvalues of the orthogonal matrix are equal in modulus to one, the rows (columns) of the orthogonal matrix are pairwise orthogonal, the sums of the squares of the elements of each row (column) of the orthogonal matrix are equal to one, the determinant of the orthogonal matrix is ± 1. If the matrix U is orthogonal, then the matrix U^{-1} is also orthogonal.

The transformation matrix U is orthogonal for the symmetric matrix A and, therefore, its transformation to a diagonal form looks as $\Lambda = U^T \cdot A \cdot U$.

Example. Let's check the similarity of matrices $A = \begin{pmatrix} 1 & 2 \\ 2 & -2 \end{pmatrix}$ and $\Lambda = \begin{pmatrix} 2 & 0 \\ 0 & -3 \end{pmatrix}$.

Let's compose the transformation matrix U taking the coordinate columns of the eigenvectors of the matrix A as its columns: $U = \begin{pmatrix} \frac{2}{\sqrt{5}} & -\frac{1}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{pmatrix}$.

Indeed:

$$U^T \cdot A \cdot U = \begin{pmatrix} \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \\ -\frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 \\ 2 & -2 \end{pmatrix} \cdot \begin{pmatrix} \frac{2}{\sqrt{5}} & -\frac{1}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{pmatrix} =$$

$$= \begin{pmatrix} \frac{4}{\sqrt{5}} & \frac{2}{\sqrt{5}} \\ \frac{3}{\sqrt{5}} & -\frac{6}{\sqrt{5}} \end{pmatrix} \cdot \begin{pmatrix} \frac{2}{\sqrt{5}} & -\frac{1}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{pmatrix} = \begin{pmatrix} \frac{10}{5} & \frac{0}{5} \\ \frac{0}{5} & \frac{-15}{5} \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 0 & -3 \end{pmatrix} = \Lambda .$$

2.14. JACOBI ROTATION METHOD

The Jacobi rotation method is applicable only for real symmetric matrices A_{nxn} ($A = A^T$) and solves the full problem of eigenvalues and eigenvectors of such matrices (in the sense that all desired values and vectors are found). It is based on finding using iterative procedures the matrix U in the similarity transformation $\Lambda = U^{-1} \cdot A \cdot U$ that preserves its eigenvalues; and since the similarity transformation matrix U is orthogonal ($U^{-1} = U^T$) for symmetric matrices A, $\Lambda = U^T \cdot A \cdot U$, where Λ is the diagonal matrix with the eigenvalues on the main diagonal:

$$\Lambda = \begin{pmatrix} \lambda_1 & \dots & 0 \\ \dots & \ddots & \dots \\ 0 & \dots & \lambda_n \end{pmatrix}.$$

Suppose we have a symmetric matrix A. It is required to calculate with an accuracy of ε all its eigenvalues and corresponding eigenvectors. The iterative algorithm of the rotation method is as follows:

$$\begin{cases} A^{(0)} = A, V^{(0)} = E \\ A^{(1)} = U^{(0)T} \cdot A^{(0)} \cdot U^{(0)}, V^{(1)} = V^{(0)} \cdot U^{(0)} \\ A^{(2)} = U^{(1)T} \cdot A^{(1)} \cdot U^{(1)}, V^{(2)} = V^{(1)} \cdot U^{(1)} \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots \\ A^{(k)} = U^{(k-1)T} \cdot A^{(k-1)} \cdot U^{(k-1)}, V^{(k)} = V^{(k-1)} \cdot U^{(k-1)} \end{cases},$$

where k is the number of iterations performed.

Suppose we know the matrix $A^{(k)}$ at the k-th iteration ($A^{(0)} = A$ for $k = 0$).

First, the maximum in modulus non-diagonal element $a_{ij}^{(k)}$ of the matrix $A^{(k)}$ is chosen.

Second, the goal is set to find such an orthogonal matrix $U^{(k)}$ that will zero out the element $a_{ij}^{(k+1)}$ of the matrix $A^{(k+1)}$ as a result of similarity transformation $A^{(k+1)} = U^{(k)T} \cdot A^{(k)} \cdot U^{(k)}$. The elementary rotation matrix (after which the method is named), which rotates by the angle of rotation $\varphi^{(k)}$ to be determined in the plane (x_i, x_j) in the n-dimensional real space R^n, is selected as an orthogonal matrix. The elementary rotation matrix has the following form in the space R^n:

$$U^{(k)} = \begin{array}{c} \begin{matrix} & & & i & & & j & & \end{matrix} \\ \begin{pmatrix} 1 & & & \vdots & & & \vdots & & \\ & \ddots & & \vdots & & & \vdots & 0 & \\ & & 1 & \vdots & & & \vdots & & \\ \cdots & \cdots & \cdots & \cos\varphi^{(k)} & \cdots & -\sin\varphi^{(k)} & \cdots & \cdots & i \\ & & & \vdots & 1 & & \vdots & & \\ & & & \vdots & & \ddots & \vdots & & \\ & & & \vdots & & & 1 & & \\ \cdots & \cdots & \cdots & \sin\varphi^{(k)} & \cdots & \cos\varphi^{(k)} & \cdots & \cdots & j \\ & & & \vdots & & & \vdots & 1 & \\ & 0 & & \vdots & & & \vdots & \ddots & \\ & & & \vdots & & & \vdots & & 1 \end{pmatrix} \end{array} .$$

The element $u_{ij}^{(k)} = -\sin\varphi^{(k)}$ is located at the intersection of the *i*-th row and the *j*-th column in the rotation matrix $U^{(k)}$; the element $u_{ji}^{(k)} = \sin\varphi^{(k)}$ is located symmetrically with respect to the main diagonal (*j*-th row, *i*-th column); the diagonal elements $u_{ii}^{(k)}$ and $u_{jj}^{(k)}$ are equal respectively to $u_{ii}^{(k)} = u_{jj}^{(k)} = \cos\varphi^{(k)}$; other diagonal elements are equal to one $u_{mm}^{(k)} = 1, m = 1, \ldots, n, m \neq i, m \neq j$; other elements in the rotation matrix are equal to zero.

The rotation angle $\varphi^{(k)}$ is determined from the condition of zeroing out the element $a_{ij}^{(k+1)} = 0$ of the matrix $A^{(k+1)}$. It is easy to express it if we consider the rotation procedure only in the plane (x_i, x_j) of the space R^n:

$$\begin{pmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{pmatrix} \cdot \begin{pmatrix} a_{ii} & a_{ij} \\ a_{ji} & a_{jj} \end{pmatrix} \cdot \begin{pmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{pmatrix} =$$

$$= \begin{pmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{pmatrix} \cdot \begin{pmatrix} a_{ii}\cos\varphi + a_{ij}\sin\varphi & -a_{ii}\sin\varphi + a_{ij}\cos\varphi \\ a_{ji}\cos\varphi + a_{jj}\sin\varphi & -a_{ji}\sin\varphi + a_{jj}\cos\varphi \end{pmatrix},$$

$$-a_{ii}\sin\varphi\cos\varphi + a_{ij}\cos^2\varphi - a_{ji}\sin^2\varphi + a_{jj}\sin\varphi\cos\varphi = 0,$$

$$a_{ij}(\cos^2\varphi - \sin^2\varphi) + (a_{jj} - a_{ii})\sin\varphi\cos\varphi = 0,$$

$$a_{ij}\cos 2\varphi + (a_{jj} - a_{ii})\frac{\sin 2\varphi}{2} = 0,\ 2a_{ij}\cos 2\varphi = (a_{ii} - a_{jj})\sin 2\varphi,$$

$$\begin{cases} \tan 2\varphi = \dfrac{2a_{ij}}{(a_{ii}-a_{jj})}, a_{ii} \neq a_{jj} \\ \cos 2\varphi = 0, a_{ii} = a_{jj} \end{cases}, \begin{cases} \varphi = \dfrac{1}{2}\arctan\dfrac{2a_{ij}}{(a_{ii}-a_{jj})}, a_{ii} \neq a_{jj} \\ \varphi = \dfrac{\pi}{4}, a_{ii} = a_{jj} \end{cases}.$$

So, the rotation angle $\varphi^{(k)} = \frac{1}{2}\arctan\frac{2a_{ij}^{(k)}}{a_{ii}^{(k)} - a_{jj}^{(k)}}$; and if $a_{ii}^{(k)} = a_{jj}^{(k)}$, then, in this case, $\varphi^{(k)} = \frac{\pi}{4}$.

As a result, the new matrix $A^{(k+1)} = U^{(k)T} \cdot A^{(k)} \cdot U^{(k)}$ is constructed in which the element $a_{ij}^{(k+1)} = a_{ji}^{(k+1)} \approx 0$. Obviously, the matrix $A^{(k+1)}$ is also symmetric.

Moreover, it can be seen that in one iteration only the elements of the *i*-th and *j*-th rows and columns change in the matrices $A^{(k+1)}$ and $V^{(k+1)}$, which, if desired, allows to significantly optimize the processes of multiplication of matrices $A^{(k+1)} = U^{(k)T} \cdot A^{(k)} \cdot U^{(k)}$ and $V^{(k+1)} = V^{(k)} \cdot U^{(k)}$ (from around n^3 arithmetic operations up to $4n$).

The condition of the smallness of the mean-square of the non-diagonal elements of the matrix $A^{(k+1)}$ is used as a criterion for iterations termination:

$$\varepsilon^{(k+1)} = t\left(A^{(k+1)}\right) = \left(\sum_{l,m;l<m}\left(a_{lm}^{(k+1)}\right)^2\right)^{1/2}.$$

If $\varepsilon^{(k+1)} = t\left(A^{(k+1)}\right) > \varepsilon$, then the iterative process continues. If $\varepsilon^{(k+1)} = t\left(A^{(k+1)}\right) \leq \varepsilon$, then the iterative process stops and the diagonal elements of the matrix $A^{(k+1)}$ are taken as the desired eigenvalues: $\lambda_1 \approx a_{11}^{(k+1)}, \lambda_2 \approx a_{22}^{(k+1)}, ..., \lambda_n \approx a_{nn}^{(k+1)}$.

The resulting matrix $A^{(k+1)}$ can be presented as

$$A^{(k+1)} = U^{(k)T} \cdot A^{(k)} \cdot U^{(k)} = U^{(k)T}U^{(k-1)T}...U^{(0)T} \cdot A^{(0)} \cdot U^{(0)}U^{(1)}...U^{(k)} = V^{(k+1)T} \cdot A^{(0)} \cdot V^{(k+1)},$$

which implies that the coordinate columns for the eigenvectors of the matrix *A* in a unit basis will be the columns of the matrix $V^{(k+1)} = U^{(0)}U^{(1)}...U^{(k)}$, i.e. $v^1 = \left(v_{11}, v_{21}, ..., v_{n1}\right)^T$, $v^2 = \left(v_{12}, v_{22}, ..., v_{n2}\right)^T$, …, $v^n = \left(v_{1n}, v_{2n}, ..., v_{nn}\right)^T$ and these eigenvectors will be orthogonal to each other:

$$\left(v^l, v^m\right) \approx 0, \; l \neq m.$$

Example. Using the Jacobi rotation method, calculate the eigenvalues and eigenvectors of the matrix $A=\begin{pmatrix}4 & 2 & 1\\ 2 & 5 & \mathbf{3}\\ 1 & 3 & 6\end{pmatrix}=A^{(0)}$ with an accuracy of $\varepsilon=0.2$.

Let's note that $\varepsilon^{(0)}=t\left(A^{(0)}\right)=\left(\sum\limits_{l,m;l<m}\left(a_{lm}^{(0)}\right)^2\right)^{1/2}=(2^2+1^2+3^2)^{1/2}=\sqrt{14}=3.742>\varepsilon$.

Let's start the iterative process. Iteration $k=1$. Let's select the maximum in modulus non-diagonal element of the matrix $A^{(0)}$, i.e. we should find such $a_{ij}^{(0)}$ that $\left|a_{ij}^{(0)}\right|=\max\limits_{l<m}\left|a_{lm}^{(0)}\right|$. It is the element $a_{23}^{(0)}=3$.

Let's find the rotation matrix corresponding to this element:

$$\varphi^{(0)}=\frac{1}{2}\arctan\frac{2\cdot 3}{5-6}=-0.703,$$

$$U^{(0)}=\begin{pmatrix}1 & 0 & 0\\ 0 & \cos\varphi^{(0)} & -\sin\varphi^{(0)}\\ 0 & \sin\varphi^{(0)} & \cos\varphi^{(0)}\end{pmatrix}=\begin{pmatrix}1 & 0 & 0\\ 0 & 0.763 & 0.646\\ 0 & -0.646 & 0.763\end{pmatrix}.$$

Calculating the matrix $A^{(1)}$: $A^{(1)}=U^{(0)T}A^{(0)}U^{(0)}=\begin{pmatrix}4 & 0.880 & \mathbf{2.056}\\ 0.880 & 2.459 & 0.000\\ 2.056 & 0.000 & 8.541\end{pmatrix}$.

In the resulting matrix, up to rounding errors, the element $a_{23}^{(1)}=0$. The sum of squares of the matrix non-diagonal elements is as follows:

$$\varepsilon^{(1)}=t\left(A^{(1)}\right)=\left(\sum_{l,m;l<m}\left(a_{lm}^{(1)}\right)^2\right)^{1/2}=(0.880^2+2.056^2+0^2)^{1/2}=2.236>\varepsilon,$$

therefore, the iterative process must be continued.

Iteration $k=2$. Let's find $a_{13}^{(1)}=2.056,\left|a_{13}^{(1)}\right|=\max\limits_{l,m;\,l<m}\left|a_{lm}^{(1)}\right|$.

$$\varphi^{(1)}=\frac{1}{2}\arctan\frac{2\cdot 2.056}{4-8.541}=-0.368,$$

$$U^{(1)}=\begin{pmatrix}\cos\varphi^{(1)} & 0 & -\sin\varphi^{(1)}\\ 0 & 1 & 0\\ \sin\varphi^{(1)} & 0 & \cos\varphi^{(1)}\end{pmatrix}=\begin{pmatrix}0.933 & 0 & 0.360\\ 0 & 1 & 0\\ -0.360 & 0 & 0.933\end{pmatrix}.$$

Calculating the matrix $A^{(2)}$: $A^{(2)} = U^{(1)T} A^{(1)} U^{(1)} = \begin{pmatrix} 3.208 & \mathbf{0.821} & 0.000 \\ 0.821 & 2.459 & 0.316 \\ 0.000 & 0.316 & 9.334 \end{pmatrix}$,

$$\varepsilon^{(2)} = t\left(A^{(2)}\right) = \left(\sum_{l,m;l<m} \left(a_{lm}^{(2)}\right)^2 \right)^{1/2} = (0.821^2 + 0^2 + 0.316^2)^{1/2} = 0.880 > \varepsilon .$$

Iteration $k = 3$. Let's find $a_{12}^{(2)} = 0.821, \left|a_{12}^{(2)}\right| = \max\limits_{l,m;\, l<m} \left|a_{lm}^{(2)}\right|$.

$$\varphi^{(2)} = \frac{1}{2} \arctan \frac{2 \cdot 0.821}{3.208 - 2.459} = 0.571,$$

$$U^{(2)} = \begin{pmatrix} \cos\varphi^{(2)} & -\sin\varphi^{(2)} & 0 \\ \sin\varphi^{(2)} & \cos\varphi^{(2)} & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0.841 & -0.541 & 0 \\ 0.541 & 0.841 & 0 \\ 0 & 0 & 1 \end{pmatrix} .$$

Calculating the matrix $A^{(3)}$: $A^{(3)} = U^{(2)T} A^{(2)} U^{(2)} = \begin{pmatrix} 3.735 & 0.000 & 0.171 \\ 0.000 & 1.931 & 0.266 \\ 0.171 & 0.266 & 9.334 \end{pmatrix}$,

$$\varepsilon^{(3)} = t\left(A^{(3)}\right) = \left(\sum_{l,m;l<m} \left(a_{lm}^{(3)}\right)^2 \right)^{1/2} = (0^2 + 0.171^2 + 0.266^2)^{1/2} = 0.316 > \varepsilon .$$

Iteration $k = 4$. Let's find $a_{23}^{(3)} = 0.266, \left|a_{23}^{(3)}\right| = \max\limits_{l,m;\, l<m} \left|a_{lm}^{(3)}\right|$.

$$\varphi^{(3)} = \frac{1}{2} \arctan \frac{2 \cdot 0.266}{1.931 - 9.334} = -0.036 ,$$

$$U^{(3)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi^{(3)} & -\sin\varphi^{(3)} \\ 0 & \sin\varphi^{(3)} & \cos\varphi^{(3)} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.999 & 0.036 \\ 0 & -0.036 & 0.999 \end{pmatrix} .$$

Calculating the matrix $A^{(4)}$: $A^{(4)} = U^{(3)T} A^{(3)} U^{(3)} = \begin{pmatrix} 3.735 & -0.006 & 0.171 \\ -0.006 & 1.921 & 0.000 \\ 0.171 & 0.000 & 9.343 \end{pmatrix}$,

$$\varepsilon^{(4)} = t\left(A^{(4)}\right) = \left(\sum_{l,m;l<m} \left(a_{lm}^{(4)}\right)^2 \right)^{1/2} = ((-0.006)^2 + 0.171^2 + 0^2)^{1/2} = 0.171 < \varepsilon .$$

Now the iterative process can be stopped.

The diagonal elements of the matrix $A^{(4)}$ can be taken as the desired eigenvalues: $\lambda_1 \approx 3.735,\ \lambda_2 \approx 1.921,\ \lambda_3 \approx 9.343$.

The eigenvectors are determined from the product:

$$V^{(4)} = U^{(0)}U^{(1)}U^{(2)}U^{(3)} = \begin{pmatrix} 0.785 & -0.517 & 0.341 \\ 0.217 & 0.745 & 0.630 \\ -0.580 & -0.421 & 0.697 \end{pmatrix}, \text{ so:}$$

$$v^1 = \begin{pmatrix} 0.785 \\ 0.217 \\ -0.580 \end{pmatrix}, \; v^2 = \begin{pmatrix} -0.517 \\ 0.745 \\ -0.421 \end{pmatrix}, \; v^3 = \begin{pmatrix} 0.341 \\ 0.630 \\ 0.697 \end{pmatrix}.$$

The resulting eigenvectors are orthogonal within the specified accuracy, i.e. $\left(v^1, v^2\right) = \left(v^1, v^3\right) = \left(v^2, v^3\right) \approx 0$.

The resulting eigenvalues satisfy the equation $Ax = \lambda x$ within the specified accuracy:

$$Av^1 - \lambda_1 v^1 = \begin{pmatrix} 0.062 \\ 0.103 \\ 0.122 \end{pmatrix}, \; Av^2 - \lambda_2 v^2 = \begin{pmatrix} -0.005 \\ -0.001 \\ 0.004 \end{pmatrix}, \; Av^3 - \lambda_3 v^3 = \begin{pmatrix} 0.134 \\ 0.037 \\ -0.099 \end{pmatrix}.$$

2.15. PROGRAM #04

Below is a proposed variant of the program algorithm for finding the eigenvalues and eigenvectors of a symmetric matrix using the Jacobi rotation method.

```
ALGORITHM “Jacobi rotation method”
INPUT     n, a[n][n], e
OUTPUT    i, j, p, l, m, f, g, h, u[n][n], b[n][n], w[n], v[n][n], k
BEGIN
    CYCLE “Rows” FOR i FROM 1 TO n BY 1
        CYCLE “Columns” FOR j FROM 1 TO n BY 1
            IF (i=j) v[i][j]:=1 ELSE v[i][j]:=0
    f:=0
    CYCLE “Rows” FOR i FROM 1 TO n BY 1
        CYCLE “Columns” FOR j FROM i+1 TO n BY 1
            f:=f+a[i][j]*a[i][j]
    f:=ROOT(f)
    k:=0
```

```
CYCLE “Iteration” WHILE (f>e)
    g:=0, l=1, m=2
    CYCLE “Rows” FOR i FROM 1 TO n BY 1
        CYCLE “Columns” FOR j FROM i+1 TO n BY 1
            IF (a[i][j]>g) g:=a[i][j], l=i, m=j
            IF (-a[i][j]>g) g:=-a[i][j], l=i, m=j
    CYCLE “Rows” FOR i FROM 1 TO n BY 1
        CYCLE “Columns” FOR j FROM 1 TO n BY 1
            IF (i=j) u[i][j]:=1 ELSE u[i][j]:=0
    IF (a[l][l]=a[m][m]) h:=PI()/4
    ELSE h:=ARCTANGENT(2*a[l][m]/(a[l][l]-a[m][m]))/2
    u[l][l]:=COSINE(h)
    u[l][m]:=-SINE(h)
    u[m][l]:=SINE(h)
    u[m][m]:=COSINE(h)
    CYCLE “Rows” FOR i FROM 1 TO n BY 1
        CYCLE “Columns” FOR j FROM 1 TO n BY 1
        IF (i=l)OR(i=m)OR(j=l)OR(j=m)
            b[i][j]:=0
            CYCLE “Element” FOR p FROM 1 TO n BY 1
                b[i][j]:=b[i][j]+u[p][i]*a[p][j]
    CYCLE “Rows” FOR i FROM 1 TO n BY 1
        CYCLE “Columns” FOR j FROM 1 TO n BY 1
        IF (i=l)OR(i=m)OR(j=l)OR(j=m)
            a[i][j]:=0
            CYCLE “Element” FOR p FROM 1 TO n BY 1
                a[i][j]:=a[i][j]+b[i][p]*u[p][j]
    CYCLE “Rows” FOR i FROM 1 TO n BY 1
        CYCLE “Columns” FOR j FROM 1 TO n BY 1
        IF (i=l)OR(i=m)OR(j=l)OR(j=m)
```

```
                    b[i][j]:=0
                    CYCLE “Element” FOR p FROM 1 TO n BY 1
                        b[i][j]:=b[i][j]+v[i][p]*u[p][j]
            CYCLE “Rows” FOR i FROM 1 TO n BY 1
                CYCLE “Columns” FOR j FROM 1 TO n BY 1
                IF (i=l)OR(i=m)OR(j=l)OR(j=m)
                    v[i][j]:=b[i][j]
            f:=0
            CYCLE “Rows” FOR i FROM 1 TO n BY 1
                CYCLE “Columns” FOR j FROM i+1 TO n BY 1
                    f:=f+a[i][j]*a[i][j]
            f:=ROOT(f)
            k:=k+1
        CYCLE “Rows” FOR i FROM 1 TO n BY 1
            w[i]:=a[i][i]
        PRINT w, v, k
END
```

2.16. POWER ITERATION METHOD

The Jacobi rotation method described above solves the full problem of eigenvalues and eigenvectors of matrices (symmetrical). However, it is often not necessary to find all eigenvalues (spectrum) and all eigenvectors, but the goal is to find only the maximum in modulus eigenvalue of the matrix (spectral radius) and the corresponding eigenvector. This problem is called the partial problem of eigenvalues and eigenvectors. The power iteration method (or power method) is intended to solve it.

Suppose we have an arbitrary matrix A and suppose its eigenvalues are ordered by absolute values: $|\lambda_1| > |\lambda_2| \geq \ldots \geq |\lambda_n|$. Then, having chosen some initial vector $y^{(0)}$, for example, the vector $y^{(0)} = (11\ldots1)^T$, the components of which are equal to one, we can construct the following iterative process to find λ_1:

$$y^{(k)} = Ay^{(k-1)}, \lambda_1^{(k)} = \frac{y_j^{(k)}}{y_j^{(k-1)}},$$

where $y_j^{(k-1)}$, $y_j^{(k)}$ are the corresponding components of the vectors $y^{(k-1)}$, $y^{(k)}$. Herewith, any number from the range $j = 1,\ldots,n$ can be used as the number *j*.

Since the vector $y^{(k)}$ at the *k*-th iteration can be presented as $y^{(k)} = Ay^{(k-1)} = \ldots = A^k y^{(0)}$, the iterative process under consideration is called the power iteration method. Due to its construction, the method is used primarily for sparse matrices.

When the condition $|\lambda_1| > |\lambda_2| \geq \ldots \geq |\lambda_n|$ is fulfilled, the iterative process converges to the desired eigenvalue λ_1 and the corresponding eigenvector $y^{(k)}$, and the convergence rate is determined by the ratio $\frac{|\lambda_2|}{|\lambda_1|}$ (the smaller it is, the higher the convergence rate is).

The following condition is used as a criterion for the termination of calculations:

$\varepsilon^{(k)} = \left|\lambda_1^{(k)} - \lambda_1^{(k-1)}\right| \leq \varepsilon$, where ε is the calculation accuracy set by the calculator.

The presented algorithm has a significant practical drawback associated with a great change in the components of the iterated vector $y^{(k)}$ during the iterative process. It is obvious that as $\left|\frac{y_j^{(k)}}{y_j^{(k-1)}}\right| \approx |\lambda_1|$, it results in an unlimited increase (if $|\lambda_1| > 1$) or decrease (if $|\lambda_1| < 1$) of the components of $y^{(k)}$ as far as the number of iterations *k* increases. To avoid this, when carrying out computer calculations, the power iteration method with the normalization of the iterated vector is usually used. For this purpose, the algorithm is modified as follows:

$z^{(k)} = Ay^{(k-1)}, \lambda_1^{(k)} = \frac{z_j^{(k)}}{y_j^{(k-1)}}, y^{(k)} = \frac{z^{(k)}}{\left\|z^{(k)}\right\|}$; herewith, a vector with a unit norm is taken as an initial approximation $y^{(0)}$.

Example. Calculate the spectral radius of the matrix $A=\begin{pmatrix}4 & 2 & 1\\ 2 & 5 & 3\\ 1 & 3 & 6\end{pmatrix}$ using the power method (with normalization of the iterated vector) with an accuracy of $\varepsilon = 0.2$.

Let's take $y^{(0)} = (1 \quad 1 \quad 1)^T$ as the initial approximation of the eigenvector. Let's implement the iterative process of the power iteration method using the norm $\|\cdot\|_c$ and assuming $j=1$: $z^{(k)} = Ay^{(k-1)}, \lambda_1^{(k)} = \frac{z_1^{(k)}}{y_1^{(k-1)}}, y^{(k)} = \frac{z^{(k)}}{\|z^{(k)}\|_c}$.

$$z^{(1)} = Ay^{(0)} = (7 \quad 10 \quad 10)^T,\ \lambda_1^{(1)} = \frac{z_1^{(1)}}{y_1^{(0)}} = \frac{7}{1} = 7,\ y^{(1)} = \frac{z^{(1)}}{\|z^{(1)}\|_c} = (0.7 \quad 1 \quad 1)^T;$$

$$z^{(2)} = Ay^{(1)} = (5.8 \quad 9.4 \quad 9.7)^T,\ \lambda_1^{(2)} = \frac{z_1^{(2)}}{y_1^{(1)}} = 8.286,$$

$$y^{(2)} = \frac{z^{(2)}}{\|z^{(2)}\|_c} = (0.598 \quad 0.969 \quad 1)^T,\ \varepsilon^{(2)} = |\lambda_1^{(2)} - \lambda_1^{(1)}| = 1.286 > \varepsilon;$$

$$z^{(3)} = Ay^{(2)} = (5.330 \quad 9.041 \quad 9.505)^T,\ \lambda_1^{(3)} = \frac{z_1^{(3)}}{y_1^{(2)}} = 8.914,$$

$$y^{(3)} = \frac{z^{(3)}}{\|z^{(3)}\|_c} = (0.561 \quad 0.951 \quad 1)^T,\ \varepsilon^{(3)} = |\lambda_1^{(3)} - \lambda_1^{(2)}| = 0.628 > \varepsilon;$$

$$z^{(4)} = Ay^{(3)} = (5.145 \quad 8.877 \quad 9.414)^T,\ \lambda_1^{(4)} = \frac{z_1^{(4)}}{y_1^{(3)}} = 9.146,$$

$$y^{(4)} = \frac{z^{(4)}}{\|z^{(4)}\|_c} = (0.547 \quad 0.943 \quad 1)^T,\ \varepsilon^{(4)} = |\lambda_1^{(4)} - \lambda_1^{(3)}| = 0.262 > \varepsilon;$$

$$z^{(5)} = Ay^{(4)} = (5.072 \quad 8.808 \quad 9.375)^T,\ \lambda_1^{(5)} = \frac{z_1^{(5)}}{y_1^{(4)}} = 9.280,$$

$$y^{(5)} = \frac{z^{(5)}}{\|z^{(5)}\|_c} = (0.541 \quad 0.939 \quad 1)^T,\ \varepsilon^{(5)} = |\lambda_1^{(5)} - \lambda_1^{(4)}| = 0.104 < \varepsilon.$$

So, the value $\lambda_1^{(5)} = 9.280$, obtained at the fifth iteration, satisfies the specified accuracy and can be taken as an approximate value λ_1, and the desired value of the spectral radius is $\rho(A) = \max_i |\lambda_i| = |\lambda_1| \approx |\lambda_1^{(5)}| = 9.280$.

Let's note that this value $\lambda_1^{(5)}$ with an accuracy of $\varepsilon = 0.2$ coincides with the value $\lambda_3 \approx 9.343$ calculated by the Jacobi rotation method, in the same way as the corresponding eigenvector $y^{(5)} = (0.541 \quad 0.939 \quad 1)^T$ represents the vector $v^3 = (0.341 \quad 0.630 \quad 0.697)^T$ after renormalization.

2.17. PROGRAM #05

Below is a proposed variant of the program algorithm for finding the maximum in modulus eigenvalue of the matrix and the corresponding eigenvector using the power iteration method.

```
ALGORITHM “Power iteration method”
INPUT     n, a[n][n], e
OUTPUT    i, j, f, r, z[n], w, y[n], k
BEGIN
    CYCLE “Rows” FOR i FROM 1 TO n BY 1
        y[i]:=1
    f:=2*e
    k:=0
    CYCLE “Iteration” WHILE (f>e)
        r:=0
        CYCLE “Rows” FOR i FROM 1 TO n BY 1
            z[i]:=0
            CYCLE “Element” FOR j FROM 1 TO n BY 1
                z[i]:=z[i]+a[i][j]*y[j]
            IF (z[i]>r) r:=z[i]
            IF (-z[i]>r) r:=-z[i]
        IF (k>0) f:=w ELSE f:=2*e+z[1]/y[1]
        w:=z[1]/y[1]
        CYCLE “Rows” FOR i FROM 1 TO n BY 1
            y[i]:=z[i]/r
```

```
            f:=w-f
            IF (f<0) f:=-f
            k:=k+1
        PRINT w, y, k
END
```