

# Introduction to Machine Learning

Christos Dimitrakakis

July 22, 2024

# Outline

## The problems of Machine Learning (1 week)

- Introduction

## Estimation

- Answering a scientific problem

- Simple modelling

## Validating models

## Course summary

- Course Contents

- Objective functions

- Pitfalls

## Reading

# The problems of Machine Learning (1 week)

## Introduction

Estimation

Validating models

Course summary

Reading

# Machine Learning And Data Mining

## The nuts and bolts

- ▶ Models
- ▶ Algorithms
- ▶ Theory
- ▶ Practice

## Workflow

- ▶ Scientific question
- ▶ Formalisation of the problem
- ▶ Data collection
- ▶ Analysis and model selection

## Types of machine learning / statistics problems

- ▶ Classification
- ▶ Regression
- ▶ Density estimation
- ▶ Clustering

# Machine learning

## Data Collection

- ▶ Downloading a clean dataset from a repository
- ▶ Performing a survey
- ▶ Scraping data from the web
- ▶ Deploying sensors, performing experiments, and obtaining measurements.

## Modelling (what we focus on this course)

- ▶ Simple: the bias of a coin
- ▶ Complex: a language model.
- ▶ The model depends on the data and the problem

## Algorithms and Decision Making

- ▶ We want to use models to make decisions.
- ▶ Decisions are made every step of the way.
- ▶ Decisions are automated algorithmically.

# The main problems in machine learning and statistics

## Prediction

- ▶ Will it rain tomorrow?
- ▶ How much will bitcoin be worth next year?

## Inference

- ▶ Does my poker opponent have two aces?
- ▶ What is the mass of the moon?
- ▶ What is the law of gravitation?

## Decision Making

- ▶ Should I go hiking tomorrow?
- ▶ Should I buy some bitcoins?
- ▶ Should I fold, call, or raise in my poker game?
- ▶ How can I get a spaceship to orbit the moon?

# The need to learn from data

## Problem definition

- ▶ What problem do we need to solve?
- ▶ How can we formalise it?
- ▶ What properties of the problem can we learn from data?

## Data collection

- ▶ Why do we need data?
- ▶ What data do we need?
- ▶ How much data do we want?
- ▶ How will we collect the data?

## Modelling and decision making

- ▶ How will we compute something useful?
- ▶ How can we use the model to make decisions?

# Problem definition

- ▶ Health, weight and height

## Health questions regarding height and weight

- ▶ What is a normal height and weight?
- ▶ How are they related to health?
- ▶ What variables affect height and weight?



# Data collection

Think about which variables we need to collect to answer our research question.

## Necessary variables

The variables we need to know about

- ▶ Weight
- ▶ Height
- ▶ Health issue:

## Auxiliary variables

Factors related to height, weight and health

## Possible confounders

Other factors that might affect health outcomes, unrelated to height and weight

# Class data

- ▶ The class enters their data into the excel file.

## Mean estimation

- ▶ What is the average height/disease prevalence?
- ▶ What is the expected height/disease prevalence in the general population?

## Supervised learning problems:

- ▶ Classification: Can we predict gender from height/weight?
- ▶ Regression: Can we predict weight from height and gender?
- ▶ In both cases we predict **output** variables from **input** variables

## Input variables

Also called features, predictors, independent variables

## Output variables

Also called response, or dependent variables.

# Variables

The class data looks like this

| First Name | Gender | Height | Weight | Age | Nationality | Smoking |
|------------|--------|--------|--------|-----|-------------|---------|
| Lee        | M      | 170    | 80     | 20  | Chinese     | 10      |
| Fatemeh    | F      | 150    | 65     | 25  | Turkey      | 0       |
| Ali        | Male   | 174    | 82     | 19  | Turkish     | 0       |
| Joan       | N      | 5'11   | 180    | 21  | Brtish      | 4       |

- ▶  $\mathbf{X}$ : Everybody's data
- ▶  $x_t$ : The  $t$ -th person's data
- ▶  $x_{t,k}$ : The  $k$ -th feature of the  $t$ -th person.
- ▶  $x_k$ : Everybody's  $k$ -th feature

## Raw versus neat data

- ▶ Neat data:  $x_t \in \mathbb{R}^n$
- ▶ Raw data: text, graphs, missing values, etc

# Python pandas for data wrangling

## Reading class data

```
import pandas as pd
X = pd.read_excel("data/class.xlsx")
X["First Name"]
```

- ▶ Array columns correspond to features
- ▶ Columns can be accessed through namesx

## Summarising class data

```
X.hist()
import matplotlib.pyplot as plt
plt.show()
```

# Pandas and DataFrames

- ▶ Data in pandas is stored in a **DataFrame**
- ▶ DataFrame is **not the same** as a numpy array.

## Core libraries

```
import pandas as pd
import numpy as np
```

## Series: A sequence of values

```
# From numpy array:
s = pd.Series(np.random.randn(3), index=["a", "b", "c"])
# From dict:
d = {"a": 1, "b": 0, "c": 2}
s = pd.Series(d)
# accessing elements
s.iloc[2] #element 2
s.iloc[1:2] #elements 1,2
s.array # gets the array object
s.to_numpy() # gets the underlying numpy array
```

# DataFrames

## Constructing from a numpy array

```
data = np.random.uniform(size = [3,2])
df = pd.DataFrame(data, index=["John", "Ali", "Sumi"],
    columns=["X1", "X2"])
```

## Constructing from a dictionary

```
d = { "one": pd.Series([1, 2], index=["a", "b"]),
      "two": pd.Series([1, 2, 3], index=["a", "b", "c"])}
df = pd.DataFrame(d)
```

## Access

```
X["First Name"] # get a column
X.loc[2] # get a row
X.at[2, "First Name"] # row 2, column 'first name'
X.loc[2].at["First Name"] # row 2, element 'first name' of the s
X.iat[2,0] # row 2, column 0
```

# Means using python

## Calculating the mean of a random variable

```
import numpy as np
X = np.random.gamma(170, 1, size=20)
X.mean()
np.mean(X)
```

## Calculating the mean of our class data

```
X.mean() # gives the mean of all the variables through pandas.co
X["Height"].mean()
np.mean(X["Weight"])
```

# One variable: expectations and distributions

## Modelling the height

- ▶ Mean: models the expected value of the population
- ▶ Variance: models the variance of the population
- ▶ Empirical distribution: models the distribution of the population

## The expected value and the mean

Assume  $x_t : \Omega \rightarrow \mathbb{R}$ , and  $\omega \sim P$

- ▶  $x_1, \dots, x_t, \dots, x_T$ : random i.i.d. variables
- ▶  $\Omega$ : random outcome space
- ▶  $P$ : distribution of outcomes  $\omega \in \Omega$
- ▶  $\mathbb{E}_P[x]$ : expectation of  $x$  under  $P$

$$\mathbb{E}_P[x_t] = \sum_{\omega \in \Omega} x_t(\omega) P(\omega) \approx \frac{1}{T} \sum_{t=1}^T x_t$$

- ▶ The sample mean is  $O(1/\sqrt{T})$ -close to the expected value.



# Reminder: expectations of random variables

## A gambling game

What are the expected winnings if you play this game?

- ▶ [a] With probability 1%, you win 100 CHF
- ▶ [b] With probability 40%, you win 20 CHF.
- ▶ [c] Otherwise, you win nothing

## Solution

# Reminder: expectations of random variables

## A gambling game

What are the expected winnings if you play this game?

- ▶ [a] With probability 1%, you win 100 CHF
- ▶ [b] With probability 40%, you win 20 CHF.
- ▶ [c] Otherwise, you win nothing

## Solution

- ▶ Let  $x$  be the amount won, then  $x(a) = 100, x(b) = 20, x(c) = 0$ .
- ▶ We need to calculate

$$\mathbb{E}_P(x) = \sum_{\omega \in \{a,b,c\}} x(\omega)P(\omega) = x(a)P(a) + x(b)P(b) + x(c)P(c)$$

- ▶  $P(c) = 59\%$ , as  $P(\Omega) = 1$ . Substituting,  
$$\mathbb{E}_P(x) = 1 + 8 + 0 = 9.$$

# Populations, samples, and distributions

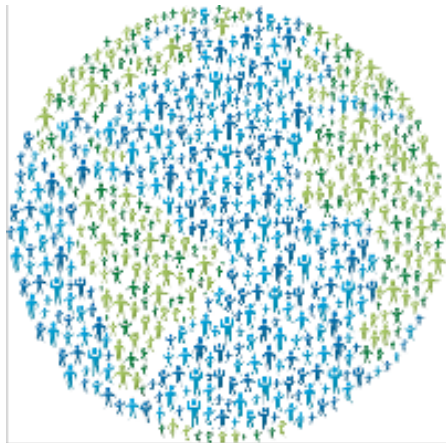


Figure: The world population



Figure: A sample

Two variables: conditional expectation

The height of different genders

# Learning from data

## Unsupervised learning

- ▶ Given data  $x_1, \dots, x_T$ .
- ▶ Learn about the data-generating process.
- ▶ Example: Estimation, compression, text/image generation

## Supervised learning

- ▶ Given data  $(x_1, y_1), \dots, (x_T, y_T)$
- ▶ Learn about the relationship between  $x_t$  and  $y_t$ .
- ▶ Example: Classification, Regression

## Online learning

- ▶ Sequence prediction: At each step  $t$ , predict  $x_{t+1}$  from  $x_1, \dots, x_t$ .
- ▶ Conditional prediction: At each step  $t$ , predict  $y_{t+1}$  from  $x_1, y_1, \dots, x_t, y_t, x_{t+1}$

## Reinforcement learning

Learn to act in an **unknown** world through interaction and rewards

# Models

## Models as summaries

- ▶ They summarise what we can see in the data
- ▶ The ultimate model of the data **is** the data

## Models as predictors

- ▶ They make predictions about **out of sample** data
- ▶ This requires some **assumptions about the data distribution**.

## Example models

- ▶ A numerical mean
- ▶ A linear classifier
- ▶ A linear regressor
- ▶ A deep neural network
- ▶ A Gaussian process
- ▶ A large language model

# Validating models

## Training data

- ▶ Calculations, optimisation
- ▶ Data exploration

## Validation data

- ▶ Fine-tuning
- ▶ Model selection

## Test data

- ▶ Performance comparison

## Simulation

- ▶ Interactive performance comparison
- ▶ White box testing

## Real-world testing

- ▶ Actual performance measurement

The simplest model: A mean



# Robust models of the mean

# Model selection

- ▶ Train/Test/Validate
- ▶ Cross-validation
- ▶ Simulation

# Course Contents

## Models

- ▶ k-Nearest Neighbours.
- ▶ Linear models and perceptrons.
- ▶ Multi-layer perceptrons (aka deep neural networks).
- ▶ Bayesian Networks

## Algorithms

- ▶ (Stochastic) Gradient Descent.
- ▶ Bayesian inference.

# Supervised learning

The general goal is learning a function  $f : X \rightarrow Y$ .

## Classification

- ▶ Input data  $x_t \in \mathbb{R}$ ,  $y_t \in [m] = \{1, 2, \dots, m\}$
- ▶ Learn a mapping  $f$  so that  $f(x_t) = y_t$  for unseen data

## Regression

- ▶ Input data  $x_t, y_t$
- ▶ Learn a mapping  $f$  so that  $f(x_t) = \mathbb{E}[y_t]$  for unseen data

# Unsupervised learning

The general goal is learning the data distribution.

## Compression

- ▶ Learn two mappings  $c, d$
- ▶  $c(x)$  compresses an image  $x$  to a small representation  $z$ .
- ▶  $d(z)$  decompresses to an approximate image  $\hat{x}$ .

## Density estimation

- ▶ Input data  $x_1, \dots, x_T$  from distribution with density  $p$
- ▶ Problem: Estimate  $p$ .

## Clustering

- ▶ Input data  $x_1, \dots, x_T$
- ▶ Assign each data  $x_t$  to cluster label  $c_t$ .

# Supervised learning objectives

- ▶ Data  $(x_t, y_t)$ ,  $x_t \in X$ ,  $y_t \in Y$ ,  $t \in [T]$ .
- ▶ i.i.d assumption:  $(x_t, y_t) \sim P$  for all  $t$ .
- ▶ Supervised decision rule  $\pi(a_t|x_t)$

## Classification

- ▶ Predict the labels correctly, i.e.  $a_t = y_t$ .
- ▶ Have an appropriate confidence level

## Regression

- ▶ Predict the mean correctly
- ▶ Have an appropriate variance around the mean

# Unsupervised learning objectives

- ▶ Reconstruct the data well
- ▶ Be able to generate data

# Reinforcement learning objectives

- ▶ Maximise total reward



# Pitfalls

## Reproducibility

- ▶ Modelling assumptions
- ▶ Distribution shift
- ▶ Interactions and feedback

## Fairness

- ▶ Implicit biases in training data
- ▶ Fair decision rules and meritocracy

## Privacy

- ▶ Accidental data disclosure
- ▶ Re-identification risk

