Diagonalizability

- A matrix $\mathbf{A}$ is referred to as *diagonalizable* if there exists a similarity transformation $\Lambda = \mathbf{X}^{-1}\mathbf{A}\mathbf{X}$ such that $\Lambda = \mathrm{diag}(\lambda_1,\lambda_2,\ldots,\lambda_n)$ is a diagonal matrix. Then $\lambda_1, \lambda_2,\ldots,\lambda_n$ are the *eigenvalues* and the *columns* of the matrix $\mathbf{X}$ contain the corresponding *eigenvectors* of the matrix $\mathbf{A}$. Then, $\mathbf{A}=\mathbf{X}\Lambda\mathbf{X}^{-1}$
- A matrix $\mathbf{A}$ is referred to as *unitarily diagonalizable* if there exists a *unitary* matrix $\mathbf{U}$ such that $\Lambda = \mathbf{U}^{\dagger}\mathbf{A}\mathbf{U}$, where $\Lambda = \mathrm{diag}(\lambda_1,\lambda_2,\ldots,\lambda_n)$ is a diagonal matrix. Then $\lambda_1, \lambda_2,\ldots,\lambda_n$ are the *eigenvalues* and the *columns* of the matrix $\mathbf{U}$ contain the corresponding *eigenvectors* of the matrix $\mathbf{A}$. Then, $\mathbf{A} = \mathbf{U}\Lambda\mathbf{U}^{\dagger}$
- The eigenvalues of a matrix $\mathbf{A}$ can be found by solving its *characteristic equation:*

$$\det(\mathbf{A} - \lambda\mathbf{I}) = 0$$

  However, it is impractical even for middle-size matrices.
- *Eigenvalue shifting*: adding the same value a to all diagonal elements of matrix $\mathbf{A}$, its eigenvectors do not change; the eigenvalues are shifted by a:

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x} \Leftrightarrow (\mathbf{A} + a\mathbf{I})\mathbf{x} = (\lambda+a)\mathbf{x}$$

Eigenvectors and -functions in some special cases

- General matrices:
  - are diagonalizable if all the eigenvalues are distinct;
  - may or may not be diagonalizable if some eigenvalues are coincident (*degenerate*).
- Hermitian and symmetric matrices:
  - are always *unitarily diagonalizable*: $\mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\dagger}$;
  - eigenvalues are always *real*;
  - eigenvectors belonging to different eigenvalues are *orthogonal*;
  - eigenvectors belonging to the same eigenvalues can be *orthogonalized*.
- Unitary and orthogonal matrices:
  - are always *unitarily diagonalizable*: $\mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\dagger}$;
  - eigenvalues are (in general, complex) such that $|\lambda_i| = 1$;
  - eigenvectors belonging to different eigenvalues are *orthogonal*;
  - eigenvectors belonging to the same eigenvalues can be *orthogonalized*.

- Normal matrices:
  - A matrix is referred to as *normal*, if it commutes with its adjoint: $\mathbf{A}\mathbf{A}^{\dagger} = \mathbf{A}^{\dagger}\mathbf{A}$;
    Hermitian and unitary matrices are particular cases of normal matrices
  - are always *unitarily diagonalizable*: $\mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\dagger}$ (*spectral decomposition*)
  - A matrix is *unitarily diagonalizable* if and only if it is normal.

More facts

- **Schur decomposition**: Any square matrix **A** is unitarily equivalent to some *upper triangular* (not necessarily diagonal) matrix **T** that contains the eigenvalues of **A** on the diagonal: $t_{ii} = \lambda_i$: $\mathbf{A} = \mathbf{U}\mathbf{T}\mathbf{U}^\dagger$
- **Jordan normal form**: Any square matrix **A** is similar to some *upper* block diagonal matrix

$$\mathbf{J} = \begin{pmatrix} \mathbf{J}_1 & & \\ & \ddots & \\ & & \mathbf{J}_p \end{pmatrix}$$

where $\mathbf{J}_i$ is so-called Jordan block:

$$\mathbf{J}_i = \begin{pmatrix} \lambda & 1 & 0 & \cdots & 0 & 0 \\ 0 & \lambda & 1 & \cdots & 0 & 0 \\ 0 & 0 & \lambda & \ddots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \ddots & \lambda & 1 \\ 0 & 0 & 0 & \cdots & 0 & \lambda \end{pmatrix}$$

Matrix Diagonalization

- Finding eigenvalues of a matrix **A** is equivalent to finding a similarity transformation that transforms it into a diagonal one:

$$\mathbf{\Lambda} = \mathbf{X}^{-1}\mathbf{A}\mathbf{X}$$

- The general strategy is to bring the matrix **A** toward a diagonal form by a series of known similarity transformations:

$$\mathbf{\Lambda} = \ldots\mathbf{X_3}^{-1}(\mathbf{X_2}^{-1}(\mathbf{X_1}^{-1}\mathbf{A}\mathbf{X_1})\mathbf{X_2})\mathbf{X_3}\ldots$$

A special case: symmetric matrices

- *Symmetric matrices* are *unitarily diagonalizable*, have *real* eigenvalues and -vectors:

  $\mathbf{\Lambda} = \mathbf{U}^T\mathbf{A}\mathbf{U}$, where **U** is a real orthogonal matrix; we have to find this matrix **U**.

A more special case: a symmetric 2×2 matrix

- $\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{pmatrix}$
- An orthogonal 2×2 matrix **U** can be always represented as a *rotation* by an angle φ:

$$\mathbf{U} = \begin{pmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{pmatrix} \equiv \begin{pmatrix} c & s \\ -s & c \end{pmatrix}; \quad c^2 + s^2 = 1$$

or as a *rotation with reflection*:

$$\mathbf{U} = \begin{pmatrix} \cos\varphi & \sin\varphi \\ \sin\varphi & -\cos\varphi \end{pmatrix} \text{ or } \begin{pmatrix} \cos\varphi & -\sin\varphi \\ -\sin\varphi & -\cos\varphi \end{pmatrix}$$

The similarity transformations given by these matrices are the same, so we wil consider the pure rotation only.

- $\mathbf{U^T A U} = \begin{pmatrix} c & -s \\ s & c \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{pmatrix} \begin{pmatrix} c & s \\ -s & c \end{pmatrix} =$

$$= \begin{pmatrix} c^2 a_{11} + s^2 a_{22} - 2sca_{12} & (c^2 - s^2)a_{12} + sc(a_{11} - a_{22}) \\ (c^2 - s^2)a_{12} + sc(a_{11} - a_{22}) & s^2 a_{11} + c^2 a_{22} + 2sca_{12} \end{pmatrix}$$

- Our purpose is to make $\mathbf{U}^T\mathbf{AU}$ diagonal by choosing appropriate values of s and c (keeping $s^2 + c^2 = 1$). To find it, we have to resolve the system of equations:

$$\begin{cases} (c^2 - s^2)a_{12} + sc(a_{11} - a_{22}) = 0 \\ c^2 + s^2 = 1 \end{cases}$$

- Solving this system dividing by c^2 and introducing $t = s/c$:

$$\begin{cases} (1-t^2)a_{12} + t(a_{11}-a_{22}) = 0 \;\Leftrightarrow\; a_{12}t^2 + \left(a_{22}-a_{11}\right)t - a_{12} = 0 \\ t^2 + 1 = 1 \,/\, c^2 \end{cases}$$

$$t = \frac{-2a_{12}}{-\left(a_{22}-a_{11}\right) \mp \sqrt{\left(a_{22}-a_{11}\right)^2 + 4a_{12}^2}} = \frac{1}{\dfrac{a_{22}-a_{11}}{2a_{12}} \pm \sqrt{\left(\dfrac{a_{22}-a_{11}}{2a_{12}}\right)^2 + 1}}$$

Here we have solved the quadratic equation $Ax^2 + Bx + C = 0$ as follows:

$$x = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A} = \frac{2C}{-B \mp \sqrt{B^2 - 4AC}}$$

Finally,

$$c = \pm\frac{1}{\sqrt{1+t^2}}$$

$$s = tc$$

- $$\mathbf{U^T A U} = \begin{pmatrix} c & -s \\ s & c \end{pmatrix}\begin{pmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{pmatrix}\begin{pmatrix} c & s \\ -s & c \end{pmatrix} = \begin{pmatrix} a_{11} - ta_{12} & 0 \\ 0 & a_{22} + ta_{12} \end{pmatrix}$$

A symmetric $N\times N$ matrix: the Jacobi diagonalization

- To eliminate an off-diagonal element a_{pq} of a symmetric matrix, we apply a similarity transformation with the following orthogonal matrix (*Jacobi rotation*):

$$\mathbf{P}_{pq} = \begin{pmatrix} 1 & & & & & & \\ & \cdots & & & & & \\ & & c & \cdots & s & & \\ & & \vdots & 1 & \vdots & & \\ & & -s & \cdots & c & & \\ & & & & & \cdots & \\ & & & & & & 1 \end{pmatrix}$$

$$\mathbf{A}' = \mathbf{P}_{pq}^T \mathbf{A} \mathbf{P}_{pq}$$

This operation will eliminate (set to zero) the values of a_{pq} and a_{qp}. Repeating this operation for all pairs p, q ($p>q$), we would finally obtain a diagonal matrix. The total transformation matrix would be given by the product of all such $\mathbf{P}_{pq}$: seems to be *good news*!

- *Bad news*: The multiplication by $\mathbf{P}_{pq}$ also affects all the matrix elements in the p-th and q-th rows and columns $\Rightarrow$ elimination of one matrix element will destroy zeros created previously.
- *Good news*: Repeating this elimination many times (in an *iterative algorithm*) will finally result to a converged diagonal matrix.

- **Algorithm overview**:

```
Set U = I
do while !iterations
  do p=1, N
      do q=1, p-1
         Calculate c and s from a_pq, a_pp, and a_qq.
         Calculate A' = P^T_pq A P_pq. Note that an explicit construction of P_pq
                                       is not needed; see below.
         Update U:  U = UP_pq.  An explicit matrix multiplication is not
                                       needed; see below
      end do !q
   end do !p
Check convergence
end do ! iterations
```

In more detail:

- Calculation of $\mathbf{A}' = \mathbf{P}_{pq}^T \mathbf{A} \mathbf{P}_{pq}$ (working formulae):

  `if` $a_{pq} = 0$ **`then`** *do nothing*;

  `else`

$$\theta = \frac{a_{qq} - a_{pp}}{2a_{pq}}$$

$$t = \frac{1}{\dfrac{a_{qq} - a_{pp}}{2a_{pq}} \pm \sqrt{\left(\dfrac{a_{qq} - a_{pp}}{2a_{pq}}\right)^2 + 1}} = \frac{1}{\theta \pm \sqrt{\theta^2 + 1}}$$

of these two solutions, t with lowest absolute value should be taken (it corresponds to a smaller rotation):

$$t = \frac{\operatorname{sgn}(\theta)}{|\,\theta\,| + \sqrt{\theta^2 + 1}}$$

watch the Fortran **`SIGN`** function!

$$c = \frac{1}{\sqrt{1+t^2}}$$

$$s = tc$$

$$a'_{pp} = a_{pp} - ta_{pq}$$

$$a'_{qq} = a_{qq} + ta_{pq}$$

$$a'_{pq} = 0 \quad \text{Evidently!}$$

The other elements in the same row and columns must be calculated for all r ($r \neq p \wedge r \neq q$):

$$a'_{rp} = ca_{rp} - sa_{rq}$$

$$a'_{rq} = ca_{rq} + sa_{rp}$$

- Calculation of **U = UP**$_{pq}$ (working formulae):

$$u'_{rp} = cu_{rp} - su_{rq}$$

$$u'_{rq} = su_{rp} + cu_{rq}$$

for all r.

```
endif
```

Convergence criterion:

$\Delta < \varepsilon$, where ε is a small number (e.g., $1 \cdot 10^{-12}$) and

$$\Delta = \sum_{p=1}^{N} \sum_{q=1}^{p-1} | a_{pq} | \quad \text{or} \quad \Delta = \sum_{p=1}^{N} \sum_{q=1}^{p-1} a_{pq}^2 \quad \text{or} \quad \Delta = \max\{| a_{pq} |\}$$

Programming task

- ***Input***:
  - The size of the matrix N;
  - The matrix itself (row by row).
- ***Output***:
  - Eigenvalues;
  - Eigenvectors.