INTRODUCTION TO MACHINE LEARNING

JESSE KRIJTHE
CSE2510 - MACHINE LEARNING

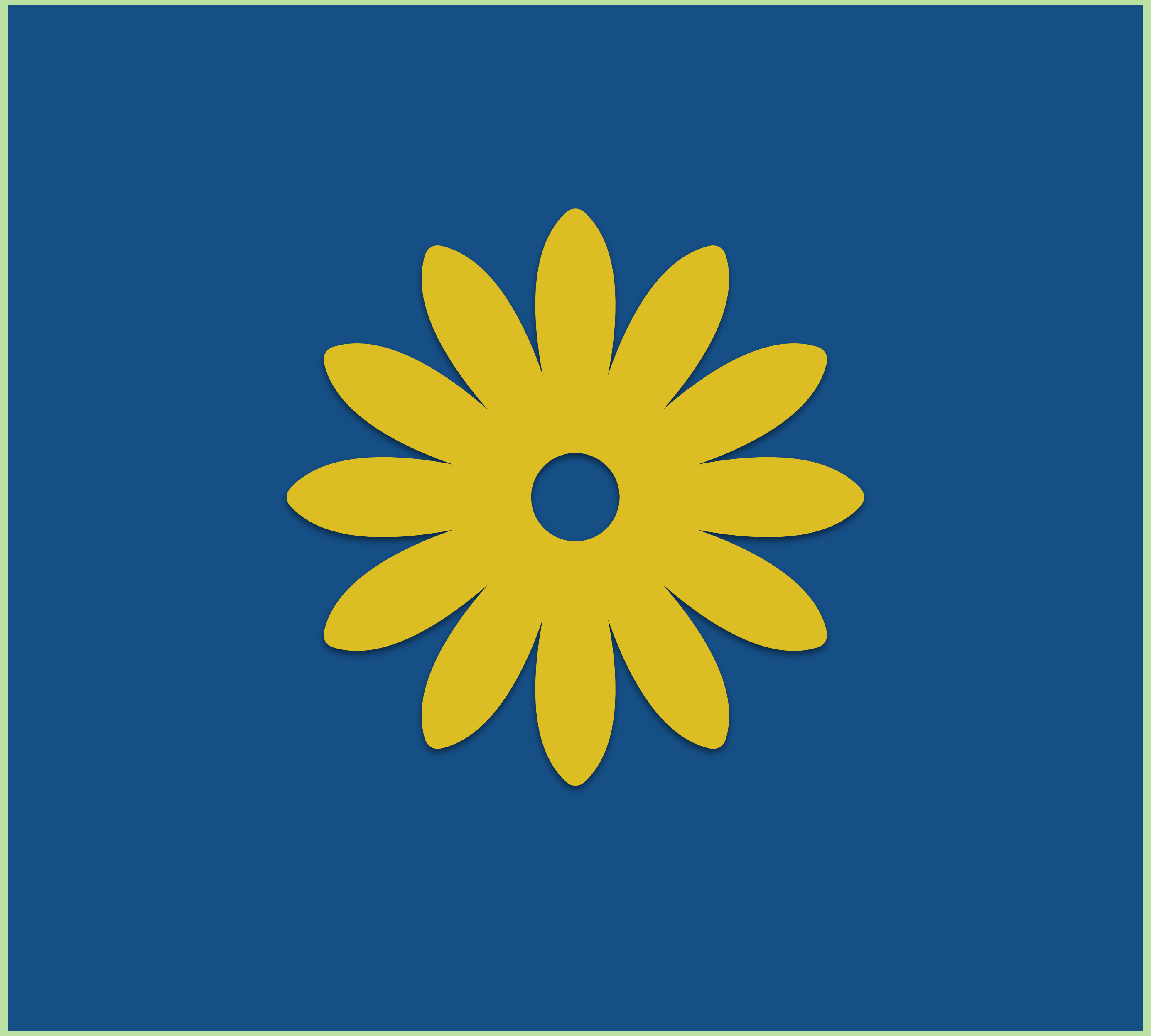
TU Delft
Delft University of Technology

LET'S PLAY

Human-generated or Computer-generated?

EXAMPLE

HUMAN



MACHINE

Easier Task?

Discussion

Suppose we want to build a system to solve this task (distinguishing human-generated from machine-generated paintings, or **distinguishing apples and pears**).

Discuss (in pairs):

- What would you need to construct such a system?
  - What physical tools would you need?
  - What additional information do you need to gather (talk to a biologist? converse with a farmer? use a search engine?)
  - How do you implement the decision making component of the system?
  - What steps would you need to take to develop such a system?

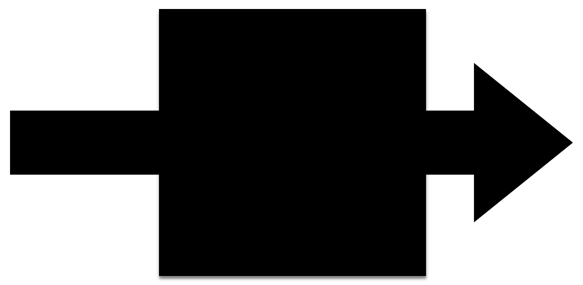
Intuition ML

- Task can seem very simple, but describing rules that a machine can use is very hard (think of all the exceptions, etc.)
- Idea: create a rule that constructs these rules by looking at many examples.
- “Finding patterns” in the data

Machine Learning

Study of algorithms that use example objects (data), to learn mappings from objects to desired outcomes

INPUT

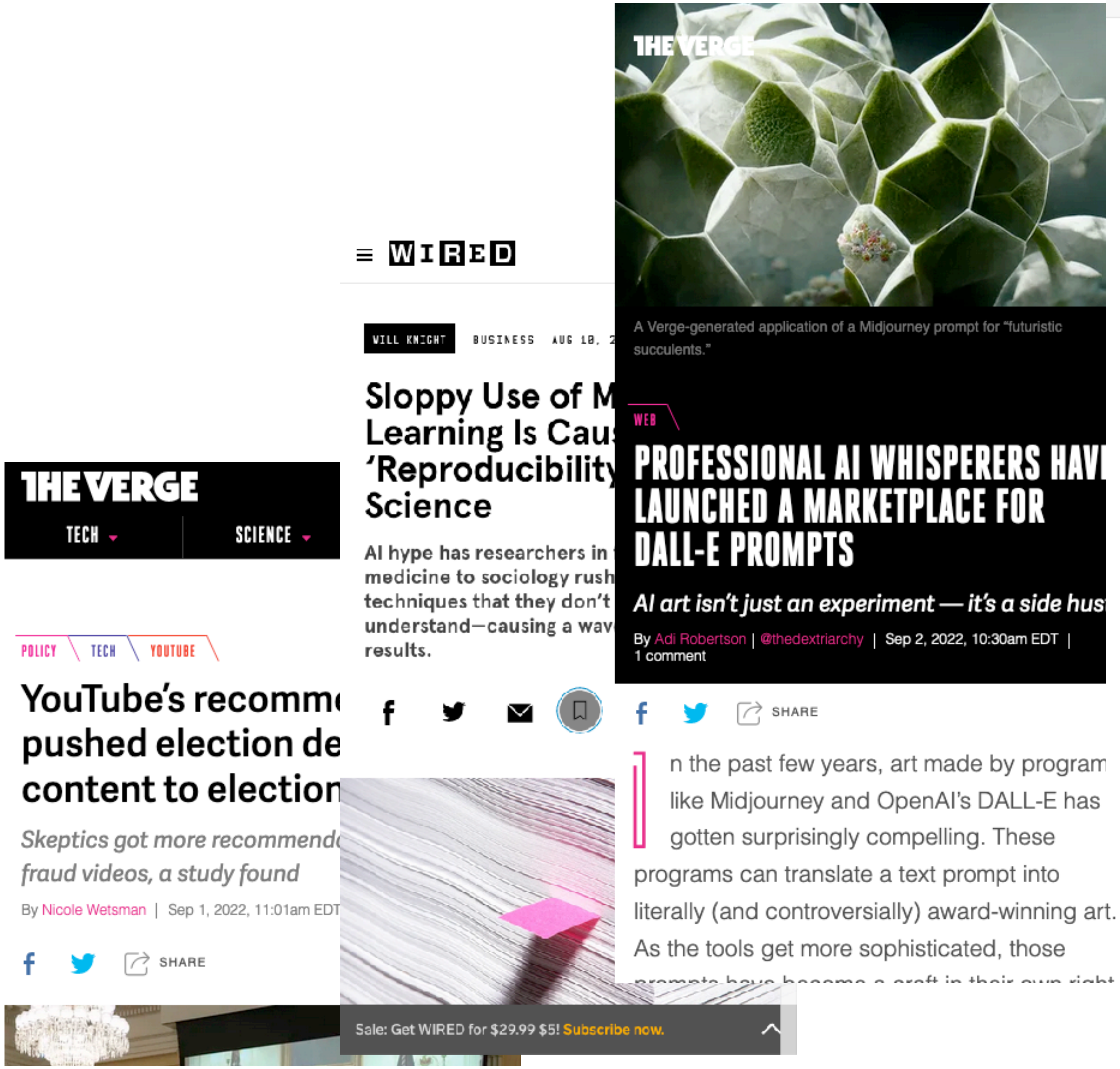


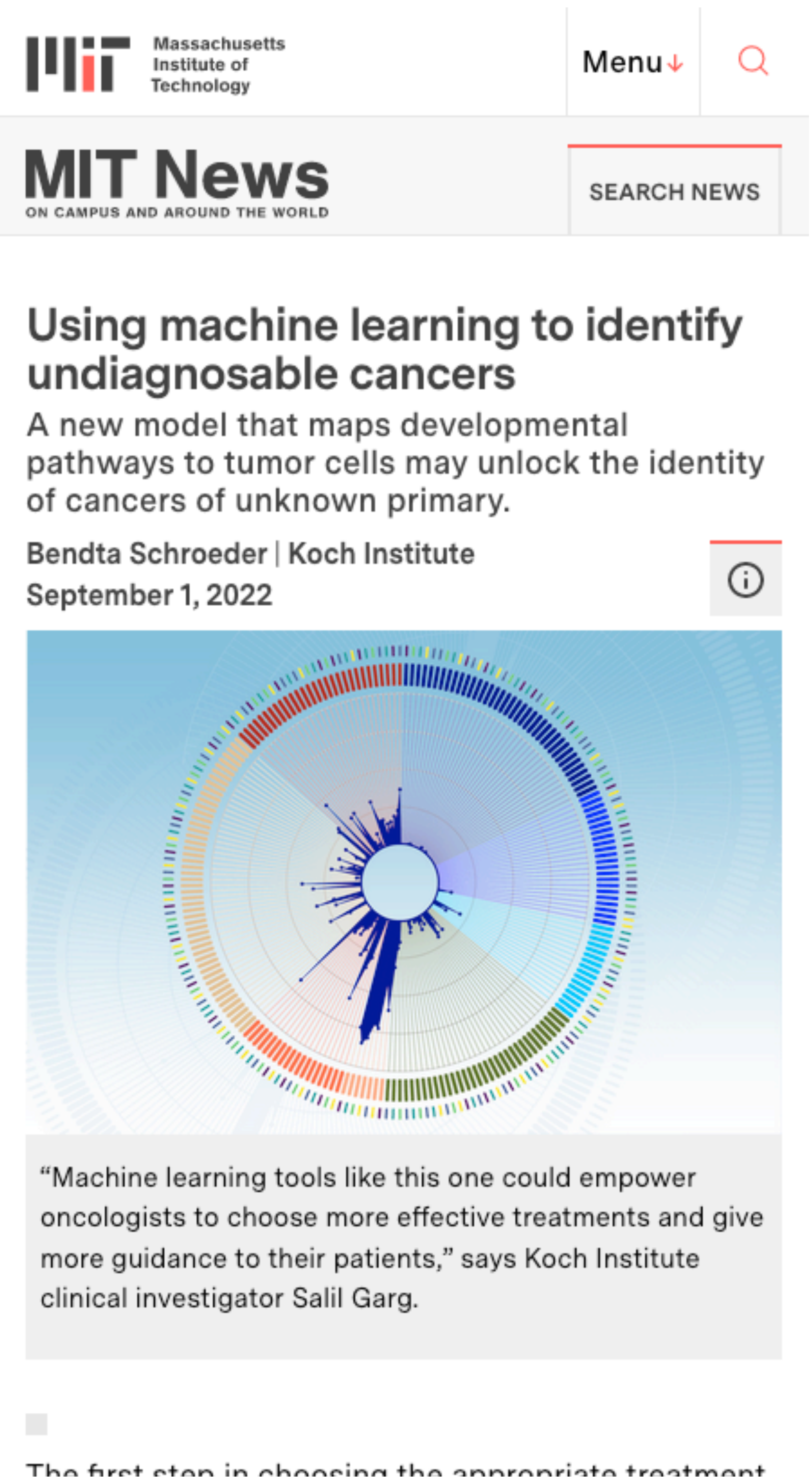
"MODEL"

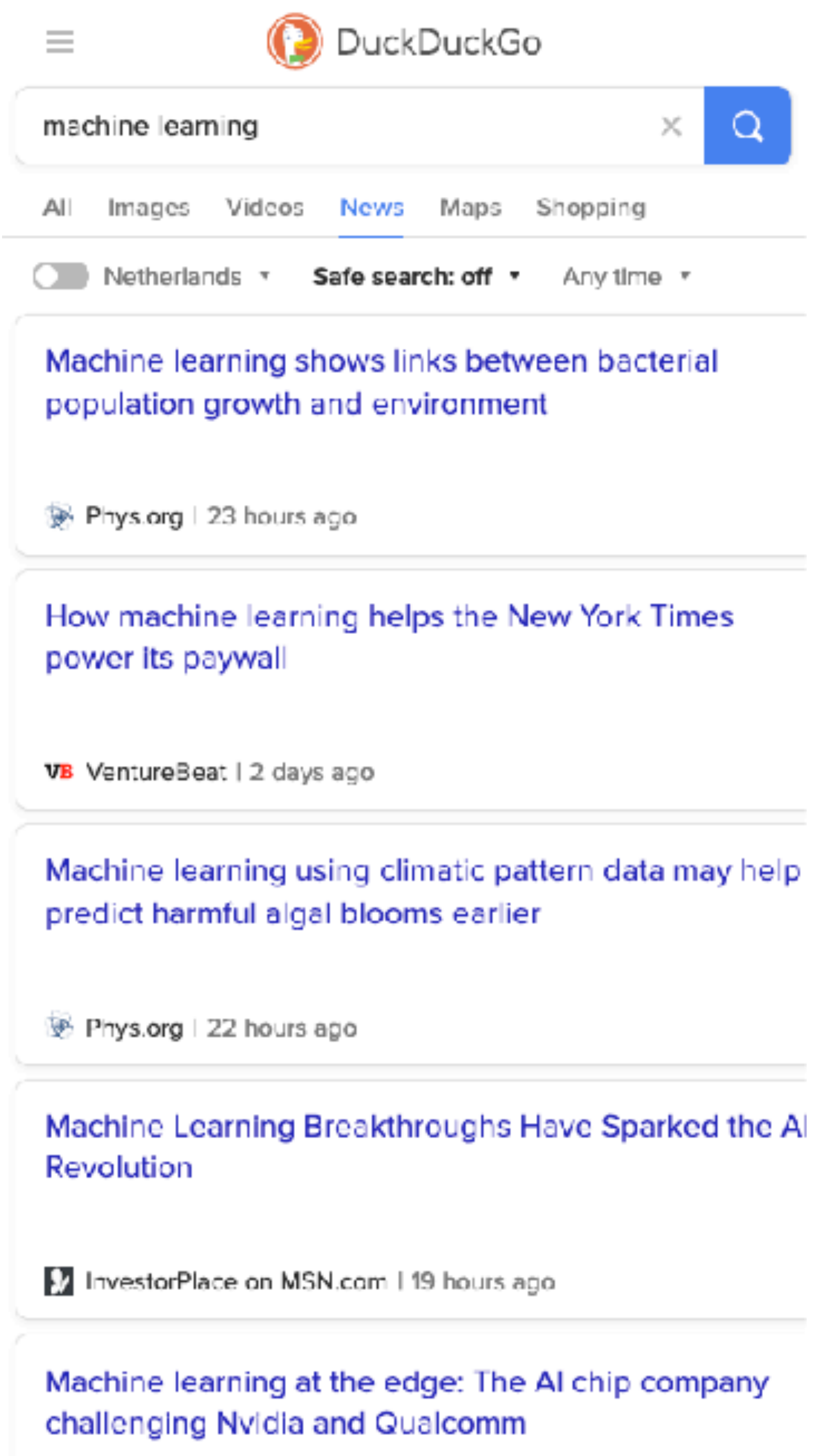
PREDICTION

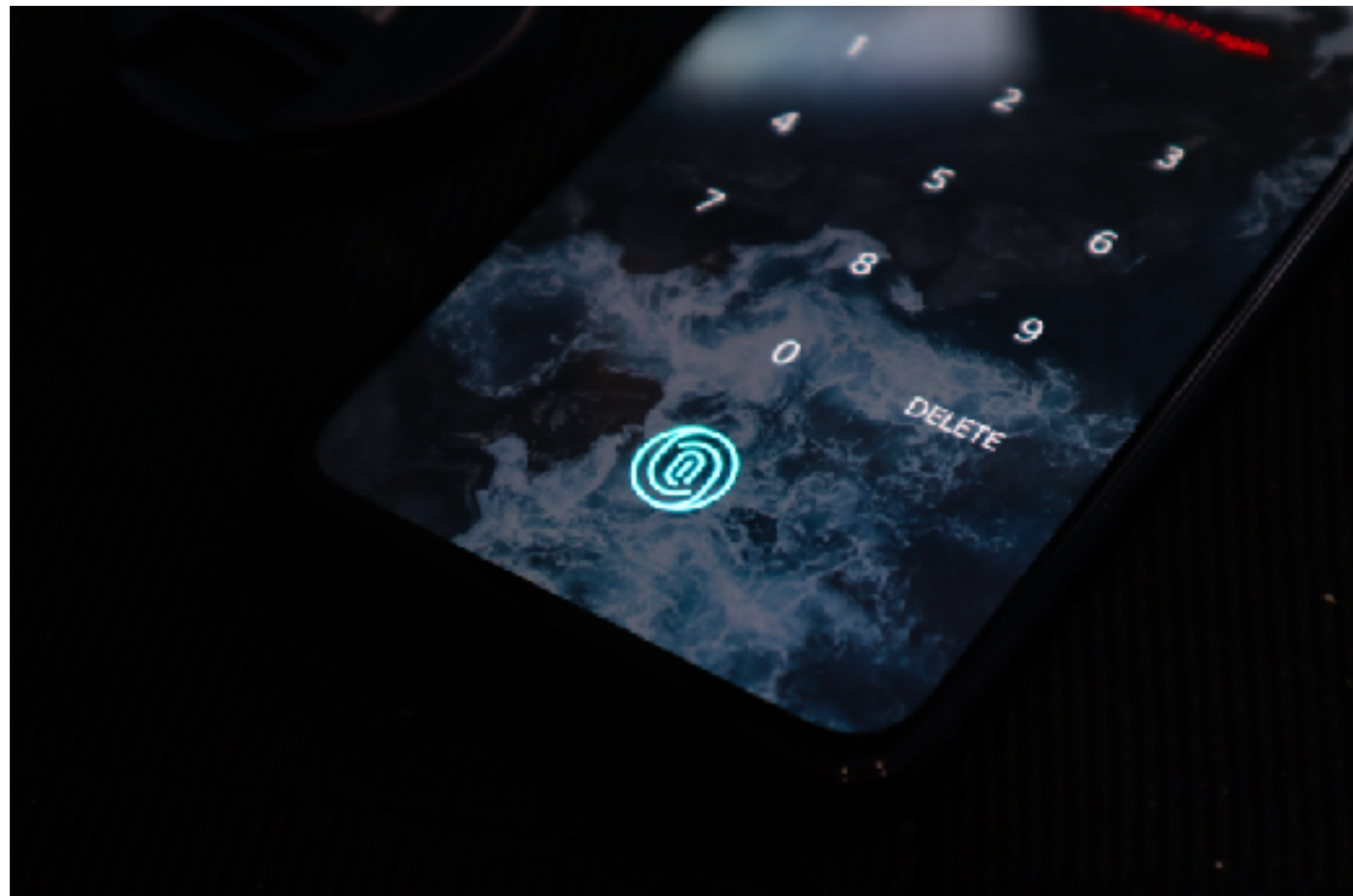
INPUT	PREDICTION
Painting	Human/Machine Made
Fruit	Apple/Pear
Photo	Objects in the photo
Opinion poll	Outcome of an election
Patient	Diagnosis/Optimal treatment
Facebook user	Advertisement
Google search	Relevant website
TikTok session	Next video
Traffic scene	Brake/Don't Brake
Chess board	Best Move

ML is Everywhere?



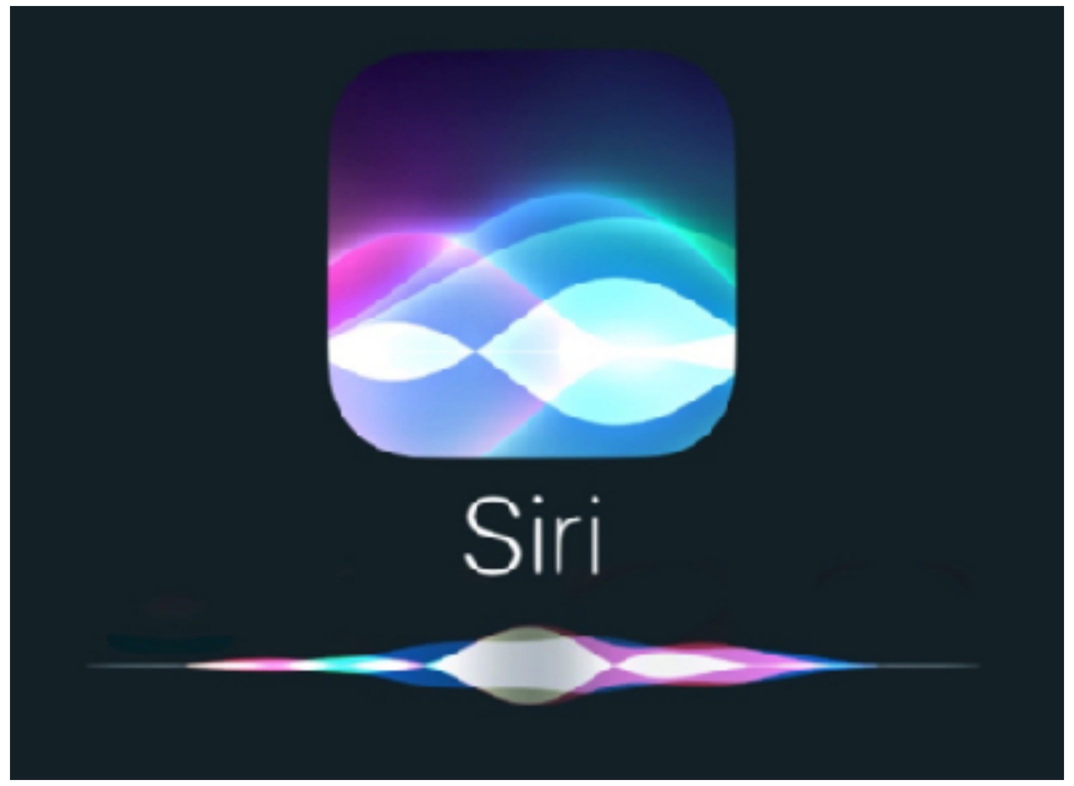


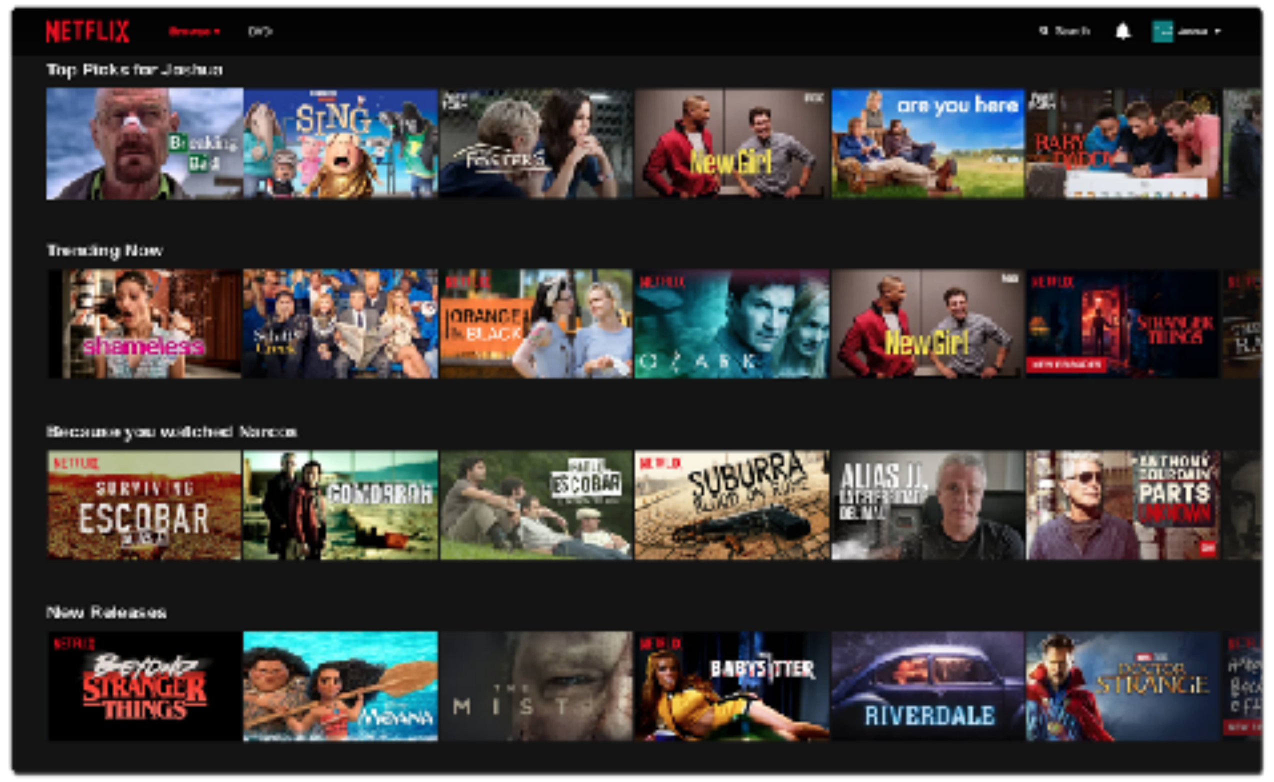




Some Examples

- Siri / Alexa / Google Assistant
- Recommending movies
- Targeting advertisements
- Game playing
    - Chess, Go
    - Videogames (Atari, DotA, Starcraft, etc.)
- Protein structure prediction
- Image generation





DeepMind's Deep Q-Network used to play Atari's Breakout game

Application Types

- Artificial Intelligence / Signal Processing / Multimedia
- Data Analysis
- Scientific ML

Learning Goals

After successfully completing this course, you are able to:

- explain the basic concepts and algorithms of machine learning and their underlying statistical concepts.
- implement, apply and evaluate basic ML algorithms in Python.
- explain the concept of and identify (implicit) bias in data and ML algorithms. (Focus in week 5)

THE COURSE

People

Lecturers

Jesse Krijthe

Gosia Migut

David Tax

TAs

Aleksander Buszydlik
Damla Ortaç
Karol Dobiczek
Aleksandra Andrasz
Simran Karnani
Bianca Cosma

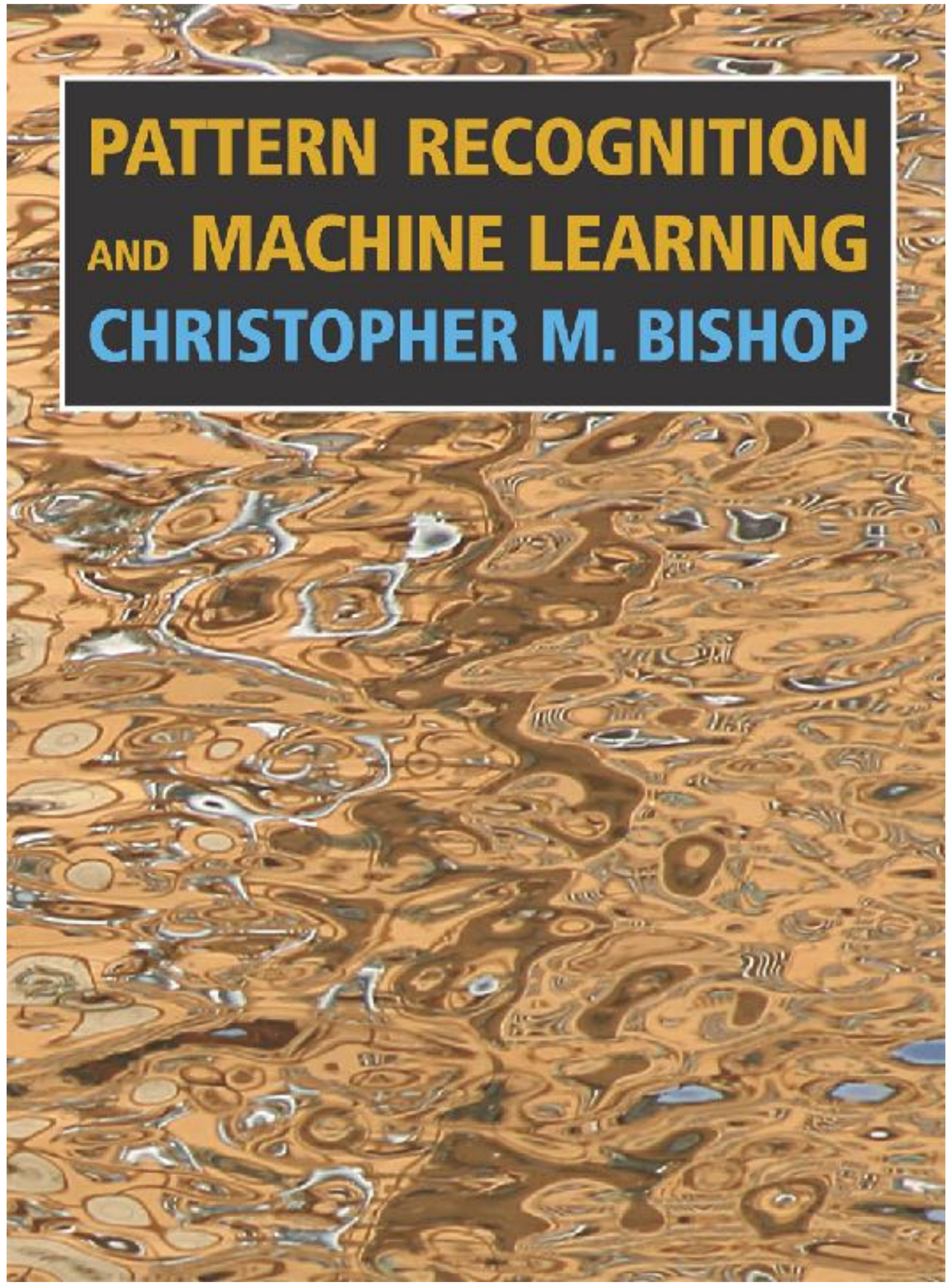
Aditi Rawat
Clio Feng
Akash Amalan
Chao Chen
Yue Chen

Structure

Carefully read the full instructions on Brightspace

- **Read the reading material** (mostly based on Bishop's Pattern Recognition & Machine Learning)
- Attend lectures and revise lecture notes
- Practice weekly theory questions in Weblab (https://weblab.tudelft.nl/cse2510/2022-2023)
- Practice implementation questions in lab notebooks
- Engage with colleagues on https://answers.ewi.tudelft.nl/
- *Exam*: mix between theory and implementation questions
- *Bonus assignment (week 8):* extra 0.5 for the final grade
- To prepare for exam: Weblab practice exams

Book!



Freely available from Brightspace and the author's website.

Some weeks we supplement the book with other reading materials.

The math can be challenging, but this is part of the learning goals.

Please use the reading materials!

Weekly Exercises (Weblab)

- https://weblab.tudelft.nl/cse2510/2022-2023/
- Should be available after the lecture
- Highly recommended to make them
- Helps you understand and practice with the material
- Many are similar to the style of the exam questions

Labs

- Each week, we provide an assignment in the form of a Python notebook that contains insight and implementation questions.
- Suggest you work on it in pairs
- TAs and lecturer are available to help you with questions
- We use Queue for management during the weekly scheduled lab sessions
- Answers will not be provided, so use the TAs to discuss exercises
- **One of the exam questions will come directly from the lab assignment**
- This week's lab (no scheduled session, so self-paced): focussed on familiarising yourself with (matrix computations in) Python

Bonus Assignment

- Published towards the end of the course
- Mostly work on this in week 8
- Teams of 2
- Hand in report + code
- Pass/Fail
- If pass & you have a passing grade on the exam -> +0.5 on your grade

Exam

- 3 hour Weblab exam on campus
- 100% of the final grade
- Note: make sure you have enrolled in Weblab before the exam (and check it is the correct edition of the course)
- Mix of theory and implementation questions
- Access to Python documentation
- No access to your notes
- See Practice Exams for examples

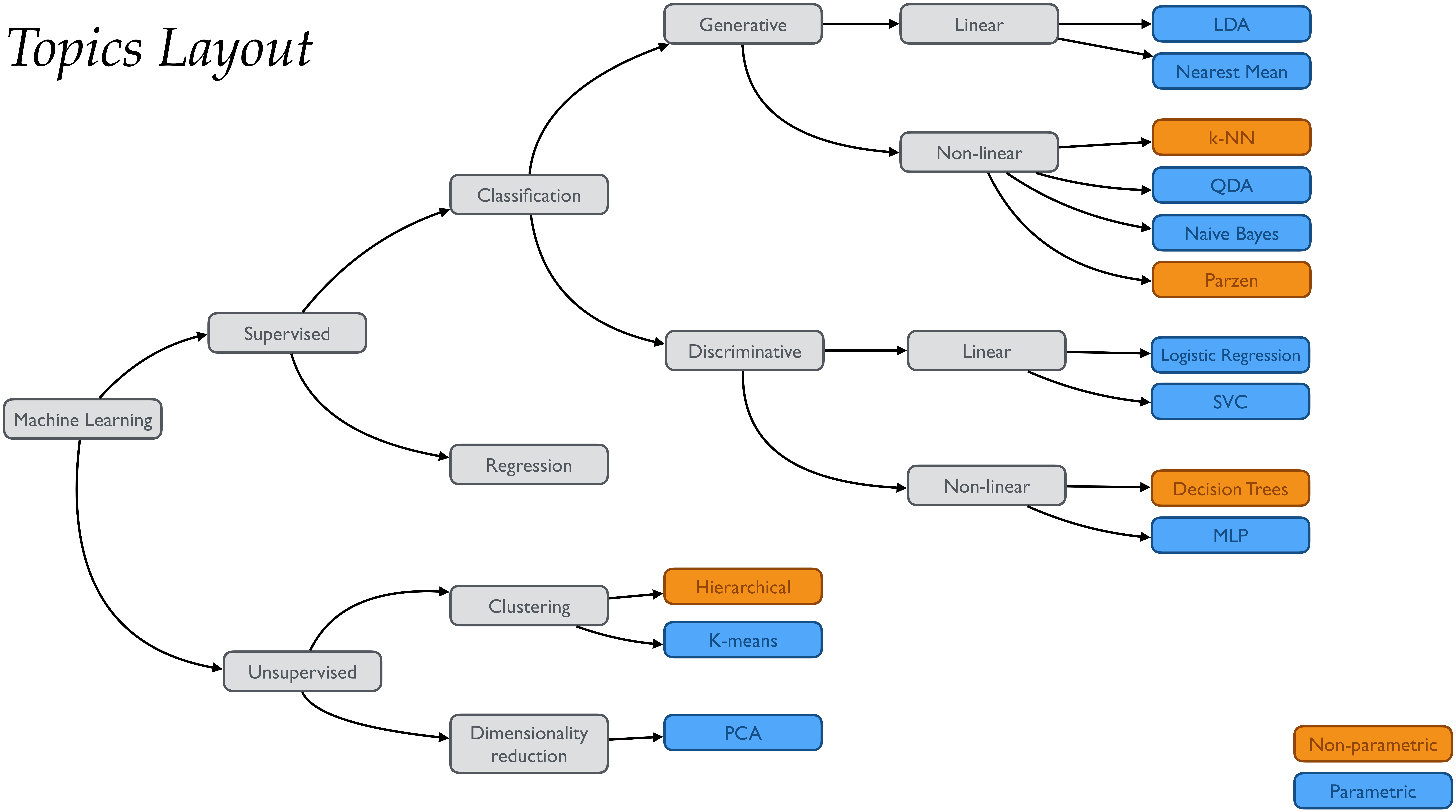
Questions/Help

1. Ask a colleague
2. Ask the TAs
3. Ask on answers platform https://answers.ewi.tudelft.nl/
4. If this failed and the question is unique to your situation:
   email us at ml-cs-ewi@tudelft.nl

Topics

Week	Topic	Lecturer
1	Introduction to ML	Jesse Krijthe & David Tax
2	Generative Parametric Models	David Tax
3	Non-parametric generative models	Gosia Migut & David Tax
4	Discriminative Linear Models	Jesse Krijthe
5	Responsible machine learning	Gosia Migut & Mark Theunissen
6	Discriminative Non-Linear Models	Jesse Krijthe
7	Unsupervised learning	Gosia Migut
8-9	Guest Lectures + Q&A	Multiple

Topics Layout



What we improved this year

- Weekly theory practice assignments in Weblab
- More consistency in the literature used
- More consistency in the mathematical notation used (work in progress)

Notes

- Math notation may seem opaque sometimes: start practicing now when things are still simple, or we'll regret it later.
- Focus of the course is on good understanding of the fundamentals, we build on this in later courses to work on more complex models and interesting applications.
- Let us know when you think we can improve something (preferably during the course, but also after).

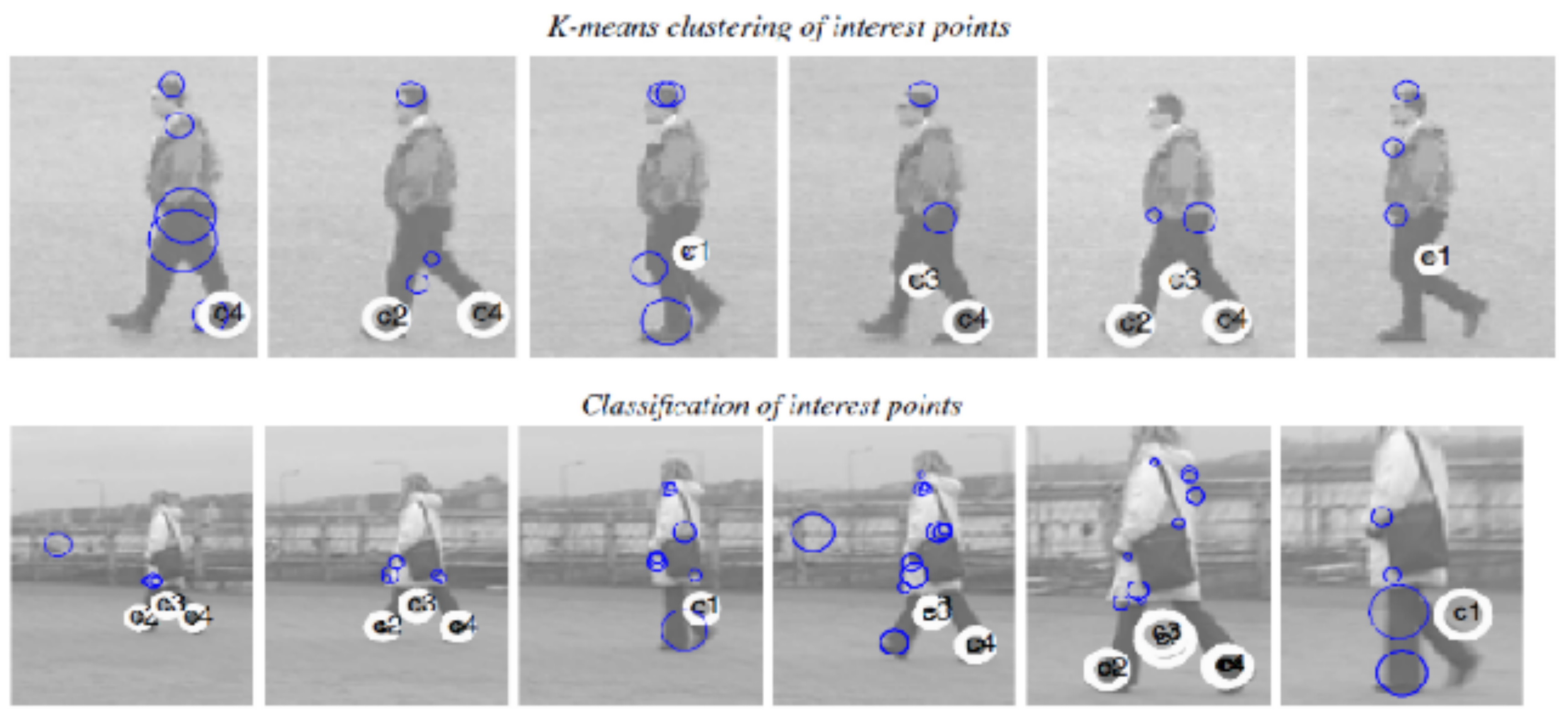
Use of ML in CSE2230 Multimedia Analysis

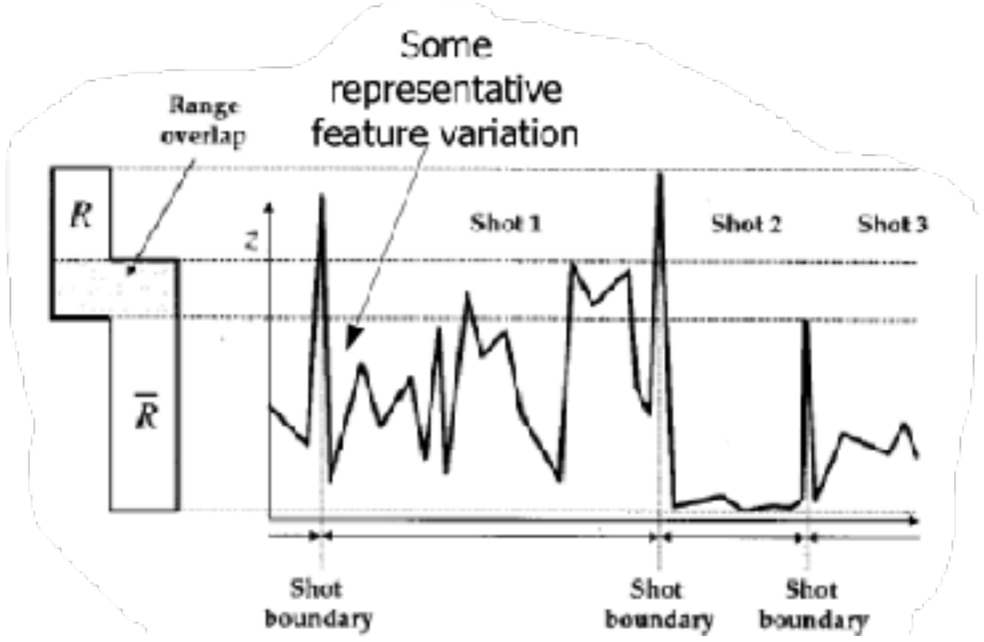
Multimedia content (text, images, speech, video) is an important source of knowledge that is useful to people.

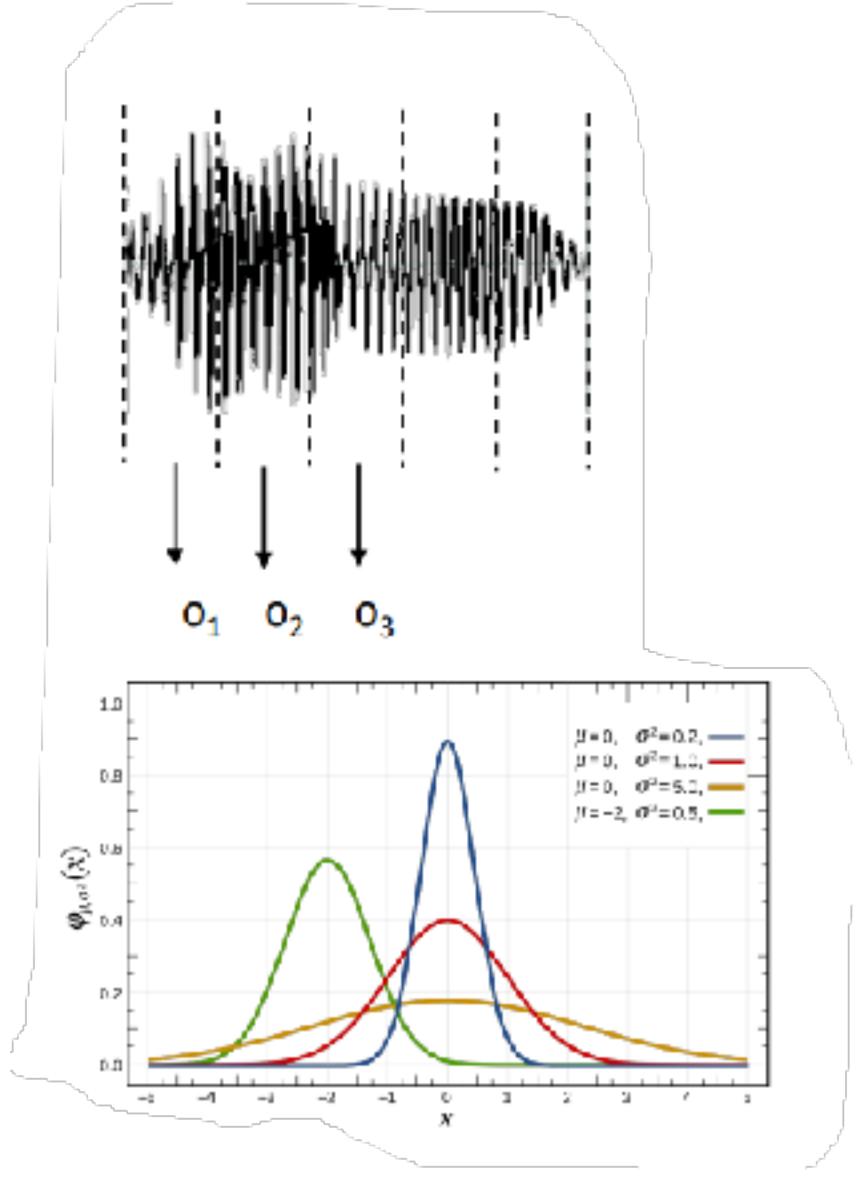
Building systems that allow people to make use of that knowledge, and evaluating such systems from both a **technological** and **human** point of view.

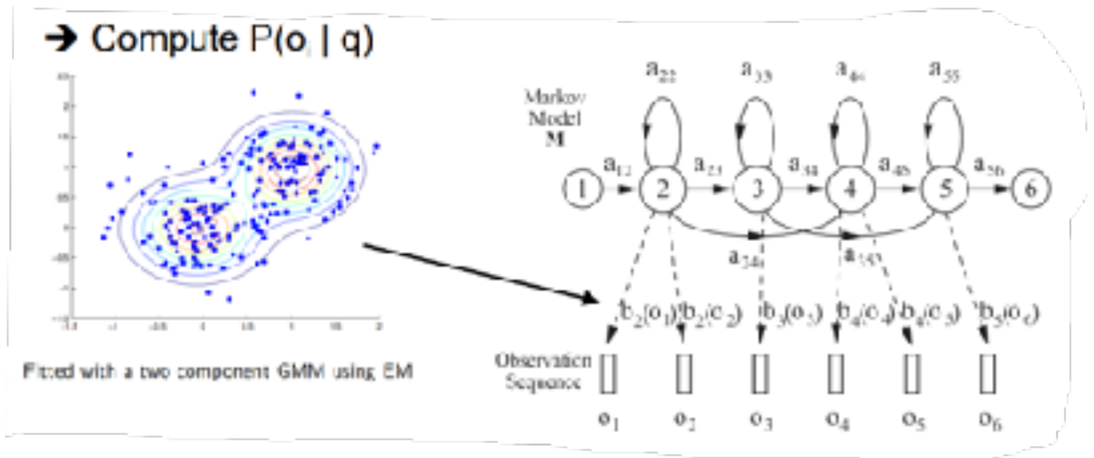
Machine Learning Topics covered:

- **K-means clustering** for e.g. image geo-location, visual words representation, audio-visual video segmentation,
- Gaussian Mixture Model for e.g. Acoustic Modelling
- **Naive Bayes Classifier** and Hidden Markov Models (not covered in ML) for e.g. Automatic Speech Recognition
- **Linear classifier** for e.g. video shot change detection,
- **k-nearest neighbour classifier** for e.g. phoneme recognition from Speech
- **Parameter training** for e.g. weight training for late multi-modal data fusion
- System Evaluation e.g. **accuracy, precision, recall**, f-measure, **ROC, AUC**… but also metrics for information retrieval systems Mean Average Precision (not covered in ML)









Slide courtesy of Hayley Hung

QUESTIONS?

Back to Machine Learning…

Example

What is Machine Learning?

Machine Learning tries to identify regularities in the world, by **learning** from examples (“data”). These regularities should **generalise**: work beyond the specific examples the model has seen before.

Why Machine Learning?

1. Many tasks are too complicated to explicitly encode by hand: learn a mapping from input to output using examples.

2. Learning about the world using data, to use objective evidence to base our decisions on.

Connections to statistics, signal processing, optimisation

Goal of Machine Learning

Model

(Mapping from inputs to outputs)

Expected Loss (Risk)

(On average, how bad is my model)

$$\mathbb{E}_O[L(g, O)]$$

Object

(Thing that I want to make a prediction for)

Loss function

(How much does a mistake "cost"?)

Goal of Machine Learning

Model

(Mapping from inputs to outputs)

Expected Loss (Risk)

(On average, how bad is my model)

$$\mathbb{E}_O[L(g, O)]$$

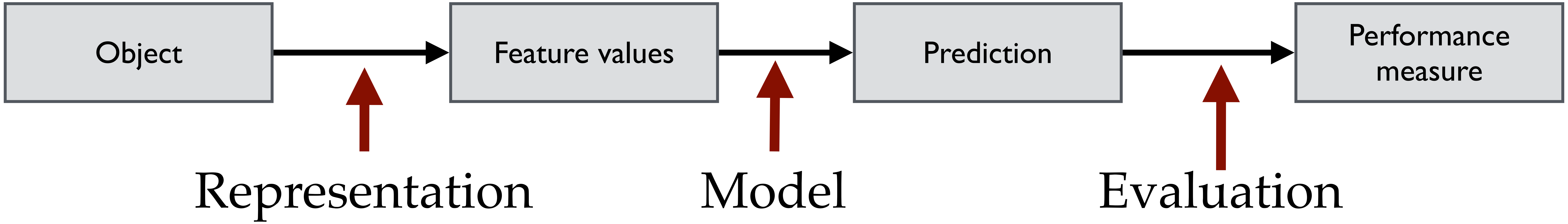
Object

(Thing that I want to make a prediction for)

Loss function

(How much does a mistake "cost"?)

Steps involved in a Machine Learning model



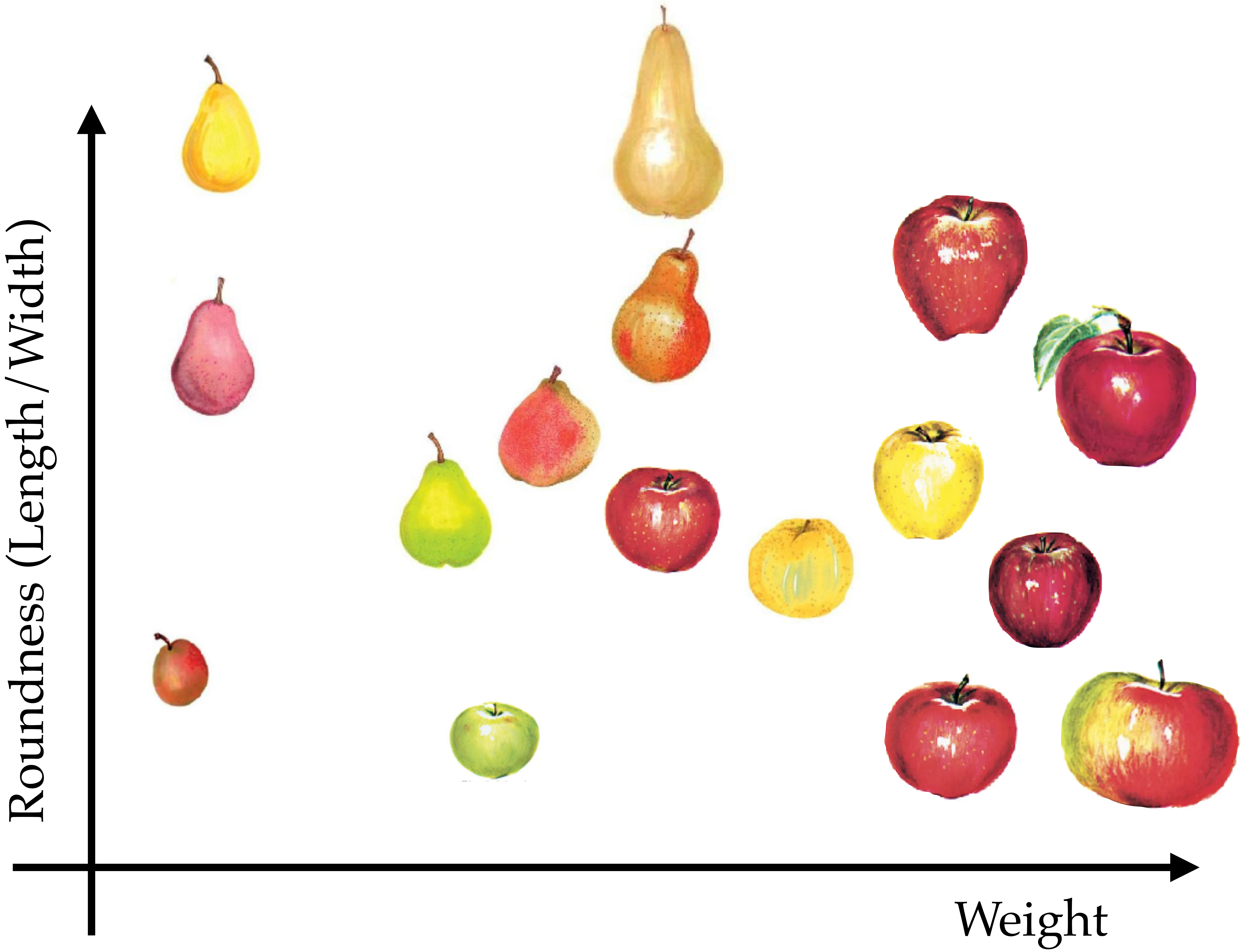
Representation -> Features

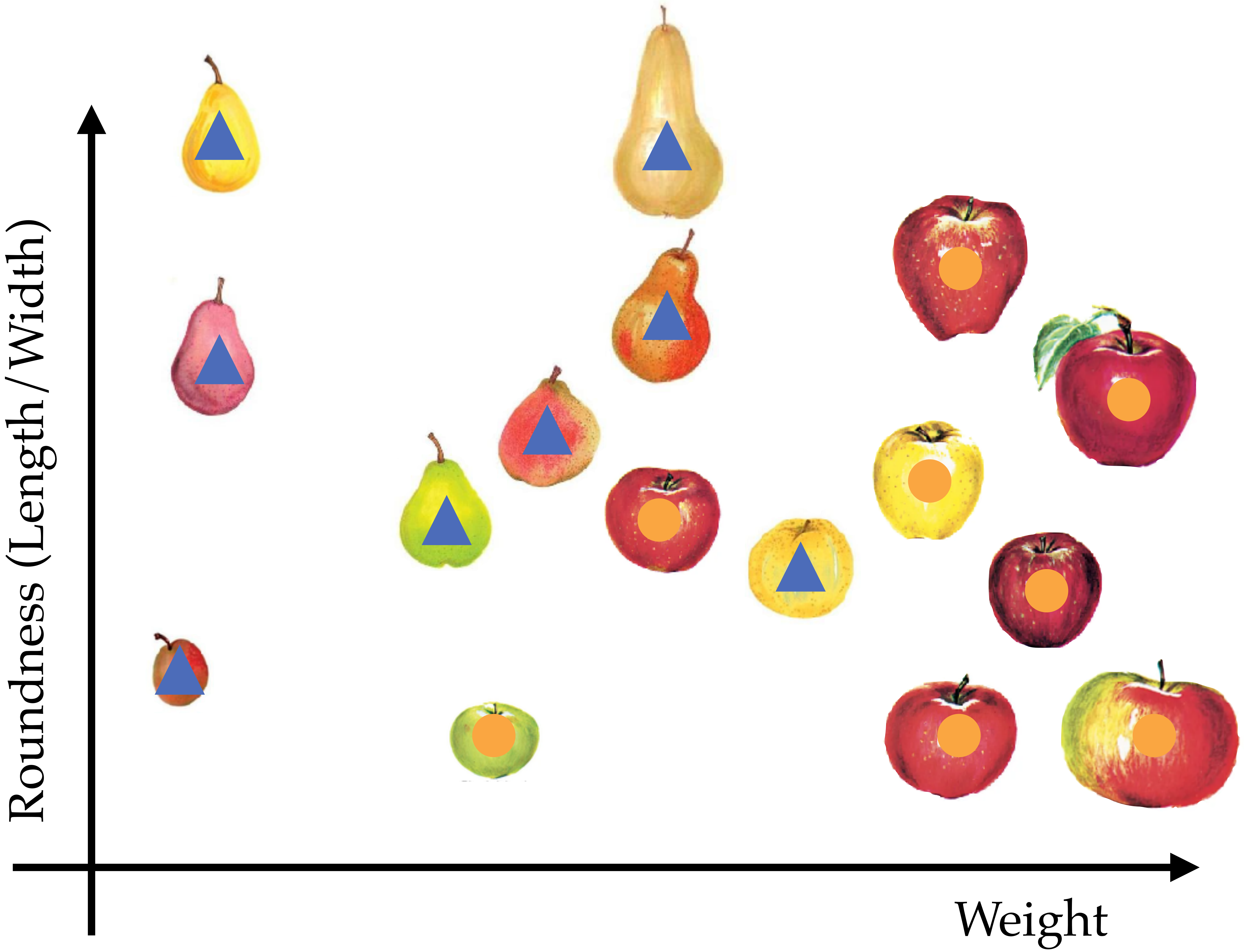
- How do I represent my object to use it in a mathematical function? What measurements do I do to represent the object?

- We call these measurements the features (variables, covariates), and depict them using a vector. We call this vector space the **feature space**. For example:

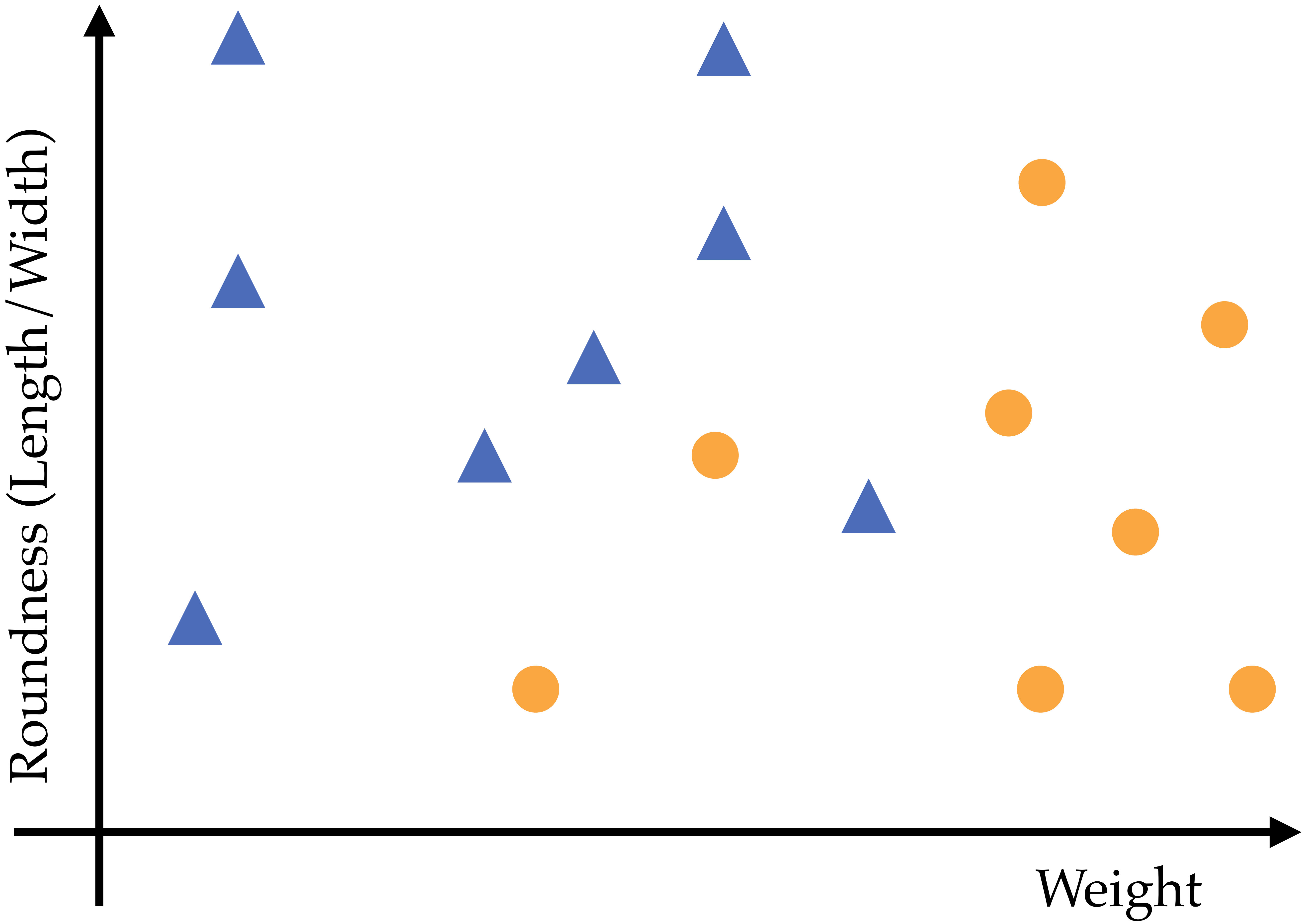
$$\mathbf{x} = \begin{bmatrix} x^1 \\ x^2 \\ ... \\ x^M \end{bmatrix} = \begin{bmatrix} 0.1 \\ -100 \\ 25 \\ 0 \end{bmatrix}$$

What measurements can I do that will help me distinguish the Apple from the Pear?

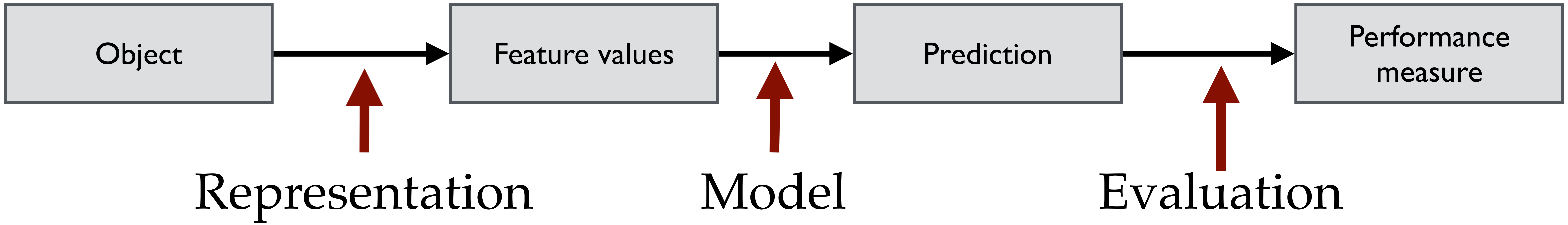

Roundness (Length/Width)
Weight


Roundness (Length/Width)
Weight

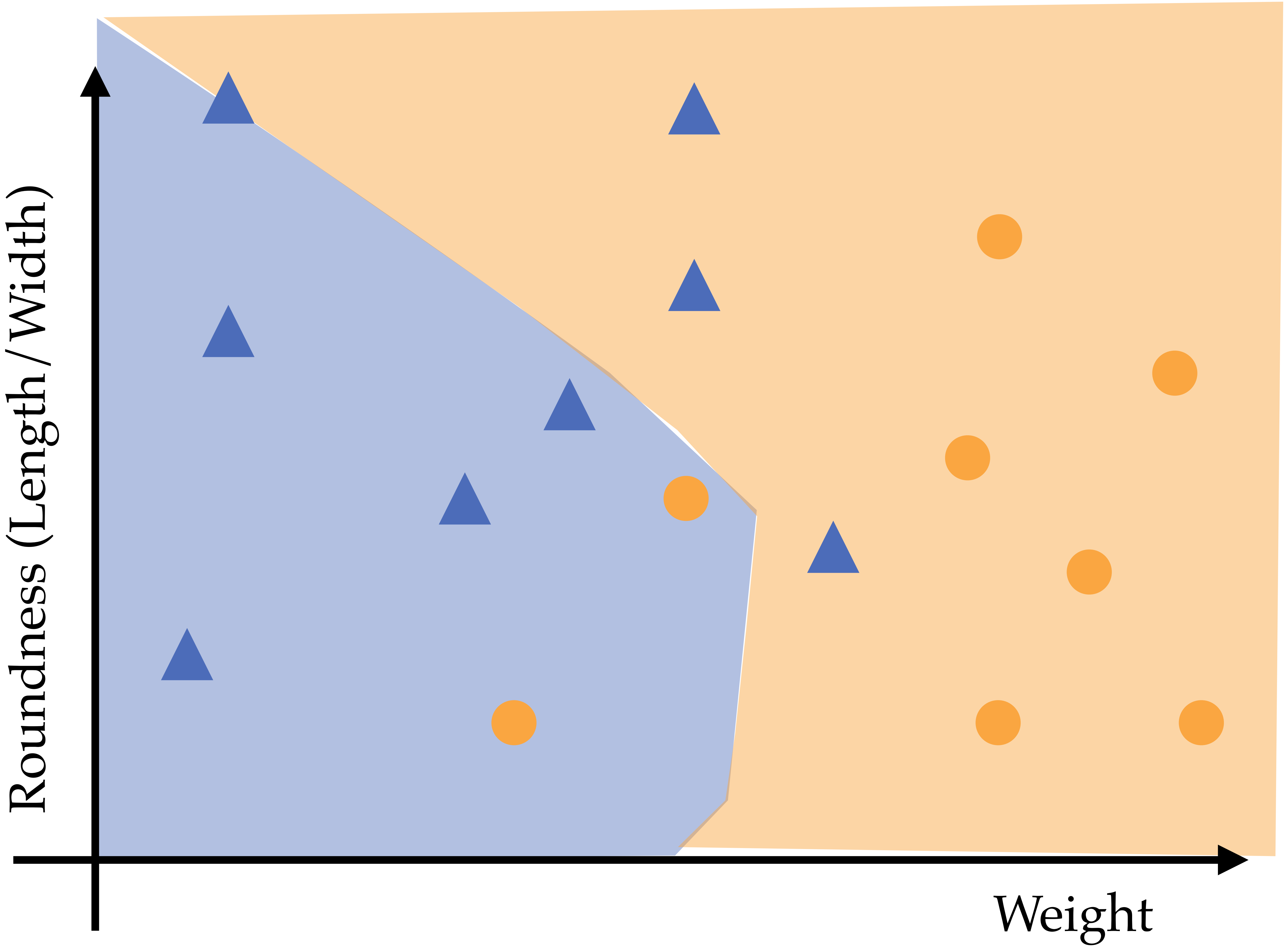
Classification Problem



Steps involved in a Machine Learning model

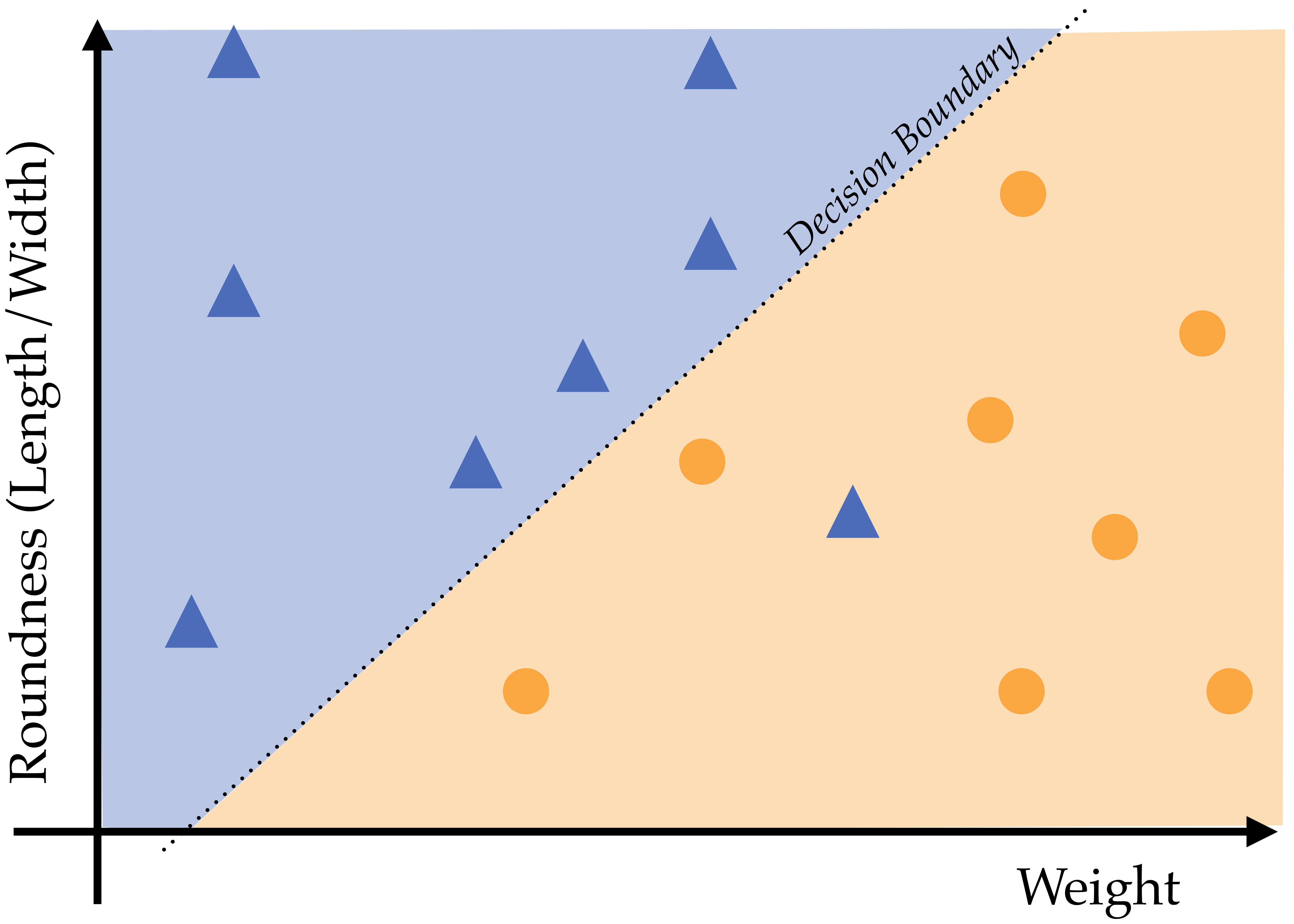


What does a mapping/classifier/model look like



A function that assigns an output to every location in the input space

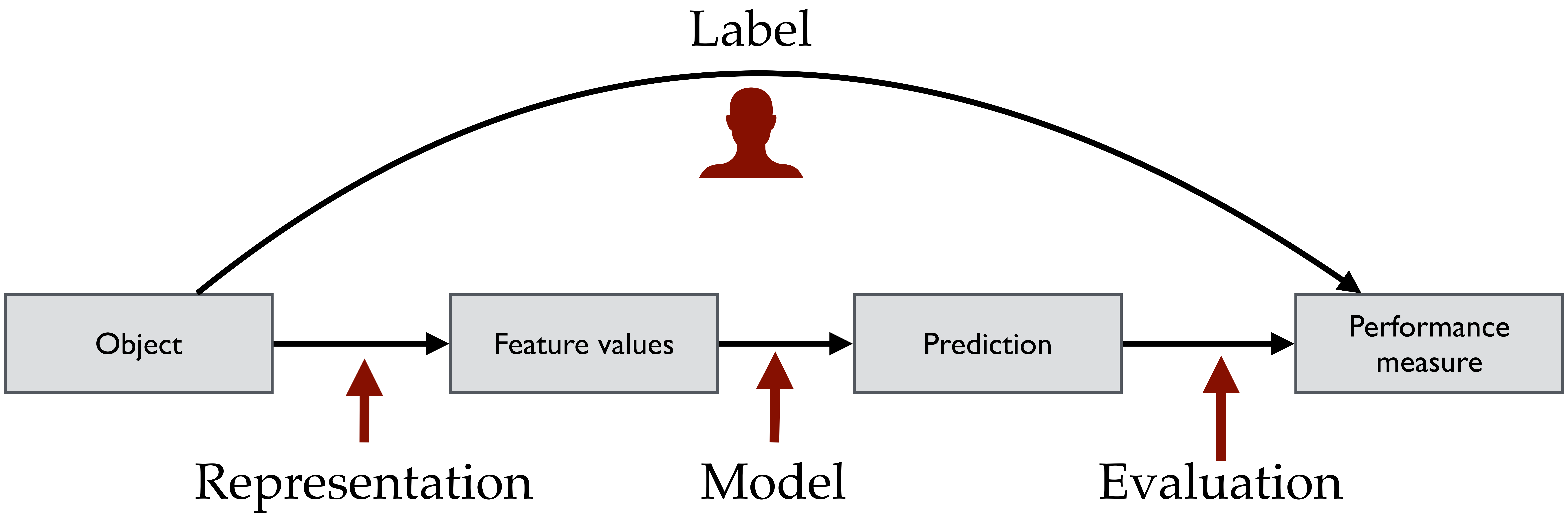
$$g : \mathbb{R}^2 \to \{\text{apple}, \text{pear}\}$$


Decision Boundary
Roundness (Length/Width)
Weight

More on Features

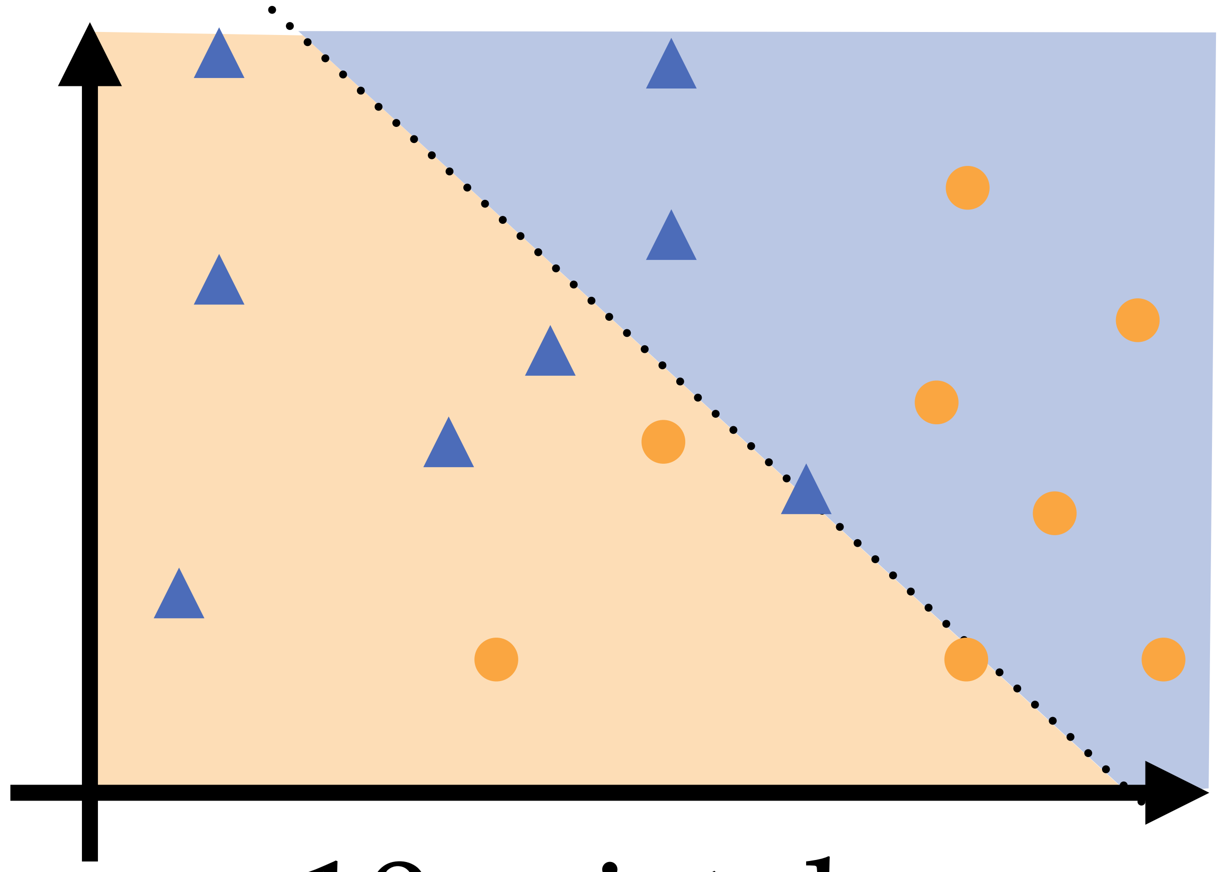
- Choosing informative features for the task makes it easy to discriminate between different classes
- If we do not measure the right things, it is impossible to have a model that performs well.
- See also: Bayes error, later this week

Steps involved in Machine Learning

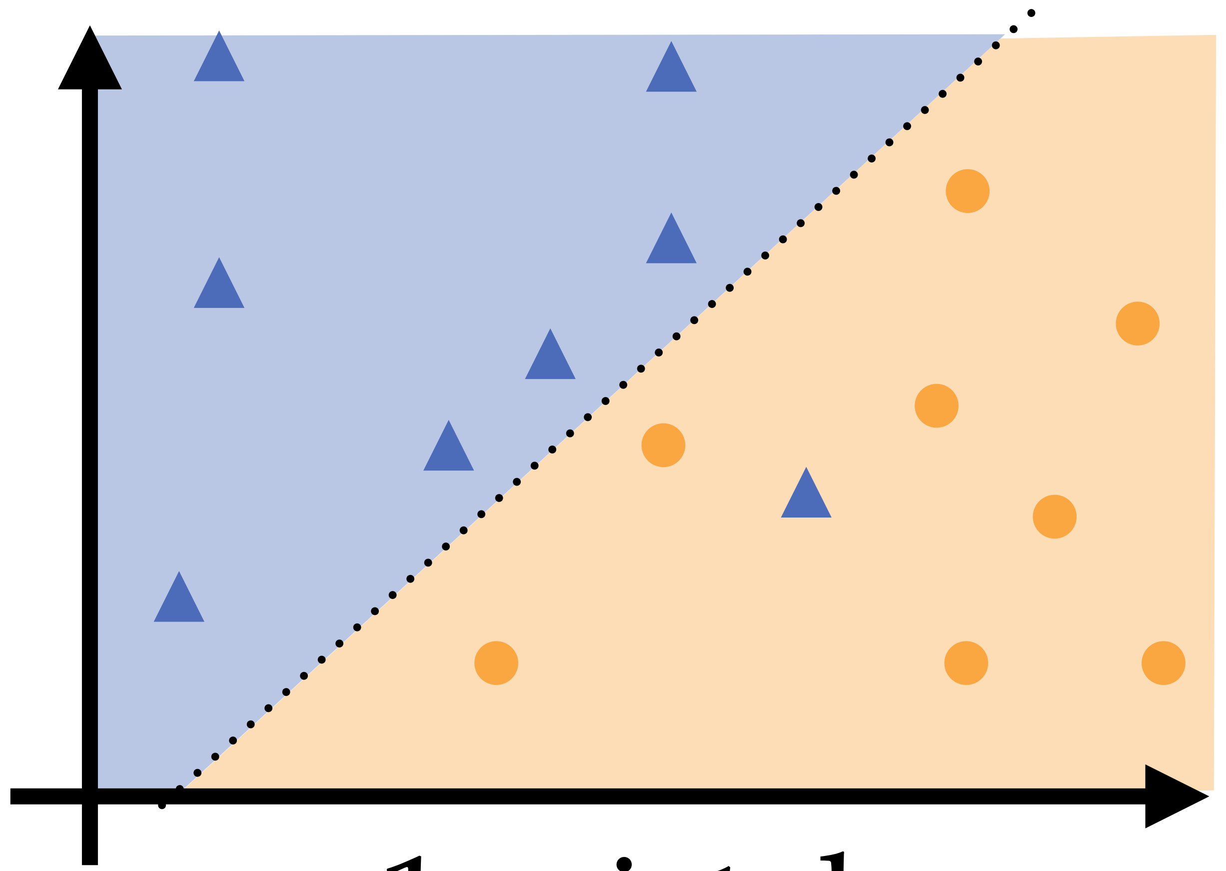


Evaluation: is this a good mapping?

- Measure the performance: how well does the mapping solve the task?
- Measure this using a “loss function”
  - Specific to the problem we are trying to solve
- Idea here: perhaps we should count the number of mistakes.

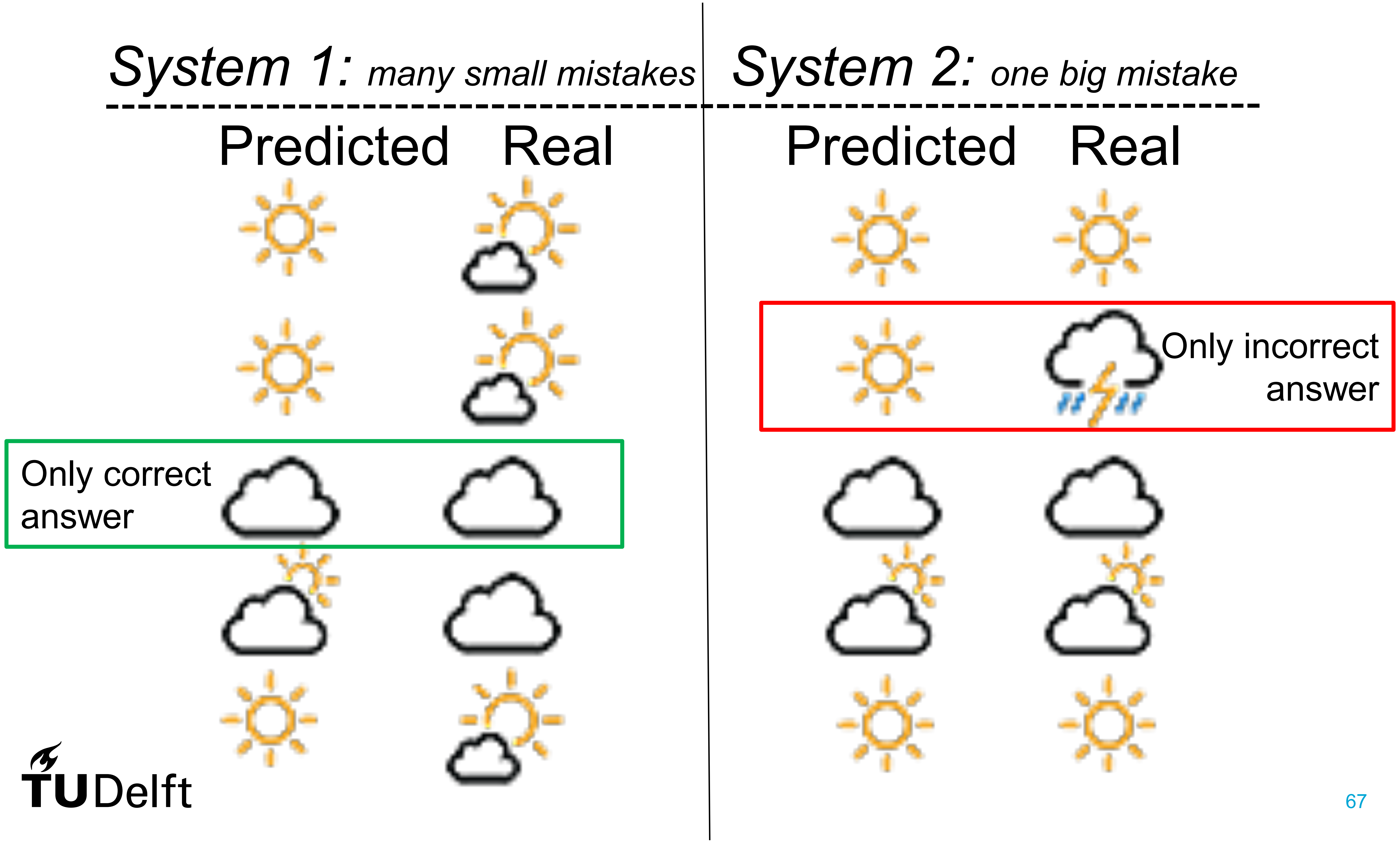


10 mistakes



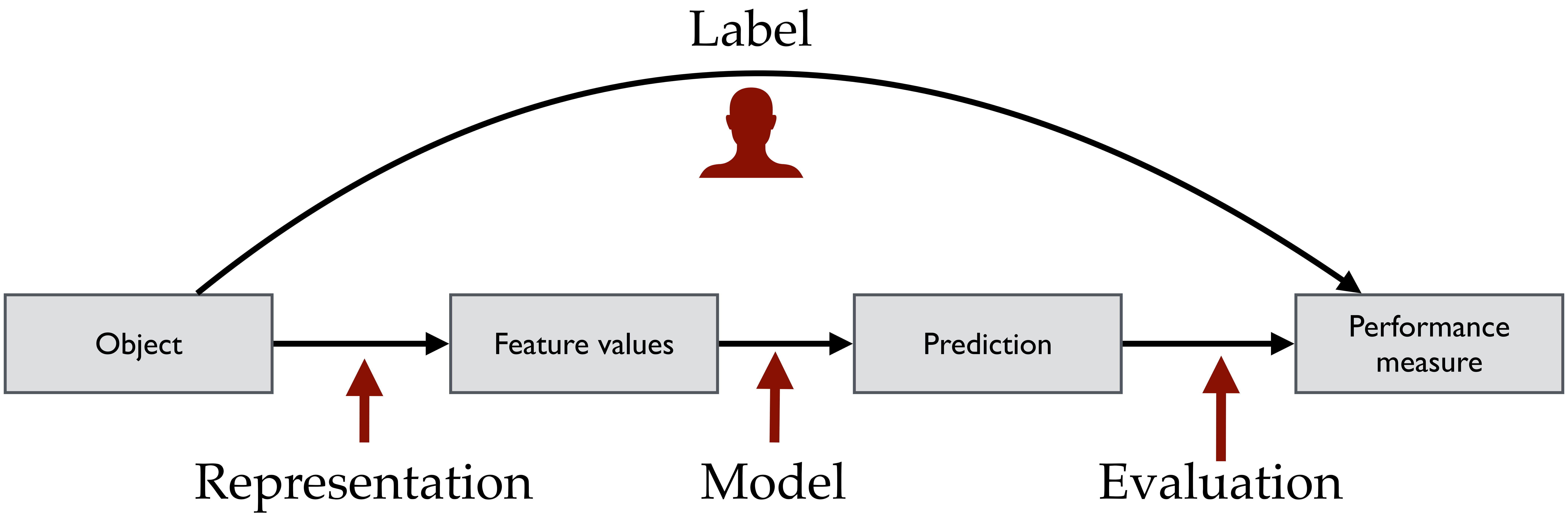
1 mistake

Evaluation: Choosing a Performance Metric



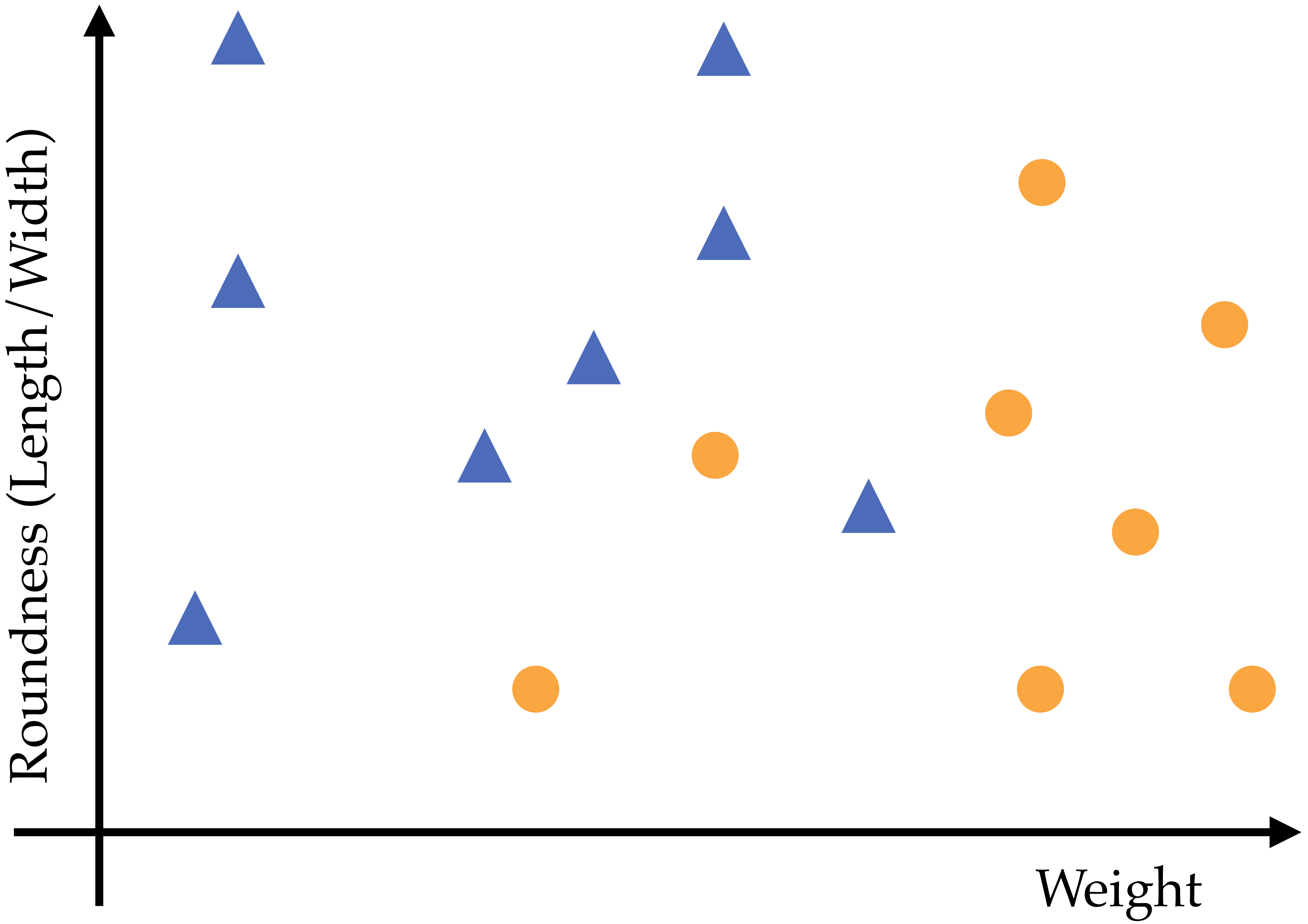
Slide courtesy of Odette Scharenborg

Steps involved in Machine Learning



What about **learning**?

Learning



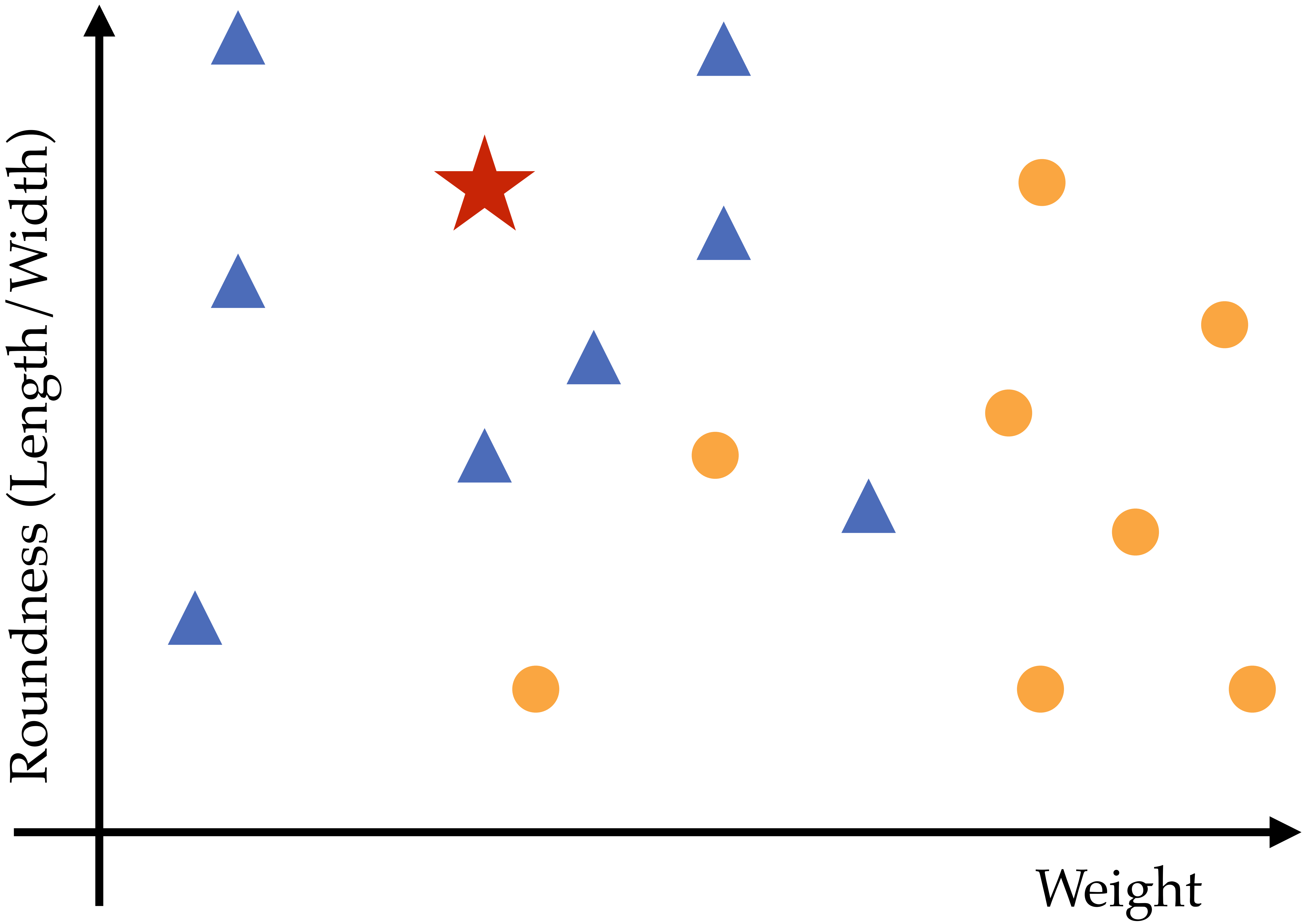
Learning and Generalisation

- What about remembering?
- Going beyond just remembering each specific instance, to generalising to new instances.
- We want to learn an input-output mapping that works for previously unseen objects as well.

"To think is to forget a difference, to generalize, to abstract."

- Jorge Luis Borges in "Funes the Memorious"

Model Predictions



Learning: how to find a mapping

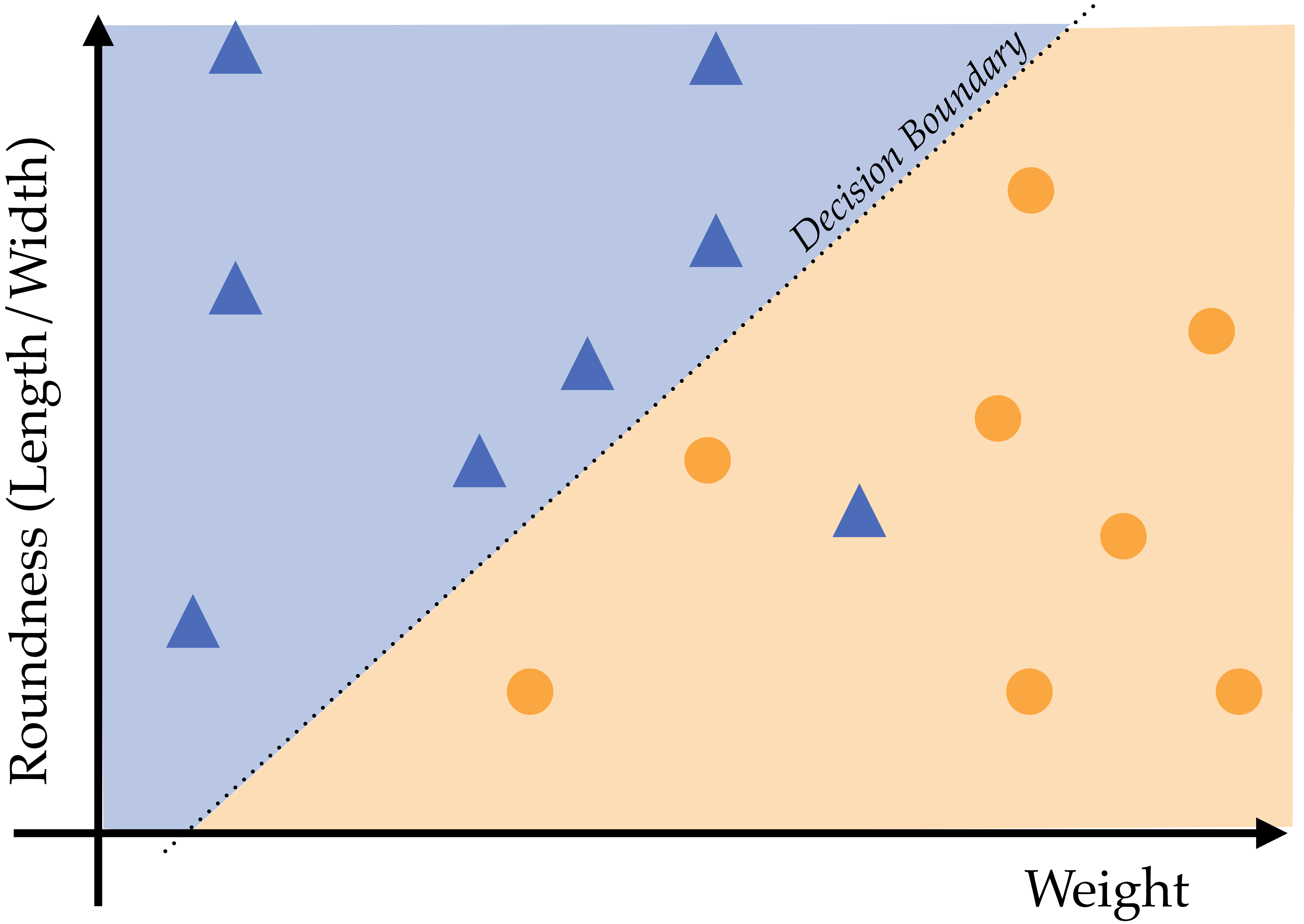
Most ML approaches: find parameters $\mathbf{w}$ for a function $g_{\mathbf{w}}$ that maximise the performance (or a related performance metric).

Problem 1: what functions should we consider?

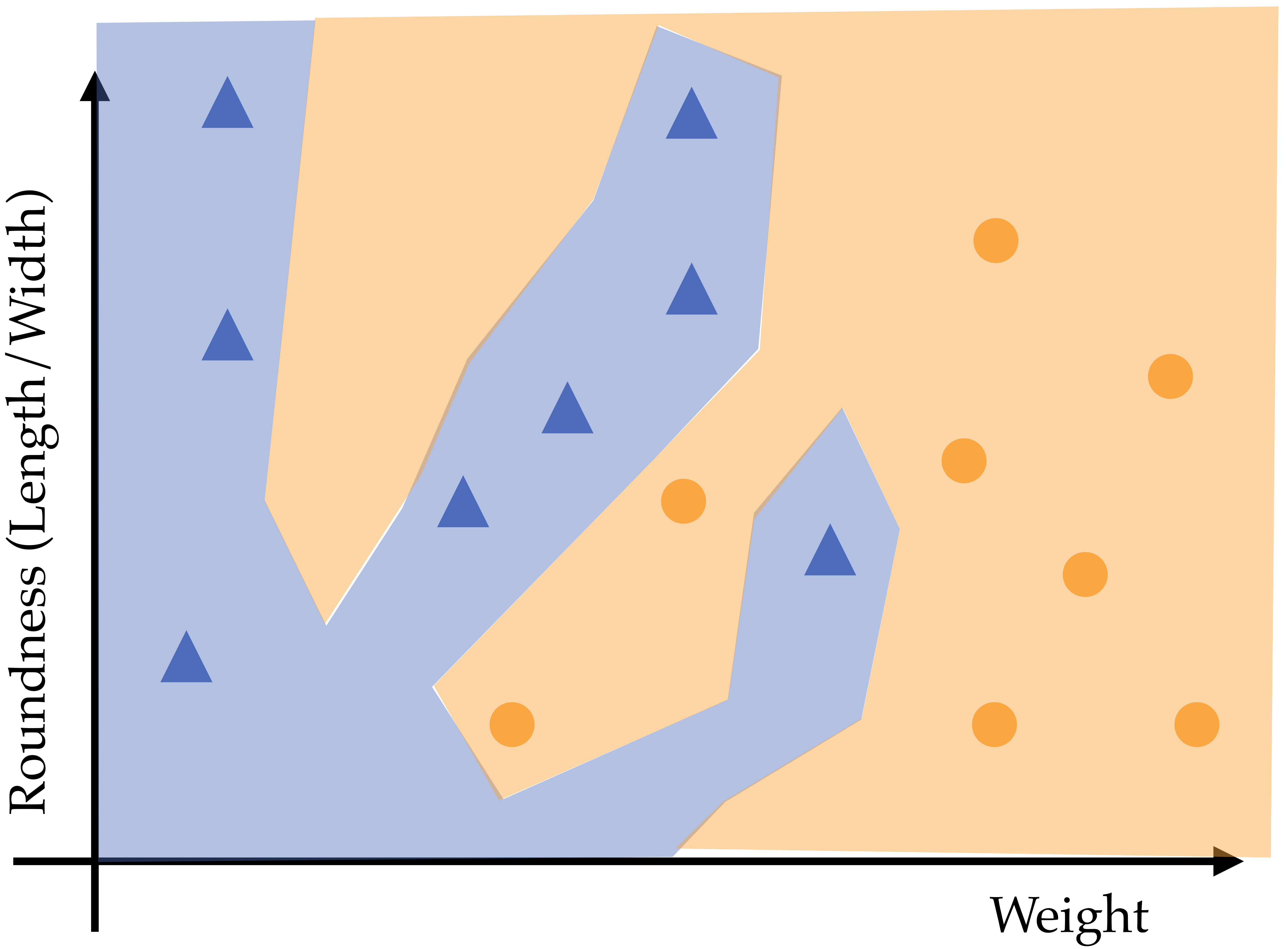
Problem 2: we don't have access to the true distribution

This course: cover many ideas on how to deal with these problems to construct learning algorithm.

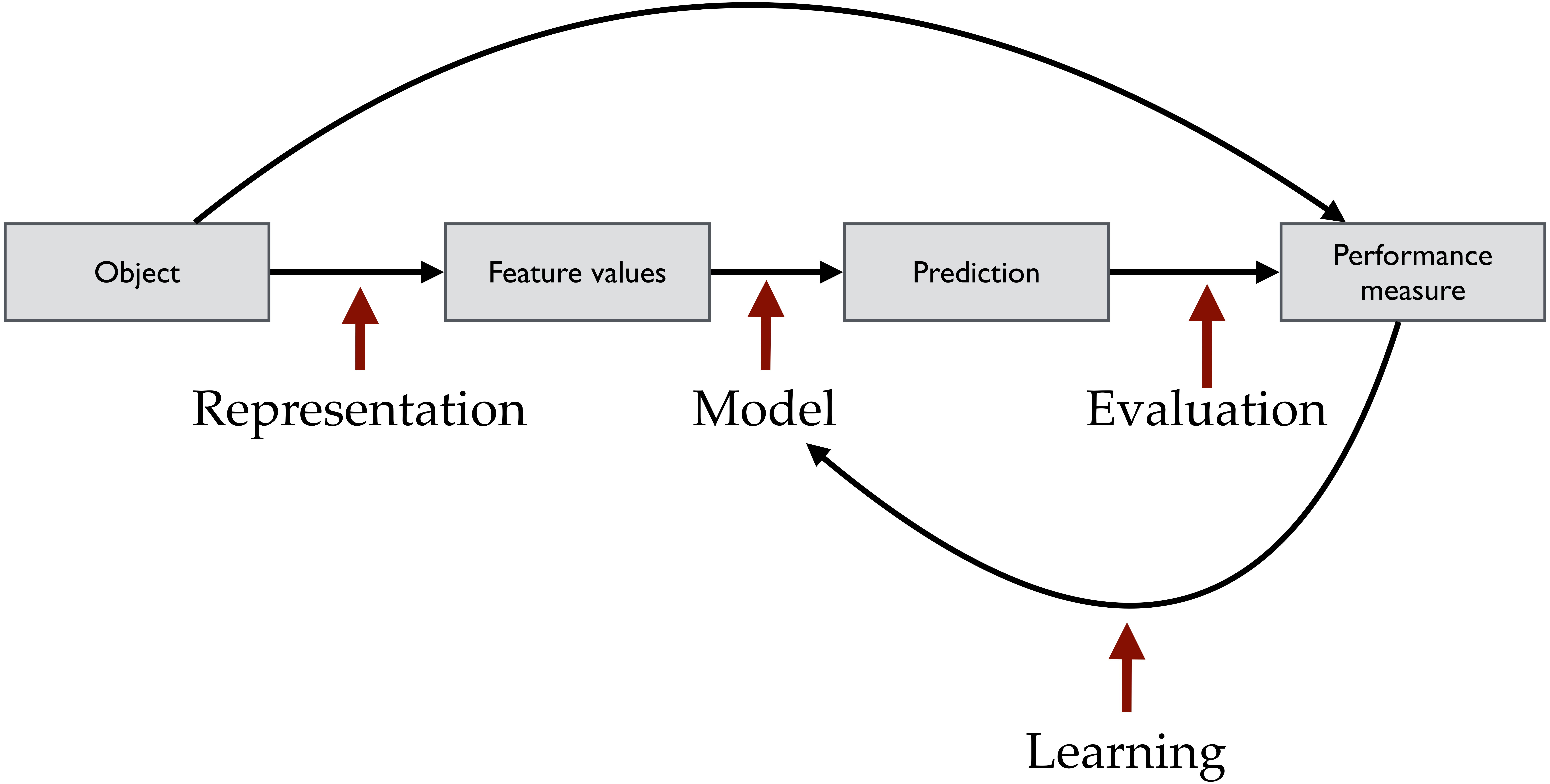
Linear Classifier



Non-Linear Classifier (Better?)



Steps involved in Machine Learning



Data

	Label	*Features*			
	Class	**Weight**	**Length/ Width**	**Greenness**	**Density**
Each object is a row → **1**	Apple	150	1.05	0.2	5
2	Apple	200	1.5	0.1	3
3	Pear	100	1.5	0.8	4
4	Apple	300	0.8	0.6	3

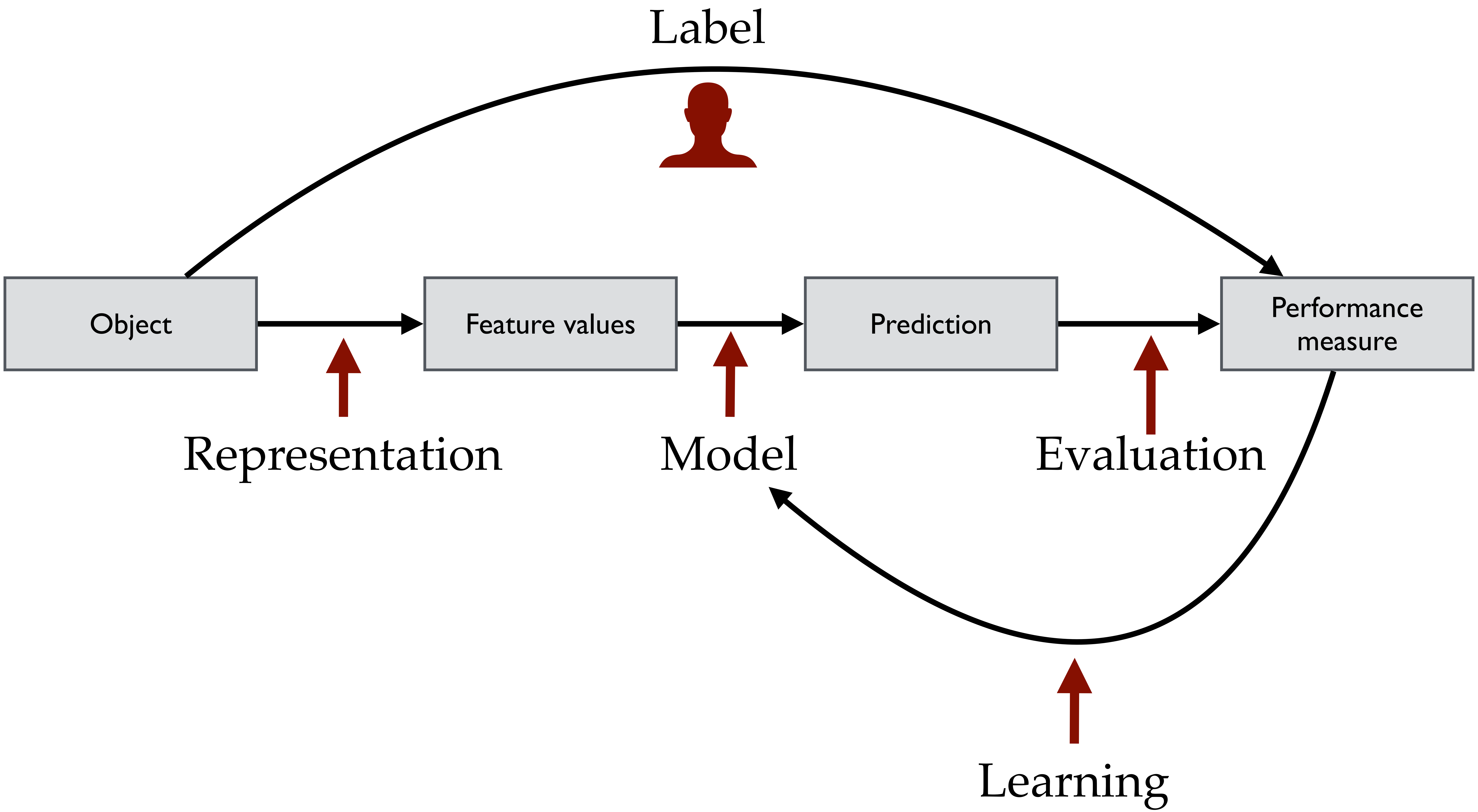
Label vector **t** (or **y**) N x M Design matrix **X**

Important common assumption: data sampled independently and identically distributed (i.i.d.) from an underlying problem distribution $P(X,t)$.

Data & Evaluation

- Remember: we are also interested in the performance on new objects that we have not seen yet (it is unlikely that new fruit will have the exact same weight and dimensions as those fruits that we have observed before).
- Idea: split data into a **training set** and a **test set**
  - Use training set to fit the model
  - Evaluate performance on the test set
  - Since the test set is an i.i.d. sample from the data generating process, this can give a fair estimate of the generalisation performance
  - More on this in week 3.2

Steps involved in Machine Learning



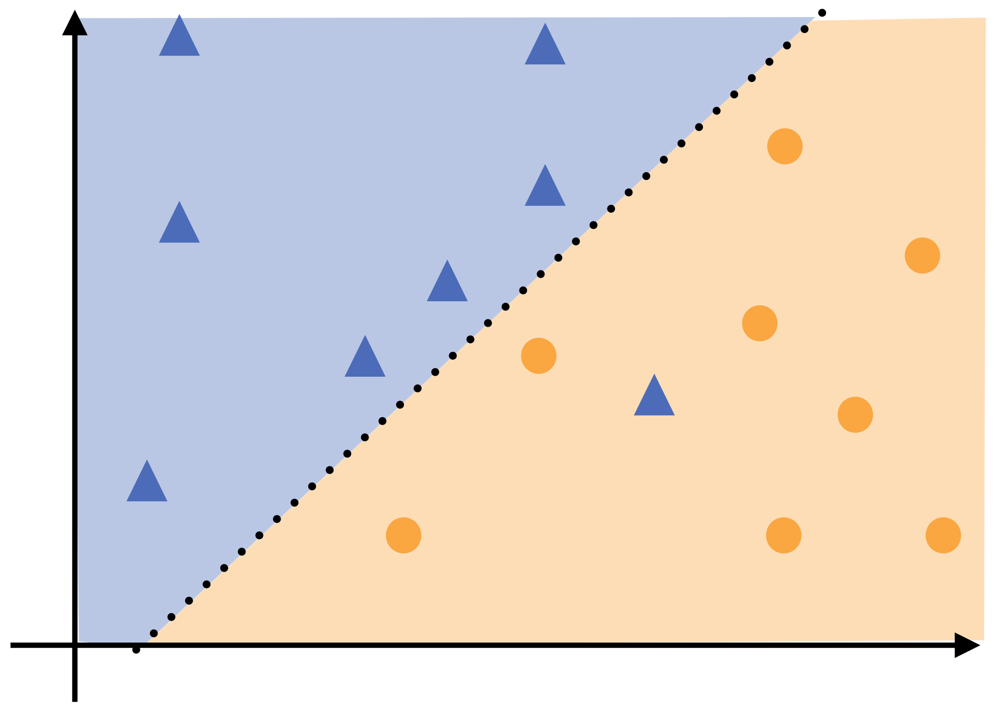
Types of Machine Learning

- Supervised Learning
  - Examples: **Classification**, Regression
- Unsupervised Learning
  - Examples: Dimensionality Reduction, Clustering
- Reinforcement Learning (not this course)
  - Example: Selecting optimal actions

Supervised Learning

CLASSIFICATION

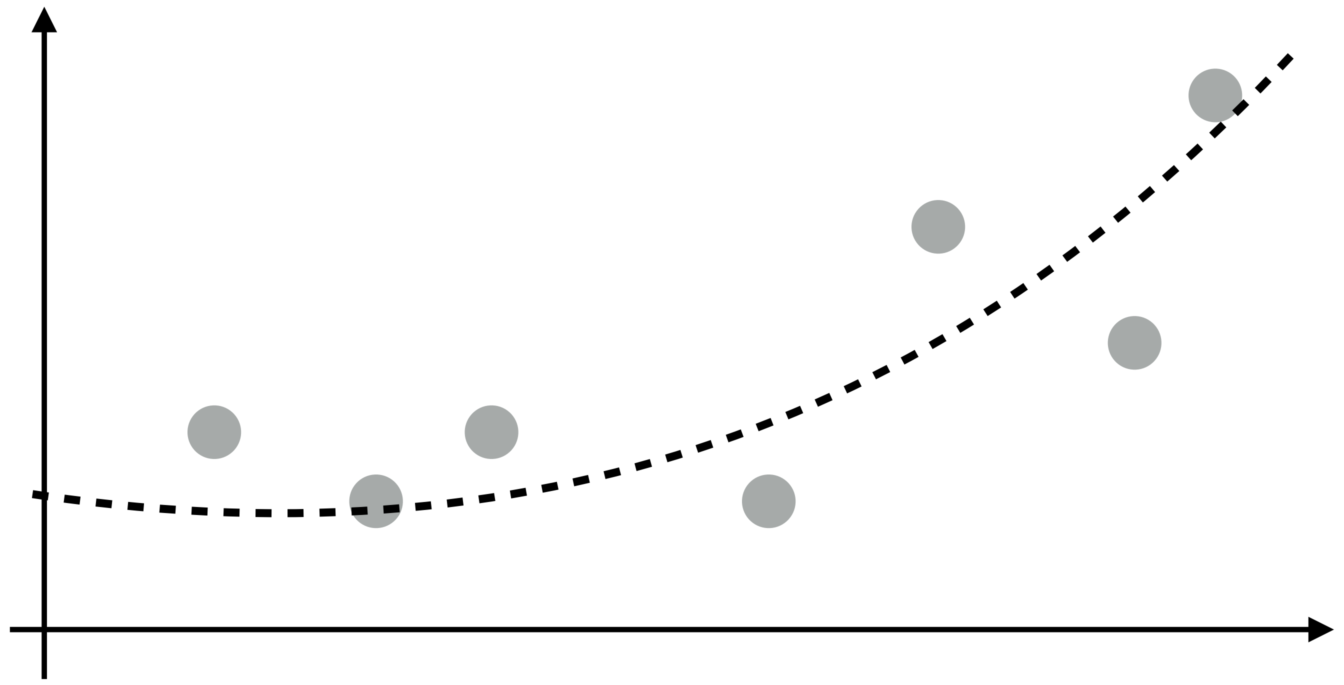
$$g : \mathbb{R}^M \to S \text{ where } |S| \text{ is small}$$



Examples
Object detection
Click prediction

REGRESSION

$$g : \mathbb{R}^M \to \mathbb{R}$$



Examples
Price prediction
Temperature prediction

“STRUCTURED OUTPUTS”

Example

$$g : \text{sentence} \to \text{image}$$

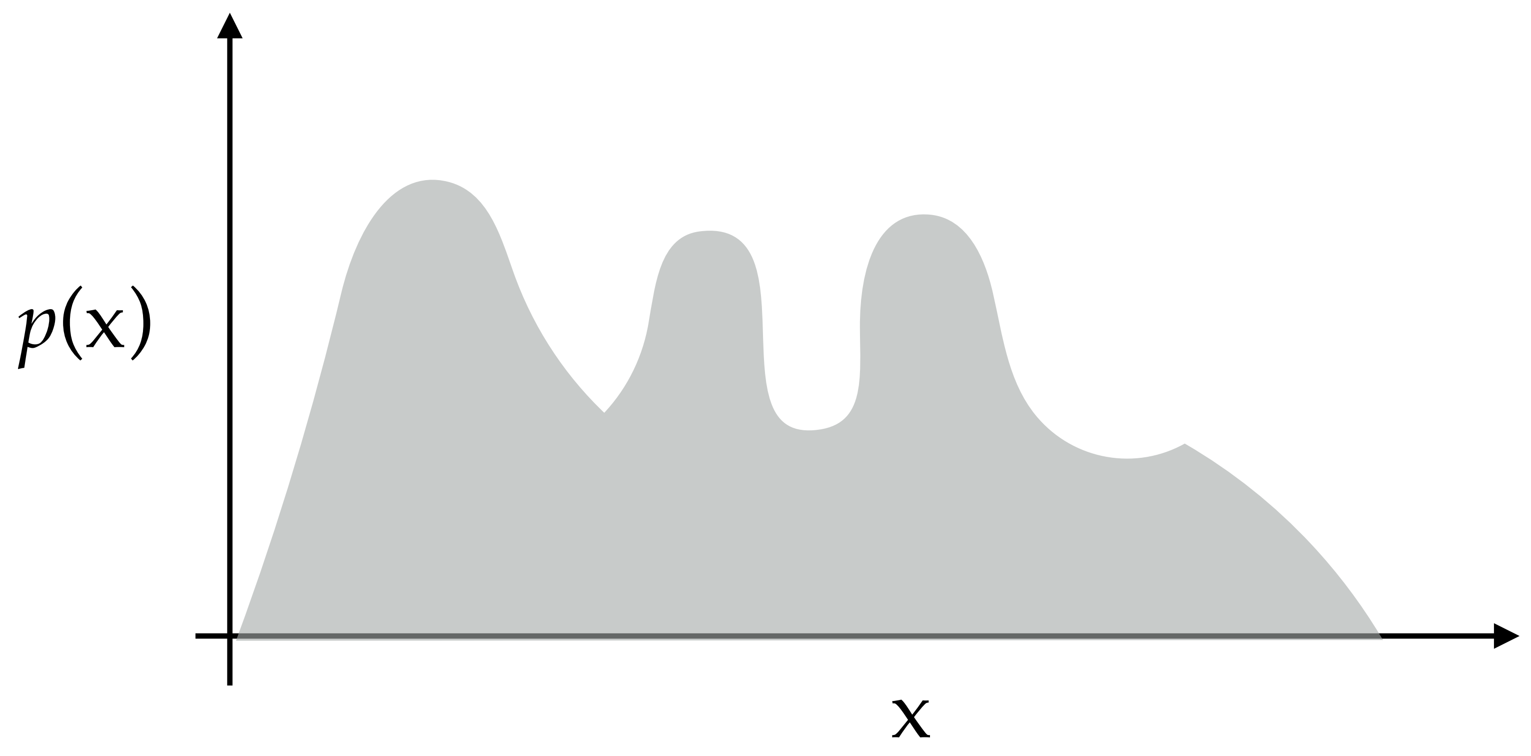
“A Hopper painting of a village” →

Unsupervised Learning

No output label given

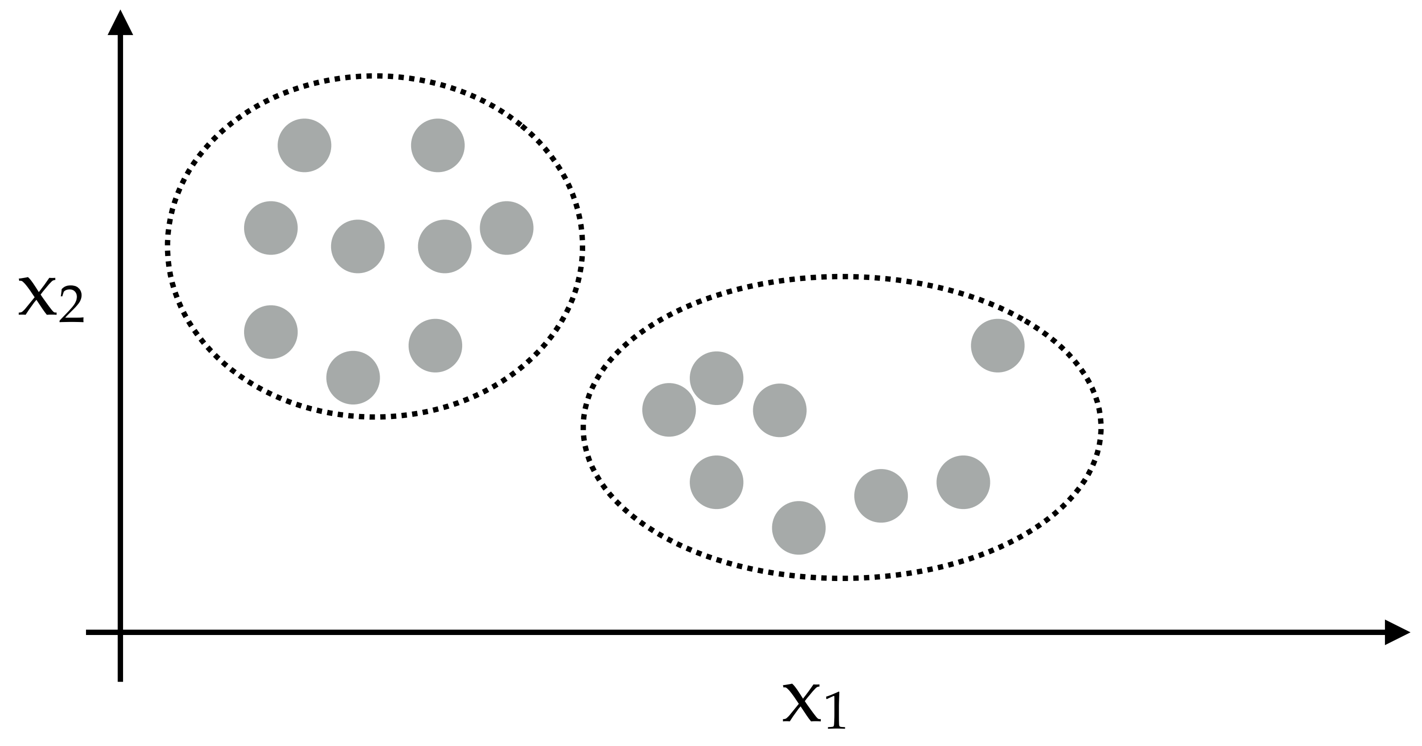
Density estimation

- For each **x**, give an estimate of the probability density $p(\mathbf{x})$.



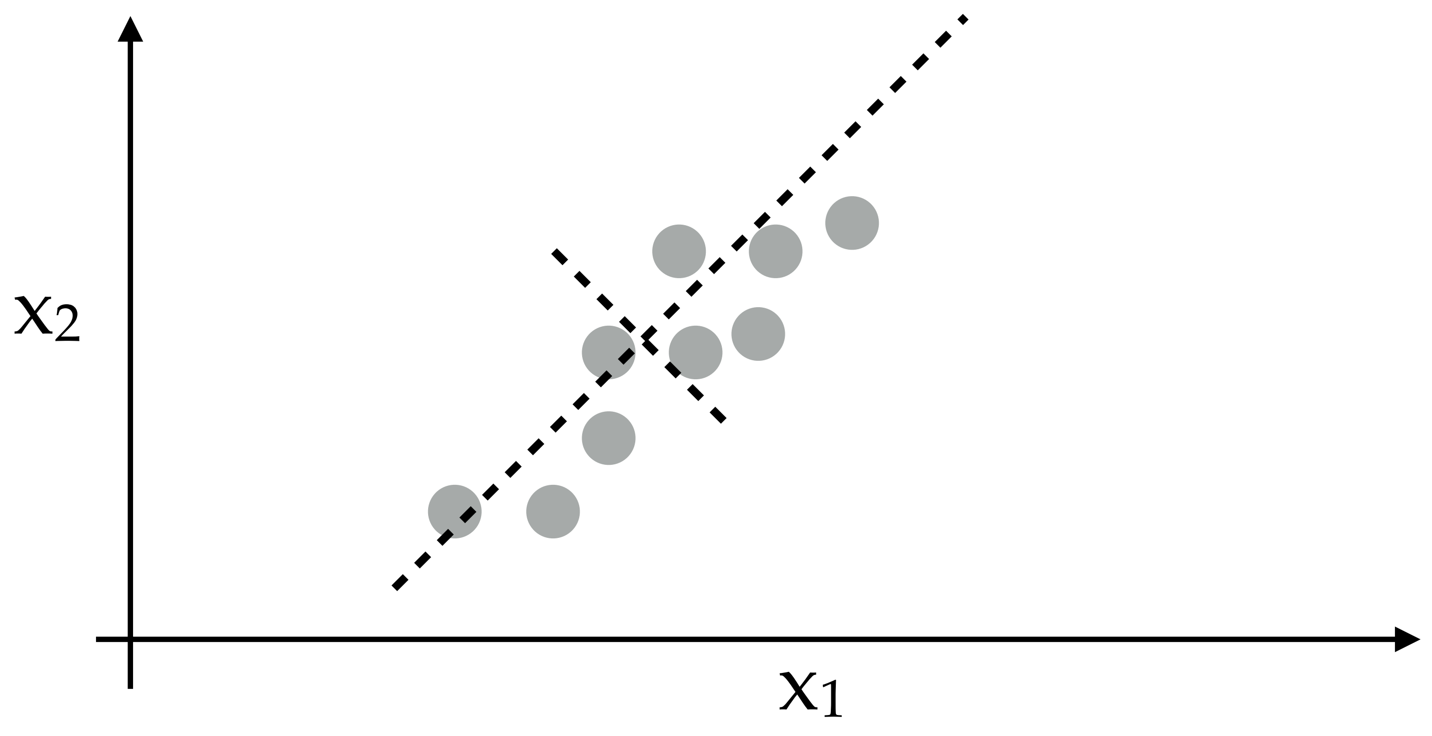
Clustering

- Identify a small number of "groups" in the data
- For instance, groups of observations that are internally similar, but different from the other groups

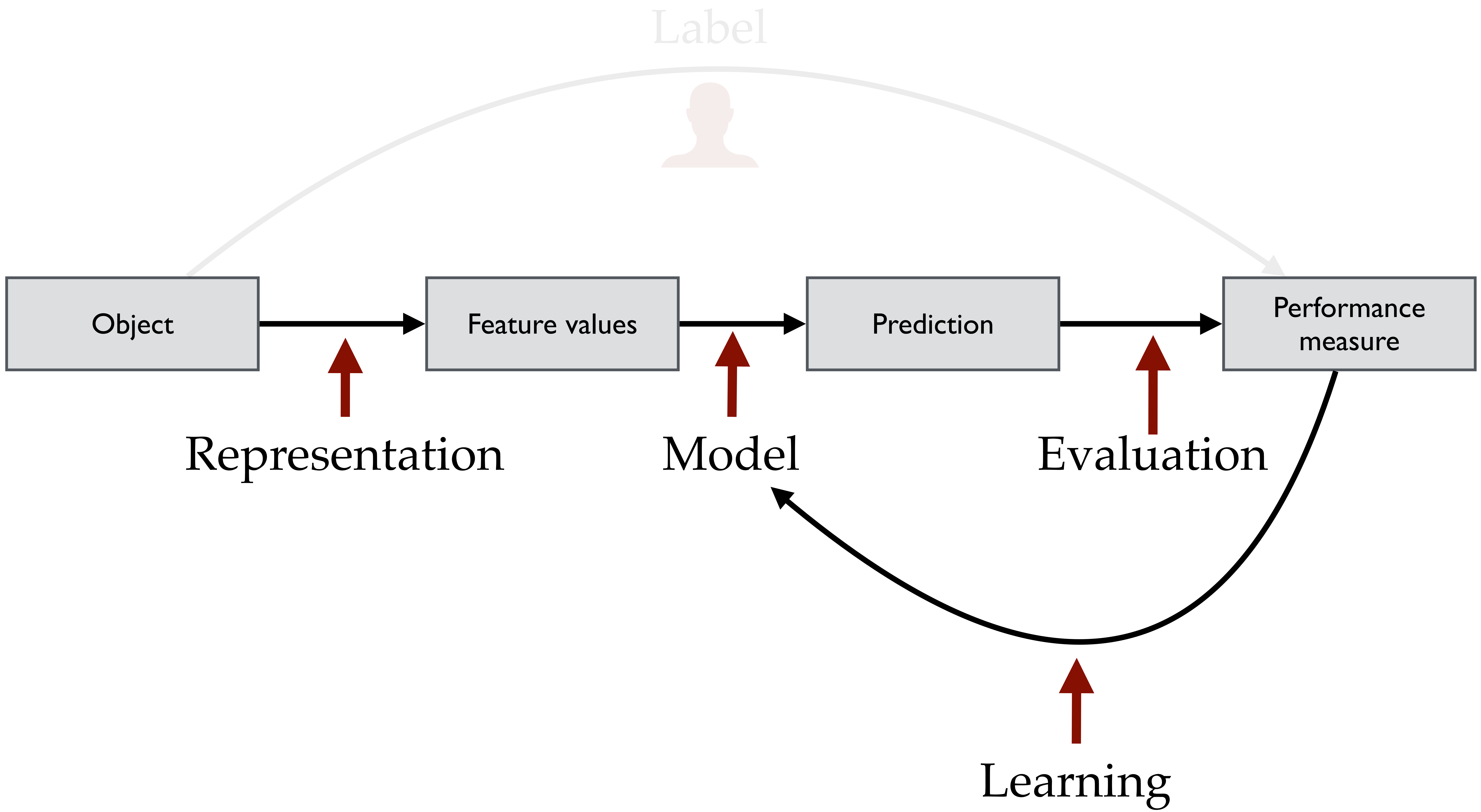


Dimensionality reduction

- Find a lower dimensional description of a high dimensional objects
- For instance, construct new features that contain most of the information in a fewer number of features.



Steps involved in Machine Learning



Practice: What Type of Learning?

- Predicting customer churn (whether a customer leaves or not)
- Finding groups of costumers that behave similarly
- Playing a video game
- Discovering different ways in which people play a videogame
- Identifying which photos contain waterfalls
- Predicting the price of a house
- Identifying which tax forms are more likely to be fraudulent
- Ranking webpages based on a search query
- Recommending the next movie a customer should watch
- Predicting which TikTok video the user is most likely to continue watching
- Predicting the amount of rainfall next Saturday

In the course we cover at least 10 classifiers.

Why do we need to study so many different learning approaches?

Why not just study the best one?

No Free Lunch Theorems and Machine Learning

- No free lunch theorems: uniformly averaged over all data-generating processes, every classification algorithm has the same out of sample prediction error. *(Note: the actual theorems are more precise and specific)*

- *For every problem a learner works well on, there is a problem on which it does not work well*

- **Lesson**: we are not looking to find the learning algorithm that is always best, but we want to find out which approach works well for which problem, and understand why.

*Wolpert DH. The Lack of A Priori Distinctions Between Learning Algorithms. Neural Computation. 1996;8(7):1341-1390. doi:*10.1162/neco.1996.8.7.1341

Take-aways

- Machine Learning is about using examples to learn a mapping from input to a desired outputs, that **generalises well** to new examples.
- In this course we focus on supervised (classification & regression) and unsupervised learning problems
- We specify a learning problem using objects, features, classes (targets), data and a performance metric.
- Learning has close connections to statistics and decision theory: next lecture!