Physics-informed neural networks

E.g., Navier-Stokes equation

Given training data of u, v, find $\boldsymbol{\lambda_1, \lambda_2}$

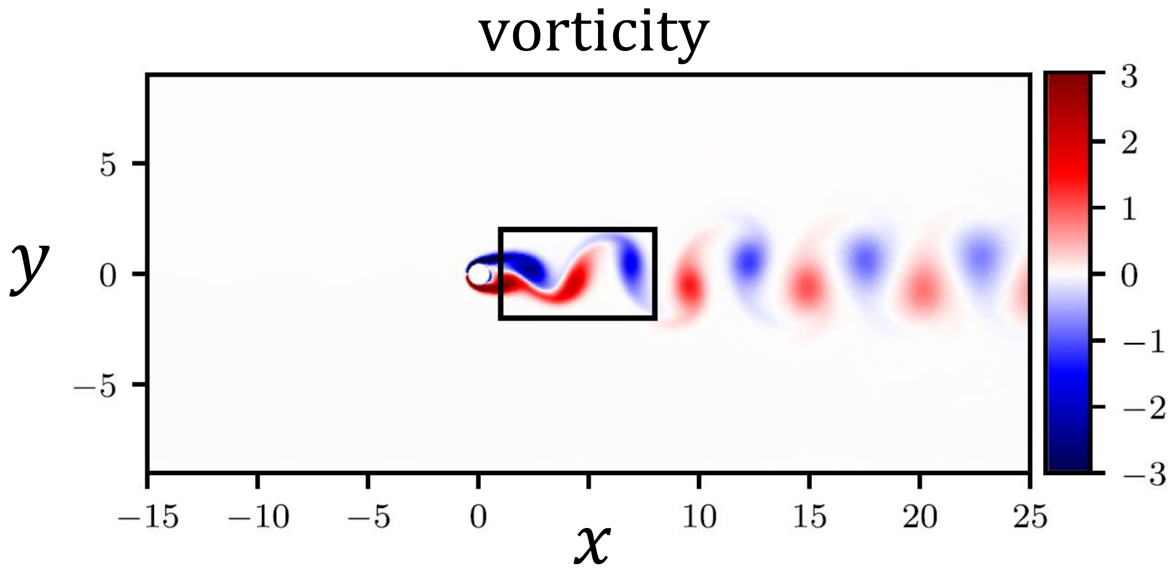
Problem statement

$$u_t + \lambda_1 (u u_x + v u_y) = -p_x + \lambda_2 (u_{xx} + u_{yy}),$$
$$v_t + \lambda_1 (u v_x + v v_y) = -p_y + \lambda_2 (v_{xx} + v_{yy}),$$

$$u_x + v_y = 0 \longrightarrow u = \psi_y, \quad v = -\psi_x$$

λ_1, λ_2: unknown parameters to be identified.
ψ: the stream function.
u, v: velocities
p: pressure

Ground truth from numerical simulation:

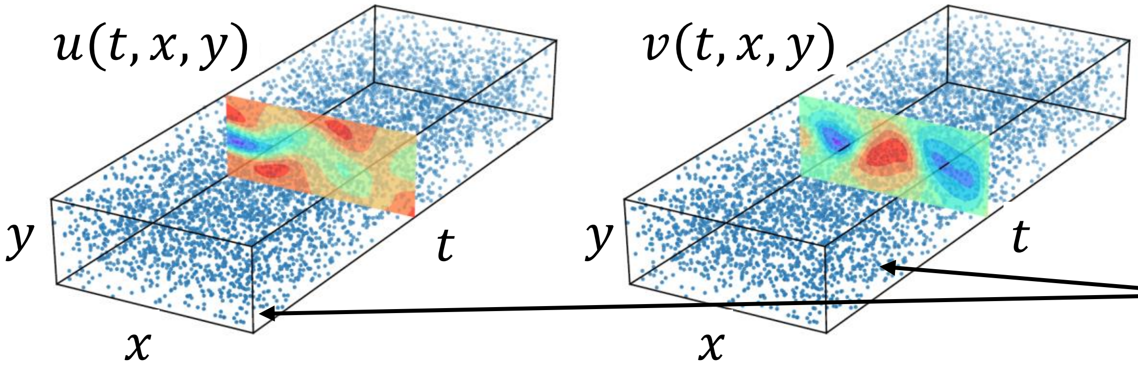


NN input: x, y, t
NN output: ψ, p
NN architecture: 9 layers 20 neurons per layer
Training parameters: weights, biases, λ_1, λ_2

Q: Why do we choose NN input/output this way?

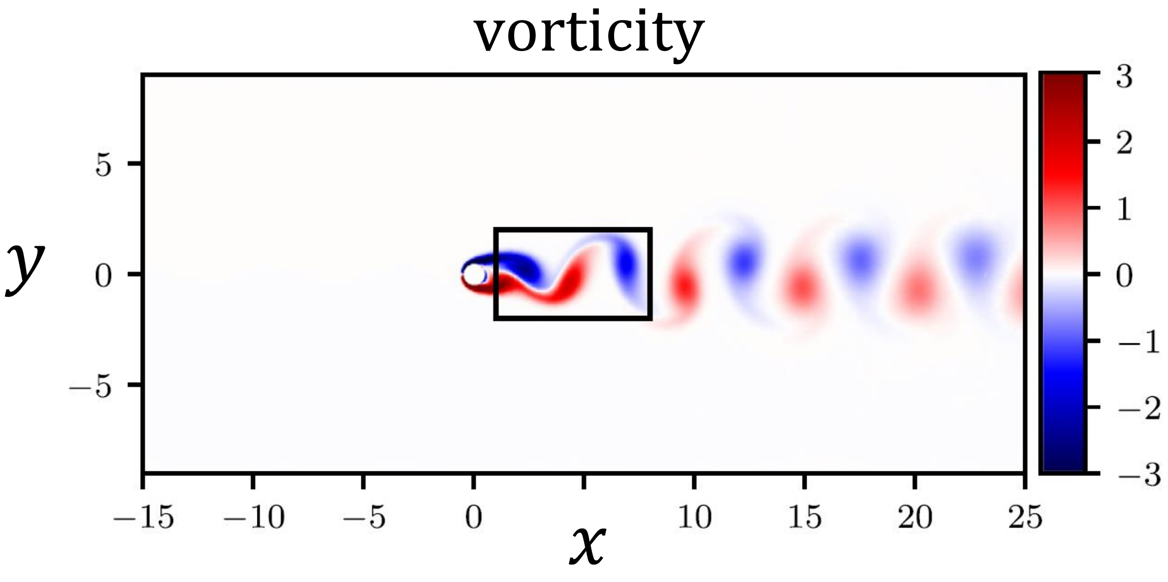
E.g., Navier-Stokes equation

Given training data of u, v, find $\boldsymbol{\lambda_1, \lambda_2}$



Training data: 5000 pts of u, v within the domain
Collocation points: 5000 pts within the domain

Ground truth from numerical simulation:

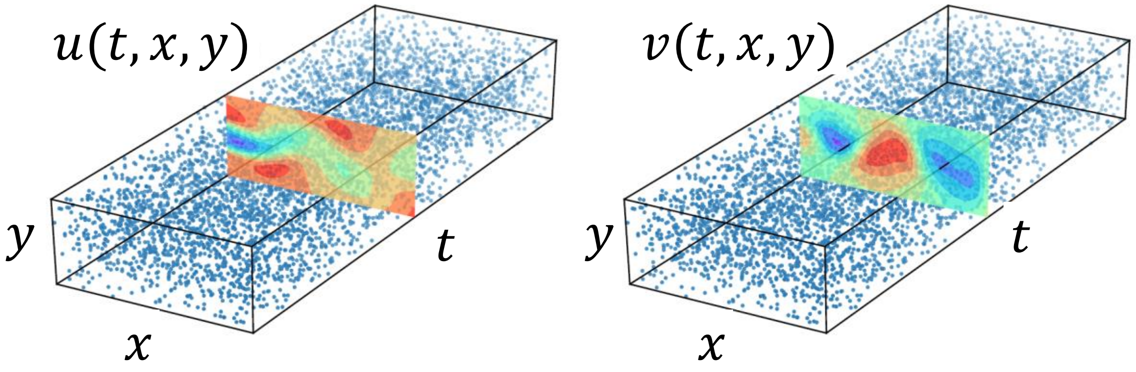


NN input: x, y, t
NN output: ψ, p
NN architecture: 9 layers 20 neurons per layer
Training parameters: weights, biases, λ_1, λ_2

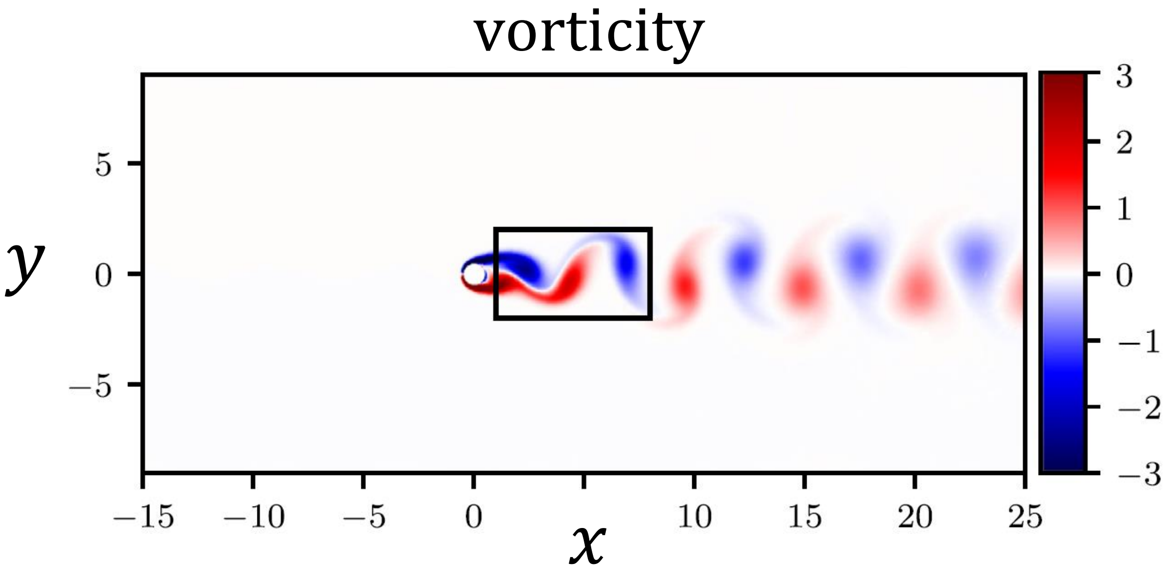
Raissi et al (2019)

E.g., Navier-Stokes equation

Given training data of u, v, find $\boldsymbol{\lambda_1, \lambda_2}$



Ground truth from numerical simulation:



Training data (from ground truth):

$\{t^i, x^i, y^i, u^i, v^i\}_{i=1}^N \qquad N = 5{,}000$

Collocation points:

$\{t^i, x^i, y^i\}_{i=1}^N \qquad N = 5{,}000$

Physics equations:

$$f := u_t + \lambda_1(uu_x + vu_y) + p_x - \lambda_2(u_{xx} + u_{yy})$$
$$g := v_t + \lambda_1(uv_x + vv_y) + p_y - \lambda_2(v_{xx} + v_{yy})$$

Loss function:

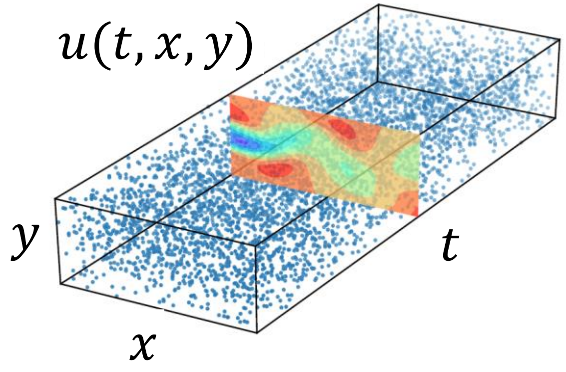
$$MSE := \frac{1}{N}\sum_{i=1}^{N}\left(|u(t^i, x^i, y^i) - u^i|^2 + |v(t^i, x^i, y^i) - v^i|^2\right)$$

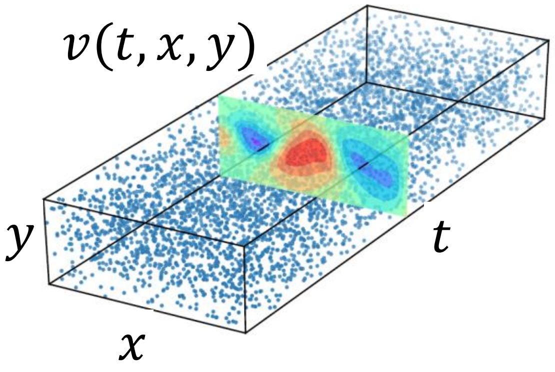
Data loss

$$+ \frac{1}{N}\sum_{i=1}^{N}\left(|f(t^i, x^i, y^i)|^2 + |g(t^i, x^i, y^i)|^2\right)$$

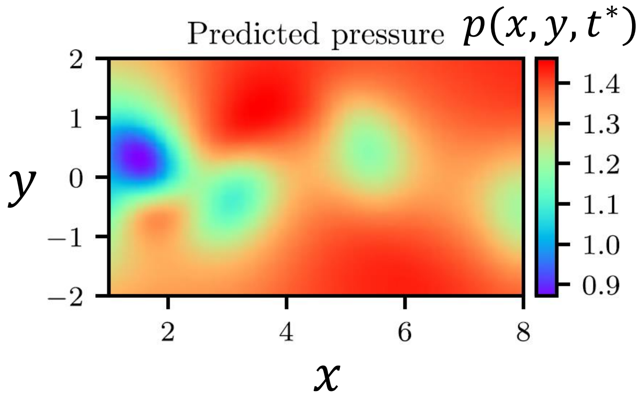
Equation loss

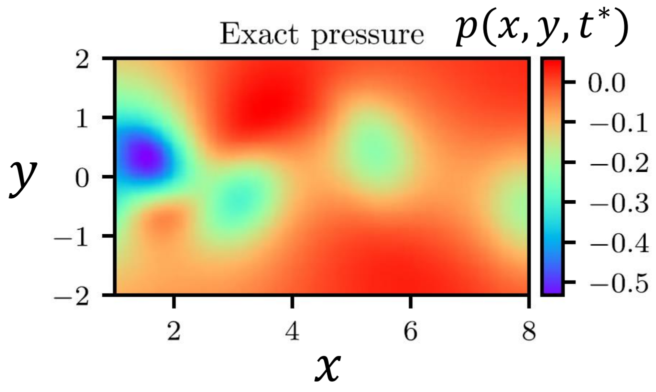
E.g., Navier-Stokes equation





Correct PDE	$u_t + (uu_x + vu_y) = -p_x + 0.01(u_{xx} + u_{yy})$ $v_t + (uv_x + vv_y) = -p_y + 0.01(v_{xx} + v_{yy})$
Identified PDE (clean data)	$u_t + 0.999(uu_x + vu_y) = -p_x + 0.01047(u_{xx} + u_{yy})$ $v_t + 0.999(uv_x + vv_y) = -p_y + 0.01047(v_{xx} + v_{yy})$
Identified PDE (1% noise)	$u_t + 0.998(uu_x + vu_y) = -p_x + 0.01057(u_{xx} + u_{yy})$ $v_t + 0.998(uv_x + vv_y) = -p_y + 0.01057(v_{xx} + v_{yy})$





Given training data of u, v, find $\boldsymbol{\lambda_1, \lambda_2}$

Training data (from ground truth):

$\{t^i, x^i, y^i, u^i, v^i\}_{i=1}^N$ $\quad N = 5{,}000$

Collocation points:

$\{t^i, x^i, y^i\}_{i=1}^N$ $\quad N = 5{,}000$

Physics equations:

$$f := u_t + \lambda_1(uu_x + vu_y) + p_x - \lambda_2(u_{xx} + u_{yy})$$
$$g := v_t + \lambda_1(uv_x + vv_y) + p_y - \lambda_2(v_{xx} + v_{yy})$$

Loss function:

$$MSE := \frac{1}{N}\sum_{i=1}^{N}\left(|u(t^i, x^i, y^i) - u^i|^2 + |v(t^i, x^i, y^i) - v^i|^2\right)$$

Data loss

$$+ \frac{1}{N}\sum_{i=1}^{N}\left(|f(t^i, x^i, y^i)|^2 + |g(t^i, x^i, y^i)|^2\right)$$

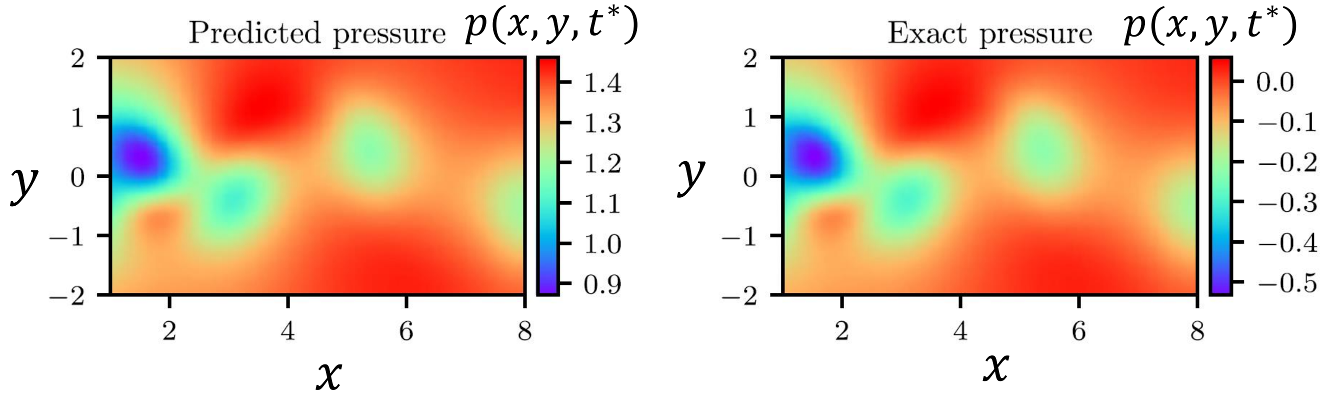
Equation loss

Navier-Stokes equation coding exercise

- TF1.14

Pause and Ponder

- Why is pressure prediction off by a constant?



- Do the input training data need to be sampled at the same inputs as the collocation points $\{t^i, x^i, y^i\}_i^N$?
- How to check if the PINN overfits the training data and collocation pts?

E.g., Nonlinear Schrodinger equation

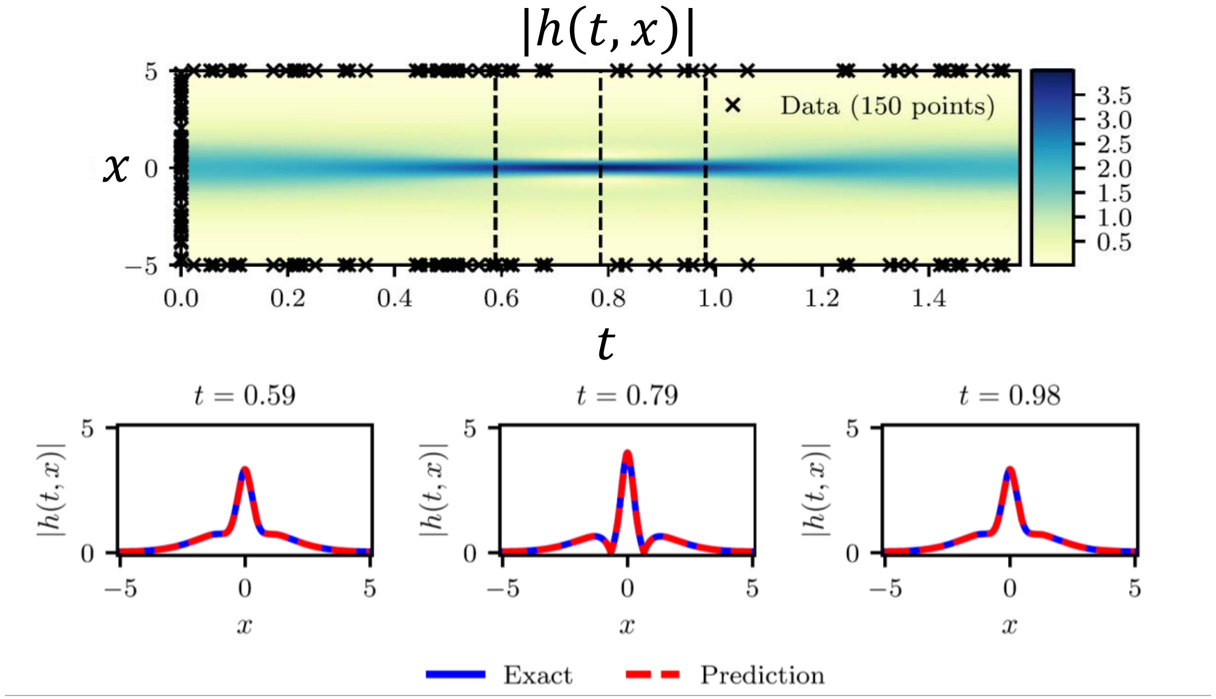
Problem statement

$$ih_t + 0.5h_{xx} + |h|^2 h = 0, \quad x \in [-5, 5], \quad t \in [0, \pi/2],$$

$$IC: \; h(0, x) = 2\,\mathrm{sech}(x),$$

$$BC: \; h(t, -5) = h(t, 5),$$

$$h_x(t, -5) = h_x(t, 5),$$



Training data (from ground truth):

$$\{x_0^i, h_0^i\}_{i=1}^{N_0}$$

Collocation points:

$$\{t_b^i\}_{i=1}^{N_b} \qquad \{t_f^i, x_f^i\}_{i=1}^{N_f}$$

Physics equations:

$$f := ih_t + 0.5h_{xx} + |h|^2 h,$$

Loss function:

$$MSE_0 = \frac{1}{N_0}\sum_{i=1}^{N_0} |h(0, x_0^i) - h_0^i|^2,$$ (1) IC

$$MSE_b = \frac{1}{N_b}\sum_{i=1}^{N_b} \left(|h^i(t_b^i, -5) - h^i(t_b^i, 5)|^2 + |h_x^i(t_b^i, -5) - h_x^i(t_b^i, 5)|^2\right)$$

(2) BC

$$MSE_f = \frac{1}{N_f}\sum_{i=1}^{N_f} |f(t_f^i, x_f^i)|^2.$$ (3) Eqn

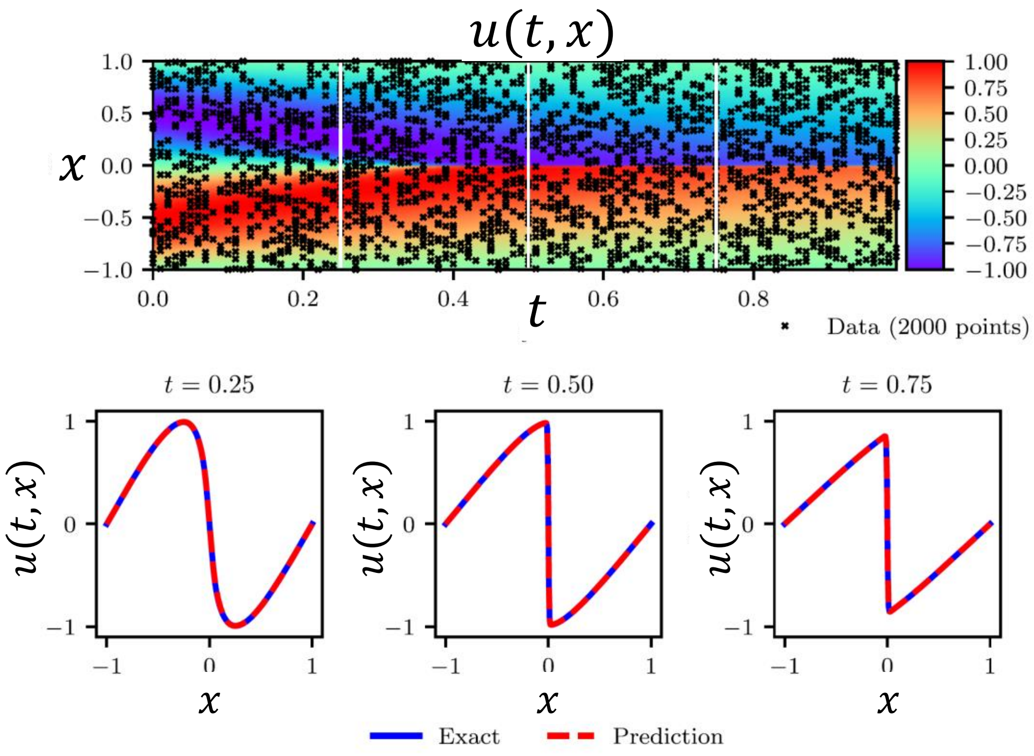
Raissi et al (2019)

E.g., Burgers' equation (identification)

Given training data of u, t, x find λ_1, λ_2

$$u_t + \lambda_1 u u_x - \lambda_2 u_{xx} = 0 \qquad x \in [-1, 1], \quad t \in [0, 1],$$

Correct PDE	$u_t + uu_x - 0.0031831u_{xx} = 0$
Identified PDE (clean data)	$u_t + 0.99915uu_x - 0.0031794u_{xx} = 0$
Identified PDE (1% noise)	$u_t + 1.00042uu_x - 0.0032098u_{xx} = 0$



Training data (from ground truth):

$\{t_u^i, x_u^i, u^i\}_{i=1}^N \qquad N = 2{,}000$

Collocation points:

$\{t_u^i, x_u^i\}_{i=1}^N \qquad N = 2{,}000$

Physics equations:

$$f := u_t + \lambda_1 u u_x - \lambda_2 u_{xx}$$

Loss function:

Data loss

$$MSE_u = \frac{1}{N_u} \sum_{i=1}^{N_u} |u(t_u^i, x_u^i) - u^i|^2$$

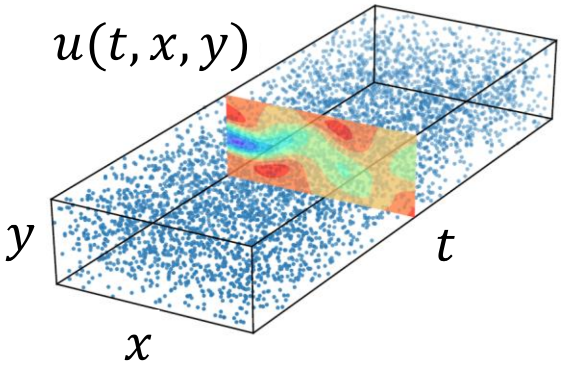
Data points

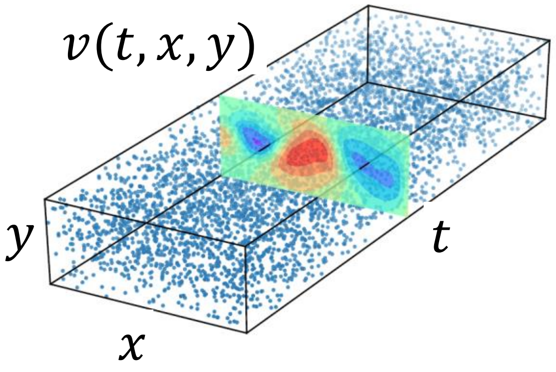
Equation loss

$$MSE_f = \frac{1}{N_f} \sum_{i=1}^{N_f} |f(t_f^i, x_f^i)|^2$$

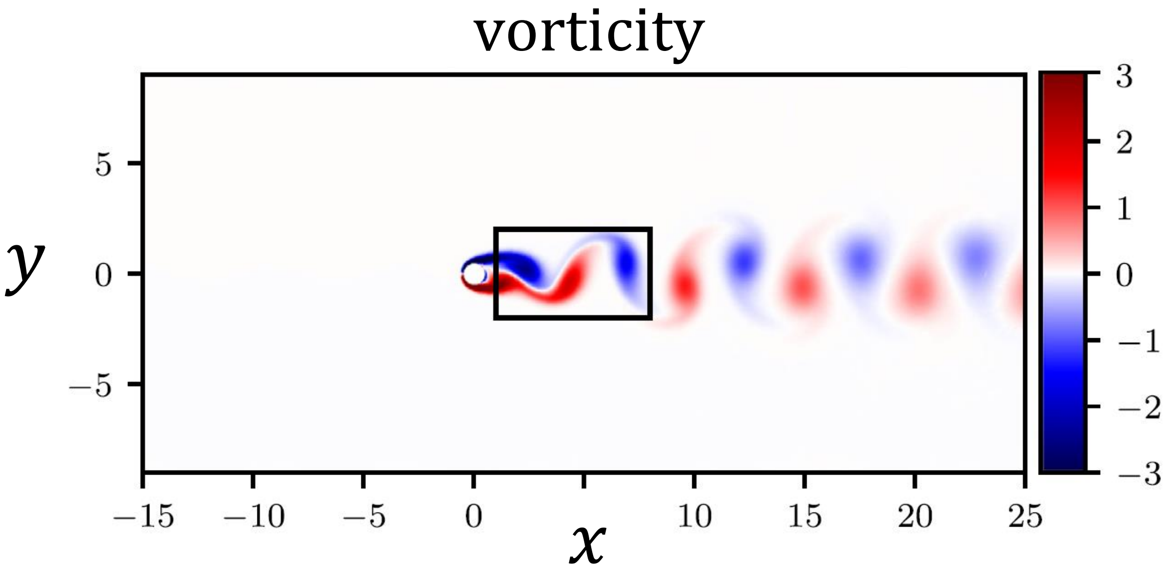
Collocation points

E.g., Navier-Stokes equation





Ground truth from numerical simulation:



Given training data of u, v, find $\boldsymbol{\lambda_1, \lambda_2}$

Training data (from ground truth):

$\{t^i, x^i, y^i, u^i, v^i\}_{i=1}^N$ $\quad N = 5{,}000$

Collocation points:

$\{t^i, x^i, y^i\}_{i=1}^N$

Physics equations:

$$f := u_t + \lambda_1(uu_x + vu_y) + p_x - \lambda_2(u_{xx} + u_{yy})$$
$$g := v_t + \lambda_1(uv_x + vv_y) + p_y - \lambda_2(v_{xx} + v_{yy})$$

Loss function:

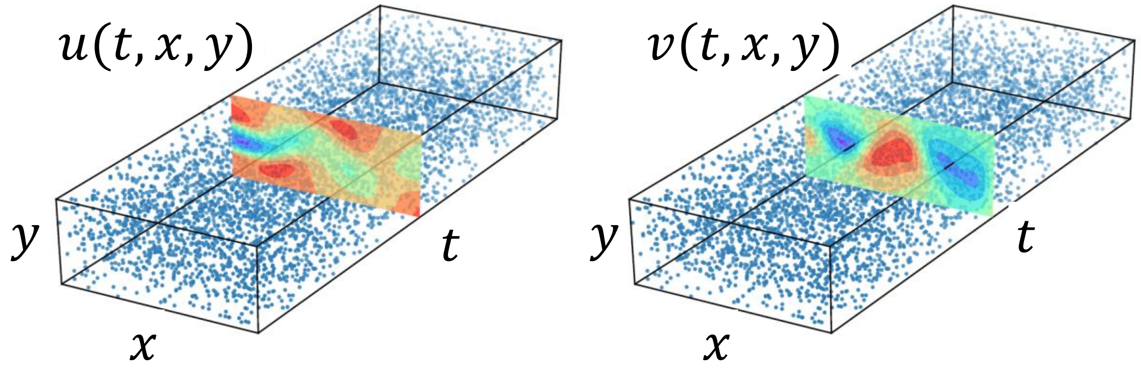
$$MSE := \frac{1}{N}\sum_{i=1}^{N}\left(|u(t^i, x^i, y^i) - u^i|^2 + |v(t^i, x^i, y^i) - v^i|^2\right)$$

Data loss

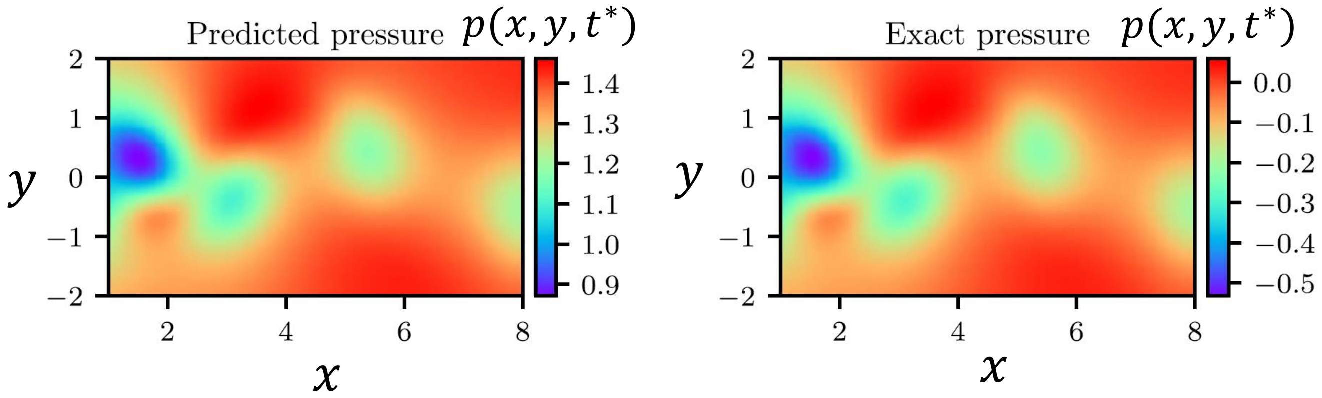
$$+ \frac{1}{N}\sum_{i=1}^{N}\left(|f(t^i, x^i, y^i)|^2 + |g(t^i, x^i, y^i)|^2\right)$$

Equation loss

E.g., Navier-Stokes equation



Correct PDE	$u_t + (uu_x + vu_y) = -p_x + 0.01(u_{xx} + u_{yy})$ $v_t + (uv_x + vv_y) = -p_y + 0.01(v_{xx} + v_{yy})$
Identified PDE (clean data)	$u_t + 0.999(uu_x + vu_y) = -p_x + 0.01047(u_{xx} + u_{yy})$ $v_t + 0.999(uv_x + vv_y) = -p_y + 0.01047(v_{xx} + v_{yy})$
Identified PDE (1% noise)	$u_t + 0.998(uu_x + vu_y) = -p_x + 0.01057(u_{xx} + u_{yy})$ $v_t + 0.998(uv_x + vv_y) = -p_y + 0.01057(v_{xx} + v_{yy})$



Given training data of u, v, find $\boldsymbol{\lambda_1, \lambda_2}$

Training data (from ground truth):

$\{t^i, x^i, y^i, u^i, v^i\}_{i=1}^N \qquad N = 5{,}000$

Collocation points:

$\{t^i, x^i, y^i\}_{i=1}^N$

Physics equations:

$$f := u_t + \lambda_1(uu_x + vu_y) + p_x - \lambda_2(u_{xx} + u_{yy})$$
$$g := v_t + \lambda_1(uv_x + vv_y) + p_y - \lambda_2(v_{xx} + v_{yy})$$

Loss function:

$$MSE := \frac{1}{N}\sum_{i=1}^{N}\left(|u(t^i, x^i, y^i) - u^i|^2 + |v(t^i, x^i, y^i) - v^i|^2\right)$$

Data loss

$$+ \frac{1}{N}\sum_{i=1}^{N}\left(|f(t^i, x^i, y^i)|^2 + |g(t^i, x^i, y^i)|^2\right)$$

Equation loss

PINN

$$\frac{\partial u}{\partial t} + N(u, \lambda) = 0, \qquad x \in [-L, L], \qquad t \in [0\ T]$$

- Application 1: Prediction of solution for a **well-posed problem**

  Given an eqn + BC + IC and parameters λ, what's the model prediction?

- Application 2: Prediction of solution when data is available within the domain but not at the IC, BC

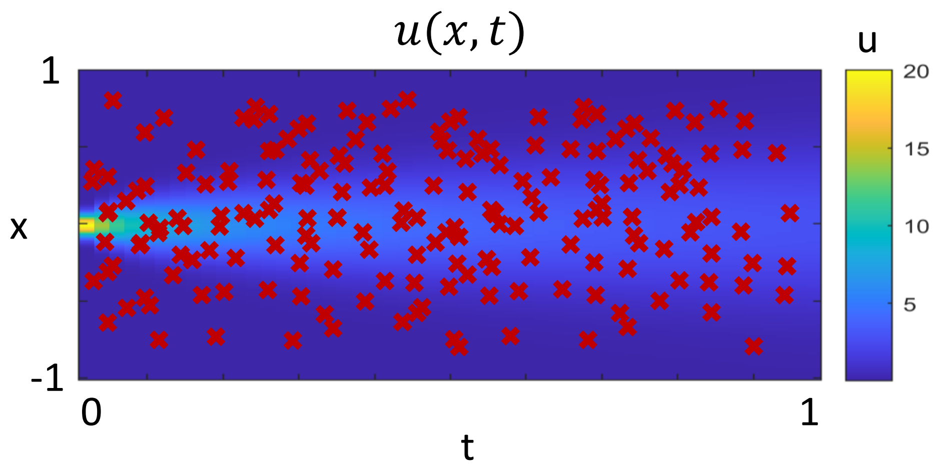
  Given an eqn and parameters λ, what's the model prediction best describes the data?

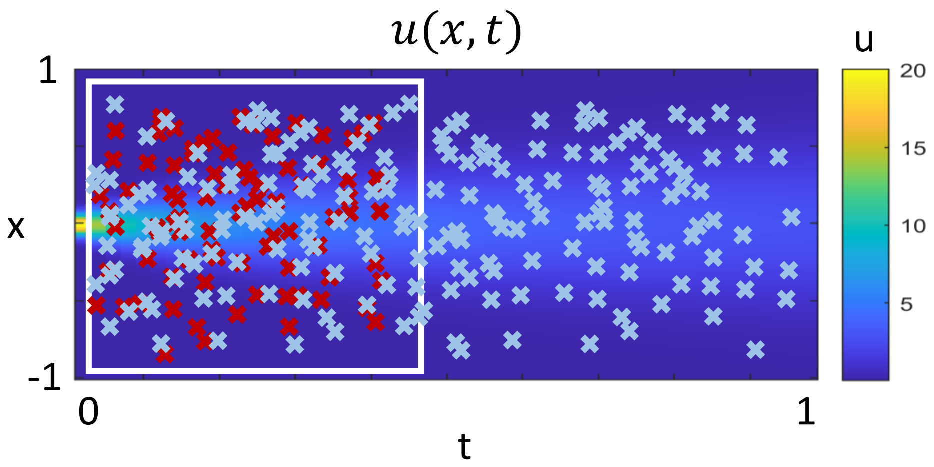
- Application 3: Data-driven discovery of **unknown parameters**

  What are the parameters λ that best describe the data and the eqn?

When can you use only data to train NN without physics constants, and still get a good NN prediction (i.e. consistent with the physics)?

- When the data is perfect without noise and available everywhere
- When the prediction is within the same {t,x} domain as the training data (no need to generalize the prediction to an unseen domain)





When can you use only data to train NN without physics constants, and still get a good NN prediction (i.e. consistent with the physics)?

- When the data is perfect without noise and available everywhere
- When the prediction is within the same {t,x} domain as the training data (no need to generalize the prediction to an unseen domain)

The point of PINN is that its prediction can be generalized to a domain without observations!