Scuola Interdipartimentale
delle **Scienze**, dell'**Ingegneria**
e della **Salute**

L. Magistrale in IA (ML&BD)

Scientific Computing
(part 2 – 6 credits)

prof. Mariarosaria Rizzardi

Centro Direzionale di Napoli – Bldg. C4
room: n. 423 – North Side, 4th floor
phone: 081 547 6545
email: mariarosaria.rizzardi@uniparthenope.it

Contents

- Eigenvalues, Eigenvectors, Eigenspaces.
- Compute the eigenvectors.
- Algebraic and geometric multiplicity of an eigenvalue.
- Properties of eigenvalues and eigenvectors of any square matrix.
- Properties of eigenvalues and eigenvectors of a symmetric matrix.

Contents

- Geometrical interpretation of eigenvalues and eigenvectors.
- Eigenvalues and eigenvectors of particular symmetric matrices.
- Eigenvalues and eigenvectors of some linear maps.
- Diagonalization of a matrix.
- Diagonalization of a symmetric matrix.

Contents

- Spectral Theorem for a symmetric matrix.
- Connection between SVD and diagonalization.
- Consequences and applications of diagonalization and of eigenvalues/ eigenvectors.

Recall: Eigenvalues and eigenvectors of a matrix

A is a square matrix $n \times n$

λ ***eigenvalue*** for A $\overset{\text{def}}{\Longleftrightarrow}$ the sistem $Ax=\lambda x$ admits infinitely many solutions $x \neq \underline{0}$

the homogeneous linear system

$$Ax = \lambda x$$

is underdetermined

$\Leftrightarrow$

x ***eigenvector*** for A related to the eigenvalue λ $\overset{\text{def}}{\Longleftrightarrow}$ $x \neq \underline{0}$ is a solution of the system

$$Ax = \lambda x$$

$$\boxed{Ax = \lambda x \quad \Leftrightarrow \quad (A - \lambda I)x = 0}$$

How can we find problem unknowns: λ and $\boldsymbol{x}$?

$$(\boldsymbol{A}-\lambda\boldsymbol{I})\boldsymbol{x}=\underline{\boldsymbol{0}} \iff \det(\boldsymbol{A}-\lambda\boldsymbol{I})=0$$

$$\det(\boldsymbol{A}-\lambda\boldsymbol{I})=\boxed{c_0+c_1\lambda+c_2\lambda^2+\ldots+c_n\lambda^n}$$

A is a square matrix $n\times n$ ⇒ n-degree polynomial

⇑

characteristic polynomial

The eigenvalues are the n roots of the characteristic polynomial

Example

$$A=\begin{pmatrix}4 & -5\\ 2 & -3\end{pmatrix} \Rightarrow A-\lambda I=\begin{pmatrix}4-\lambda & -5\\ 2 & -3-\lambda\end{pmatrix} \Rightarrow \det(A-\lambda I)=\lambda^2-\lambda-2$$

```
A=[4 -5;2 -3];
syms lambda; B=A - lambda*eye(size(A));
S=solve(det(B),lambda)
S =
-1
 2
```

$\det(A-\lambda I)=0$

roots: $\lambda_1=-1,\quad \lambda_2=+2$

eigenvalues

Compute eigenvalues from definition in MATLAB

$$A=\begin{pmatrix}4 & -5\\ 2 & -3\end{pmatrix}$$

numerical

```
A=[ 4 -5;2 -3];
disp(eig(A).')
  2     -1
disp(charpoly(A))
     1     -1     -2
disp(roots(charpoly(A)).')
  2     -1
```

```
syms x real
disp(charpoly(A,x))
x^2 - x - 2
```

both numerical and symbolic

```
d=roots(charpoly(A));
```

symbolic

```
A=sym([4 -5;2 -3]);
disp(eig(A).')
[-1,  2]
disp(charpoly(A))
[1, -1, -2]
disp(roots(charpoly(A)).')
[-1,  2]
```

charpoly(A) returns the coefficients of the characteristic polynomial of **A**

charpoly(A,x) (only symbolic) returns the characteristic polynomial of **A** in terms of **x**

$$A=\begin{pmatrix}3 & -2 & 0\\ -2 & 3 & 0\\ 0 & 0 & 5\end{pmatrix}$$

numerical

```
A=[3 -2 0;-2 3 0;0 0 5];
disp(eig(A).')
     1     5     5
disp(charpoly(A))
   1    -11     35    -25
disp(roots(charpoly(A)).')
     5     5     1
```

```
syms x real
disp(charpoly(A,x))
x^3 - 11*x^2 + 35*x - 25
```

both numerical and symbolic

```
d=roots(charpoly(A));
```

symbolic

```
A=sym([3 -2 0;-2 3 0;0 0 5]);
disp(eig(A).')
[1, 5, 5]
disp(charpoly(A))
[1, -11, 35, -25]
disp(roots(charpoly(A)).')
[1, 5, 5]
```

If a matrix is real, its characteristic polynomial has real coefficients: what type are its roots?

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 5 & 0 & -1 \\ 3 & 1 & 0 \end{pmatrix} \Rightarrow A-\lambda I = \begin{pmatrix} 1-\lambda & 0 & 0 \\ 5 & -\lambda & -1 \\ 3 & 1 & -\lambda \end{pmatrix}$$

$\det(A-\lambda I) = (1-\lambda)(\lambda^2+1)$, roots: $\lambda_1 = 1,\ \lambda_2 = +i,\ \lambda_3 = -i$

complex eigenvalues!

```
A=[1 0 0;5 0 -1;3 1 0]; disp(charpoly(A))
 1    -1     1    -1
disp(eig(A).')
        0 + 1i       0 - 1i       1 + 0i
```

Also if the A matrix is real, **nothing** can be said about the existence of real eigenvalues. The **Fundamental Theorem of Algebra** (also known as d'Alembert's Theorem or the d'Alembert-Gauss Theorem "a n-degree polynomial has exactly n complex roots") implies that:

A matrix of dimension n has n complex eigenvalues: some of them may be real, and each complex eigenvalue always appears paired with its complex conjugate.

Computing the eigenvalues is immediate when the matrix is diagonal or triangular; in these cases the eigenvalues are exactly the elements on the main diagonal.

The $\boldsymbol{LU}$ factorization does **NOT** preserve the eigenvalues.

Examples

Lower triangular matrix

$$\boldsymbol{A} = \begin{pmatrix} 1 & & \\ 0 & 2 & \\ 4 & 5 & 3 \end{pmatrix} \qquad \det(\boldsymbol{A} - \lambda I) = \det \begin{pmatrix} 1-\lambda & & \\ 0 & 2-\lambda & \\ 4 & 5 & 3-\lambda \end{pmatrix}$$

Its eigenvalues are: **1**, **2**, **3**.

$$\boldsymbol{A} = \begin{pmatrix} 4 & -5 \\ 2 & -3 \end{pmatrix}$$

eigenvalues of $\boldsymbol{A}$: $\lambda_1 = -1, \quad \lambda_2 = +2$

$\updownarrow$ different

$$\boldsymbol{A} = \boldsymbol{LU} = \begin{pmatrix} 1 & \\ ½ & 1 \end{pmatrix} \begin{pmatrix} 4 & -5 \\ 0 & -½ \end{pmatrix}$$

eigenvalues of $\boldsymbol{U}$: $\lambda_1 = -1/2, \quad \lambda_2 = +4$

To find all the **eigenvectors** related to a particular eigenvalue λ^*, we have to compute the infinitely many solutions x^* of the following underdetermined homogeneous linear system

$$(A-\lambda^* I)\,x^* = \underline{0}$$

i.e. we have to compute the *Null Space* of the matrix $A-\lambda^* I$. This subspace V_{λ^*} is called the

eigenspace related to λ^*:

$$V_{\lambda^*} = \mathcal{N}(A-\lambda^* I)$$

Examples: computing eigenvectors

$$A=\begin{pmatrix}4 & -5\\ 2 & -3\end{pmatrix}$$ eigenvalues: $\lambda_1=-1,\ \lambda_2=+2$

both numerical and symbolic

```
d=roots(charpoly(A));
```

```
A=[4 -5;2 -3];
d=eig(A)
d =
   2
  -1
```

```
A=sym([4 -5;2 -3]);
d=eig(A)
d =
-1
 2
```

eigenvalues

$$A-\lambda_1 I=\begin{pmatrix}5 & -5\\ 2 & -2\end{pmatrix}$$

$\lambda_1=-1$

$$V_{-1}=\mathcal{N}(A-\lambda_1 I)=\mathrm{span}\{(1,1)^T\}$$

```
null(A-d(2)*eye(2))
ans =
     0.7071
     0.7071
```

```
null(A-d(1)*eye(2))
ans =
1
1
```

eigenspaces

```
null(A-d(1)*eye(2))
ans =
     0.9285
     0.3714
```

```
null(A-d(2)*eye(2))
ans =
5/2
  1
```

$\lambda_2=+2$

$$A-\lambda_2 I=\begin{pmatrix}2 & -5\\ 2 & -5\end{pmatrix}$$

$$V_{+2}=\mathcal{N}(A-\lambda_2 I)=\mathrm{span}\{(5,2)^T\}$$

```
[V,d]=eig(A,'vector')
d =
  2
 -1
```

```
V =
     0.9285     0.7071
     0.3714     0.7071
```

bases of the two eigenspaces

```
[V,D]=eig(A)
V =
[1, 5/2]
[1,   1]
```

```
D =
[-1, 0]
[ 0, 2]
```

MATLAB eig fun: computing eigenvalues/eigenvectors

numerically

```
A=[4 -5;2 -3];
[V, D]=eig(A)
V =
   0.9285     0.7071
   0.3714     0.7071
D =
      2       0
      0      -1
```

normalized vectors ($\|\cdot\|_2=1$) but non-orthogonal

V: each column represents a basis for the eigenspace of the related eigenvalue

```
disp(V'*V)
   1.0000     0.9191
   0.9191     1.0000
```

D: the main diagonal contains the eigenvalues

```
[V, d] = eig(A,'vector')
V =
   0.92848      0.70711
   0.37139      0.70711
d =
      2
     -1
```

vector of eigenvalues

```
V1=null(A-d(1)*eye(size(A)))
V1 =
   0.92848
   0.37139
V2=null(A-d(2)*eye(size(A)))
V2 =
   0.70711
   0.70711
```

symbolically

```
A=sym([4 -5;2 -3]);
[V, D]=eig(A)
V =
[   1,  5/2]
[   1,    1]
D =
[   -1     0]
[    0     2]
disp(rank([V1 V(:,2)]))
1
disp(rank([V2 V(:,1)]))
1
```

non-normalized vectors

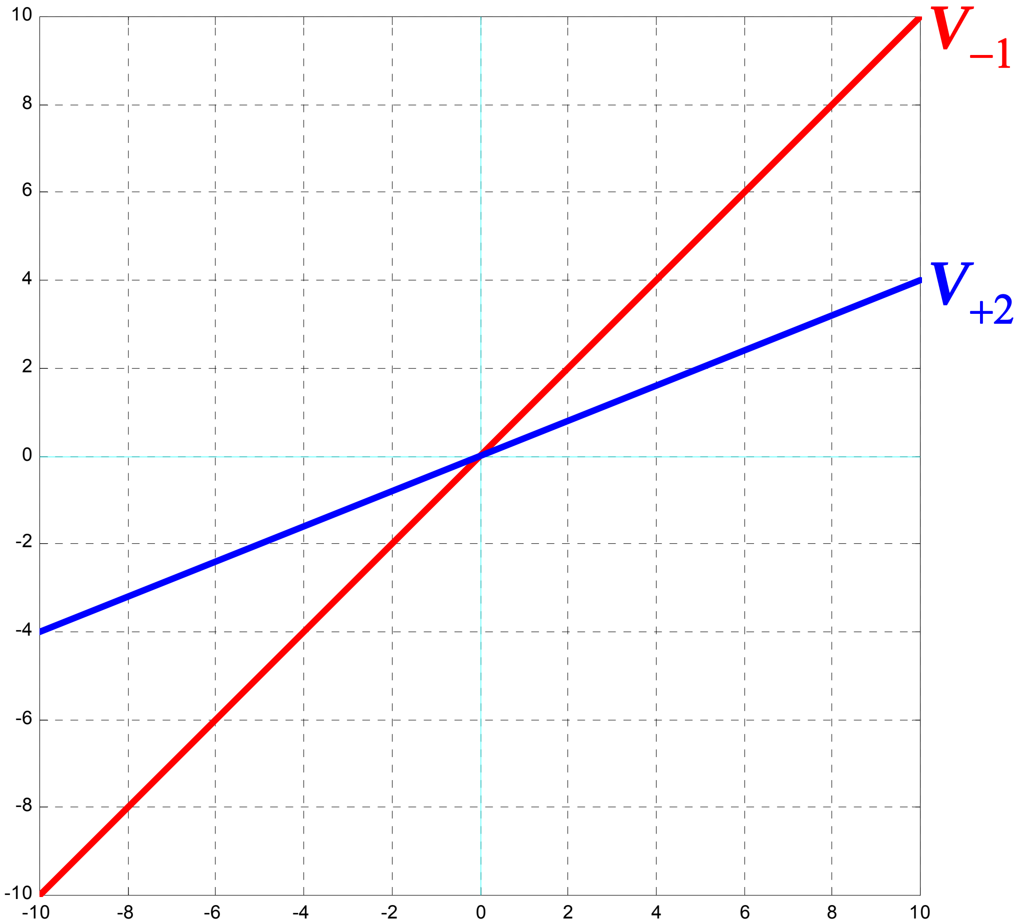
eigenspaces of $A = \begin{pmatrix} 4 & -5 \\ 2 & -3 \end{pmatrix}$

Spectrum (set of eigenvalues) of **A**

$$\sigma(\mathbf{A}) = \{\lambda_1 = -1,\ \lambda_2 = +2\}$$

$$V_{-1} = \mathcal{N}(A - \lambda_1 I) = \mathbf{span}\{(1,1)^T\}$$

$$V_{+2} = \mathcal{N}(A - \lambda_2 I) = \mathbf{span}\{(5,2)^T\}$$



MATLAB eig fun: computing eigenvalues/eigenvectors

$$A=\begin{pmatrix}3&-2&1\\-2&3&0\\0&0&5\end{pmatrix}$$

simple root

eigenvalues: $\lambda_1=1,\ \lambda_2=\lambda_3=5$

double root

```
A=[3 -2 1; -2 3 0; 0 0 5];
[V, D] = eig(A)
V =
    0.7071    0.7071    -0.7071
   -0.7071    0.7071     0.7071
         0         0 1.5701e-15
D =
         5         0          0
         0         1          0
         0         0          5
```

parallel

$$A-\lambda_1 I=\begin{pmatrix}2&-2&1\\-2&2&0\\0&0&4\end{pmatrix}$$

$$V_{\lambda=1}=\mathcal{N}(A-\lambda_1 I)=\mathrm{span}\{(1,1,0)^T\}$$

```
V1=null(A-D(1,1)*eye(size(A)))
V1 =
1
1
0
```

```
[V, D] = eig(sym(A))
V =
[  1, -1]
[  1,  1]
[  0,  0]
D =
[ 1, 0, 0]
[ 0, 5, 0]
[ 0, 0, 5]
```

eigenspaces

$$A-\lambda_2 I=\begin{pmatrix}-2&-2&1\\-2&-2&0\\0&0&0\end{pmatrix}$$

$$V_{\lambda=5}=\mathcal{N}(A-\lambda_2 I)=\mathrm{span}\{(-1,1,0)^T\}$$

```
V2=null(A-D(2,2)*eye(size(A)))
V2 =
-1
 1
 0
```

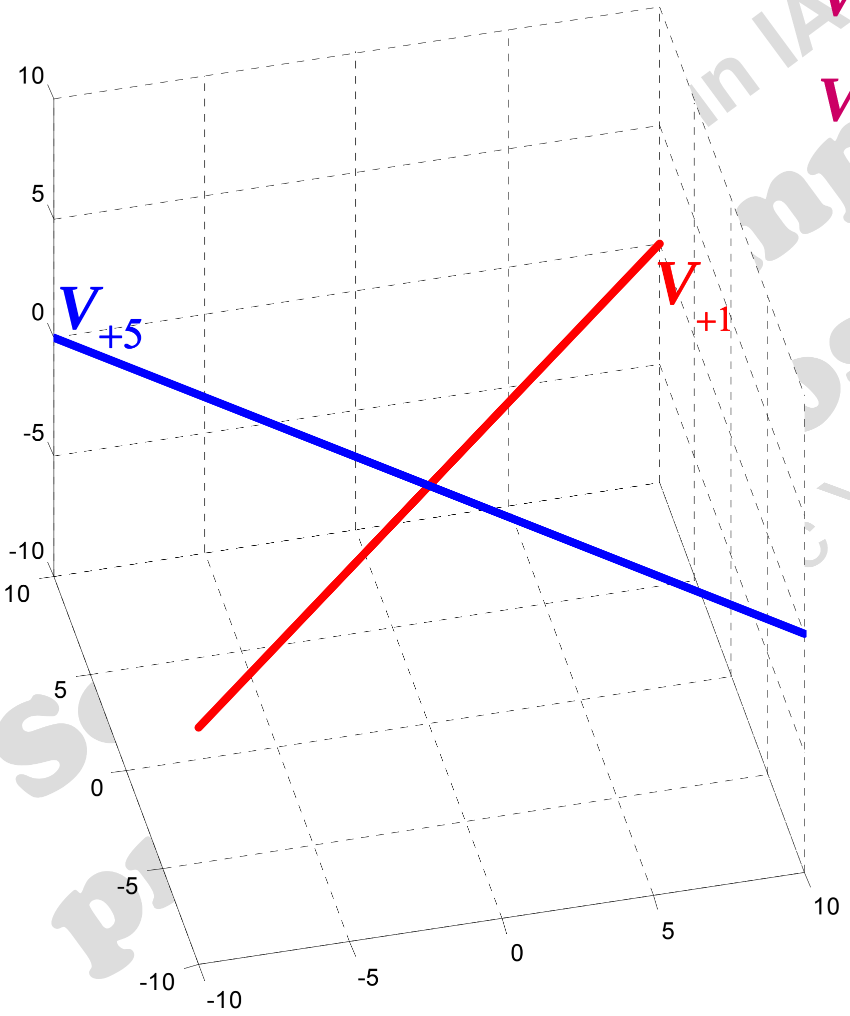
eigenspaces of $A= \begin{pmatrix} 3 & -2 & 1 \\ -2 & 3 & 0 \\ 0 & 0 & 5 \end{pmatrix}$

Spectrum (set of eigenvalues) of $\mathbf{A}$

$$\sigma(\mathbf{A}) = \{\lambda_1=+1,\ \lambda_2=+5\}$$

$$V_{+1}=\mathcal{N}(A-\lambda_1 I) = \mathbf{span}\{(1,1,0)^\mathsf{T}\}$$

$$V_{+5}=\mathcal{N}(A-\lambda_2 I) = \mathbf{span}\{(-1,1,0)^\mathsf{T}\}$$



MATLAB eig fun: computing eigenvalues/eigenvectors

$$A = \begin{pmatrix} 3 & -2 & 0 \\ -2 & 3 & 0 \\ 0 & 0 & 5 \end{pmatrix}$$ symmetric

eigenvalues: $\lambda_1 = 1,\ \lambda_2 = \lambda_3 = 5$

$$A - \lambda_1 I = \begin{pmatrix} 2 & -2 & 0 \\ -2 & 2 & 0 \\ 0 & 0 & 4 \end{pmatrix}$$

$$V_1 = \mathcal{N}(A - \lambda_1 I) = \text{span}\{(1,1,0)^T\}$$

$$\Uparrow$$

eigenspaces

$$\Downarrow$$

$$A - \lambda_2 I = \begin{pmatrix} -2 & -2 & 0 \\ -2 & -2 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$V_2 = \mathcal{N}(A - \lambda_2 I) = \text{span}\{(-1,1,0)^T, (0,0,1)^T\}$$

```
A=[3 -2 0;-2 3 0; 0 0 5];
[V, D]=eig(sym(A))
V =
[1, -1, 0]
[1,  1, 0]
[0,  0, 1]
D =
[1, 0, 0]
[0, 5, 0]
[0, 0, 5]
```

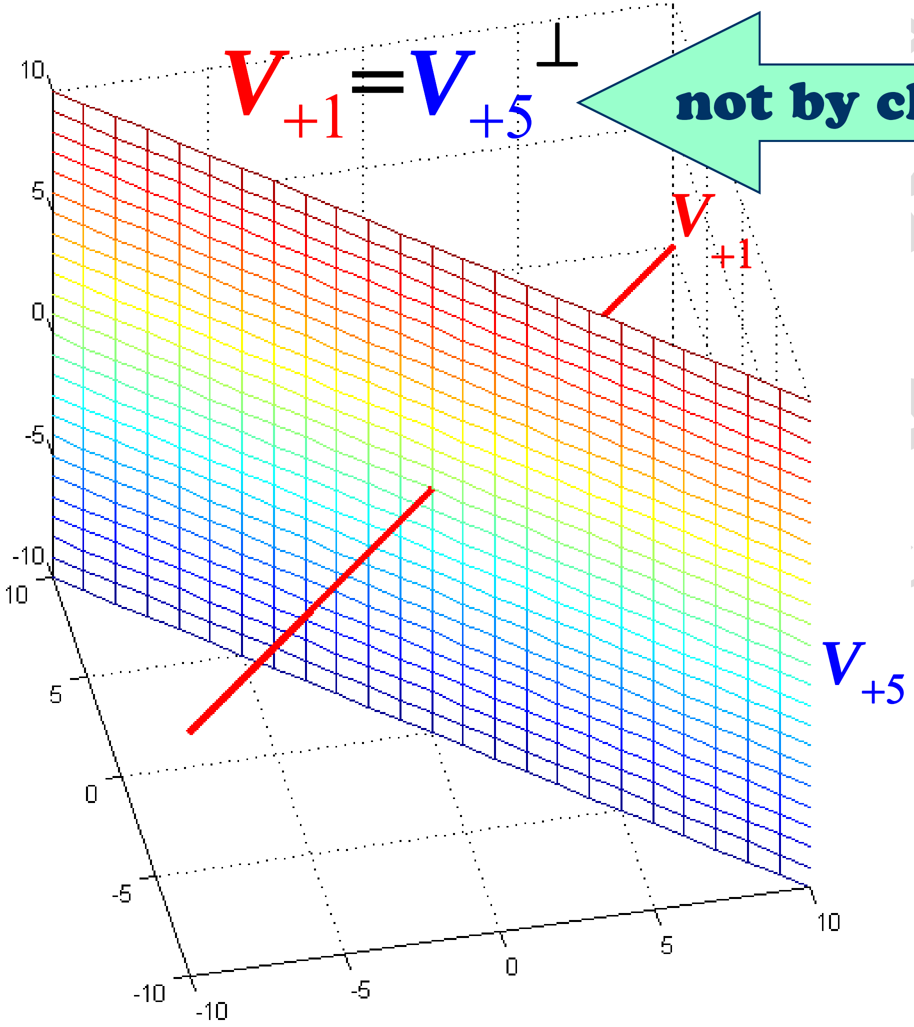
```
d=roots(charpoly(A));
syms lambda
B=A-lambda*eye(size(A));
```

```
V1=null(subs(B,lambda,D(1,1)))
V1 =
1
1
0
```

```
V2=null(subs(B,lambda,D(2,2)))
V2 =
[-1, 0]
[ 1, 0]
[ 0, 1]
```

eigenspaces of $A=\begin{pmatrix} 3 & -2 & 0 \\ -2 & 3 & 0 \\ 0 & 0 & 5 \end{pmatrix}$

symmetric matrix



$V_{+1}=V_{+5}^{\perp}$

not by chance!

```
A=[3 -2 0;-2 3 0; 0 0 5];
[V,D]=eig(sym(A));
disp(rank(V))
3
syms lambda
B=A-lambda*eye(size(A));
V1=null(subs(B,lambda,D(1,1)));
V2=null(subs(B,lambda,D(2,2)));
subspace(V1,V2)
ans =
pi/2
```

orthogonal eigenspaces

$\boldsymbol{\nu}(\lambda)$ ***algebraic multiplicity*** of an eigenvalue λ

$$\overset{\textbf{def}}{\Longleftrightarrow}$$

multiplicity of λ as a root of the characteristic polynomial

$\boldsymbol{\mu}(\lambda)$ ***geometric multiplicity*** of an eigenvalue λ

$$\overset{\textbf{def}}{\Longleftrightarrow}$$

dimension of the related eigenspace $\boldsymbol{V}_\lambda$

It can be proved that:

1) $\boxed{\boldsymbol{\mu}(\lambda) \le \boldsymbol{\nu}(\lambda),\ \forall\lambda}$

(geometric multiplicity does not exceed algebraic multiplicity)

$\boldsymbol{A}(n\times n) \Rightarrow$ 2) $\boxed{\sum_j \boldsymbol{\nu}(\lambda_j) = n}$

1) + 2) $\Rightarrow$ the sum of dimensions of all the eigenspaces is n at most

Properties of eigenvalues/eigenvectors of any square matrix $A_{(n\times n)}$

Let $A_{n\times n}$ be any real square matrix

1. Each eigenvalue is uniquely determined by its eigenvector, while there are infinitely many eigenvectors related to a particular eigenvalue;
2. If λ is an eigenvalue of A and x is a corresponding eigenvector, then λ^2 is an eigenvalue of A^2 and x its related eigenvector;
3. If $\lambda\neq 0$ is an eigenvalue of A (invertible) and x is a corresponding non-zero eigenvector, then λ^{-1} is an eigenvalue of A^{-1} and x is its related eigenvector;
4. The matrices A and A^T have the same eigenvalues;

Properties of eigenvalues/eigenvectors of any square matrix $A_{(n\times n)}$

5. Eigenvectors related to different eigenvalues are linearly independent;

6. The product of all the eigenvalues of a matrix equals the determinant of the matrix
   (in particular, if ∃ $\lambda=0$ then $\det(A)=0$);

7. The sum of all the eigenvalues of a matrix equals the sum of elements on the main diagonal of A (called the trace of the matrix):

$$\sum_{k=1}^{n}\lambda_k = \mathrm{Tr}(A) = \sum_{k=1}^{n} a_{k,k}$$

Properties of eigenvalues/eigenvectors of a symmetric real matrix $A_{(n\times n)}$

$$A^T = A$$

in addition:

8. The eigenvalues are all real ($\sigma(A) \subset \mathbb{R}$)*;

* particular case of "eigenvalues of a hermitian matrix are real"

9. Eigenspaces corresponding to different eigenvalues are mutually orthogonal;
10. If λ is an eigenvalue of algebraic multiplicity $\nu(\lambda)$, then the corresponding eigenspace has dimension: $\mu(\lambda)=\nu(\lambda)$, that is the geometrical multiplicity $\mu(\lambda)$ equals the algebraic multiplicity $\nu(\lambda)$;
11. $\mathbb{R}^n$ can be obtained as a direct sum of all the eigenspaces;
12. A is diagonalizable (as we will see later …)

A symmetric and positive definite ⇨ all its eigenvalues are positive

Geometrical interpretation of eigenvalues/eigenvectors

If we consider the endomorphism associated with the A matrix

$$t_A \ : \ x\in\mathbb{R}^n \longrightarrow t_A(x) = Ax\in\mathbb{R}^n$$

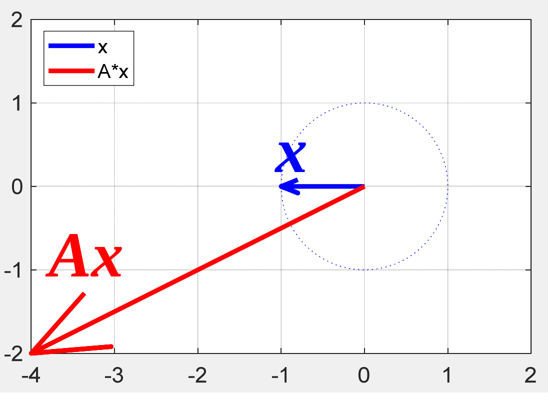
solving the equation $Ax=\lambda x$ is the same as looking for a vector x whose image $y=t_A(x)=Ax$ lies parallel to x, since only parallel vectors have proportional components:

$$y=t_A(x)=Ax=\lambda x$$

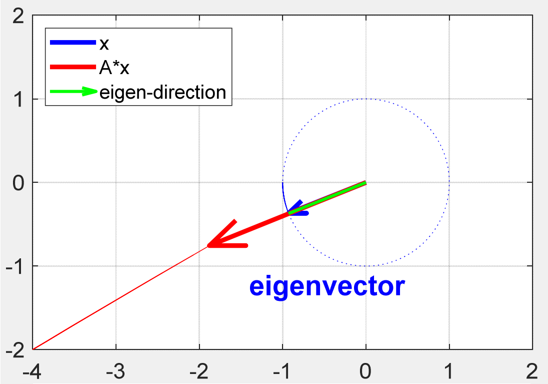
The proportionality constant (i.e. the scaling factor) between the components of an eigenvector x and its image $y=t_A(x)$ is the *eigenvalue* λ_x of A and the related *eigenspace* V_{λ_x} consists of all the vectors that have the same property of x.

Therefore each eigenspace remains overall **unchanged** by the mapping t_A, i.e. $t_A(V_\lambda)=V_\lambda$ (hence the term *eigen*space), although the individual vectors do not necessarily transform into themselves unless the eigenvalue is **1**.

Example: download and run `eigenvectors.p` (eigenvectors of a matrix)



x moves on the unitary circle

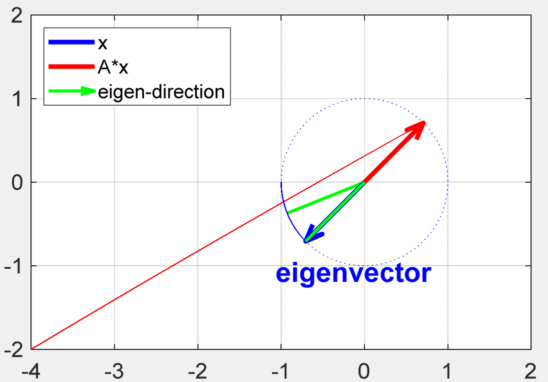


eigenvector related to $\lambda_1 = +2$

Ax is twice the length of x

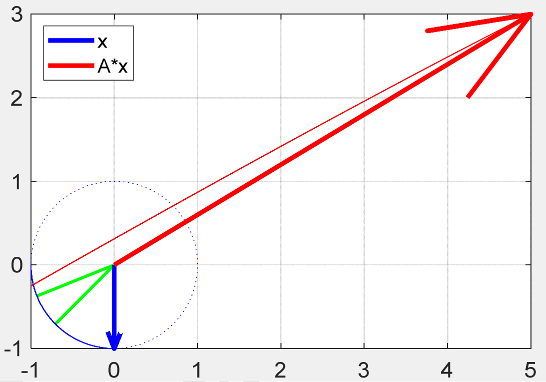
$$A = \begin{pmatrix} 4 & -5 \\ 2 & -3 \end{pmatrix}$$

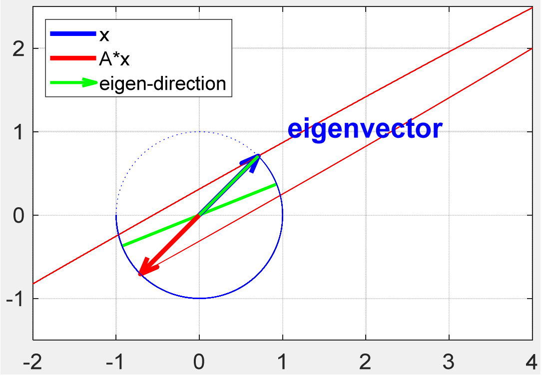
```
A=[4 -5;2 -3];
[V,D]=eig(A,'vector')
V =
   0.92848    0.70711
   0.37139    0.70711
D =
          2  λ1
         -1  λ2
```



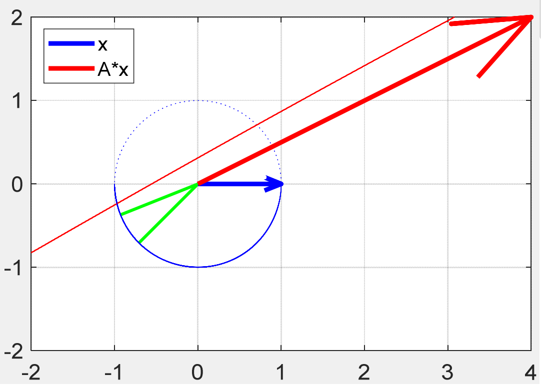
eigenvector related to $\lambda_2 = -1$

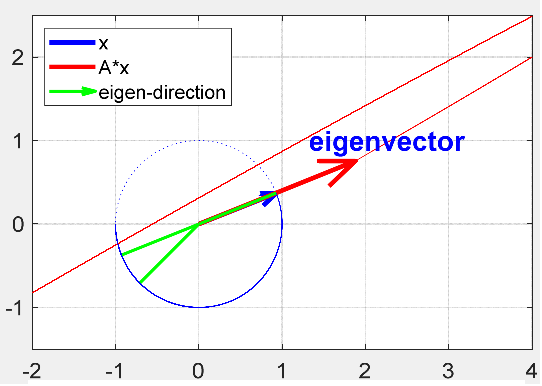
Ax is the opposite of x



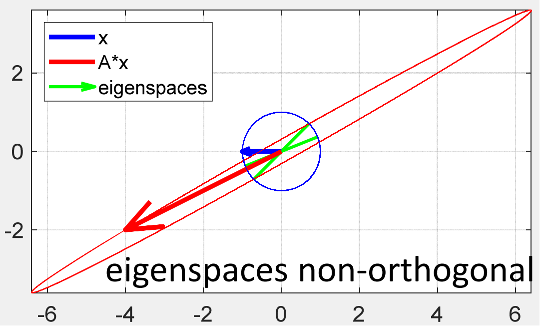


eigenvector related to $\lambda_2 = -1$



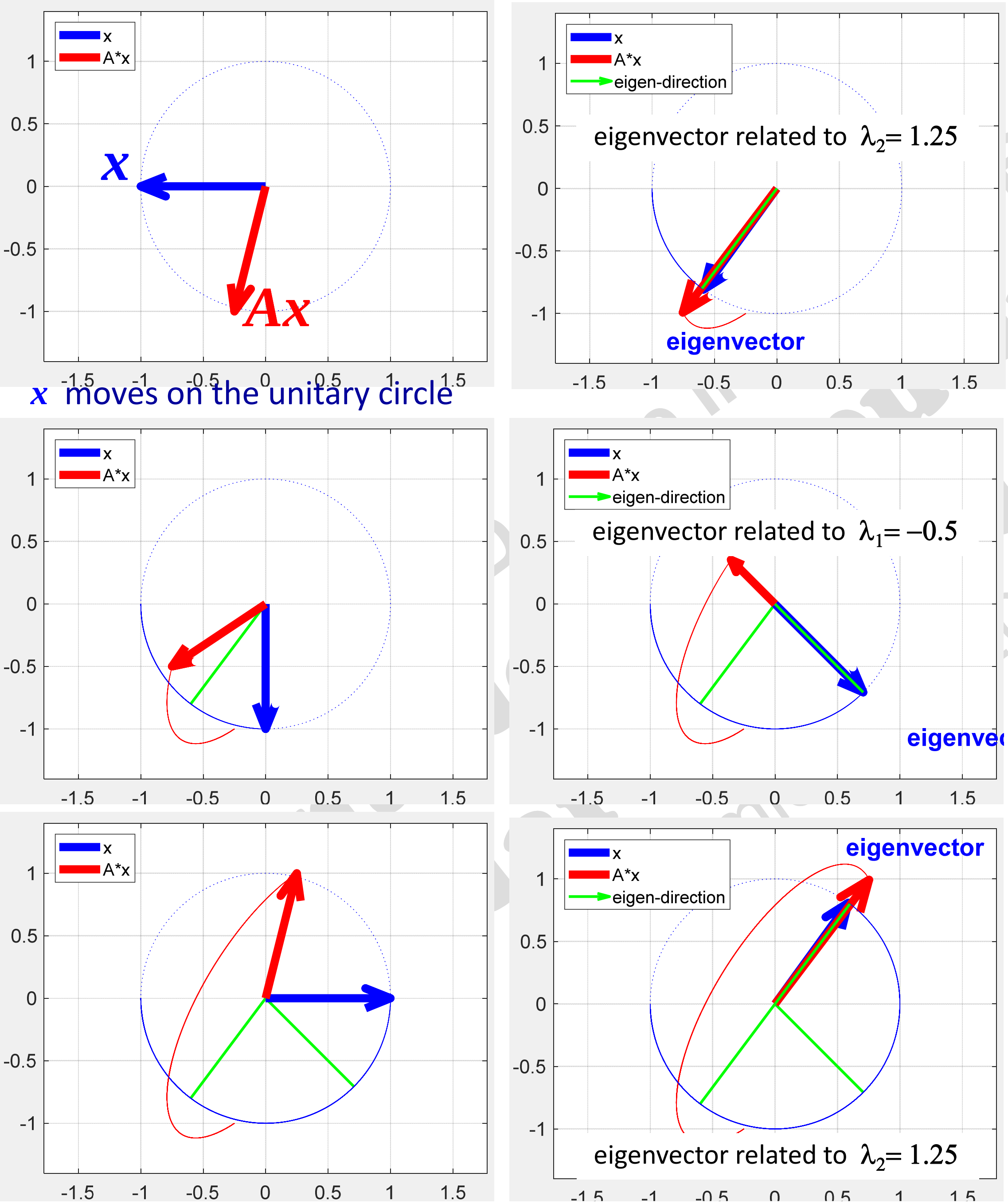


eigenvector related to $\lambda_1 = +2$



eigenspaces non-orthogonal

Example: download and run `eigenvectors.p` (eigenvectors of a matrix)

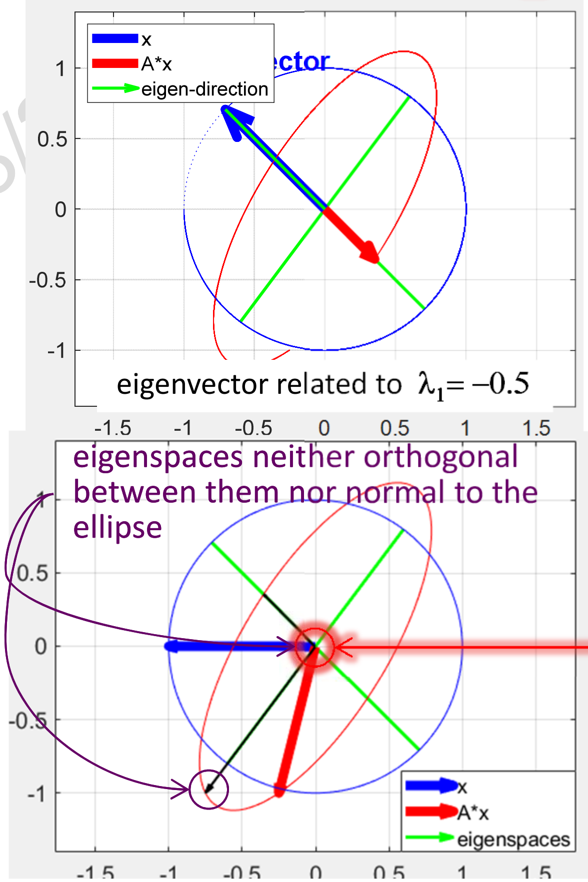


x moves on the unitary circle

$$A=\begin{pmatrix}\frac{1}{4} & \frac{3}{4}\\ 1 & \frac{1}{2}\end{pmatrix}$$

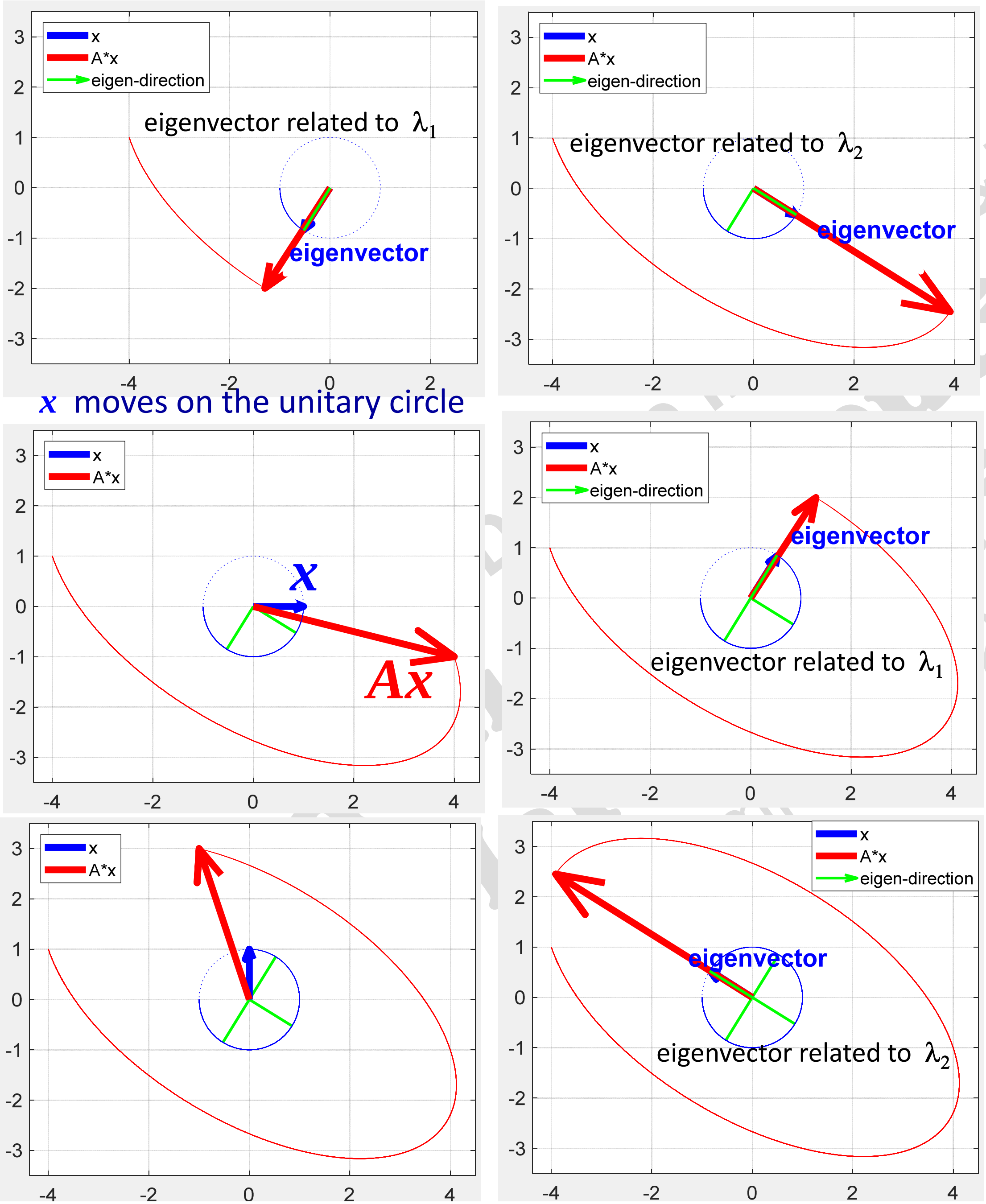
```
A=[1/4 3/4;1 1/2];
[V,D]=eig(A ,'vector')
V =
    -0.70711       -0.6
     0.70711       -0.8
D =
     -0.5  λ1
     1.25  λ2
```

```
subspace( ...
V(:,1), ...
V(:,2))*180/pi
ans =
               81.87
```



eigenspaces neither orthogonal between them nor normal to the ellipse

Example: download and run `eigenvectors.p` (eigenvectors of a matrix)



x moves on the unitary circle

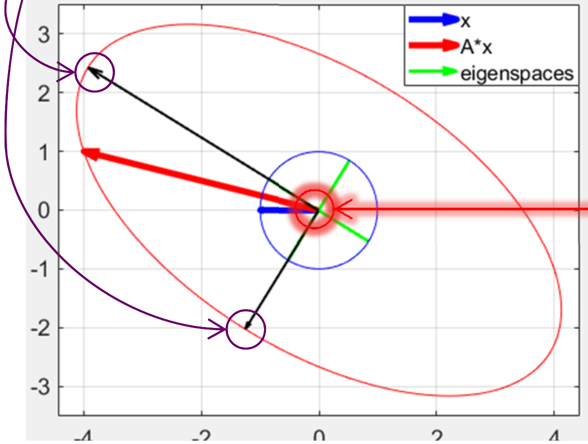
symmetric $$A=\begin{pmatrix}4 & -1\\ -1 & 3\end{pmatrix}$$

positive definite matrix

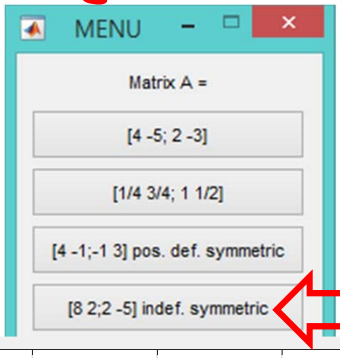
```
A=[4 -1;-1 3];
[V,D]=eig(A ,'vector')
V =
    -0.52573    -0.85065
    -0.85065     0.52573
D =
          2.382 λ1>0
          4.618 λ2>0
```

```
subspace( ...
V(:,1), ...
V(:,2))*180/pi
ans =
       90
```

eigenspaces orthogonal between them and normal to the ellipse



Example: download and run `eigenvectors.p` (eigenvectors of a matrix)



Matrix A =

[4 -5; 2 -3]

[1/4 3/4; 1 1/2]

[4 -1;-1 3] pos. def. symmetric

[8 2;2 -5] indef. symmetric

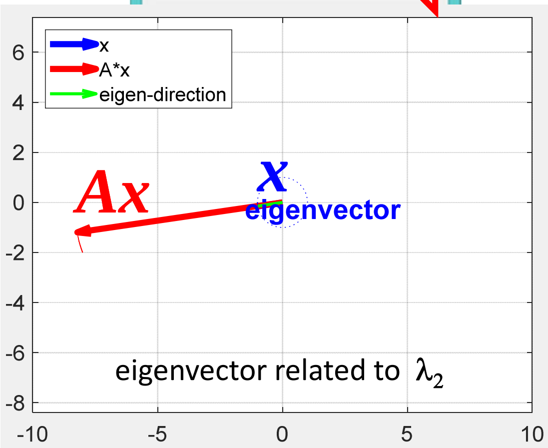
```
>> eigenvectors
```

symmetric $A = \begin{pmatrix} 8 & 2 \\ 2 & -5 \end{pmatrix}$

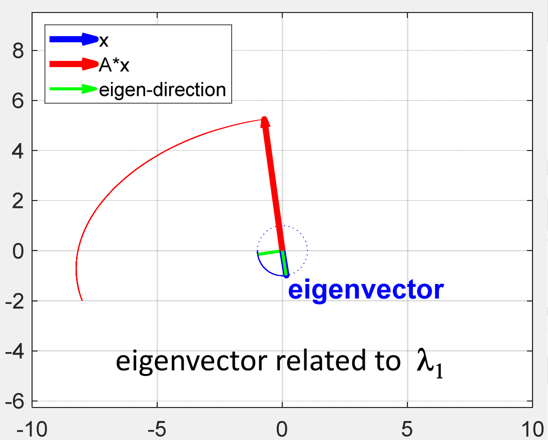
indefinite matrix

```
A=[8 2;2 -5];
[V,D]=eig(A,'vector')
V =
      0.1487     -0.98888
    -0.98888      -0.1487
D =
     -5.3007  λ1
      8.3007  λ2
```

```
subspace( ...
V(:,1), ...
V(:,2))*180/pi
ans =
       90
```



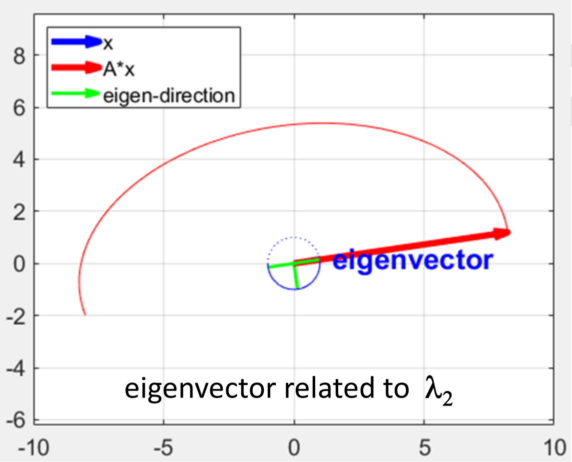
eigenvector related to λ_2



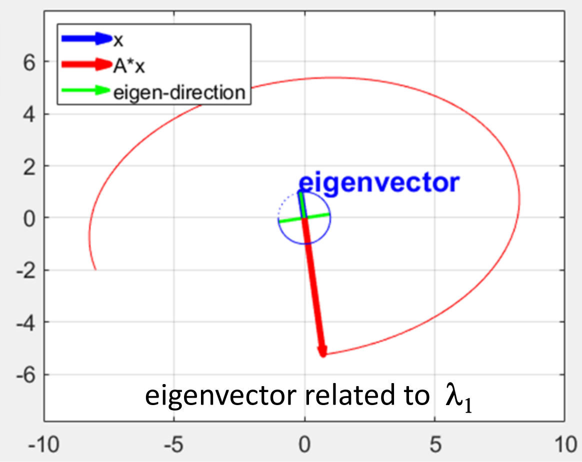
eigenvector related to λ_1

x moves on the unitary circle

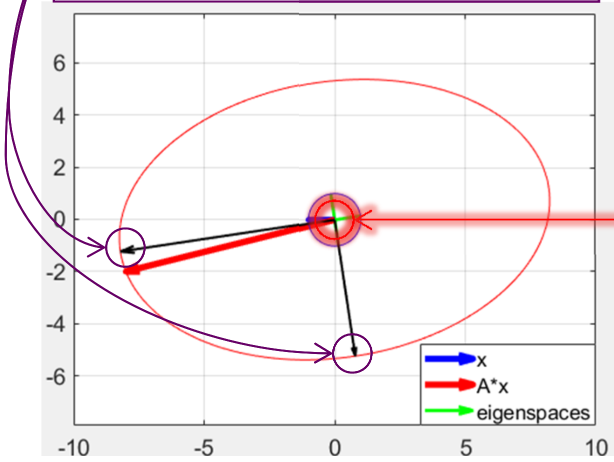
eigenspaces orthogonal each other and normal to the ellipse



eigenvector related to λ_2



eigenvector related to λ_1



A symmetric matrix and its ellipse

A symmetric matrix A can be:

positive definite ($x^T A x > 0$) : $\lambda_k > 0$, negative definite ($x^T A x < 0$) : $\lambda_k < 0$,
semipositive definite ($x^T A x \geq 0$) : $\lambda_k \geq 0$, seminegative definite ($x^T A x \leq 0$) : $\lambda_k \leq 0$,
indefinite (otherwise).

```
A=...; [V,D]=eig(A,'vector')
...
t=linspace(-pi,pi,201); x=[cos(t);sin(t)]; Ax=A*x;
plot(Ax(1,:),Ax(2,:),'k'); axis equal; grid on; hold on
quiver(0,0,V(1,1),V(2,1),abs(D(1)),'Color','b','LineWidth',2)
quiver(0,0,V(1,2),V(2,2),abs(D(2)),'Color','r','LineWidth',2)
```

scale factor

symmetric

$$A = \begin{pmatrix} 4 & -1 \\ -1 & 3 \end{pmatrix}$$

```
V =
  -0.52573   -0.85065
  -0.85065    0.52573
D =
   2.382
   4.618
```

$\lambda > 0$

positive definite matrix

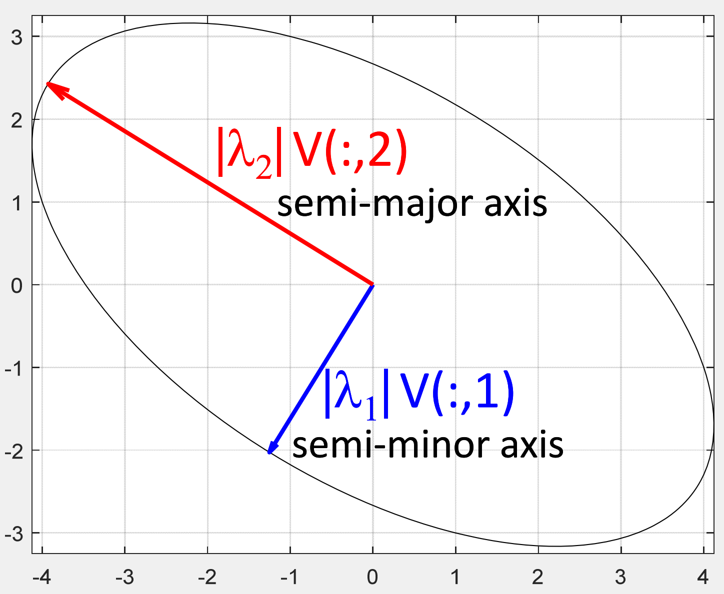
symmetric

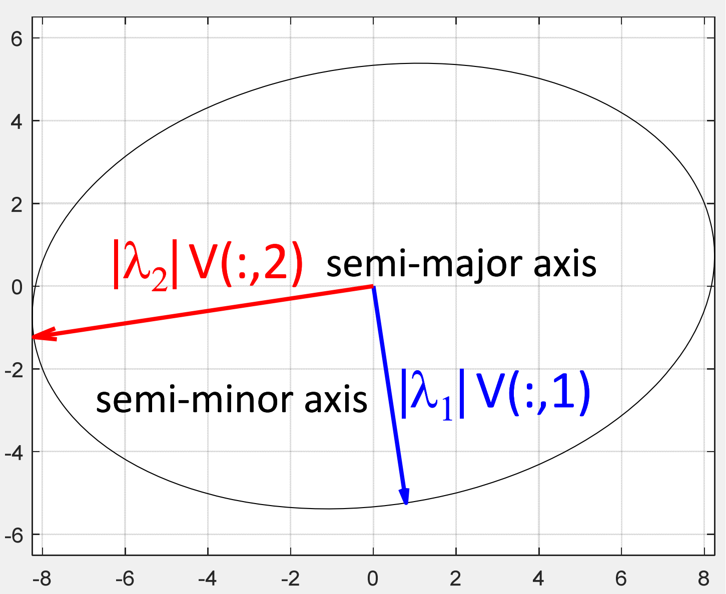
$$A = \begin{pmatrix} 8 & 2 \\ 2 & -5 \end{pmatrix}$$

```
V =
    0.1487    -0.98888
  -0.98888    -0.1487
D =
  -5.3007
   8.3007
```

$\lambda \underset{\leq}{\geq} 0$

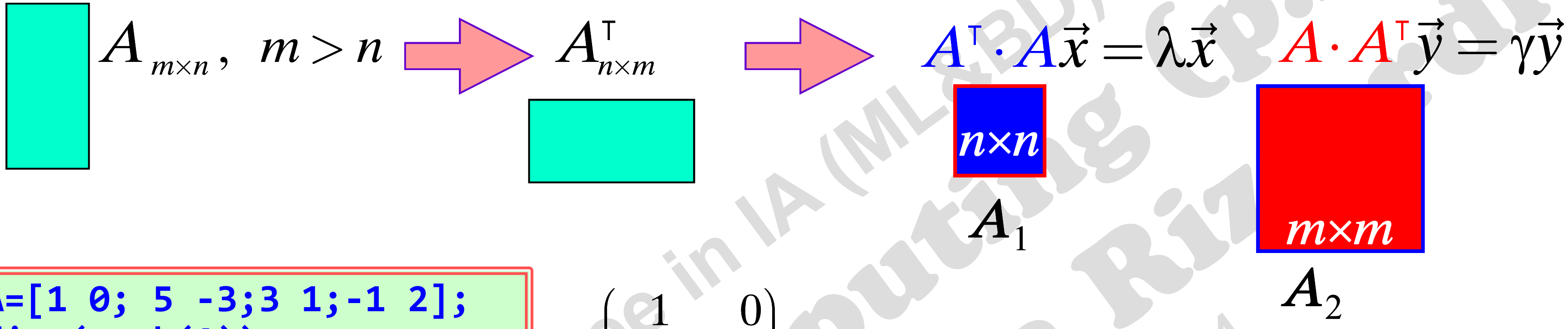
indefinite matrix





Eigenvalues of particular symmetric matrices:

$A^{\mathsf{T}}A$, AA^{T}



```
A=[1 0; 5 -3;3 1;-1 2];
disp(rank(A))
2
A1=A'*A
A1 =
     36    -14
    -14     14
disp(rank(A1))
2
[V1,d1]=eig(A1,'vector')
V1 =
   -0.4371   -0.8994
   -0.8994    0.4371
d1 =
    7.1955
   42.8045
```

$A^{\mathsf{T}}\cdot A\vec{x}=\lambda\vec{x}$ $\boxed{A_1}$

$$A=\begin{pmatrix}1&0\\5&-3\\3&1\\-1&2\end{pmatrix}$$

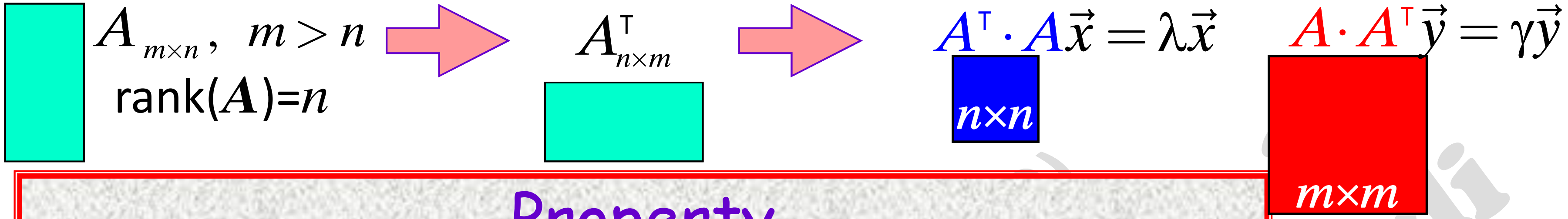
$A^{\mathsf{T}}\cdot A\vec{x}=\lambda\vec{x}$

$A\cdot A^{\mathsf{T}}\vec{y}=\gamma\vec{y}$

```
A2=A*A'
A2 =
     1     5     3    -1
     5    34    12   -11
     3    12    10    -1
    -1   -11    -1     5
disp(rank(A2))
2
[V2,d2]=eig(A2,'vector')
V2 =
  -0.9770   -0.0003    0.1630   -0.1375
   0.0932   -0.4082   -0.1911   -0.8878
   0.1860    0.4083    0.8242   -0.3456
   0.0468   -0.8165    0.5076    0.2711
d2 =
   0.00
   0.00
   7.1955
  42.8045
```

$A\cdot A^{\mathsf{T}}\vec{y}=\gamma\vec{y}$ $\boxed{A_2}$

non-zero eigenvalues are equal



$A_{m\times n},\ \ m>n$

rank($\mathbf{A}$)=n

$A^{\mathsf{T}}_{n\times m}$

$A^{\mathsf{T}}\cdot A\vec{x}=\lambda\vec{x}$

$n\times n$

$A\cdot A^{\mathsf{T}}\vec{y}=\gamma\vec{y}$

$m\times m$

Property

spectrum $\sigma([A\,A^{\mathsf{T}}]_{m\times m})$ = spettro $\sigma([A^{\mathsf{T}}A]_{n\times n})\cup\{0\}$

Proof

1) Let λ be any eigenvalue of $A^{\mathsf{T}}\cdot A$ and $\boldsymbol{x}$ a related eigenvector:

$$A^{\mathsf{T}}\cdot A\vec{x}=\lambda\vec{x}$$

If we premultiply by $\mathbf{A}$: $A(A^{\mathsf{T}}\cdot A\vec{x})=A(\lambda\vec{x})$, then we have

$$AA^{\mathsf{T}}\cdot(A\vec{x})=\lambda(A\vec{x})$$

i.e. λ is also an eigenvalue of $A\cdot A^{\mathsf{T}}$ and the related eigenvector is $A\vec{x}$

$$\Rightarrow\quad \sigma([A^{\mathsf{T}}\cdot A]_{n\times n})\subseteq\sigma([A\cdot A^{\mathsf{T}}]_{m\times m})$$

2) Conversely, let $\gamma\neq 0$ an eigenvalue of $A\cdot A^{\mathsf{T}}$ and y a related eigenvector: $A\cdot A^{\mathsf{T}}\vec{y}=\gamma\vec{y}$

Premultiply by $\mathbf{A}^{\mathsf{T}}$: then $A^{\mathsf{T}}A\cdot(A^{\mathsf{T}}\vec{y})=\gamma(A^{\mathsf{T}}\vec{y})$, i.e. γ is also eigenvalue of $A^{\mathsf{T}}\cdot A$ and a related eigenvector is $A^{\mathsf{T}}\vec{y}$

$$\Rightarrow\quad \sigma([A\cdot A^{\mathsf{T}}]_{m\times m})-\{0\}\subseteq\sigma([A^{\mathsf{T}}A]_{n\times n})$$

Examples of eigenvalues/eigenvectors of 2D linear maps: rotation

$$Ax = \lambda x$$

```
syms th real; A=[cos(th) -sin(th);sin(th) cos(th)]
A =
[cos(th), -sin(th)]
[sin(th),  cos(th)]
syms lambda; B=A-lambda*eye(2);
d=simplify(det(B))
d =
lambda^2 - 2*cos(th)*lambda + 1
S=simplify(solve(d,lambda),10)
S =
cos(th) + (cos(th)^2 - 1)^(1/2)
cos(th) - (cos(th)^2 - 1)^(1/2)
```

A: 2D rotation matrix

```
d=simplify(charpoly(A,lambda))
```

```
[V,D]=eig(A)
V =
[(cos(th)-sin(th)*1i)/sin(th)-cos(th)/sin(th),
   (cos(th)+sin(th)*1i)/sin(th)-cos(th)/sin(th)]
[1,                                          1]
D =
[cos(th) - sin(th)*1i,                   0]
[                   0, cos(th) + sin(th)*1i]
```

$\cos(\theta)^2 - 1 \leq 0$ ⇒ complex eigenvalues, except for $\cos(\theta) = \pm 1$, i.e. $\theta=0$ or $\theta=\pi$

Indeed, in general, there is no real direction that remains the same after the plane rotation.

$\theta=0$: ⇒ identity map ⇒ $\lambda=1$ and eigenspace$=\mathbb{R}^2$

all the vectors in $\mathbb{R}^2$ remain the same

$\theta=\pi$: ⇒ reflection across **O** ⇒ $\lambda=-1$ and eigenspace$=\mathbb{R}^2$

each vector in $\mathbb{R}^2$ becomes its opposite;

but the eigenspace remains the same

Examples of eigenvalues/eigenvectors of 2D linear maps:

orthogonal reflection across x-axis

```
A=[1 0;0 -1];
syms lambda; B=A-lambda*eye(2);
d=simplify(det(B))
d =
lambda^2 - 1
S=simplify(solve(d,lambda),10)
S =
-1
 1
```

```
d=simplify(charpoly(A,lambda))
```

$$\boxed{Ax=\lambda x}$$

$$A=\begin{pmatrix}1&0\\0&-1\end{pmatrix}$$

```
[V,D]=eig(A)
V =
       0         1
       1         0
D =
      -1         0
       0         1
```

λ_k

$\lambda=1$: $\Rightarrow$ eigenspace=x-axis $\Rightarrow$

only vectors on x-axis remain the same

```
B1=subs(B,lambda,1);
N1=null(B1)
N1 =
1
0
```

$\lambda=-1$: $\Rightarrow$ eigenspace=y-axis $\Rightarrow$

only vectors on y-axis become their opposite;
but the eigenspace remains the same

```
B2=subs(B,lambda,-1);
N2=null(B2)
N2 =
0
1
```

Examples of eigenvalues/eigenvectors of 2D linear maps: orthogonal reflection across the line $\boldsymbol{r}$=span$\{\underline{a}\}$=span$\{(2,1)^T\}$

$$y = t_A(x) = \left[\frac{2}{\|a\|^2} a a^T - I_2\right] x = \left[\frac{2}{5}\begin{pmatrix}2\\1\end{pmatrix}\begin{pmatrix}2 & 1\end{pmatrix} - \begin{pmatrix}1 & 0\\0 & 1\end{pmatrix}\right] x = \begin{pmatrix}\frac{3}{5} & \frac{4}{5}\\ \frac{4}{5} & -\frac{3}{5}\end{pmatrix} x$$

```
a=[2 1]';
A=2/norm(a)^2*a*a' - eye(2);
syms lambda; B=A-lambda*eye(2);
d=simplify(det(B))
d =
lambda^2 - 1
S=simplify(solve(d,lambda),10)
S =
-1
 1
```

eigenvalues

```
B1=subs(B,lambda,1);
N1=null(B1)
N1 =
2
1
```

eigenspace basis of $\lambda=1$

```
B2=subs(B,lambda,-1);
N2=null(B2)
N2 =
-1/2
   1
```

eigenspace basis of $\lambda=-1$

```
[V,D]=eig(sym(A))
V =
[2, -1/2]
[1,    1]
D =
[1,  0]
[0, -1]
```

$$\boxed{Ax = \lambda x}$$

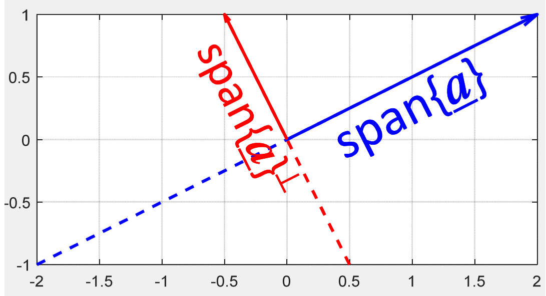
$\lambda=1$: ⇒ eigenspace = span{**a**} ⇓

only vectors on the reflection axis remain the same

$\lambda=-1$: ⇒ eigenspace = span{**a**}$^\perp$ ⇓

only vectors on span{**a**}$^\perp$ become their opposite

the eigenspace remains the same



The eigenvalues of a 2D orthogonal reflection over a generic line $\boldsymbol{r} = \text{span}\{\underline{a}\}$ are always $\lambda = \pm 1$

Proof

Let $\underline{a} = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}$; then the orthogonal reflection across $\boldsymbol{r} = \text{span}\{\underline{a}\}$ is:

$$y = t_A(x) = \underbrace{\left[\frac{2}{\|a\|^2} a a^T - I_2\right]}_{A} x = \begin{pmatrix} \dfrac{a_1^2 - a_2^2}{a_1^2 + a_2^2} & \dfrac{2a_1 a_2}{a_1^2 + a_2^2} \\ \dfrac{2a_1 a_2}{a_1^2 + a_2^2} & -\dfrac{a_1^2 - a_2^2}{a_1^2 + a_2^2} \end{pmatrix} x$$

Its characteristic polynomial is:

$$\det(A - \lambda I) = \det \begin{pmatrix} \dfrac{a_1^2 - a_2^2}{a_1^2 + a_2^2} - \lambda & \dfrac{2a_1 a_2}{a_1^2 + a_2^2} \\ \dfrac{2a_1 a_2}{a_1^2 + a_2^2} & -\dfrac{a_1^2 - a_2^2}{a_1^2 + a_2^2} - \lambda \end{pmatrix} = \lambda^2 - \left[\left(\frac{a_1^2 - a_2^2}{a_1^2 + a_2^2}\right)^2 + \left(\frac{2a_1 a_2}{a_1^2 + a_2^2}\right)^2\right] = \lambda^2 - 1$$

```
syms a [2 1] real;  A=simplify(2/norm(a)^2*a*a' - eye(numel(a)))
A =
[(a1^2 - a2^2)/(a1^2 + a2^2),      (2*a1*a2)/(a1^2 + a2^2)]
[    (2*a1*a2)/(a1^2 + a2^2), -(a1^2 - a2^2)/(a1^2 + a2^2)]
p=simplify(charpoly(A))
p =
[1, 0, -1]
disp(roots(p))
-1
 1
```

```
syms x real
Px=charpoly(A,x)
Px =
x^2 - 1
```

The eigenvalues are always $\lambda = \pm 1$

The eigenspaces of a 2D orthogonal reflection over a generic line $\boldsymbol{r} = \text{span}\{\underline{\boldsymbol{a}}\}$ are: $\text{span}\{\underline{\boldsymbol{a}}\}$ and $\text{span}\{\underline{\boldsymbol{a}}\}^{\perp}$

Proof

The eigenspace V_1 related to $\lambda = +1$ is given by:

$$A - \lambda I|_{\lambda=1} = \begin{pmatrix} \frac{a_1^2 - a_2^2}{a_1^2 + a_2^2} - 1 & \frac{2a_1 a_2}{a_1^2 + a_2^2} \\ \frac{2a_1 a_2}{a_1^2 + a_2^2} & -\frac{a_1^2 - a_2^2}{a_1^2 + a_2^2} - 1 \end{pmatrix} = \frac{2}{a_1^2 + a_2^2} \begin{pmatrix} a_2 & 0 \\ 0 & a_1 \end{pmatrix} \begin{pmatrix} -a_2 & a_1 \\ a_2 & -a_1 \end{pmatrix}$$

$$\mathscr{N}(A - 1I) = \mathscr{N} \begin{pmatrix} -a_2 & a_1 \\ a_2 & -a_1 \end{pmatrix} = \text{span}\left\{ \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \right\}$$

```
…
syms lambda; B=A-lambda*eye(2)
B1=subs(B,lambda,1); N1=null(B1)
N1 =
a1/a2
    1
```

eigenspace basis for $\lambda=1$

The eigenspace V_2 related to $\lambda = -1$ is given by:

$$A - \lambda I|_{\lambda=-1} = \begin{pmatrix} \frac{a_1^2 - a_2^2}{a_1^2 + a_2^2} + 1 & \frac{2a_1 a_2}{a_1^2 + a_2^2} \\ \frac{2a_1 a_2}{a_1^2 + a_2^2} & -\frac{a_1^2 - a_2^2}{a_1^2 + a_2^2} + 1 \end{pmatrix} = \frac{2}{a_1^2 + a_2^2} \begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix} \begin{pmatrix} a_1 & a_2 \\ a_1 & a_2 \end{pmatrix}$$

$$\mathscr{N}(A + 1I) = \mathscr{N} \begin{pmatrix} a_1 & a_2 \\ a_1 & a_2 \end{pmatrix} = \text{span}\left\{ \begin{pmatrix} -a_2 \\ a_1 \end{pmatrix} \right\} = V_1^{\perp}$$

```
B2=subs(B,lambda,-1); N2=null(B2)
N2 =
-a2/a1
    1
```

eigenspace basis for $\lambda=-1$
orthogonal to N1

Examples of eigenvalues/eigenvectors of 3D linear maps: orthogonal projection onto the plane $\pi=\mathrm{span}\{(1,0,1)^{\mathsf{T}},(1,1,0)^{\mathsf{T}}\}$

$$A=\begin{pmatrix}1 & 1\\ 0 & 1\\ 1 & 0\end{pmatrix} \qquad \pi=\mathscr{R}(A)$$

non-orthonormal columns

$$y=\boxed{\left[A\left(A^{\mathsf{T}}A\right)^{-1}A^{\mathsf{T}}\right]}x$$

projection matrix P⊥

orthonormal columns

$$y=\boxed{UU^{\mathsf{T}}}x$$

projection matrix P⊥

It is preferable to switch to an orthonormal basis for the plane π: in this way we can use the simplified formula for the projection matrix P⊥.

```
A=[1 0 1; 1 1 0]';
[Q,~]=qr(A,0); P1=Q*Q'
P1 =
   0.66667   0.33333   0.33333
   0.33333   0.66667  -0.33333
   0.33333  -0.33333   0.66667
O=orth(A); P2=O*O'
P2 =
   0.66667   0.33333   0.33333
   0.33333   0.66667  -0.33333
   0.33333  -0.33333   0.66667
```

P1, **P2** are singular matrices

$$\boxed{Ax=\lambda x}$$

```
[V,D] = eig(sym(P2))
V =
[ -1, 1, 1]
[  1, 1, 0]
[  1, 0, 1]
D =
[ 0, 0, 0]
[ 0, 1, 0]
[ 0, 0, 1]
```

The mapping has a double eigenvalue: $\lambda_1=1$ (related to the vectors that remain fixed) and the corresponding eigenspace is the plane itself $\mathscr{R}(A)$.

Exercises

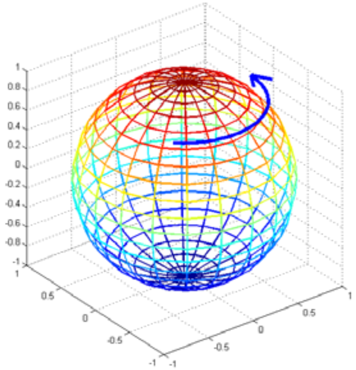
Find the eigenvalues and the eigenspaces of the 3D orthogonal reflection over the line

$$\boldsymbol{r} = \text{span}\{(2,1,1)^T\}.$$

Also by means of *Symbolic Math Toolbox*, find the eigenvalues and the eigenspaces of the 3D orthogonal reflection over a generic line $\boldsymbol{r}=\text{span}\{\underline{\boldsymbol{a}}\}$ where $\underline{\boldsymbol{a}}=(a_1,a_2,a_3)^T$.

Hint: remember the matrix form of the mapping

Also by means of *Symbolic Math Toolbox*, find the real eigenvalues and their eigenspaces of the 3D rotation around z-axis.



Matrix diagonalization

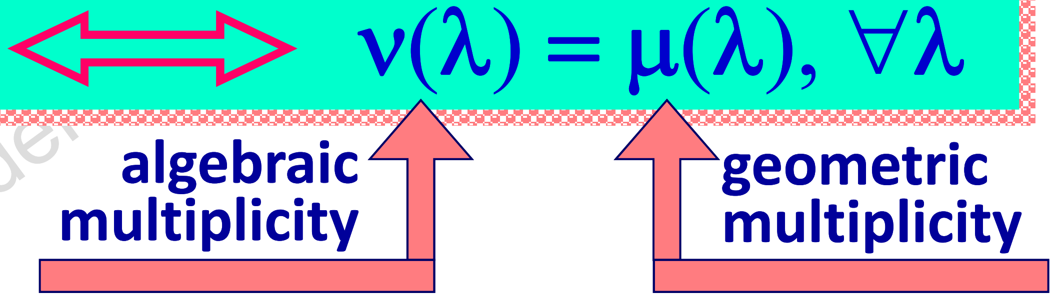
Two square matrices, of size n, A and B are called "**similar**" if there exists an invertible matrix S such that $B = S^{-1} A S$ (S: similarity matrix).

Two similar matrices have the same eigenvalues.

In general, a matrix $A_{(n\times n)}$ is "diagonalizable" if it admits n linearly independent eigenvectors, which can form a basis of $\mathbb{R}^n$, and so that they constitute the columns of S.

This occurs, for example, if all the eigenvalues of A differ.

Diagonalizability criterion



$A_{(n\times n)}$ is diagonalizable $\Longleftrightarrow$ $\nu(\lambda) = \mu(\lambda),\ \forall\lambda$

Equivalent statements:

(1) $A_{(n\times n)}$ is "diagonalizable".
(2) The sum of the geometric multiplicites equals n.
(3) $\nu(\lambda) = \mu(\lambda),\ \forall\lambda$.

Matrix diagonalization: example

$$A = \begin{pmatrix} 1/4 & 3/4 \\ 1 & 1/2 \end{pmatrix} \qquad t_A \; : \; x \in \mathbb{R}^2 \longrightarrow t_A(x) = Ax \in \mathbb{R}^2$$

eigenvalues: $\lambda_1 = -1/2$, $\lambda_2 = 5/4$ (simple roots of characteristic polynomial)

eigenspaces: $V_{\lambda_1} = \text{span}\{(-1,1)^T\}$

$V_{\lambda_2} = \text{span}\{(3,4)^T\}$

$$\begin{array}{l} \nu(\lambda_1) = \mu(\lambda_1) \\ \nu(\lambda_2) = \mu(\lambda_2) \end{array} \Rightarrow S = \begin{pmatrix} -1 & 3 \\ 1 & 4 \end{pmatrix}$$

Since the two eigenvectors are linearly independent, they can form a new basis for $\mathbb{R}^2$, so that each vector x can be expressed as

$$x = \alpha_1 \begin{pmatrix} -1 \\ 1 \end{pmatrix} + \alpha_2 \begin{pmatrix} 3 \\ 4 \end{pmatrix} = S\underline{\alpha}$$

$$\Rightarrow \quad t_A(x) = Ax = AS\underline{\alpha} = \alpha_1 A \begin{pmatrix} -1 \\ 1 \end{pmatrix} + \alpha_2 A \begin{pmatrix} 3 \\ 4 \end{pmatrix} =$$

$$\Lambda = \begin{pmatrix} \lambda_1 & \\ & \lambda_2 \end{pmatrix} \qquad = \alpha_1 \lambda_1 \begin{pmatrix} -1 \\ 1 \end{pmatrix} + \alpha_2 \lambda_2 \begin{pmatrix} 3 \\ 4 \end{pmatrix} = S\Lambda\underline{\alpha}$$

w.r.t. the new basis of $\mathbb{R}^2$ in S, the mapping t_A becomes as a non-uniform scaling

$$t_A : x \in \mathbb{R}^n, x = S\boxed{\underline{\alpha}} \longrightarrow t_A(x) = Ax \in \mathbb{R}^n, t_A(x) = AS\underline{\alpha} = S\boxed{\Lambda\underline{\alpha}}$$

$$\Leftrightarrow \quad \Lambda = S^{-1} A S$$

A has been diagonalized by S

Matrix diagonalization: MATLAB examples

$$A = \begin{pmatrix} 1/4 & 3/4 \\ 1 & 1/2 \end{pmatrix}$$

non-symmetric

numerical

```
A=[1/4 3/4;1 1/2]
A =
      0.25          0.75
         1           0.5
P=charpoly(A)
P =
 1  -0.75  -0.625
w=roots(P)
w =
            1.25
            -0.5
V1=null(A-w(1)*eye(2))
V1 =
             0.6
             0.8
V2=null(A-w(2)*eye(2))
V2 =
 -0.70711
  0.70711
S=[V1 V2];
inv(S)*A*S
ans =
 1.25   -2.22e-16
-2.77e-17   -0.5
```

V1, **V2** are normalized but not orthogonal

symbolic

```
A=sym(A)
A =
[1/4, 3/4]
[  1, 1/2]
P=charpoly(A)
P =
[1, -3/4, -5/8]
w=roots(P)
w =
-1/2
 5/4
V1=null(A-w(1)*eye(2))
V1 =
-1
 1
V2=null(A-w(2)*eye(2))
V2 =
3/4
  1
S=[V1 V2];
inv(S)*A*S
ans =
[-1/2,    0]
[   0, 5/4]
```

V1, **V2** neither normalized nor orthogonal

characteristic polynomial

eigenvalues

eigenvectors

S diagonalizes **A**

`S\A*S`

diagonalization

```
[S,d]=eig(A,'vector');
inv(S)*A*S
ans =
         -0.5    2.7756e-17
   1.1102e-16          1.25
```

diagonal matrix

```
[S,D]=eig(A);
inv(S)*A*S
ans =
[-1/2,    0]
[   0, 5/4]
```

* **A symmetric matrix is always diagonalizable.**

* Spectral Theorem for symmetric real matrices

Let A be an $n \times n$ real matrix. Then A is symmetric if, and only if, it is orthogonally diagonalizable, that is:

Q orthogonal matrix

$$\exists Q \; : \; Q^{-1}AQ = \Lambda \text{ where } \Lambda \text{ is diagonal and } Q^{T}Q = QQ^{T} = I$$

$$\Longleftrightarrow \boxed{\Lambda = Q^{T}AQ}$$

the inverse no longer needs to be computed

i.e.: “if A is a real symmetric matrix, then there exists an orthonormal basis of $\mathbb{R}^n$ formed by eigenvectors of A”

In order to find Q:

- compute eigenvalues and eigenvectors of A;
- form the diagonalizing matrix S having the eigenvectors of A as columns;
- compute Q by ortonormalizing cols of S: $\begin{cases} \text{a) compute the factorization } S = QR; \\ \text{b) apply Gram-Schmidt ortonormaliz.} \end{cases}$
- Q is the orthogonal diagonalizing matrix.

Matrix diagonalization: MATLAB examples

$A = \begin{pmatrix} 5 & 4 \\ 4 & 5 \end{pmatrix}$ symmetric

numerical

```
A=[5 4;4 5]
A =
         5         4
         4         5
P=charpoly(A)
P =
 1  -10  9
w=roots(P)
w =
     9
     1
V1=null(A-w(1)*eye(2))
V1 =
   0.70711
   0.70711
V2=null(A-w(2)*eye(2))
V2 =
  -0.70711
   0.70711
V=[V1 V2];
inv(V)*A*V
ans =
         9     0
         0     1
V'*A*V
ans =
         9     0
         0     1
```

V1, V2 are orthonormal

```
[V,d]=eig(A,'vector');
V'*A*V
ans =
         1         0
         0         9
```

diagonal matrix

characteristic polynomial

eigenvalues

eigenvectors

V diagonalizes A

V\A*V

diagonalization

symbolic

```
A=sym(A)
A =
[5, 4]
[4, 5]
P=charpoly(A)
P =
[1, -10, 9]
w=roots(P)
w =
1
9
V1=null(A-w(1)*eye(2))
V1 =
-1
 1
V2=null(A-w(2)*eye(2))
V2 =
1
1
V=[V1 V2];
inv(V)*A*V
ans =
[1, 0]
[0, 9]
V'*A*V
ans =
[2,  0]
[0, 18]
```

V1, V2 are orthogonal but non-normalized

```
O=orth(V);
O'*A*O
ans =
[1, 0]
[0, 9]
```

```
[V,D]=eig(A);
V'*A*V
ans =
[2,  0]
[0, 18]
```

Why twice the eigenvalues?

Matrix diagonalization: MATLAB `eig()` recap

non-symmetric

```
A=[1/4 3/4;1 1/2];
[V, D] = eig(A)
V =
     -0.70711      -0.6
      0.70711      -0.8
D =
         -0.5         0
            0      1.25
disp(V'*V)
            1   -0.14142
  -0.14142             1
disp(V*V')
         0.86      -0.02
     -0.02          1.14
disp(inv(V)*A*V)
        -0.5   2.7756e-17
  1.1102e-16         1.25
```

```
disp(norm(V(:,1)))
          1
disp(norm(V(:,2)))
          1
```

For a non-symmetric matrix, **eig** (num) function returns **V** as a non-orthogonal matrix, but with normalized columns

symmetric

```
A=[5 4;4 5];
[V, D] = eig(A)
V =
     -0.70711      0.70711
      0.70711      0.70711
D =
      1      0
      0      9
disp(V'*V)
           1            0
           0            1
disp(V*V')
           1            0
           0            1
disp(V'*A*V)
             1          0
             0          9
```

For a symmetric matrix, **eig** (num) function returns **V** as an orthogonal matrix

Exercise

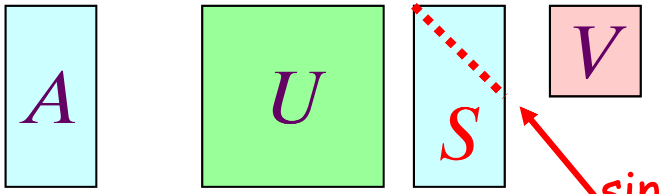
Are the following maps **diagonalizable**?

- A horizontal shear: $A = \begin{pmatrix} 1 & r \\ 0 & 1 \end{pmatrix}$, $r=2$
- A rotation: $A = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$
- An orthogonal projection: $A = \frac{1}{\|a\|^2} aa^\top$, $a=[2,1]^\top$

Connection between SVD and diagonalization

Singular Value Decomposition of $A_{m\times n}$

$$A_{m\times n} = U_{m\times m} S_{m\times n} V^{\mathsf{T}}_{n\times n}$$



singular values σ_k of A

U,V orthogonal real square matrices (or unitary complex square matrices)

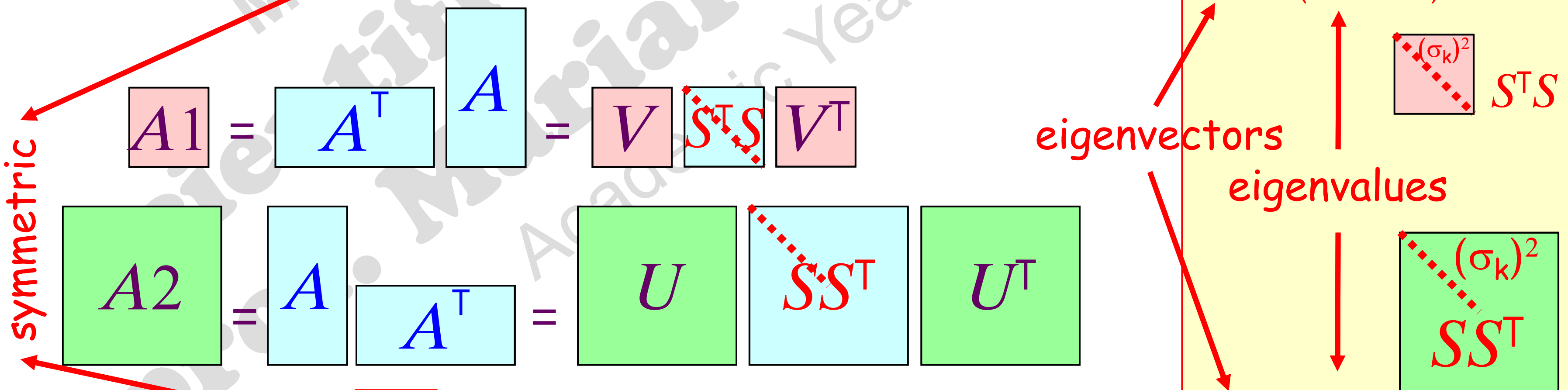
$$U^{\mathsf{T}}U = I_{m\times m}$$

$$V^{\mathsf{T}}V = I_{n\times n}$$

MATLAB ′ is the conjugate transpose operator, while .′ is the transpose operator

S is uniquely determined by A

$$\boxed{A1} = A^{\mathsf{T}}A = V\cdot S^{\mathsf{T}}\cdot U^{\mathsf{T}}\cdot U\cdot S\cdot V^{\mathsf{T}} = V\cdot\left(S^{\mathsf{T}}\cdot S\right)\cdot V^{\mathsf{T}}$$



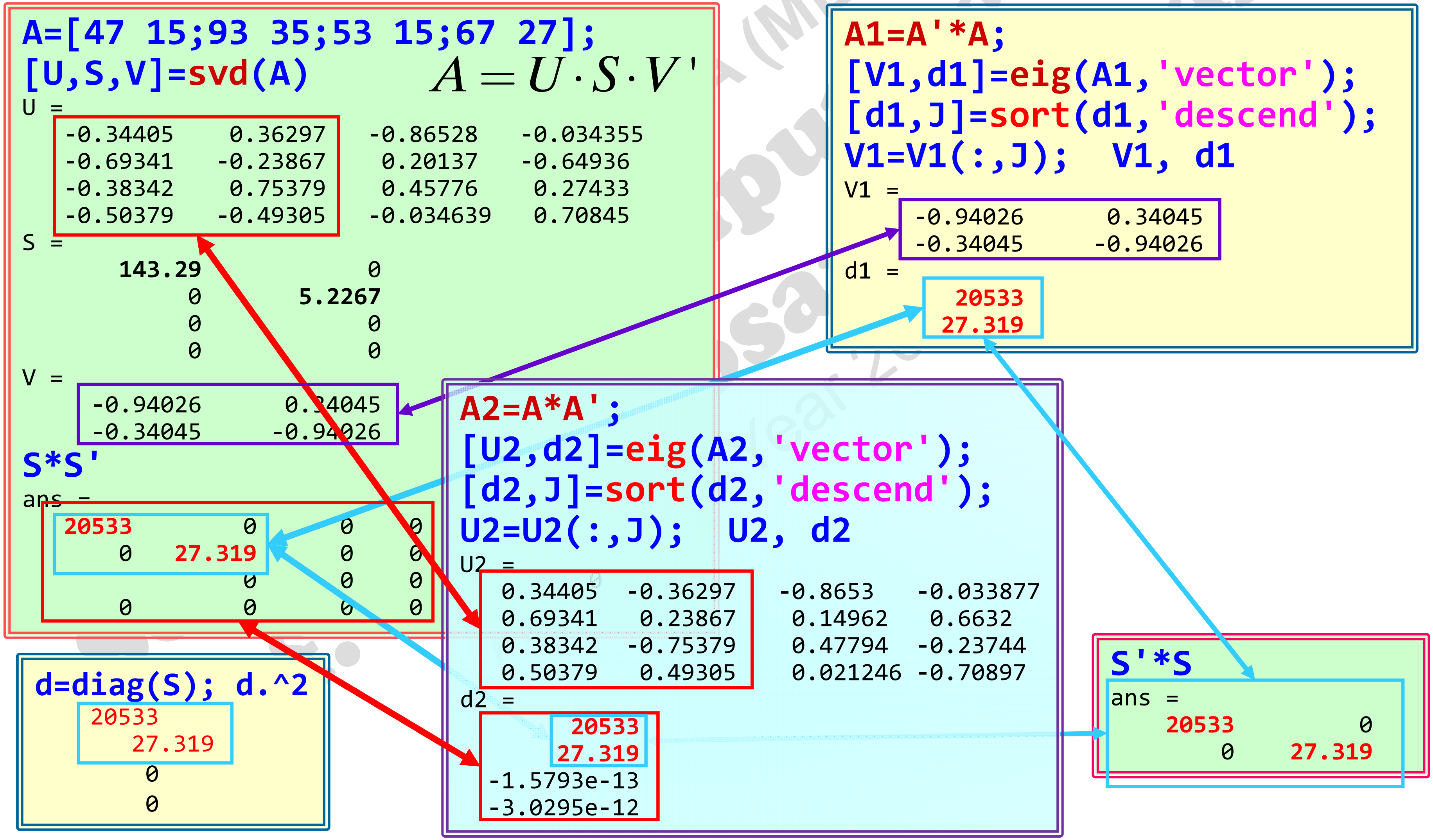
diagonalization

$$\boxed{A2} = A\cdot A^{\mathsf{T}} = U\cdot S\cdot V^{\mathsf{T}}\cdot V\cdot S^{\mathsf{T}}\cdot U^{\mathsf{T}} = U\cdot\left(S\cdot S^{\mathsf{T}}\right)\cdot U^{\mathsf{T}}$$

Connection between SVD and diagonalization

Example

$$A=\begin{pmatrix}47 & 15\\ 93 & 35\\ 53 & 15\\ 67 & 27\end{pmatrix}\qquad A^{\mathsf{T}}\cdot A=\begin{pmatrix}18156 & 6564\\ 6564 & 2404\end{pmatrix}\qquad A\cdot A^{\mathsf{T}}=\begin{pmatrix}2434 & 4896 & 2716 & 3554\\ 4896 & 9874 & 5454 & 7176\\ 2716 & 5454 & 3034 & 3956\\ 3554 & 7176 & 3956 & 5218\end{pmatrix}$$



```
A=[47 15;93 35;53 15;67 27];
[U,S,V]=svd(A)
U =
   -0.34405     0.36297    -0.86528    -0.034355
   -0.69341    -0.23867     0.20137    -0.64936
   -0.38342     0.75379     0.45776     0.27433
   -0.50379    -0.49305    -0.034639    0.70845
S =
     143.29           0
          0      5.2267
          0           0
          0           0
V =
   -0.94026     0.34045
   -0.34045    -0.94026
S*S'
ans =
  20533         0        0    0
      0    27.319        0    0
      0         0        0    0
      0         0        0    0
```

$$A=U\cdot S\cdot V'$$

```
A1=A'*A;
[V1,d1]=eig(A1,'vector');
[d1,J]=sort(d1,'descend');
V1=V1(:,J);  V1, d1
V1 =
   -0.94026        0.34045
   -0.34045       -0.94026
d1 =
      20533
     27.319
```

```
A2=A*A';
[U2,d2]=eig(A2,'vector');
[d2,J]=sort(d2,'descend');
U2=U2(:,J);  U2, d2
U2 =
    0.34405   -0.36297   -0.8653    -0.033877
    0.69341    0.23867    0.14962    0.6632
    0.38342   -0.75379    0.47794   -0.23744
    0.50379    0.49305    0.021246  -0.70897
d2 =
        20533
       27.319
  -1.5793e-13
  -3.0295e-12
```

```
d=diag(S); d.^2
   20533
   27.319
        0
        0
```

```
S'*S
ans =
     20533          0
         0     27.319
```

Applications of diagonalization: example 1

A^n: n^{th} power of a matrix

In order to compute A^n, the most efficient algorithm is to diagonalize A, and then compute the power as:

$$\Lambda = \begin{pmatrix} a & 0 & 0 & \cdots & 0 \\ 0 & b & 0 & \cdots & 0 \\ 0 & 0 & c & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & z \end{pmatrix}$$

$$A = S\Lambda S^{-1} \Leftrightarrow A^n = AA\cdots A = (S\Lambda S^{-1})^n = S\Lambda^n S^{-1}$$

$$\Downarrow$$

$$\Lambda^n = \begin{pmatrix} a^n & 0 & 0 & \cdots & 0 \\ 0 & b^n & 0 & \cdots & 0 \\ 0 & 0 & c^n & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & z^n \end{pmatrix}$$

What is the matrix power used for?

1a) In dynamic systems **transition matrices** are used to pass from a state to another.

$$\begin{cases} x_0 & \text{initial state} \\ x_{k+1} = A\,x_k \end{cases}$$

(A Transition matrix of a Markov chain)

Consider the dynamic system consisting of the population movement between a city and its suburbs. Let **x=[c;s]**$\in\mathbb{R}^2$ be the state population vector, where **c** is the population of the city and **s** is the population of the suburbs. For simplicity, we assume that **c+s=1**, i.e., **c** and **s** are percentages of the total population. Suppose that in the year 1900, the city population was $\mathbf{c_0}$ and the suburban population was $\mathbf{s_0}$ and suppose that after each year 5% of the city's population moves to the suburbs and that 3% of the suburban population moves to the city. Hence, the population in the city is:

```
A=[0.95 0.03;0.05 0.97];
[c1;s1]=A*[c0;s0]; % in year 1901
[c2;s2]=A*[c1;s1]; % in year 1902
```

⇒

```
[c2;s2]=A^2*[c0;s0];  % in year 1902
 ...
[c;s]=A^100*[c0;s0]; % in year 2000
```

Application of diagonalization: example 1a (cont.)

A^n: n^{th} power of a matrix

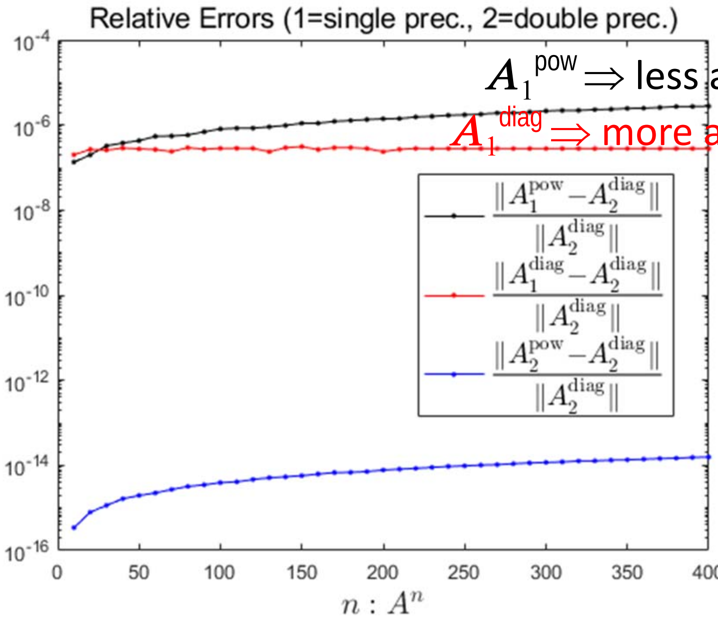
```
A2=[0.95 0.03;0.05 0.97]; % double precision
A1=single(A2);            % single precision
[V2,D2]=eig(A2);  [V1,D1]=eig(A1);
nMIN=10; nMAX=400; nSTEP=10; nVals=(nMIN:nSTEP:nMAX)';
E22=zeros(numel(nVals),1); E12diag=zeros(numel(nVals),1); E12pow=zeros(numel(nVals),1);
for k=1:numel(nVals)
    n=nVals(k);
    A2pow=A2^n; % A^n in double prec.
    A1pow=A1^n; % A^n in single prec.
    A2diag=V2*diag(diag(D2).^n)/V2; % *inv(V2)
    A1diag=V1*diag(diag(D1).^n)/V1; % *inv(V1)
    E22(k)=norm(A2pow-A2diag)/norm(A2diag);
    E12pow(k)=norm(A1pow-A2diag)/norm(A2diag);
    E12diag(k)=norm(A1diag-A2diag)/norm(A2diag);
end
figure(1); semilogy(nVals,[E12pow E12diag E22])
figure(2); plot(nVals,E12pow./E12diag)
```

$$A^n = AA\cdots A$$

$$A^n = V\Lambda^n V^{-1}$$

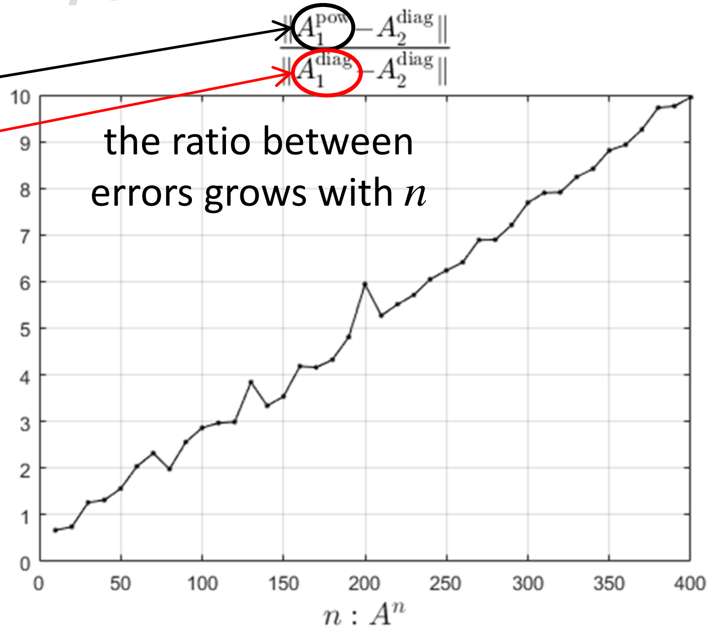
compare single prec. A_1^{pow} and A_1^{diag} with double prec. A_2^{diag}

relative errors w.r.t. A_2^{diag}



$A_1^{\text{pow}} \Rightarrow$ less accurate

$A_1^{\text{diag}} \Rightarrow$ more accurate



the ratio between errors grows with n

Applications of diagonalization: example 1b

A^n: n^{th} power of a matrix

In order to compute A^n, the most efficient algorithm is to diagonalize A, and then compute the power as:

$$A = S\Lambda S^{-1} \Leftrightarrow A^n = AA\cdots A = (S\Lambda S^{-1})^n = S\Lambda^n S^{-1}$$

$$\Lambda = \begin{pmatrix} a & 0 & 0 & \cdots & 0 \\ 0 & b & 0 & \cdots & 0 \\ 0 & 0 & c & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & z \end{pmatrix} \Downarrow \quad \Lambda^n = \begin{pmatrix} a^n & 0 & 0 & \cdots & 0 \\ 0 & b^n & 0 & \cdots & 0 \\ 0 & 0 & c^n & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & z^n \end{pmatrix}$$

What is the matrix power used for?

1b) In **graph theory**:

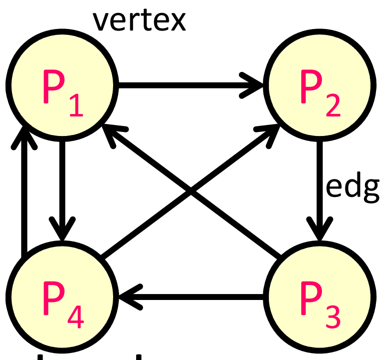
Example We consider n people $P_1, P_2, \ldots, P_n$, and form a **digraph** (**directed graph** or **oriented graph**), where a directed edge from P_i to P_j denotes that P_i can send some information to P_j. Then we write the corresponding **adjacency matrix** $\mathbf{A}$:

$$\mathbf{A} = \begin{array}{c} \\ \text{from} \end{array} \begin{array}{c} \text{to} \\ \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{pmatrix} \\ \begin{array}{cccc} P_1 & P_2 & P_3 & P_4 \end{array} \end{array} \begin{array}{c} P_1 \\ P_2 \\ P_3 \\ P_4 \end{array}$$

adjacency matrix

$P_1 \rightarrow P_2$, $P_1 \rightarrow P_4$
$P_2 \rightarrow P_3$
$P_3 \rightarrow P_1$, $P_3 \rightarrow P_4$
$P_4 \rightarrow P_1$, $P_4 \rightarrow P_2$

Graph = {vertices} ∪ {edges}



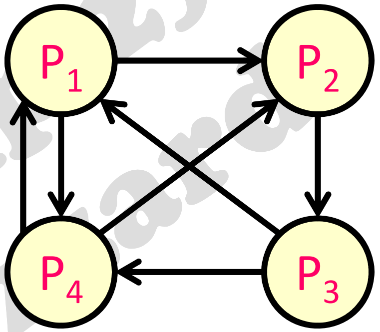
directed graph = one-way edges

Applications of diagonalization: example 1b (cont.)

A^n: n^{th} power of a matrix

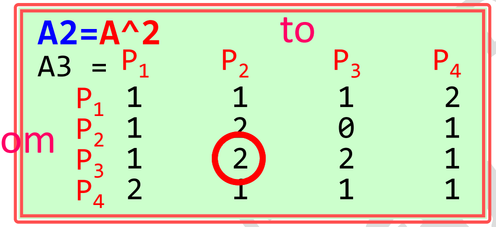
$$\mathbf{A} = \begin{array}{c|cccc|c} & P_1 & P_2 & P_3 & P_4 & \\ \hline & 0 & 1 & 0 & 1 & P_1 \\ & 0 & 0 & 1 & 0 & P_2 \\ & 1 & 0 & 0 & 1 & P_3 \\ & 1 & 1 & 0 & 0 & P_4 \end{array}$$

$P_1 \to P_2$, $P_1 \to P_4$
$P_2 \to P_3$
$P_3 \to P_1$, $P_3 \to P_4$
$P_4 \to P_1$, $P_4 \to P_2$



How can we interpret $\mathbf{A}^3$, and more generally $\mathbf{A}^k$?

```
A=[0 1 0 1;0 0 1 0;1 0 0 1;1 1 0 0];
```



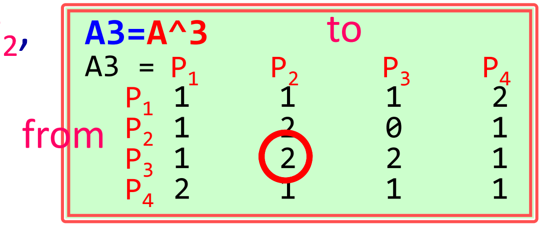
```
     A2=A^2          to
     A3 = P1     P2     P3     P4
       P1  1      1      1      2
from   P2  1      2      0      1
       P3  1      2      2      1
       P4  2      1      1      1
```

The entry $\mathbf{a}_{32}$=2 in $\mathbf{A}^2$ denotes that P_3 can send information to P_2, in 2 stages, by 2 different ways:

$$\underset{\text{stage 1}}{P_3 \to P_4} \wedge \underset{\text{stage 2}}{P_4 \to P_2} \quad \text{or} \quad \underset{\text{stage 1}}{P_3 \to P_1} \wedge \underset{\text{stage 2}}{P_1 \to P_2}$$

The entry $\mathbf{a}_{32}$=2 in $\mathbf{A}^3$ denotes that P_3 can send informations to P_2, in 3 stages, by 2 different ways:

$$P_3 \to P_4 \to P_1 \to P_2 \quad \text{or} \quad P_3 \to P_1 \to P_4 \to P_2$$



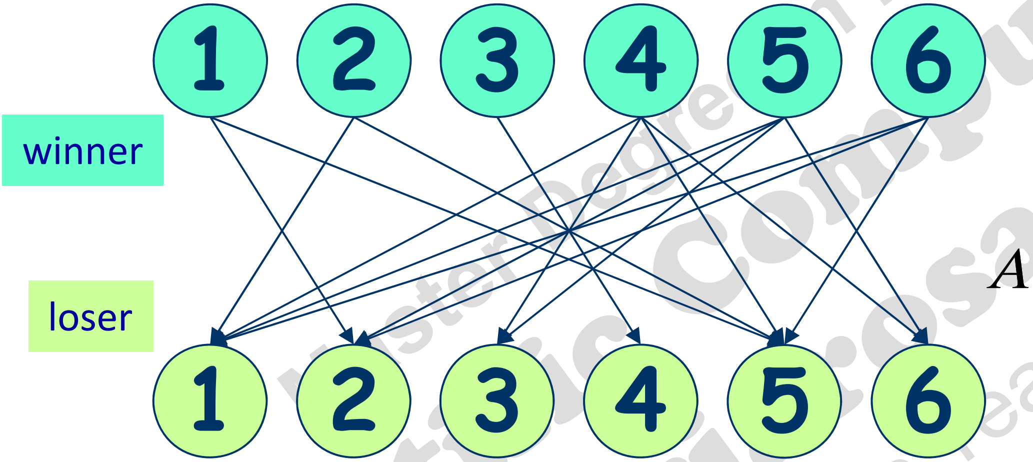
```
     A3=A^3          to
     A3 = P1     P2     P3     P4
       P1  1      1      1      2
from   P2  1      2      0      1
       P3  1      2      2      1
       P4  2      1      1      1
```

In general, the number of walks of length k from P_i to P_j is given by the entry $\mathbf{a}_{ij}$ of the matrix $\mathbf{A}^k$: $\mathbf{a}_{ij} = (\mathbf{A}^k)_{ij}$

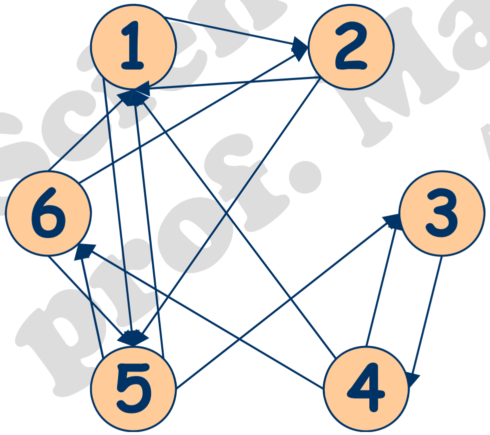
In general, the number of ways in which P_i can send information to P_j in **at most** k stages is given by the entry a_{ij} of the matrix: $\mathbf{A} + \mathbf{A}^2 + \mathbf{A}^3 + \cdots + \mathbf{A}^k$

Applications of diagonalization: example 2

We want to compute the ranking of the best soccer teams in a tournament. We form the bipartite directed graph where the nodes are the teams, and the directed edge from node i to node j denotes that team i won against team j. The graph is represented by its adjacency matrix (non-symmetric).



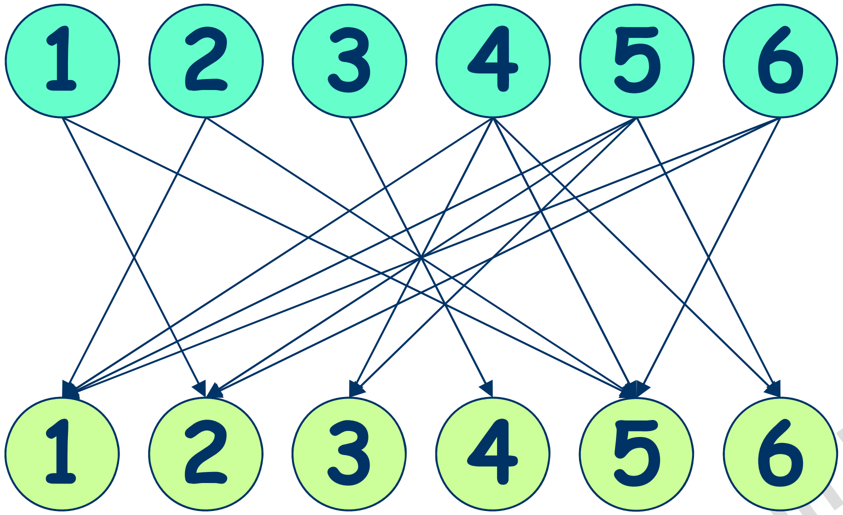
$$A = \begin{array}{cccccc|c|c} 1 & 2 & 3 & 4 & 5 & 6 & & \\ \hline 0 & 1 & 0 & 0 & 1 & 0 & 1 & \text{wins} \\ 1 & 0 & 0 & 0 & 1 & 0 & 2 & \\ 0 & 0 & 0 & 1 & 0 & 0 & 3 & \\ 1 & 0 & 1 & 0 & 1 & 1 & 4 & \\ 1 & 1 & 1 & 0 & 0 & 1 & 5 & \\ 1 & 1 & 0 & 0 & 1 & 0 & 6 & \end{array}$$



A describes the tournament

James P. Keener – *The Perron-Frobenius Theorem and the ranking of football teams*. SIAM Review, 35 (1), 1993
http://stat.wharton.upenn.edu/~steele/Courses/956/Ranking/RankingFootballSIAM93.pdf

Dario A. Bini – *Il problema del PageRank*. Appunti del corso di Calcolo Scientifico (2015)
https://pagine.dm.unipi.it/bini/Didattica/CalSci/dispense/google.pdf



$$A=\begin{array}{cccccc|c} 1 & 2 & 3 & 4 & 5 & 6 & \\ \hline 0 & 1 & 0 & 0 & 1 & 0 & 1 \quad \text{wins} \\ 1 & 0 & 0 & 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 1 & 0 & 0 & 3 \\ 1 & 0 & 1 & 0 & 1 & 1 & 4 \\ 1 & 1 & 1 & 0 & 0 & 1 & 5 \\ 1 & 1 & 0 & 0 & 1 & 0 & 6 \end{array}$$

```
A=[0 1 0 0 1 0 ...];
[V,d]=eig(A,'vector')
[d,J]=max(abs(d));
v=abs(V(:,J));
[v,J]=sort(v,'descend');
disp(' team ranking')
disp([J v])
team        ranking
   4        0.56648
   5        0.49581
   6        0.43459
   1        0.31328
   2        0.31328
   3        0.21934
```

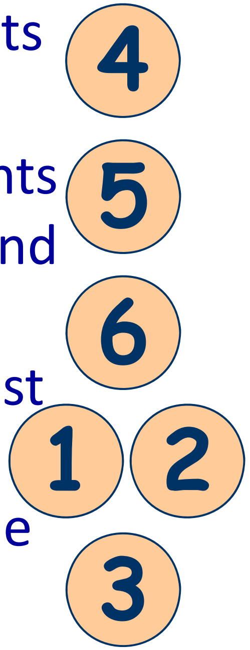
The maximum modulus eigenvalue is

$$\lambda=2.58...$$

and its related eigenvector represents the ranking.

We sort the modulus of the components of this eigenvector in descend order, and we obtain the score of each team.

According to this ranking system, the best team is the 4th, followed by the 5th, 6th, then tied first and second, and as last the 3rd.



If we change the adjacency matrix so that the element A_{ij} denotes the goal difference (>0), we can then take advantage of additional information that leads to a change in the ranking.

$$A = \begin{array}{cccccc|l} 1 & 2 & 3 & 4 & 5 & 6 & \\ \hline 0 & 1 & 0 & 0 & 1 & 0 & 1 \quad \text{wins} \\ 1 & 0 & 0 & 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 1 & 0 & 0 & 3 \\ 1 & 0 & 1 & 0 & 1 & 1 & 4 \\ 1 & 1 & 2 & 0 & 0 & 2 & 5 \\ 1 & 1 & 0 & 0 & 1 & 0 & 6 \end{array}$$

```
A=[0 1 0 0 1 0 ...];
[V,D]=eig(A);
[d,J]=max(abs(diag(D)));
v=abs(V(:,J));
[v,J]=sort(v,'descend');
disp(' team ranking')
disp([J v])
team          ranking
   5          0.60052
   4          0.50336
   6          0.41068
   2          0.30689
   1          0.30689
   3          0.17024
```

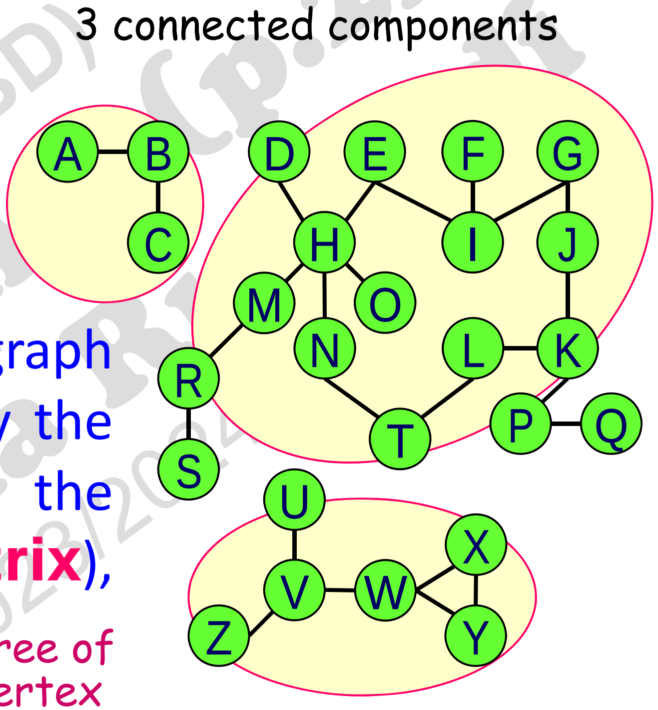
now
5
4
6
1
2
3

Taking goal difference into account, the ranking changes!

before
4
5
6
1, 2
3

Applications of diagonalization: example 3

Detect the number of
connected components
of a graph



The number of connected components of a graph (of nodes V_k, and set of edges E) is given by the multiplicity of $\lambda=0$ as an eigenvalue of the **Laplacian matrix** L (or **Kirchhoff matrix**), which is defined as

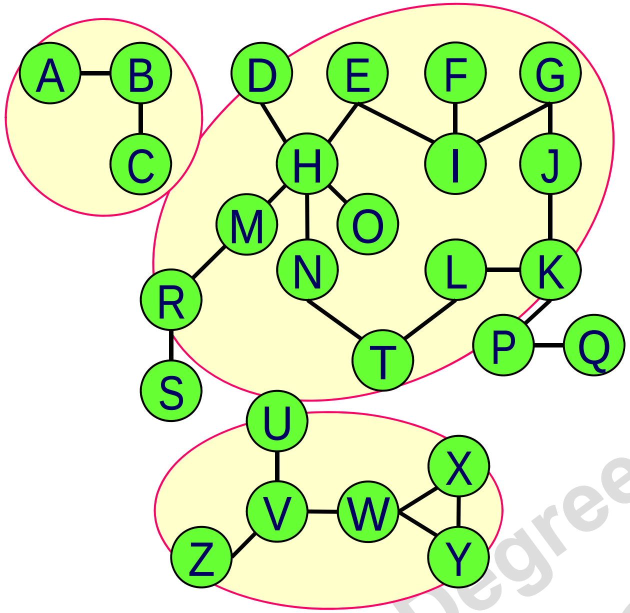
degree of a vertex

$$L=\left(\ell_{ij}\right)=\begin{cases}\deg\left(V_i\right) & i=j\\ -1 & i\neq j\wedge\left(V_i,V_j\right)\in E\\ 0 & \text{otherwise}\end{cases}$$

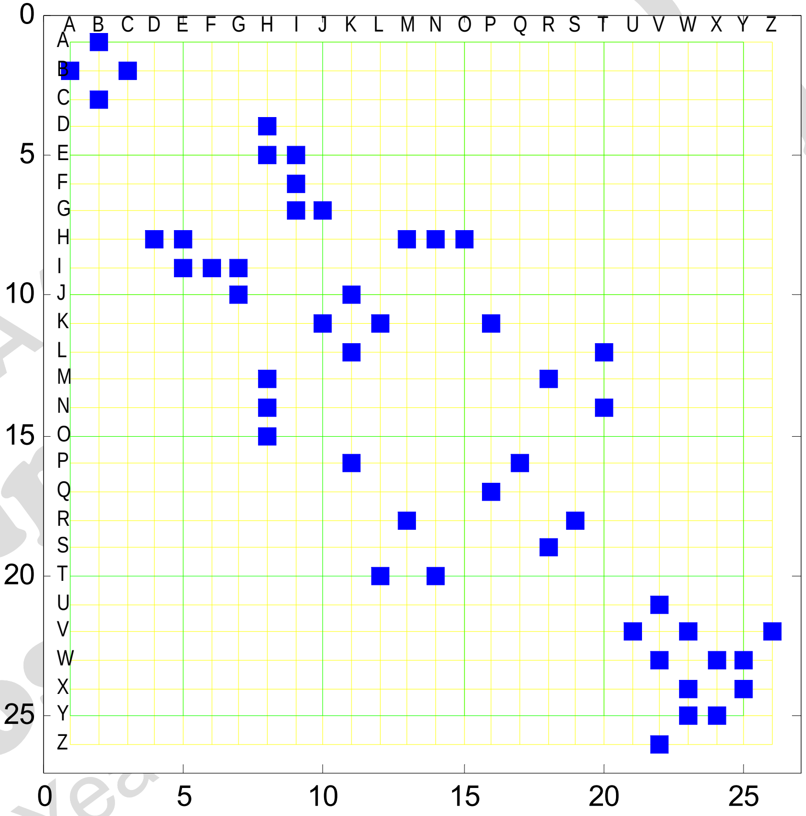
degree of a vertex: it is the number of edges that are incident to the vertex

The Laplacian matrix L is computed as $L=D-A$, i.e. the difference between the degree matrix D (a diagonal matrix which contains information about the degree of each vertex - that is, the number of edges attached to each vertex) and the adjacency matrix A of the graph.

undirected graph



adjancency matrix



spy(A)

symmetric matrix

```
A = [0 1 0 0 0 0 ...];
spy(A)
deg=diag(sum(A));
L = deg - A;
d=abs(eig(L));
J=find(d < 1e-8);
numel(J)
ans =
       3
```

find the algebraic multiplicity of the null eigenvalue

otherwise

```
d=abs(eigs(L,25,'smallestabs'));
```

eigs: subset of the 25 smallest eigenvalues

Exercise

Write a MATLAB function to detect the number of connected components of a graph, given its adjacency matrix as input.

Download the file `graph2.mat` for an adjacency matrix, or use another of your choice.