Lecture 21: Diagonalizable matrices (class)

Reading: Meyer 7.5 Normal matrices
7.6 Positive semi-definite matrices

SPECTRAL THEORY

Questions:	Answers
① What matrices have an eigenvector?	all square matrices
② What matrices can be diagonalized? (ie, they have a full basis of eigenvectors, $A = UDU^{-1}$)	diagonalizable matrices including matrices with n distinct eigenvalues (multiplicity 1)
③ Which can be diagonalized with a basis of orthogonal eigenvectors?	? TODAY

THE "POWER METHOD" FOR FINDING THE LARGEST EIGENVALUE EIGENVECTOR

Matlab eig( ) and eigs(,) commands

```
>> help eig
 eig    Eigenvalues and eigenvectors.
    E = eig(A) produces a column vector E containing the eigenvalues of
    a square matrix A.

    [V,D] = eig(A) produces a diagonal matrix D of eigenvalues and
    a full matrix V whose columns are the corresponding eigenvectors
    so that A*V = V*D.

>> help eigs
 eigs  Find a few eigenvalues and eigenvectors of a matrix
    D = eigs(A) returns a vector of A's 6 largest magnitude eigenvalues.
    A must be square and should be large and sparse.
```

```
>> help sprandsym
 sprandsym Sparse random symmetric matrix.
    R = sprandsym(S) is a symmetric random matrix whose lower triangle
        and diagonal have the same structure as S.  The elements are
        normally distributed, with mean 0 and variance 1.

    R = sprandsym(n,density) is a symmetric random, n-by-n, sparse
        matrix with approximately density*n*n nonzeros; each entry is
        the sum of one or more normally distributed random samples.
```

Example:

```
>> n = 2000;
>> density = 10/n;      (roughly 10 nonzero entries per row)
>> A = sprandsym(n, density);
>> tic; eig(A); toc
Elapsed time is 18.907694 seconds.
```

find all e-values

```
>> n = 10^6;
>> density = 10/n;
>> A = sprandsym(n, density);
>> tic; eigs(A, 1); toc
Elapsed time is 32.109028 seconds.
```

find largest-magnitude e-value

How this works

Let A be a diagonalizable matrix (for simplicity)
Sort its eigenvalues by magnitude:

$$|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n| .$$

Algorithm (simplest version)

① Start with a "generic" vector $\vec{x}_0$
② Repeat for $t = 1, 2, 3, \ldots$
let $\vec{x}_t = A\vec{x}_{t-1} = A^t \vec{x}_0$

Analysis:

Expand $\vec{x}_0$ in the basis of eigenvectors $\underset{\lambda_1}{\vec{v}_1}, \ldots, \underset{\lambda_n}{\vec{v}_n}$:

$$\vec{x}_0 = \sum_{i}^{n} c_i \vec{v}_i$$

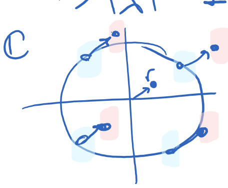
e.g. any unitary matrix U

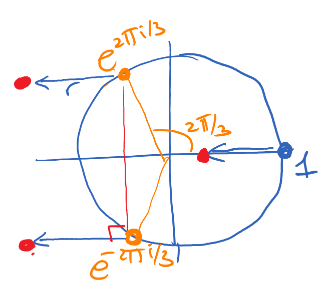
$U\vec{v} = \lambda\vec{v}$

$\|v\| = \|U\vec{v}\| = |\lambda| \cdot \|v\|$

$\Rightarrow |\lambda| = 1$

$\mathbb{C}$





$$P = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

$\lambda_1 \qquad \lambda_n$

$$\vec{x}_0 = \sum_{j=1}^{n} c_j \vec{v}_j$$

$$\Rightarrow A^t \vec{x}_0 = \sum_j c_j A^t \vec{v}_j = \sum_j c_j \lambda_j^t \vec{v}_j$$

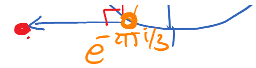
$$= c_1 \lambda_1^t \vec{v}_1 + \lambda_1^t \underbrace{\sum_{j \geq 2} c_j \left(\frac{\lambda_j}{\lambda_1}\right)^t \vec{v}_j}_{\xrightarrow{t\to\infty} 0}$$

if $|\lambda_1| > \max_{j \geq 2} |\lambda_j|$

$$\xrightarrow{t\to\infty} \lambda_1^t (c_1 \vec{v}_1)$$

$$P = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

$P^3 = I$



Solution: Instead of P, consider $\underbrace{Q}_{\lambda + r} = \underbrace{P}_{\lambda} + \underbrace{r}_{\in \mathbb{C}} I$.

Problem: What if $c_1 = 0$? ← Never happens (prob. 0)

Problem: Numerical overflow

```
>> n = 3;
>> A = randn(n, n);
>>
>> v = randn(n, 1);
>> for t = 1:10000
     v = A * v;
   end
>> v

v =

   NaN
   NaN
   NaN
```

numbers got too big!

Solution: renormalize occasionally

```
>> v = randn(n, 1);
>> for t = 1:10000
     v = A * v;
     v = v / norm(v);
   end
>> v

v =

    0.3950
    0.7841
    0.4786
```

here I do it every round

```
>> (A * v) ./ v

ans =

    1.1099
    1.1099
    1.1099
```

Problem: What if $|\lambda_1| = |\lambda_2|$?

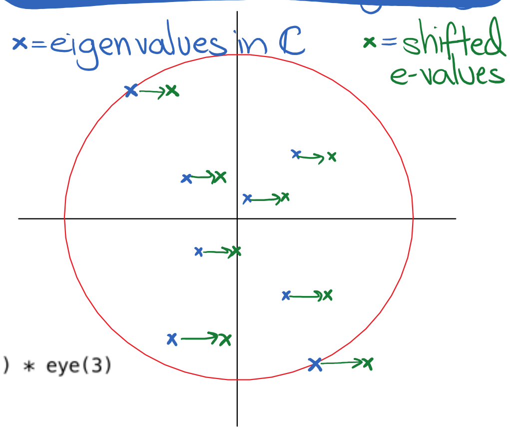
Example: Permutation matrix

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

A =

     0     1     0
     0     0     1
     1     0     0
```

[redacted]




```
>> v = randn(3, 1);
>> for t = 1:10000
     v = A * v;
   end
>> v

v =

    0.3426
    3.5784
   -0.4336
```

← not an e-vector!

```
>> Ashift = A + .01*randn() * eye(3)

Ashift =

    0.0163    1.0000         0
         0    0.0163    1.0000
    1.0000         0    0.0163

>> v = randn(3, 1);
>> for t = 1:10000
     v = Ashift * v;
```

```
>> v = randn(3, 1);
>> for t = 1:10000
    v = Ashift * v;
    v = v / v(1);
   end
>> v

v =

    1.0000
    1.0000
    1.0000
```

different way to prevent blowup

This works more reliably if the e-values are all real

Problem: What about other e-values?

We'll see...

Smallest magnitude e-value

eigs(A, 1, 'sm') finds smallest magnitude

Exercise: How does this work?

$x \leftarrow$ random vector
for $t = 1, 2, \ldots$
  $x \leftarrow A \backslash x$
  $x \leftarrow x / \|x\|$

$\longleftarrow$ don't do $\vec{v} \leftarrow A^{-1}\vec{v}$
precomputing the LU decomposition will be faster

Exercise: Using Matlab, find the e-value closest to 10.

Answer 1: eigs(A-10*eye(n), 1, 'SM') $\mathbb{C}$

Answer 2: Using the power method:

```
B = A - 10*eye(n);
x = randn(n,1);
for j = 1:10000
    x = B \ x;
    x = x/norm(x);
end
```

$\leftarrow$ it would be better to check for convergence...

$\leftarrow$ compute $B^{-1}x$
it would be faster to precompute the LU decomposition of B

10

WHEN IS A MATRIX DIAGONALIZABLE?

When does it have a complete set of eigenvectors?

Exercise: Which of these matrices can be diagonalized?

$$A = \begin{pmatrix} 1 & 1 \\ 0 & 2 \end{pmatrix} \checkmark \qquad B = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \times \qquad C = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \checkmark$$

$$B = I + \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

A matrix is diagonalizable when

- for each eigenvalue, the dimension of the associated eigenspace equals the multiplicity of the eigenvalue.
- In other words, if

$$\det(A - \lambda I) = (\lambda - \lambda_1)^{\alpha_1} \cdot (\lambda - \lambda_2)^{\alpha_2} \cdot \ldots \cdot (\lambda - \lambda_k)^{\alpha_k}$$

distinct e-values $\lambda_1, \ldots, \lambda_k$, with multiplicities

$\alpha_1, \dots, \alpha_k$
$\alpha_1 + \alpha_2 + \cdots + \alpha_k = n$

$\dim N(A - \lambda_j I) = \alpha_j$ for all j

$$\Rightarrow A = \underbrace{\begin{pmatrix} \boxed{\text{basis for } N(A-\lambda_1 I)} & \boxed{\text{basis for } N(A-\lambda_2 I)} & \cdots \end{pmatrix}}_{U} \begin{pmatrix} \underbrace{\lambda_1 \ddots \lambda_1}_{\substack{\alpha_1 \\ \shortparallel \\ \dim N(A-\lambda_1 I) \\ \text{times}}} & & & \\ & \overbrace{\lambda_2 \ddots \lambda_2}^{\alpha_2 \text{ times}} & & \\ & & \lambda_3 \ddots & \end{pmatrix} U^{-1}$$

Corollary: If an $n \times n$ matrix A has n distinct eigenvalues, then A must be diagonalizable.

But not every diagonalizable matrix has n distinct eigenvalues.

e.g., $I = \begin{pmatrix} 1 & & & 0 \\ & 1 & & \\ & & \ddots & \\ 0 & & & 1 \end{pmatrix}$ or $\begin{pmatrix} 1 & & & & & 0 \\ & 1 & & & & \\ & & 1 & & & \\ & & & 2 & & \\ & & & & 2 & \\ 0 & & & & & 2 \end{pmatrix}$.

e-value 1, multiplicity n

$\lambda_1 = 1$, multiplicity $\alpha_1 = 3$
$\lambda_2 = 2$, $\alpha_2 = 3$

Even if a matrix can be diagonalized, its eigenvectors might not be orthogonal.

Eg., $\begin{pmatrix} 1 & 0 \\ 1 & 2 \end{pmatrix}$

Evalue	E-vector
2	$(0,1)$
1	$(1,-1)$

not orthogonal!

$N\left(\begin{pmatrix} 1 & 0 \\ 1 & 2 \end{pmatrix} - \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\right) = N\left(\begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}\right) = \text{Span}\left(\begin{pmatrix} 1 \\ -1 \end{pmatrix}\right)$

Why doesn't Gram-Schmidt help?

- Performed on eigenvectors with different eigenvalues, it will output orthogonal vectors spanning the same space, but they won't (in general) still be eigenvectors.

E.g., $\left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix} \right\} \xrightarrow[\text{Schmidt}]{\text{Gram-}} \left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\}$

not an eigenvector of $\begin{pmatrix} 1 & 0 \\ 1 & 2 \end{pmatrix}$!

Example:

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

$A^T A = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \neq A A^T = \begin{pmatrix} 2 & 0 \\ 0 & 0 \end{pmatrix}$

but A is diagonalizable!

$\lambda_1 = 1$
$\lambda_2 = 0$

(e.g., since $\text{Trace}(A) = 1 = \lambda_1 + \lambda_2$ and $\text{Det}(A) = 0 = \lambda_1 \cdot \lambda_2$)

All distinct eigenvalues $\Rightarrow$ diagonalizable.

Today: Lots of matrices are diagonalizable, with orthonormal eigenvectors.

For example, all symmetric matrices $(A = A^T)$.
(And real symmetric matrices even have real eigenvalues)

Recall: Adjoint = conjugate transpose

$\begin{pmatrix} a+bi & c+di \end{pmatrix}^{\dagger} = \begin{pmatrix} a-bi & e-fi \end{pmatrix}$

4 : minutes

4 : minutes

Recall: Adjoint = conjugate transpose

$$\begin{pmatrix} a+bi & c+di \\ e+fi & g+hi \end{pmatrix}^{\dagger} = \begin{pmatrix} a-bi & e-fi \\ c-di & g-hi \end{pmatrix}$$

(same as transpose for real matrices).

THEOREM: A has a complete, orthogonal set of eigenvectors

$$\Updownarrow$$

$$A^{\dagger}A = AA^{\dagger}.$$

(if A is real, $A^TA = AA^T$)

(**Definition**: A is "normal" $\Leftrightarrow A^{\dagger}A = AA^{\dagger}$)

Examples

real symmetric $\Rightarrow$ normal
$A = A^T$ $\quad AA^T = A^2 = A^TA$

unitary $\Rightarrow$ normal
$A^{-1} = A^{\dagger}$ $\quad A^{\dagger}A = I = AA^{\dagger}$

Proof:

$\Downarrow$: One direction is trivial.

Let $\vec{u}_1, \dots, \vec{u}_n$ be an orthonormal set of e-vectors for A

let $U = \begin{pmatrix} | & & | \\ u_1 & \cdots & u_n \\ | & & | \end{pmatrix}$, so $A = UDU^{-1}$

$\uparrow$ diagonal matrix of e-values

$$A^{\dagger}A = (UDU^{-1})^{\dagger}(UDU^{-1})$$
$$= (U^{-1})^{\dagger}D^{\dagger}U^{\dagger}(UDU^{-1})$$
$$= UD^{\dagger}DU^{-1}$$

but $U^{\dagger} = U^{-1}$

$$D^{\dagger}D = DD^{\dagger} = \begin{pmatrix} |\lambda_1|^2 & & 0 \\ & |\lambda_2|^2 & \\ 0 & & \ddots \end{pmatrix}$$

$$AA^{\dagger} = (UDU^{-1})(UDU^{-1})^{\dagger}$$
$$= UDU^{-1}UD^{\dagger}U^{\dagger}$$
$$= UDD^{\dagger}U^{-1}$$

$=$

A normal ✓

The other direction (A normal $\Rightarrow A = UDU^{-1}$) is much more interesting. First let me prove two claims:

Claim 1: A normal $\Rightarrow R(A) = R(A^T)$.

(**Note**: If $R(A) \neq R(A^T)$, then there is no hope of finding a basis of eigenvectors.
A maps rowspace, $R(A^T)$, to columnspace, $R(A)$.
If these spaces are different, then A does more than just scale some vectors.)

Proof:

$$R(A) \underset{SVD}{=} R(AA^{\dagger}) \underset{\text{normal}}{=} R(A^{\dagger}A) \underset{SVD}{=} R(A^{\dagger}) \checkmark$$

similarly $N(A) = R(A^{\dagger})^{\perp} = R(A)^{\perp} = N(A^{\dagger})$ ✓

Claim 2: A normal $\Rightarrow$ there exists a unitary matrix U such that
$$U^\dagger A U = \left(\begin{array}{c|c} D & 0 \\ \hline 0 & 0 \end{array}\right)$$
for some nonsingular matrix D (a $\operatorname{rank}(A) \times \operatorname{rank}(A)$ matrix)

Proof: By Claim 1, $R(A) = R(A^\dagger)$, and $N(A) = N(A^\dagger) = R(A)^\perp$.

Let
$$U = \left(\begin{array}{c|c} \text{o.n. basis for } R(A) & \text{o.n. basis for } N(A) \end{array}\right)$$

$$U^\dagger A U = \begin{pmatrix} \overset{B^\dagger}{\text{o.n. basis}^\dagger \text{ for } R(A)} \\ \hline \underset{C^\dagger}{\text{o.n. basis}^\dagger \text{ for } N(A)} \end{pmatrix} A \left(\begin{array}{c|c} \overset{B}{R(A)} & \overset{C}{N(A)} \end{array}\right)$$
$$= \left(\begin{array}{c|c} B^\dagger A B & B^\dagger A C \to 0 \\ \hline C^\dagger A B \to 0 & C^\dagger A C \to 0 \end{array}\right) = \left(\begin{array}{c|c} B^\dagger A B & 0 \\ \hline 0 & 0 \end{array}\right).$$

Now we're ready to prove the interesting direction:

Theorem: $A^\dagger A = A A^\dagger \Rightarrow A = \underset{\text{unitary}}{U} \underset{\text{diagonal}}{D} U^\dagger$ using **Claim 2:** $A^\dagger A = A A^\dagger \Rightarrow U^\dagger A U = \begin{pmatrix} \underset{\text{nonsingular}}{C} & 0 \\ 0 & 0 \end{pmatrix}$

Proof:

Let $\lambda_1, \lambda_2, \ldots, \lambda_k$ be A's distinct eigenvalues.

A normal $\Rightarrow A - \lambda_1 I$ normal

$\left((A-\lambda_1 I)(A-\lambda_1 I)^\dagger = (A-\lambda_1 I)^\dagger (A - \lambda_1 I)\right)$

$\overset{}{\underset{\text{Claim 2}}{\Longrightarrow}} \exists\, U_1$ s.t.
$$U_1^\dagger (A - \lambda_1 I) U_1 = \begin{pmatrix} C_1 & 0 \\ 0 & 0 \end{pmatrix}$$
$$\Rightarrow U_1^\dagger A U_1 = \begin{pmatrix} C_1 & 0 \\ 0 & 0 \end{pmatrix} + \lambda_1 \underbrace{U_1^\dagger U_1}_{I}$$
$$= \left(\begin{array}{c|c} C_1 + \lambda_1 I & 0 \\ \hline 0 & \lambda_1 I \end{array}\right)$$

Let $A_1 = C_1 + \lambda_1 I$.

Observe:
- λ_1 is not an eigenvalue of A_1 (or $A_1 - \lambda_1 I$ would be singular)
- $\lambda_2, \ldots, \lambda_k$ are still eigenvalues of A_1

(since conjugating A by U does)

(or $A_1 - \lambda_1 I$ would be singular)

- $\lambda_2, \ldots, \lambda_k$ are still eigenvalues of A_1

  (since conjugating A by U_1 does not change the set of e-values, and $\lambda_2, \ldots, \lambda_k$ are definitely not e-values of the second block $\lambda_1 I$)

- A_1 is normal!

  (b/c $U_1^+ A U_1 = \begin{pmatrix} A_1 & 0 \\ 0 & \lambda_1 I \end{pmatrix}$ is normal — conjugating by a unitary does not change normality. Since the matrix is block-diagonal, each block must be normal separately.)

Therefore we can just recurse: find a unitary U_2 so

$$U_2^+ (A_1 - \lambda_2 I) U_2 = \begin{pmatrix} \underset{\text{nonsingular}}{C_2} & 0 \\ 0 & 0 \end{pmatrix}$$

$$\Rightarrow U_2^+ A_1 U_2 = \begin{pmatrix} \underset{A_3!}{C_2 + \lambda_2 I} & 0 \\ 0 & \lambda_2 I \end{pmatrix} \quad \text{etc.}$$

Putting everything together, we find that for

$$U^+ = \cdots \begin{pmatrix} \begin{matrix} U_3^+ & 0 \\ 0 & I \end{matrix} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} U_2^+ & 0 \\ 0 & \underset{\text{to leave the } \lambda_1 I \text{ term unchanged}}{I} \end{pmatrix} U_1^+ ,$$

$$\underline{U^+ A U} = \begin{pmatrix} \lambda_k I & & & 0 \\ & \lambda_{k-1} I & & \\ & & \ddots \lambda_2 I & \\ 0 & & & \lambda_1 I \end{pmatrix} \quad \checkmark$$

$\square$

Important: The theorem ($A^+ A = A A^+ \Rightarrow$ unitarily diagonalizable) is very important. So is the proof technique: Find one eigenspace, split it off, and recurse with the remainder.

Example: How can we use the power method to find the **second**-largest magnitude eigenvalue and the corresponding eigenvector?

One approach, in Matlab:

```
% using the power method to find the second-largest-magnitude eigenvalue
eigenvector
n = 100;
A = randn(n,n);
A = A + A';          % symmetric matrix => normal matrix

% first find the principal eigenvector using the power method
x = randn(n,1);
for j = 1:10000
  x = A * x;
  x = x / norm(x);
  x';
end
A*x ./ x    % using component-wise division, check that we've found an e-vector

% this starts with a vector perpendicular to the principal eigenvector,
%  but numerical errors explode, causing it to be pushed parallel to the
%  principal eigenvector
y = randn(n,1);
y = y - (x'*y)*x;
for j = 1:10000
  y = A * y;
  y = y / norm(y);
  y';
end
A * y ./ y

% to get the power method to work, we need to project orthogonal to the
%  principal eigenvector after every step (or at least occasionally)
y = randn(n,1);
y = y - (x'*y)*x;
for j = 1:10000
  y = A * y;
  y = y - (x'*y)*x;
  y = y / norm(y);
  y';
end
A * y ./ y

% we can also find the k largest-magnitude eigenvalues simultaneously, using
the Gram-Schmidt procedure at every step of the power method
```

THEOREM: A has a complete, orthogonal set of eigenvectors

$$\Updownarrow$$

$$A^\dagger A = A A^\dagger. \qquad (\text{"A is normal"})$$

$$\Rightarrow A = \underset{\text{unitary w/ e-vector columns}}{\mathcal{U}} \; \underset{\text{diagonal w/ e-values}}{D} \; \mathcal{U}^\dagger$$

Examples:

- Every diagonal matrix is normal ($\mathcal{U} = I$ above)
- No upper- or lower-triangular matrix is normal, unless it is diagonal! (see the homework)

```
>> n = 4;
>> A = randn(n,n);
>> A = A + A';      % make the matrix symmetric (and hence normal)
>> [U, D] = eigs(A)
```

U =

$\vec{v}_1$	$\vec{v}_2$	$\vec{v}_3$	$\vec{v}_4$
0.6207	-0.5958	0.1753	-0.4785
0.6988	0.1282	-0.1866	0.6786
-0.1355	-0.3777	-0.9151	-0.0408
-0.3287	-0.6971	0.3116	0.5558

the e-vectors are orthonormal:

```
>> U' * U

ans =

    1.0000         0    0.0000    0.0000
```

D =

λ_1

```
  -0.1355   -0.3777   -0.9151   -0.0408
  -0.3287   -0.6971    0.3116    0.5558
```

D =

$$\begin{array}{cccc} \lambda_1 & \lambda_2 & \lambda_3 & \lambda_4 \\ -7.2760 & 0 & 0 & 0 \\ 0 & 3.7712 & 0 & 0 \\ 0 & 0 & 1.8551 & 0 \\ 0 & 0 & 0 & -0.3472 \end{array}$$

```
>> A * U - U * D

ans =

   1.0e-14 *

    0.3553   -0.1776    0.0056   -0.1832
   -0.0888   -0.0722    0.1554   -0.3608
   -0.1221   -0.0444    0.1110    0.0073
    0.0444         0   -0.0888   -0.0416
```

$A\vec{v}_j = \lambda_j \vec{v}_j$

```
>> U' * U

ans =

    1.0000         0    0.0000    0.0000
         0    1.0000    0.0000    0.0000
    0.0000    0.0000    1.0000   -0.0000
    0.0000    0.0000   -0.0000    1.0000
```

Proposition: A normal $\Rightarrow \|A\| = \max_{\text{eigenvalues } \lambda} |\lambda|$.

Proof:

Recall $\|A\| = \max_{x : \|x\|=1} \|Ax\|$. Since unitaries don't change lengths,

$$\begin{aligned} \|A\| &= \|UDU^{-1}\| \\ &= \|D\| \\ &= \max_i |D_{i,i}| \checkmark \end{aligned}$$

$\square$

This proposition is **false** for non-normal matrices, e.g., both eigenvalues of $\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$ are 0, but $\left\|\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}\right\| = 1 \neq 0$. ✓

Key point:

Normal matrix $\Rightarrow$ Different eigenspaces are orthogonal

Because the nullspace (e-value 0 e-space) satisfies

$$N(A) = N(A^T) = R(A)^\perp$$

and all other e-spaces are in $R(A)$