

Introduction to intelligent systems

Introduction

Mikkel N. Schmidt

Technical University of Denmark,
DTU Compute, Department of Applied Mathematics and Computer Science.

Overview

- ① Welcome to Introduction to Intelligent Systems
- ② Expectations survey
- ③ The historical origin of AI
- ④ Intelligence and the brain
- ⑤ Methods of reasoning
- ⑥ Demo of image recognition
- ⑦ Python installation instructions
- ⑧ Tasks

Feedback group

- Jakob Lauge Reeh
- David Lindahl
- Philip Thinggaard
- Poul Skov

Learning objectives

I Reasoning: Induction, deduction, abduction

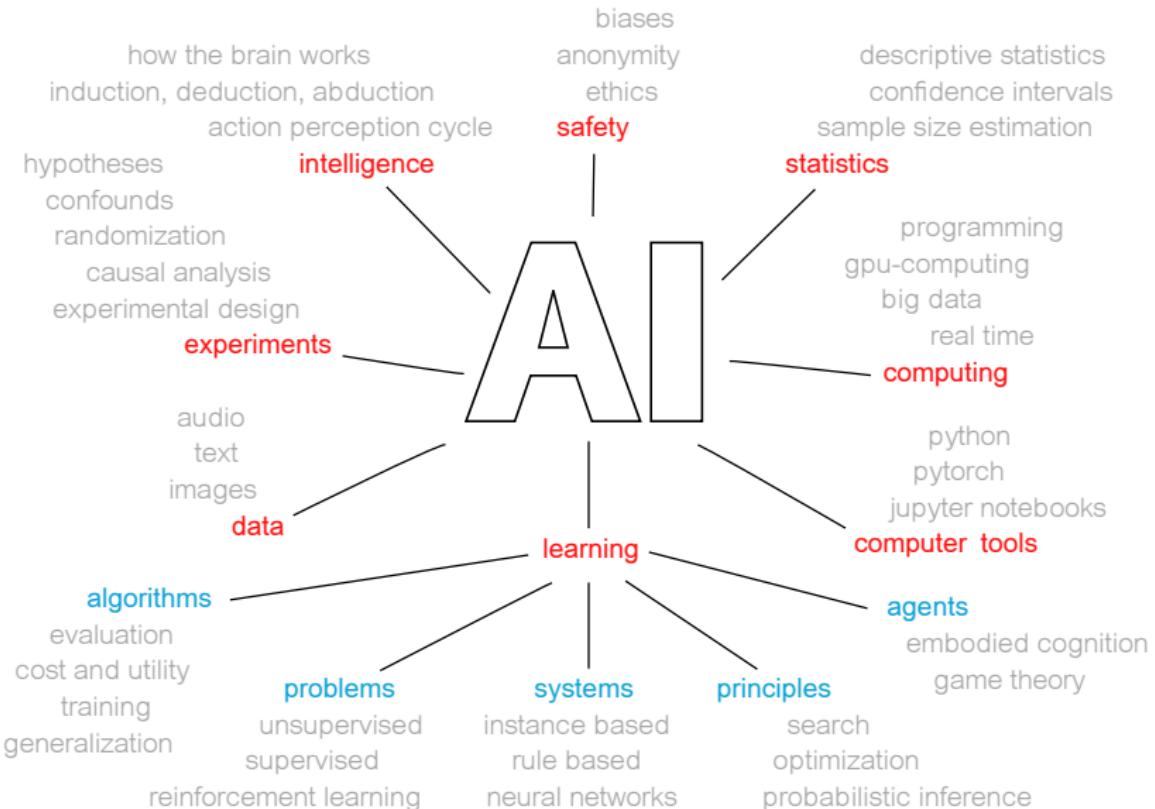
- I Understand the concepts and definitions, and know their application. Reason about the concepts in the context of an example. Use correct technical terminology.
- II As above plus: Read, manipulate, and work with technical definitions and expressions (mathematical and Python code). Carry out practical computations. Interpret and evaluate results.

Welcome to Introduction to Intelligent Systems

Welcome to Introduction to Intelligent Systems

Who am I?

Course overview



Course objectives

To provide the participants a basic knowledge of

- defining aspects of intelligent systems,
- application of intelligent systems in image, audio, text and game data,
- computational and collaborative tools for artificial intelligence.

Intelligent systems on DTU Learn

`learn.inside.dtu.dk`

Exercise: Examples of AI

Find the most significant and profound example of (one of) the following topics

A. Superhuman AI Artificial intelligence that outperforms humans

<https://finnaarupnielsen.wordpress.com/2015/03/15/status-on-human-vs-machines>

B. Emulating human creativity AI that emulates human creativity

<http://www.thepaintingfool.com>

C. Intelligent animal behavior Animal behavior

[https://www.thespruce.com/understanding-bird-intelligence-386440,](https://www.thespruce.com/understanding-bird-intelligence-386440)

https://en.wikipedia.org/wiki/Dog_intelligence

D. Augmented intelligence Enhancing human performance using AI

[https://www.technologyreview.com/s/603951/this-is-your-brain-on-gps-navigation,](https://www.technologyreview.com/s/603951/this-is-your-brain-on-gps-navigation)

<https://deepmind.com/blog/2017-deepminds-year-review>

Prepare to present your example with *two sentences*:

1. Describe the example briefly.
2. Describe why you think this example is significant.

Expectations survey

Survey

Please fill out the *Expectations survey*, including

1. Goals and expectations
2. Self-assessment
3. Mini IQ-test

Find links to the surveys on DTU Learn

The historical origin of AI

Background

Cybernetics (Wiener, 1940s) Study of feedback, control and communication in the animal and the machine.

Cellular automata (Ulam and von Neumann, 1940s) Iteration of very simple rules can produce intricate and complex patterns.

You insist...

John von Neumann (1950)

“You insist that there is something a machine cannot do. If you will tell me precisely what it is that a machine cannot do, then I can always make a machine which will do just that!”



Universal machines

Alan Turing

"This special property of digital computers, that they can mimic any discrete state machine, is described by saying that they are universal machines. The existence of machines with this property has the important consequence that, considerations of speed apart, it is unnecessary to design various new machines to do various computing processes. They can all be done with one digital computer, suitably programmed for each case. It will be seen that as a consequence of this all digital computers are in a same equivalent."

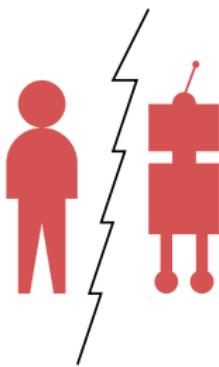


Turing's test (1950)

- Alan Turing speculated about the possibility of creating thinking machines.
- Since “thinking” is difficult to define, he devised his famous Turing Test.

If a machine could carry on a conversation (over a teleprinter) that was indistinguishable from a conversation with a human being, then it was reasonable to say that the machine was “thinking”.

- Turing argues that a thinking machine is at least plausible.



Exercise: Turings objections

Is it possible to create a *thinking machine*? Turing outlined 9 objections:

Theological Only God can create thinking machines.

Heads in the sand The consequences of thinking machines are too dreadful.

Mathematical Fundamental limitations to the power of state machines.

Consciousness The machine can merely imitate—it cannot feel.

Disabilities Okay you can do all these things, but you can't do X...

Determinism The machine can only do what we tell it.

Discrete The human nervous system is continuous.

Informality We cannot define rules for every conceivable circumstance.

Extra-sensory As-of-yet undiscovered laws of physics govern thinking.

Discuss in groups

- Do you believe it is possible to create a thinking machine?
- Which of these objections you agree/disagree with
- Can you come up with any other objections?

Prepare to present your argument for or against thinking machines.

Dartmouth Summer Research Project on Artificial Intelligence

John McCarthy (1956) coined the term “artificial intelligence”

We propose that a 2 month, 10 man study of artificial intelligence be carried out during the summer of 1956 at Dartmouth College in Hanover, New Hampshire. The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. An attempt will be made to find how to make machines use language, form abstractions and concepts, solve kinds of problems now reserved for humans, and improve themselves. We think that a significant advance can be made in one or more of these problems if a carefully selected group of scientists work on it together for a summer.

The following are some aspects of the artificial intelligence problem:

1. *Automatic computers*
2. *How can a computer be programmed to use a language*
3. *Neuron nets*
4. *Theory of the size of a calculation*
5. *Self-improvement*
6. *Abstractions*
7. *Randomness and creativity*

Intelligence and the brain

What is intelligence

Can you come up with the 3 most important factors that are required, in order to define a system (computer, human, animal, etc.) as intelligent?

A definition of intelligence

Interact Perceive the world and take actions which affect the world.

Learn Form mental representations of percepts and environment.
Remember and generalize from past events.

Achieve Have goals and act in a way that helps achieve them.

A definition of intelligence

The ability to learn and apply individual knowledge and skills to achieve goals.

Artificial intelligence and data

Intelligence The ability to learn and apply individual knowledge and skills to achieve goals.

Artificial intel. The science of creating machines with intelligent behavior.

Data Information, facts, and statistics collected and stored in a computer.

Human intelligence

General intelligence factor

- Spearman (1904) argues that there exists a general mental capacity, the g factor, that influences performance on all cognitive tasks.
- According to this view, human intelligence can be measured as a single number, such as an IQ score.

Multiple intelligences

- Gardner's (1983) eight intelligences
 - musical-rhythmic
 - visual-spatial
 - verbal-linguistic
 - logical-mathematical
 - bodily-kinesthetic
 - interpersonal
 - intrapersonal
 - naturalistic
- Sternberg's (1985) triarchic theory of intelligence
 - componential-analytic
 - experiential-creative
 - practical-contextual

Why do we have a brain?

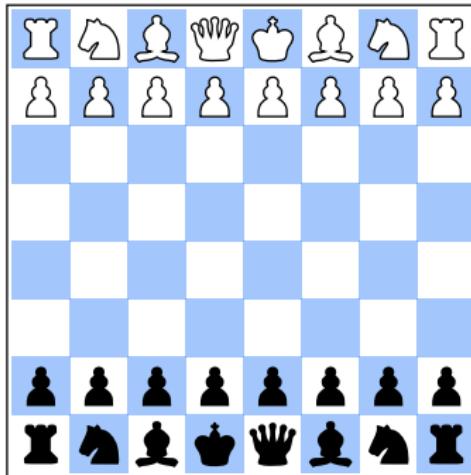
According to Daniel Wolpert

Not all species on the planet have brains, so if we want to know what the brain is for, let's think about why we evolved one. Now you may reason that we have one to perceive the world, or to think, and that is completely wrong.

If you think about this question for any length of time, it's blindingly obvious why we have a brain. We have a brain for one reason, and one reason only, and that is to produce adaptable and complex movements. There is no other reason to have a brain.

Think about it: Movement is the only way you have of affecting the world around you.

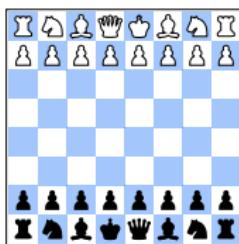
Which task is most difficult?



Determine what piece
to move where

Physically pick up
a piece and place it

Which task is most difficult?



Determine what piece
to move where

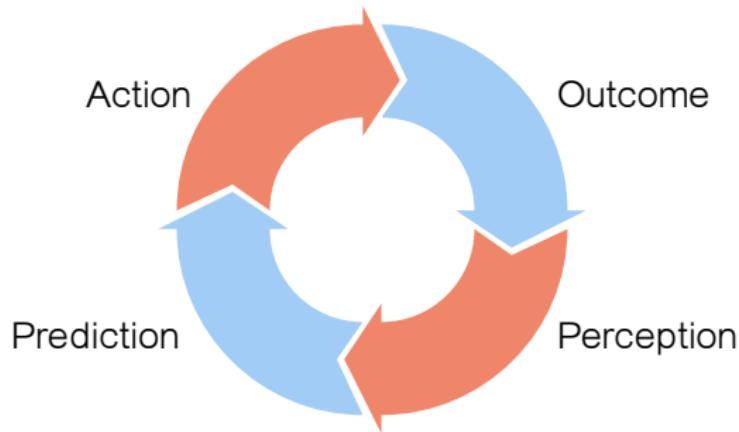
Physically pick up
a piece and place it

Decide which piece to move where:

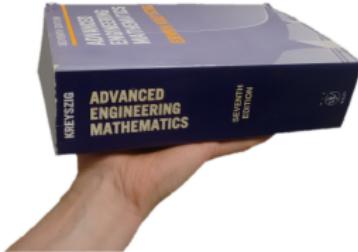
Simple algorithm: Consider all possible moves to the end of the game. Choose the one that makes you win.

Physically pick up a piece and place it:
Algorithm: ???

The action-perception cycle



Heavy book experiment



Try this experiment with a friend. Pick up a heavy object, like a large book, and hold it underneath with your left hand. If you now use your right hand to lift the book off of your left hand, you'll notice that your left hand stays steady. However, if your friend lifts the book off of your hand, your brain will not be able to predict exactly when that will happen. Your left hand will rise up just a little after the book is gone, until your brain realizes it no longer needs to compensate for the book's weight. When your own movement removed the book, your brain was able to cancel out that action and predict with certainty when to adjust your left hand's support.

Methods of reasoning

Deduction, induction, and abduction

Deduction Reasoning from general premises to specific conclusions. Infer what is necessarily true based on the premises.

Can be disproven by finding a flaw in the premise.

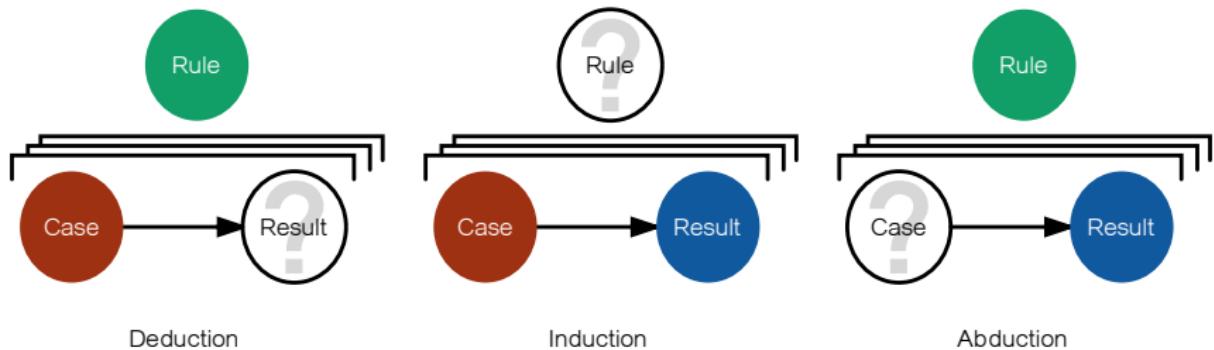
Induction Reasoning from specific facts to general rules. Infer what are likely conclusions based on statistics.

Can be disproven by a counterexample.

Abduction Reasoning about the simplest or most likely explanation.

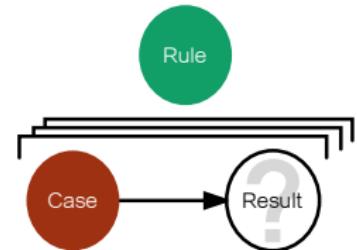
Can be disproven by finding a flaw in the explanation.

Deduction, induction, and abduction



Examples of deduction

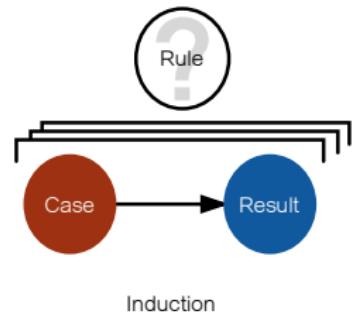
Rule All humans are mortal
Case Socrates is a human
Result Socrates is mortal



Deduction
Reasoning from general premises to specific conclusions. Infer what is necessarily true based on the premises.
Can be disproven by finding a flaw in the premise.

Example of induction

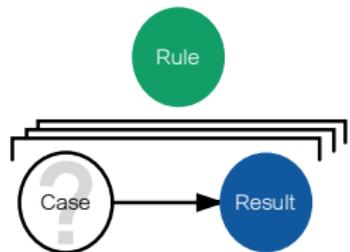
Case My brother, my father, my mother
Result Tall, tall, tall
Rule All my relatives are tall



Induction
Reasoning from specific facts to general rules.
Infer what are likely conclusions based on statistics.
Can be disproven by a counterexample.

Example of abduction

- Rule** The library has many books
- Result** I have a book in my hand
- Case** The book was probably taken from the library



Abduction
Reasoning about the simplest or most likely explanation.
Can be disproven by finding a flaw in the explanation.

Deduction, induction, or abduction?

Q1 In the following number sequence, the next number is always equal to the sum of the two previous numbers. What are the missing numbers?

- a) 4 - 9 b) 4 - 8 c) 5 - 9 d) 5 - 8

Q2 What are the missing numbers in this sequence?

- $$1 - 3 - 6 - 10 - 15 - 21 - ? - ?$$

a) 27 - 34 b) 28 - 36 c) 27 - 36 d) 28 - 34

Q4 What are the missing numbers in this sequence?

Hint: the answer is not 1, so you really have to search from A to Z to find the solution.

- a) 2 b) 3 c) 4 d) 5

Q3 The following numbers are listed in random order. What are the missing numbers?

- 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1
1 - 1 - 2 - 1 - 1 - 1 - 1 - 2
1 - 1 - 1 - 2 - 1 - 1 - 1 - 1
1 - 1 - 1 - 1 - 1 - 1 - 1 - 3
1 - 1 - 1 - 1 - 1 - 1 - 1 - 1
1 - 1 - 1 - 1 - 1 - 1 - 1 - 1
1 - 1 - 1 - 1 - 1 - 1 - ? - ?

- a) 1 - 1 b) 2 - 3 c) 3 - 2 d) 4 - 2

Demo of image recognition

Python installation instructions

Python installation instructions

1. Install Python

Get version 3.11 from python.org

2. Install VSCode

Install the extensions Python and Jupyter

3. Create a virtual environment called IntelligentSystems

```
python -m venv IntelligentSystems
```

4. Activate the virtual environment

```
Windows powershell IntelligentSystems\Scripts\Activate.ps1  
MacOS/Linux source IntelligentSystems/bin/activate
```

5. Install required packages with pip

```
pip install -r requirements.txt
```

6. Select the virtual environment as interpreter in VSCode

Python script Click on the interpreter version in the bottom right corner

Jupyter notebook Click on the kernel version in the top right corner

and choose IntelligentSystems

Detailed instructions: Go to pythonsupport.dtu.dk and find course **02461**.

Tasks

Tasks

Today

1. Install Python (according to instructions)
2. Try out the following notebooks and programs in VSCode
 - The notebook *01-Introduction.ipynb*
 - The script *01-ImageRecognition.py*
 - The game Lunar Lander: *Play_LunarLander.py*
3. Today's feedback group
 - Jakob Lauge Reeh
 - David Lindahl
 - Philip Thinggaard
 - Poul Skov

For next time

1. Read the notes “Introduction” + Solve problems
2. Read the notes “Statistics” + Solve problems

Introduction to intelligent systems

Statistics

Mikkel N. Schmidt

Technical University of Denmark,
DTU Compute, Department of Applied Mathematics and Computer Science.

Overview

- ① Descriptive statistics
- ② Probability distributions
- ③ Population and sample
- ④ Central limit theorem
- ⑤ Standard error of the mean
- ⑥ Confidence intervals
- ⑦ Tasks

Feedback group

- Magnus Alexander Mollatt van Capel
- Rasmus Johansen Rieneck
- Christian Rahbæk Warburg
- Clara Louise Brodt

Learning objectives

- I Descriptive and inferential statistics: Population, sample, statistic, parameters, estimator.
 - II Population mean and standard deviation.
 - II Sample estimate of mean and standard deviation.
 - II Confidence interval for mean and proportion.
 - II Sample size estimate for mean and proportion.
-
- I Understand the concepts and definitions, and know their application. Reason about the concepts in the context of an example. Use correct technical terminology.
 - II As above plus: Read, manipulate, and work with technical definitions and expressions (mathematical and Python code). Carry out practical computations. Interpret and evaluate results.

Descriptive statistics

Mean and variance of a population

Population mean Average value of the population

$$\mu_x = \frac{1}{N} \sum_{i=1}^N x_i$$

Population variance Average squared distance to the mean

$$\sigma_x^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu_x)^2$$

Population standard deviation Square root of variance

Exercise: Population mean and variance

Consider a population of $N = 3$ observations.

$$x = \{1, 4, 10\}$$

- What is the population mean μ_x and variance σ_x^2 ?

Definitions

$$\mu_x = \frac{1}{N} \sum_{i=1}^N x_i$$

$$\sigma_x^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu_x)^2$$

Exercise: Population mean and variance

Consider a population of $N = 3$ observations.

$$x = \{1, 4, 10\}$$

Definitions

$$\mu_x = \frac{1}{N} \sum_{i=1}^N x_i$$

- What is the population mean μ_x and variance σ_x^2 ?

$$\sigma_x^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu_x)^2$$

Solution

$$\mu_x = \frac{1}{3}(1 + 4 + 10) = 5$$

$$\sigma_x^2 = \frac{1}{3}((1 - 5)^2 + (4 - 5)^2 + (10 - 5)^2) = 14$$

Probability distributions

Probability distributions

Probability distribution

- Mathematical function that gives the probability of each possible outcome in an experiment
- Describes a random phenomenon in terms of probabilities of events
- Can be used as an infinite population

Probability distributions

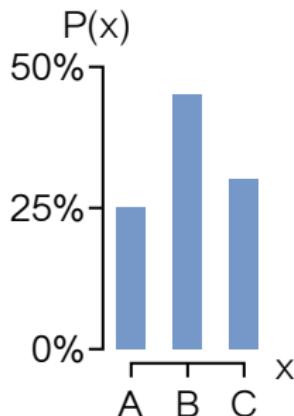
Probability distribution

- Mathematical function that gives the probability of each possible outcome in an experiment
- Describes a random phenomenon in terms of probabilities of events
- Can be used as an infinite population

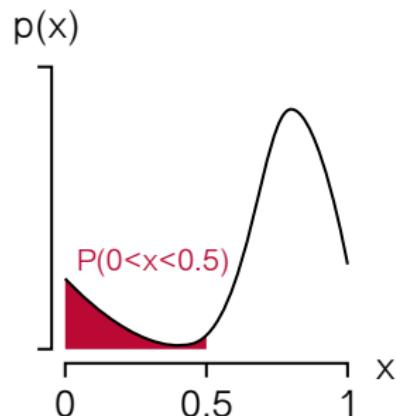
- | | |
|----------------------------|--|
| Discrete outcomes | ■ Probability mass function provides probability of each outcome. |
| Continuous outcomes | <ul style="list-style-type: none">■ Probability density function (PDF) provides <i>relative</i> probability of each outcome.■ The probability of any particular outcome is zero.■ The probability of an outcome within some range is the area under the PDF curve. |

Probability distributions

Probability mass function



Probability density function



Mean and variance of a population / distribution

Finite population

- Sum over all values in population
- Weigh each equally by $1/N$

$$\mu_x = \frac{1}{N} \sum_{i=1}^N x_i$$

$$\sigma_x^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu_x)^2$$

Mean and variance of a population / distribution

Finite population

- Sum over all values in population
- Weigh each equally by $1/N$

$$\mu_x = \frac{1}{N} \sum_{i=1}^N x_i$$

$$\sigma_x^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu_x)^2 \quad \sigma_x^2 = \sum_{i=1}^K P(x_k) \cdot (x_k - \mu_x)^2$$

Discrete distribution

- Sum over all K possible outcomes
- Weigh each by their probability

$$\mu_x = \sum_{k=1}^K P(x_k) \cdot x_k$$

Mean and variance of a population / distribution

Finite population

- Sum over all values in population
- Weigh each equally by $1/N$

$$\mu_x = \frac{1}{N} \sum_{i=1}^N x_i$$

$$\sigma_x^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu_x)^2$$

Discrete distribution

- Sum over all K possible outcomes
- Weigh each by their probability

$$\mu_x = \sum_{k=1}^K P(x_k) \cdot x_k$$

$$\sigma_x^2 = \sum_{i=1}^K P(x_k) \cdot (x_k - \mu_x)^2$$

Continuous distribution

- Integral over all possible outcomes
- Weigh each by their probability density

$$\mu_x = \int_{-\infty}^{\infty} p(x) \cdot x \cdot dx$$

$$\sigma_x^2 = \int_{-\infty}^{\infty} p(x) \cdot (x - \mu_x)^2 \cdot dx$$

Population and sample

Population and sample

Population Entire set of entities under study

- Can be a hypothetical population
- Typically impossible to survey/measure entire population
- May be infinite

Sample Random subset of entire population

- When studying a population using a sample, we may obtain slightly different answers depending on the sample

Terms used in inferential statistics

Population The entire set of items of interest.

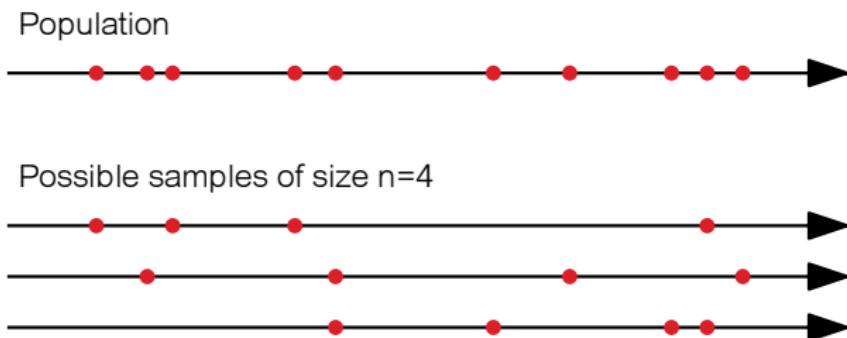
Sample A subset of the population.

Statistic A numerical value computed from the sample.

Parameter A characteristic of the population under study.

Estimator A statistic used to estimate a parameter.

Example: Population and sample



Sample estimate of mean and variance

Sample estimate of mean Average value of the sample

$$m_x = \frac{1}{n} \sum_{i=1}^n x_i$$

Sample estimate of variance Estimate of average squared distance to the mean. Notice, we divide by $n - 1$, as opposed to n

$$s_x^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - m_x)^2$$

Why divide by $n - 1$?

Sample estimate of variance

$$s_x^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - m_x)^2$$

Intuition

- Sample mean is always between the lowest and highest value in the sample—the population mean need not be
- Variance observed in a sample tends to underestimate the population variance
- We would like the *average of all possible sample variances* to equal the population variance

Exercise: Why divide by $n - 1$?

Consider a population of $N = 3$ observations

$$x = \{1, 4, 10\}$$

with population mean and variance

$$\mu_x = 5 \quad \sigma_x^2 = 14$$

- List all possible ordered samples with replacement of size $n = 2$.
(Hint: There are 9 such possible samples)

Exercise: Why divide by $n - 1$?

Consider a population of $N = 3$ observations

$$x = \{1, 4, 10\}$$

with population mean and variance

$$\mu_x = 5 \quad \sigma_x^2 = 14$$

- List all possible ordered samples with replacement of size $n = 2$.
(Hint: There are 9 such possible samples)

Solution

The 9 possible samples are

$$\{1, 1\}, \{1, 4\}, \{1, 10\}, \{4, 1\}, \{4, 4\}, \{4, 10\}, \{10, 1\}, \{10, 4\}, \{10, 10\}$$

Exercise: Why divide by $n - 1$? (II)

Consider a population of $N = 3$ observations

$$x = \{1, 4, 10\}$$

with population mean and variance

$$\mu_x = 5 \quad \sigma_x^2 = 14$$

Sample estimate

$$m_x = \frac{1}{n} \sum_{i=1}^n x_i$$

$$s_x^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - m_x)^2$$

- Compute the sample estimate of the mean and variance, m_x and s_{n-1}^2 for each possible sample

$$\{1, 1\}, \{1, 4\}, \{1, 10\}, \{4, 1\}, \{4, 4\}, \{4, 10\}, \{10, 1\}, \{10, 4\}, \{10, 10\}$$

- What is the average sample estimate of the mean and variance (averaged over all possible samples)?

Exercise: Why divide by $n - 1$? (II)

Solution

Sample	m_x	s_x^2
{1, 1}	1	0
{1, 4}	2.5	2.25
{1, 10}	5.5	30.25
{4, 1}	2.5	2.25
{4, 4}	4	0
{4, 10}	7	49
{10, 1}	5.5	30.25
{10, 4}	7	49
{10, 10}	10	0

Exercise: Why divide by $n - 1$? (II)

Solution

Sample	m_x	s_x^2
{1, 1}	$\frac{1+1}{2} = 1$	$\frac{(1-1)^2 + (1-1)^2}{2-1} = 0$
{1, 4}		
{1, 10}		
{4, 1}		
{4, 4}		
{4, 10}		
{10, 1}		
{10, 4}		
{10, 10}		

Exercise: Why divide by $n - 1$? (II)

Solution

Sample	m_x	s_x^2
{1, 1}	$\frac{1+1}{2} = 1$	$\frac{(1-1)^2 + (1-1)^2}{2-1} = 0$
{1, 4}	$\frac{1+4}{2} = 2.5$	$\frac{(1-2.5)^2 + (4-2.5)^2}{2-1} = 4.5$
{1, 10}		
{4, 1}		
{4, 4}		
{4, 10}		
{10, 1}		
{10, 4}		
{10, 10}		

Exercise: Why divide by $n - 1$? (II)

Solution

Sample	m_x	s_x^2
{1, 1}	$\frac{1+1}{2} = 1$	$\frac{(1-1)^2 + (1-1)^2}{2-1} = 0$
{1, 4}	$\frac{1+4}{2} = 2.5$	$\frac{(1-2.5)^2 + (4-2.5)^2}{2-1} = 4.5$
{1, 10}	$\frac{1+10}{2} = 5.5$	$\frac{(1-5.5)^2 + (10-5.5)^2}{2-1} = 40.5$
{4, 1}		
{4, 4}		
{4, 10}		
{10, 1}		
{10, 4}		
{10, 10}		

Exercise: Why divide by $n - 1$? (II)

Solution

Sample	m_x	s_x^2
{1, 1}	$\frac{1+1}{2} = 1$	$\frac{(1-1)^2 + (1-1)^2}{2-1} = 0$
{1, 4}	$\frac{1+4}{2} = 2.5$	$\frac{(1-2.5)^2 + (4-2.5)^2}{2-1} = 4.5$
{1, 10}	$\frac{1+10}{2} = 5.5$	$\frac{(1-5.5)^2 + (10-5.5)^2}{2-1} = 40.5$
{4, 1}	2.5	4.5
{4, 4}	4	0
{4, 10}	7	18
{10, 1}	5.5	40.5
{10, 4}	7	18
{10, 10}	10	0

Exercise: Why divide by $n - 1$? (II)

Solution

Sample	m_x	s_x^2
{1, 1}	$\frac{1+1}{2} = 1$	$\frac{(1-1)^2 + (1-1)^2}{2-1} = 0$
{1, 4}	$\frac{1+4}{2} = 2.5$	$\frac{(1-2.5)^2 + (4-2.5)^2}{2-1} = 4.5$
{1, 10}	$\frac{1+10}{2} = 5.5$	$\frac{(1-5.5)^2 + (10-5.5)^2}{2-1} = 40.5$
{4, 1}	2.5	4.5
{4, 4}	4	0
{4, 10}	7	18
{10, 1}	5.5	40.5
{10, 4}	7	18
{10, 10}	10	0

Average s_x^2 over all possible samples

$$\text{avg}(m_x) = \frac{1 + 2.5 + 5.5 + 2.5 + 4 + 7 + 5.5 + 7 + 10}{9} = \frac{45}{9} = 5 = \mu_x$$

$$\text{avg}(s_x^2) = \frac{0 + 4.5 + 40.5 + 4.5 + 0 + 18 + 40.5 + 18 + 0}{9} = \frac{126}{9} = 14 = \sigma_x^2$$

Sampling distribution

The *sampling distribution* is the distribution of sample means that occurs as we draw samples (of size n) from the population

Example We want to study how many kilometers DTU students commute every morning

- We cannot ask all students in the population, so we randomly choose a sample of $n = 10$ persons
- We then compute the mean distance travelled for the 10 persons

Sampling If we then repeat this 5 times, each time asking another sample of 10 persons, we might get the following results

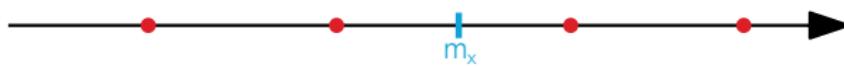
Sample number	Mean number of kilometers
1	5.5
2	7.6
3	2.1
4	6.2
5	1.9

Example: Population and sample means

Population



Possible samples of size n=4



Central limit theorem

Central limit

What happens to the sampling distribution of the mean, as we increase the sample size n ?

- The sample tends to follow a normal distribution
- The sample mean tends to cluster closer around the population mean

Demo: Central limit theorem

Demo: `02-CentralLimitTheorem.ipynb` notebook

Standard error of the mean

Standard error of the mean

We have seen that, as we increase the sample size

- The sample tends to follow a normal distribution
- The sample mean tends to cluster closer around the population mean

If we know the population variance and the sample size n , is there a way to predict the variance of the sampling distribution?

Variance of the sample mean

$$\sigma_{m_x}^2 = \frac{\sigma^2}{n}$$

Standard error of the mean

$$\sigma_{m_x} = \frac{\sigma}{\sqrt{n}}$$

Standard error of the mean: Proof

Variance of a sum of independent random variables

$$\text{var}(x_1 + x_2 + \cdots + x_n) = \text{var}(x_1) + \text{var}(x_2) + \cdots + \text{var}(x_n)$$

Variance scales quadratically

$$\text{var}(a \cdot x) = a^2 \cdot \text{var}(x)$$

Variance of the sample mean

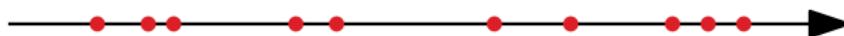
$$\sigma_{m_x}^2 = \text{var}(m_x) = \text{var}\left(\frac{1}{n} \sum_{i=1}^n x_i\right) = \frac{1}{n^2} \text{var}\left(\sum_{i=1}^n x_i\right) = \frac{\sigma^2}{n}$$

Using the central limit theorem

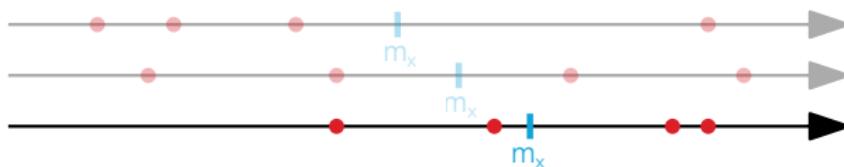
- We know the sample means approximately follow a normal distribution (when the sample size is moderately large)
- We can calculate the mean and variance of that distribution (it is simply $\mu_{m_x} = \mu$ and $\sigma_{m_x}^2 = \frac{\sigma^2}{n}$)
- We can use this to predict which values of the sample mean are most likely

Example: Population and sample

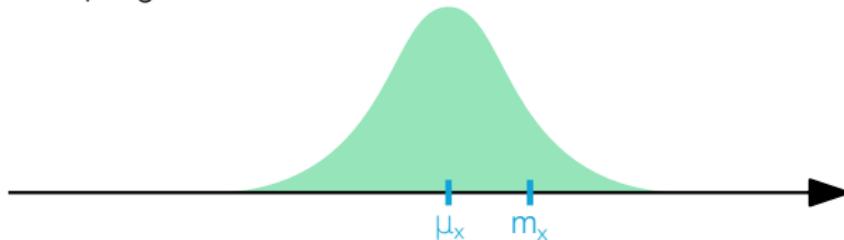
Population



Possible samples of size $n=4$



Sampling distribution of the mean



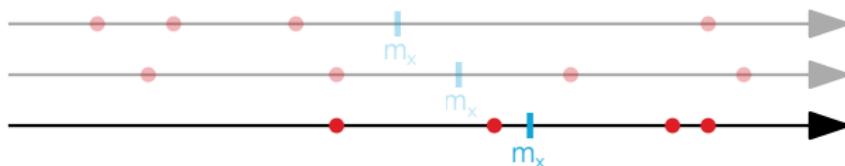
Confidence intervals

Example: Population and sample

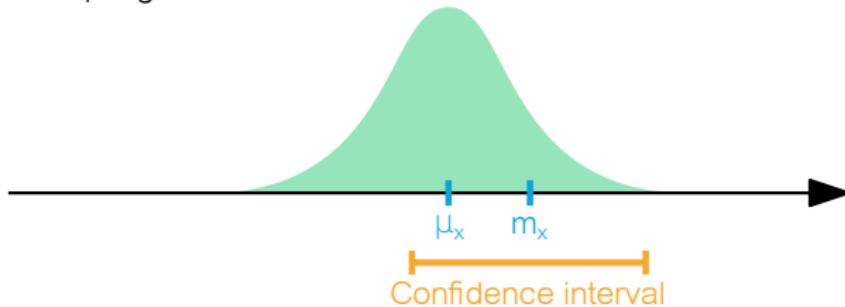
Population



Possible samples of size $n=4$



Sampling distribution of the mean



Confidence interval

Confidence interval

point estimate \pm margin of error

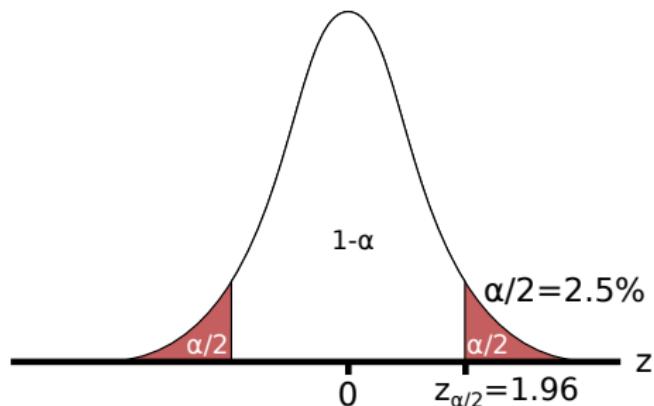
$$m_x \pm \underbrace{z_{\alpha/2}}_{\text{critical value}} \cdot \underbrace{\sqrt{\frac{\sigma^2}{n}}}_{\text{standard error}}$$

Confidence interval for mean We want to estimate the population mean μ

- Sample n observations
- Compute the sample mean $m_x = \frac{1}{n} \sum_{i=1}^n x_i$
- Choose confidence level, e.g. $1 - \alpha = 95\%$, and look up critical value
- Compute standard error and multiply by critical value

Critical value

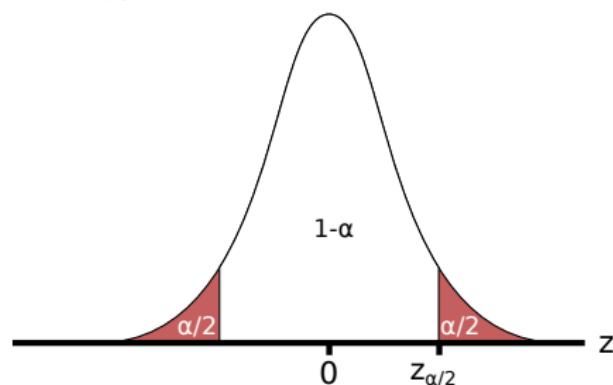
Example: Critical value for $1 - \alpha = 95\%$ interval



Look up in a table or compute in Python using `scipy.stats.norm.ppf`

Table of critical values

Central area	Tail area	Critical value
$1 - \alpha$	$\alpha/2$	$z_{\alpha/2}$
50%	0.25	0.67
90%	0.05	1.64
95%	0.025	1.96
99%	0.005	2.58



Interpretation of confidence interval

- Expresses error due to random sampling
- A larger sample size gives a smaller margin of error
- Is defined at a desired confidence level, e.g. $1 - \alpha = 95\%$
- 95% confidence means that out of 100 samples, we expect that 95 times the true population parameter is within the margin of error

Exercise: Mean and variance of a 6-sided dice

Mean and standard deviation of a discrete distribution

- Sum over all possible outcomes
- Weigh each by their probability

$$\mu_x = \sum_{k=1}^K P(x_k) \cdot x_k \quad \sigma_x^2 = \sum_{k=1}^K P(x_k) \cdot (x_k - \mu)^2$$

- What is μ_x and σ_x^2 for a normal 6-sided dice?

$$K = 6, \quad x_1 = 1, x_2 = 2, \dots, x_6 = 6, \quad P(x_1) = P(x_2) = \dots = P(x_6) = \frac{1}{6}$$

Exercise: Mean and variance of a 6-sided dice

Mean and standard deviation of a discrete distribution

- Sum over all possible outcomes
- Weigh each by their probability

$$\mu_x = \sum_{k=1}^K P(x_k) \cdot x_k \quad \sigma_x^2 = \sum_{k=1}^K P(x_k) \cdot (x_k - \mu)^2$$

- What is μ_x and σ_x^2 for a normal 6-sided dice?

$$K = 6, \quad x_1 = 1, x_2 = 2, \dots, x_6 = 6, \quad P(x_1) = P(x_2) = \dots = P(x_6) = \frac{1}{6}$$

Solution

$$\mu_x = \frac{1}{6} \cdot 1 + \frac{1}{6} \cdot 2 + \frac{1}{6} \cdot 3 + \frac{1}{6} \cdot 4 + \frac{1}{6} \cdot 5 + \frac{1}{6} \cdot 6 = \frac{21}{6} = 3.5$$

$$\begin{aligned}\sigma_x^2 &= \frac{1}{6} \left((1-3.5)^2 + (2-3.5)^2 + (3-3.5)^2 + (4-3.5)^2 + (5-3.5)^2 + (6-3.5)^2 \right) \\ &= \frac{1}{6} (6.25 + 2.25 + 0.25 + 0.25 + 2.25 + 6.25) \approx 2.917\end{aligned}$$

Exercise: Confidence interval of 10 dice throws

- Throw a 6-side dice 10 times and record the results
(e.g. use www.random.org/dice)
- Compute the 50% confidence interval for the mean
Express it as a range [low, high]

Confidence interval

$$m_x \pm \underbrace{z_{\alpha/2}}_{\text{critical value}} \cdot \underbrace{\sqrt{\frac{s_x^2}{n}}}_{\text{standard error}}$$

$(z_{0.25} = 0.67)$

Exercise: Confidence interval of 10 dice throws

- Throw a 6-side dice 10 times and record the results
(e.g. use www.random.org/dice)
- Compute the 50% confidence interval for the mean
Express it as a range [low, high]

Confidence interval

$$m_x \pm \underbrace{z_{\alpha/2}}_{\text{critical value}} \cdot \underbrace{\sqrt{\frac{s_x^2}{n}}}_{\text{standard error}}$$
$$(z_{0.25} = 0.67)$$

Solution example



Exercise: Confidence interval of 10 dice throws

- Throw a 6-side dice 10 times and record the results
(e.g. use www.random.org/dice)
- Compute the 50% confidence interval for the mean
Express it as a range [low, high]

Confidence interval

$$m_x \pm \underbrace{z_{\alpha/2}}_{\text{critical value}} \cdot \underbrace{\sqrt{\frac{s_x^2}{n}}}_{\text{standard error}}$$
$$(z_{0.25} = 0.67)$$

Solution example



$$m_x = \frac{1}{10}(1 + 6 + 5 + 6 + 1 + 6 + 3 + 1 + 3 + 1) = \frac{33}{10} = 3.3$$

$$s_x^2 = \frac{1}{10-1} ((1 - 3.3)^2 + (6 - 3.3)^2 + (5 - 3.3)^2 + \dots + (1 - 1)^2) \approx 5.12$$

Confidence interval

$$m_x \pm z_{\alpha/2} \cdot \sqrt{\frac{s_x^2}{n}} = 3.3 \pm 0.67 \cdot \sqrt{\frac{5.12}{10}} = 3.3 \pm 0.48$$
$$[2.82, 3.78]$$

The population mean is 3.5 and we expect 50% of the computed confidence intervals to include it

Estimating a proportion

Bernoulli distribution

$$P(k) = \pi^k \cdot (1 - \pi)^{1-k} = \begin{cases} (1 - \pi), & k = 0 \\ \pi, & k = 1 \end{cases}$$

What is the mean and variance?

Estimating a proportion

Bernoulli distribution

$$P(k) = \pi^k \cdot (1 - \pi)^{1-k} = \begin{cases} (1 - \pi), & k = 0 \\ \pi, & k = 1 \end{cases}$$

What is the mean and variance?

Mean

$$\begin{aligned}\mu_k &= \sum_{k \in \{0,1\}} P(k) \cdot k \\ &= \underbrace{(1 - \pi) \cdot 0}_{k=0} + \underbrace{\pi \cdot 1}_{k=1} = \pi\end{aligned}$$

Estimating a proportion

Bernoulli distribution

$$P(k) = \pi^k \cdot (1 - \pi)^{1-k} = \begin{cases} (1 - \pi), & k = 0 \\ \pi, & k = 1 \end{cases}$$

What is the mean and variance?

Mean

$$\begin{aligned}\mu_k &= \sum_{k \in \{0,1\}} P(k) \cdot k \\ &= \underbrace{(1 - \pi) \cdot 0}_{k=0} + \underbrace{\pi \cdot 1}_{k=1} = \pi\end{aligned}$$

Variance

$$\begin{aligned}\sigma_k^2 &= \sum_{k \in \{0,1\}} (k - \mu)^2 \cdot P(k) \\ &= \underbrace{(0 - \pi)^2 \cdot (1 - \pi)}_{k=0} + \underbrace{(1 - \pi)^2 \cdot \pi}_{k=1} \\ &= \pi^2(1 - \pi) + (1 + \pi^2 - 2\pi)\pi \\ &= \pi^2 - \pi^3 + \pi + \pi^3 - 2\pi^2 = \pi - \pi^2 = \pi(1 - \pi)\end{aligned}$$

Confidence interval for proportion

Confidence interval for proportion

point estimate \pm margin of error

$$p \pm \underbrace{z_{\alpha/2}}_{\text{critical value}} \cdot \underbrace{\sqrt{\frac{p(1-p)}{n}}}_{\text{standard error}}$$

Confidence interval for proportion

- Estimate unknown true proportion parameter, π
- Sample n observations
- Compute the sample proportion $p = \frac{\#\text{correct}}{n}$
- Choose confidence level, e.g. $1 - \alpha = 95\%$, and look up critical value.
- Compute standard error and multiply by critical value

Sample size for proportion

- The equation for the confidence interval can be solved for the sample size
- This gives a formula for the required sample size to give a desired margin of error

Sample size for proportion

$$n = z_{\alpha/2}^2 \frac{p(1-p)}{e^2}$$

n: Required sample size

p: Expected population proportion

Unknown, but we can substitute best guess or worst case $p = 50\%$

e: Desired margin of error

α : Significance level

Tasks

Tasks

Tasks today

- Start working on lab report 1: Read description on DTU Learn
- Today's feedback group
 - Magnus Alexander Mollatt van Capel
 - Rasmus Johansen Rieneck
 - Christian Rahbæk Warburg
 - Clara Louise Brodt

Lab report hand in

- Lab 1: Image recognition (Deadline: Thursday 14 September 20:00)

Next time

- Algorithms. Preparation: Read the note “Algorithms” + solve problems

Introduction to intelligent systems

Algorithms

Mikkel N. Schmidt

Technical University of Denmark,
DTU Compute, Department of Applied Mathematics and Computer Science.

Overview

① Algorithms

② Algorithmic complexity

③ Divide and conquer / recursion

④ Software tools

⑤ Tasks

Feedback group

- Viet Hoang Nguyen
- Lukas Peter Dyhr
- Philip Kierkegaard
- Ali Mohammed Fathi Afif

Learning objectives

- I Algorithmic complexity.
 - I Understand an algorithm from description or Python code.
 - II Time complexity function.
 - II Best, average and worst case complexity.
 - II Big-O notation.

- II Understand the concepts and definitions, and know their application. Reason about the concepts in the context of an example. Use correct technical terminology.
- II As above plus: Read, manipulate, and work with technical definitions and expressions (mathematical and Python code). Carry out practical computations. Interpret and evaluate results.

Algorithms

What is an algorithm

- Unambiguous specification of how to solve a problem
- Expressed in a well-defined formal language
- Starts from initial state and input
- Proceeds through a finite number of steps
- Eventually terminates and produces an output

Levels of description

High level description Describes algorithm in normal language, ignoring implementation detail.

Implementation description Detailed description of exactly which actions must be performed by the computer.

Example: Finding the largest number in a list

1. Assume the first number is the largest number
2. For each remaining number in the set: if this number is larger than the current largest number, consider this number to be the largest number
3. When there are no numbers left in the set to iterate over, consider the current largest number to be the largest number of the list

Example list

99 83 125 12 5 256 31 192

Exercise: An algorithm for sorting

1. Write down the numbers below on 8 small pieces of paper
2. Lay them in a random sequence on the table
3. Sort them starting with the smallest, and take notice of exactly which procedure you use
4. Write down a high level description of your sorting algorithm
5. Randomize the order of the numbers again, and follow your written procedure to the letter to sort the numbers again

Example list

99 83 125 12 5 256 31 192

Prepare to present your algorithm to the class

A simple sorting algorithm

1. Find the smallest number on the list
2. Remove the smallest number from the list and append it to the list of sorted numbers
3. Repeat the above steps until the list is empty

Example list

99 83 125 12 5 256 31 192

Good algorithms

Knuth: “...we want good algorithms in some loosely defined aesthetic sense. One criterion ... is the length of time taken to perform the algorithm. (...) [Other] criteria are adaptability of the algorithm to computers, its simplicity and elegance, etc”

Chaitin: “a program is elegant, by which I mean that it's the smallest possible program for producing the output that it does”

Algorithmic complexity

Time complexity

The time complexity function, $T(n)$:

- The “runtime” of an algorithm that operates on an input of length n .
- Can e.g. measure the number of required computational operations.

Best, worst, and average case

Best case Minimal complexity for the most favorable input.

Worst case Maximal complexity for the least favorable input.

Average case Typical complexity (for some definition of typical input).

Algorithmic complexity

- Classify algorithms according to their performance
- Time function $T(n)$ measures runtime
- Big-O notation expresses *runtime complexity*
- Considers only the highest order term of $T(n)$
- Upper bound on growth rate

Definition

The computational complexity is

$$T(n) \in O(f(n))$$

if and only if there exists a constant c such that $T(n) < cf(n)$ for all $n > n_0$. We say $f(n)$ is an asymptotic upper bound for $T(n)$.

Simplification

The term in $T(n)$ that grows most quickly will eventually dominate all other terms.

We can make the following simplifications

- Only keep the fastest growing term.
- Omit any multiplicative constants.

For example $T(n) = 4x^3 + 5x^2 + 10$ can be simplified to $T(n) \in O(n^3)$.

Example

An algorithm with time complexity

$$T(n) = 1\,000\,000n + n^2$$

is still $O(n^2)$ because for $n > 1\,000\,000$ the term n^2 is largest.

Algorithmic complexity

Constant time, $O(1)$ Same amount of computation regardless of input size
Example: Access a specific element in a list

Logarithmic time, $O(\log n)$ Computation proportional to logarithm of input size
Example: Binary search (find element in sorted list)

Linear time, $O(n)$ Computation proportional to the input size
Example: Find minimum element in a list

Quadratic time, $O(n^2)$ Computation proportional to the square of the input size
Example: Selection sort

Factorial time, $O(n!)$ Computation proportional to the factorial of the input size
Example: Tabulate all permutations of a list

Example lists

<i>Unsorted</i>	99	83	125	12	5	256	31	192
<i>Sorted</i>	5	12	31	83	99	125	192	256

A simple sorting algorithm

Algorithm

1. Find the smallest number on the list
2. Remove the smallest number from the list and append it to the list of sorted numbers
3. Repeat the above steps until the list is empty

A simple sorting algorithm

Algorithm

1. Find the smallest number on the list
2. Remove the smallest number from the list and append it to the list of sorted numbers
3. Repeat the above steps until the list is empty

Complexity

(Number of comparisons needed to sort a list of n numbers)

$$T(n) = (n - 1) + (n - 2) + \cdots + 2 + 1 = \frac{n(n - 1)}{2} = \frac{1}{2}n^2 - \frac{1}{2}n$$

$$T(n) \in O(n^2)$$

Exercise: Merge sort

Algorithm

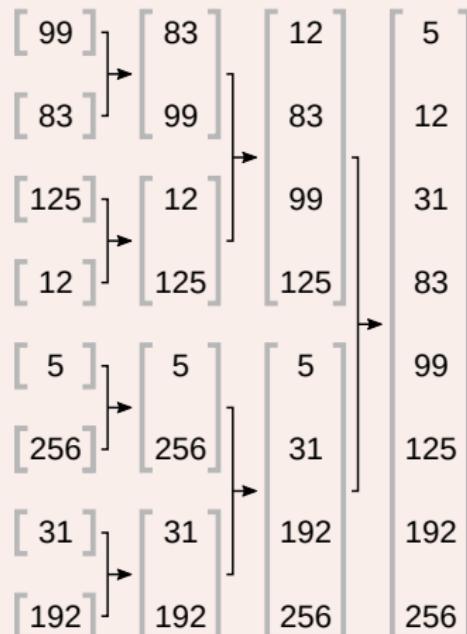
At all times, maintain a set of sorted sublists

Initially each element is a sorted sublist

1. Merge each pair of sublists to form a new sorted sublist
2. Repeat until all sublists have been merged

Question

- How many operations (comparisons) are required (in the worst case) to sort a list of 8 items?
- What is the algorithmic complexity of merge sort?
Assume for simplicity that the number of elements is a power of two, $n = 2^\ell$.



Exercise: Merge sort

Algorithm

At all times, maintain a set of sorted sublists

Initially each element is a sorted sublist

1. Merge each pair of sublists to form a new sorted sublist
2. Repeat until all sublists have been merged

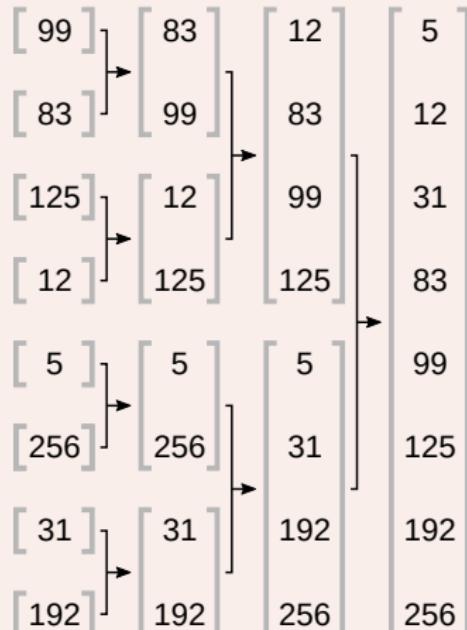
Question

- How many operations (comparisons) are required (in the worst case) to sort a list of 8 items?
- What is the algorithmic complexity of merge sort?
Assume for simplicity that the number of elements is a power of two, $n = 2^\ell$.

Solution

$$T(n) = \frac{n}{2} \cdot 1 + \frac{n}{4} \cdot 3 + \frac{n}{8} \cdot 7 + \dots = n\left(\frac{1}{2} + \frac{3}{4} + \frac{7}{8} + \dots\right) < n\ell \quad T(8) = 17$$

$$n = 2^\ell \Leftrightarrow \ell = \log_2(n), \quad T(n) \in O(n \log n)$$



Divide and conquer / recursion

Divide and conquer

Divide and conquer

1. Divide the problem into smaller sub-problems
2. Solve sub-problems (recursively) until solved
3. Combine the sub-problems to get the solution to the full problem

Find a peak

Input A sequence of n numbers, x_1, x_2, \dots, x_n .

Objective Find a peak, defined as a position $i \in (1, \dots, n)$ in the sequence such that

$$x_{i-1} \leq x_i \geq x_{i+1}$$

or at the edges a peak is defined as

$$x_1 \geq x_2 \quad \text{and/or} \quad x_{n-1} \leq x_n.$$

Find a peak

Input A sequence of n numbers, x_1, x_2, \dots, x_n .

Objective Find a peak, defined as a position $i \in (1, \dots, n)$ in the sequence such that

$$x_{i-1} \leq x_i \geq x_{i+1}$$

or at the edges a peak is defined as

$$x_1 \geq x_2 \quad \text{and/or} \quad x_{n-1} \leq x_n.$$

1. Look at the “middle” number at position $[n/2]$

Find a peak

Input A sequence of n numbers, x_1, x_2, \dots, x_n .

Objective Find a peak, defined as a position $i \in (1, \dots, n)$ in the sequence such that

$$x_{i-1} \leq x_i \geq x_{i+1}$$

or at the edges a peak is defined as

$$x_1 \geq x_2 \quad \text{and/or} \quad x_{n-1} \leq x_n.$$

1. Look at the “middle” number at position $[n/2]$
2. If $x_{n/2-1} > x_{n/2}$ then consider the sequence to the left, $x_1, \dots, x_{n/2-1}$

Find a peak

Input A sequence of n numbers, x_1, x_2, \dots, x_n .

Objective Find a peak, defined as a position $i \in (1, \dots, n)$ in the sequence such that

$$x_{i-1} \leq x_i \geq x_{i+1}$$

or at the edges a peak is defined as

$$x_1 \geq x_2 \quad \text{and/or} \quad x_{n-1} \leq x_n.$$

1. Look at the “middle” number at position $[n/2]$
2. If $x_{n/2-1} > x_{n/2}$ then consider the sequence to the left, $x_1, \dots, x_{n/2-1}$
3. Else, if $x_{n/2} < x_{n/2+1}$ then consider the sequence to the right, $x_{n/2+1}, \dots, x_n$.

Find a peak

Input A sequence of n numbers, x_1, x_2, \dots, x_n .

Objective Find a peak, defined as a position $i \in (1, \dots, n)$ in the sequence such that

$$x_{i-1} \leq x_i \geq x_{i+1}$$

or at the edges a peak is defined as

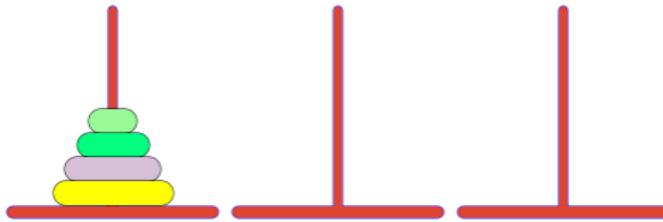
$$x_1 \geq x_2 \quad \text{and/or} \quad x_{n-1} \leq x_n.$$

1. Look at the “middle” number at position $[n/2]$
2. If $x_{n/2-1} > x_{n/2}$ then consider the sequence to the left, $x_1, \dots, x_{n/2-1}$
3. Else, if $x_{n/2} < x_{n/2+1}$ then consider the sequence to the right, $x_{n/2+1}, \dots, x_n$.
4. Else, $x_{n/2}$ is a peak.

Recursion: Tower of Hanoi

```
def thanoi(pieces, movefrom=1, moveto=2, other=3):
    if pieces == 1:
        print(f'Move ring from {movefrom} to {moveto}')
    else:
        thanoi(pieces-1, movefrom, other, moveto)
        thanoi(1, movefrom, moveto, other)
        thanoi(pieces-1, other, moveto, movefrom)

thanoi(4)
```



Output

Move ring from 1 to 3
Move ring from 1 to 2
Move ring from 3 to 2
Move ring from 1 to 3
Move ring from 2 to 1
Move ring from 2 to 3
Move ring from 1 to 3
Move ring from 1 to 2
Move ring from 3 to 2
Move ring from 3 to 1
Move ring from 2 to 1
Move ring from 3 to 2
Move ring from 1 to 3
Move ring from 1 to 2
Move ring from 3 to 2

Tower of Hanoi

Problem

- What is the time complexity $T(n)$ for the solution to the Tower of Hanoi problem? (The number of moves as a function of the number of rings n)

Hint: With one ring it takes one move, $T(1) = 1$. Two rings requires three moves, $T(2) = 3$. To solve an $n + 1$ -rings problem we solve an n -rings problem, move 1 ring, and solve an n -rings problem again, so we have $T(n + 1) = 2 \cdot T(n) + 1$. You can use this to find $T(n)$ for $n = 1, 2, 3, 4 \dots$ and see if you can spot the pattern.

Tower of Hanoi

Problem

- What is the time complexity $T(n)$ for the solution to the Tower of Hanoi problem? (The number of moves as a function of the number of rings n)

Hint: With one ring it takes one move, $T(1) = 1$. Two rings requires three moves, $T(2) = 3$. To solve an $n + 1$ -rings problem we solve an n -rings problem, move 1 ring, and solve an n -rings problem again, so we have $T(n + 1) = 2 \cdot T(n) + 1$. You can use this to find $T(n)$ for $n = 1, 2, 3, 4 \dots$ and see if you can spot the pattern.

Solution

Using the recursion we get

n	1	2	3	4	5	6	7	8
$T(n)$	1	3	7	15	31	63	127	255

From this we can spot the pattern $T(n) = 2^n - 1$

Software tools

L^AT_EX

L^AT_EX is a markup language for writing documents. Once you get used to it, it is way better than everything else.

- Install on your own computer or *or use online at overleaf.com*
- Get started with the tutorial at latex-tutorial.com/tutorials

A bit of a learning curve, so you might as well get started.

Git

Git is a distributed system for version control, especially useful for software development in a team.

- Use a commercial system like github or *DTU Compute's free lab.compute.dtu.dk*
- Worksheet “git_tutorial.pdf” on DTU Learn.
- A good free book/reference at git-scm.com/book/en/v2

A bit of a learning curve, so you might as well get started.

Tasks

Tasks for today

Tasks today

- Continue work on lab report
- Start learning about on latex-tutorial.com/tutorials/ and overleaf.com
- Start learning about Git with the worksheet “git_tutorial.pdf” on DTU Learn.

Today's feedback group

- Viet Hoang Nguyen
- Lukas Peter Dyhr
- Philip Kierkegaard
- Ali Mohammed Fathi Afif

Lab report hand in

- Lab 1: Image recognition (Deadline: Thursday 14 September 20:00)

Next time

- Read the notes “Symbolic AI” + Solve all problems

Introduction to intelligent systems

Symbolic AI

Mikkel N. Schmidt

Technical University of Denmark,
DTU Compute, Department of Applied Mathematics and Computer Science.

Overview

- ① Logic
- ② Expert systems
- ③ Search
- ④ Demo: Lunar Lander and 2048
- ⑤ Tasks

Feedback group

- Lucas Rieneck Gottfried Pedersen
- Benjamin Banks
- Tobias Emde Ralsted Jensen
- Solvej Amélie Brun Sønderbæk

Learning objectives

- I Symbolic AI.
 - I Forward and backward chaining.
 - I Monte carlo search.
 - II Boolean logic.
 - II Rule-based systems and expert systems.
-
- I Understand the concepts and definitions, and know their application. Reason about the concepts in the context of an example. Use correct technical terminology.
 - II As above plus: Read, manipulate, and work with technical definitions and expressions (mathematical and Python code). Carry out practical computations. Interpret and evaluate results.

Logic

Boolean algebra notation

- Variables

True	$x = T$	$x = 1$
False	$x = F$	$x = 0$

- Operators

and	$x \wedge y$	$x \cdot y$
or	$x \vee y$	$x + y$
not	$\neg x$	\bar{x}

Boolean algebra

Commutative law $a + b = b + a$

$$a \cdot b = b \cdot a$$

Associative law $a + (b + c) = (a + b) + c$

$$a \cdot (b \cdot c) = (a \cdot b) \cdot c$$

Distributive law $(a \cdot b) + (a \cdot c) = a \cdot (b + c)$

$$(a + b) \cdot (a + c) = a + (b \cdot c)$$

Double negative law $\overline{\overline{a}} = a$

Identities $a + 0 = a, \quad a + 1 = 1$

$$a \cdot 1 = a, \quad a \cdot 0 = 0$$

Complement law $a + \overline{a} = 1$

$$a \cdot \overline{a} = 0$$

Absorbtion laws $a + a \cdot b = a$

$$a + \overline{a} \cdot b = a + b$$

DeMorgan's law $\overline{a \cdot b} = \overline{a} + \overline{b}$

$$\overline{a + b} = \overline{a} \cdot \overline{b}$$

Boolean function

$$f(x_1, x_2, \dots, x_n)$$

- Inputs: n Boolean values
- Output: A Boolean value

Will I eat chocolate?

The following Boolean function governs when I eat chocolate:

$$f(a, b, c) = \bar{a} \cdot b + b \cdot \bar{c} + b \cdot c + a \cdot \bar{b} \cdot \bar{c}$$

f =I'll eat chocolate a =I'm hungry b =I'm tired c =I have proper food

Question 1 (easy) Will I eat chocolate if im hungry, not tired, and have proper food? I.e., what is the value of $f(1, 0, 1)$?

Question 2 (difficult) Simplify the expression for $f(a, b, c)$ as much as possible.

Will I eat chocolate?

The following Boolean function governs when I eat chocolate:

$$f(a, b, c) = \bar{a} \cdot b + b \cdot \bar{c} + b \cdot c + a \cdot \bar{b} \cdot \bar{c}$$

f =I'll eat chocolate a =I'm hungry b =I'm tired c =I have proper food

Question 1 (easy) Will I eat chocolate if im hungry, not tired, and have proper food? I.e., what is the value of $f(1, 0, 1)$?

Question 2 (difficult) Simplify the expression for $f(a, b, c)$ as much as possible.

Solution

1. $f(1, 0, 1) = \bar{1} \cdot 0 + 0 \cdot \bar{1} + 0 \cdot 1 + 1 \cdot \bar{0} \cdot \bar{1} = 0 \cdot 0 + 0 \cdot 0 + 0 \cdot 1 + 1 \cdot 1 \cdot 0 = 0$
2. $f(a, b, c) = b \cdot (\bar{a} + \bar{c} + c) + a \cdot \bar{b} \cdot \bar{c} = b + a \cdot \bar{b} \cdot \bar{c} = b + a \cdot \bar{c}$

Expert systems

Expert systems

- A collection of facts and rules

Facts Observable / inferred variables

Rules If-statements

if <condition>

then <consequent>

- Rules are created by experts

Example: Will I eat chocolate?

if tired

then eat chocolate

if hungry and not have proper food

then eat chocolate

if have proper food

then dont eat chocolate

if tired and not hungry

then dont eat chocolate

Conflicts and undefined cases

Constructing rule that cover all cases without conflict is difficult

Conflict

- Two or more rules fire with conflicting consequents
- Solution: Prioritize rules

Undefined cases

- No rules fire to generate the consequent we are interested in
- Solution: Add more rules or define defaults

Conflicts and undefined cases

Can you identify a conflict and an undefined case in the example?

if tired

then eat chocolate

if hungry and not have proper food

then eat chocolate

if have proper food

then dont eat chocolate

if tired and not hungry

then dont eat chocolate

Conflicts and undefined cases

Can you identify a conflict and an undefined case in the example?

if tired

then eat chocolate

if hungry and not have proper food

then eat chocolate

if have proper food

then dont eat chocolate

if tired and not hungry

then dont eat chocolate

Solution

- Conflict: Rule 1 and 4 fire if *tired* and not *hungry*.
- Undefined: No rule fires if not *tired*, not *hungry*, and not *have proper food*.

Forward and backward chaining

Forward chaining Reasoning from the data

1. Apply the first rule that fires
2. The fired rule generates new facts, influencing which rules may fire
3. Repeat from 1, making sure each rule fires only once

Backward chaining Reasoning to prove an outcome

1. Find the rule(s) that could generate the fact we want to prove
2. Check if the conditions of the rule(s) are true
3. If unknown, find the rule(s) that could generate the facts needed for the condition
4. Continue until the fact is proven or we have failed our attempt

Forward and backward chaining

Consider the example on the right (it has no conflicts)

if *happy*=True or *birthday*=True
then *sing*=True

if *birthday*=True and *diet*=False
then *cake*=True

if *cake*=True and (*sing*=True or *diet*=True)
then *problem*=True

1. What can you infer from the fact *birthday*=True?
(Use forward chaining)
2. Can you prove that *problem*=True from *sing*=True and *diet*=False?
(Use backward chaining)

Forward and backward chaining

Consider the example on the right (it has no conflicts)

```
if happy=True or birthday=True  
then sing=True  
  
if birthday=True and diet=False  
then cake=True  
  
if cake=True and (sing=True or diet=True)  
then problem=True
```

1. What can you infer from the fact *birthday*=True?
(Use forward chaining)
2. Can you prove that *problem*=True from *sing*=True and *diet*=False?
(Use backward chaining)

Solution

1. *sing*=True follows from rule 1. Nothing more can be inferred
2. No it cannot be proven. Only rule 3 could generate *problem*=True but we need *cake*=True. This could only be generated from rule 2 we then need *birthday*=True which we do not have.

Search

AI by search

Search as an AI strategy

- Observables: State of the environment
- Action: Change environment state
- Goal: Take actions to reach a certain desirable state

Monte Carlo search

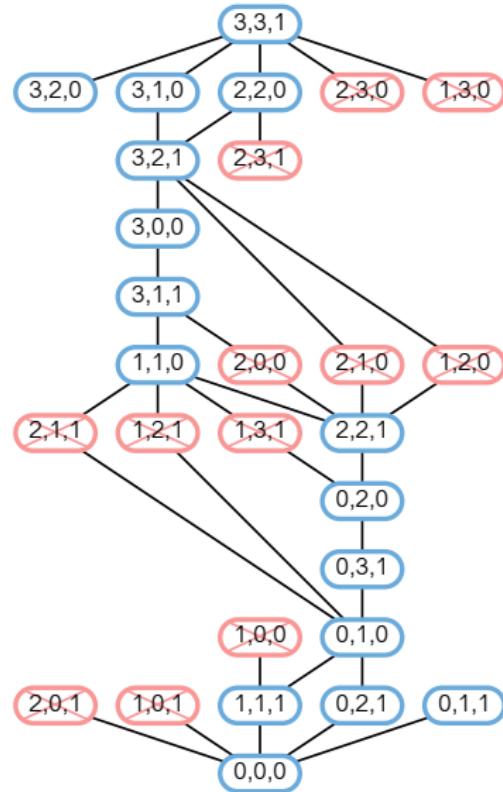
Exact search is often not feasible

- Large state space: Search all paths not possible
- Approximate search: Find next step that likely leads to a solution
- Requires measure of reward to rank states/partial paths

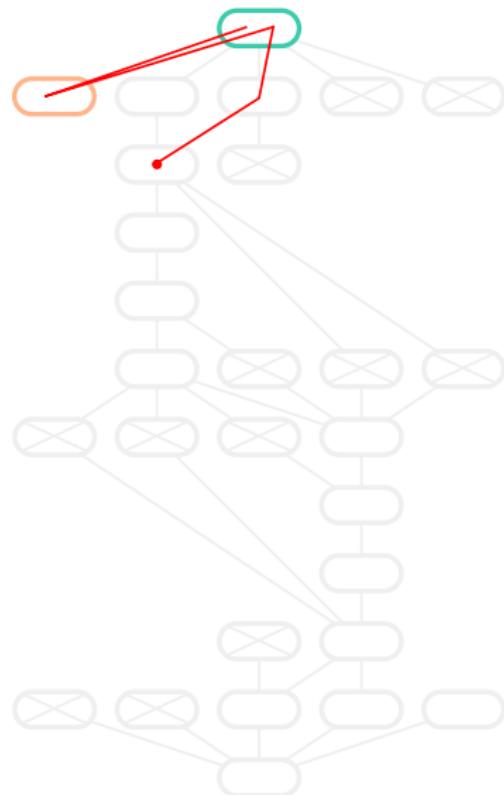
Monte Carlo search

1. Explore N random paths from current state
2. Score the explored paths by their reward
3. Take the first step that leads to the highest average score
4. Repeat

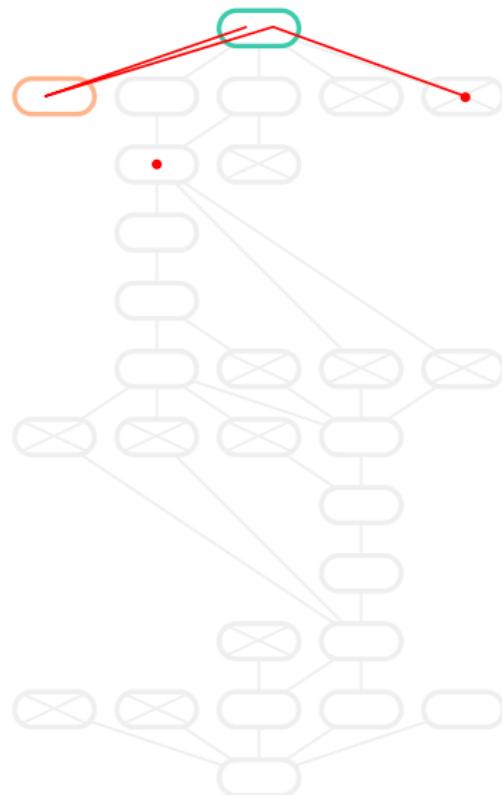
Monte Carlo search



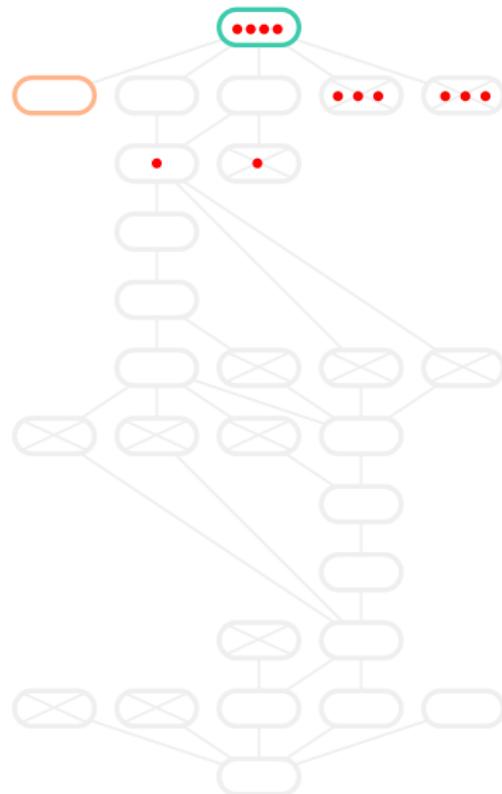
Monte Carlo search



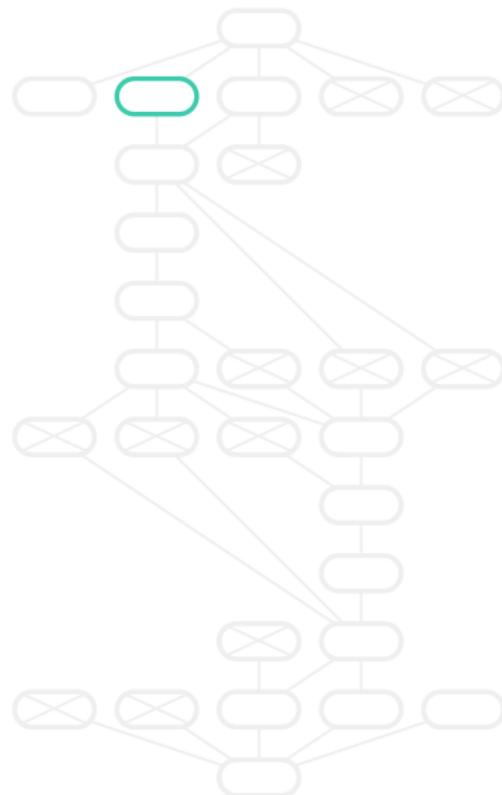
Monte Carlo search



Monte Carlo search



Monte Carlo search



Demo: Lunar Lander and 2048

Tasks

Tasks for today

Tasks today

- Start working on Lab Report 2: Read description on DTU Learn
- Today's feedback group
 - Lucas Rieneck Gottfried Pedersen
 - Benjamin Banks
 - Tobias Emde Ralsted Jensen
 - Solvej Amélie Brun Sønderbæk

Lab report hand in

- Lab 2: Symbolic AI (Deadline: Thursday 28 September 20:00)

Next time

- Read the notes “Data representation” + Solve all problems

Introduction to intelligent systems

Natural language processing

Mikkel N. Schmidt

Technical University of Denmark,
DTU Compute, Department of Applied Mathematics and Computer Science.

Overview

① Text processing

② Demo: List manipulation in Python

③ Tasks

Feedback group

- Marius Drachmann Niss
- Joseph An Duy Nguyen
- Tobias Rodrigues Bjerre
- Marcus Zabell Olssen

Learning objectives

- I Data representation in the computer.
 - I Bag of words representation.
 - II Term frequency-inverse document frequency (TF-IDF).
-
- I Understand the concepts and definitions, and know their application. Reason about the concepts in the context of an example. Use correct technical terminology.
 - II As above plus: Read, manipulate, and work with technical definitions and expressions (mathematical and Python code). Carry out practical computations. Interpret and evaluate results.

Text processing

Natural language processing

Syntax Part-of-speech tagging, parsing, grammar induction etc.

Semantics Lexical semantics, translation, named entity recognition, sentiment analysis, topic analysis, etc.

Discourse Summarization, discourse analysis

Levels of analysis

- Sequence of characters
- Sequence of words
- Sequence of sentences

Document search

Corpus A set of documents (text)

Query A text string

Goal Return top- N relevant documents according to the query

Which document is most relevant

Consider the search query:

what is a cat

Which of the following “documents” (sentences) is most relevant, and why?

1. Cats are small domesticated mammals with soft fur and retractile claws.
2. We need to promote direct foreign investment, which **is what a** free trade agreement would do.
3. **Is** a banana **a** fruit or **a** herb? The banana plant **is** technically regarded as **a** herb, because the stem does not contain woody tissue.
4. The Caterpillar Inc. (**CAT**) stock price **is** now well beyond **what** most analysts predicted.

(Words matching the query are highlighted, case and punctuation ignored.)

Desiderata

- Word endings should be ignored
- Common words should be ignored and rare words should be given emphasis
- System should distinguish between homographs (words spelled the same with different meaning)
- User should write a better query
- ... more?

Stemming

- Reduce words to their word stem, base or root form.
- Related words map to the same stem
- Many search engines treat words with the same stem as synonyms (conflation)

Example: **argue**, **argued**, **argues**, **arguing** all reduce to the stem **argu**

Example of stemming

Before stemming

Zebras are several species of African equids (horse family) united by their distinctive black and white striped coats.

After stemming

zebra are sever speci of african equid hors famili unit by their distinct black and white stripe coat

Bag-of-words representation

	Doc. 1	Doc. 2
african	1	0
although	0	1
and	1	0
are	1	0
bear	0	1
black	1	0
by	1	0
close	0	1
coat	1	0
distinct	1	0
equid	1	0
famili	1	0
giraff	0	1
hors	1	0
is	0	1
it	0	1
mark	0	1
most	0	1
of	1	1
okapi	0	1
relat	0	1
reminisc	0	1
sever	1	0
speci	1	0
stripe	1	1
the	0	2
their	1	0
to	0	1
unit	1	0
white	1	0
zebra	1	1

Sentences

1. Zebras are several species of African equids (horse family) united by their distinctive black and white striped coats.
2. Although the okapi bears striped markings reminiscent of zebras it is most closely related to the giraffe.

- A bag-of-words sentence/document can be seen as a point in a high-dimensional vector space

Exercise: Dot product

We can use the dot product between the word occurrence vectors as a measure of similarity between documents

- Compute the dot product between the two sentences
- Can you think of pros and cons of using the dot product to measure similarity?

Sentences

1. Zebras are several species of African equids (horse family) united by their distinctive black and white striped coats.
2. Although the okapi bears striped markings reminiscent of zebras it is most closely related to the giraffe.

Words in common in the sentences

	Doc. 1	Doc. 2
of	1	1
stripe	1	1
zebra	1	1

TF and IDF

TF: Term frequency How *frequently* a term occurs in a document. Since documents can have different length. TF is often normalized by the document length.

$$\text{TF}(t, d) = \frac{\#\text{occurrences of term } t \text{ in document } d}{\#\text{words in document } d}$$

IDF: Inverse document frequency How *important* a term is. Some terms like *is*, *that*, and *the* may appear a lot, but have little importance.

$$\text{IDF}(t) = \log \left(\frac{\#\text{documents in corpus}}{\#\text{documents with term } t} \right)$$

Exercise: TF-IDF

- Consider a document that contains 100 words, wherein
 - the word *the* appears 3 times and
 - the word *cat* appears 3 times
- The document is part of a 10 000 document corpus, wherein
 - 4 900 of the documents contain the word *the* and
 - 123 of the documents contain the word *cat*

TF and IDF

$$TF = \frac{n_{t,d}}{n_d} \quad IDF = \log \left(\frac{N}{n_t} \right)$$

$n_{t,d}$ Number of occurrences of term t in document d

n_d Number of terms in document d

n_t Number of documents with term t

N Total number of documents

Compute the TF and IDF for the terms *the* and *cat*

Exercise: TF-IDF

- Consider a document that contains 100 words, wherein
 - the word *the* appears 3 times and
 - the word *cat* appears 3 times
- The document is part of a 10 000 document corpus, wherein
 - 4 900 of the documents contain the word *the* and
 - 123 of the documents contain the word *cat*

TF and IDF

$$TF = \frac{n_{t,d}}{n_d} \quad IDF = \log \left(\frac{N}{n_t} \right)$$

$n_{t,d}$ Number of occurrences of term t in document d

n_d Number of terms in document d

n_t Number of documents with term t

N Total number of documents

Compute the TF and IDF for the terms *the* and *cat*

Solution

the

$$TF = \frac{3}{100} = 0.03$$

$$IDF = \log \left(\frac{10\,000}{4\,900} \right) \approx 0.7133$$

cat

$$TF = \frac{3}{100} = 0.03$$

$$IDF = \log \left(\frac{10\,000}{123} \right) \approx 4.398$$

TF-IDF score

$$\text{TF-IDF}(d, q) = \underbrace{\sum_{t \in q}}_{\text{Sum over all terms in query } q} \frac{n_{t,d}}{n_d} \cdot \log \left(\frac{N}{n_t} \right)$$

$n_{t,d}$ Number of occurrences of term t in document d

n_d Number of terms in document d

n_t Number of documents with term t

N Total number of documents

Exercise: TF-IDF

TF-IDF

$$\text{TF-IDF}(d, q) = \sum_{t \in q} \frac{n_{t,d}}{n_d} \cdot \log \left(\frac{N}{n_t} \right)$$

- What happens if no documents contain one of the search terms?

$n_{t,d}$ Number of occurrences of term t in document d

n_d Number of terms in document d

n_t Number of documents with term t

N Total number of documents

Exercise: TF-IDF

TF-IDF

$$\text{TF-IDF}(d, q) = \sum_{t \in q} \frac{n_{t,d}}{n_d} \cdot \log \left(\frac{N}{n_t} \right)$$

- What happens if no documents contain one of the search terms?

$n_{t,d}$ Number of occurrences of term t in document d

n_d Number of terms in document d

n_t Number of documents with term t

N Total number of documents

Solution: Division by zero!

Okapi BM25

$$\text{BM25}(d, q) = \sum_{t \in q} \frac{n_{t,d} \cdot (k_1 + 1)}{n_{t,d} + k_1 \cdot (1 - b + b \cdot \frac{n_d}{\text{avgdl}})} \cdot \log \left(\frac{N - n_t + 0.5}{n_t + 0.5} \right)$$

$n_{t,d}$ Number of occurrences of term t in document d

n_d Number of terms in document d

n_t Number of documents with term t

N Total number of documents

avgdl Average document length $\frac{1}{N} \sum_d n_d$

b Parameter ($b \in [0, 1]$, default $b = 0.75$)

k_1 Parameter ($k_1 > 0$, default $k_1 = 1.2$)

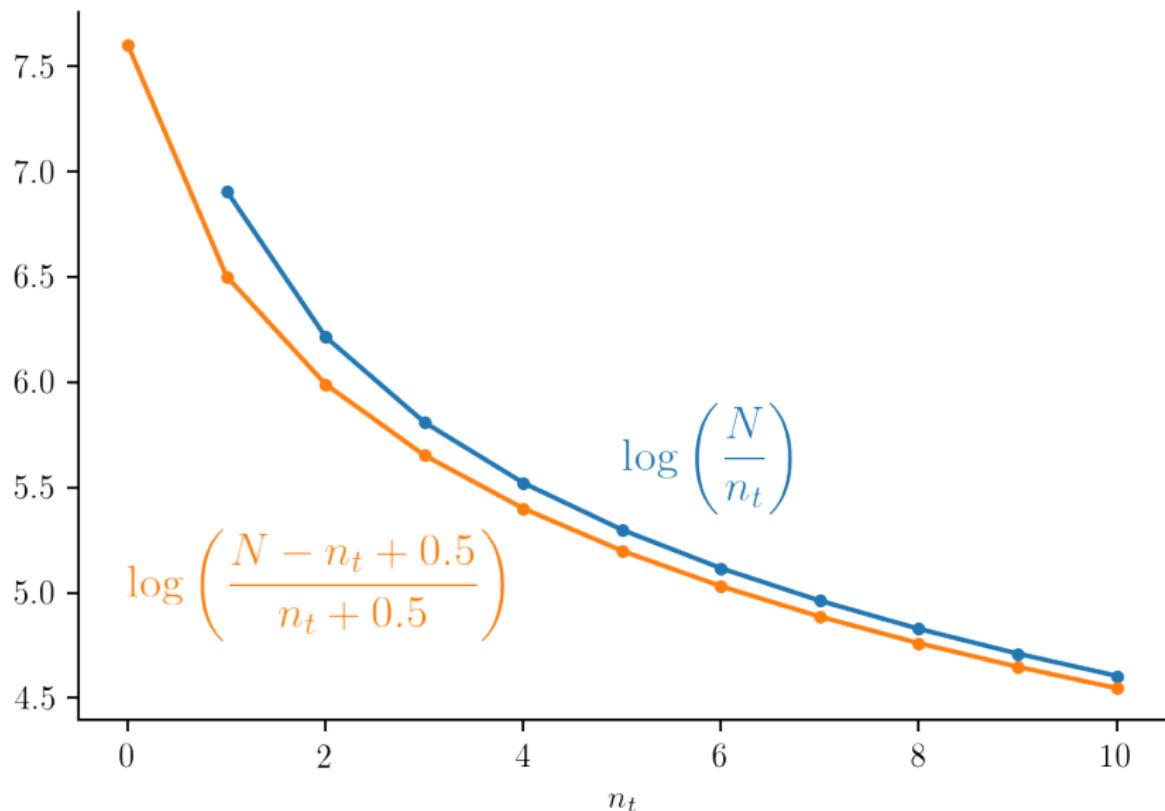
Okapi BM25 parameters

$$\text{BM25}(d, q) = \sum_{t \in q} \frac{n_{t,d} \cdot (k_1 + 1)}{n_{t,d} + k_1 \cdot (1 - b + b \cdot \frac{n_d}{\text{avgdl}})} \cdot \log \left(\frac{N - n_t + 0.5}{n_t + 0.5} \right)$$

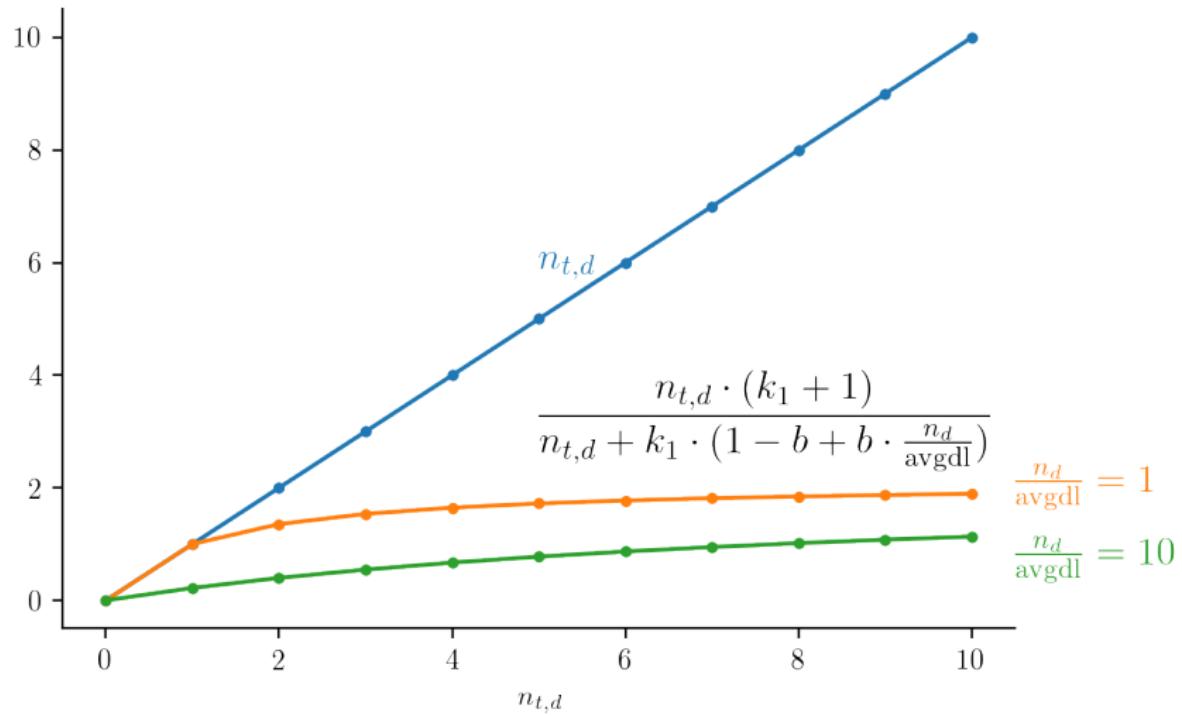
What do the parameters b and k_1 control?

- k_1 How much to weigh the term-frequency
 $k_1 = 0$: No term-frequency. $k_1 \rightarrow \infty$: Raw term-frequency.
- b How much to scale with the document length
 $b = 0$: No scaling. $b = 1$: Full scaling.

The “inverse document frequency” term in Okapi BM25



The “term-frequency” in Okapi BM25



Exercise: Okapi BM25

BM25

$$\text{BM25}(d, q) = \sum_{t \in q} \frac{n_{t,d} \cdot (k_1 + 1)}{n_{t,d} + k_1 \cdot (1 - b + b \cdot \frac{n_d}{\text{avgdl}})} \cdot \log \left(\frac{N - n_t + 0.5}{n_t + 0.5} \right)$$

- Consider a document that contains 100 words, wherein
 - the word *the* appears 3 times and
 - the word *cat* appears 3 times
- The document is part of a 10 000 document corpus, wherein
 - 4 900 of the documents contain the word *the* and
 - 123 of the documents contain the word *cat*
- The average document length in the corpus is 150

$n_{t,d}$ Number of occurrences of term t in document d

n_d Number of terms in document d

n_t Number of documents with term t

N Total number of documents

avgdl Average document length

b $b = 0.75$

k_1 $k_1 = 1.2$

Compute the BM25-score for the query *the cat*

Exercise: Okapi BM25

Solution

$$\begin{aligned} \text{BM25}(d, q) &= \sum_{t \in q} \frac{n_{t,d} \cdot (k_1 + 1)}{n_{t,d} + k_1 \cdot (1 - b + b \cdot \frac{n_d}{\text{avgdl}})} \cdot \log \left(\frac{N - n_t + 0.5}{n_t + 0.5} \right) \\ &= \frac{3 \cdot (1.2 + 1)}{3 + 1.2 \cdot (1 - 0.75 + 0.75 \cdot \frac{100}{150})} \cdot \log \left(\frac{10\,000 - 4\,900 + 0.5}{4\,900 + 0.5} \right) + \\ &\quad \frac{3 \cdot (1.2 + 1)}{3 + 1.2 \cdot (1 - 0.75 + 0.75 \cdot \frac{100}{150})} \cdot \log \left(\frac{10\,000 - 123 + 0.5}{123 + 0.5} \right) \\ &\approx 1.692 \cdot 0.040 + 1.692 \cdot 4.382 \approx \underline{\underline{7.483}} \end{aligned}$$

Demo: List manipulation in Python

Demo: List manipulation in Python

```
>>> my_list = [1, 1, 2, 3, 5, 8, 13]
```

Indexing (accessing elements)

```
>>> my_list[0]
```

1

```
>>> my_list[4]
```

5

Length

```
>>> len(my_list)
```

7

Check if element occurs

```
>>> 1 in my_list
```

True

```
>>> 17 in my_list
```

False

Count occurrences

```
>>> my_list.count(1)
```

2

```
>>> my_list.count(2)
```

1

```
>>> my_list.count(17)
```

0

List comprehension

```
>>> [x**2 for x in my_list]
```

[1, 1, 4, 9, 25, 64, 169]

Demo: List manipulation in Python

```
>>> my_list = [1, 1, 2, 3, 5, 8, 13]
>>> my_query = [3, 4, 5]
```

How many of the numbers in `my_query` occur in `my_list`?

For-loop

```
>>> occurrences = 0
>>> for q in my_query:
...     if q in my_list:
...         occurrences += 1
```

List comprehension

```
>>> occurrences = [q in my_list for q in my_query].count(True)
```

Tasks

Tasks

Today

1. Implement document search for the *animals.txt* data set
 - Work through the tasks in the script `05-DocumentSearch.ipynb` (contains a solution template)
 - Given a query (one or more words), your program must return the top-5 best matching documents
 - Start by implementing TF-IDF and then move on to Okapi BM25
2. Today's feedback group
 - Marius Drachmann Niss
 - Joseph An Duy Nguyen
 - Tobias Rodrigues Bjerre
 - Marcus Zabell Olssen

Lab report

- Lab 2: Symbolic AI (Deadline: Thursday 28 September 20:00)

Introduction to intelligent systems

Machine learning

Mikkel N. Schmidt

Technical University of Denmark,
DTU Compute, Department of Applied Mathematics and Computer Science.

Overview

- ① Machine learning problems
- ② Machine learning algorithms
- ③ Linear regression
- ④ Generalization
- ⑤ Tasks

Feedback group

- David Svane-Petersen
- Yuxuan Zhang
- sebastian vargr
- Peter Vestereng Larsen

Learning objectives

- I Types of machine learning problems.
 - I Generalization: Training and test error.
 - II Linear regression. Model, parameters, and cost function.
-
- I Understand the concepts and definitions, and know their application. Reason about the concepts in the context of an example. Use correct technical terminology.
 - II As above plus: Read, manipulate, and work with technical definitions and expressions (mathematical and Python code). Carry out practical computations. Interpret and evaluate results.

Machine learning problems

Machine learning problems

Categorization of learning problems

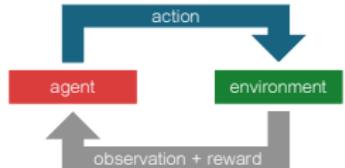
Unsupervised Learn function that describes the structure in data



Supervised Learn function that maps input to output to optimize cost



Reinforcement Learn a function (policy) that maps inputs to actions to optimize cumulative reward



Unsupervised learning

Learn a function that describes the structure in a data set

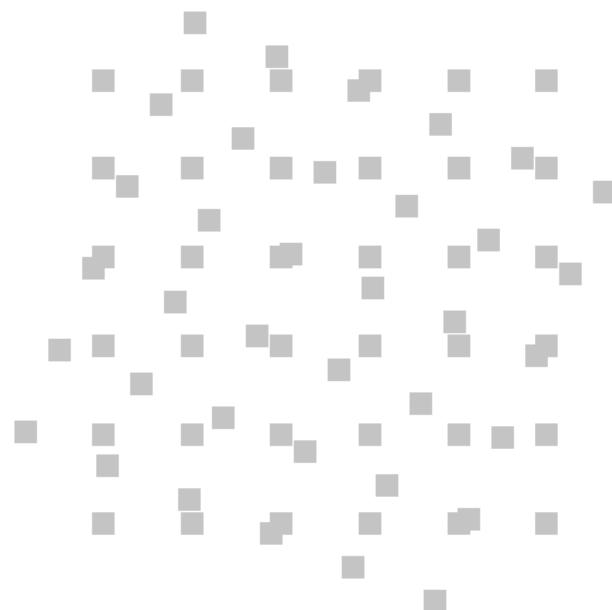
Clustering Find a way to group data points into meaningful components

Dimensionality reduction Find a lower-dimensional representation of the data

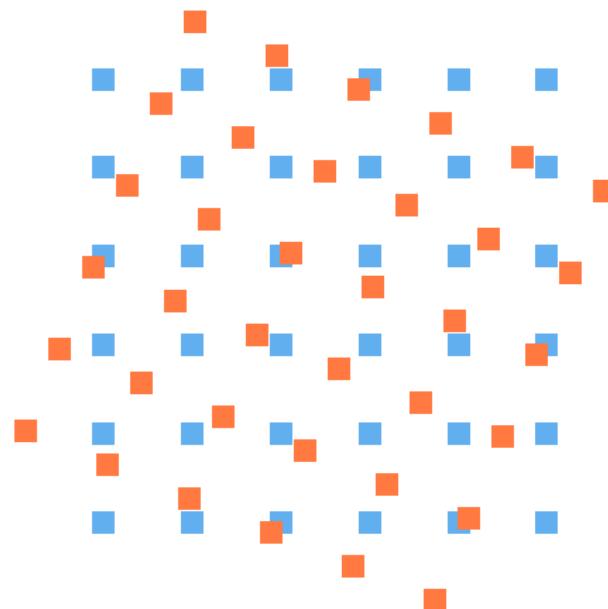
Anomaly detection Find data points that deviate from “normal” behaviour



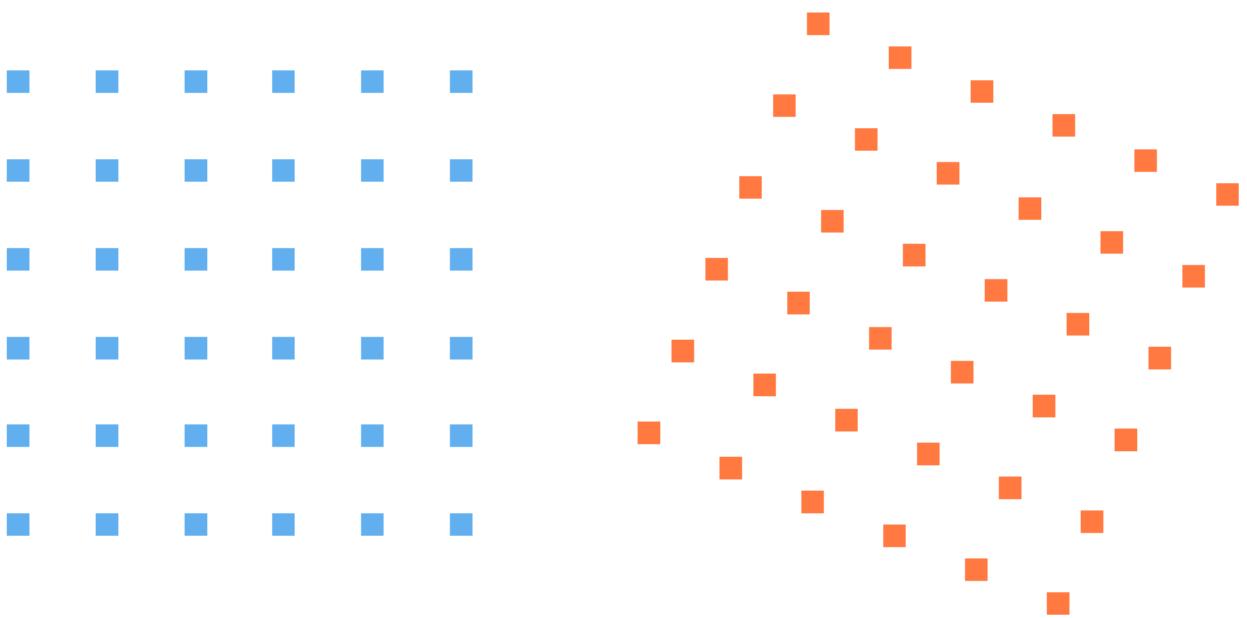
Unsupervised learning: Discover patterns in data



Unsupervised learning: Discover patterns in data



Unsupervised learning: Discover patterns in data



Supervised learning

Learn a function that maps an input to an output to optimize a cost

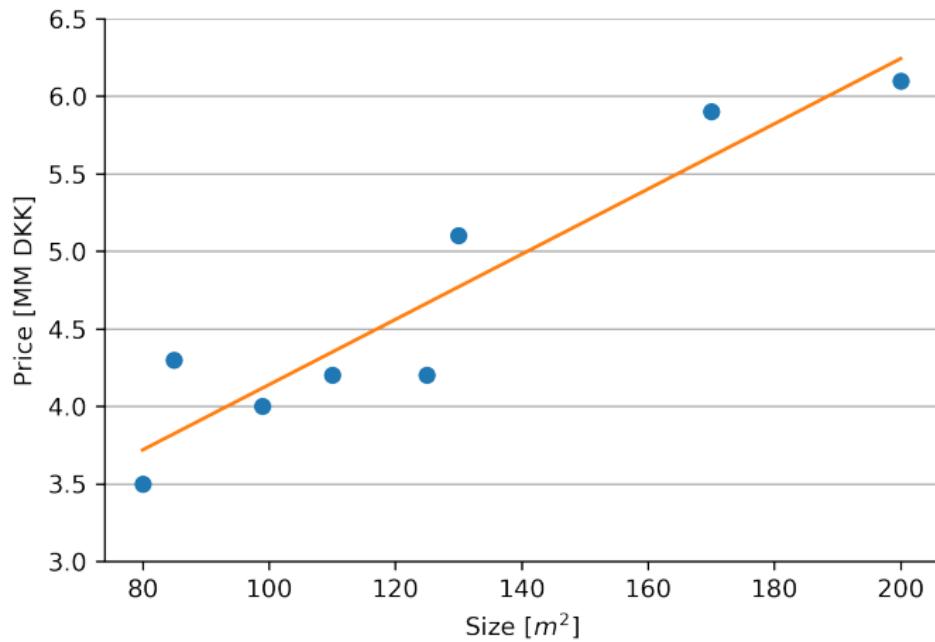
Regression Outputs are continuous variables

Classification Outputs are discrete classes

Ranking Output is a ranking of the data objects



Supervised learning: House price prediction



Reinforcement learning

Learn a function (policy) that maps inputs to actions to optimize cumulative reward

Evaluation vs. control

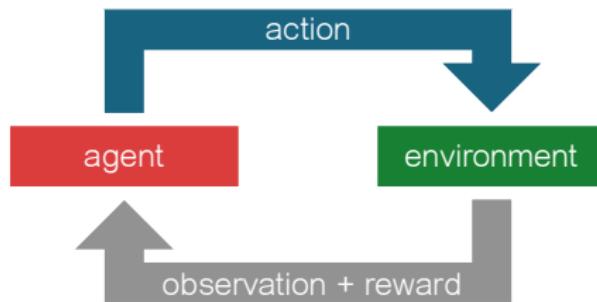
Passive Evaluate the future reward for a given policy

Active Estimate the optimal policy by exploration

Observability of the environment

Full Agent knows the state of the environment

Partial Agent must learn a representation of the environment



Exercise: What is human learning?

Is human learning best characterized as

- Unsupervised learning
- Supervised learning
- Reinforcement learning

(If you think the answer is somehow obvious, see if you can come up with an argument against)

Machine learning algorithms

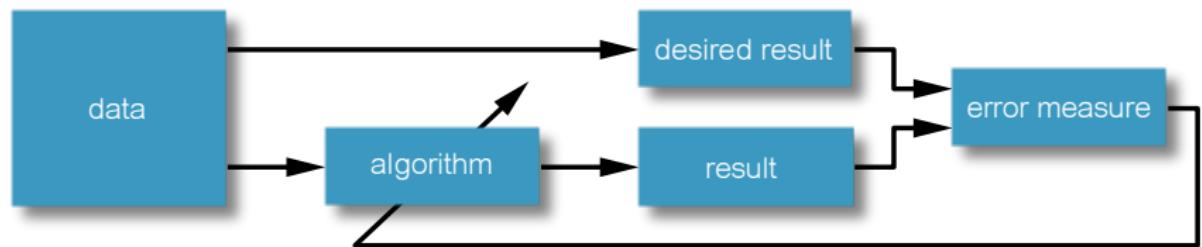
Machine learning algorithms

Machine learning

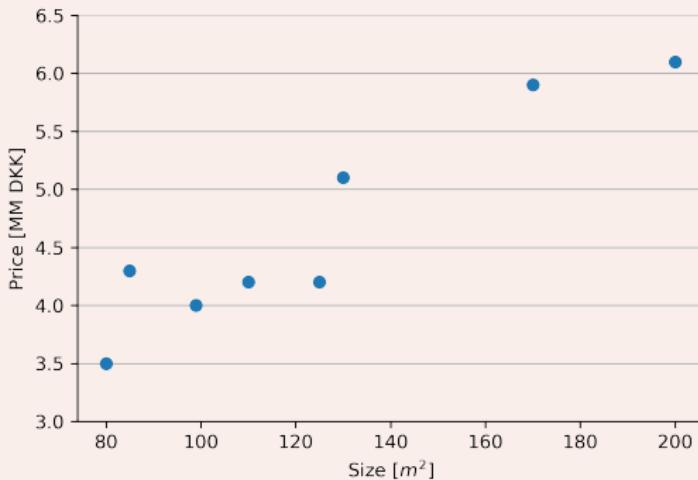
- Algorithm with tunable parameters
- Takes in some data and produces some output
- Measure error between algorithm's output and the desired output
- Tune parameters to minimize error

Goal: Generalization = good performance on future/unseen data

Machine learning algorithms

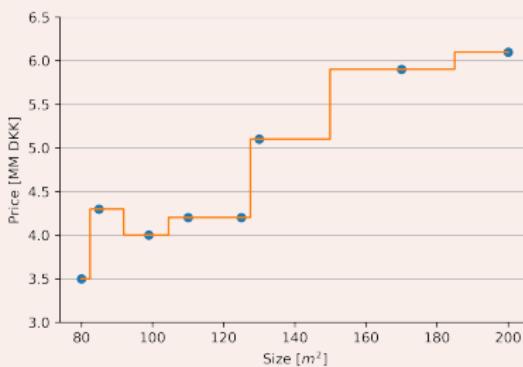
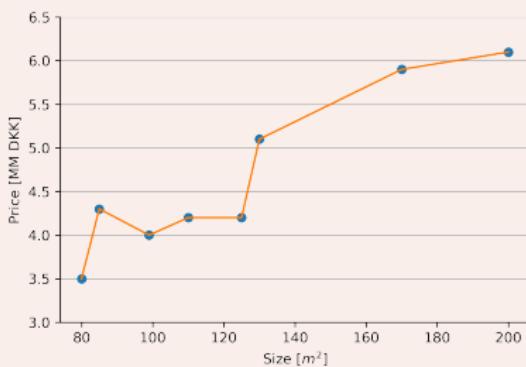
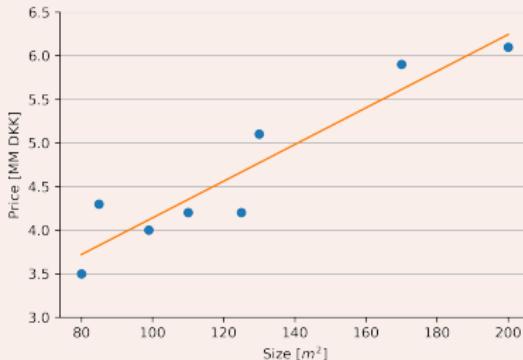
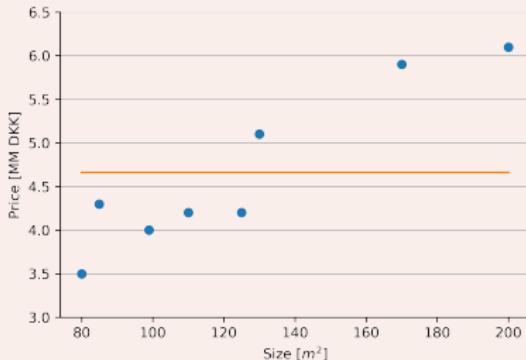


Exercise: Price of a $150\ m^2$ house



- What would you expect the price of a $150\ m^2$ house to be?
- Discuss which “algorithm” you used to come up with your answer

Exercise: House price regression



- Which of the above regression curves is best?
- Discuss how you could define a criteria for which is “best”

House price regression

Possible criteria for a good regression line

Fit the observed data well

Robust to small changes in the data

Generalize to (unseen) future data

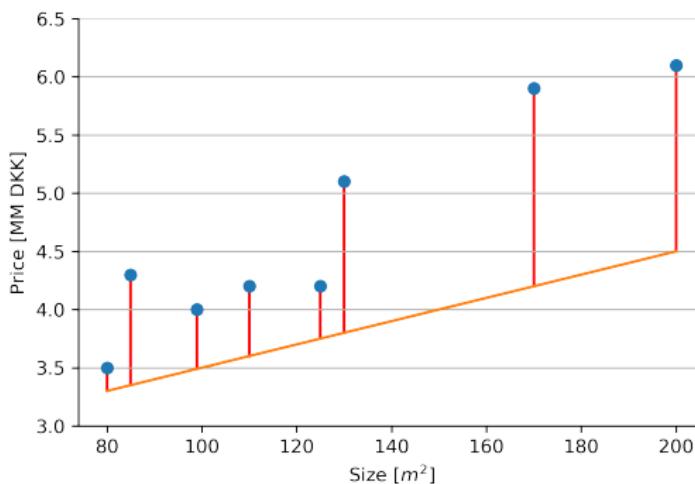
Linear regression

Linear regression

- Regression line: $f(x) = ax + b$
- Error: Squared distance between data and regression line

$$E = \sum_{n=1}^N (y_n - f(x_n))^2$$

- Find values of a and b to minimize E

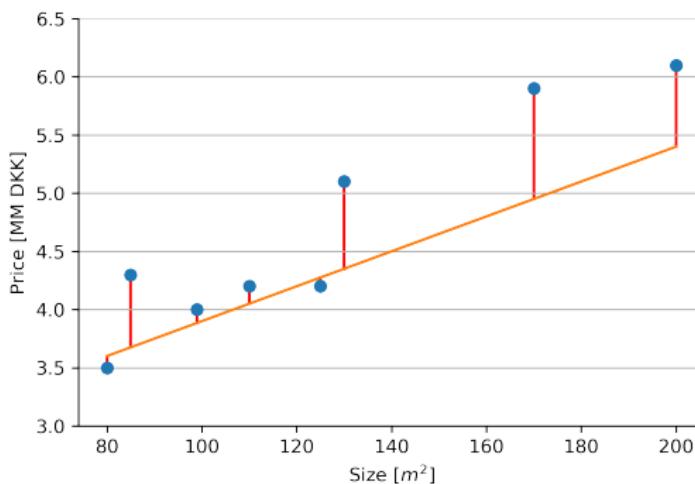


Linear regression

- Regression line: $f(x) = ax + b$
- Error: Squared distance between data and regression line

$$E = \sum_{n=1}^N (y_n - f(x_n))^2$$

- Find values of a and b to minimize E

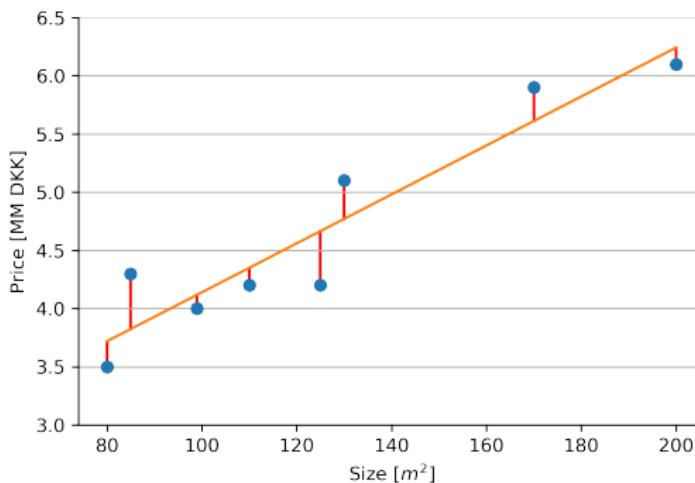


Linear regression

- Regression line: $f(x) = ax + b$
- Error: Squared distance between data and regression line

$$E = \sum_{n=1}^N (y_n - f(x_n))^2$$

- Find values of a and b to minimize E



Exercise: Least squares regression

Solve the least square regression problem by minimizing the error

- Differentiate the error measure wrt. the parameters a and b
- This gives you two equations in two unknowns to solve

Problem specification

- Data

$$x = \{80, 85, 99, 110, 125, 130, 170, 200\}$$

$$y = \{3.5, 4.3, 4, 4.2, 4.2, 5.1, 5.9, 6.1\}$$

- Regression function

$$f(x) = ax + b$$

- Error measure

$$E = \sum_{n=1}^N (y_n - f(x_n))^2$$

Some useful definitions

$$\bar{x} = \sum_{n=1}^N x_n = 999$$

$$\bar{y} = \sum_{n=1}^N y_n = 37.3$$

$$\overline{xy} = \sum_{n=1}^N x_n y_n = 4914.5$$

$$\overline{xx} = \sum_{n=1}^N x_n^2 = 136951$$

Solution: Equation for a

Differentiate wrt. a and equate to zero

$$\begin{aligned} E &= \sum_{n=1}^N (y_n - f(x_n))^2 = \sum_{n=1}^N (y_n - ax_n - b)^2 \\ \frac{dE}{da} &= \sum_{n=1}^N -2(y_n - ax_n - b)x_n \\ &= -2 \sum_{n=1}^N y_n x_n + 2a \sum_{n=1}^N x_n^2 + 2b \sum_{n=1}^N x_n \\ &= -2\bar{xy} + 2a\bar{x}\bar{x} + 2b\bar{x} = 0 \\ \Rightarrow \underline{\bar{x}x \cdot a + \bar{x} \cdot b} &= \underline{\bar{xy}} \end{aligned}$$

Some useful definitions

$$\bar{x} = \sum_{n=1}^N x_n = 999$$

$$\bar{y} = \sum_{n=1}^N y_n = 37.3$$

$$\bar{xy} = \sum_{n=1}^N x_n y_n = 4914.5$$

$$\bar{x}\bar{x} = \sum_{n=1}^N x_n^2 = 136951$$

Solution: Equation for b

Differentiate wrt. b and equate to zero

$$\begin{aligned} E &= \sum_{n=1}^N (y_n - f(x_n))^2 = \sum_{n=1}^N (y_n - ax_n - b)^2 \\ \frac{dE}{db} &= \sum_{n=1}^N -2(y_n - ax_n - b) \\ &= -2 \sum_{n=1}^N y_n + 2a \sum_{n=1}^N x_n + 2Nb \\ &= -2\bar{y} + 2a\bar{x} + 2Nb = 0 \\ \Rightarrow \underline{\bar{x} \cdot a + N \cdot b} &= \bar{y} \end{aligned}$$

Some useful definitions

$$\bar{x} = \sum_{n=1}^N x_n = 999$$

$$\bar{y} = \sum_{n=1}^N y_n = 37.3$$

$$\overline{xy} = \sum_{n=1}^N x_n y_n = 4914.5$$

$$\overline{x^2} = \sum_{n=1}^N x_n^2 = 136951$$

Solution: Two equations in two unknowns

Two equations

$$\overline{xx} \cdot a + \bar{x} \cdot b = \overline{xy}$$

$$\bar{x} \cdot a + N \cdot b = \bar{y}$$

In matrix notation

$$\begin{bmatrix} \overline{xx} & \bar{x} \\ \bar{x} & N \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \overline{xy} \\ \bar{y} \end{bmatrix}$$

Some useful definitions

$$\bar{x} = \sum_{n=1}^N x_n = 999$$

$$\bar{y} = \sum_{n=1}^N y_n = 37.3$$

$$\overline{xy} = \sum_{n=1}^N x_n y_n = 4914.5$$

$$\overline{xx} = \sum_{n=1}^N x_n^2 = 136951$$

Solution by substitution

Solve for b in eq. (B)

$$b = \frac{\bar{y} - \bar{x} \cdot a}{N}$$

Insert in eq. (A)

$$\bar{x}\bar{x} \cdot a + \bar{x} \cdot \underbrace{\frac{\bar{y} - \bar{x} \cdot a}{N}}_b = \bar{x}\bar{y}$$

and solve for a

$$\begin{aligned} a &= \frac{N \cdot \bar{x}\bar{y} - \bar{x} \cdot \bar{y}}{N \cdot \bar{x}\bar{x} - \bar{x}^2} \\ &= \frac{8 \cdot 4914.5 - 999 \cdot 37.3}{8 \cdot 136951 - 999^2} \approx \underline{0.0210} \end{aligned}$$

Insert, and solve for b

$$b = \frac{37.3 - 999 \cdot 0.0210}{8} \approx \underline{2.04}$$

Equations

$$\begin{aligned} (A) \quad &\bar{x}\bar{x} \cdot a + \bar{x} \cdot b = \bar{x}\bar{y} \\ (B) \quad &\bar{x} \cdot a + N \cdot b = \bar{y} \end{aligned}$$

Constants

$$\bar{x} = 999$$

$$\bar{y} = 37.3$$

$$\bar{x}\bar{y} = 4914.5$$

$$\bar{x}\bar{x} = 136951$$

Solution by solving matrix equation in Python

Two equations in matrix notation

$$\begin{bmatrix} \bar{x}\bar{x} & \bar{x} \\ \bar{x} & N \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \bar{x}\bar{y} \\ \bar{y} \end{bmatrix}$$

```
>>> N, x, y, xy, xx = 8, 999, 37.3, 4914.5, 136951
```

```
>>> X = np.array([[xx, x],[x, N]])
>>> print(X)
[[136951    999]
 [   999     8]]
```

```
>>> y = np.array([xy, y])
>>> print(y)
[4914.5  37.3]
```

```
>>> a, b = np.linalg.solve(X, y)
>>> print(f'a = {a:.3}, b = {b:.3}')
a = 0.021, b = 2.04
```

Some useful definitions

$$\bar{x} = \sum_{n=1}^N x_n = 999$$

$$\bar{y} = \sum_{n=1}^N y_n = 37.3$$

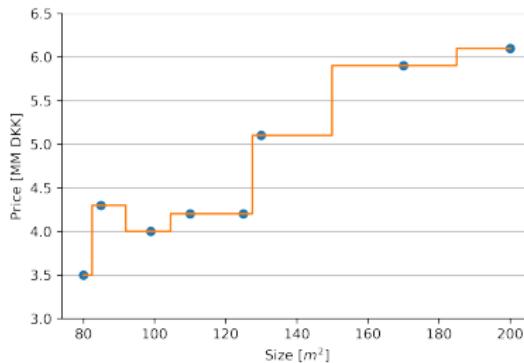
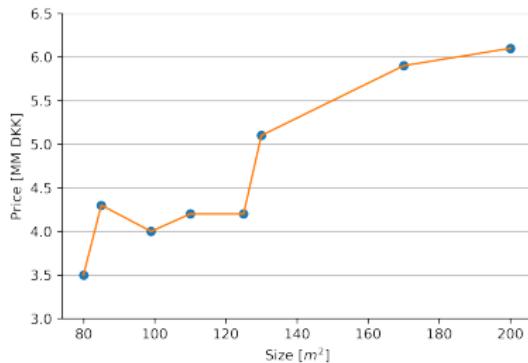
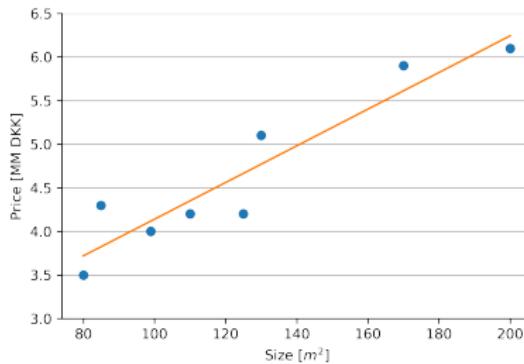
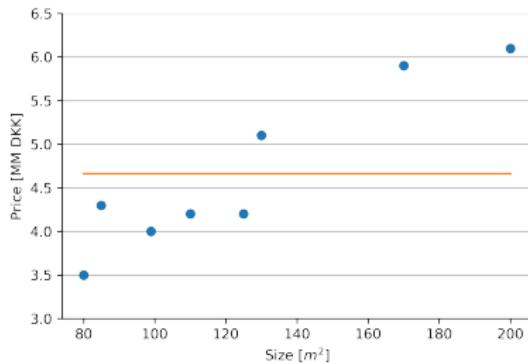
$$\bar{x}\bar{y} = \sum_{n=1}^N x_n y_n = 4914.5$$

$$\bar{x}\bar{x} = \sum_{n=1}^N x_n^2 = 136951$$

Generalization

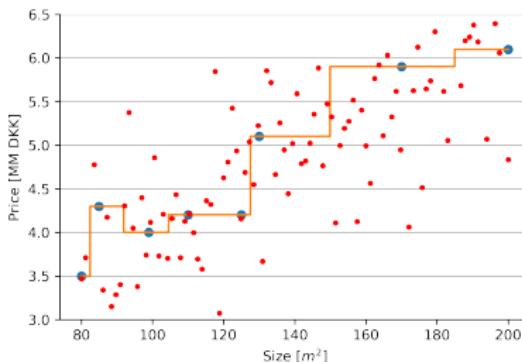
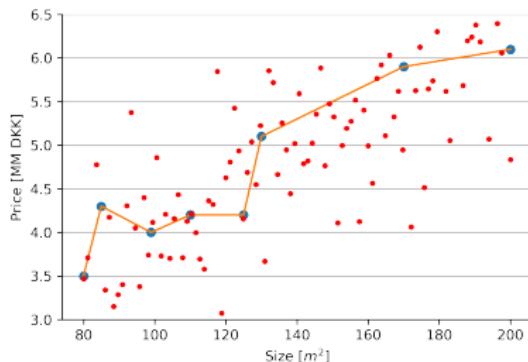
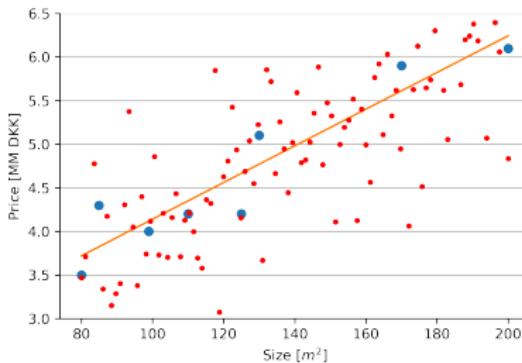
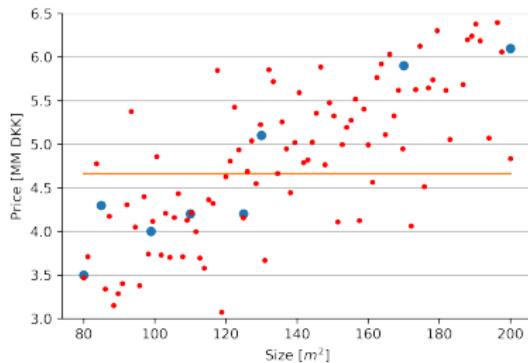
House price regression: Generalization

- If we knew future house prices, we could measure generalization error



House price regression: Generalization

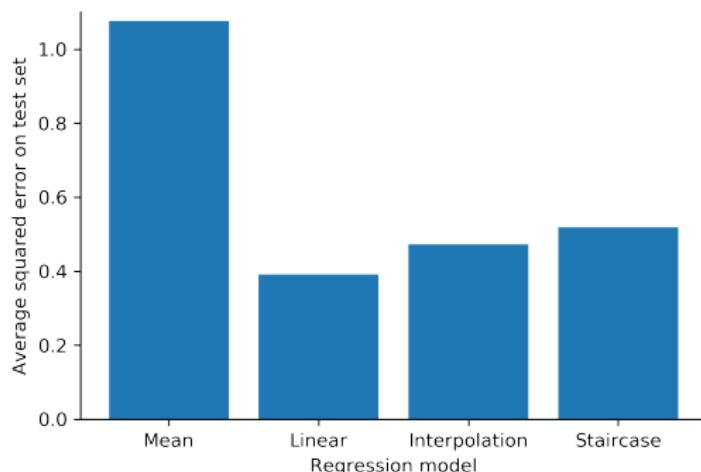
- If we knew future house prices, we could measure generalization error



House price regression: Generalization

Generalization error / out-of-sample error

- Average error on future data

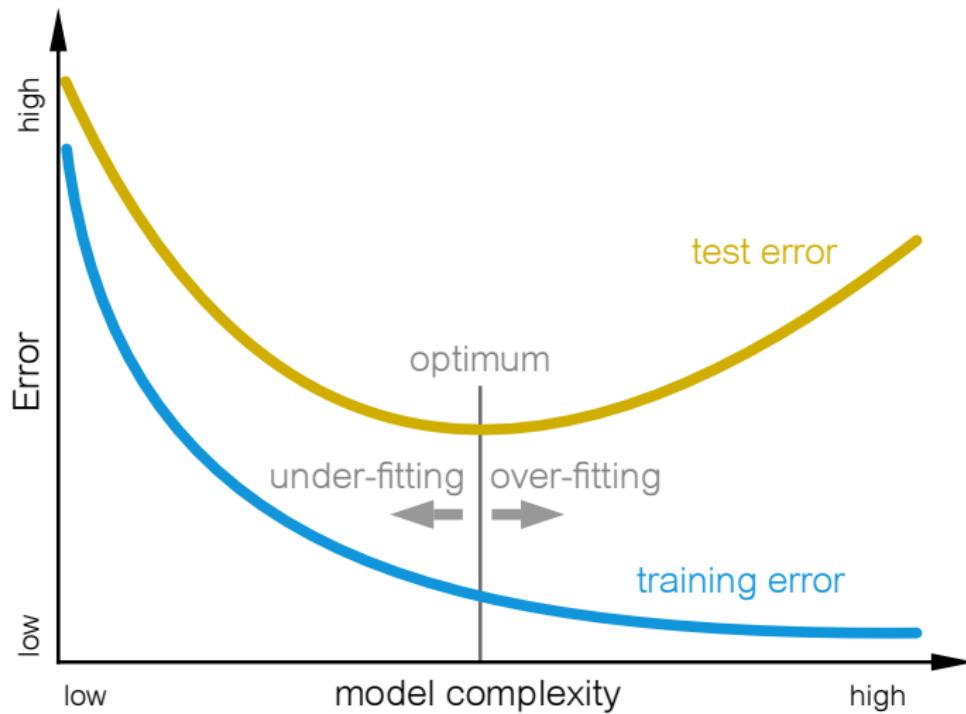


But, of course, we don't actually have access to future data

Cross-validation

- We only have access to a finite data sample
- Split the data sample into two parts called the *training set* and the *test set*
- Fit the models using the training set
- Evaluate and compare model performance on the test set

Model complexity



Tasks

Tasks for today

Tasks today

1. Work through the *regression complexity* notebook
`06-RegressionComplexity.ipynb`

Today's feedback group

- David Svane-Petersen
- Yuxuan Zhang
- sebastian vargr
- Peter Vestereng Larsen

Introduction to intelligent systems

Image processing

Mikkel N. Schmidt

Technical University of Denmark,
DTU Compute, Department of Applied Mathematics and Computer Science.

Overview

- ① K-means clustering
- ② Data geometry: Image data
- ③ Technical writing
- ④ Tasks

Feedback group

- Carl Borg
- Magnus Nordtorp Mabeck
- Aleks Laith Gryn
- Christine Amalie Meinert Cardel

Learning objectives

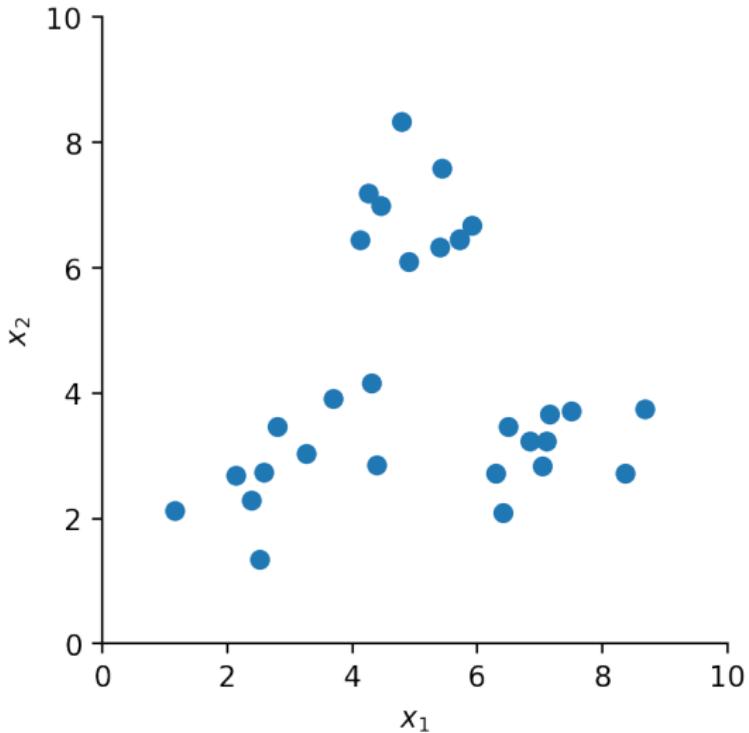
- II** Feature normalization and standardization.
 - II** K-means clustering. Model, cost function, parameters, and algorithm.
-
- I Understand the concepts and definitions, and know their application. Reason about the concepts in the context of an example. Use correct technical terminology.
 - II As above plus: Read, manipulate, and work with technical definitions and expressions (mathematical and Python code). Carry out practical computations. Interpret and evaluate results.

K-means clustering

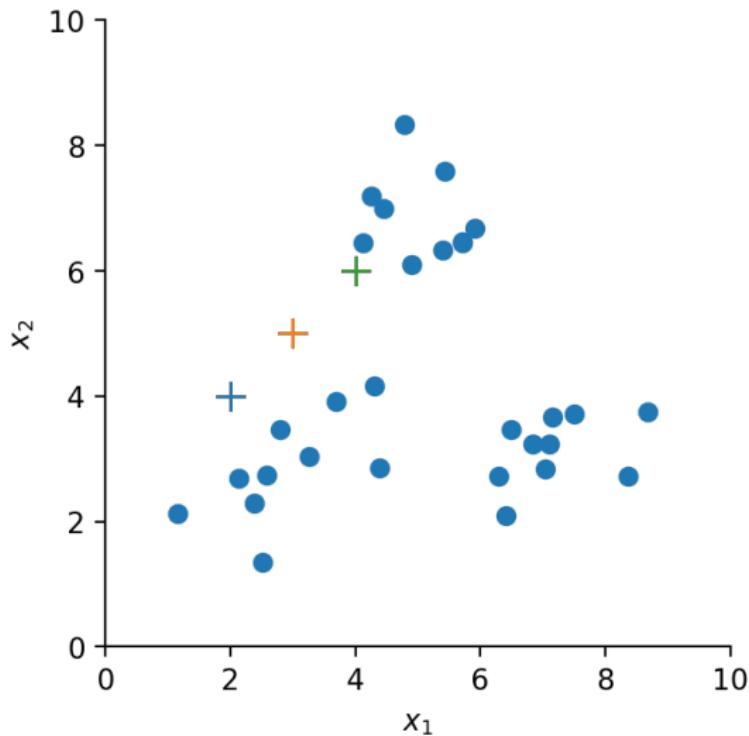
K-means clustering

- Basic idea: Group n observations into K clusters
- Observations belong to the cluster with the closest mean
- The cluster mean serves as a prototype of the cluster

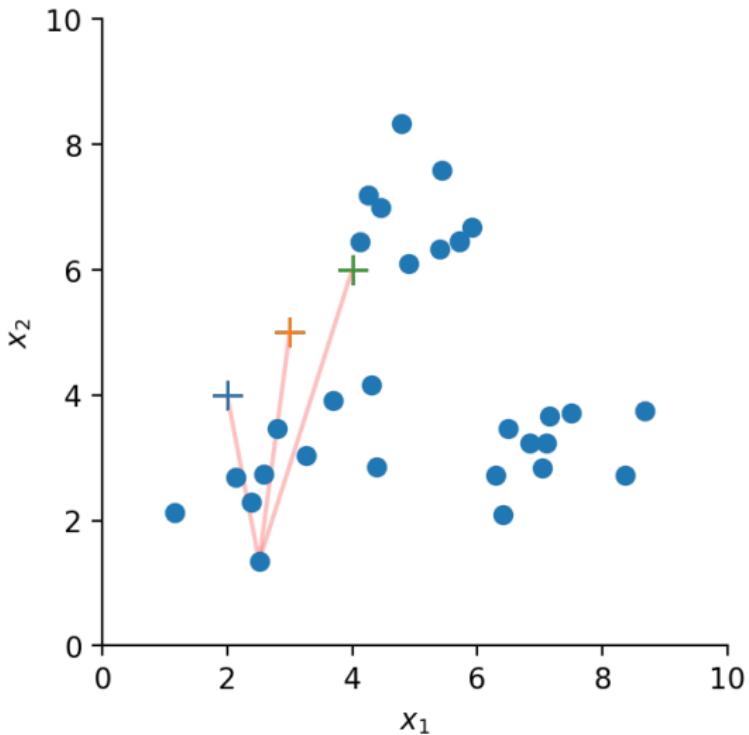
K-means example



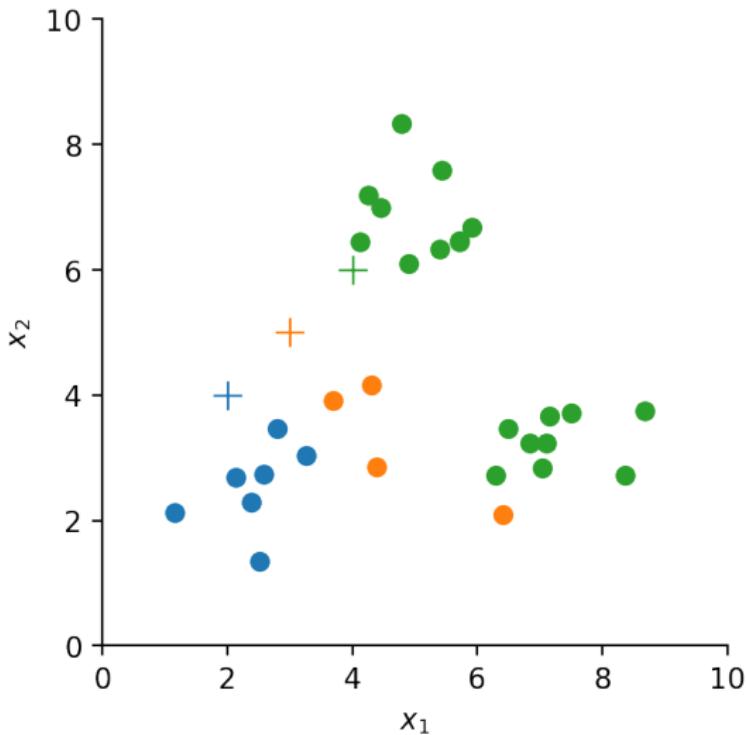
K-means example



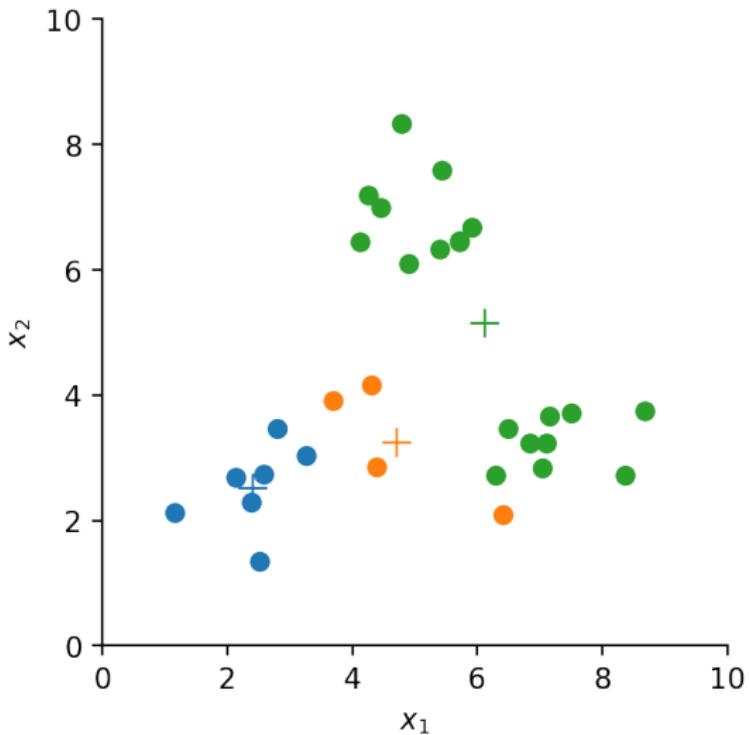
K-means example



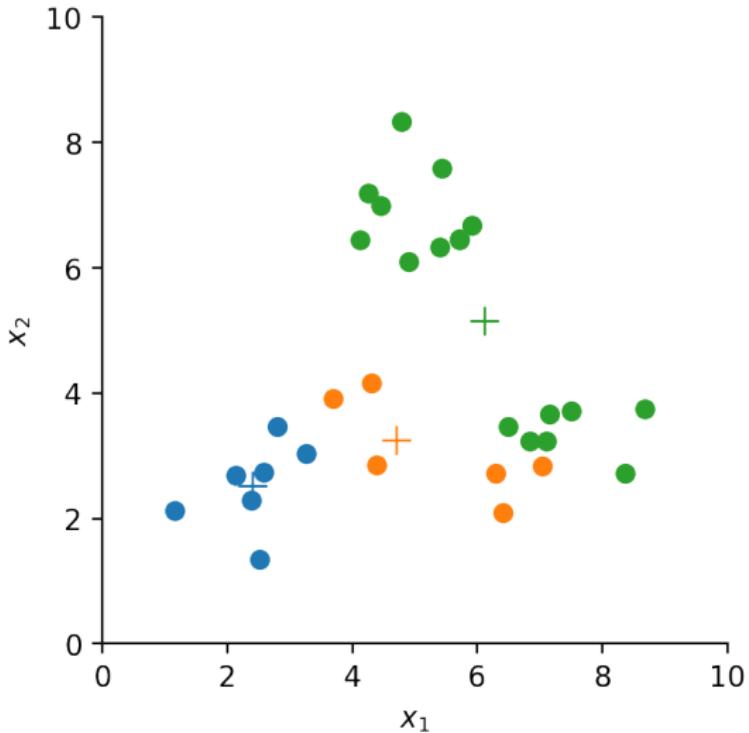
K-means example



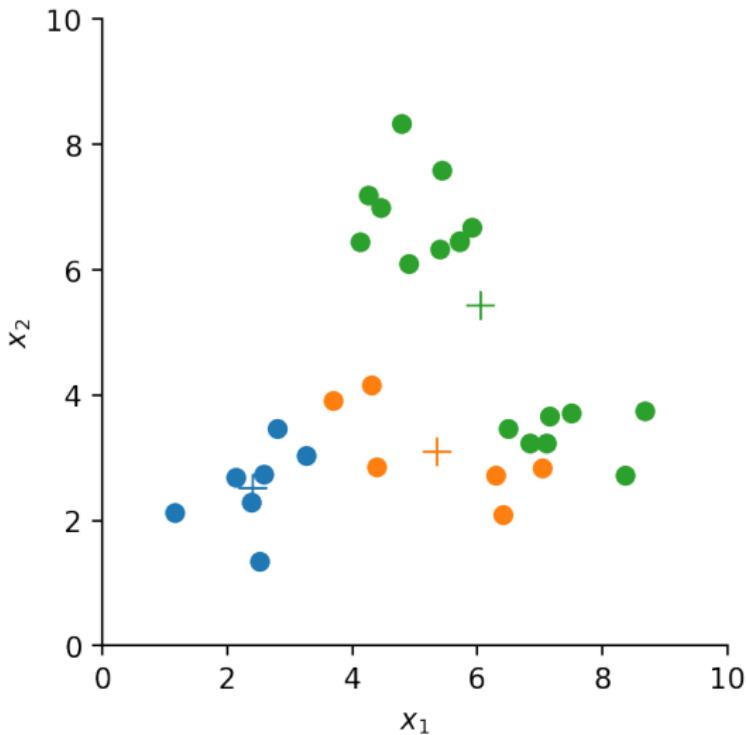
K-means example



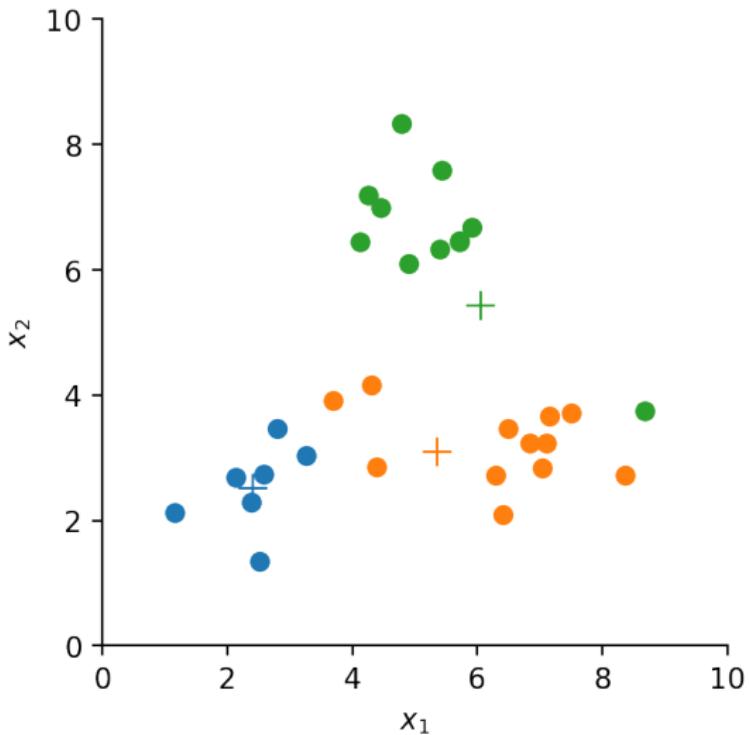
K-means example



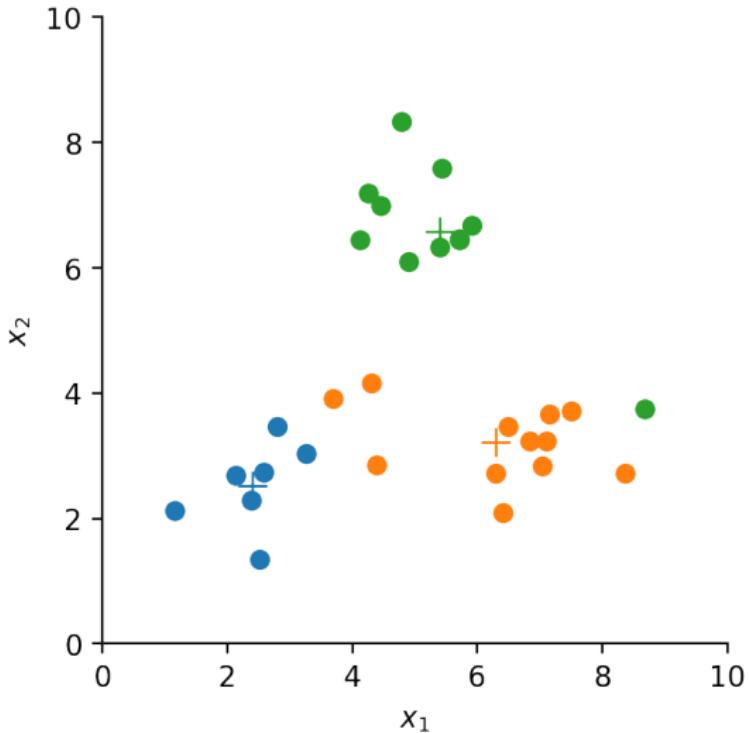
K-means example



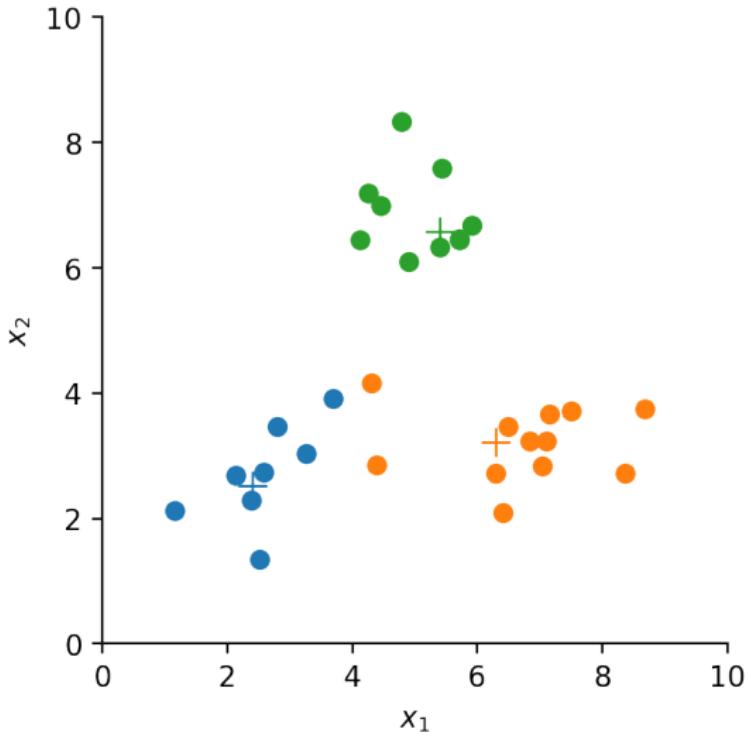
K-means example



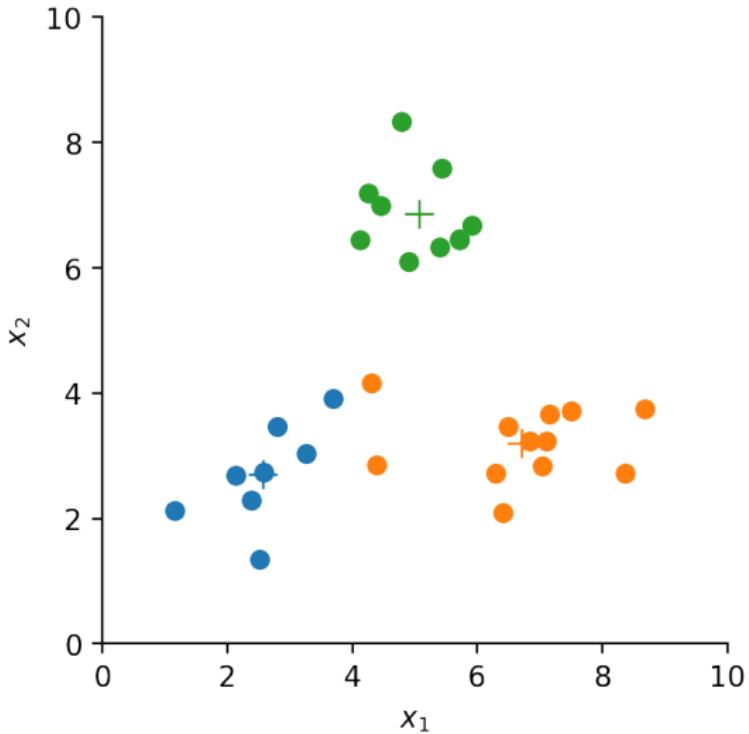
K-means example



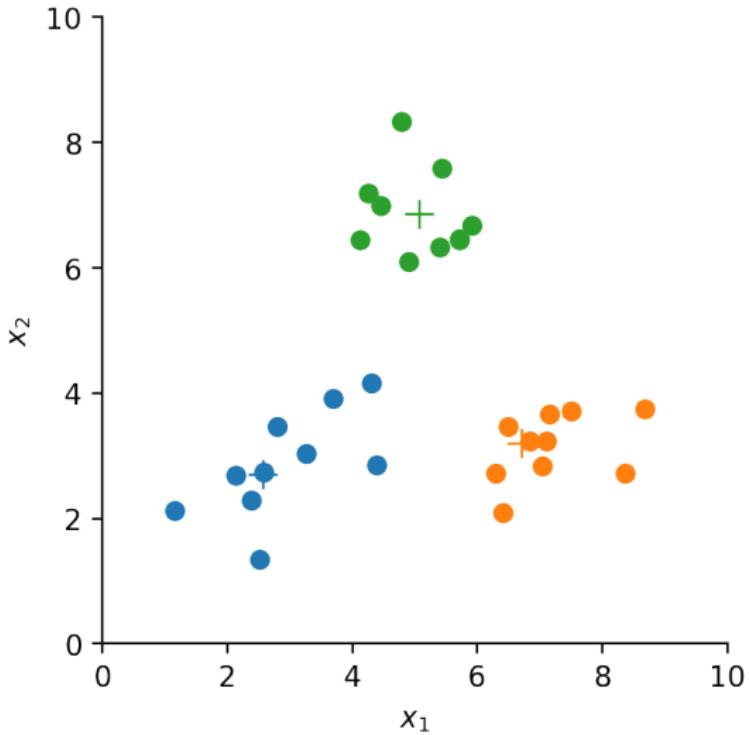
K-means example



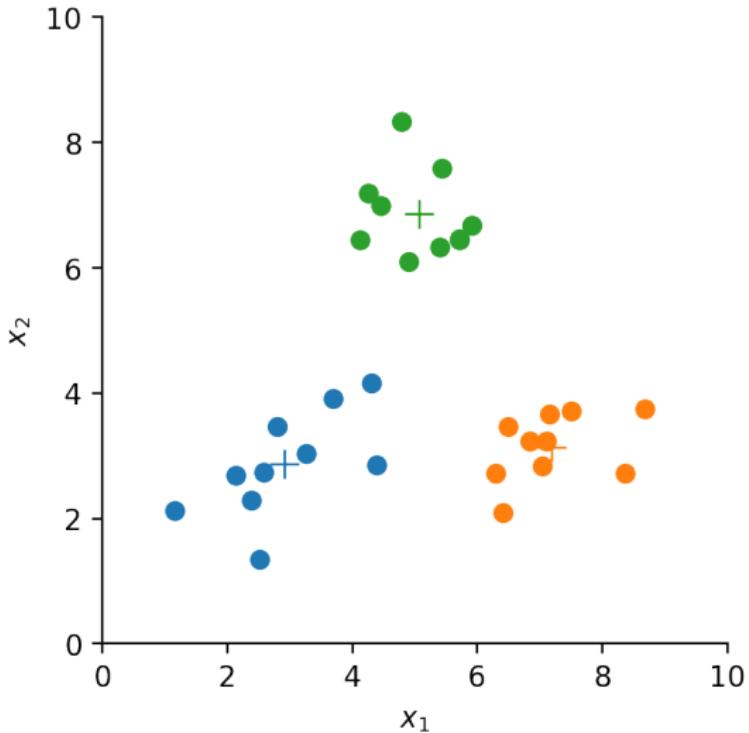
K-means example



K-means example



K-means example



K-means algorithm

Minimization of the objective

$$\underbrace{\min_{\{z_1, \dots, z_K\}}}_{\text{Cluster means}} \quad \underbrace{\min_{\{c_1, \dots, c_N\}}}_{\text{Cluster assignments}} \quad \underbrace{\sum_{n=1}^N \|x_n - z_{c_n}\|^2}_{\text{Squared distance to cluster mean}}$$

K-means algorithm

Minimization of the objective

$$\underbrace{\min_{\{z_1, \dots, z_K\}}}_{\text{Cluster means}} \quad \underbrace{\min_{\{c_1, \dots, c_N\}}}_{\text{Cluster assignments}} \quad \underbrace{\sum_{n=1}^N \|x_n - z_{c_n}\|^2}_{\text{Squared distance to cluster mean}}$$

Algorithm

1. Fix cluster means, optimize cluster assignment

$$\min_{\{c_1, \dots, c_N\}} \sum_{n=1}^N \|x_n - z_{c_n}\|^2$$

K-means algorithm

Minimization of the objective

$$\underbrace{\min_{\{z_1, \dots, z_K\}}}_{\text{Cluster means}} \quad \underbrace{\min_{\{c_1, \dots, c_N\}}}_{\text{Cluster assignments}} \quad \underbrace{\sum_{n=1}^N \|x_n - z_{c_n}\|^2}_{\text{Squared distance to cluster mean}}$$

Algorithm

1. Fix cluster means, optimize cluster assignment

$$\min_{\{c_1, \dots, c_N\}} \sum_{n=1}^N \|x_n - z_{c_n}\|^2$$

2. Fix cluster assignments, optimize cluster means

$$\min_{\{z_1, \dots, z_K\}} \sum_{n=1}^N \|x_n - z_{c_n}\|^2$$

Notation for the objective

The objective can be written as a sum over all data points

$$L = \sum_{n=1}^N \|\mathbf{x}_n - \mathbf{z}_{c_n}\|^2$$

Notation for the objective

The objective can be written as a sum over all data points

$$L = \sum_{n=1}^N \|\mathbf{x}_n - \mathbf{z}_{c_n}\|^2$$

or as a sum over all data points in each cluster inside a sum over all clusters

$$L = \underbrace{\sum_{k=1}^K}_{\text{Clusters}} \underbrace{\sum_{n: c_n=k} \|\mathbf{x}_n - \mathbf{z}_k\|^2}_{\text{Observations in cluster } k}$$

Notation for the distances

- The squared distance from a data point to a cluster is the squared norm of the difference

$$\|\mathbf{x}_n - \mathbf{z}_k\|^2$$

where \mathbf{x}_n and \mathbf{z}_k are vectors.

Notation for the distances

- The squared distance from a data point to a cluster is the squared norm of the difference

$$\|\mathbf{x}_n - \mathbf{z}_k\|^2$$

where \mathbf{x}_n and \mathbf{z}_k are vectors.

- In two dimension, for example, we have

$$\mathbf{x}_n = \begin{bmatrix} x_n^{(1)} \\ x_n^{(2)} \end{bmatrix}, \quad \mathbf{z}_k = \begin{bmatrix} z_k^{(1)} \\ z_k^{(2)} \end{bmatrix}$$

so the squared distance is given as

$$\|\mathbf{x}_n - \mathbf{z}_k\|^2 = (x_n^{(1)} - z_k^{(1)})^2 + (x_n^{(2)} - z_k^{(2)})^2$$

Derivative with respect to a vector

- The partial derivative of the squared difference with respect to one component

$$\begin{aligned}\frac{\partial}{\partial z_k^{(d)}} \|\mathbf{x}_n - \mathbf{z}_k\|^2 &= \frac{\partial}{\partial z_k^{(d)}} \left((x_n^{(1)} - z_k^{(1)})^2 + (x_n^{(2)} - z_k^{(2)})^2 \right) \\ &= -2(\mathbf{x}_n^{(d)} - \mathbf{z}_k^{(d)})\end{aligned}$$

Derivative with respect to a vector

- The partial derivative of the squared difference with respect to one component

$$\begin{aligned}\frac{\partial}{\partial z_k^{(d)}} \|\mathbf{x}_n - \mathbf{z}_k\|^2 &= \frac{\partial}{\partial z_k^{(d)}} \left((x_n^{(1)} - z_k^{(1)})^2 + (x_n^{(2)} - z_k^{(2)})^2 \right) \\ &= -2(\mathbf{x}_n^{(d)} - \mathbf{z}_k^{(d)})\end{aligned}$$

- All partial derivatives can be collected in a vector

$$\frac{\partial L}{\partial \mathbf{z}_k} = \left[\begin{array}{c} \frac{\partial L}{\partial z_k^{(1)}} \\ \frac{\partial L}{\partial z_k^{(2)}} \end{array} \right] = \left[\begin{array}{c} -2(x_n^{(1)} - z_k^{(1)}) \\ -2(x_n^{(2)} - z_k^{(2)}) \end{array} \right] = -2(\mathbf{x}_n - \mathbf{z}_k)$$

Derivative with respect to a vector

- The partial derivative of the squared difference with respect to one component

$$\begin{aligned}\frac{\partial}{\partial z_k^{(d)}} \|\mathbf{x}_n - \mathbf{z}_k\|^2 &= \frac{\partial}{\partial z_k^{(d)}} \left((x_n^{(1)} - z_k^{(1)})^2 + (x_n^{(2)} - z_k^{(2)})^2 \right) \\ &= -2(\mathbf{x}_n^{(d)} - \mathbf{z}_k^{(d)})\end{aligned}$$

- All partial derivatives can be collected in a vector

$$\frac{\partial L}{\partial \mathbf{z}_k} = \left[\begin{array}{c} \frac{\partial L}{\partial z_k^{(1)}} \\ \frac{\partial L}{\partial z_k^{(2)}} \end{array} \right] = \left[\begin{array}{c} -2(x_n^{(1)} - z_k^{(1)}) \\ -2(x_n^{(2)} - z_k^{(2)}) \end{array} \right] = -2(\mathbf{x}_n - \mathbf{z}_k)$$

- So we have the rule

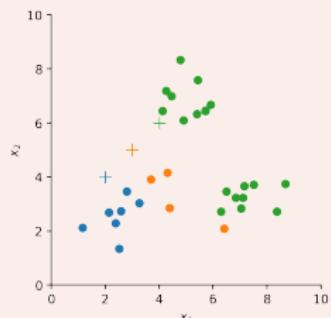
$$\frac{\partial}{\partial z_k} \|\mathbf{x}_n - \mathbf{z}_k\|^2 = -2(\mathbf{x}_n - \mathbf{z}_k)$$

Exercise: Optimal cluster center

Fix cluster assignments, optimize cluster means

$$\min_{\{\mathbf{z}_1, \dots, \mathbf{z}_K\}} \underbrace{\sum_{k=1}^K}_{\text{Clusters}} \underbrace{\sum_{n: c_n=k} \| \mathbf{x}_n - \mathbf{z}_k \|^2}_{\text{Observations in cluster } k}$$

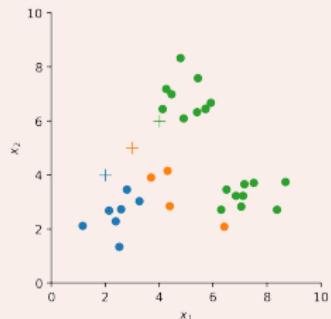
- What is the optimum value of the cluster means \mathbf{z}_k ?
- Hint: Optimize the expression by computing the derivative wrt. \mathbf{z}_k , equate to zero and solve for \mathbf{z}_k



Exercise: Optimal cluster center

Fix cluster assignments, optimize cluster means

$$\min_{\{\mathbf{z}_1, \dots, \mathbf{z}_K\}} \underbrace{\sum_{k=1}^K}_{\text{Clusters}} \underbrace{\sum_{n: c_n=k} \| \mathbf{x}_n - \mathbf{z}_k \|^2}_{\text{Observations in cluster } k}$$



- What is the optimum value of the cluster means \mathbf{z}_k ?
- Hint: Optimize the expression by computing the derivative wrt. \mathbf{z}_k , equate to zero and solve for \mathbf{z}_k

Solution

$$\frac{\partial L}{\partial \mathbf{z}_k} \sum_{n: c_n=k} -2(\mathbf{x}_n - \mathbf{z}_k) = 2N_k \mathbf{z}_k - 2 \sum_{n: c_n=k} \mathbf{x}_n = 0 \Rightarrow \mathbf{z}_k = \frac{1}{N_k} \sum_{n: c_n=k} \mathbf{x}_n$$

Exercise: Pen-and-paper k-means

Using pen-and-paper k-means, cluster the following 1-dimensional data objects

Data $\{10, 18, 32, 70, 81, 89\}$

Num. clusters $K = 2$

Initialization Set means to the first two data points

Algorithm

1. Fix cluster means
Assign each observation to closest cluster
2. Fix cluster assignments
Set cluster means to average of data points in cluster

Exercise: K-means computational complexity

Algorithm

- What is the computational complexity of the k-means algorithm?
- Express it in big-O notation in terms of the number of data points N and the number of clusters K

1. Fix cluster means,
optimize cluster assignment

$$\min_{\{c_1, \dots, c_N\}} \sum_{n=1}^N \|x_n - z_{c_n}\|^2$$

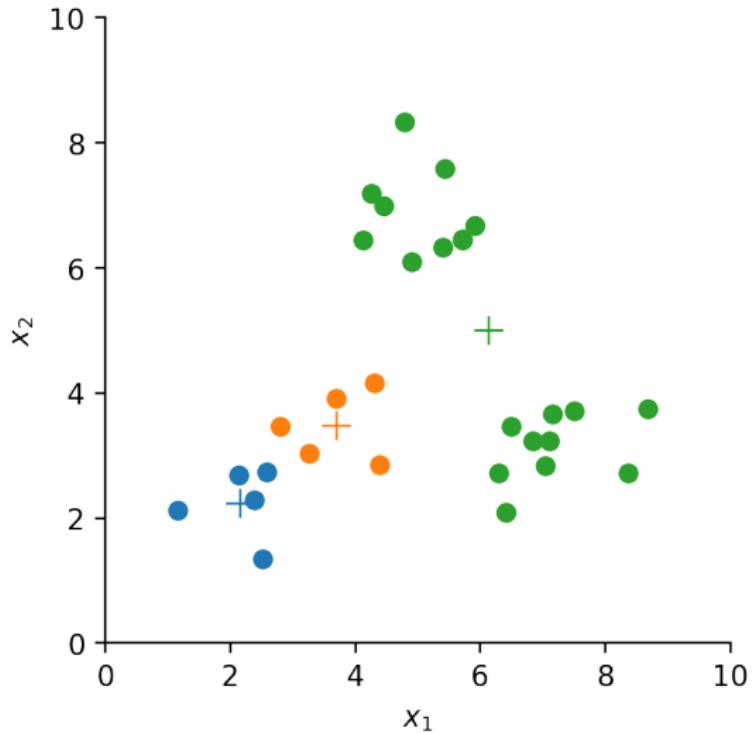
2. Fix cluster assignments,
optimize cluster means

$$\min_{\{z_1, \dots, z_K\}} \sum_{n=1}^N \|x_n - z_{c_n}\|^2$$

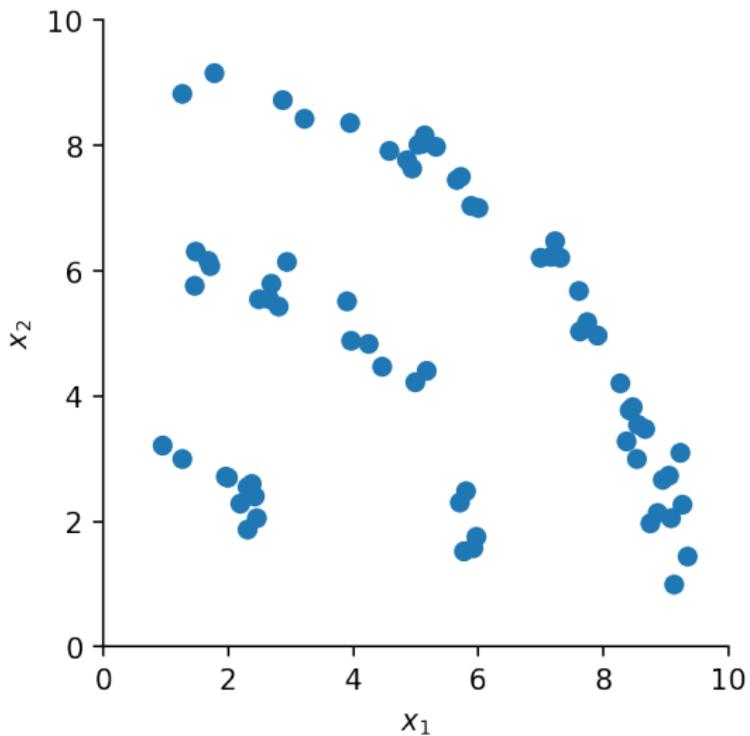
Initialization

- Algorithm requires set of initial cluster means
- Not guaranteed to converge to global optimum
- Many good heuristic initialization strategies exist

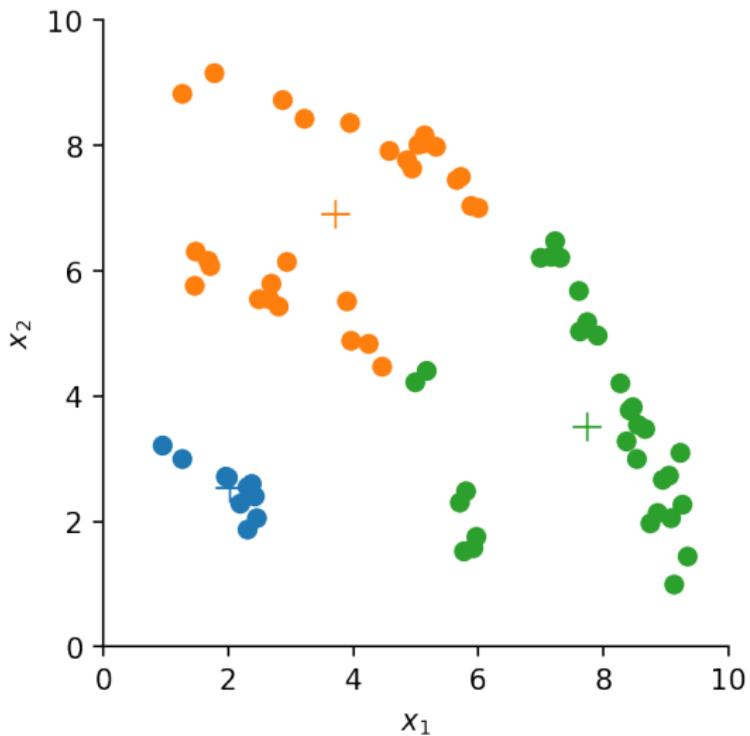
Example of local optimum



Depends on input features

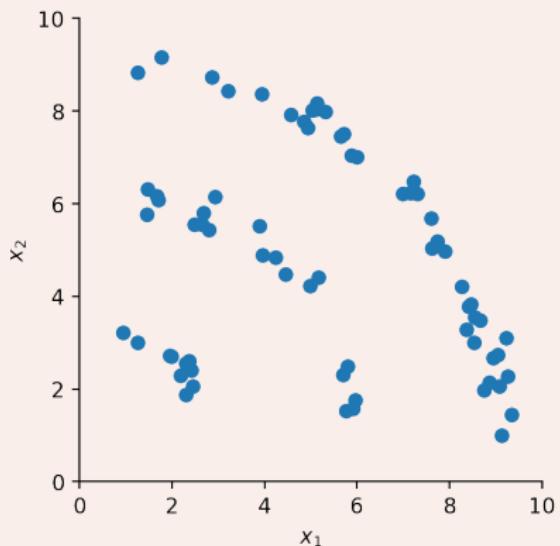


Depends on input features

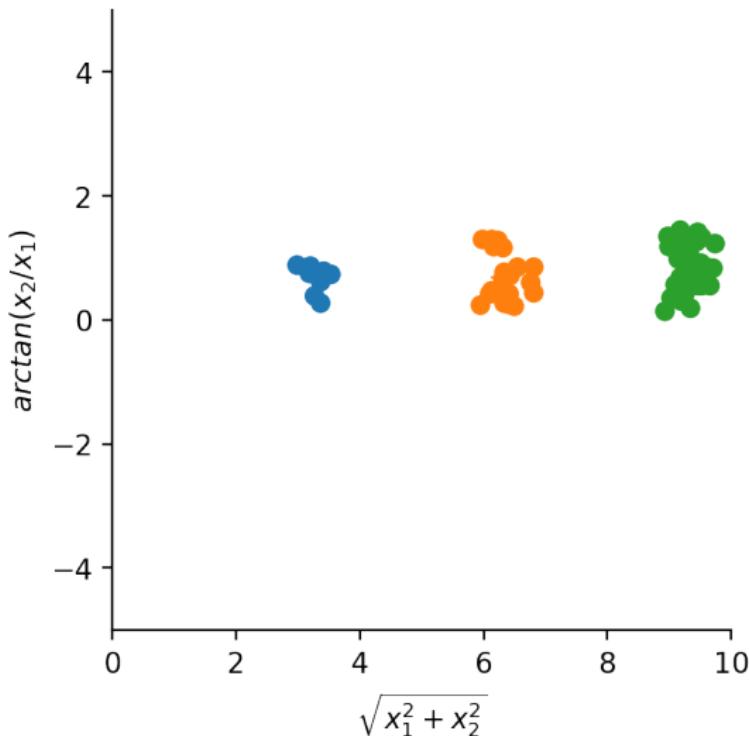


Exercise: Transformation of input features

- Can you come up with a way to transform the input features, so that k-means will find the three clusters?



Transformation of input features



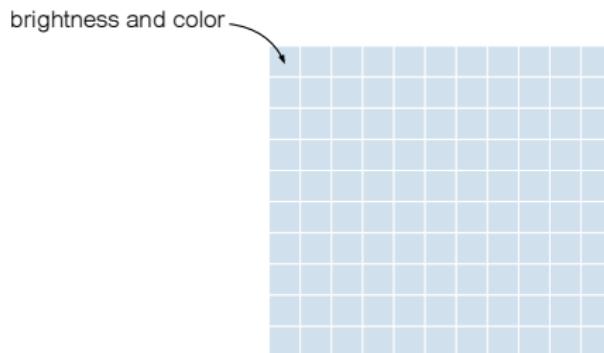
Data geometry: Image data

Exercise: What is an image?

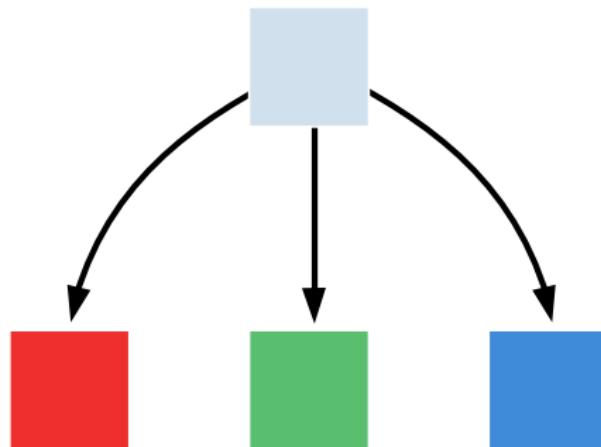
- Try to make a definition of what an *image* is without using technical terms such as pixels etc.

Image defined in technical terms

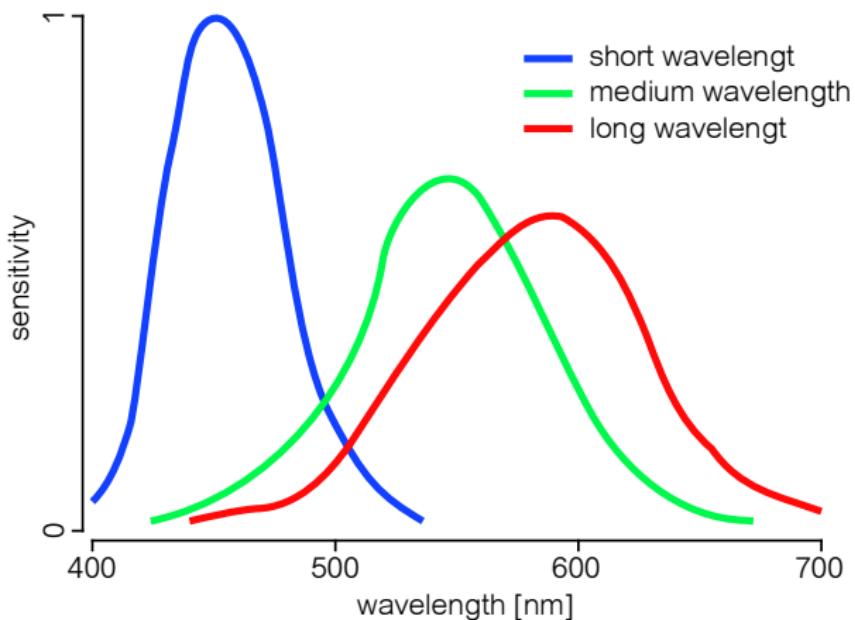
- Grid of pixels
- Every pixel contains information about color and brightness



Red, green, and blue



The eye's sensitivity to light



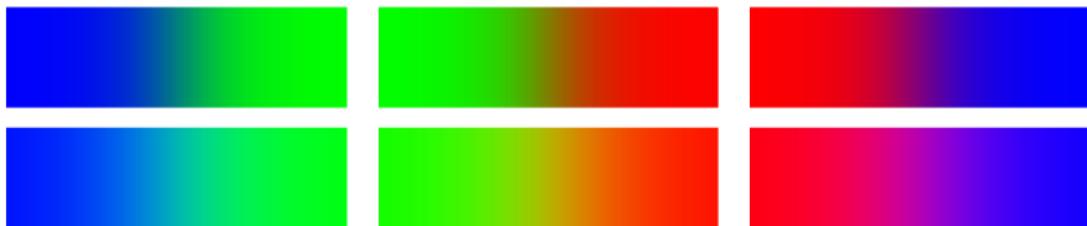
Gamma correction

- Human perception of light follows approximately a power function
- In most display systems, intensities are encoded non-linearly as

$$v_{\text{out}} = v_{\text{in}}^{\gamma}$$

- Most systems use a value of approximately $\gamma = .45$

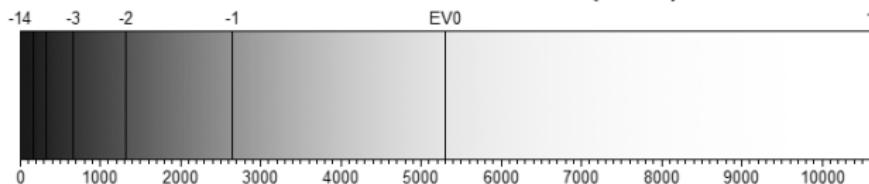
Example: Color gradients without and with gamma correction



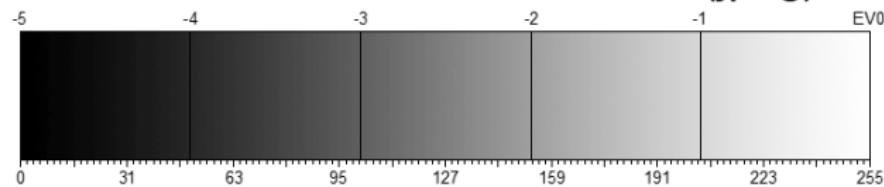
Gamma correction

Example: Linear vs. gamma corrected distribution

Linear Distribution (raw)



Gamma Corrected Distribution (jpeg)



CIELAB color space

The CIELAB color space

- Defined by the International Commission on Illumination (CIE) in 1976
- Expresses color as three numerical values,
 - L for the lightness
 - A for the green-red color component
 - B for the blue-yellow color components
- Perceptually uniform with respect to human color vision
- Same amount of numerical change = same amount of visually perceived change

Example: Image segmentation



Technical writing

The IMRaD model

Original research articles are typically structured in this basic order

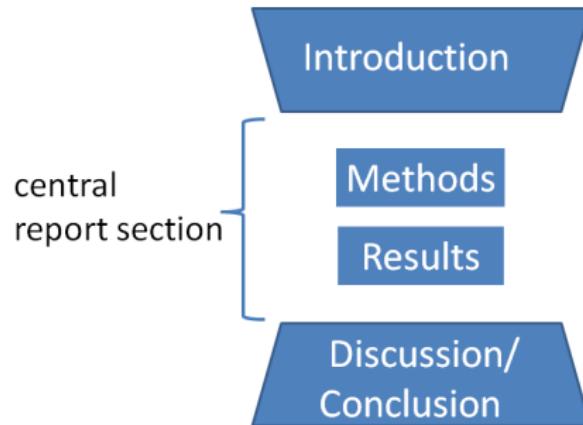
Introduction Why was the study undertaken? What was the research question, the tested hypothesis or the purpose of the research?

Methods When, where, and how was the study done? What materials were used or who was included in the study groups (patients, etc.)?

Results What answer was found to the research question; what did the study find? Was the tested hypothesis true?

Discussion What might the answer imply and why does it matter? How does it fit in with what other researchers have found? What are the perspectives for future research?

The IMRaD wineglass



IMRaD pros and cons

Pros

- Presents an idealized report of the ideas
- Clear and logical presentation
- Easy for readers to navigate

Cons

- Does not correspond to the actual research process
- Can be too rigid and simplistic

Abstract

- Brief summary at the beginning of a manuscript
- Help the reader ascertain the paper's purpose
- Should be a self-contained text
- Mirrors the IMRaD model

Abstract template

5-sentence abstract template

Motivation Why do we care?

Problem What problem will we solve?

What have others done, and why is that not enough?

Approach What is our big idea? How did we solve it?

Which research, analysis, and experiments did we do?

Results What is the answer?

Conclusions What are the implications?

Example of abstract

Machine-learning tasks frequently involve problems of manipulating and classifying large numbers of vectors in high-dimensional spaces. Classical algorithms for solving such problems typically take time polynomial in the number of vectors and the dimension of the space. Quantum computers are good at manipulating high-dimensional vectors in large tensor product spaces. This paper provides supervised and unsupervised quantum machine learning algorithms for cluster assignment and cluster finding. Quantum machine learning can take time logarithmic in both the number of vectors and their dimension, an exponential speed-up over classical algorithms.

5-sentence abstract template

Motivation Why do we care?

Problem What problem will we solve? What have others done, and why is that not enough?

Approach What is our big idea? How did we solve it? Which research, analysis, and experiments did we do?

Results What is the answer?

Conclusions What are the implications?

Example of abstract

[*Some*] tasks frequently involve problems of [*something difficult*]. Classical algorithms for solving such problems typically take [*very long time*]. Quantum computers are good at [*something really complicated*]. This paper provides [*some new algorithm*]. Quantum machine learning [*is much faster than*] classical algorithms.

5-sentence abstract template

Motivation Why do we care?

Problem What problem will we solve? What have others done, and why is that not enough?

Approach What is our big idea? How did we solve it? Which research, analysis, and experiments did we do?

Results What is the answer?

Conclusions What are the implications?

Example of abstract

Motivation *[Some]* tasks frequently involve problems of *[something difficult]*.

Problem Classical algorithms for solving such problems typically take *[very long time]*.

Approach Quantum computers are good at *[something really complicated]*.

Results This paper provides *[some new algorithm]*.

Conclusions Quantum machine learning *[is much faster than]* classical algorithms.

5-sentence abstract template

Motivation Why do we care?

Problem What problem will we solve? What have others done, and why is that not enough?

Approach What is our big idea? How did we solve it? Which research, analysis, and experiments did we do?

Results What is the answer?

Conclusions What are the implications?

Tasks

Tasks for today

Tasks today

1. Work through the two *clustering* notebooks

07-KMeansClustering.ipynb

07-ImageSegmentation.ipynb

2. Start working on Lab Report 3.

3. Today's feedback group

- Carl Borg
- Magnus Nordtorp Mabeck
- Aleks Laith Gryn
- Christine Amalie Meinert Cardel

Lab report hand in

- Lab 3: Image segmentation (Deadline: Thursday 26 October 20:00)

Introduction to intelligent systems

Optimization

Mikkel N. Schmidt

Technical University of Denmark,
DTU Compute, Department of Applied Mathematics and Computer Science.

Overview

- ① Gradient descent
- ② Linear regression (with gradient descent)
- ③ Neural network (with gradient descent)
- ④ Tasks

Feedback group

- Nicholas Borch
- Alfred Fonnesbech Aqraou
- Josefine Høgsted Voglhofer
- Rasmus Bernth Linnemann

Learning objectives

- II Gradient descent algorithm.
 - I Stochastic gradient descent.
 - II Gradient of cost function.
 - II Neural networks: Model (layers, activation functions), parameters, cost function.
-
- I Understand the concepts and definitions, and know their application. Reason about the concepts in the context of an example. Use correct technical terminology.
 - II As above plus: Read, manipulate, and work with technical definitions and expressions (mathematical and Python code). Carry out practical computations. Interpret and evaluate results.

Gradient descent

Gradient descent

- Iterative method for finding optimum of a function
- Start at an initial point
- Updates parameters by taking step proportional to negative of the gradient
- Repeat until convergence

Partial derivative

- Derivative of a function of *several variable* with respect to *one* of those variables, with the others *held constant*.
- Notation

$$\frac{\partial f}{\partial x_1}, \quad \frac{\partial f(x_1, x_2)}{\partial x_1}, \quad \frac{\partial f}{\partial x_1}(x_1, x_2)$$

- The partial derivative evaluated at a certain point

$$\left. \frac{\partial f}{\partial x_1} \right|_{x_1=5, x_2=7}$$

Partial derivative, definition

Derivative, function of single variable, $f(x)$

$$\frac{df}{dx}(x) = \lim_{h \rightarrow 0} \frac{f(x + h) - f(x)}{h}$$

Partial derivative, definition

Derivative, function of single variable, $f(x)$

$$\frac{df}{dx}(x) = \lim_{h \rightarrow 0} \frac{f(x + h) - f(x)}{h}$$

Partial derivative, function of multiple variables, $f(x_1, x_2)$

$$\frac{\partial f}{\partial x_1}(x_1, x_2) = \lim_{h \rightarrow 0} \frac{f(x_1 + h, x_2) - f(x_1, x_2)}{h}$$

Partial derivative, definition

Derivative, function of single variable, $f(x)$

$$\frac{df}{dx}(x) = \lim_{h \rightarrow 0} \frac{f(x + h) - f(x)}{h}$$

Partial derivative, function of multiple variables, $f(x_1, x_2)$

$$\frac{\partial f}{\partial x_1}(x_1, x_2) = \lim_{h \rightarrow 0} \frac{f(x_1 + h, x_2) - f(x_1, x_2)}{h}$$

$$\frac{\partial f}{\partial x_2}(x_1, x_2) = \lim_{h \rightarrow 0} \frac{f(x_1, x_2 + h) - f(x_1, x_2)}{h}$$

Gradient

Definition

$$\nabla f(x_1, x_2, \dots) = \begin{bmatrix} \frac{\partial f(x_1, x_2, \dots)}{\partial x_1} \\ \frac{\partial f(x_1, x_2, \dots)}{\partial x_2} \\ \vdots \end{bmatrix}$$

Exercise: Gradient calculation

Multivariate function

$$f(x, y) = x^2 \cos(y)$$

What is the gradient?

Gradient definition

$$\nabla f(x, y) = \begin{bmatrix} \frac{\partial f(x, y)}{\partial x} \\ \frac{\partial f(x, y)}{\partial y} \end{bmatrix}$$

Exercise: Gradient calculation

Multivariate function

$$f(x, y) = x^2 \cos(y)$$

What is the gradient?

Partial derivatives

$$\frac{\partial f(x, y)}{\partial x} = 2x \cos(y)$$

$$\frac{\partial f(x, y)}{\partial y} = -x^2 \sin(y)$$

Gradient

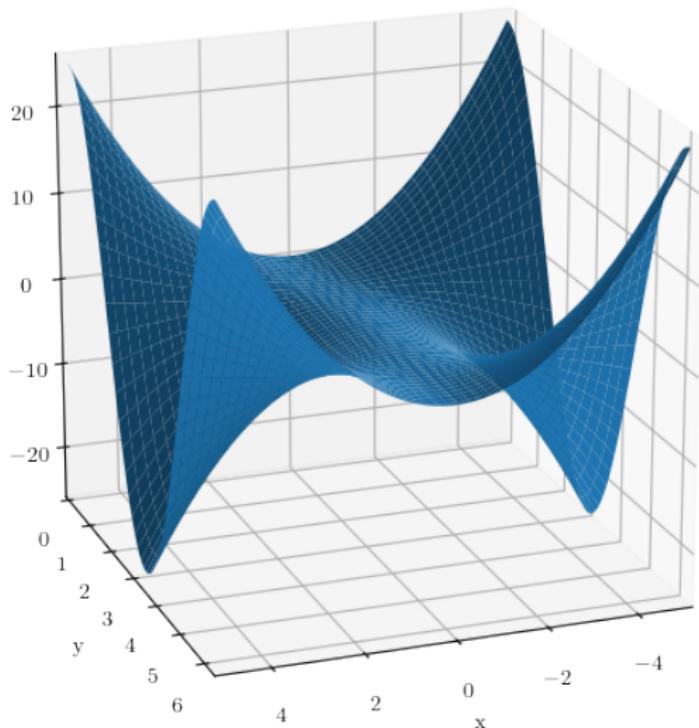
$$\nabla f(x, y) = \begin{bmatrix} \frac{\partial f(x, y)}{\partial x} \\ \frac{\partial f(x, y)}{\partial y} \end{bmatrix} = \begin{bmatrix} 2x \cos(y) \\ -x^2 \sin(y) \end{bmatrix}$$

Gradient definition

$$\nabla f(x, y) = \begin{bmatrix} \frac{\partial f(x, y)}{\partial x} \\ \frac{\partial f(x, y)}{\partial y} \end{bmatrix}$$

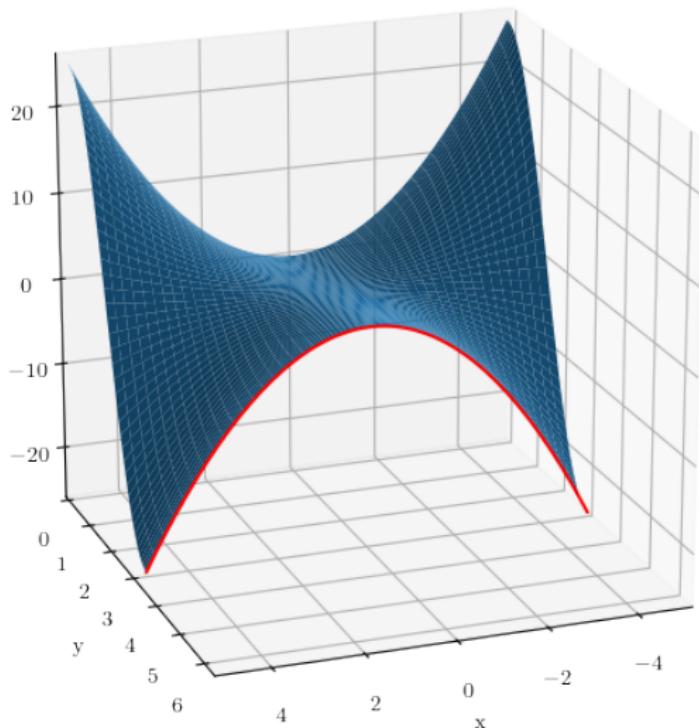
Gradient

$$f(x, y) = x^2 \cos(y)$$



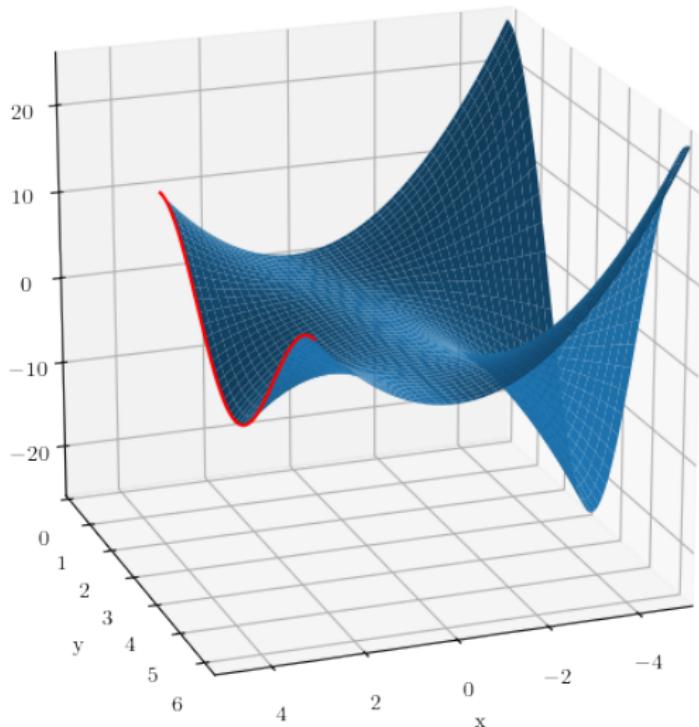
Gradient

$$f(x, y) = x^2 \cos(y)$$



Gradient

$$f(x, y) = x^2 \cos(y)$$



Gradient descent

Initialize $x^{(0)}$

Repeat, $t = 0, 1, 2, \dots$

$$\underbrace{x^{(t+1)}}_{\text{new parameter value}} = \underbrace{x^{(t)}}_{\text{old parameter value}} - \underbrace{\alpha}_{\text{step size}} \cdot \underbrace{\nabla f(x^{(t)})}_{\text{gradient}}$$

until convergence

Partial derivative in vector form

Partial derivative, function of multiple variables, $f(x_1, x_2)$

$$\frac{\partial f}{\partial x_1}(x_1, x_2) = \lim_{h \rightarrow 0} \frac{f(x_1 + h, x_2) - f(x_1, x_2)}{h}$$

$$\frac{\partial f}{\partial x_2}(x_1, x_2) = \lim_{h \rightarrow 0} \frac{f(x_1, x_2 + h) - f(x_1, x_2)}{h}$$

Partial derivative in vector form

Partial derivative, function of multiple variables, $f(x_1, x_2)$

$$\frac{\partial f}{\partial x_1}(x_1, x_2) = \lim_{h \rightarrow 0} \frac{f(x_1 + h, x_2) - f(x_1, x_2)}{h}$$

$$\frac{\partial f}{\partial x_2}(x_1, x_2) = \lim_{h \rightarrow 0} \frac{f(x_1, x_2 + h) - f(x_1, x_2)}{h}$$

Partial derivative in vector form, function of a vector, $f(\bar{x})$, where $\bar{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

$$\frac{\partial f}{\partial x_1}(\bar{x}) = \lim_{h \rightarrow 0} \frac{f(\bar{x} + h\bar{e}_1) - f(\bar{x})}{h}$$

$$\frac{\partial f}{\partial x_2}(\bar{x}) = \lim_{h \rightarrow 0} \frac{f(\bar{x} + h\bar{e}_2) - f(\bar{x})}{h}$$

$$\bar{e}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \bar{e}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Directional derivative

How much does the function change if we move

the parameters $\bar{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ in the direction $\bar{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$

Directional derivative

How much does the function change if we move

the parameters $\bar{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ in the direction $\bar{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$

$$\nabla_{\bar{v}} f(\bar{x}) = \lim_{h \rightarrow 0} \frac{f(x_1 + h \cdot v_1, x_2 + h \cdot v_2) - f(x_1, x_2)}{h}$$

Directional derivative

How much does the function change if we move

the parameters $\bar{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ in the direction $\bar{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$

$$\begin{aligned}\nabla_{\bar{v}} f(\bar{x}) &= \lim_{h \rightarrow 0} \frac{f(x_1 + h \cdot v_1, x_2 + h \cdot v_2) - f(x_1, x_2)}{h} \\ &= \lim_{h \rightarrow 0} \frac{f(\bar{x} + h\bar{v}) - f(\bar{x})}{h}\end{aligned}$$

Directional derivative

How much does the function change if we move

the parameters $\bar{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ in the direction $\bar{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$

$$\begin{aligned}\nabla_{\bar{v}} f(\bar{x}) &= \lim_{h \rightarrow 0} \frac{f(x_1 + h \cdot v_1, x_2 + h \cdot v_2) - f(x_1, x_2)}{h} \\ &= \lim_{h \rightarrow 0} \frac{f(\bar{x} + h\bar{v}) - f(\bar{x})}{h} \\ &= \nabla f(\bar{x}) \cdot \bar{v} \quad \leftarrow \text{We will show this}\end{aligned}$$

Directional derivative: Proof

$$\nabla_{\bar{v}} f(\bar{x}) = \lim_{h \rightarrow 0} \frac{f(x_1 + h \cdot v_1, x_2 + h \cdot v_2) - f(x_1, x_2)}{h} = \lim_{h \rightarrow 0} \frac{f(\bar{x} + h\bar{v}) - f(\bar{x})}{h}, \quad \bar{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

Directional derivative: Proof

$$\nabla_{\bar{v}} f(\bar{x}) = \lim_{h \rightarrow 0} \frac{f(x_1 + h \cdot v_1, x_2 + h \cdot v_2) - f(x_1, x_2)}{h} = \lim_{h \rightarrow 0} \frac{f(\bar{x} + h\bar{v}) - f(\bar{x})}{h}, \quad \bar{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$g(h) = f(\underbrace{x_1 + h \cdot v_1}_{z_1(h)}, \underbrace{x_2 + h \cdot v_2}_{z_2(h)}) = f(\bar{x} + h\bar{v})$$

Directional derivative: Proof

$$\nabla_{\bar{v}} f(\bar{x}) = \lim_{h \rightarrow 0} \frac{f(x_1 + h \cdot v_1, x_2 + h \cdot v_2) - f(x_1, x_2)}{h} = \lim_{h \rightarrow 0} \frac{f(\bar{x} + h\bar{v}) - f(\bar{x})}{h}, \quad \bar{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$g(h) = f(\underbrace{x_1 + h \cdot v_1}_{z_1(h)}, \underbrace{x_2 + h \cdot v_2}_{z_2(h)}) = f(\bar{x} + h\bar{v})$$

$$\frac{dg}{dh}\Big|_{h=0} = \lim_{\epsilon \rightarrow 0} \frac{g(0 + \epsilon) - g(0)}{\epsilon}$$

Directional derivative: Proof

$$\nabla_{\bar{v}} f(\bar{x}) = \lim_{h \rightarrow 0} \frac{f(x_1 + h \cdot v_1, x_2 + h \cdot v_2) - f(x_1, x_2)}{h} = \lim_{h \rightarrow 0} \frac{f(\bar{x} + h\bar{v}) - f(\bar{x})}{h}, \quad \bar{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$g(h) = f(\underbrace{x_1 + h \cdot v_1}_{z_1(h)}, \underbrace{x_2 + h \cdot v_2}_{z_2(h)}) = f(\bar{x} + h\bar{v})$$

$$\begin{aligned} \frac{dg}{dh} \Big|_{h=0} &= \lim_{\epsilon \rightarrow 0} \frac{g(0 + \epsilon) - g(0)}{\epsilon} \\ &= \lim_{\epsilon \rightarrow 0} \frac{f(\bar{x} + \epsilon\bar{v}) - f(\bar{x})}{\epsilon} = \underline{\nabla_{\bar{v}} f(\bar{x})} \end{aligned}$$

Directional derivative: Proof

$$\nabla_{\bar{v}} f(\bar{x}) = \lim_{h \rightarrow 0} \frac{f(x_1 + h \cdot v_1, x_2 + h \cdot v_2) - f(x_1, x_2)}{h} = \lim_{h \rightarrow 0} \frac{f(\bar{x} + h\bar{v}) - f(\bar{x})}{h}, \quad \bar{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$g(h) = f(\underbrace{x_1 + h \cdot v_1}_{z_1(h)}, \underbrace{x_2 + h \cdot v_2}_{z_2(h)}) = f(\bar{x} + h\bar{v})$$

$$\begin{aligned} \frac{dg}{dh} \Big|_{h=0} &= \lim_{\epsilon \rightarrow 0} \frac{g(0 + \epsilon) - g(0)}{\epsilon} \\ &= \lim_{\epsilon \rightarrow 0} \frac{f(\bar{x} + \epsilon\bar{v}) - f(\bar{x})}{\epsilon} = \underline{\nabla_{\bar{v}} f(\bar{x})} \end{aligned}$$

$$= \frac{\partial f}{\partial z_1} \frac{\partial z_1}{\partial h} + \frac{\partial f}{\partial z_2} \frac{\partial z_2}{\partial h}$$

Directional derivative: Proof

$$\nabla_{\bar{v}} f(\bar{x}) = \lim_{h \rightarrow 0} \frac{f(x_1 + h \cdot v_1, x_2 + h \cdot v_2) - f(x_1, x_2)}{h} = \lim_{h \rightarrow 0} \frac{f(\bar{x} + h\bar{v}) - f(\bar{x})}{h}, \quad \bar{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$g(h) = f(\underbrace{x_1 + h \cdot v_1}_{z_1(h)}, \underbrace{x_2 + h \cdot v_2}_{z_2(h)}) = f(\bar{x} + h\bar{v})$$

$$\begin{aligned} \frac{dg}{dh}\Big|_{h=0} &= \lim_{\epsilon \rightarrow 0} \frac{g(0 + \epsilon) - g(0)}{\epsilon} \\ &= \lim_{\epsilon \rightarrow 0} \frac{f(\bar{x} + \epsilon\bar{v}) - f(\bar{x})}{\epsilon} = \underline{\nabla_{\bar{v}} f(\bar{x})} \end{aligned}$$

$$\begin{aligned} &= \frac{\partial f}{\partial z_1} \frac{\partial z_1}{\partial h} + \frac{\partial f}{\partial z_2} \frac{\partial z_2}{\partial h} \\ &= \frac{\partial f}{\partial z_1} v_1 + \frac{\partial f}{\partial z_2} v_2 = \underline{\nabla f \cdot \bar{v}} \end{aligned}$$

Direction of steepest descent

Directional derivative

$$\nabla_{\bar{v}} f(\bar{x}) = \nabla f(\bar{x}) \cdot \bar{v}$$

- Measures how much the function changes when we move a bit in the direction \bar{v}

Direction of steepest descent

Directional derivative

$$\nabla_{\bar{v}} f(\bar{x}) = \nabla f(\bar{x}) \cdot \bar{v}$$

- Measures how much the function changes when we move a bit in the direction \bar{v}

Which direction maximizes the directional derivative?

Direction of steepest descent

Directional derivative

$$\nabla_{\bar{v}} f(\bar{x}) = \nabla f(\bar{x}) \cdot \bar{v}$$

- Measures how much the function changes when we move a bit in the direction \bar{v}

Which direction maximizes the directional derivative?

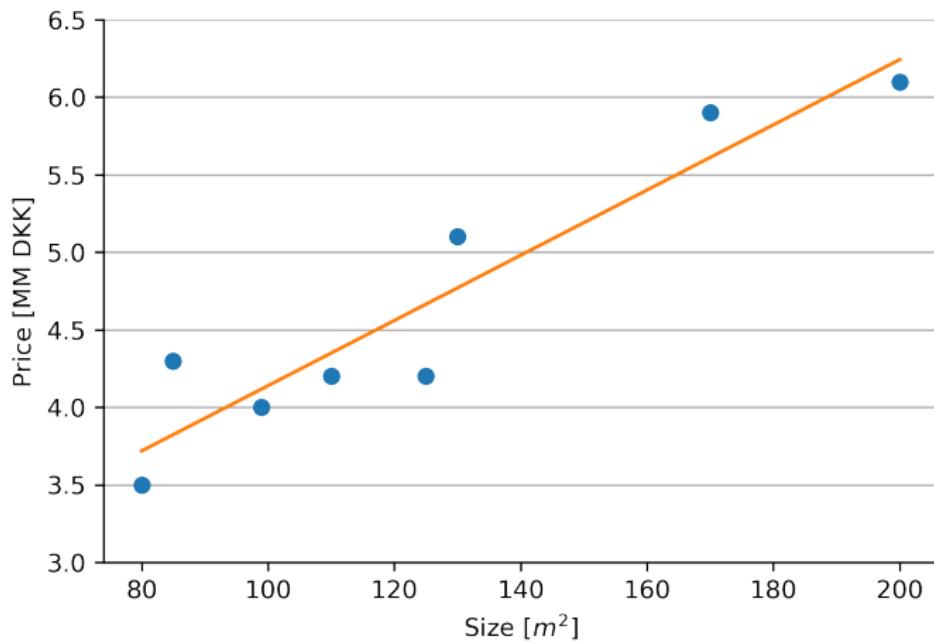
The dot product is maximal when the two vectors are parallel

$$\bar{v} = \frac{\nabla f(\bar{x})}{\|\nabla f(\bar{x})\|}$$

I.e. the gradient points in the direction of steepest ascent.

Linear regression (with gradient descent)

Remember linear regression



Gradient descent in linear regression

Linear regression

- Regression line: $f(x) = ax + b$
- Cost: Squared distance between data and regression line

$$E = \sum_{n=1}^N (y_n - f(x_n))^2$$

What is the gradient?

$$\nabla E(a, b) = \begin{bmatrix} \frac{\partial E(a, b)}{\partial a} \\ \frac{\partial E(a, b)}{\partial b} \end{bmatrix}$$

Gradient descent in linear regression

Linear regression

- Regression line: $f(x) = ax + b$
- Cost: Squared distance between data and regression line

$$E = \sum_{n=1}^N (y_n - f(x_n))^2$$

What is the gradient?

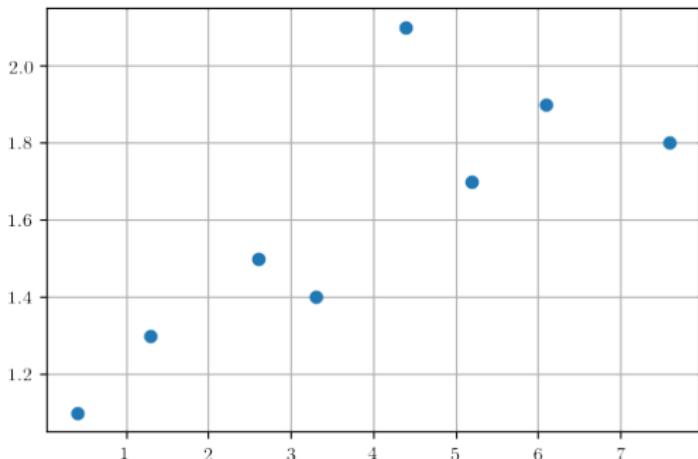
$$\nabla E(a, b) = \begin{bmatrix} \frac{\partial E(a, b)}{\partial a} \\ \frac{\partial E(a, b)}{\partial b} \end{bmatrix}$$

Solution

$$\frac{\partial E}{\partial a} = \sum_{n=1}^N -2(y_n - ax_n - b)x_n$$

$$\frac{\partial E}{\partial b} = \sum_{n=1}^N -2(y_n - ax_n - b)$$

Linear regression data

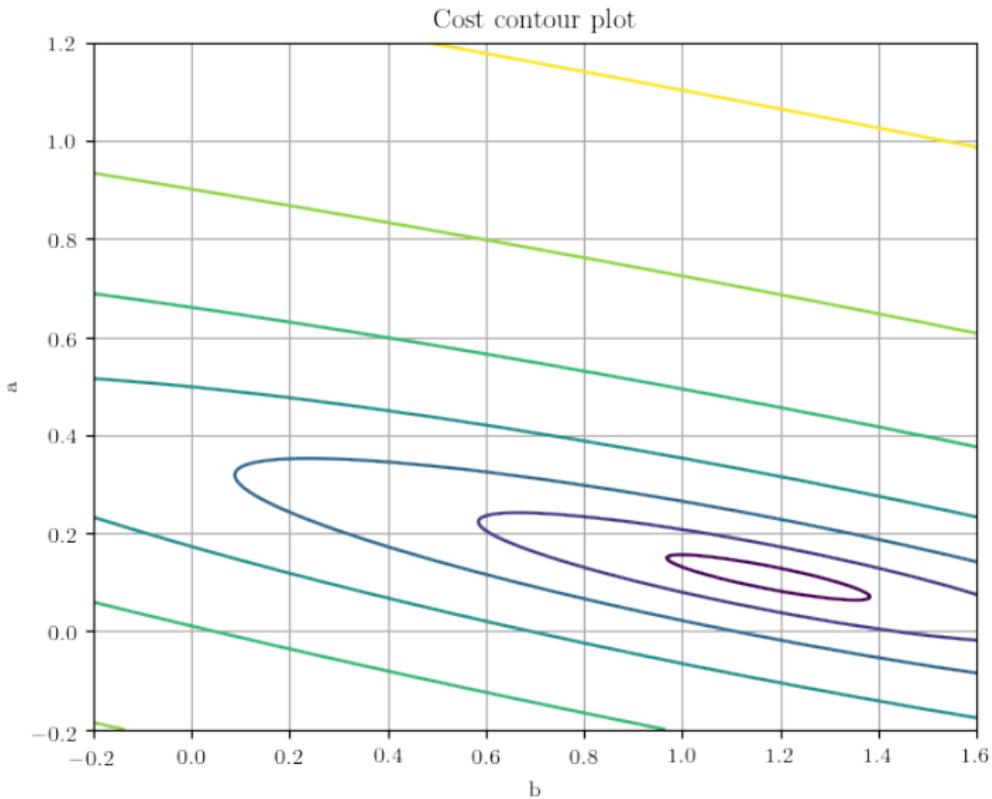


- Regression line:
 $f(x) = ax + b$
- Cost: Squared distance between data and regression line

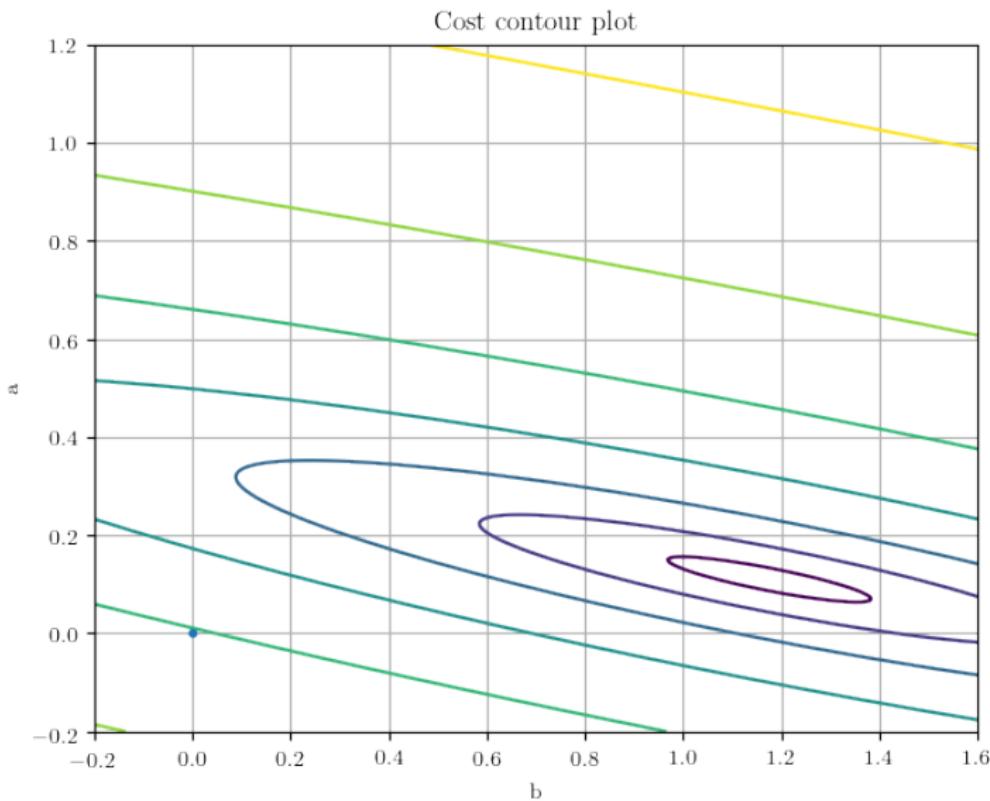
$$E = \sum_{n=1}^N (y_n - f(x_n))^2$$

The cost, $E(a, b)$, is a function of two variables. What does it look like?

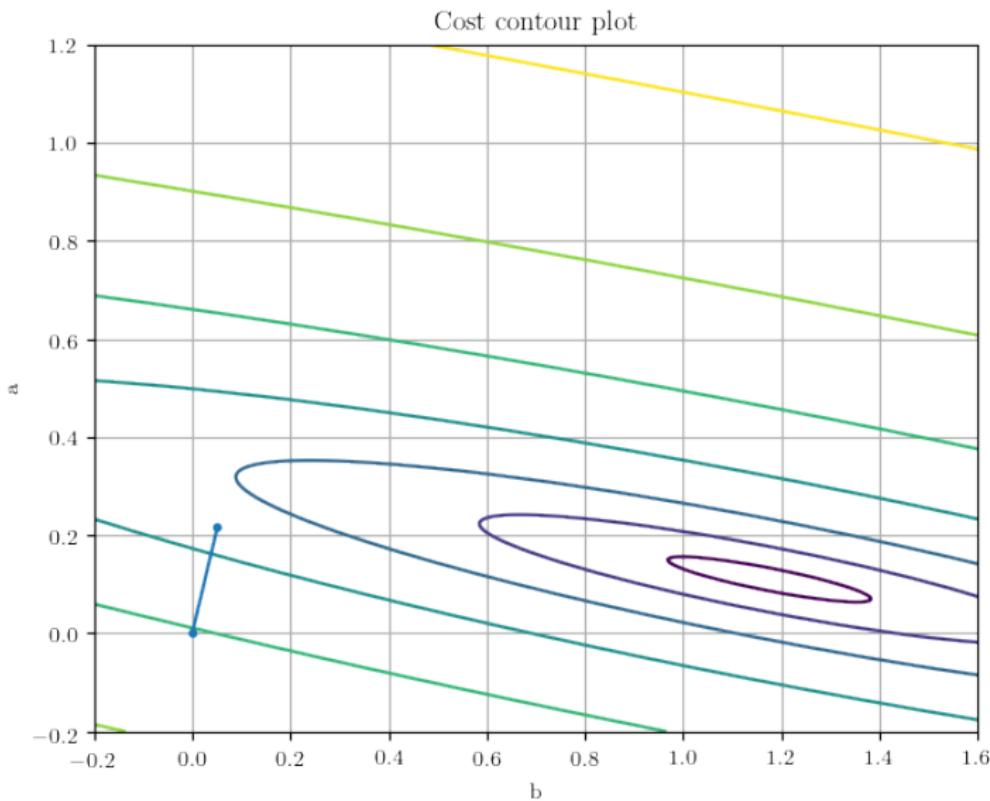
Gradient steps



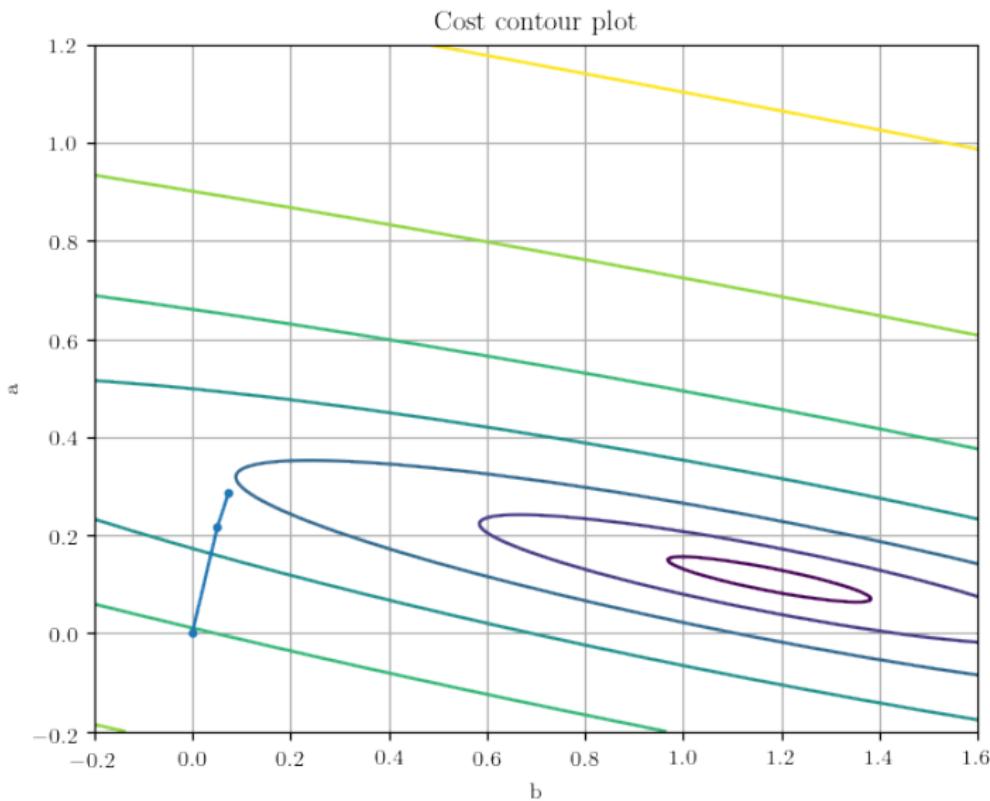
Gradient steps



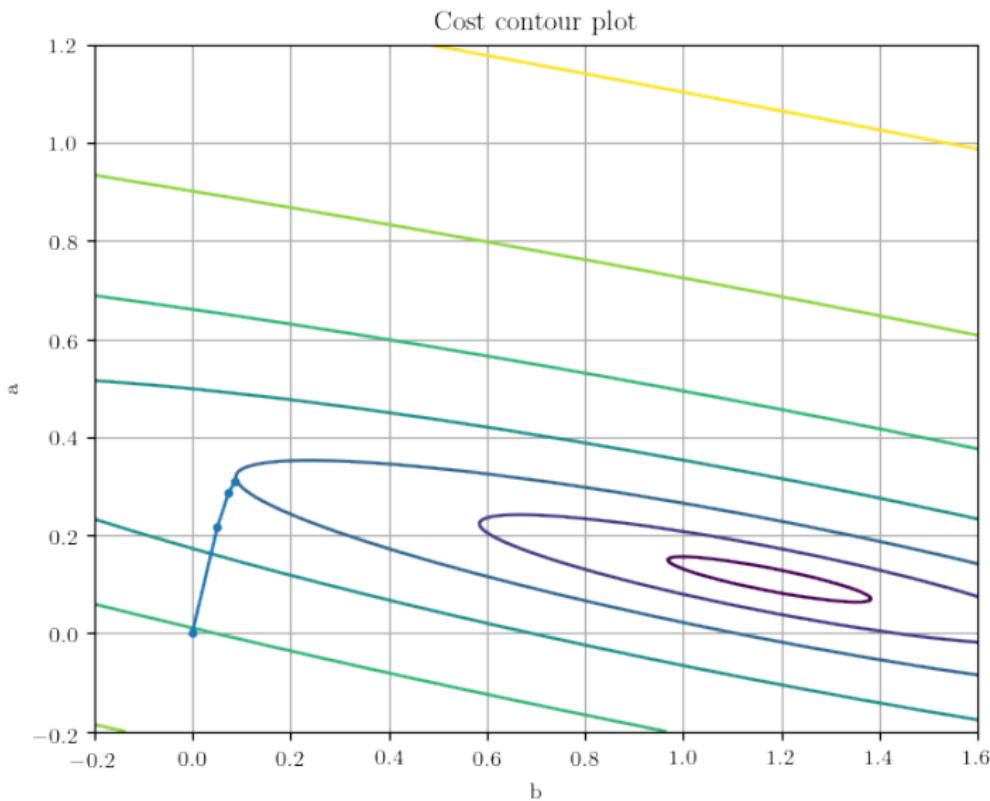
Gradient steps



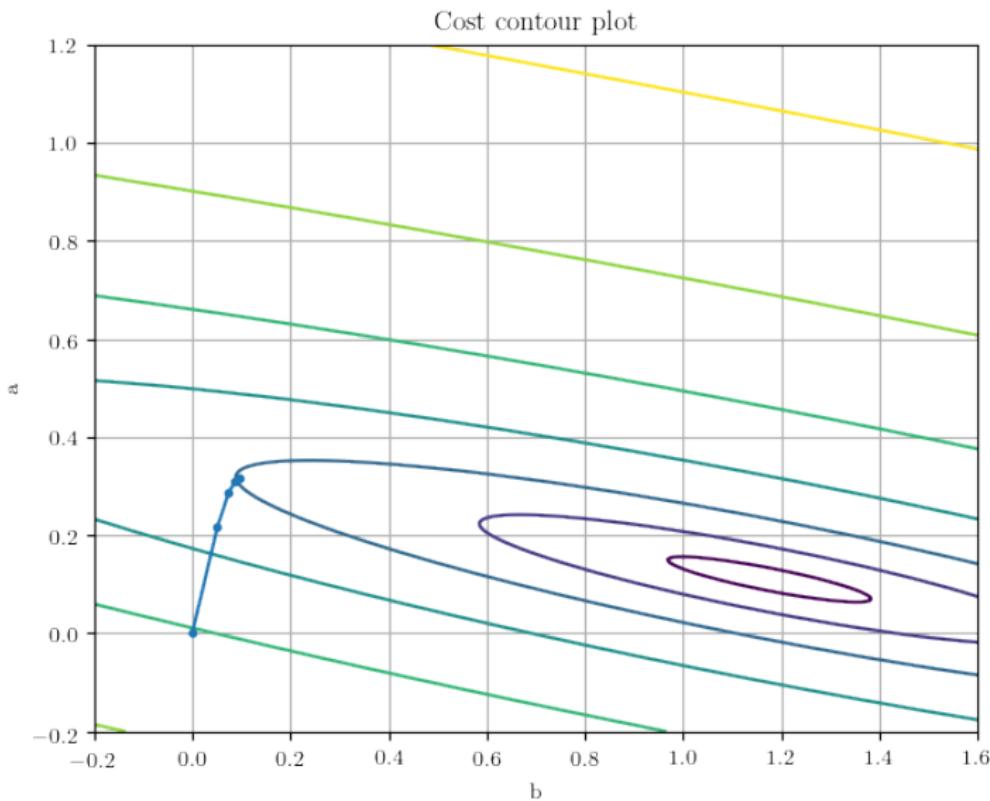
Gradient steps



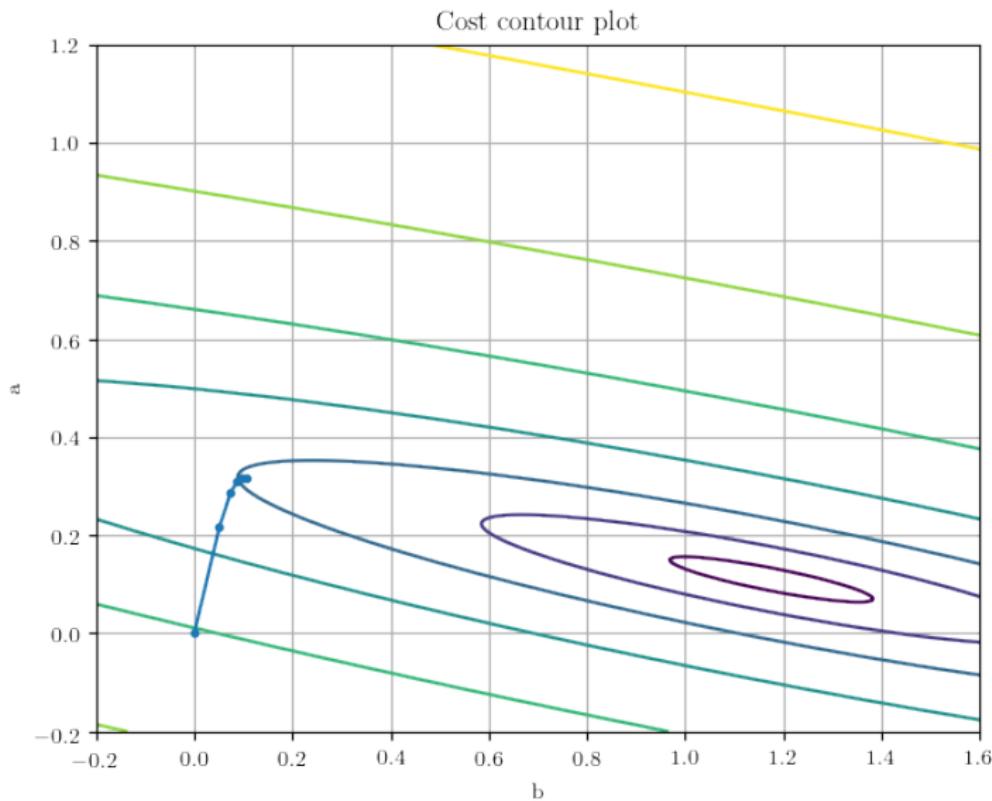
Gradient steps



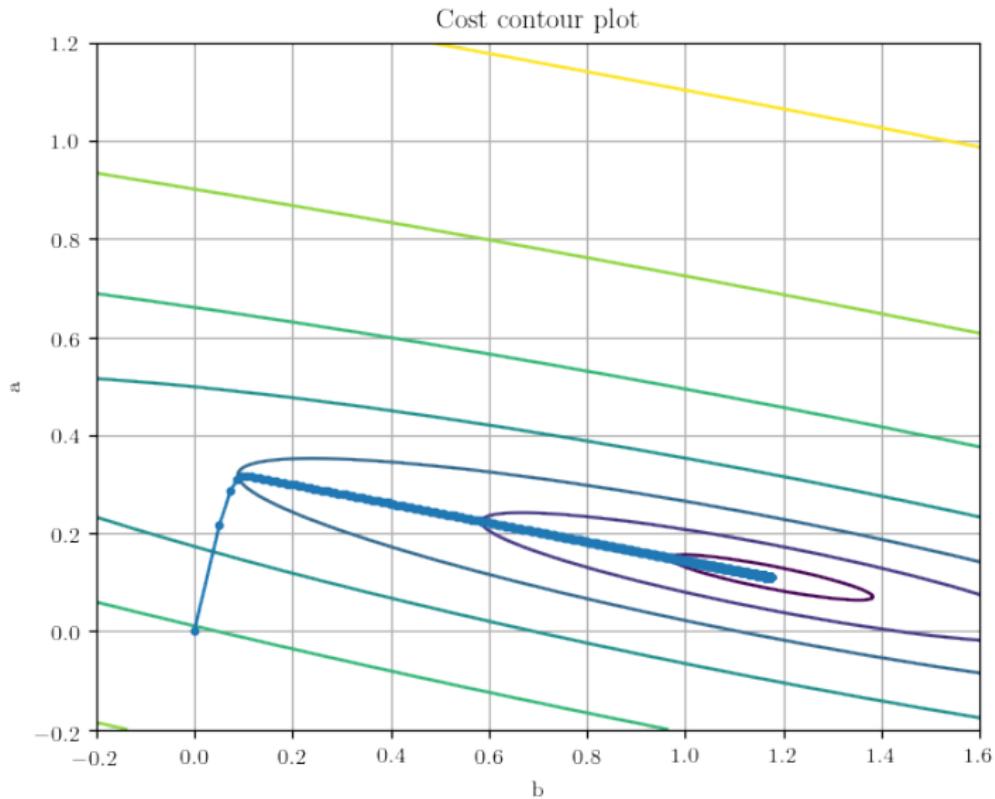
Gradient steps



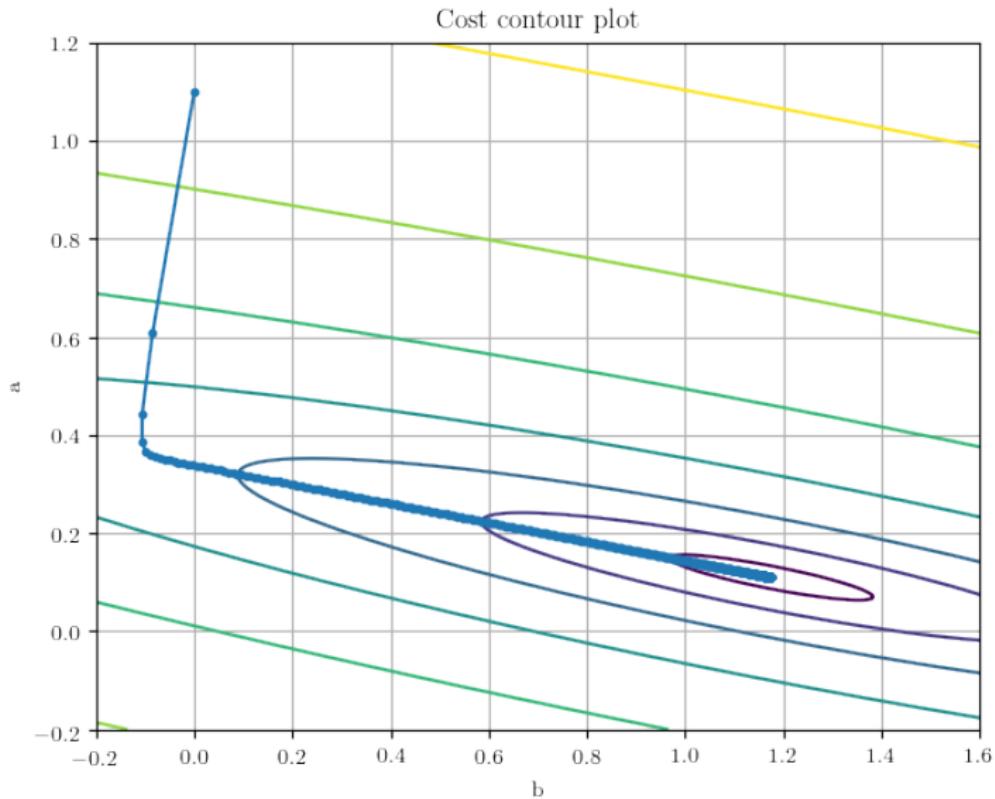
Gradient steps



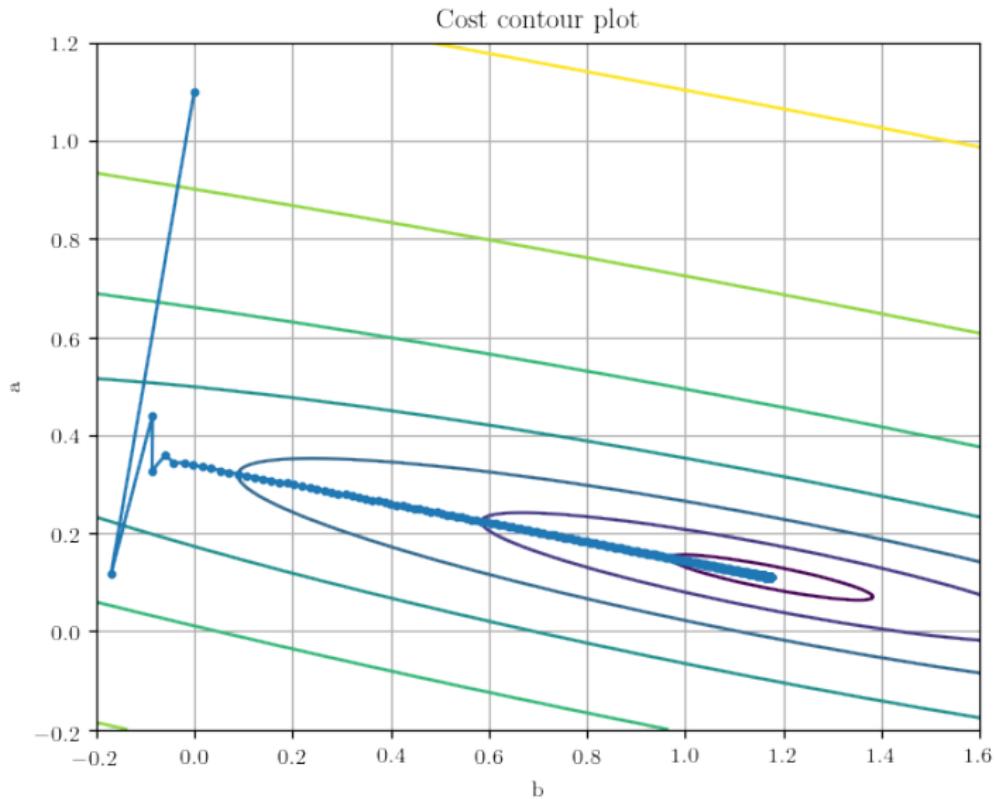
Gradient descent



Gradient descent



Gradient descent



Learning rate

- Small learning rate can lead to slow convergence
- Large learning rate may lead to divergence

Comparison with setting derivative equal to zero

Derivative equal to zero and solve

- No parameters to tune
- Closed form solution
- Slow for many features
Need to solve N equations in N unknowns

Gradient descent

- Need to select step size
- Needs many iterations
- Fast for many features
Need only compute the gradient

Feature scaling

- Is gradient descent sensitive to the scale of features? YES
- Features on different scale = parameters on different scale

Feature scaling

Min-max normalization

Rescale the range to $[0, 1]$

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Standardization

Rescale to have zero mean and unit variance

$$x' = \frac{x - \bar{x}}{\sigma_x}$$

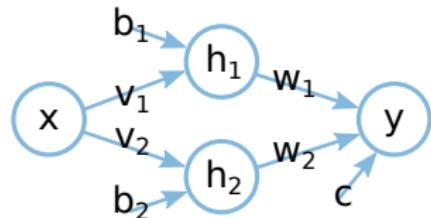
(\bar{x} : mean, σ_x : standard deviation)

Neural network (with gradient descent)

Neural network

Cost function

$$E = \sum_{n=1}^N (y_n - \hat{y}_n)^2$$



Network structure

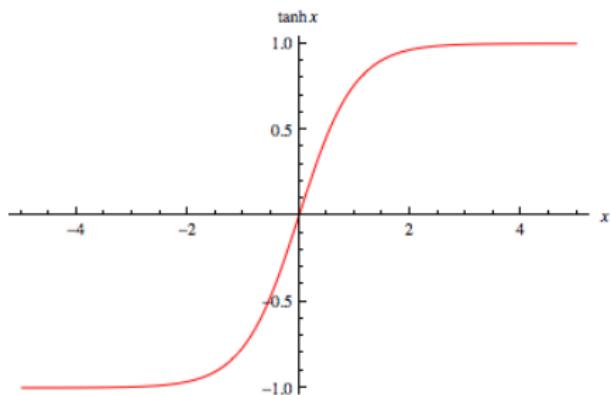
$$\hat{y}_n = w_1 h_1(x_n) + w_2 h_2(x_n) + c$$

$$h_1(x_n) = \tanh(v_1 x_n + b_1)$$

$$h_2(x_n) = \tanh(v_2 x_n + b_2)$$

Model parameters

$$c, w_1, w_2, v_1, v_2, b_1, b_2$$



Exercise: Gradient of neural network

Compute the partial derivatives

$$\frac{\partial E}{\partial c}, \quad \frac{\partial E}{\partial w_1}, \quad \frac{\partial E}{\partial b_1}, \quad \frac{\partial E}{\partial v_1}$$

Cost function

$$E = \sum_{n=1}^N (y_n - \hat{y}_n)^2$$

Hints

1. Use the chain rule
2. $\frac{\partial \tanh(x)}{\partial x} = 1 - \tanh^2(x)$
3. Don't expand terms needlessly.
Express in terms of e.g. \hat{y}_n and $h_1(x_n)$ where possible.

Neural network model

$$\hat{y}_n = w_1 h_1(x_n) + w_2 h_2(x_n) + c$$

$$h_1(x_n) = \tanh(v_1 x_n + b_1)$$

$$h_2(x_n) = \tanh(v_2 x_n + b_2)$$

Solution: Gradient of neural network

$$\frac{\partial E}{\partial c} = -2 \sum_{n=1}^N \left(y_n - \hat{y}_n \right)$$

Solution: Gradient of neural network

$$\frac{\partial E}{\partial c} = -2 \sum_{n=1}^N \left(y_n - \hat{y}_n \right)$$

$$\frac{\partial E}{\partial w_1} = -2 \sum_{n=1}^N \left((y_n - \hat{y}_n) h_1(x_n) \right)$$

Solution: Gradient of neural network

$$\frac{\partial E}{\partial c} = -2 \sum_{n=1}^N \left(y_n - \hat{y}_n \right)$$

$$\frac{\partial E}{\partial w_1} = -2 \sum_{n=1}^N \left((y_n - \hat{y}_n) h_1(x_n) \right)$$

$$\frac{\partial E}{\partial b_1} = -2 \sum_{n=1}^N \left((y_n - \hat{y}_n) w_1 (1 - h_1^2(x_n)) \right)$$

Solution: Gradient of neural network

$$\frac{\partial E}{\partial c} = -2 \sum_{n=1}^N \left(y_n - \hat{y}_n \right)$$

$$\frac{\partial E}{\partial w_1} = -2 \sum_{n=1}^N \left((y_n - \hat{y}_n) h_1(x_n) \right)$$

$$\frac{\partial E}{\partial b_1} = -2 \sum_{n=1}^N \left((y_n - \hat{y}_n) w_1 (1 - h_1^2(x_n)) \right)$$

$$\frac{\partial E}{\partial v_1} = -2 \sum_{n=1}^N \left((y_n - \hat{y}_n) w_1 (1 - h_1^2(x_n)) \right) x_n$$

Analysis of gradient

Partial derivatives

- Gradient scales with the error $y_n - \hat{y}_n$
- If $h_1(x_n)$ saturates at -1 or +1, the term $1 - h_1^2(n)$ is zero

$$\frac{\partial E}{\partial c} = -2 \sum_{n=1}^N \left(y_n - \hat{y}_n \right)$$

$$\frac{\partial E}{\partial w_1} = -2 \sum_{n=1}^N \left((y_n - \hat{y}_n) h_1(x_n) \right)$$

$$\frac{\partial E}{\partial b_1} = -2 \sum_{n=1}^N \left((y_n - \hat{y}_n) w_1 (1 - h_1^2(n)) \right)$$

$$\frac{\partial E}{\partial v_1} = -2 \sum_{n=1}^N \left((y_n - \hat{y}_n) w_1 (1 - h_1^2(n)) \right) x_n$$

Tasks

Tasks

Tasks today

1. Work through the notebook

`08-GradientDescentLinearRegression.ipynb`

2. Work through the notebook `08-GradientDescentNeuralNet.ipynb`

3. Today's feedback group

- Nicholas Borch
- Alfred Fonnesbech Aqraou
- Josefine Høgsted Voglhofer
- Rasmus Bernth Linnemann

Lab report hand in

- Lab 3: Image segmentation (Deadline: Thursday 26 October 20:00)

Introduction to intelligent systems

Automatic differentiation

Mikkel N. Schmidt

Technical University of Denmark,
DTU Compute, Department of Applied Mathematics and Computer Science.

Overview

① Automatic differentiation

② Dual numbers

③ PyTorch

④ Tasks

Feedback group

- Mathias Kraemer Eberhardt Sørensen
- Oskar Gotthardt Bak
- Christian Ludvig Meinert Sørensen
- Alexander Baumkirchner

Learning objectives

- I Automatic differentiation: Forward and reverse accumulation.
 - II Computation graphs.
 - II Automatic differentiation in Pytorch.
 - II Implementation of neural networks in Pytorch.
-
- I Understand the concepts and definitions, and know their application. Reason about the concepts in the context of an example. Use correct technical terminology.
 - II As above plus: Read, manipulate, and work with technical definitions and expressions (mathematical and Python code). Carry out practical computations. Interpret and evaluate results.

Automatic differentiation

Gradient descent

Initialize x_0

Repeat, $t = 0, 1, 2, \dots$

$$\underbrace{x_{t+1}}_{\text{new parameter value}} = \underbrace{x_t}_{\text{old parameter value}} - \underbrace{\alpha}_{\text{step size}} \cdot \underbrace{\nabla f(x_t)}_{\text{gradient}}$$

until convergence

Definition of gradient

$$\nabla f(x, y, \dots) = \begin{bmatrix} \frac{\partial f(x, y)}{\partial x} \\ \frac{\partial f(x, y)}{\partial y} \\ \vdots \end{bmatrix}$$

Symbolic, numerical, and automatic differentiation

Symbolic Automatic manipulation of mathematical expressions to get derivatives (e.g. Mathematica, Maple)

Numerical Approximation of derivatives by finite differences

Automatic Automatic computation of the derivative of a compound expression by applying the chain rule

The chain rule

Derivative of composition of functions, $z(x) = (f \circ g)(x) = f(g(x))$

$$z' = (f \circ g)' = (f' \circ g) \cdot g'$$

In Leibnitz's notation

$$\frac{dz}{dx} = \frac{df}{dy} \cdot \frac{dy}{dx}$$

where $z = f(y)$ and $y = g(x)$

Chain rule for functions of multiple variables

Function of two variables $z(t) = f(x(t), y(t))$

$$\frac{dz}{dt} = \frac{\partial f}{\partial x} \cdot \frac{dx}{dt} + \frac{\partial f}{\partial y} \cdot \frac{dy}{dt}$$

Exercise: Chain rule

Compute the derivative $\frac{dz}{dt}$ of the following function

$$z(t) = f(x, y) = xy + x^2$$

where

$$x(t) = \sin(t)$$

$$y(t) = t^2$$

Exercise: Chain rule

Compute the derivative $\frac{dz}{dt}$ of the following function

$$z(t) = f(x, y) = xy + x^2$$

where

$$x(t) = \sin(t)$$

$$y(t) = t^2$$

Solution

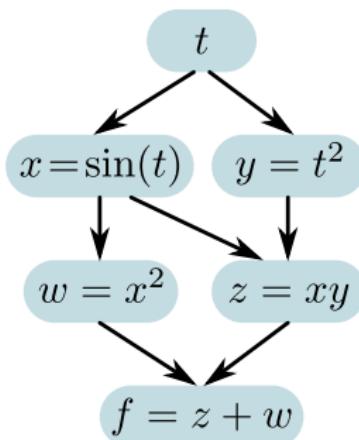
$$\begin{aligned}\frac{dz}{dt} &= \frac{\partial f}{\partial x} \cdot \frac{dx}{dt} + \frac{\partial f}{\partial y} \cdot \frac{dy}{dt} \\ &= (y + 2x) \cdot \cos(t) + x \cdot (2t)\end{aligned}$$

Computation graph

$$f(t) = \sin(t)t^2 + \sin^2(t)$$

Computation graph

$$f(t) = \sin(t)t^2 + \sin^2(t)$$



Exercise: Computation graph

Draw the computation graph for the function

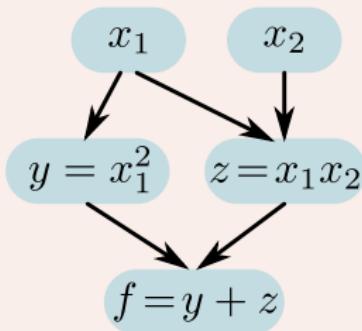
$$f(x_1, x_2) = x_1^2 + x_1 \cdot x_2$$

Exercise: Computation graph

Draw the computation graph for the function

$$f(x_1, x_2) = x_1^2 + x_1 \cdot x_2$$

Solution



Forward accumulation

Function and derivatives

$$f(x_1, x_2) = x_1^2 + x_1 \cdot x_2$$

$$\frac{\partial f}{\partial x_1} = 2x_1 + x_2, \quad \frac{\partial f}{\partial x_2} = x_1, \quad \nabla f(3, 4) = \begin{bmatrix} 10 \\ 3 \end{bmatrix}$$

Evaluate $f(3, 4)$

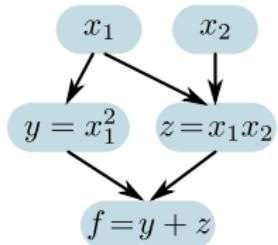
$$x_1 = 3$$

$$x_2 = 4$$

$$y = x_1^2$$

$$z = x_1 x_2$$

$$f = y + z$$

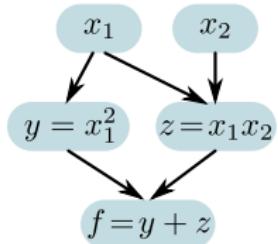


Forward accumulation

Function and derivatives

$$f(x_1, x_2) = x_1^2 + x_1 \cdot x_2$$

$$\frac{\partial f}{\partial x_1} = 2x_1 + x_2, \quad \frac{\partial f}{\partial x_2} = x_1, \quad \nabla f(3, 4) = \begin{bmatrix} 10 \\ 3 \end{bmatrix}$$



Evaluate $f(3, 4)$

$$x_1 = 3$$

$$x_2 = 4$$

$$y = x_1^2$$

$$z = x_1 x_2$$

$$f = y + z$$

Evaluate $\nabla_{x_1} f(3, 4)$

$$\dot{x}_1 = \frac{\partial x_1}{\partial x_1} = 1$$

$$\dot{x}_2 = \frac{\partial x_2}{\partial x_1} = 0$$

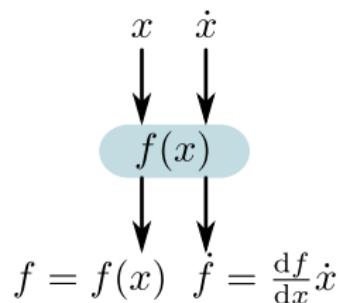
$$\dot{y} = \frac{\partial y}{\partial x_1} \dot{x}_1 = 2x_1 \cdot \dot{x}_1 = 2 \cdot 3 \cdot 1 = 6$$

$$\dot{z} = \frac{\partial z}{\partial x_1} \dot{x}_1 + \frac{\partial z}{\partial x_2} \dot{x}_2 = x_2 \cdot \dot{x}_1 + x_1 \cdot \dot{x}_2 = 4 \cdot 1 + 3 \cdot 0 = 4$$

$$\dot{f} = \frac{\partial f}{\partial y} \dot{y} + \frac{\partial f}{\partial z} \dot{z} = 1 \cdot \dot{y} + 1 \cdot \dot{z} = 1 \cdot 6 + 1 \cdot 4 = \underline{10}$$

Forward accumulation

Function of one variable



Forward accumulation

Function of multiple variables

$$\begin{array}{ccccc} x & \dot{x} & y & \dot{y} \\ \downarrow & \downarrow & \downarrow & \downarrow \\ f(x, y) \\ \downarrow & & \downarrow \\ f = f(x, y) & \dot{f} = \frac{\partial f}{\partial x} \dot{x} + \frac{\partial f}{\partial y} \dot{y} \end{array}$$

Forward accumulation

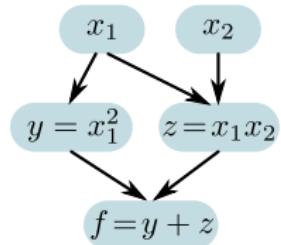
- Direct application of chain rule (going from input to output)
- Computation involves one forward pass through the graph per derivative
- Computationally expensive with many inputs

Reverse accumulation

Function and derivatives

$$f(x_1, x_2) = x_1^2 + x_1 \cdot x_2$$

$$\frac{\partial f}{\partial x_1} = 2x_1 + x_2, \quad \frac{\partial f}{\partial x_2} = x_1, \quad \nabla f(3, 4) = \begin{bmatrix} 10 \\ 3 \end{bmatrix}$$



Evaluate $f(3, 4)$

$$x_1 = 3$$

$$x_2 = 4$$

$$y = x_1^2$$

$$z = x_1 \cdot x_2$$

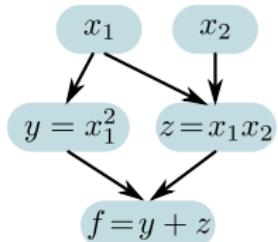
$$f = y + z$$

Reverse accumulation

Function and derivatives

$$f(x_1, x_2) = x_1^2 + x_1 \cdot x_2$$

$$\frac{\partial f}{\partial x_1} = 2x_1 + x_2, \quad \frac{\partial f}{\partial x_2} = x_1, \quad \nabla f(3, 4) = \begin{bmatrix} 10 \\ 3 \end{bmatrix}$$



Evaluate $f(3, 4)$

$$x_1 = 3$$

$$x_2 = 4$$

$$y = x_1^2$$

$$z = x_1 x_2$$

$$f = y + z$$

Evaluate $\nabla f(3, 4)$

$$\bar{f} = \frac{\partial f}{\partial f} = 1$$

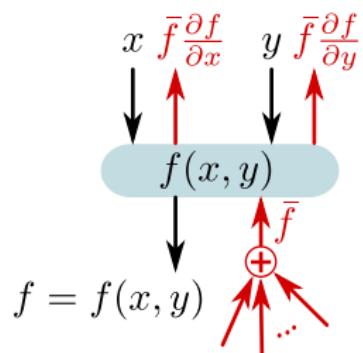
$$\bar{y} = \bar{f} \frac{\partial f}{\partial y} = \bar{f} \cdot 1 = 1 \cdot 1 = 1$$

$$\bar{z} = \bar{f} \frac{\partial f}{\partial z} = \bar{f} \cdot 1 = 1 \cdot 1 = 1$$

$$\bar{x}_1 = \bar{y} \frac{\partial y}{\partial x_1} + \bar{z} \frac{\partial z}{\partial x_1} = \bar{y} \cdot 2 \cdot x_1 + \bar{z} \cdot x_2 = 1 \cdot 2 \cdot 3 + 1 \cdot 4 = 10$$

$$\bar{x}_2 = \bar{z} \frac{\partial z}{\partial x_2} = \bar{x}_2 + \bar{z} \cdot x_1 = 0 + 1 \cdot 3 = 3$$

Reverse accumulation



Reverse accumulation

- Direct application of chain rule (going from output to input)
- Computation involves one backward pass through the graph to compute all derivatives
- Requires a bit more “book-keeping” to keep track of dependencies and trace the graph backwards

Dual numbers

Complex and dual numbers

\mathbb{C} : Complex numbers

$$a + ib$$

$$i^2 = -1$$

Addition

$$(a + ib) + (c + id) = a + c + i(b + d)$$

Multiplication

$$\begin{aligned}(a + ib)(c + id) &= ac + iad + ibc + i^2 bd \\ &= (ac - bd) + i(ad + bc)\end{aligned}$$

\mathbb{D} : Dual numbers

$$a + \epsilon b$$

$$\epsilon^2 = 0$$

Addition

$$(a + \epsilon b) + (c + \epsilon d) = a + c + \epsilon(b + d)$$

Multiplication

$$\begin{aligned}(a + \epsilon b)(c + \epsilon d) &= ac + \epsilon ad + \epsilon bc + \epsilon^2 bd \\ &= ac + \epsilon(ad + bc)\end{aligned}$$

Example: $f(x) = x^2$

Example

$$f(x) = x^2$$

Example: $f(x) = x^2$

Example

$$f(x) = x^2$$

Evaluating $f(x)$ on $x = a + \epsilon$ we get

$$f(a + \epsilon) = (a + \epsilon)^2 = a^2 + 2a\epsilon + \epsilon^2 = \underbrace{a^2}_{f(a)} + \epsilon \underbrace{2a}_{f'(a)}$$

The dual part happens to be $f'(a) = 2a$. Coincidence?

Exercise: $f(x) = Ax^2 + Bx + C$

Consider the function

$$f(x) = Ax^2 + Bx + C$$

Evaluate the function on $x = a + \epsilon$

Exercise: $f(x) = Ax^2 + Bx + C$

Consider the function

$$f(x) = Ax^2 + Bx + C$$

Evaluate the function on $x = a + \epsilon$

Solution

$$\begin{aligned} f(a + \epsilon) &= A(a + \epsilon)^2 + B(a + \epsilon) + C \\ &= A(a^2 + 2a\epsilon + \epsilon^2) + B(a + \epsilon) + C \\ &= \underbrace{(Aa^2 + Ba + C)}_{f(a)} + \epsilon \underbrace{(2Aa + B)}_{f'(a)} \end{aligned}$$

Taylor series

Taylor series around a

$$f(x) = f(a) + \frac{f'(a)}{1!}(x - a) + \frac{f''(a)}{2!}(x - a)^2 + \frac{f'''(a)}{3!}(x - a)^3 + \dots$$

Taylor series

Taylor series around a

$$f(x) = f(a) + \frac{f'(a)}{1!}(x - a) + \frac{f''(a)}{2!}(x - a)^2 + \frac{f'''(a)}{3!}(x - a)^3 + \dots$$

Inserting $x = a + \epsilon$

$$\begin{aligned}f(a + \epsilon) &= f(a) + \frac{f'(a)}{1!}\epsilon + \frac{f''(a)}{2!}\epsilon^2 + \frac{f'''(a)}{3!}\epsilon^3 + \dots \\&= f(a) + \epsilon f'(a)\end{aligned}$$

PyTorch

PyTorch

Demonstration

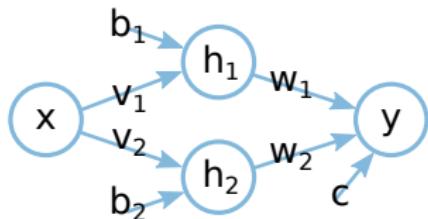
Demo: PyTorch

```
>>> import torch
>>> x1 = torch.tensor(3., requires_grad=True)
>>> x2 = torch.tensor(4., requires_grad=True)
>>> f = x1**2+x1*x2
>>> f.backward()
>>> x1.grad
tensor(10.)
>>> x2.grad
tensor(3.)
```

Neural network notebook

Cost function

$$E = \sum_{n=1}^N (y(n) - \hat{y}(n))^2$$



Network structure

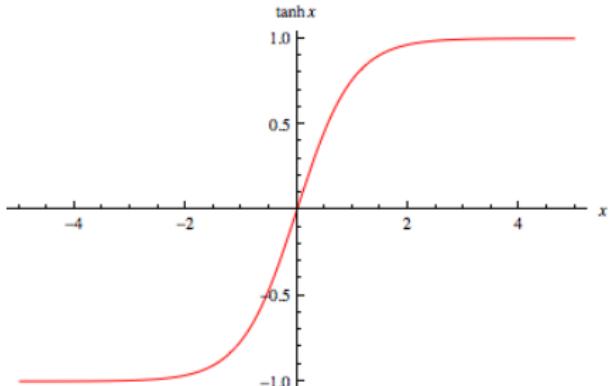
$$\hat{y}(n) = w_1 h_1(n) + w_2 h_2(n) + c$$

$$h_1(n) = \tanh(v_1 x(n) + b_1)$$

$$h_2(n) = \tanh(v_2 x(n) + b_2)$$

Model parameters

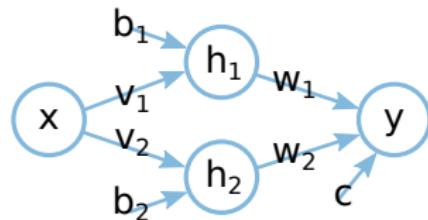
$$c, w_1, w_2, v_1, v_2, b_1, b_2$$



Neural network notebook

Cost function

$$E = \sum_{n=1}^N (y(n) - \hat{y}(n))^2$$



Network structure

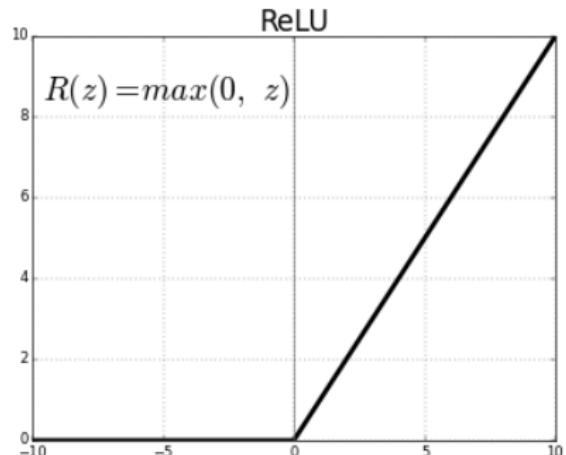
$$\hat{y}(n) = w_1 h_1(n) + w_2 h_2(n) + c$$

$$h_1(n) = \text{ReLU}(v_1 x(n) + b_1)$$

$$h_2(n) = \text{ReLU}(v_2 x(n) + b_2)$$

Model parameters

$$c, w_1, w_2, v_1, v_2, b_1, b_2$$



Tasks

Tasks

1. Work through introduction to PyTorch notebooks
See `09-PyTorchTutorial1.ipynb` and `09-PyTorchTutorial2.ipynb` on the fileshare
2. Work through introduction to PyTorch notebooks
See `09-TwoLayerNet-x.ipynb` on the fileshare
3. Experiment with the neural network challenge notebook
See `09-NeuralNetworkChallenge.ipynb` on the fileshare
4. Today's feedback group
 - Mathias Kræmer Eberhardt Sørensen
 - Oskar Gotthardt Bak
 - Christian Ludvig Meinert Sørensen
 - Alexander Baumkirchner

Lab report

- Lab 4: Neural networks (Deadline: Thursday 9 November 20:00)

Introduction to intelligent systems

Audio processing

Mikkel N. Schmidt

Technical University of Denmark,
DTU Compute, Department of Applied Mathematics and Computer Science.

Overview

① Machine learning systems

② Feature transformations

③ Audio

④ Tasks

Feedback group

- Selma Bundgaard Langvik
- Andreas Holm Matthiassen
- Jacob Danvad Nalholm
- Mikkel Nielsen Broch-Lips

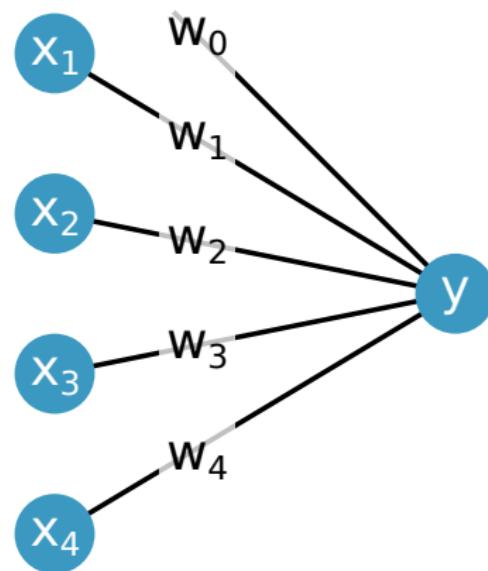
Learning objectives

- I** Frequency spectrum and spectrogram.
- II** Feature transformations and basis change.

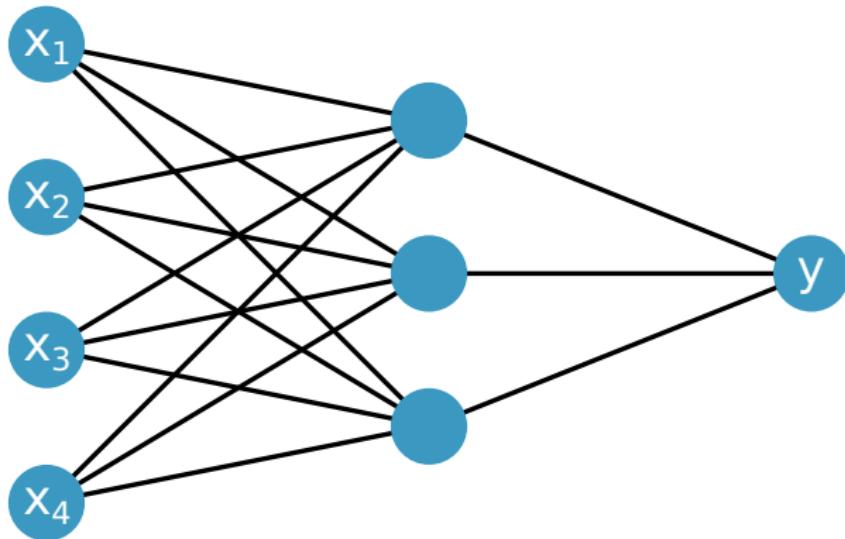
- I Understand the concepts and definitions, and know their application. Reason about the concepts in the context of an example. Use correct technical terminology.
- II As above plus: Read, manipulate, and work with technical definitions and expressions (mathematical and Python code). Carry out practical computations. Interpret and evaluate results.

Machine learning systems

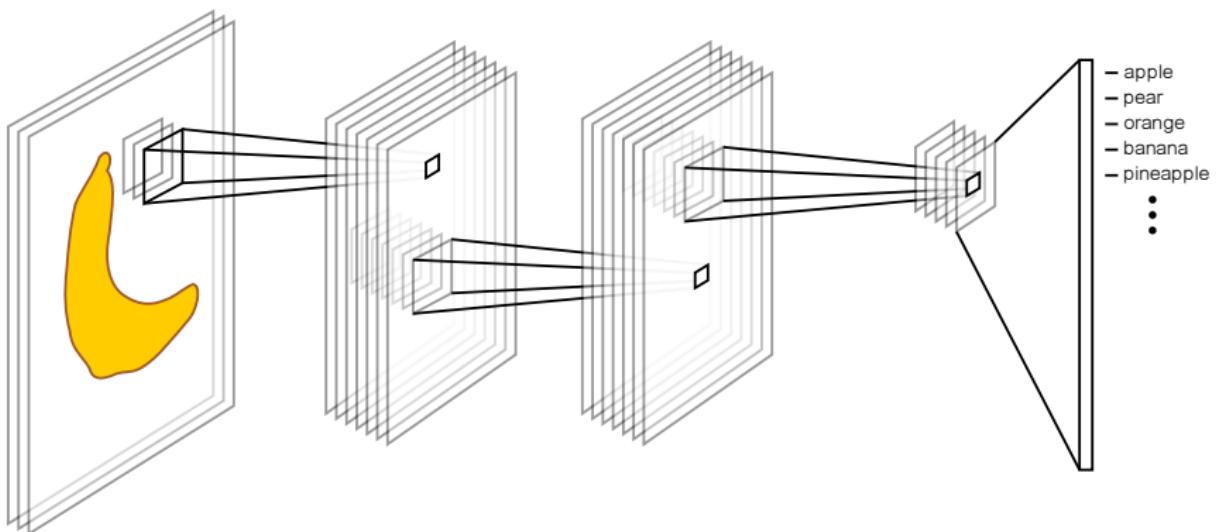
Linear models



Neural networks

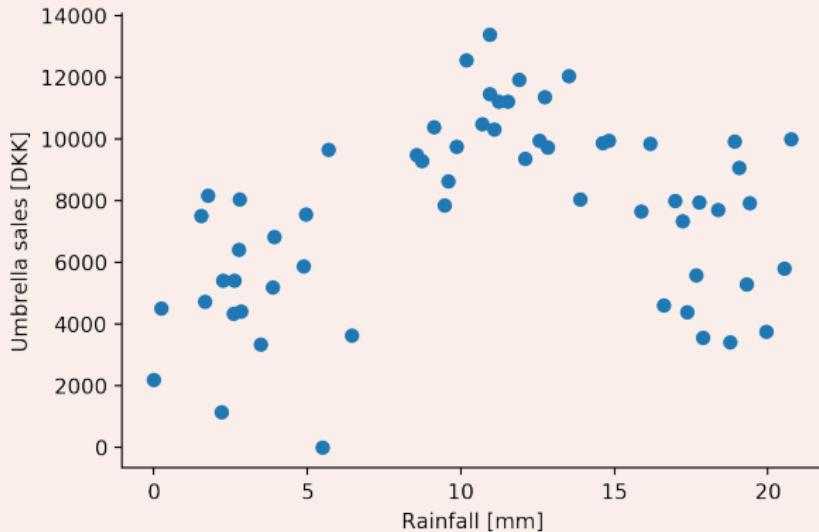


Convulsive neural network



Feature transformations

Clustering umbrella sales



- We want to examine if there are any clusters in the umbrella sales data
- We decide to use the k-means algorithm with the Euclidean distance
- What will go wrong, and how can we fix it?

Feature transformations

Mapping a set of data to a new set of values

Reasons to do feature transformations:

- To make representation more suitable for some particular algorithm
- To focus on relevant aspects of the data
- To make the data easier to process
- To remove unwanted noise
- To reduce dimensionality

Feature scaling

- Some machine learning methods are sensitive to the range of variables
- Standardize the range of a variable

Min-max normalization Rescale the range to $[0, 1]$

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Standardization Rescale to have zero mean and unit variance

$$x' = \frac{x - \bar{x}}{\sigma_x}$$

(\bar{x} : mean, σ_x : standard deviation)

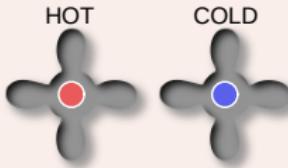
Change of basis

One way to transform a set of features is to change basis

- Represent data as a set of linear combinations of existing features
- Improve interpretation / more meaningful features

Hot and cold tap

- In the good old days, a shower just had a *hot* and a *cold* tap

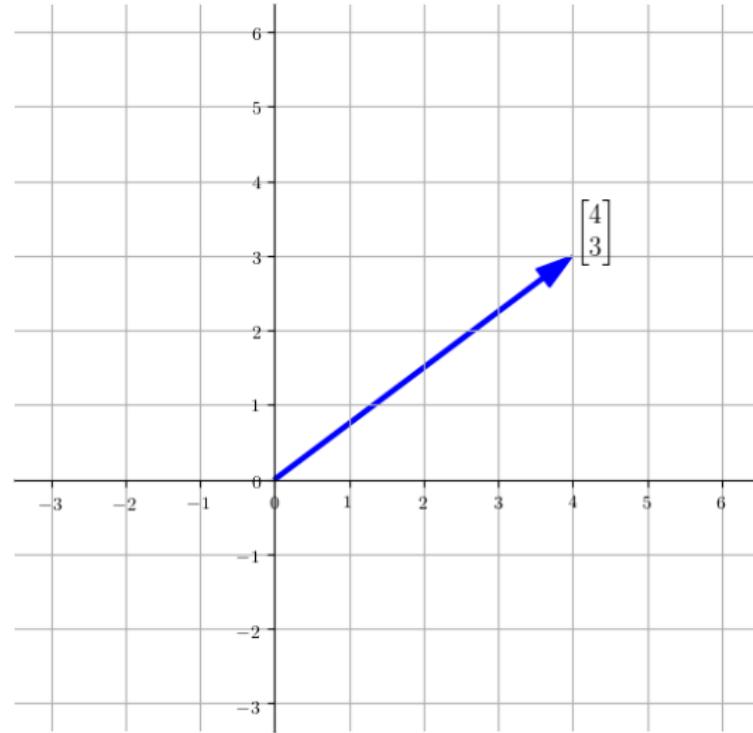


- To make the water more hot while keeping the same pressure, you would
 1. Turn up the hot tap a bit
 2. Turn down the cold tap a bit
 3. Adjust by turning the hot a little down again
 4. Hmm. Now turn up the cold a little ...
 5. Aargh... Too cold now...
- Ideally, we would like a *temperature* and a *pressure* tap
- How can you make a linear combination of *hot* and *cold* to achieve this?
(fill in the missing numbers)

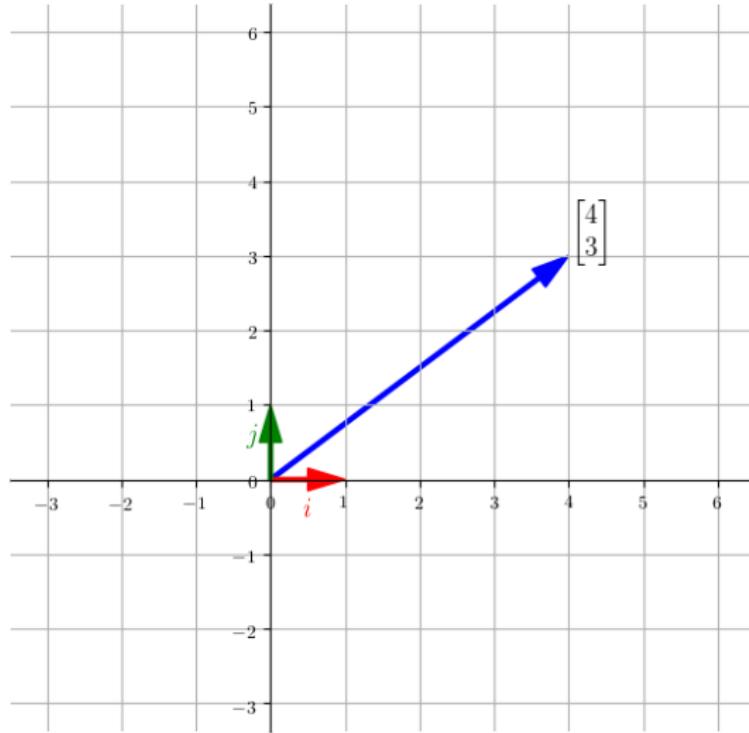
$$\text{temperature} = \underline{\quad} \cdot \text{cold} + \underline{\quad} \cdot \text{hot}$$

$$\text{pressure} = \underline{\quad} \cdot \text{cold} + \underline{\quad} \cdot \text{hot}$$

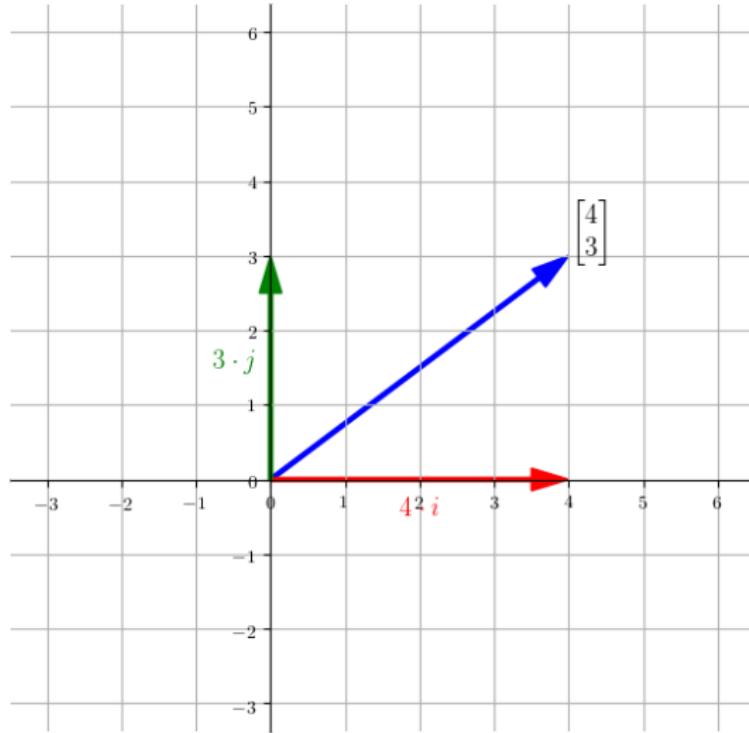
Basis of a vector space



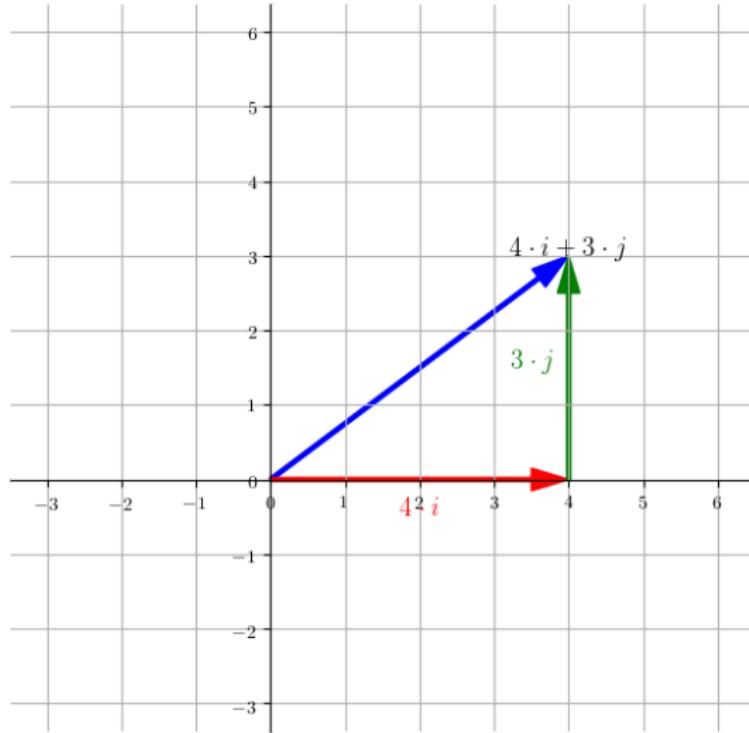
Basis of a vector space



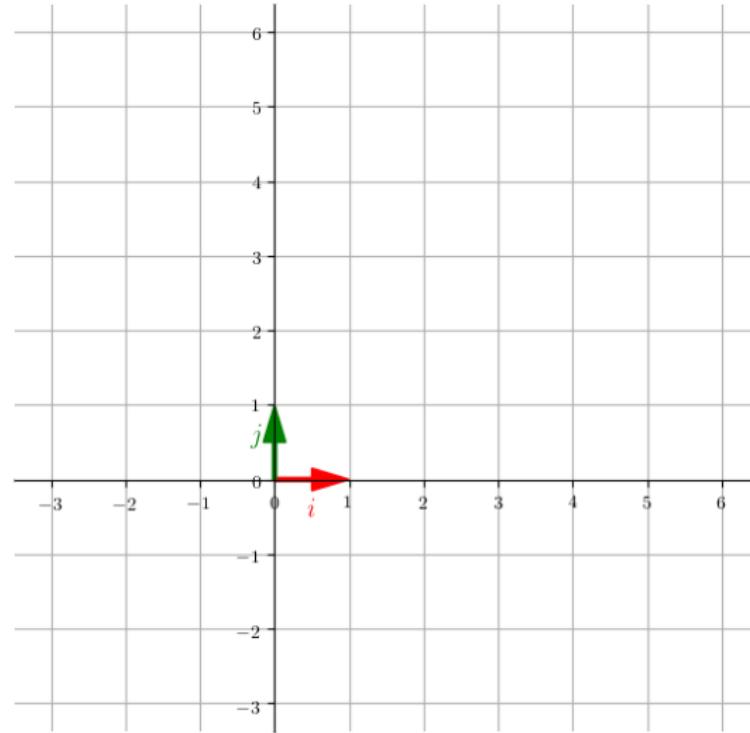
Basis of a vector space



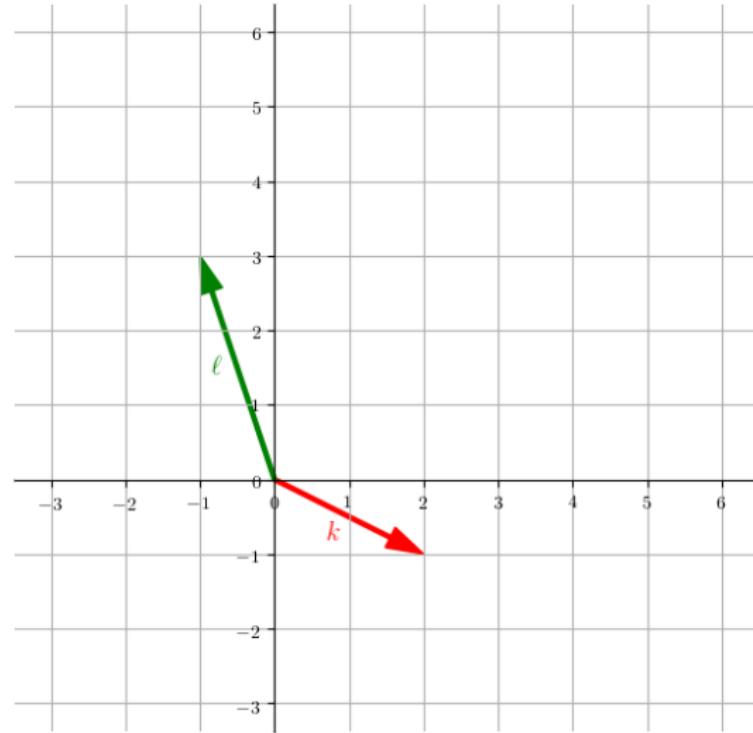
Basis of a vector space



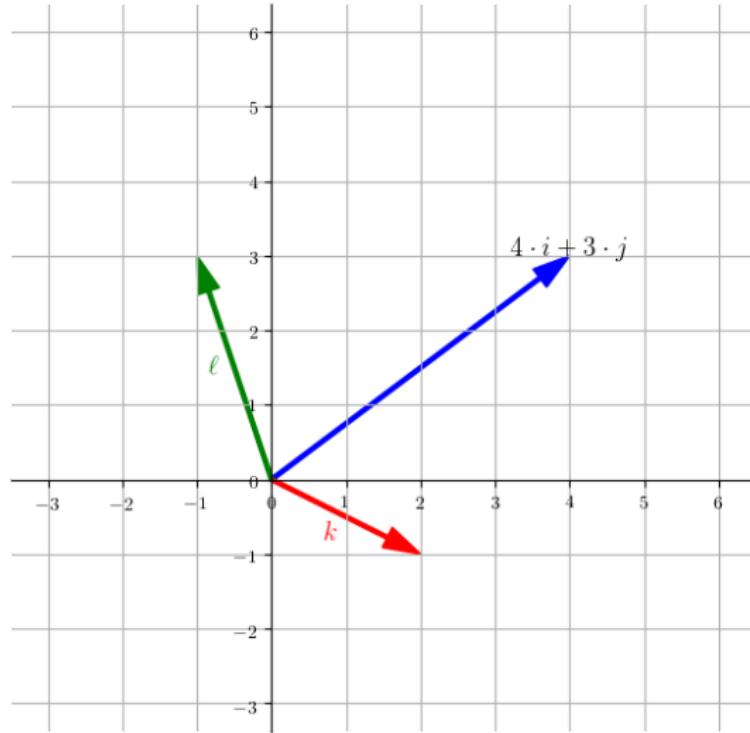
Basis of a vector space



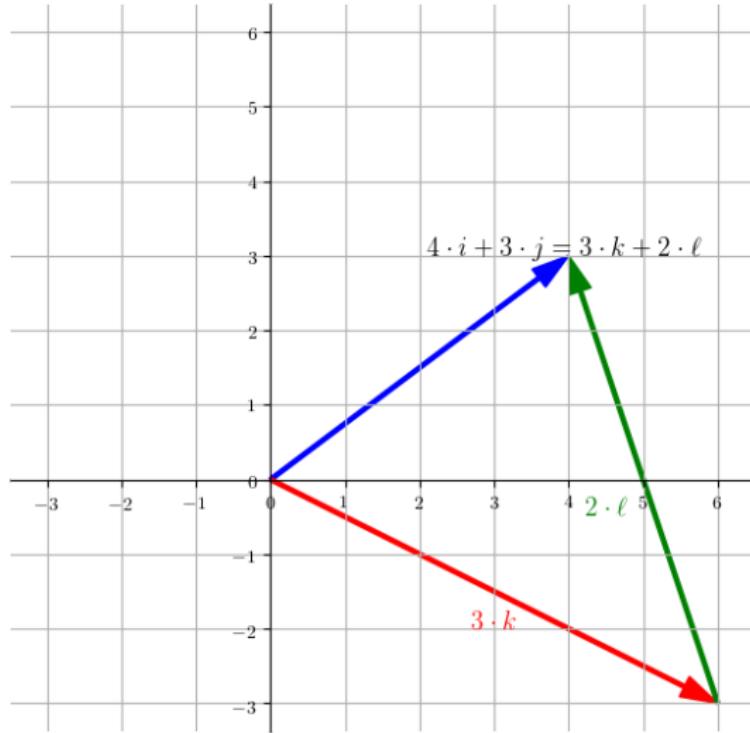
Basis of a vector space



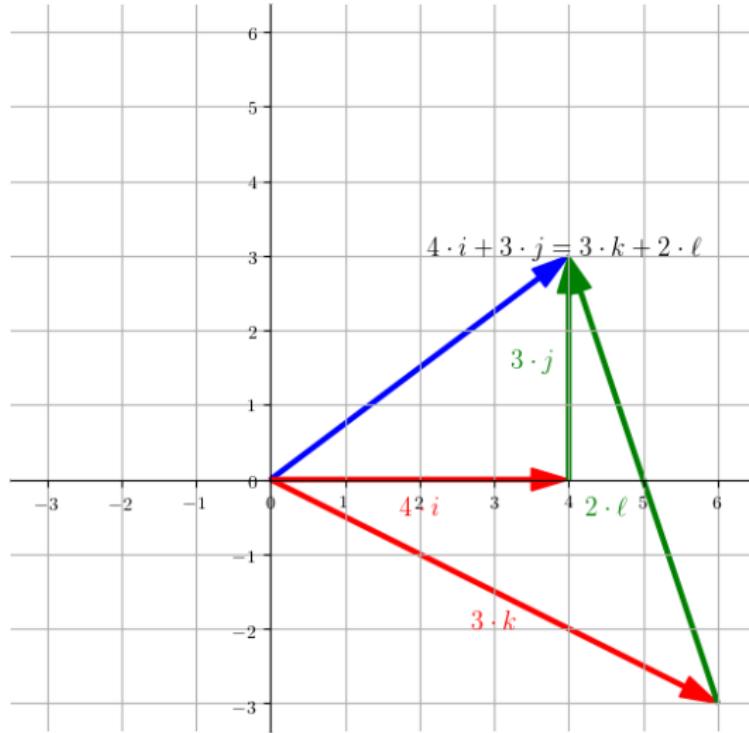
Basis of a vector space



Basis of a vector space



Basis of a vector space



Changing basis

- In the standard basis, the point is $\begin{bmatrix} 4 \\ 3 \end{bmatrix} = 4 \cdot i + 3 \cdot j$
- In the new basis, the point is $\begin{bmatrix} 3 \\ 2 \end{bmatrix} = 3 \cdot k + 2 \cdot \ell$
- The new basis is defined by

$$k = 2 \cdot i - 1 \cdot j \quad \ell = -1 \cdot i + 3 \cdot j$$

- We can transform the point from the new basis to the standard basis

$$\begin{aligned}\begin{bmatrix} 3 \\ 2 \end{bmatrix} &= 3 \cdot k + 2 \cdot \ell \\ &= 3(2 \cdot i - 1 \cdot j) + 2(-1 \cdot i + 3 \cdot j) \\ &= 4 \cdot i + 3 \cdot j = \begin{bmatrix} 4 \\ 3 \end{bmatrix}\end{aligned}$$

Excercise: Changing basis

- In the standard basis, a point is given by $\begin{bmatrix} 5 \\ 7 \end{bmatrix} = 5 \cdot i + 7 \cdot j$
- The new basis is defined by

$$k = 2 \cdot i - 1 \cdot j \quad \ell = -1 \cdot i + 3 \cdot j$$

- Express the point as a coordinate in the new basis

Hint: Solve for i and j in terms of k and ℓ and insert the result

Excercise: Changing basis

- In the standard basis, a point is given by $\begin{bmatrix} 5 \\ 7 \end{bmatrix} = 5 \cdot i + 7 \cdot j$
- The new basis is defined by

$$k = 2 \cdot i - 1 \cdot j \quad \ell = -1 \cdot i + 3 \cdot j$$

- Express the point as a coordinate in the new basis

Hint: Solve for i and j in terms of k and ℓ and insert the result

Solution

Solving for i and j yields

$$i = 0.6k + 0.2\ell \quad j = 0.2k + 0.4\ell$$

$$\begin{aligned}\begin{bmatrix} 5 \\ 7 \end{bmatrix} &= 5i + 7j \\ &= 5(0.6k + 0.2\ell) + 7(0.2k + 0.4\ell) \\ &= 4.4k + 3.8\ell = \begin{bmatrix} 4.4 \\ 3.8 \end{bmatrix}\end{aligned}$$

Basis matrix

- New basis, defined in the standard basis

$$k = 2 \cdot i - 1 \cdot j \quad \ell = -1 \cdot i + 3 \cdot j$$

- Standard basis, defined in the new basis

$$i = 0.6k + 0.2\ell \quad j = 0.2k + 0.4\ell$$

We can write the bases as matrices where each column is a basis vector

$$\begin{bmatrix} 2 & -1 \\ -1 & 3 \end{bmatrix}^{-1} = \begin{bmatrix} 0.6 & 0.2 \\ 0.2 & 0.4 \end{bmatrix} \quad (1)$$

Basis change as matrix multiplication

- In the standard basis, the point is $\begin{bmatrix} 5 \\ 7 \end{bmatrix} = 5 \cdot i + 7 \cdot j$
- The basis is defined by

$$i = 0.6k + 0.2\ell \quad j = 0.2k + 0.4\ell$$

- We transform the point to the new basis as

$$\begin{aligned}\begin{bmatrix} 5 \\ 7 \end{bmatrix} &= 5i + 7j \\ &= 5(0.6k + 0.2\ell) + 7(0.2k + 0.4\ell) \\ &= 5 \begin{bmatrix} 0.6 \\ 0.2 \end{bmatrix} + 7 \begin{bmatrix} 0.2 \\ 0.4 \end{bmatrix} \\ &= \begin{bmatrix} 0.6 & 0.2 \\ 0.2 & 0.4 \end{bmatrix} \begin{bmatrix} 5 \\ 7 \end{bmatrix} = \begin{bmatrix} 4.4 \\ 3.8 \end{bmatrix}\end{aligned}$$

Basis change as matrix multiplication

- General formula for basis change

$$\mathbf{y} = \underbrace{\mathbf{T}^{-1} \mathbf{x}}_{\text{matrix multiplication}}$$

\mathbf{x} Vector in the original coordinate system

\mathbf{T} Matrix where each column is a basis vector of the new coordinate system expressed in the old coordinate system

\mathbf{y} Vector expressed in the new coordinate system

- We can map the other way as

$$\mathbf{x} = \mathbf{T}\mathbf{y}$$

- Orthonormal basis: $\mathbf{T}^{-1} = \mathbf{T}^\top$

Audio

What is sound?

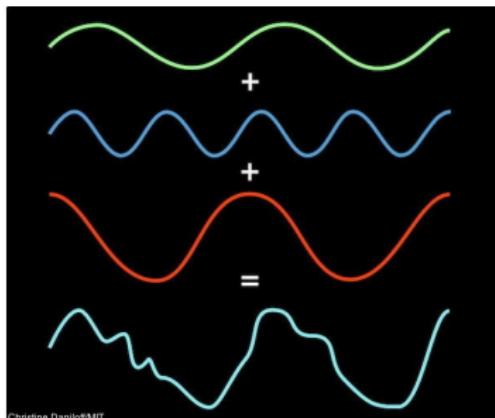
In physics

- Vibrations that propagate as a pressure wave through a transmission medium (such as air)

In psychology

- The reception of a sound wave and its perception by the brain

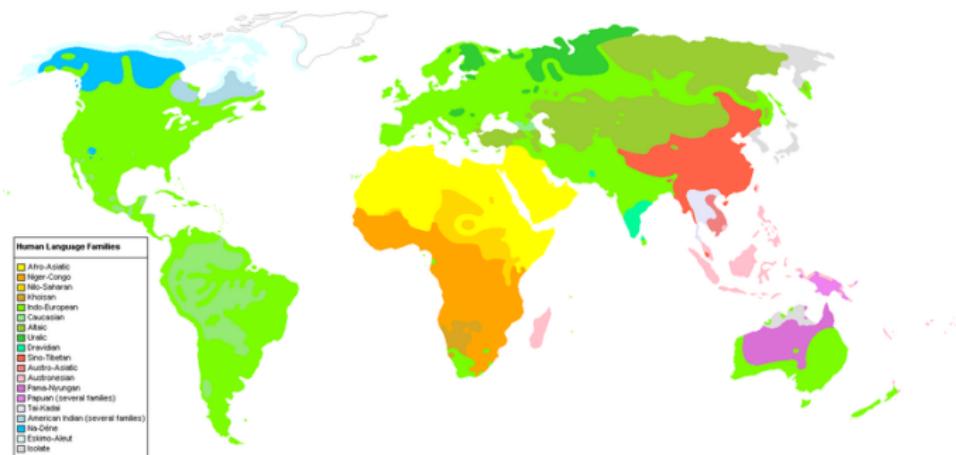
Frequency content



Christine Quillen/MIT

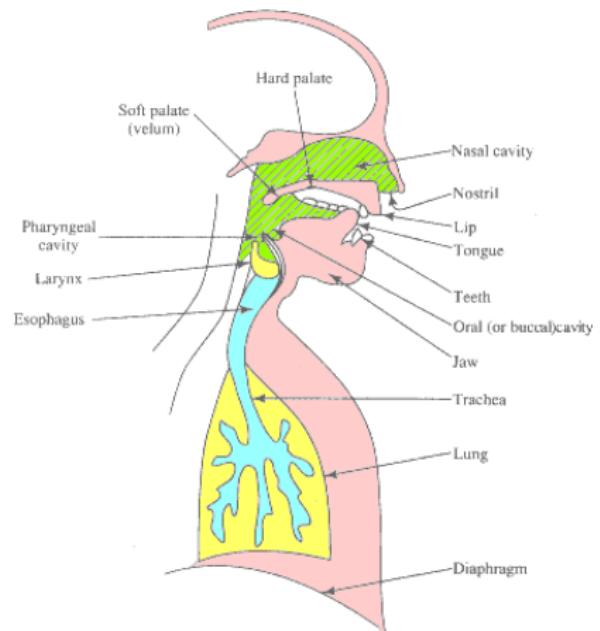
Speech

- Speech signals are sequences of sounds
- The basic sounds and the transitions between them serve as symbolic representation of information: *semantics*
- The arrangement of these sounds (symbols) is governed by the rules of language
- The study of these rules and their implications in human communication is called *linguistics*
- The study of and classification of the sounds of speech is called *phonetics*



Speech production

- Speech is produced by the human vocal tract
- The vocal tract is excited either by short burst of *periodic* signal or by “*noise*” from the flow
- Periodic signals (voiced sounds) are produced by air flow through tight and vibrating vocal cords
- Noise (unvoiced sounds) are produced by turbulent flow with relaxed vocal cords



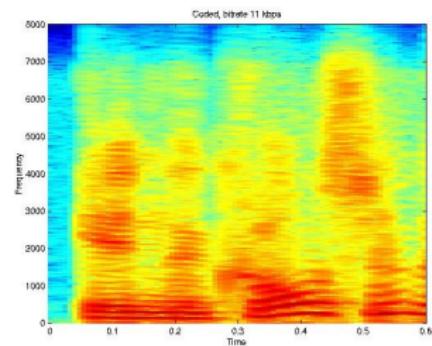
Speech production

- The vocal tract (nasal and oral cavities) transforms the voiced or unvoiced sounds to phonemes (speech sounds)
- Speech sounds are classified broadly into phonemes classes
 - Vowels (voiced)
 - Consonants (unvoiced)
- Phonemes corresponds to *formants*—peaks in the power spectrum modulation (red areas in figure)
- Formant frequencies in the range

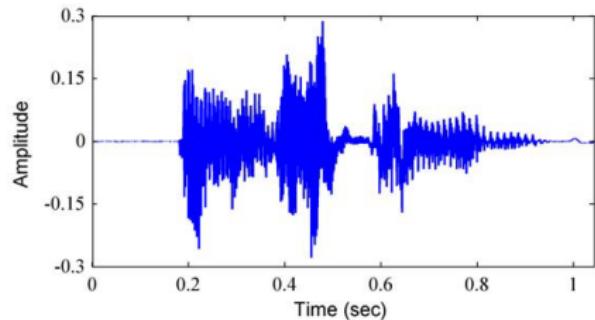
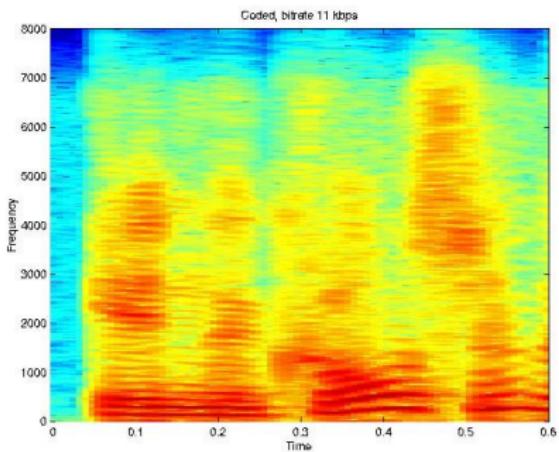
F1 270-730 Hz

F2 840-2290 Hz

F3 1690-3010 Hz

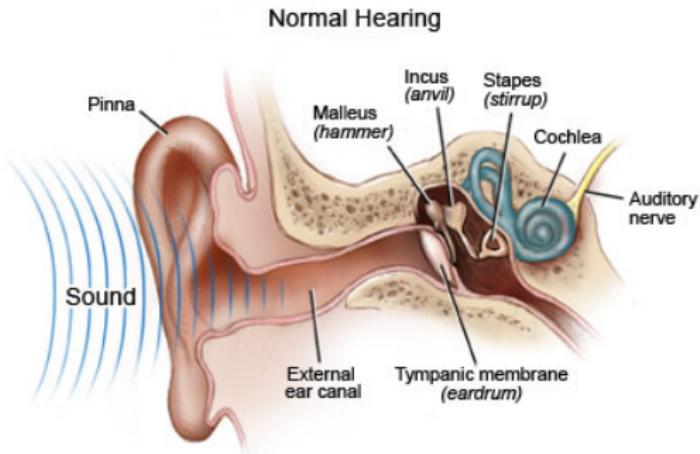


Speech spectrogram



| b | ey | z | th | ih | er | em |
| Bayes' | Theorem |

Human hearing performs frequency analysis



- Cochlea is filled with a watery liquid
- Liquid moves in response to the vibrations coming from the middle ear via the oval window
- Hair cells sense the motion—convert motion to electrical signals
- Communicated via neurotransmitters to many thousands of nerve cells

Human hearing performs frequency analysis

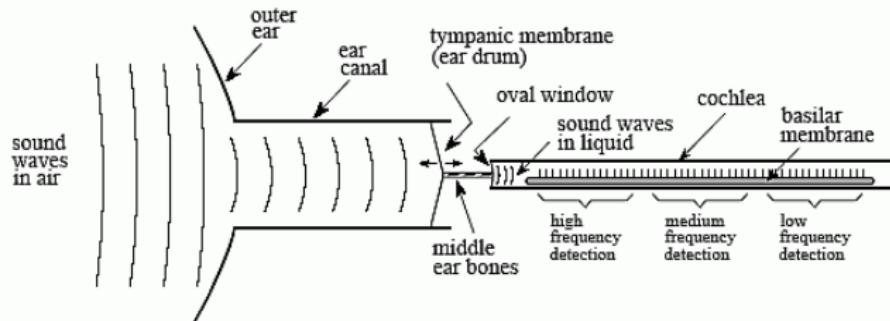


FIGURE 22-1

Functional diagram of the human ear. The outer ear collects sound waves from the environment and channels them to the tympanic membrane (ear drum), a thin sheet of tissue that vibrates in synchronization with the air waveform. The middle ear bones (hammer, anvil and stirrup) transmit these vibrations to the oval window, a flexible membrane in the fluid filled cochlea. Contained within the cochlea is the basilar membrane, the supporting structure for about 12,000 nerve cells that form the cochlear nerve. Due to the varying stiffness of the basilar membrane, each nerve cell only responds to a narrow range of audio frequencies, making the ear a frequency spectrum analyzer.

Sound intensity

TABLE 22-1
Units of sound intensity. Sound intensity is expressed as power per unit area (such as watts/cm²), or more commonly on a logarithmic scale called *decibels SPL*. As this table shows, human hearing is the most sensitive between 1 kHz and 4 kHz.

Watts/cm ²	Decibels SPL	Example sound
10^{-2}	140 dB	Pain
10^{-3}	130 dB	
10^{-4}	120 dB	Discomfort
10^{-5}	110 dB	Jack hammers and rock concerts
10^{-6}	100 dB	
10^{-7}	90 dB	OSHA limit for industrial noise
10^{-8}	80 dB	
10^{-9}	70 dB	
10^{-10}	60 dB	Normal conversation
10^{-11}	50 dB	
10^{-12}	40 dB	Weakest audible at 100 hertz
10^{-13}	30 dB	
10^{-14}	20 dB	Weakest audible at 10kHz
10^{-15}	10 dB	
10^{-16}	0 dB	Weakest audible at 3 kHz
10^{-17}	-10 dB	
10^{-18}	-20 dB	

Softer Louder

Sampling and waveforms

The following is done in order to process audio in a computer

Low pass filtering Frequency content above some upper level is discarded

Sampling The signal is measured at discrete time intervals

Quantization The signal amplitudes are represented as (usually discrete) numbers

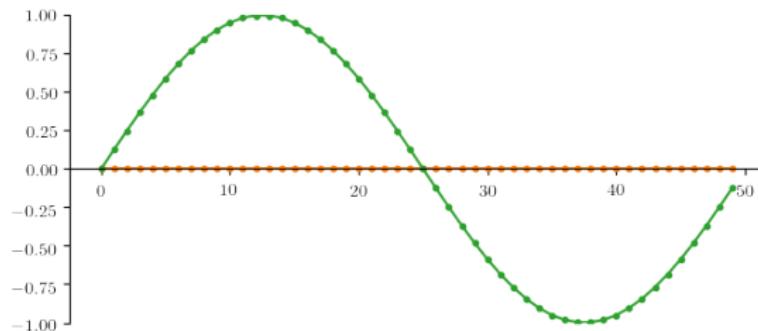
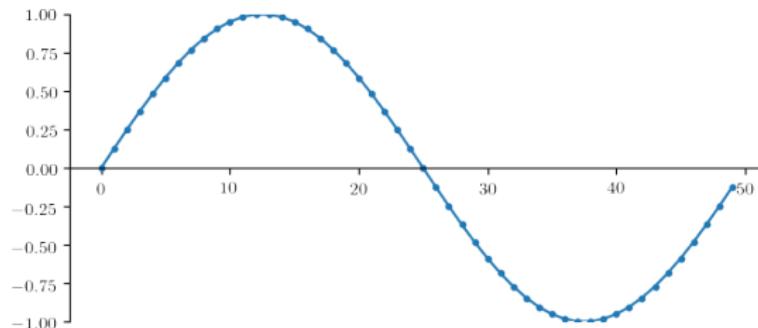
Exercise: Audio as a point in a vector space

An audio signal of length N can be thought of as a point in an N -dimensional vector space, \mathbb{R}^N

- What is the standard basis of this vector space?
- How can we construct any possible audio signal by a linear combination of such basis vectors?
- How do you think each of these basis vectors sounds
- Is this a good basis for representing sound? Can you come up with a better basis, perhaps inspired by the human auditory system?

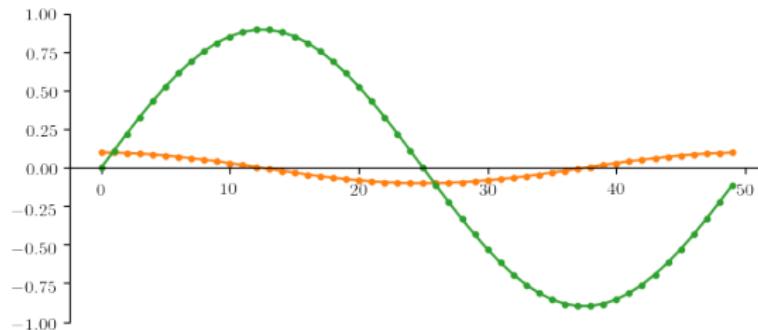
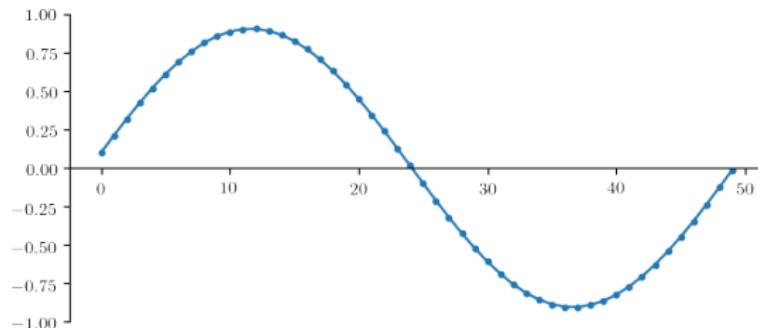
Sum of sinusoids

A weighted sum of a sine and cosine function can give a sinusoid with arbitrary phase-shift



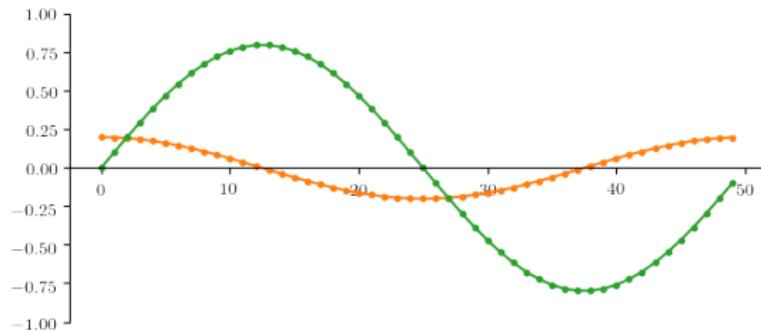
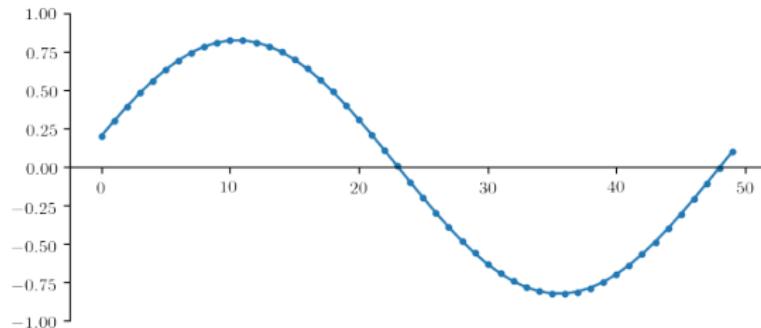
Sum of sinusoids

A weighted sum of a sine and cosine function can give a sinusoid with arbitrary phase-shift



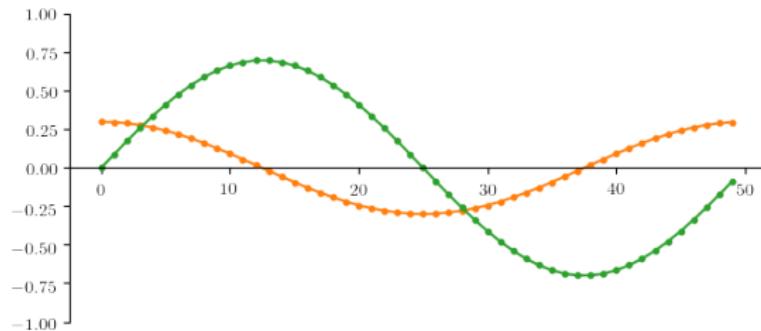
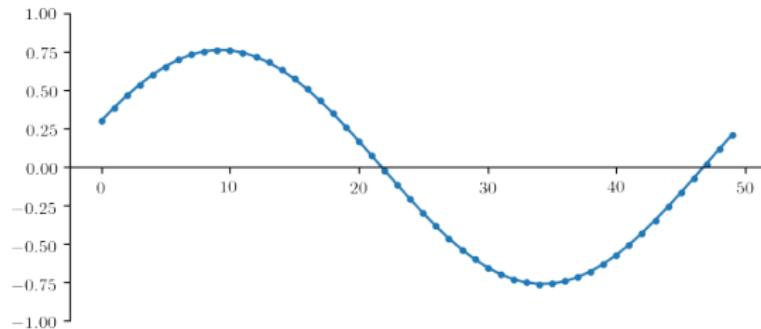
Sum of sinusoids

A weighted sum of a sine and cosine function can give a sinusoid with arbitrary phase-shift



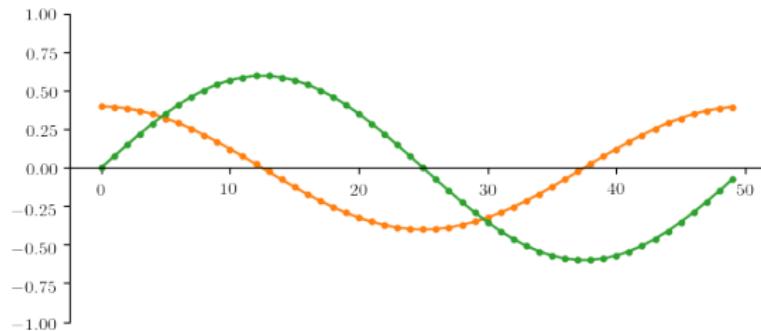
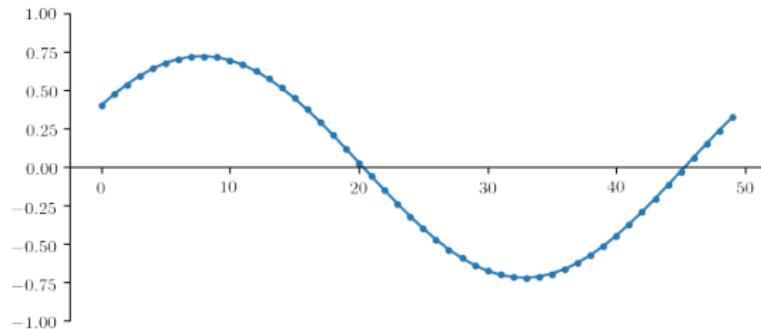
Sum of sinusoids

A weighted sum of a sine and cosine function can give a sinusoid with arbitrary phase-shift



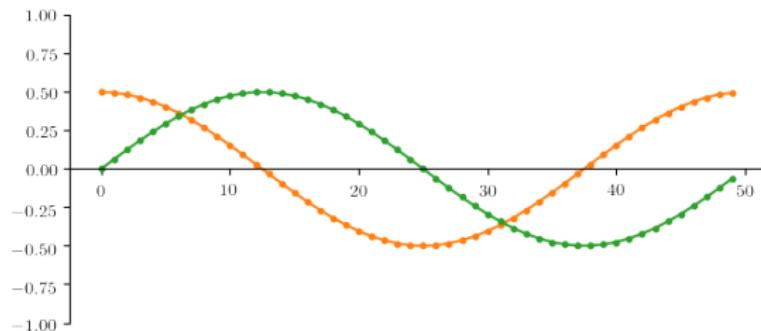
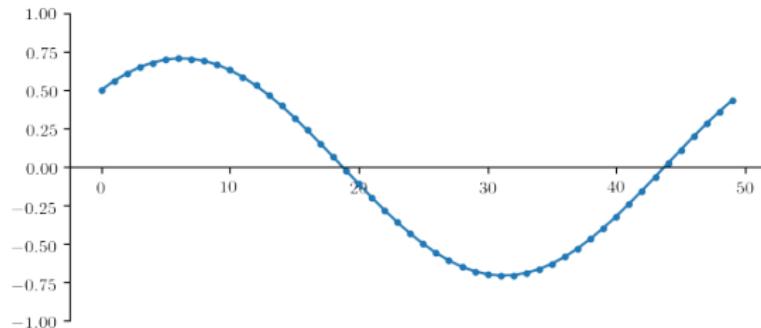
Sum of sinusoids

A weighted sum of a sine and cosine function can give a sinusoid with arbitrary phase-shift



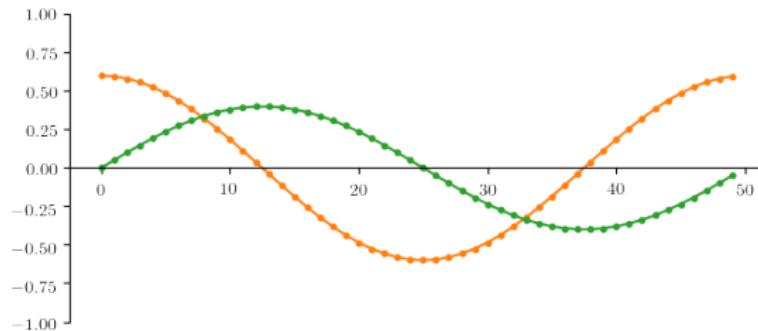
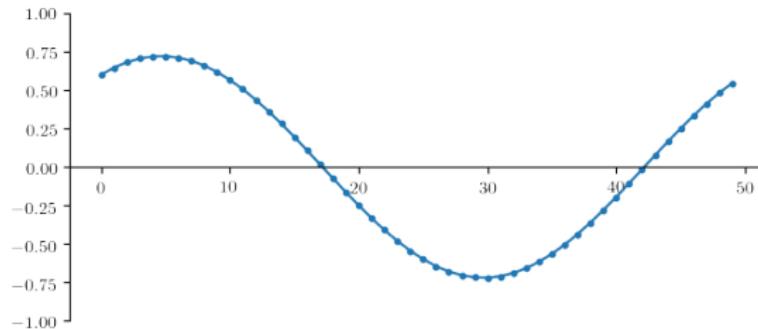
Sum of sinusoids

A weighted sum of a sine and cosine function can give a sinusoid with arbitrary phase-shift



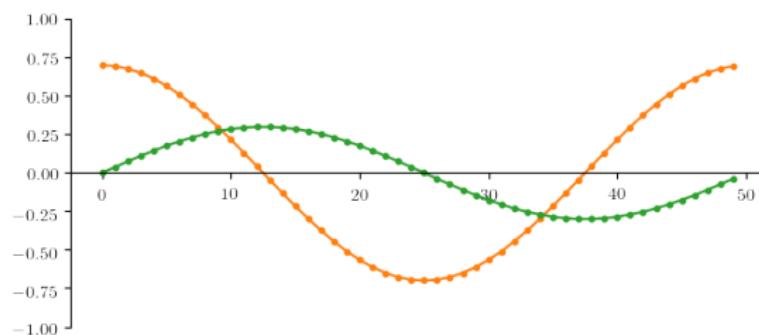
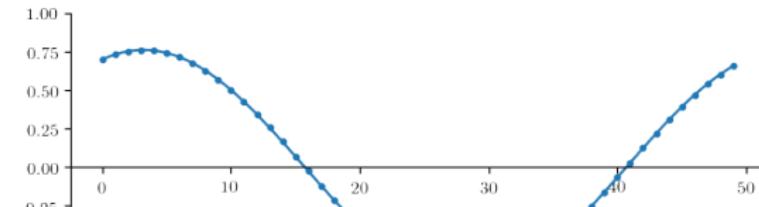
Sum of sinusoids

A weighted sum of a sine and cosine function can give a sinusoid with arbitrary phase-shift



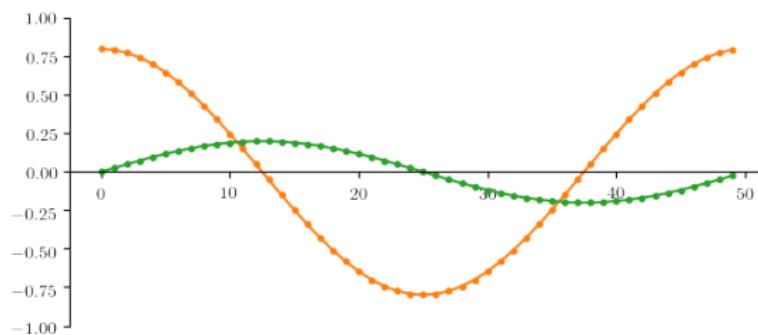
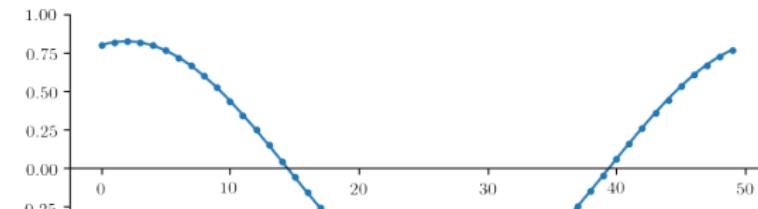
Sum of sinusoids

A weighted sum of a sine and cosine function can give a sinusoid with arbitrary phase-shift



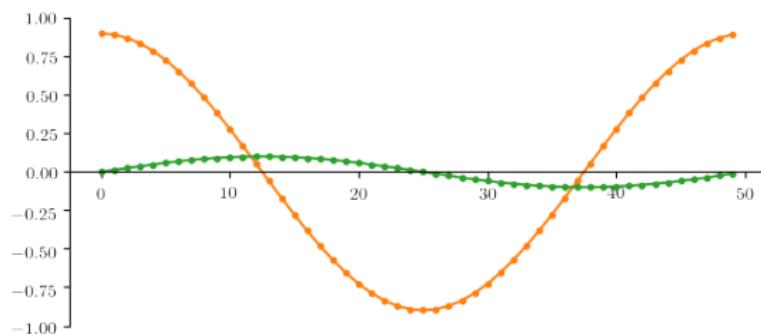
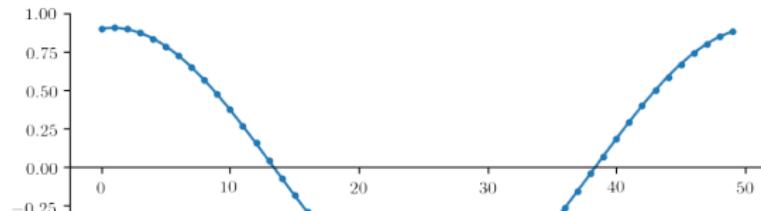
Sum of sinusoids

A weighted sum of a sine and cosine function can give a sinusoid with arbitrary phase-shift



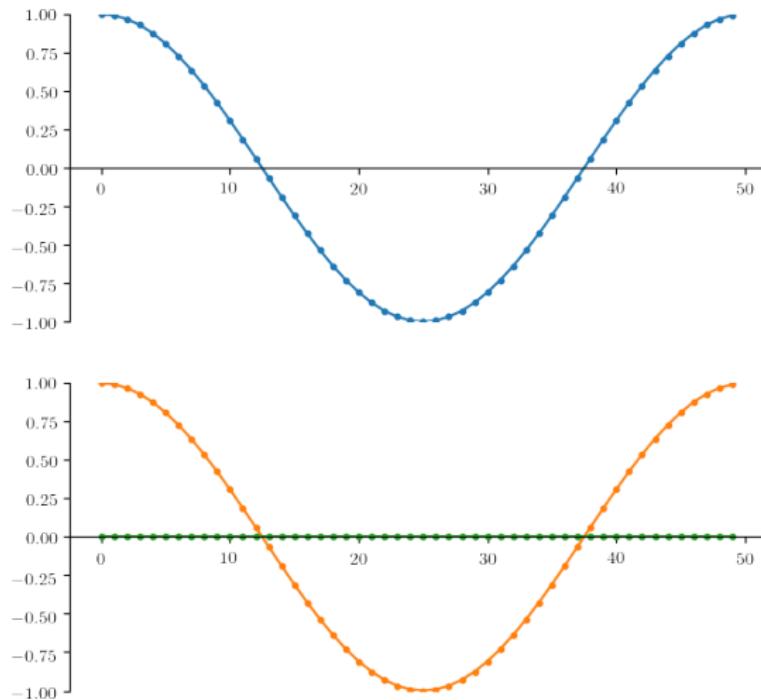
Sum of sinusoids

A weighted sum of a sine and cosine function can give a sinusoid with arbitrary phase-shift



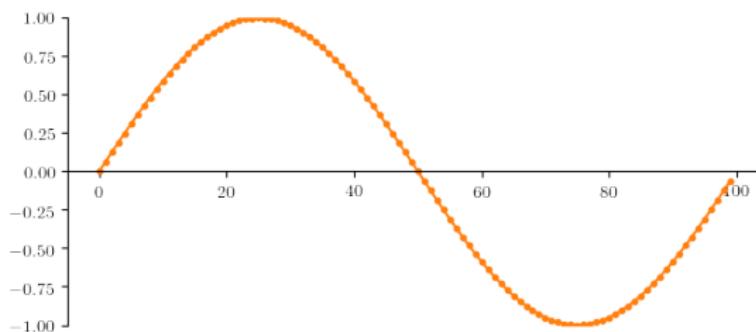
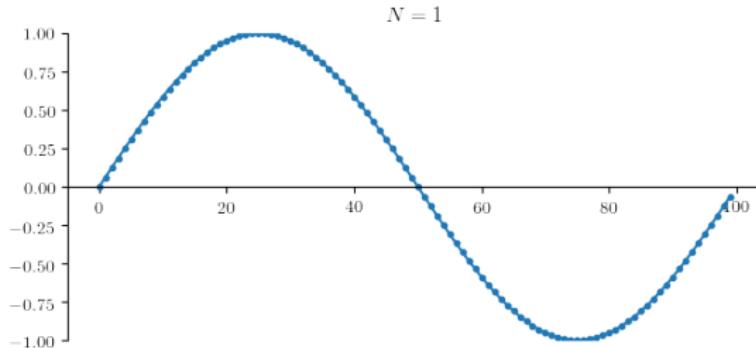
Sum of sinusoids

A weighted sum of a sine and cosine function can give a sinusoid with arbitrary phase-shift



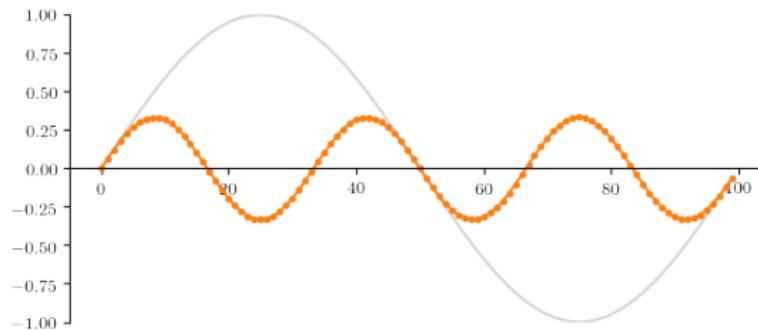
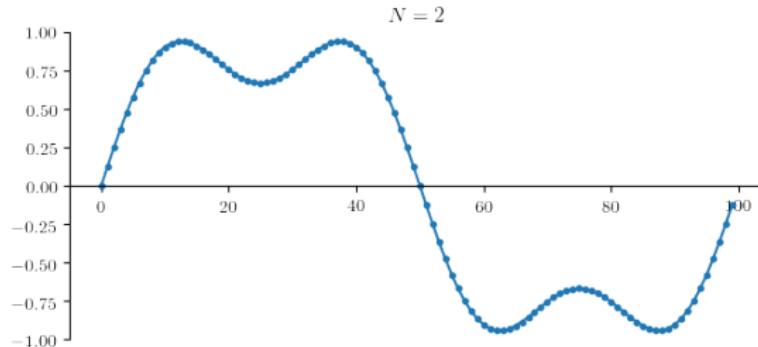
Sum of sinusoids

Any signal can be decomposed as a sum of sinusoids



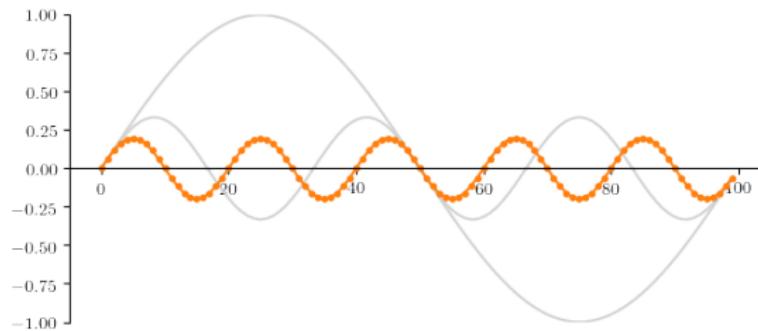
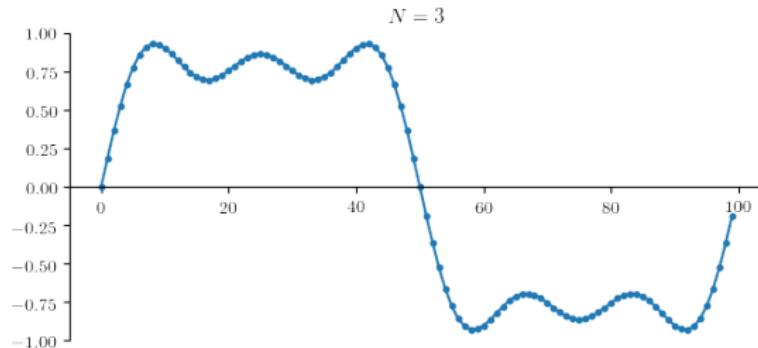
Sum of sinusoids

Any signal can be decomposed as a sum of sinusoids



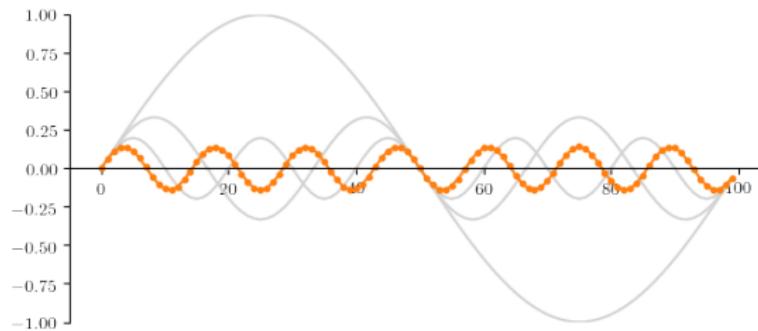
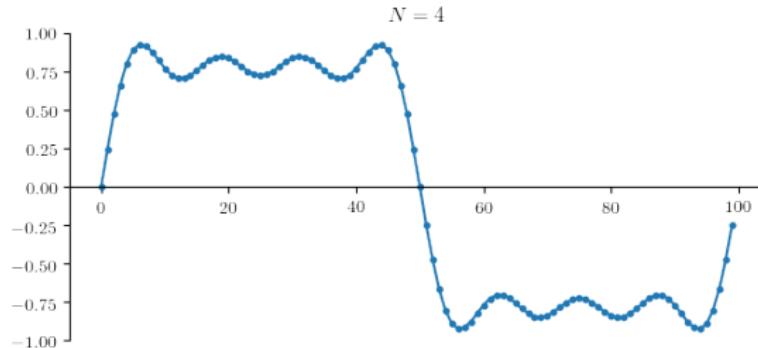
Sum of sinusoids

Any signal can be decomposed as a sum of sinusoids



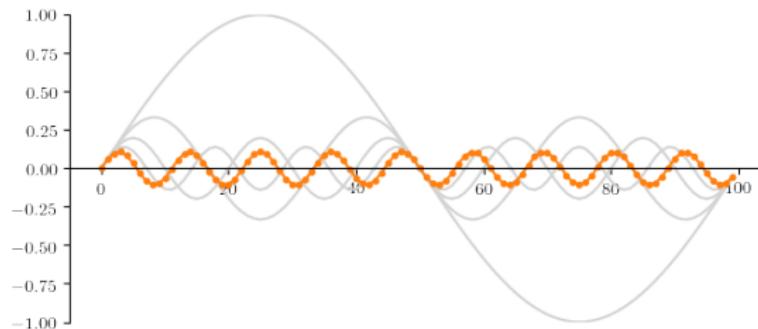
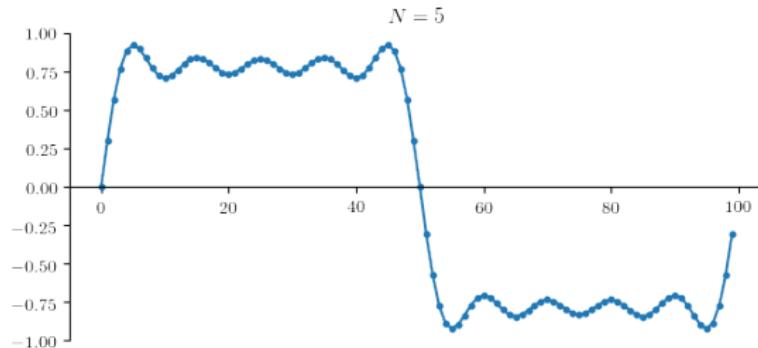
Sum of sinusoids

Any signal can be decomposed as a sum of sinusoids



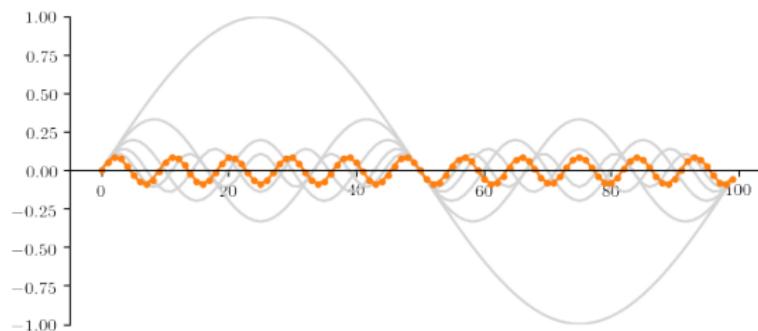
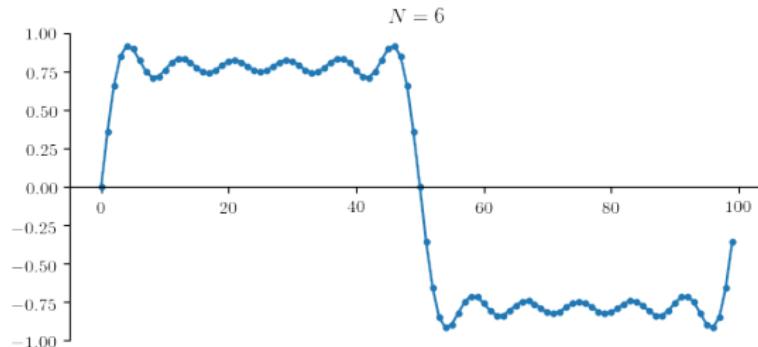
Sum of sinusoids

Any signal can be decomposed as a sum of sinusoids



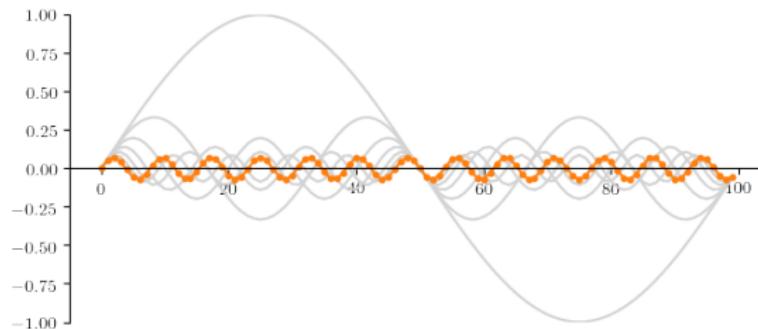
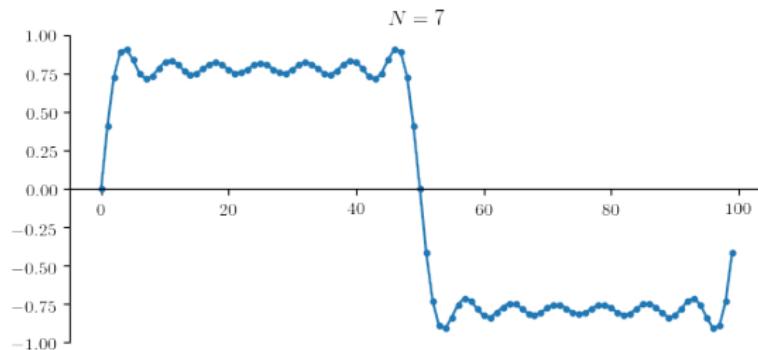
Sum of sinusoids

Any signal can be decomposed as a sum of sinusoids



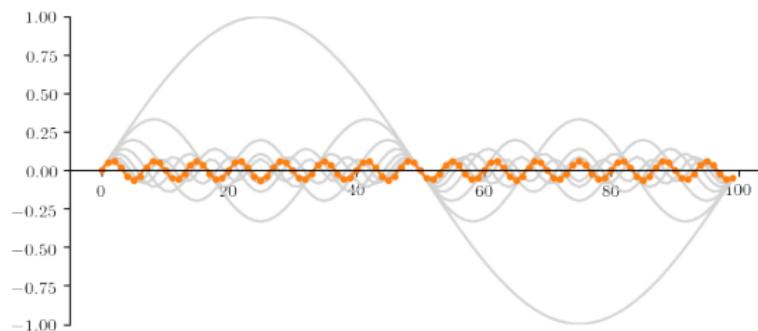
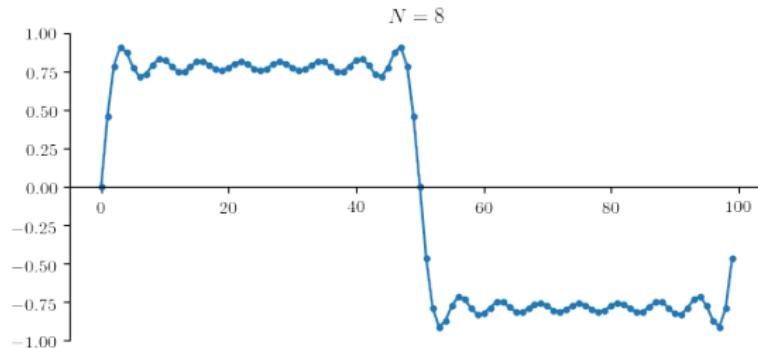
Sum of sinusoids

Any signal can be decomposed as a sum of sinusoids



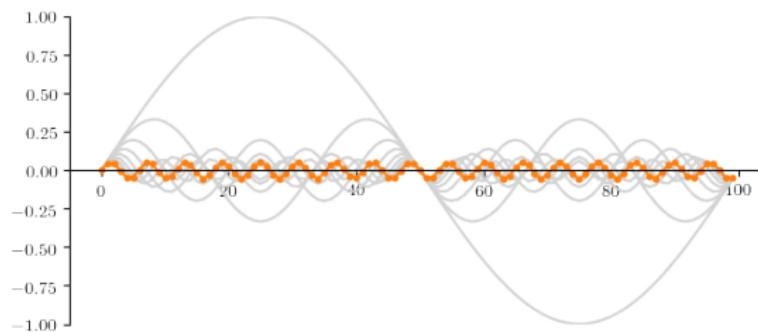
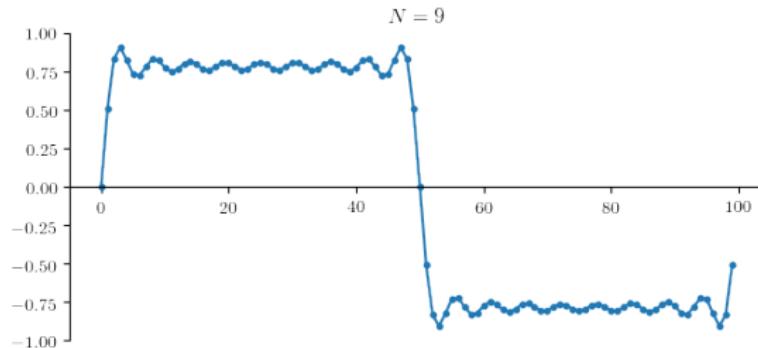
Sum of sinusoids

Any signal can be decomposed as a sum of sinusoids



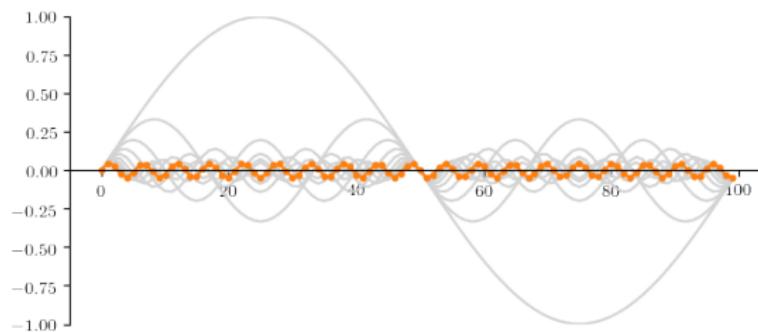
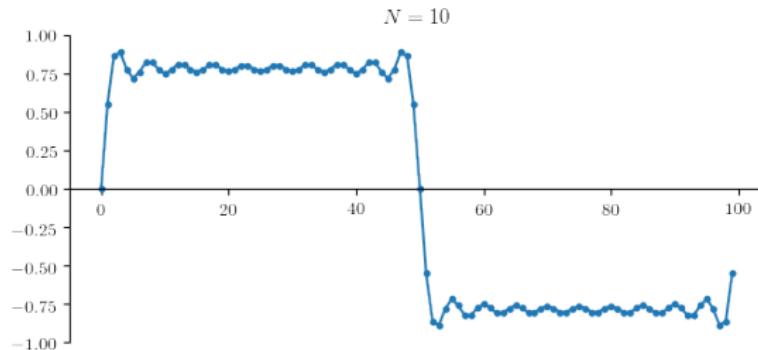
Sum of sinusoids

Any signal can be decomposed as a sum of sinusoids



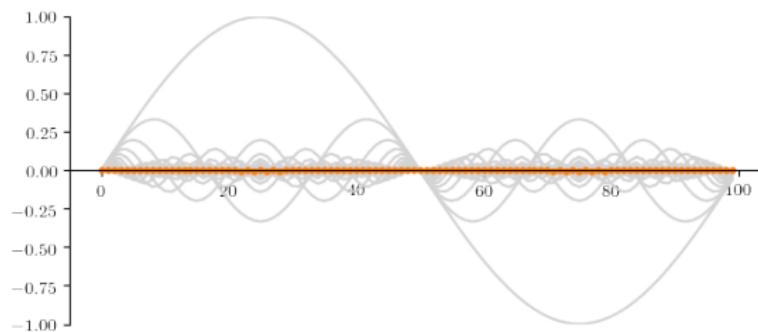
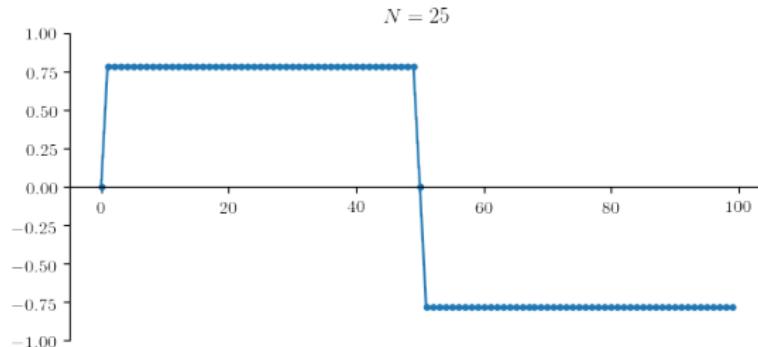
Sum of sinusoids

Any signal can be decomposed as a sum of sinusoids



Sum of sinusoids

Any signal can be decomposed as a sum of sinusoids



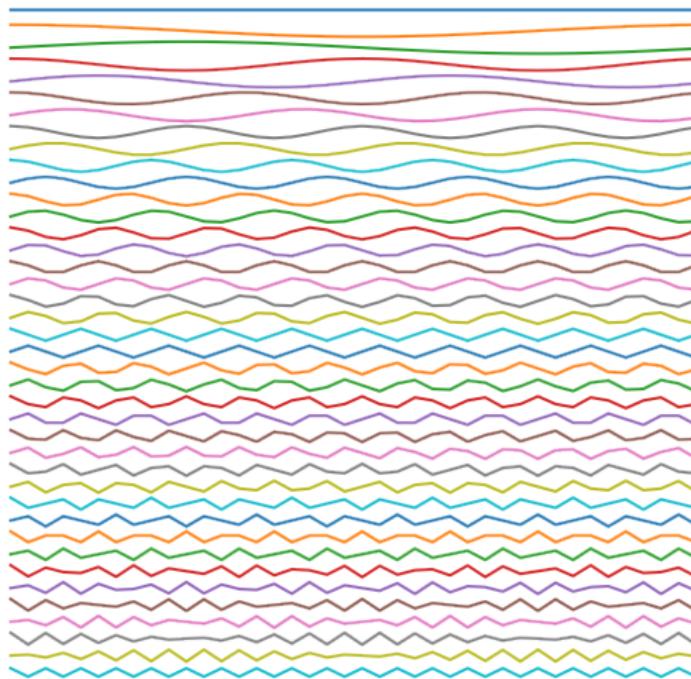
A sine+cosine basis

A basis of sine and cosine functions (for even N)

$$T = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 \\ \sqrt{2} \cos(t) \\ \sqrt{2} \sin(t) \\ \sqrt{2} \cos(2t) \\ \sqrt{2} \sin(2t) \\ \vdots \\ \sqrt{2} \cos([\frac{N}{2} - 1]t) \\ \sqrt{2} \sin([\frac{N}{2} - 1]t) \\ \cos(\frac{N}{2}t) \end{bmatrix}^\top$$

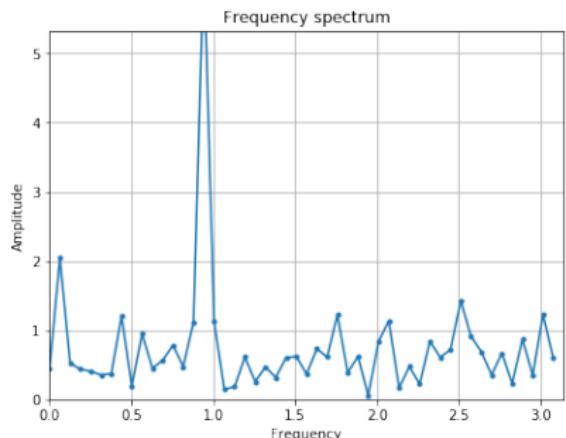
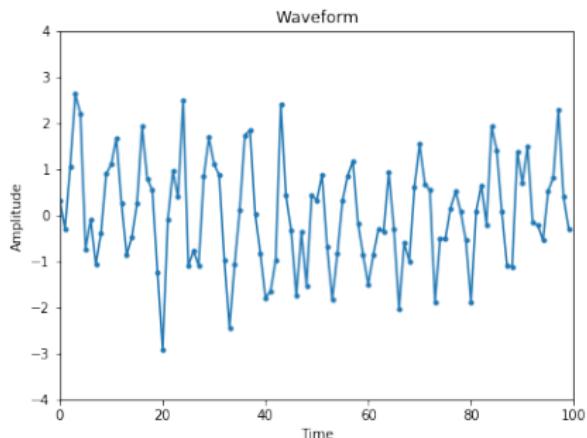
$$t = \frac{2\pi n}{N}, \quad n = [0, 1, 2, \dots, N-1]$$

A sine+cosine basis



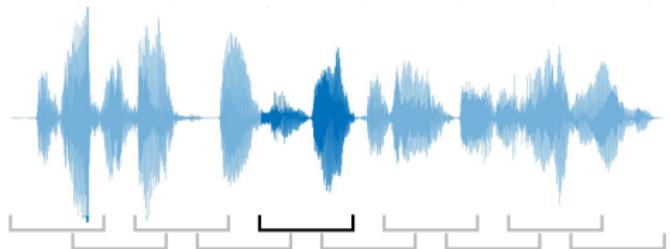
Frequency spectrum

- Map waveform onto sine+cosine basis
- Combine amplitude of sine and cosine at each frequency
- Amplitude spectrum (invariant to phase shift)

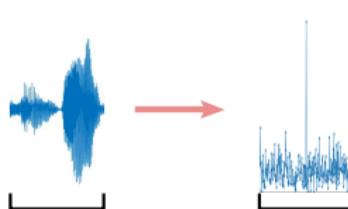


Spectrogram

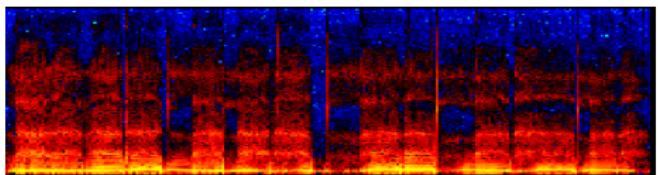
Split the signal
into blocks



For each block,
compute the spectrum



Gather spectra as columns
in matrix and plot heat map



Tasks

Tasks for today

1. Work through the three notebooks on audio analysis
`10-SinusoidsInNoise.ipynb` `10-Spectrogram.ipynb`
`10-AudioClassification.ipynb`
2. Today's feedback group
 - Selma Bundgaard Langvik
 - Andreas Holm Matthiassen
 - Jacob Danvad Nalholm
 - Mikkel Nielsen Broch-Lips

Lab report

- Lab 4: Neural networks (Deadline: Thursday 9 November 20:00)

Introduction to intelligent systems

Reinforcement learning

Mikkel N. Schmidt

Technical University of Denmark,
DTU Compute, Department of Applied Mathematics and Computer Science.

Overview

① Reinforcement learning

② Python dictionaries

③ Tasks

Feedback group

- Karl Johan Murphy Mogensen
- Rasmus Grønnegaard Arnmark
- Mikel Taotao Yu
- Haaris Usman Syed

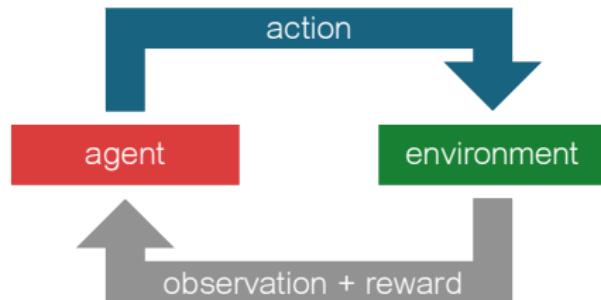
Learning objectives

- I Reinforcement learning: Markov decision process (state, action, reward).
 - I Epsilon-greedy action selection and optimistic initialization.
 - II RL algorithms: Value iteration and Q-learning.
 - II Optimal action and optimal policy.
 - II Discount factor.
 - II Value (of a state) and quality (of a state-action pair).
-
- I Understand the concepts and definitions, and know their application. Reason about the concepts in the context of an example. Use correct technical terminology.
 - II As above plus: Read, manipulate, and work with technical definitions and expressions (mathematical and Python code). Carry out practical computations. Interpret and evaluate results.

Reinforcement learning

Reinforcement learning

Learn a function (policy) that maps inputs to actions to optimize cumulative reward



Markov decision process (MDP)

In the general setting where state transitions and rewards are stochastic, the MDP is defined by

\mathcal{S} Set of states.

\mathcal{A} Set of actions.

$p(s'|s, a)$ Probability of next state s' given current state s and action a .

$p(r|s', s, a)$ Distribution of reward for transitioning to state s' from state s using action a .

Markov decision process (deterministic setting)

In the deterministic setting, the MDP is defined by

\mathcal{S} Set of states.

\mathcal{A} Set of actions.

$s' = f(s, a)$ Next state s' is determined from current state s and action a .

$r = r(s, a)$ Reward for taking action a in state s .

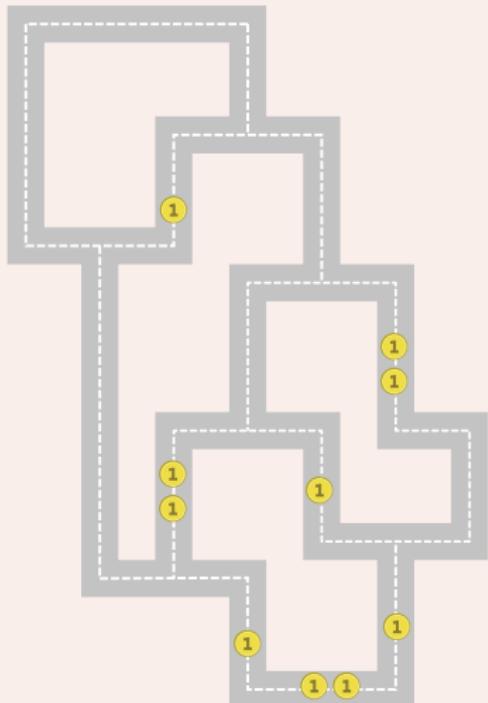
- The next state is deterministic, i.e. given as a function of the current state and action.
- Rewards depend only on current state and action.

Exercise: Collecting gold coins

Consider a game, where we drive around and collect gold coins (coins can be picked up multiple times.)

How could we meaningfully define:

- The set of states
- The set of actions
- The next-state function
- The reward



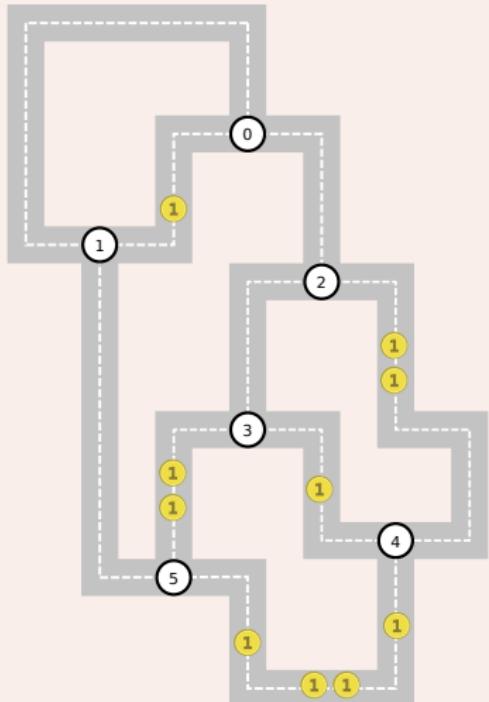
Discussion: Collecting gold coins

- The set of states could be defined as places where the road forks.
- The set of actions could be defined as north, south, east west. We need to consider what would happen if we take an “illegal” action.
- The next state is deterministic: Follow the road to the next fork.
- The reward could be the number of coins collected.

Exercise: Optimal policy

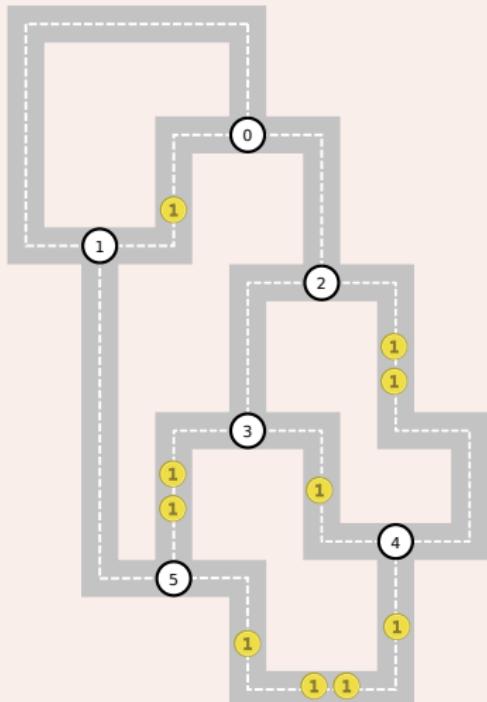
What is the *optimal policy*?

Hint: What should we end up doing, if we follow the optimal policy?



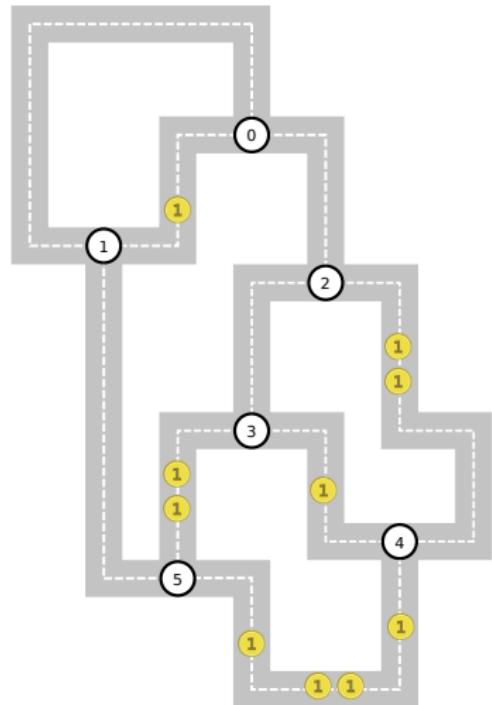
Solution: Optimal policy

- The optimal policy should end up going back and forth between state 4 and 5, collecting 4 gold coins in each step.
- From state 3, the optimal policy would be to go to state 5, since this gives us 2 gold coins.
- From state 2 go to state 4.
- From state 1 go to state 5.
- From state 0 go to state 1 or 2. This depends on whether we prioritize getting 1 coin immediately or 2 coins a bit later.



Reward

Reward		Action, a			
	$r(s, a)$	N	S	E	W
State, s	0	0	0	0	1
	1	0	0	1	0
	2	0	0	2	0
	3	0	0	1	2
	4	0	4	2	1
	5	2	0	4	0



Value function

In the deterministic setting, the value function is defined recursively as

Value function (deterministic setting)

$$v(s) = \max_a (r(s, a) + \gamma v(s'))$$

Value of state 5

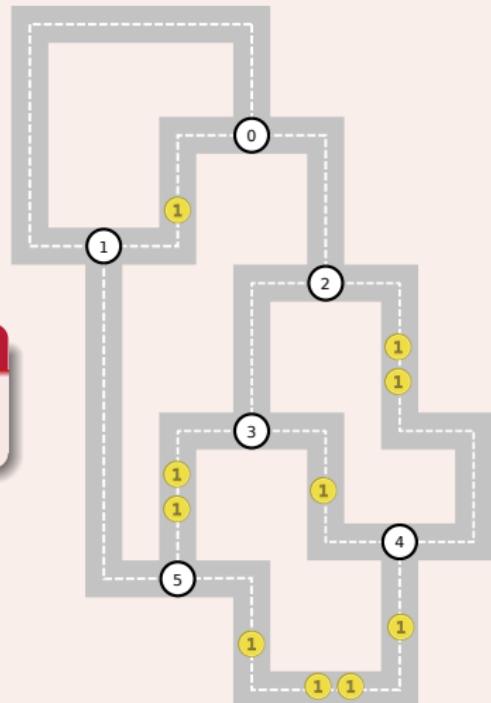
- The optimal policy will end up going back and forth between state 4 and 5.
- We will use $\gamma = 0.9$

What is the value of state 5?

Hint: We can deduce that $v(4) = v(5)$ from the optimal policy.

Value function (deterministic setting)

$$v(s) = \max_a (r(s, a) + \gamma v(s'))$$



Value of state 5

- The optimal policy will end up going back and forth between state 4 and 5.
- We will use $\gamma = 0.9$

What is the value of state 5?

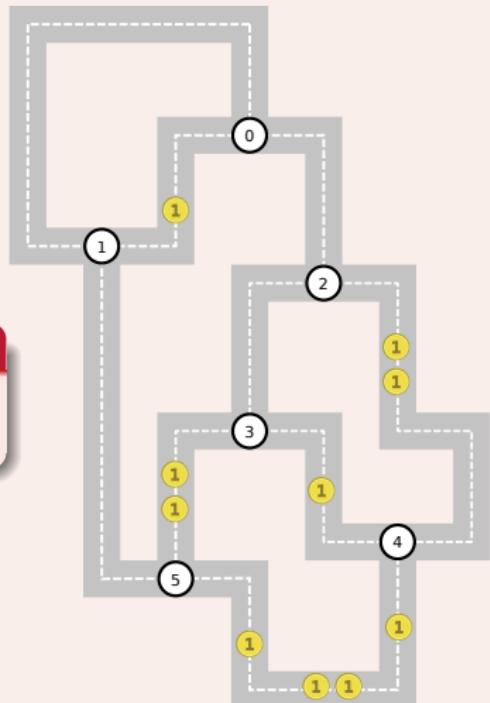
Hint: We can deduce that $v(4) = v(5)$ from the optimal policy.

Value function (deterministic setting)

$$v(s) = \max_a (r(s, a) + \gamma v(s'))$$

Solution

$$\begin{aligned}v(5) &= r(5, \text{East}) + \gamma v(4) = 4 + \gamma v(5) \\&= \frac{4}{1 - \gamma} = \underline{40}\end{aligned}$$



Value iteration

- Loop through all states and update according to

$$v(s) = \max_a (r(s, a) + \gamma v(s'))$$

- Repeat until convergence

Value iteration

```
# Initial values          # 1000 value iterations
V = [0,0,0,0,0,0]         for t in range(1000):
# Discount                # Loop over all states
gamma = 0.9                 for s in range(6):
# Actions: 0=North,        # Update value
# 1=South, 2=East, 3=West    V[s] = max([r+gamma*V[sp] for r,sp in zip(R[s], F[s])])
actions = [0, 1, 2, 3]

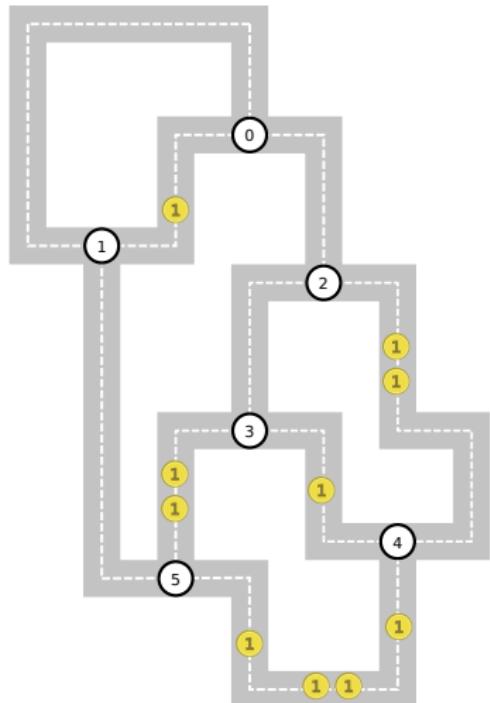
# Next state table
F = [[1, 0, 2, 1],
      [1, 5, 0, 0],
      [0, 2, 4, 3],
      [2, 3, 4, 5],
      [4, 5, 2, 3],
      [3, 5, 4, 1]]

# Reward table
R = [[0, 0, 0, 1],
      [0, 0, 1, 0],
      [0, 0, 2, 0],
      [0, 0, 1, 2],
      [0, 4, 2, 1],
      [2, 0, 4, 0]]
```

Estimated value function

After running the code, we arrive at the following value function

State, s	0	1	2	3	4	5
Value, $v(s)$	34.2	36	38	38	40	40



Model-based and model-free

Model based We know the state transition function.

Example: Value iteration

Model free We can only learn about state transitions by interacting with the environment

Example: Q-learning

Quality function

In the deterministic setting, the quality function is defined recursively as

Quality function (deterministic setting)

$$q(s, a) = r(s, a) + \gamma \max_{a'} q(s', a')$$

The quality of taking action a in state s is

- The immediate associated reward +
- The discounted quality of the best action in the next state.

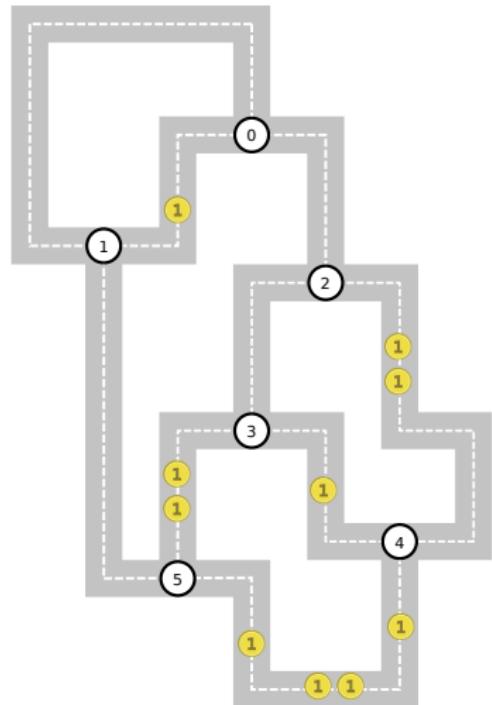
Q-learning

- Explore the environment according to some policy that ensures visiting all state-action pair
- At each step, update the quality function according to

$$q(s, a) = r(s, a) + \gamma \max_{a'} q(s', a')$$

Q-learning example

State, s	Quality $q(s, a)$				Action, a
	N	S	E	W	
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	

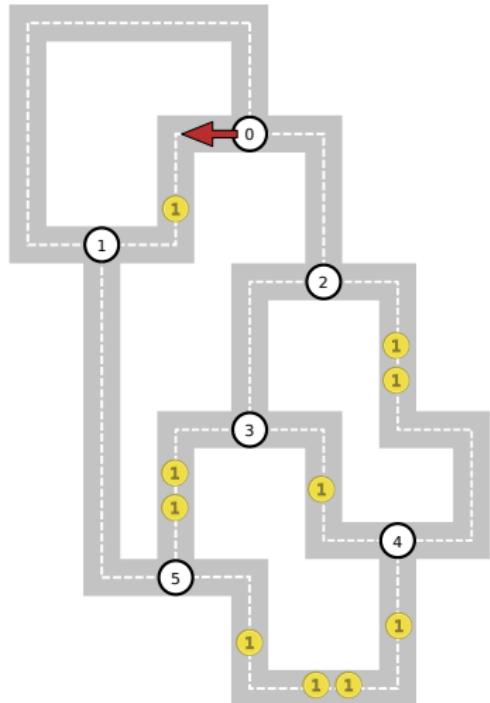


Q-learning example

State, s	Quality $q(s, a)$				Action, a
	N	S	E	W	
0	0	0	0	1	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	

$$q(0, \text{W}) = r(0, \text{W}) + \gamma \max_{a'} q(1, a')$$

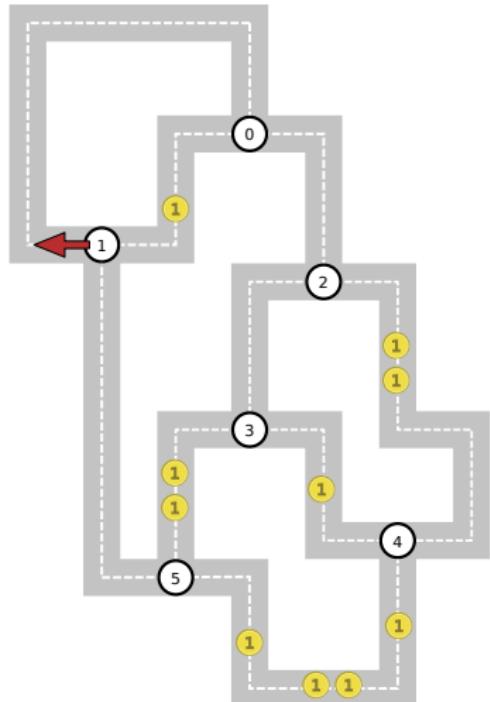
$$= 1 + 0 = 1$$



Q-learning example

Quality $q(s, a)$		Action, a			
State, s		N	S	E	W
0		0	0	0	1
1		0	0	0	0.9
2		0	0	0	0
3		0	0	0	0
4		0	0	0	0
5		0	0	0	0

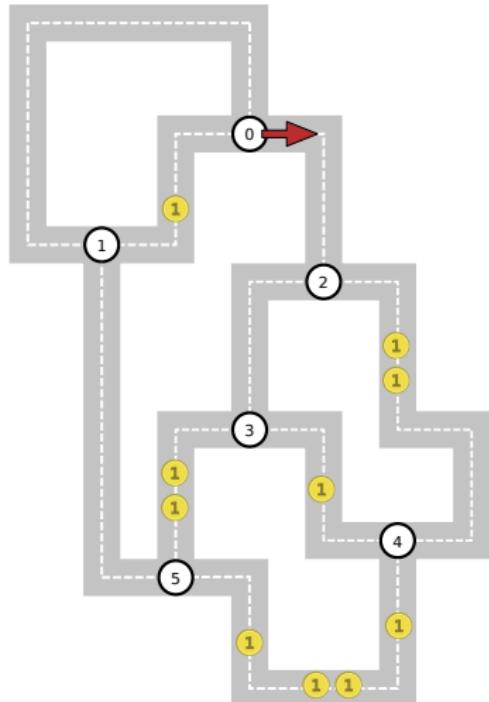
$$\begin{aligned}
 q(1, \text{W}) &= r(1, \text{W}) + \gamma \max_{a'} q(0, a') \\
 &= 0 + 0.9 \cdot 1 = 0.9
 \end{aligned}$$



Q-learning example

State, s	Quality $q(s, a)$				Action, a
	N	S	E	W	
0	0	0	0	1	
1	0	0	0	0.9	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	

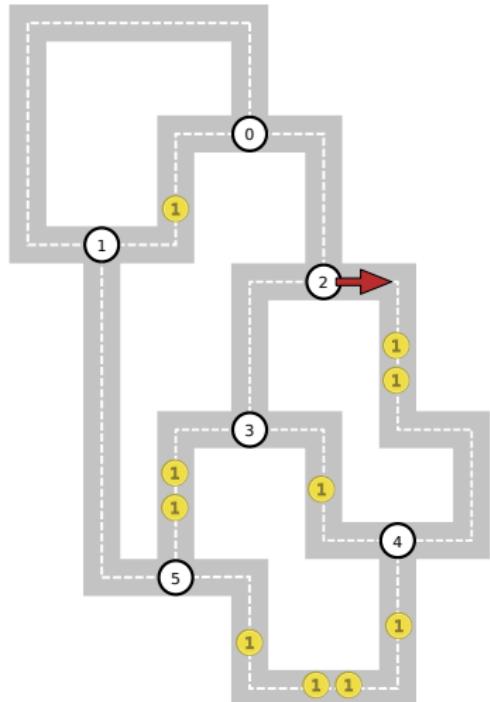
$$\begin{aligned} q(0, \text{E}) &= r(0, \text{E}) + \gamma \max_{a'} q(2, a') \\ &= 0 + 0.9 \cdot 0 = 0 \end{aligned}$$



Q-learning example

State, s	Quality $q(s, a)$				Action, a
	N	S	E	W	
0	0	0	0	1	
1	0	0	0	0.9	
2	0	0	2	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	

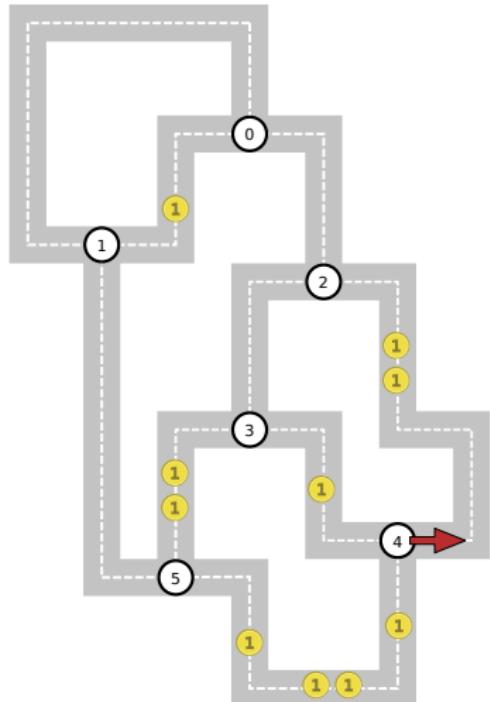
$$\begin{aligned}
 q(2, \text{E}) &= r(2, \text{E}) + \gamma \max_{a'} q(4, a') \\
 &= 2 + 0.9 \cdot 0 = 2
 \end{aligned}$$



Q-learning example

Quality $q(s, a)$		Action, a			
		N	S	E	W
State, s	0	0	0	0	1
	1	0	0	0	0.9
	2	0	0	2	0
	3	0	0	0	0
	4	0	0	3.8	0
	5	0	0	0	0

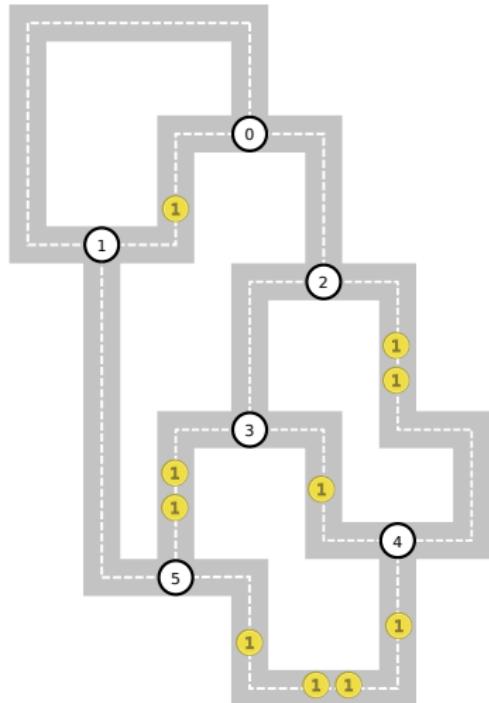
$$\begin{aligned}
 q(4, \text{E}) &= r(4, \text{E}) + \gamma \max_{a'} q(2, a') \\
 &= 2 + 0.9 \cdot 2 = 3.8
 \end{aligned}$$



Q-learning example

Final q-table

		Action, a			
		N	S	E	W
Quality $q(s, a)$		0	32.4	30.8	34.2
State, s	0	32.4	30.8	34.2	33.4
	1	32.4	36.0	31.8	30.8
	2	30.8	34.2	38.0	34.2
	3	34.2	34.2	37.0	38.0
	4	36.0	40.0	36.2	35.2
	5	36.2	36.0	40.0	32.4



Epsilon-greedy exploration

The epsilon-greedy policy is one way to explore the environment, that mixes *exploration* and *exploitation*.

With probability

- ϵ Take a random action.
- $1 - \epsilon$ Take the best action according to the current estimate of the quality function.

The best action is simply given as

$$a^* = \max_a q(s, a)$$

Exploration by optimistic initialization

Another way to ensure exploration is to use *optimistic initialization*.

Here, we always take the best action according to the current estimate of the quality function.

- The quality of all state-action pairs are initialized with a (relatively) high value.
- When the agent receives its reward, it will be lower than the initial values.
- The agent then avoids actions that lead to this low reward.
- After a while, all actions have been explored, and the quality function converges.

Python dictionaries

Dictionaries

A Python dictionary is a data structure that associates *keys* with *values*.

```
>>> my_dict = {'a': [0,1,2], 'b': [3,4,5]}\n>>> my_dict\n{'a': [0, 1, 2], 'b': [3, 4, 5]}
```

```
>>> my_dict['a']\n[0, 1, 2]
```

```
>>> my_dict['b'][2]\n5
```

```
>>> my_dict['b'][2] = 10\n>>> my_dict\n{'a': [0, 1, 2], 'b': [3, 4, 10]}
```

```
>>> my_dict['c']\nTraceback (most recent call last):\n  File "<stdin>", line 1, in <module>\nKeyError: 'c'
```

Default-dictionaries

In a default-dictionary, we have a function that specifies the default value for undefined keys.

```
>>> from collections import defaultdict
>>> my defaultdict = defaultdict(lambda: [0, 0, 0])
>>> my defaultdict
defaultdict(<function <lambda> at 0x7f6836046200>, {})
>>> my defaultdict['a'] = [1,2,3]
>>> my defaultdict['a']
[1, 2, 3]
>>> my defaultdict['c']
[0, 0, 0]
```

Tasks

Tasks for today

1. Today's feedback group

- Karl Johan Murphy Mogensen
- Rasmus Grønnegaard Arnmark
- Mikel Taotao Yu
- Haaris Usman Syed

Lab report

- Lab 5: Reinforcement learning (Deadline: Thursday 23 November 20:00)

12 Causality

In causal estimation one of the primary goals is to estimate the strength of the causal effect of one variable on another. In other words: We are interested in finding out how much the value of one variable influences the value of another variable. This will allow us to answer causal questions such as: “If I were to modify variable x , how much would I expect variable y to change?”. This is not always an easy problem, and the answer most often depends on additional knowledge that is not evident in the statistical data alone.

Example 12.1 Bullet holes in air planes¹

During the second world it was noticed that air planes returning from battle had more bullet holes in the fuselage than in the engine. At first, this finding suggested that more armour should be added to the fuselage to protect them where they seemed to get hit the most. However, a statistician who examined the case came to the opposite conclusion: They should add more armour to the engine. While at first this might seem backward, the argument is clear: If we assume that the bombers are equally likely to get hit everywhere, the fact that the air planes that return have more bullet holes in the fuselage suggests that planes that are hit in the engine are more likely to crash and therefore not return. The planes that do return are the ones that are hit where it does not matter so much.

Often causal estimation is discussed in the language of experimental design, where we imagine a hypothetical study where some units are given a treatment x and we measure the outcome y . The treatment could be a binary variable (some units receive the treatment, others get no treatment) or it could be a

¹ This example is based on a true story as told by Jordan Ellenberg in the book “How Not to Be Wrong: The Power of Mathematical Thinking” regarding the statistician Abraham Wald who published his results in the 1943 report “A Method of Estimating Plane Vulnerability Based on Damage of Survivors”

continuous variable (for example the size of a dose of medicine). Similarly, the outcome could be binary or continuous.

12.1 Statistical dependence

Since causal estimation is concerned with measuring the causal dependence between variables, we begin by discussion how to measure statistical dependence. Here, it is easier to begin by defining when two variables are *not* associated, i.e. there is no statistical dependence.

Definition 12.1 Statistical independence

Two variables x and y are said to be statistically independent if there is no association between the variables:

Knowing something about x tells us absolutely nothing about y and vice versa. Technically we say that the joint probability of x and y factorizes as the product of the individual distributions $p(x)$ and $p(y)$,

$$p(x, y) = p(x)p(y).$$

Example 12.2 Independent coin flips

If we flip a fair coin we get a random outcome of either heads or tails, and each outcome occurs with 50% probability. If we flip the coin twice and call the two outcomes x and y , we will observe one of the following sequences, each of which occur with probability 25%:

x	y	$p(x, y)$
heads	heads	0.25
heads	tails	0.25
tails	heads	0.25
tails	tails	0.25

In this case x and y are statistically independent: Knowledge about the outcome of one of the coin flips tells us nothing about the outcome of the other. We can verify this using the definition of statistical independence. Using the

individual probabilities of the outcomes

$$\begin{aligned} p(x = \text{heads}) &= p(x = \text{tails}) = 0.5 \\ p(y = \text{heads}) &= p(y = \text{tails}) = 0.5 \end{aligned}$$

we can verify that $p(x,y) = p(x)p(y) = 0.25$ for any value of x and y .

12.1.1 Linear dependence

Obviously, if two variables are not statistically independent, we say that they are statistically dependent. In that case, information about one of the variables tells us something about the other variable. The dependence between two variables x and y can either be positive or negative: If observing some particular value of x makes it more likely to observe some particular value of y , we say there is a positive dependence. If some particular value of x makes some particular value of y more unlikely, the dependence is negative.

Example 12.3 Dependent coin flips

Let us consider flipping a coin to get a random outcome of either heads or tails. Each of these possible outcomes occur with probability 50%. Let us call the outcome x , and let us define another variable y to be the *opposite* possible outcome. In other words, if x is tails, the y is heads and vice versa. The possible combined outcomes we can see and their probability would then be given by:

x	y	$p(x,y)$
heads	tails	0.5
tails	heads	0.5

Clearly, x and y are not independent, since they are always the opposite of each other. We can verify this by checking the definition of statistical independence, namely that the joint probability is equal to the product of the individual marginal probabilities, $p(x,y) = p(x)p(y)$, for any values of x and y . As in the previous example we have that the

probability of each individual outcome of x and y is 50%

$$p(x = \text{heads}) = p(x = \text{tails}) = 0.5$$

$$p(y = \text{heads}) = p(y = \text{tails}) = 0.5$$

However, the joint probability $p(x = \text{heads}, y = \text{tails}) = 0.5$

$$p(x = \text{heads}, y = \text{tails}) = 0.5$$

$$\neq$$

$$p(x = \text{heads})p(y = \text{tails}) = 0.5 \cdot 0.5 = 0.25$$

This proves that x and y are not independent. In this case there is a negative dependence between observing $x = \text{heads}$ and $y = \text{heads}$, because if we see $x = \text{heads}$ it makes it less likely (impossible actually) to see $y = \text{heads}$.

One particular way that we can measure the degree of dependence between two variables is by their *covariance* and the related *correlation coefficient*. Both of these are measures of the *linear* dependence between two variables². Covariance is defined in a similar way as variance is defined for a single variable. Where the variance is the average squared deviation from the mean, the covariance is the average product of the difference of the mean for the two variables.

Definition 12.2 Covariance and correlation

Covariance

The covariance between two variables is a measure of the linear dependence between two variables. It can be computed as:

$$\text{cov}(x, y) = \frac{1}{N} \sum_{i=1}^N (x_i - \mu_x)(y_i - \mu_y).$$

Correlation coefficient

The correlation coefficient between two variables is their covariance normalized by the product of their standard deviations, and is computed as:

$$\rho_{x,y} = \frac{\text{cov}(x, y)}{\sigma_x \cdot \sigma_y}.$$

The correlation coefficient is a number between -1 and 1.

² Note that it is possible for two variables to have a non-linear dependence while having zero covariance; thus, if the covariance between two variables is zero, it does not necessarily mean that the variables are independent; however, if they are independent the covariance will be zero.

Example 12.4 Covariance and correlation coefficient

Consider two data sets which consist of five numbers each:

$$x = \{1, 8, 4, 10, 2\}, \quad y = \{7, 5, 4, 3, 6\}.$$

Recall from Example ?? that the means and standard deviations of these two data sets are:

$$\mu_x = 5, \quad \mu_y = 5, \quad \sigma_x \approx 3.46, \quad \sigma_y \approx 1.41.$$

Using the means, we can compute the covariance between x and y as

$$\begin{aligned} \text{cov}(x, y) &= \frac{1}{N} \sum_{i=1}^N (x_i - \mu_x)(y_i - \mu_y) \\ &= \frac{1}{5} \left((1-5)(7-5) + (8-5)(5-5) + (4-5)(4-5) \right. \\ &\quad \left. + (10-5)(3-5) + (2-5)(6-5) \right) = -4 \end{aligned}$$

The fact that the covariance is negative means that when x tends to be relatively high (greater than its mean) then y tends to be relatively low (smaller than its mean).

Finally, we can compute the correlation coefficient, which measures the degree of correlation as a number between -1 and 1 :

$$\rho_{x,y} = \frac{\text{cov}(x, y)}{\sigma_x \cdot \sigma_y} \approx \frac{-4}{3.46 \cdot 1.41} \approx -0.82$$

12.2 Correlation is not causation

If two variables are strongly correlated, we might think that there is a causal link between them; however, we should be very careful when we try to explain *why* the correlation occurs, because there could be many equally good explanations and we might not have enough information to determine the correct one.

Causal relation It could be the case, that there is a causal relation between the variables: This would mean that the changes observed in one of the variables is caused by changes in the other variable. But since the measure of correlation is symmetrical, we cannot say which variable is the cause and

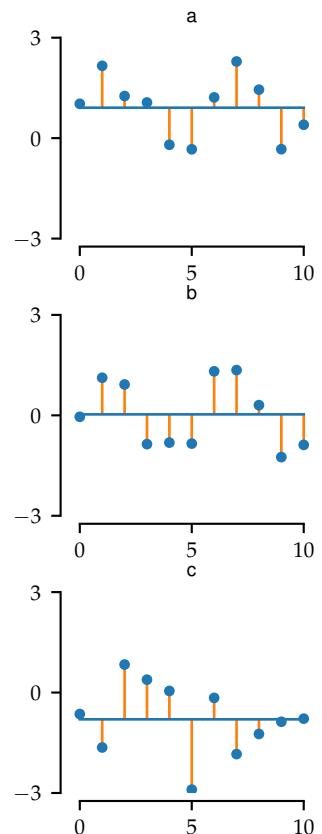


Figure 12.1: Examples of correlations. The plots show the mean and 11 observations of three different variables. The data in the two top panes are correlated: When one variable is above its mean the other also tends to be above its mean and vice versa. The bottom data is uncorrelated with the data in the top two panes.

which is the effect. To determine the direction of the causal link requires further information or assumptions.

Bidirectional relation The cause-effect relation might also go both ways, as for example in a predator-prey relation: As the number of predators increases, the number of prey will decrease (as they are eaten), which in turn leads to fewer predators (as they starve) and so on.

Confounding Another possibility could be that there is some third, unobserved variable, (known as a confounder) that is a common cause of both of our observed variables, and the observed correlation between our variables simply occurs because they are both influenced by this common cause.

Spurious correlation Finally, the observed correlation might simply be a random coincidence, a phenomenon known as *spurious correlation*.

Definition 12.3 Correlation and causation

Correlation is a statistical measure that describes the strength of the linear relationship between two variables.

Causation means that one variable (the effect) is the result of another variable (the cause). If there is a causal relationship between two variables, and we were to manually modify the cause-variable, the effect-variable would be affected as well.

Example 12.5 Income earned and hours worked

Let us say that we have measured two variables “income earned” and “hours worked”, and we have estimated a strong positive correlation (people who work more hours earn more income and vice versa). Is it likely that there is a direct causal effect? Well yes, we could imagine that if we increased the number of hours worked the income would increase as well, for example if we are talking about people working on hourly wages. So the “hours worked” could possibly have a causal effect on “income earned”.

A causal effect the other way around is perhaps not so easy to imagine: If we increase their income, people would

probably not start working more hours.

But before we make any conclusions we should think things through: Imagine a scenario where everybody have a fixed monthly wage and no overtime pay, and imagine that people with a high wage tend to put in more overtime without compensation. In that case “hours worked” and “income earned” would be correlated. But since no-one gets paid for their overtime, changing the number of hours worked will have no causal effect on the income.

The bottom line is that it requires strong assumptions about how the world works to go from an observed correlation to a causal explanation.

12.2.1 Confounding variables

In many situations the direction of the causal relation between two variable is fairly obvious, for example if the cause logically precedes the effect in time or there is some other well established theory about their causal relation. In that case we often refer to the hypothesized cause as the “treatment” and the effect as the “outcome”. But even in that situation we should be very cautious about using a measure of correlation to say something about the strength of the causal relationship between treatment and outcome. One reason is that there could be *confounding* variables that influence the measured correlation by affecting both the treatment and the outcome (see Figure 12.2).

Definition 12.4 Confounding variable

A *confounder* is a variable (often unobserved) that influence both the treatment and the outcome. In the presence of unobserved confounding variables, the observed correlation cannot be taken as evidence for the strength of a hypothesized causal relation.

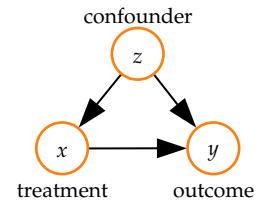


Figure 12.2: Causal diagram:
The treatment is a direct cause of the outcome, but the confounder is a cause of both the treatment and the outcome.

When an unobserved confounder is present, the observed correlation between the treatment and the outcome could be completely different from the causal relation. It might even be the case that there is a positive observed correlation even if the underlying causal relation is negative.

Example 12.6 The best AI teacher

Let us say that a university course in AI runs two times per year. In the spring semester the course is taught by Teacher A, and in the fall it is taught by Teacher B. We now observe the number of students who pass the standardized exam in the two semesters:

- 95 out of 100 students (95%) pass the course with Teacher A in the spring.
- 72 out of 80 students (90%) pass the course with Teacher B in the fall.

We can think of the teacher as the treatment and the proportion of students who pass the exam as the outcome: It seems that Teacher A does a better job than Teacher B, and it would appear that taking the course with Teacher A more likely leads to passing the exam.

Now, assume we asked the teachers about their classes and they gave us the following further information.

- In the spring, 80 of the students were experienced graduate students 78 of whom passed the exam, and the remaining 20 were inexperienced freshmen 17 of whom passed.
- In the fall, 10 of the students were graduate students who all passed and 70 were freshmen of whom 62 passed.

If we examine how well the two types of students do, we might break down the numbers as follows:

	Graduates	Freshmen	Total
Teacher A	78/80≈98%	17/20=85%	95/100=95%
Teacher B	10/10=100%	62/70≈89%	72/80=90%

Now we can see that while Teacher A overall has the best passing rate, the passing rate for Teacher B is better *both for the graduates and the freshmen*.

In this example, the type of student is a confounder. The passing rate with Teacher A in the spring is higher in part because more experienced students take the course in

the spring. While this more detailed analysis might suggest that Teacher B is best after all, there is no guarantee that there are not other unobserved confounders, which might lead us to reverse our conclusion again.

12.2.2 Randomized trials

If we want to interpret an observed correlation as a causal relation, we need to make sure that we take into account all possible confounding variables. In an observational study, where we gather data from some “real life” situation, there is no general way to find out which confounders that are present. If we did have access to measurements of all the confounders, we could in principle adjust for them to estimate the strength of the causal effect. One way to do this would be to divide our observations into subgroups in which the confounder is practically constant, and perform our analysis on each subgroup separately, as we did for the different types of students in Example 12.6. However, this approach relies strongly on the assumption that we have measured all relevant confounders.

Another approach, which completely sidesteps the issue with possible confounders is to conduct a *randomized trial*. The idea in a randomize trial is that the decision about who gets the treatment (or how large a treatment) is completely randomized, which by definition guarantees that there are no possible confounders. Since the decision about the treatment is completely specified by a randomized procedure, there can be no other hidden causes that affect the treatment, and thus there is no confounders. In a randomized trial, it is suddenly easy to estimate the strength of a causal relation, because any observed correlation immediately can be interpreted as causal.

Problems

1. What is the covariance and correlation coefficient between x and y in Example 12.2 concerning the dependent coin flips? Hint: Consider a sample that includes every possible outcome.
2. I have observed that my average speed (in my car) is 62 km/h when I use my summer tires, and 58 km/h when I use my winter tires. If the type of tire is the treatment and the average speed of the car is the outcome, what are possible confounders?
3. What do you think are the most probable explanations for the following observed relations
 - (a) Dancing at parties is correlated with throwing up.
 - (b) Snow is correlated with road accidents.
 - (c) Cheese consumption is correlated with the risk of dying by becoming entangled in bedsheets.
 - (d) Smoking is correlated with cancer.
 - (e) Moderate alcohol consumption is correlated with increased life expectancy.

Solutions

1. $\text{cov}(x, y) = -0.25$ and $\rho_{xy} = -1$.
2. One possible confounder could be the weather (winter tires are used in winter weather, which also might cause me to slow down.)
3. While there is not necessarily any correct answer here, possible explanations could be
 - (a) A common cause could be alcohol consumption.
 - (b) This is most likely direct causal relation.
 - (c) This is probably a spurious correlation.
 - (d) This is most surely a direct causal relation.
 - (e) Who knows — the jury is still out on this one.

Introduction to intelligent systems

Algorithmic fairness

Mikkel N. Schmidt

Technical University of Denmark,
DTU Compute, Department of Applied Mathematics and Computer Science.

Overview

① Fairness

② AI ethics

③ Recap

- Statistics
- Machine learning
- Algorithms
- Optimization
- Data

④ Written exam

⑤ Project work

- Activities in the project period

Feedback group

- Casper Andresen
- Jasmin Lundager Aasbjerg Petersen
- Gabriel Sejr Hornstrup
- Sofus Alexander Kjelgaard Carstens

Learning objectives

- II Fairness criteria: Demographic parity, equalized odds, equal opportunity.
- I Ethical challenges and dilemmas in AI

- I Understand the concepts and definitions, and know their application. Reason about the concepts in the context of an example. Use correct technical terminology.
- II As above plus: Read, manipulate, and work with technical definitions and expressions (mathematical and Python code). Carry out practical computations. Interpret and evaluate results.

Fairness

Setting and notation

We restrict ourselves to a binary decision process setting, with two protected groups

$G \in \{a, b\}$ Protected group (such as race, gender, religion etc.)

X Attributes used in decision making

$\hat{S} = f(X) \in \mathbb{R}$ Score function

$\hat{Y} \in \{0, 1\}$ Binary decision found by thresholding \hat{S}

$Y \in \{0, 1\}$ Correct decision (might not be available)

Binary decision making

- Score function

$$\hat{S} = f(X)$$

E.g. determined by machine learning to predict a measure of success.

- Binary decision

$$\hat{Y} = \begin{cases} 1 & \text{if } \hat{S} > \tau, \\ 0 & \text{otherwise.} \end{cases}$$

Decision matrix

Decision matrix

		$G = a$		$G = b$	
		$\hat{Y} = 0$	$\hat{Y} = 1$	$\hat{Y} = 0$	$\hat{Y} = 1$
$Y = 0$	a_{00}	a_{01}	b_{00}	b_{01}	
	a_{10}	a_{11}	b_{10}	b_{11}	

- Our decision \hat{Y} is known.
- Outcome Y sometimes only known for $\hat{Y} = 1$.
- Outcome Y for $\hat{Y} = 0$ can be a counterfactual.

Demographic parity

Decide $\hat{Y} = 1$ and $\hat{Y} = 0$ in the same fraction of cases in each protected group

$$P(\hat{Y} = 1 | G = a) = P(\hat{Y} = 1 | G = b)$$

		$G = a$		$G = b$	
		$\hat{Y} = 0$	$\hat{Y} = 1$	$\hat{Y} = 0$	$\hat{Y} = 1$
$Y = 0$	a_{00}	a_{01}	b_{00}	b_{01}	
$Y = 1$	a_{10}	a_{11}	b_{10}	b_{11}	

Selection of competitors for the math competition

- A school participates in a math competition and can run with 12 students.
- The school has students from two protected groups
 - a : (200 students) and
 - b : (100 students).

We have access to the students' assessment marks \hat{S} , which we believe is a reasonable predictor of success.

How can we select students in a fair manner according to the demographic parity criterion?

Selection of competitors for the math competition

- A school participates in a math competition and can run with 12 students.
- The school has students from two protected groups
 - a : (200 students) and
 - b : (100 students).

We have access to the students' assessment marks \hat{S} , which we believe is a reasonable predictor of success.

How can we select students in a fair manner according to the demographic parity criterion?

Solution

We can select $\frac{200}{200+100} \cdot 12 = 8$ from group a and the remaining 4 from group b , ranked by their assessment marks.

Equalized odds

Probability of correct decision equal in the protected groups

$$P(\hat{Y} = 0 | Y = 0, G = a) = P(\hat{Y} = 0 | Y = 0, G = b)$$

$$P(\hat{Y} = 1 | Y = 1, G = a) = P(\hat{Y} = 1 | Y = 1, G = b)$$

		$G = a$		$G = b$	
		$\hat{Y} = 0$	$\hat{Y} = 1$	$\hat{Y} = 0$	$\hat{Y} = 1$
$Y = 0$	$\hat{Y} = 0$	a_{00}	a_{01}	b_{00}	b_{01}
	$\hat{Y} = 1$	a_{10}	a_{11}	b_{10}	b_{11}

Selection of competitors for the math competition II

After the math competition, the math problems are released and all students have a go. It turns out that students in group b were a lot better on average, and that the assessment marks were not a good predictor of success.

		$G = a$		$G = b$	
		$\hat{Y} = 0$	$\hat{Y} = 1$	$\hat{Y} = 0$	$\hat{Y} = 1$
$Y = 0$	96	4	24	1	
$Y = 1$	96	4	72	3	

With this new data in mind, was the school's selection fair according to the equalized odds criterion?

Selection of competitors for the math competition II

After the math competition, the math problems are released and all students have a go. It turns out that students in group b were a lot better on average, and that the assessment marks were not a good predictor of success.

		$G = a$		$G = b$	
		$\hat{Y} = 0$	$\hat{Y} = 1$	$\hat{Y} = 0$	$\hat{Y} = 1$
$Y = 0$	96	4	24	1	
$Y = 1$	96	4	72	3	

With this new data in mind, was the school's selection fair according to the equalized odds criterion?

Solution

$$P(\hat{Y} = 0 | Y = 0, G = a) = P(\hat{Y} = 0 | Y = 0, G = b)$$

$$\frac{96}{96+4} = \frac{24}{24+5} = 96\%$$

$$P(\hat{Y} = 1 | Y = 1, G = a) = P(\hat{Y} = 1 | Y = 1, G = b)$$

$$\frac{4}{96+4} = \frac{3}{72+3} = 4\%$$

Yes, this is fair according to equalized odds.

Equalized opportunity

Probability that decision is correct among positive decisions equal in the protected groups

$$P(Y = 1 | \hat{Y} = 1, G = a) = P(Y = 1 | \hat{Y} = 1, G = b)$$

		$G = a$		$G = b$	
		$\hat{Y} = 0$	$\hat{Y} = 1$	$\hat{Y} = 0$	$\hat{Y} = 1$
$Y = 0$	a_{00}	a_{01}	b_{00}	b_{01}	
$Y = 1$	a_{10}	a_{11}	b_{10}	b_{11}	

Selection of competitors for the math competition III

		$G = a$		$G = b$	
		$\hat{Y} = 0$	$\hat{Y} = 1$	$\hat{Y} = 0$	$\hat{Y} = 1$
$Y = 0$	96	4	24	1	
$Y = 1$	96	4	72	3	

Was the school's selection fair according to the equalized opportunity criterion?

Selection of competitors for the math competition III

		$G = a$		$G = b$	
		$\hat{Y} = 0$	$\hat{Y} = 1$	$\hat{Y} = 0$	$\hat{Y} = 1$
$Y = 0$	96	4	24	1	
	96	4	72	3	

Was the school's selection fair according to the equalized opportunity criterion?

Solution

$$P(Y = 1 | \hat{Y} = 1, G = a) = P(Y = 1 | \hat{Y} = 1, G = b)$$

$$\frac{4}{4+4} = 50\% = \frac{3}{1+3} = 75\% \text{ False}$$

No, this does not appear to be fair according to equalized opportunity. Caveat: The sample size is very small, so the difference is not statistically significant.

AI ethics

Ethical challenges in AI

- Increased reliance on AI
- Explainable AI / right to explanation
- Bias and fairness in AI systems
- Behaviour manipulation
- Human-robot interaction
- Autonomous systems
- AI faking technologies
- Automation and employment
- Privacy and surveillance
- Strong AI, super intelligence, robot rights

Increased reliance on AI systems

Boeing anti-stall patch MCAS (Maneuvering Characteristics Augmentation System).

- As an automated corrective measure, the MCAS was given full authority to bring the aircraft nose down, and could not be overridden by pilot resistance against the control wheel.
- Pilots were unaware of the existence of MCAS due to its omission from the crew manual and no coverage in training.¹
- In October 2018 and March 2019, Boeing 737 MAX passenger jets crashed just after takeoff with nearly 350 killed.
- All 737 MAX planes were afterwards grounded worldwide.

¹ Wikipedia, https://en.wikipedia.org/wiki/Maneuvering_Characteristics_Augmentation_System

Increased reliance on AI systems

Boeing anti-stall patch MCAS (Maneuvering Characteristics Augmentation System).

- As an automated corrective measure, the MCAS was given full authority to bring the aircraft nose down, and could not be overridden by pilot resistance against the control wheel.
- Pilots were unaware of the existence of MCAS due to its omission from the crew manual and no coverage in training.¹
- In October 2018 and March 2019, Boeing 737 MAX passenger jets crashed just after takeoff with nearly 350 killed.
- All 737 MAX planes were afterwards grounded worldwide.

“[MCAS] was designed per our standards, certified per our standards, and we’re confident in that process. So, it operated according to those design and certification standards. So, we haven’t seen a technical slip or gap in terms of the fundamental design and certification of the approach.” (Boeing CEO Dennis Muilenburg 2019)

¹ Wikipedia, https://en.wikipedia.org/wiki/Maneuvering_Characteristics_Augmentation_System

Questions

- In what daily situations do you rely on automation and AI?
- Is it a problem if humans rely on computer decisions without understanding how they operate or draw conclusions?
- Should humans be made aware of all AI systems they come in contact with? And should they understand how they work?

Dilemma

You use a chatbot to make phone calls to order a service on your behalf. The service provider never finds out that they talked to a bot. Is that okay?

Yes As long as everything works for everybody

No It has to be clear if you talk to a human or a robot

Explainable AI

- At ML Symposium, NIPS 2017, Facebook chief AI Scientist Yann LeCun suggests that a models reasoning can be inferred from observing how it acts, such that:
 - Rigorous testing is enough to provide an explanation.
- Cassie Kozyrkov (Google Intelligence Engineer) asks the question: “Imagine choosing between two spaceships.
 - Spaceship 1 comes with exact equations explaining how it works, but has never been flown.
 - How Spaceship 2 flies is a mystery, but it has undergone extensive testing, with years of successful flights like the one you’re going on.

Which spaceship would you choose?”

Questions

- If the model is predictable and thoroughly tested, do you think there is still a need for an explanation?
- What in cases were obtaining testdata for all cases is infeasible or costly (healthcare, self-driving cars)?
- Do you hold other humans to the same standards?

Dilemma

Use of AI can lead to faster decisions within the public sector, such as building permissions and early retirement benefits.

Good idea It makes processing times faster.

Bad idea Cases should be processed by humans.

Right to explanation

Generally in the EU, consumers have a legal *right to explanation* if a decision is based solely on automated processing (including profiling). Profiling can be the basis for decision making in certain situations (such as fraud and tax-evasion monitoring), or when the data subject has given his or her explicit consent.

- Do you think this is a fair principle?
- Should “right to explanation” limit what AI models we can use for decision making?
- Should this also be the case when humans make the decisions?

Bias in AI systems

- In 2014 Amazon developed a machine learning system to review job applicants' resumes to automatically extract the top talents.
- The model was trained to score applicants based on patterns observed in resumes the company had received for over a decade.
- Most of the training data were applications from men, reflecting the demographics of the tech industry.

² Reuters, Oct 11, 2018, <https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G>

Bias in AI systems

- In 2014 Amazon developed a machine learning system to review job applicants' resumes to automatically extract the top talents.
- The model was trained to score applicants based on patterns observed in resumes the company had received for over a decade.
- Most of the training data were applications from men, reflecting the demographics of the tech industry.

“In effect, Amazon’s system taught itself that male candidates were preferable. It penalized resumes that included the word women’s, as in women’s chess club captain. And it downgraded graduates of two all-women’s colleges [...] Amazon edited the programs to make them neutral to these particular terms. But that was no guarantee that the machines would not devise other ways of sorting candidates that could prove discriminatory, the people said.”²

² Reuters, Oct 11, 2018, <https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G>

Questions

AI systems learn from training data, which can reflect biased human decisions or historical constructs, even when sensitive variables are removed.

- How should we address this?
- Is it possible to agree on when a decision is fair?

Behaviour manipulation

- On March 23, 2016, Microsoft launched *Tay*, an artificial intelligence chatbot on Twitter, developed to conduct research on conversational understanding.
- Tay was designed to mimic the language of a young adult learning by interacting with human Twitter users.
- Users could follow and interact with the bot on Twitter and it would tweet back, learning conversation from other users' posts.
- Just 16 hours after its launch it was shut down

Behaviour manipulation

- On March 23, 2016, Microsoft launched *Tay*, an artificial intelligence chatbot on Twitter, developed to conduct research on conversational understanding.
- Tay was designed to mimic the language of a young adult learning by interacting with human Twitter users.
- Users could follow and interact with the bot on Twitter and it would tweet back, learning conversation from other users' posts.
- Just 16 hours after its launch it was shut down

“As many of you know by now, on Wednesday we launched a chatbot called Tay. We are deeply sorry for the unintended offensive and hurtful tweets from Tay, which do not represent who we are or what we stand for, nor how we designed Tay. Tay is now offline and we'll look to bring Tay back only when we are confident we can better anticipate malicious intent that conflicts with our principles and values.” (Mar 25, 2016, Peter Lee, Corporate Vice President, Microsoft Healthcare)

Questions

- Who is responsible for Tay's behaviour?

Questions

- Who is responsible for Tay's behaviour?

Just as the AI's behaviour can be manipulated, so can humans. Manipulating online behaviour is a core internet business model, which includes exploitation of behaviour, addiction and deception.

- Sale of addictive goods and services (alcohol, tobacco, gambling) are highly regulated, should this be the same for technology that exploit or manipulate behaviour?
- The Hippocratic oath historically taken by physicians states "I will abstain from all intentional wrong-doing and harm". Do we have similar ethical obligations as AI engineers?

Human-Robot interactions

- “A California hospital delivered end-of-life news to a 78-year-old patient via a robotic machine this week, prompting the man’s family to go public with their frustration.”³

³ Headline, USAtoday 2018

⁴ Aalborg University, <https://www.kommunikation.aau.dk/forskning/vidensgrupper/e-learning-lab/nyhedsliste/nyhed/demente-liver-op-i-interaktion-med-menneskelignende-robotter.cid365882>

Human-Robot interactions

- “A California hospital delivered end-of-life news to a 78-year-old patient via a robotic machine this week, prompting the man’s family to go public with their frustration.”³
- People with severe dementia benefit from contact with social robots: It makes them come alive and socially active. But there is a risk that the robots may be used to passify troublesome patients.⁴

³ Headline, USAtoday 2018

⁴ Aalborg University, <https://www.kommunikation.aau.dk/forskning/vidensgrupper/e-learning-lab/nyhedsliste/nyhed/demente-liver-op-i-interaktion-med-menneskelignende-robotter.cid365882>

Dilemma

Is it okay that a nursing home uses robot pets to interact with residents with dementia, who do not know it is a robot?

Yes It brings joy and increases life quality

No Residents need to know if they interact with robots

Recap

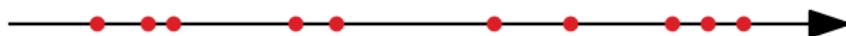
Overview

Statistics	■ Population and sample ■ Standard error and confidence intervals ■ Sample size calculation ■ Correlation and causality ■ Experimental design ■ Fairness criteria	Optimiz.	■ Cost function and parameter estimation ■ Gradient descent ■ Automatic differentiation
ML	■ Unsupervised, supervised, and reinforcement learning ■ Training and test error ■ Feature transformation: Scaling and basis change	Data	■ Image data: Color spaces ■ Audio data: Spectrogram ■ Text data: Bag of words
Algorithms	■ Algorithmic complexity ■ K-means clustering ■ Least squares regression ■ Neural networks ■ TF-IDF / Okapi BM25 ■ Value iteration / q-learning		

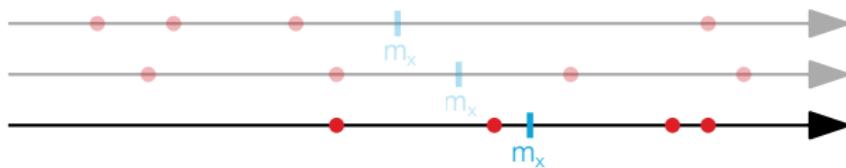
Recap: Statistics

Population and sample

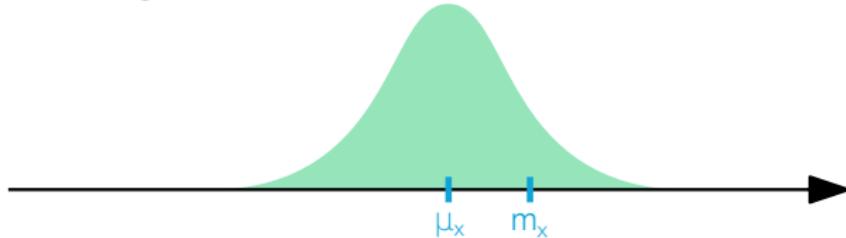
Population



Possible samples of size $n=4$



Sampling distribution of the mean



Standard error and confidence intervals

Confidence interval

point estimate \pm margin of error

$$\bar{x} \pm \underbrace{z_{\alpha/2}}_{\text{critical value}} \cdot \underbrace{\sqrt{\frac{\sigma^2}{n}}}_{\text{standard error}}$$

We want to estimate the population mean μ

- Sample n observations
- Compute the sample mean $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$
- Choose confidence level, e.g. $1 - \alpha = 95\%$, and look up critical value
- Compute standard error and multiply by critical value

Sample size calculation

- The equation for the confidence interval can be solved for the sample size
- This gives a formula for the required sample size to give a desired margin of error

Sample size for proportion

$$n = z_{\alpha/2}^2 \frac{p(1-p)}{E^2}$$

E : Desired margin of error

α : Significance level

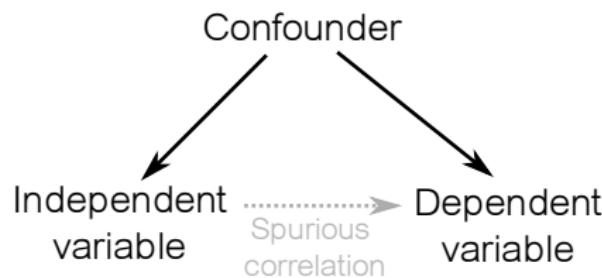
p : True proportion

Correlation and causality

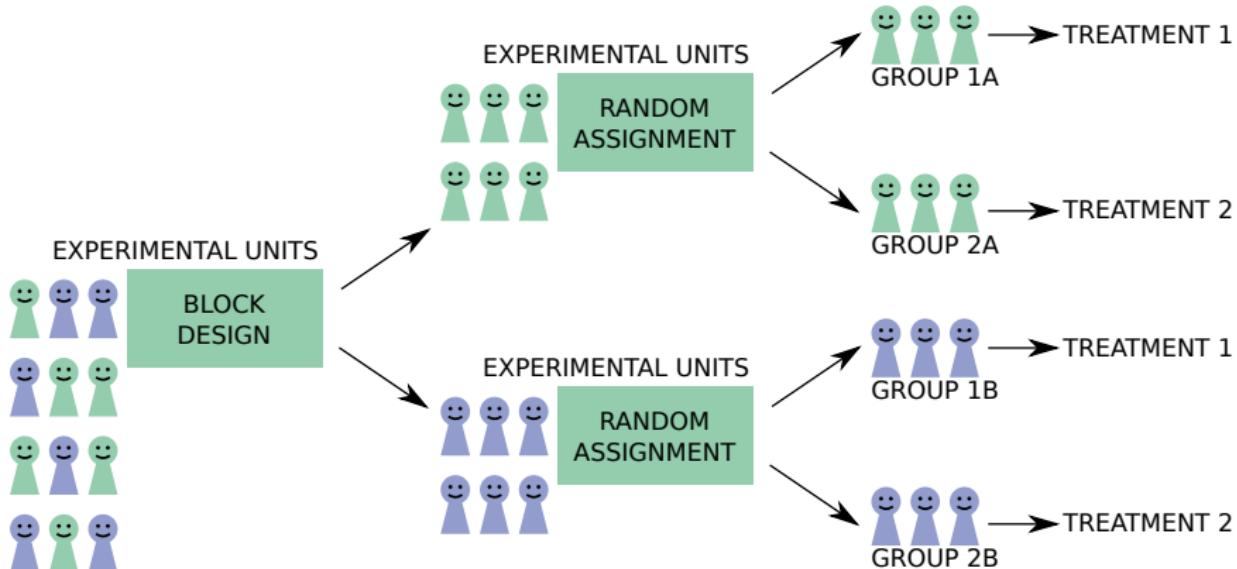
Pearson correlation coefficient

$$\rho_{X,Y} = \frac{\text{cov}(X, Y)}{\sigma_X \cdot \sigma_Y}$$

Causality



Experimental design



- Blocking eliminates one confounding variable
- Other confounders will be randomly distributed between treatment groups

Recap: Machine learning

Unsupervised, supervised, and reinforcement learning

Categorization of learning problems

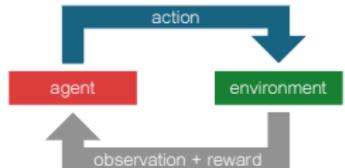
Unsupervised Learn function that describes the structure in data



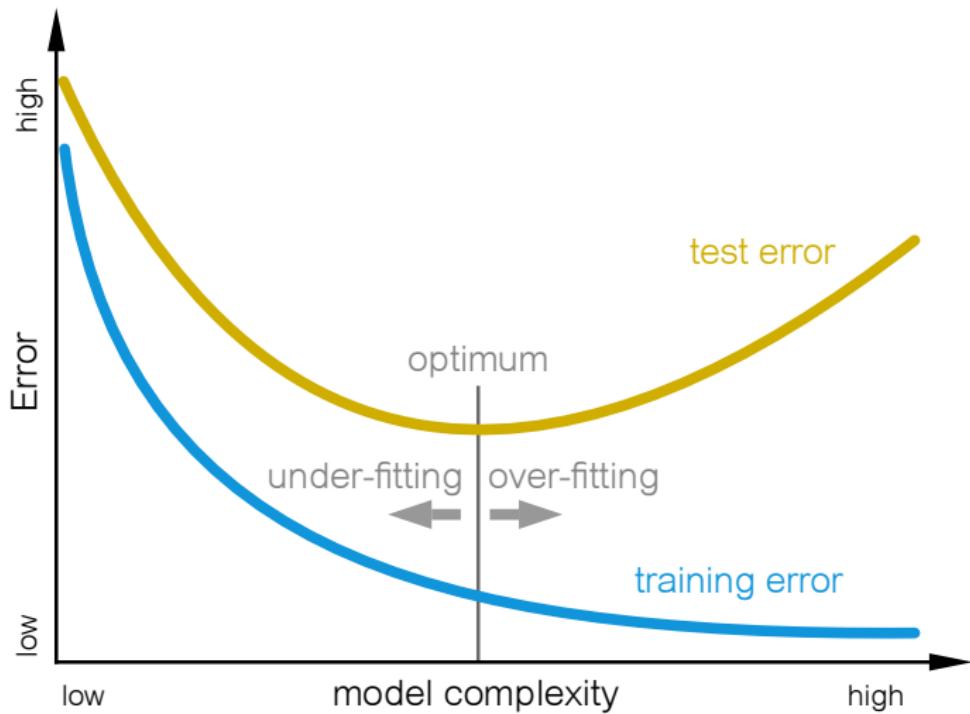
Supervised Learn function that maps input to output to optimize cost



Reinforcement Learn a function (policy) that maps inputs to actions to optimize cumulative reward



Training and test error



Feature transformation: Scaling and basis change

Min-max normalization

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Standardization

$$x' = \frac{x - \bar{x}}{\sigma_x}$$

\bar{x} Mean

σ_x standard deviation

Basis change

$$\mathbf{y} = \underbrace{\mathbf{T}^{-1} \mathbf{x}}_{\text{matrix multiplication}}$$

\mathbf{x} Vector in the original coordinate system

\mathbf{T} Matrix where each column is a basis vector of the new coordinate system expressed in the old coordinate system

\mathbf{y} Vector expressed in the new coordinate system

Recap: Algorithms

Algorithmic complexity

- Classify algorithms according to their performance
- Time function $T(n)$ measures runtime
- Big-O notation expresses *runtime complexity*
- Considers only the highest order term of $T(n)$
- Upper bound on growth rate

Formal definition

$T(n) \in O(f(n))$ iff there exists a constant c such that $T(n) < cf(n)$ for all $n > n_0$

We say $f(n)$ is an asymptotic upper bound for $T(n)$

K-means clustering

Objective

$$\underbrace{\min_{\{\mu_1, \dots, \mu_K\}}}_{\text{Cluster means}} \underbrace{\min_{\{c_1, \dots, c_N\}}}_{\text{Cluster assignments}} \underbrace{\sum_{n=1}^N \|x_n - \mu_{c_n}\|^2}_{\text{Squared distance to cluster mean}}$$

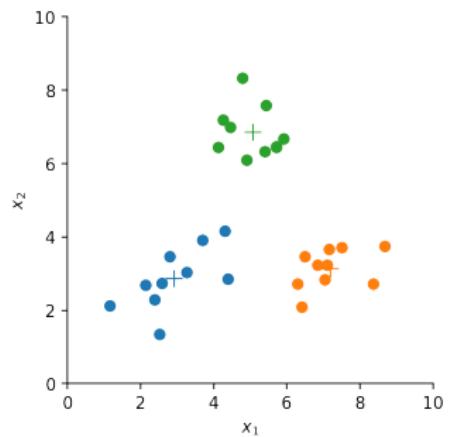
Algorithm

1. Fix cluster means, optimize cluster assignment

$$\min_{\{c_1, \dots, c_N\}} \sum_{n=1}^N \|x_n - \mu_{c_n}\|^2$$

2. Fix cluster assignments, optimize cluster means

$$\min_{\{\mu_1, \dots, \mu_K\}} \sum_{n=1}^N \|x_n - \mu_{c_n}\|^2$$

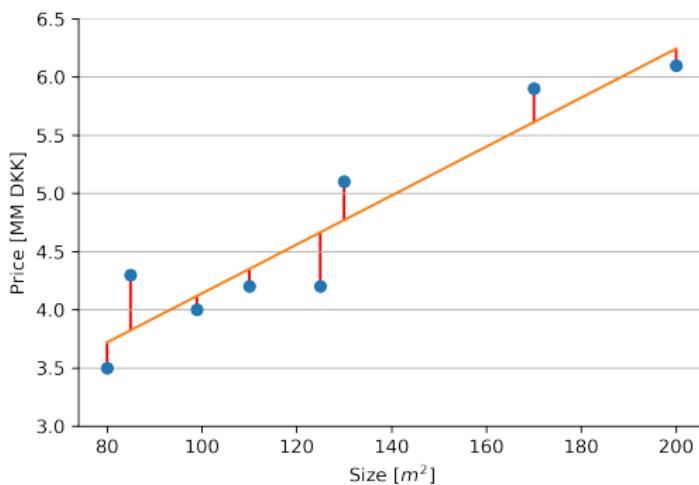


Least squares regression

- Regression line: $f(x) = ax + b$
- Error: Squared distance between data and regression line

$$E = \sum_{n=1}^N (y_n - f(x_n))^2$$

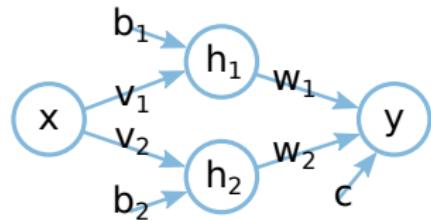
- Find values of a and b to minimize E



Neural networks

Cost function

$$E = \sum_{n=1}^N (y(n) - \hat{y}(n))^2$$



Network structure

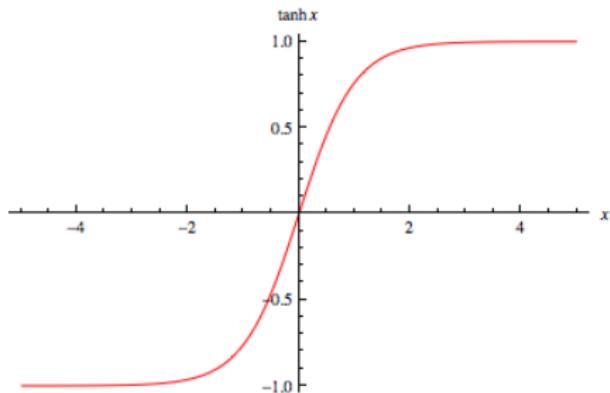
$$\hat{y}(n) = w_1 h_1(n) + w_2 h_2(n) + c$$

$$h_1(n) = \tanh(v_1 x(n) + b_1)$$

$$h_2(n) = \tanh(v_2 x(n) + b_2)$$

Model parameters

$$c, w_1, w_2, v_1, v_2, b_1, b_2$$



Okapi BM25

$$\text{BM25}(d, q) = \sum_{t \in q} \frac{n_{t,d} \cdot (k_1 + 1)}{n_{t,d} + k_1 \cdot (1 - b + b \cdot \frac{n_d}{\text{avgdl}})} \cdot \log \left(\frac{N - n_t + 0.5}{n_t + 0.5} \right)$$

$n_{t,d}$ Number of occurrences of term t in document d

n_d Number of terms in document d

n_t Number of documents with term t

N Total number of documents

avgdl Average document length $\frac{1}{N} \sum_d n_d$

b Parameter ($b \in [0, 1]$, default $b = 0.75$)

k_1 Parameter ($k_1 > 0$, default $k_1 = 1.2$)

Value iteration and Q-learning

Value iteration

- Loop through all states and update according to

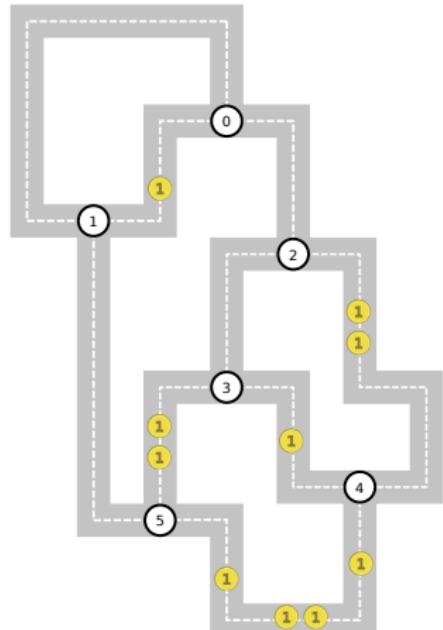
$$v(s) = \max_a (r(s, a) + \gamma v(s'))$$

- Repeat until convergence

Q-learning

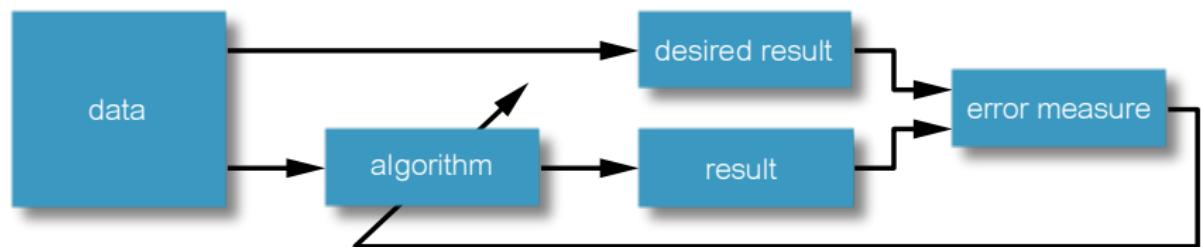
- Explore the environment according to some policy that ensures visiting all state-action pair
- At each step, update the quality function according to

$$q(s, a) = r(s, a) + \gamma \max_{a'} q(s', a')$$

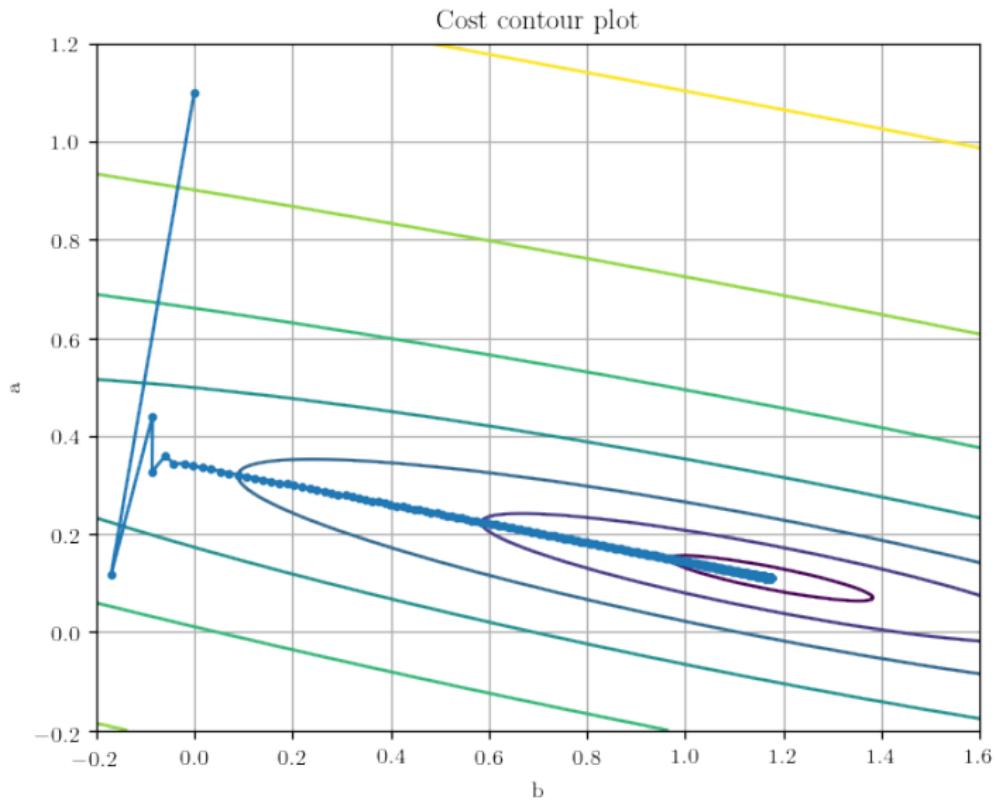


Recap: Optimization

Cost function and parameter estimation



Gradient descent

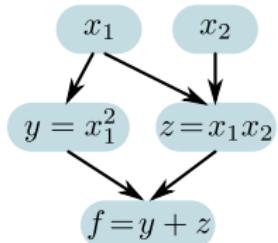


Automatic differentiation

Function and derivatives

$$f(x_1, x_2) = x_1^2 + x_1 \cdot x_2$$

$$\frac{\partial f}{\partial x_1} = 2x_1 + x_2, \quad \frac{\partial f}{\partial x_2} = x_1, \quad \nabla f(3, 4) = \begin{bmatrix} 10 \\ 3 \end{bmatrix}$$



Evaluate $f(3, 4)$

$$x_1 = 3$$

$$x_2 = 4$$

$$y = x_1^2$$

$$z = x_1 \cdot x_2$$

$$f = y + z$$

$$\bar{f} = 1$$

$$\bar{y} = \bar{y} + \bar{f} \frac{\partial f}{\partial y} = \bar{y} + \bar{f} \cdot 1 = 0 + 1 \cdot 1 = 1$$

$$\bar{z} = \bar{z} + \bar{f} \frac{\partial f}{\partial z} = \bar{z} + \bar{f} \cdot 1 = 0 + 1 \cdot 1 = 1$$

$$\bar{x}_1 = \bar{x}_1 + \bar{y} \frac{\partial y}{\partial x_1} = \bar{x}_1 + \bar{y} \cdot 2 \cdot x_1 = 0 + 1 \cdot 2 \cdot 3 = 6$$

$$\bar{x}_1 = \bar{x}_1 + \bar{z} \frac{\partial z}{\partial x_1} = x_1 + \bar{z} \cdot x_2 = 6 + 1 \cdot 4 = 10$$

$$\bar{x}_2 = \bar{x}_2 + \bar{z} \frac{\partial z}{\partial x_2} = x_2 + \bar{z} \cdot x_1 = 0 + 1 \cdot 3 = 3$$

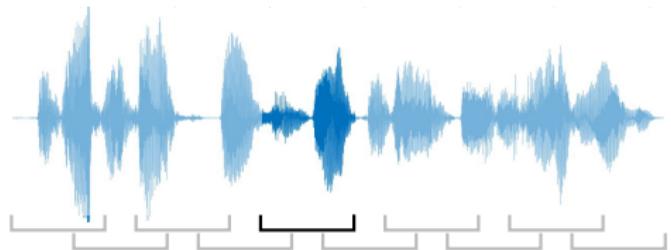
Recap: Data

Image data: Color spaces

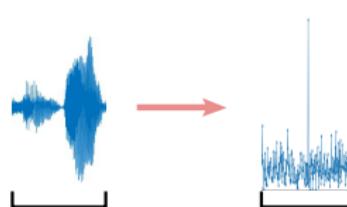


Audio data: Spectrogram

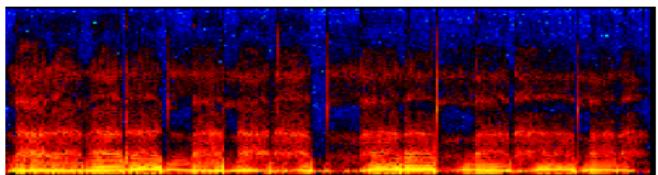
Split the signal
into blocks



For each block,
compute the spectrum



Gather spectra as columns
in matrix and plot heat map



Text data: Bag of words

	Doc. 1	Doc. 2
african	1	0
although	0	1
and	1	0
are	1	0
bear	0	1
black	1	0
by	1	0
close	0	1
coat	1	0
distinct	1	0
equid	1	0
famili	1	0
giraff	0	1
hors	1	0
is	0	1
it	0	1
mark	0	1
most	0	1
of	1	1
okapi	0	1
relat	0	1
reminisc	0	1
sever	1	0
speci	1	0
stripe	1	1
the	0	2
their	1	0
to	0	1
unit	1	0
white	1	0
zebra	1	1

Sentences

1. Zebras are several species of African equids (horse family) united by their distinctive black and white striped coats.
2. Although the okapi bears striped markings reminiscent of zebras it is most closely related to the giraffe.

- A bag-of-words sentence/document can be seen as a point in a high-dimensional vector space

Written exam

Individual written examination

Written exam in December

- Topics covered in lecture slides and notes (see “list of topics”)
- 2 hours
- 5 exercises with 4 questions each
- Hand in digitally (pdf file)

Final grade

- Written exam (weight 40%)
- Individualized group report (weight 60%)

List of topics

 *Show list of topics*

Project work

Project work

- Groups of 3 students is strongly preferred
- Design and conduct an experiment
- Write and hand in group report.
 - 5-pages.
 - Individualized (who did what?)

Week 1

Come up with idea The first step is to come up with an idea for your experiment—something you would like to study. You can formulate this as a research question, and state your expectations and tentative explanation as a hypothesis.

Design experiment Next, you must design your experiment. Agree on a protocol to follow and write it down. Ideally, the experiment should be designed so carefully that others would be able to reproduce the experiment and get the same result except for statistical variation.

Decide on your approach and start building At this time, you should also agree on which method to use to analyze the data, and consider how large a sample size you need to support your claims. You should think about what code you need to write, or existing toolboxes you need to familiarize yourself with, to be able to conduct your experiment. Start prototyping your experimental setup.

Deliverables

- Description of experiment idea
- Experimental plan

Week 2

Build your experimental platform At this time, you should put together the computer code etc. you need for your experiment. Test that everything works as required and finalize your setup. Perhaps it could be a good idea to run a small pilot study to make sure everything is in place.

Carry out your experiment Now, you are ready to carry out your experiment. Follow your protocol carefully and record the results. If something goes wrong or you realize there is an issue with your experimental design, you might need to go back to the previous step and modify your design.

Start writing At this time, you should also start to draft your report.

Deliverables

- Experimental data
- Report draft

Week 3

Document and communicate your experiment Finally, you must write up a report, following the template we have given you. The report must be approximately 5 pages long, and you must hand it in as a pdf file. The report must clearly describe the experimental design and protocol, include visualization and summaries of the data gathered in the experiment, present the results of the experiment, and include a discussion where you comment on perspectives, ethics etc.

Deliverables

- Final report

Experiment

- The experiment must be centered around a problem where artificial intelligence or machine learning is relevant
- The experiment can be based on the methods and computer code discussed in the course, including
 - Image classification
 - Symbolic AI
 - Linear / neural network regression
 - K-means clustering
 - Text search using Okapi
 - Audio classification
 - Tabular value iteration / q-learning
- You must gather your own experimental data

Report

- Approximately 5 pages
- Follow the IMRaD format
- Include a short abstract
- Include appropriate visualization of the dataset
- Include statistical considerations regarding sample size
- Back up claims regarding recognition up by statistics
- Include your own code as an appendix
- Write using LaTeX

Assessment criteria

In addition to the requirements above, the degree of fulfillment of following objectives will be taken into account in the assessment of the project:

- Description of key components of intelligent systems: Sensing and active data collection, machine learning, evaluations and communication
- Application of AI tools to data (such as image, audio, text and games)
- Discussion and analysis of performance
- Application of visualization techniques for evaluation of performance and basic debugging
- Application of scientific Python programming tools
- Discussion of the role of AI tools in the chosen application domain
- Discussion safety and ethical challenges in AI, biases and stereotypes, privacy and societal impact

Project work: Activities in the project period

Project period

Assistance Our teaching assistants will be available many days throughout the period. They can help with almost everything, and you can book individual consultations with them.

Consultations Groups can book individual consultations with the teacher in 20 minute time slots.

Feedback sessions Two times during the period we will meet 4–5 groups together with the teacher to present project status and give each other feedback.

Recap lectures Depending on your demand, we will give recap lectures in any topic from the course or beyond.

Project idea brain storm

Discuss in groups

Phase I: Generate ideas

- Present and write down very briefly all your ideas, and come up with more
- Criticism is not allowed

Phase II: Review, select, combine, and improve

- Lead a discussion to identify the most interesting ideas and try to combine or improve the ideas
- Agree on two ideas
 1. The most impactful and world changing idea.
 2. The most absurdly hilarious idea.

Be prepared to present to the class.

Report your project groups to us

- When you decide on your idea and form a group, please self-sign-up your group on DTU Learn
- Please use the discussion forum “Group formation and idea exchange” on DTU Learn
 - If you have an idea, but miss a few group members
 - If you lack ideas and group members
- It would be great if groups are formed before we start in January.

Project topic ideas

Visual learning

- Image recognition in variable lightning conditions / with obstruction
- Recognition of rotated objects
- Face recognition / facial expression recognition

Audio learning

- Classify presence or absence of music
- Noise level vs subjective noise level
- Tone / instrument classification

Text modeling

- Okapi search performance evaluation (e.g. in parliament documents or Wikipedia)
- Sentiment scoring compared with subjective evaluation (e.g parliament opening speeches or RSS news feed),

Other modalities

- EEG analysis
- Cardiac monitoring

Project examples

Emotional image analysis	Image classification with convolutional neural network and transfer learning
Generating hand-drawn circles	Image generation using generative adversarial network
Arousal in danish news media	Sentiment analysis of news articles using the AFINN lexicon
Learn to play "Game 2048"	Deep reinforcement learning using policy gradient methods
Super-resolution imaging	Image up-scaling using generative adversarial network
Decoding mental states in EEG	Binary classification using support vector machine
Recognition of hand-written digits	Image classification using convolutional neural network
Face recognition	How many training images are needed for face recognition using convolutional neural network
Learn to play "Snake"	Reinforcement learning using tabular Q-learning
Colorizing black/white images	Comparison of convolutional neural network and U-net on image colorization task
Counting objects in images	Exploration of multi-task convolutional neural network for counting objects in images
Most readable background color	Comparison of nearest neighbor, logistic regression, and neural network for classifying text readability

Thank you

See you in January