Lecture 1

08.09.2025

Plan for today

- Administrative
  - Staff, grading, communication
- Introduction to AI
- Introduction to ML
  - Supervised versus unsupervised learning
  - Linear regression

Administrative

Welcome to ME-390

Administrative

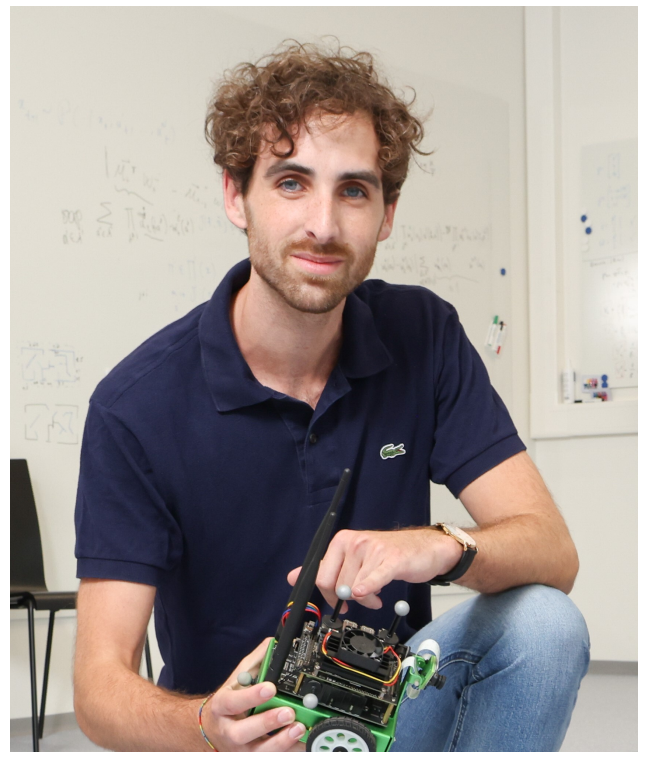
Teacher Staff

Teacher

Maryam Kamgarpour

TAs

Kai Ren



Giulio Salizzoni

Saurabh Vaishampayan

Anna Maddux

Tingting Ni

Philip Jordan

Gabriel Vallat

Assessment and grading

- Assessment
  - 2 in-person quizzes: 10% each (Oct 15, Nov 26, during exercise hour)
  - 1 Python assignment: 10% (due Dec 17)
  - Final exam: at least 70% (date set by EPFL)
- Grading policy: your grade will be the best of several possible combinations:
  - Final exam 100%
  - Final exam 80% + best 2 other components (quizzes/assignment)
  - Final exam 70% + all 3 other components

- Implies:
  - Semester work is optional, it only counts if it helps your grade
  - No special requests on quiz/assignment dates or grading will be considered

Course material and ommunication

- We use Moodle for lecture notes and videos, programming exercises, problem sets, past quizzes and exams
- We use EdDiscussions for Q&A: please write your questions on Ed Discussion or bring them to class/exercise hour

- Question: what do I do if I have a question about course material?
  - talk to me during class break/after lecture
  - ask your TAs during exercise hour
  - post your question on Ed Discussion

Classroom expectation

- Let’s make our time in class useful for everyone. Please:
- Be present and attentive during lecture
- If you need to work on something else, feel free to step outside
- Save conversations for the breaks so everyone can focus

- In all our interactions, with each other and with instructors, we aim for respect and professionalism

Technical background requirements

Domains covered by ML

- Linear algebra
- Calculus
- Optimization
- Probability & statistics
- Programming (in Python)

Additional resources

The notes are based on a few sources:

- Machine Learning for Engineers, Using Data to Solve Problems for Physical Systems by Ryan G. McClarren
- EE 104, Stanford

Online courses and resources

- Posted on Moodle
- background: Introduction to Applied Linear Algebra – Vectors, Matrices, and Least Squares

Introduction to AI

An example in robotics - Imitation learning

Learn how to drive a car based on past drivers' driving data

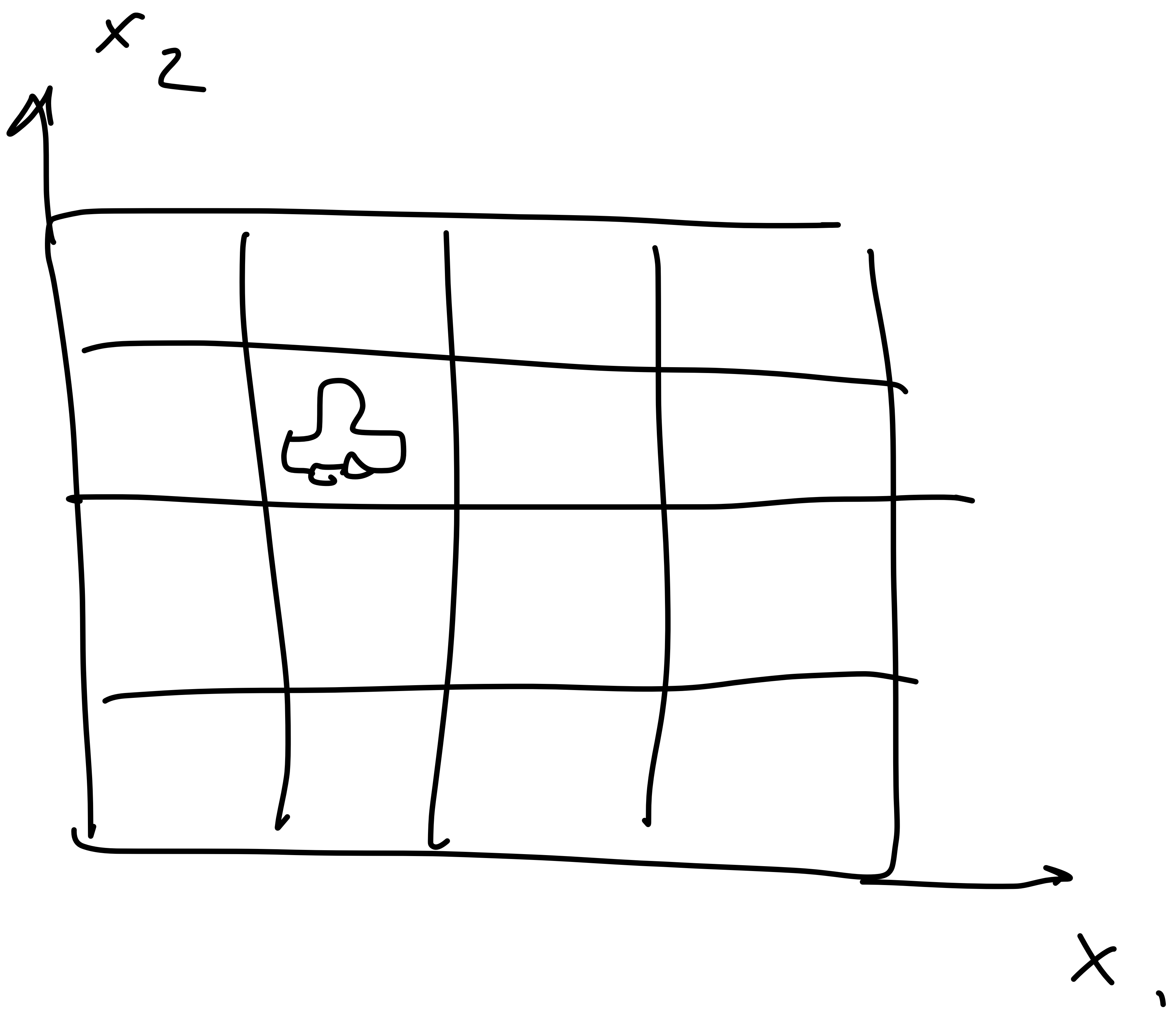
From expert demos to learning

$x^i \in \mathbb{R}^2$

car position

$y^i \in \{ \text{up} \uparrow, \text{down} \downarrow, \rightarrow \text{right}, \leftarrow \text{left} \}$

actions



History of AI

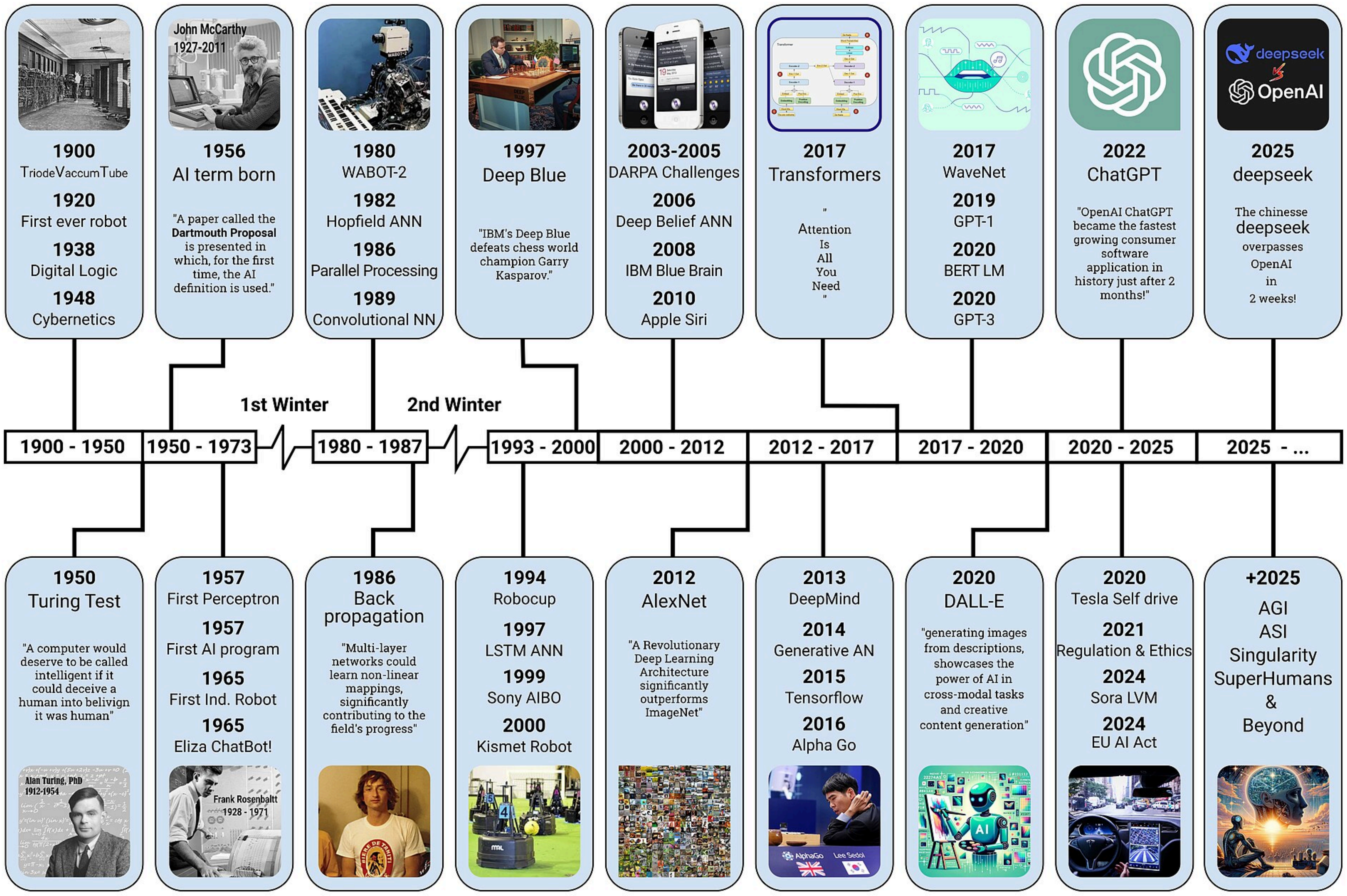
What is Artificial Intelligence?

AI is intelligence demonstrated by machines…

But how do we define intelligence?

Activity: Think, Pair, Share

AI timeline from 1900-2025 (wikipedia)



Discussion on intelligence and AI

Intelligence is multi-faceted: technical, social, emotional, ethical..

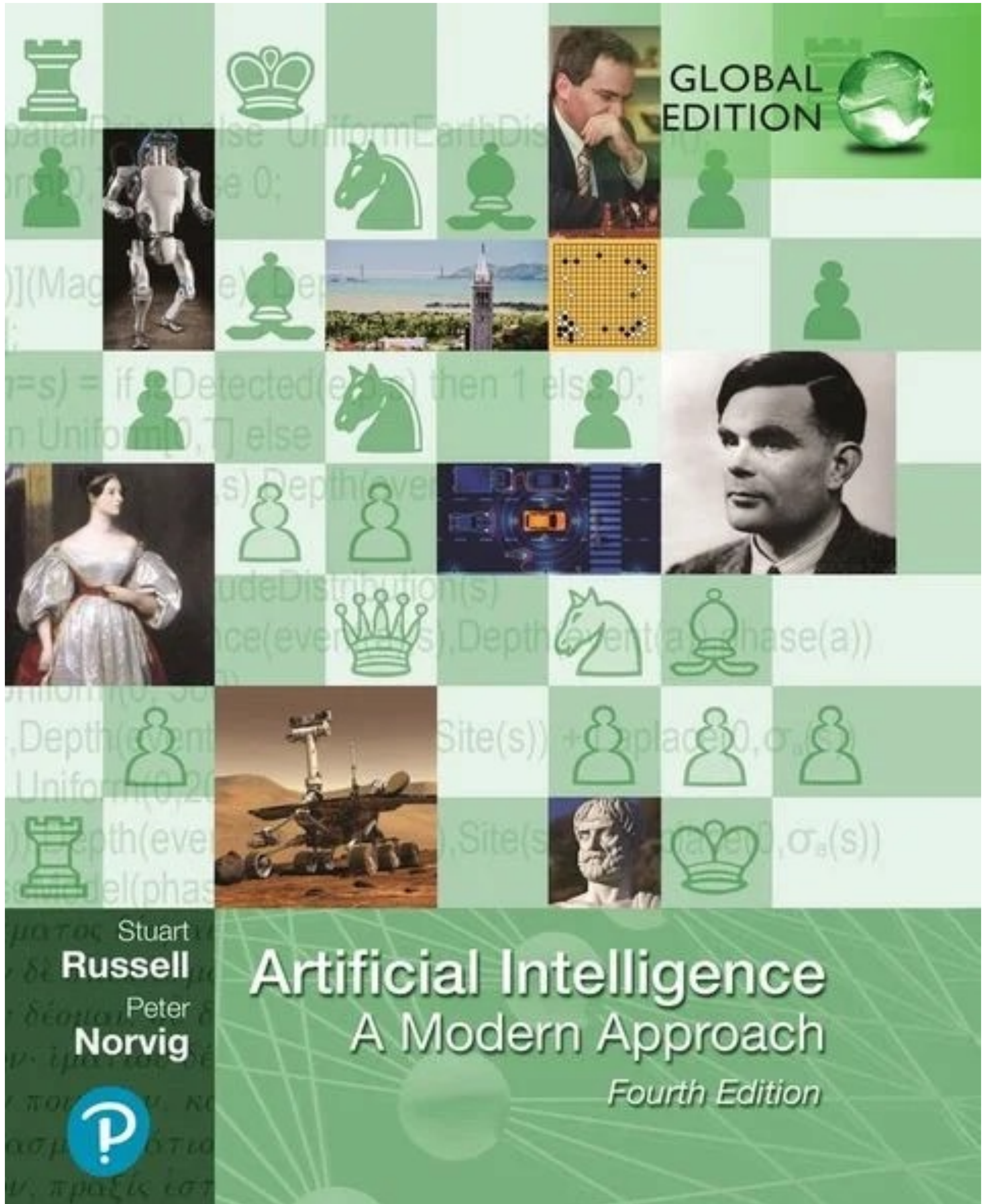
Defining intelligence reflects what we value as a society: Let’s think: what world do we want to live in? And how will that guide us in the kind of intelligence we want to have both for humans and machines..

Current AI is strong on technical aspects of intelligence

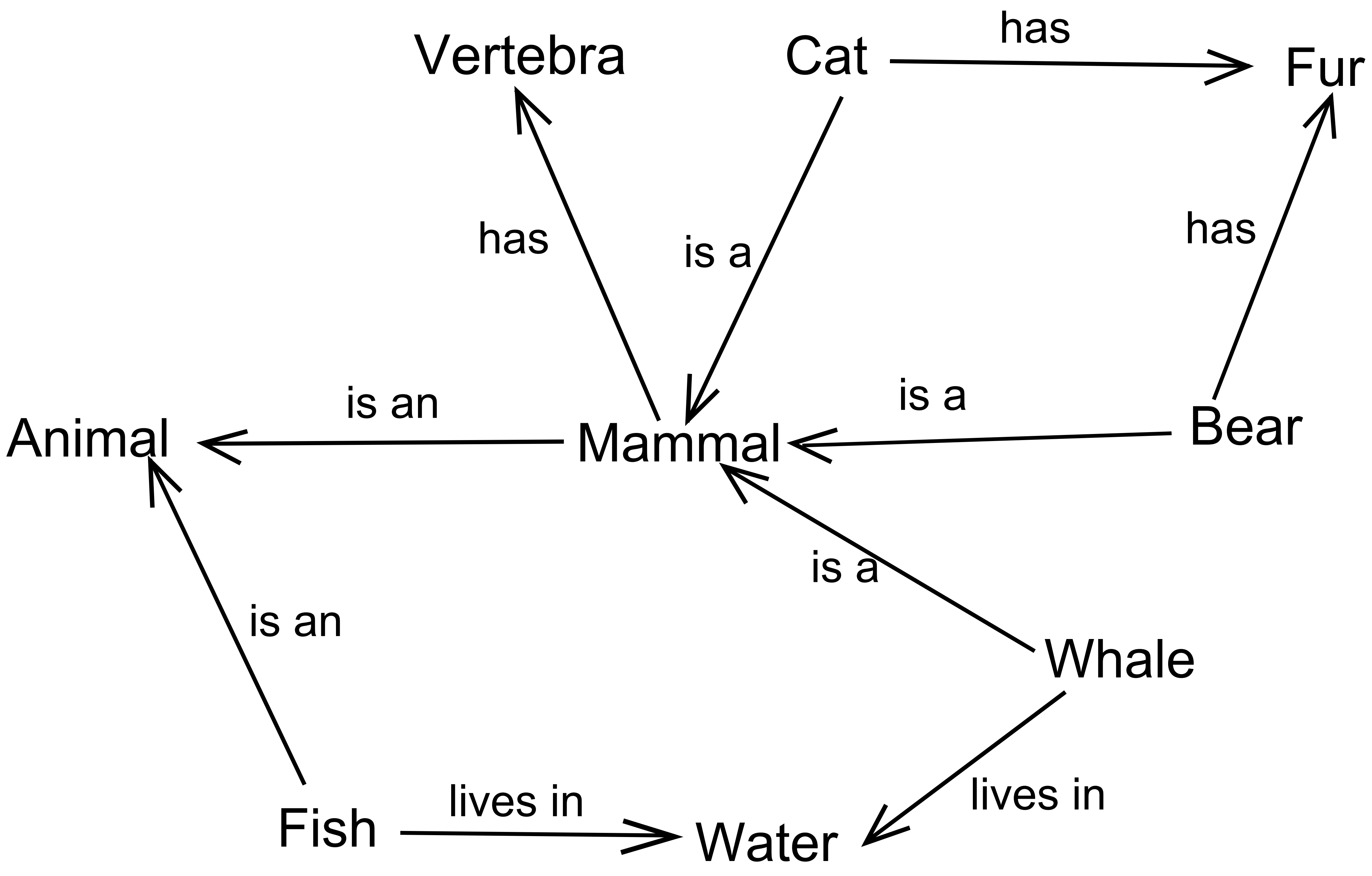
What impacts do our algorithms have locally and globally?

Whose agenda are we fulfilling when we build an intelligent system?

AI Textbook - a classic

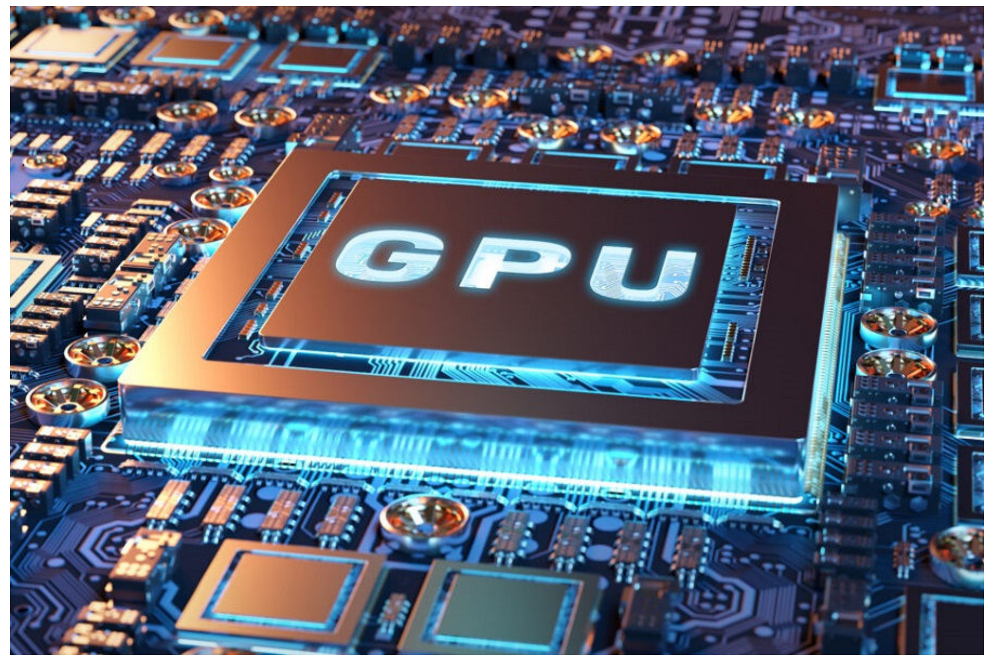


AI: problem-solving, reasoning, knowledge, planning, logic, search, …



AI now…

Pillars of AI theory are not new.. but there has been some advances that help use this theory..

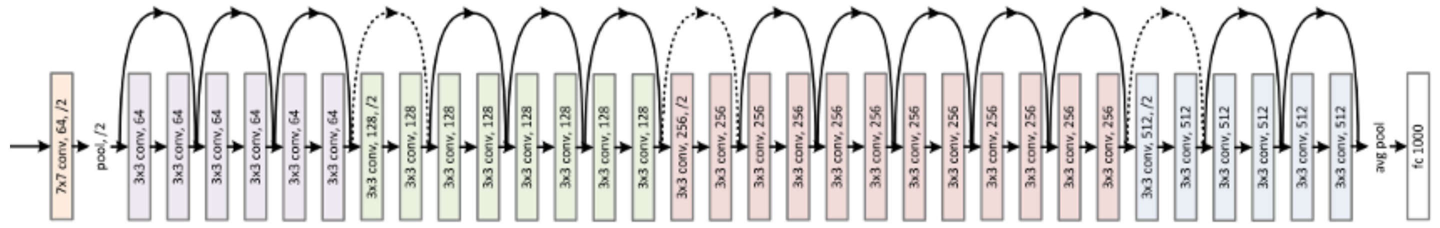


Hardware

Progress in hardware such as Graphical Processing Unit (GPU)

Data

Massive amount of labeled data



Open Science/ Software

Shared models/frameworks

Machine Learning is the tool that leverages all these.

Our focus in the course is on the ML theory

Machine learning approach to AI

- Instead of listing out the rules, let machines automatically learn how input **data** is correlated with a given task/objective/outcome/output.

- Let machines learns from experience/examples/data/feedback

- In this course, we focus on the machine learning (ML) approach as the paradigm that dominates today's applications, but remember: it's not the whole story of AI

This course

- Focused on AI in the technical aspect
- Reason:
  - learn the basis of current tools to understand their background and potential

- But: I like you to think about the broader aspect

What impacts do our algorithms have locally and globally?

Whose agenda are we fullfilling when we build an intelligent system?

AI versus ML - Deep Blue

- Deep Blue system (IBM,1997) defeated the world chess champion
  - Programmed to know the chess rules, used brute force search

- AlphaGo (DeepMind, 2015) defeated Go champion
  - Taught itself using human and computer game playing data

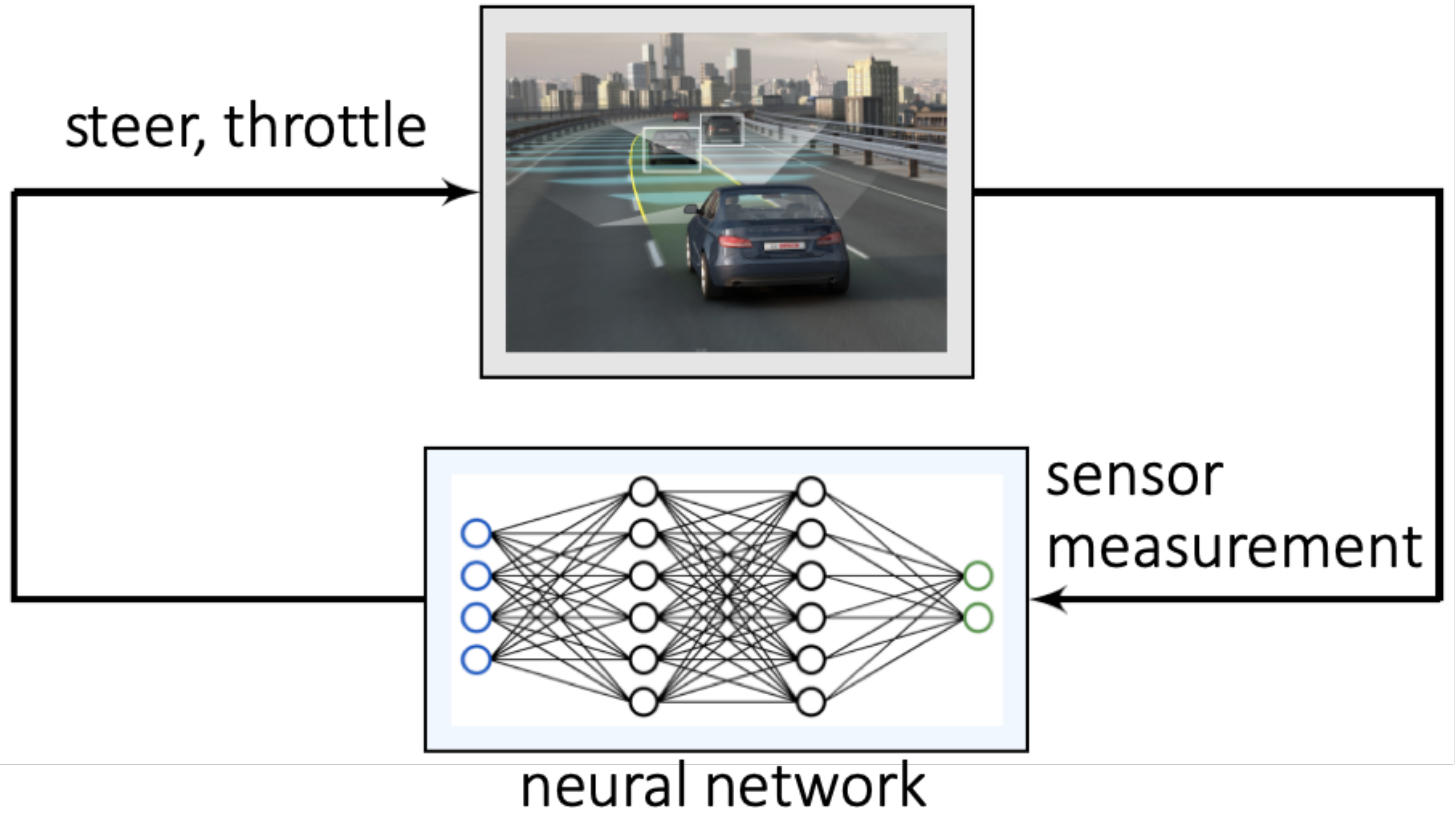
Examples of AI achieved with ML

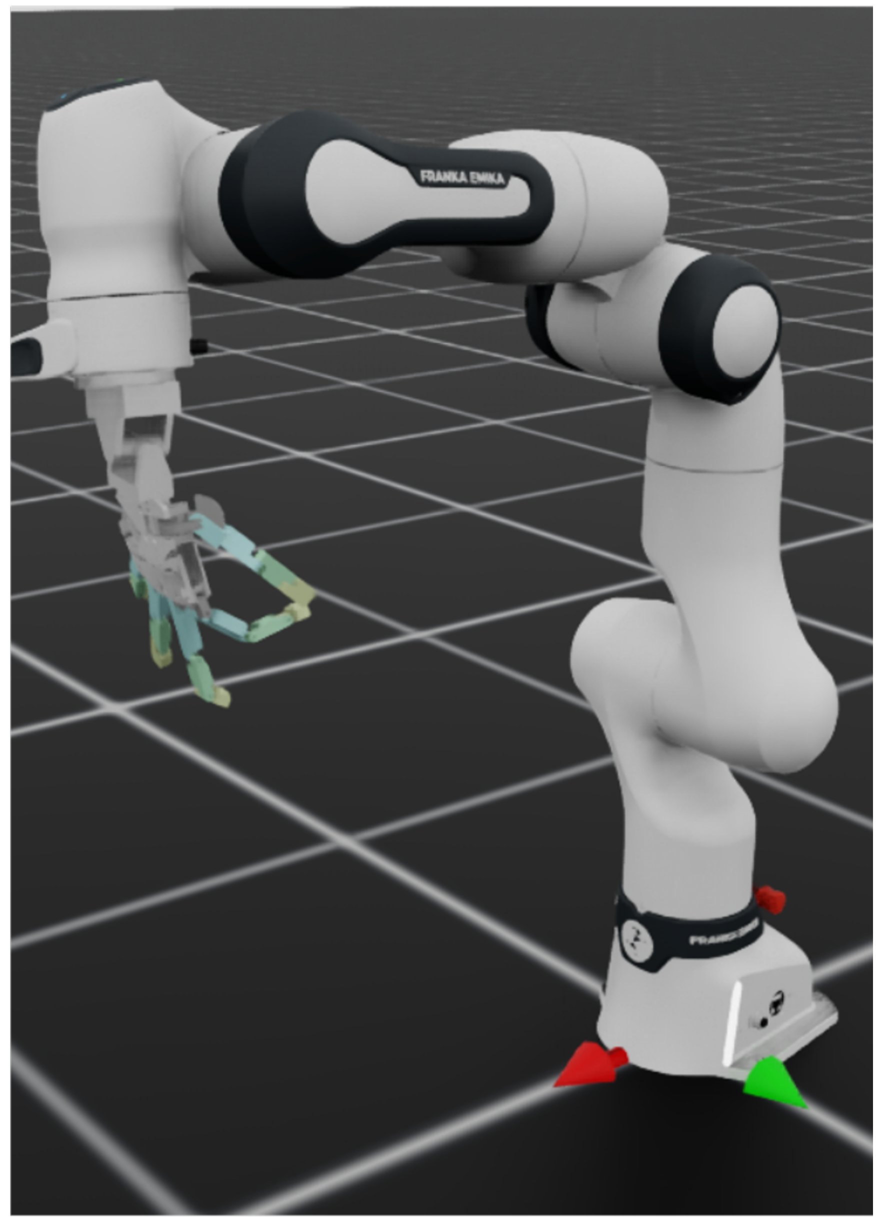
Video credit: Tesla
https://www.tesla.com/autopilot

The following people don't exist… They are generated by an AI

Why this course matters to you

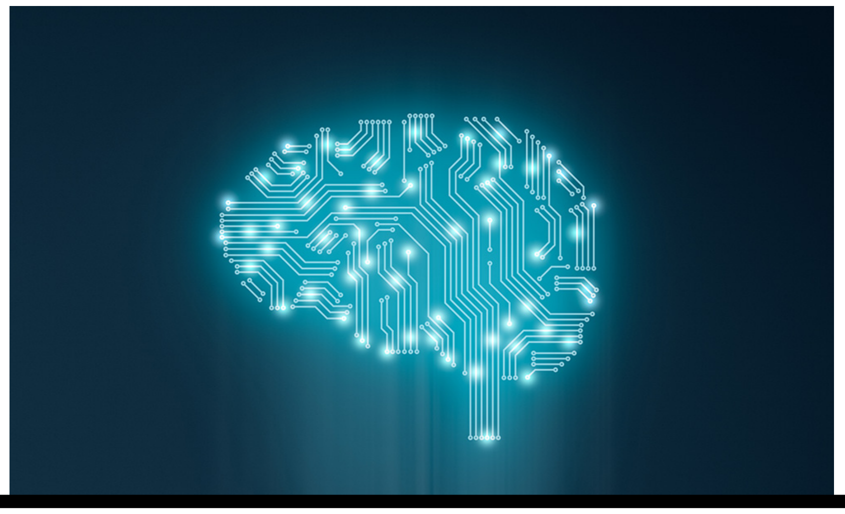
- You already use AI daily
- ML has entered mechanical engineering: design, maintenance, energy, control, solving PDEs, ….
- Industry is looking for engineers with fundamental ML understanding
- Knowledge = power -> Less mystery, better decisions



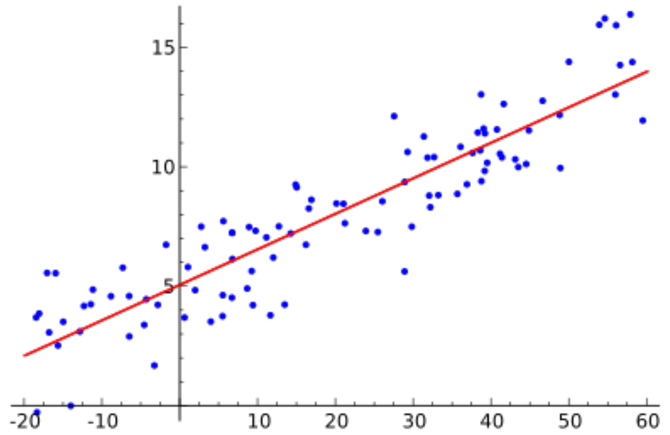


Course topics

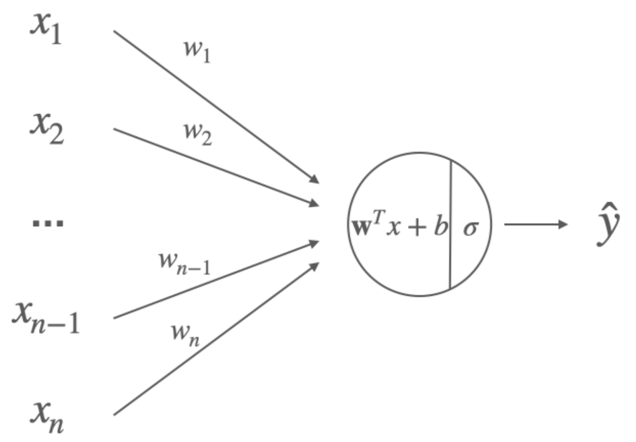
Introduction



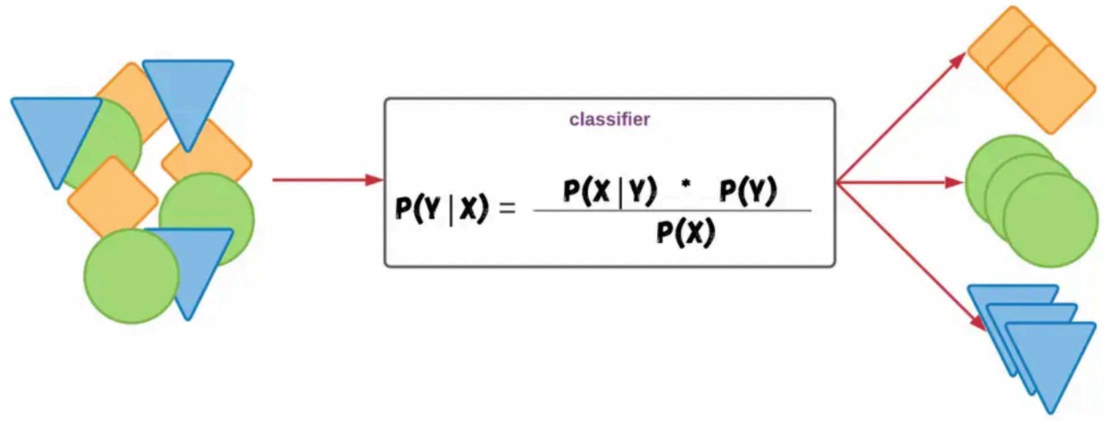
Linear regression



Logistic regression



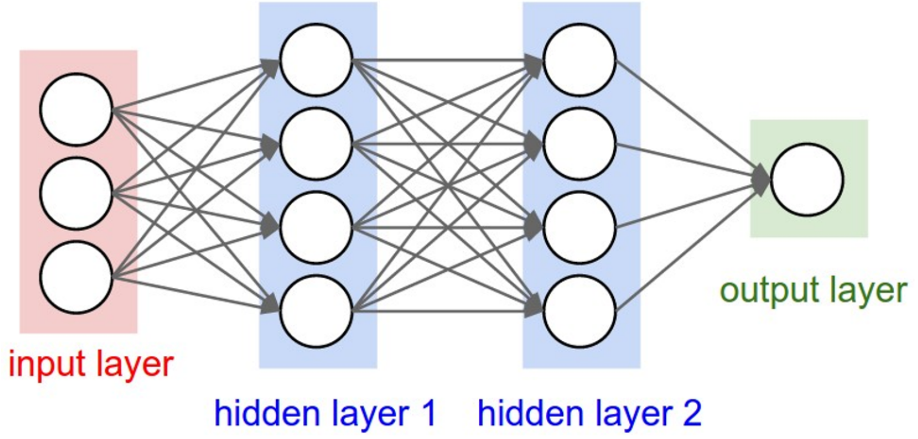
Naive Bayes



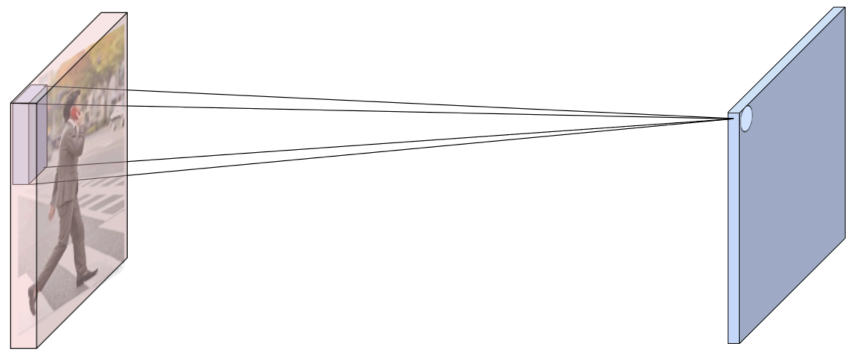
KNN

AI & sustainability

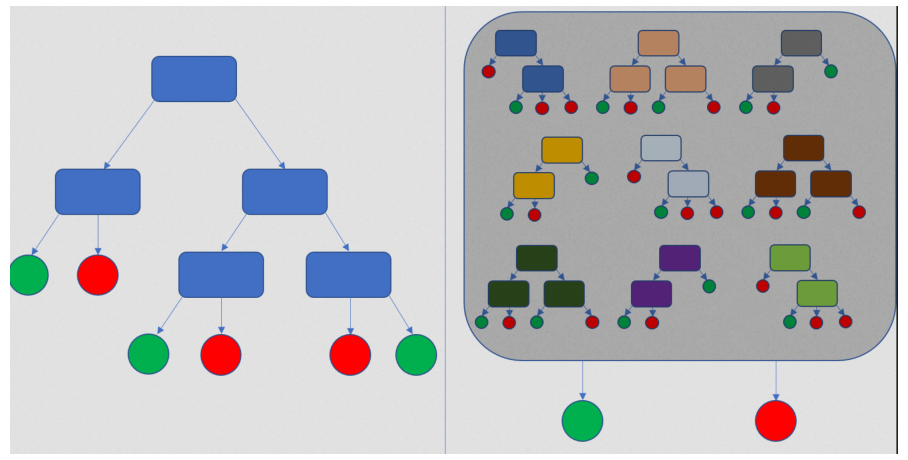
Neural networks



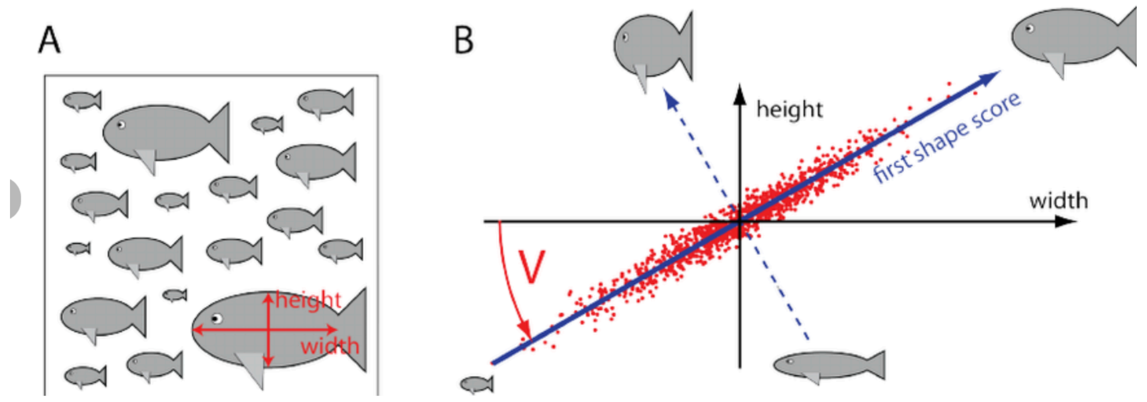
Convolutional neural networks



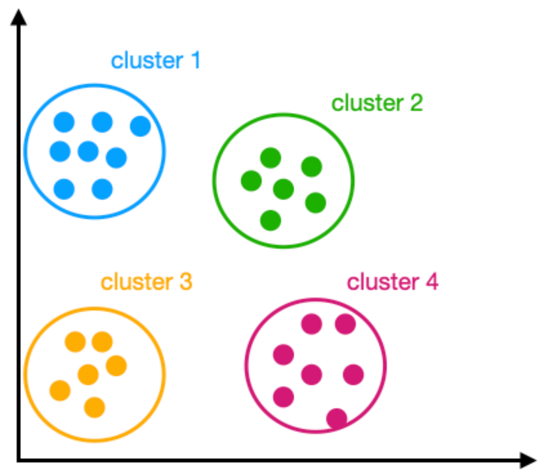
Decision-trees



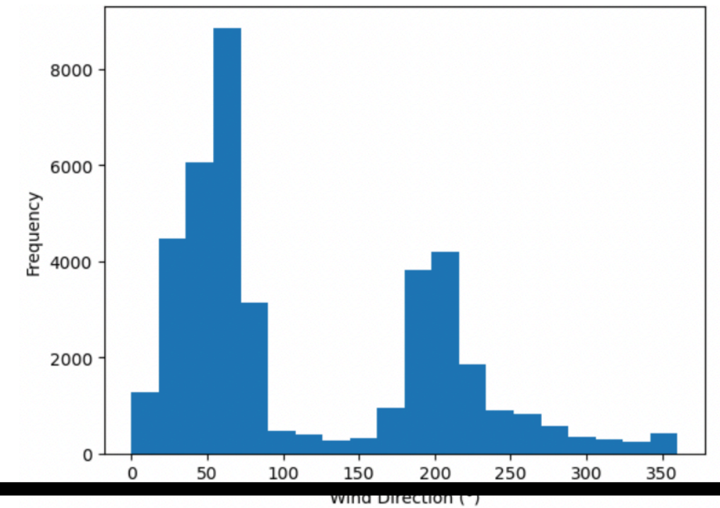
Dimensionality reduction



Clustering



Reinforcement learning



Fundamentals

- Linear algebra
  - Vectors, linear equations, matrices: eigenvalues, positive (semi) definite, invertibility
- Optimizaiton
  - Gradient descent, optimality conditions, regularization
- Statistics
  - Mean, variance, covariance, mode, probability distribution, conditional distribution, Bayes rule, empirical estimates
- Dynamical systems and control
  - Transition kernel, policy, reinforcement learning

Introduction to ML

Notation: sets, vectors, matrices

A <u>set</u> with k elements $S = \{i_1, i_2, \ldots, i_k\}$

also, written as $S = \{i_\ell\}_{\ell=1}^{k}$

Cartesian product of two sets,

$$S_a \times S_b = \{(s_a, s_b) \mid s_a \in S_a, s_b \in S_b\}$$

<u>Example</u>: $S_a = \{1, 2\}$, $S_b = \{2, 3\}$

$(1, 3) \in S_a \times S_b$, $(1, 1) \notin S_a \times S_b$

Notation: sets, vectors, matrices

vector $a \in \mathbb{R}^n$, $a = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}$, $a = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix}$

$a = (a_1, a_2, \cdots, a_n) \in \mathbb{R}^n$

all equivalent representation

matrix $A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$

$A \in \mathbb{R}^{m \times n}$

Note: $(1,2) \in \mathbb{R}^2$ vs. $(1,2) \subset \mathbb{R}$

Notation: data

$\{(x^i, y^i)\}_{i=1}^N$ labelled data, N data points

x^i : input features, y^i : output labels
(independent variables) (dependent variables)

<u>example</u> : driving dataset

$x^i \in \mathbb{R}^2$ Cartesian position of cars

$y^i \in \{\uparrow, \downarrow, \rightarrow, \leftarrow\}$: possible actions

EPFL

ML Ingredients

- Experience (data)
  - Example: wind speed, wind direction, power output

- Task (what we hope to do with this data)
  - Example: predict power output of a wind turbine given wind forecast

- Performance (how we measure the quality of the ML in doing the task):
  - usually through a “loss” function (also referred to as cost or objective function)
  - Example: error in prediction

Types of experience

- **Supervised** learning
  - Labels (outputs) are available for each input
  - Goal: learn a mapping from input to output

- **Unsupervised** learning
  - No labels
  - Goal: find pattern or structure in data

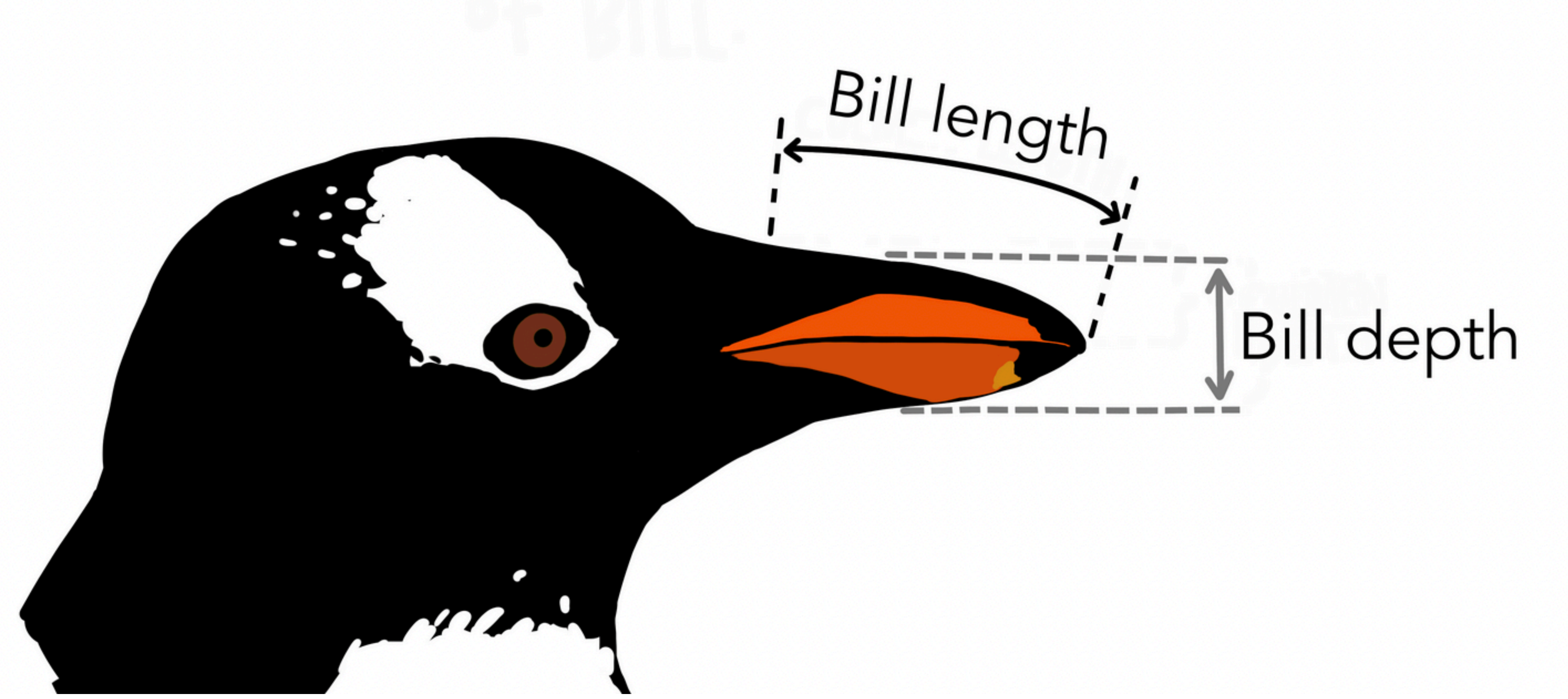
- **Reinforcement** learning
  - No labels, feedback comes as reward/punishment in dynamic environment
  - Goal: learn a strategy that maximises long-term reward

Supervised learning example

Palmer Penguins

	species	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g
0	Chinstrap	49.0	19.5	210.0	3950.0
1	Chinstrap	50.9	19.1	196.0	3550.0
2	Gentoo	42.7	13.7	208.0	3950.0
3	Chinstrap	43.5	18.1	202.0	3400.0
4	Chinstrap	49.8	17.3	198.0	3675.0

$i = 1$
$i = 2$
$\vdots$
$i = 5$



- The species of the penguin is used as an output label.

$x^i \in \mathbb{R}^4$

$y^i \in \{Adelie, Gentoo, Chinstrap\}$

Unsupervised learning example

Iris dataset

	sepal_length	sepal_width	petal_length	petal_width
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2

$x^i \in \mathbb{R}^4$

x^i_1 : sepal length, ..., x^i_4 : petal width

Supervised vs. Unsupervised

Palmer Penguins

	species	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g
0	Chinstrap	49.0	19.5	210.0	3950.0
1	Chinstrap	50.9	19.1	196.0	3550.0
2	Gentoo	42.7	13.7	208.0	3950.0
3	Chinstrap	43.5	18.1	202.0	3400.0
4	Chinstrap	49.8	17.3	198.0	3675.0

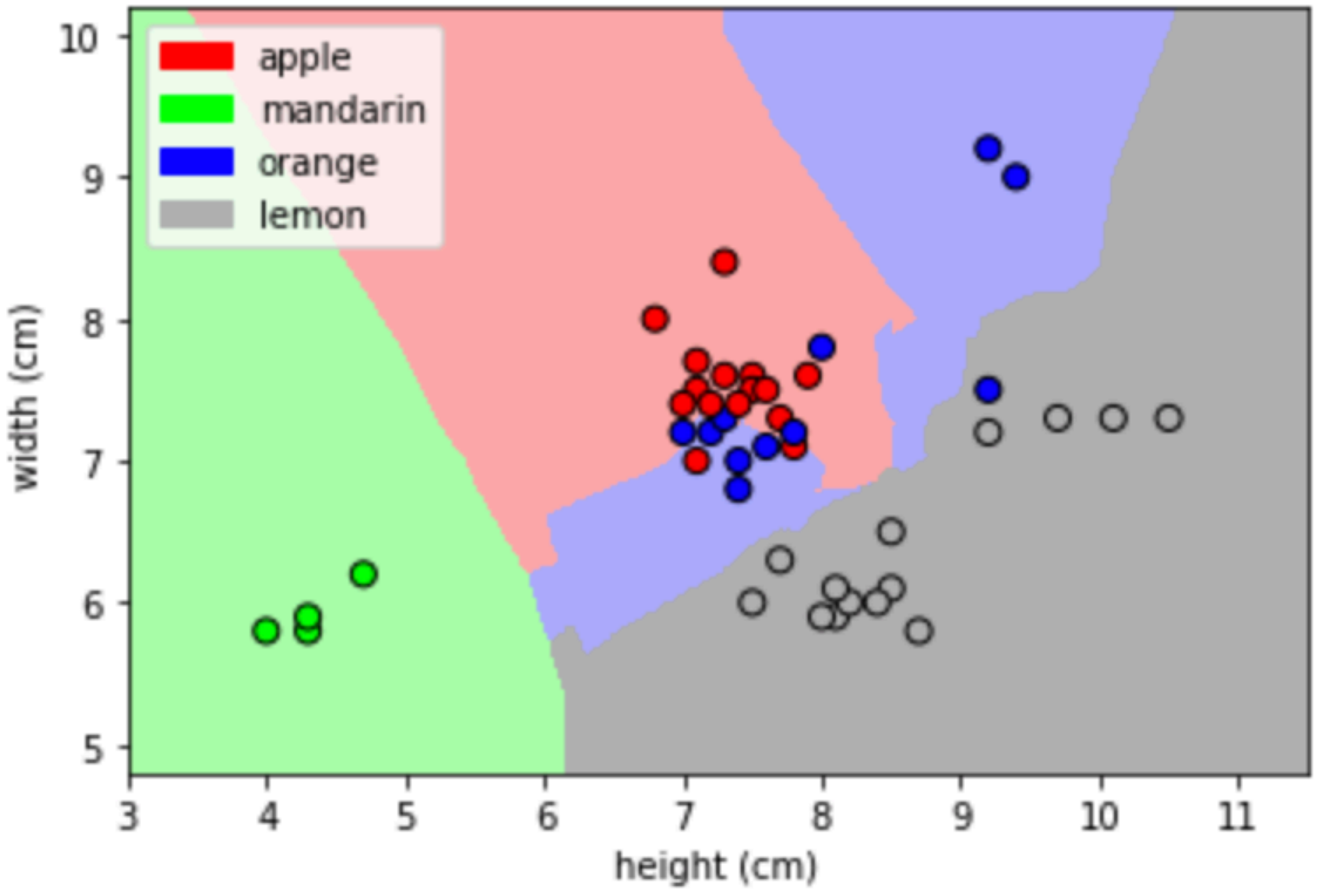
Iris dataset

	sepal_length	sepal_width	petal_length	petal_width
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2

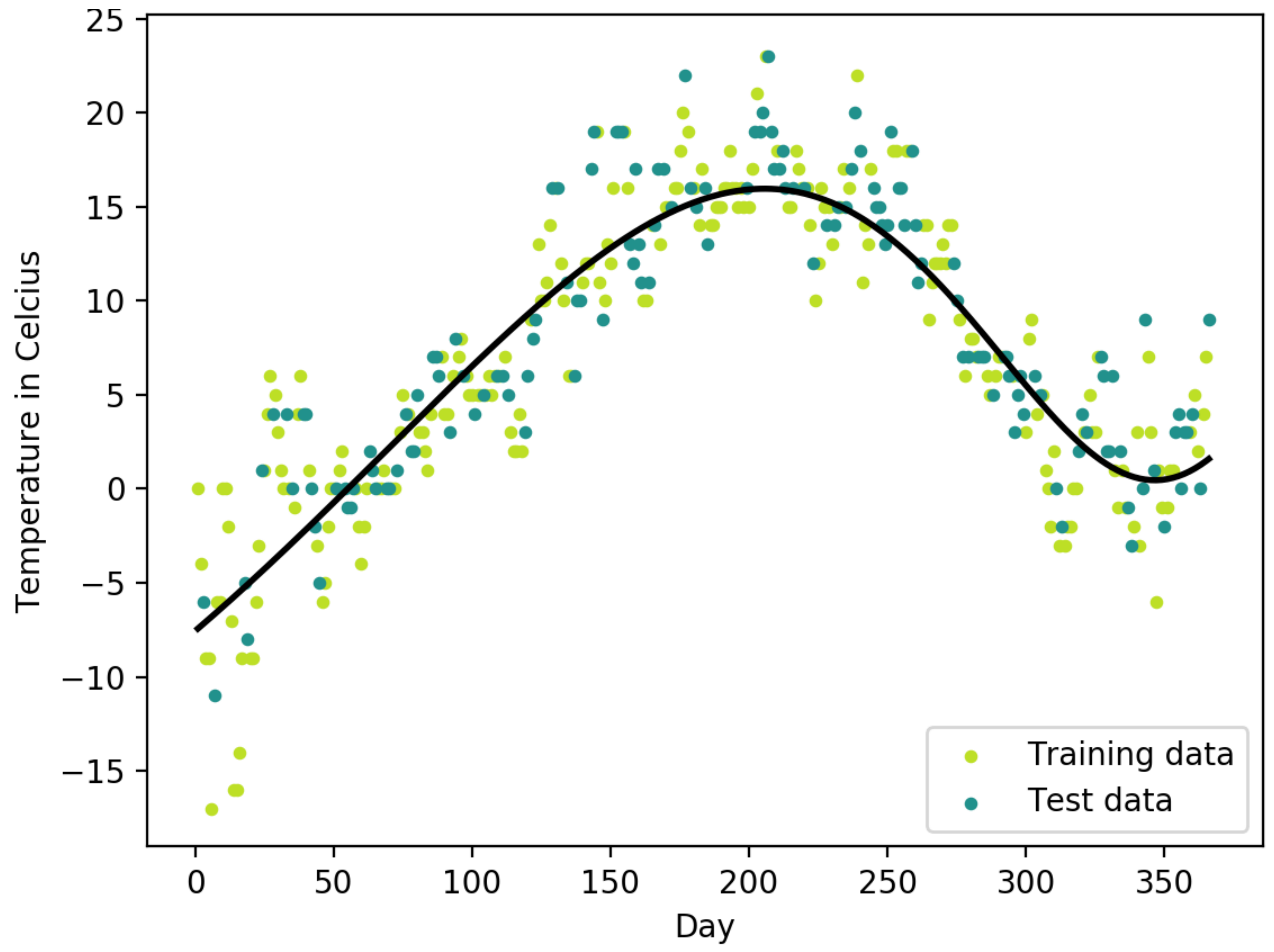
Tasks in Supervised learning
Regression versus Classification

Regression vs. Classification

Classification



Regression



$y^i \in \{1, 2, \dots, K\}$

ex: {apple, mandarin, orange, lemon}

$y^i \in \mathbb{R}^m$

ex: $m = 1$

Regression vs. Classification
Real-world problem

The boundary between regression and classification is not always obvious:

possible to transform a regression problem into a classification problem (and vice versa).

Examples:

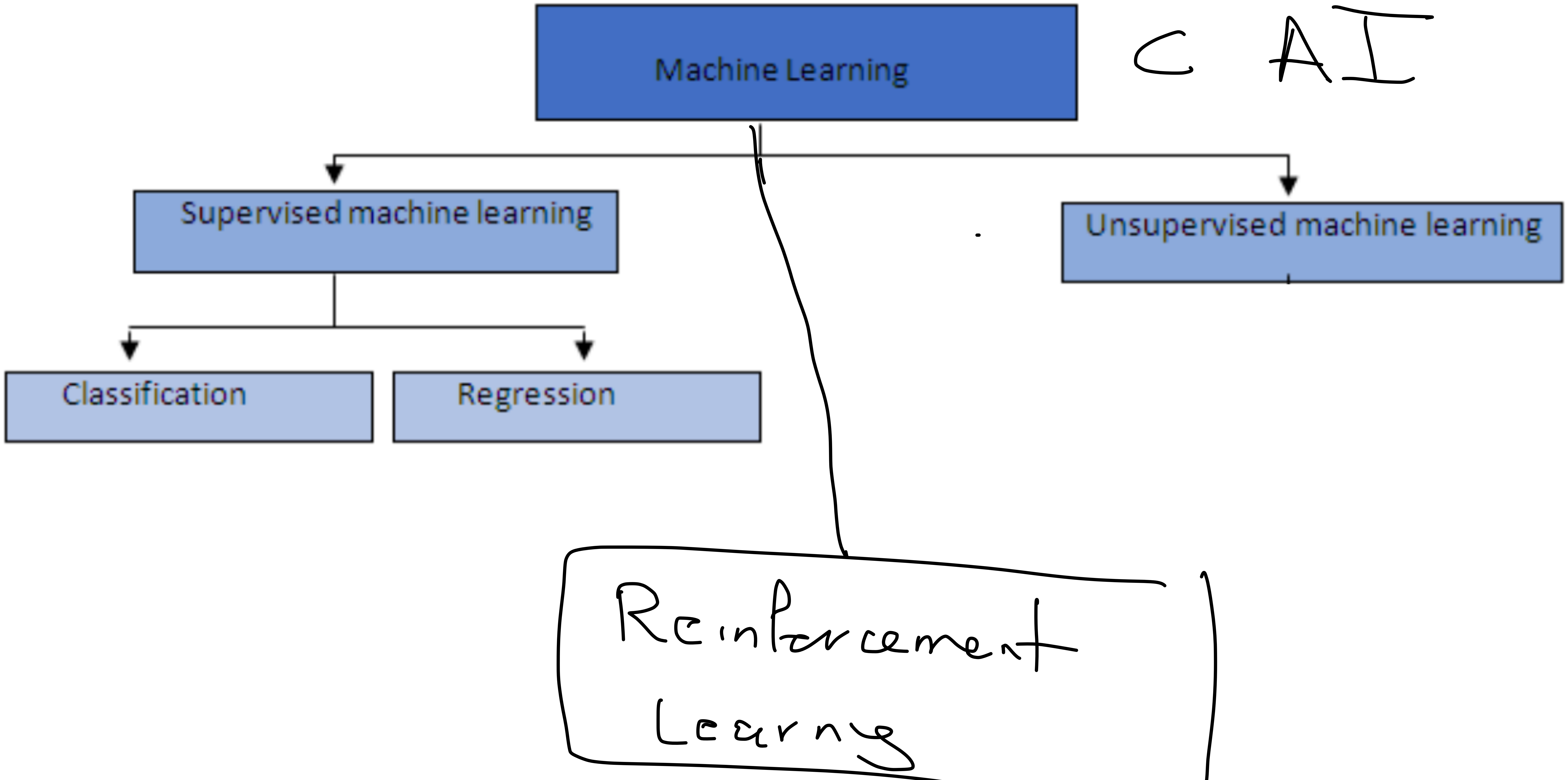
- Predict power generated by solar panels → predict power range

regression → $y^i \in \mathbb{R}$, classification → $[0, 100], [100, 200], [200, 300]$

- Predict faults → predict the probability, p, of a fault

classification: {fault, no fault} , regression: prob of being faulty $\in \mathbb{R}$ $[0, 1]$

Regression vs. classification



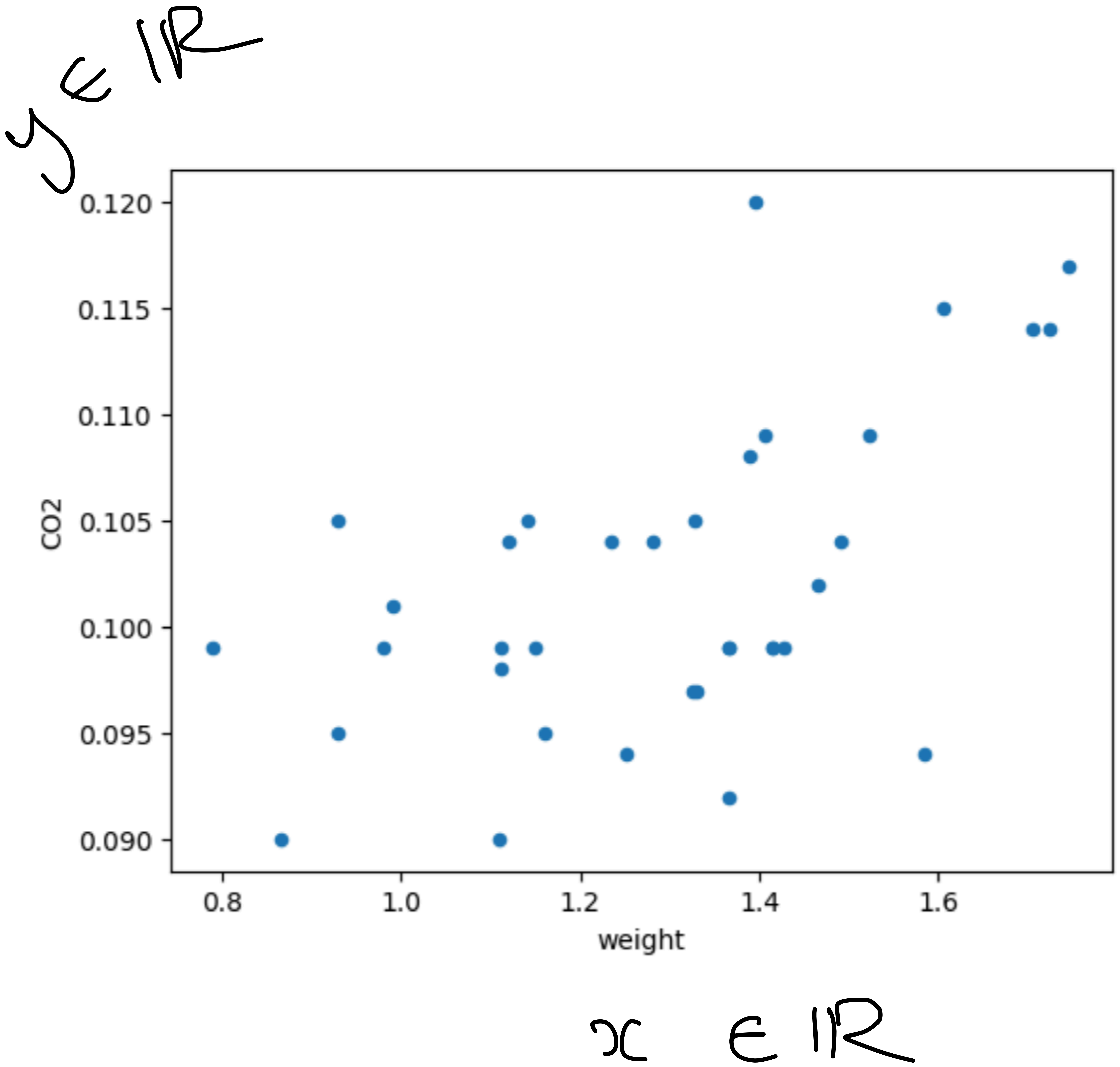
Linear regression

Linear regression

- **Example exercise on python**
  - Data: CO_2 emission and weight of cars
  - Goal**:** Predict the emissions of a car as a function of its weight

Linear regression

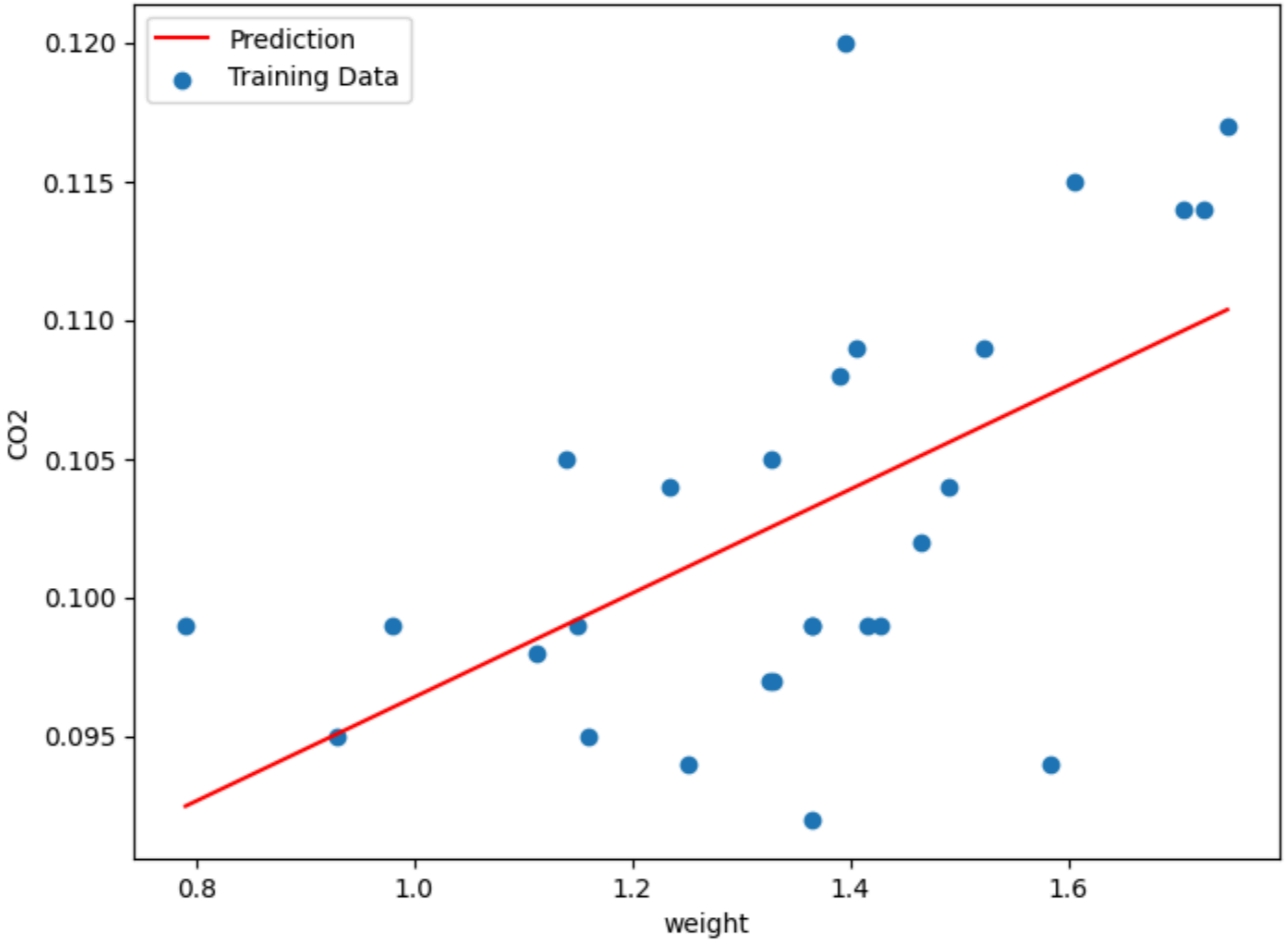
Introduction



- Problem:
  - Predict CO_2 emission of cars based on their weights
  - Why?
    - assist car manufacturers in designing more fuel-efficient and environmentally friendly vehicles
    - Assist policymakers in formulating regulations to curb emissions.

Linear regression

Introduction



- Approach:
  - Model the relationship between weights and CO2 emission as linear
  - Why?
    - It's just a hypothesis…
    - We'd have to verify whether it's a good hypothesis..

Definition: Linear functions, affine functions

<u>Definition</u> $f : \mathbb{R}^d \longrightarrow \mathbb{R}^m$ <u>linear</u> if

$\forall \alpha, \beta \in \mathbb{R}, \quad a, b \in \mathbb{R}^d$

$$f(\alpha a + \beta b) = \alpha f(a) + \beta f(b)$$

ex: $f(x) = Ax \ , \quad x \in \mathbb{R}^d, \ A \in \mathbb{R}^{m \times d}$

$f : \mathbb{R}^d \longrightarrow \mathbb{R}^m$ <u>affine</u> if $\exists \ c \in \mathbb{R}^m$

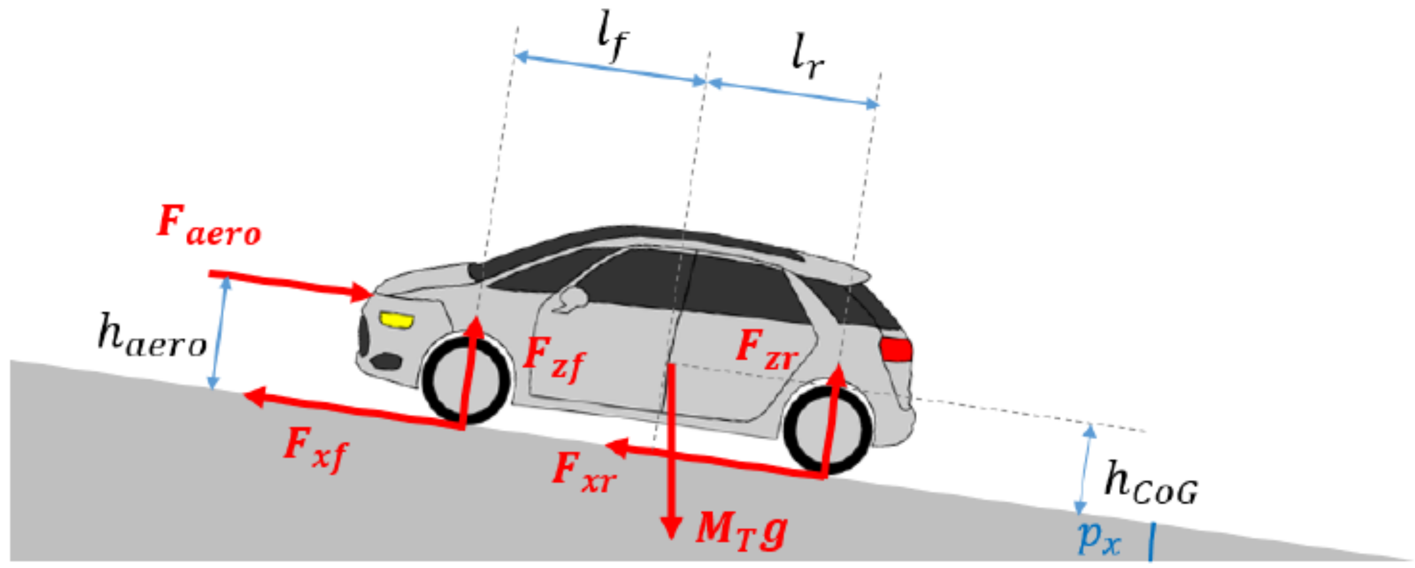
such that (s.t.) $f - c : \mathbb{R}^d \longrightarrow \mathbb{R}^m$ is

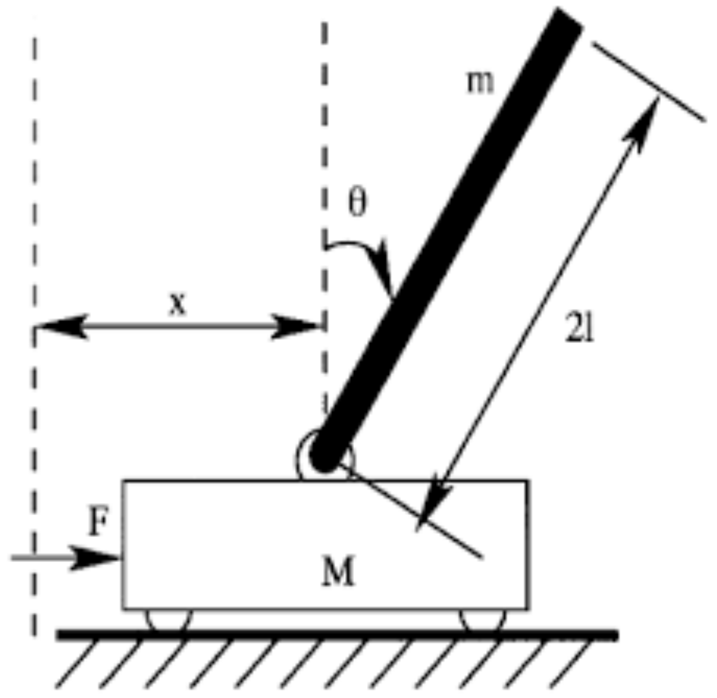
linear. ex: $f(x) = Ax + c$

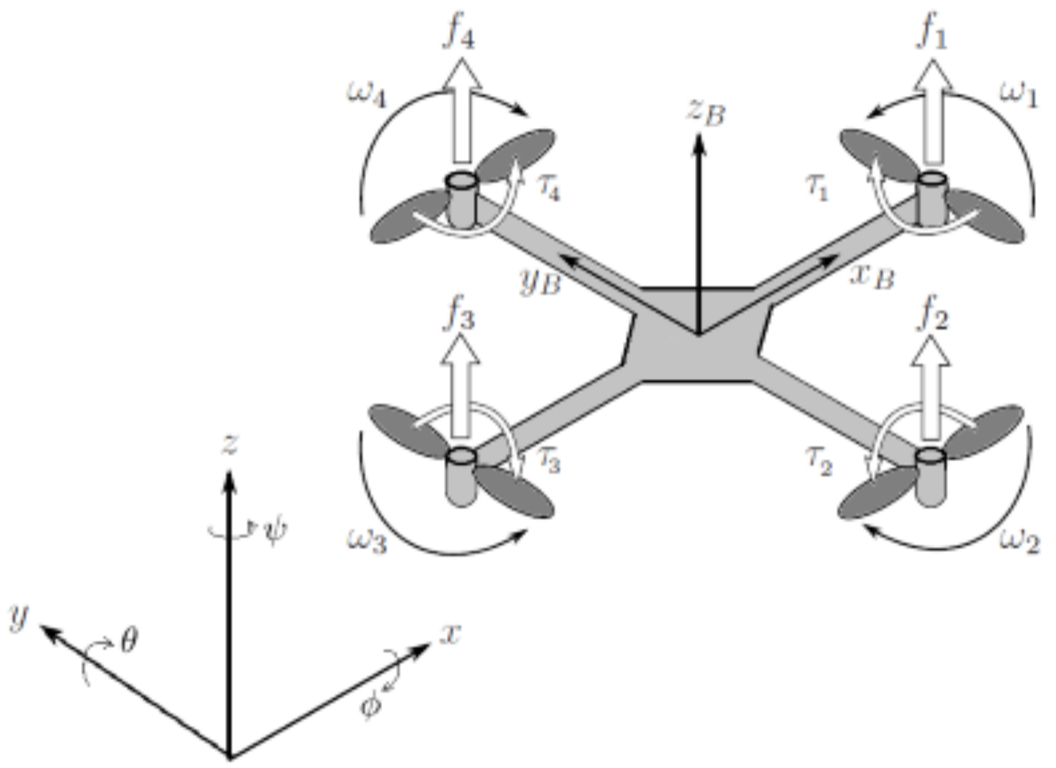
Motivation for linear regression

Linearity is a property of vector spaces and functions widely used in engineering

Examples of linear dynamical models







Linear predictors

Training data $\{x^i, y^i\}_{i=1}^{N}$, $x^i \in \mathbb{R}^d$, $y^i \in \mathbb{R}$

Linear predictor $f_w(x) = b + w_1x_1 + \ldots + w_dx_d$, $x^i \mapsto f_w(x^i)$

with $w' = [b, w_1, \ldots, w_d]^T$, $x' = [1, x_1, \ldots, x_d]^T$, $f(x) = w'^T x'$

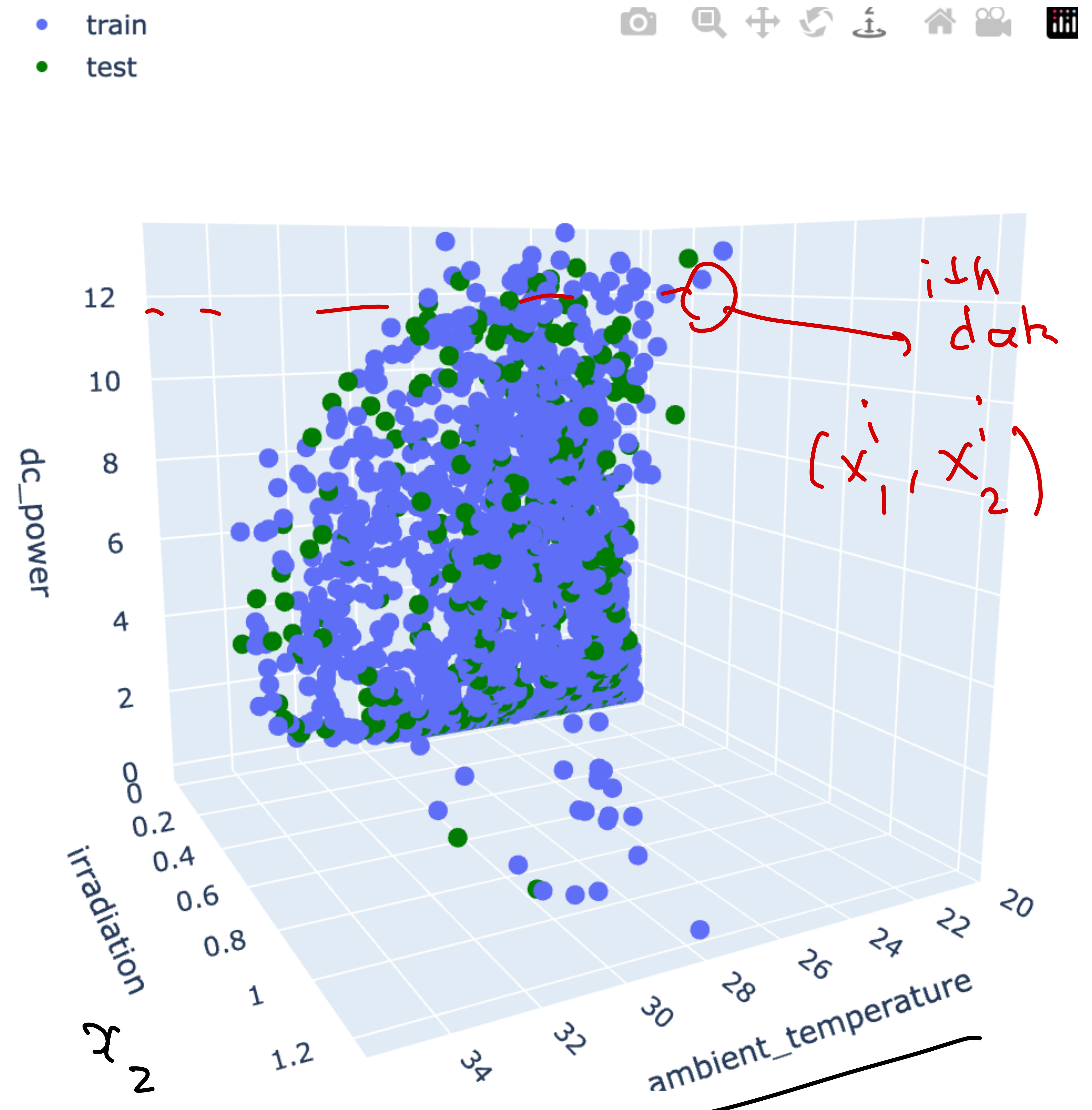
$w' \in \mathbb{R}^{d+1}$

Tuning, training, fitting a parameter: finding the best parameter:

finding the best $[b, w_1, \ldots, w_d]$

Exercise

- Goal: predict solar power generation as a function of measurements of irradiation (sunlight intensity) and ambient temperature.
- Why? Optimize performance of a solar power plant
- Questions $x^i \in \mathbb{R}^2$
  - What are the input features (independent variables)?
  - What is the output label (dependent variable)? $y^i \in \mathbb{R}$
  - How would you find a linear predictor given data?



hypothesize $f_w(x) = b + w_1 x_1 + w_2 x_2$, b, w_1, w_2

Summary

- Intro to course admin structure

- Intro to AI and ML
- Supervised versus unsupervised learning
- Notation for vectors and functions
- Linear regression as a supervised learning task
- Linear functions, affine functions, linear predictor

Your tasks

- Go through the “Course information” on Moodle to familiarise yourself with the course
- Go through the lecture notes and material posted on Moodle
- Exercise hour on 10.09.2025
  - Bring your laptop
  - You’ll get familiar with Python
    - Get further familiar with python

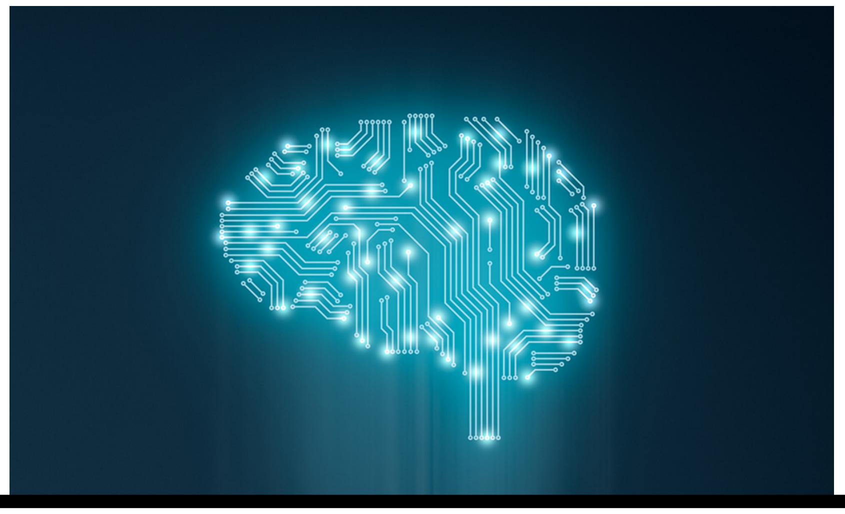
- Next lecture: linear regression through a loss function

Your tasks

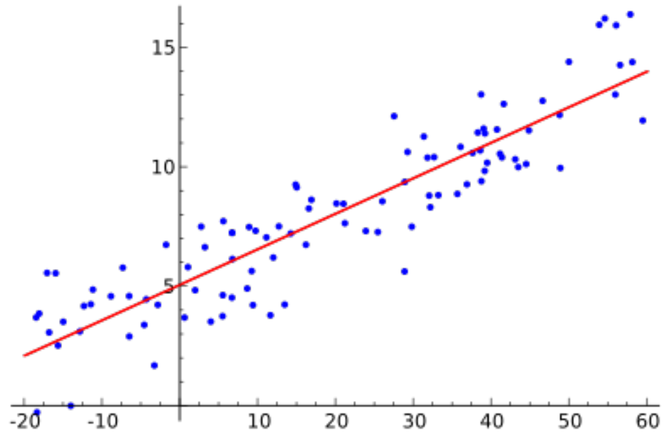
- Go through the “Course information” on Moodle to familiarise yourself with the course
- Go through the lecture notes and material posted on Moodle
- Exercise hour on 10.09.2025
  - Bring your laptop
  - You’ll get familiar with Python
    - Get further familiar with python

- Next lecture: linear regression through a loss function

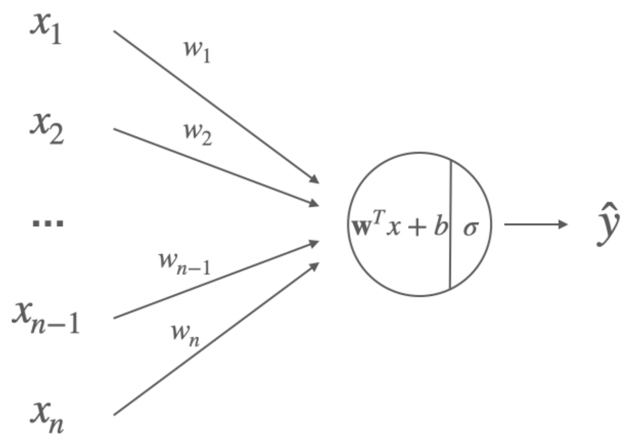
Introduction



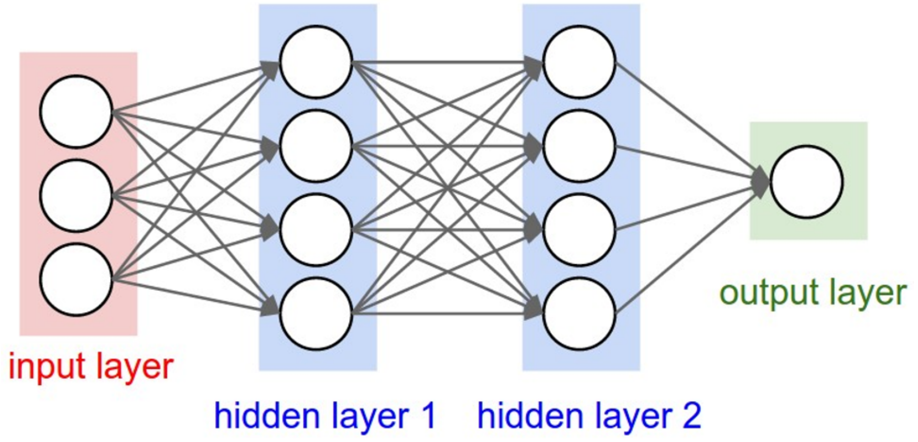
Linear regression



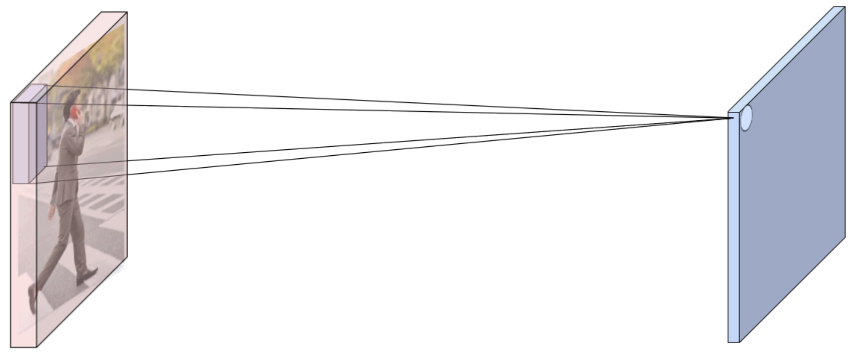
Logistic regression



Neural networks

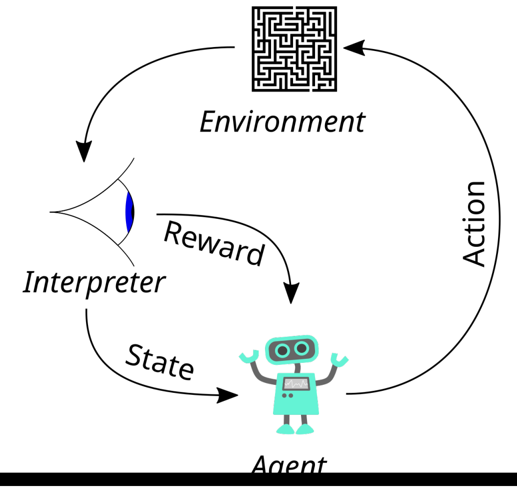


Convolutional neural networks



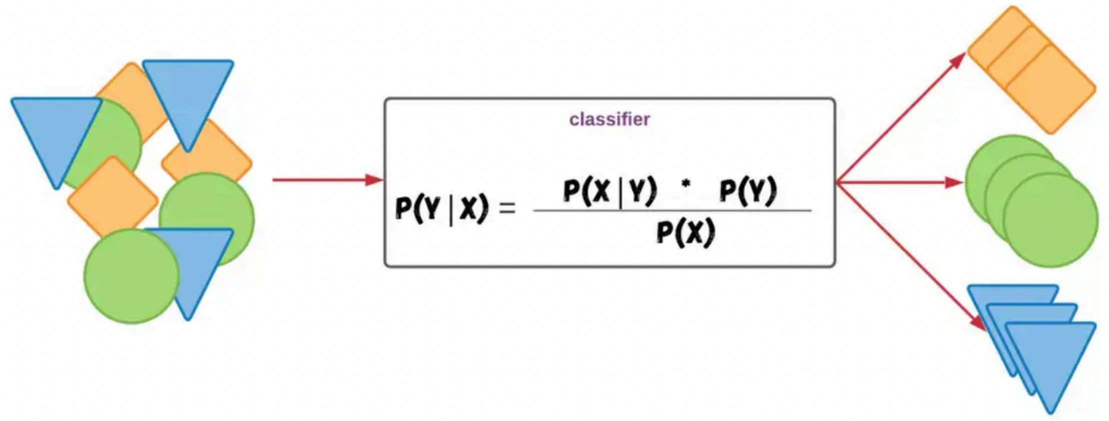
AI & sustainability

Reinforcement learning

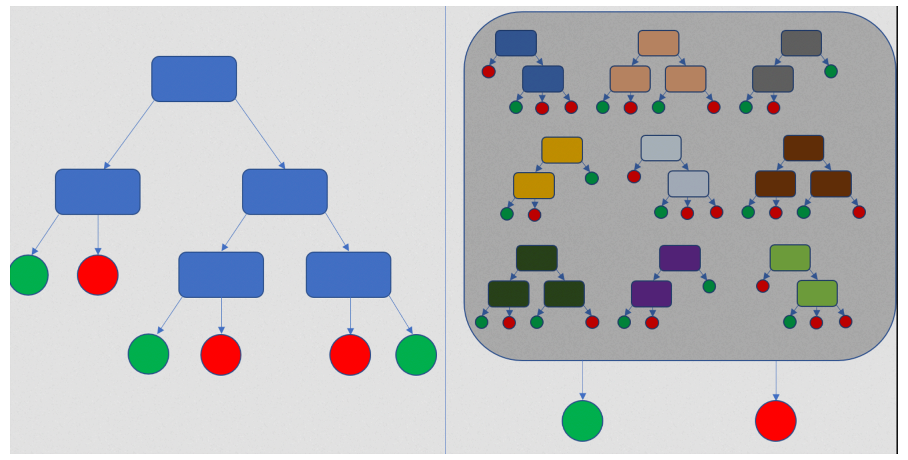


KNN

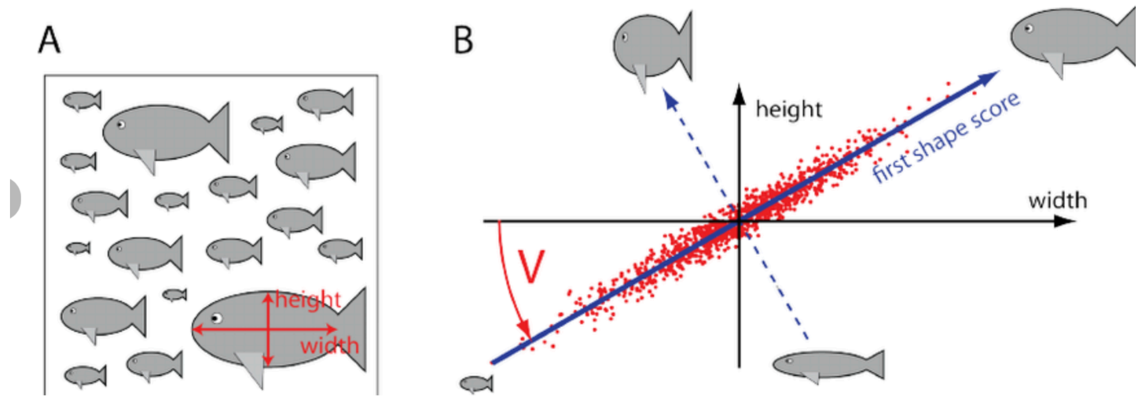
Naive Bayes



Decision-trees



Dimensionality reduction



Clustering