AOS 551
Deep Learning in Geophysical Fluid Dynamics

Instructor: Ching-Yao Lai
Office location: 418A Guyot Hall
Email: cylai@princeton.edu
Office hours: Thursdays 1-2pm

Today

- What will be covered in this course?
- Examples of topics that will be covered
- Grading, presentations, course project, final paper
- Tools: Anaconda python, Jupyter notebook, Github

Who is the class for?

- Knowledge in differential equations (undergrad level).
- Fluid dynamicists, climate scientists, earth scientists.
- Have domain knowledge in one of the above fields, interested in learning about neural networks and its application to your research field.
- Don’t need to have experience in machine learning.

Why take this class?

- Learn useful concepts about neural networks
- Learn how to incorporate our physics knowledge into a neural network training
- See examples of how these can be applied to GFD
- Learn how to use/implement the methods covered in class
- Develop a course project that applies these novel and exciting tools to your research!

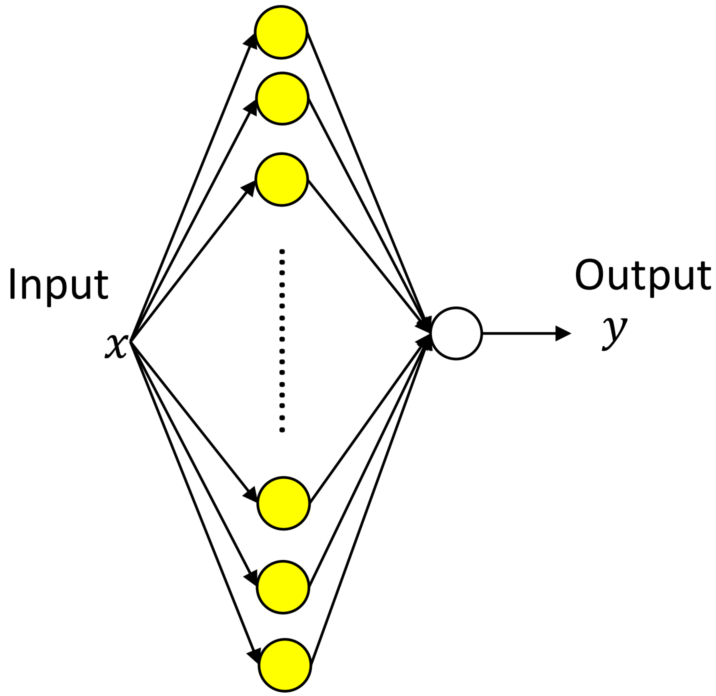
What will be covered in this course?

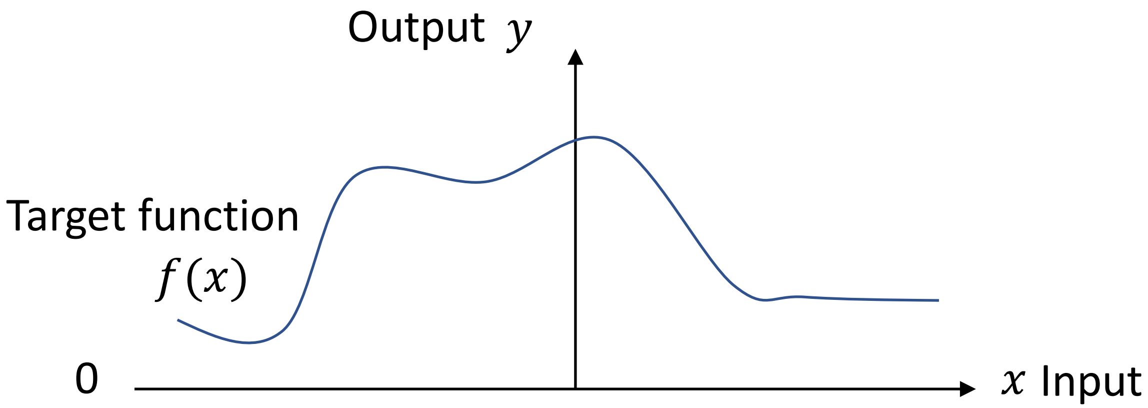
- Basics of neural networks
- How to constrain a neural network with physical principles?
- How to use a neural network to solve differential equations?
- How can neural networks informed by physics and data solve inverse problems?
- Discovering governing equations from data
- Discovering dynamical regimes

Examples of topics you will see in this course

What is a neural network?

- Universal function approximator: NN(x) = y(x)
  e.g. curve fitting



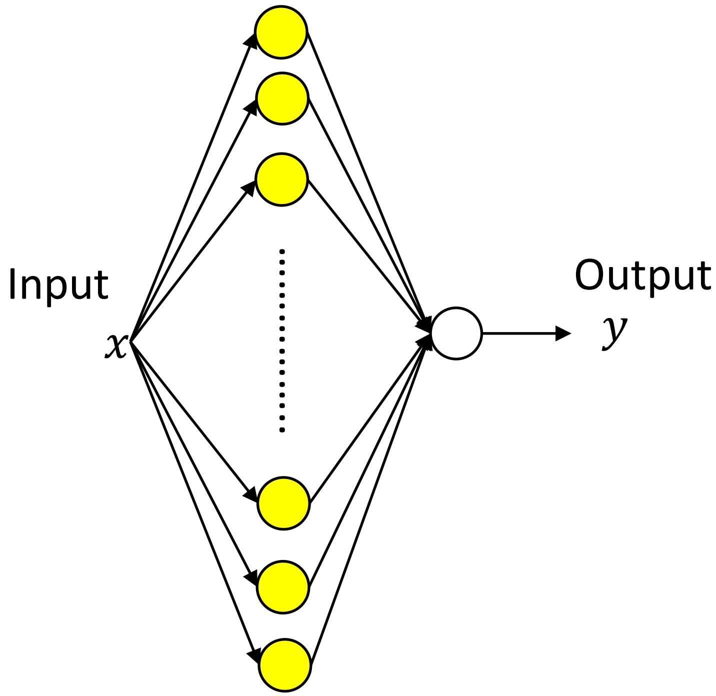


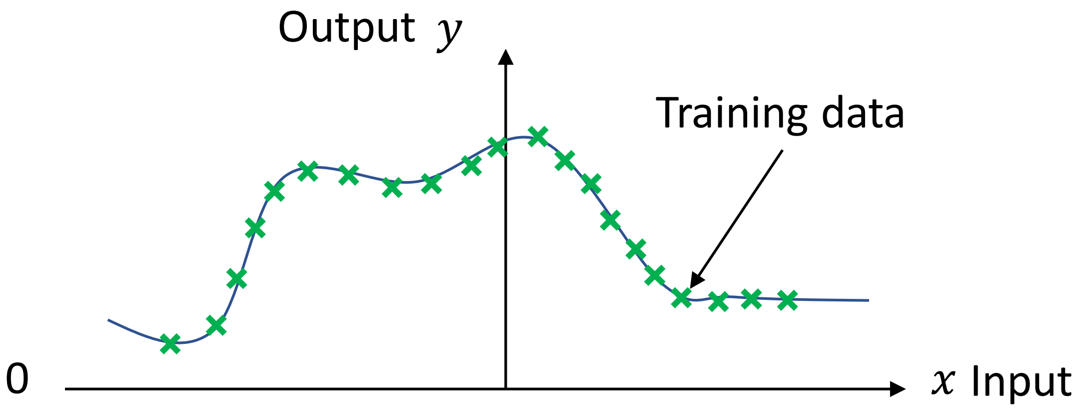
For a given continuous smooth $f(x)$ and an arbitrarily small $\epsilon > 0$
There exist NN prediction $\mathrm{y}(x)$ so that

$$\int |y(x) - f(x)|\, dx < \epsilon$$

What is a neural network?

- Universal function approximator: NN(x) = y(x)
  e.g. curve fitting (discrete data)



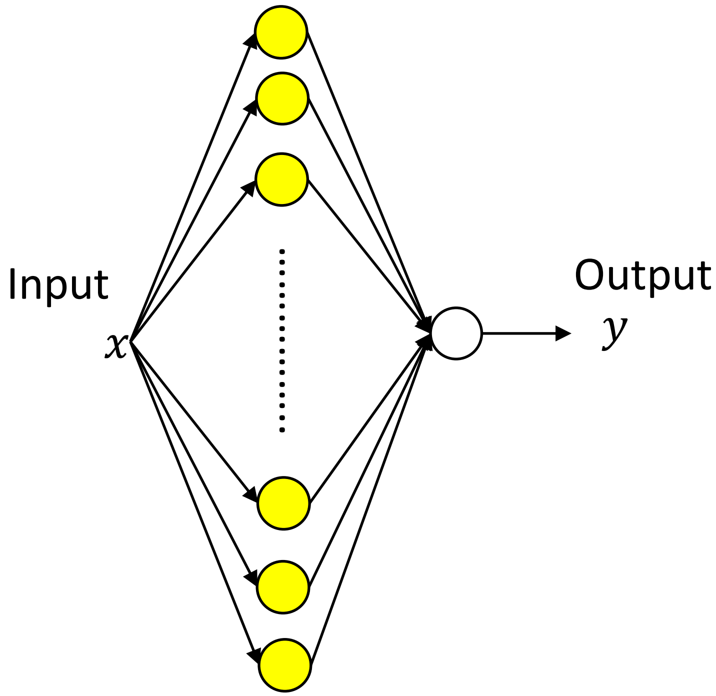


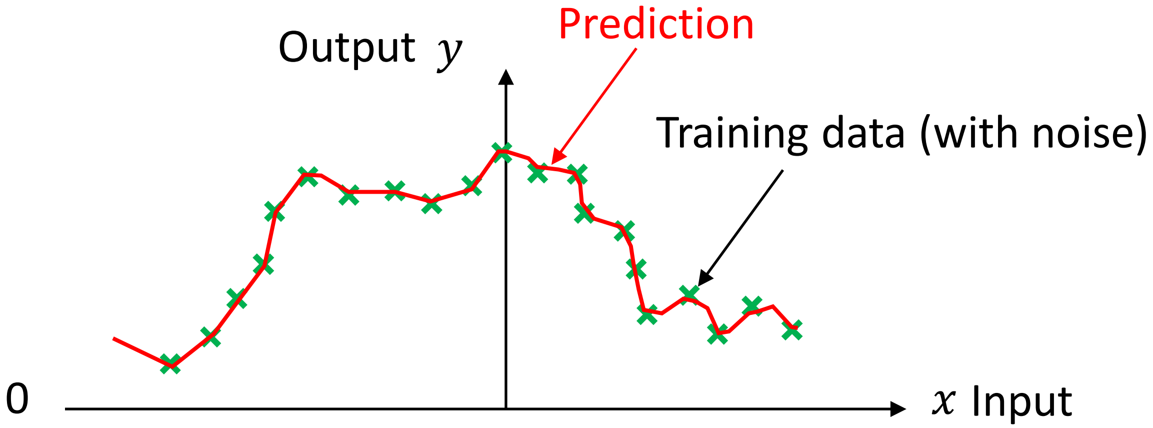
For a given data set $\{x_d^i\ ,\ y_d^i\}$ and an arbitrarily small $\epsilon > 0$
There exist NN prediction $y\left(x_d^i\right)$ so that

$$\sum_{i=1}^{n}\left(y\left(x_d^i\right) - y_d^i\right)^2 < \epsilon$$

What is a neural network?

- NN can easily overfit noisy data!



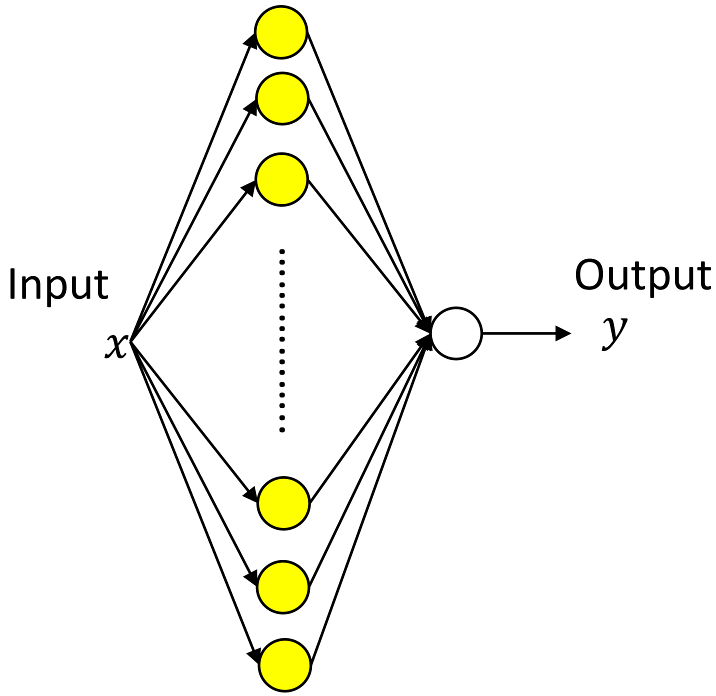


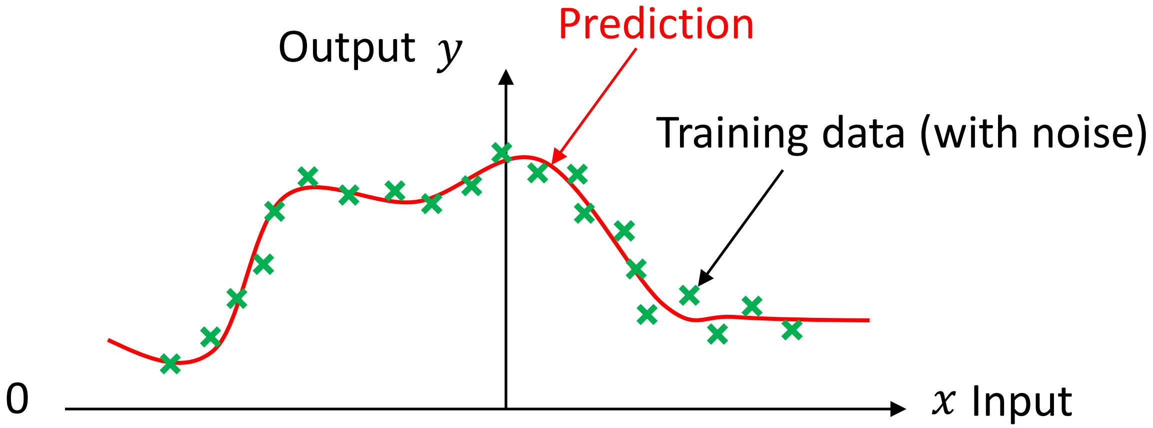
For a given data set $\{x_d^i\ ,\ y_d^i\}$ and an arbitrarily small $\epsilon > 0$
There exist NN prediction $y(x_d^i)$ so that

$$\sum_{i=1}^{n} \left(y(x_d^i) - y_d^i\right)^2 < \epsilon$$

What is a neural network?

- NN can easily overfit noisy data!
- In this class we will introduce the use of **physical principles** to regularize the NN prediction



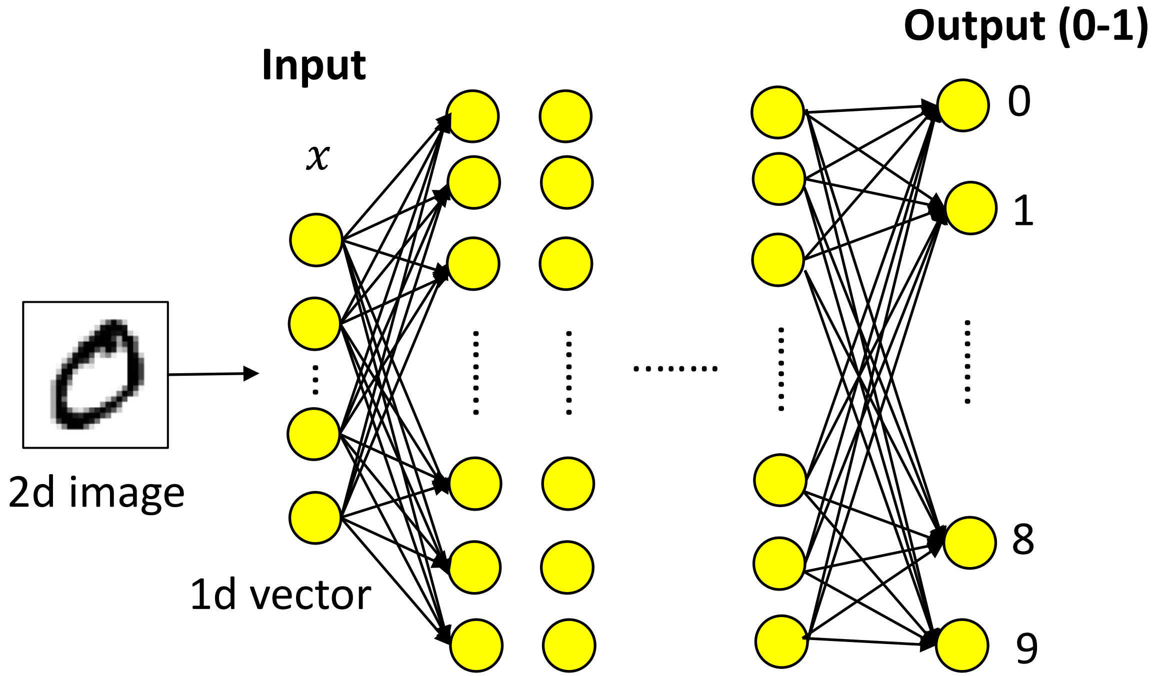


For a given data set $\{x_d^i\ ,\ y_d^i\}$ and an arbitrarily small $\epsilon > 0$
There exist NN prediction $y\left(x_d^i\right)$ so that

$$\sum_{i=1}^{n}\left(y\left(x_d^i\right) - y_d^i\right)^2 < \epsilon$$

Who cares about curve fitting?

- Curve fitting is a simple example. More generally NN is good at finding the relationship between output and input.

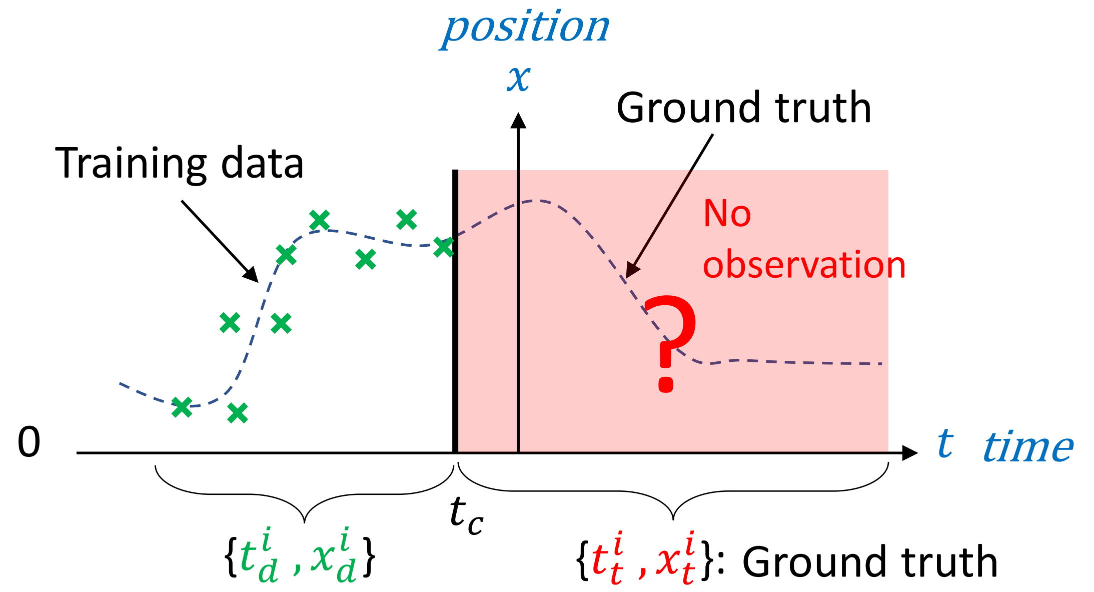


Training data

Can NN be used to predict dynamics?

Dynamics $\equiv x(t)$

Can a NN trained with data within a specific domain ($t < t_c$) be **extrapolated** to make prediction in an unseen domain ($t > t_c$)?



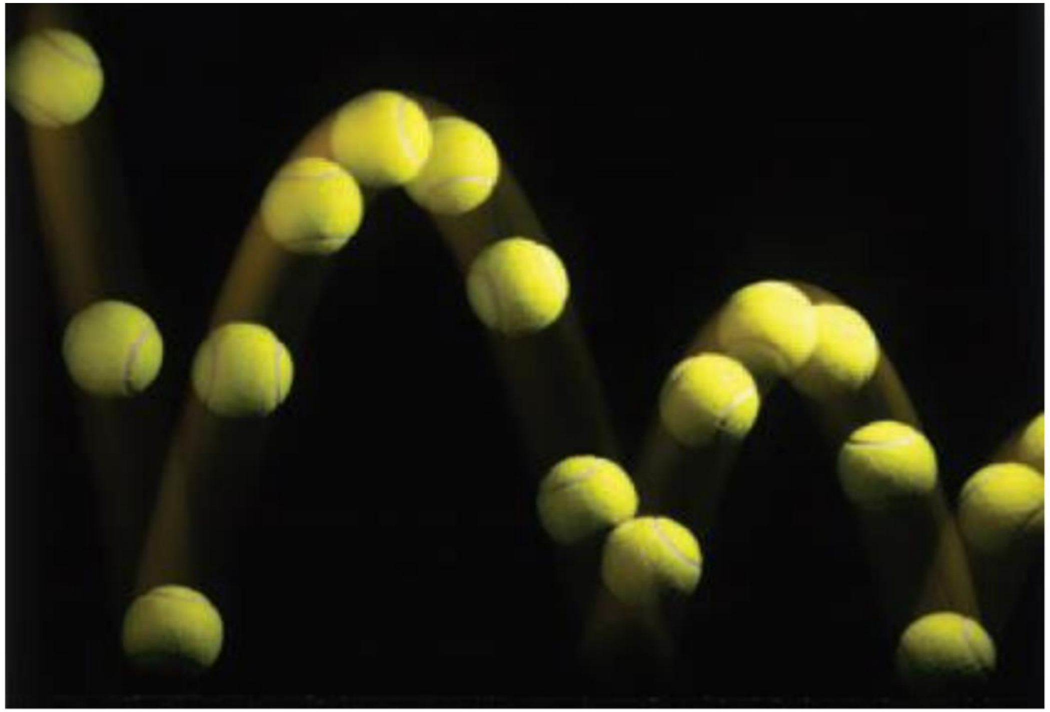
Will NN prediction NN(t) = $x(t_t^i)$ in the domain in which training data doesn't exit obey

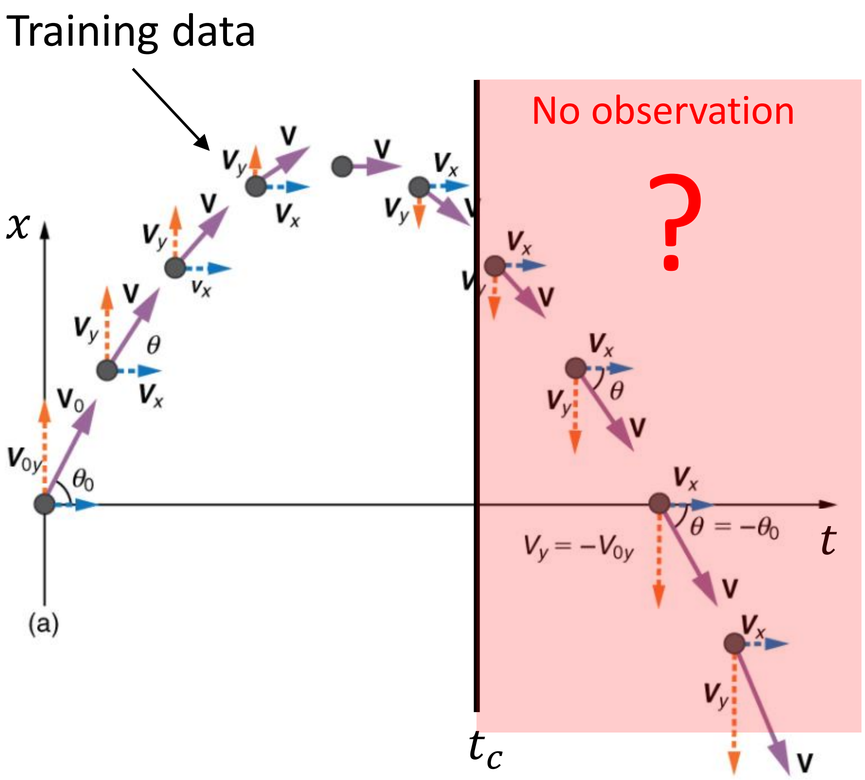
$$\sum_{i=1}^{n}\left(x(t_t^i) - x_t^i\right)^2 < \epsilon \quad ?$$

Can NN be used to predict dynamics?

Learn from data that the second derivative of position w.r.t. time is a constant → can be **extrapolated** to make prediction for future trajectories

$$\boxed{F = m\frac{d^2x}{dt^2}}$$





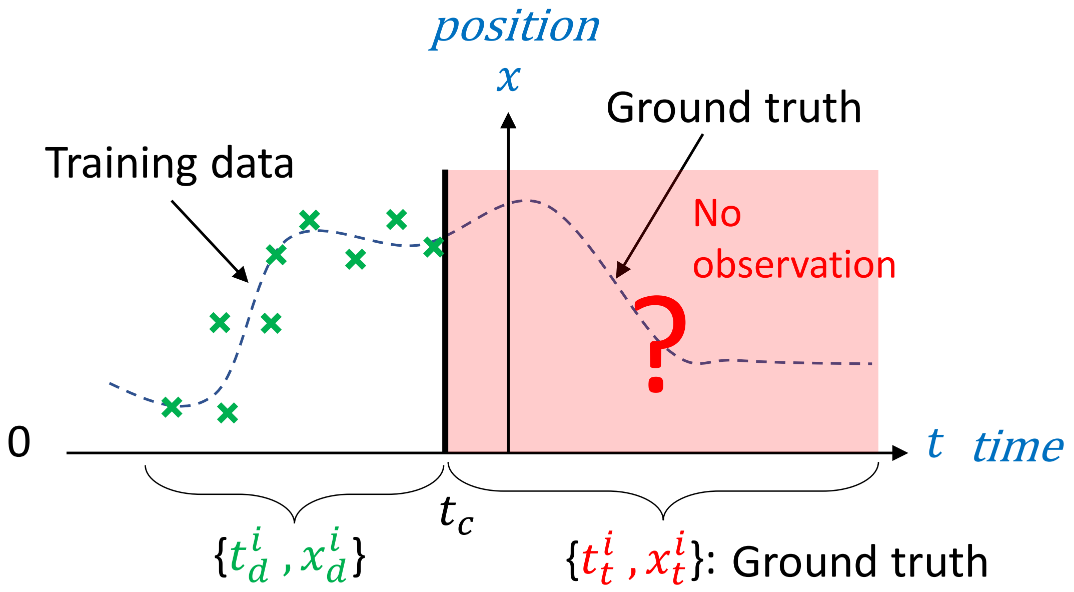
Can NN be used to predict dynamics?

However, if data is **noisy**.....

The NN will very likely overfit the data instead of learning the actual operators in the governing laws (e.g., $\frac{d^2x}{dt^2} = ?, \frac{dx}{dt} = ?$)

Can't be extrapolated to $t > t_c$.

In this class we will teach you how to address this issue!



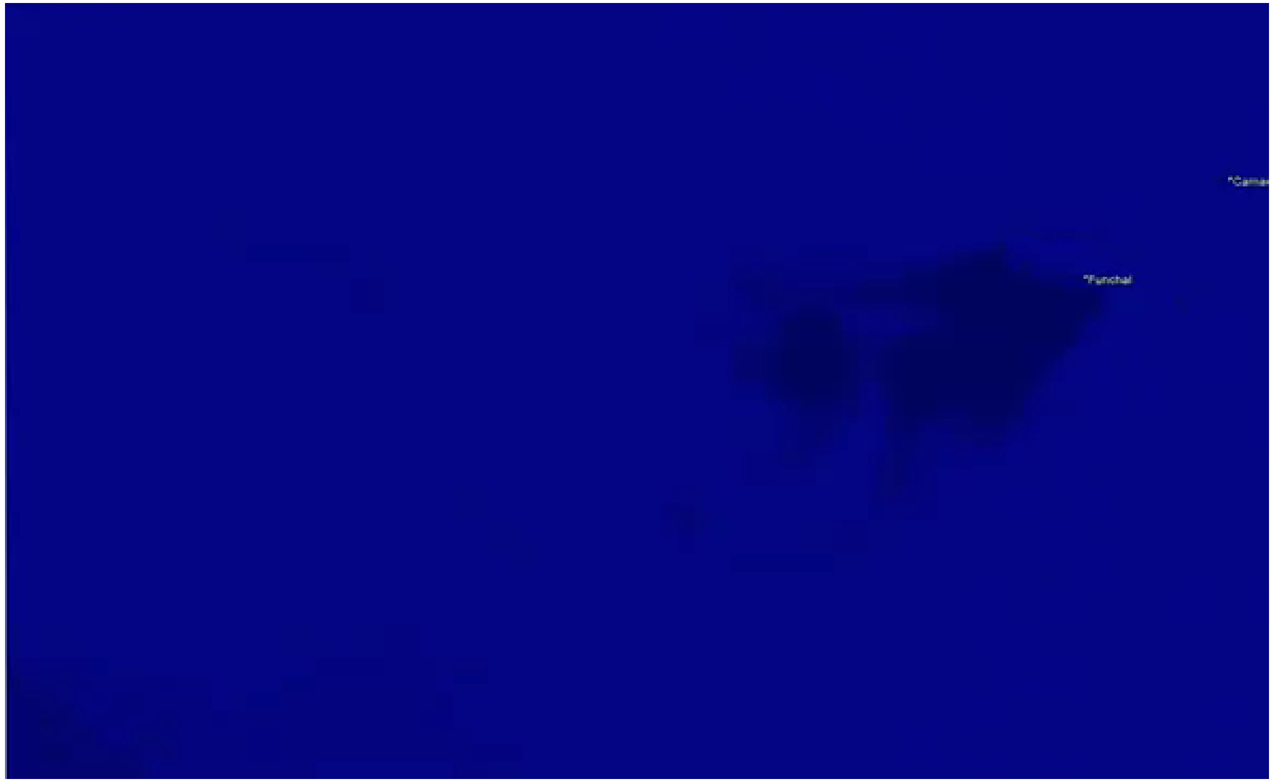
Will NN prediction $x(t_t^i)$ in the domain in which training data doesn't exit obey

$$\sum_{i=1}^{n} \left(x(t_t^i) - x_t^i\right)^2 < \epsilon \quad ?$$

$*$Reynolds number = inertia/viscous forces, $\mathrm{Re} = \frac{\rho U L}{\mu}$

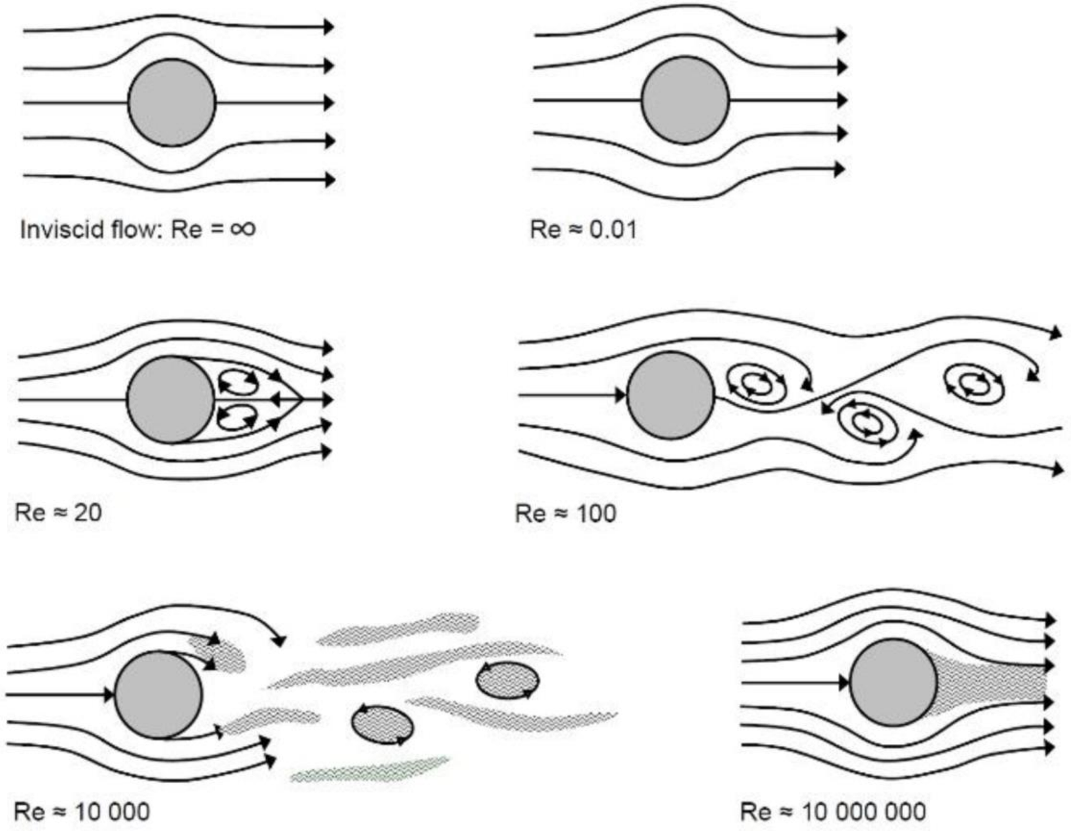
Inverse problems

E.g. what’s the Reynolds number (Re)*?



Von-Kármán vortex street downwind of the island of Madeira (northeastern Atlantic Ocean).

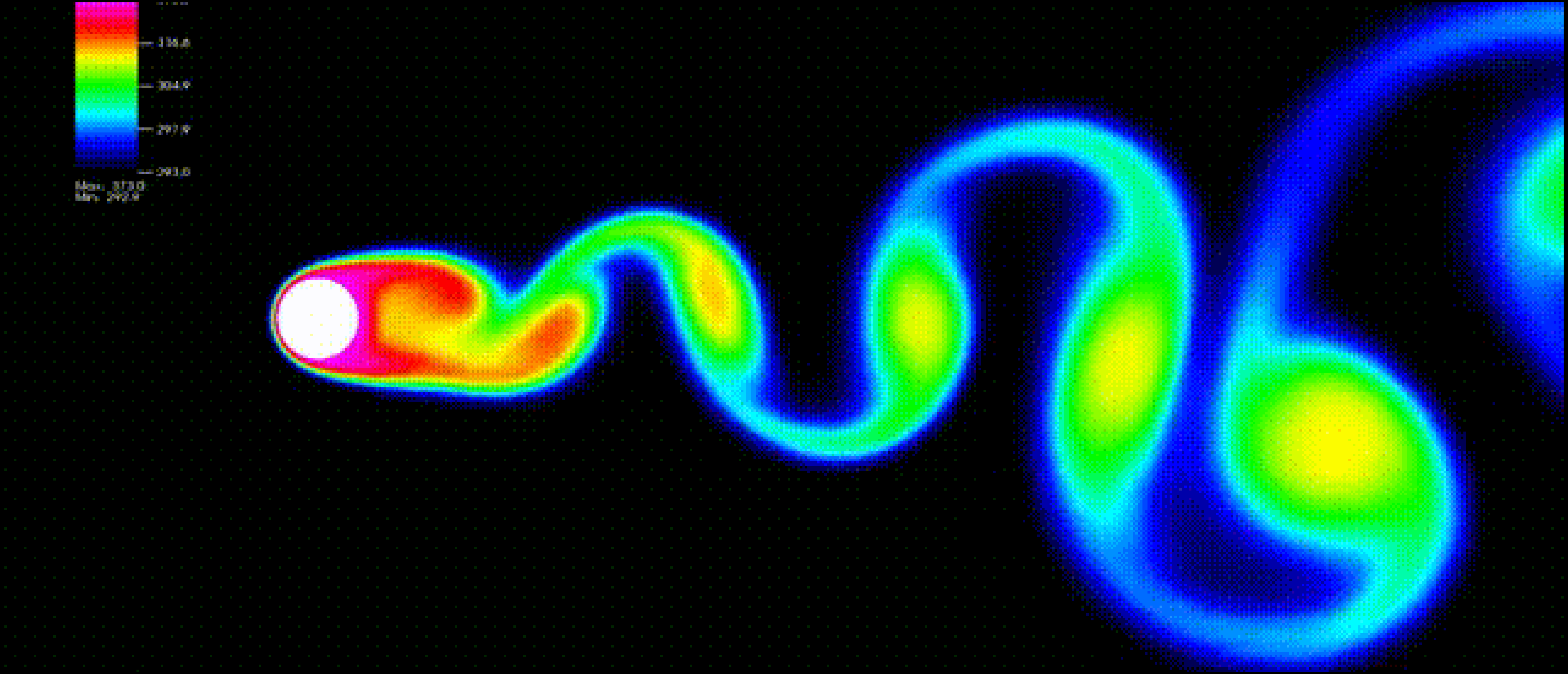
The length of this mountainous island is 57 kilometers, reaching at its highest point 1862 meters (February 2015).



*Reynolds number = inertia/viscous forces, $\mathrm{Re} = \frac{\rho U L}{\mu}$

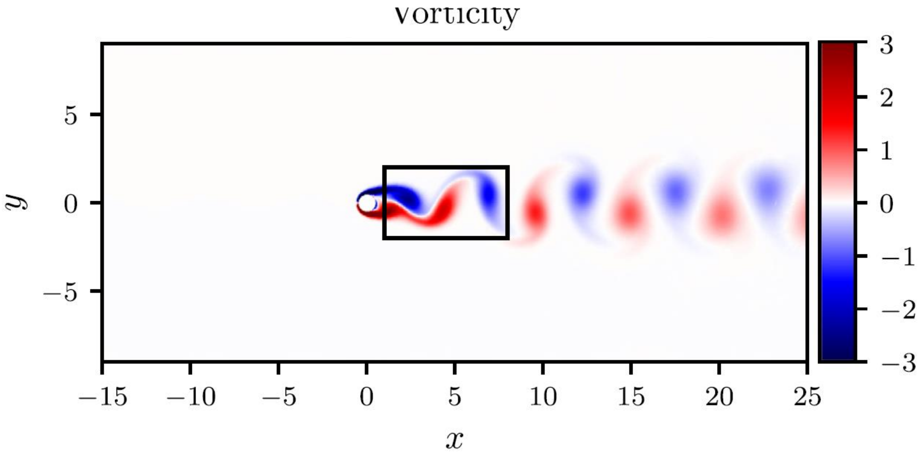
Inverse problems

E.g. what's the Reynolds number (Re)*?



Inverse problems

E.g. what’s the Reynolds number (Re)*?

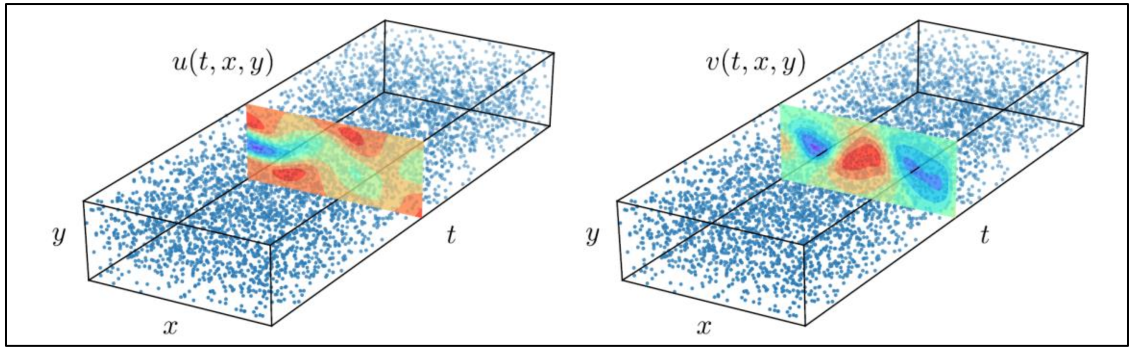


Given training data of velocities, find λ_1, λ_2 (unknown parameters)

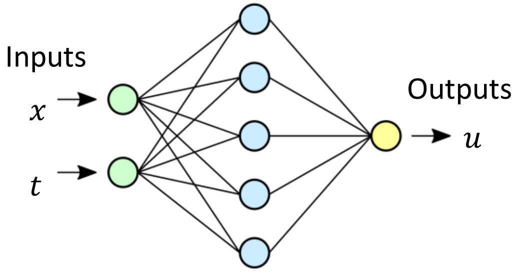
Navier-Stokes equation:

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

1% of the total available data



Model in



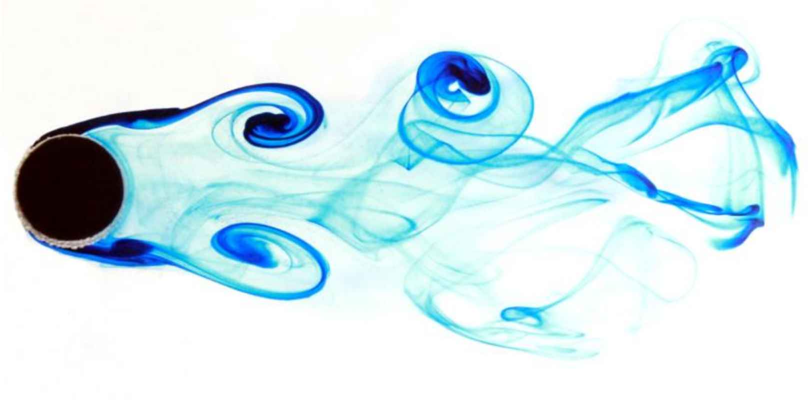
Model out

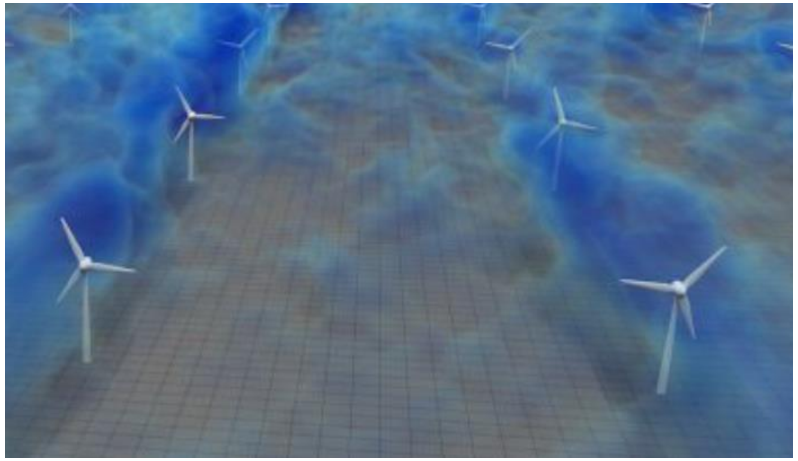
Correct PDE	$u_t + (uu_x + vu_y) = -p_x + 0.01(u_{xx} + u_{yy})$ $v_t + (uv_x + vv_y) = -p_y + 0.01(v_{xx} + v_{yy})$
Identified PDE (clean data)	$u_t + 0.999(uu_x + vu_y) = -p_x + 0.01047(u_{xx} + u_{yy})$ $v_t + 0.999(uv_x + vv_y) = -p_y + 0.01047(v_{xx} + v_{yy})$
Identified PDE (1% noise)	$u_t +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

Raissi et al (2020)

Discover parameters from data + physics

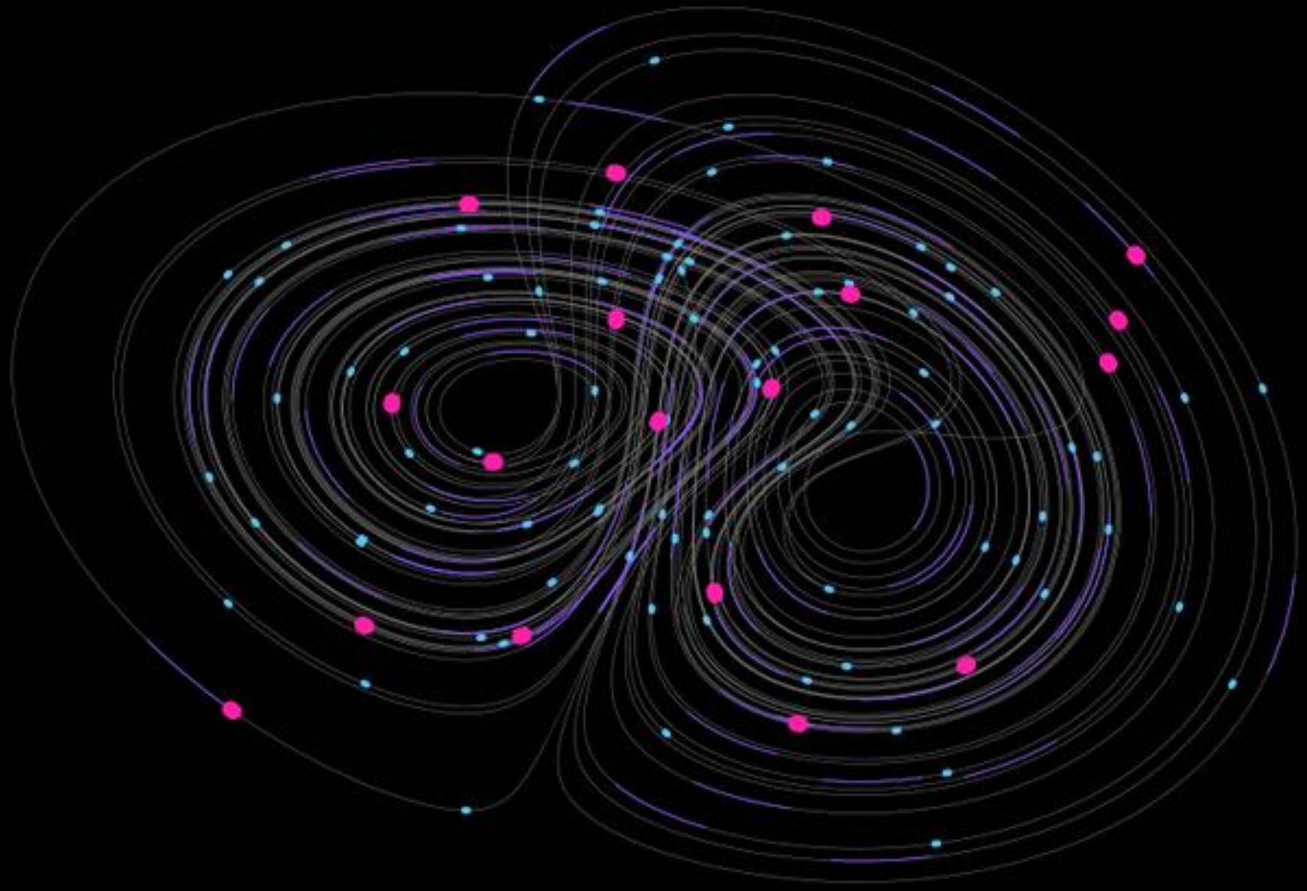
Infer parameters (e.g., velocity fields, pressure fields) from tracer dye?





Discover governing equations from data

Lorenz system (Nonlinear ODE)



$$\frac{dx}{dt} = \sigma(y - x)$$

$$\frac{dy}{dt} = x(\rho - z) - y$$

$$\frac{dz}{dt} = xy - \beta z$$

$x \sim$ the rate of convection
$y \sim$ the horizontal temperature variation
$z \sim$ the vertical temperature variation

Brunton et al (2016)

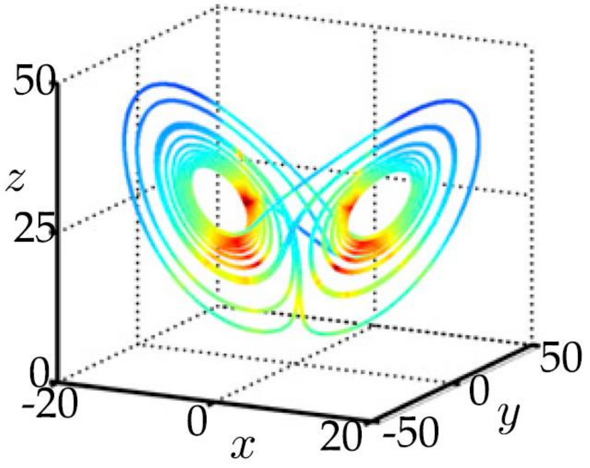
Discover governing equations from data

Lorenz system (Nonlinear ODE)

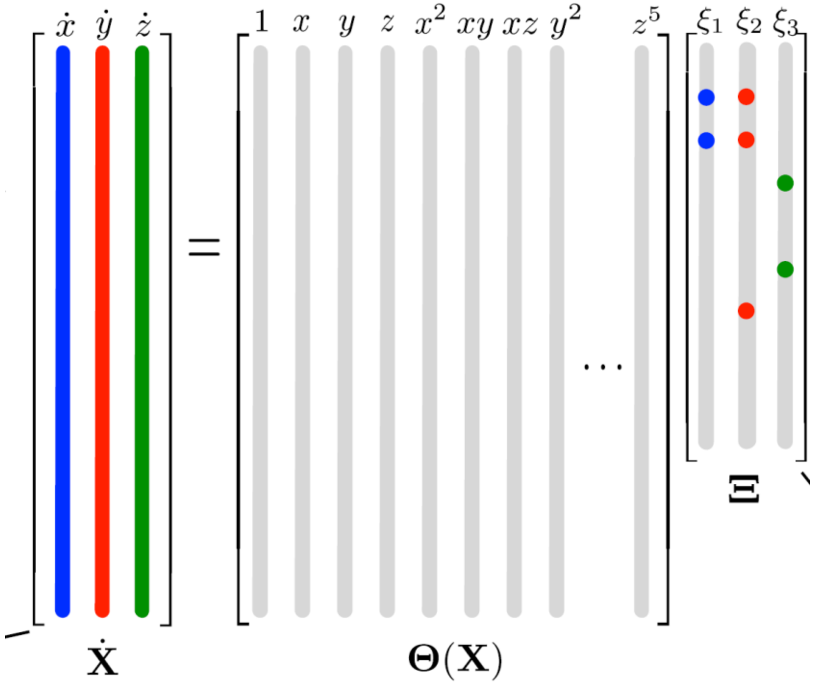
Algorithm can identify the correct mathematical forms and the coefficients!

I. True Lorenz System

$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = x(\rho - z) - y$$
$$\dot{z} = xy - \beta z.$$



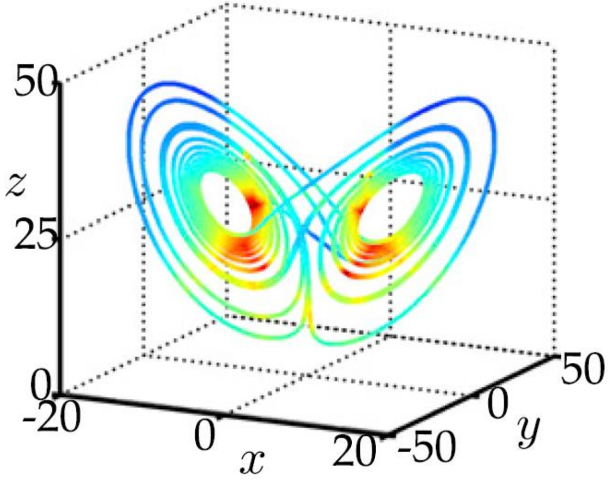
Model in



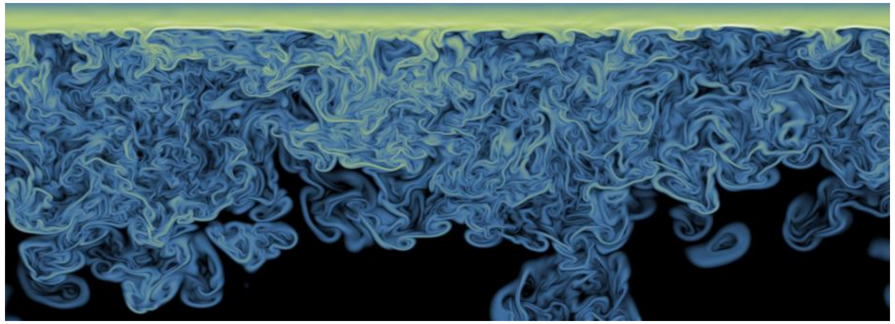
Model out

III. Identified System

$$\dot{x} = \Theta(\mathbf{x}^T)\xi_1$$
$$\dot{y} = \Theta(\mathbf{x}^T)\xi_2$$
$$\dot{z} = \Theta(\mathbf{x}^T)\xi_3$$



Discover dynamical regimes



Navier-Stokes eqn

$$\boxed{\rho \frac{\partial \boldsymbol{u}}{\partial t}} = -\boldsymbol{\nabla} p + \rho \boldsymbol{g} + \boxed{\mu \nabla^2 \boldsymbol{u}}$$

Inertia term

Pressure gradient

Gravitational force

Viscous term

$$Re \equiv \frac{\text{Inertia}}{\text{Viscous}}$$

$Re \gg 1 \rightarrow$ turbulent

$Re \ll 1 \rightarrow$ laminar

Advection-diffusion eqn

$$\frac{\partial c}{\partial t} = \boxed{\alpha \nabla^2 c} - \boxed{u \cdot \nabla c}$$

Rate of change of concentration

Diffusion term

Advection term

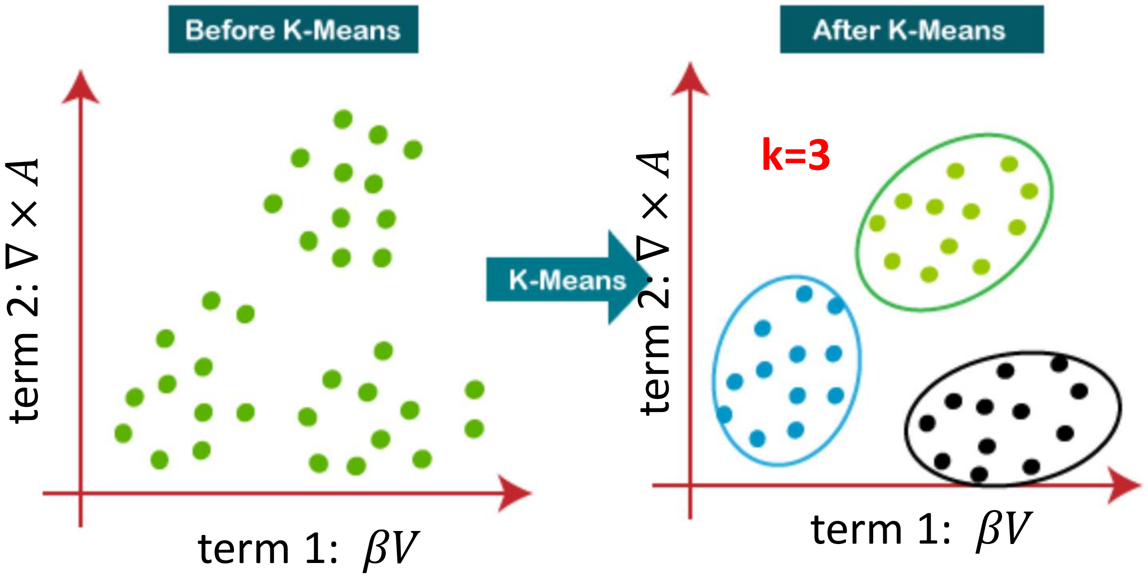
$$Pe \equiv \frac{\text{Advection}}{\text{Diffusion}}$$

Comparing the magnitudes of terms to know the regimes

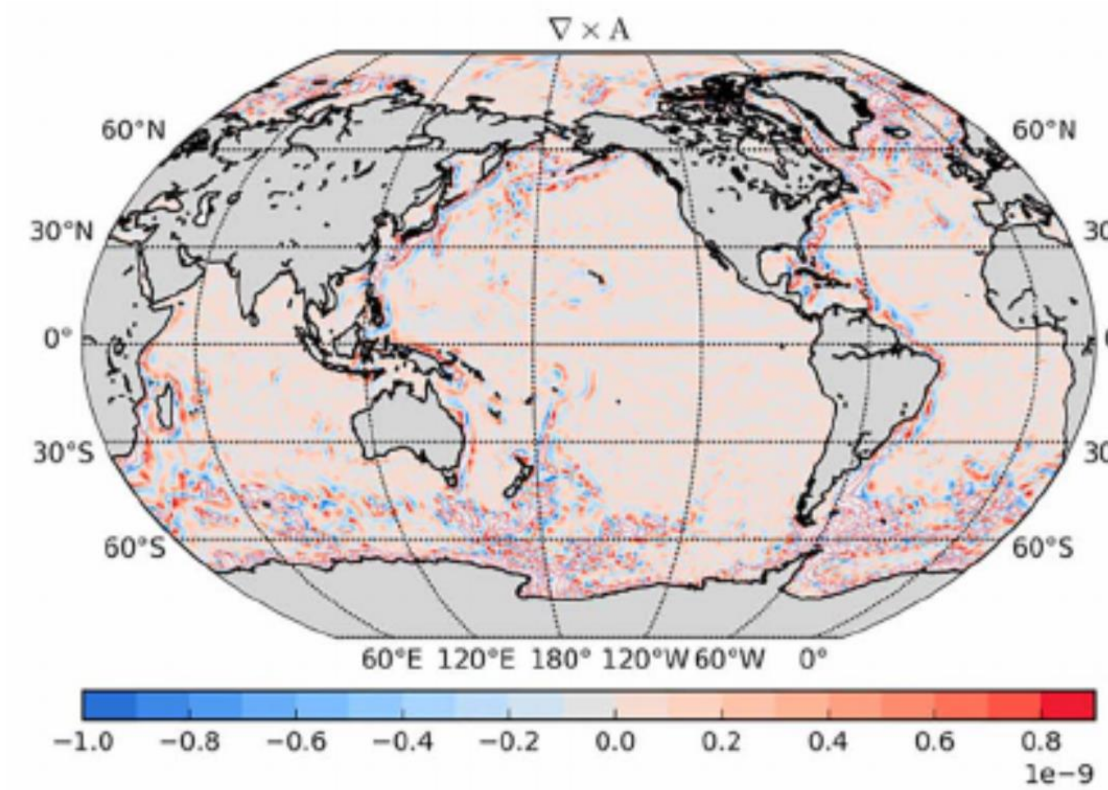
Discover dynamical regimes

Momentum equation for ocean on a rotating sphere

$$\beta V = \frac{1}{\rho_0}\nabla p_b \times \nabla H + \frac{1}{\rho_0}\nabla \times \tau + \nabla \times \mathbf{A} + \nabla \times \mathbf{B}$$



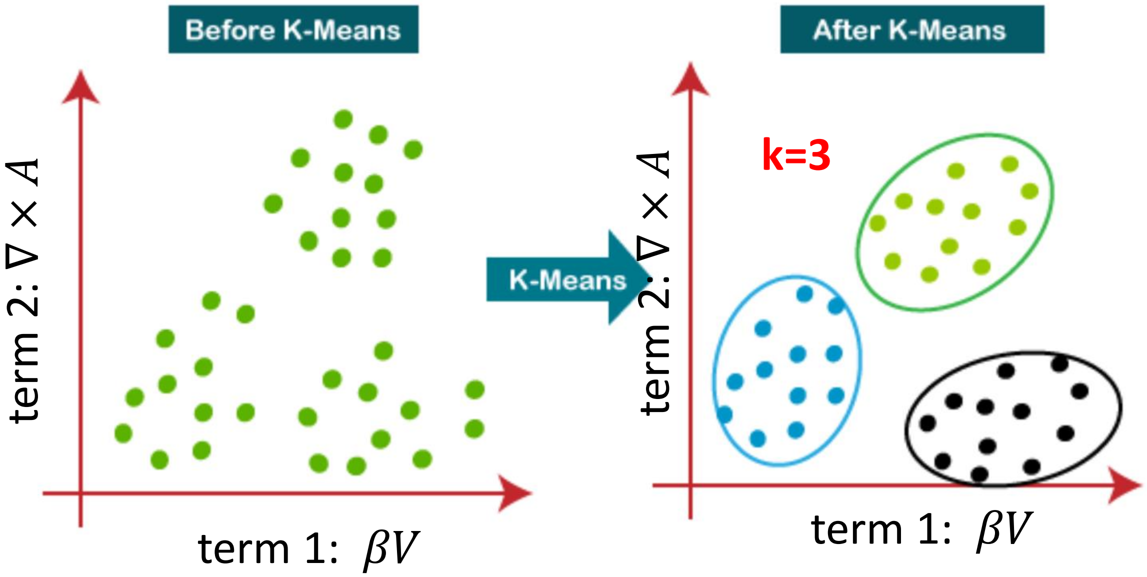
E. g., Discover global Ocean Dynamical Regions with unsupervised learning



Discover dynamical regimes

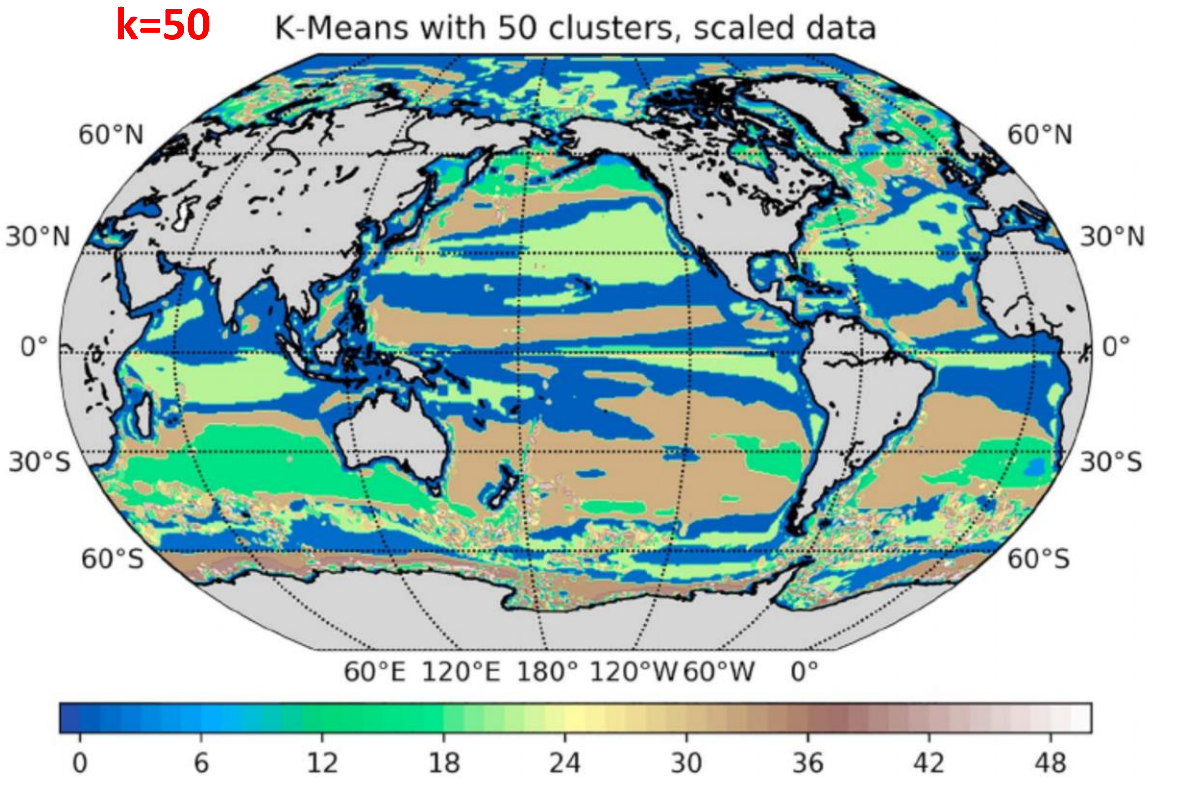
Momentum equation for ocean on a rotating sphere

$$\beta V = -\frac{1}{\rho_0}\nabla p_b \times \nabla H + \frac{1}{\rho_0}\nabla \times \tau + \nabla \times \mathbf{A} + \nabla \times \mathbf{B}$$



E. g., Discover global Ocean Dynamical Regions with unsupervised learning

k=50



(a) Area selected by the clusters.

Schedule

Lecture 1: Introduction
Lectures 2-4: Basics of neural networks (back propagation, universal function approximation)
Lectures 5-7: Physics-informed neural networks (PINN)
Lecture 8: Inferring hidden parameters in fluid dynamics
Lecture 9: Example of PINN applying to ice dynamics
Lecture 10: Collocation points, meaning of equation weights, and optimal weights
Lecture 11-13: Student presentations of selected papers (list of suggested papers will be available)
Fall break (10/16-24)
Lecture 14: High-frequency function approximation
Lecture 15-17: Discovering governing equations from data
Lecture 18-19: Basics of convolutional neural networks (CNN) and application to fluid modeling
Lecture 20: No Free Lunch: How ML can be used (or mis-used) to uncover dynamical regimes in the ocean and beyond. Guest lecturer: Dr. Maike Sonnewald
Thanksgiving break (11/24-28)
Lecture 21- 24: Student presentations on course projects.
Reading period (12/7-14)
Final exam period (12/15-21)
Final paper due on 12/21 5pm

Schedule

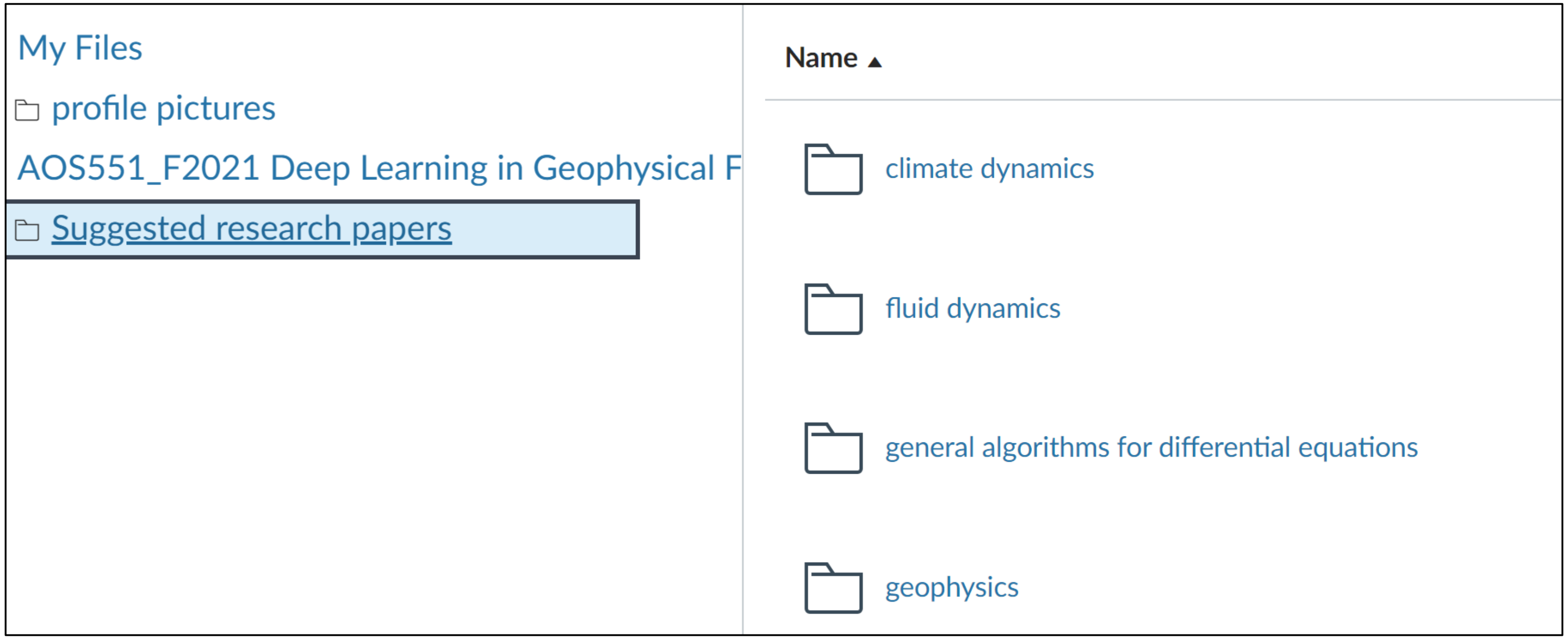
- 24 Lectures: Tue, Thur 11-12:20pm; Office hours: Thus 1-2pm
- 6 of the lectures will involve in-class coding tutorial (dates TBD)
- I’m out of town on 11/16, 11/23
  - We will find time to do make-up lectures.
- Paper presentations (10/7, 12, 14)
- Course project presentations (11/30, 12/2, TBD)
- No exams

Grading

- Coding exercises- won’t be graded.
- 15% Presentation of 1 paper- select a paper relevant to the intersection of ML and your research filed.
- 60% Final paper that summarizes the course project (due on 12/21 5pm)
- 15% Presentation of course project

Paper presentation

Select a paper relevant to the intersection of ML and your research filed.
Suggested papers are available on Canvas. You can select a paper from the suggested list or find a paper and discuss with me if it would fit.



Course project

- 1 independent project/per person.
- Schedule a meeting with me near the end of September to discuss the project idea.
- Have most of the analysis done before the presentations on 11/30, 12/2.
- Suggestion: Start with an idealized and well-defined problem. (there are some examples in class)
- Training data can be **computational, experimental, or field data**

Final paper (<10 pages*) due on 12/21 5pm

- Introduction and background knowledge

  What specific problem in this project was difficult with existing method?

  Define all terminology that is specific in your field. Avoid jargons if you can.

- Problem statement

  What is the task that a ML model need to do?

  What are the physics constraints that will be used?

- Result and Discussion
- Open questions for future work

*The final paper will not be judged by its length

Evaluation of final paper

- Basic criteria for the evaluation written paper are:
  1. Sophistication of material presented
  2. Quality of exposition
  3. Amount of effort

Tools:

- Python, Tensorflow package.
  If you are unfamiliar with python, don’t worry. We have many examples to help you be familiar with it
- Jupyter notebook. (in-class tutorial)

Before the next lecture:

- Install Anaconda: https://www.anaconda.com/products/individual
- Create a Github account. Class Github: https://github.com/AOS551

What if I have seen some of these topics before?

- You may still benefit from this class
- Explore the intersection of traditional physics-based models and deep learning
- Develop a course project that applies these novel and exciting tools to your research!

Schedule

Lecture 1: Introduction
Lectures 2-4: Basics of neural networks (back propagation, universal function approximation)
Lectures 5-7: Physics-informed neural networks (PINN)
Lecture 8: Inferring hidden parameters in fluid dynamics
Lecture 9: Example of PINN applying to ice dynamics
Lecture 10: Collocation points, meaning of equation weights, and optimal weights
Lecture 11-13: Student presentations of selected papers (list of suggested papers will be available)
Fall break (10/16-24)
Lecture 14: High-frequency function approximation
Lecture 15-17: Discovering governing equations from data
Lecture 18-19: Basics of convolutional neural networks (CNN) and application to fluid modeling
Lecture 20: No Free Lunch: How ML can be used (or mis-used) to uncover dynamical regimes in the ocean and beyond. Guest lecturer: Dr. Maike Sonnewald
Thanksgiving break (11/24-28)
Lecture 21- 24: Student presentations on course projects.
Reading period (12/7-14)
Final exam period (12/15-21)
Final paper due on 12/21 5pm