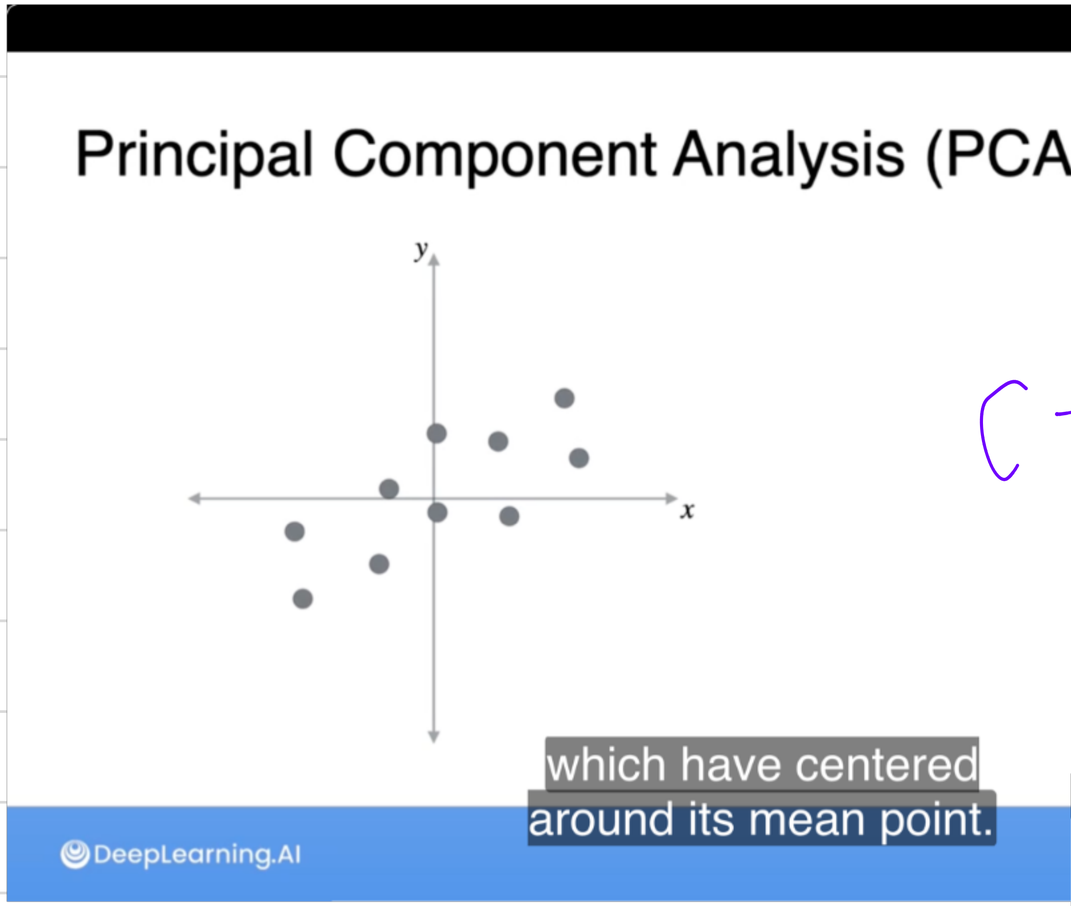
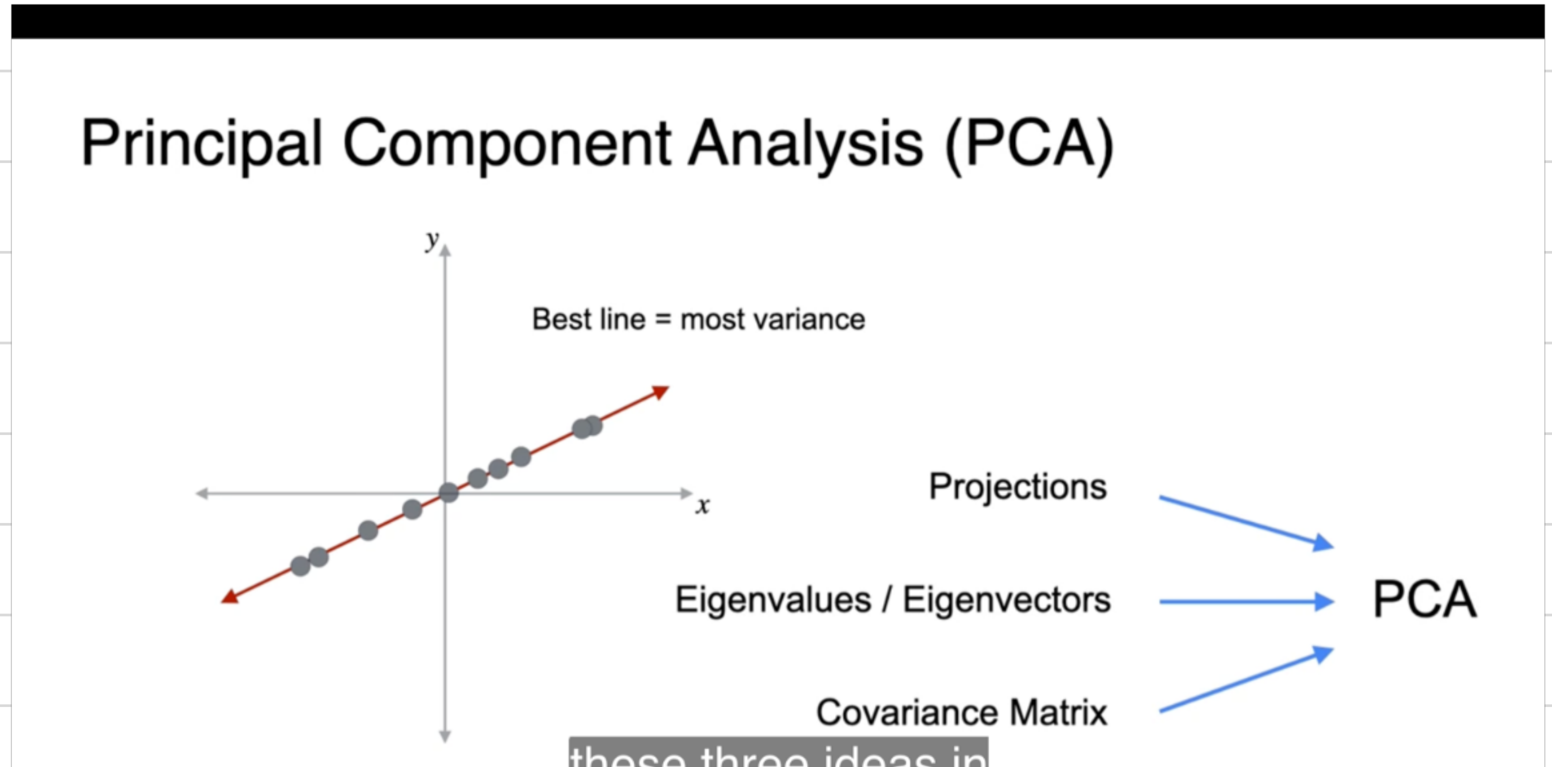
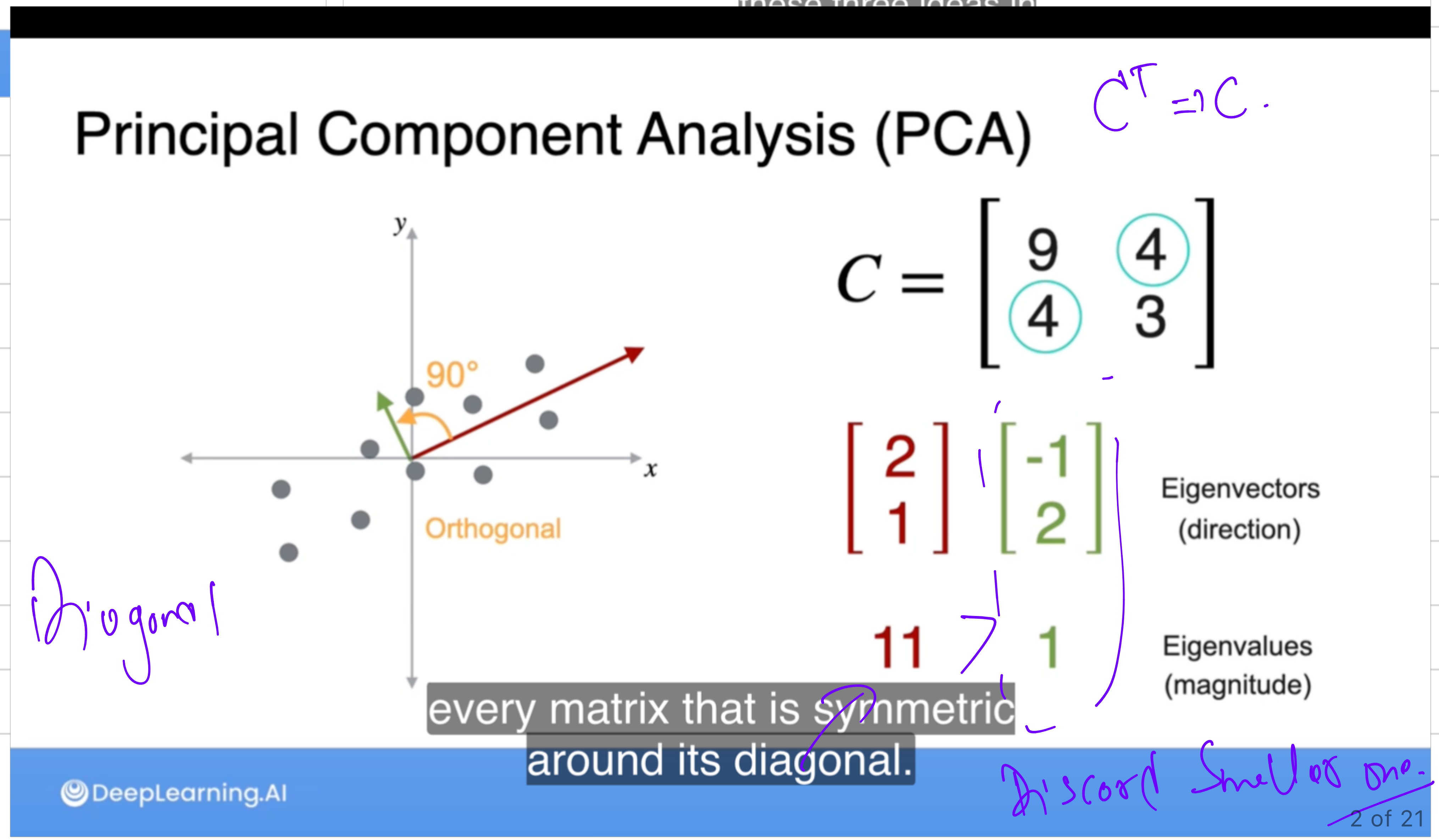
Day-94, Falgun 20, 2081 (Mar 4, 2025)

1. Finding unusual events → Unsupervised learning.
2. Anomaly Detection Algorithm
3. Developing and Evaluating An Anomaly Detection System
4. PCA Overview, Variance and Covariance, How PCA works ?
5. Discrete Dynamical Systems.
6. Co-Variance Matrix.



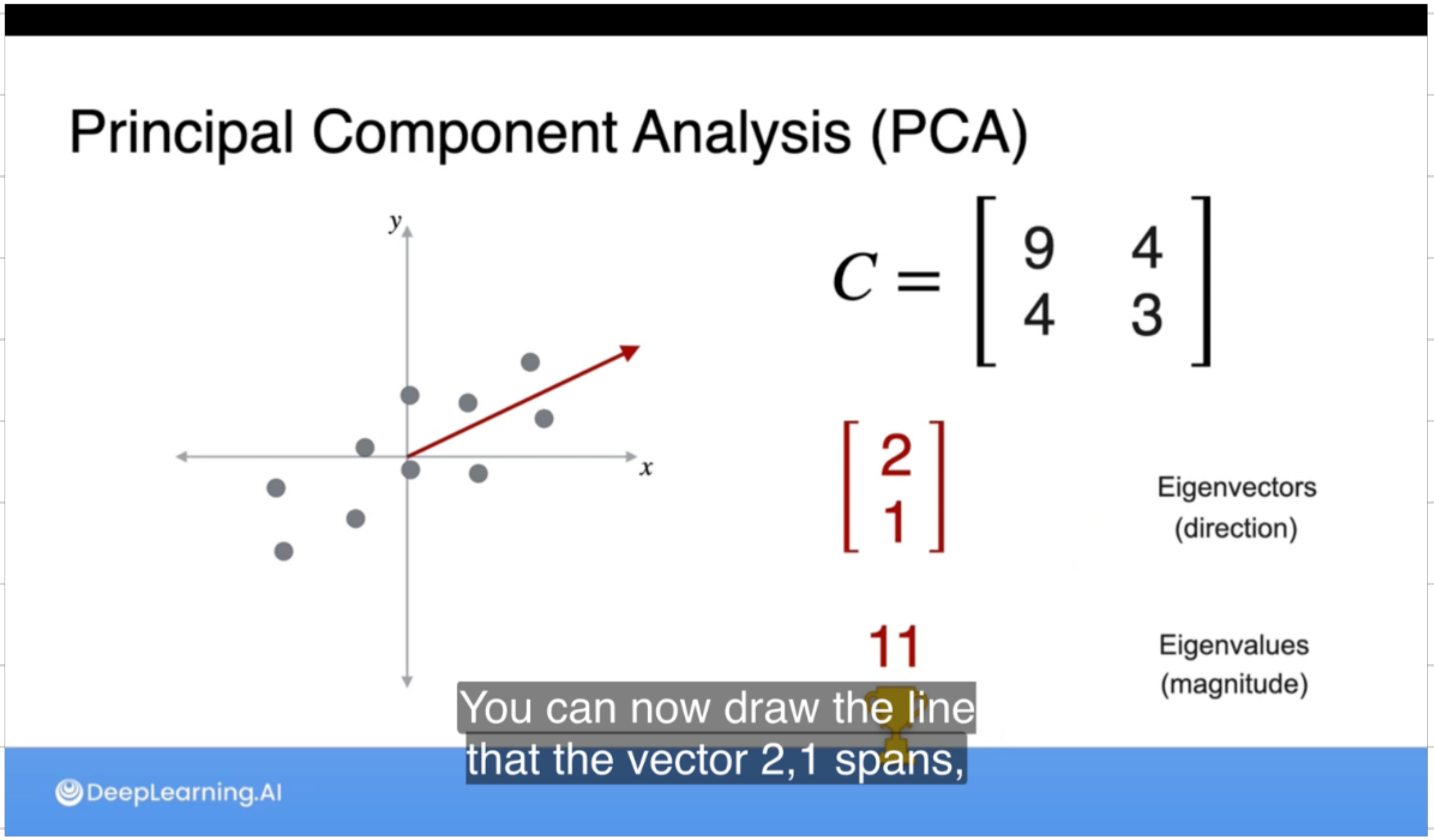


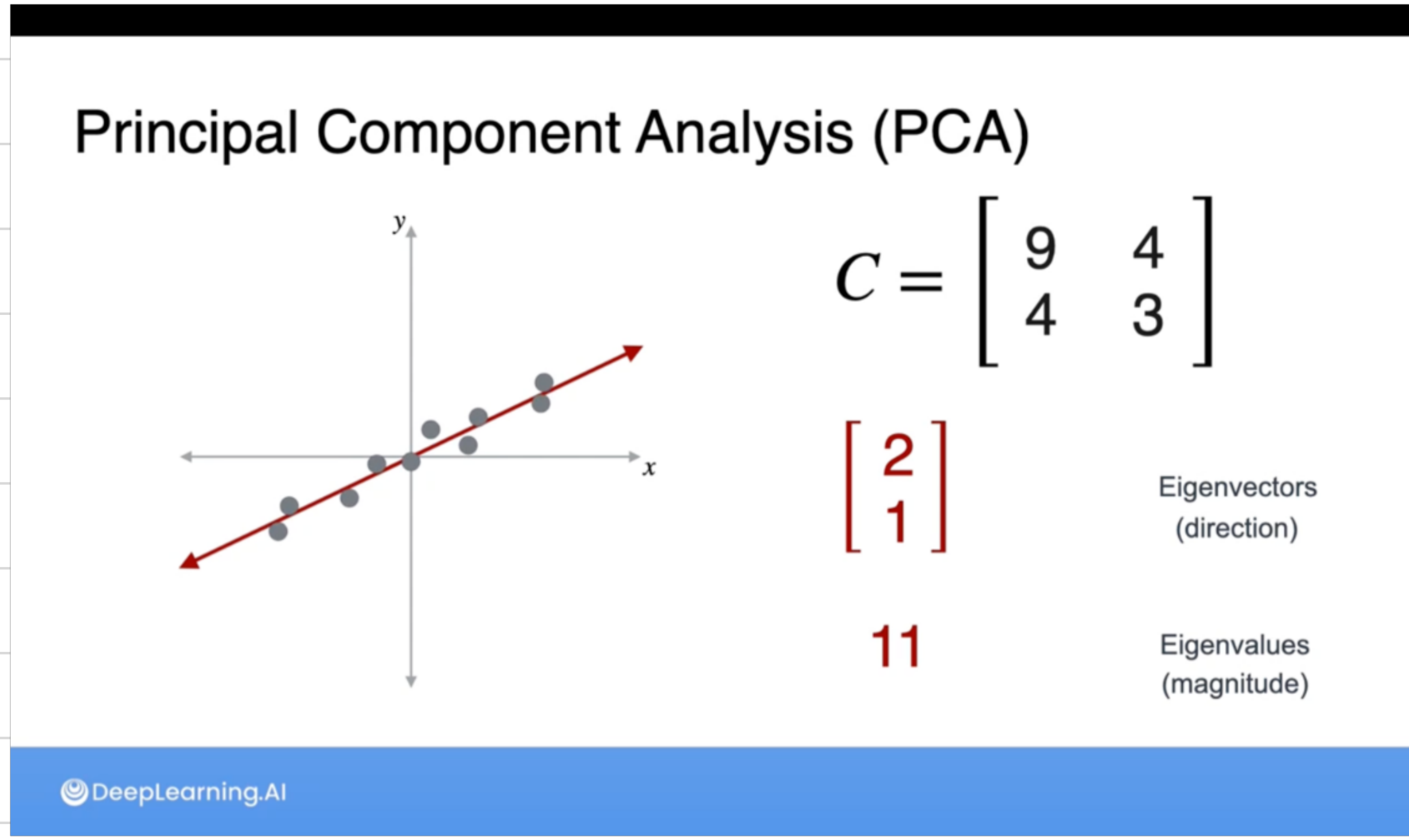


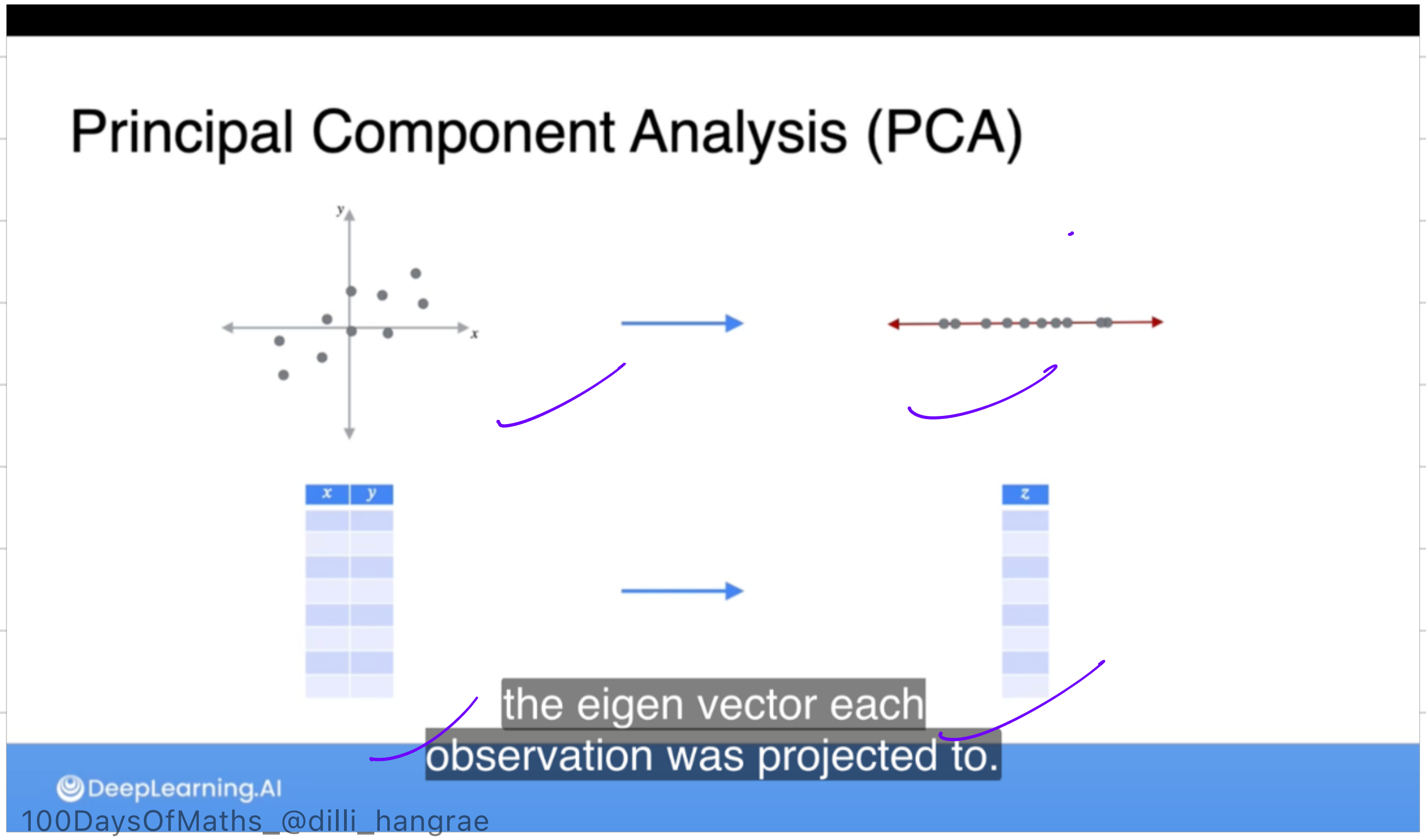
↳ Orthogonal is true for Eigen Vectors & Values.

↳ Every Co-Variance is Symmetric about Diagonal

$$\begin{bmatrix} 2 & -1 \\ 1 & 2 \end{bmatrix}$$







3D → 2D

2D → 1D.

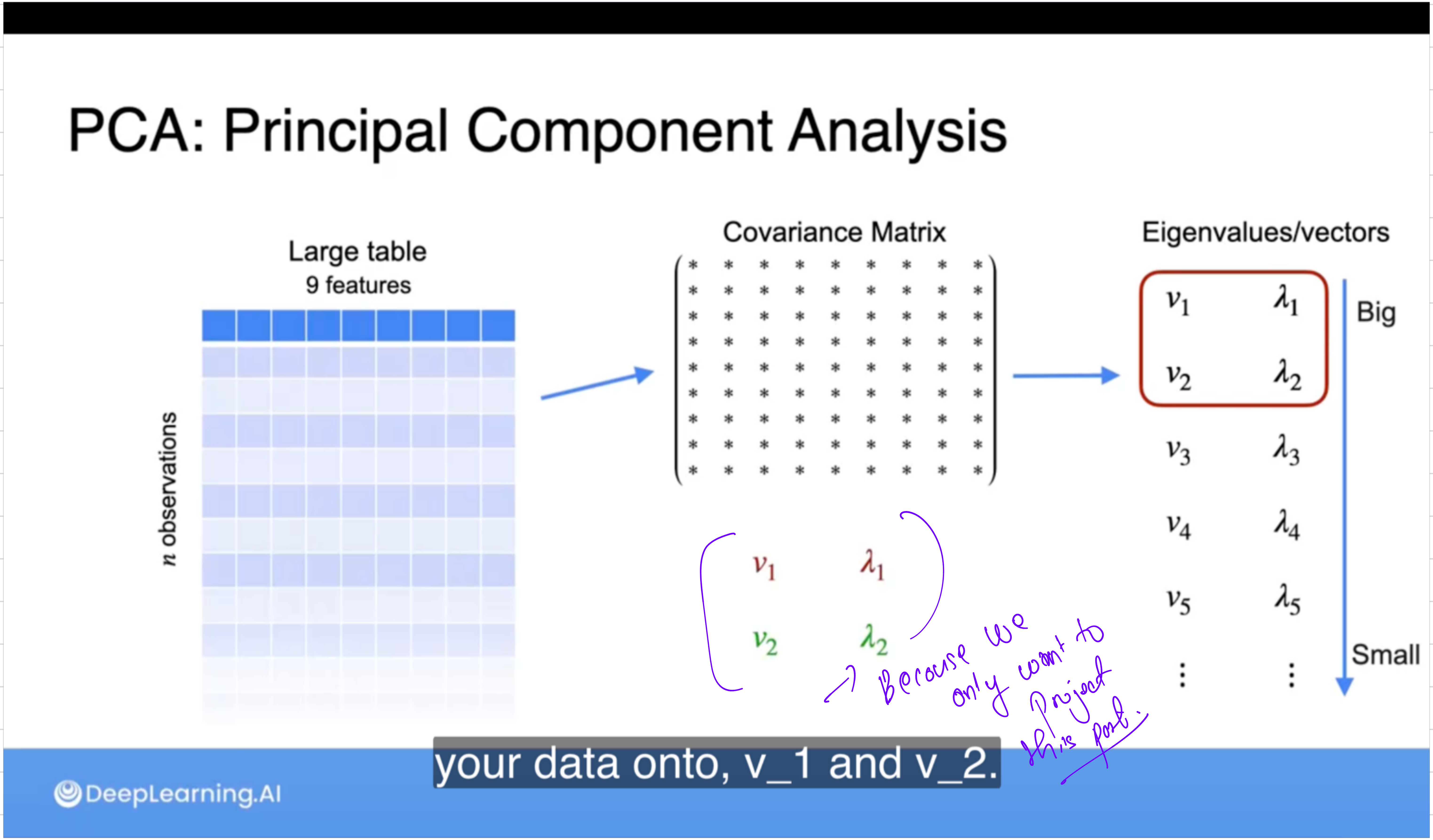
How $C = \begin{bmatrix} 9 & 4 \\ 4 & 3 \end{bmatrix}$

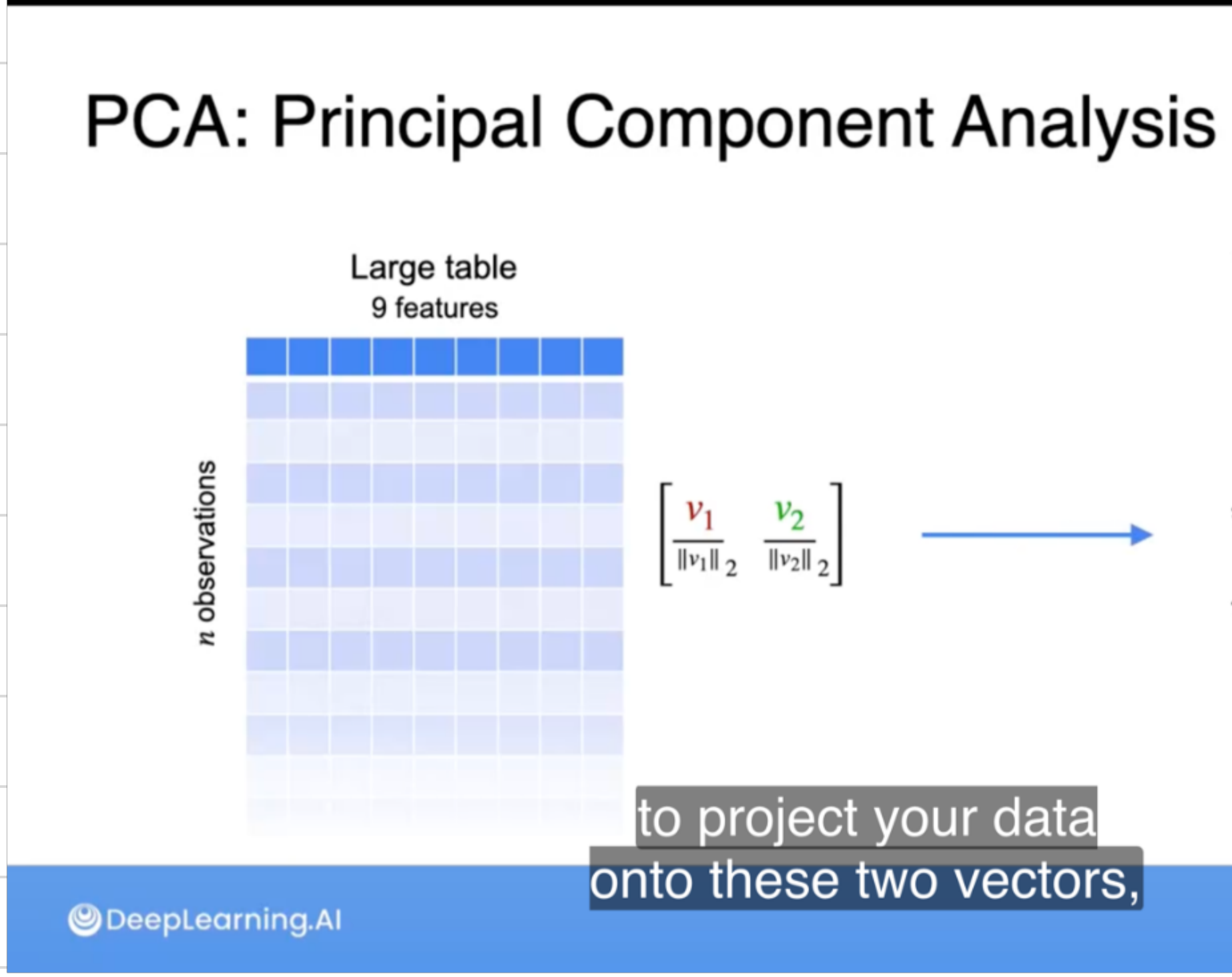
$C^T = \begin{bmatrix} 9 & 4 \\ 4 & 3 \end{bmatrix}$

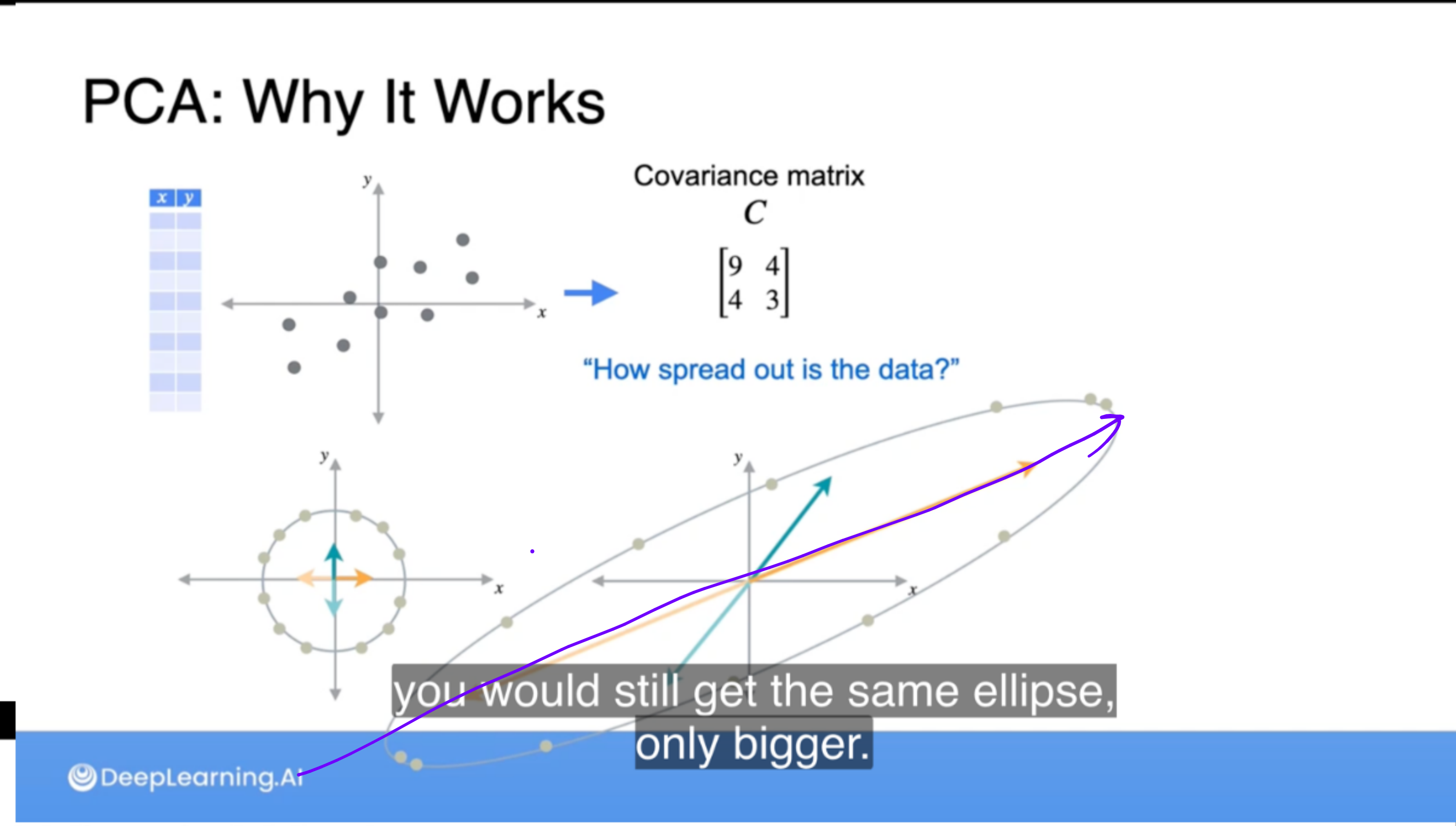
and Diagonal are Symmetry.

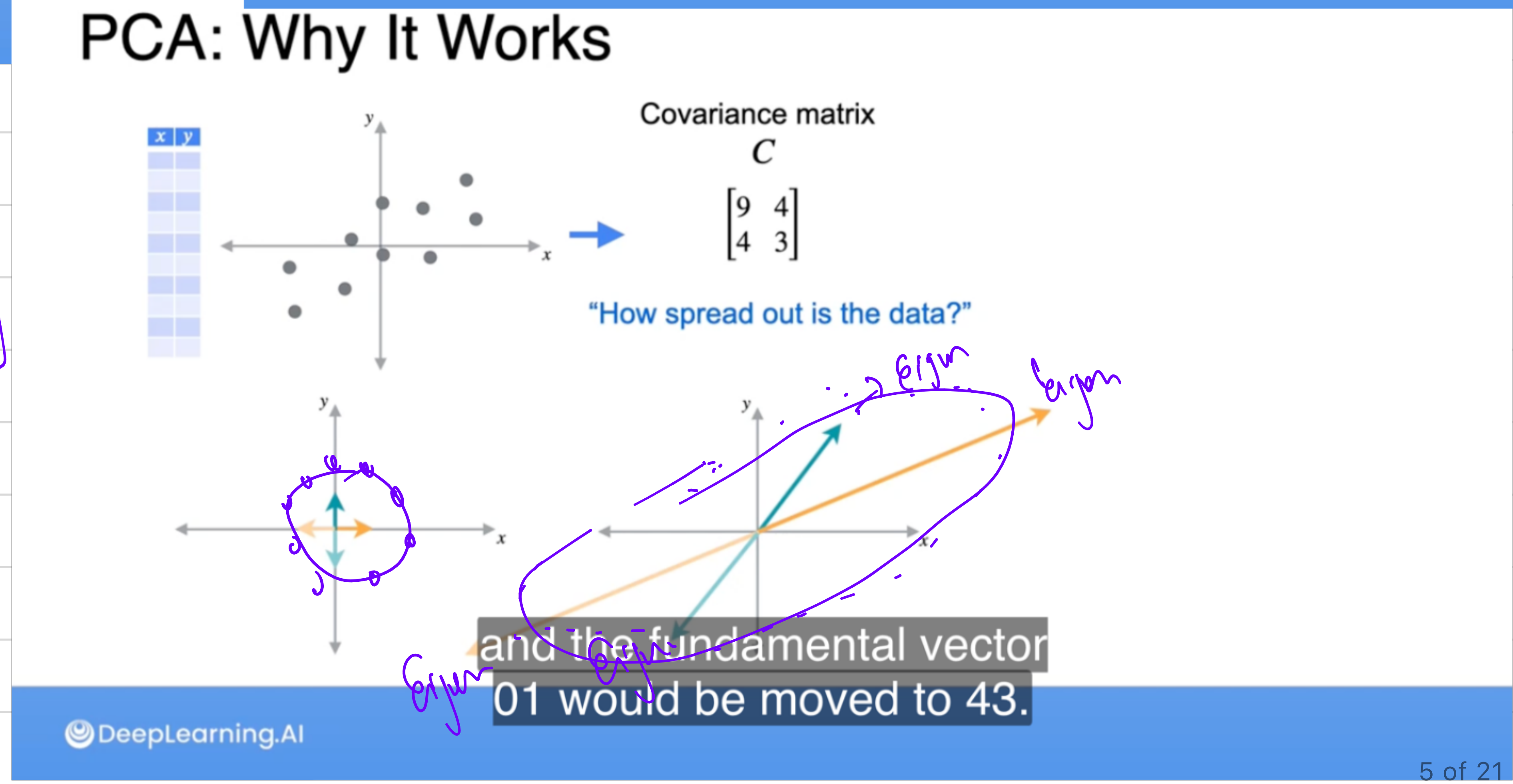
PCA:

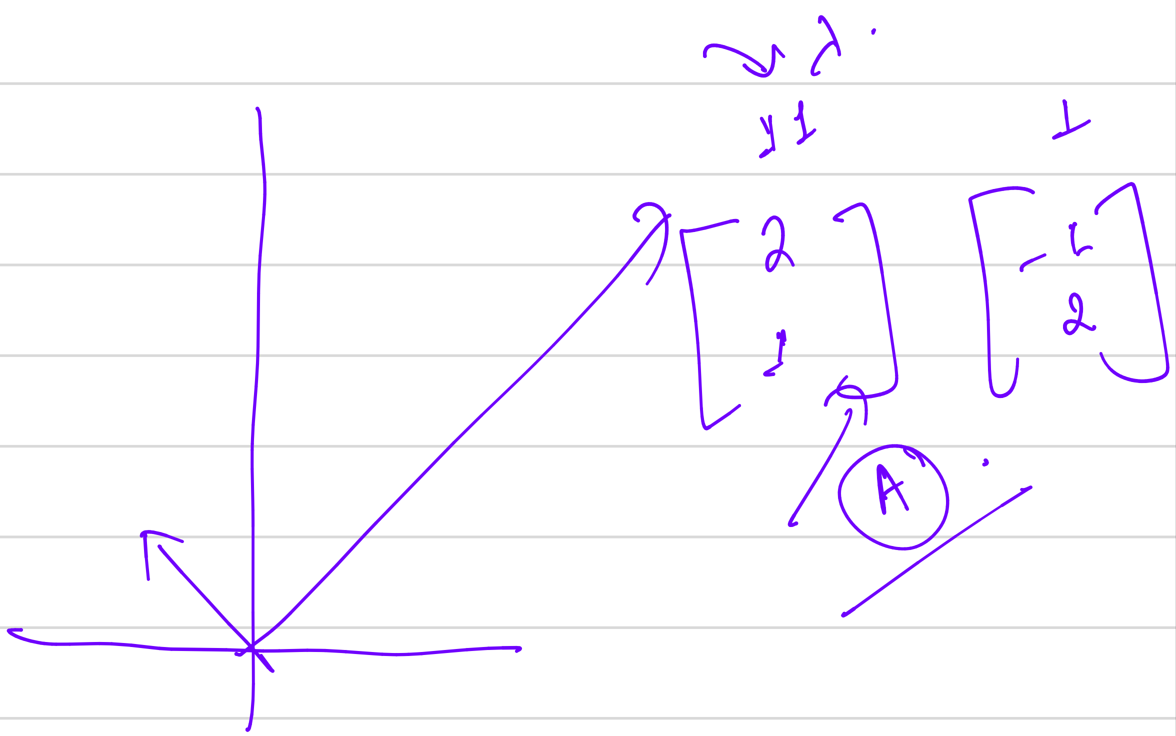
Eigen Vectors & Values are orthogonal (0,0).

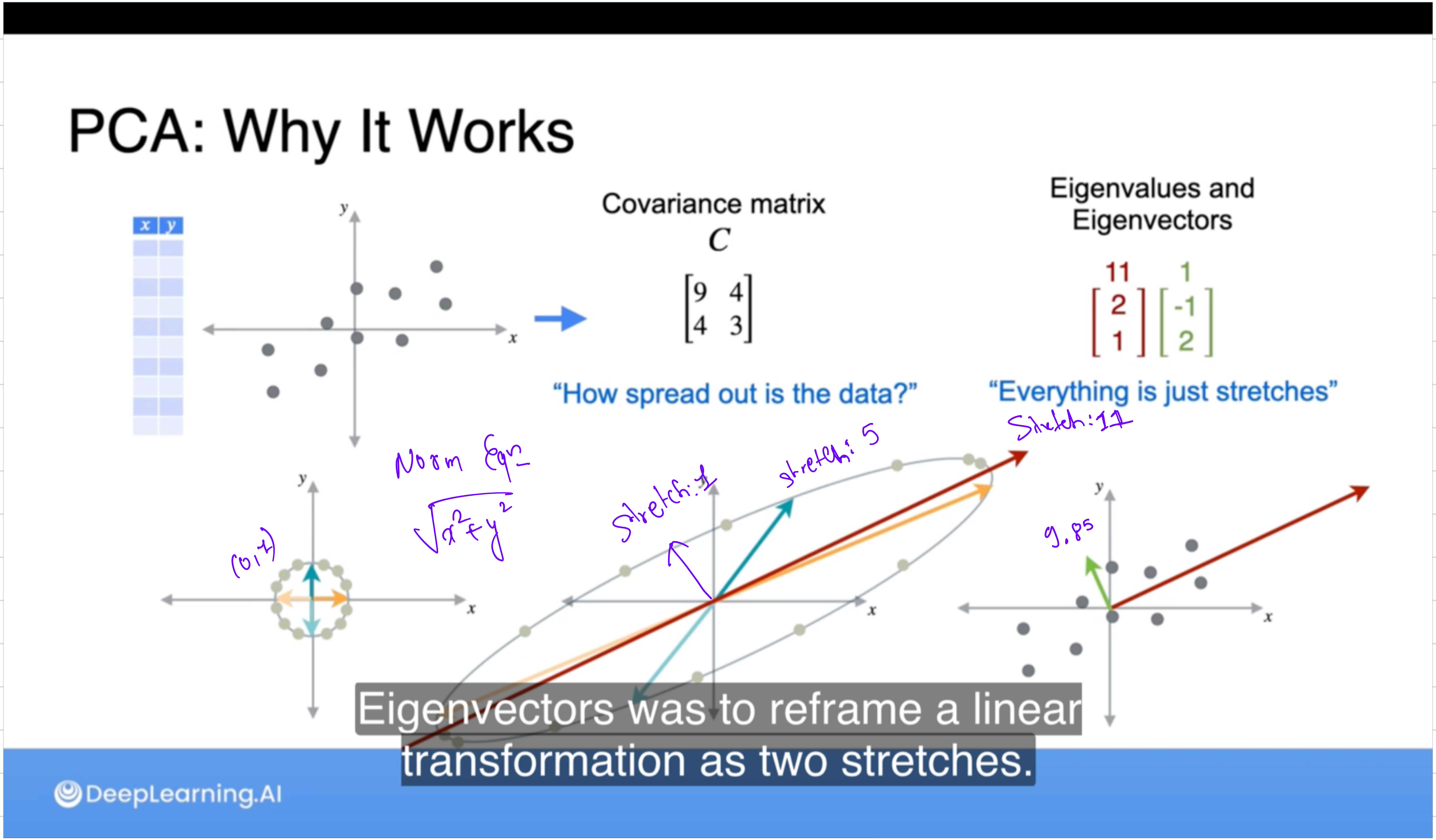

PCA: Principal Component Analysis
Large table
9 features
n observations
Covariance Matrix
Eigenvalues/vectors
v_1 λ_1
v_2 λ_2
v_3 λ_3
v_4 λ_4
v_5 λ_5
Big
Small
v_1 λ_1
v_2 λ_2
Because we only want to Project this part.
your data onto, v_1 and v_2.
DeepLearning.AI


PCA: Principal Component Analysis
Large table
9 features
n observations
Small table
2 features
n observations
to project your data
onto these two vectors,
DeepLearning.AI

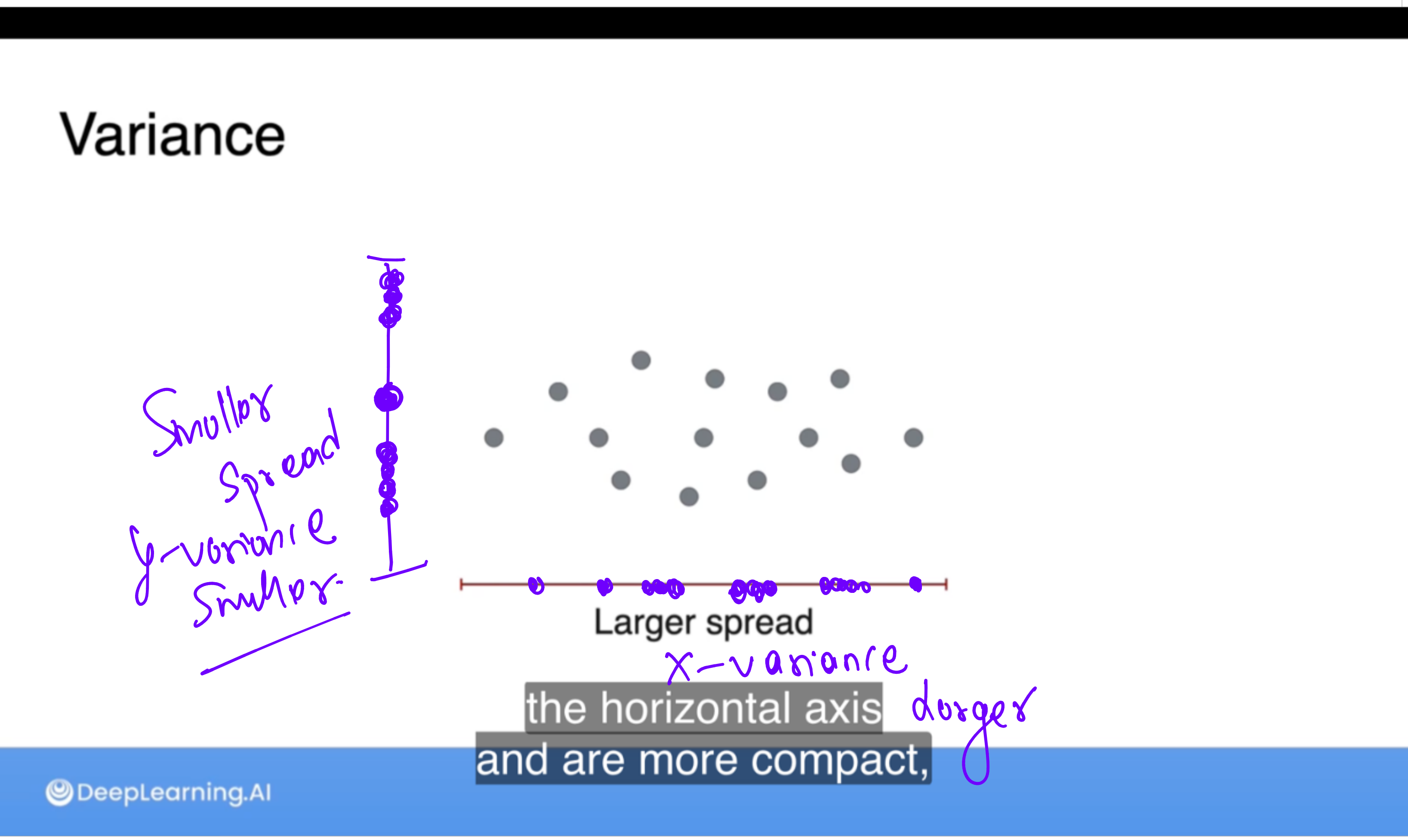

PCA: Why It Works
Covariance matrix
C
"How spread out is the data?"
you would still get the same ellipse,
only bigger.
DeepLearning.AI

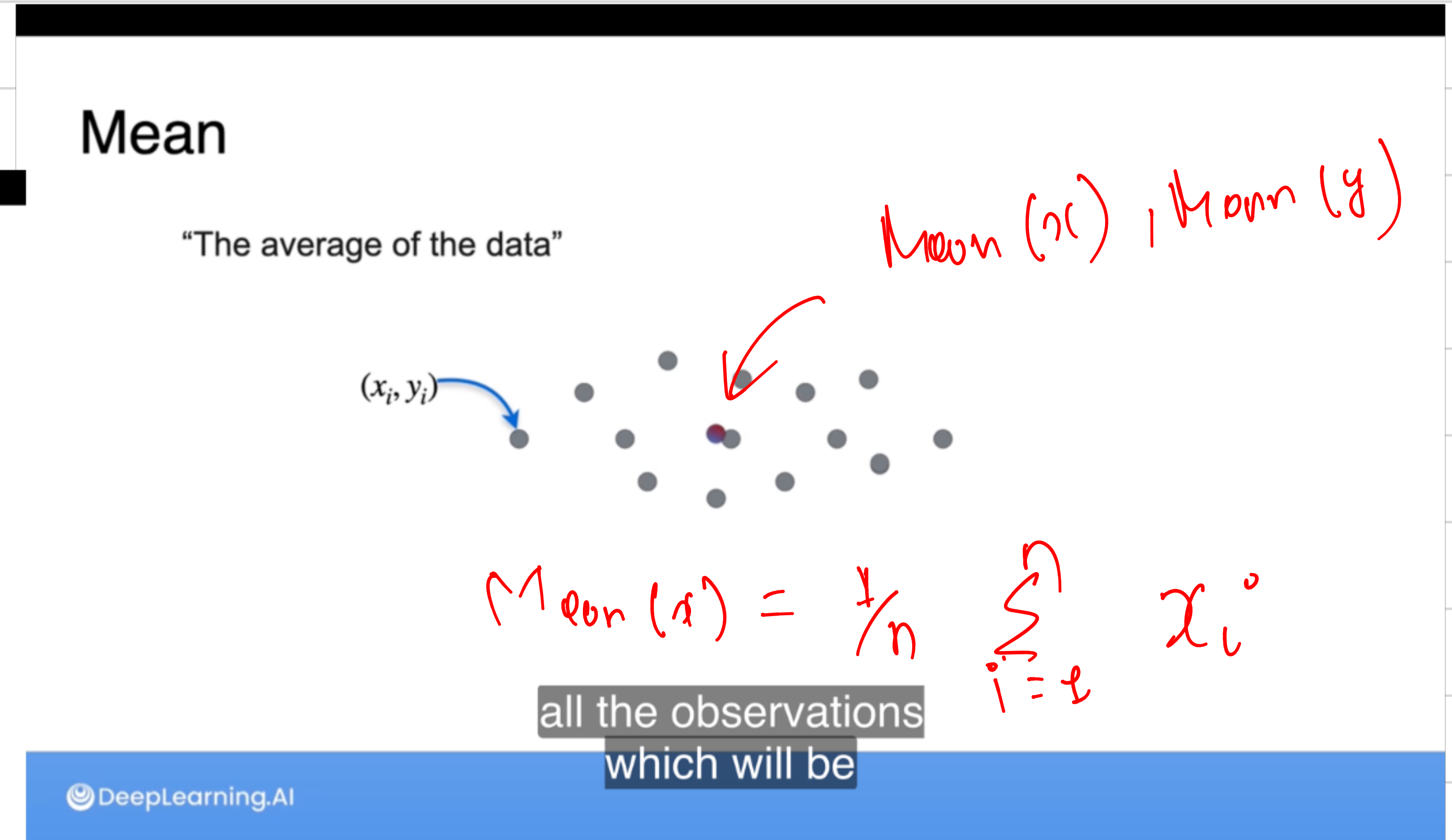

PCA: Why It Works
Covariance matrix
C
"How spread out is the data?"
and the fundamental vector
01 would be moved to 43.
DeepLearning.AI




PCA: Why It Works
x
y
y
x
Covariance matrix
C
[9 4]
[4 3]
"How spread out is the data?"
Eigenvalues and
Eigenvectors
11
[2 1]
1
[-1 2]
"Everything is just stretches"
Stretch: 11
Norm Eqn
√(x²+y²)
(0,1)
Stretch: 1
stretch: 5
9.85
y
x
y
x
y
x
Eigenvectors was to reframe a linear
transformation as two stretches.
DeepLearning.AI

Variance and Covariance:





Suppose $\bar{x} = 9$ for $n = 5$

$x_i - (\bar{x})^2$

$\boxed{\text{Var}(X) \Rightarrow 16}$

$$\text{Variance}(x) = \frac{1}{n-1} \sum_{i=1}^{n} \left(x_i - \text{Mean}(\bar{x})\right)^2$$

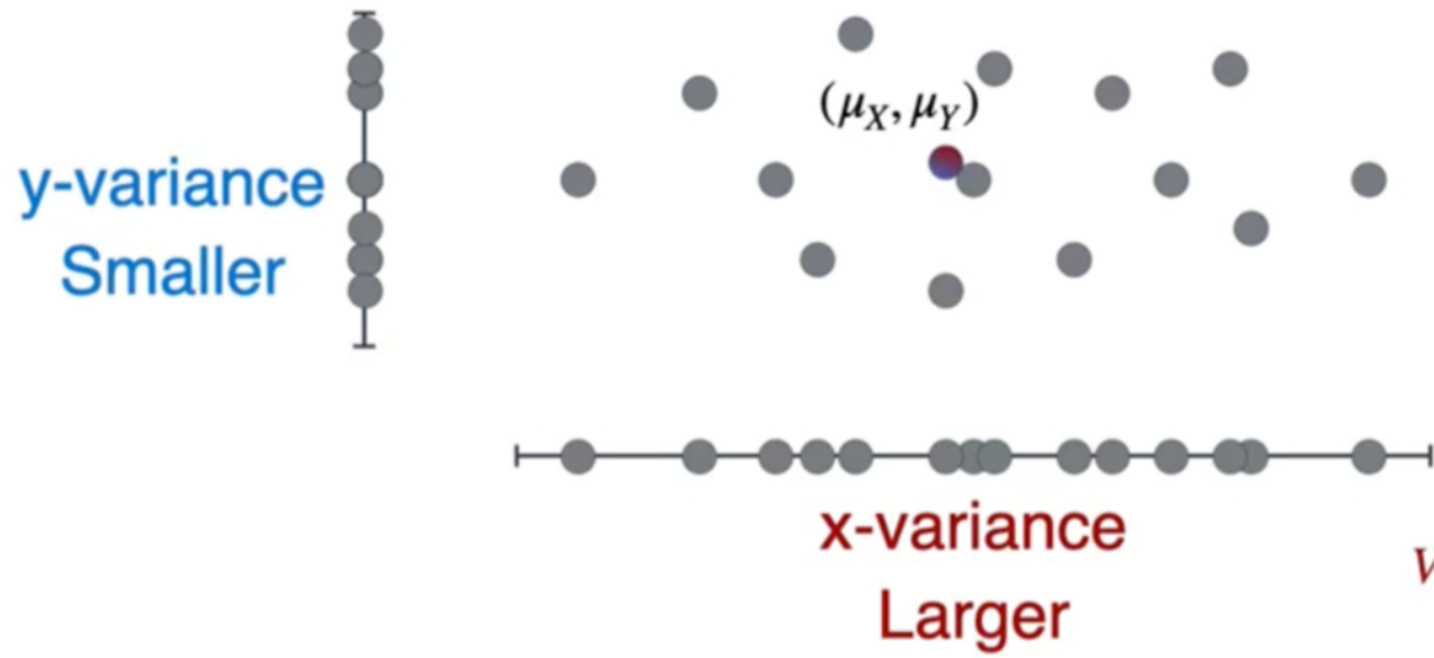
$$\text{Var}(x) = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \mu_x)^2$$

Average Squared distance from μ.

Variance

(Helps to quantify How Spread the data is).

$$Var(y) = \frac{1}{n-1}\sum_{i=1}^{n}(y_i - \mu_Y)^2$$



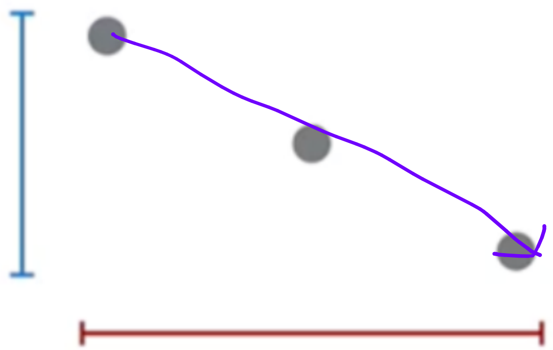
$$Var(x) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \mu_x)^2$$

DeepLearning.AI

Meanwhile, along the y-axis,

Problem

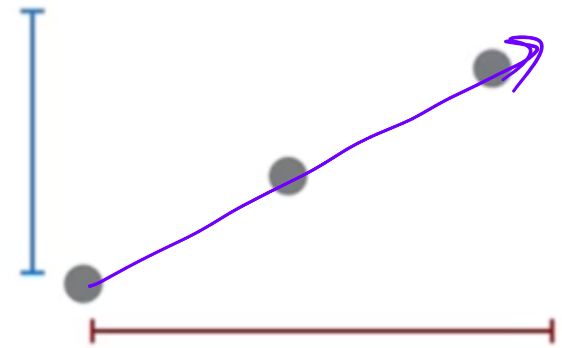
$$Cov(x,y) = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \mu_x)(y_i - \mu_y)$$



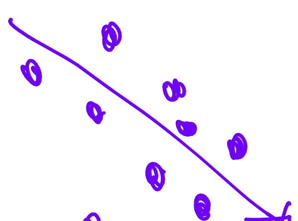
-ve
Cov

Solution: Covariance

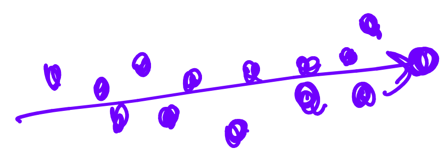
the direction of the relationship between two variables



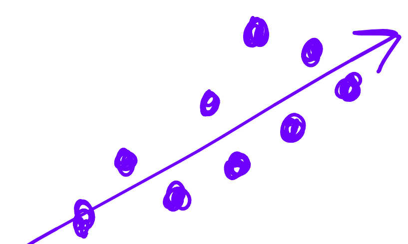
+ve (cov)



-ve
Covariance



Covariance



+ve
Covariance.

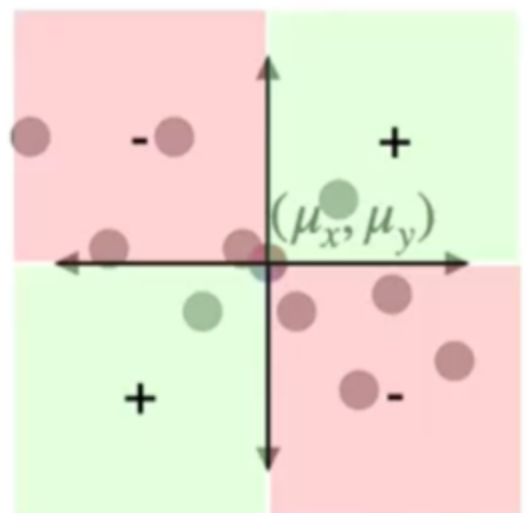
a dataset varies with
respect to one another.

DeepLearning.AI

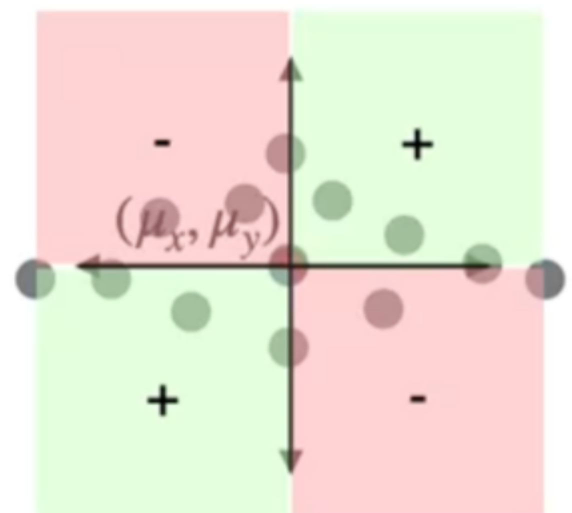
Covariance

"Take the average"

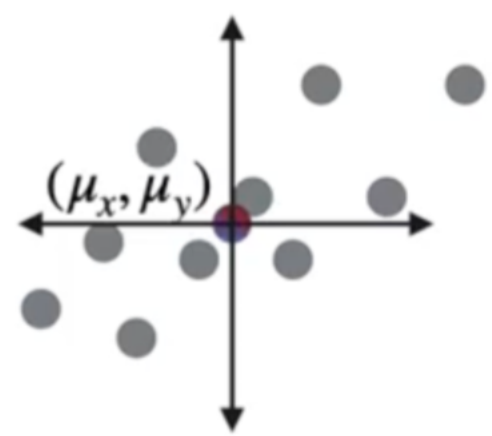
$$Cov(x, y) = \boxed{\frac{1}{n-1}\sum_{i=1}^{n}}(x_i - \mu_x)(y_i - \mu_y)$$



negative covariance



covariance zero (or very small)

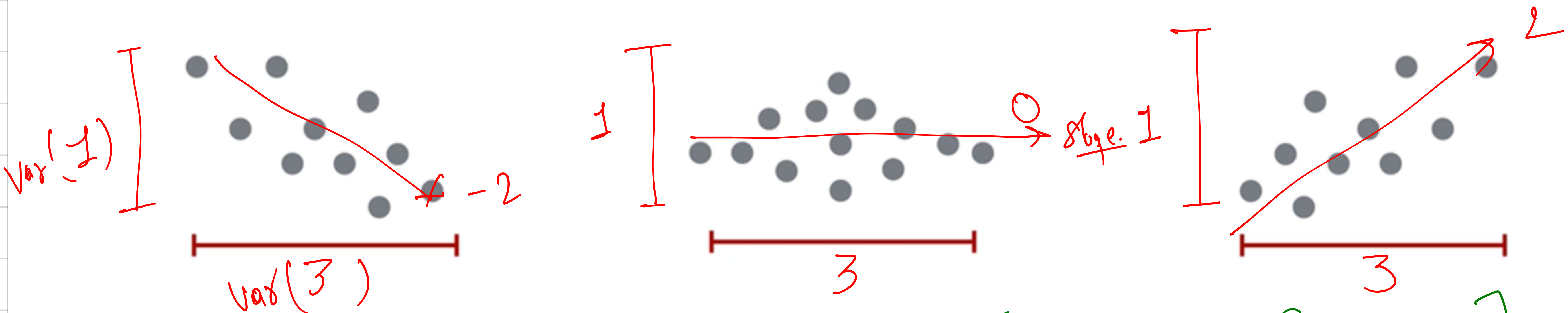


positive covariance

In the third, most of the dots

DeepLearning.AI

Covariance matrix

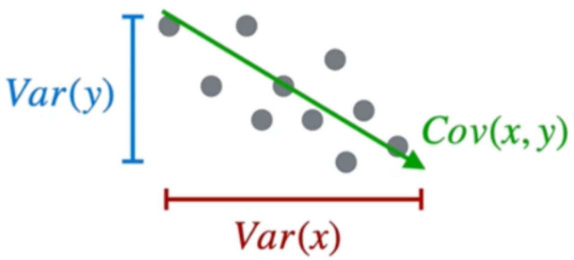


$$\begin{bmatrix} 3 & -2 \\ -2 & 1 \end{bmatrix} \qquad \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix} \qquad \begin{bmatrix} 3 & 2 \\ 2 & 1 \end{bmatrix}$$

the variance here
precisely so I'll

DeepLearning.AI

Covariance matrix



$$C = \begin{array}{c c} & \begin{array}{cc} x & y \end{array} \\ \begin{array}{c} x \\ y \end{array} & \begin{bmatrix} Cov(x,x) & Cov(x,y) \\ Cov(y,x) & Cov(y,y) \end{bmatrix} \end{array}$$

$$\left[\underline{Cov(x,x) = Var(x)} . \right]$$

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

Covariance matrix

$$= \frac{1}{n-1}(A-\mu)^T(A-\mu) = \frac{1}{n-1}\left(\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_n & y_n \end{bmatrix} - \begin{bmatrix} \mu_x & \mu_y \\ \mu_x & \mu_y \\ \vdots & \vdots \\ \mu_x & \mu_y \end{bmatrix}\right)^T \left(\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_n & y_n \end{bmatrix} - \begin{bmatrix} \mu_x & \mu_y \\ \mu_x & \mu_y \\ \vdots & \vdots \\ \mu_x & \mu_y \end{bmatrix}\right)$$

$$= \frac{1}{n-1}\begin{bmatrix} x_1-\mu_x & y_1-\mu_y \\ x_2-\mu_x & y_2-\mu_y \\ \vdots & \vdots \\ x_n-\mu_x & y_n-\mu_y \end{bmatrix}^T \begin{bmatrix} x_1-\mu_x & y_1-\mu_y \\ x_2-\mu_x & y_2-\mu_y \\ \vdots & \vdots \\ x_n-\mu_x & y_n-\mu_y \end{bmatrix}$$

$$= \frac{1}{n-1}\underbrace{\begin{bmatrix} x_1-\mu_x & x_2-\mu_x & \cdots & x_n-\mu_x \\ y_1-\mu_y & y_2-\mu_y & \cdots & y_n-\mu_y \end{bmatrix}}_{\boxed{2}\times n} \underbrace{\begin{bmatrix} x_1-\mu_x & y_1-\mu_y \\ x_2-\mu_x & y_2-\mu_y \\ \vdots & \vdots \\ x_n-\mu_x & y_n-\mu_y \end{bmatrix}}_{n\times\boxed{2}}$$

eepLearning.AI

iance matrix

$$A = \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_n & y_n \end{bmatrix} \qquad \mu = \begin{bmatrix} \mu_x & \mu_y \\ \mu_x & \mu_y \\ \vdots & \vdots \\ \mu_x & \mu_y \end{bmatrix}$$

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

using these two matrices,

ıg.AI

Covariance matrix

$$C = \frac{1}{n-1}(A-\mu)^T(A-\mu) = \frac{1}{n-1}\left(\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_n & y_n \end{bmatrix} - \begin{bmatrix} \mu_x & \mu_y \\ \mu_x & \mu_y \\ \vdots & \vdots \\ \mu_x & \mu_y \end{bmatrix}\right)^T \left(\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_n & y_n \end{bmatrix} - \begin{bmatrix} \mu_x & \mu_y \\ \mu_x & \mu_y \\ \vdots & \vdots \\ \mu_x & \mu_y \end{bmatrix}\right)$$

$$= \frac{1}{n-1}\begin{bmatrix} x_1-\mu_x & y_1-\mu_y \\ x_2-\mu_x & y_2-\mu_y \\ \vdots & \vdots \\ x_n-\mu_x & y_n-\mu_y \end{bmatrix}^T \begin{bmatrix} x_1-\mu_x & y_1-\mu_y \\ x_2-\mu_x & y_2-\mu_y \\ \vdots & \vdots \\ x_n-\mu_x & y_n-\mu_y \end{bmatrix}$$

$$= \frac{1}{n-1}\begin{bmatrix} x_1-\mu_x & x_2-\mu_x & \cdots & x_n-\mu_x \\ y_1-\mu_y & y_2-\mu_y & \cdots & y_n-\mu_y \end{bmatrix} \begin{bmatrix} x_1-\mu_x & y_1-\mu_y \\ x_2-\mu_x & y_2-\mu_y \\ \vdots & \vdots \\ x_n-\mu_x & y_n-\mu_y \end{bmatrix}$$

which is simply x_1

$$(x_1-\mu_x)(x_1-\mu_x)(x_2-\mu_x)(x_2-\mu_x) \;\cdots\cdots\; (x_n-\mu_x)(x_n-\mu_x)$$

Similar to

$$\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\mu_x)^2$$

$$\Rightarrow Var(x)$$

Covariance matrix

$$C = \frac{1}{n-1}(A-\mu)^T(A-\mu) = \frac{1}{n-1}\left(\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_n & y_n \end{bmatrix} - \begin{bmatrix} \mu_x & \mu_y \\ \mu_x & \mu_y \\ \vdots & \vdots \\ \mu_x & \mu_y \end{bmatrix}\right)^T \left(\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_n & y_n \end{bmatrix} - \begin{bmatrix} \mu_x & \mu_y \\ \mu_x & \mu_y \\ \vdots & \vdots \\ \mu_x & \mu_y \end{bmatrix}\right)$$

$$= \frac{1}{n-1}\begin{bmatrix} x_1-\mu_x & y_1-\mu_y \\ x_2-\mu_x & y_2-\mu_y \\ \vdots & \vdots \\ x_n-\mu_x & y_n-\mu_y \end{bmatrix}^T \begin{bmatrix} x_1-\mu_x & y_1-\mu_y \\ x_2-\mu_x & y_2-\mu_y \\ \vdots & \vdots \\ x_n-\mu_x & y_n-\mu_y \end{bmatrix}$$

$$= \frac{1}{n-1}\begin{bmatrix} x_1-\mu_x & x_2-\mu_x & \cdots & x_n-\mu_x \\ y_1-\mu_y & y_2-\mu_y & \cdots & y_n-\mu_y \end{bmatrix} \begin{bmatrix} x_1-\mu_x & y_1-\mu_y \\ x_2-\mu_x & y_2-\mu_y \\ \vdots & \vdots \\ x_n-\mu_x & y_n-\mu_y \end{bmatrix} = \begin{bmatrix} Var(x) & Cov(x,y) \\ Cov(y,x) & Var(y) \end{bmatrix}$$

$(x_1-\mu_x)(y_1 \ldots (y_n-\mu_y)$

Mu_x*y_1-y+x_2 minus

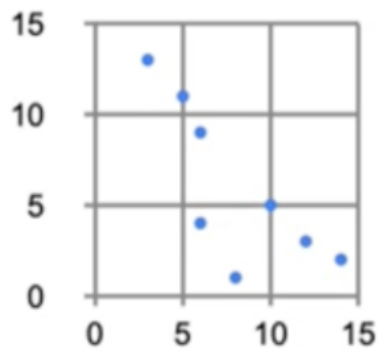
Mu_x*y_2 minus Mu_y etc

So, final form is

$$\Rightarrow \begin{bmatrix} Var(x) & Cov(x,y) \\ Cov(y,x) & Var(y) \end{bmatrix}$$

Matrix formula

$$A - \mu = \begin{bmatrix} x_1 - \mu_x & y_1 - \mu_y \\ \vdots & \vdots \\ x_n - \mu_x & y_n - \mu_y \end{bmatrix} \qquad C = \frac{1}{n-1}(A-\mu)^T(A-\mu)$$



Similar X and Y variances
Negative covariance

A

x	y
10	5
12	3
6	9
6	4
5	11
14	2
8	1
3	13

$\mu_x = 8 \quad \mu_y = 6$

$A - \mu$

$x - \mu_x$	$y - \mu_y$
2	-1
4	-3
-2	3
-2	-2
-3	5
6	-4
0	-5
-5	8

$\frac{1}{7}$

$(A - \mu)^T$

$x - \mu_x$	2	4	-2	-2	-3	6	0	-5
$y - \mu_y$	-1	-3	3	-2	5	-4	-5	8

Because $\frac{1}{n-1} = \frac{1}{7}$

$A - \mu$

$x - \mu_x$	$y - \mu_y$
2	-1
4	-3
-2	3
-2	-2
-3	5
6	-4
0	-5
-5	8

$$C = \begin{bmatrix} 14 & -11.86 \\ -11.86 & 19.71 \end{bmatrix}$$

which is C the covariance matrix

DeepLearning.AI

To exit full screen, press ⎋.

Matrix formula

$$A = \begin{bmatrix} x_1 & y_1 & z_1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & z_n \end{bmatrix} \qquad C = \frac{1}{n-1}(A-\mu)^T(A-\mu)$$

1. Arrange data with a different feature in each column
2. Calculate column averages
3. Subtract each average from their respective column to generate $A - \mu$
4. $\frac{1}{n-1}(A-\mu)^T(A-\mu)$ gives the covariance matrix C

All these steps gives C.

more detail in the third
course of this specialization.

DeepLearning.AI

Discrete Dynamical Systems:

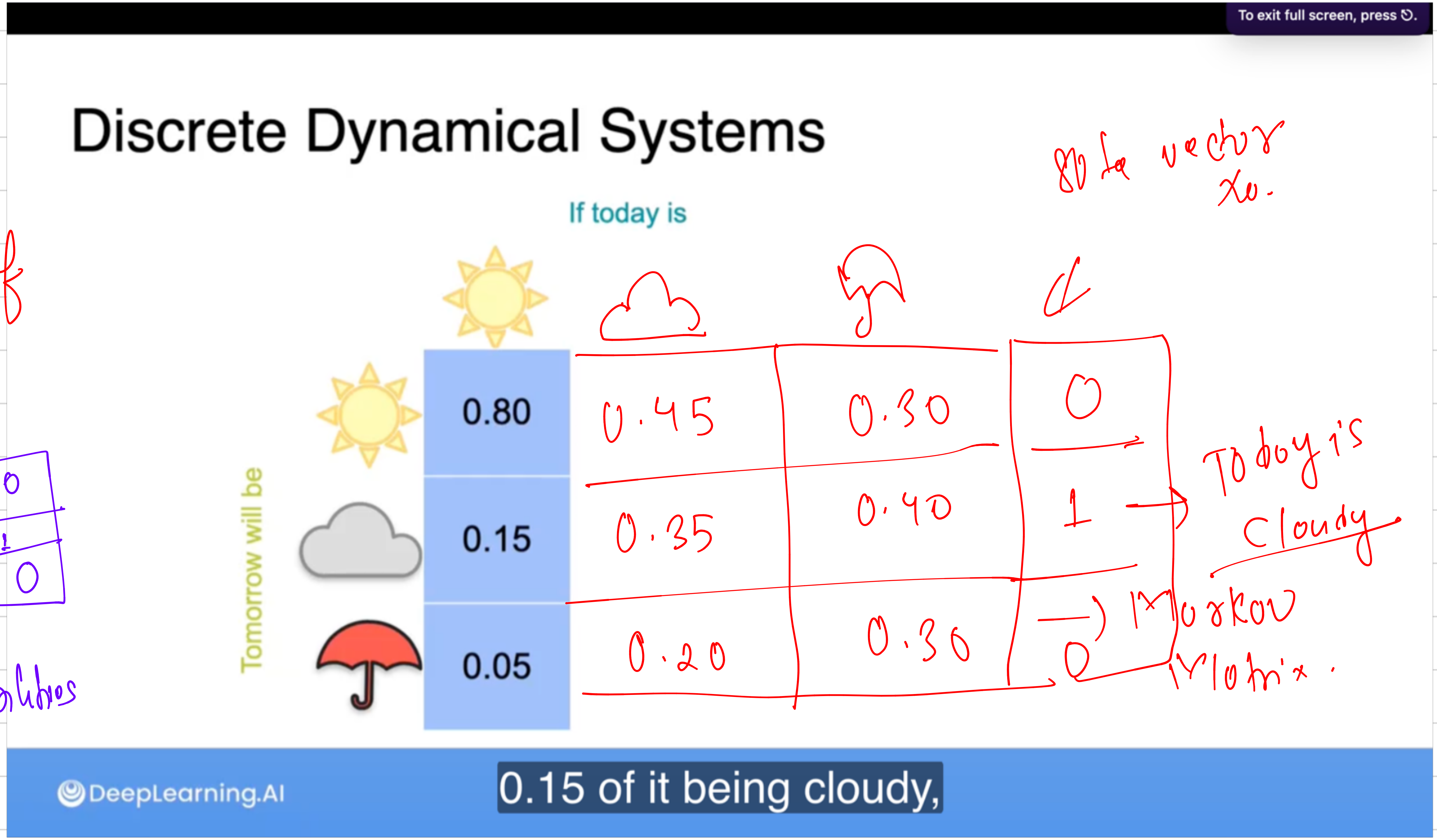
Tommorow ?

Take Dot product of

→ Matrix

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

gives the probabilities of tommorow's weather.



Day After tomorrow $X_2, X_3 \dots$?

$X_n = ?$

Same as dot product between previous state (present) and current state.

Discrete Dynamical Systems

1 day ahead X_1

X_2 ?

If today is | State vector X_0 | Tomorrow will be X_1

Tomorrow will be

				State vector X_0		Tomorrow will be X_1
	0.80	0.45	0.30	0		0.45
	0.15	0.35	0.40	1	=	0.35
	0.05	0.20	0.30	0		0.2

0.45, 0.35, and 0.2.

Discrete Dynamical Systems

Eigenvector

If today is | 11 days ahead

Tomorrow will be

				X_{10}		X_{11}
	0.80	0.45	0.30	0.6665		0.6665
	0.15	0.35	0.40	0.2223	= 1.	0.2223
	0.05	0.20	0.30	0.1112		0.1112

Eigenvalue

Essentially, you are saying that

Stabilize.

DeepLearning.AI

Discrete Dynamical Systems

If today is

Tomorrow will be

	☀	☁	☂	Eigenvector X_{10}		Eigenvalue X_{11}
☀	0.80	0.45	0.30	0.6665		0.6665
☁	0.15	0.35	0.40	0.2223	$= 1 \cdot$	0.2223
☂	0.05	0.20	0.30	0.1112		0.1112

it's an eigenvector

Discrete Dynamical Systems

Transition matrix P			Equilibrium vector X_∞	
0.80	0.45	0.30	0.6665	☀
0.15	0.35	0.40	0.2223	☁
0.05	0.20	0.30	0.1112	☂

Long run probabilities

a go with the
programming assignment.

At ∞ states are become stabilize using Markov Model.

Source: Linear Algebra for Machine Learning and Data Science Coursera.

Unsupervised Learning Algorithm:

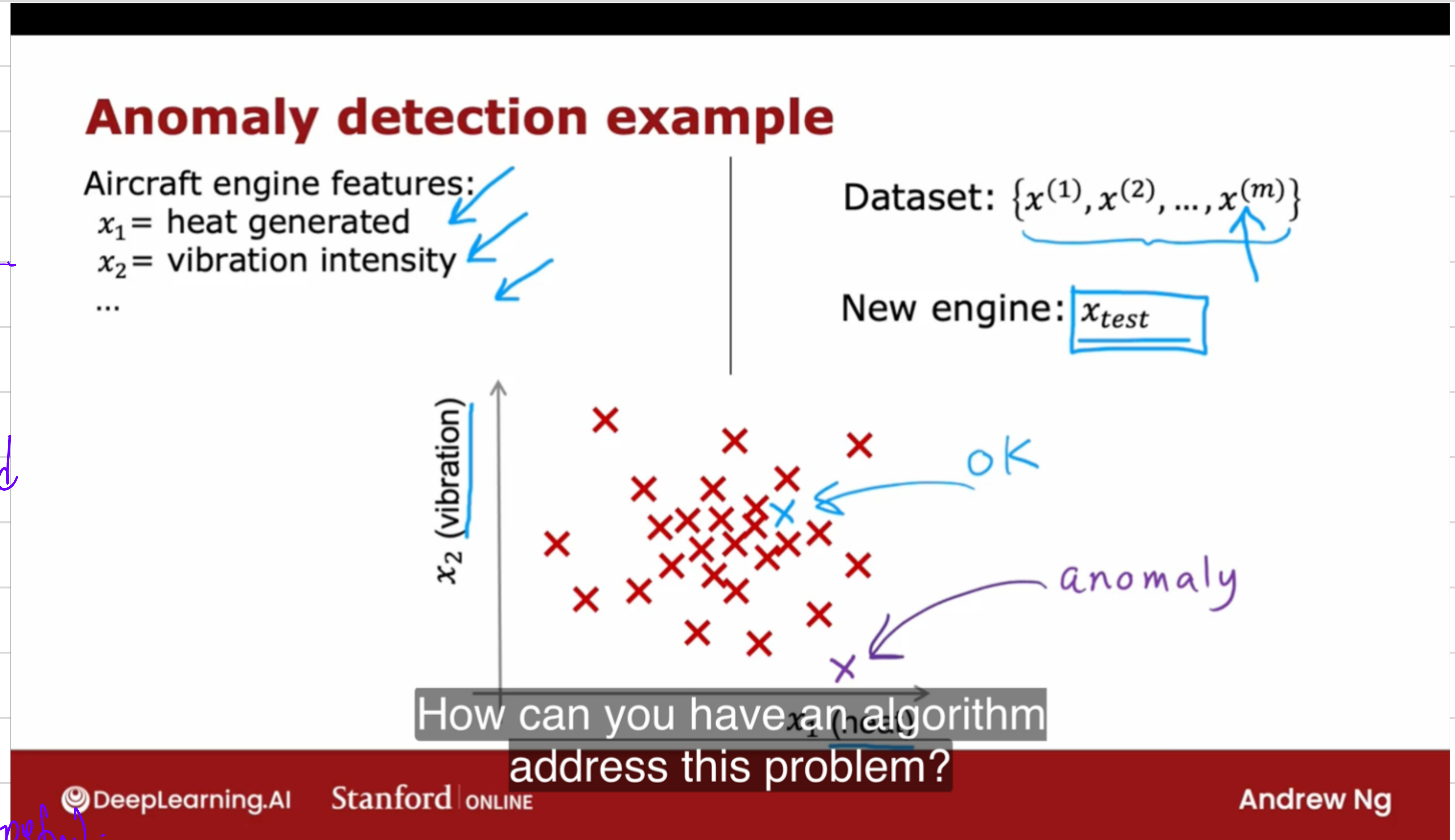
Anomaly Detection for Aircraft Engines.

↳ Using Unsupervised ML

↳ learns from past data and

new engine data X test

if similar → Normal
else Anomaly (needs Inspection).



Goal: Prevent potential failures by flagging outliers for further checks.

$p(x)$ estimates the Probability of observing a given data points.

Model $p(x)$

Density estimation

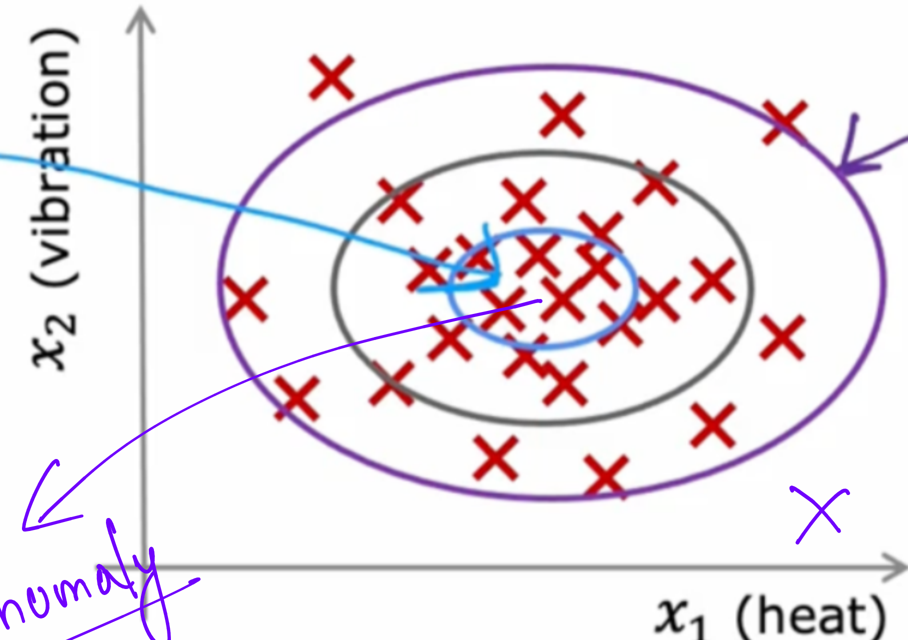
Dataset: $\{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}$ probability of x being seen in dataset

Model $p(x)$

Is x_{test} anomalous?

high probability

$p(x_{test}) \geq \varepsilon$



Okay No Anomaly

low probability

$p(x_{test}) < \varepsilon$

epsilon : Small number

Flag as Anomaly

Unlikely. anomaly

DeepLearning.AI Stanford ONLINE Andrew Ng

Applications of Anomoly Detection (ML):

Anomaly detection example

how often log in?
how many web pages visited?
transactions?
posts? typing speed?

Fraud detection:

- $x^{(i)}$ = features of user i's activities
- Model $p(x)$ from data.
- Identify unusual users by checking which have $p(x) < \varepsilon$

perform additional checks to identify real fraud vs. false alarms

Manufacturing:
$x^{(i)}$ = features of product i

airplane engine
circuit board
smartphone

Monitoring computers in a data center:
$x^{(i)}$ = features of machine i

- x_1 = memory use,
- x_2 = number of disk accesses/sec,
- x_3 = CPU load,
- x_4 = CPU load/network traffic.

ratios

DeepLearning.AI Stanford ONLINE Andrew Ng