

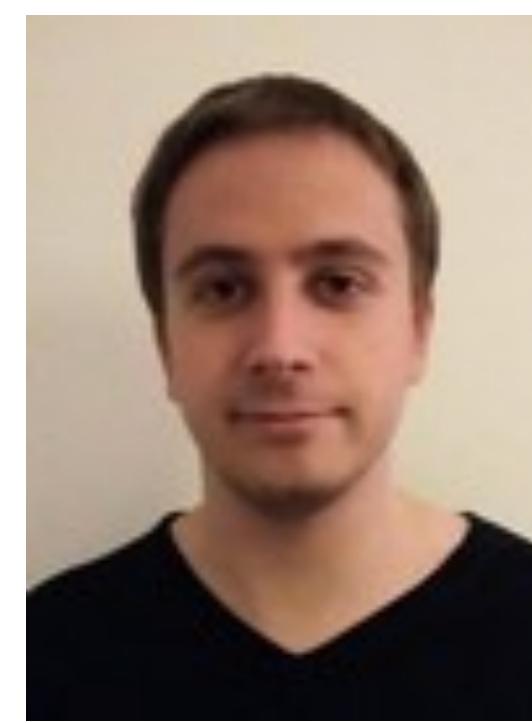
# Deep Learning for Ocean and Atmosphere (OA) Dynamics

## June 26-30, 2023

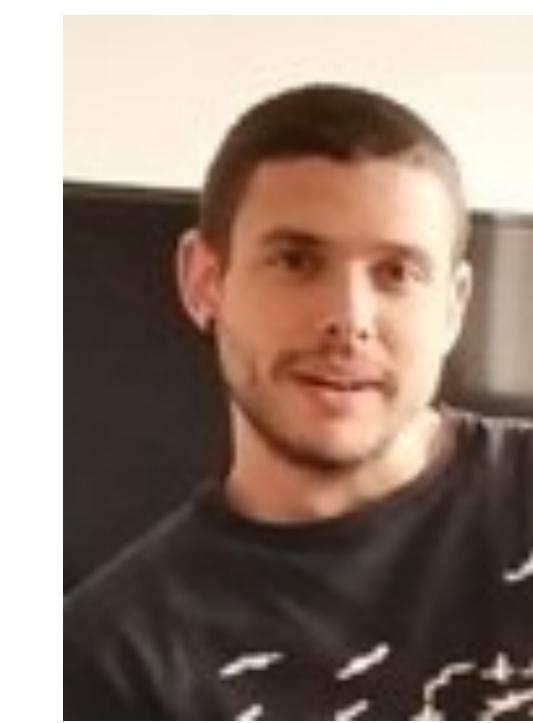
### Brest



Ronan  
Fablet



Lucas  
Drumetz



Carlos Granero  
Belinchon



Bruno  
Deremble



Audrey  
Monsimer



Emmanuel  
Cosme

### Grenoble

# Objective

---

- Introduction of the main Deep learning concepts (we think relevant for OA sciences)
- Introduction to PyTorch Deep Learning “ecosystem”
- Training through practice (lab sessions and project sessions)
- ability to deploy a deep learning approach for OA-related topics

# Program

<p><b>Day #1, 9.30am-12.30pm</b></p> <p><b>Introduction to Deep Learning</b></p> <ul style="list-style-type: none"><li>• What's learning ?</li><li>• MLP / backprop</li><li>• Practice on toy regression/classification examples</li></ul>	<p><b>Day #1, 2.00pm-5.30pm</b></p> <p><b>Project session #1:</b></p> <ul style="list-style-type: none"><li>• Project selection</li><li>• Learning-based problem formulation</li><li>• Exploratory analysis/visualization of the considered dataset</li></ul>
<p><b>Day #2, 9.30am-12.30pm</b></p> <p><b>Convolutional Neural Networks</b></p> <ul style="list-style-type: none"><li>• From MLP to CNN</li><li>• Deep Learning methodology</li><li>• Practice on MNIST digit classification</li></ul>	<p><b>Day #2, 2.00pm-5.30pm</b></p> <p><b>Project session #2:</b></p> <ul style="list-style-type: none"><li>• Tutorial on PyTorch Lightning</li><li>• Selection of neural architectures</li><li>• Design of the training scheme</li></ul>
<p><b>Day #3, 9.30am-12.30pm</b></p> <p><b>Auto-encoders and Generative models</b></p> <ul style="list-style-type: none"><li>• Auto-encoder architectures (Dense AEs/PCA, Convolutional AEs, U-Net)</li><li>• Practice on MNIST dataset</li><li>• Opening towards generative models: VAE, NF, GAN</li></ul>	<p><b>Day #3, 2.00pm-5.30pm</b></p> <p><b>Project session #3:</b></p> <ul style="list-style-type: none"><li>• First results for a simple architecture</li><li>• Sensitivity analysis</li><li>• Updated architectures</li></ul>
<p><b>Day #4, 9.30am-12.30pm</b></p> <p><b>Recurrent Neural Networks</b></p> <ul style="list-style-type: none"><li>• RNN / LSTM</li><li>• Neural ODE / PINN</li><li>• Practice on L63 system</li></ul>	<p><b>Day #4, 2.00pm-5.30pm</b></p> <p><b>Project session #4:</b></p> <ul style="list-style-type: none"><li>• Optimization of the architecture</li><li>• Synthesis of the experiments, incl. the benchmarking of several architectures</li></ul>
<p><b>Day #5, 9.30am-12.30pm</b></p> <p><b>Deep Learning and Inverse Problems</b></p> <ul style="list-style-type: none"><li>• DL, AutoDiff and minimization</li><li>• Deep inverse models</li><li>• Deep unfolded architectures</li><li>• Practice on L63 system</li></ul>	<p><b>Day #5, 2.00pm-5.30pm</b></p> <p><b>Project session #5:</b></p> <ul style="list-style-type: none"><li>• Presentation for each project (10'+5' for each project)</li></ul>

- Book: Deep Learning

Goodfellow, Bengio, Courville, MIT Press

Online version <http://www.deeplearningbook.org/>

- Online course by Andrew Ng (Stanford/Baidu)

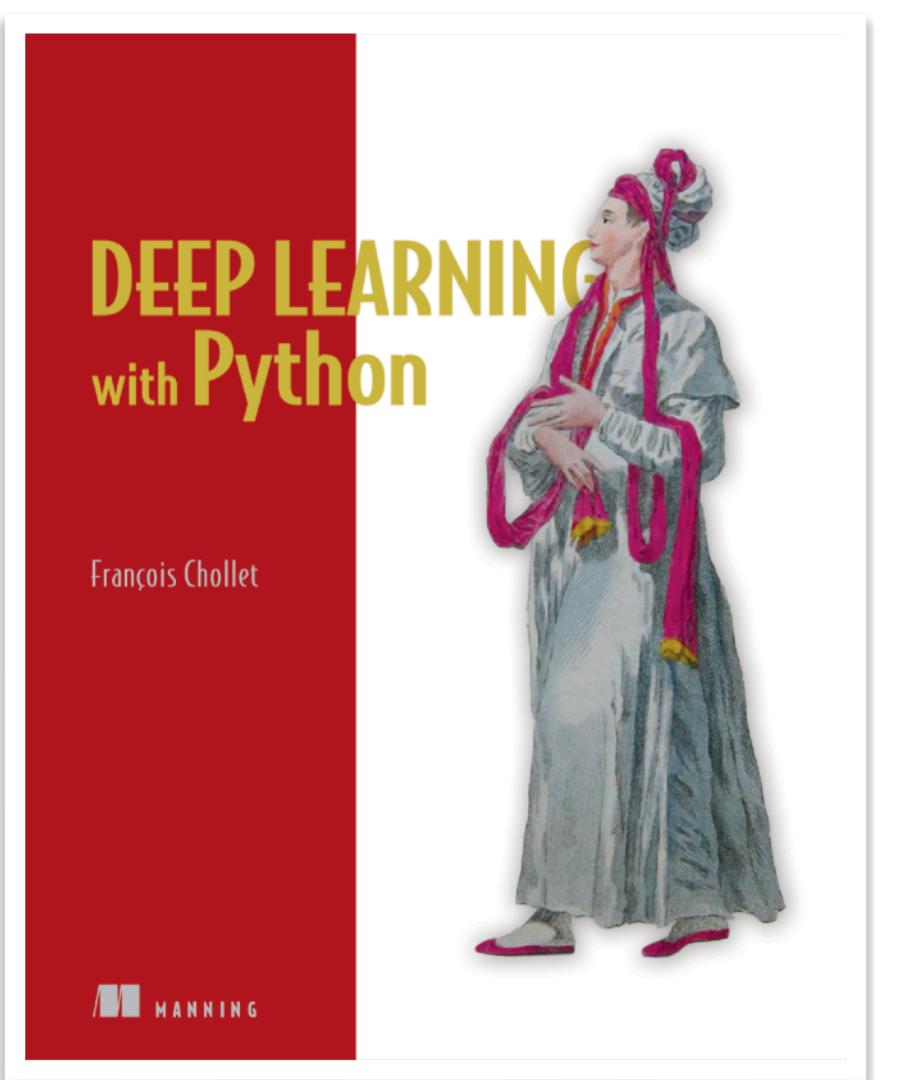
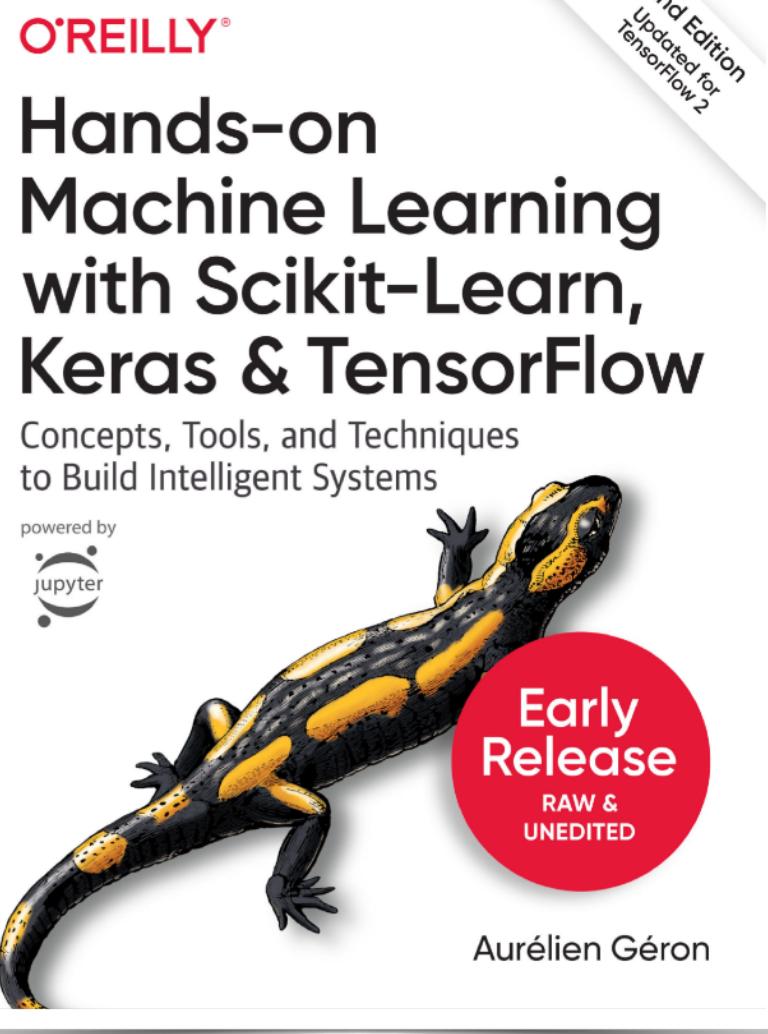
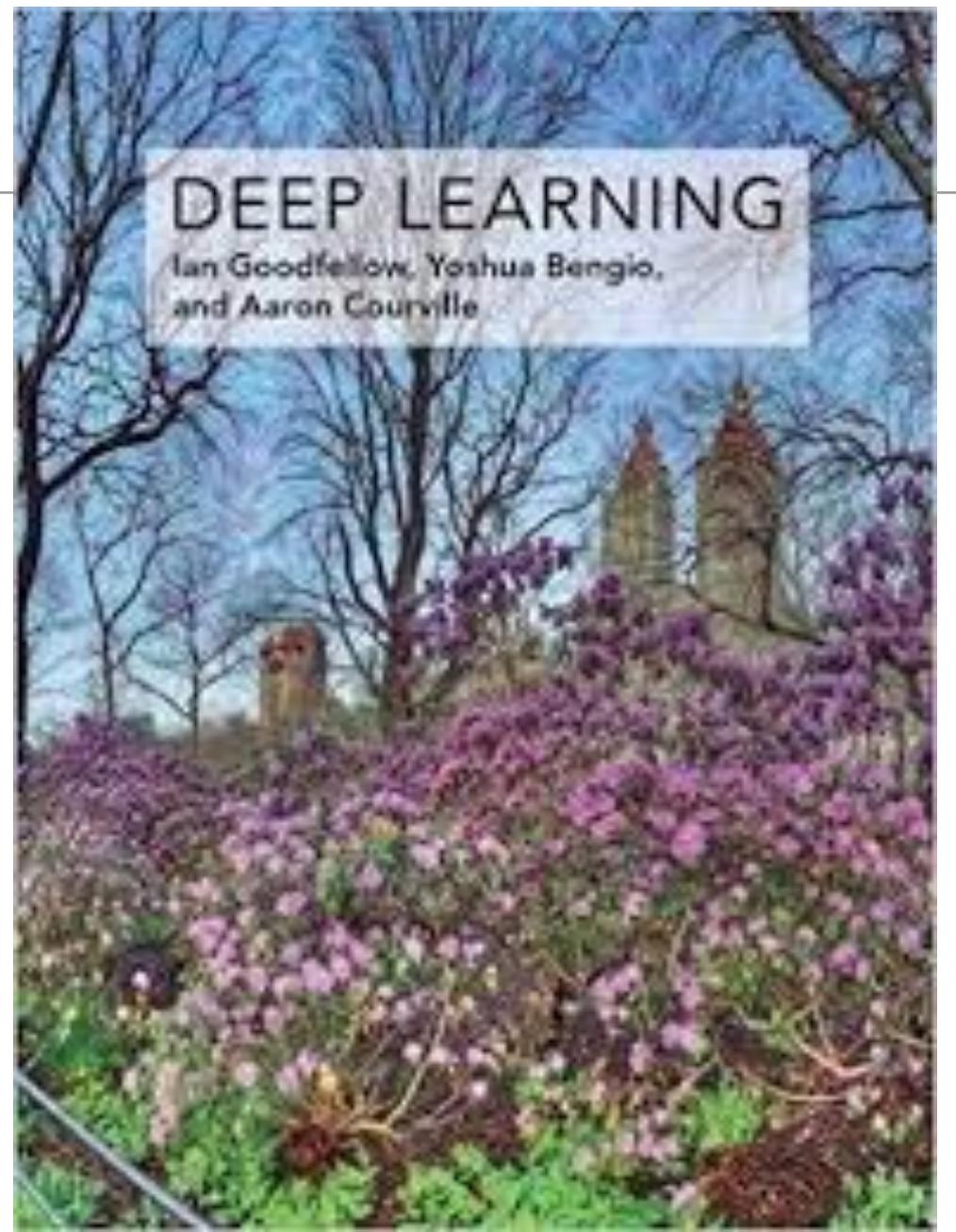
Youtube: [link](#)

Online course on Coursera: [link](#)

- Review paper: Deep learning in neural networks by

J. Schmidhuber pdf: [link](#)

- Github repo: <https://github.com/CIA-Oceanix/DLOA2023>



Neural Networks 61 (2015) 85–117

Contents lists available at ScienceDirect

Neural Networks

journal homepage: [www.elsevier.com/locate/neunet](http://www.elsevier.com/locate/neunet)

CrossMark

**Review**

**Deep learning in neural networks: An overview**

Jürgen Schmidhuber

The Swiss AI Lab IDSIA, Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, University of Lugano & SUPSI, Galleria 2, 6928 Manno-Lugano, Switzerland

**ARTICLE INFO**

**Article history:**  
Received 2 May 2014  
Received in revised form 12 September 2014  
Accepted 14 September 2014  
Available online 13 October 2014

**Keywords:**  
Deep learning  
Supervised learning  
Unsupervised learning  
Reinforcement learning  
Evolutionary computation

In recent years, deep artificial neural networks (including recurrent ones) have won numerous contests in pattern recognition and machine learning. This historical survey compactly summarizes relevant work, much of it from the previous millennium. Shallow and Deep Learners are distinguished by the depth of their *credit assignment paths*, which are chains of possibly learnable, causal links between actions and effects. I review deep supervised learning (also recapitulating the history of backpropagation), unsupervised learning, reinforcement learning & evolutionary computation, and indirect search for short programs encoding deep and large networks.

© 2014 Published by Elsevier Ltd.

# Requirements

---

- access to discord server OK?
- google colab account?

# Lecture #1: The basics

- Deep Learning facts
  - definition, emergence, uses, culture, softwares, carbon footprint
- ~~How learning works~~

---

  - description, mathematical formulation, basic concepts, vocabulary
- The Multi Layer Perceptron
  - Perceptron, activation functions, feedforward networks, layers...
- The learning process
  - Loss function, Gradient descent, backpropagation
- Introduction to Pytorch
  - Basic tips and practice

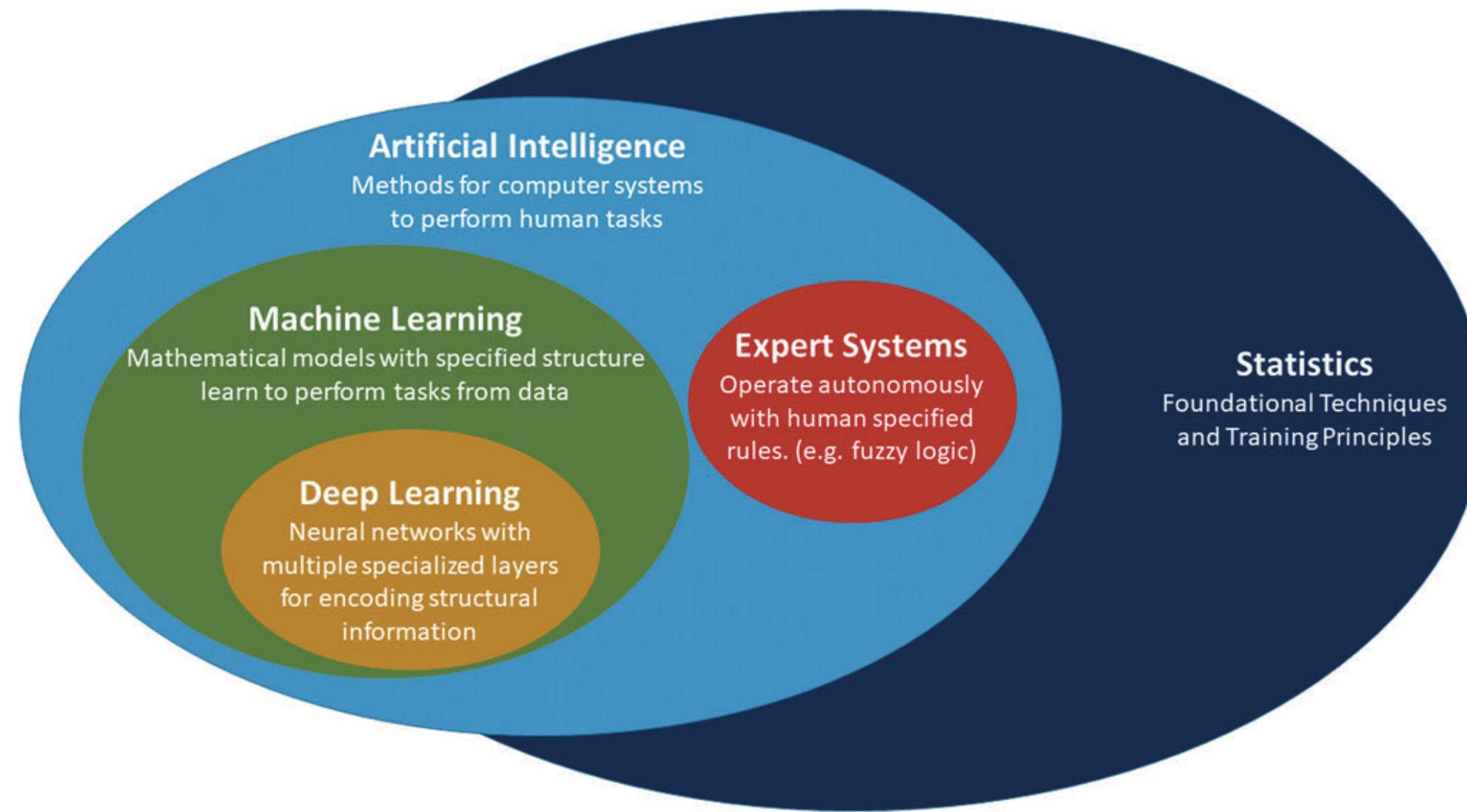
# Deep Learning facts

---

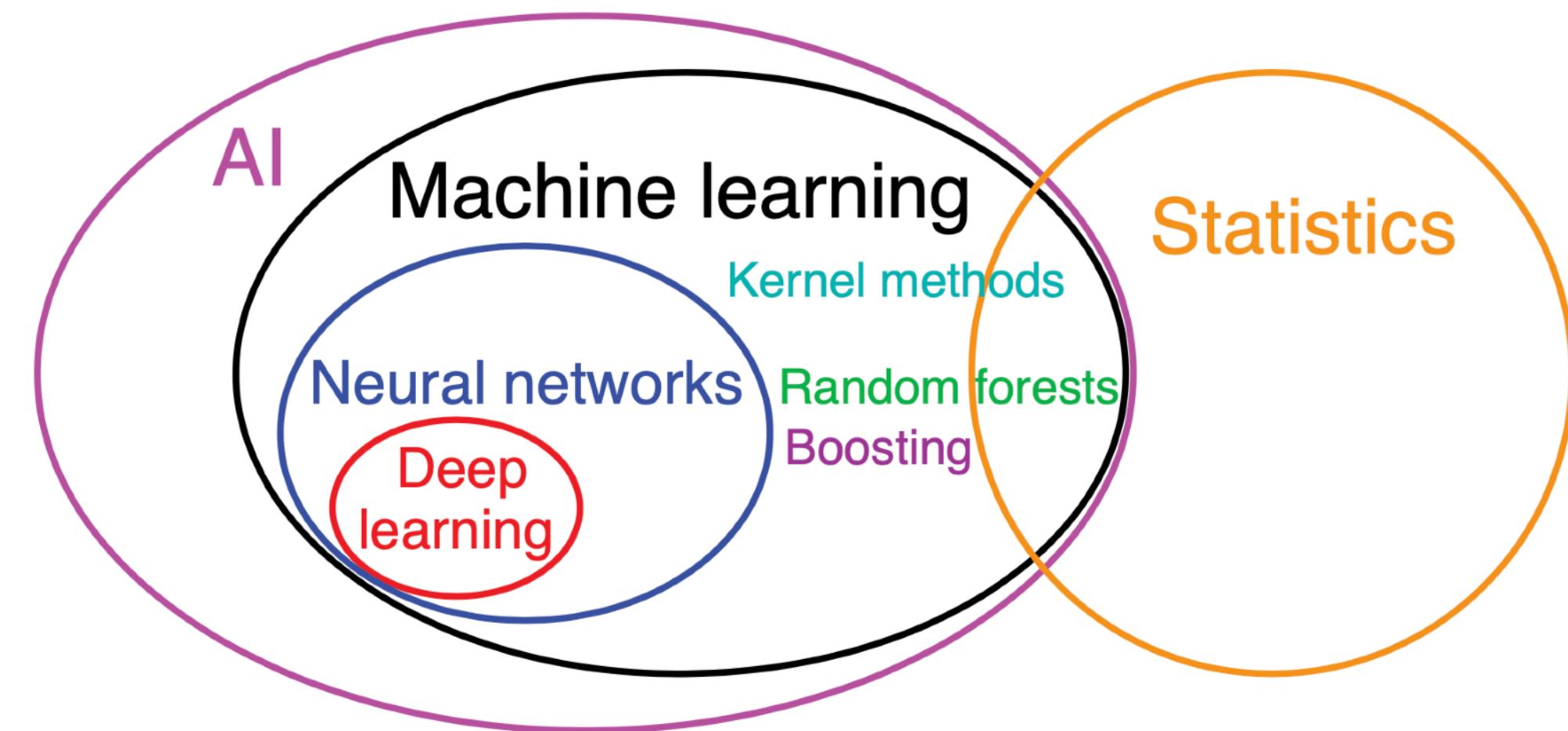
Definition, emergence, uses, culture, softwares, carbon footprint

# AI, Machine Learning, Deep Learning, etc

---



From Haupt et al (2022)



From Hsieh (2022)

Deep learning has become the standard framework for a wide range of applications in computer vision, natural language processing, signal processing...

# Machine Learning and "traditional" simulation

---

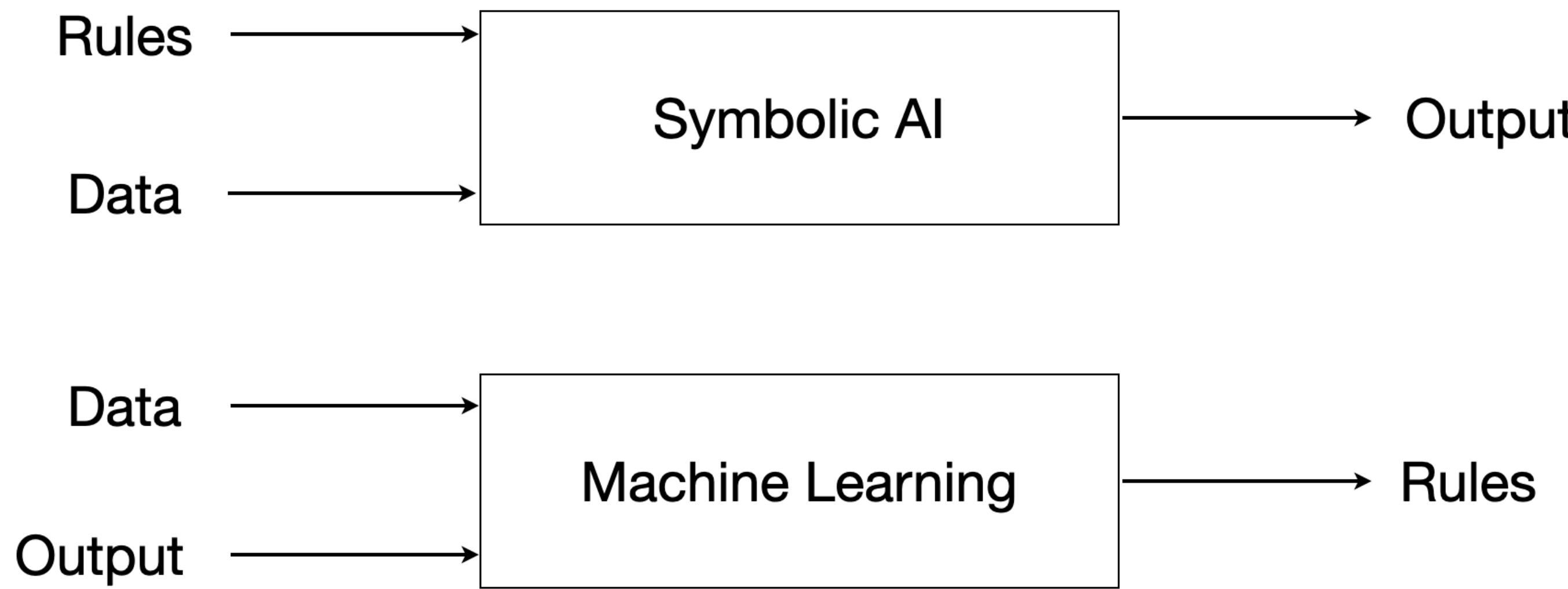
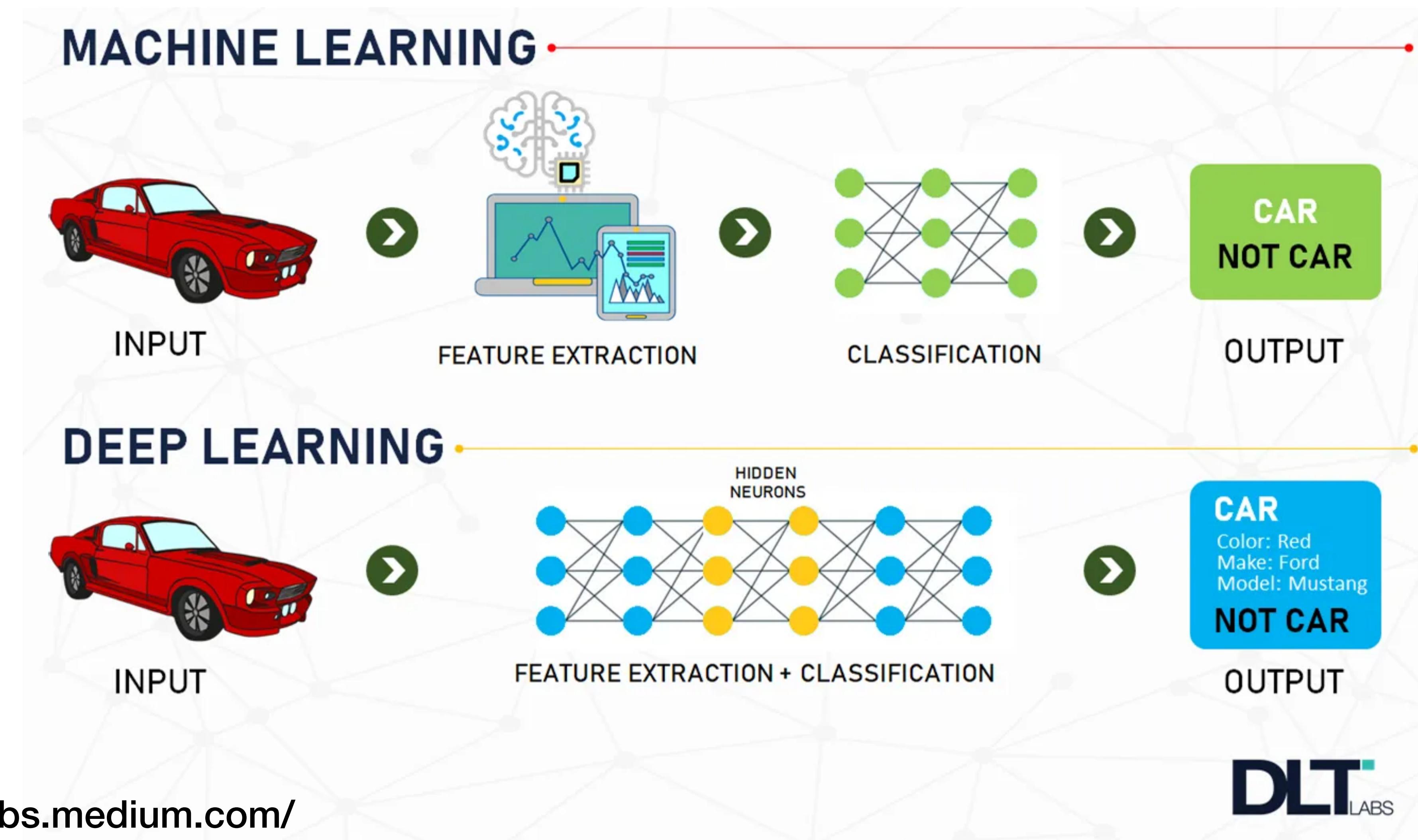


Figure largely inspired by Chollet (2017)

# Machine Learning vs Deep Learning

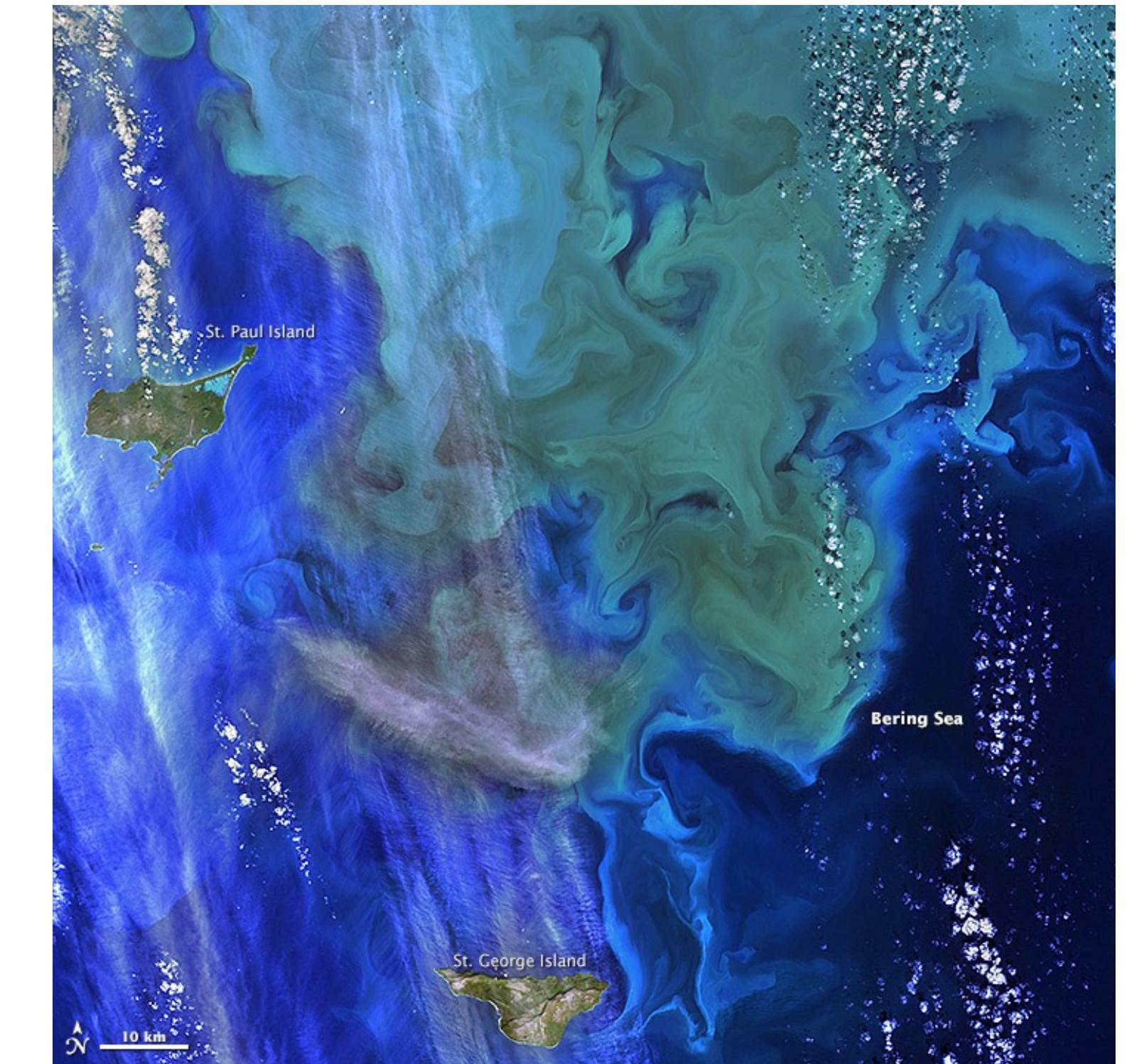


# Emergence of deep learning

$$\begin{aligned}\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} &= fv - \frac{1}{\rho} \frac{\partial p}{\partial x} + K_u \frac{\partial^2 u}{\partial z^2} \\ \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} &= -fu - \frac{1}{\rho} \frac{\partial p}{\partial y} + K_v \frac{\partial^2 v}{\partial z^2} \\ -\frac{\partial p}{\partial z} &= \rho g\end{aligned}$$

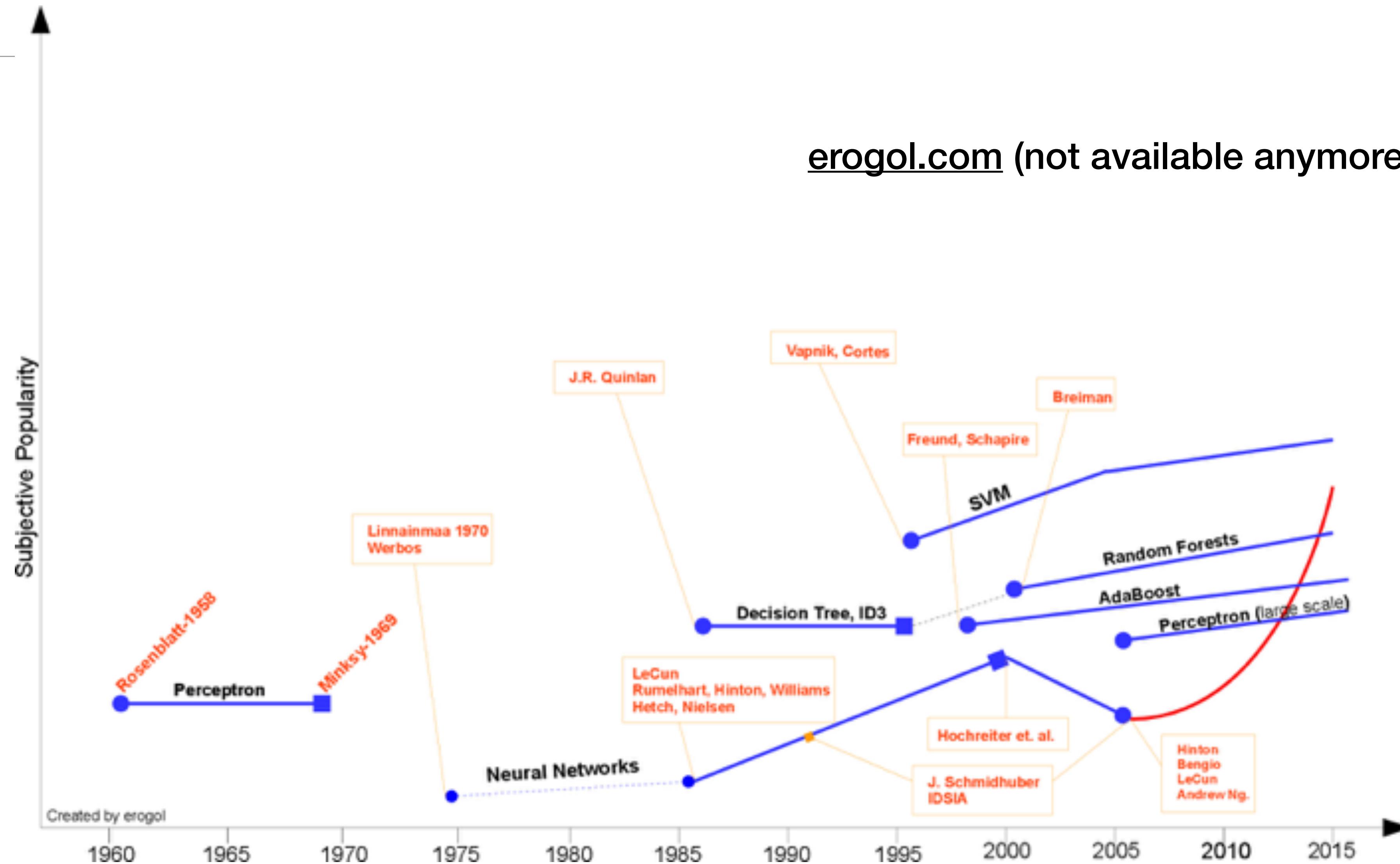
Can you solve these equations  
to picture the ocean circulation?

What is on this picture?

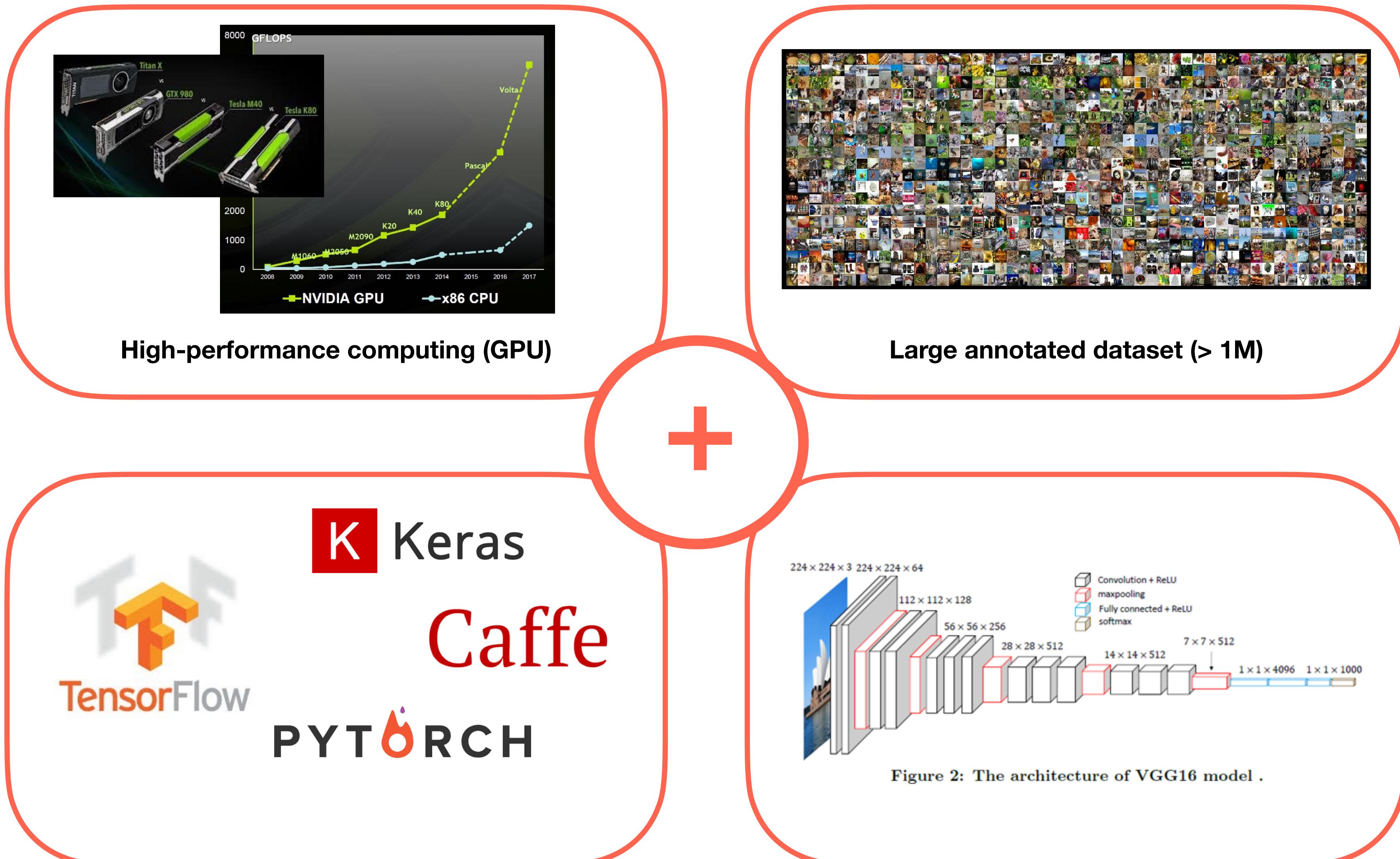


This is a satellite view of phytoplankton at  
the ocean surface. Any idea of how  
surface currents are?

# Emergence of deep learning



# Emergence of deep learning



# An history of DL

Neural Networks 61 (2015) 85–117

Contents lists available at ScienceDirect

**Neural Networks**

journal homepage: [www.elsevier.com/locate/neunet](http://www.elsevier.com/locate/neunet)





---

**Review**

**Deep learning in neural networks: An overview**

 CrossMark

**Jürgen Schmidhuber**

*The Swiss AI Lab IDSIA, Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, University of Lugano & SUPSI, Galleria 2, 6928 Manno-Lugano, Switzerland*

---

**ARTICLE INFO**

*Article history:*  
Received 2 May 2014  
Received in revised form 12 September 2014  
Accepted 14 September 2014  
Available online 13 October 2014

---

*Keywords:*  
Deep learning  
Supervised learning  
Unsupervised learning  
Reinforcement learning  
Evolutionary computation

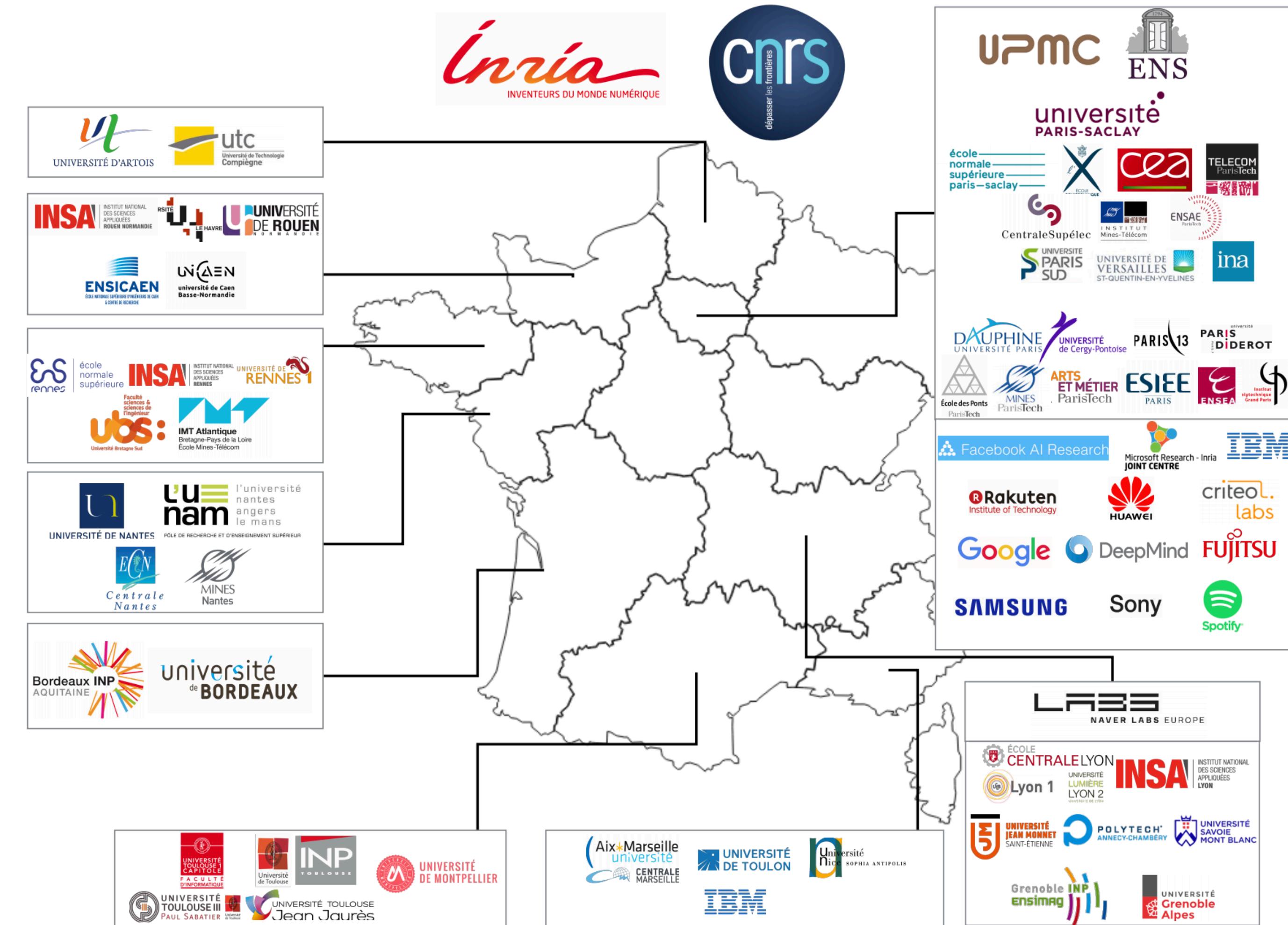
---

**ABSTRACT**

In recent years, deep artificial neural networks (including recurrent ones) have won numerous contests in pattern recognition and machine learning. This historical survey compactly summarizes relevant work, much of it from the previous millennium. Shallow and Deep Learners are distinguished by the depth of their *credit assignment paths*, which are chains of possibly learnable, causal links between actions and effects. I review deep supervised learning (also recapitulating the history of backpropagation), unsupervised learning, reinforcement learning & evolutionary computation, and indirect search for short programs encoding deep and large networks.

© 2014 Published by Elsevier Ltd.

# Today's research in AI in France

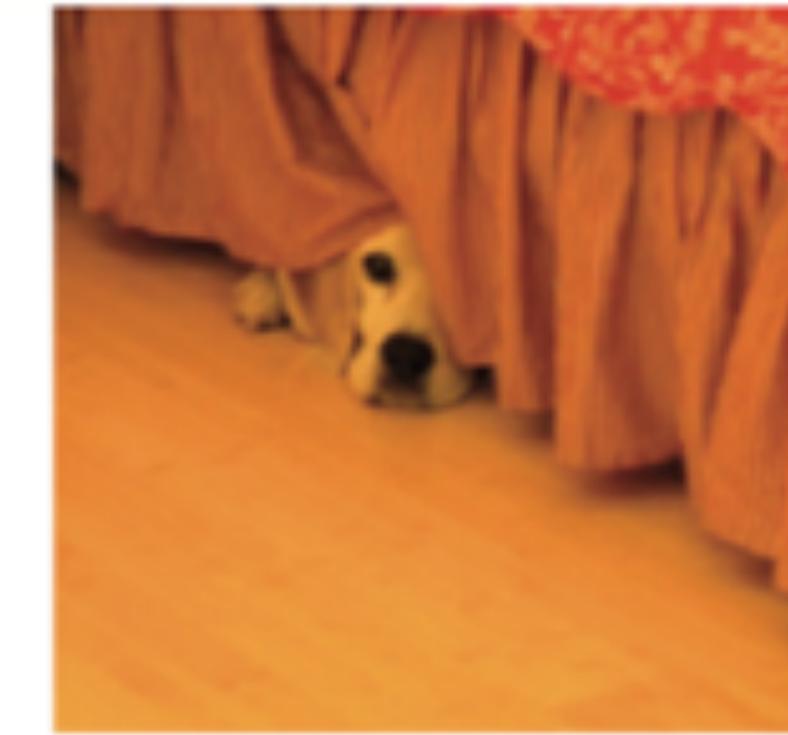


# Image captioning (image to text)

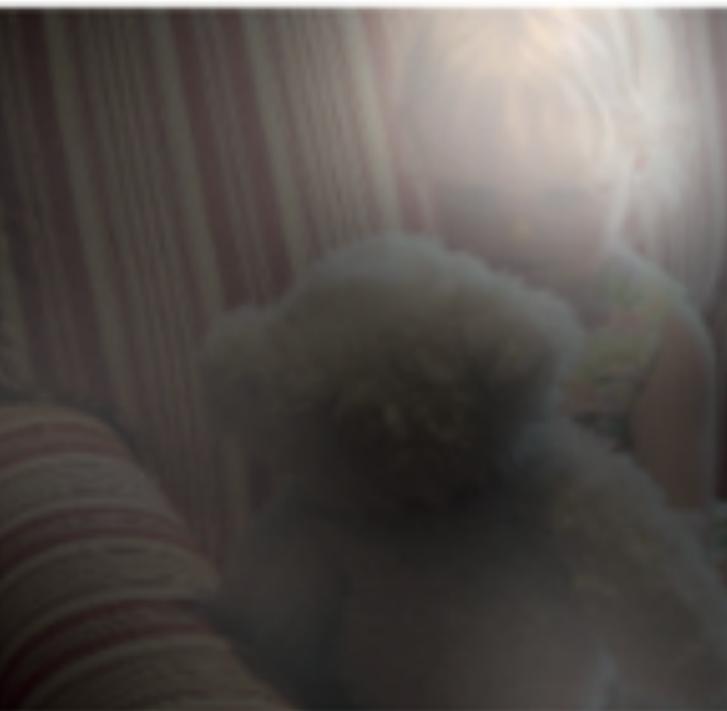
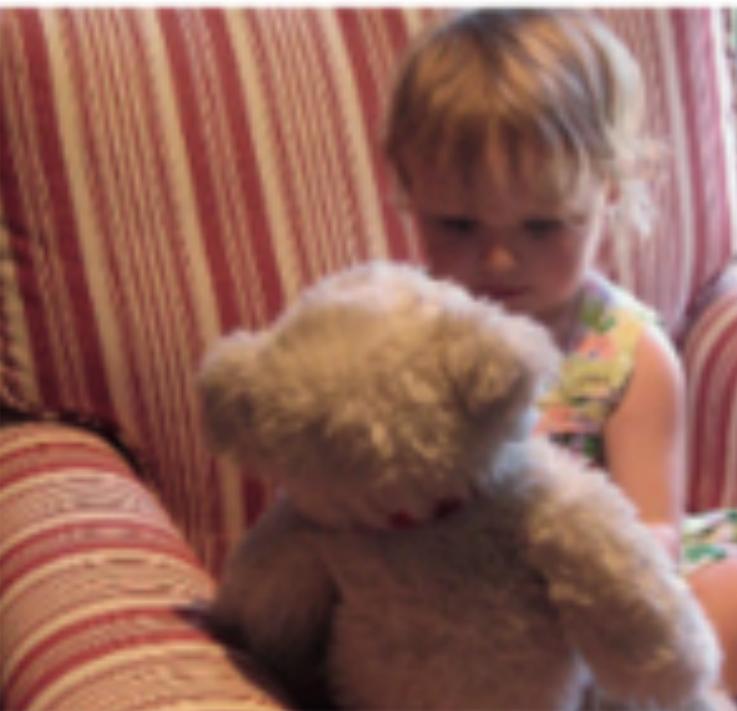
---



A woman is throwing a **frisbee** in a park.



A **dog** is standing on a hardwood floor.



A little **girl** sitting on a bed with a teddy bear.



A group of **people** sitting on a boat in the water.

... and generation (text to image)

---



**Imagine if the best AI models  
were open and free.**

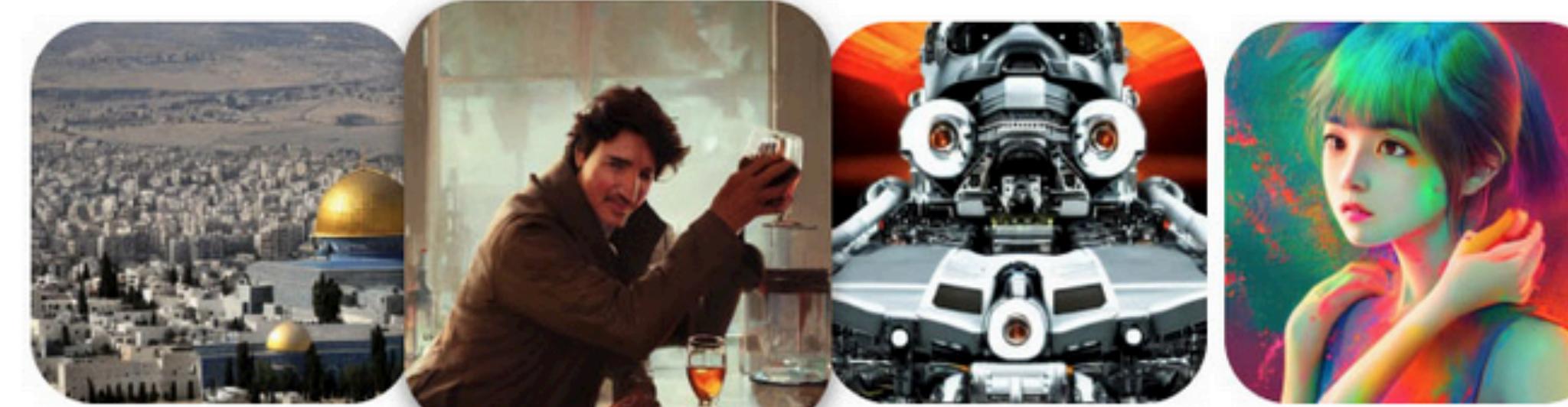
**What would you do with them?**

Open AI is working to make the  
world's best AI models open.

In the meantime...

**Generate any image imaginable.**

Generate



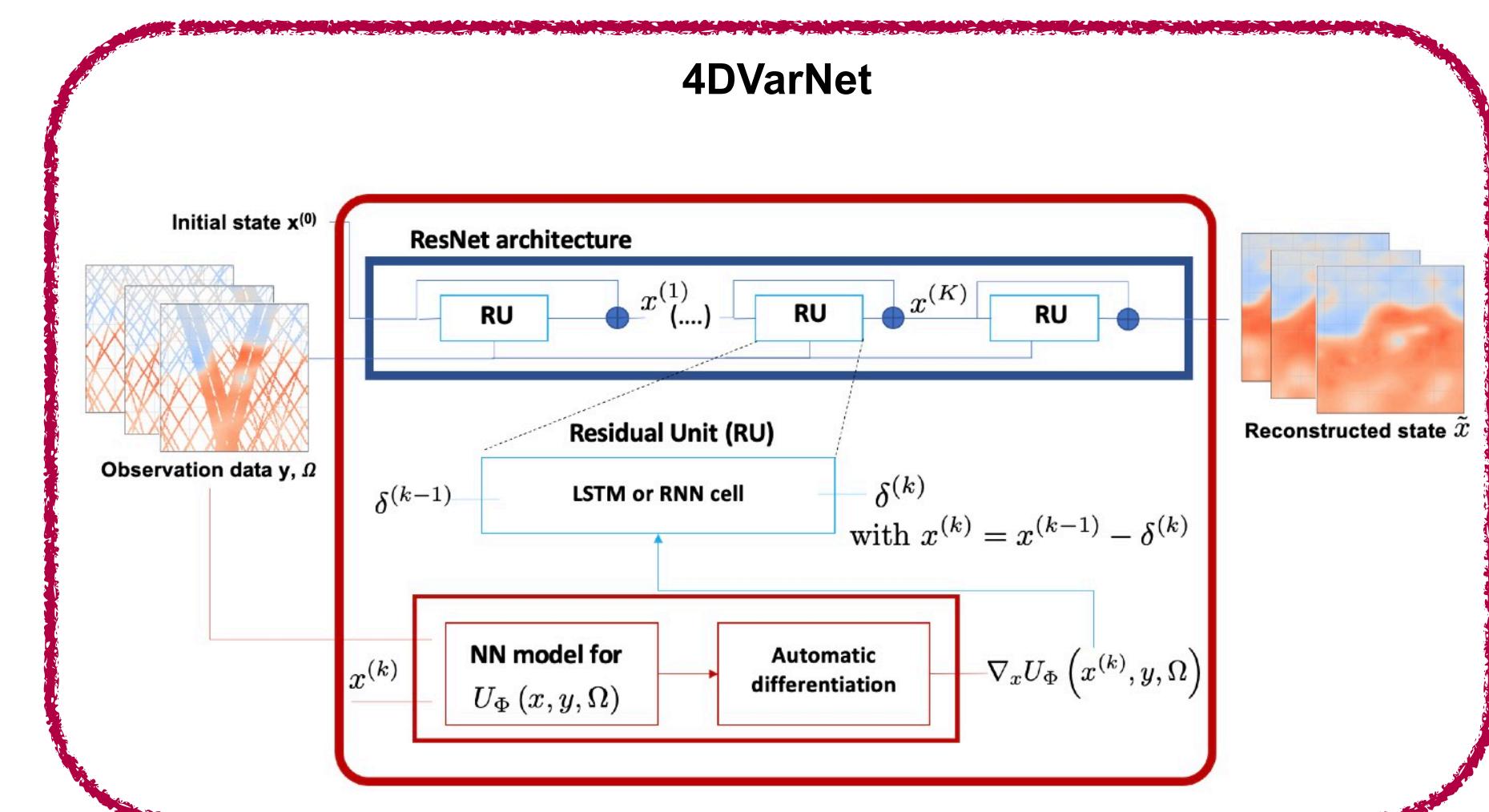
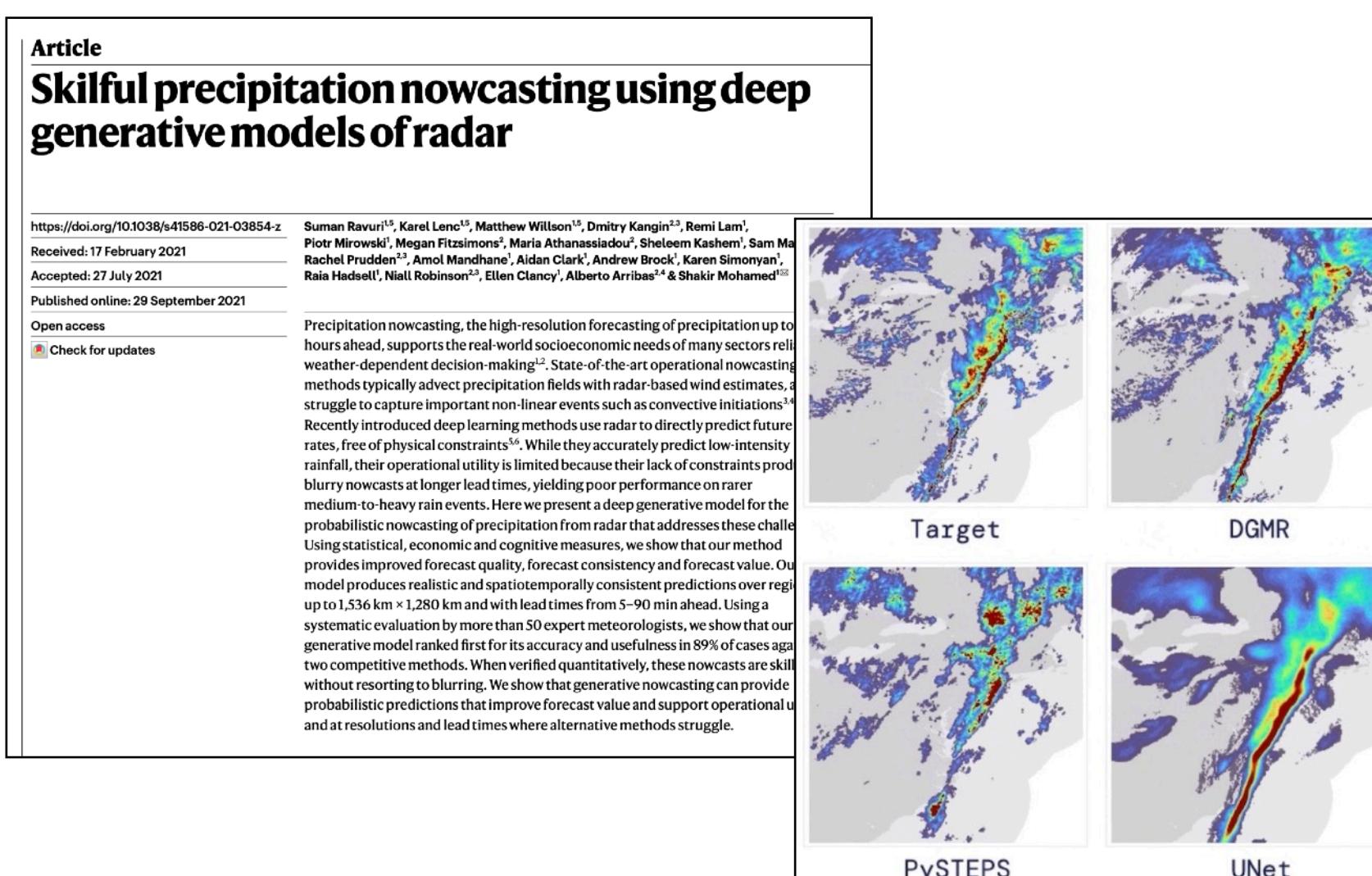
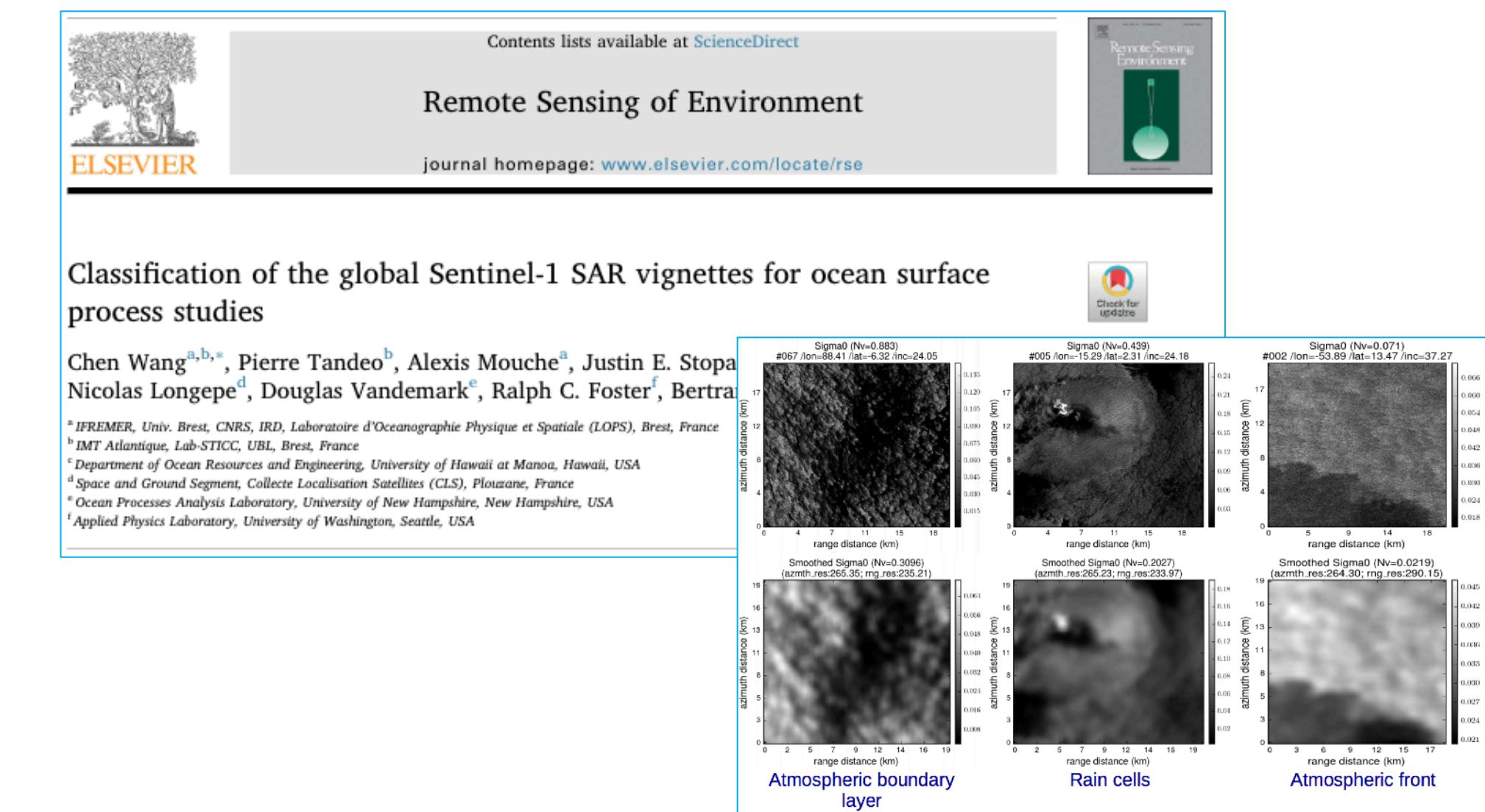
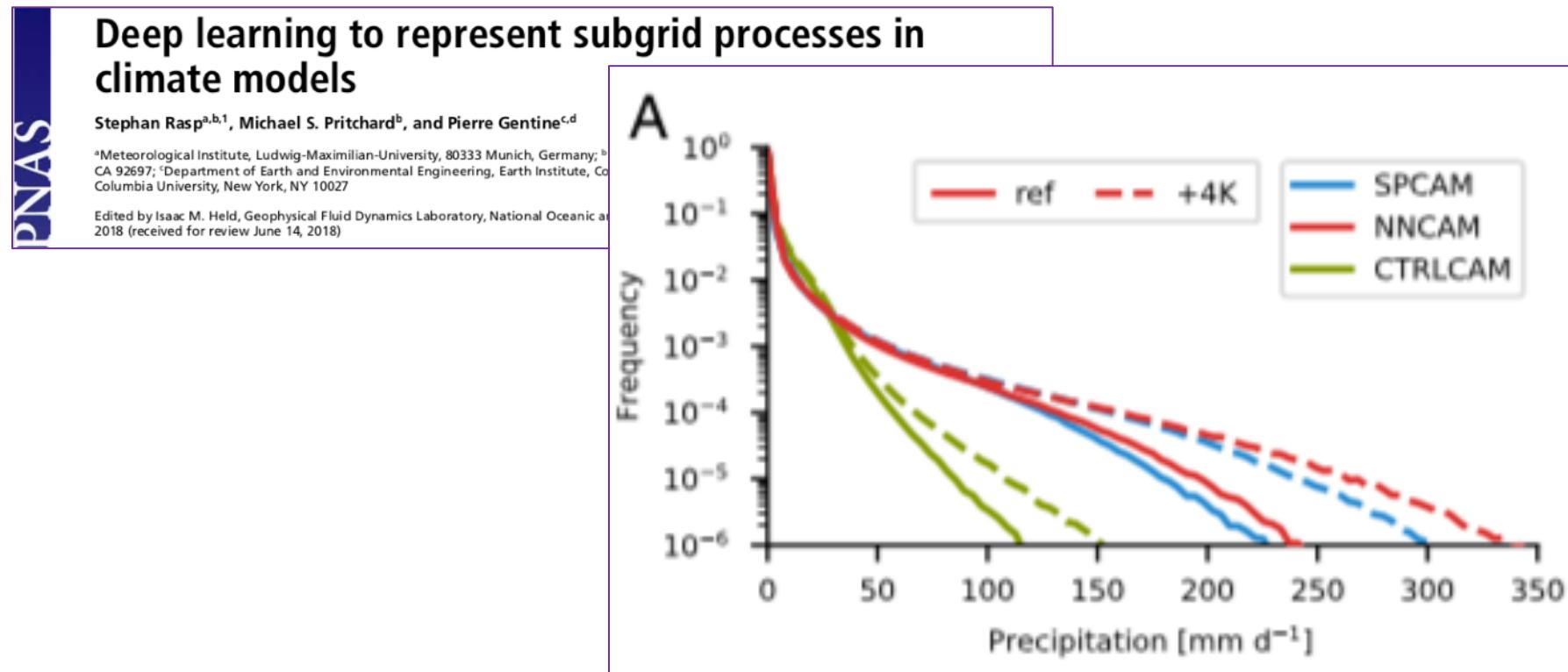
justin trudeau style, highly detailed, digital painting, artstation, concept art, wallpaper, smooth, sharp focus, illustration, art by artgerm and greg rutkowski and alphonse mucha trending on

# Interactive discussions... (ChatGPT)

---

- for summaries
- brainstorming
- get a quick answer to a question
- exploring
- wasting your time...

# In OA sciences



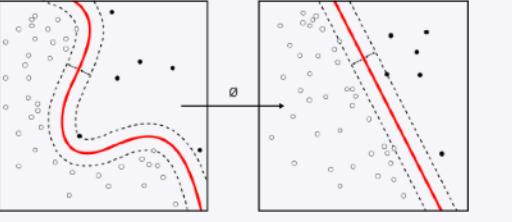
# Classical datasets

## List of datasets for machine-learning research

From Wikipedia, the free encyclopedia

These [datasets](#) are applied for [machine learning](#) research and have been cited in [peer-reviewed](#) academic journals. Datasets are an integral part of the field of machine learning. Major advances in this field can result from advances in learning [algorithms](#) (such as [deep learning](#)), computer hardware, and, less-intuitively, the availability of high-quality training datasets.<sup>[1]</sup> High-quality labeled training datasets for [supervised](#) and [semi-supervised](#) machine learning algorithms are usually difficult and expensive to produce because of the large amount of time needed to label the data. Although they do not need to be labeled, high-quality datasets for [unsupervised](#) learning can also be difficult and costly to produce.<sup>[2][3][4][5]</sup>

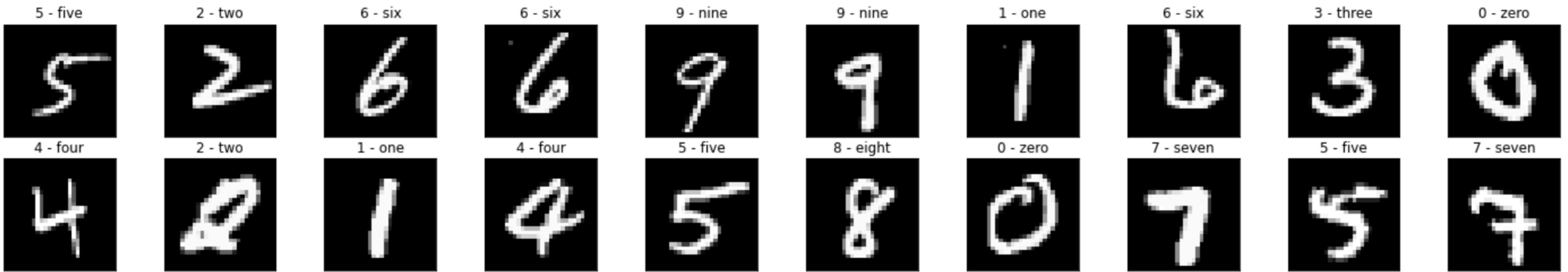
<b>Contents [hide]</b>
1 <a href="#">Image data</a> <ul style="list-style-type: none"><li>1.1 <a href="#">Facial recognition</a></li><li>1.2 <a href="#">Action recognition</a></li><li>1.3 <a href="#">Object detection and recognition</a></li><li>1.4 <a href="#">Handwriting and character recognition</a></li><li>1.5 <a href="#">Aerial images</a></li><li>1.6 <a href="#">Other images</a></li></ul>
2 <a href="#">Text data</a> <ul style="list-style-type: none"><li>2.1 <a href="#">Reviews</a></li><li>2.2 <a href="#">News articles</a></li><li>2.3 <a href="#">Messages</a></li><li>2.4 <a href="#">Twitter and tweets</a></li><li>2.5 <a href="#">Dialogues</a></li><li>2.6 <a href="#">Other text</a></li></ul>
3 <a href="#">Sound data</a> <ul style="list-style-type: none"><li>3.1 <a href="#">Speech</a></li><li>3.2 <a href="#">Music</a></li><li>3.3 <a href="#">Other sounds</a></li></ul>
4 <a href="#">Signal data</a> <ul style="list-style-type: none"><li>4.1 <a href="#">Electrical</a></li><li>4.2 <a href="#">Motion-tracking</a></li><li>4.3 <a href="#">Other signals</a></li></ul>
5 <a href="#">Physical data</a> <ul style="list-style-type: none"><li>5.1 <a href="#">High-energy physics</a></li><li>5.2 <a href="#">Systems</a></li><li>5.3 <a href="#">Astronomy</a></li><li>5.4 <a href="#">Earth science</a></li><li>5.5 <a href="#">Other physical</a></li></ul>
6 <a href="#">Biological data</a> <ul style="list-style-type: none"><li>6.1 <a href="#">Human</a></li><li>6.2 <a href="#">Animal</a></li><li>6.3 <a href="#">Fungi</a></li><li>6.4 <a href="#">Plant</a></li><li>6.5 <a href="#">Microbe</a></li><li>6.6 <a href="#">Drug Discovery</a></li></ul>
7 <a href="#">Anomaly data</a>
8 <a href="#">Question Answering data</a>
9 <a href="#">Multivariate data</a> <ul style="list-style-type: none"><li>9.1 <a href="#">Financial</a></li><li>9.2 <a href="#">Weather</a></li><li>9.3 <a href="#">Census</a></li></ul>

Part of a series on	
<b>Machine learning and data mining</b>	
	<b>Problems</b> [show]
	<b>Supervised learning</b> [show] (classification • regression)
	<b>Clustering</b> [show]
	<b>Dimensionality reduction</b> [show]
	<b>Structured prediction</b> [show]
	<b>Anomaly detection</b> [show]
	<b>Artificial neural network</b> [show]
	<b>Reinforcement learning</b> [show]
	<b>Learning with humans</b> [show]
	<b>Model diagnostics</b> [show]
	<b>Theory</b> [show]
	<b>Machine-learning venues</b> [show]
	<b>Related articles</b> [show]

V · T · E

# Classical datasets: MNIST

---



# Softwares

---

- Python
- numpy
- matplotlib
- pandas
- tensorflow/keras
- PyTorch, pytorch-lightning
- Scikit-learn
- xarray
- and many others

# Data challenges

**kaggle** Competitions Datasets Code Discussions Courses ...

Search Sign In Register

Inside Kaggle you'll find all the code & data you need to do your data science work. Use over 50,000 public datasets and 400,000 public notebooks to conquer any analysis in no time.

Maintained by Kaggle Starter Code Finance Datasets Linguistics Datasets Data Visualization Kernels

The screenshot shows the Kaggle homepage with a search bar and navigation links. Below the main heading, there's a large text block about the platform's offerings. At the bottom, there are four dataset cards with images and titles:

- Financial Tweets** by David Wallach: CSV Dataset | 50 upvotes
- Face Detection in Images** by DataTurks: JSON Dataset | 68 upvotes
- Star Trek Scripts** by Gary Broughton: JSON Dataset | 12 upvotes
- Avocado Prices** by Justin Kiggins: CSV Dataset | 546 upvotes

# Data challenges



- Home
- Competitions
- Datasets
- Code
- Discussions
- Learn
- More

GettingStarted Prediction Competition

## Titanic - Machine Learning from Disaster

Start here! Predict survival on the Titanic and get familiar with ML basics

Kaggle · 13,209 teams · Ongoing

Overview Data Code Discussion Leaderboard Rules

Join Competition

### Overview

#### Description

Ahoy, welcome to Kaggle! You're in the right place.

#### Evaluation

This is the legendary Titanic ML competition – the best, first challenge for you to dive into ML competitions and familiarize yourself with how the Kaggle platform works.

#### Frequently Asked Questions

The competition is simple: use machine learning to create a model that predicts which passengers survived the Titanic shipwreck.

Read on or watch the video below to explore more details. Once you're ready to start competing, click on the "Join Competition button" to create an account and gain access to the [competition data](#). Then check out [Alexis Cook's Titanic Tutorial](#) that walks you through step by step how to make your first submission!



### The Challenge

The sinking of the Titanic is one of the most infamous shipwrecks in history.

On April 15, 1912, during her maiden voyage, the widely considered "unsinkable" RMS Titanic sank after colliding with an iceberg. Unfortunately, there weren't enough lifeboats for everyone onboard, resulting in the death of 1502 out of 2224 passengers and crew.

While there was some element of luck involved in surviving, it seems some groups of people were more likely to survive than others.

In this challenge, we ask you to build a predictive model that answers the question: "what sorts of people were more likely to survive?" using passenger data (ie name, age, gender, socio-economic class, etc).

<https://github.com/ocean-data-challenges>

# Data challenges

Screenshot of the GitHub repository page for "ocean-data-challenges".

The repository name is "Ocean Data Challenges". Description: "Hosting collaborative data challenge related to ocean sciences.".

Pinned repositories:

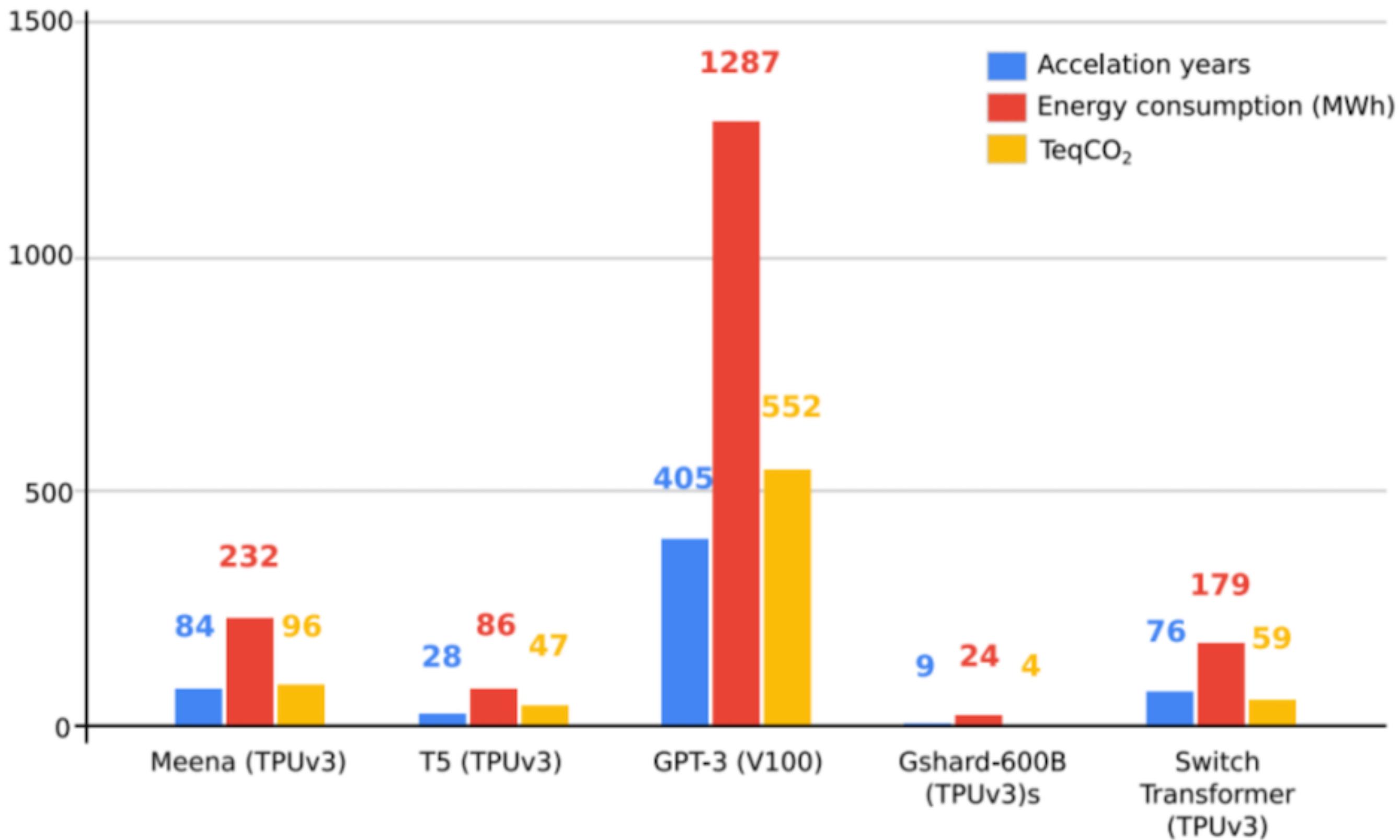
- 2020a\_SSH\_mapping\_NATL60** (Public)  
A challenge on the mapping of satellite altimeter sea surface height data organised by MEOM@IGE, Ocean-Next and CLS.  
Jupyter Notebook, 18 stars, 7 forks.
- 2021a\_SSH\_mapping\_OSE** (Public)  
A challenge on the mapping of real satellite altimeter sea surface height data organised by MEOM@IGE, Ocean-Next and CLS.  
Jupyter Notebook, 3 stars, 4 forks.
- 2022a\_SWOT\_karin\_error\_filtering** (Public)  
A challenge on the SWOT Karin instrumental error filtering organised by Datlas, IMT Atlantique and CLS.  
Jupyter Notebook.

Customize pins, View as: Public, You are viewing the README and pinned repositories as a public user. You can create a README file visible to anyone. Get started with tasks that most successful organizations complete.

People:



# Energy consumption of AI



Quelques facteurs d'énergie utilisés pour l'entraînement de plusieurs réseaux de neurones modernes (ici pour le traitement du langage). Nombre d'années équivalent-GPU (bleu), facture électrique en MWh (pour information, 1MWh équivaut environ à 100 heures) en rouge, et **tonnage équivalent en CO<sub>2</sub> (pour information, 1TeqCO<sub>2</sub> équivaut à 1 A/R Paris-New York), en jaune**. Le simple coût d'entraînement d'un réseau de neurones sur une application ciblée est 250 fois supérieur au maximum annuel autorisé à chaque Européen (2TeqCO<sub>2</sub>) pour atteindre l'équilibre carbone en 2050.

Figure from Patterson et al, 2021, <https://arxiv.org/abs/2104.10350> ; Caption from Trystram et al, 2021

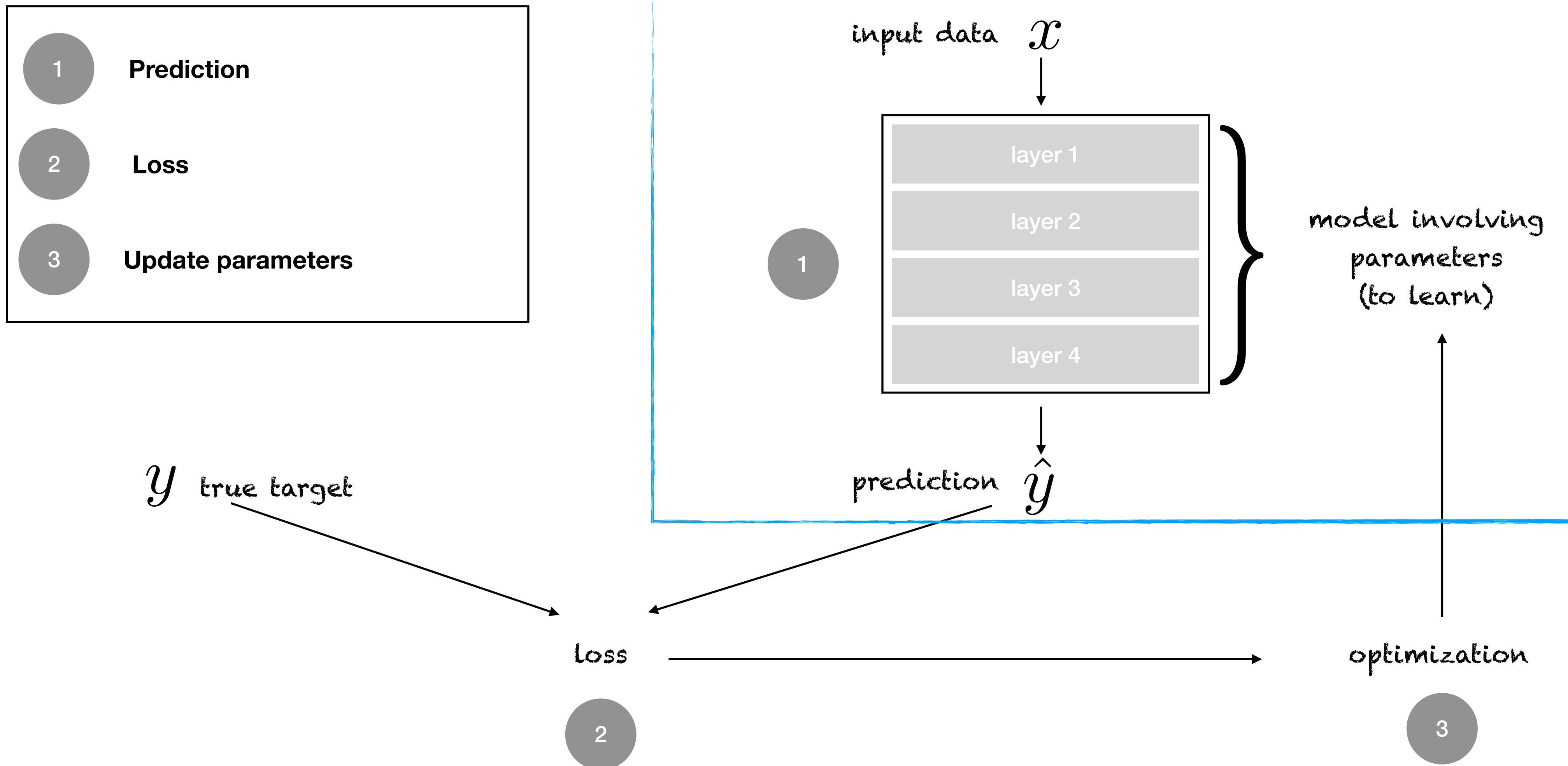
<https://theconversation.com/apprentissage-profound-et-consommation-energetique-la-partie-immergee-de-lia-ceberg-172341>

# How learning works

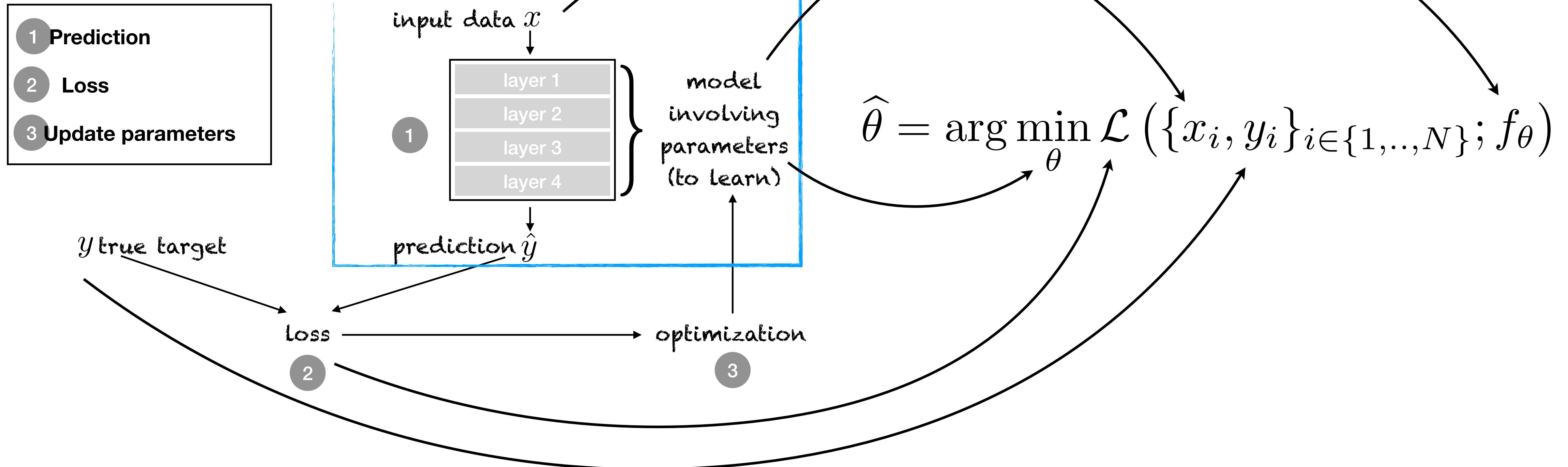
---

Description, mathematical formulation, basic concepts, vocabulary

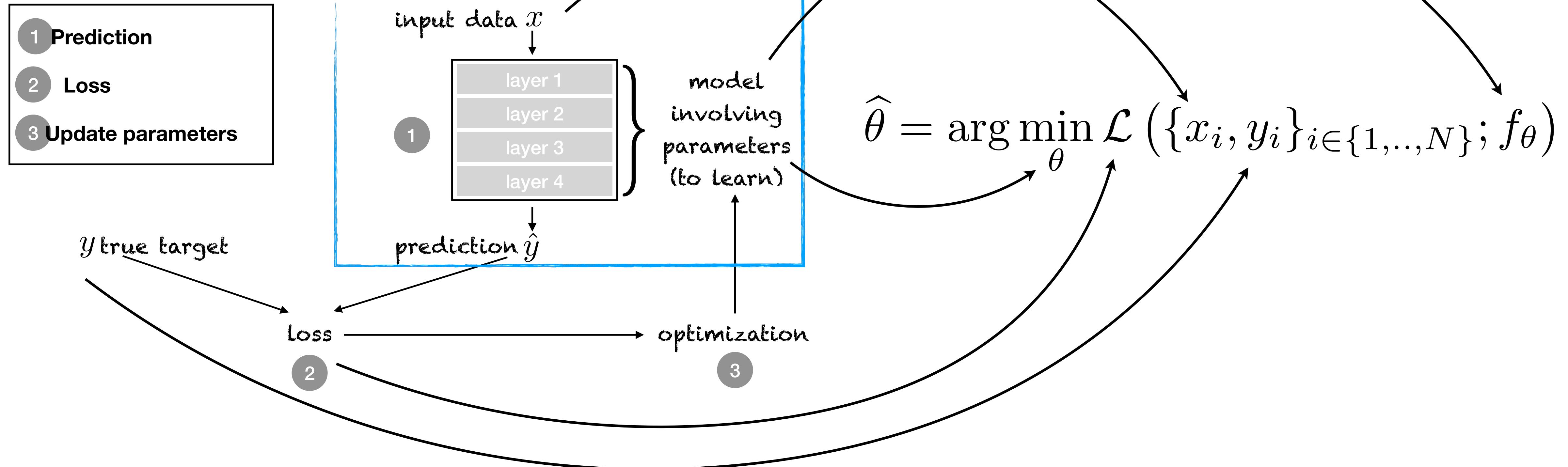
# Machine/Deep Learning



# Machine/Deep Learning



# Machine/Deep Learning



Key questions:

- how to design the model?
- How to choose the loss function?

# Machine Learning terminology

---

- Training and test dataset (Jeux de données d'apprentissage et test)
- Training loss (Coût d'apprentissage)
- Model (Modèle)
- Supervised learning (Apprentissage supervisé)
- Unsupervised learning (Apprentissage non-supervisé)
- Regression
- Classification

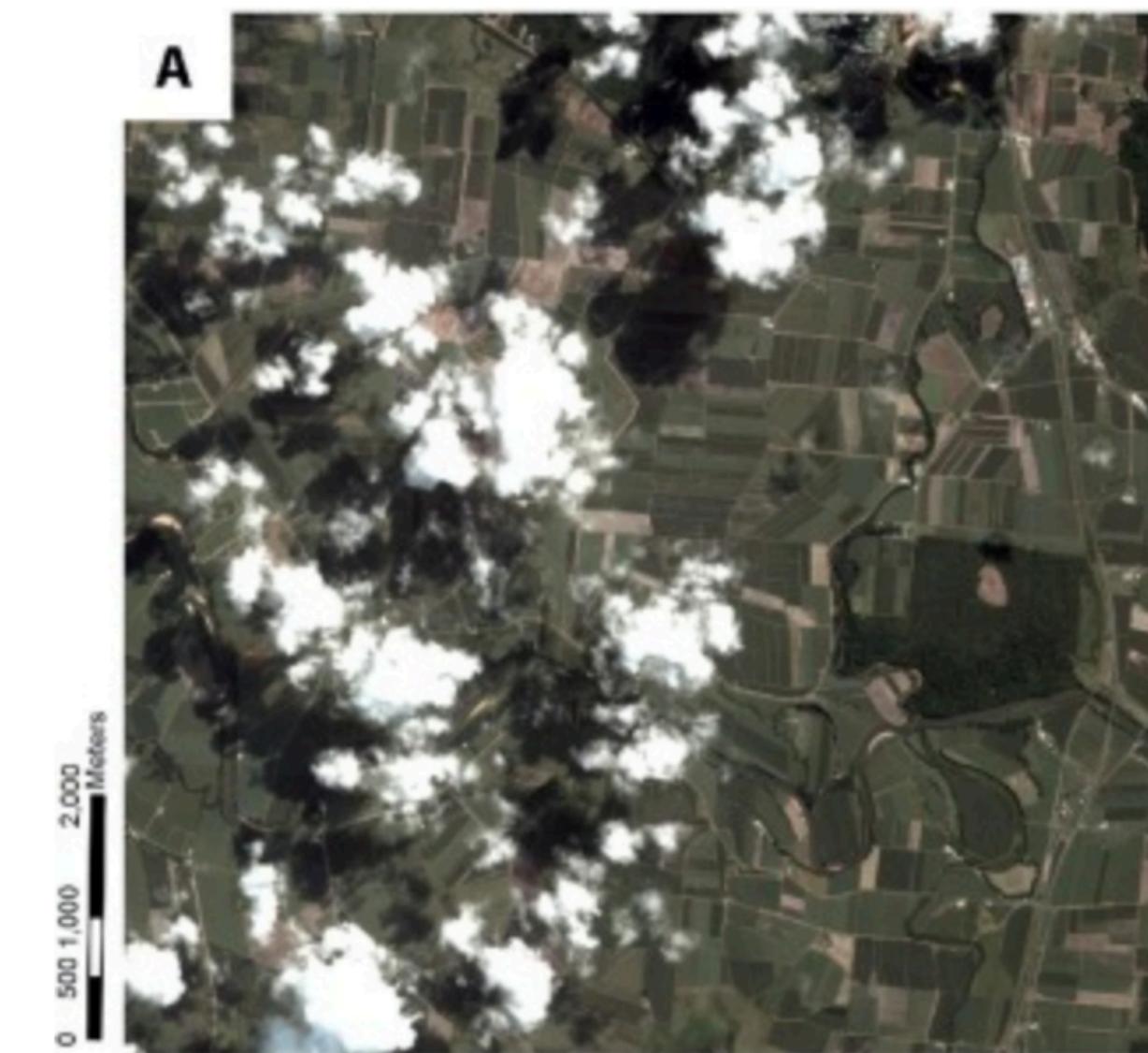
# Types of Machine Learning problems

---

**Classification** answers to:  
what (pre-defined) class does  
this item belong to?

**Supervised learning.**

**Binary** classification



?

Cloudy

Not Cloudy

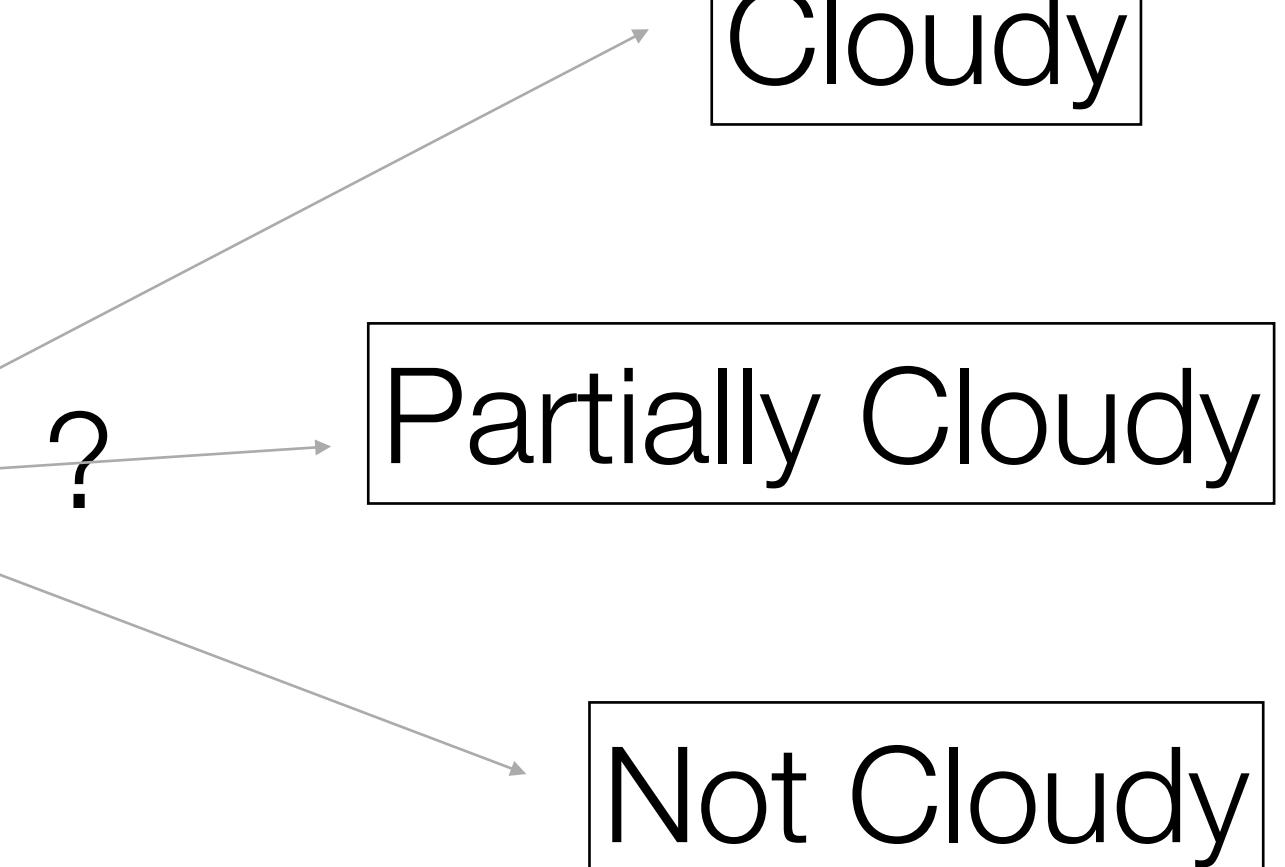
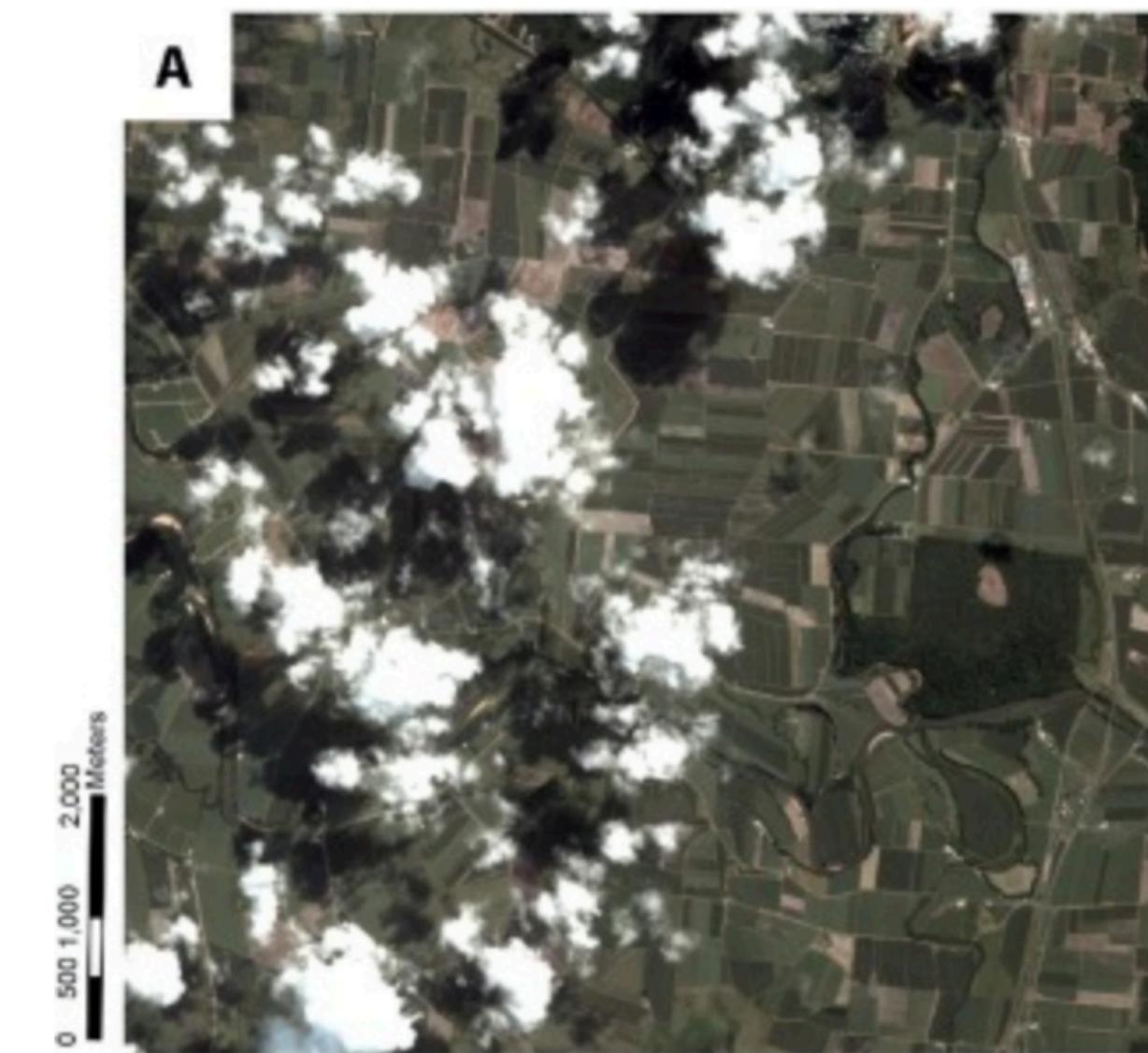
# Types of Machine Learning problems

---

**Classification** answers to:  
what (pre-defined) class does  
this item belong to?

**Supervised learning.**

**Multi-class** classification



# Types of Machine Learning problems

**Classification** answers to: what (pre-defined) class does this item belong to?

**Supervised learning.**

**Multi-label** classification: "contain habitation", "partly cloudy", "cloudy", "partly shaded", "shaded"

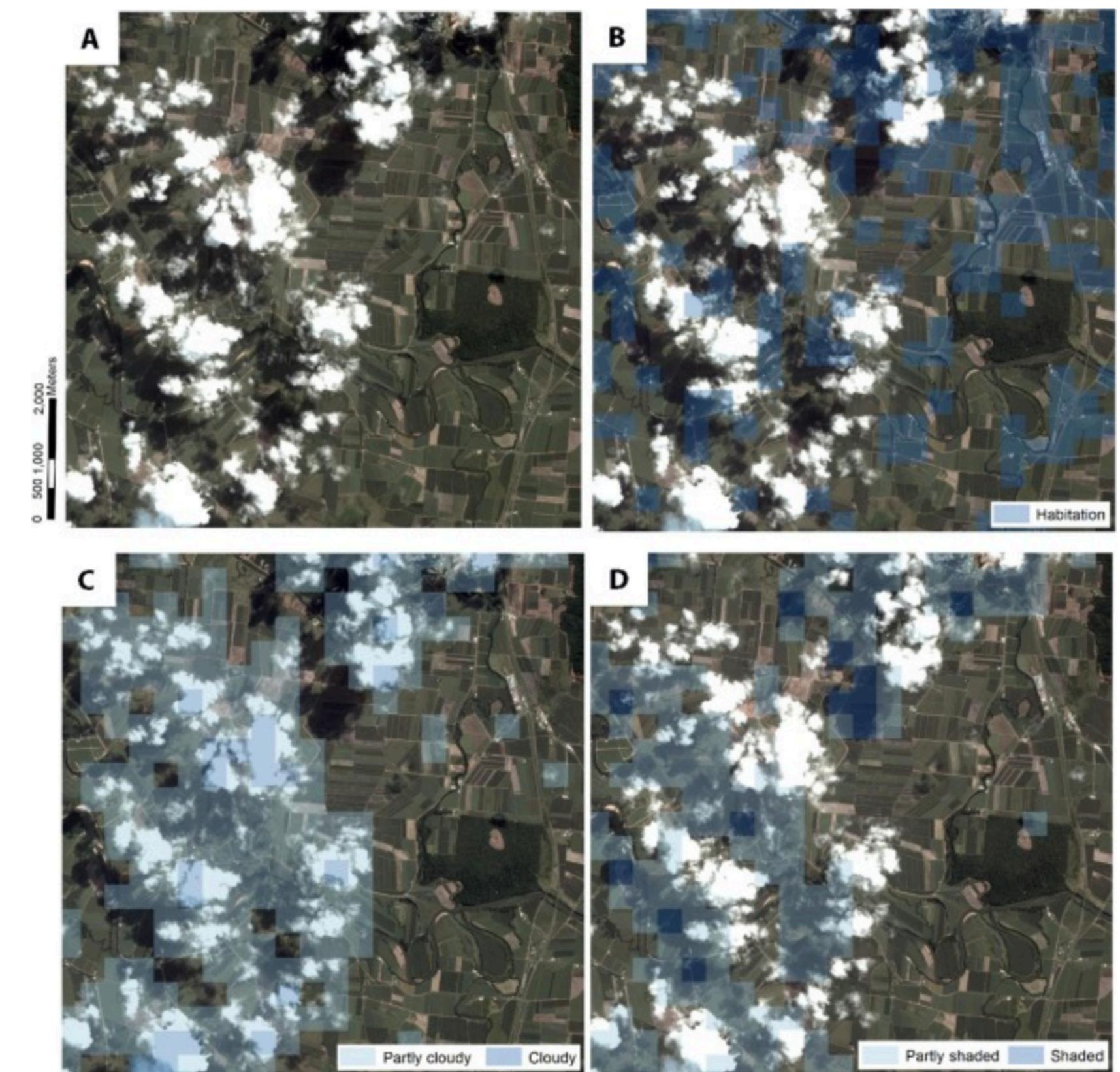


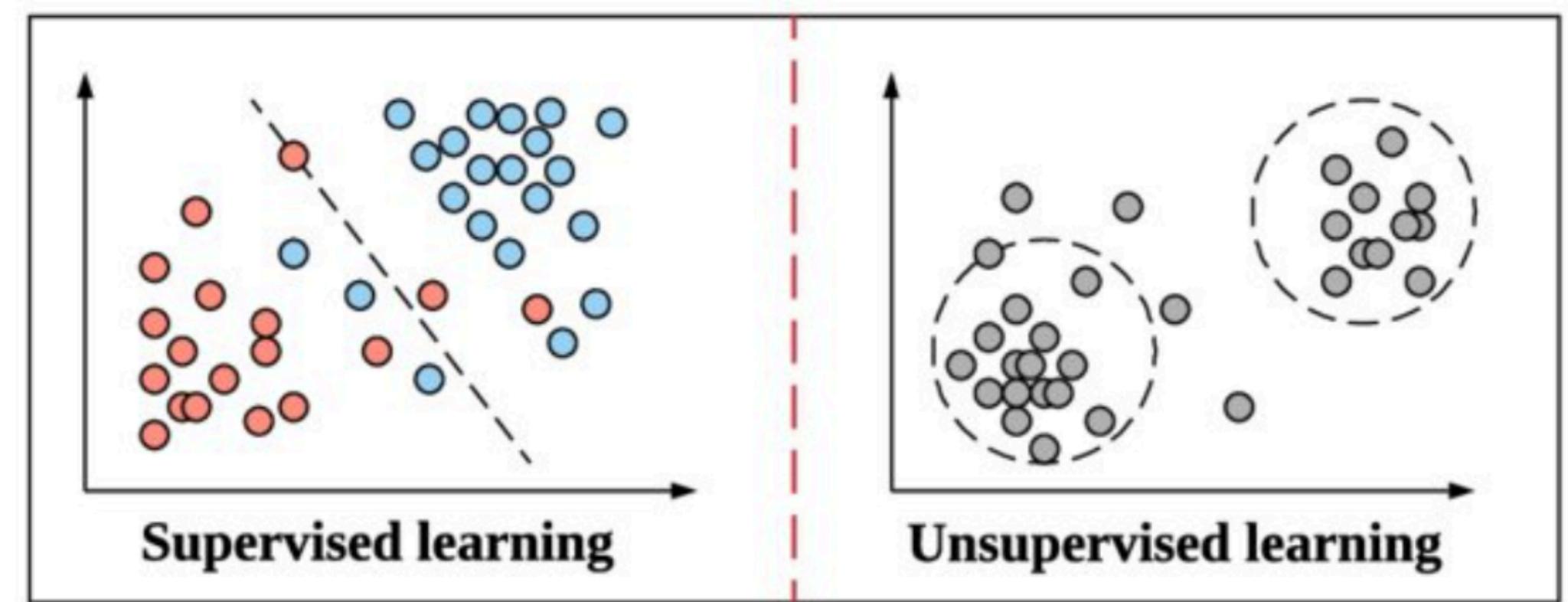
Fig. 4. An example of a multi-label scene [classification](#) using our ensemble CNN model of a PlanetScope image collected over the Wet Tropics of Australia on May 9, 2017: (A) original RGB image, (B) scenes identified to contain 'habitation' labels (as an example of a land cover label group), (C) scenes identified to contain 'partly cloudy' and 'cloudy' labels, (D) scenes identified to contain 'partly shaded' and 'shaded' labels. Each scene is  $128 \times 128$  pixels (i.e.  $400 \times 400$  m) in size.

# Types of Machine Learning problems

---

**Clustering** answers to: how can I group these items? No pre-defined classes.

**Unsupervised learning.**



Qian et al, 2019

# Types of Machine Learning problems

**Regression** refers to the case where the task is to provide a real-valued quantity.

**Supervised learning.**

## Article

# Skilful precipitation nowcasting using deep generative models of radar

<https://doi.org/10.1038/s41586-021-03854-z>

Received: 17 February 2021

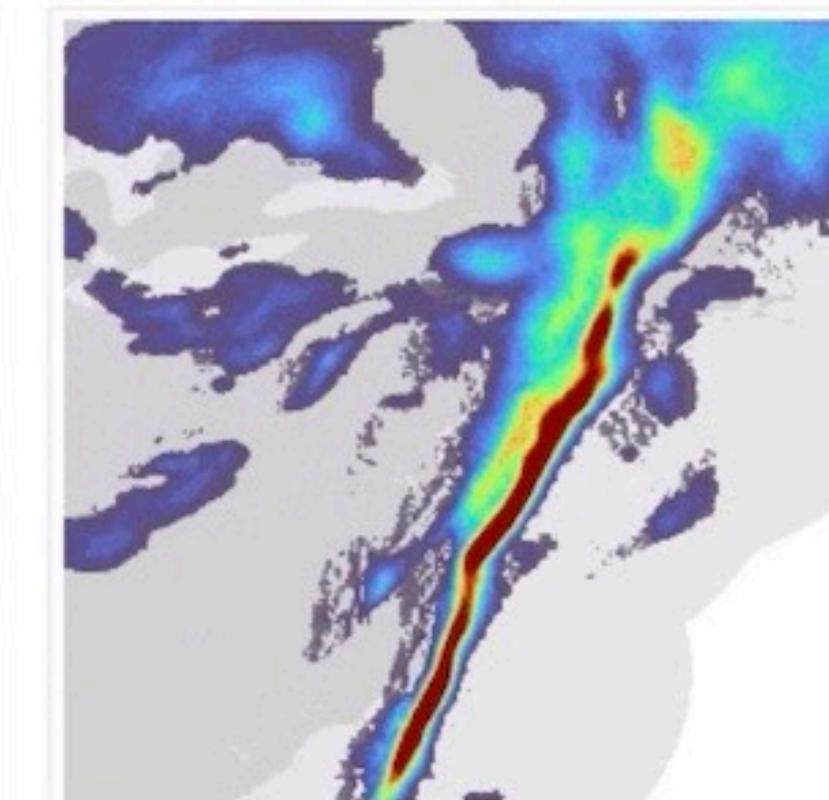
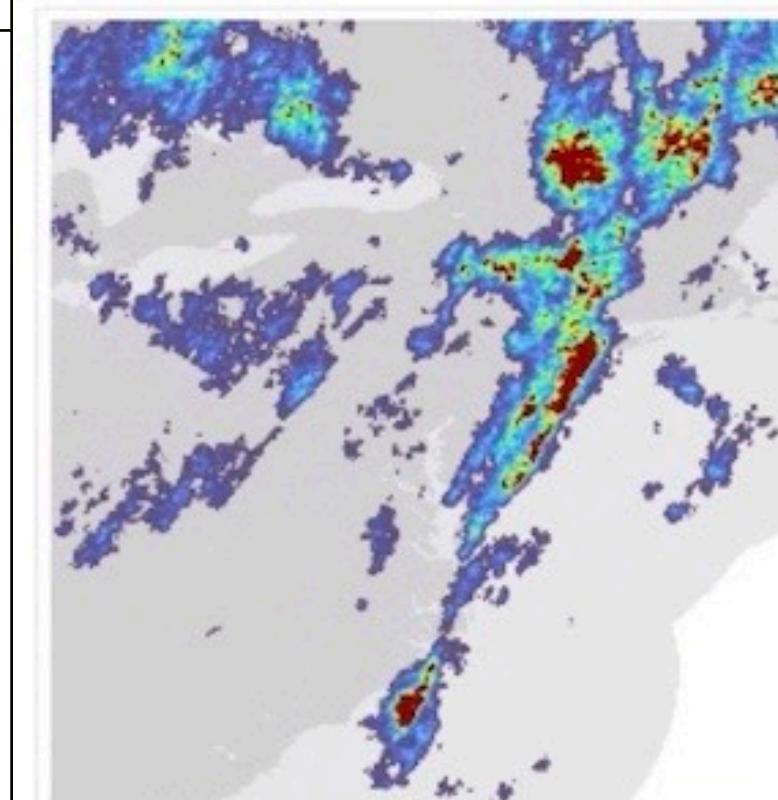
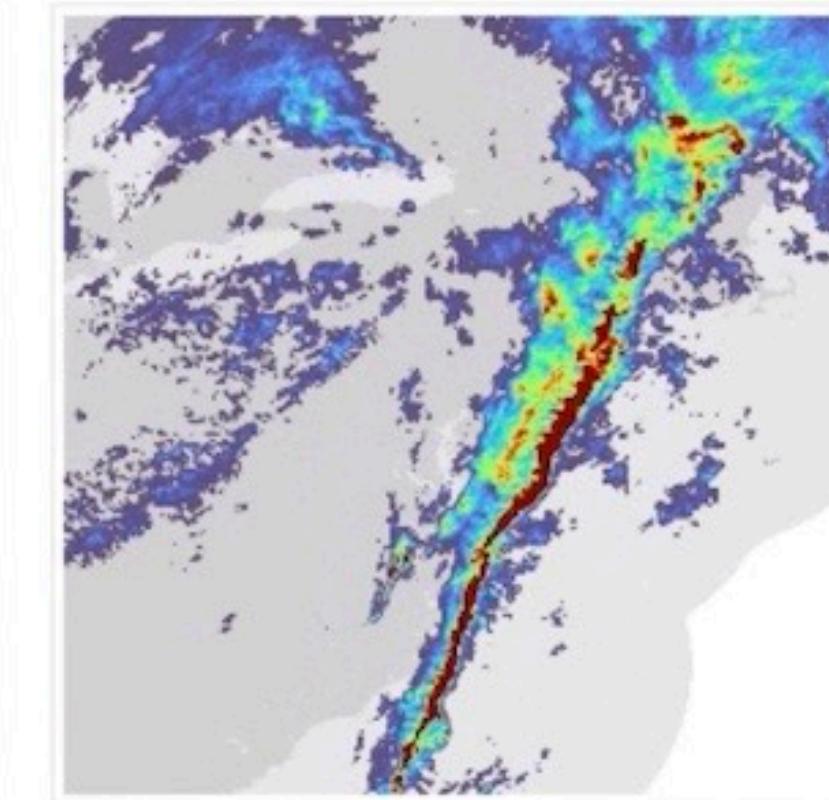
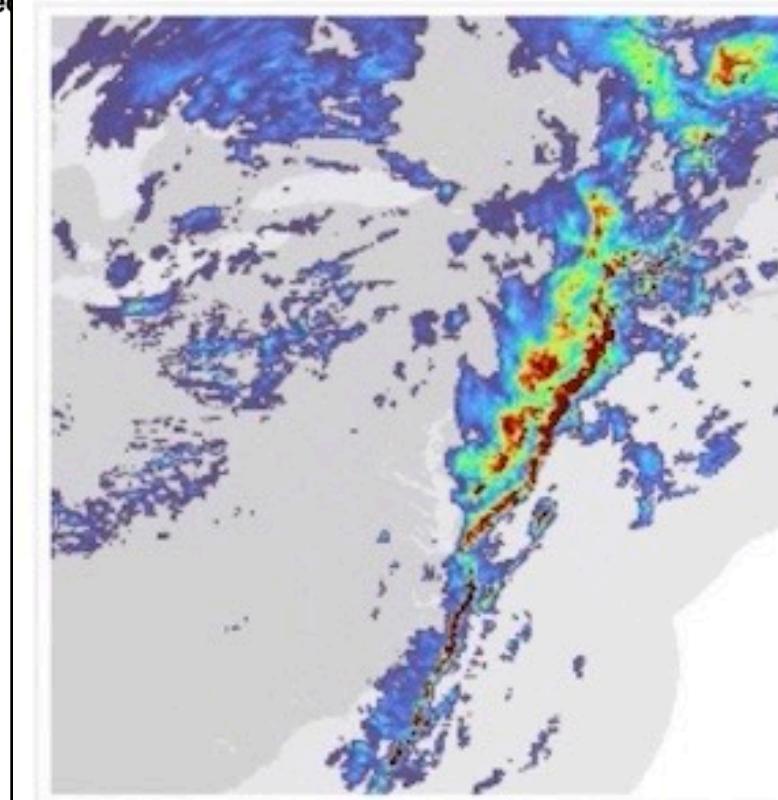
Accepted: 27 July 2021

Published online: 29 September 2021

Open access



Precipitation nowcasting, the high-resolution forecasting of precipitation up to two hours ahead, supports the real world socio-economic needs of many sectors relevant on



Suman Ravuri<sup>1,5</sup>, Karel Lenc<sup>1,5</sup>, Matthew Willson<sup>1,5</sup>, Dmitry Kangin<sup>2,3</sup>, Remi Lam<sup>1</sup>, Piotr Mirowski<sup>1</sup>, Megan Fitzsimons<sup>2</sup>, Maria Athanassiadou<sup>2</sup>, Sheleem Kashem<sup>1</sup>, Sam Madge<sup>2</sup>, Rachel Prudden<sup>2,3</sup>, Amol Mandhane<sup>1</sup>, Aidan Clark<sup>1</sup>, Andrew Brock<sup>1</sup>, Karen Simonyan<sup>1</sup>, Raia Hadsell<sup>1</sup>, Niall Robinson<sup>2,3</sup>, Ellen Clancy<sup>1</sup>, Alberto Arribas<sup>2,4</sup> & Shakir Mohamed<sup>1,✉</sup>

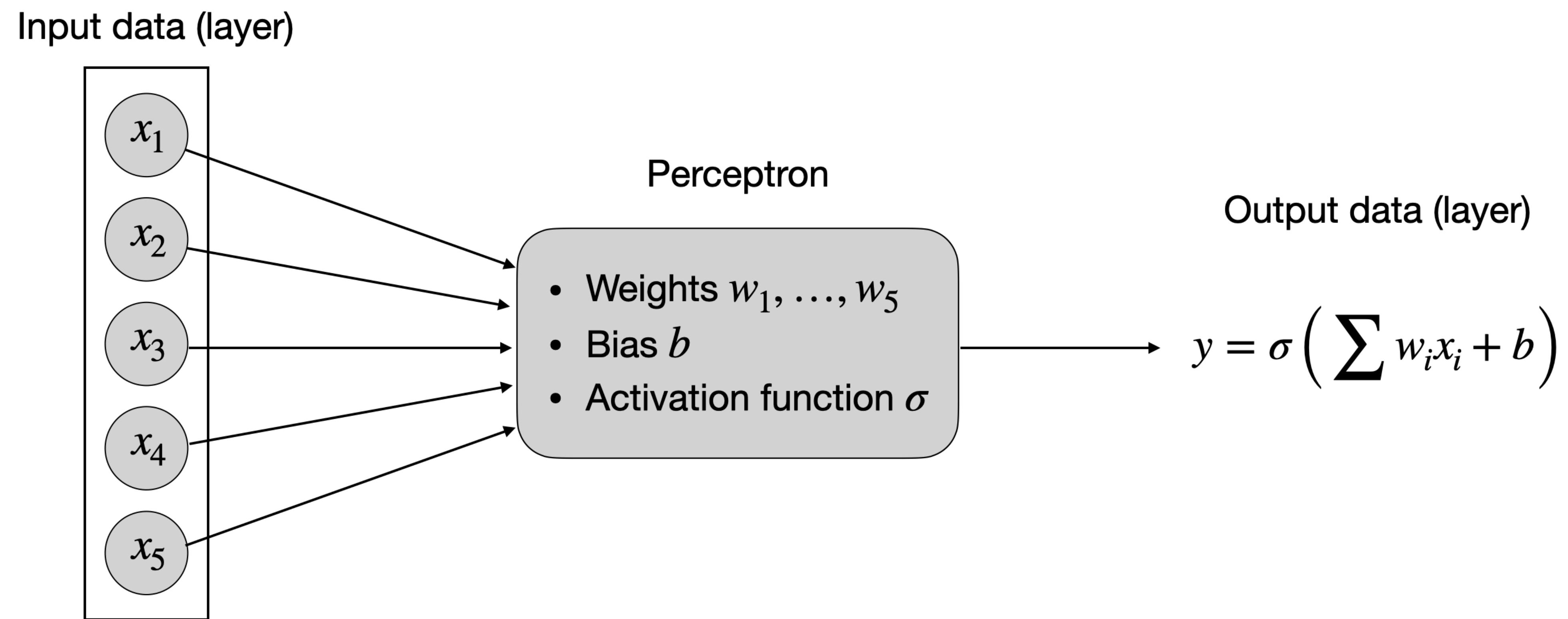
# The Multi-Layer Perceptron

---

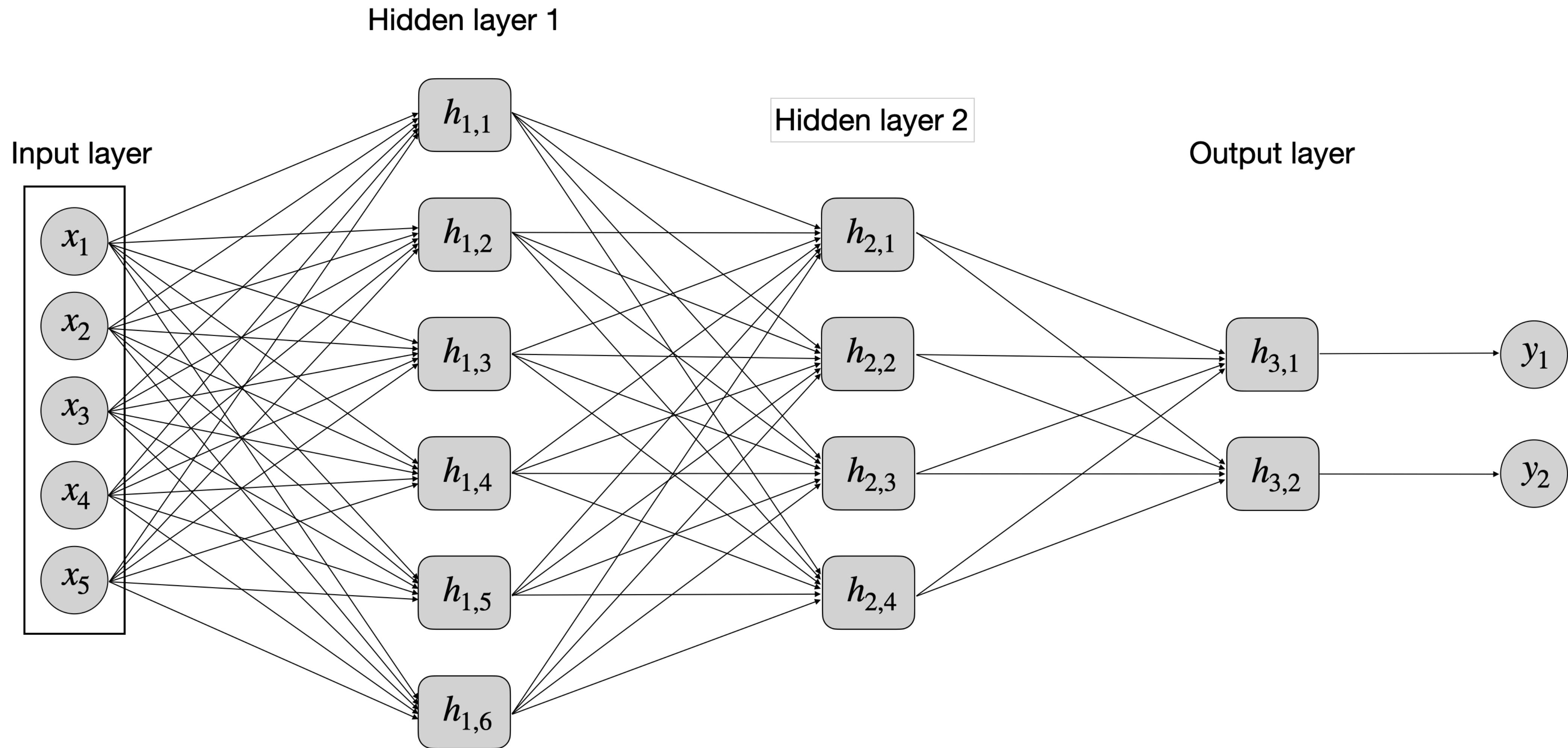
Perceptron, activation functions, feedforward networks, layers...

# The artificial neuron (perceptron)

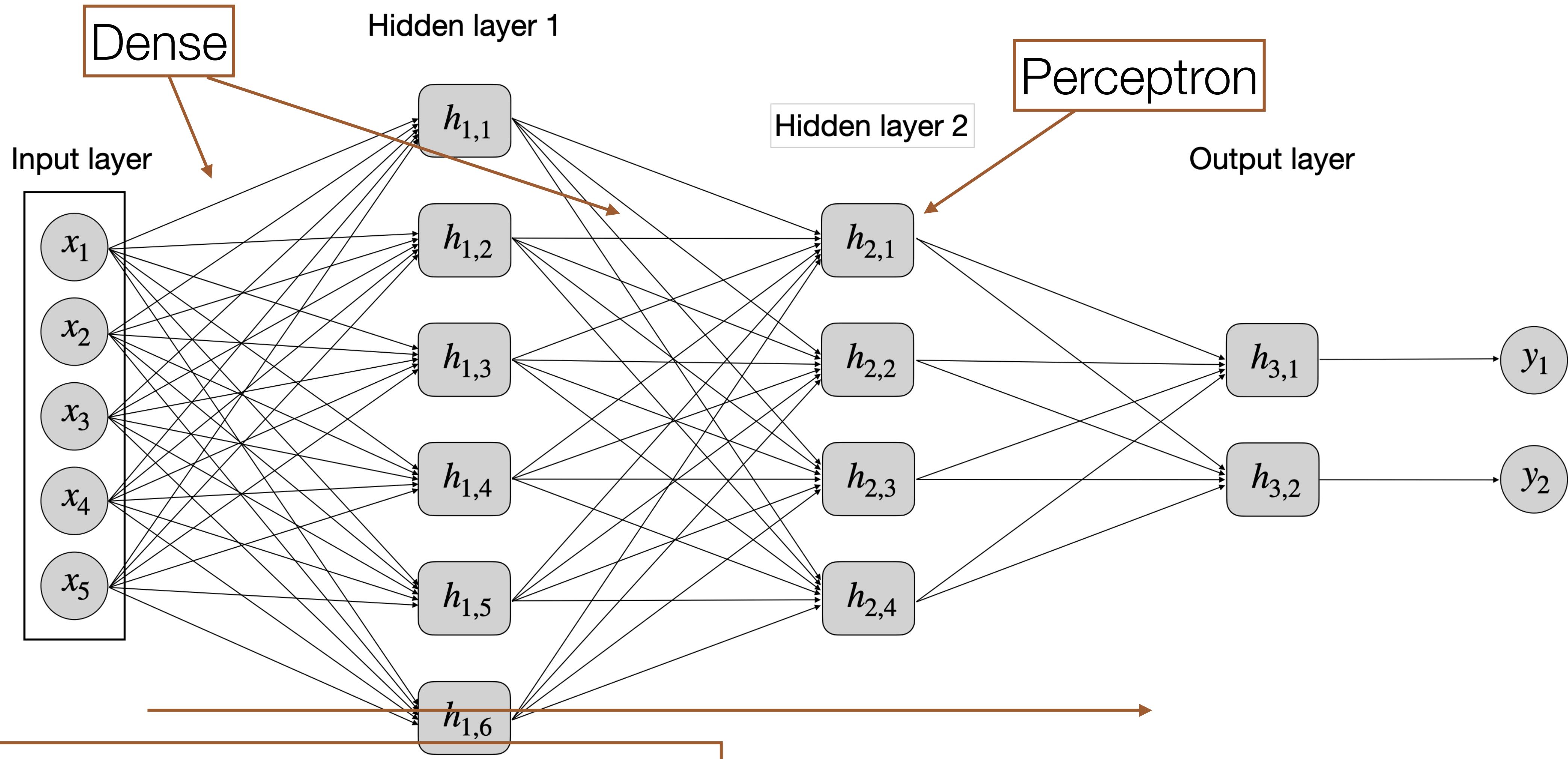
---



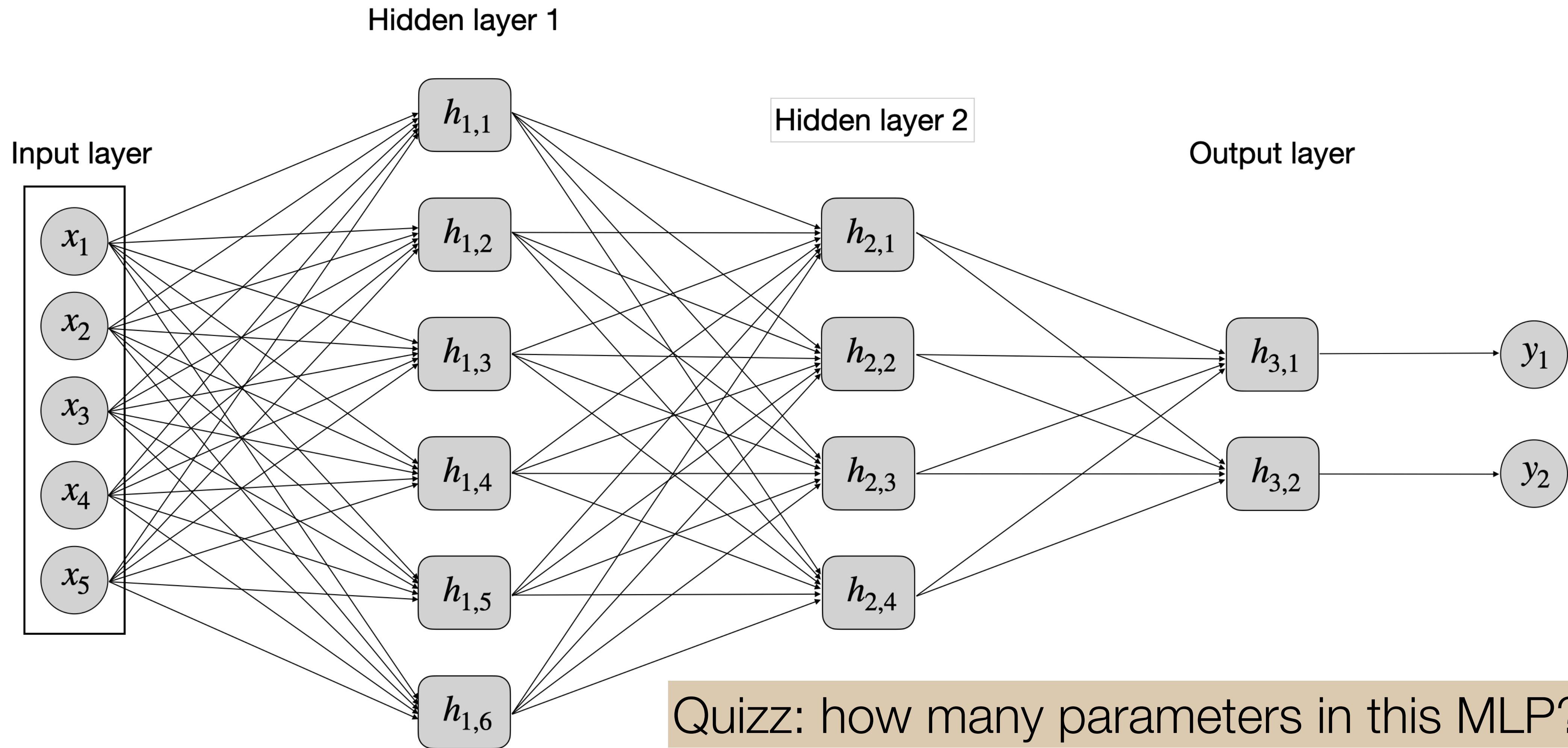
# Multi Layer Perceptron (MLP) or Dense Neural Network (DNN)



# Multi Layer Perceptron (MLP) or Dense Neural Network (DNN)

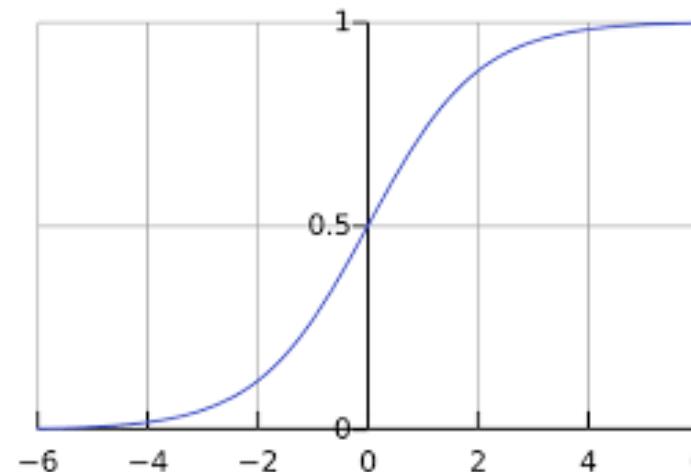


# Multi Layer Perceptron (MLP) or Dense Neural Network (DNN)



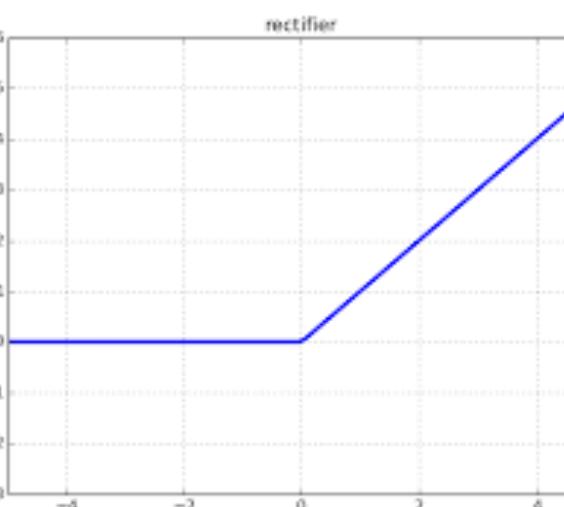
# Activation functions

**softmax**



$$\sigma : \mathbb{R}^K \rightarrow (0, 1)^K$$
$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad \text{for } j = 1, \dots, K.$$

**ReLU**



$$f(x) = x^+ = \max(0, x)$$

**selu**



$$\text{selu}(x) = \lambda \begin{cases} x & \text{if } x > 0 \\ \alpha e^x - \alpha & \text{if } x \leq 0 \end{cases}$$

**etc.**

# Let's play with MLPs

---

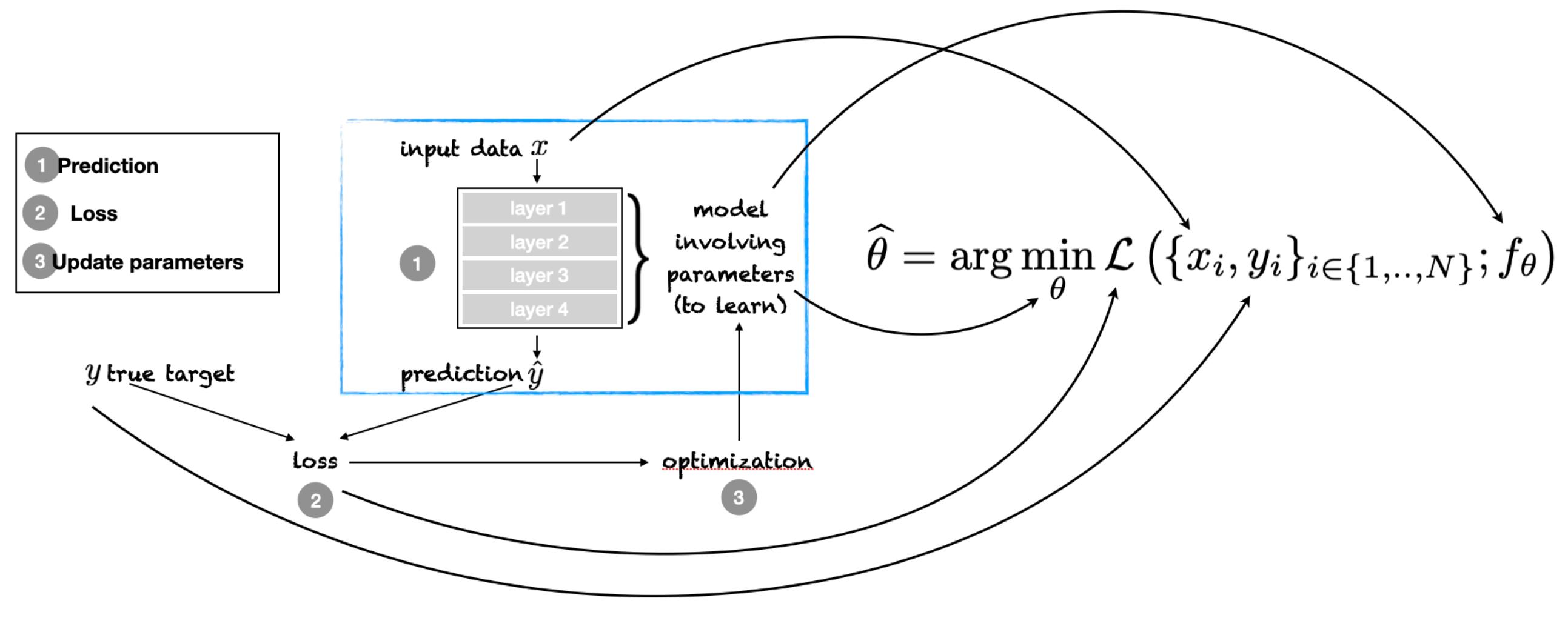
- <https://playground.tensorflow.org>

# The learning process

---

Loss function, Gradient descent, backpropagation

# Loss functions



$$J(\theta) = \frac{1}{m} \sum_{i=1}^m |y^{(i)} - f(x^{(i)}; \theta)|$$

$$J(\theta) = \frac{1}{m} \sum_{i=1}^m (y^{(i)} - f(x^{(i)}; \theta))^2$$

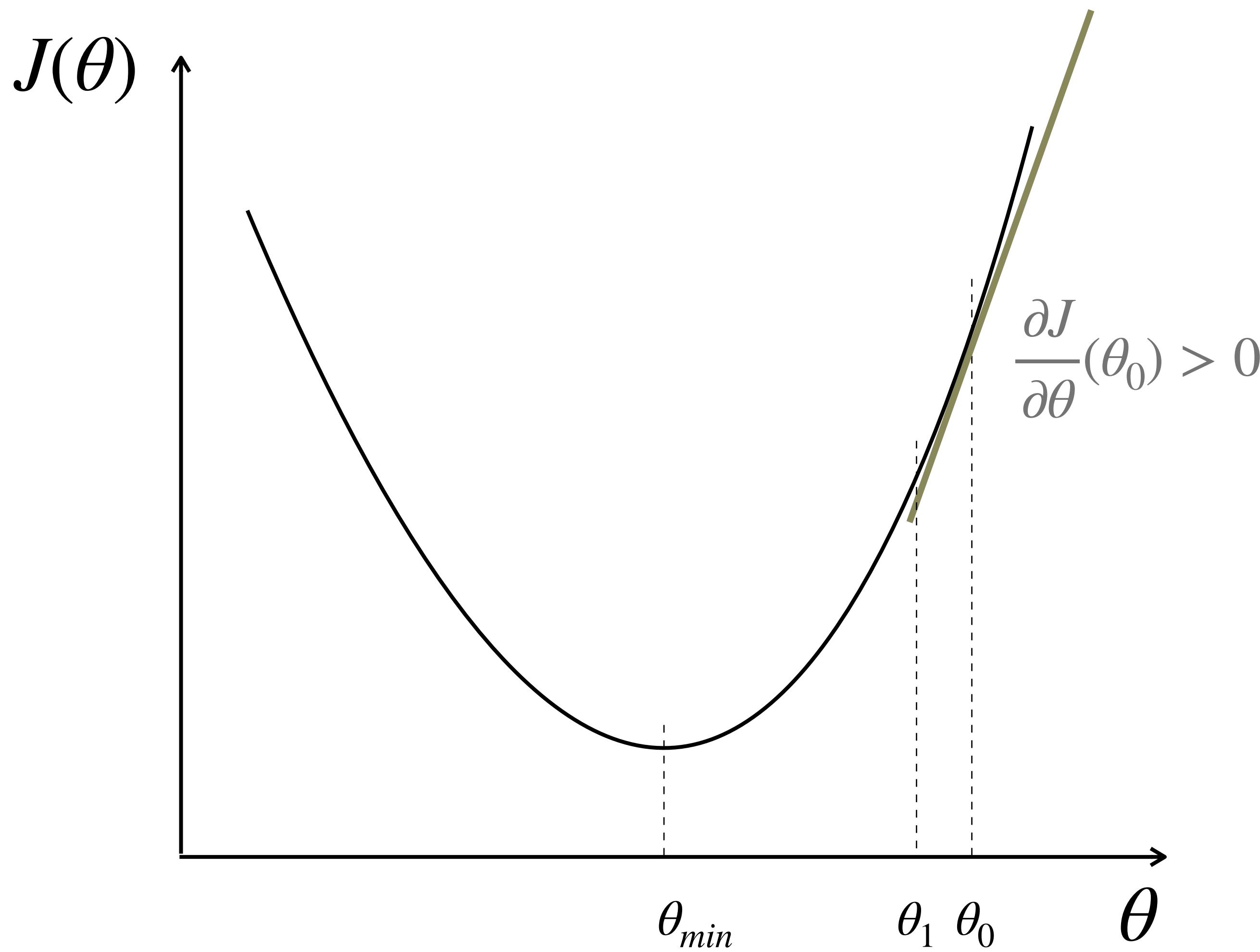
$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m y^{(i)} \log f(x^{(i)}; \theta)$$

etc.

$(\mathcal{L}$  has become  $J$ )

# Gradient descent

---



Gradient descent:

$$\theta_{n+1} = \theta_n - \lambda \nabla J(\theta_N)$$

Learning rate

# Calculating the gradient: backpropagation

---

- on the black board

# Introduction to pytorch

---

Basic tips and practice

# Basic tips for pytorch

---

- High-level python library (platform-independent, CPU/GPU, distributed computing,...)
- Very similar to numpy syntax but different (eg, `torch.tensor` vs. `numpy.array`)
- Specific types and classes to implement forward and backward passes through a network
- Object-based coding through classes for neural models
- Among the two main state-of-the-art DL frameworks with Tensorflow (but there are others, JuliaDiff, Jax, Caffe,...)

# Practice

---

- [https://github.com/CIA-Oceanix/DLOA2023/blob/main/lectures/notebooks/intro\\_pytorch\\_learning.ipynb](https://github.com/CIA-Oceanix/DLOA2023/blob/main/lectures/notebooks/intro_pytorch_learning.ipynb)