

Introduction

09.09.2024

- Administrative
 - Staff, Grading
- Introduction to AI
- Introduction to ML
 - Supervised versus Unsupervised learning
 - Linear regression

Administrative

Welcome to ME-390

Administrative Teacher Staff

Teacher

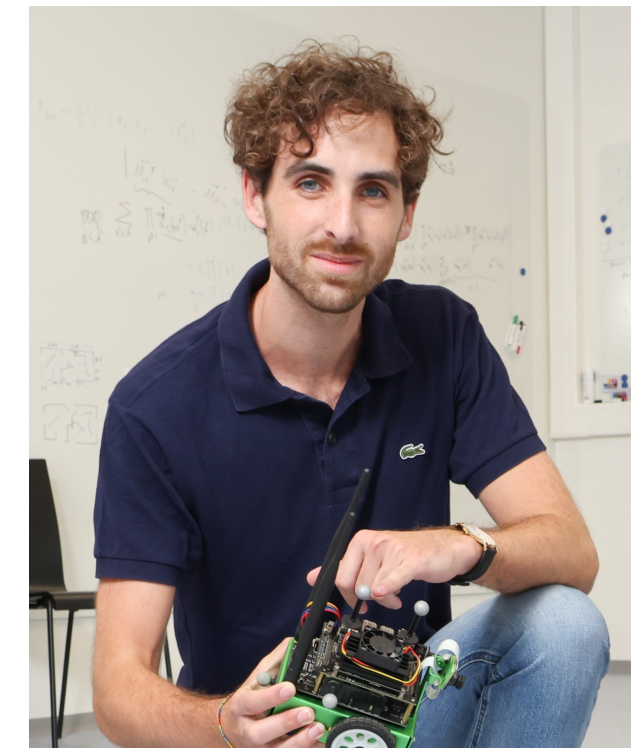


Maryam Kamgarpour

TAs



Kai Ren



Giulio Salizzoni



Saurabh
Vaishampayan



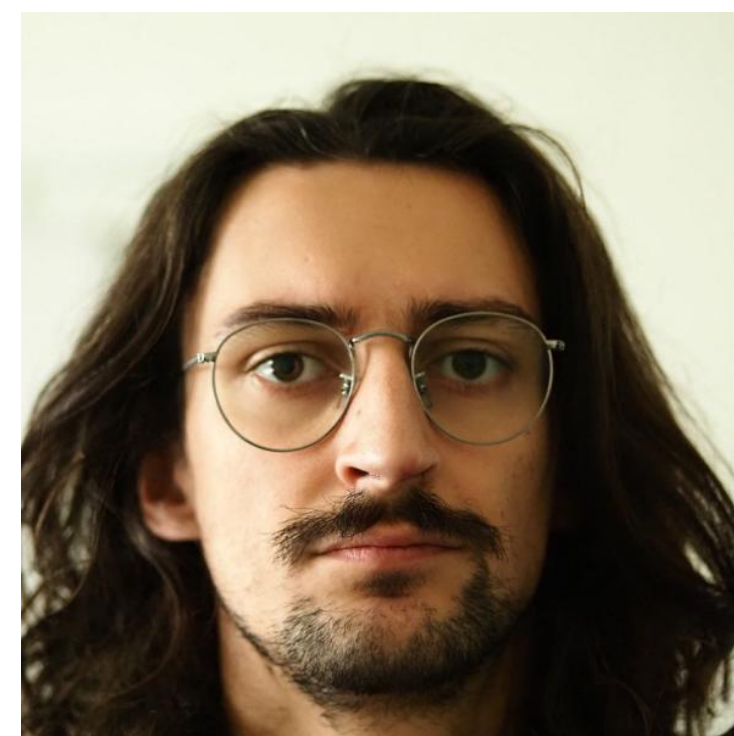
Anna Maddux



Tingting Ni



Andreas
Schlaginhaufen



Gabriel Vallat

Administrative

Grading

This course will be graded as follows:

- Math problem sets and python exercises (not graded)
 - Every week during the exercise hour
 - Solutions will be provided after one week
- 3 in-class **quizzes**: 10% each
 - Based on lectures, in-class activities, problem sets and python exercises provided
 - You will submit in-class
 - **Dates: Oct 16, Nov. 20, Dec 11**
- **Final Exam**: at least 70% (see below for grade calculation)
 - At the end of the semester during exam period

Your final grade: in-class quizzes count if they help your final mark

Grade =

$\max(70\% \text{final} + 10\% \text{quiz 1} + 10\% \text{quiz 2} + 10\% \text{quiz 3}, 80\% + 10\% \text{quiz1} + 10\% \text{quiz2}, \dots, 100\% \text{final})$

Administrative Technical requirements

Domains covered by ML

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

Administrative

Communication/class requirements

We use Moodle for lecture notes and videos

We use EdDiscussions for Q&A: please write your questions on EdDiscussions or bring them to class/exercise hour instead of individual emails

In-class: take time to give your full attention to the course, away from distractions. Be respectful (no talking/no phone)

Administrative

Additional resources

The notes are based on a few sources:

- CIVIL-226
- Machine Learning for Engineers, Using Data to Solve Problems for Physical Systems by Ryan G. McClarren
- EE 104, Stanford

Online courses and resources

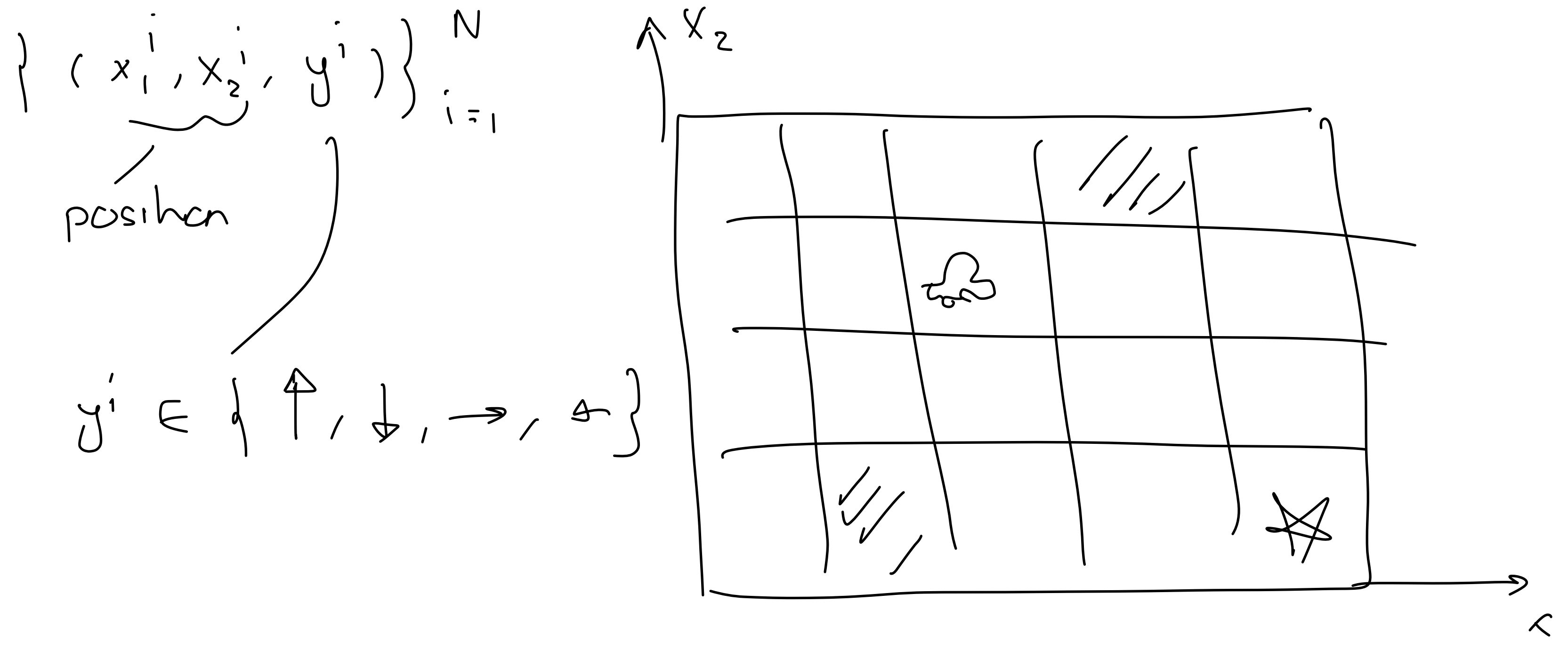
- There are (too) many online resources, I have listed some on our Moodle website
- The following book covers the relevant mathematical background for the course:
Introduction to Applied Linear Algebra – Vectors, Matrices, and Least Squares

Introduction to AI

An Example in robotics - Imitation learning

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

From expert demos to learning



What is AI?

AI is intelligence demonstrated by machines...

But how do we define intelligence?

Activity: Think, Pair, Share

History of AI

History of AI

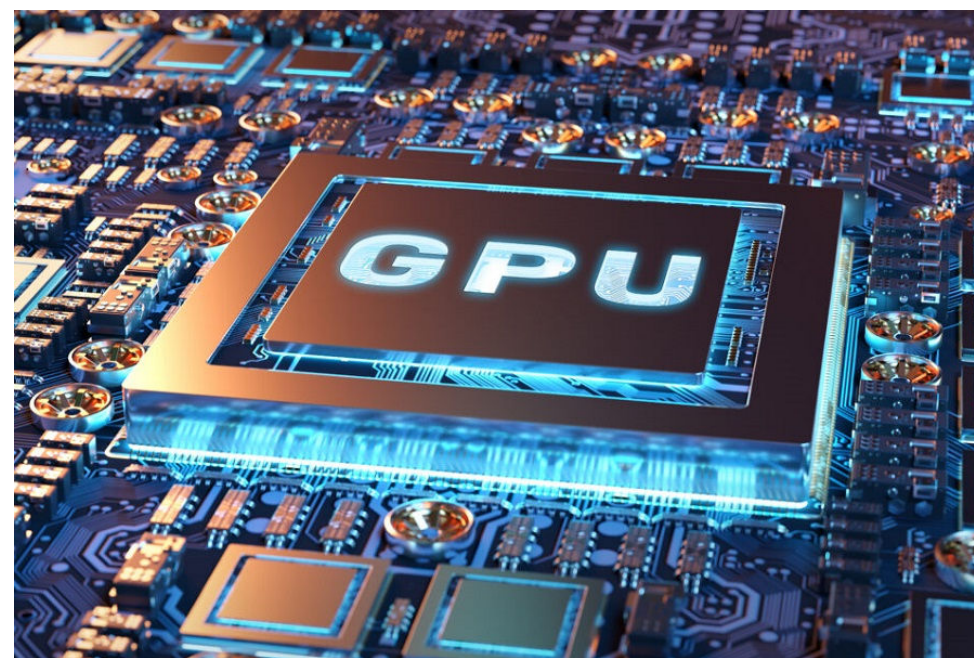
Interplay between **Data** and algorithms

Most AI breakthroughs used old algorithms but recent data sets and computational tools.

Year	Breakthroughs in AI	Datasets (First Available)	Algorithms (First Proposed)
1994	Human-level spontaneous speech recognition	Spoken Wall Street Journal articles and other texts (1991)	Hidden Markov Model (1984)
1997	IBM Deep Blue defeated Garry Kasparov	700,000 Grandmaster chess games, aka "The Extended Book" (1991)	Negascout planning algorithm (1983)
2005	Google's Arabic- and Chinese-to-English translation	1.8 trillion tokens from Google Web and News pages (collected in 2005)	Statistical machine translation algorithm (1988)
2011	IBM Watson became the world Jeopardy! champion	8.6 million documents from Wikipedia, Wiktionary, Wikiquote, and Project Gutenberg (updated in 2010)	Mixture-of-Experts algorithm (1991)
2014	Google's GoogLeNet object classification at near-human performance	ImageNet corpus of 1.5 million labeled images and 1,000 object categories (2010)	Convolution neural network algorithm (1989)
2015	Google's Deepmind achieved human parity in playing 29 Atari games by learning general control from video	Arcade Learning Environment dataset of over 50 Atari games (2013)	Q-learning algorithm (1992)
Average No. of Years to Breakthrough:		3 years	18 years

AI now...

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



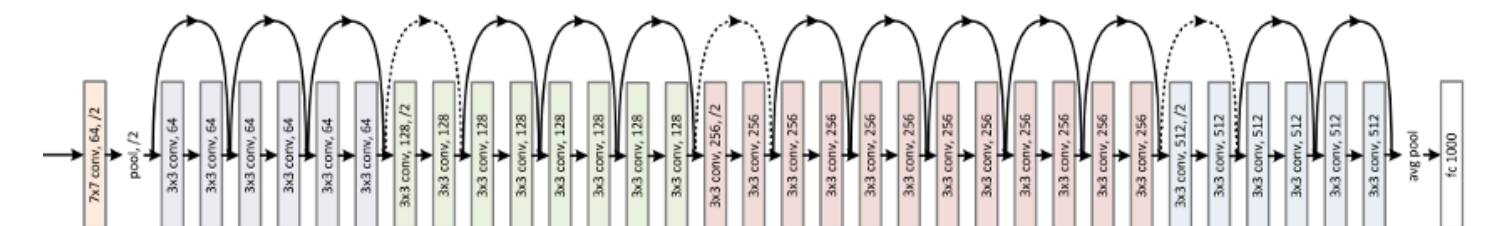
Hardware

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

What is ML? Is it AI?

AI vs ML



Artificial Intelligence

What is ML? Is it AI?

AI vs ML



Artificial Intelligence



Machine Learning

What is ML? Is it AI?

AI vs ML



Artificial Intelligence

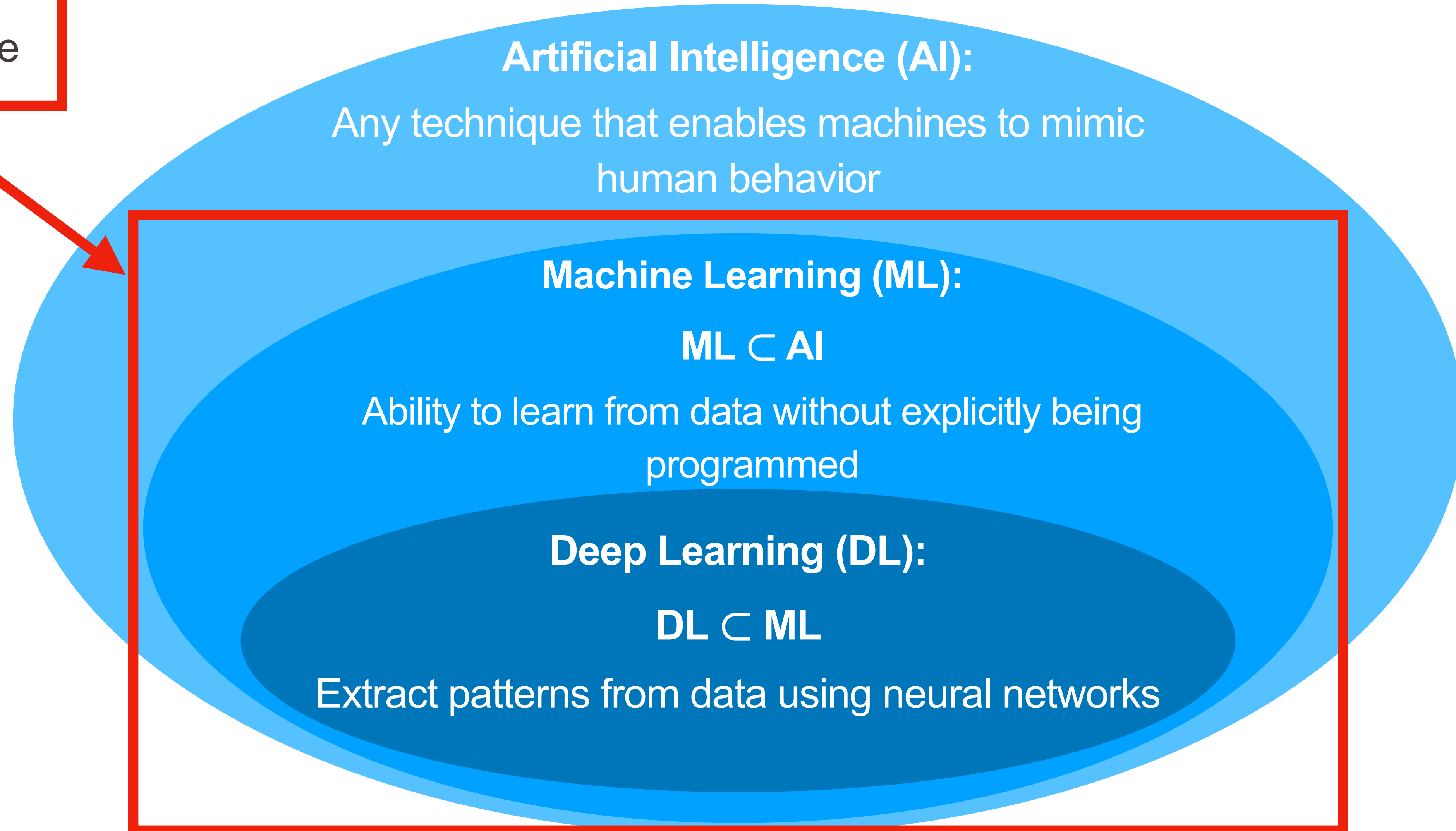


Machine Learning

Introduction to ML

AI vs ML vs DL - The differences

This course



What is ML?

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

AI vs ML - Deep Blue

- In 1997, Deep Blue system (IBM) defeated the world chess champion, Gary Kasparov.
- It used brute force by choosing the move that would lead to the best possible final board position → used computing advantage.



AI vs ML - AlphaGo

- In 2015, AlphaGo (DeepMind) defeated one of the strongest Go players in the world.
- AlphaGo uses neural networks to teach itself to play from data → machine learning.



Introduction to ML

AI vs DL - Deep Blue vs. AlphaGo

Deep Blue (AI)

- Deep Blue is an AI but not ML because it doesn't learn from data.
- It was programmed to know the chess rules and it used brute force to construct a search tree over all the possible final positions based on its move.

AlphaGo (ML)

- The possible positions of Go is $> 10^{100}$ → we can't use brute force.
- AlphaGo teaches itself to play Go using a neural network and based on human and computer Go playing data. No explicitly coded strategies of Go.

Examples of AI achieved with ML



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



Alexa



Siri



Google Now



Cortana

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



Research example - Text to image

The following image were created from scratch based on the input text.

Input:	“An arm chair in the shape of an avocado”	“A collection of glasses is sitting on a table”	“A cross-section view of a walnut”
Output:	 A 3D rendered image of a green, avocado-shaped armchair. The chair has a smooth, rounded back and a matching circular seat, all in a vibrant green color. It is positioned on a light-colored wooden surface against a plain white background.	 A photograph showing a collection of several different styles of eyeglasses, including round, rectangular, and aviator frames, scattered on a light brown wooden table. The glasses are arranged in a somewhat overlapping manner.	 A close-up photograph of a walnut that has been split open, revealing the light-colored, wrinkled kernel inside the dark brown, textured shell. The background is a soft, out-of-focus light blue.

Activity: Think, Pair, Share

- Where you have used AI in the past
- Was the AI approach an ML approach (i.e. based on data)?

Why should you care about this AI course as an IGM student?

- You daily interact with ML techniques (examples based on your activity)
- You most likely will use some of these techniques in your future courses or semester projects (I will give you some examples)
- You will most likely use ML techniques in upcoming internships, and jobs
- Knowledge is power!
 - You'd get to know the basics and the AI will appear less mysterious to you

Examples of ML in IGM

Data-driven optimisation and control of VAWT

@ EPFL
UNSTEADY FLOW
DIAGNOSTICS LAB
UNFOLD

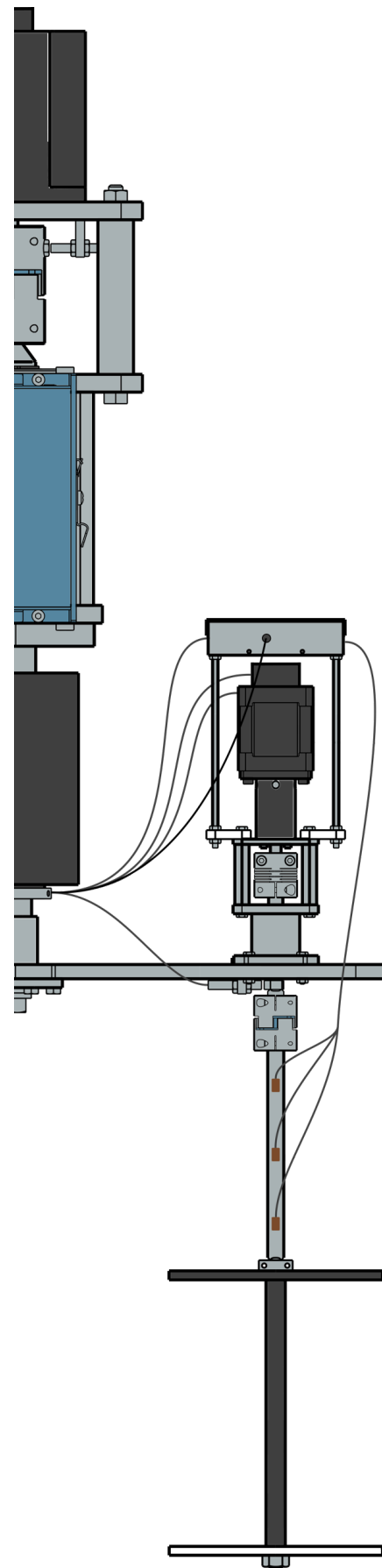
VAWT = vertical axis wind turbine

HAWT = horizontal axis wind turbine

Why VAWT?

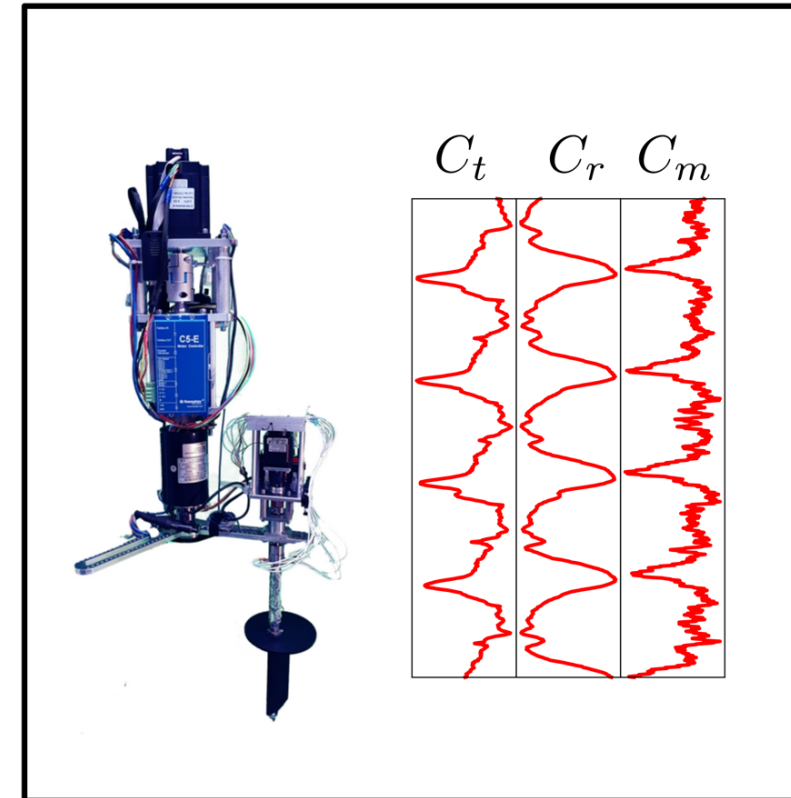
- less noisy
- insensitive to wind direction
- easier to commission and maintain
- higher power density

Data-driven control of VAWT

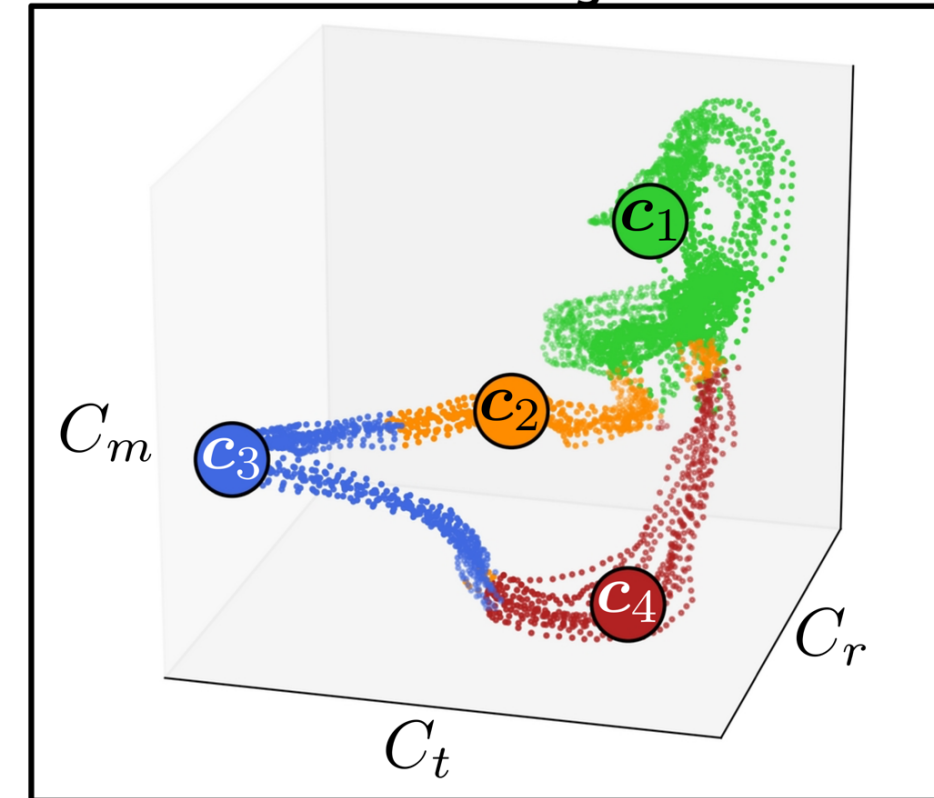


OFFLINE

1. Baseline data collection

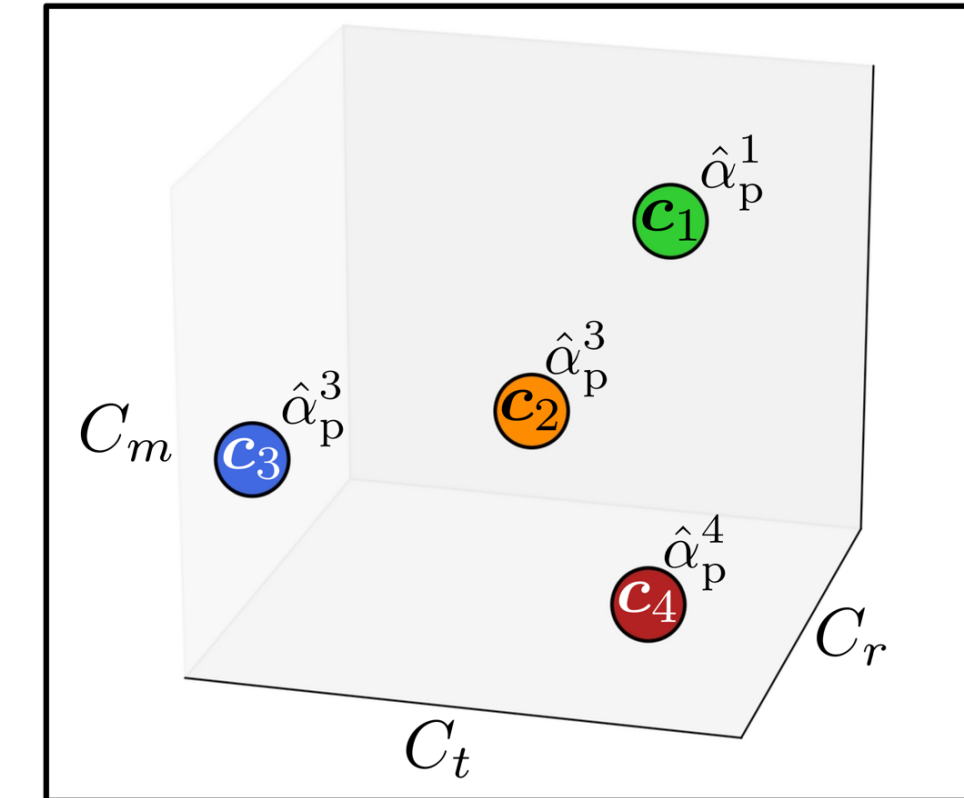


2. Clustering

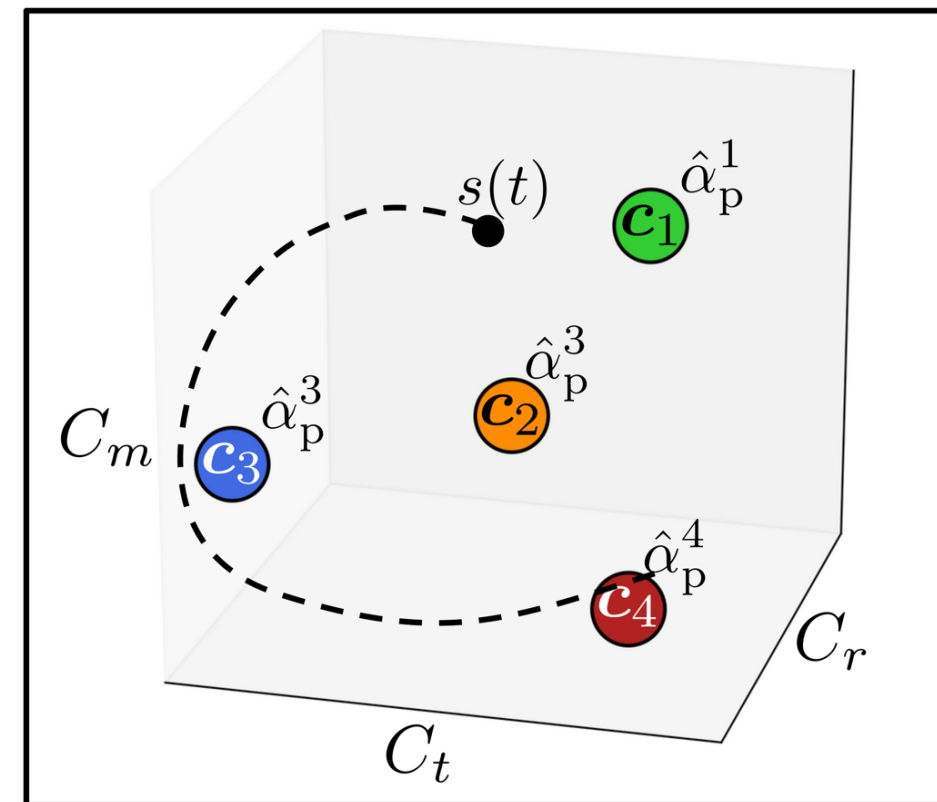


3. Selection of a control law

$$\hat{\alpha}_{\text{pitch}} = [\hat{\alpha}_p^1, \dots, \hat{\alpha}_p^4]$$



4. Real time monitoring of the state
 $s(t) = [C_t(t), C_r(t), C_m(t)]$



5. Real time computation of the next pitch angle

$$\alpha_{\text{pitch}}(t) = \frac{\sum_{k=1}^K \hat{\alpha}_p^k e^{-\|s(t) - c_k\|^2}}{\sum_{k=1}^K \hat{\alpha}_p^k e^{-\|s(t) - c_k\|^2}}$$

ONLINE

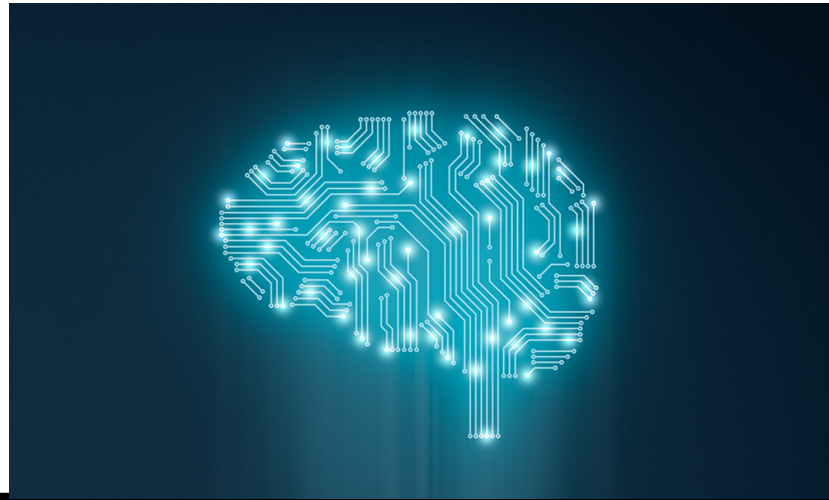
At the end of the course

You will:

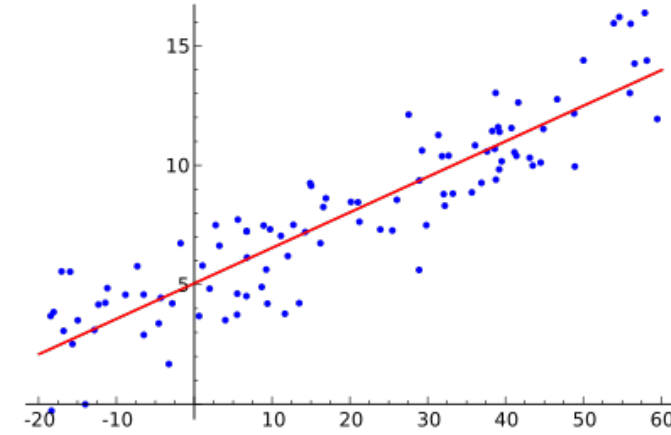
- Know the theory behind common ML techniques
- Know how to implement ML techniques in Python using popular data science libraries
- Have the tools needed to start a ML project on your own
- Be able to read ML literature

Course topics

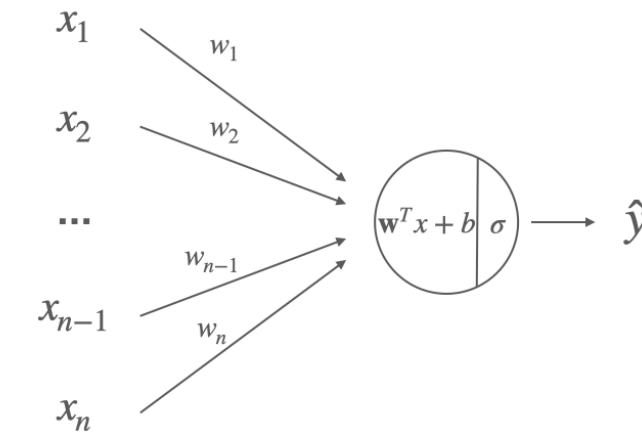
Introduction



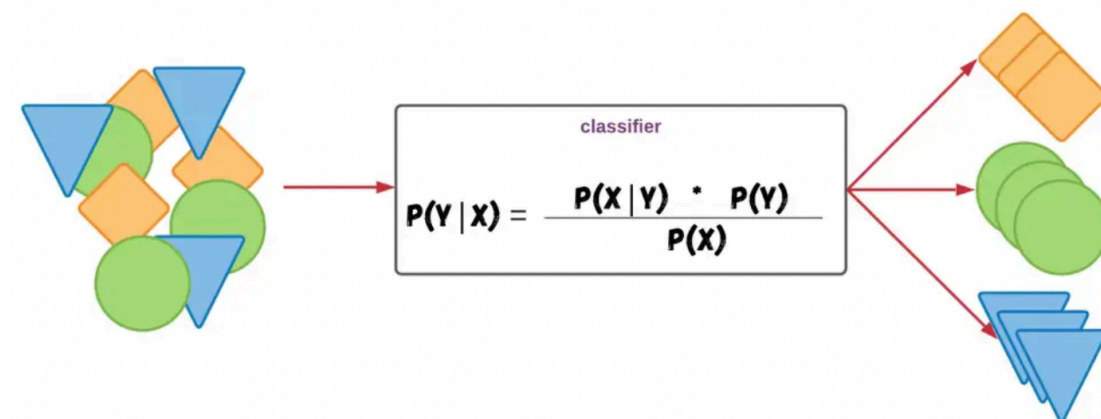
Linear regression



Logistic regression



Naive Bayes



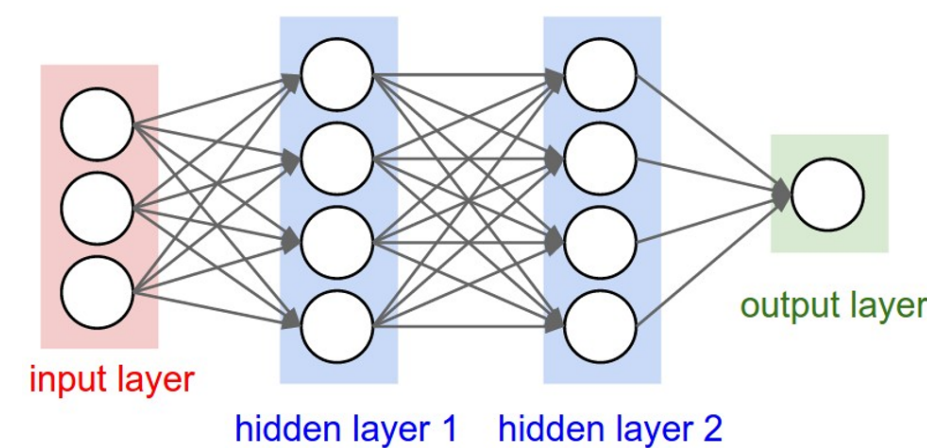
KNN



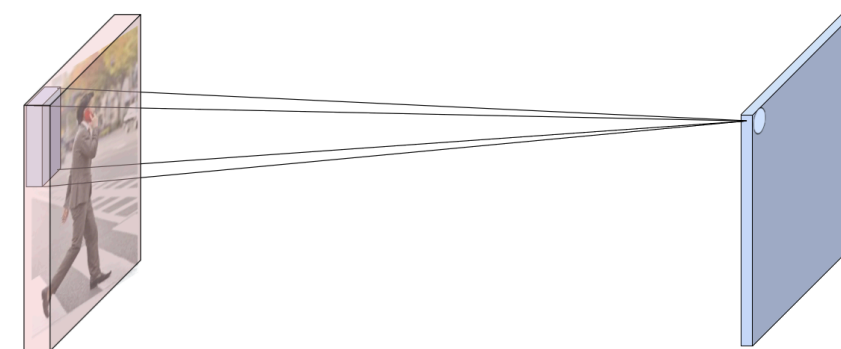
AI ethics



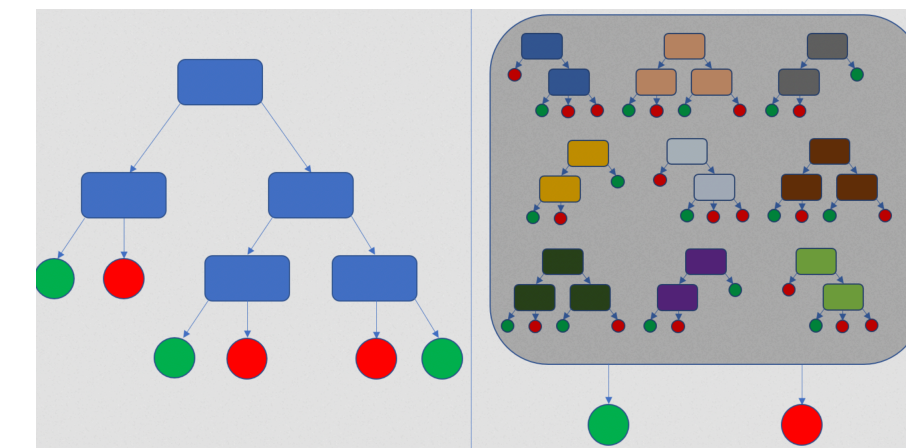
Neural networks



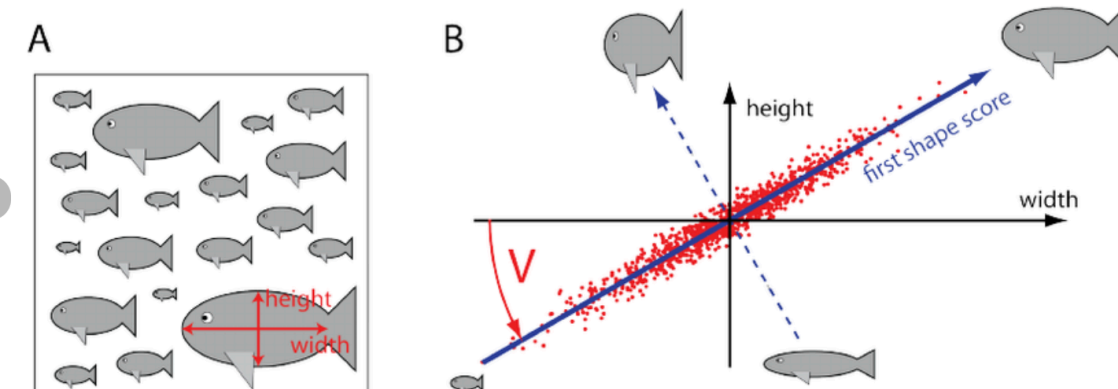
Convolutional neural networks



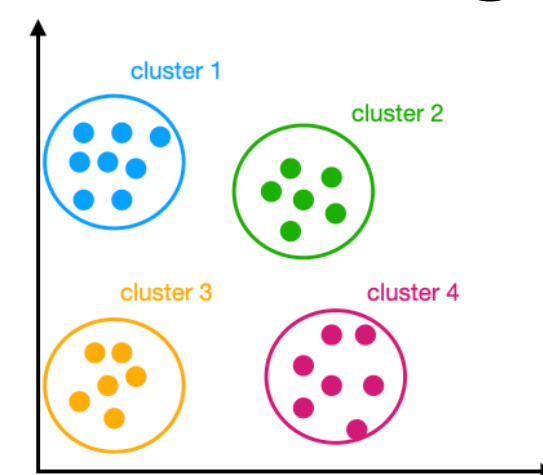
Decision-trees



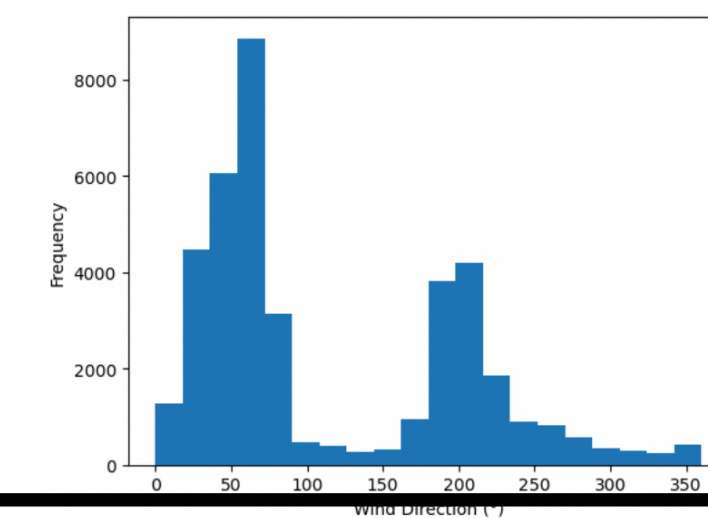
Dimensionality reduction



Clustering



Reinforcement learning



- Supervised learning
 - Linear regression
 - Logistic regression
 - Naive Bayes
 - K-nearest neighbour
 - Decision-trees
- Unsupervised learning
 - Dimensionality reduction: PCA
 - Clustering: k-means
- Reinforcement learning

- Linear algebra
 - Vectors, linear equations
 - Matrices: eigenvalues, positive (semi) definite, invertibility
- Optimizaiton
 - Gradient descent
 - Optimality conditions
 - Regularization
- Statistics
 - Mean, variance, covariance, mode
 - probability distribution
 - Empirical estimates
- Feature engineering

Introduction to ML

- Experience (data)
 - e.g. wind speed, wind direction, power output
- Task (what we hope to do with this data)
 - e.g.: 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 function), e.g: error in prediction

Types of experience

- **Supervised learning**

- Labels are available (desired outputs)

- **Semi-supervised learning**

- Some labels are missing

- **Unsupervised learning**

- No labels

- **Reinforcement learning**

- Labels (in form of rewards and punishments) are given only as feedback to the program's actions in a dynamic environment

- **Active learning**

- Limited set of labels are available and can be interactively selected

Notation: sets, vectors, matrices

Set : a set of k elements $S = \{i_1, i_2, \dots, i_k\}$, equivalently as $\{i_l\}_{l=1}^k$. example, $S_a = \{1, 2\}$, $S_b = \{2, 3\}$.

Cartesian product of ^{two} sets : $S_a \times S_b = \{s_1, s_2 \mid s_1 \in S_a, s_2 \in S_b\}$

ex : $(1, 3) \in S_a \times S_b$ but $(1, 1) \notin S_a \times S_b$.

vectors : $a \in \mathbb{R}^n$ $a = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}$, $a = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix}$, $a = (a_1, a_2, \dots, a_n) \in \mathbb{R}^n$

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

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

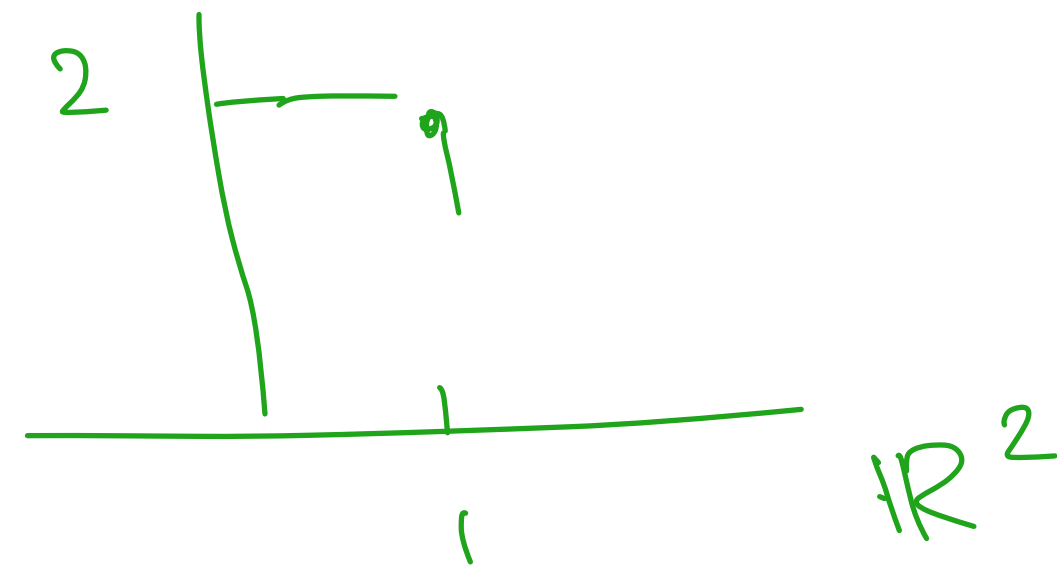
A^T : transpose of a matrix

Notation: sets, vectors, matrices

What's the difference between

$$(1, 2) \in \mathbb{R}^2$$

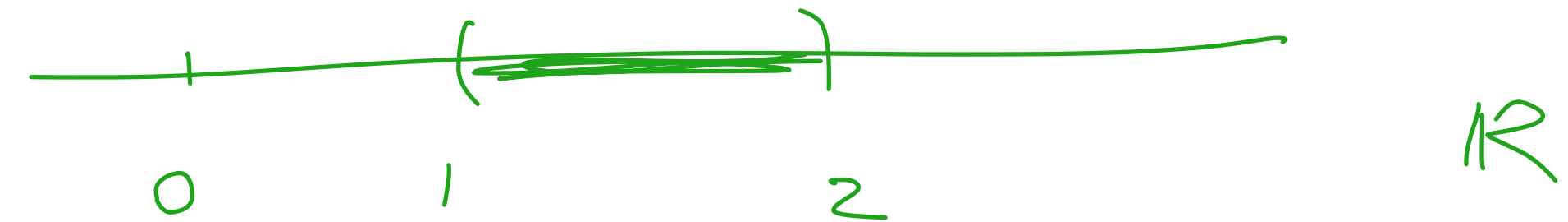
vector in $\mathbb{R}^2$



vs.

$$(1, 2) \subset \mathbb{R} \quad ?$$

open interval in $\mathbb{R}$



Notation: data

$\{(x^i, y^i)\}_{i=1}^N$ data set of N elements

ex: $x^i \in \mathbb{R}^2$ position of a car in 2D plane

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

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

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

$y^i \in \mathbb{R}^m$ continuous set.

x^i : independent variables, covariates, explanatory variables, feature vector
 y^i : dependent variables, target, outcome, label, response

Supervised vs. Unsupervised

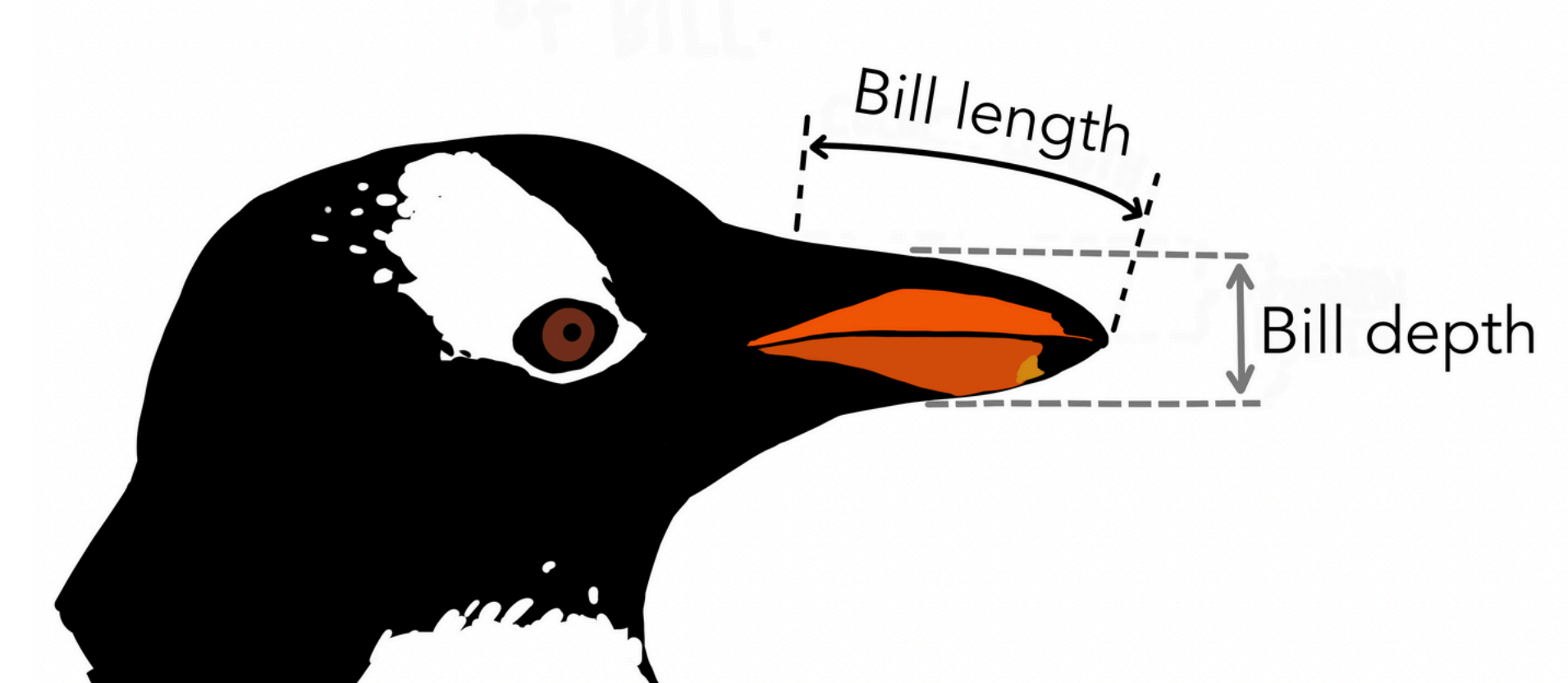
Supervised - Example

Palmer Penguins

y^1
 y^2

	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

$x^1 \in \mathbb{R}^4$
 $x^2 \in \mathbb{R}^4$



- The species of the penguin is used as a target label.

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

Supervised vs. Unsupervised

Unsupervised - Example

Iris dataset

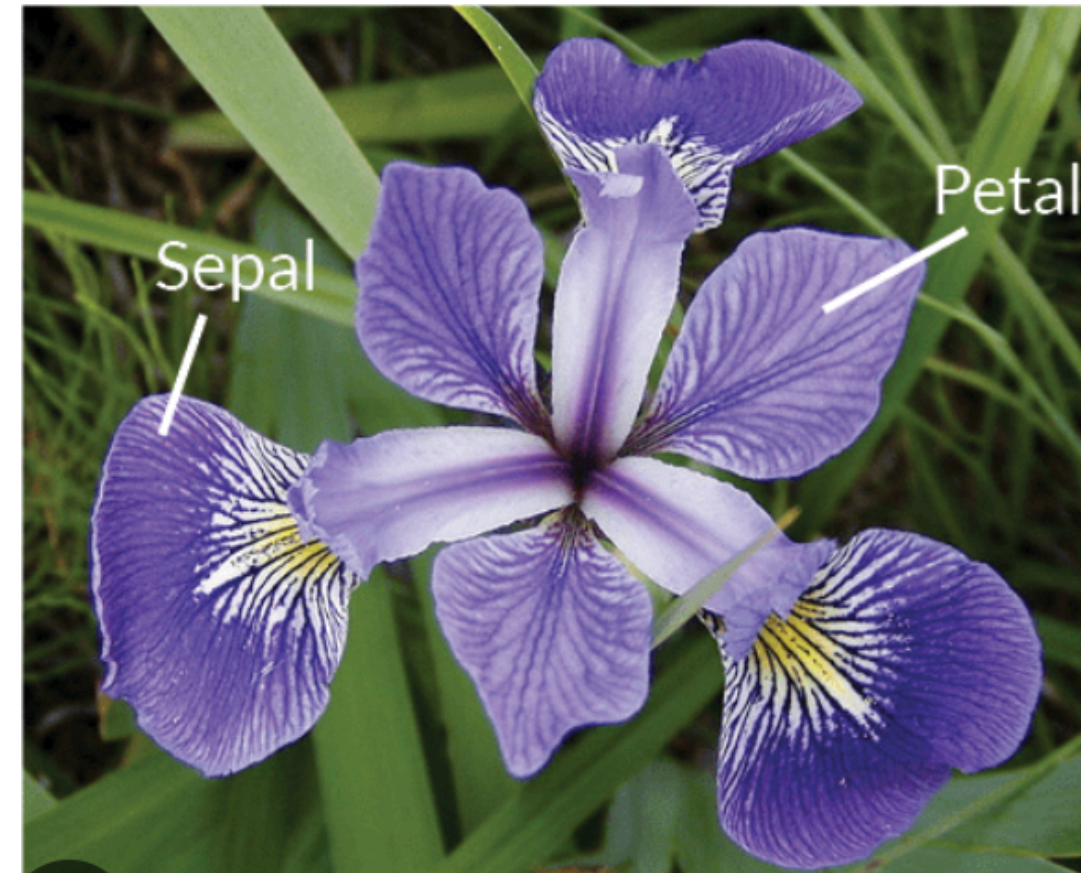
	sepal_length	sepal_width	petal_length	petal_width
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2

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

.

.

$$x'^s \in \mathbb{R}^4$$



Supervised vs. Unsupervised

Supervised learning: the data set contains both **features** and **labels**.

Unsupervised learning: the data set contains only **features** and **no label**.

The model infers natural structure in the data without using labels.

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



Supervised vs. Unsupervised

Some Examples:

- 1. **Customer segment**: cluster your clients among different groups of customers who share common buying patterns. *unsupervised*
- 2. **Object detection**: Is there a car in this picture? *supervised*

Supervised vs. Unsupervised

Supervised - Theory

Supervised learning: the data set contains both **features** and **labels**.

The model learns a function, f , that takes the features and outputs a prediction of the target label:

$$x^i \in \mathbb{R}^d \quad f(x^i) = \hat{y}^i, \text{ true label } y^i$$
$$f : \mathbb{R}^d \rightarrow \text{set of labels}$$

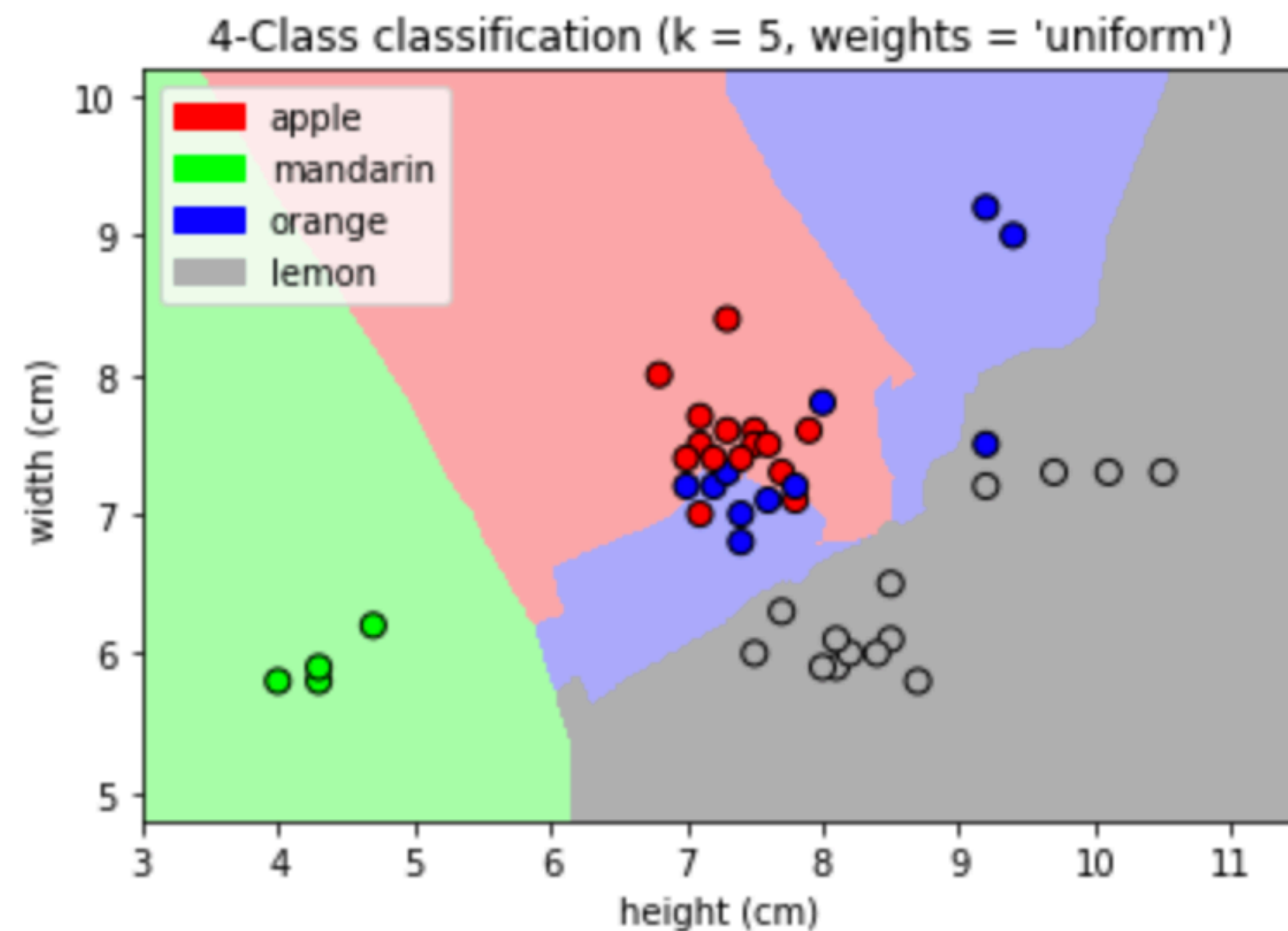
The samples in the data provide ground truths examples $\rightarrow$ it teaches f to predict *label* from *features*.

Tasks in Supervised learning

Regression versus Classification

Regression vs. Classification

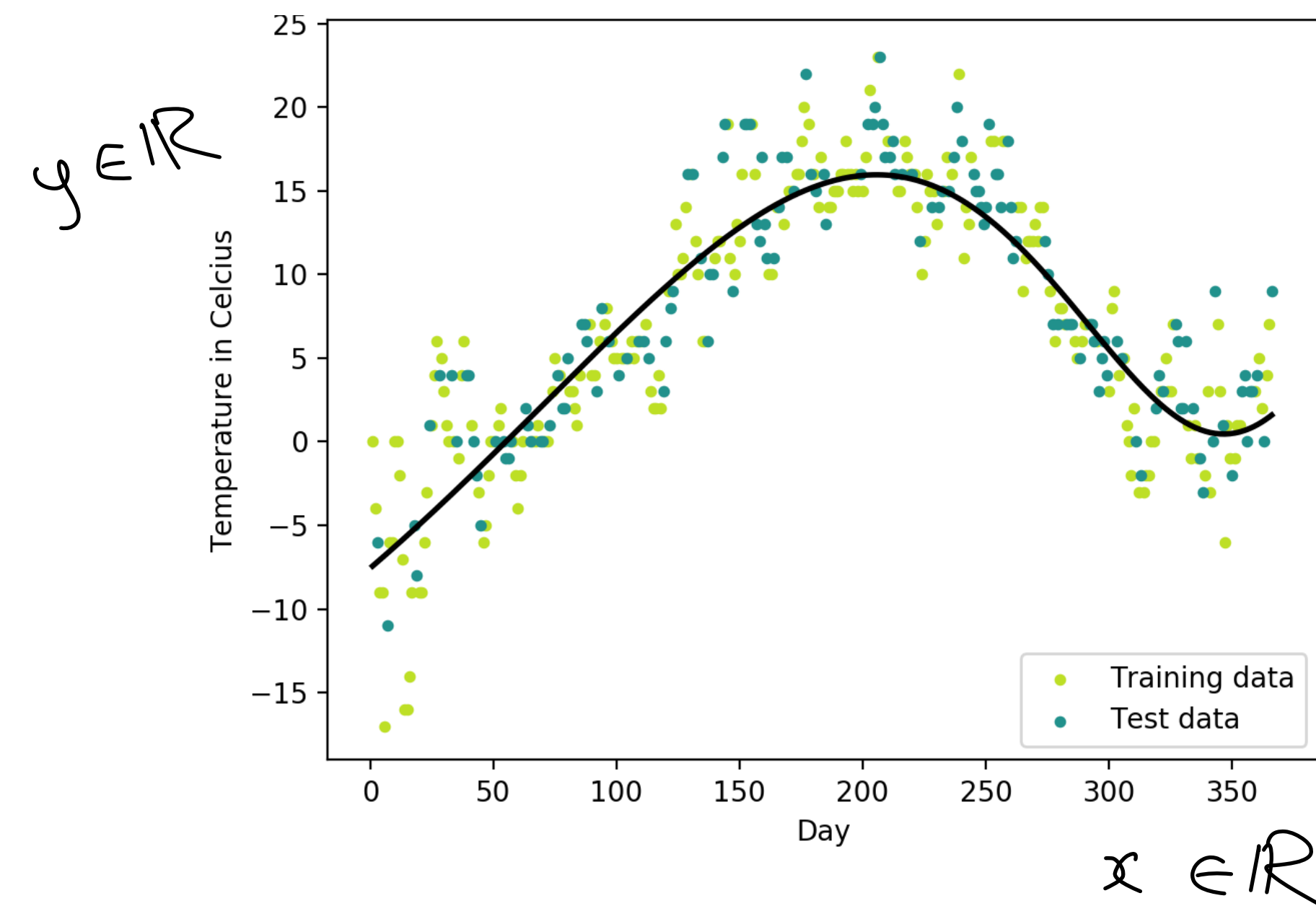
Classification



- Supervised Learning
- Output variable is discrete (categorical)

$$x \in \mathbb{R}^2 \text{ (width, height)}$$
$$y \in \{\text{apple}, \dots, \text{lemon}\}$$

Regression



- Supervised Learning
- Output variable is continuous

Supervised learning

Regression versus classification theory

The model learns a function, f , that takes the features and outputs a prediction of the target label:

$$f(\text{features}) = \hat{\text{label}}$$

where $\hat{\text{label}}$ is the prediction of label .

classification

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

regression

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

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

$\{0, 1\}$

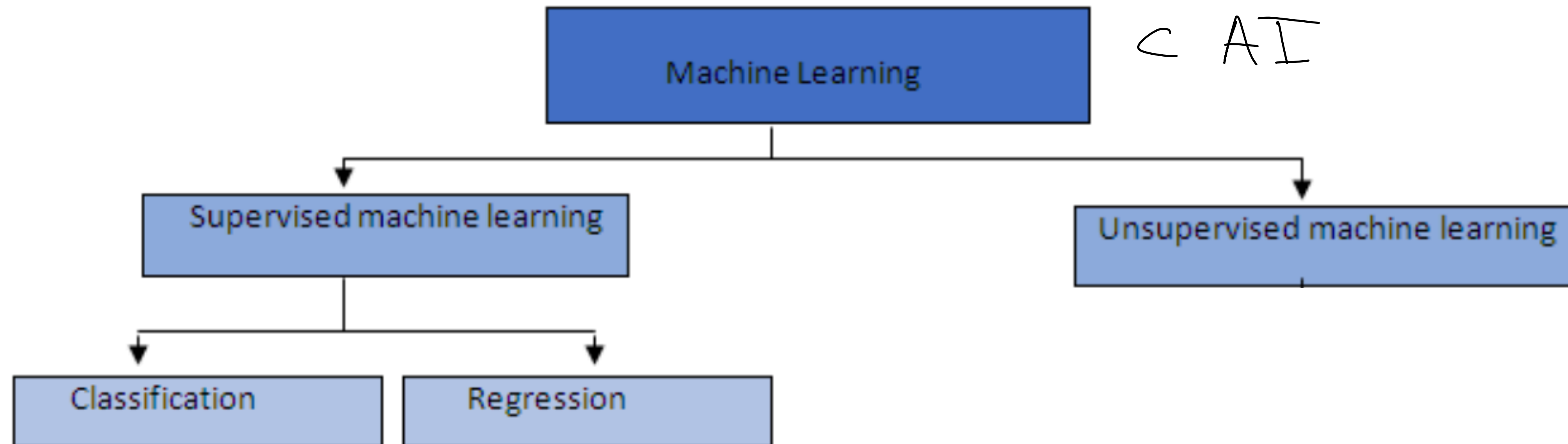
$p \in \mathbb{R}$

$y \in \mathbb{R}$

$\{low, medium, high\}$

Regression vs. Classification

Summary



Linear regression

Linear Regression - your exercise on python

■ Datasets:

- 1) CO2 Emissions of Cars , *weight of the cars*
- 2) Solar Power Generation , *temperature & radiation*

■ Goals:

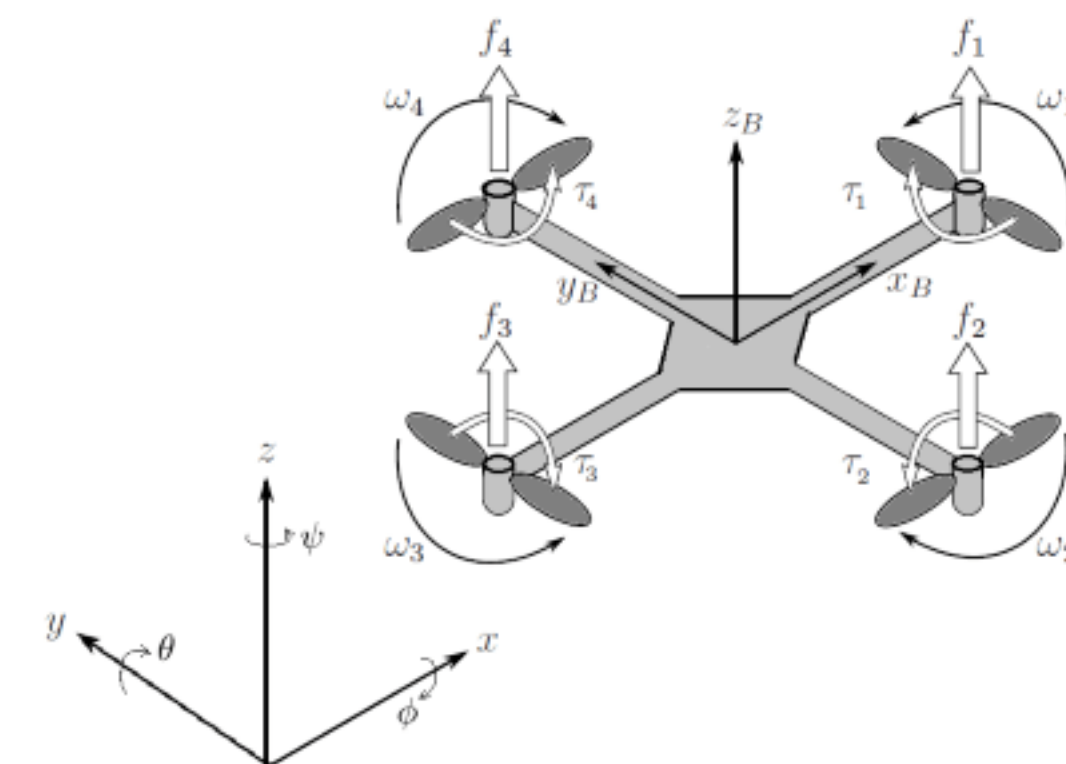
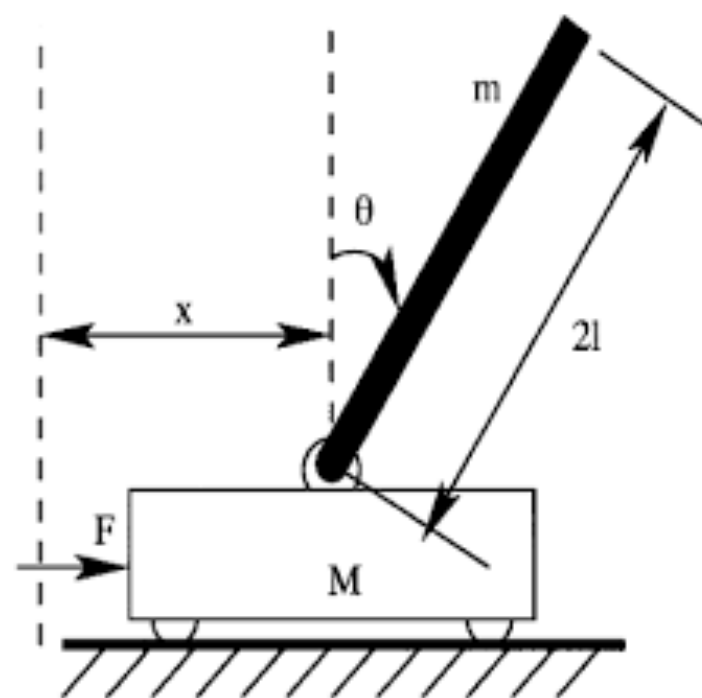
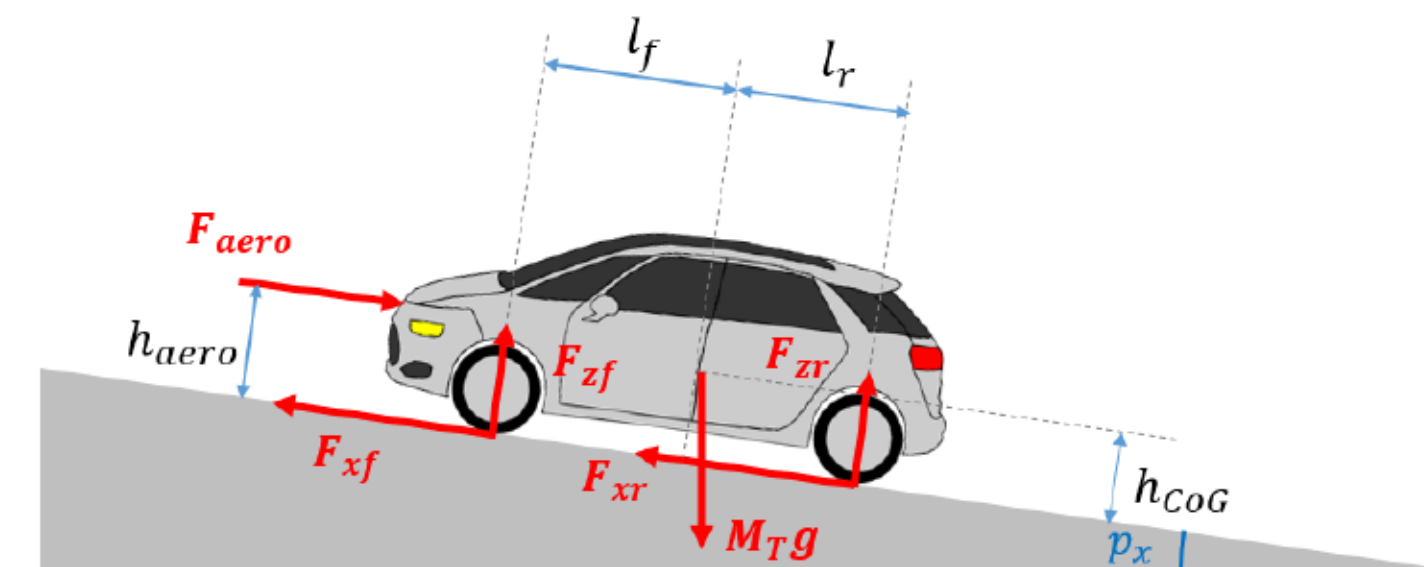
- 1) Predict the emissions of a car as a function of its weight
- 2) Predict the power generated by a solar power plant



Motivation for linear regression

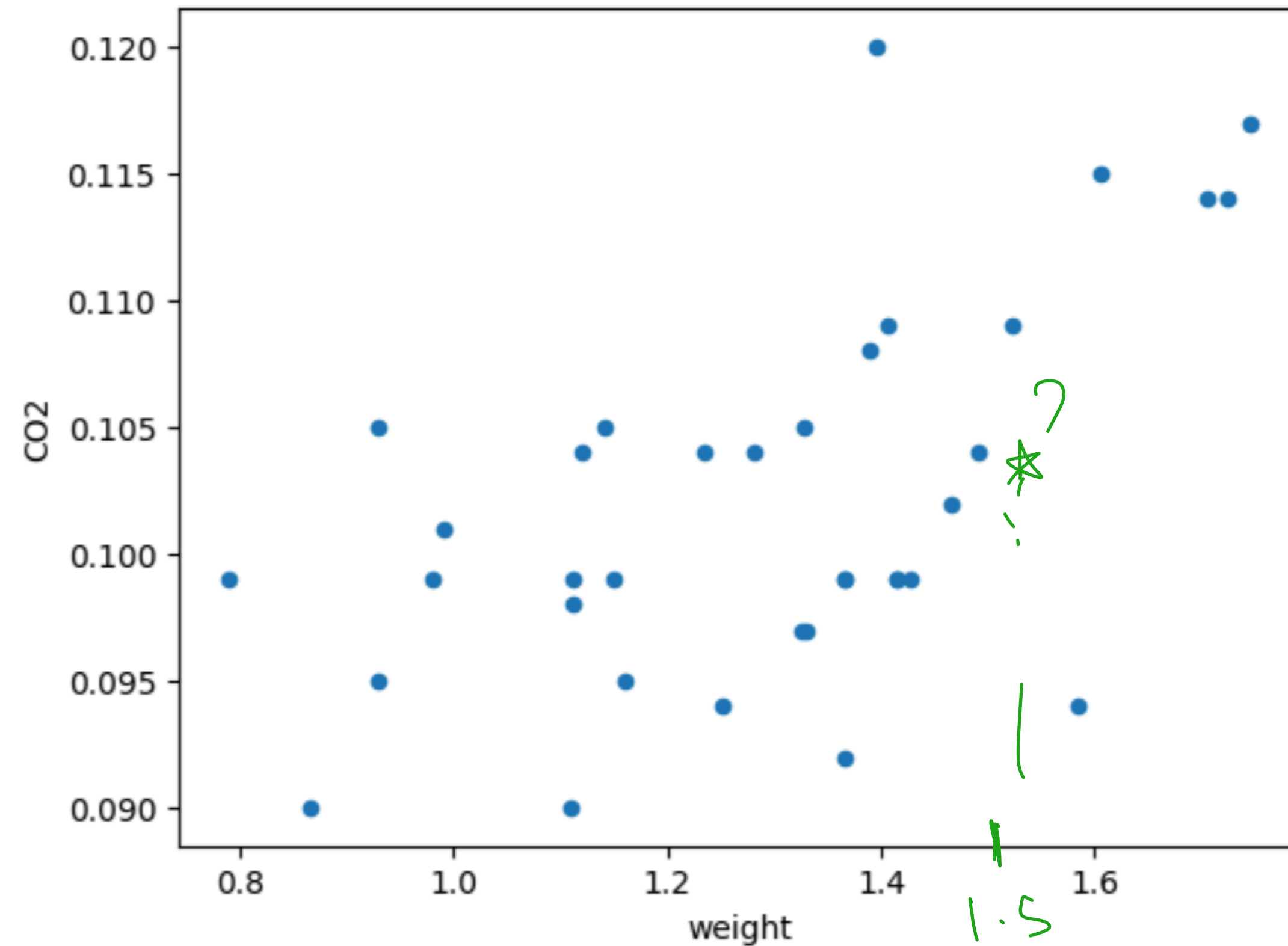
Linearity is a property of spaces and functions widely assumed and used in engineering

Examples of linear dynamical models used in IGM...



Linear Regression

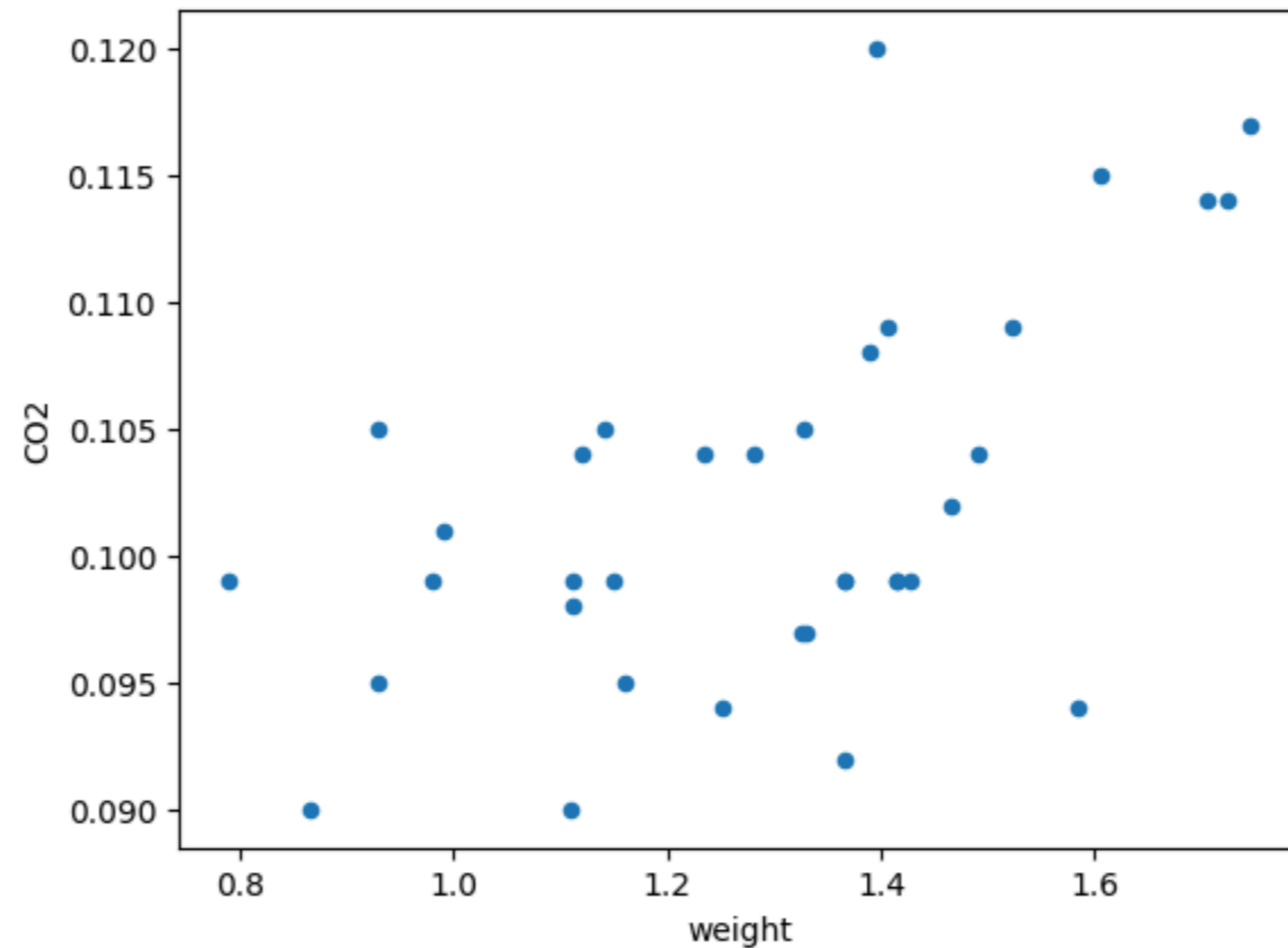
Introduction



- Problem:
 - Predict CO2 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

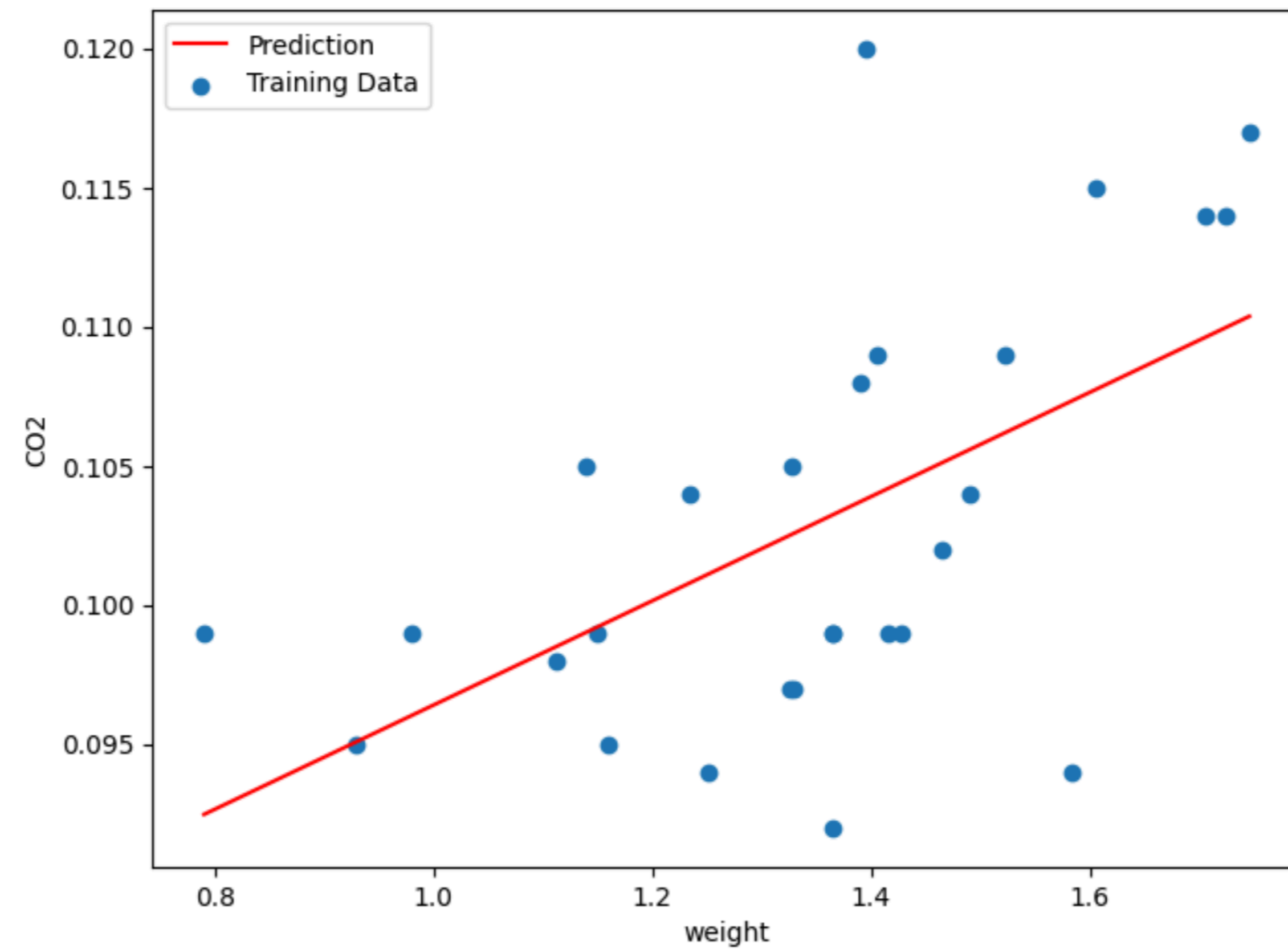
Introduction



- 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..

Parametric approach - Linear Regression

Introduction



Linear Regression

Abstraction

Problem: Fitting the data with a linear function. $\{ (x^i, y^i) \}_{i=1}^N$, $y^i \in \mathbb{R}^m$, $x^i \in \mathbb{R}^d$

Hypothesis: the relationship between the input and output data is linear.

It fits a line of slope w_1 and of intercept b such that for any data x^i , the prediction $\hat{y}^i$ is:

$$\hat{y}^i = wx^i + b \quad x^i \in \mathbb{R}, \quad y^i \in \mathbb{R}$$

Example: the linear regression for the car example..

Linear Regression

Multiple linear regression

When there are multiple features, the same principles apply

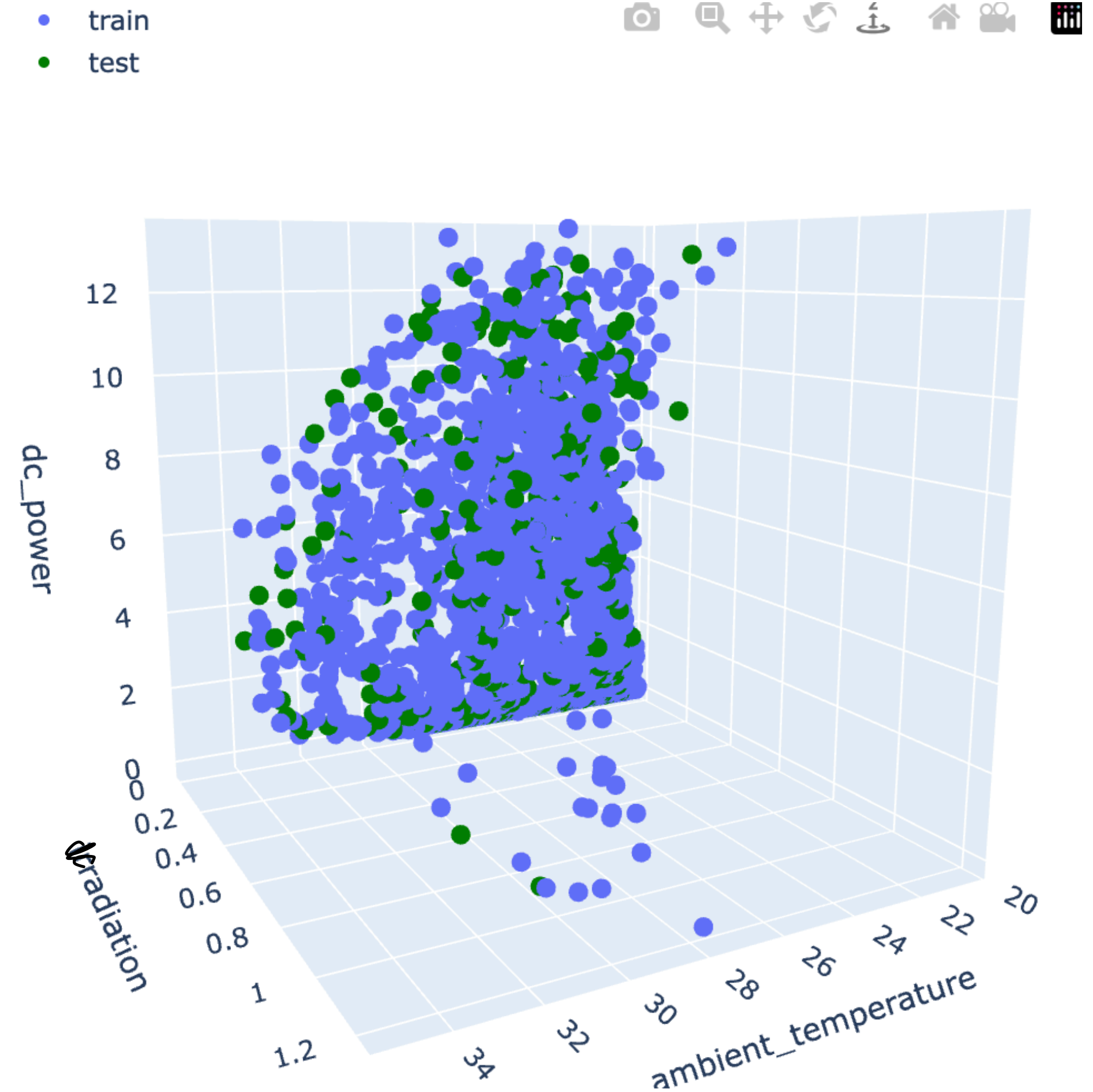
Goal: predict solar power generation as a function of measurements of irradiation (sunlight intensity) and ambient temperature.

Why? Optimize performance of a solar power plant

What are the independent variables?

What is the dependent variable/label?

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



Linear Regression

Multiple linear regression

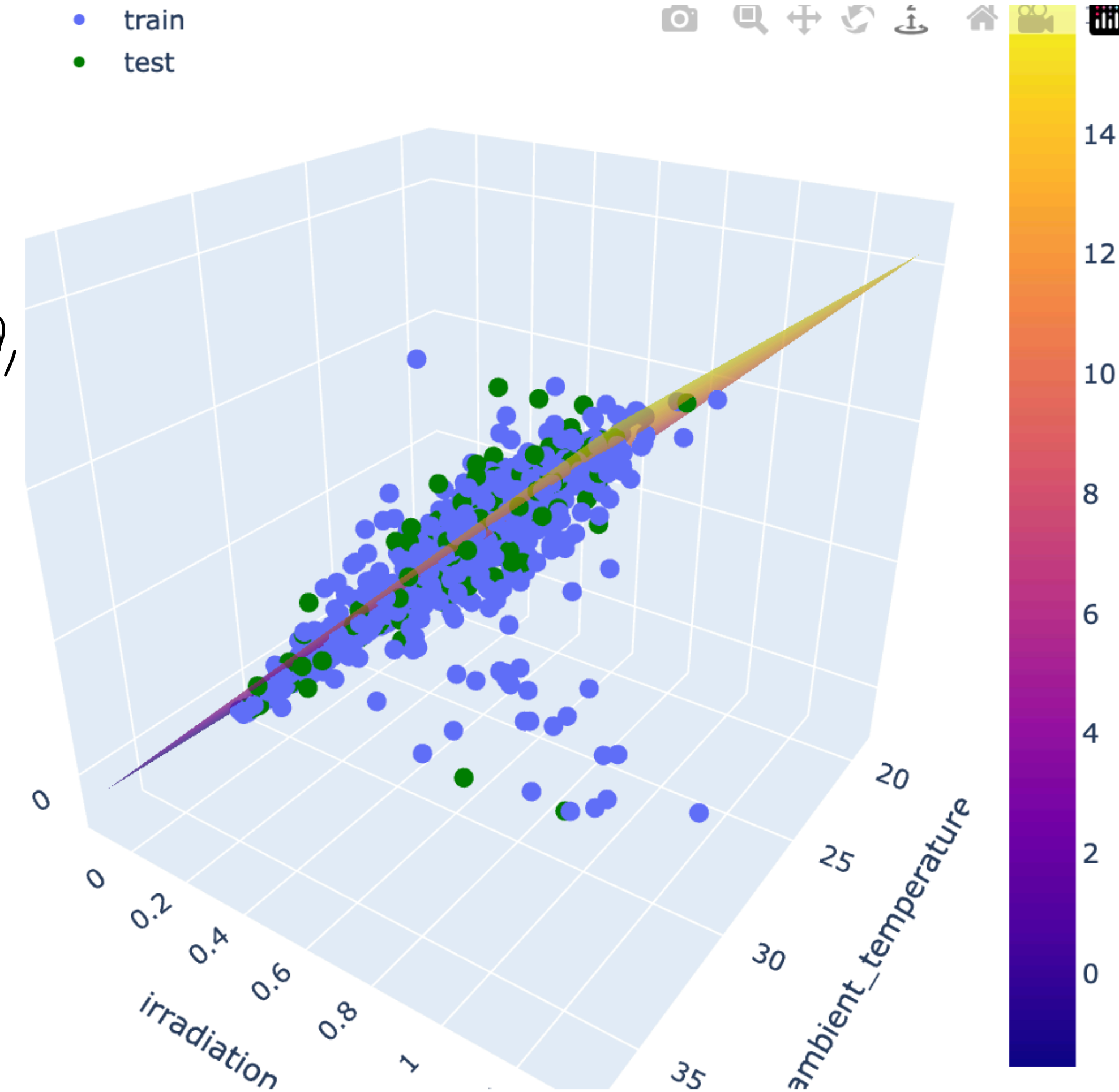
When there are multiple features, the same principles apply

The model parameters are combined in the **model parameter** w ,

$$\hat{y}^i = b + w_1 x_1^i + w_2 x_2^i + \dots + w_d x_d^i$$

where:

- w is the model's parameter vector ($w = [b, w_1, w_2, \dots, w_d]^T$)
- x is the instance's feature vector ($x = [1, x_1, x_2, \dots, x_d]^T$)



$h: \mathbb{R}^n \rightarrow \mathbb{R}^m$ is linear if $\forall \alpha, \beta \in \mathbb{R}, a, b \in \mathbb{R}^n$

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

$h: \mathbb{R}^n \rightarrow \mathbb{R}^m$ is affine if $\exists c \in \mathbb{R}$ s.t. $h - c =: \tilde{h}$ is linear.

example $h: \mathbb{R}^d \rightarrow \mathbb{R}$, given by $h(x) = b + w_1 x_1 + w_2 x_2 + \dots + w_d x_d$ is affine...

Let's consider $\tilde{h} = h - b =: \tilde{h}$ $\tilde{h}(x) = w_1 x_1 + w_2 x_2 + \dots + w_d x_d = \omega^T x$

$$\tilde{h}(\alpha a + \beta b) = \omega^T (\alpha a + \beta b) = \alpha \omega^T a + \beta \omega^T b = \alpha \tilde{h}(a) + \beta \tilde{h}(b)$$

Since $\tilde{h}$ is linear, it follows that h is affine.

Example: linear functions, affine functions

Let $g : \mathbb{R}^2 \rightarrow \mathbb{R}$, with $g(w) = w_1 \sin x + w_2 \cos x$

Is this function linear ?

yes, it is linear .

(but as a function of x , it is not)

Theory - linear predictors

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

use this data to "learn" "good" $w_1, w_2, \dots, w_d, b$

Linear predictor $f_w(x) = b + w_1x_1 + \dots + w_dx_d = w^T x$

with $w = [b, w_1, \dots, w_d]^T, x = [1, x_1, \dots, x_d]^T$

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

- 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

Exercise hours and plans

This week and next week

- Exercise hour on Wednesday 11.09
 - Bring your laptop - you will get familiar with python
- No lecture on Monday 16.09 but there will be exercise hour on Wednesday 18.09
- Exercise hour on Wednesday 18.09
 - Go through the notes online and the exercises there
 - Get further familiar with python