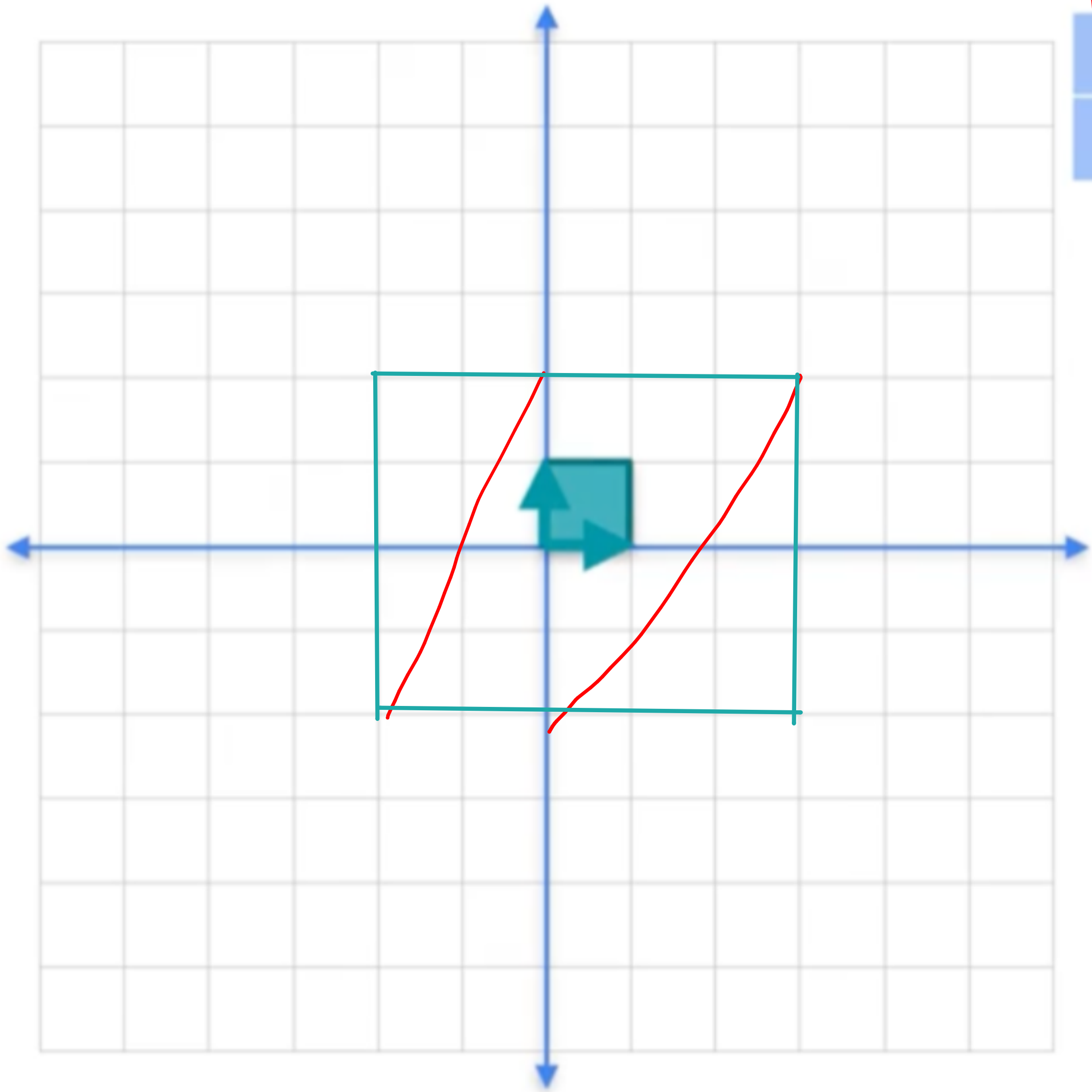
Day-93, Mar 3, 2025 (Falgun 19, 2081 B.S.).

1. PCA Motivating
2. Variance and Covariance
3. On the Number of Eigen Vectors
4. Dimensionality Reduction and Projection
5. Covariance Matrix
6. PCA overview and why it works, Mathematical formulation
7. Discrete Dynamical Systems.
8. Eigenvalues, Eigen Vectors.

Basis

$$\begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \Rightarrow \begin{bmatrix} 3 \\ 3 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$$



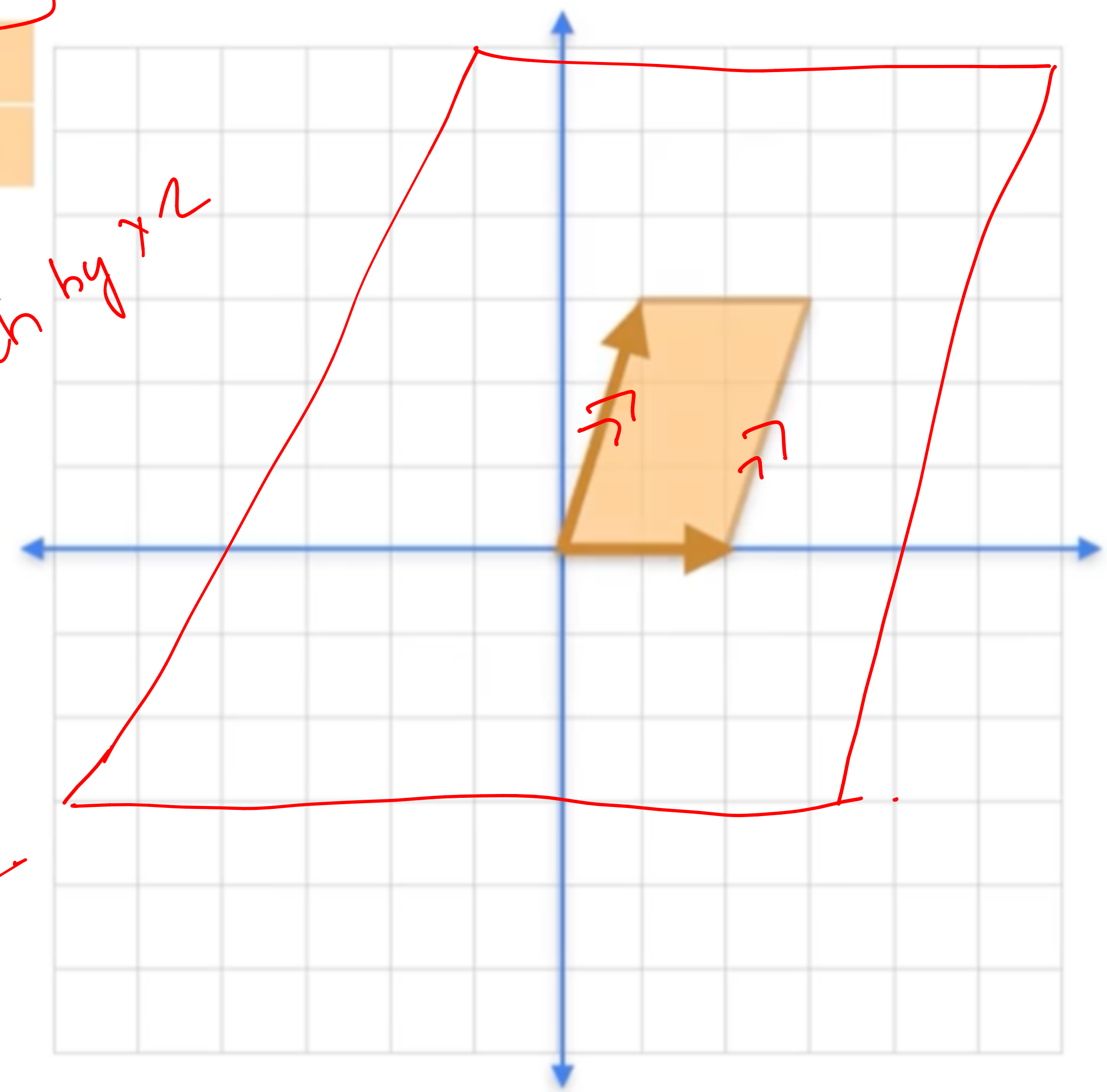
$\xrightarrow{\times 2}$ sketch by ×2

(1,0) → (2,0)

(0,1) → (1,3)

×3

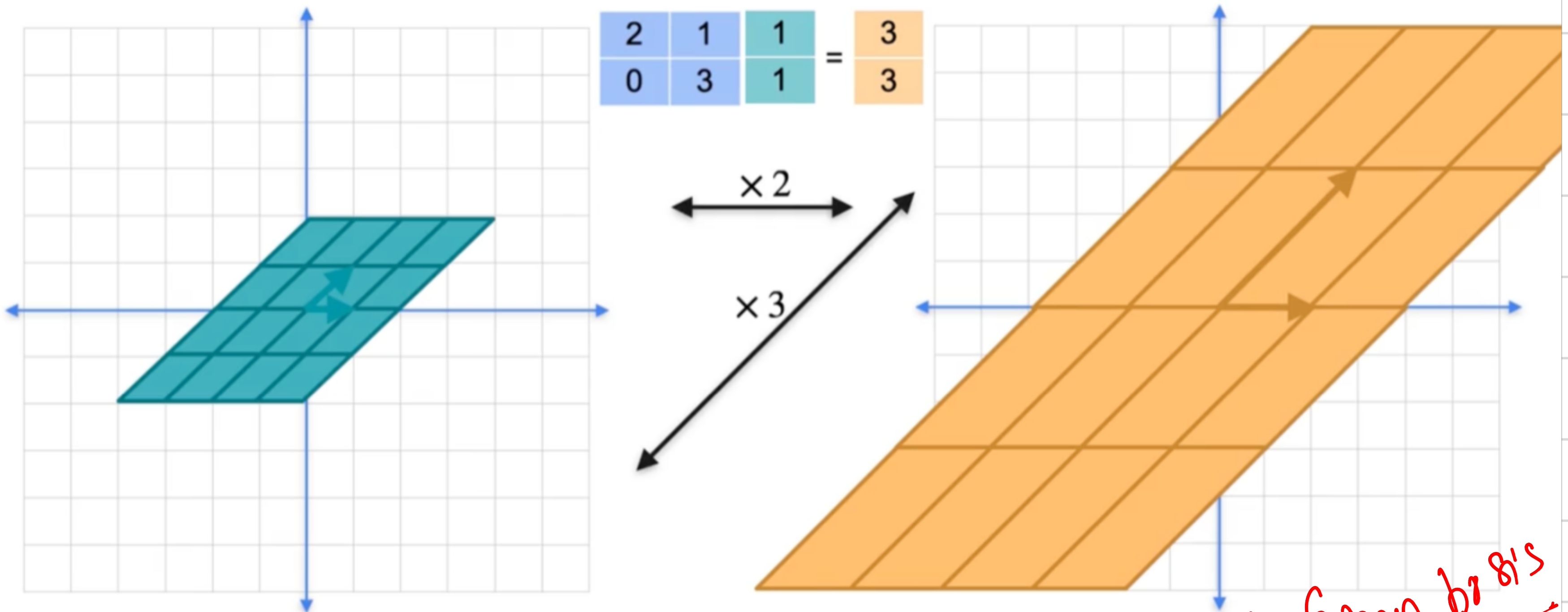
by ×3.



this parallelogram on the right

DeepLearning.AI

Eigenbasis



This is a very special basis.

DeepLearning.AI

Called Eigenbasis

Eigenbasis sends one parallelogram to another parallelogram with sides parallel to the original one

$$[A \lambda = \mu.]$$

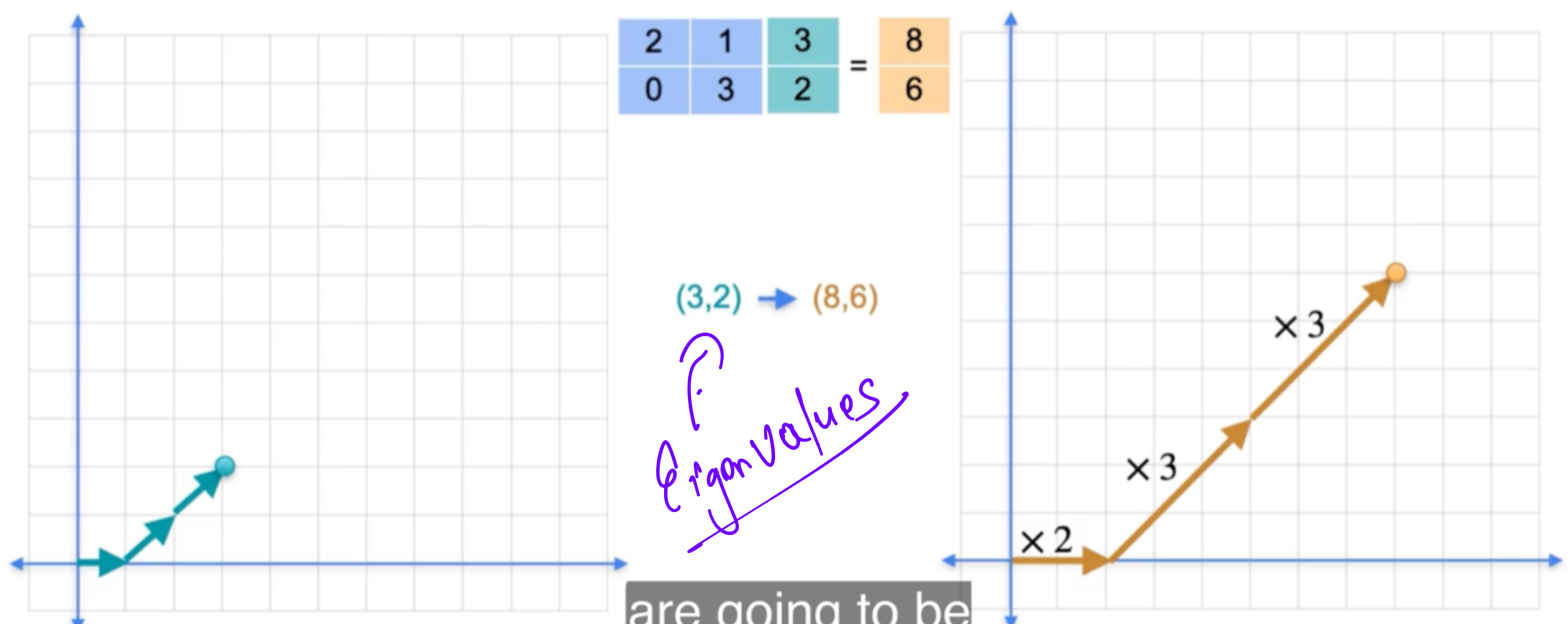
$$\begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 3 \\ 2 \end{bmatrix} = \begin{bmatrix} 8 \\ 6 \end{bmatrix}$$

$$[A \quad \lambda \quad = \quad x.]$$

find $\begin{bmatrix} 3 \\ 2 \end{bmatrix}$ using A and x = !!!

Eigenbasis

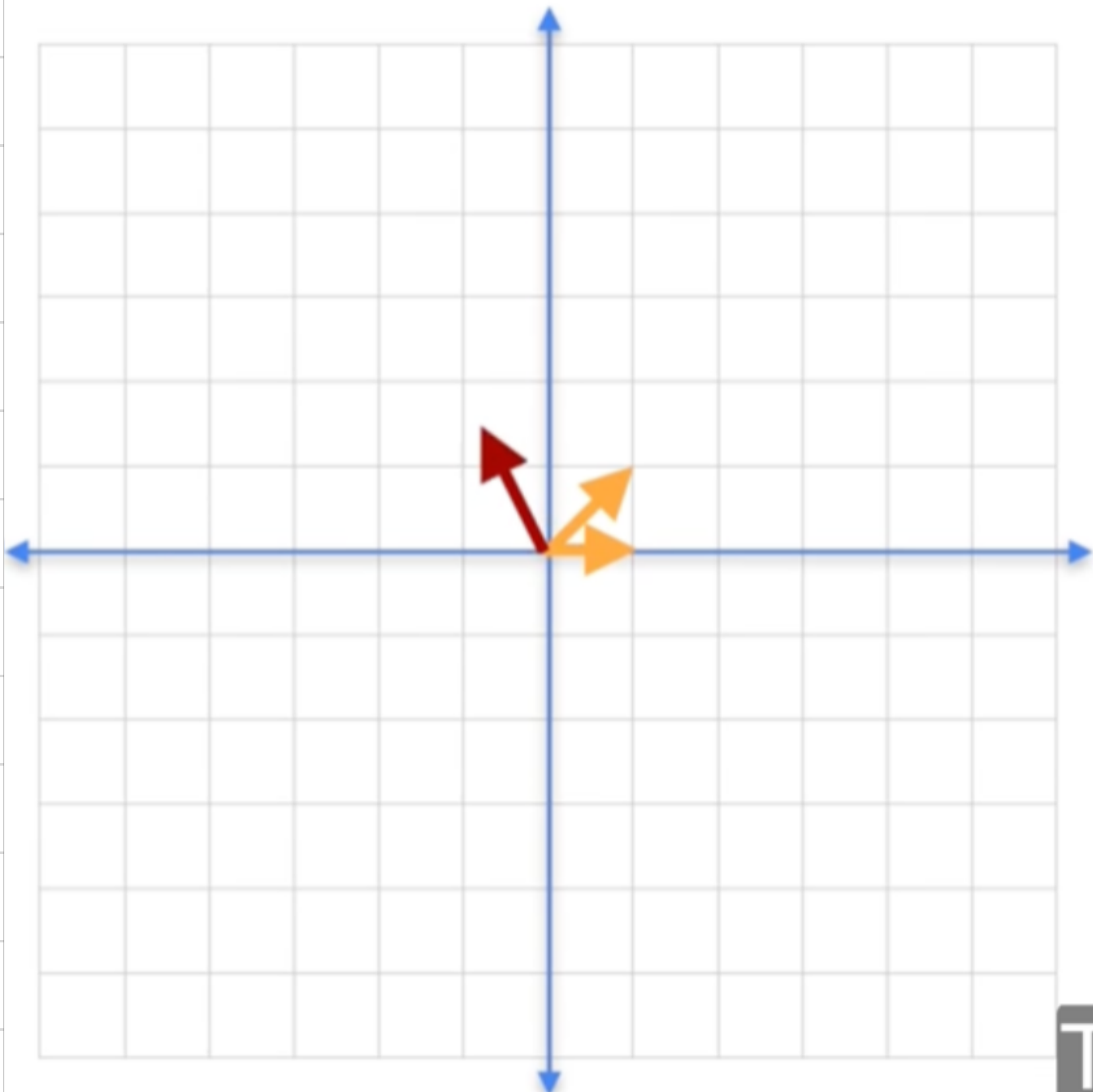


?

Eigenvalues

are going to be called eigenvalues.

Eigenvalues and eigenvectors

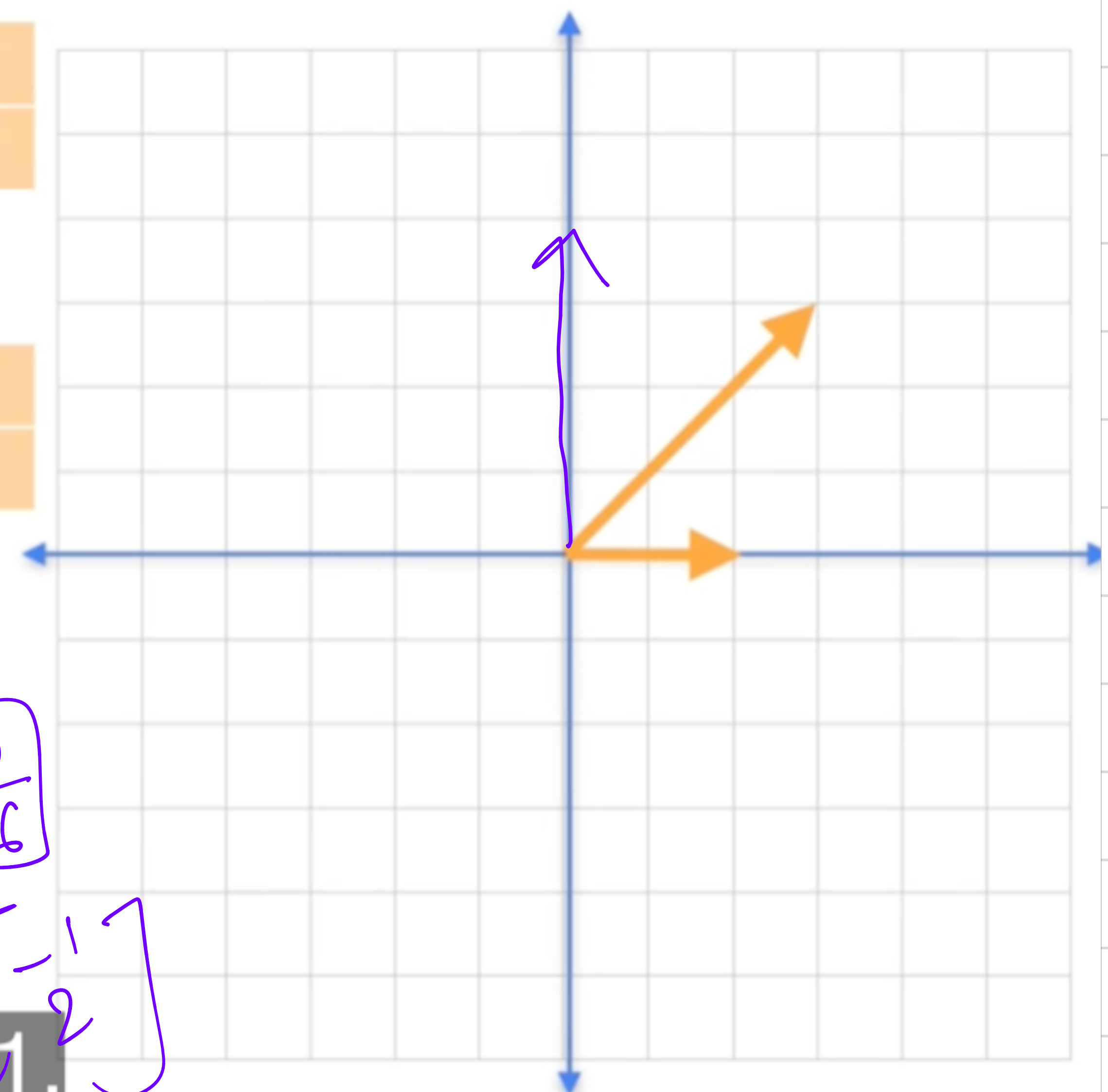


$$\begin{bmatrix}2 & 1\\0 & 3\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = 2\begin{bmatrix}1\\0\end{bmatrix}$$

$$\begin{bmatrix}2 & 1\\0 & 3\end{bmatrix}\begin{bmatrix}1\\1\end{bmatrix} = 3\begin{bmatrix}1\\1\end{bmatrix}$$

$$\begin{bmatrix}2 & 1\\0 & 3\end{bmatrix}\begin{bmatrix}-1\\2\end{bmatrix} = \boxed{\begin{bmatrix}0\\6\end{bmatrix}}$$

$$\neq \lambda \begin{bmatrix}-1\\2\end{bmatrix}$$



The final vector -1, 2 is not an eigenvector.

DeepLearning.AI

Eigenvalues and eigenvectors

Matrix Multiplication → More work.

Scalar Multiplication → less work.

$$\begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 2 \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

8 multiplication 2 multiplication.

$$\begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = 3 \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Scalar

$$A v_1 = \boxed{\lambda_1 v_1}$$

→ first pair

$$A v_2 = \boxed{\lambda_2 v_2}$$

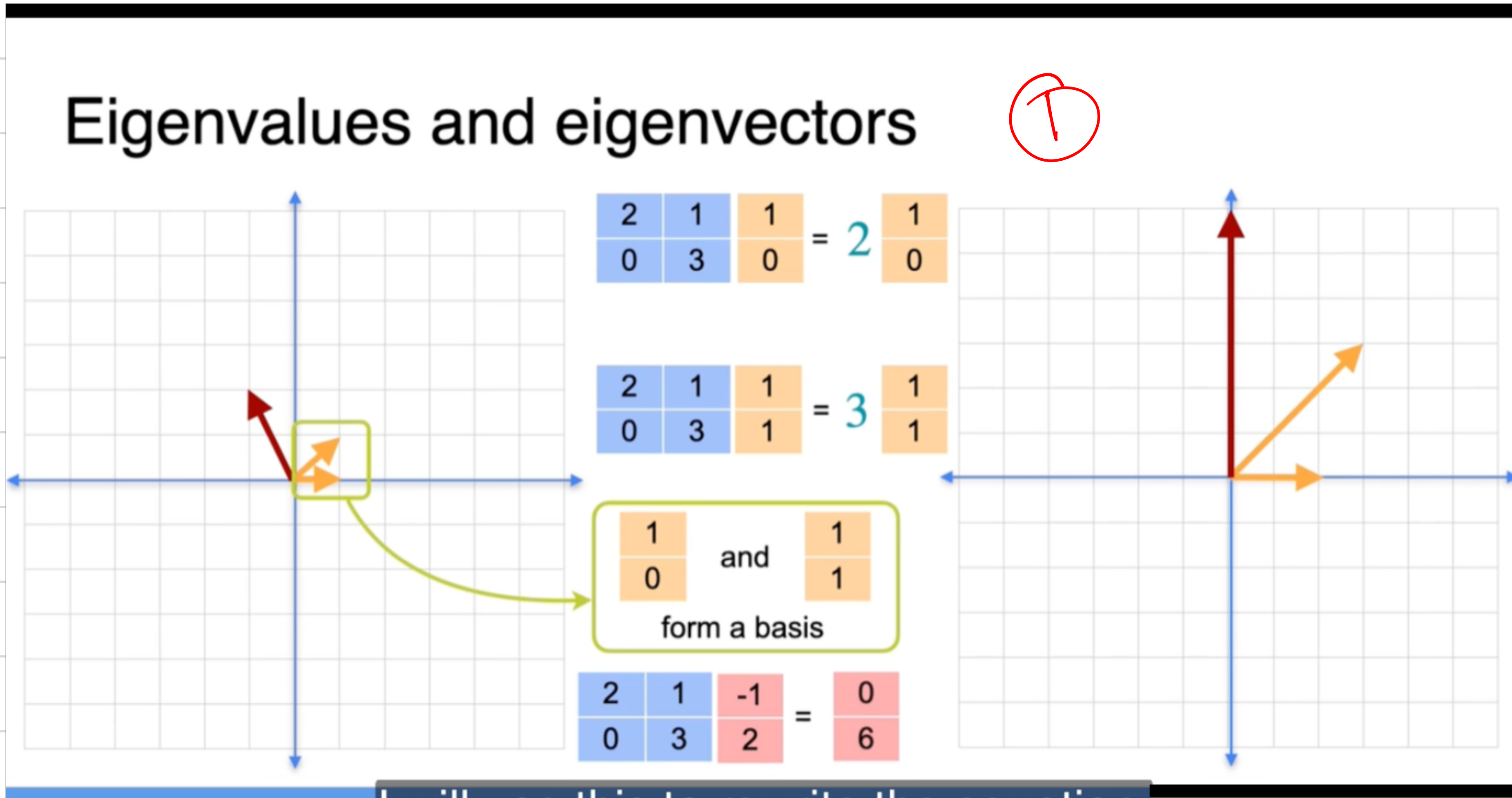
→ Second pair

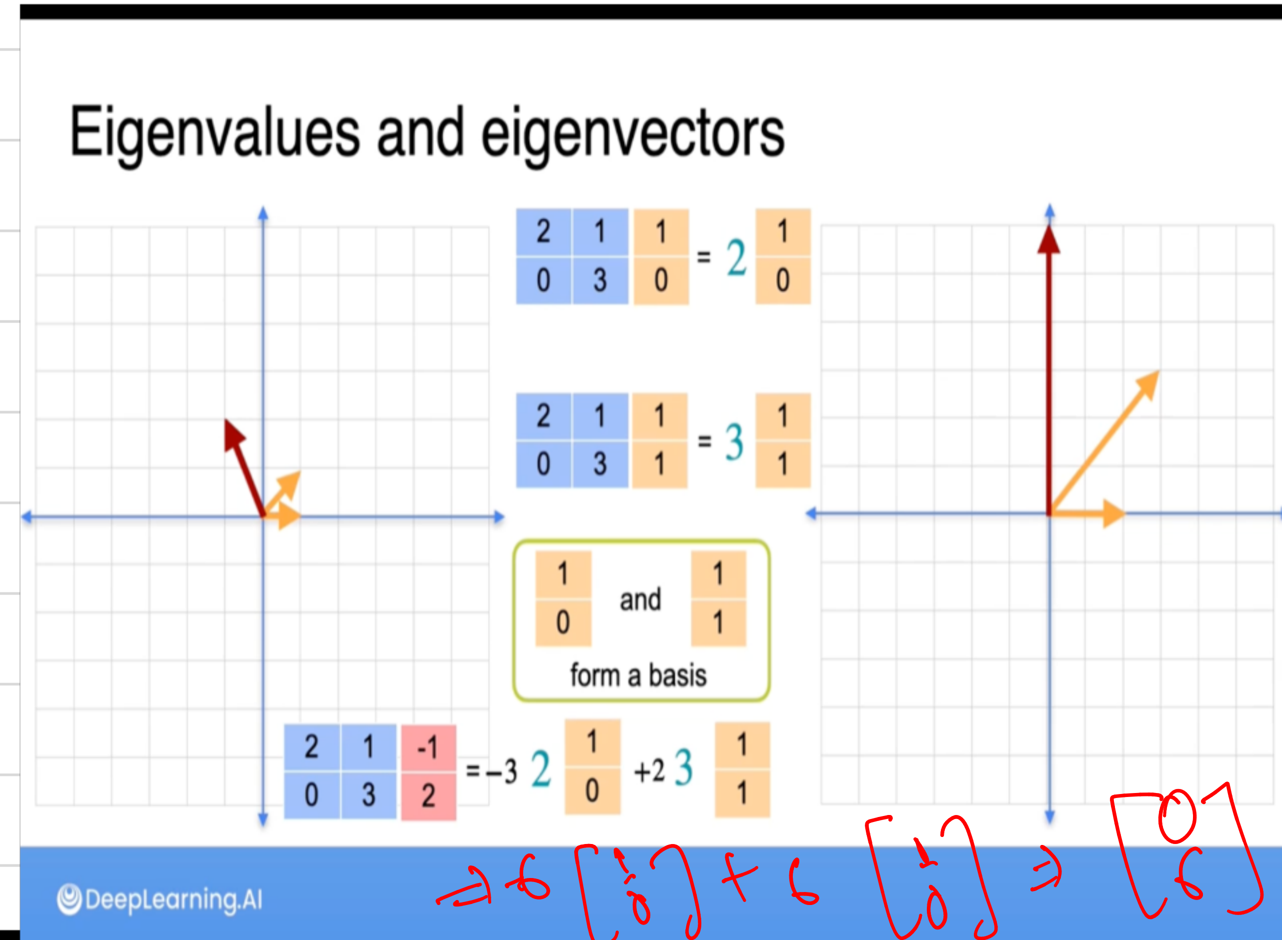
Eigen values

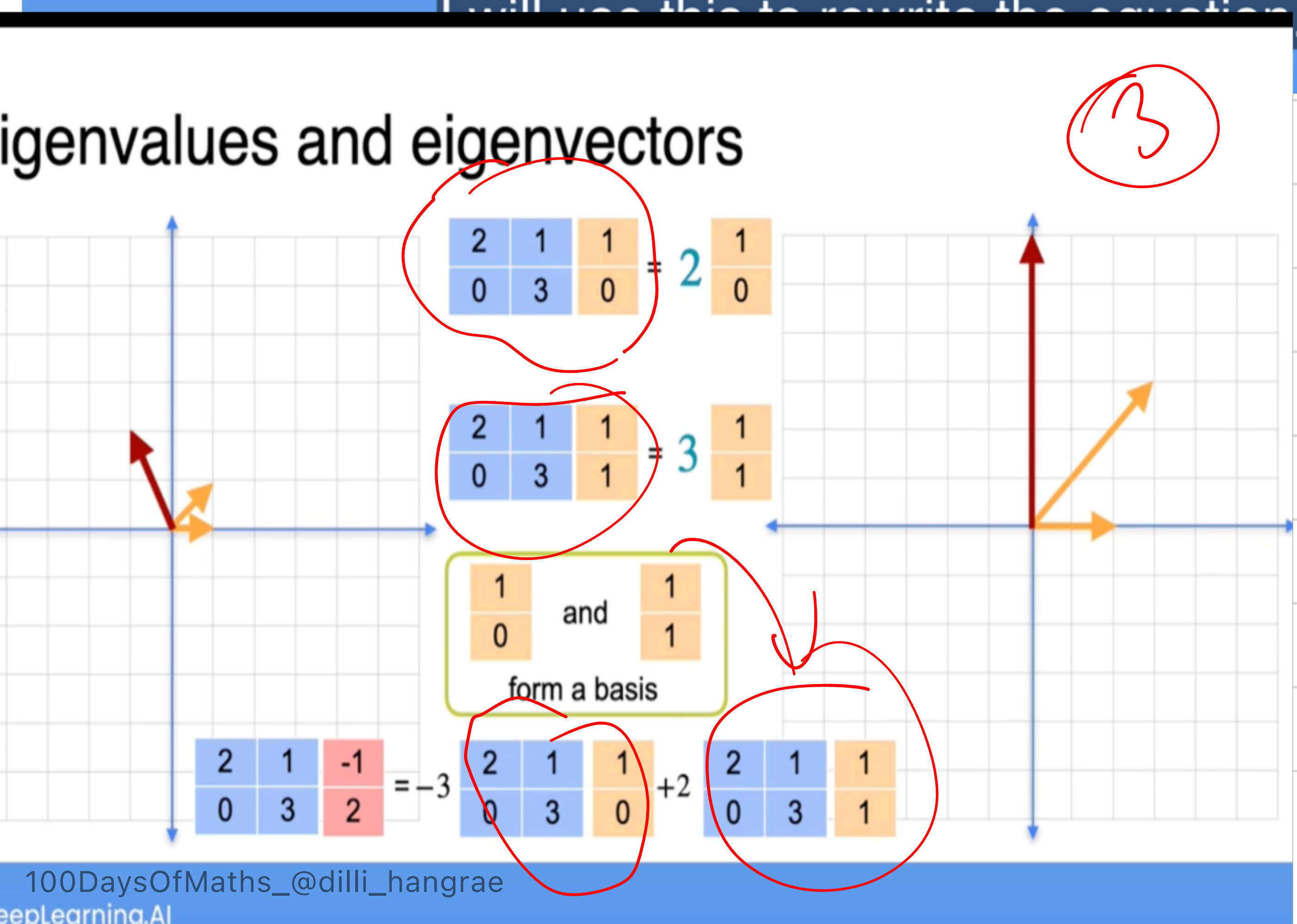
→ Eigen vector

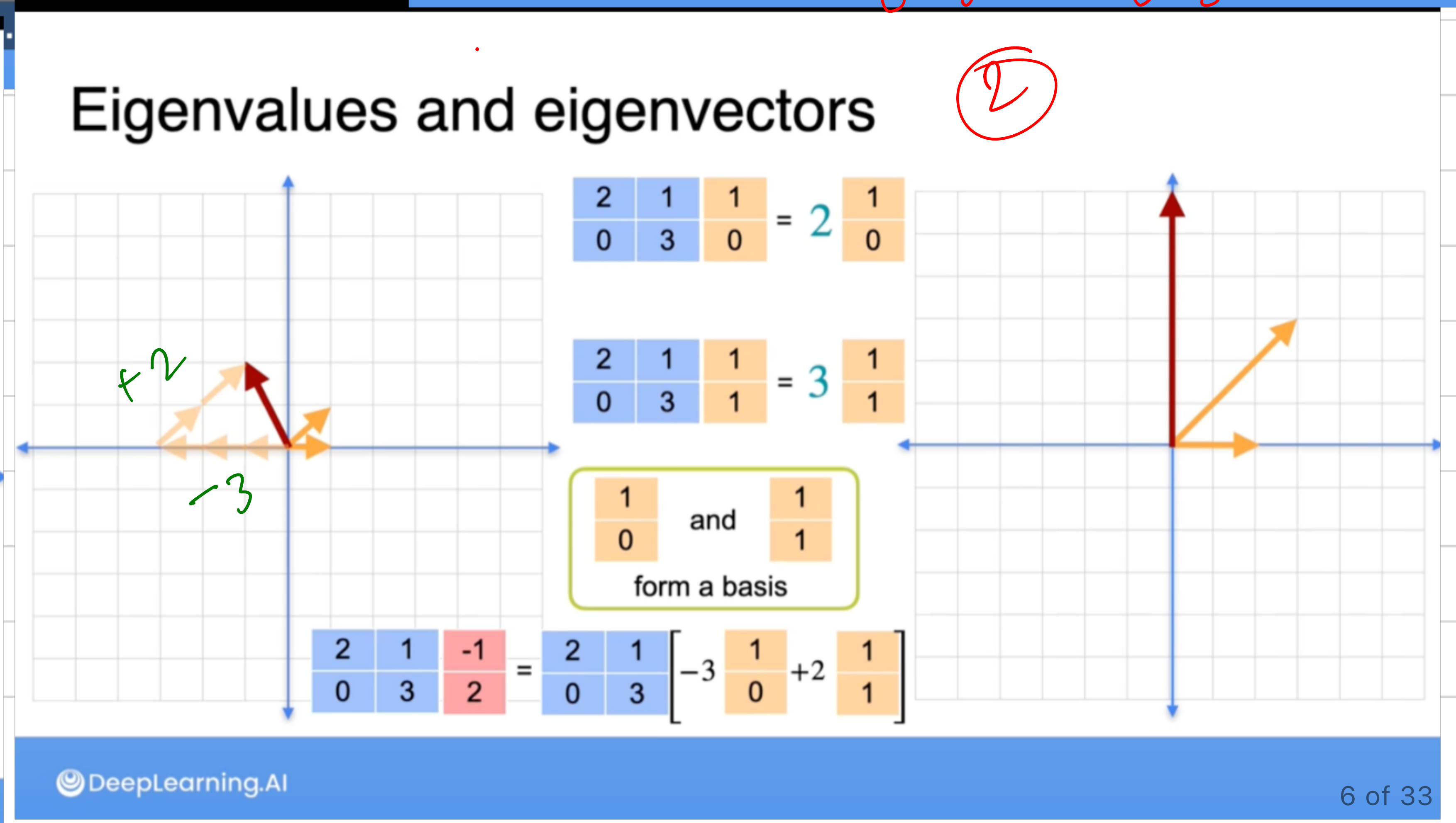
you already know v1 is the vector 1, 0 and lambda 1 is the scalar 2.

DeepLearning.AI


Eigenvalues and eigenvectors
①
2 1 / 0 3 · 1 / 0 = 2 · 1 / 0
2 1 / 0 3 · 1 / 1 = 3 · 1 / 1
1 / 0 and 1 / 1 form a basis
2 1 / 0 3 · -1 / 2 = 0 / 6


Eigenvalues and eigenvectors
2 1 / 0 3 · 1 / 0 = 2 · 1 / 0
2 1 / 0 3 · 1 / 1 = 3 · 1 / 1
1 / 0 and 1 / 1 form a basis
2 1 / 0 3 · -1 / 2 = −3 2 · 1 / 0 +2 3 · 1 / 1
DeepLearning.AI


igenvalues and eigenvectors
③
2 1 / 0 3 · 1 / 0 = 2 · 1 / 0
2 1 / 0 3 · 1 / 1 = 3 · 1 / 1
1 / 0 and 1 / 1 form a basis
2 1 / 0 3 · -1 / 2 = −3 · 2 1 / 0 3 · 1 / 0 +2 · 2 1 / 0 3 · 1 / 1
100DaysOfMaths_@dilli_hangrae
eepLearning.AI

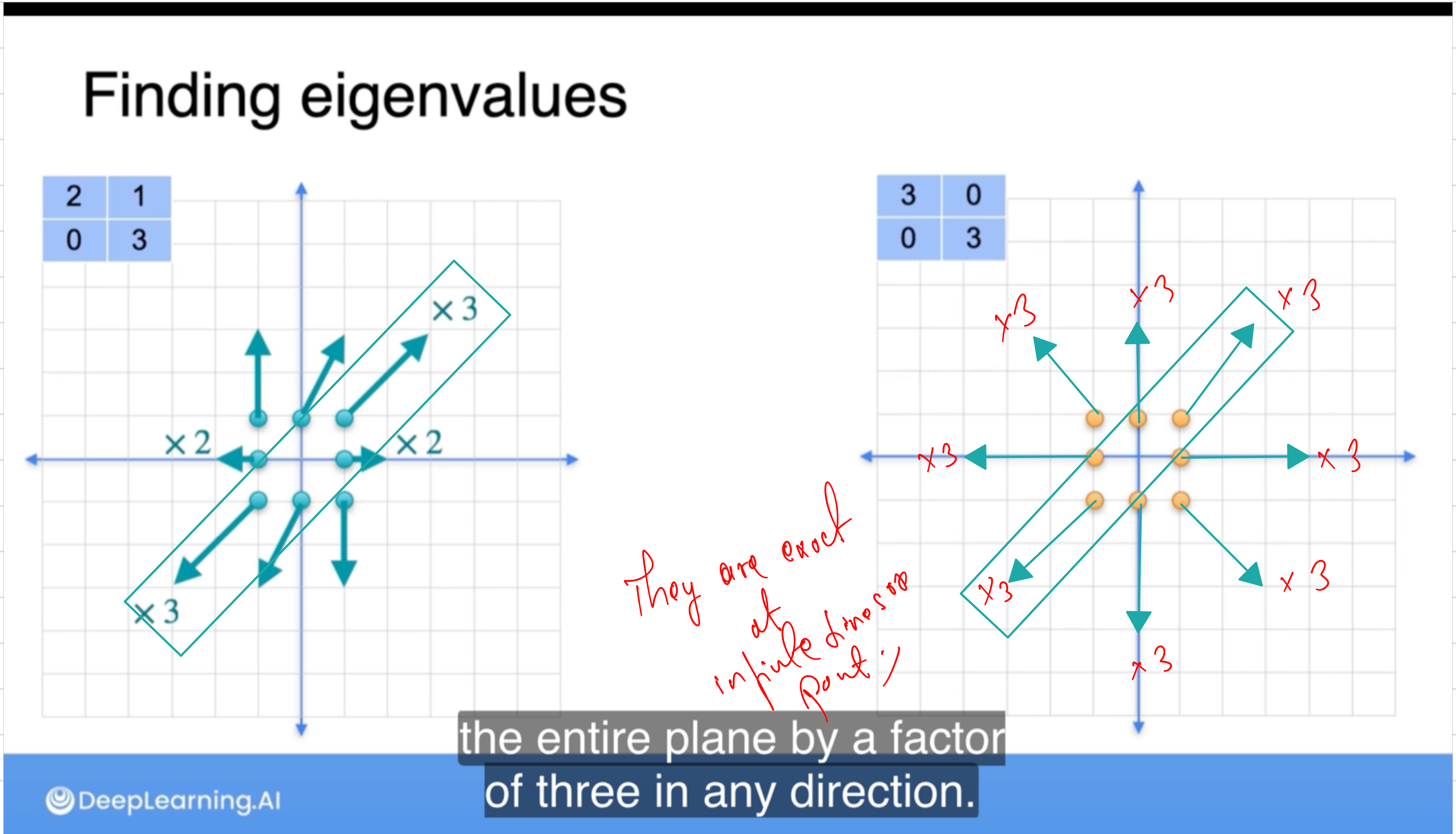

Eigenvalues and eigenvectors
②
+2
−3
2 1 / 0 3 · 1 / 0 = 2 · 1 / 0
2 1 / 0 3 · 1 / 1 = 3 · 1 / 1
1 / 0 and 1 / 1 form a basis
2 1 / 0 3 · -1 / 2 = 2 1 / 0 3 [−3 · 1 / 0 +2 · 1 / 1]
DeepLearning.AI
6 of 33

So, in the previous we used (-3, 2) but for other case we need to compute using

1) Calculate the inverse of the eigenbasis
2) Multiply it by your vector

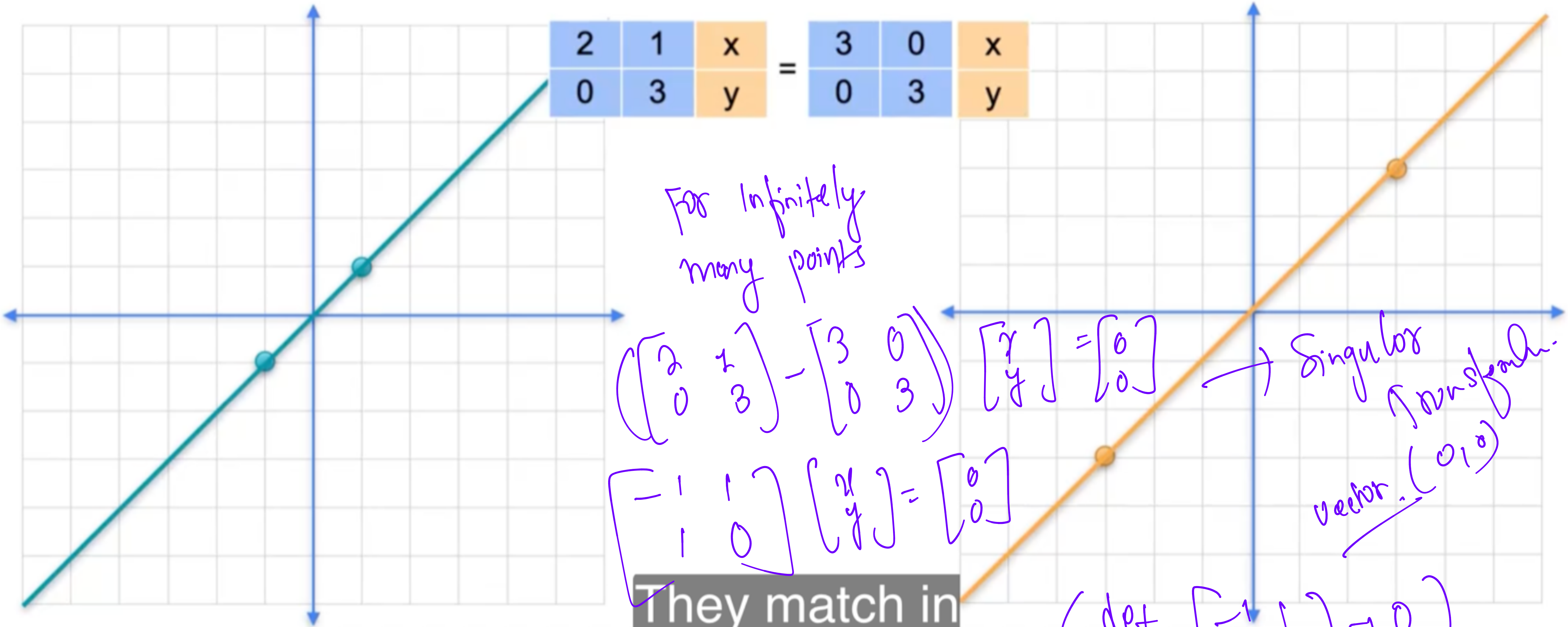
- $Av = \lambda v$ for each eigenvector / eigenvalue
  - EigenVectors: direction of sketch
  - EigenValues: How much stretch
  - Eigenbasis: the set of matrix's eigenvectors, can be arranged as a matrix with one eigenvector in each column.
- Save work and characterize a transformation.

Eigenvectors and Eigen Values:



Finding eigenvalues

$$\begin{bmatrix}2&1\\0&3\end{bmatrix}\begin{bmatrix}x\\y\end{bmatrix}=\begin{bmatrix}3&0\\0&3\end{bmatrix}\begin{bmatrix}x\\y\end{bmatrix}$$



For infinitely many points

$$\left(\begin{bmatrix}2&1\\0&3\end{bmatrix}-\begin{bmatrix}3&0\\0&3\end{bmatrix}\right)\begin{bmatrix}x\\y\end{bmatrix}=\begin{bmatrix}0\\0\end{bmatrix}$$

$$\begin{bmatrix}-1&1\\1&0\end{bmatrix}\begin{bmatrix}x\\y\end{bmatrix}=\begin{bmatrix}0\\0\end{bmatrix}$$

→ Singular transformation.

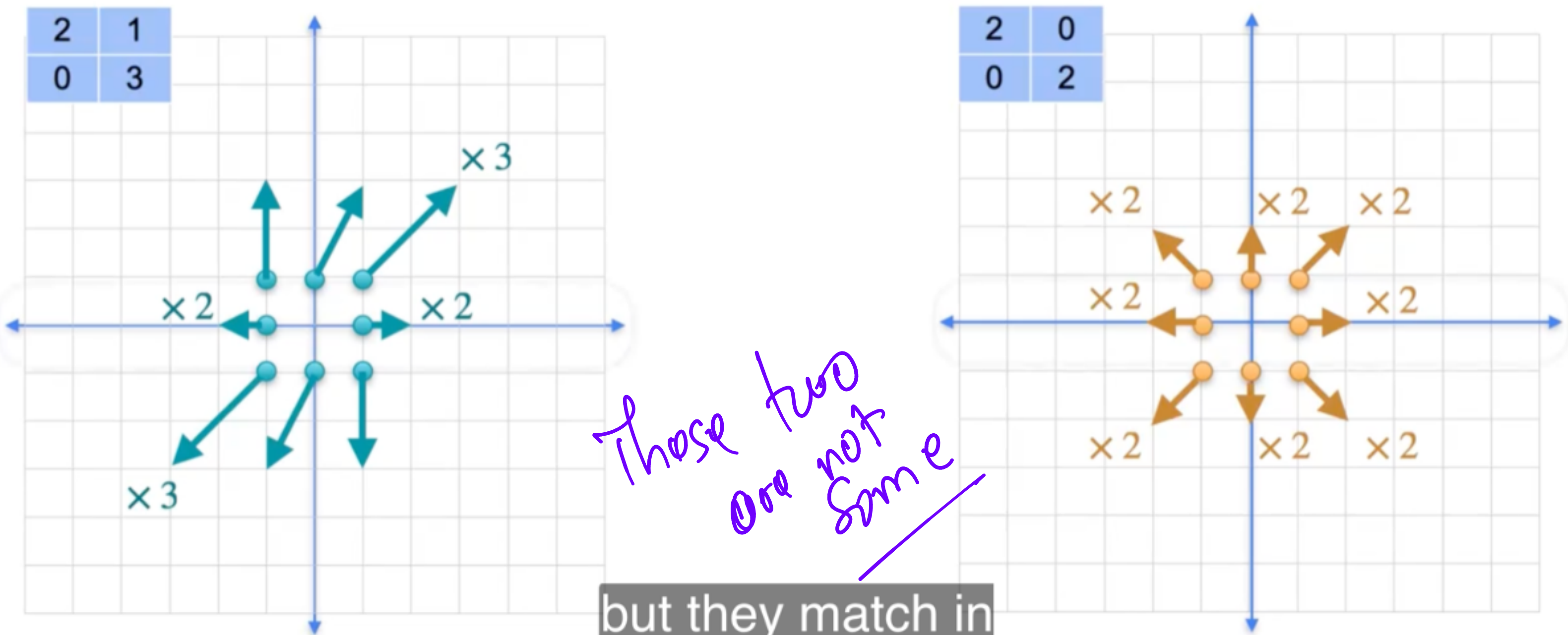
vector (0,0)

$$\left(\det\begin{bmatrix}-1&1\\0&0\end{bmatrix}=0\right)$$

They match in infinitely many points.

DeepLearning.AI

Finding eigenvalues



These two are not same

but they match in this entire line.

DeepLearning.AI

Must be $\det \begin{bmatrix} 2-\lambda & 1 \\ 0 & 3-\lambda \end{bmatrix} \Rightarrow 0.$

to Avoid infinitely solution.

Special about eigenvalue?

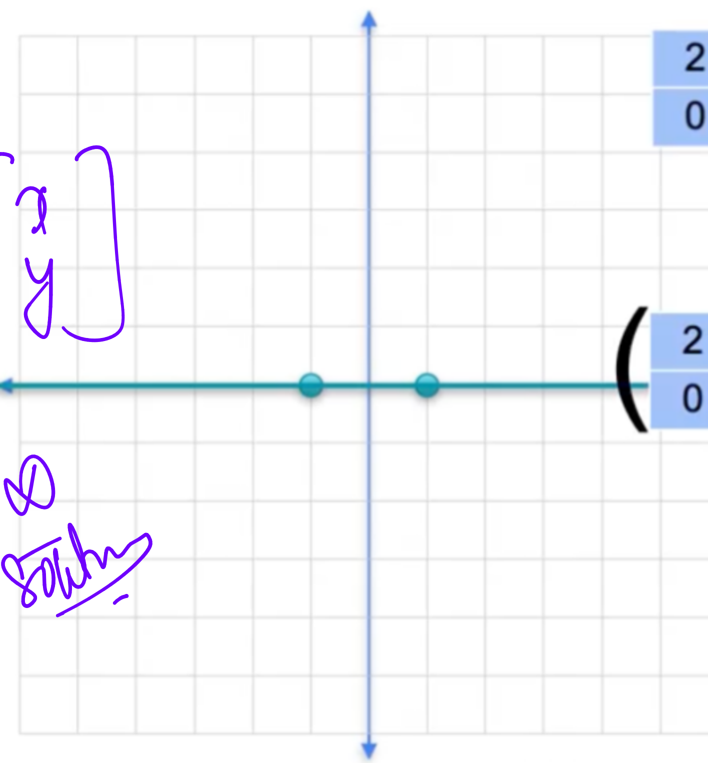
$$\lambda = \begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\begin{bmatrix} 2-\lambda & 1 \\ 0 & 3-\lambda \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

∞ solution

$$\det \begin{bmatrix} 2-\lambda & 1 \\ 0 & 3-\lambda \end{bmatrix} = 0.$$

Finding eigenvalues



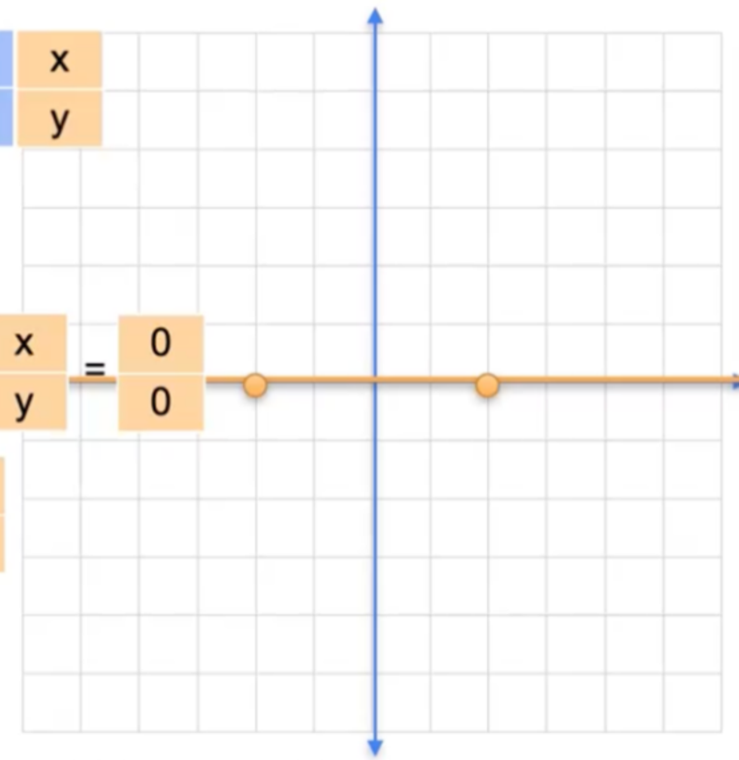
$$\begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

For infinitely many points

$$\left(\begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix} - \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \right) \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\det \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} = 0$$



is a singular matrix.

DeepLearning.AI

Finding eigenvalues

If λ is an eigenvalue: $\begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$ For infinitely many (x,y)

$\begin{bmatrix} 2-\lambda & 1 \\ 0 & 3-\lambda \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ Has infinitely many solutions

$\det \begin{bmatrix} 2-\lambda & 1 \\ 0 & 3-\lambda \end{bmatrix} = 0$

Characteristic polynomial $(2-\lambda)(3-\lambda) - 1 \cdot 0 = 0$ $\quad \lambda = 2$, $\lambda = 3$

In this case, they're going to be two and three.

DeepLearning.AI

Eigenvalues: $\lambda = 2$
$\lambda = 3$

$$\begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 2 \begin{bmatrix} x \\ y \end{bmatrix}$$

$2x + y = 2x$ $\quad x = 1$
$0x + 3y = 2y$ $\quad y = 0$

$$\begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \Rightarrow 3 \begin{bmatrix} x \\ y \end{bmatrix}$$

$2x + y \Rightarrow 3x$ $\quad x = 1$
$0x + 3y \Rightarrow 3y$ $\quad y = 1$

$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$

Solution

- Eigenvalues: 11, 1
- Eigenvectors: (2,1), (-1,2)

9	4
4	3

- The characteristic polynomial is $\det \begin{bmatrix} 9-\lambda & 4 \\ 4 & 3-\lambda \end{bmatrix} = (9-\lambda)(3-\lambda) - 4 \cdot 4 = 0$
- Which factors as $\lambda^2 - 12\lambda + 11 = (\lambda - 11)(\lambda - 1)$
- The solutions are $\lambda = 11$, $\lambda = 1$

Qu

Question

Find the eigenvalues and eigenvectors of this matrix:

Start by finding the **eigenvalues.** Remember you need to calculate the determinant of the matrix $A - \lambda I$, where A is your matrix.

$$\begin{bmatrix} 9 & 4 \\ 4 & 3 \end{bmatrix}$$

- ○ 10, 1
- ◉ 11, 1
- ○ 0, 11
- ○ 11, 2

Correct

The characteristic polynomial is:

$$det \begin{bmatrix} 9-\lambda & 4 \\ 4 & 3-\lambda \end{bmatrix} = (9-\lambda)(3-\lambda) - 4 \cdot 4 = 0.$$

When you simplify the expression, you get:

$$\lambda^2 - 12\lambda + 11$$

The solutions for the equation are: $\lambda_1 = 11; \lambda_2 = 1.$

Skip

Continue

eigenvectors of this matrix.

Qu

Question

Select the option containing eigenvectors for this matrix:

Now use your results to find the **eigenvectors**. You'll need to use the eigenvalues you calculated in the last question. In case you didn't get them, they are 11 and 1. Then, for each eigenvector, solve the equation $Av = \lambda v$, where λ is your eigenvalue, A is your matrix, and v is the eigenvector you're solving for.

$$\begin{bmatrix} 9 & 4 \\ 4 & 3 \end{bmatrix}$$

- (2,2), (-1,0)
- (2,0), (-1,2)
- (2,1), (-1,2)
- (2,1), (1,2)

Correct

Correct! By using the eigenvalues you found $\lambda_1 = 11, \lambda_2 = 1$, you solve the system of equations to find the eigenvectors.

$$\begin{bmatrix} 9 & 4 \\ 4 & 3 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = 11 \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

Therefore, solving the linear system, you get the relation

$$x = 2y$$

So, any (x, y) satisfying such relation is an eigenvector. In this case, $x = 2, y = 1$ satisfies. So, the vector $\begin{bmatrix} 2 \\ 1 \end{bmatrix}$ is an eigenvector for the matrix. For the next eigenvalue, we have to solve the following set of equations:

$$\begin{bmatrix} 9 & 4 \\ 4 & 3 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = 1 \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

Leading to the following relationship between x and y:

$$2x = -y$$

And $x = -1, y = 2$ satisfies such relation, so the vector $\begin{bmatrix} -1 \\ 2 \end{bmatrix}$ is also an eigenvector for the matrix.

Skip

Continue

Deep

Finding eigenvalues

$$A = \begin{bmatrix} 2 & 1 & -1 \\ 1 & 0 & -3 \\ -1 & -3 & 0 \end{bmatrix} \qquad \lambda I = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}$$

Characteristic polynomial: $\det(A - \lambda I) = 0$

$$\det \begin{bmatrix} 2-\lambda & 1 & -1 \\ 1 & -\lambda & -3 \\ -1 & -3 & -\lambda \end{bmatrix} = 0$$

$$(2-\lambda)^2 + 3 + 3 - 9(2-\lambda) + \lambda + \lambda$$

$$\Rightarrow -\lambda^3 + 2\lambda^2 + 11\lambda - 12 = 0$$

$$\Rightarrow -(\lambda+3)(\lambda-1)(\lambda-4) = 0$$

[Eigen values $\lambda = -3, 1, 4$]

Eigen vector = ?

In this case, Lambda
minus I will be

DeepLearning.AI

Finding eigenvalues

$A =$

2	1	-1
1	0	-3
-1	-3	0

Eigenvalues: $-3, 1, \boxed{4}$

$$Av = \lambda v$$

$$\begin{bmatrix} 2 & 1 & -1 \\ 1 & 0 & -3 \\ -1 & -3 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \Rightarrow 4 \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

$$\begin{matrix} 2x_1 + x_2 - x_3 \\ x_1 - 3x_3 \\ -x_1 - 3x_2 \end{matrix} = \begin{bmatrix} 4x_1 \\ 4x_2 \\ 4x_3 \end{bmatrix}$$

for this case

$$\begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix}$$

$$2x_1 + x_2 - x_3 = 4x_1$$
$$x_1 - 3x_3 = 4x_2$$
$$-x_1 - 3x_2 = 4x_3$$

R_1: $-2x_1 + x_2 - x_3 = 0$

R_2: $x_1 - 4x_2 - 3x_3 = 0$

R_3: $-x_1 - 3x_2 - 4x_3 = 0$

$R_2 + R_3$

$-7x_2 - 7x_3 = 0$

$x_2 = -x_3$

$3R_1 + R_3$

$-7x_1 - 7x_3 = 0$

$x_1 = -x_3$

$$\begin{bmatrix} x_1 = k \\ x_2 = k \\ x_3 = -k \end{bmatrix}$$

Infinite solutions.

Let's begin with our last eigenvalue, four.

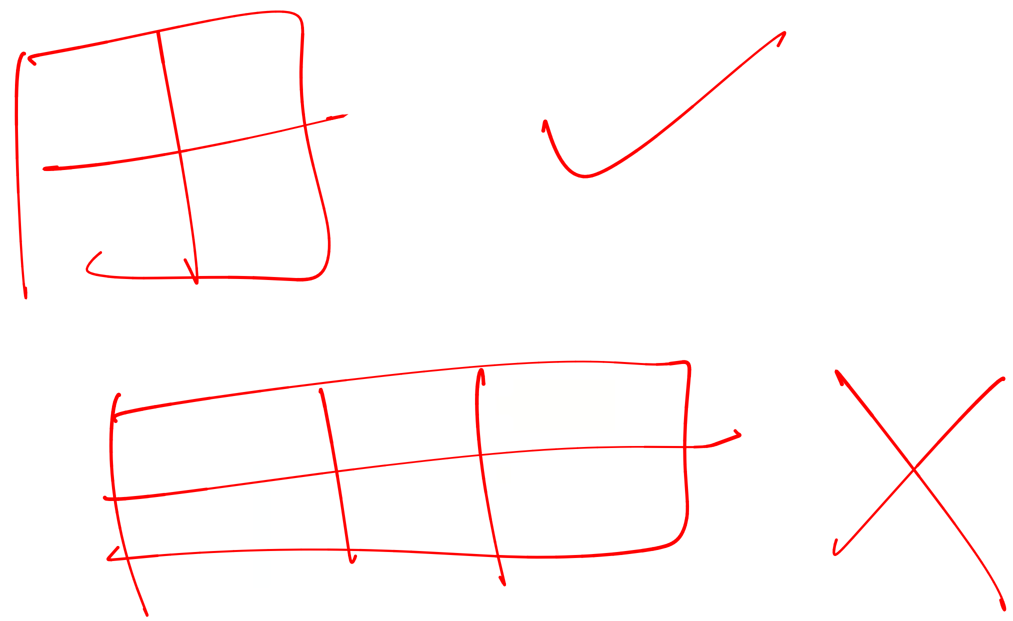
DeepLearning.AI

Note on dimensions

Eigenvalues $\longrightarrow$ Determinant $\longrightarrow$ Square Matrix.

$\downarrow$

Only for Square Matrix.

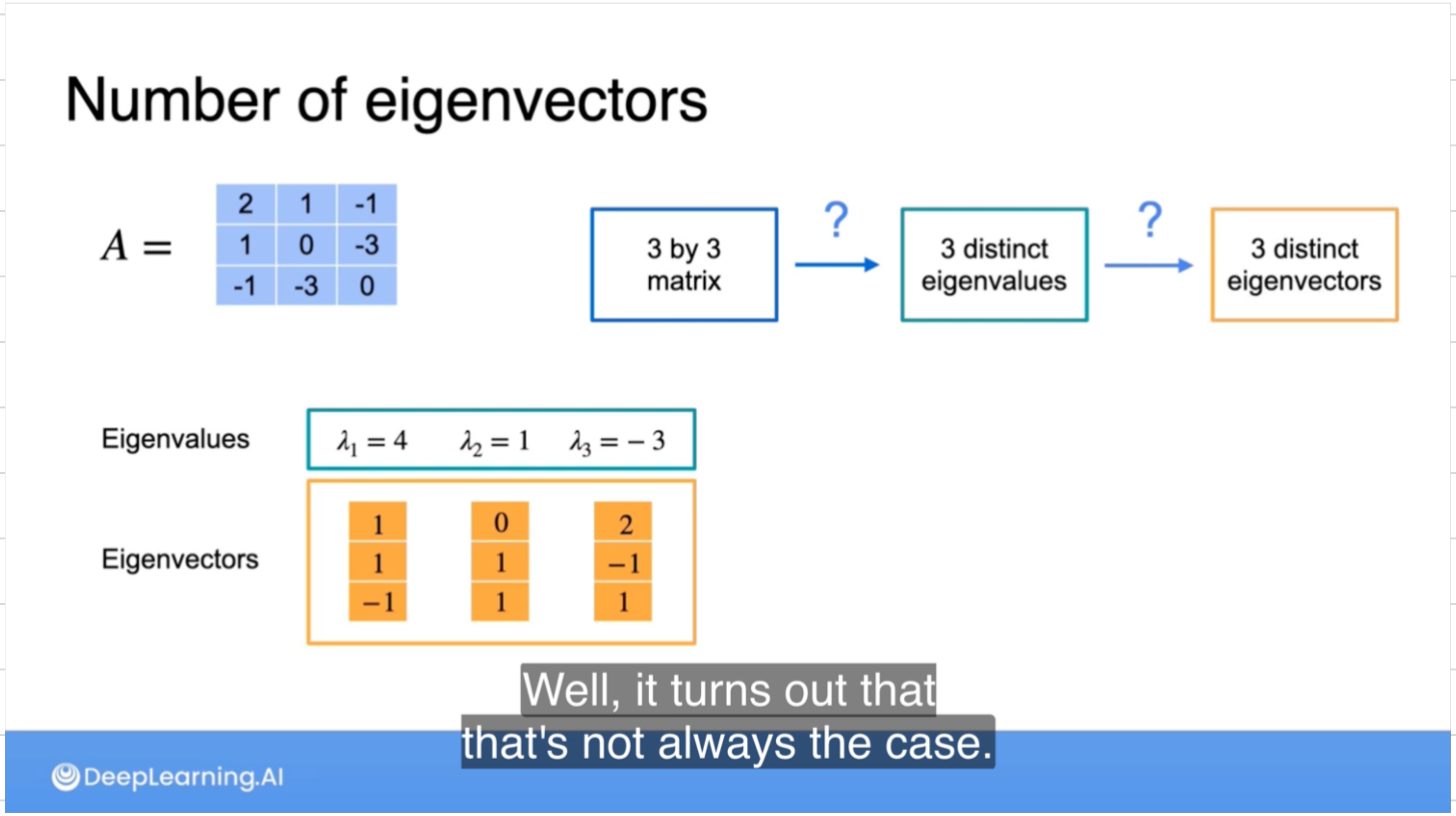


If you notice the two
examples you worked on

DeepLearning.AI

On the Number of Eigen Vectors

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



Characteristics polynomial =

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

$$\begin{bmatrix} 2-\lambda & 0 & 0 \\ 1 & 4-\lambda & 0\ 5 \\ 0 & 0 & 2-\lambda \end{bmatrix}$$

$$(2-\lambda)^2(4-\lambda) + 0 + 0 - 0 - 0 = 0$$

Eigen Values: $4, 2, 2$.

Among 4, 2, 2 take 4.

Repeated eigenvalues - Example 1

$$A = \begin{bmatrix} 2 & 0 & 0 \\ -1 & 4 & -0.5 \\ 0 & 0 & 2 \end{bmatrix} \quad \text{Eigenvalue: } 4$$

$$Av = 4v$$

$$\underbrace{\begin{bmatrix} 2 & 0 & 0 \\ -1 & 4 & -0.5 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{\begin{bmatrix} 2x_1 \\ -x_1 + 4x_2 - 0.5x_3 \\ 2x_3 \end{bmatrix}} = 4 \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4x_1 \\ 4x_2 \\ 4x_3 \end{bmatrix}$$

$2x_1 = 4x_1$

$-x_1 + 4x_2 - 0.5x_3 = 4x_2$

$2x_3 = 4x_3$

$-2x_1 = 0$

$-x_1 - 0.5x_3 = 0$

$-2x_3 = 0$

$x_1 = 0$ $\quad$ $x_2 = k$ (any number).

$x_3 = 0$

Eigenvector.

$$\begin{array}{l} x_1 \rightarrow \\ x_2 \rightarrow \\ x_3 \rightarrow \end{array} \boxed{\begin{matrix} 0 \\ 1 \\ 0 \end{matrix}}$$

as the vector with entries

2x_1, -x_1+4x_2-0.5x_3,

DeepLearning.AI

Repeated eigenvalues - Example 1

$$A = \begin{bmatrix} 2 & 0 & 0 \\ -1 & 4 & -0.5 \\ 0 & 0 & 2 \end{bmatrix}$$

Eigenvalue: 2

$$\begin{aligned} 2x_1 &= 2x_1 \\ -x_1 + 4x_2 - 0.5x_3 &= 2x_2 \\ 2x_3 &= 2x_3 \end{aligned}$$

$$Av = 2v$$

$$\underbrace{\begin{bmatrix} 2 & 0 & 0 \\ -1 & 4 & -0.5 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{\begin{bmatrix} 2x_1 \\ -x_1 + 4x_2 - 0.5x_3 \\ 2x_3 \end{bmatrix}} = 2 \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2x_1 \\ 2x_2 \\ 2x_3 \end{bmatrix}$$

$$\begin{aligned} 0 &= 0 \\ -x_1 + 2x_2 - 0.5x_3 &= 0 \\ 0 &= 0 \end{aligned}$$

$$x_1 = 2x_2 - 0.5x_3$$

Point in different directions

2	$x_1 = 2$
1	$x_2 = 1$
0	$x_3 = 0$

1	$= x_1 = 1$
1	$x_2 = 1$
2	$x_3 = 2$

Different Eigenvectors

This equation has infinitely many solutions

DeepLearning.AI

Repeated eigenvalues - Example 1

$$A = \begin{bmatrix} 2 & 0 & 0 \\ -1 & 4 & -0.5 \\ 0 & 0 & 2 \end{bmatrix}$$

$$\begin{matrix} 2 & 0 & 0 \\ -1 & 4 & -0.5 \\ 4 & 0 & 2 \end{matrix}$$

changed.

Eigenvalues	$\lambda_1 = 4$	$\lambda_2 = 2$	$\lambda_3 = 2$
Eigenvectors	$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 2 \\ 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}$

Let's look into one more

Repeated eigenvalues - Example 2

$$A = \begin{bmatrix} 2 & 0 & 0 \\ -1 & 4 & -0.5 \\ 4 & 0 & 2 \end{bmatrix}$$

Characteristic polynomial = $\det(A - \lambda I)$ = det $\begin{bmatrix} 2-\lambda & 0 & 0 \\ 1 & 4-\lambda & 0.5 \\ -4 & 0 & 2-\lambda \end{bmatrix}$

$$(2-\lambda)^2(4-\lambda) + 0 + 0 - 0 - 0 - 0$$

Eigenvalues: 4, $\boxed{2, 2}$ Repeated eigenvalue

Let's repeat the process from before to find

Repeated eigenvalues - Example 2

$$A = \begin{bmatrix} 2 & 0 & 0 \\ -1 & 4 & -0.5 \\ 4 & 0 & 2 \end{bmatrix}$$

Eigenvalue: 4

$$Av = 4v$$

$$\underbrace{\begin{bmatrix} 2 & 0 & 0 \\ -1 & 4 & -0.5 \\ 4 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{\begin{bmatrix} 2x_1 \\ -x_1 + 4x_2 - 0.5x_3 \\ 4x_1 + 2x_3 \end{bmatrix}} = 4 \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4x_1 \\ 4x_2 \\ 4x_3 \end{bmatrix}$$

changes

$$\begin{aligned} 2x_1 &= 4x_1 \\ -x_1 + 4x_2 - 0.5x_3 &= 4x_2 \\ 4x_1 + 2x_3 &= 4x_3 \end{aligned}$$

$$\begin{aligned} -2x_1 &= 0 \\ -x_1 - 0.5x_3 &= 0 \\ 4x_1 - 2x_3 &= 0 \end{aligned}$$

$x_1 = 0$ $\quad x_3 = 0$ $\quad x_2 =$ any number

$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$ $\quad x_1 = 0$, $x_2 = 1$, $x_3 = 0$ $\quad$ Same as before!

initial equation Matrix.

It's the same one for the previous matrix.

Same for Eigenvalue: 2.

DeepLearning.AI

① Eigen Values Repeats you get fewer Directions

Repeated eigenvalues - Example 2

$$A = \begin{bmatrix} 2 & 0 & 0 \\ -1 & 4 & -0.5 \\ 4 & 0 & 2 \end{bmatrix}$$

Eigenvalue: 2

$$\begin{aligned} 2x_1 &= 2x_1 \\ -x_1 + 4x_2 - 0.5x_3 &= 2x_2 \\ 4x_1 + 2x_3 &= 2x_3 \end{aligned}$$

$$Av = 2v$$

$$\underbrace{\begin{bmatrix} 2 & 0 & 0 \\ -1 & 4 & -0.5 \\ 4 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{\begin{bmatrix} 2x_1 \\ -x_1 + 4x_2 - 0.5x_3 \\ 4x_1 + 2x_3 \end{bmatrix}} = 2 \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2x_1 \\ 2x_2 \\ 2x_3 \end{bmatrix}$$

$$\begin{aligned} 0 &= 0 \\ -x_1 + 2x_2 - 0.5x_3 &= 0 \\ 4x_1 &= 0 \end{aligned}$$

$$\begin{bmatrix} 0 \\ k \\ 4k \end{bmatrix}$$

$x_1 = 0$ $\quad$ $x_3 = 4x_2$

$\begin{bmatrix} 0 \\ 1 \\ 4 \end{bmatrix}$ $x_1 = 0$, $x_2 = 1$, $x_3 = 4$

$\begin{bmatrix} 0 \\ 0.5 \\ 2 \end{bmatrix}$ $x_1 = 0$, $x_2 = 0.5$, $x_3 = 2$

On the same line

Same eigenvector

2 and 3 that spans a different line.

DeepLearning.AI

② Number of Repetition (eigen value) doesn't mean you get more directions

③ Without enough directions, matrix cannot be broken into Simple parts.

④ If matrix has different eigenvalues, it has special directions.

Repeated eigenvalues - Example 2

$$A = \begin{bmatrix} 2 & 0 & 0 \\ -1 & 4 & -0.5 \\ 4 & 0 & 2 \end{bmatrix}$$

Eigenvalues	$\lambda_1 = 4$	$\lambda_2 = 2$	$\lambda_3 = 2$
Eigenvectors	$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \\ 4 \end{bmatrix}$	Can't create an eigenbasis from this matrix

the three dimensional space because

DeepLearning.AI

Summary

	$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$		$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$	
Eigenvalues	λ_1, λ_2		$\lambda_1, \lambda_2, \lambda_3$	
	If $\lambda_1 \neq \lambda_2$	2 eigenvectors (2 different directions)	If $\lambda_1 \neq \lambda_2 \neq \lambda_3$	3 eigenvectors (3 different directions)
	If $\lambda_1 = \lambda_2$	1 eigenvector (1 direction) 2 eigenvectors (2 different directions)	If $\lambda_1 = \lambda_2 \neq \lambda_3$	2 eigenvectors (2 different directions) 3 eigenvectors (3 different directions)
			If $\lambda_1 = \lambda_2 = \lambda_3$	1 eigenvector (1 direction) 2 eigenvectors (2 different directions) 3 eigenvectors (3 different directions)

In this week's programming assignments,

DeepLearning.AI

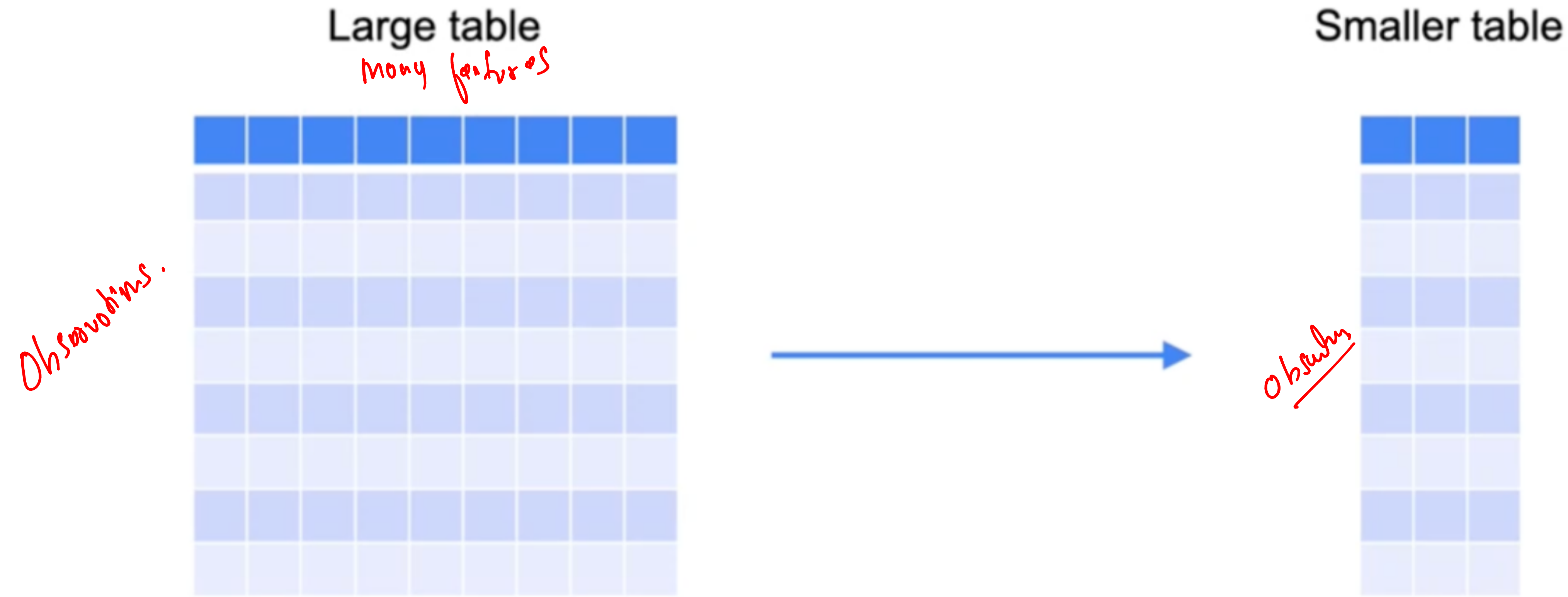
So, Eigenvalues is responsible to detect the changes of magnitude of eigenvectors.

Dimensionality Reduction and Projection:

(PCA - delete the columns preserving valuable information)

Dimensionality Reduction

- Reduce dimensions (# of columns) of dataset
- Preserve as much information as possible

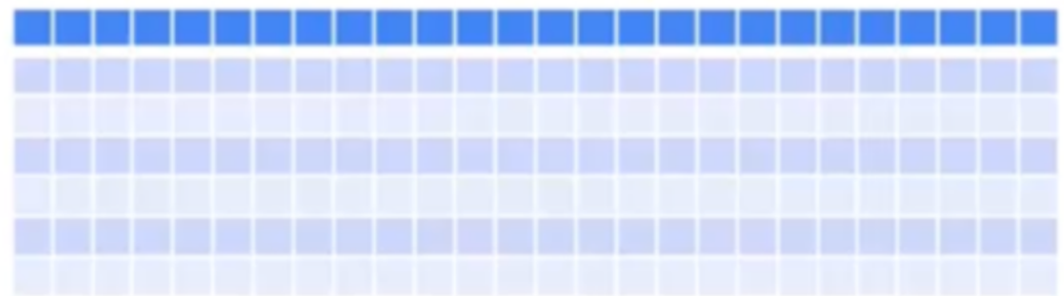


converts it into a smaller one.

DeepLearning.AI

Dimensionality Reduction

- Leads to smaller datasets
- Easier to visualize



visulize

Customer Age	Account Age	Days Since Login	Total Purchases	Total $ Spent
23	1 month	10 days	1	$100
71	45 months	2 days	5	$150
54	30 months	15 days	2	$70
36	22 months	12 days	4	$210

Reduce the table size,

And the question is, how should you do it?

DeepLearning.AI

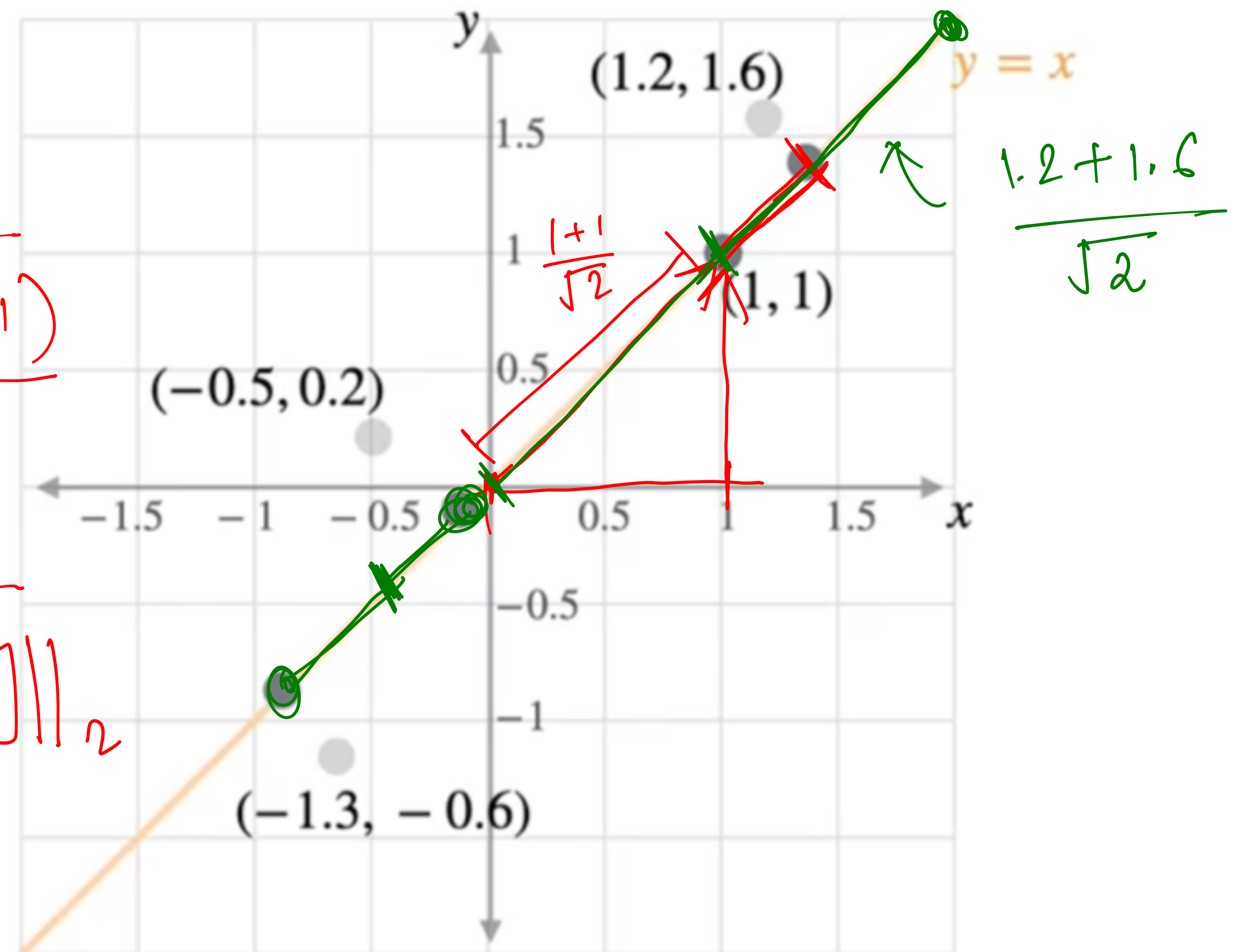
Projections

x	y
1.0	1.0
1.2	1.6
-0.5	0.2
-1.3	-0.6

Norm of $\frac{1}{\cdot}$

$\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{(1+1)}{\sqrt{2}}$

$\Rightarrow \frac{1}{\left\| \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\|_2}$



I'll start with the easiest example,
the point 1, 1 which didn't move at all.

DeepLearning.AI

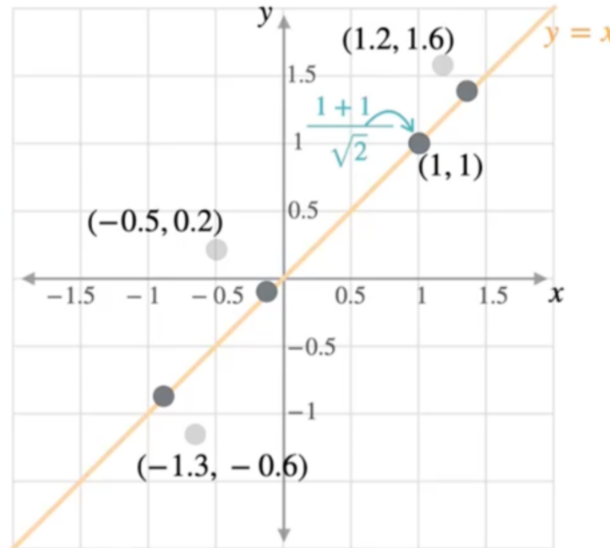
Projections

x	y
1.0	1.0
1.2	1.6
-0.5	0.2
-1.3	-0.6

Norm of 1

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix} \frac{1}{\sqrt{2}} = (1+1)/\sqrt{2}$$

$$\frac{1}{\left\| \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\|_2}$$



1

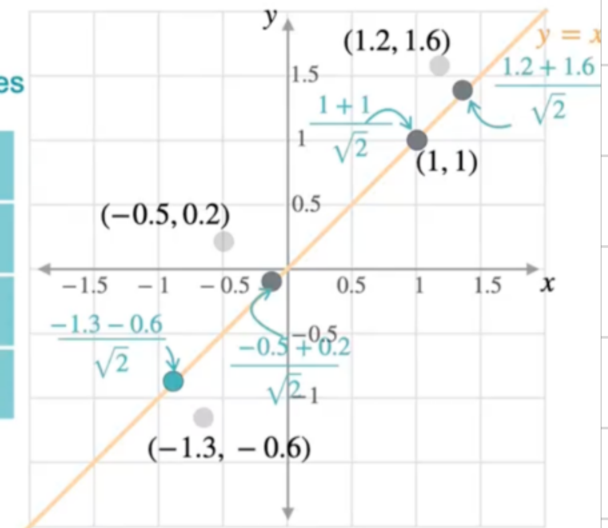
Multiplying by the vector projects the points along that vector and

Projections

x	y
1.0	1.0
1.2	1.6
-0.5	0.2
-1.3	-0.6

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix} \frac{1}{\sqrt{2}} =$$

Final coordinates
1.4142
1.9799
-0.2121
-1.344



4

So the four points with two variables each simplify to

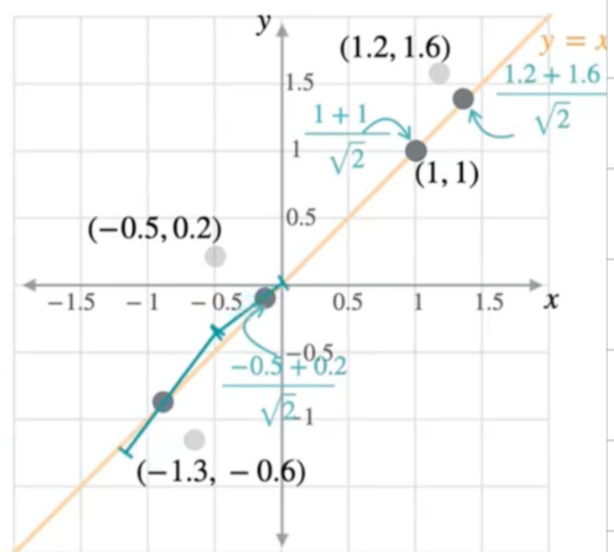
Projections

3

x	y
1.0	1.0
1.2	1.6
-0.5	0.2
-1.3	-0.6

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix} \frac{1}{\sqrt{2}} =$$

$(1+1)/\sqrt{2}$
$(1.2+1.6)/\sqrt{2}$
$(-0.5+0.2)/\sqrt{2}$



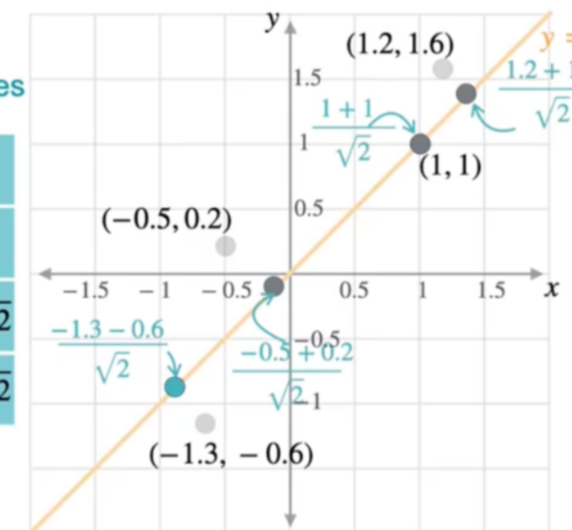
Projections

2

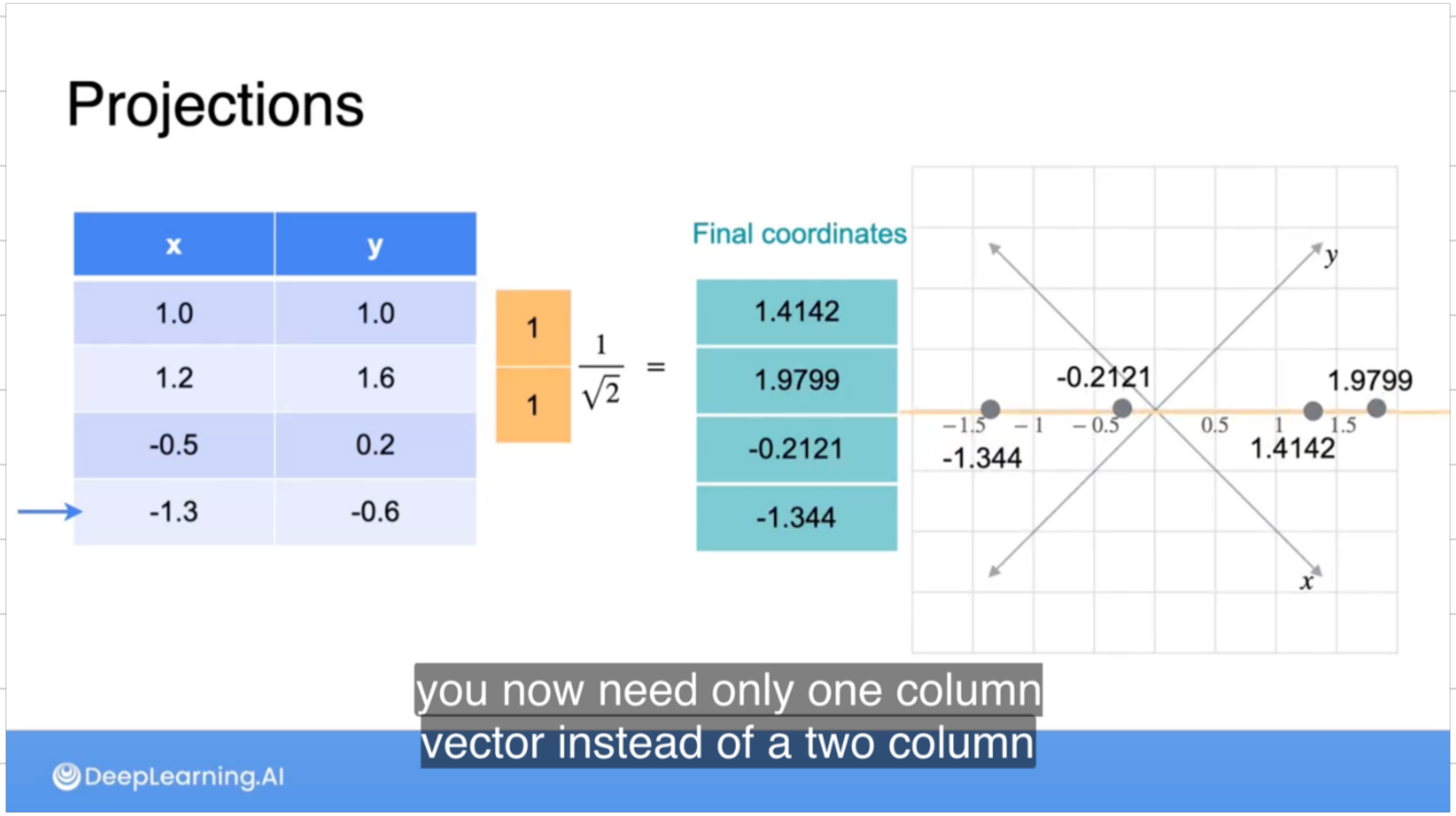
x	y
1.0	1.0
1.2	1.6
-0.5	0.2
-1.3	-0.6

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix} \frac{1}{\sqrt{2}} =$$

Final coordinates
$(1+1)/\sqrt{2}$
$(1.2+1.6)/\sqrt{2}$
$(-0.5+0.2)/\sqrt{2}$
$(-1.3,-0.6)/\sqrt{2}$



each point onto the x = y line.



x	y
1.0	1.0
1.2	1.6
-0.5	0.2
-1.3	-0.6

Projections:

To project a Matrix onto a vector v

$$A_p = A \cdot \frac{v}{\|v\|_2}$$

$$\underbrace{r \times c \quad c \times 1}$$

so, $r \times 1$

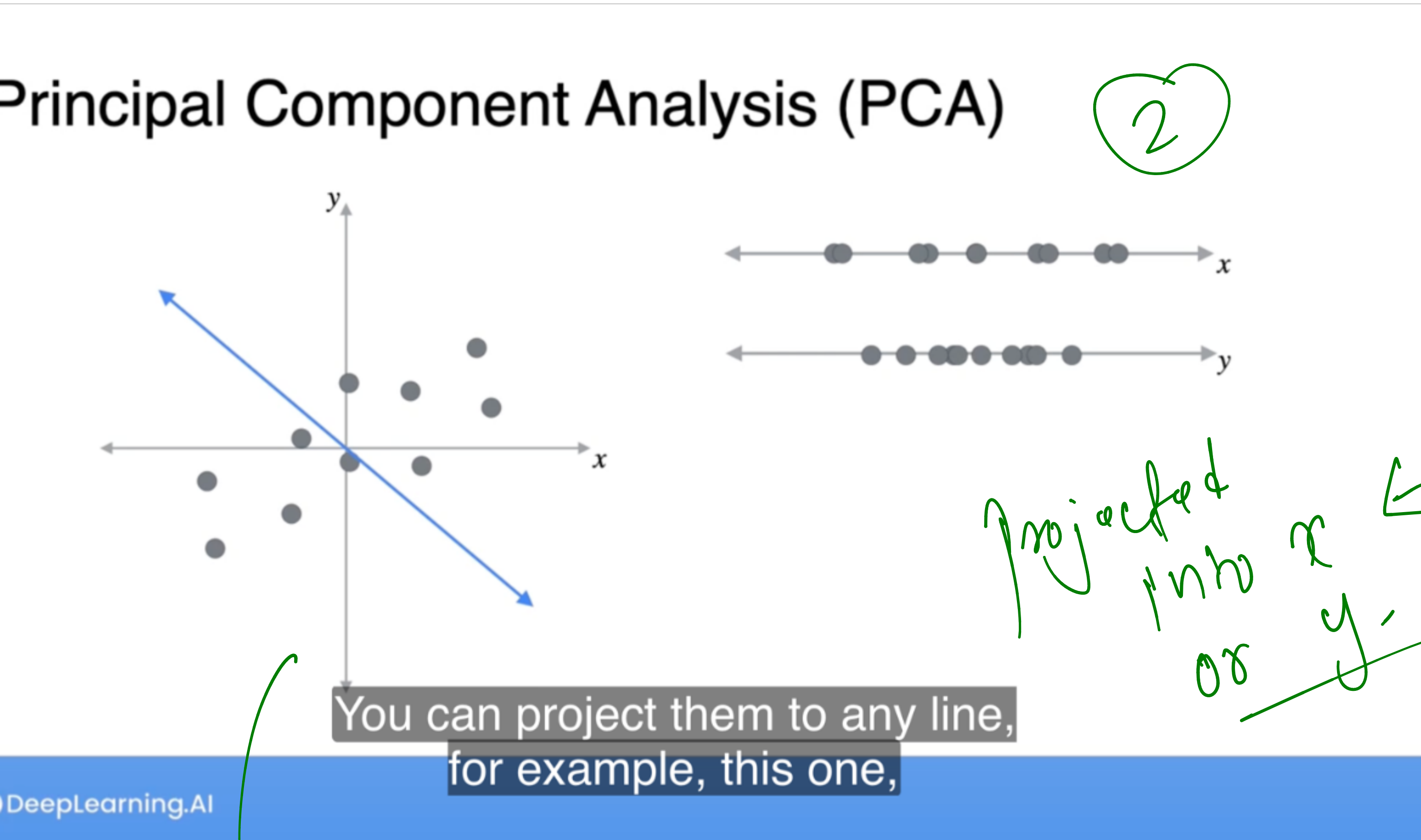
$$A_p = A \left[\frac{v_1}{\|v_1\|_2} \quad \frac{v_2}{\|v_2\|_2} \right]$$

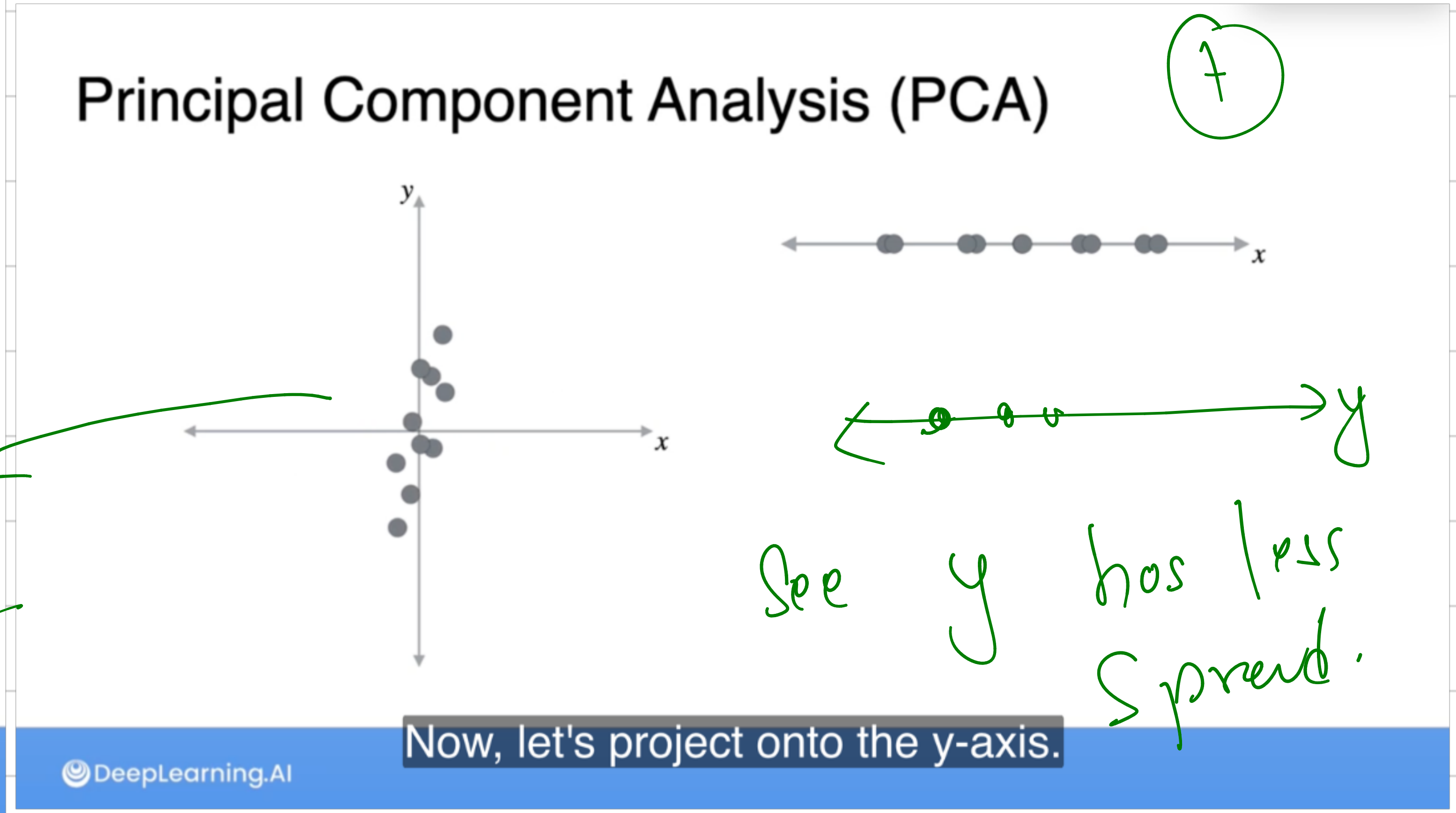
$r \times 2 \Leftarrow r \times c \quad c \times 2$.

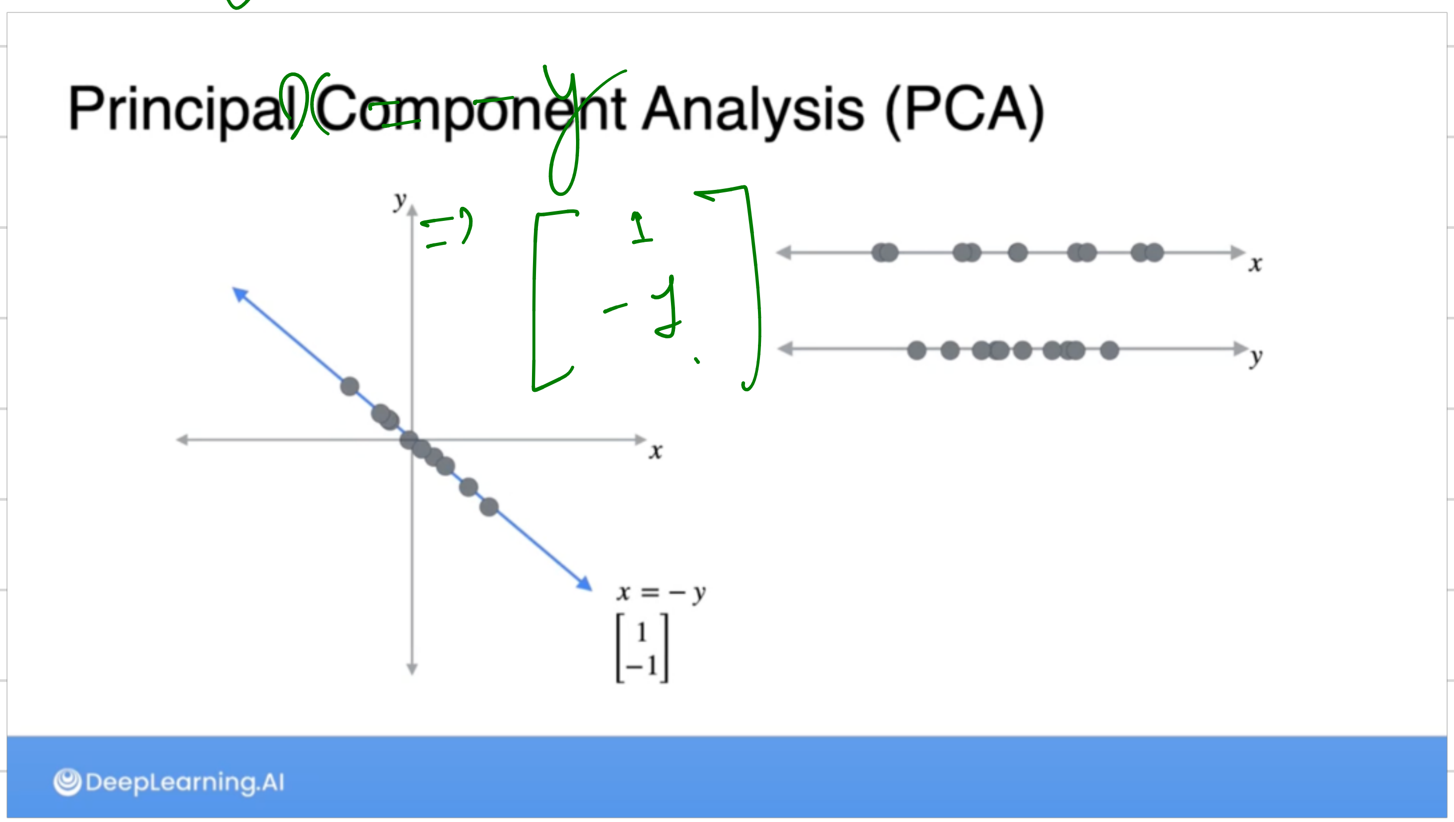
$\boxed{r \times 2}$

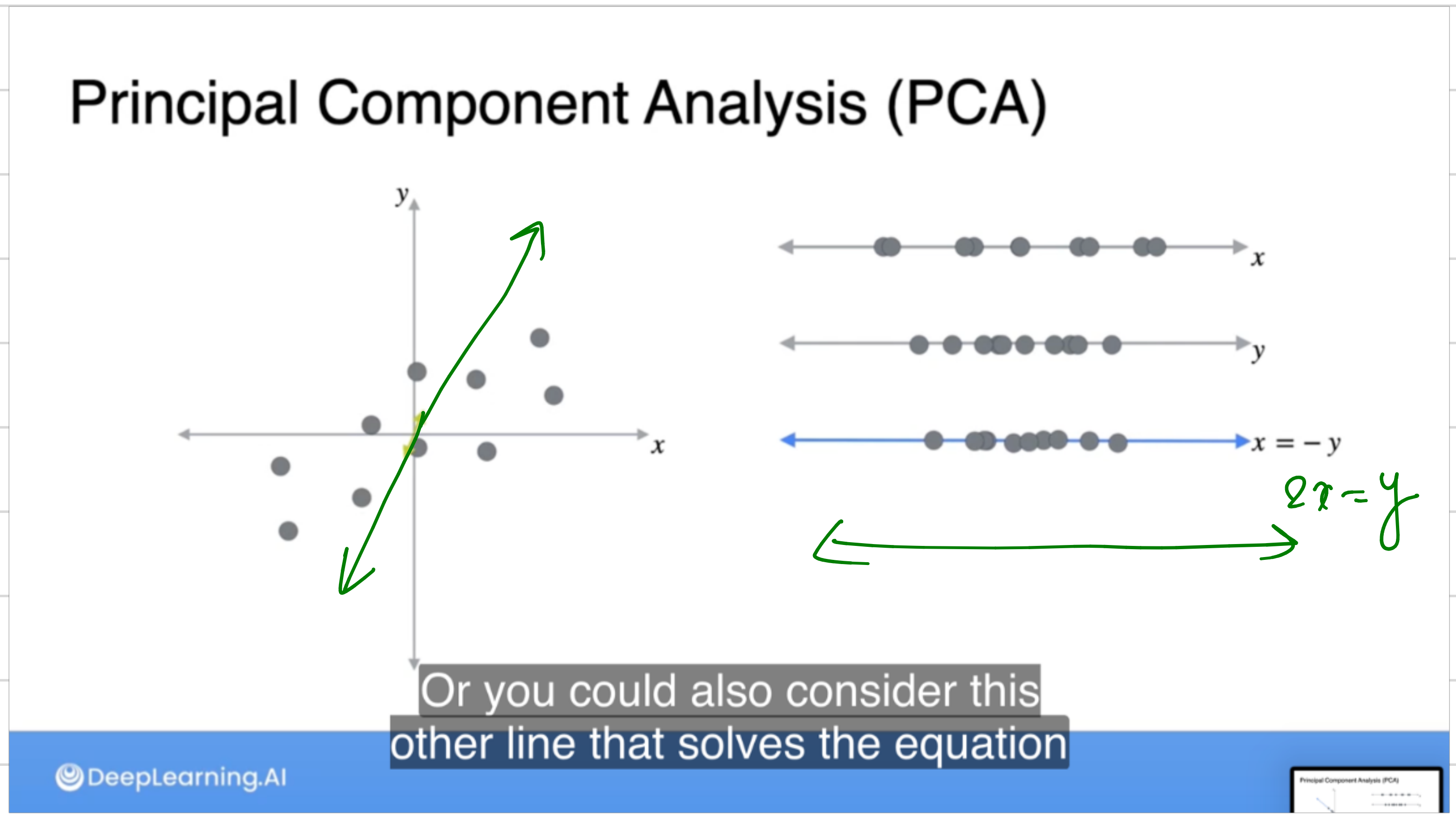
$$\boxed{A_p = AV}$$

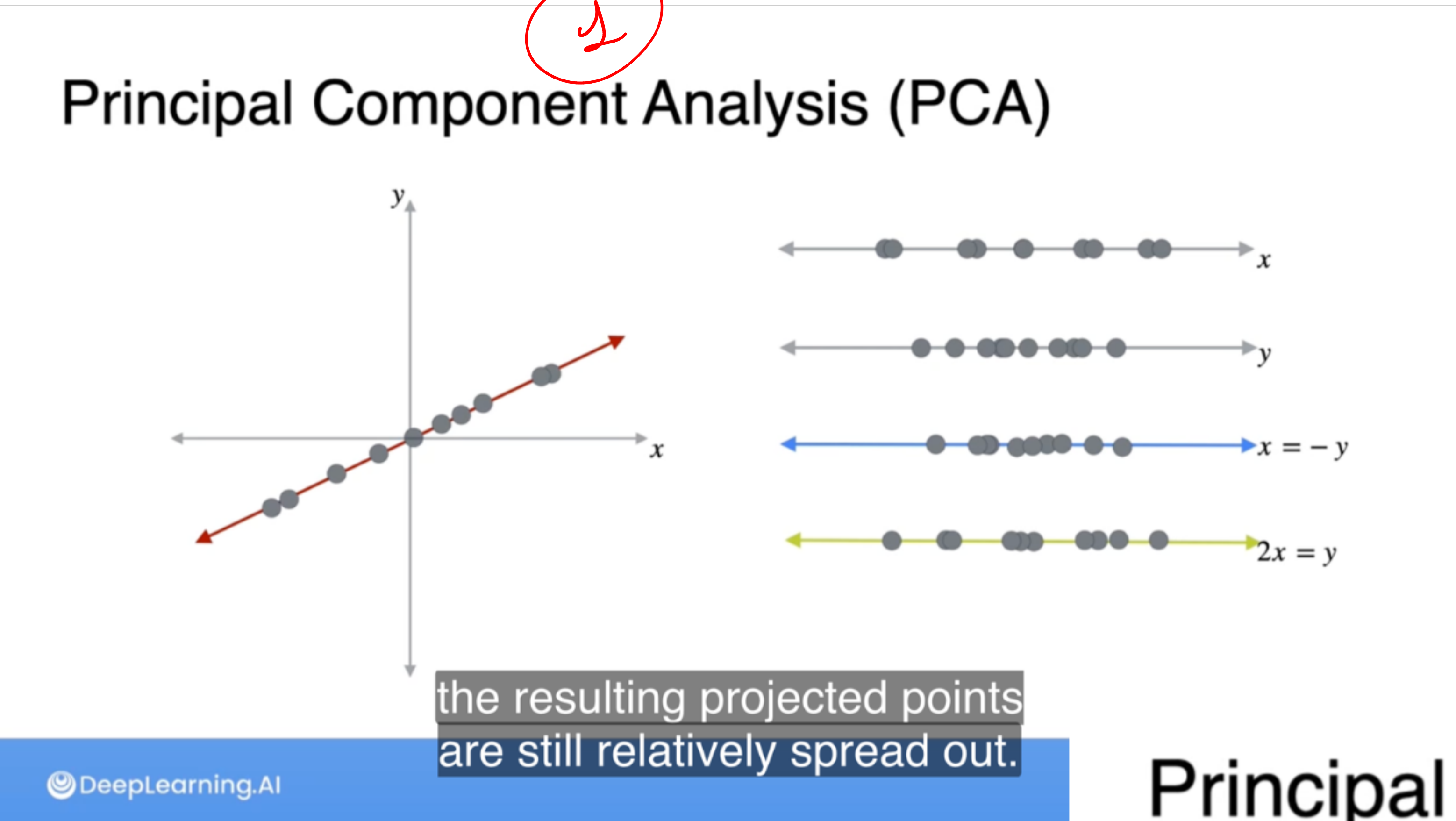
↳ Projection helps in information Reduction.

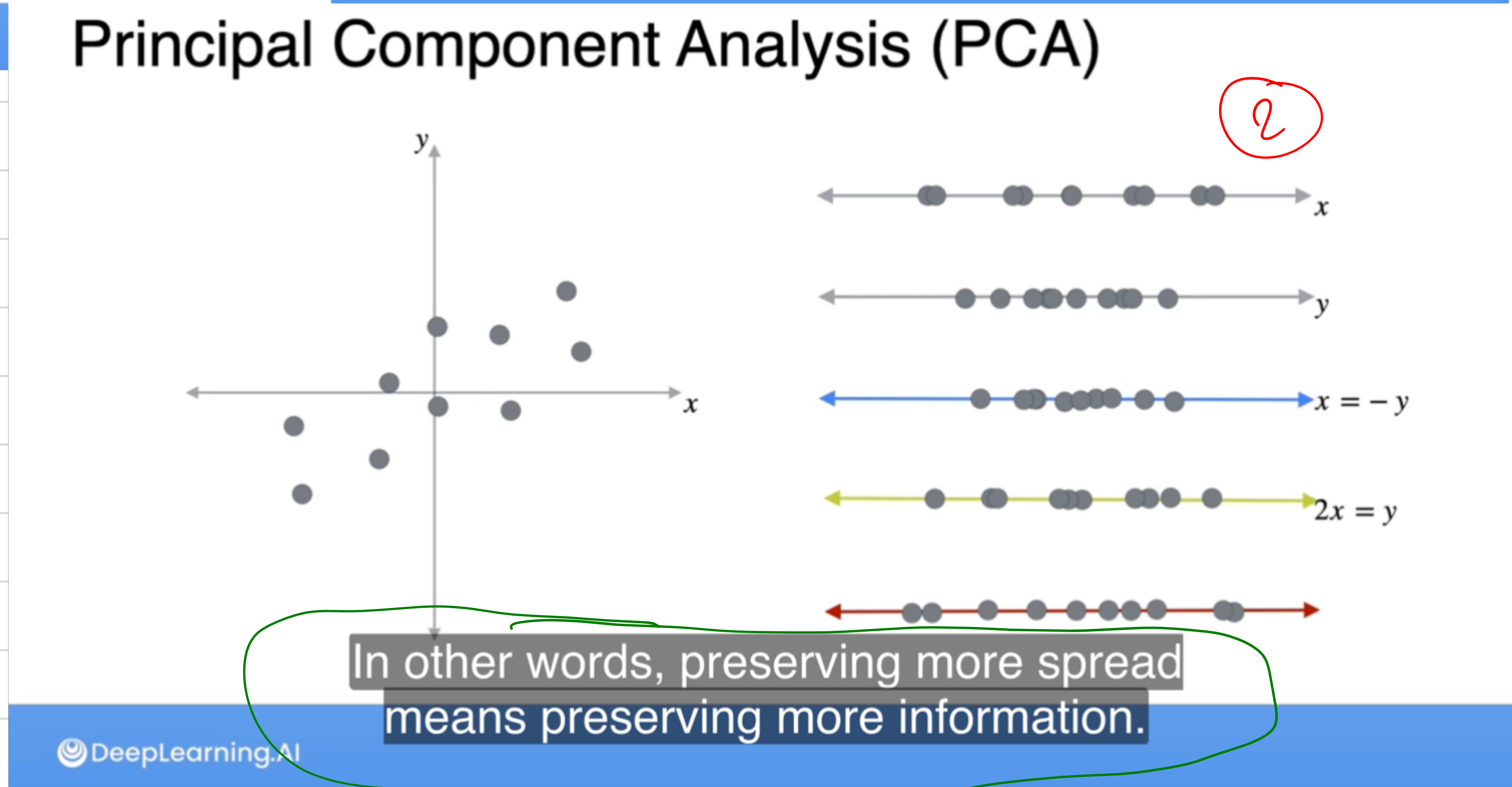

Principal Component Analysis (PCA)
2
y
x
x
y
Projected into x or y.
You can project them to any line,
for example, this one,
DeepLearning.AI

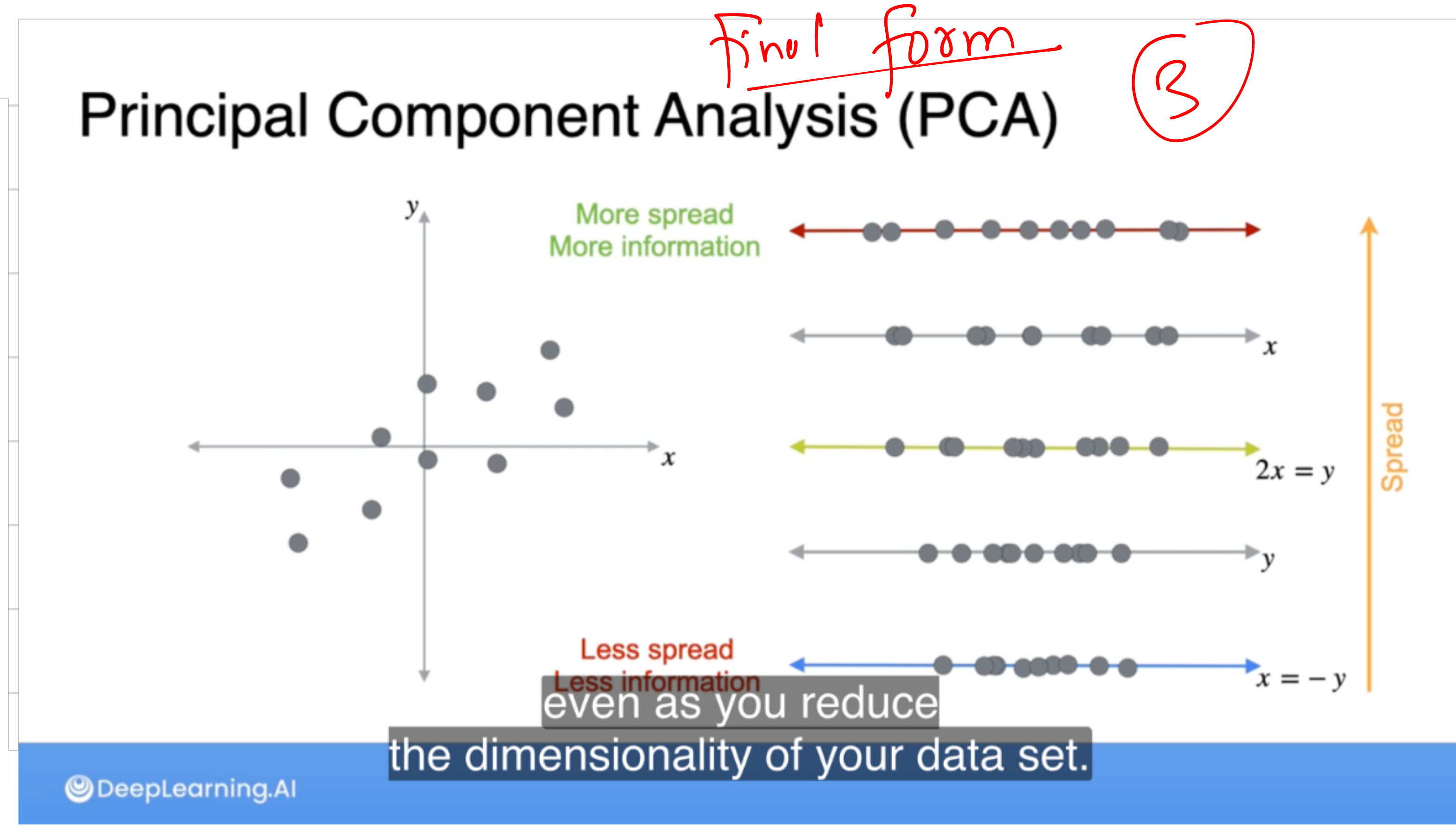

Principal Component Analysis (PCA)
7
y
x
x
y
See y has less spread.
Now, let's project onto the y-axis.
DeepLearning.AI


Principal Component Analysis (PCA)
x = -y
=> [1, -1]
y
x
x
y
x = − y
[1, −1]
DeepLearning.AI


Principal Component Analysis (PCA)
y
x
x
y
x = − y
2x = y
Or you could also consider this
other line that solves the equation
DeepLearning.AI







PCA → final form is sorted according to the more to less spread.

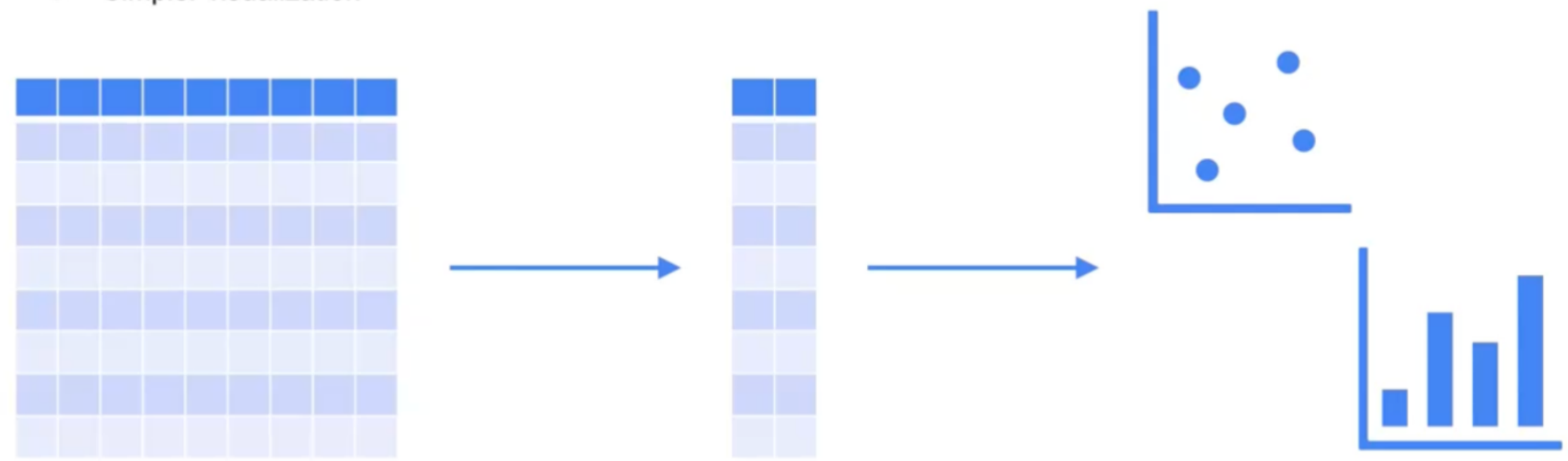
PCA Mathematical Formulation

$$n = (x_1, x_2, x_3, x_4, x_5)$$

Goal: Reduce to 2 variables

Benefits of Dimensionality Reduction

- Easier dataset to manage
- PCA reduces dimensions while minimizing information loss
- Simpler visualization



DeepLearning.AI

① Create Matrix.

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{15} \\ x_{21} & x_{22} & & x_{25} \\ \vdots & & & \vdots \\ x_{n1} & x_{n2} & & x_{n5} \end{bmatrix}$$

② Center the Data $\rightarrow$

$$X - \mu = \begin{bmatrix} x_{11} - \mu_1 & x_{12} - \mu_2 & \cdots & x_{1n} - \mu_n \\ x_{21} - \mu_1 & x_{22} - \mu_2 & \cdots & x_{2n} - \mu_n \\ \vdots & \vdots & & \vdots \\ x_{n1} - \mu_n & x_{n2} - \mu_n & \cdots & x_{mn} - \mu_n \end{bmatrix}$$

(3) Calculate Co-variance Matrix.

$$C = \frac{1}{n-1}(X-\mu)^T(X-\mu)$$

(4) Calculate Eigen Vectors and Eigen Values

Big → Small → sort them. tho.

(5) Create Projection Matrix: $V = \begin{bmatrix} \frac{v_1}{\|v_1\|_2} & \frac{v_2}{\|v_2\|_2} \end{bmatrix}$

(6) Project Centered Data: $X_{PCA} = (X-\mu)V$

Source: Linear Algebra for Machine Learning & Data Science offered by Coursera —

PCA Mathematical formulation

You have n observations of 5 variables $(x_1, x_2, x_3, x_4, x_5)$ Goal: Reduce to 2 variables

1. Create matrix

$$\begin{array}{cc} & \text{5 variables} \\ X = & \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{15} \\ x_{21} & x_{22} & \cdots & x_{25} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{n5} \end{bmatrix} \end{array} \; n \text{ Observations}$$

2. Center the data

$$X - \mu = \begin{bmatrix} x_{11}-\mu_1 & x_{12}-\mu_2 & \cdots & x_{15}-\mu_5 \\ x_{21}-\mu_1 & x_{22}-\mu_2 & \cdots & x_{25}-\mu_5 \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1}-\mu_1 & x_{n2}-\mu_2 & \cdots & x_{n5}-\mu_5 \end{bmatrix}$$

PCA Mathematical formulation

You have n observations of 5 variables $(x_1, x_2, x_3, x_4, x_5)$ Goal: Reduce to 2 variables

4. Calculate Eigenvectors and Eigenvalues

5. Create Projection Matrix

6. Project Centered Data

Big

$\lambda_1 \quad v_1$

$\lambda_2 \quad v_2$

$\lambda_3 \quad v_3$

$\lambda_4 \quad v_4$

$\lambda_5 \quad v_5$

Small

$$V = \begin{bmatrix} \frac{v_1}{\|v_1\|_2} & \frac{v_2}{\|v_2\|_2} \end{bmatrix}$$

$$X_{PCA} = (X - \mu)V$$

perform PCA for dimensionality reduction of your data.

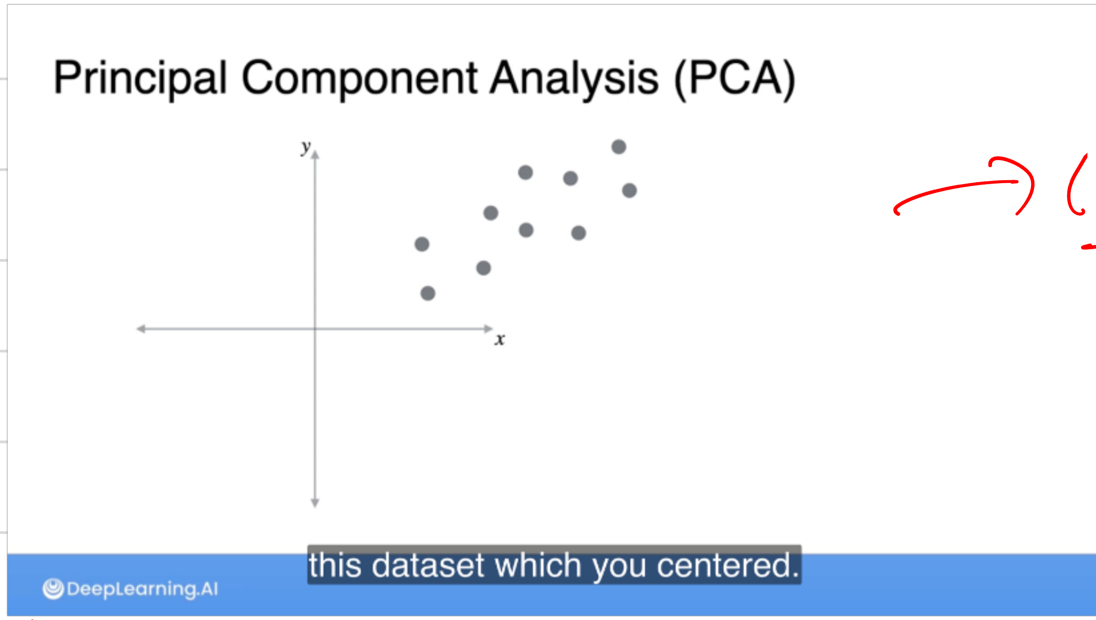
DeepLearning.AI

PCA - Overview

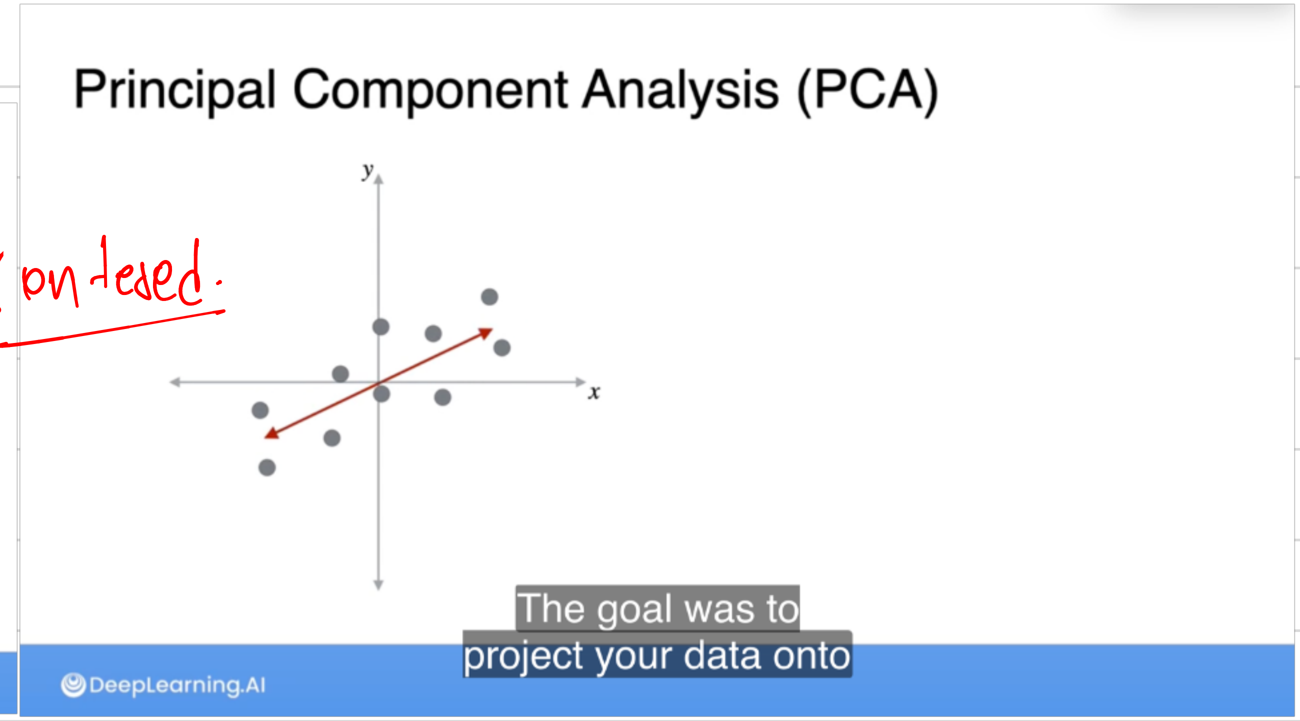
Eigen Values

Eigenvectors

Only stretches No Change in direction.



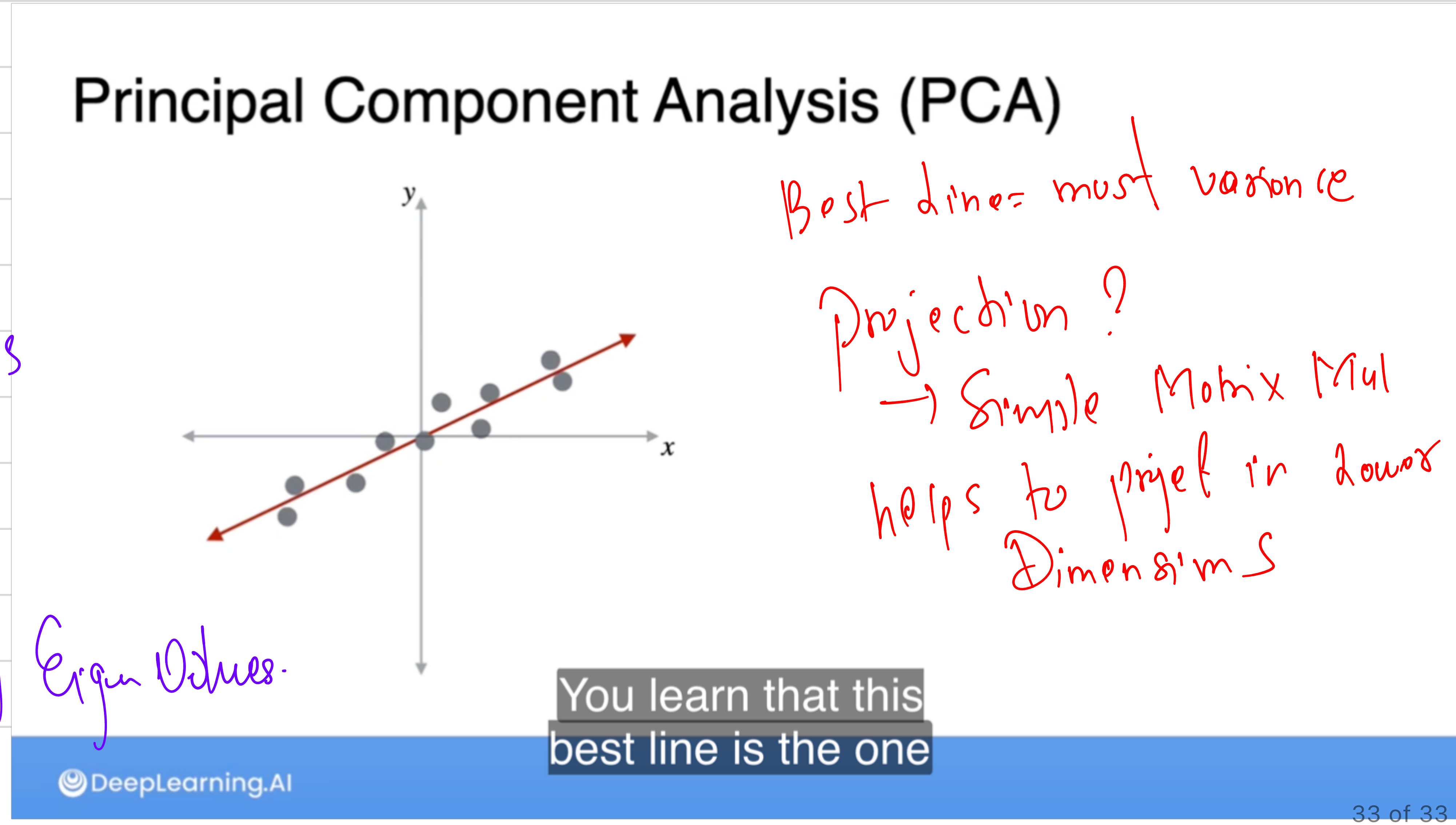
→ Centered.



Covariance Matrix.

Relationship in Variables

PCA ← Projection, Covariance, Eigen Vectors | Eigen Values.



Best line = most variance

Projection ?

→ Simple Matrix Mul

helps to project in lower Dimensions