Title: Lecture - Scientific Machine Learning, PHYS 777

Speakers: Mohammad Kohandel

Collection/Series: Scientific Machine Learning (Elective), PHYS 777, February 23 - March 27, 2026

Subject: Condensed Matter, Other

Date: March 11, 2026 - 2:15 PM

URL: https://pirsa.org/26030046

Lecture 8

- Workflow for Training NNs
- Including Physics in NNs
- Physics-informed NNs

- Practical workflow

1) Data Pre-processing

- Practical workflow

1) Data Pre-processing

- Feature Normalization

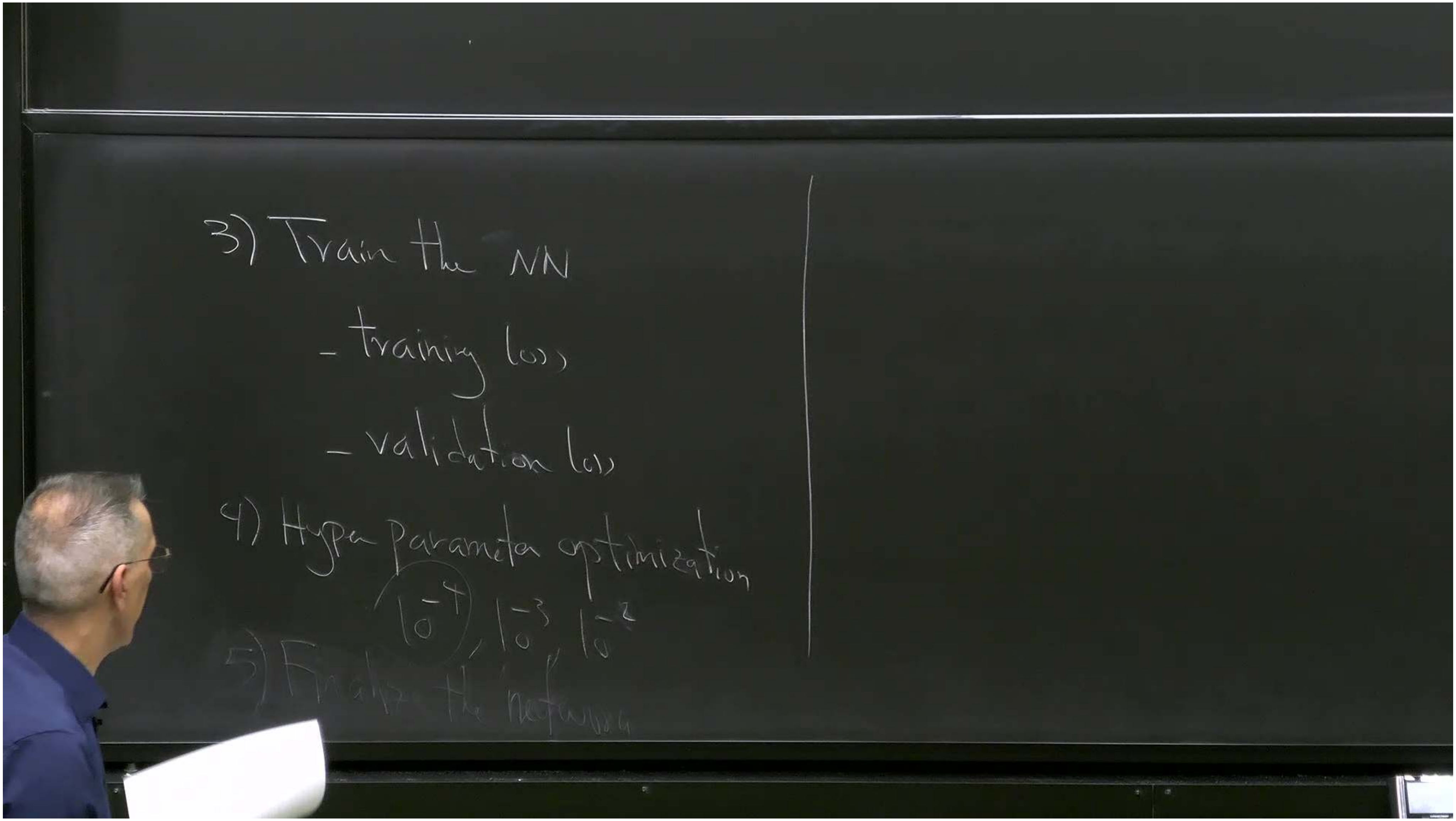
(Batch normalization $\longrightarrow$ in the intermediate activations)

2) Initial model configuration

- Network architecture (number of layers and neurons)
- Activation func.
- Loss function
- optimization algorithm (e.g. mini-batch gradient descent)
- Learning rate
- Epochs

the)

tivations


3) Train the NN
- training loss
- validation loss
4) Hyper parameter optimization
10^-4, 10^-3, 10^-2
5) Finalize the network

How can we include physics into a NN?

1) Physics in the loss function

$u(x,t)$: unknown physical quantity

$\hookrightarrow$ satisfies some governing equation into a NN?

$$u(\vec{x},t) \simeq u_{\theta}(\vec{x},t)$$

θ ↳ Parameters of the NN

n

ntly

quation

$\vec{x}$, t $\longrightarrow$ NN $\longrightarrow u_{\theta}(\vec{x},t)$

Input

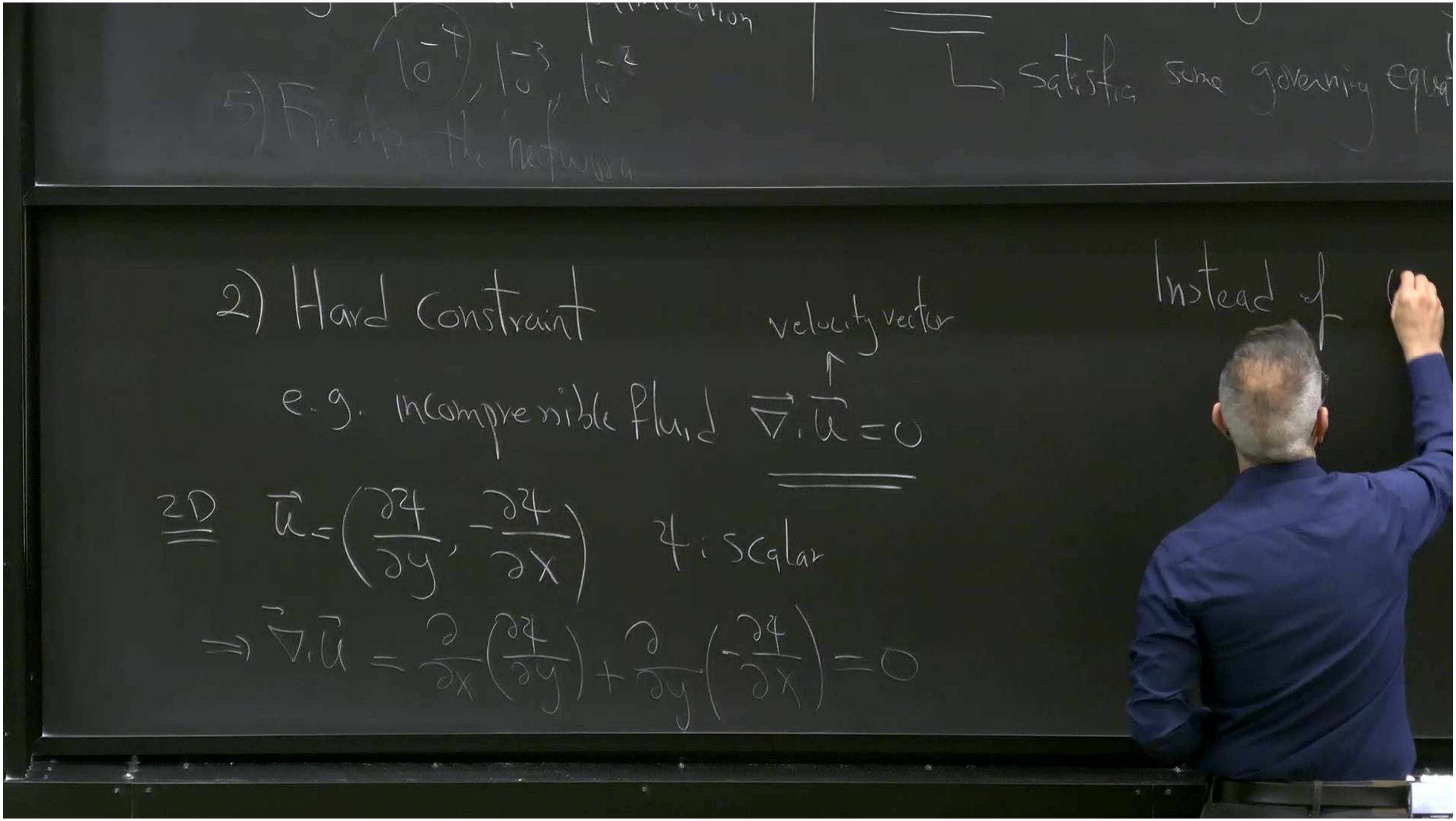
(i) Data: $\{(\vec{x}_i, t_i, u_i^{obs})\}_{i=1}^{N}$

(ii) Physics Law

$$L = \frac{1}{N}\sum_{i=1}^{N}\left[u_i^{obs} - \hat{u}_i\right]^2 + \lambda \, L_{phys}$$

$\nabla \cdot u = 0$

(soft constraint)



$10^{-4}, 10^{-3}, 10^{-2}$
5) Finalize the network
↳ satisfies some governing equat
2) Hard Constraint
velocity vector
e.g. incompressible fluid $\vec{\nabla}\cdot\vec{u} = 0$
2D $\vec{u} = \left(\frac{\partial \psi}{\partial y}, -\frac{\partial \psi}{\partial x}\right)$ ψ : scalar
$\Rightarrow \vec{\nabla}\cdot\vec{u} = \frac{\partial}{\partial x}\left(\frac{\partial \psi}{\partial y}\right) + \frac{\partial}{\partial y}\left(-\frac{\partial \psi}{\partial x}\right) = 0$
Instead of

...own physical quantity t $u_\theta(x, t)$ (ii) Physics Law

... some governing equation Input $L = \frac{1}{N}\sum_{i=1}^{N}\left[u_i^{obs} - \hat{u}_i\right]^2 + \lambda L_{Phys}$ $\nabla . u = 0$

(soft constraint)

Instead of $u_\theta \simeq u(\vec{x}, t)$, we use

$\psi_\theta \simeq \psi(\vec{x}, t)$

$$\begin{array}{l} t \\ \vec{x} \\ \text{Input} \end{array} \longrightarrow NN \longrightarrow \psi_\theta(\vec{x}, t) \Rightarrow \begin{array}{l} u_x = \frac{\partial \psi}{\partial y} \Rightarrow \hat{\nabla} . \hat{u} = 0 \\ u_y = -\frac{\partial \psi}{\partial x} \end{array}$$

(ii) Physics Law

equation

Input

$$L = \frac{1}{N}\sum_{i=1}^{N}\left[u_i^{obs} - \hat{u}_i\right]^2 + \lambda L_{phys}$$

$\nabla \cdot u = 0$

(soft constraint)

$u_\theta \simeq u(\vec{x}, t)$, we use

$(\vec{x}, t)$

$$\rho\left[\frac{\partial u}{\partial t} + (u \cdot \nabla) u\right] = -\nabla P + \mu \nabla^2 u$$

Physics Loss

$$\vec{\nabla} \cdot \vec{u} = 0$$

$$(\vec{x}, t) \Rightarrow u_x = \frac{\partial \psi}{\partial y} \Rightarrow \hat{\nabla} \cdot \hat{u} = 0$$

$$u_y = -\frac{\partial \psi}{\partial x}$$

$$L = L_{Data} + L_{phys.}$$

3) Physics in input features

choose a representation that makes learning easier

e.g. Buckingham - Pi theorem

$$\frac{dN}{dt} = rN\left(1-\frac{N}{K}\right) \rightsquigarrow \boxed{\frac{d\tilde{N}}{d\tilde{t}} = \tilde{N}\left(1-\tilde{N}\right)}$$

$$x = \frac{1}{2} g t^2 + v_0 t$$

Physical quantities x, v_0, g, t } $4-2=2 \rightarrow$ two dimensionless quantities Π_1, Π_2

Fundamental units: T, L $\Rightarrow \Pi_2 = f(\Pi_1)$

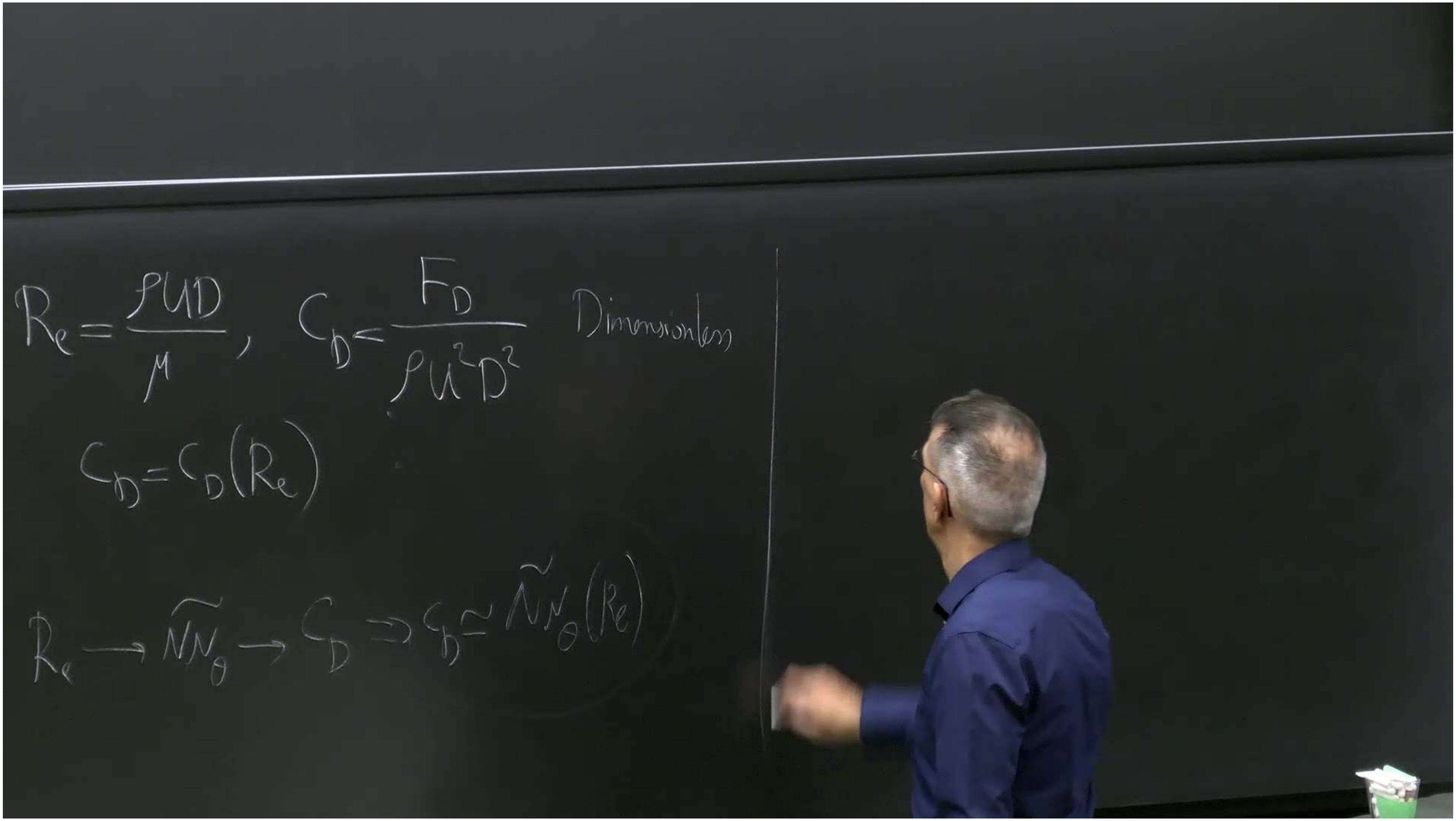
$$x = f(g, t, v_0)$$

Ex.

$$F_D = f(\rho, \mu, \underbrace{D, u})$$

drag force — density — viscosity — characteristic length and velocity

$$\begin{matrix}\rho \\ \mu \\ D \\ u\end{matrix} \rightarrow NN \rightarrow F_D \Rightarrow F_D = NN_\theta(\rho, \mu, D, u)$$


$R_e = \frac{\rho U D}{\mu}$, $C_D = \frac{F_D}{\rho U^2 D^2}$ Dimensionless
$C_D = C_D(R_e)$
$R_e \longrightarrow \widetilde{NN}_\theta \longrightarrow C_D \Rightarrow C_D \simeq \widetilde{NN}_\theta(R_e)$

$\frac{\rho U D}{\mu}$, $C_D = \frac{F_D}{\rho U^2 D^2}$ Dimensionless

$= C_D(R_e)$

$\widetilde{NN}_\theta \rightarrow C_D \Rightarrow C_D \simeq \widetilde{NN}_\theta(R_e)$

We can also apply symbolic Regression to find the eq.

PySR

Physics-informed NNs (PINNs) $\rightarrow$ soft-constraint

Physical law: $\underbrace{\mathcal{N}}_{\text{operator}}(\underbrace{u(\vec{x},t)}_{\text{field}}, \underbrace{\vec{\lambda}}_{\text{parameters}}) = 0 \Rightarrow \underline{\underline{\text{PDE}}}$

+ IC + BC

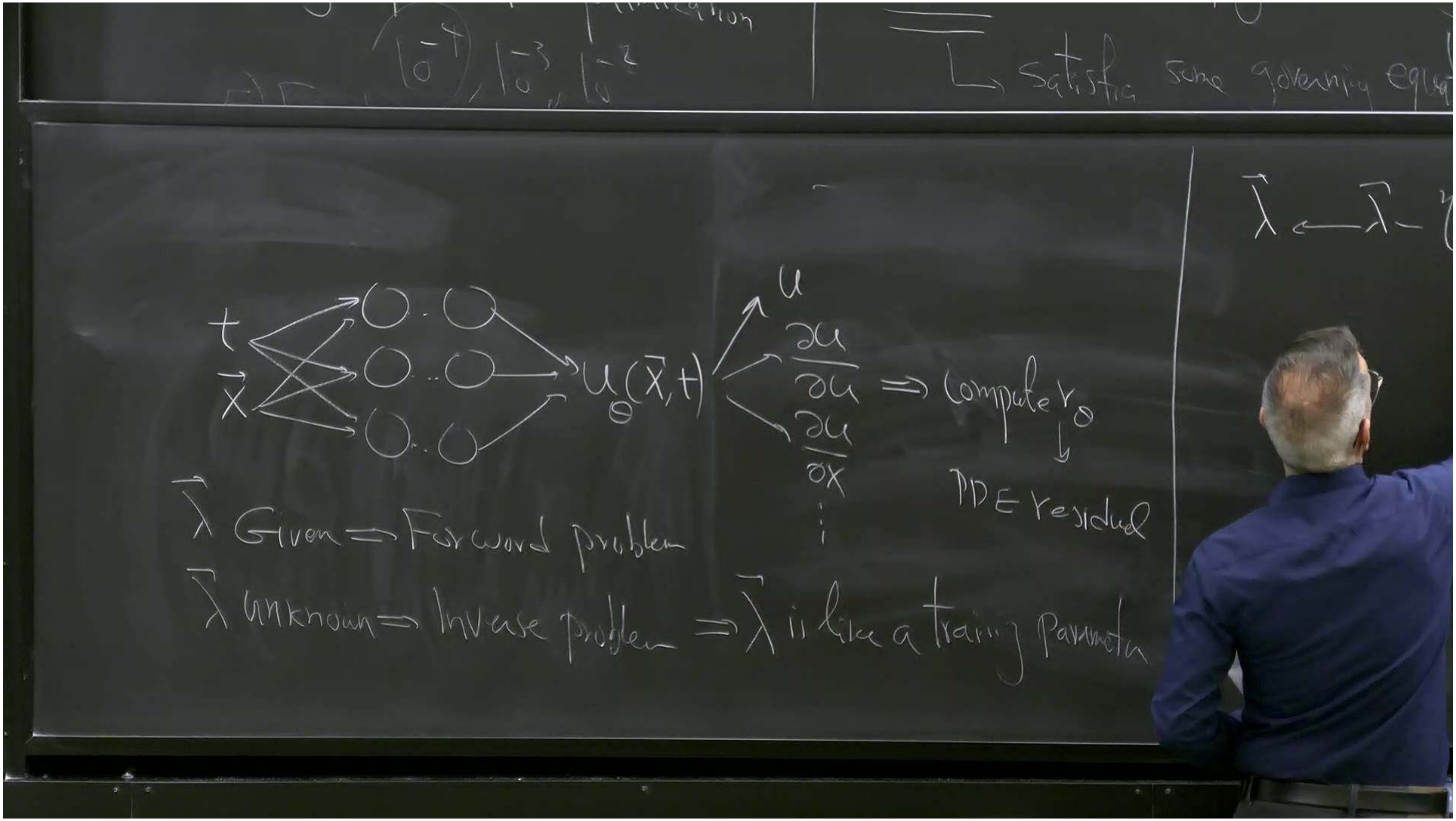
$$u(\vec{x},t) \simeq \underbrace{u_\theta(\vec{x},t)}_{NN}$$

collucation or virtual points

$$\partial_t u = \partial_{xx} u \Rightarrow r_\theta = \partial_t u - \partial_{xx} u$$

$$L = \frac{1}{N} \underbrace{\sum_{i=1}^{N} \left[u^{obs}(\vec{x},t) - u_\theta(\vec{x},t) \right]^2}_{\text{data loss}} + \sum_{i=1}^{M} \left[N(u_\theta(\vec{x},t), \vec{\lambda}) \right]^2 + \sum_{i=1}^{N_{IC}} \left[u_\theta(\vec{x},0) - u(\vec{x},0) \right]^2 + BC$$

$$\vec{x} \subset \Omega, \; t \in [0,T]$$



satisfies some governing equa
t
x
u_θ(x, t)
u
∂u/∂u
∂u/∂x
⇒ compute r_θ
PDE residual
λ Given ⇒ Forward problem
λ unknown ⇒ Inverse problem ⇒ λ is like a training parameter

physical quantity

governing equation

$t \longrightarrow NN \longrightarrow u_\theta(\vec{x}, t)$

Input

(i) Data, $\{(\vec{x}_i, t_i, u_i^{obs})\}_{i=1}^{N}$

(ii) Physics Law

$$L = \frac{1}{N}\sum^{N}\left[u_i^{obs} - \hat{u}_i\right]^2 + \lambda L_{Ph}$$

$\nabla \cdot u = 0$

$$\vec{\lambda} \longleftarrow \vec{\lambda} - \eta \frac{\partial L}{\partial \vec{\lambda}}$$

EX. Damped pendulum $\quad \theta'' + \gamma\theta' + \frac{g}{L}\sin\theta = 0$

$t \longrightarrow NN \longrightarrow \theta_\psi(t)$, $\psi \longrightarrow$ Parameters

g, L, γ Given

$$\Rightarrow L = L_{data} + \sum_{i=1}^{M}\left[\theta_\psi'' + \gamma\theta_\psi' + \frac{g}{L}\sin\theta_\psi\right]^2 + \sum_{i=1}^{N_c}\left[\theta_\psi(0) - \omega_0\right]^2$$

If γ is unknown $\Rightarrow \gamma \longleftarrow \gamma - \eta\frac{\partial L}{\partial \gamma}$

$10^{-4}, 10^{-3}, 10^{-2}$

5) Finalize the network

↳ satisfies some governing equat…

We assumed $F_{\text{damping}} \sim \theta'$

$$\theta'' + \frac{g}{L}\sin\theta + \underline{\underline{F_D(\theta, \theta')}} = 0$$

Data: $\{t_i, \theta_i, \theta'_i\}_{i=1}^{N}$

$\theta = \Theta_{\psi}(t)$, $F_D = NN_{\phi}$

Instead of u_θ

$u_\theta \simeq u(\vec{x}$

t

$\vec{x} \longrightarrow NN \longrightarrow u_\theta(\vec{x}$

Input