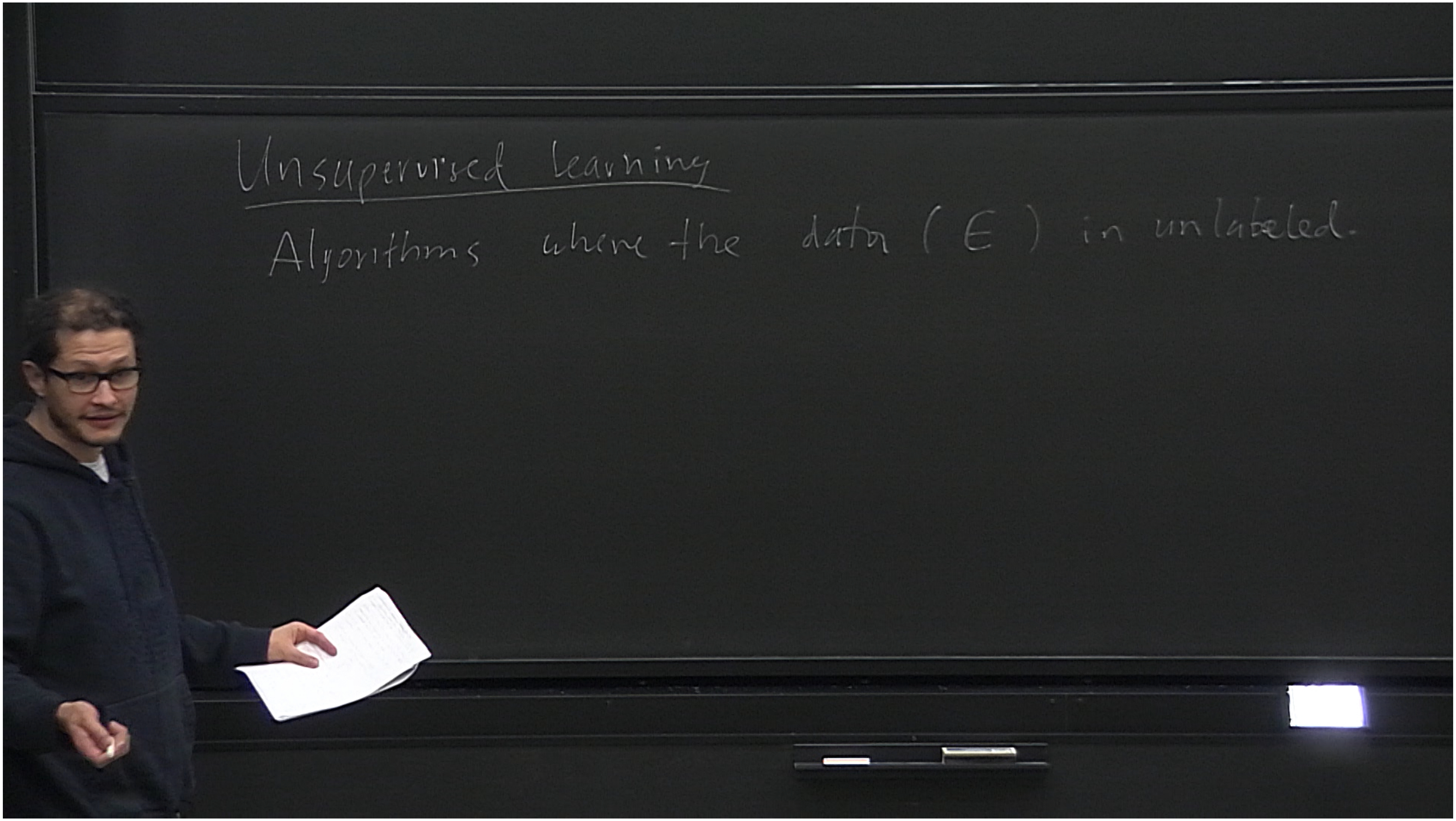
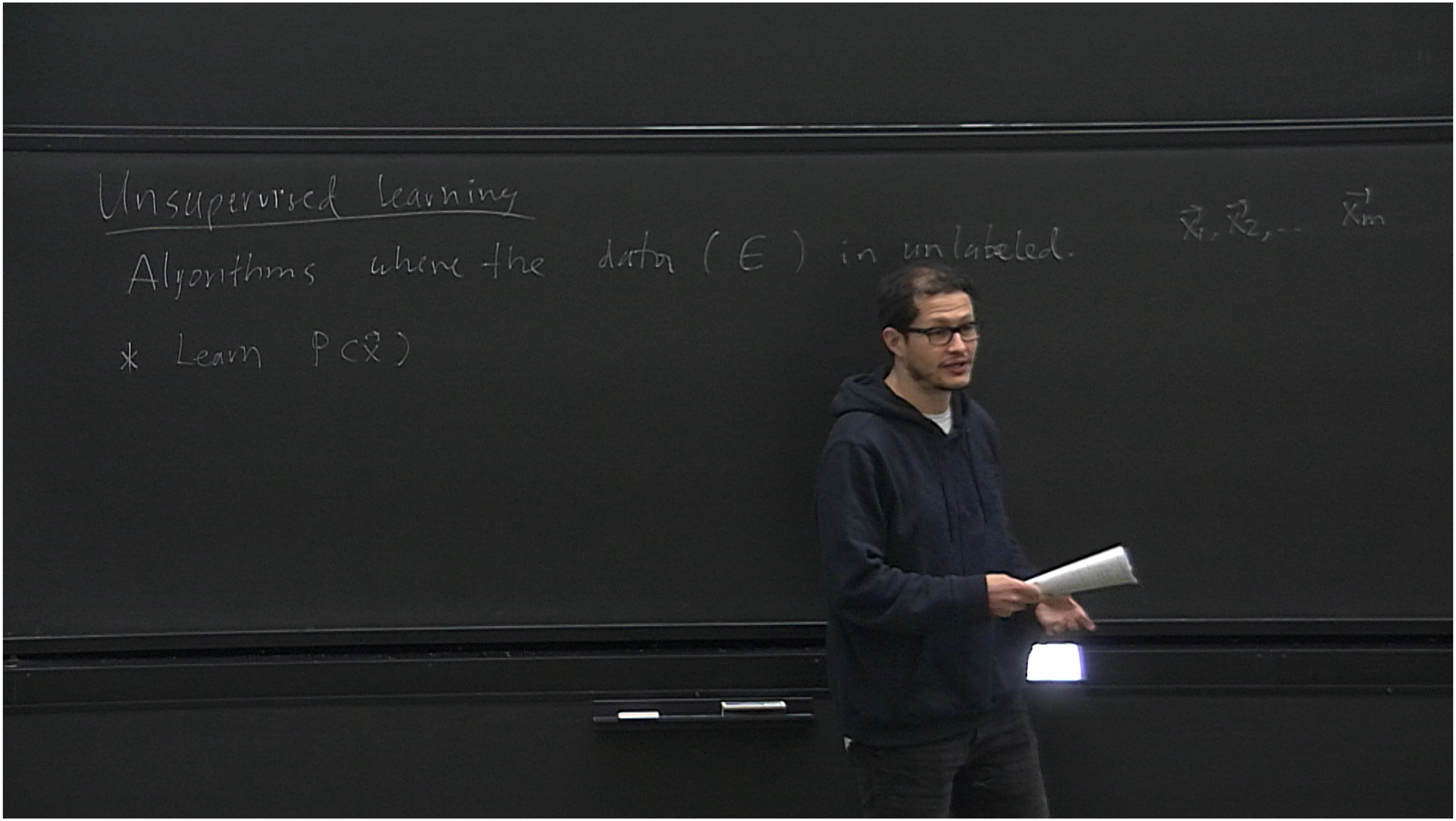
Title: PSI 2017/2018 - Machine Learning for Many Body Physics - Lecture 7

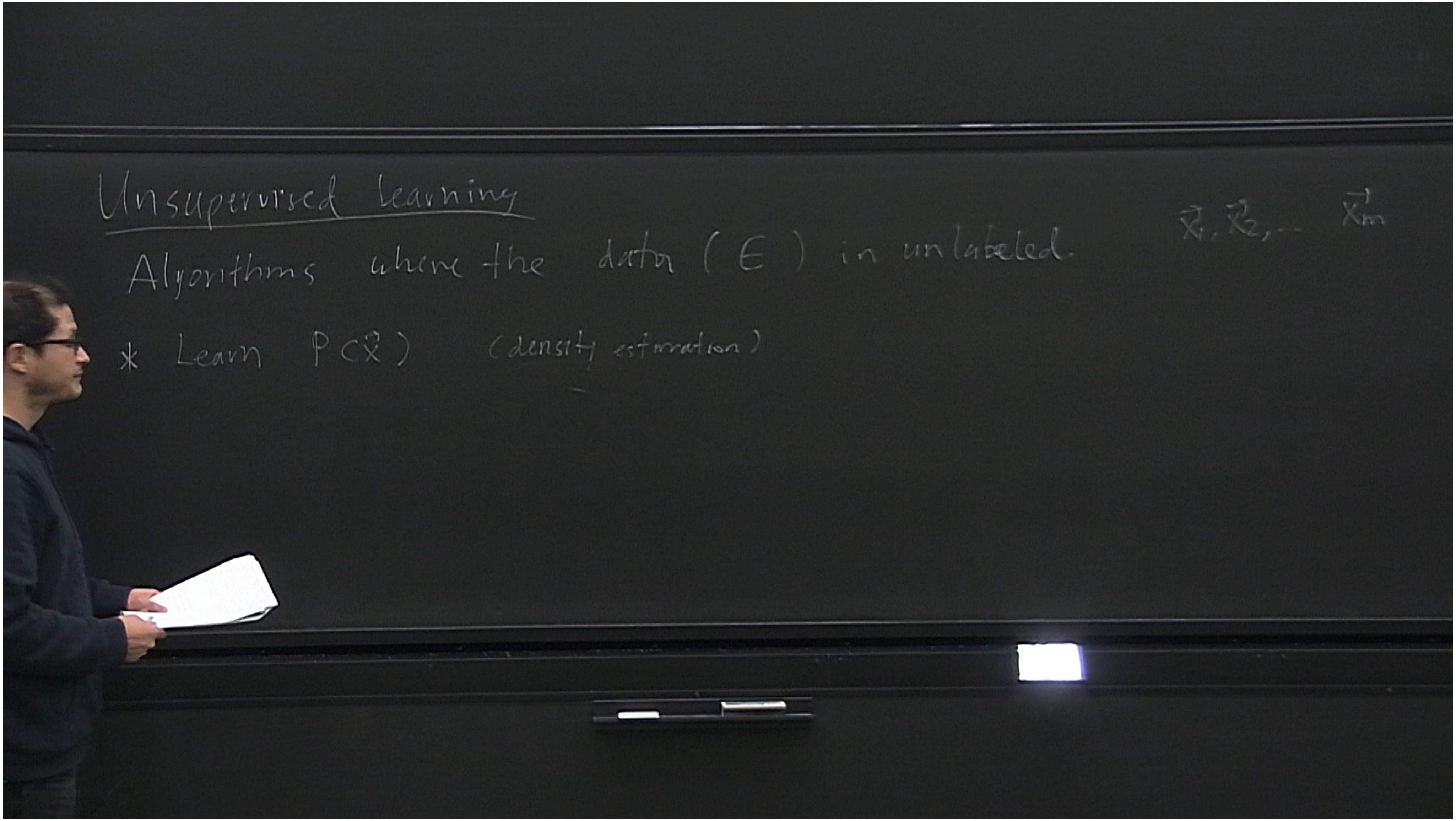
Date: Apr 17, 2018 09:00 AM

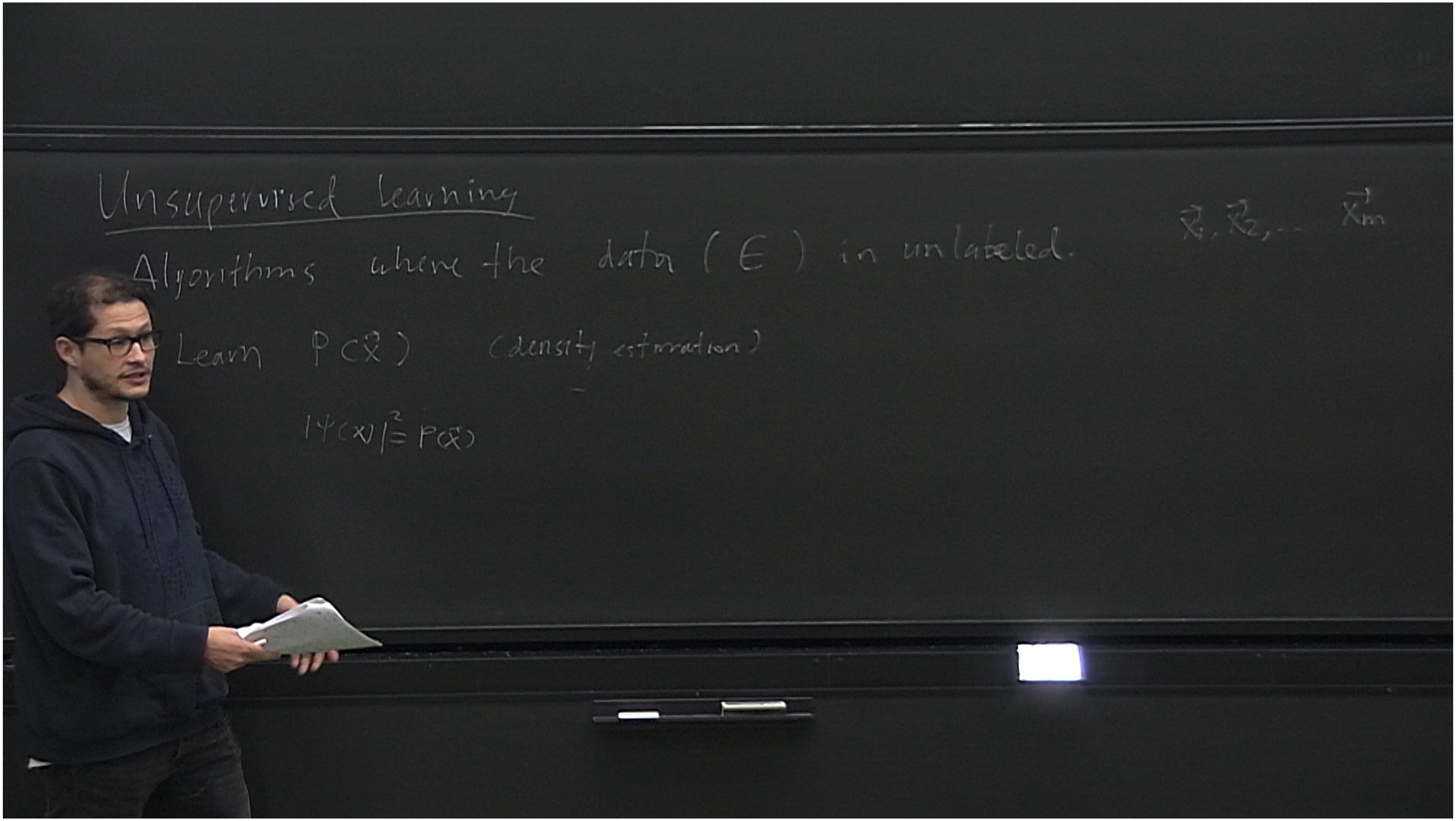
URL: http://pirsa.org/18040060

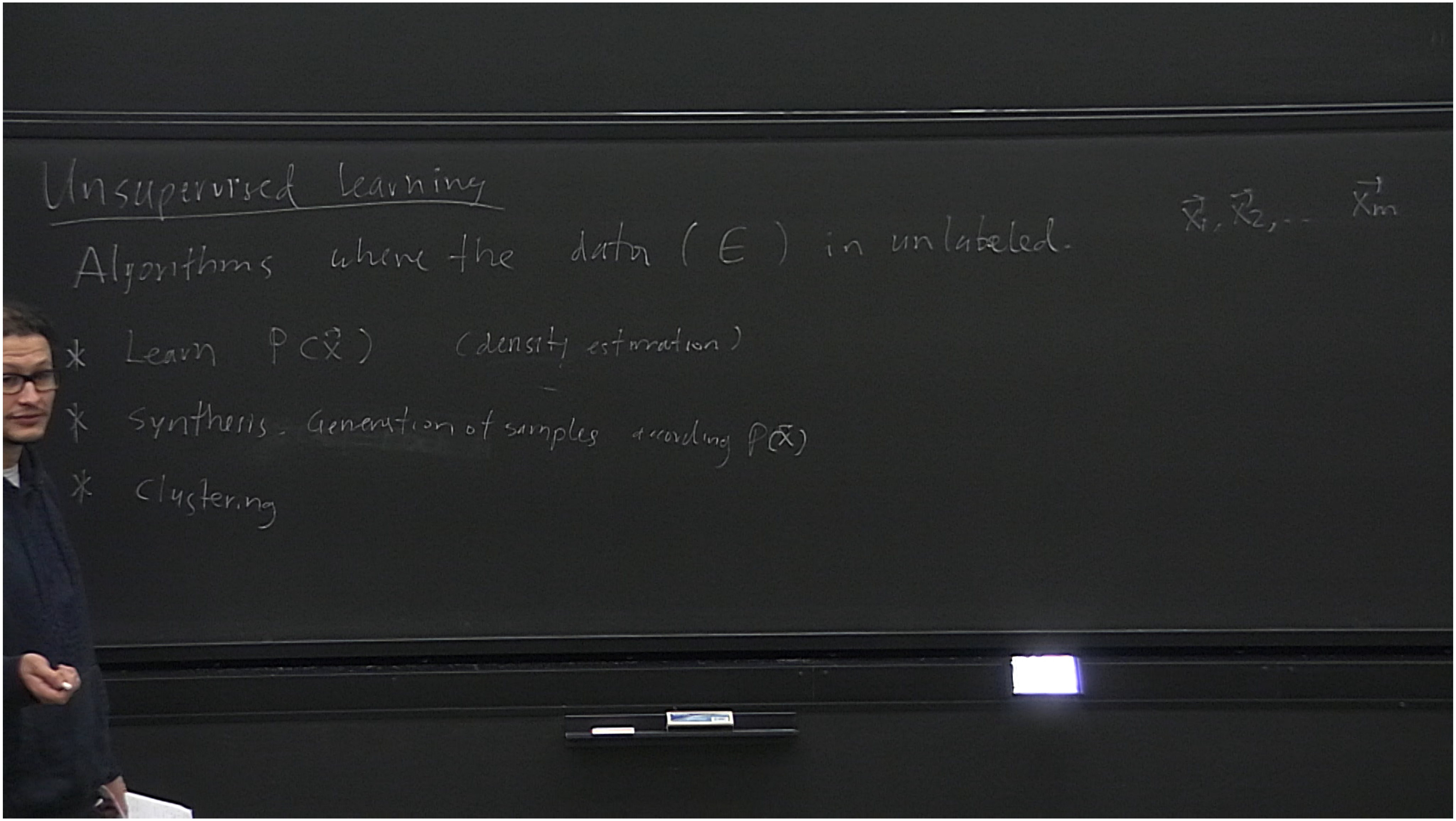
Abstract:

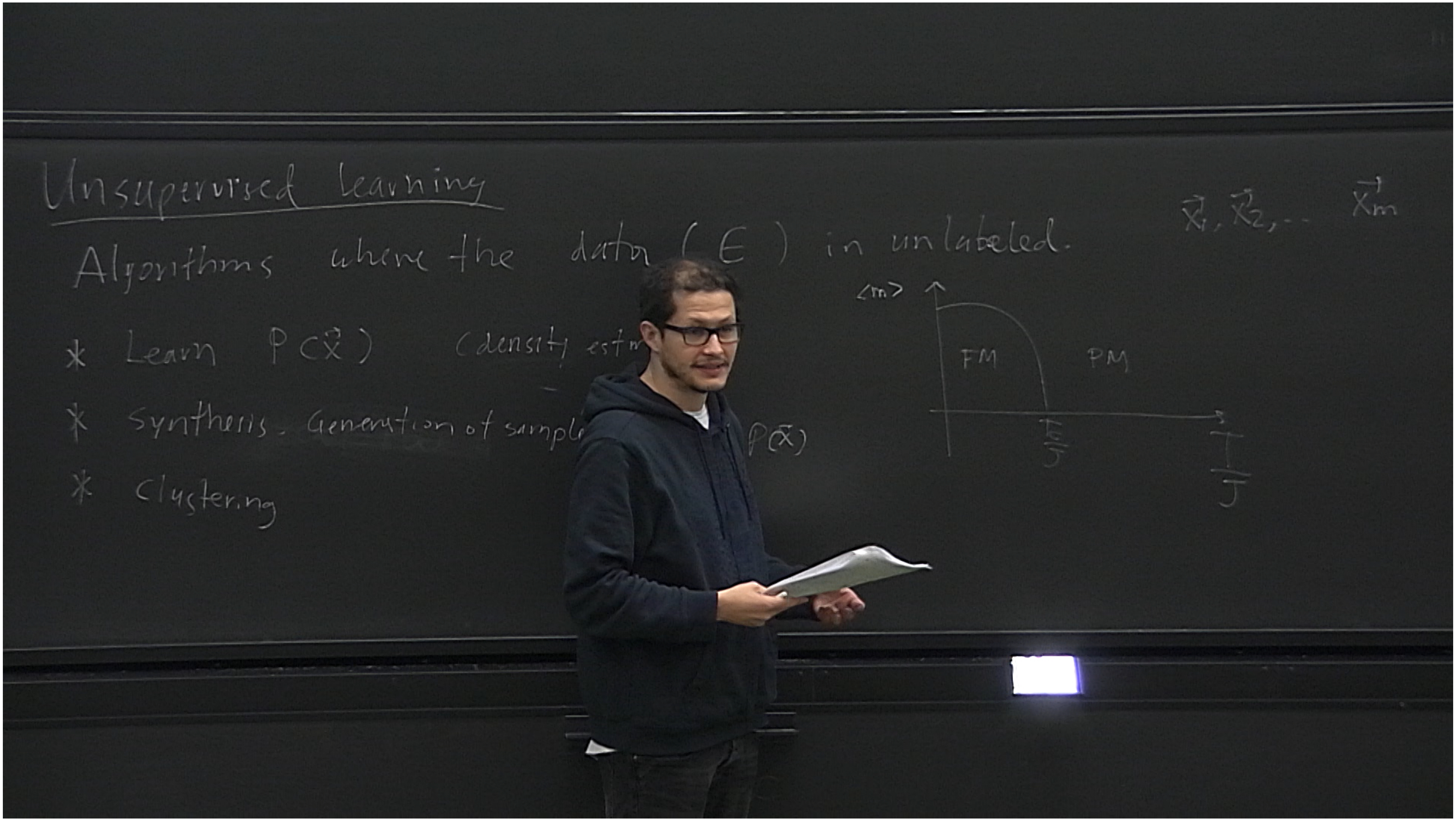

Unsupervised learning
Algorithms where the data (∈) in unlabeled.

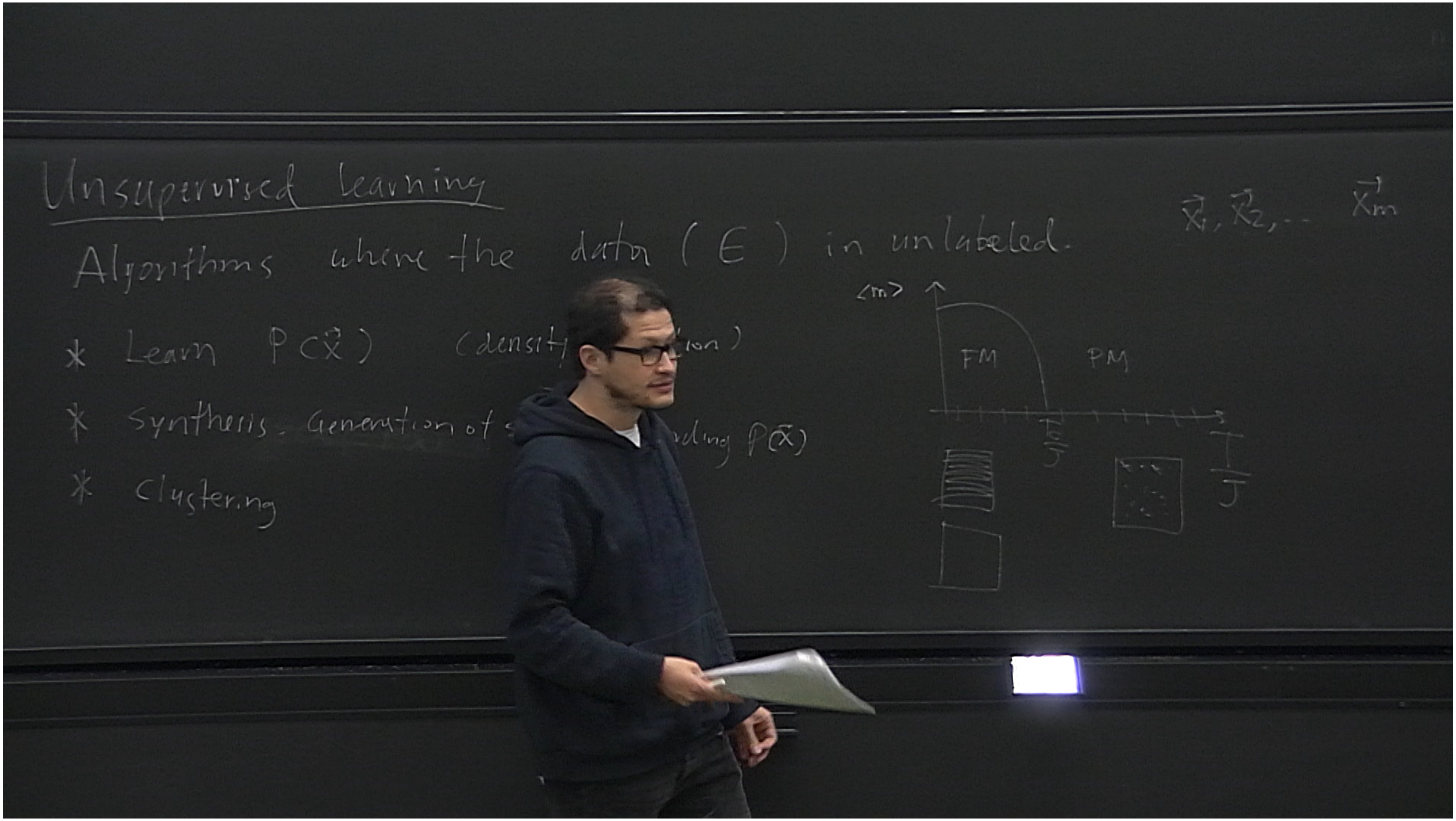

Unsupervised learning
Algorithms where the data (E) in unlabeled.
$\vec{x}_1, \vec{x}_2, \ldots \vec{x}_m$
* Learn $P(\vec{x})$

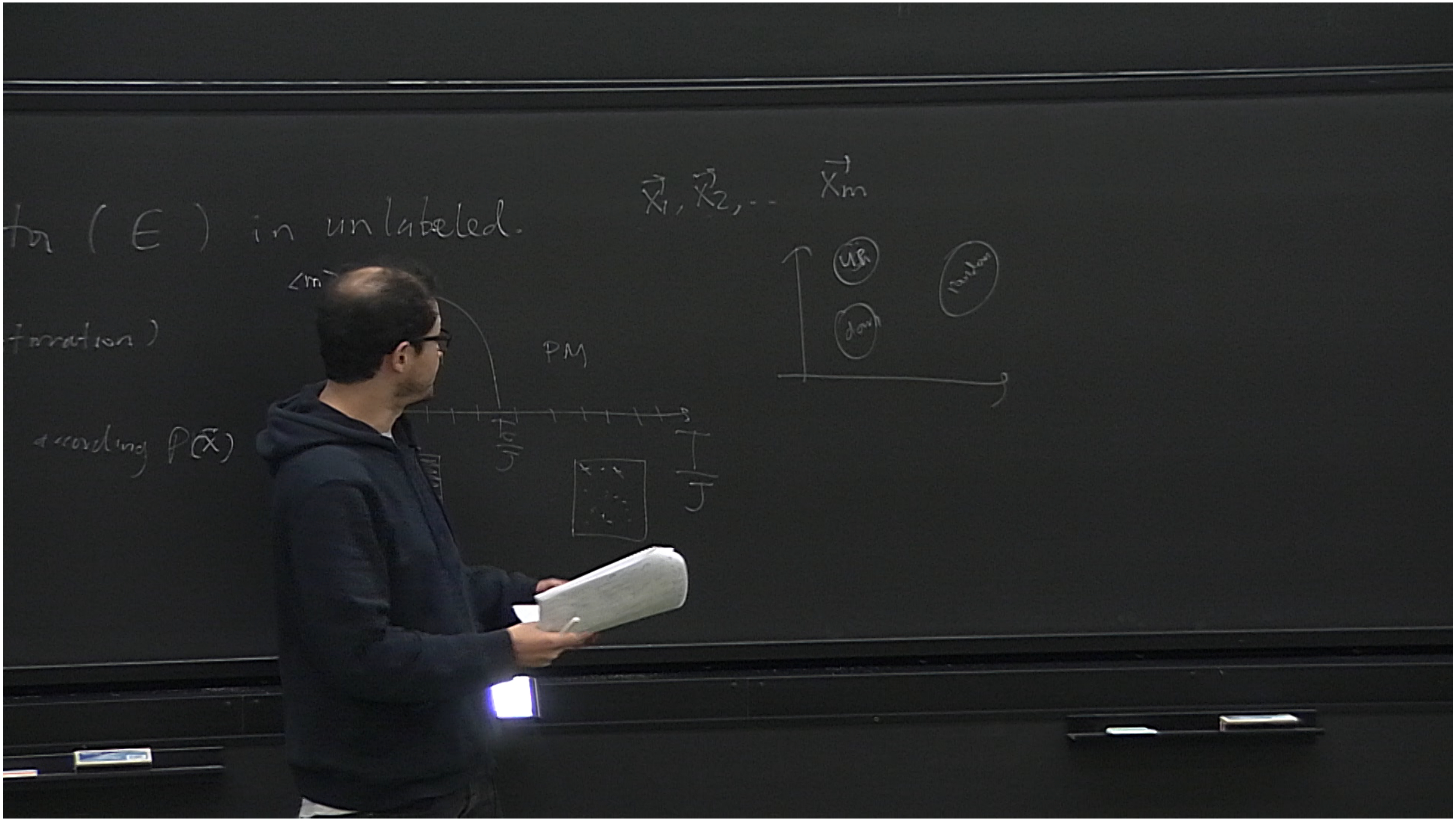

Unsupervised learning
Algorithms where the data (E) in unlabeled.
$\vec{x}_1, \vec{x}_2, \ldots \vec{x}_m$
* Learn P($\vec{x}$) (density estimation)


Unsupervised learning
Algorithms where the data (E) in unlabeled.
$\vec{x}_1, \vec{x}_2, \ldots \vec{x}_m$
Learn $P(\vec{x})$ (density estimation)
$|\psi(x)|^2 = P(\vec{x})$

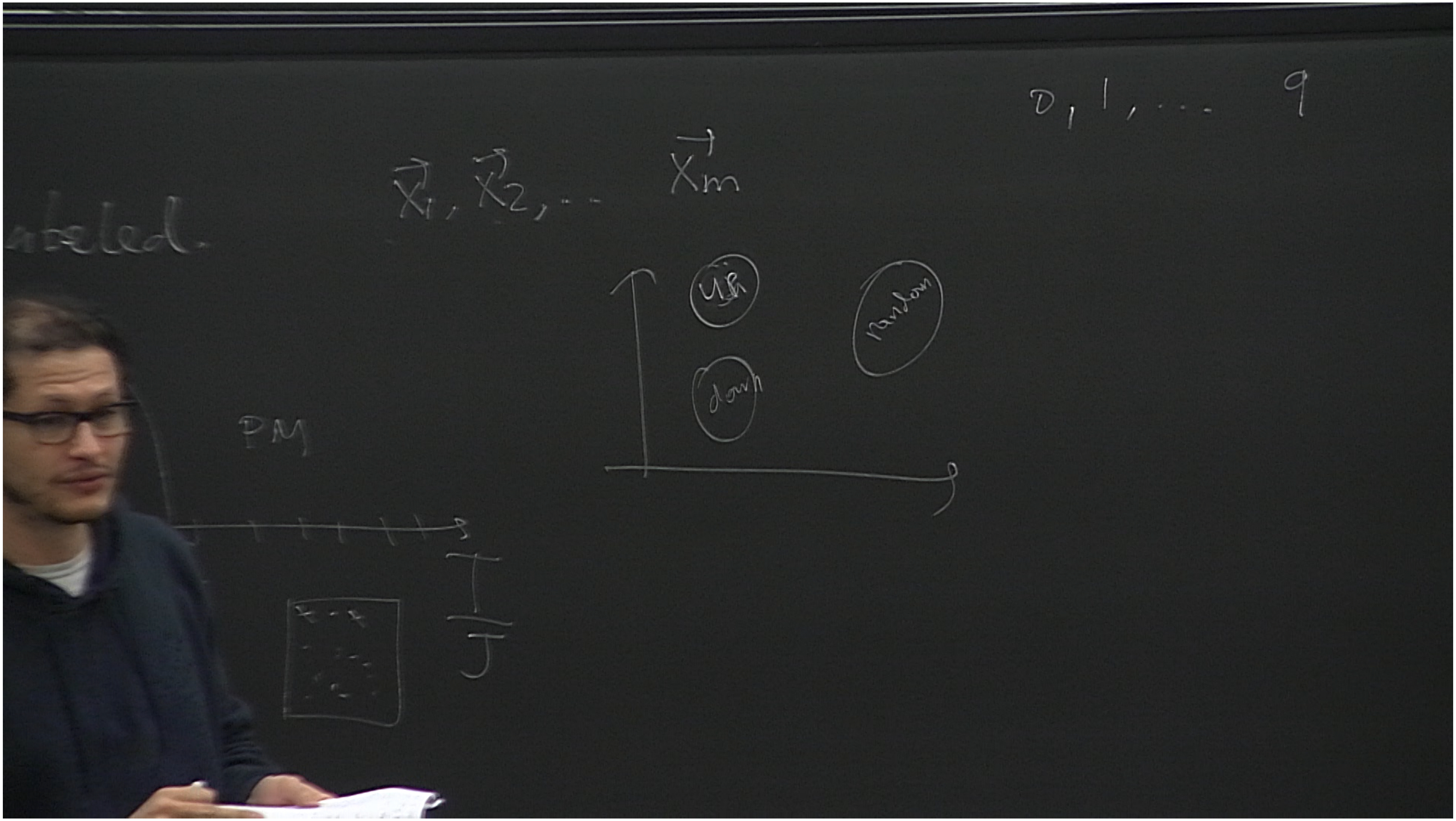

Unsupervised learning
$\vec{x}_1, \vec{x}_2, \ldots \vec{x}_m$
Algorithms where the data (∈) in unlabeled.
* Learn $P(\vec{x})$ (density estimation)
* Synthesis. Generation of samples according $P(\vec{x})$
* Clustering

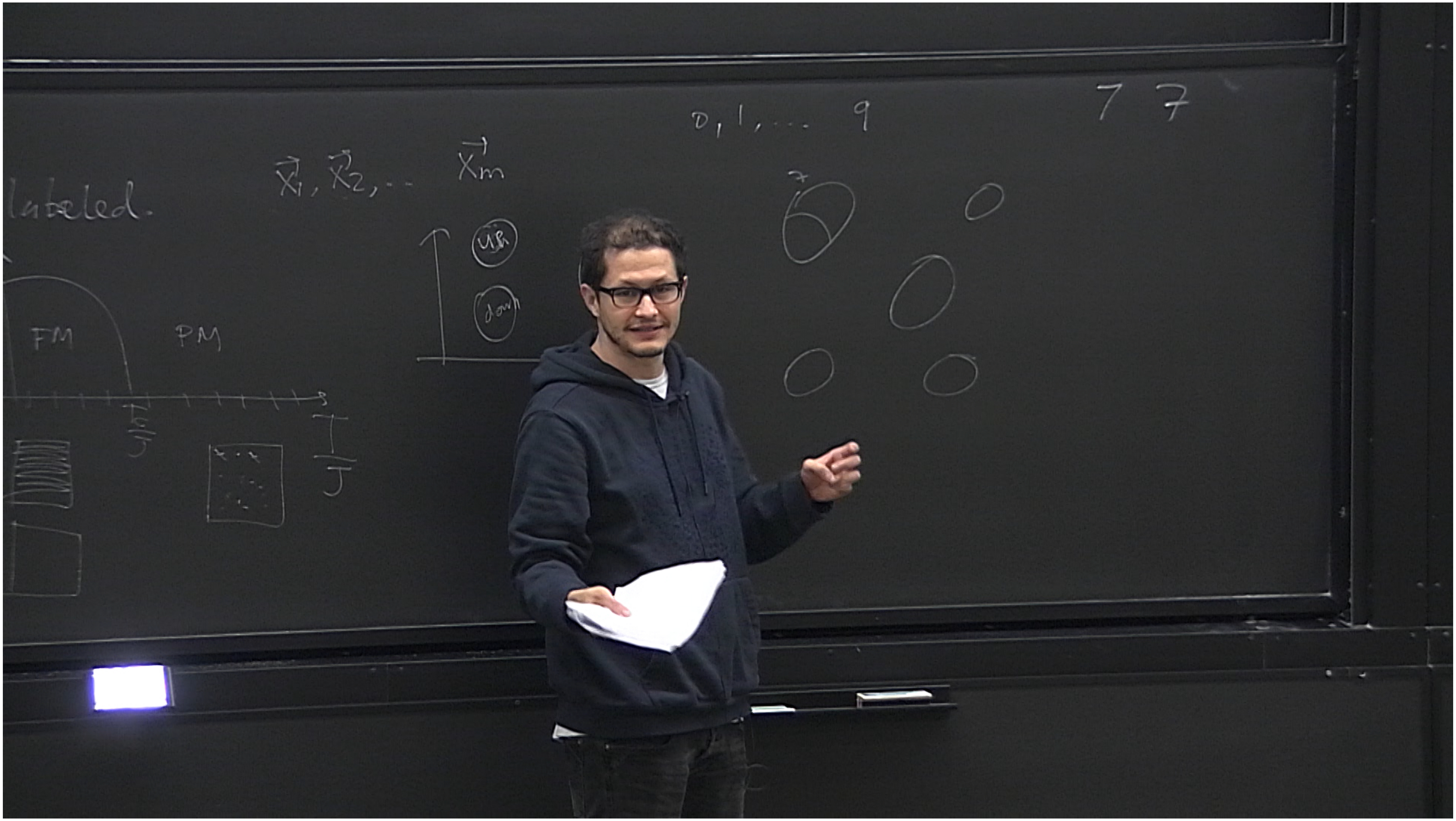

Unsupervised learning
Algorithms where the data (E) in unlabeled.
$\vec{x}_1, \vec{x}_2, \ldots \vec{x}_m$
* Learn $P(\vec{x})$
(density
* Synthesis. Generation of
$P(\vec{x})$
* Clustering
$\langle m \rangle$
FM
PM


Unsupervised learning
Algorithms where the data (∈) in unlabeled.
* Learn P(x)
* Synthesis. Generation of
* Clustering
FM
PM
T/J

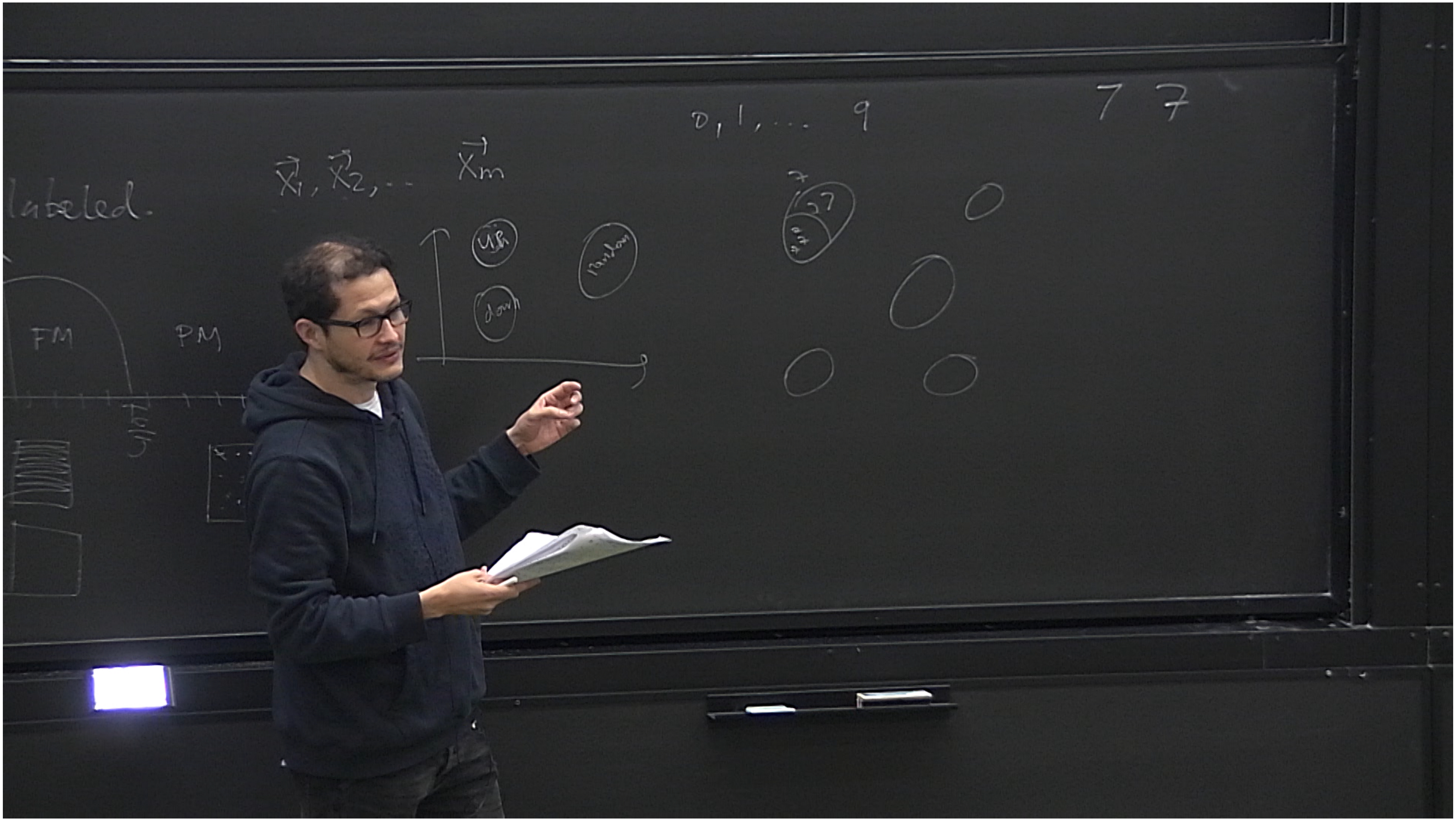


tor (E) in unlabeled.
formation)
according P(x)
PM
x1, x2, ... xm
up
down
random

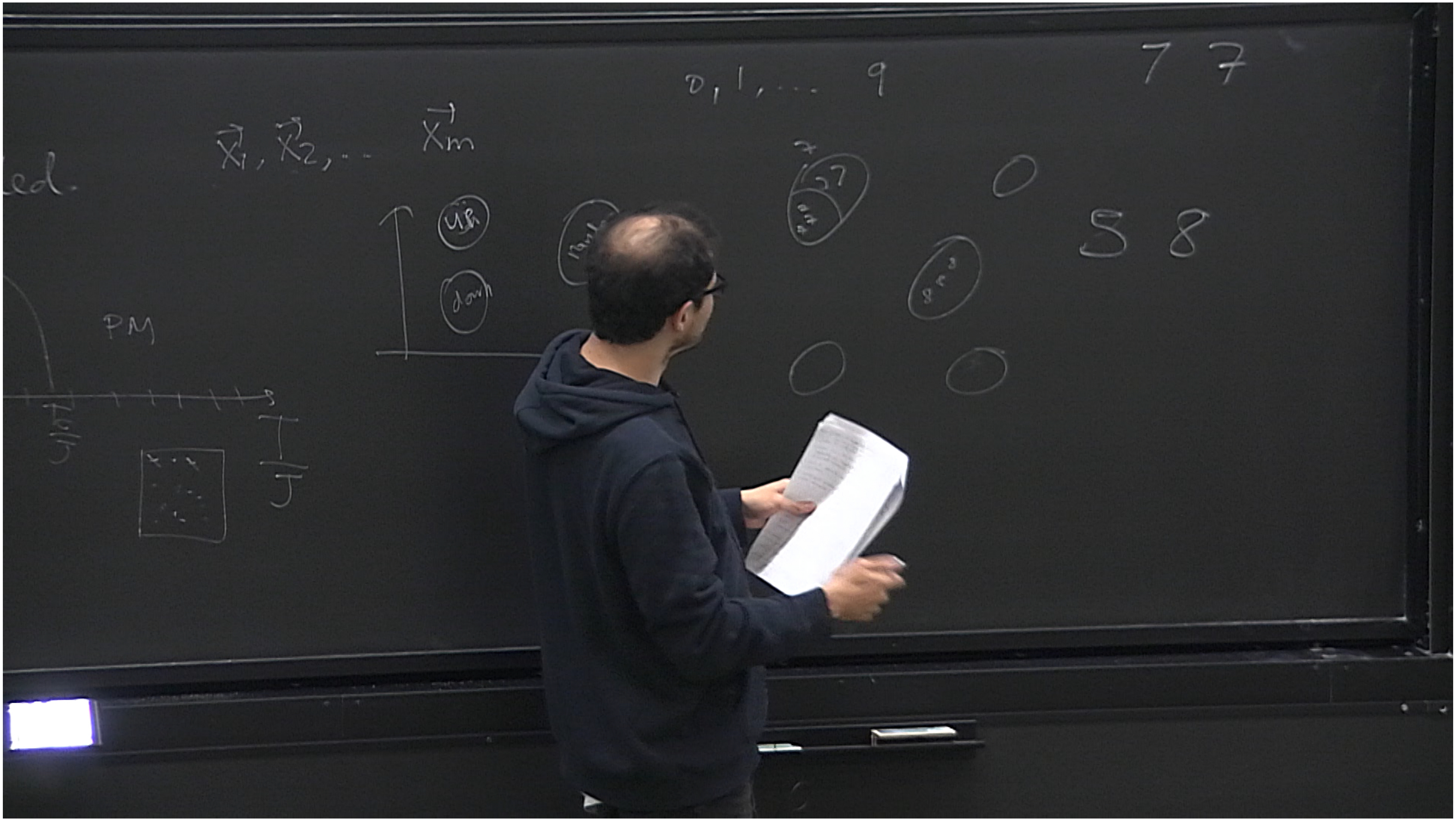

0, 1, ... 9
$\vec{X}_1, \vec{X}_2, \dots \vec{X}_m$
abeled.
up
down
random
PM
J



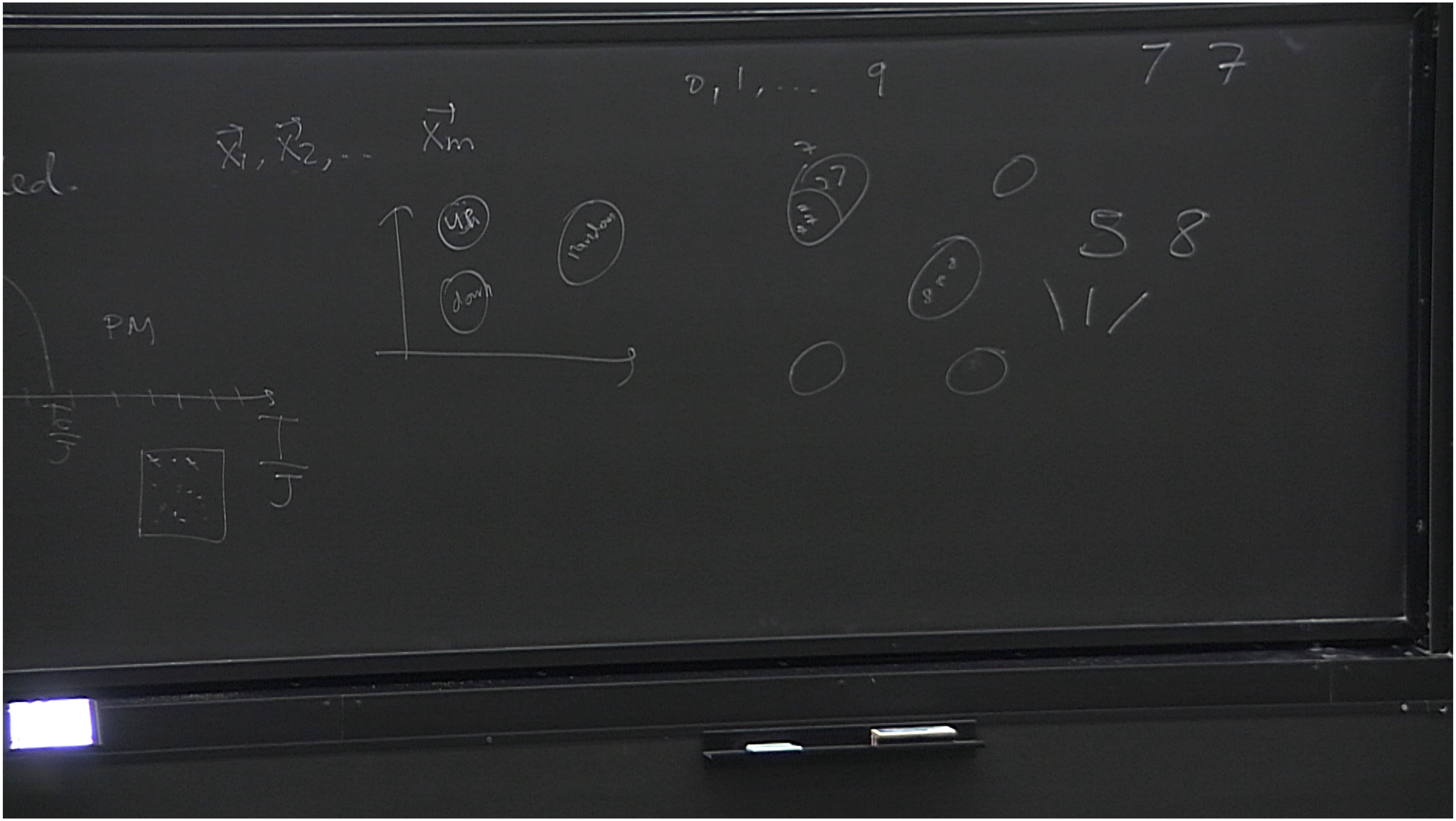
labeled.
FM
PM
0, 1, ... 9
7 7
up
down

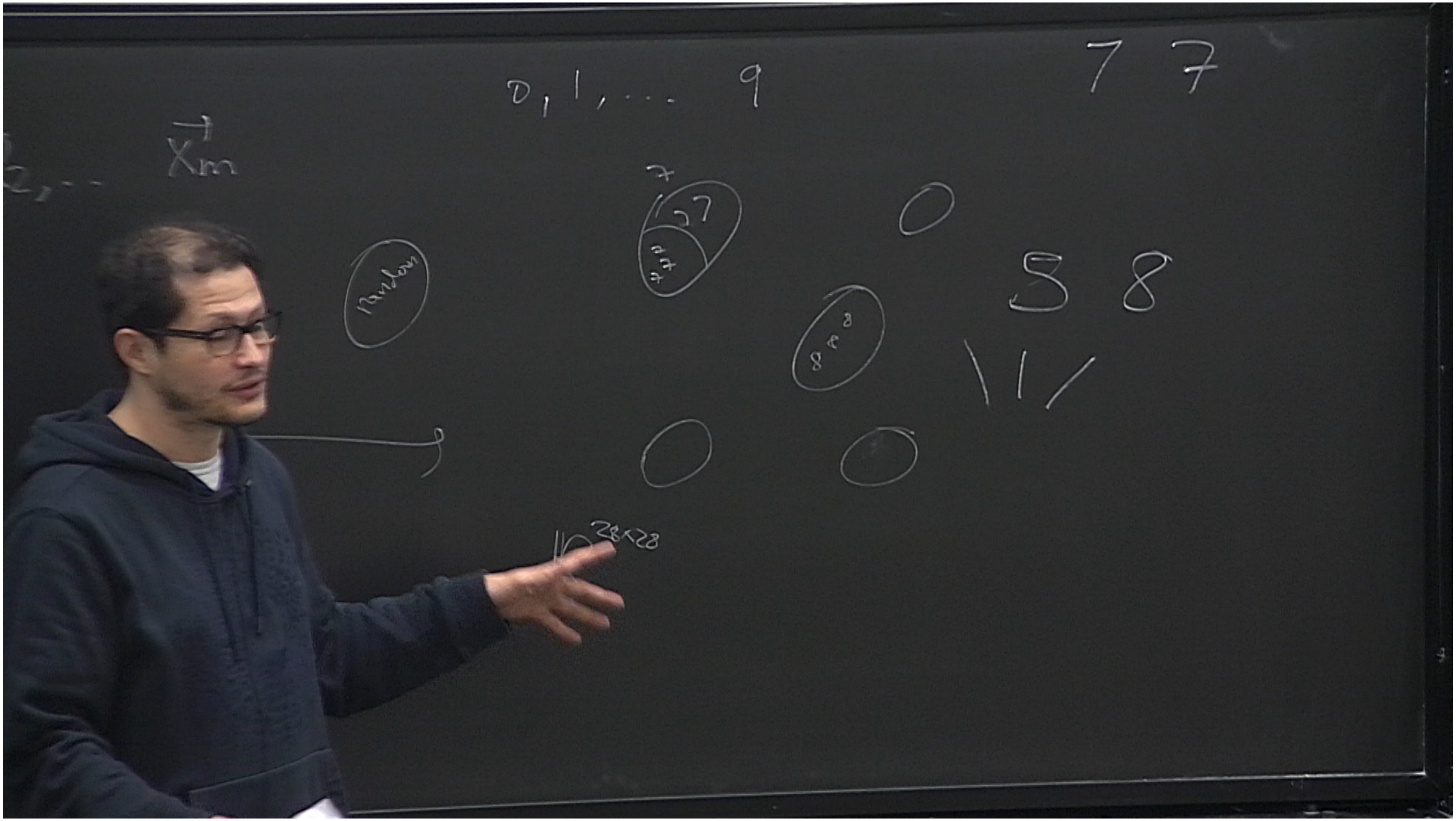


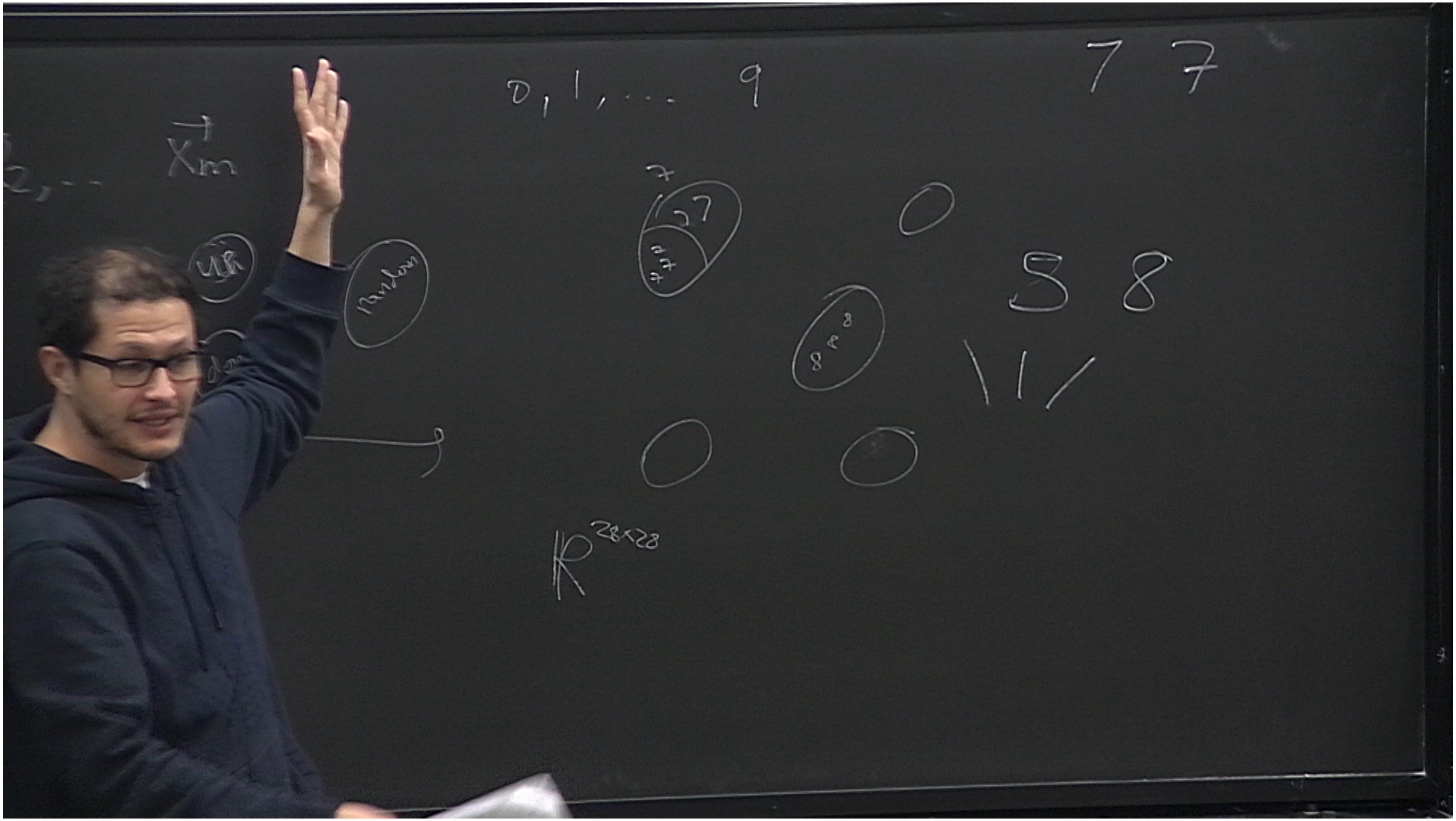
labeled.
X1, X2, ... Xm
0, 1, ... 9
7 7
up
down
random
FM
PM

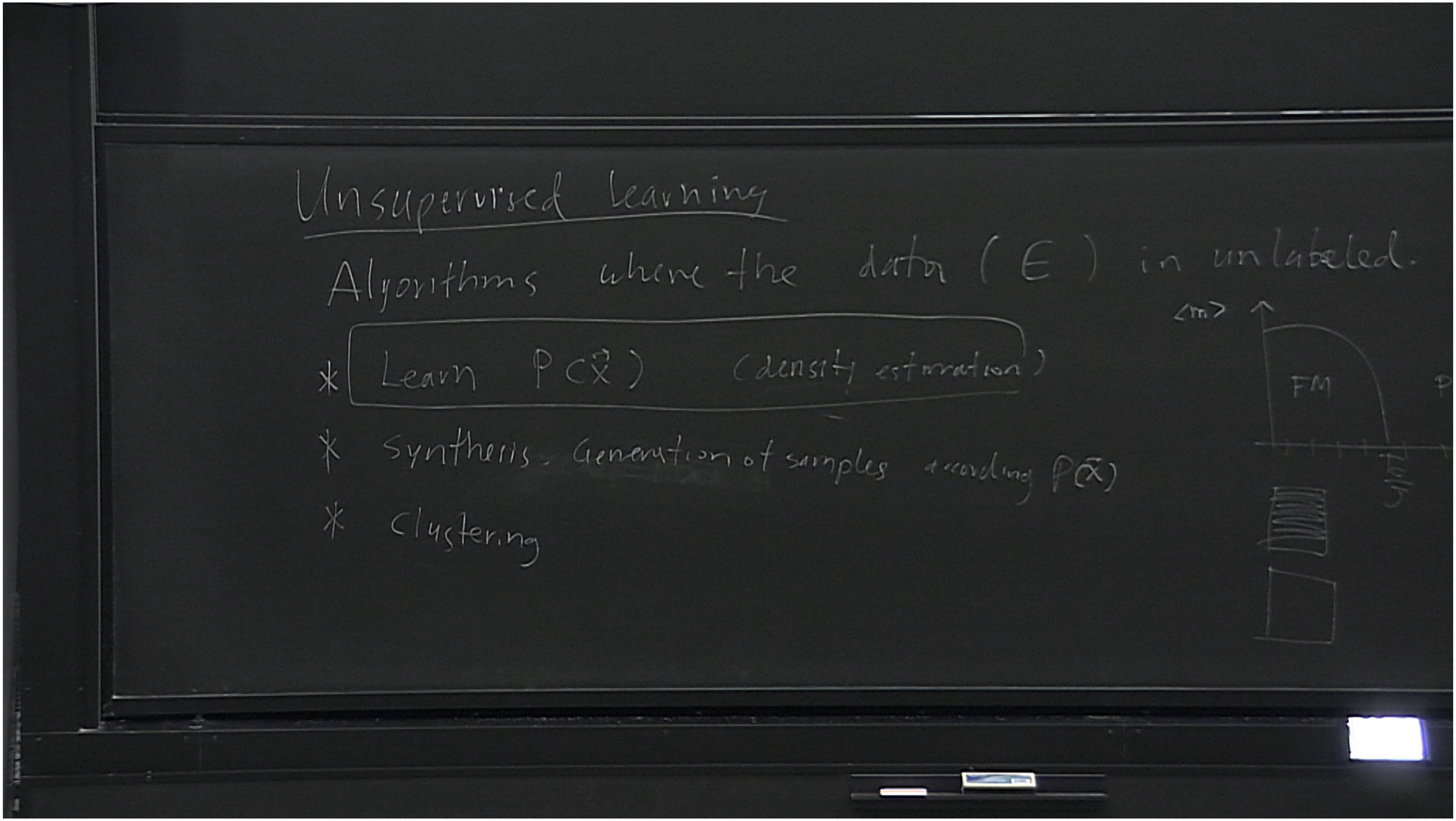


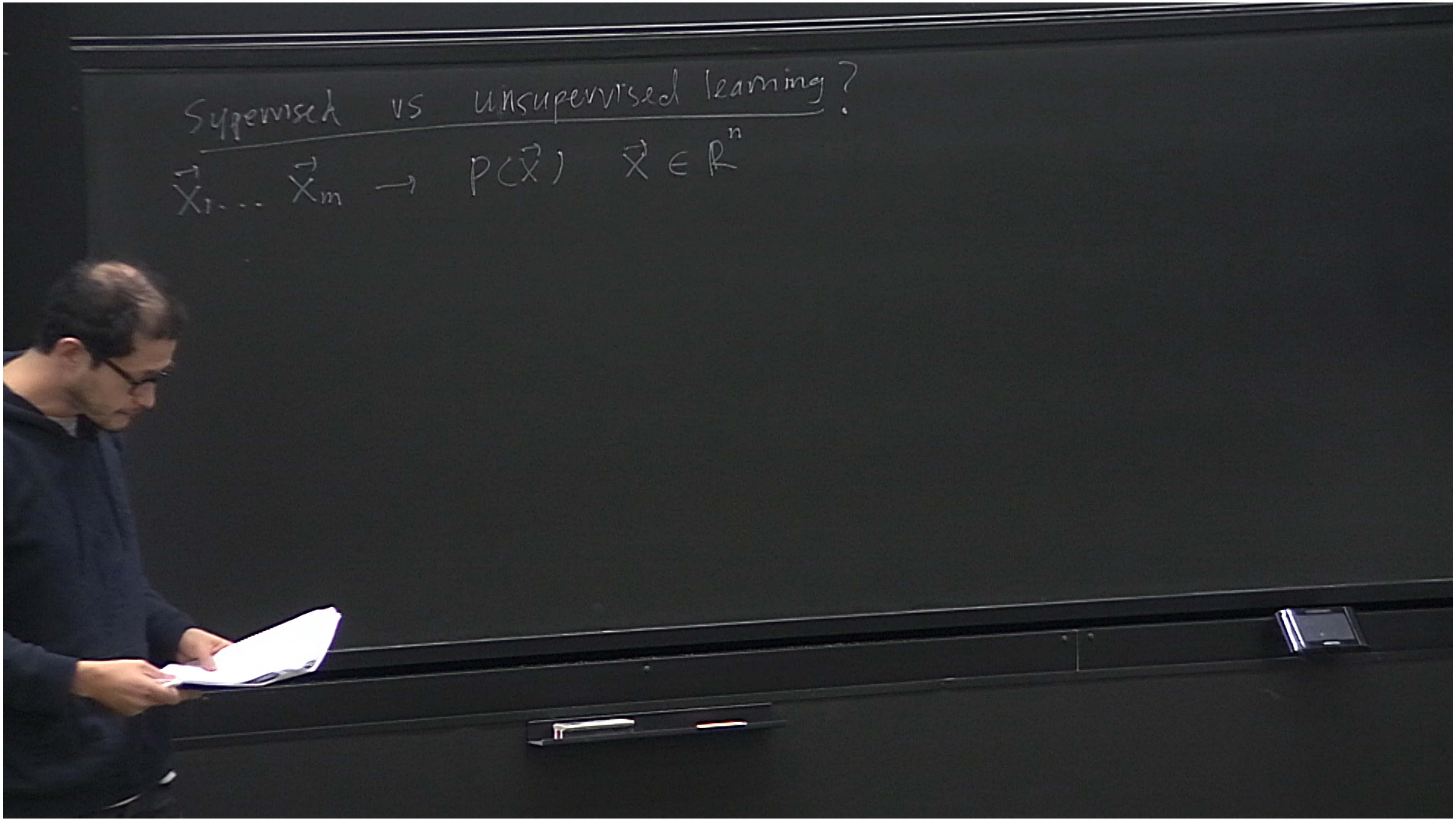
0, 1, ... 9
7 7
$\vec{X}_1, \vec{X}_2, \ldots \vec{X}_m$
up
down
5 8
PM

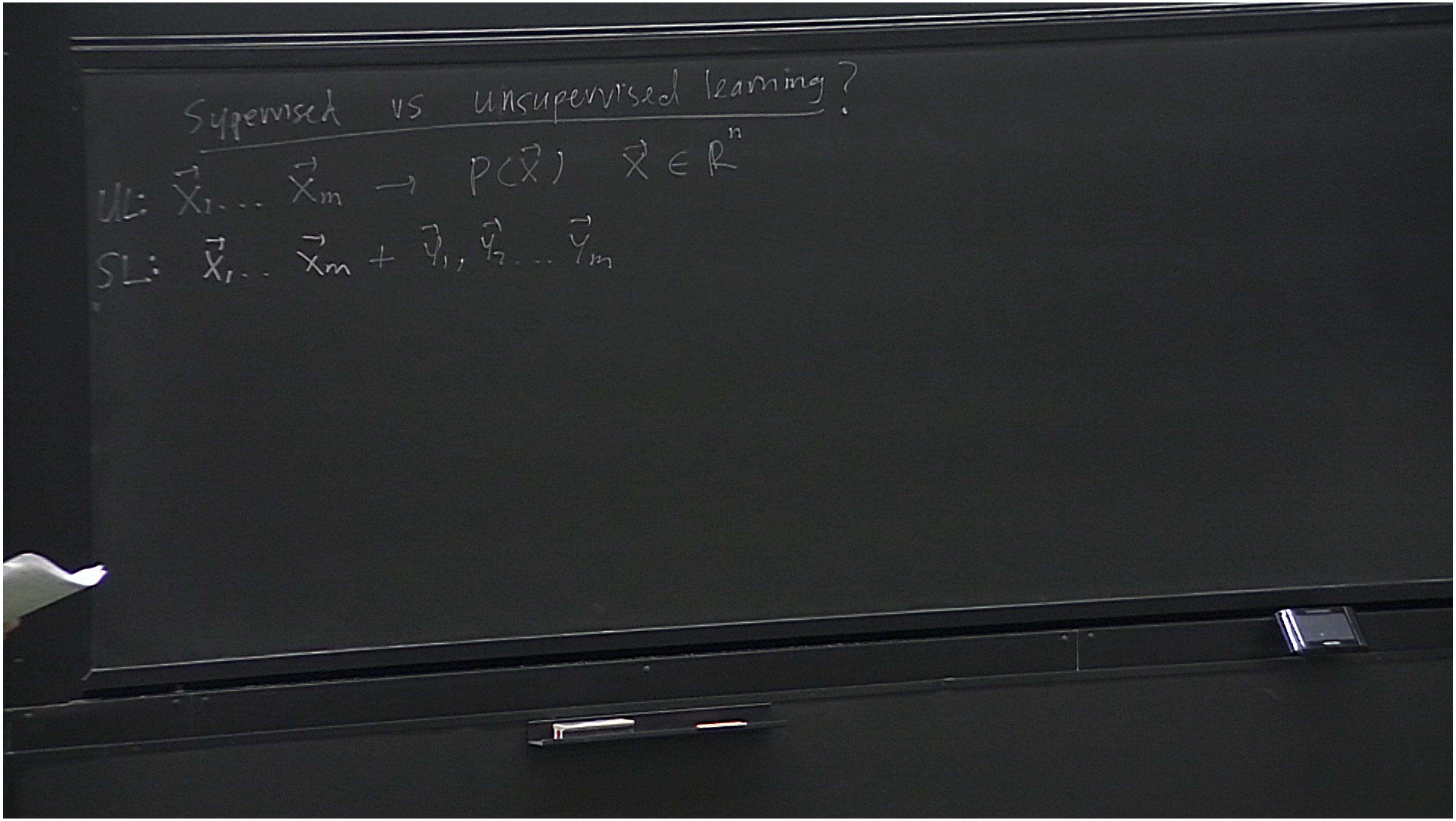

ed.
X1, X2, ... Xm
0, 1, ... 9
7 7
PM
up
down
random
0
5 8

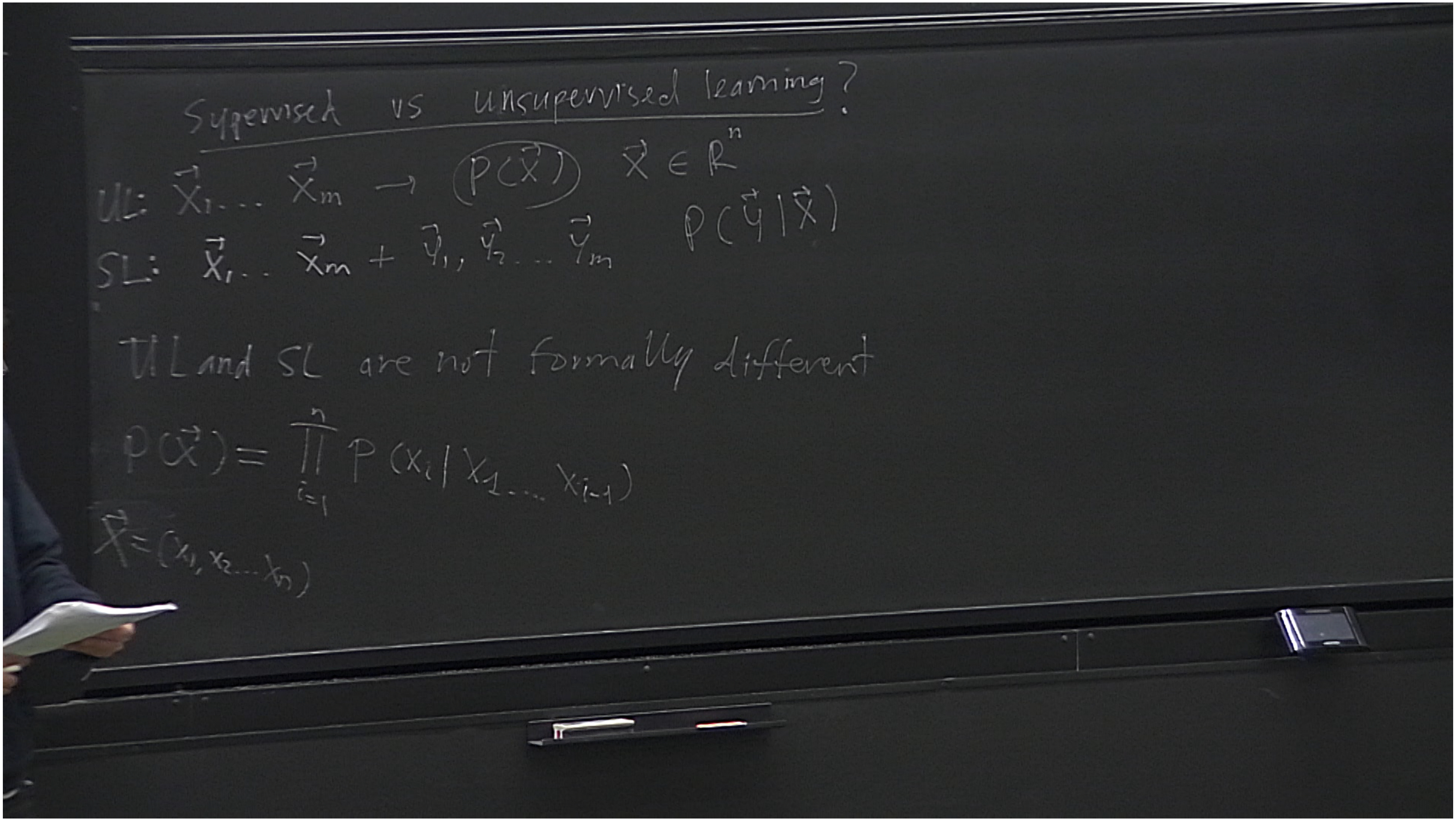

0, 1, ... 9
7 7
Xm
random
5 8
28×28

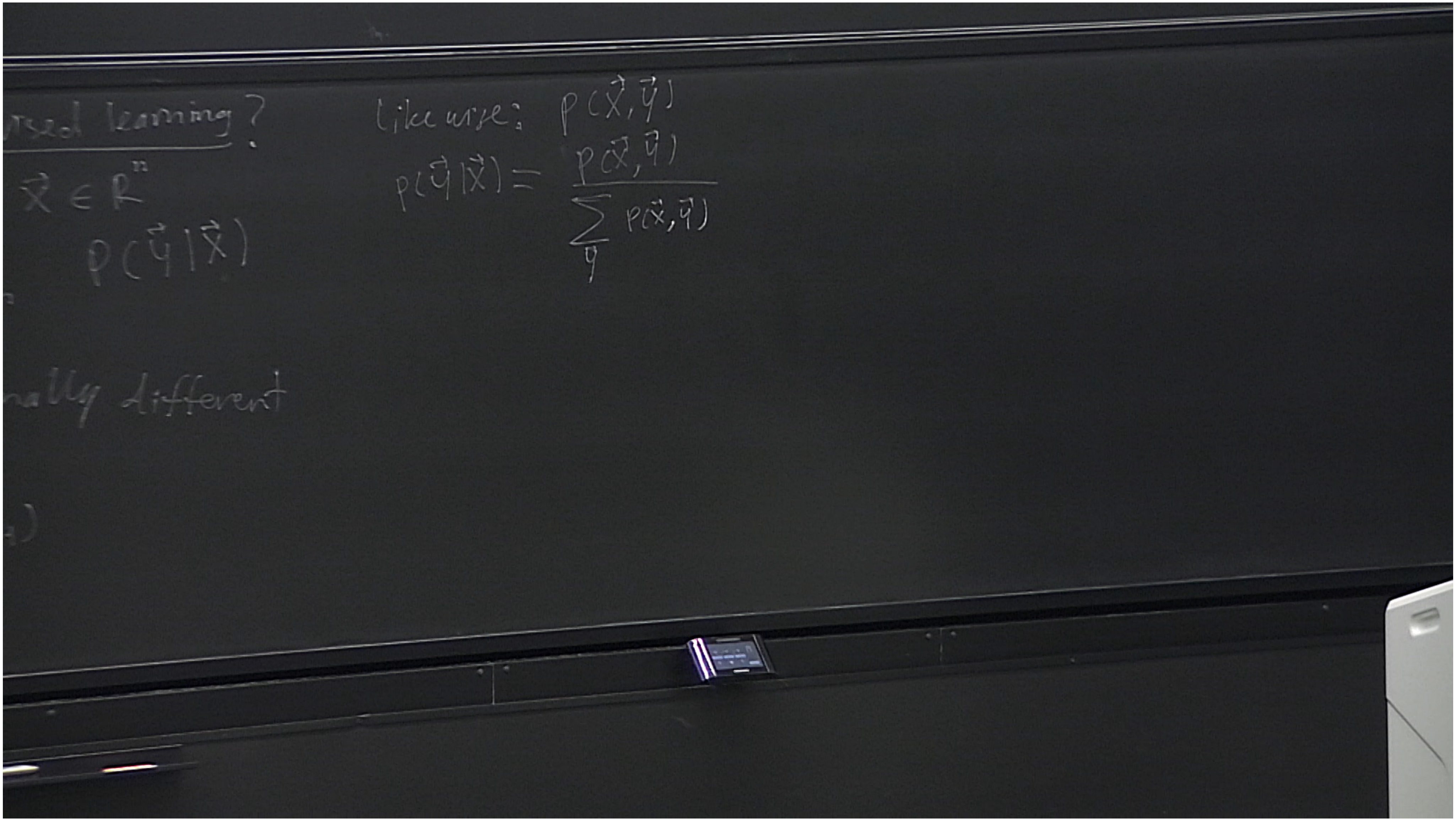

0, 1, ... 9
7 7
random
5 8

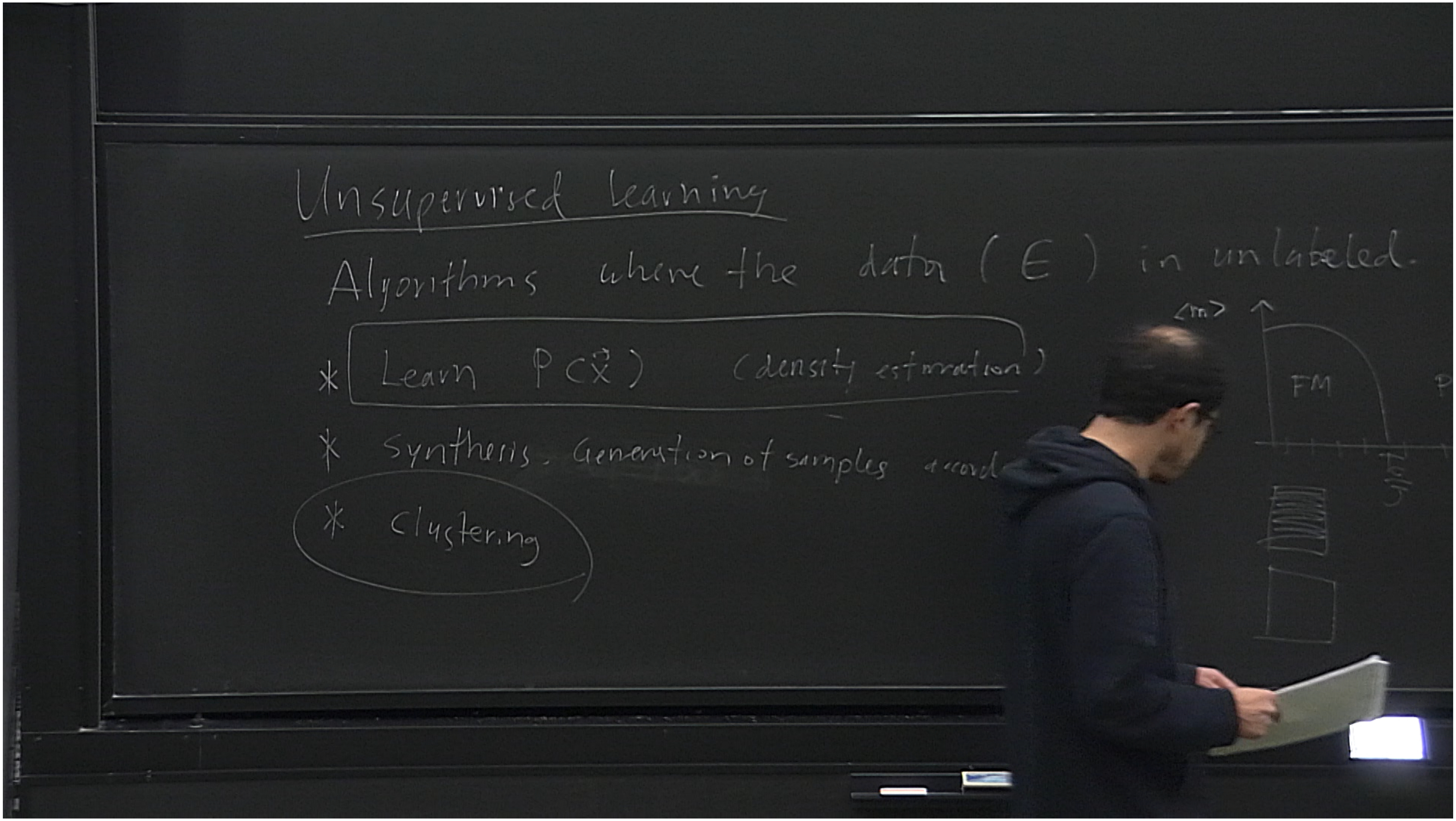

Unsupervised learning
Algorithms where the data (E) in unlabeled.
* Learn P(x) (density estimation)
* Synthesis. Generation of samples according P(x)
* Clustering
<m>
FM


Supervised vs unsupervised learning?
$\vec{X}_1 \ldots \vec{X}_m \longrightarrow P(\vec{X})$ $\vec{X} \in \mathbb{R}^n$

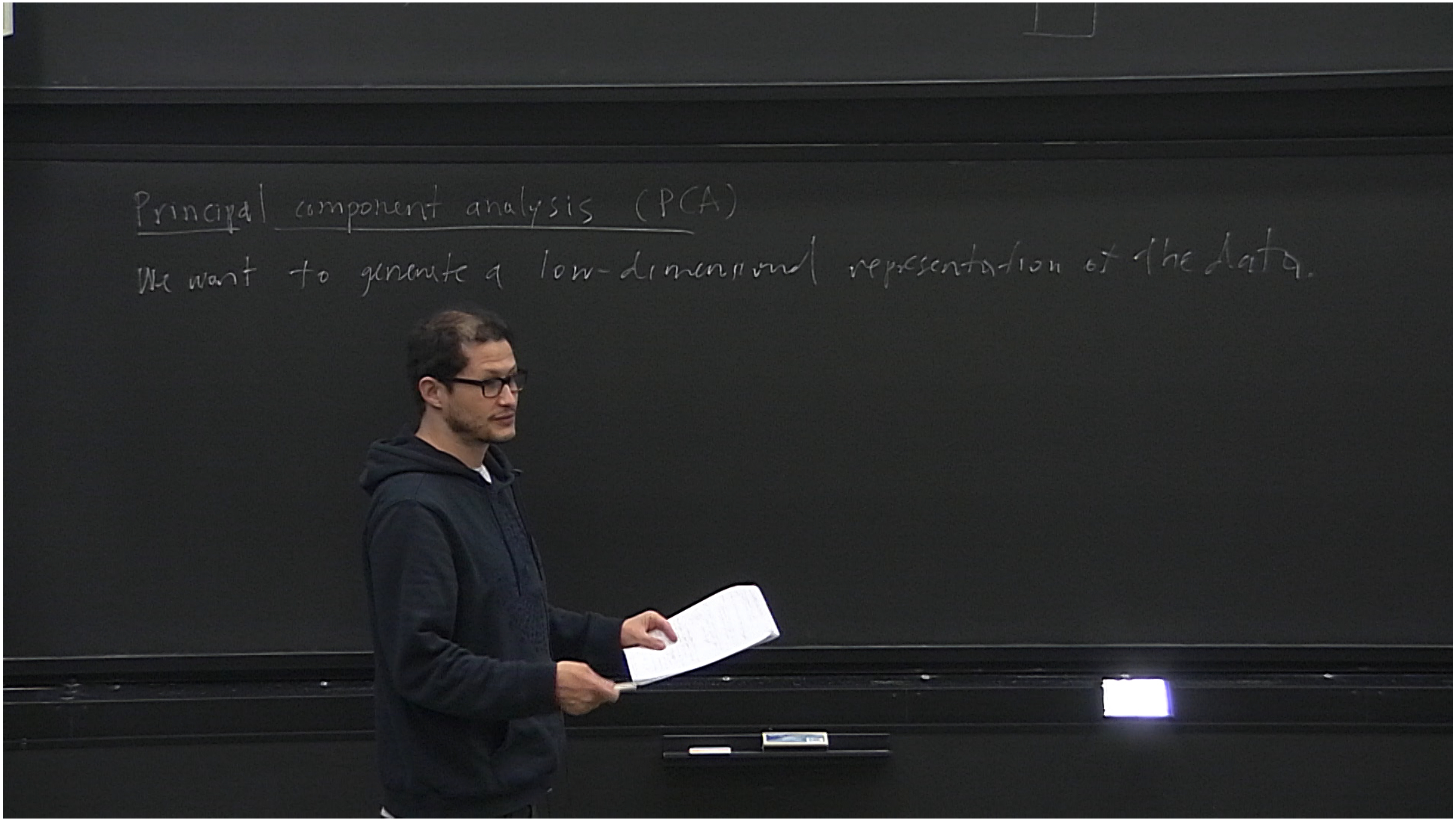

Supervised vs unsupervised learning?
UL: $\vec{X}_1 \ldots \vec{X}_m \to P(\vec{X})$ $\vec{X} \in R^n$
SL: $\vec{X}_1 \ldots \vec{X}_m + \vec{y}_1, \vec{y}_2 \ldots \vec{y}_m$


Supervised vs unsupervised learning ?
UL: $\vec{x}_1 \dots \vec{x}_m \longrightarrow P(\vec{x})$ $\vec{x} \in R^n$
SL: $\vec{x}_1 \dots \vec{x}_m + \vec{y}_1, \vec{y}_2 \dots \vec{y}_m$ $P(\vec{y}|\vec{x})$
UL and SL are not formally different
$P(\vec{x}) = \prod_{i=1}^{n} P(x_i | x_1 \dots x_{i-1})$
$\vec{x} = (x_1, x_2 \dots x_n)$


rsed learning?
likewise: $p(\vec{x},\vec{y})$
$p(\vec{y}|\vec{x}) = \frac{p(\vec{x},\vec{y})}{\sum_{\vec{y}} p(\vec{x},\vec{y})}$
$\vec{x} \in R^n$
$p(\vec{y}|\vec{x})$
nally different



Unsupervised learning
Algorithms where the data (E) in unlabeled.
* Learn P(x⃗) (density estimation)
* Synthesis. Generation of samples
* Clustering
<m>
FM


Principal component analysis (PCA)
We want to generate a low-dimensional representation of the data.

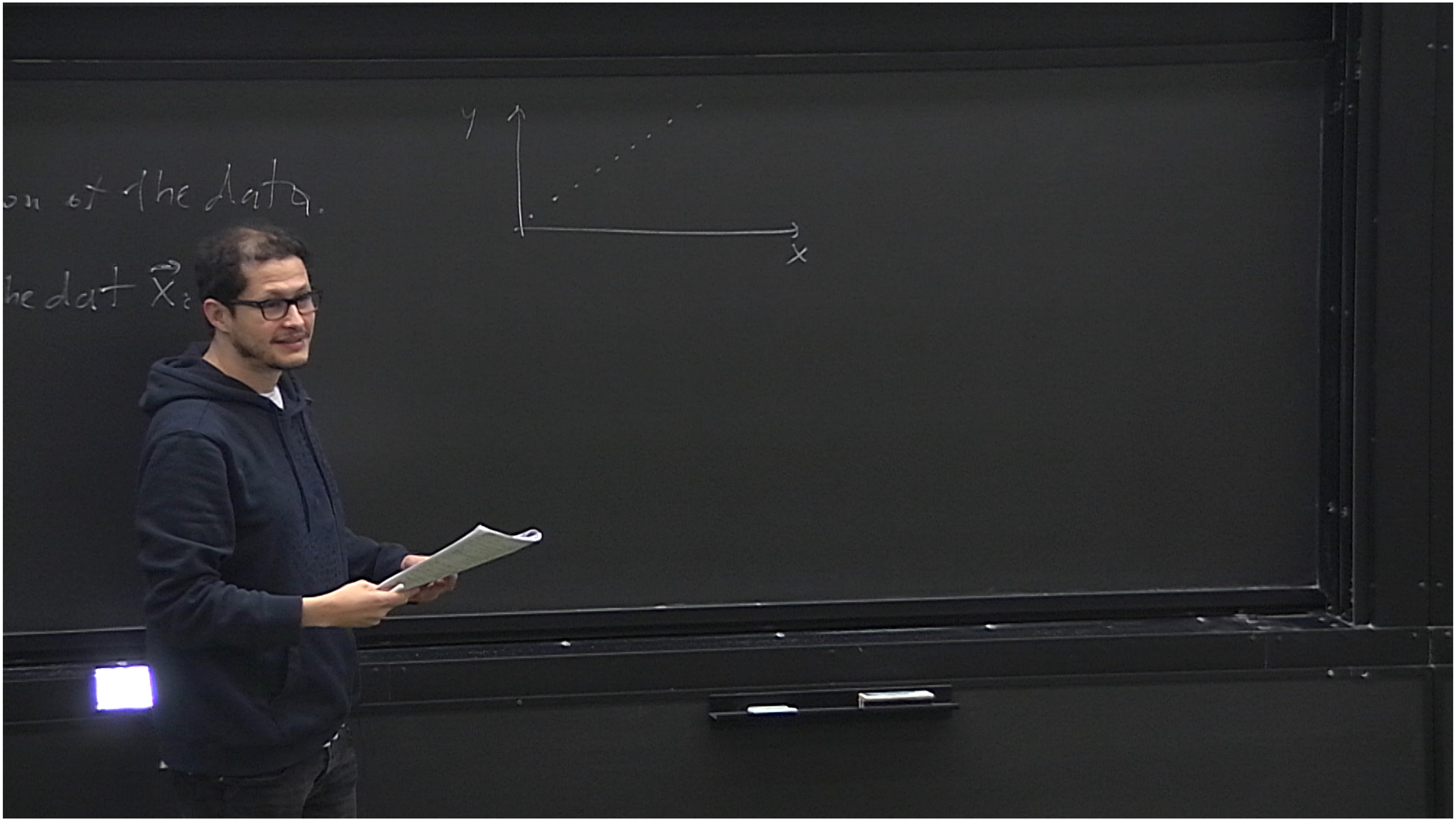
Principal component analysis (PCA)

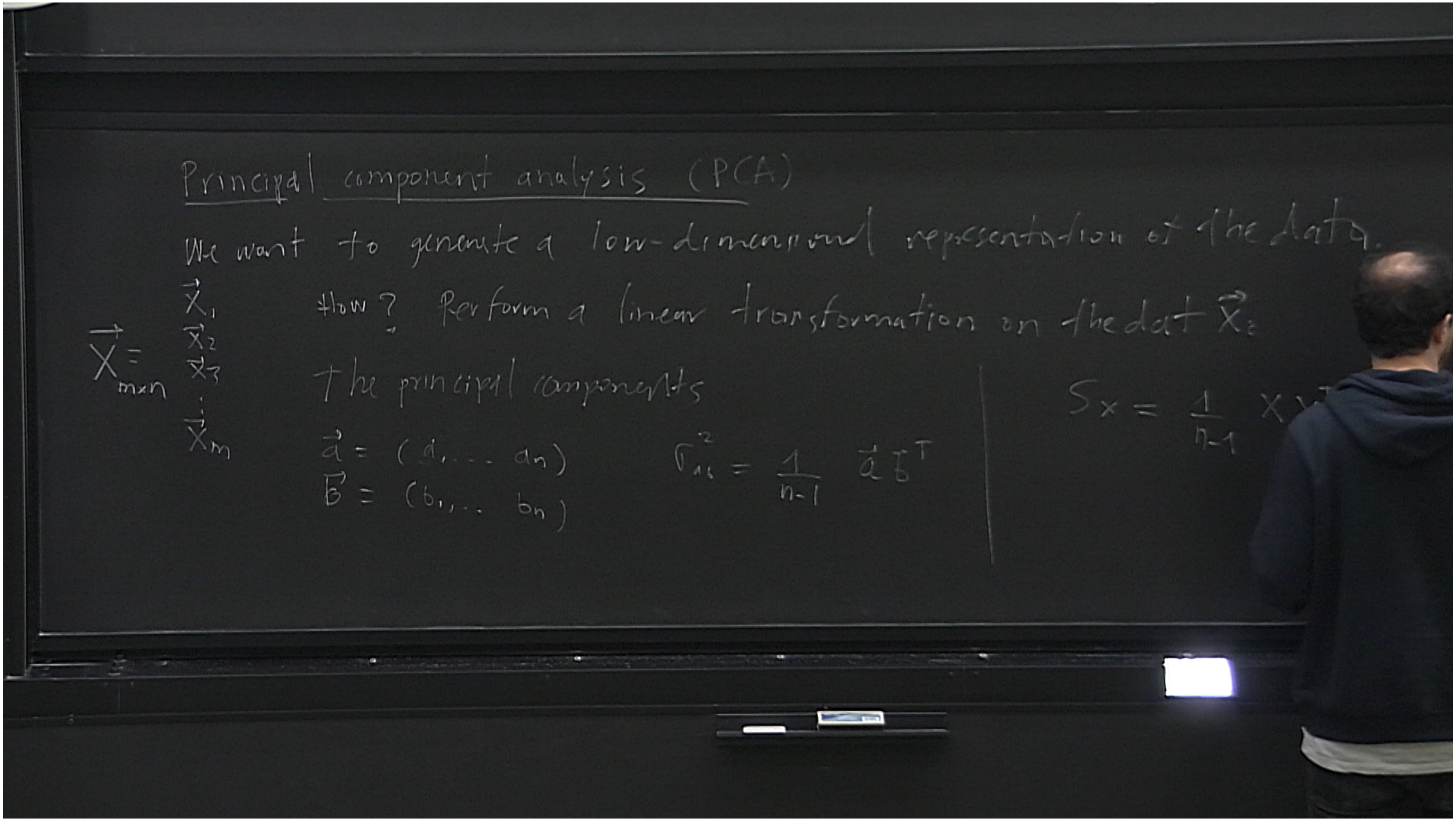
We want to generate a low-dimensional representation of the data.

$$\vec{X}_{m\times n} = \begin{matrix} \vec{x}_1 \\ \vec{x}_2 \\ \vec{x}_3 \\ \vdots \\ \vec{x}_m \end{matrix}$$

How? Perform a linear transformation on the dat $\vec{x}_i$

The principal components


of the data.
y
x


Principal component analysis (PCA)
We want to generate a low-dimensional representation of the data.
X = x1, x2, x3, ..., xm (m×n)
How? Perform a linear transformation on the dat xi
The principal components
a = (a1, ... an)
b = (b1, ... bn)
σab² = 1/(n-1) a b^T
Sx = 1/(n-1) X

Principal component analysis (PCA)

We want to generate a low-dimensional representation of the data.

$$\vec{X}_{m\times n} = \begin{matrix} \vec{x}_1 \\ \vec{x}_2 \\ \vec{x}_3 \\ \vdots \\ \vec{x}_m \end{matrix}$$

How? Perform a linear transformation on the data $\vec{x}_i$

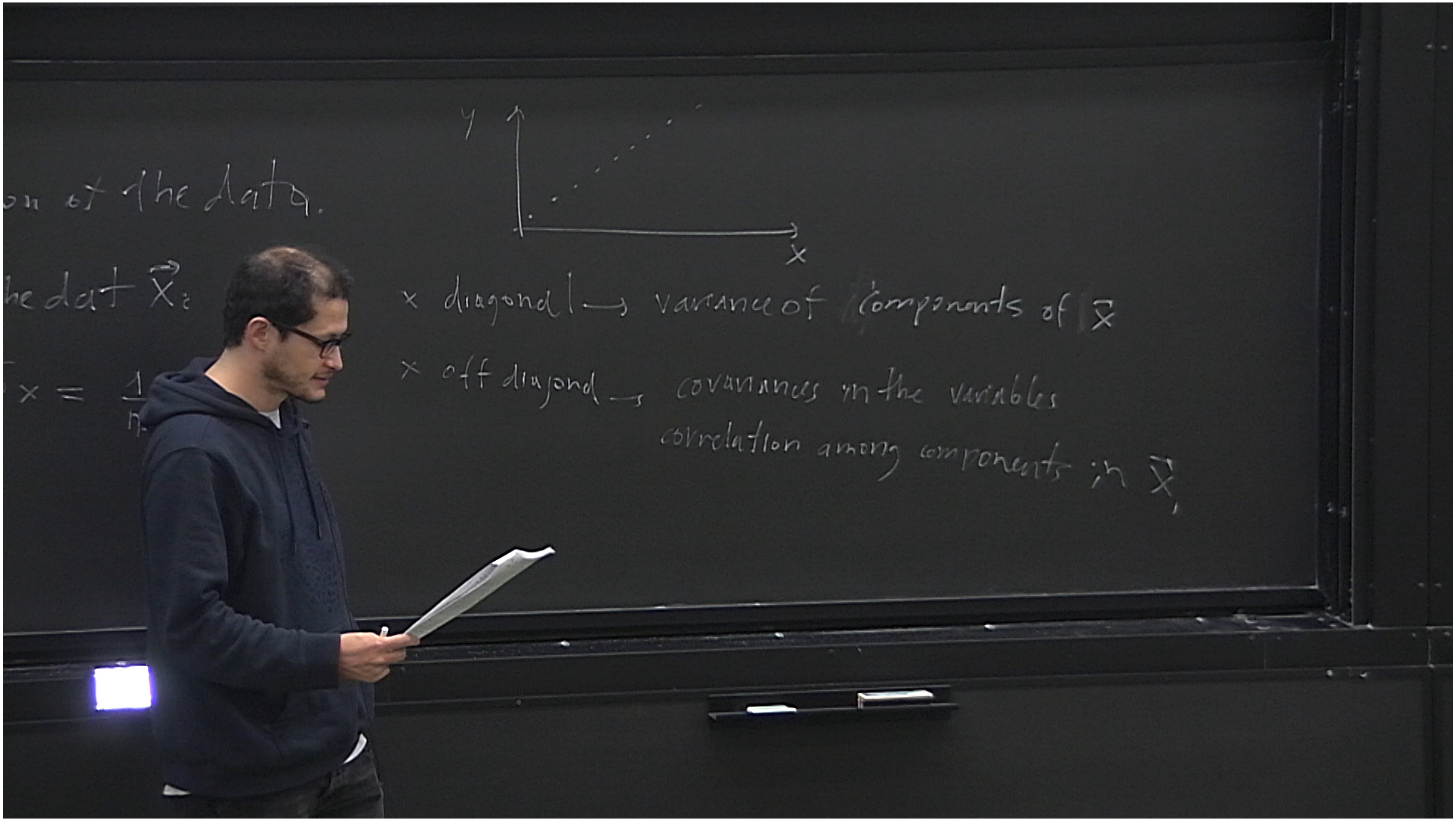
The principal components

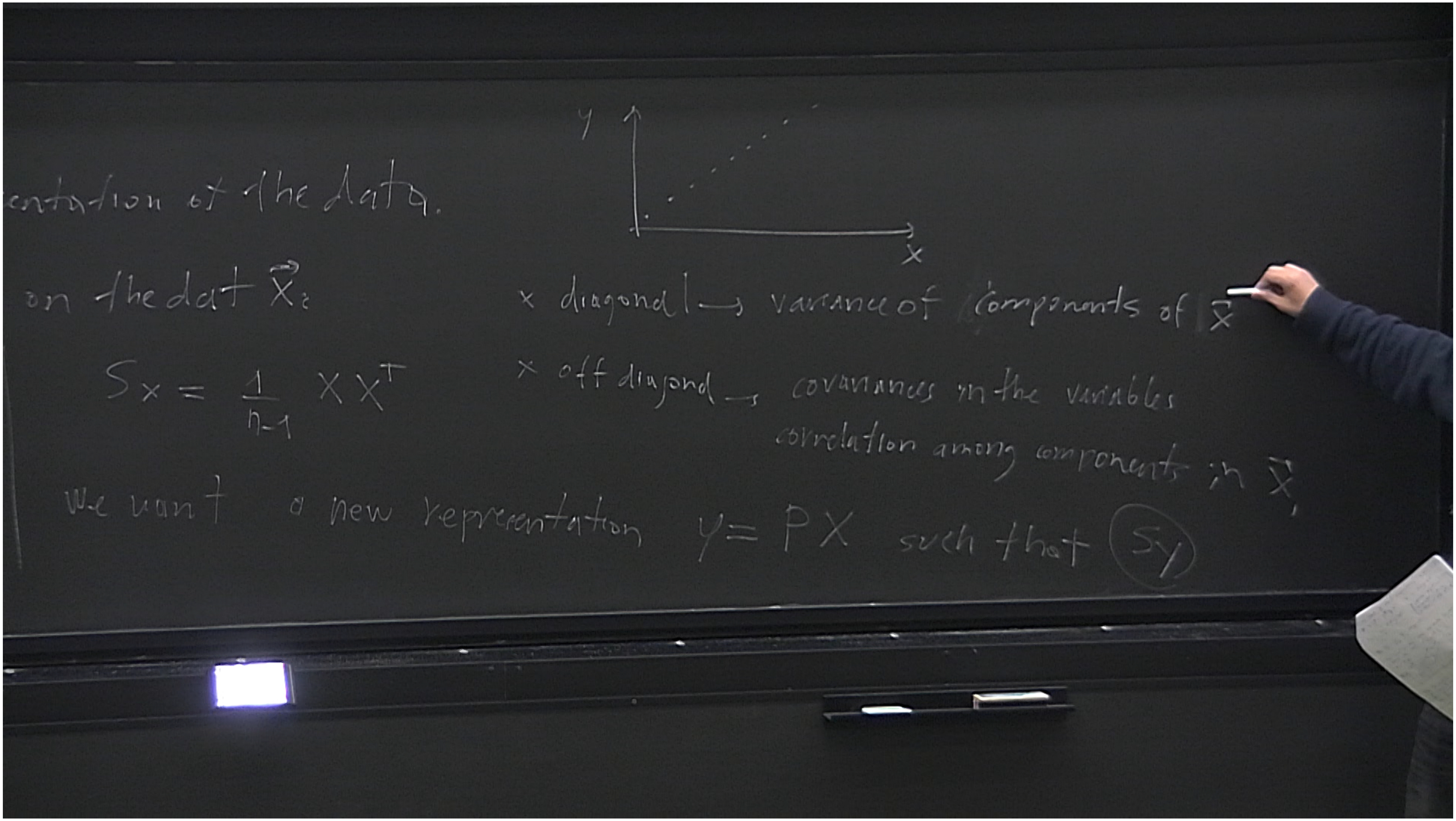
$\vec{a} = (a_1, \dots a_n)$

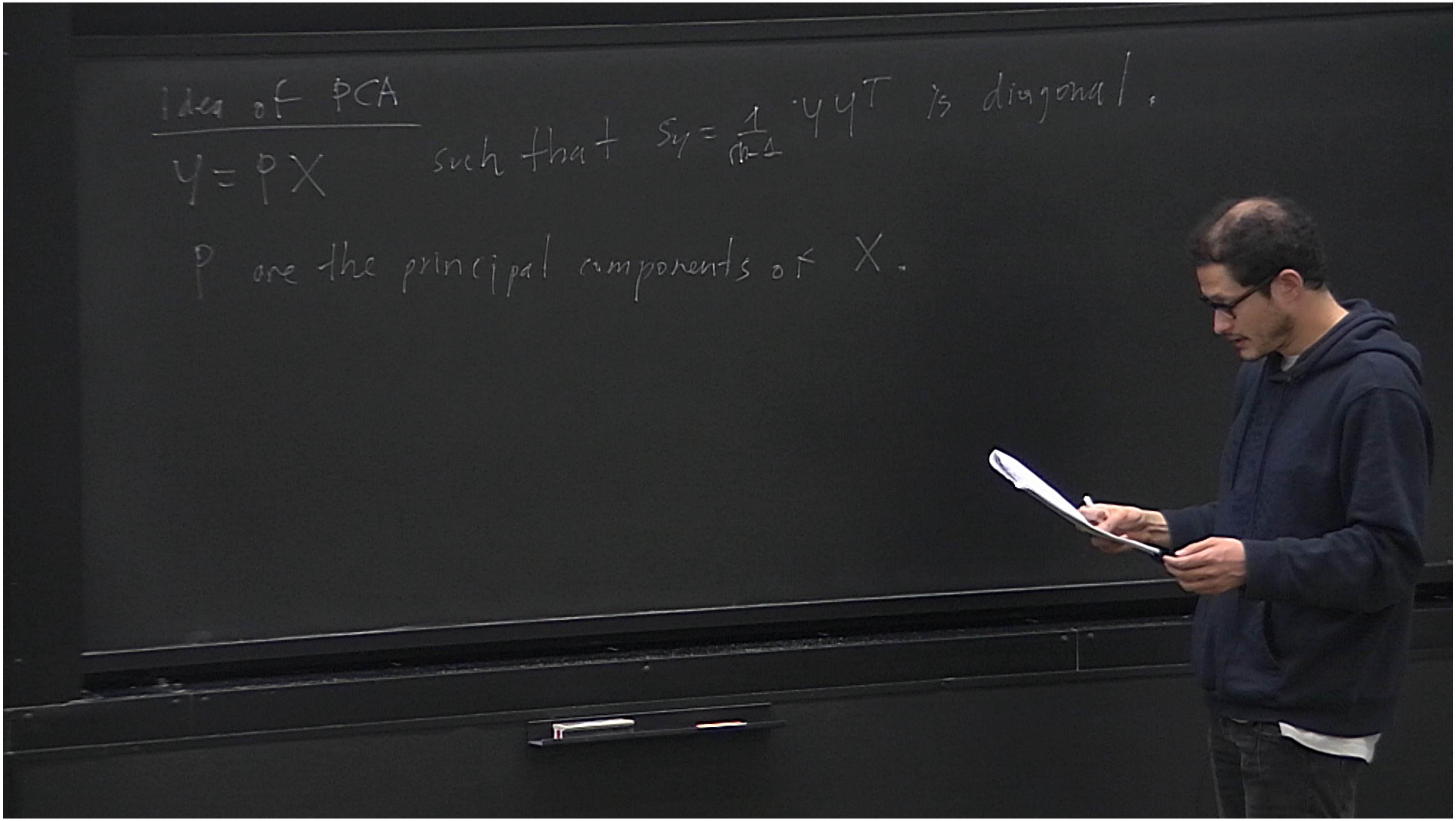
$\vec{b} = (b_1, \dots b_n)$

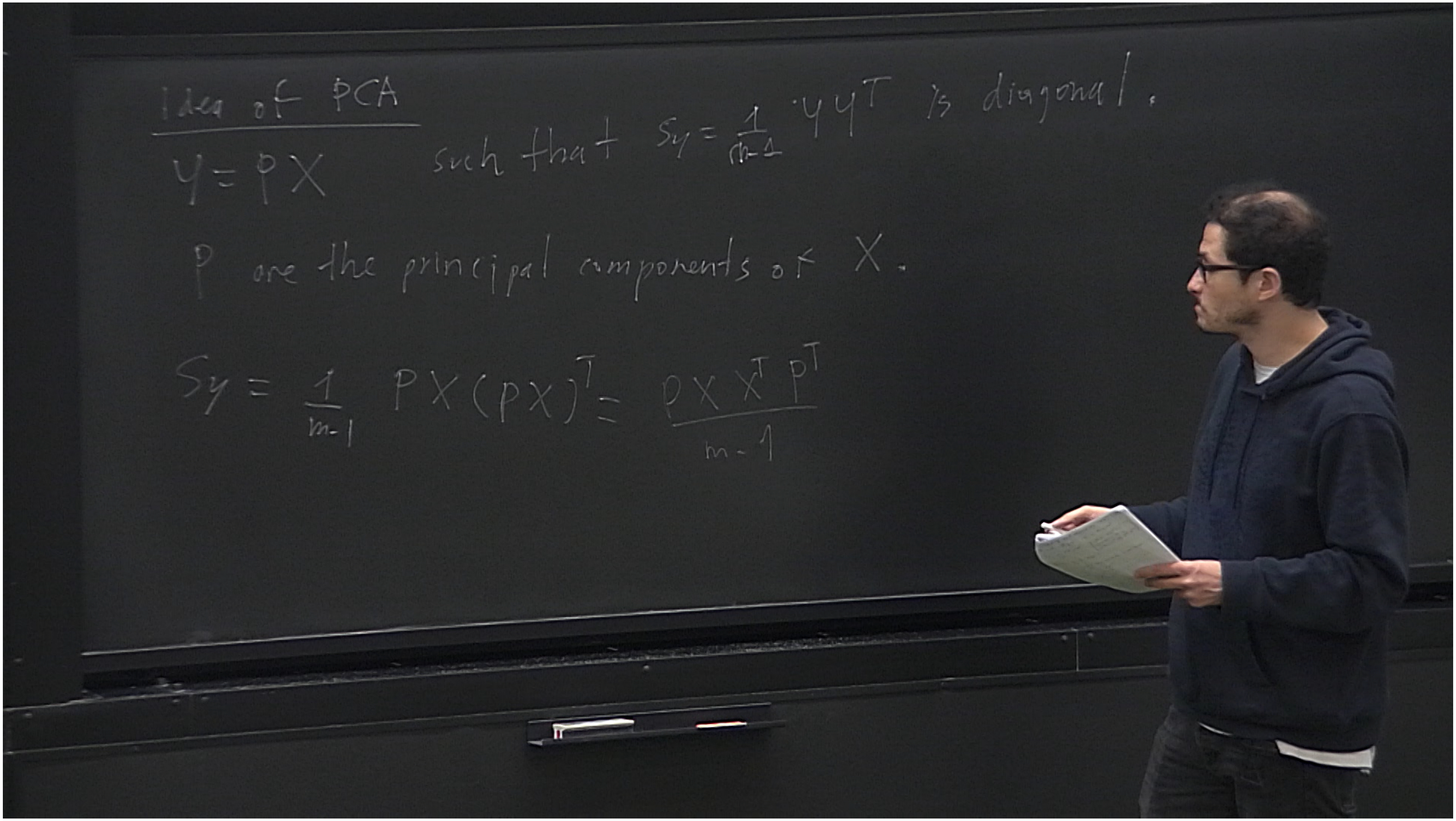
$$\sigma_{ab}^2 = \frac{1}{n-1} \vec{a}\,\vec{b}^T$$

$$S_x = \frac{1}{n-1} X X^T$$


on of the data.
y
x
he dat X
× diagonal → variance of components of X
× off diagonal → covariances in the variables
correlation among components in X


entation of the data.
on the dat $\vec{X}_i$
$S_X = \frac{1}{n-1} X X^T$
y
x
× diagonal → variance of components of $\vec{X}$
× off diagonal → covariances in the variables
correlation among components in $\vec{X}$,
We want a new representation $Y = PX$ such that S_Y


Idea of PCA
Y = PX such that S_Y = 1/(n-1) · YY^T is diagonal.
P are the principal components of X.


Idea of PCA
$Y = PX$ such that $S_Y = \frac{1}{m-1} YY^T$ is diagonal.
P are the principal components of X.
$S_Y = \frac{1}{m-1} PX(PX)^T = \frac{PXX^TP^T}{m-1}$

(A)

...ensional representation of the data.

...transformation on the dat $\vec{X}_i$

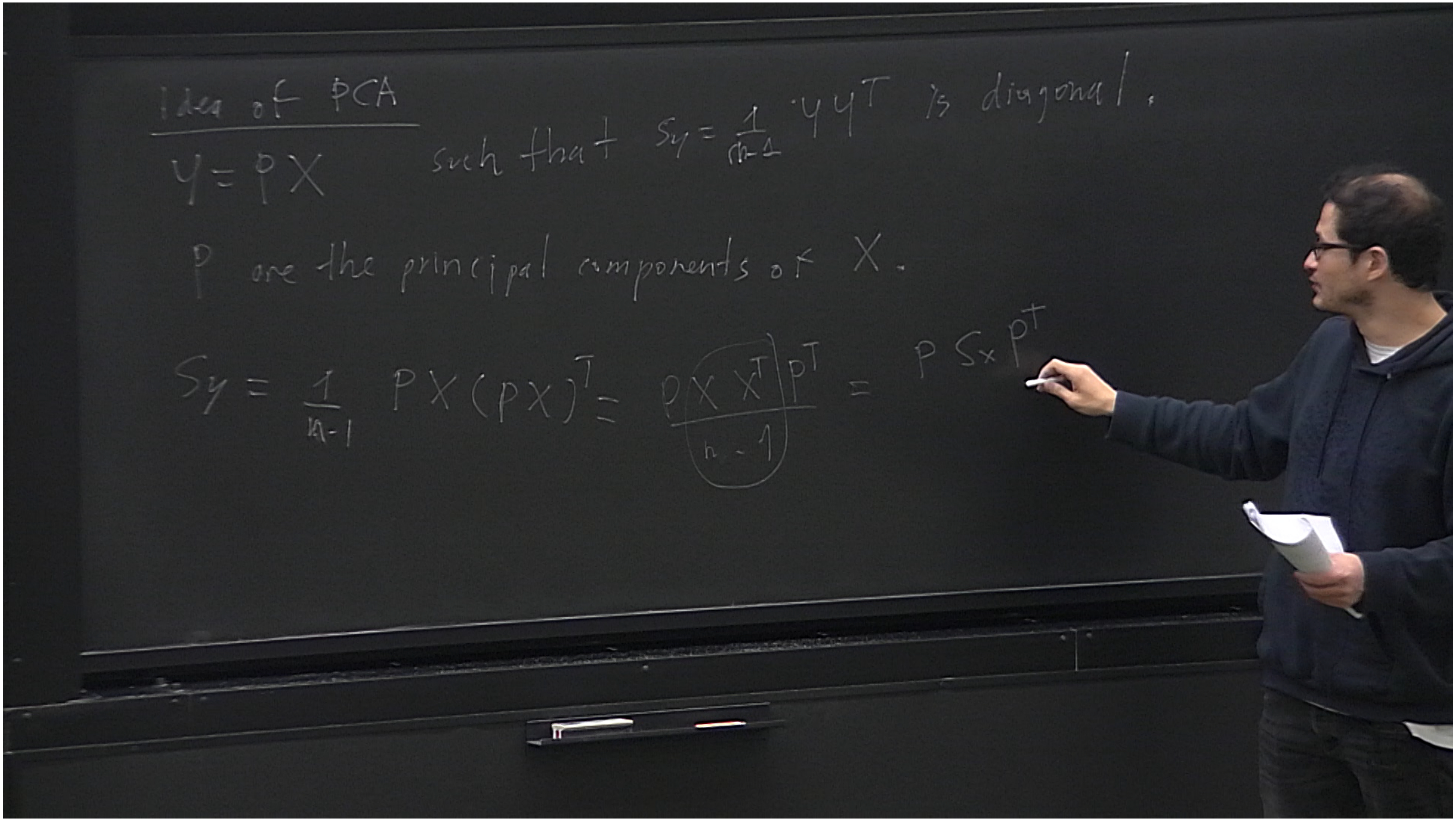
× diagonal → variance of components of

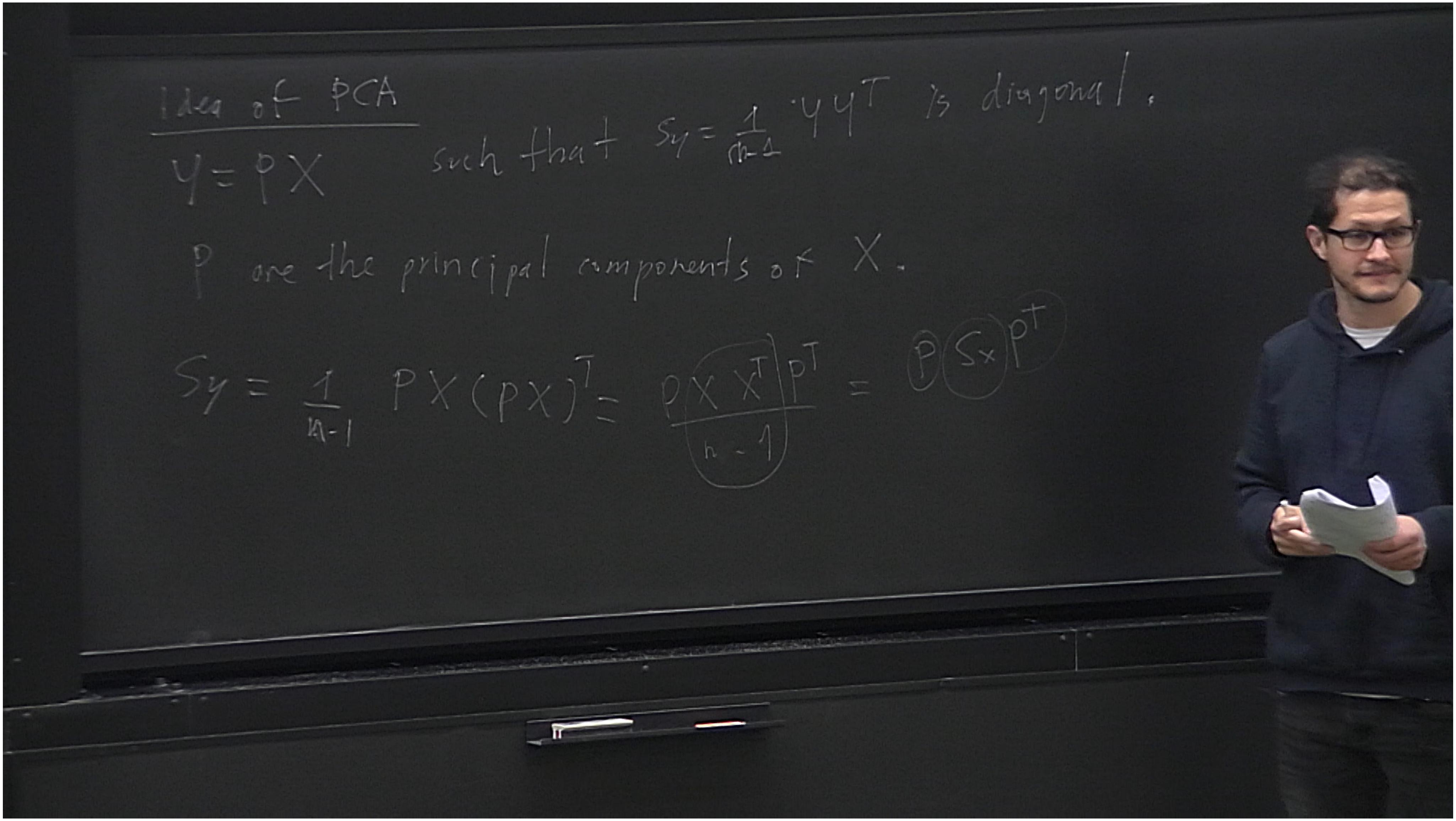
× off diagonal → covariances in the variables
correlation among components

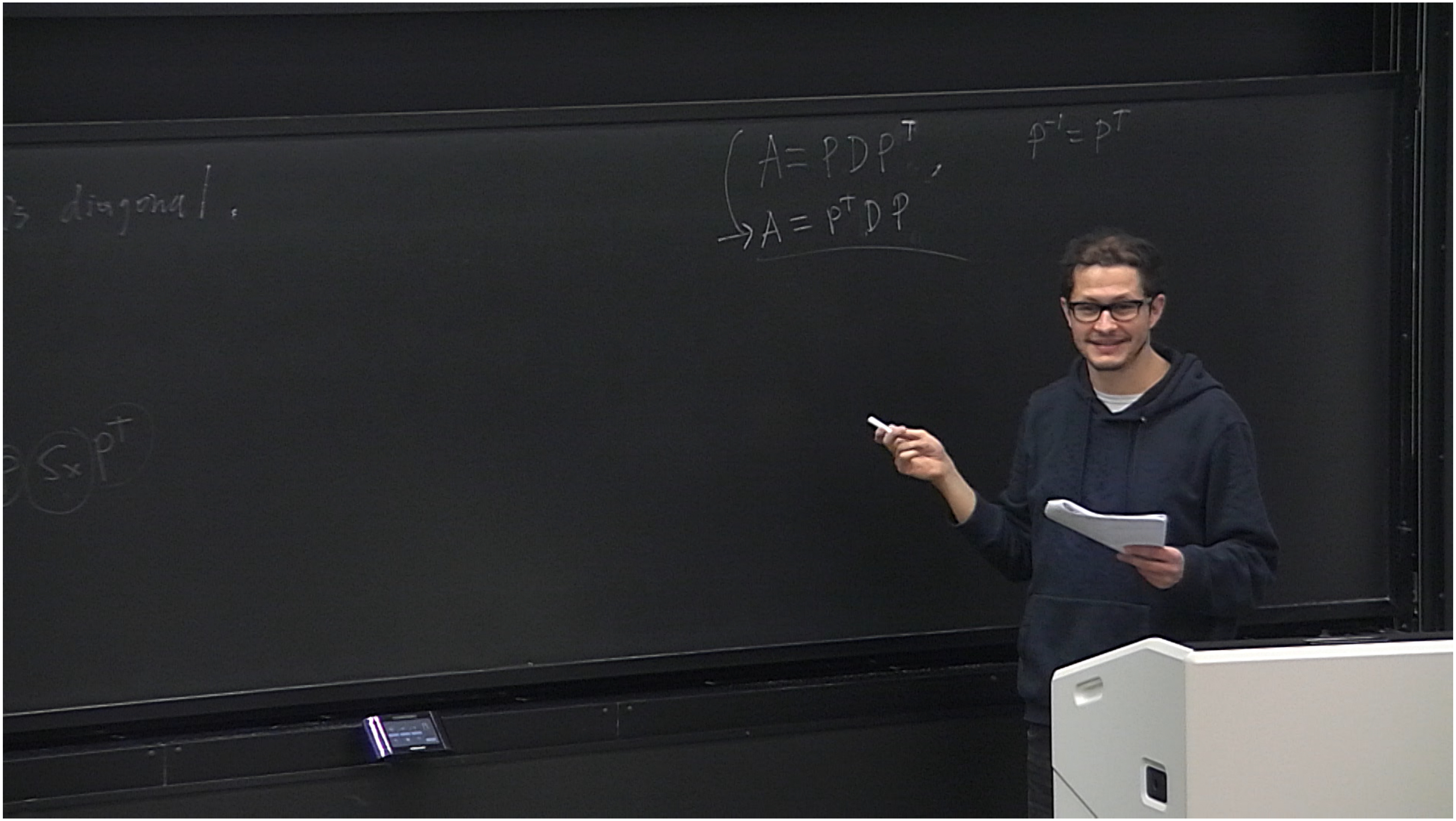
$$= \frac{1}{n-1} \vec{a}\,\vec{b}^T$$

$$S_X = \frac{1}{n-1} X X^T$$

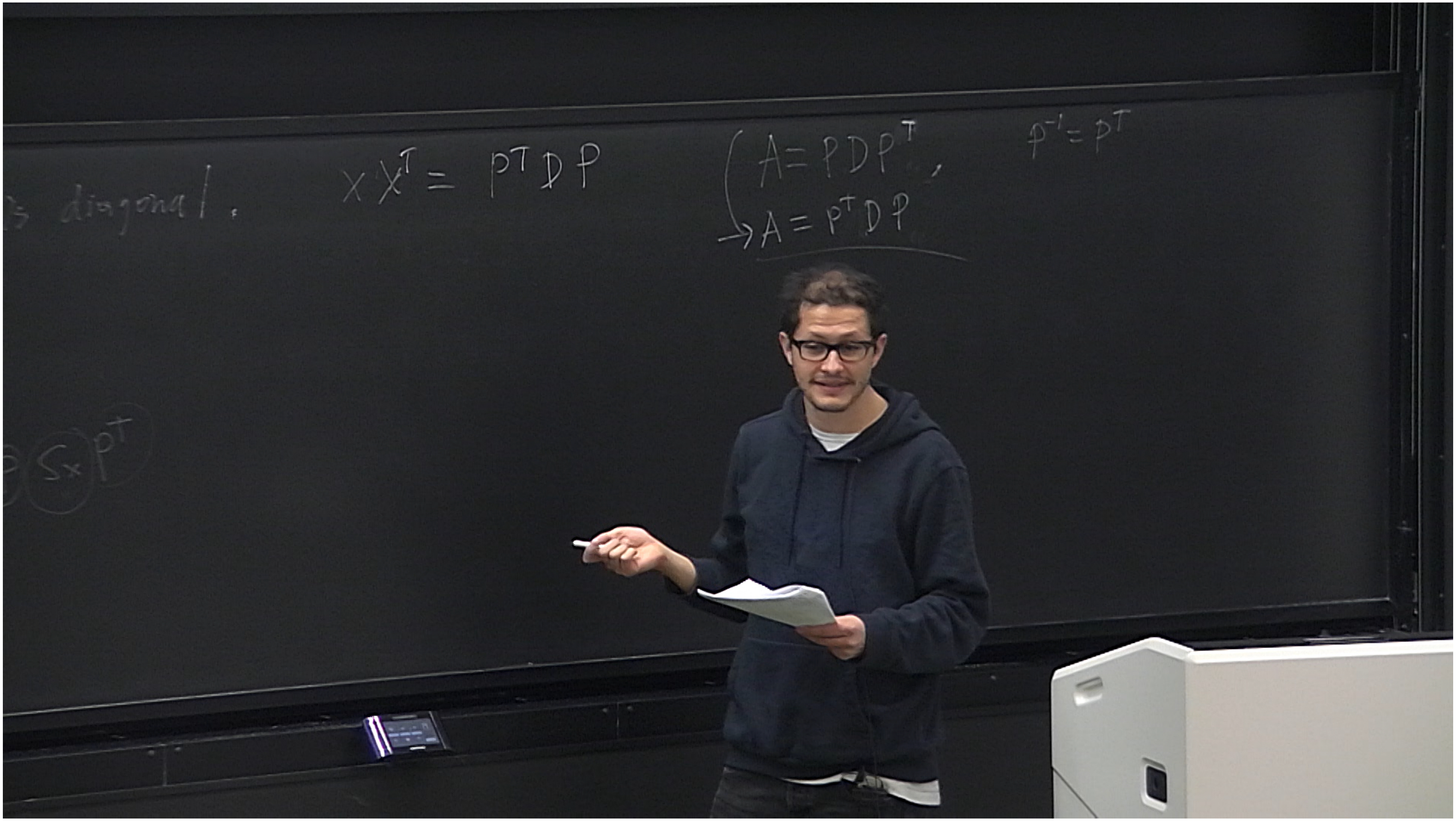
We want a new representation $Y = PX$ such that S_Y

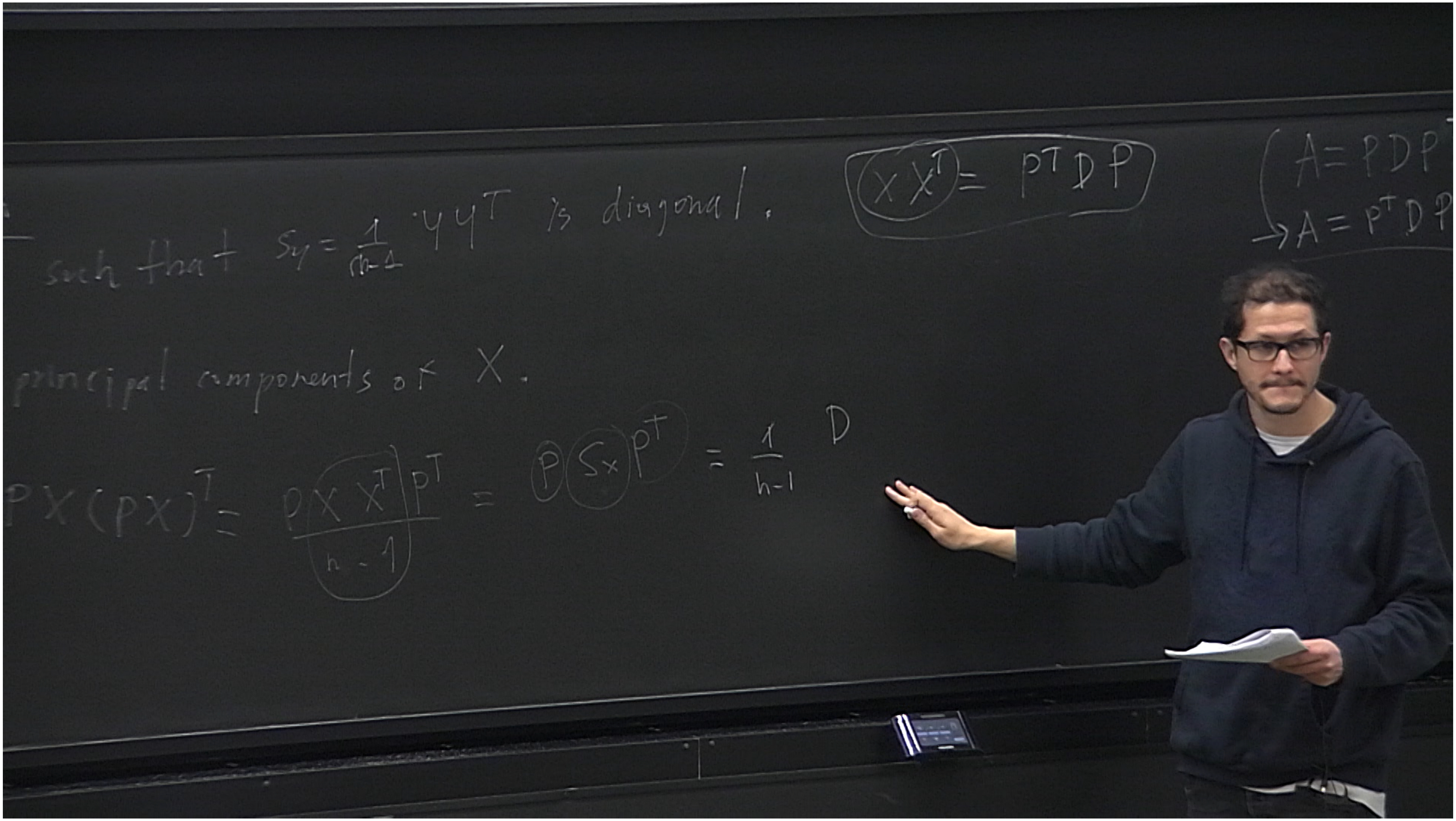

Idea of PCA
Y = PX such that Sy = 1/(n-1) YY^T is diagonal.
P are the principal components of X.
Sy = 1/(n-1) PX(PX)^T = P XX^T P^T / (n-1) = P Sx P^T

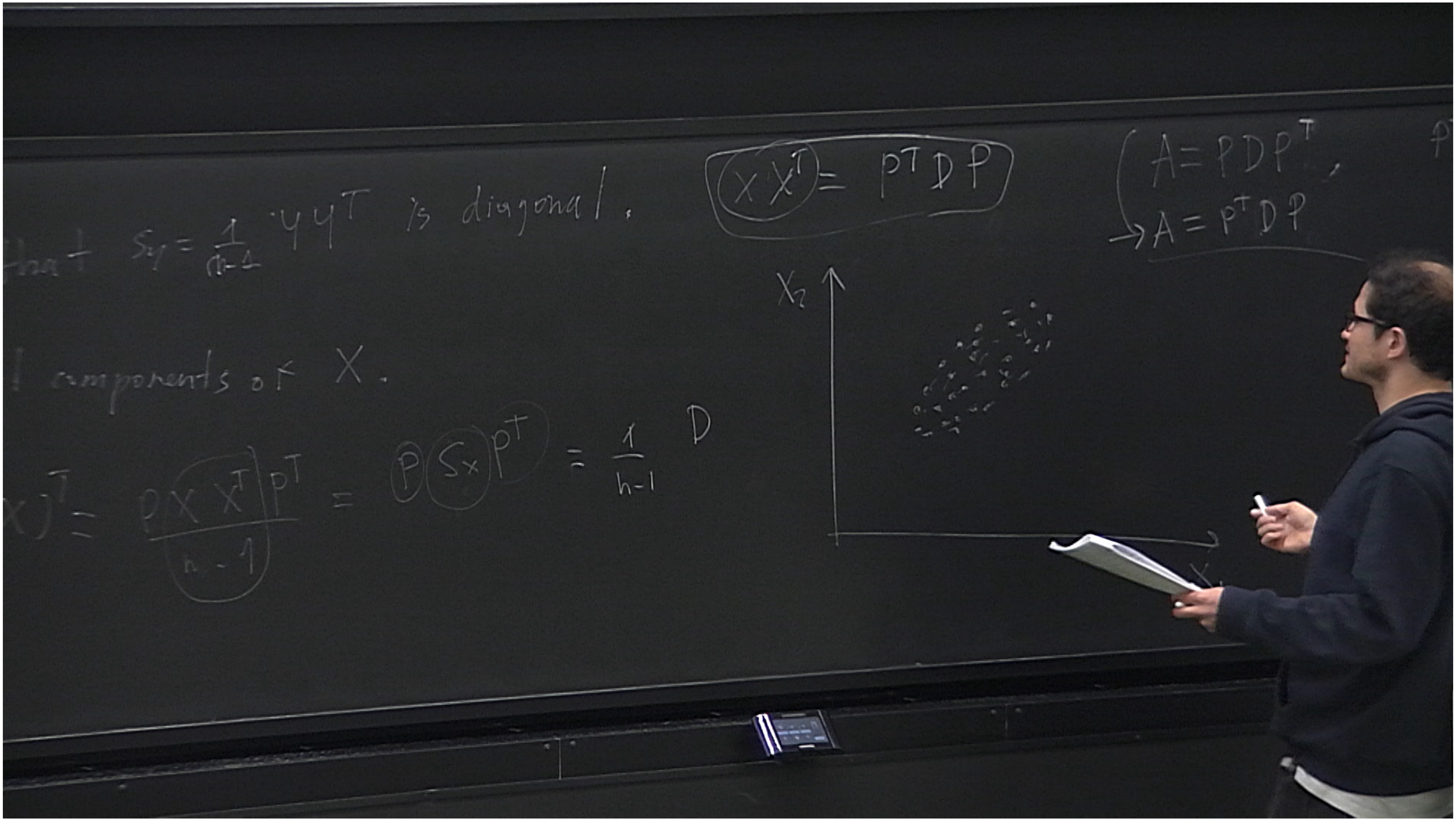

Idea of PCA
Y = PX such that S_y = 1/(n-1) YY^T is diagonal.
P are the principal components of X.
S_y = 1/(n-1) PX(PX)^T = P XX^T/(n-1) P^T = P S_x P^T

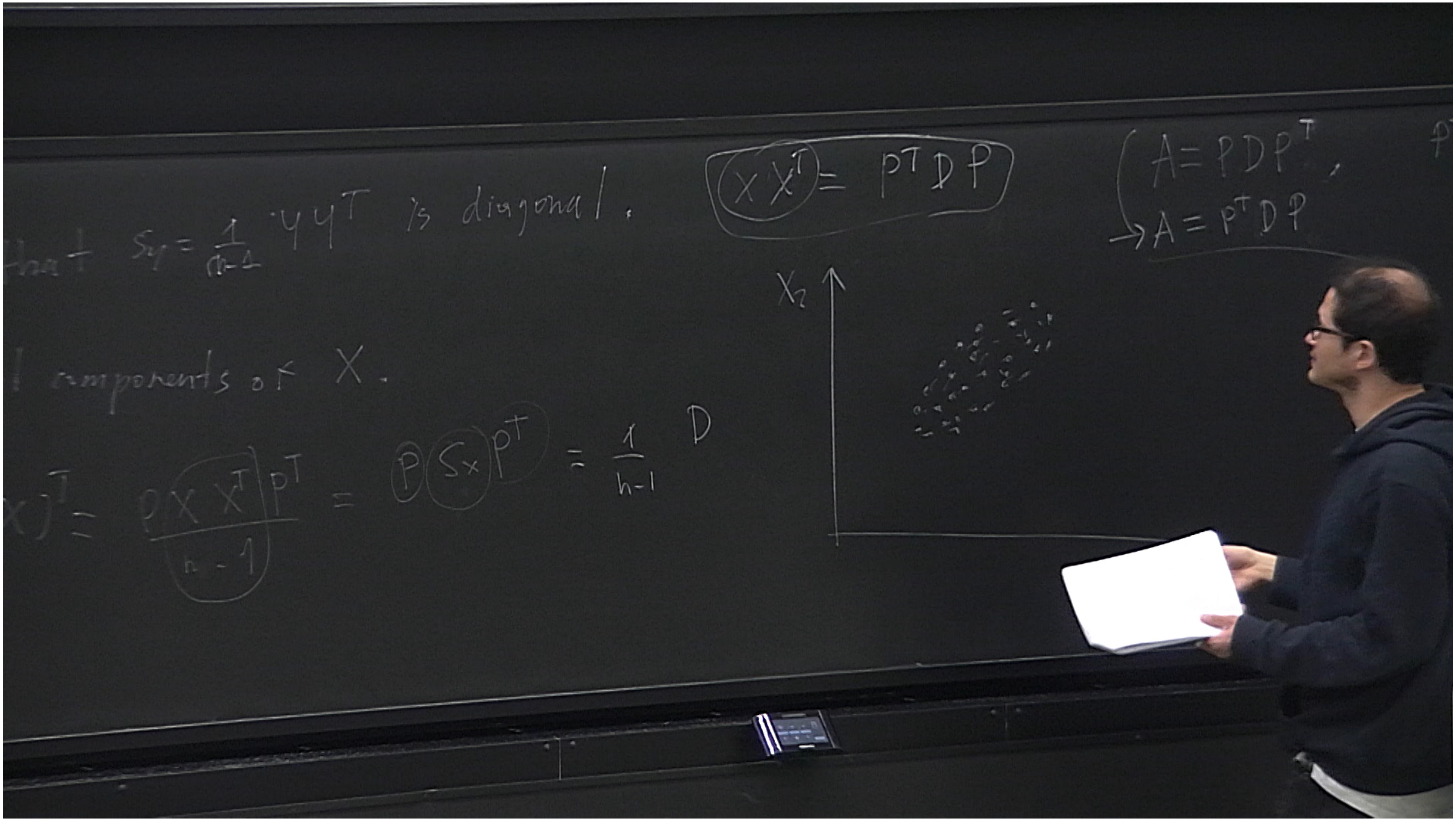


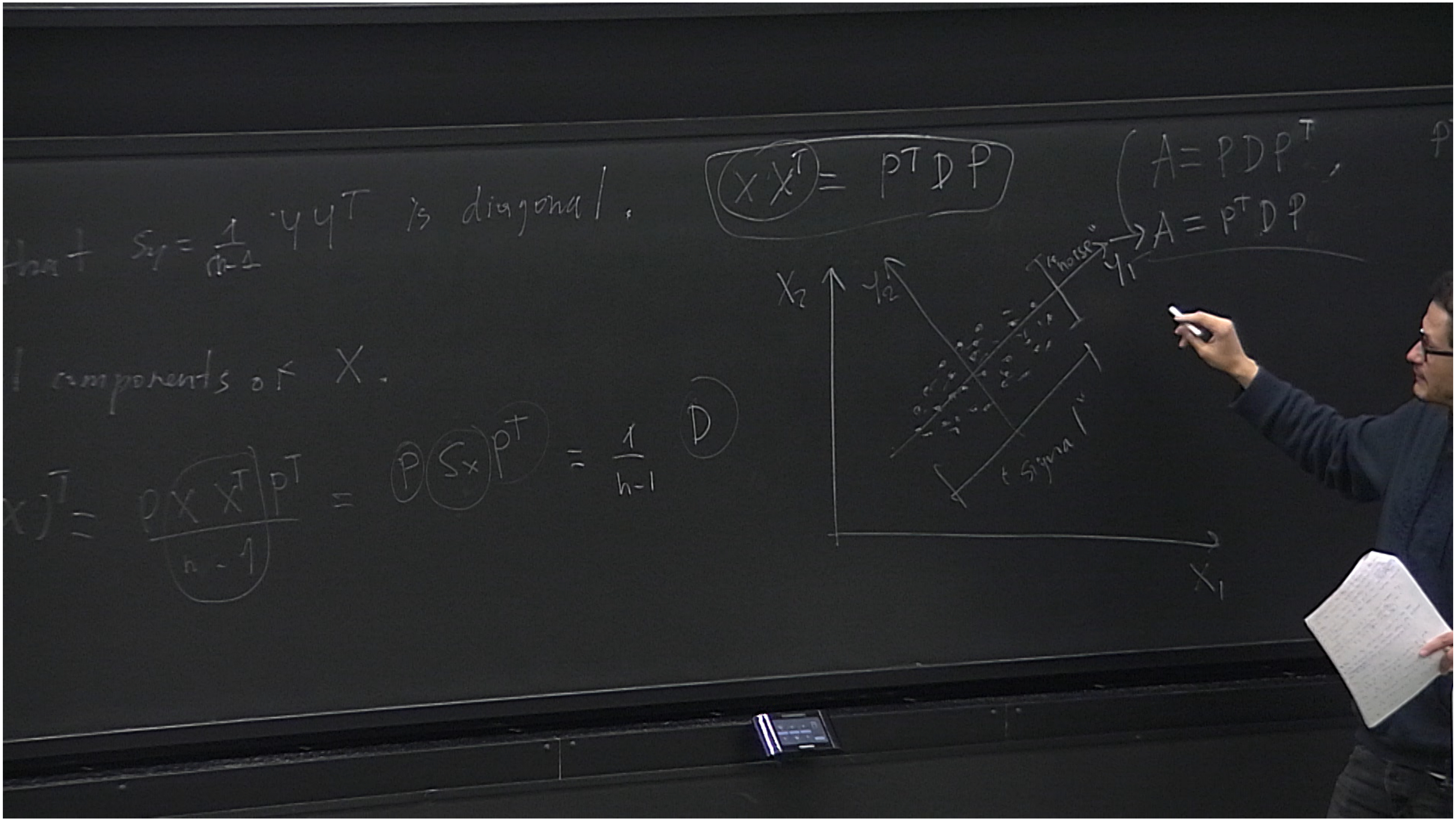
is diagonal.
A = PDP^T,
A = P^T D P
P^-1 = P^T
Sx P^T

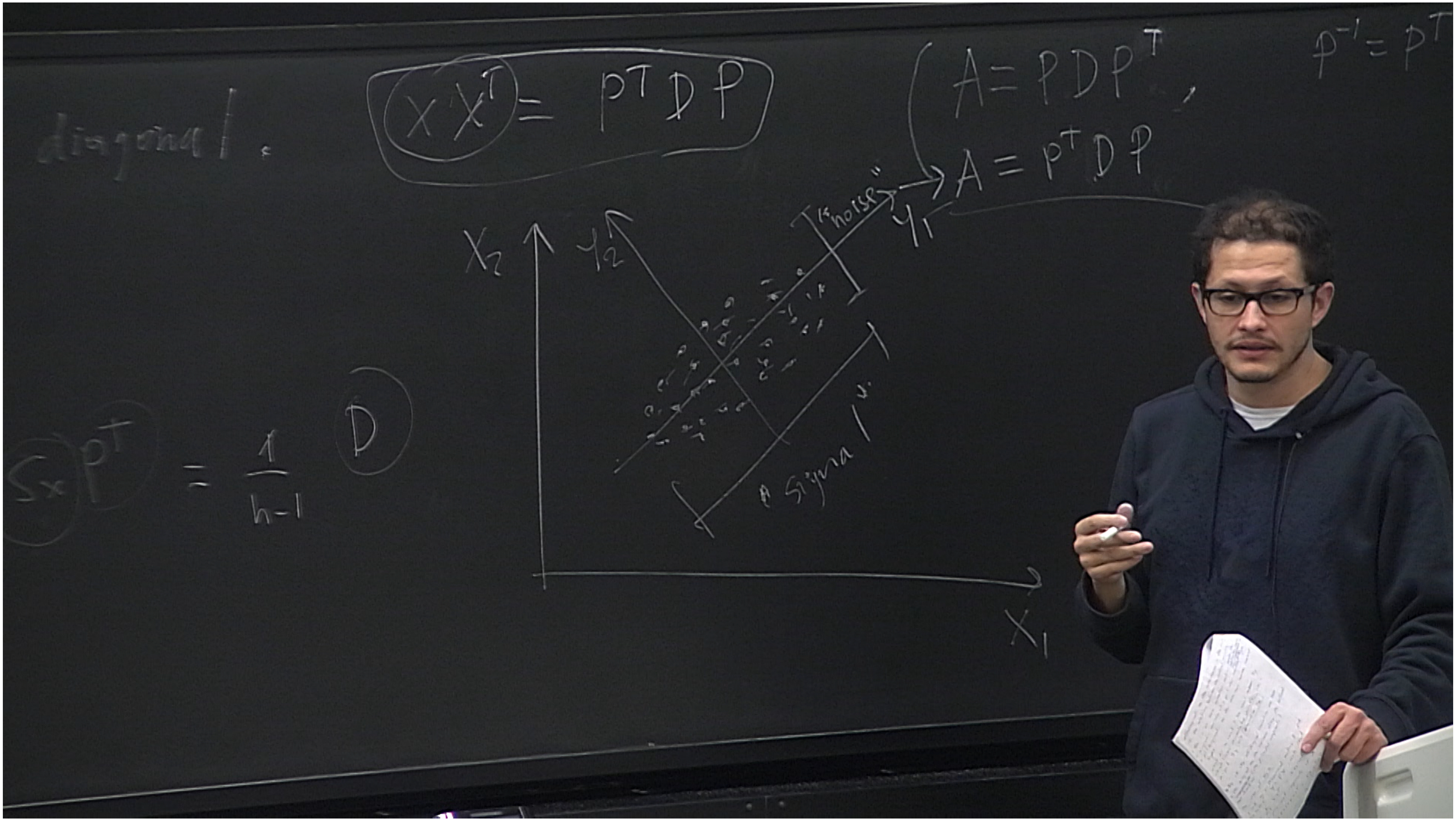

is diagonal.
$XX^T = P^T D P$
$A = PDP^T$,
$A = P^T D P$
$P^{-1} = P^T$
$S_x P^T$

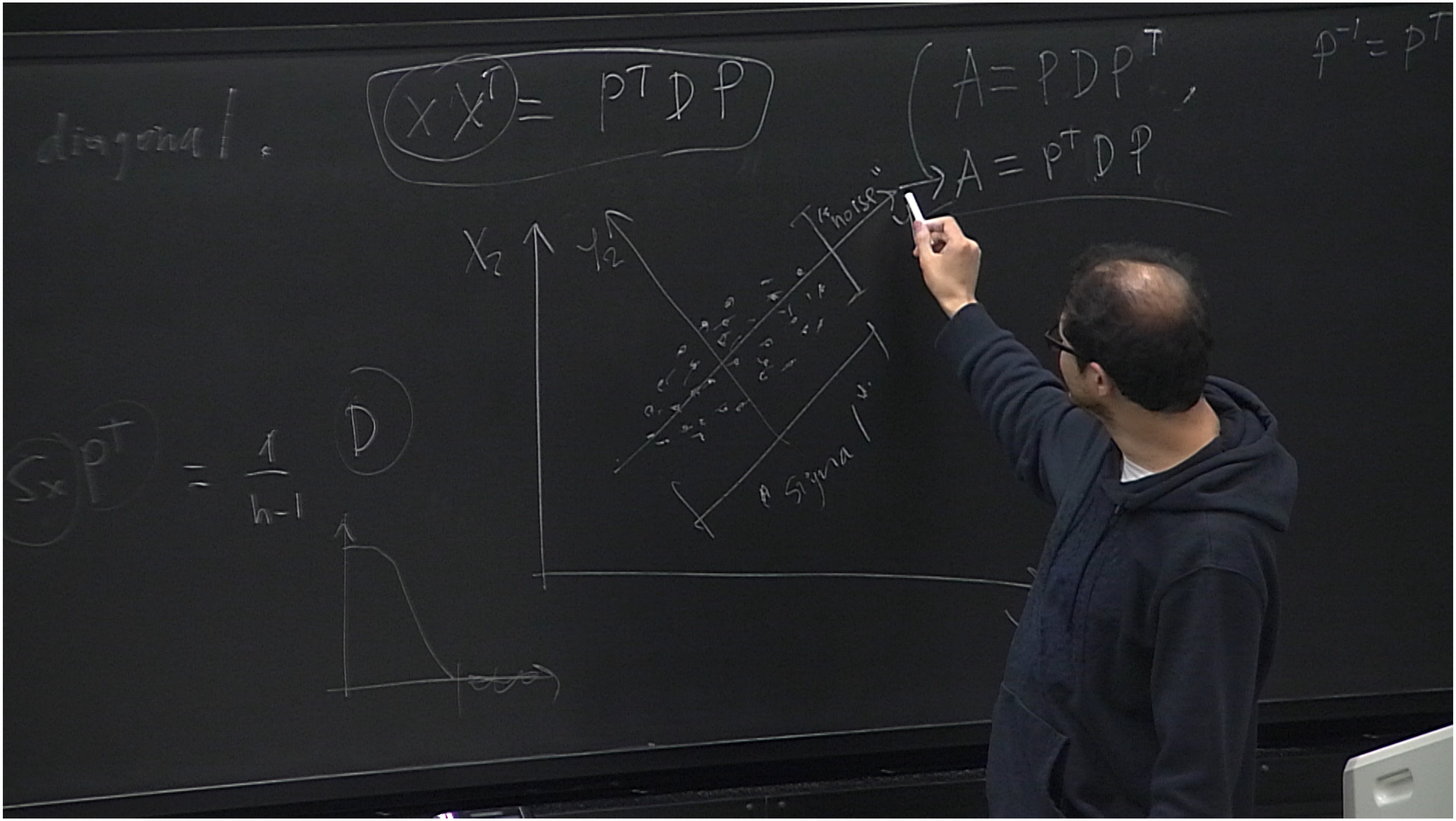

such that $S_Y = \frac{1}{n-1} YY^T$ is diagonal.
$XX^T = P^T D P$
$A = PDP^T$
$A = P^T D P$
principal components of X.
$PX(PX)^T = \frac{PXX^TP^T}{n-1} = P S_X P^T = \frac{1}{n-1} D$

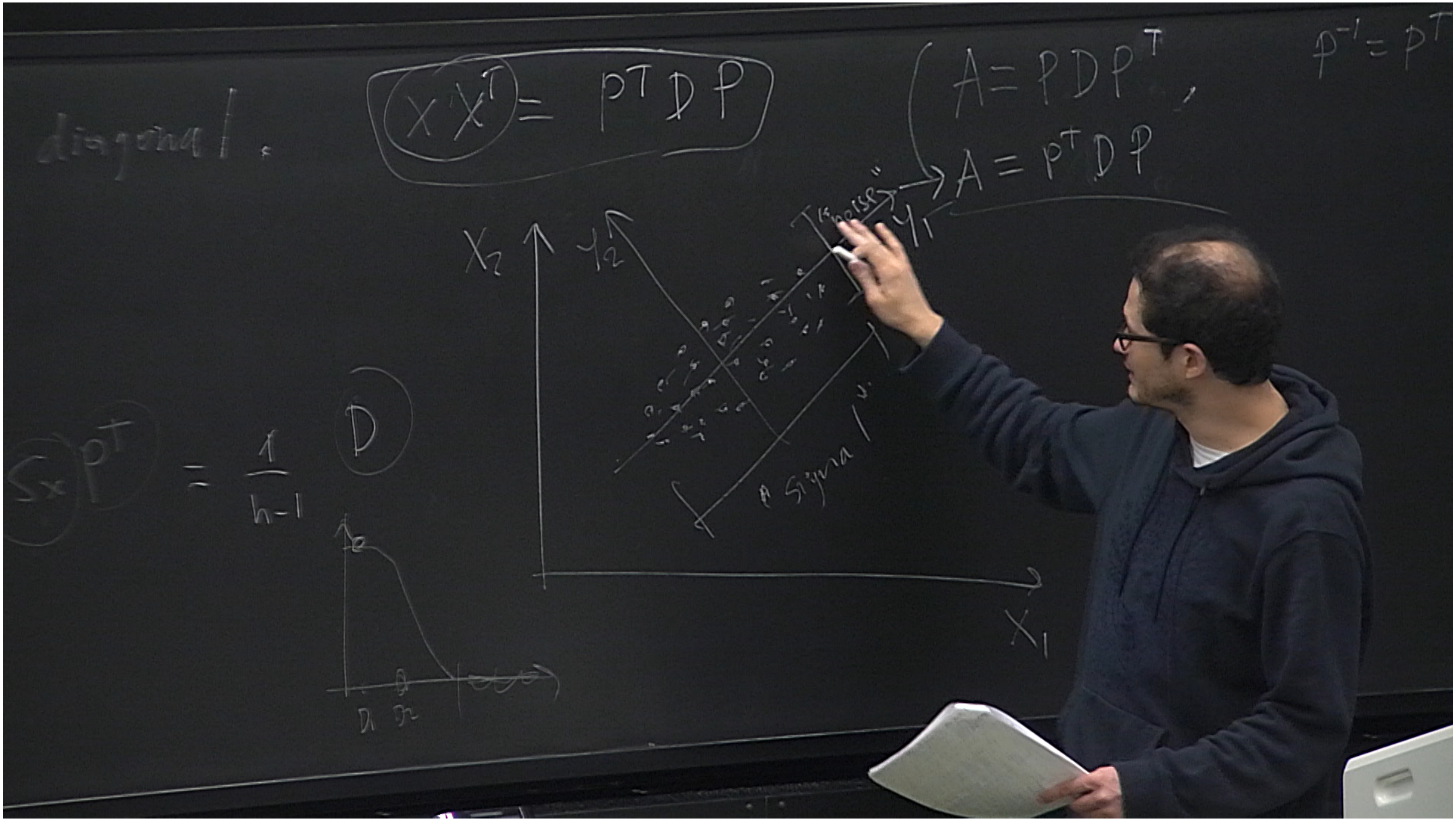

that Sy = 1/(n-1) YY^T is diagonal.
XX^T = P^T D P
A = PDP^T,
A = P^T D P
components of X.
= PXX^T P^T / (n-1) = P Sx P^T = 1/(n-1) D
X2

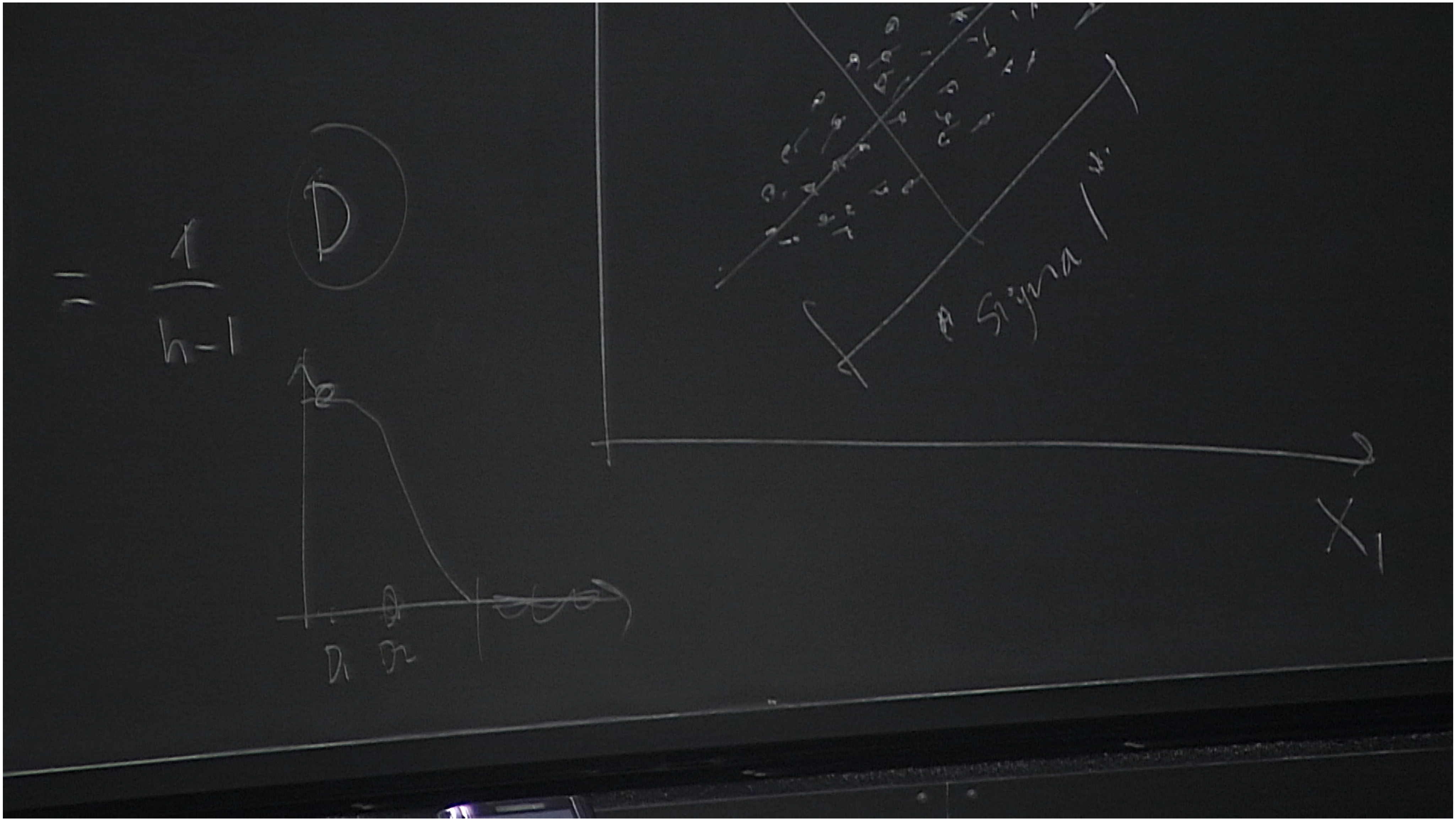

YY^T is diagonal.
that $S_Y = \frac{1}{n-1}$
components of X.
$= \frac{P(XX^T)P^T}{n-1} = PS_xP^T = \frac{1}{n-1}$ D
$XX^T = P^T D P$
$A = PDP^T$,
$A = P^T D P$
X_2


that S_Y = 1/(n-1) YY^T is diagonal.
components of X.
XX^T = P^T D P
A = PDP^T,
A = P^T D P
"noise"
"signal"
X_2
Y_2
Y_1
X_1
= 1/(n-1) D

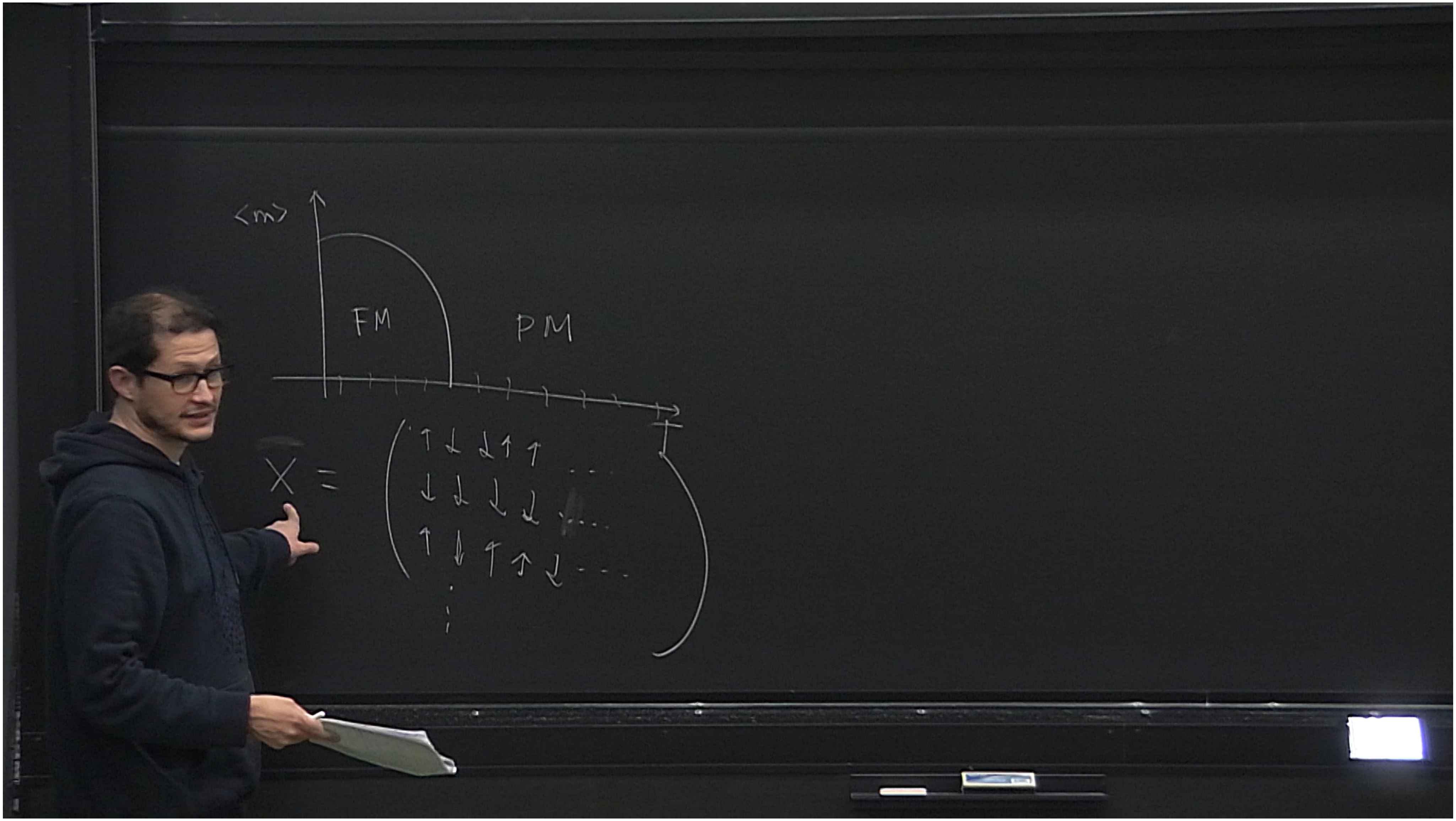

diagonal
$XX^T = P^T D P$
$A = PDP^T$
$A = P^T D P$
$P^{-1} = P^T$
"noise"
a signal
x_2
y_2
y_1
x_1
$S_x P^T = \frac{1}{n-1} D$


diagonal.
XX^T = P^T D P
A = PDP^T
A = P^T D P
P^-1 = P^T
X_2
Y_2
"noise"
signal
S_x P^T = 1/(n-1) D

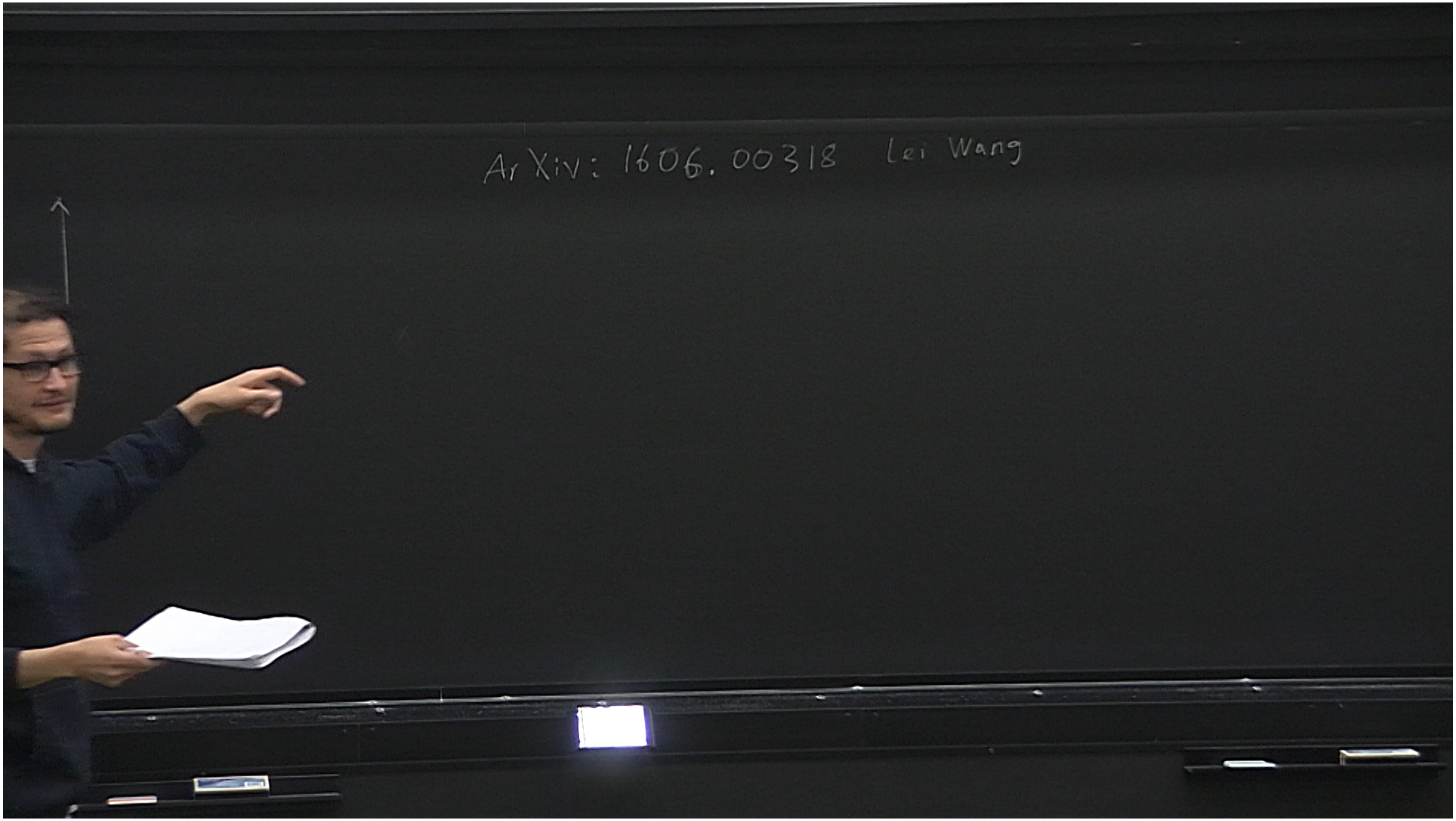

diagonal
$X X^T = P^T D P$
$A = P D P^T$
$A = P^T D P$
$P^{-1} = P^T$
noise
Y_1
Y_2
X_1
X_2
signal
$S_x P^T = \frac{1}{n-1} D$
D_1 D_2

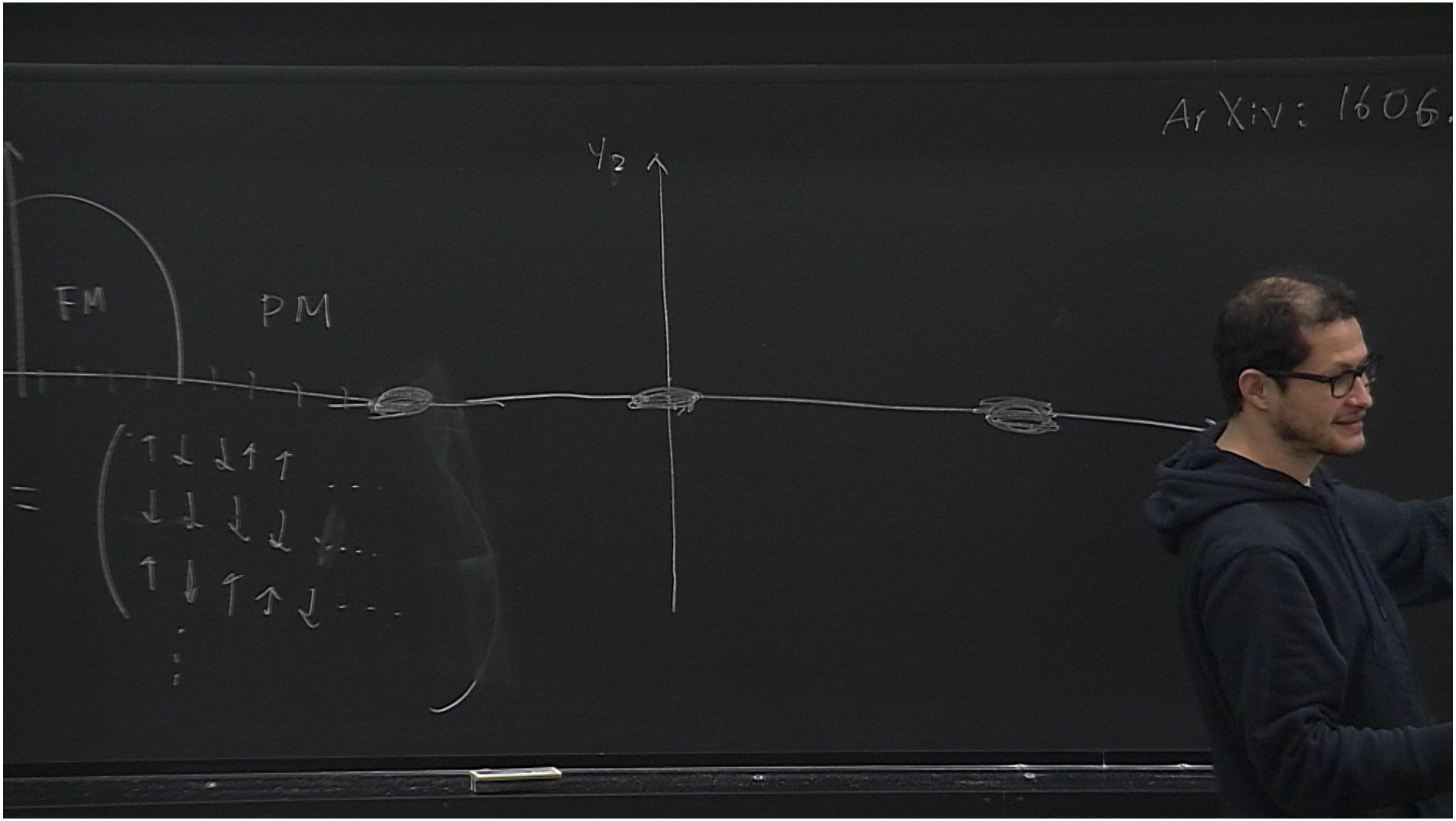


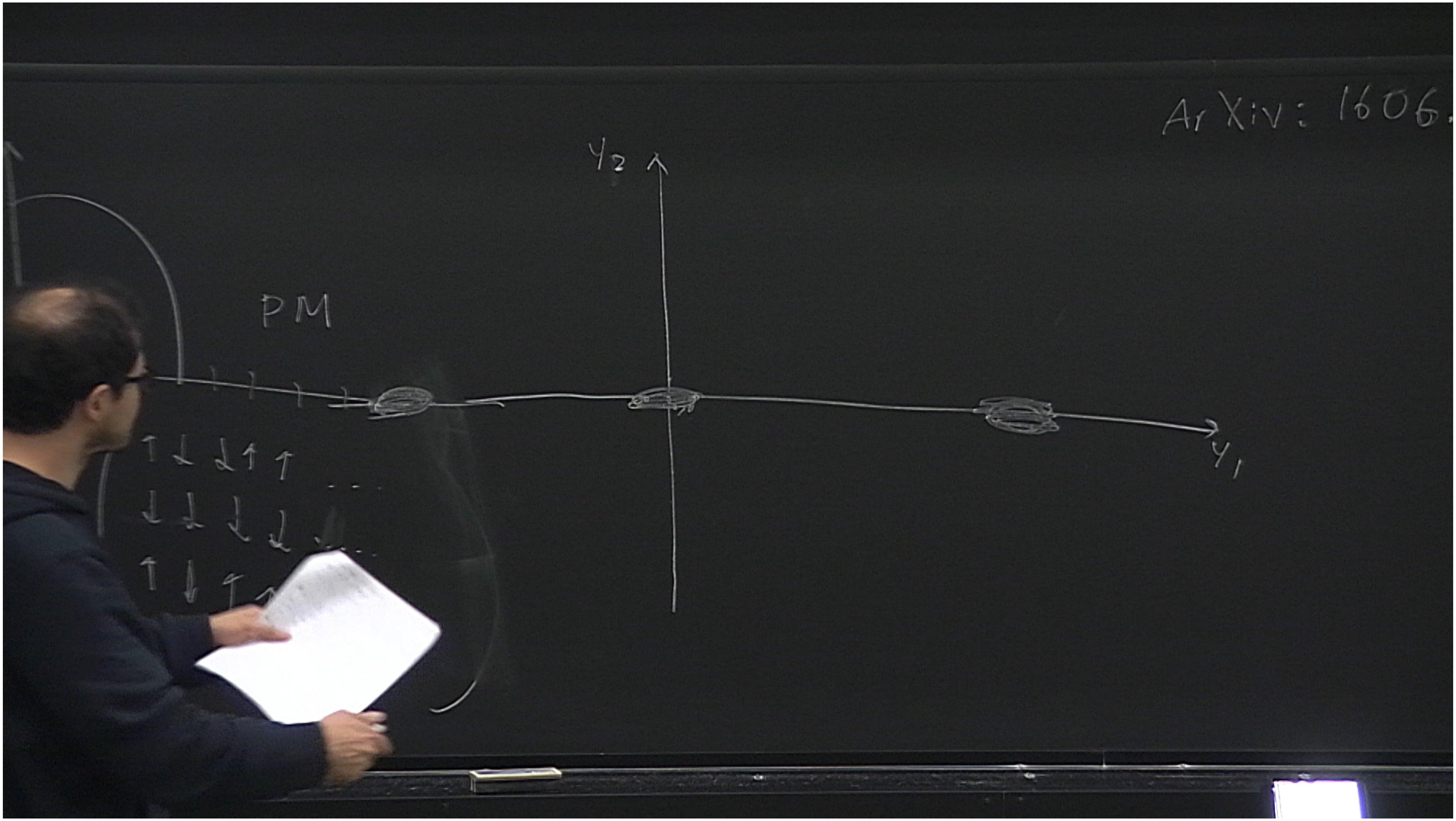
Sigma
X_1

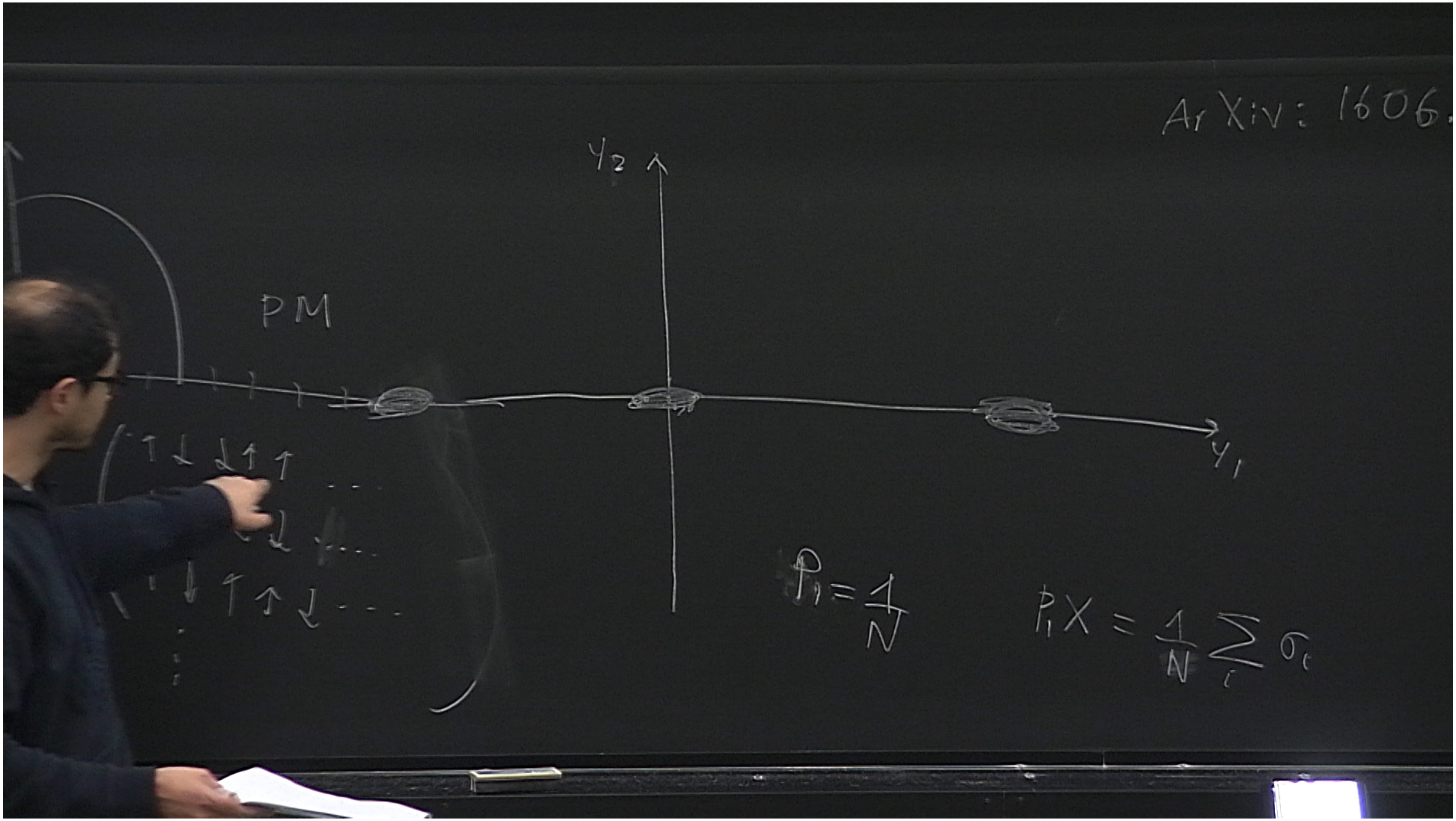


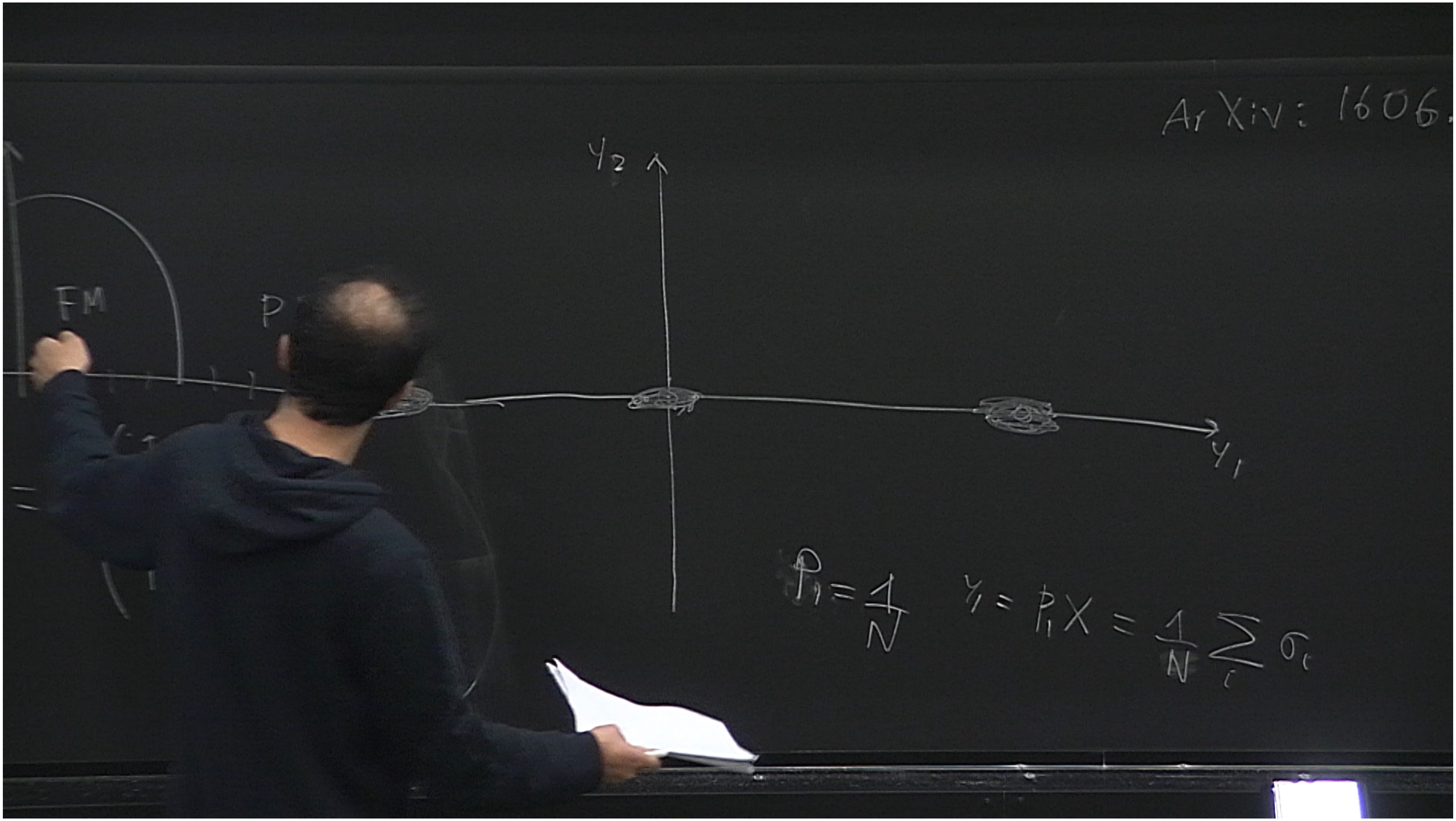
⟨m⟩
FM
PM
T
X =

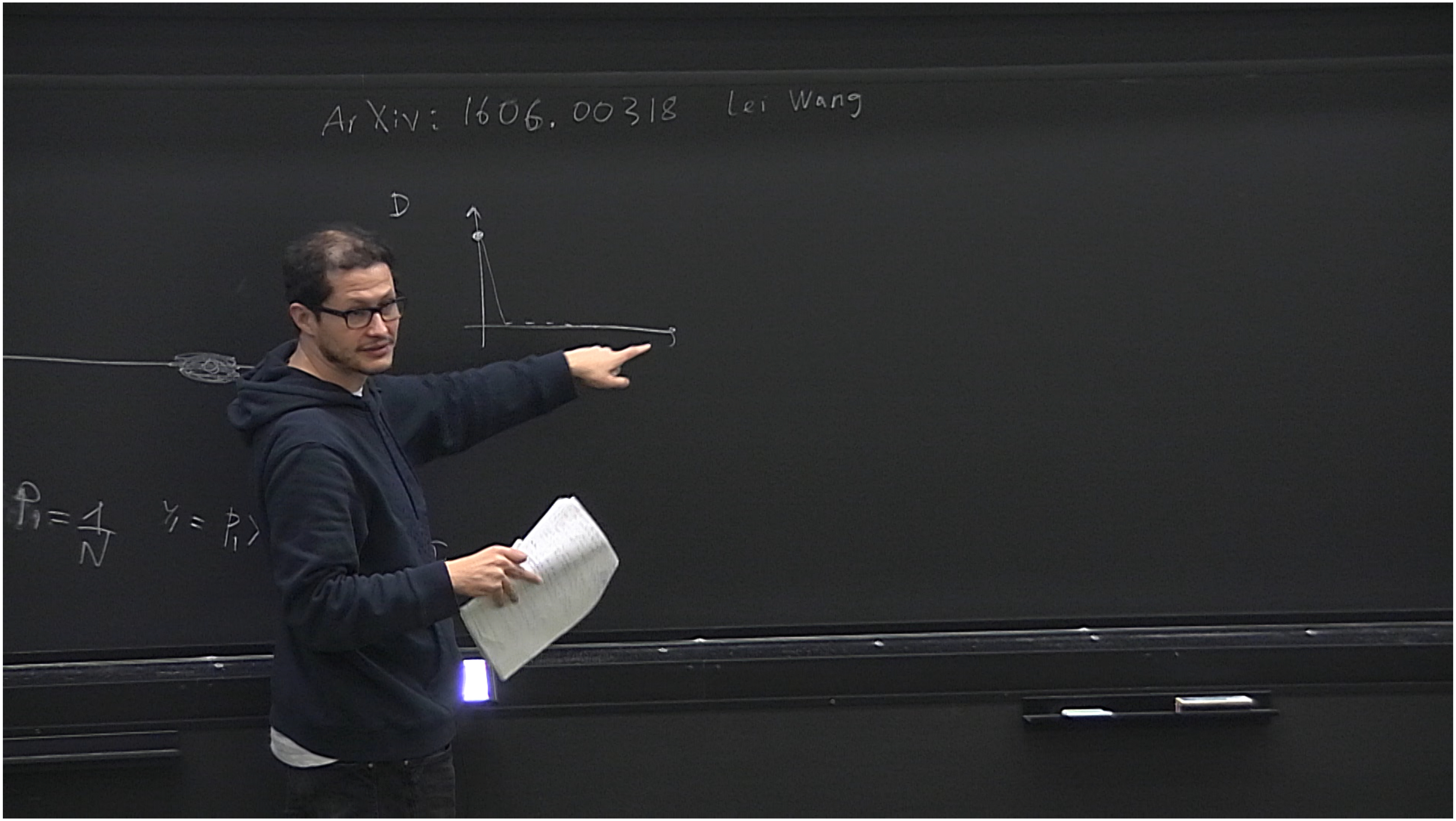

ArXiv: 1606.00318 Lei Wang

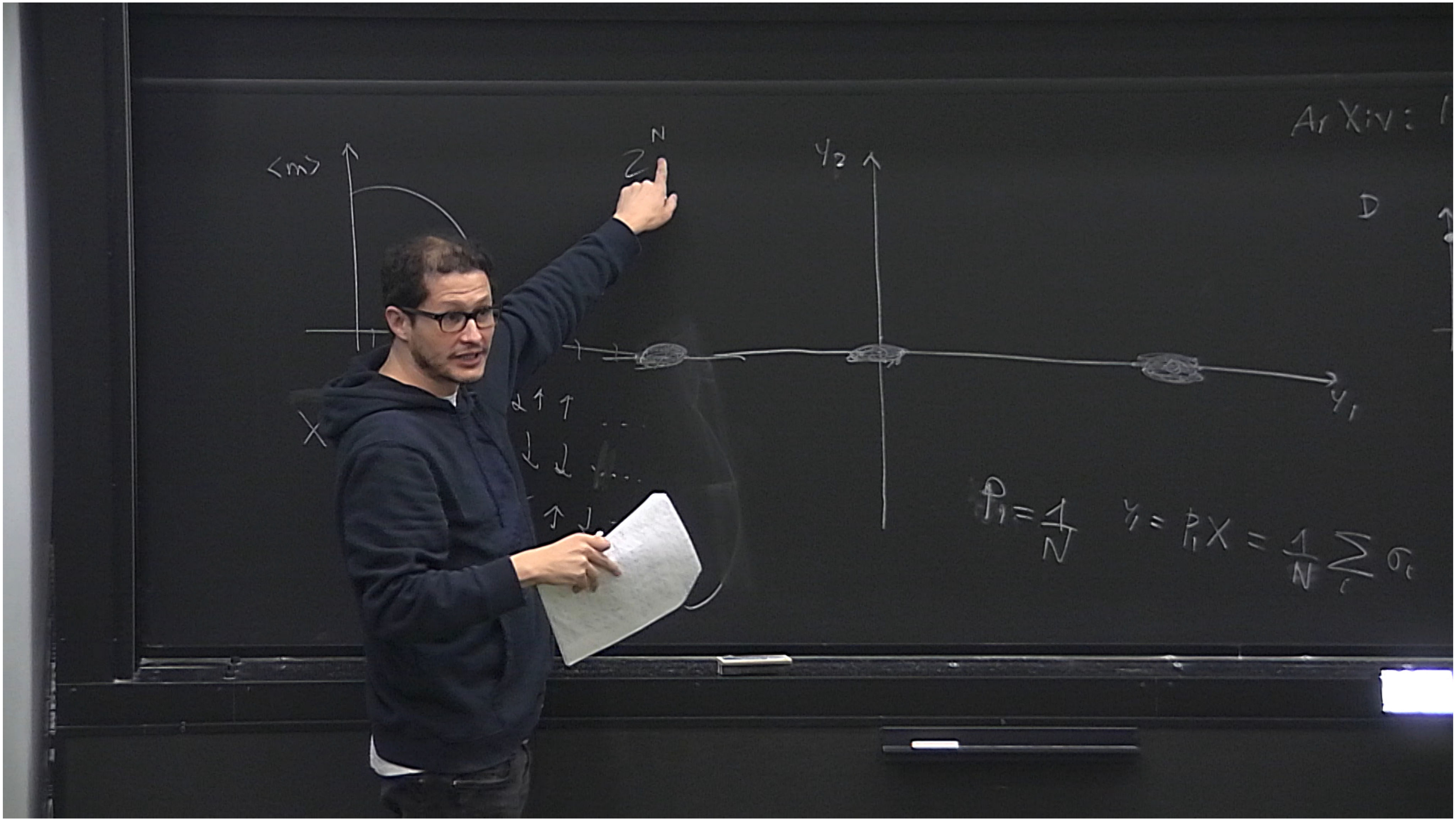

ArXiv: 1606.
y_2
FM
PM


ArXiv: 1606.
PM
y_2
y_1


ArXiv: 1606.
y_2
PM
y_1
$P_1 = \frac{1}{N}$
$P_1 X = \frac{1}{N} \sum_i \sigma_i$


ArXiv: 1606.
y2
y1
FM
P
P1 = 1/N
y1 = P1X = 1/N Σi σi


ArXiv: 1606.00318 Lei Wang
D


ArXiv:
⟨m⟩
X
2^N
y_2
y_1
D
$P_1 = \frac{1}{N}$
$y_1 = P_1 X = \frac{1}{N}\sum_i \sigma_i$