Algorithms for Machine Learning

Lecture 1a: Introduction

Thomas Sauerwald

University of Cambridge, Department of Computer Science and Technology
email: tsauerwa@gmail.com

3 Dec 2020

Outline

Lecturer Intro

Brief Bio:

- Diploma ($\approx$ MSc) in Math, then switched to Computer Science for PhD
- Postdocs at Berkeley, Vancouver and Max Planck Institute for Informatics
- Reader in Computer Science at Cambridge (joined 2013)
- ERC Starting Grant 2015 on Stochastic Processes and Randomised Algorithms

Lecturer Intro

Brief Bio:

- Diploma ($\approx$ MSc) in Math, then switched to Computer Science for PhD
- Postdocs at Berkeley, Vancouver and Max Planck Institute for Informatics
- Reader in Computer Science at Cambridge (joined 2013)
- ERC Starting Grant 2015 on Stochastic Processes and Randomised Algorithms

Teaching:

- several courses related to Algorithms, Data Science, Graph Theory, Machine Learning and Probability

Outline

Short Bio

Highlights (and Challenges) in AI/ML

What is Machine Learning and Artificial Intelligence?

Content of this Programme

Garry Kasparov faced off against Deep Blue, IBM's chess-playing computer in 1997. Deep Blue was able to imagine an average of 200,000,000 positions per second. Kasparov ended up losing the match. (AP Photo/Adam Nadel)

Adam Nadel/AP

1996: Gary Kasparov-Deep Blue 4:2

Garry Kasparov faced off against Deep Blue, IBM's chess-playing computer in 1997. Deep Blue was able to imagine an average of 200,000,000 positions per second. Kasparov ended up losing the match. (AP Photo/Adam Nadel)

Adam Nadel/AP

1996: Gary Kasparov-Deep Blue 4:2
1997: Gary Kasparov-Deep Blue 2,5:3,5

AlphaGo

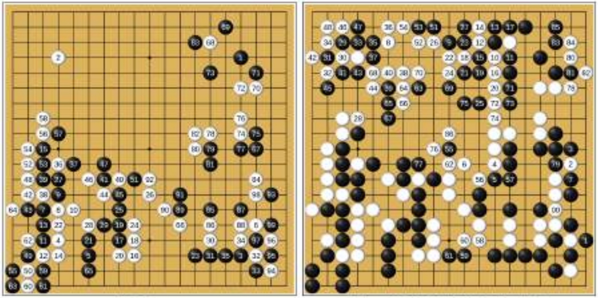
Source: Wikipedia

2016: Lee Sedol-Alpha Go 1:4

AlphaGo

Example game [edit]

AlphaGo Master (white) v. Tang Weixing (31 December 2016), AlphaGo won by resignation. White 36 was widely praised.



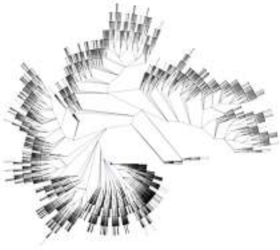
First 99 moves

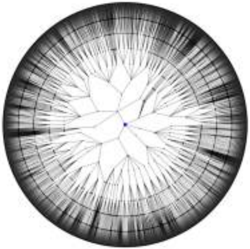
Moves 100–186 (149 at 131, 150 at 130)

Source: Wikipedia

2016: Lee Sedol-Alpha Go 1:4
2019: Lee Sedol retired from Professional Go

From Chess to Go: Evolution of Tree Search





Source: `https://www.cs.cornell.edu/courses/cs6700/2016sp/lectures/CS6700-UCT.pdf`

UCT Tree Search

- selective and asymmetric search
- best performing method for Go

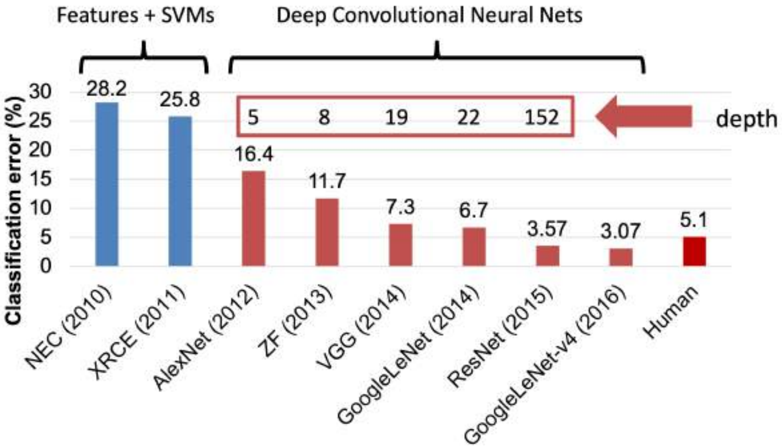
Minimax Search

- full-width tree up to some depth
- (used to be) best method for Chess

Source: Google

Figure 5: Example of frontal upright face images used for training.

ImageNet Large Scale Visual Recognition Challenge



Source: `https://cs.uwaterloo.ca/~ppoupart/teaching/cs885-spring18/slides/cs885-lecture4a.pdf`

Source: Japan Times


Who Is Responsible In A Crash With A Self-Driving Car?

Elizabeth Fernandez

Science

On March 18, 2018, at nearly 10 PM, a self-driving Volvo hit and killed a pedestrian, a woman named Elaine Herzberg. Herzberg's death was the first pedestrian fatality involving a self-driving car. The self-driving car was a test vehicle, a car that Uber was testing in Arizona. It could not figure out if the woman was a pedestrian, a bicycle, or another car, nor predict where she was going. Video showed that the driver of the self-driving car, acting as a "safety backup", was not looking at the road at the time of the collision. Instead, she was watching an episode of "The Voice".

Sources: The Economist and Forbes

https://www.nbcnews.com/tech/tech-news/self-driving-uber-car-hit-killed-woman-did-not-recognize-n1079281

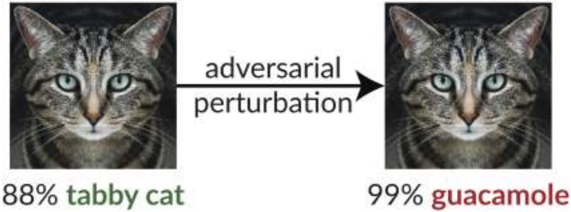
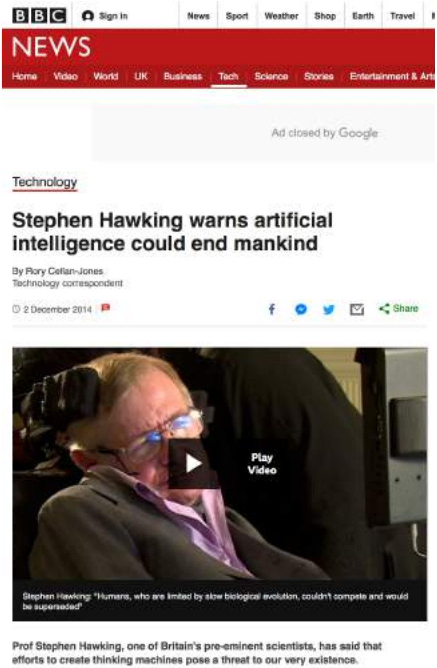


Fig. 1. A small change imperceptible to humans misleads the InceptionV3 network into classifying an image of a tabby cat as guacamole. Image taken from https://github.com/anishathalye/obfuscated-gradients.

A Simple Explanation for the Existence of Adversarial Examples with Small Hamming Distance

Adi Shamir[1], Itay Safran[1], Eyal Ronen[2], and Orr Dunkelman[3]

[1] Computer Science Department, The Weizmann Institute, Rehovot, Israel
[2] Computer Science Department, Tel Aviv University, Tel Aviv, Israel
[3] Computer Science Department, University of Haifa, Israel



BBC Sign in News Sport Weather Shop Earth Travel

NEWS

Home Video World UK Business Tech Science Stories Entertainment & Arts

Ad closed by Google

Technology

Stephen Hawking warns artificial intelligence could end mankind

By Rory Cellan-Jones
Technology correspondent

2 December 2014

Share

Play Video

Stephen Hawking: "Humans, who are limited by slow biological evolution, couldn't compete and would be superseded"

Prof Stephen Hawking, one of Britain's pre-eminent scientists, has said that efforts to create thinking machines pose a threat to our very existence.

Source: BBC News

Outline

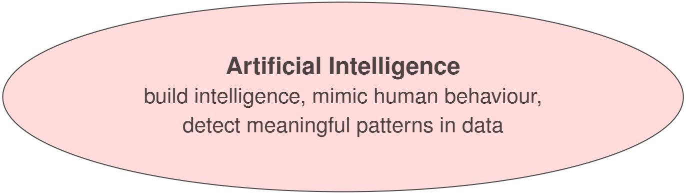
Short Bio

Highlights (and Challenges) in AI/ML

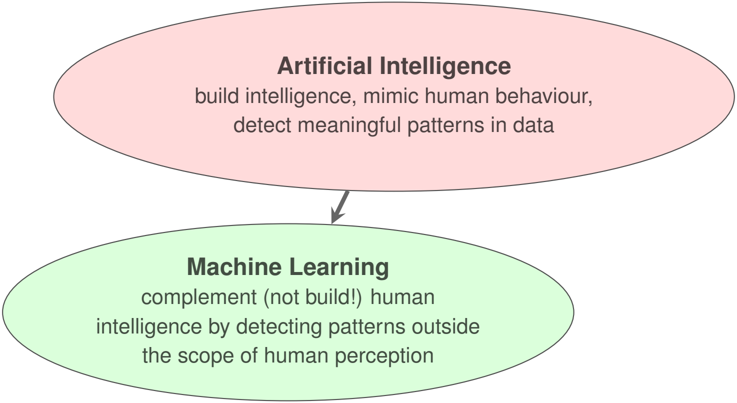
What is Machine Learning and Artificial Intelligence?

Content of this Programme

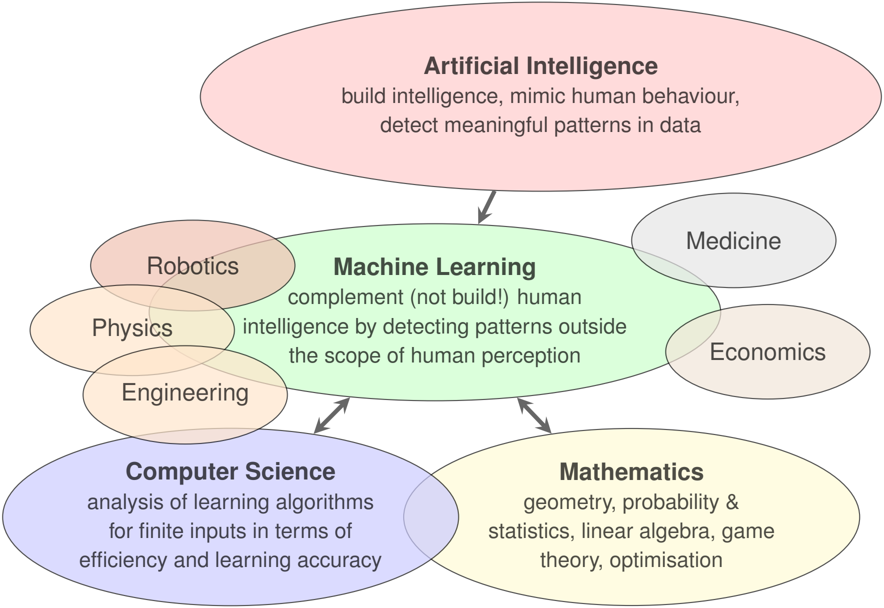
Machine Learning and Artificial Intelligence



Machine Learning and Artificial Intelligence

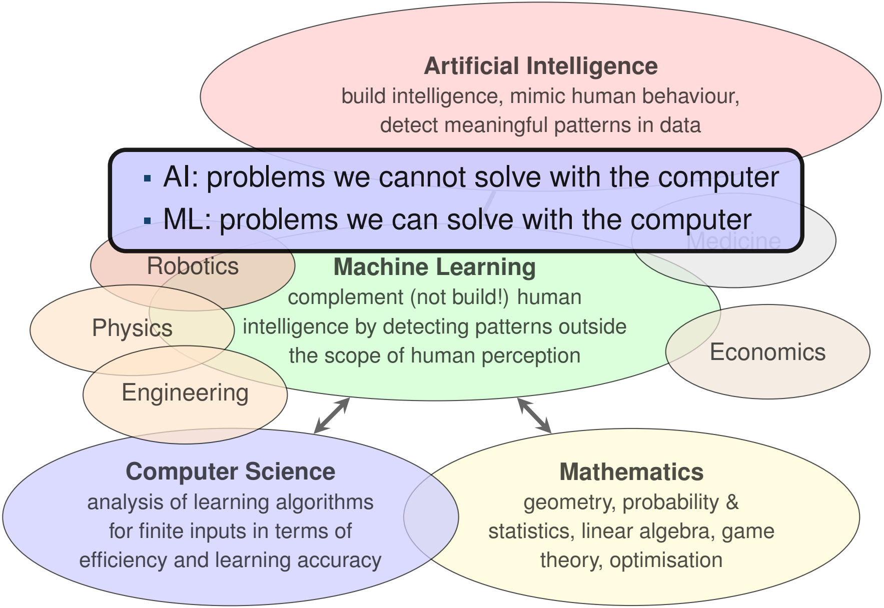


Machine Learning and Artificial Intelligence



Machine Learning and Artificial Intelligence

- AI: problems we cannot solve with the computer
- ML: problems we can solve with the computer



Motivation

Data is everywhere:

- we generate it continuously
- big data is freely available
- storage is cheap

Machine Learning is everywhere:

- spam filtering
- text and speech recognition
- medicine and healthcare
- self-driving cars
- recommender system
- photo search

Motivation

Data is everywhere:

- we generate it continuously
- big data is freely available
- storage is cheap

Machine Learning is everywhere:

- spam filtering
- text and speech recognition
- medicine and healthcare
- self-driving cars
- recommender system
- photo search

We have access to huge computational power.
We have new sophisticated (and parallel!) machine learning algorithms.

Motivation

Data is everywhere:

- we generate it continuously
- big data is freely available
- storage is cheap

Machine Learning is everywhere:

- spam filtering
- text and speech recognition
- medicine and healthcare
- self-driving cars
- recommender system
- photo search

We have access to huge computational power.
We have new sophisticated (and parallel!) machine learning algorithms.

Analyse large amounts of data so that learning algorithms can:

- spot patterns and understand better behaviour of entity (**descriptive** learning);

Motivation

Data is everywhere:

- we generate it continuously
- big data is freely available
- storage is cheap

facebook

Machine Learning is everywhere:

- spam filtering
- text and speech recognition
- medicine and healthcare
- self-driving cars
- recommender system
- photo search

Discover hidden trends in data, Extract clusters in a network, Segment image into pieces

ver.

We ... hing algorithms.

Analyse large amounts of data so that learning algorithms can:

- spot patterns and understand better behaviour of entity (**descriptive** learning);

Motivation

Data is everywhere:

- we generate it continuously
- big data is freely available
- storage is cheap

Machine Learning is everywhere:

- spam filtering
- text and speech recognition
- medicine and healthcare
- self-driving cars
- recommender system
- photo search

Discover hidden trends in data, Extract clusters in a network, Segment image into pieces

ver.

We hing algorithms.

Analyse large amounts of data so that learning algorithms can:

- spot patterns and understand better behaviour of entity (**descriptive** learning);
- learn behaviour of entity to predict future behaviour (or unknown feature) (**predictive** learning).

Motivation

Data is everywhere:

- we generate it continuously
- big data is freely available
- storage is cheap

Machine Learning is everywhere:

- spam filtering
- text and speech recognition
- medicine and healthcare
- self-driving cars
- recommender system
- photo search

Discover hidden trends in data, Extract clusters in a network, Segment image into pieces

ver.

We hing algorithms.

Analyse large amounts of data so that learning algorithms can:

- spot patterns and understand better behaviour of entity (**descriptive** learning);
- learn behaviour of entity to predict future behaviour (or unknown feature) (**predictive** learning).

Patient X does not have diabetes, this image is a face, the weather tomorrow is sunny, the expected grade of student Y is 8.5, ...

Landscape of Machine Learning Algorithms

Training Set provided initially

Supervised Learning

Nearest Neighbour, Regression, Logistic Regression, Decision Trees and Random Forests, Perceptron, Naive Bayes, Boosting, Support Vector Machines, Neural Networks

Predict unseen data

Landscape of Machine Learning Algorithms

Training Set provided initially

No Training Set

Supervised Learning

Nearest Neighbour, Regression, Logistic Regression, Decision Trees and Random Forests, Perceptron, Naive Bayes, Boosting, Support Vector Machines, Neural Networks

Unsupervised Learning

Density Estimation (Maximum Likelihood), Feature Extraction: Dimensionality Reduction, Principal Component Analysis, Singular Value Decomposition, Clustering

Predict unseen data

Extract Knowledge

Landscape of Machine Learning Algorithms

Training Set provided initially

Feedback after Decisions

No Training Set

Supervised Learning

Nearest Neighbour, Regression, Logistic Regression, Decision Trees and Random Forests, Perceptron, Naive Bayes, Boosting, Support Vector Machines, Neural Networks

Online/Reinforcement Learning

Online Perceptron, Weighted-Majority, Markov Chains, Hidden-Markov Models, Markov Decision Processes: Temporal Difference, Q-Learning, SARSA

Unsupervised Learning

Density Estimation (Maximum Likelihood), Feature Extraction: Dimensionality Reduction, Principal Component Analysis, Singular Value Decomposition, Clustering

Predict unseen data

Maximise Reward

Extract Knowledge

Machine Learning: A New Perspective of Computing

Traditional Computing:

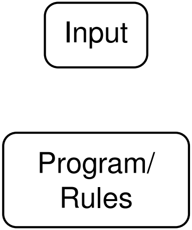
Machine Learning: A New Perspective of Computing

Input

Traditional Computing:

- User specifies input (e.g., data)

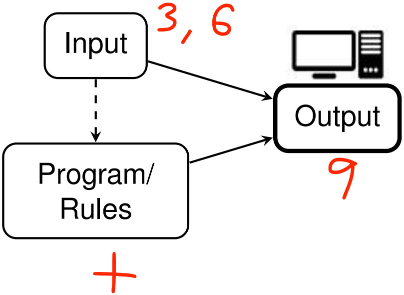
Machine Learning: A New Perspective of Computing



Traditional Computing:

- User specifies input (e.g., data) and set of instructions (e.g., program)

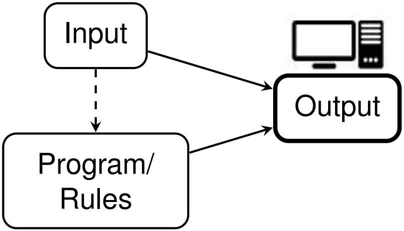
Machine Learning: A New Perspective of Computing



Traditional Computing:

- User specifies input (e.g., data) and set of instructions (e.g., program)
- Computer calculates output

Machine Learning: A New Perspective of Computing

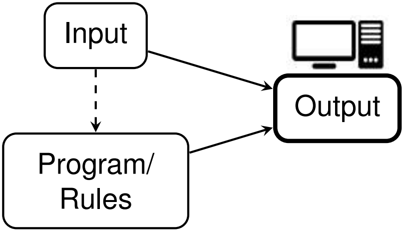


Traditional Computing:

- User specifies input (e.g., data) and set of instructions (e.g., program)
- Computer calculates output

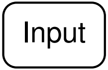
Machine Learning:

Machine Learning: A New Perspective of Computing



Traditional Computing:

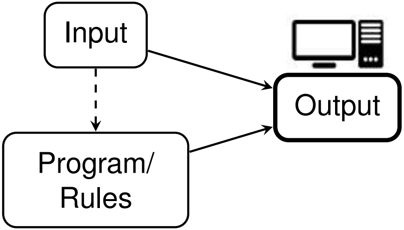
- User specifies input (e.g., data) and set of instructions (e.g., program)
- Computer calculates output



Machine Learning:

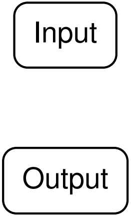
- User specifies input (e.g., data)

Machine Learning: A New Perspective of Computing



Traditional Computing:

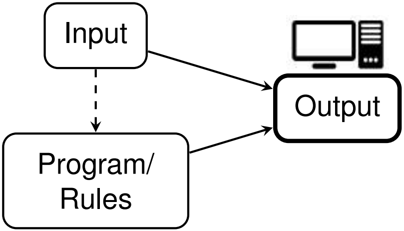
- User specifies input (e.g., data) and set of instructions (e.g., program)
- Computer calculates output



Machine Learning:

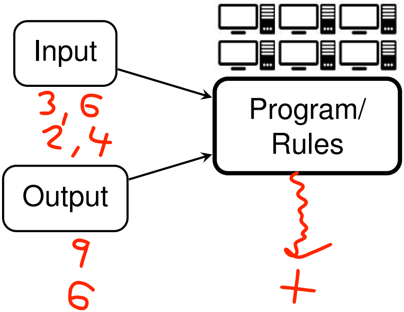
- User specifies input (e.g., data) and output (e.g., classifications)

Machine Learning: A New Perspective of Computing



Traditional Computing:

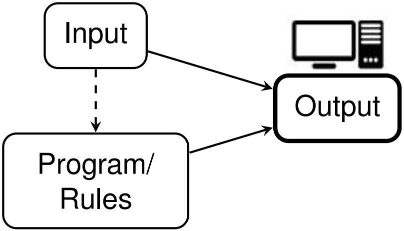
- User specifies input (e.g., data) and set of instructions (e.g., program)
- Computer calculates output



Machine Learning:

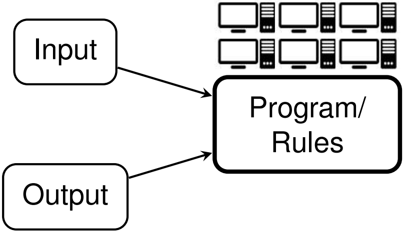
- User specifies input (e.g., data) and output (e.g., classifications)
- Computer calculates a rule/model

Machine Learning: A New Perspective of Computing



Traditional Computing:

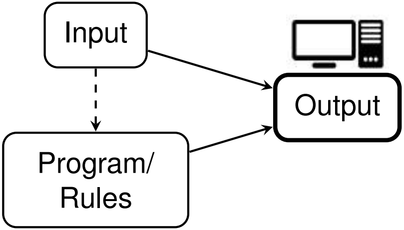
- User specifies input (e.g., data) and set of instructions (e.g., program)
- Computer calculates output



Machine Learning:

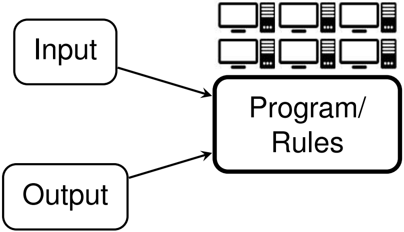
- User specifies input (e.g., data) and output (e.g., classifications)
- Computer calculates a rule/model which should be:

Machine Learning: A New Perspective of Computing



Traditional Computing:

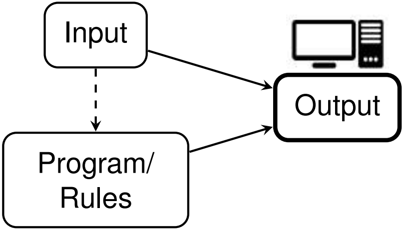
- User specifies input (e.g., data) and set of instructions (e.g., program)
- Computer calculates output



Machine Learning:

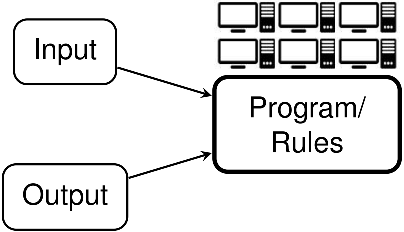
- User specifies input (e.g., data) and output (e.g., classifications)
- Computer calculates a rule/model which should be:
  - logical
  - probable
  - simple (interpretable)
  - efficient

Machine Learning: A New Perspective of Computing



Traditional Computing:

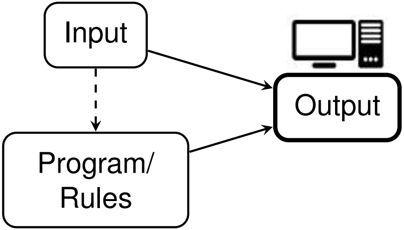
- User specifies input (e.g., data) and set of instructions (e.g., program)
- Computer calculates output



Machine Learning:

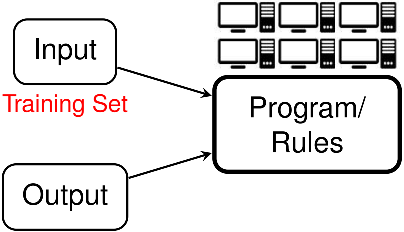
- User specifies input (e.g., data) and output (e.g., classifications)
- Computer calculates a rule/model which should be:
  - logical
  - probable
  - simple (interpretable)
  - efficient
  - accurate

Machine Learning: A New Perspective of Computing



Traditional Computing:

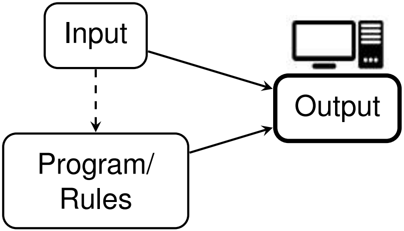
- User specifies input (e.g., data) and set of instructions (e.g., program)
- Computer calculates output



Machine Learning:

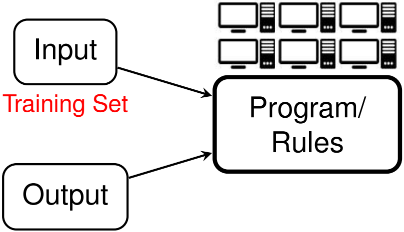
- User specifies input (e.g., data) and output (e.g., classifications)
- Computer calculates a rule/model which should be:
  - logical
  - probable
  - simple (interpretable)
  - efficient
  - accurate (Training Set)

Machine Learning: A New Perspective of Computing



Traditional Computing:

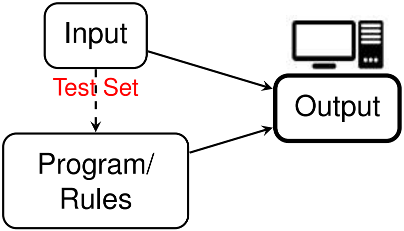
- User specifies input (e.g., data) and set of instructions (e.g., program)
- Computer calculates output



Machine Learning:

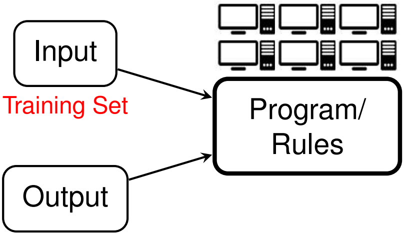
- User specifies input (e.g., data) and output (e.g., classifications)
- Computer calculates a rule/model which should be:
  - logical
  - probable
  - simple (interpretable)
  - efficient
  - accurate (Training Set)
  - generalisable

Machine Learning: A New Perspective of Computing



Traditional Computing:

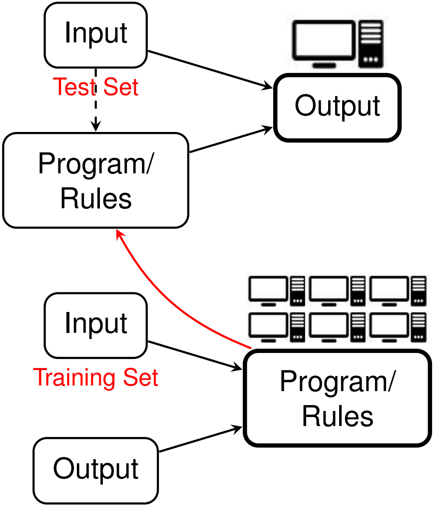
- User specifies input (e.g., data) and set of instructions (e.g., program)
- Computer calculates output



Machine Learning:

- User specifies input (e.g., data) and output (e.g., classifications)
- Computer calculates a rule/model which should be:
  - logical
  - probable
  - simple (interpretable)
  - efficient
  - accurate (Training Set)
  - generalisable (Test Set)

Machine Learning: A New Perspective of Computing



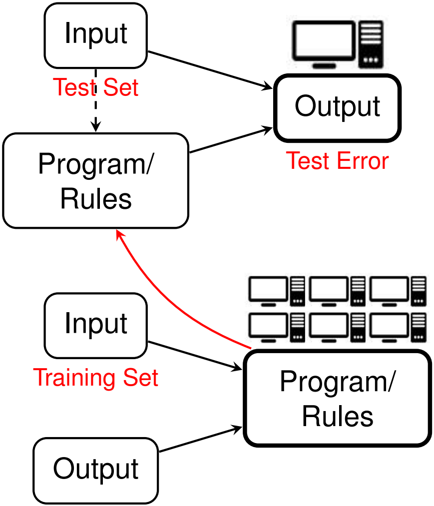
Traditional Computing:

- User specifies input (e.g., data) and set of instructions (e.g., program)
- Computer calculates output

Machine Learning:

- User specifies input (e.g., data) and output (e.g., classifications)
- Computer calculates a rule/model which should be:
  - logical
  - probable
  - simple (interpretable)
  - efficient
  - accurate (Training Set)
  - generalisable (Test Set)

Machine Learning: A New Perspective of Computing



Traditional Computing:

- User specifies input (e.g., data) and set of instructions (e.g., program)
- Computer calculates output

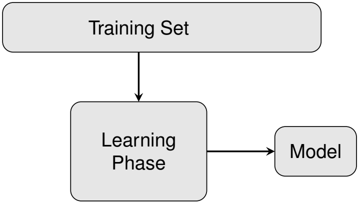
Machine Learning:

- User specifies input (e.g., data) and output (e.g., classifications)
- Computer calculates a rule/model which should be:
  - logical
  - probable
  - simple (interpretable)
  - efficient
  - accurate (Training Set)
  - generalisable (Test Set)

Supervised Learning Model

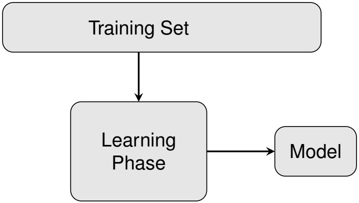
- Training data: used to build a model or learn a classifier for the data.

Supervised Learning Model



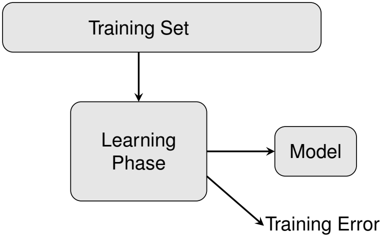
- Training data: used to build a model or learn a classifier for the data.

Supervised Learning Model



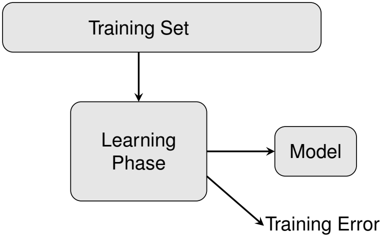
- Training data: used to build a model or learn a classifier for the data.
  - training error: number of wrong classifications by model on training data

Supervised Learning Model



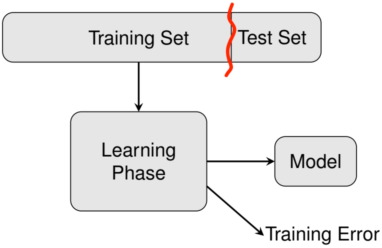
- Training data: used to build a model or learn a classifier for the data.
  - training error: number of wrong classifications by model on training data

Supervised Learning Model



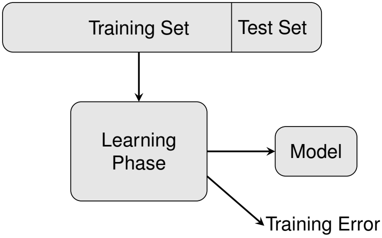
- Training data: used to build a model or learn a classifier for the data.
  - training error: number of wrong classifications by model on training data
- Test data: helps validating it and prevent overfitting.

Supervised Learning Model



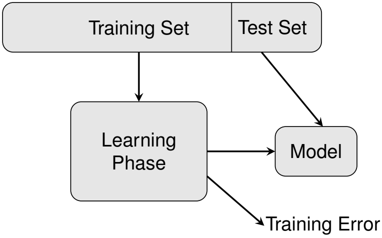
- Training data: used to build a model or learn a classifier for the data.
  - training error: number of wrong classifications by model on training data
- Test data: helps validating it and prevent overfitting.

Supervised Learning Model



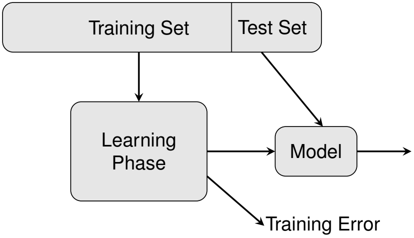
- Training data: used to build a model or learn a classifier for the data.
  - training error: number of wrong classifications by model on training data
- Test data: helps validating it and prevent overfitting.
  - test error: number of wrong classifications by model on test data

Supervised Learning Model



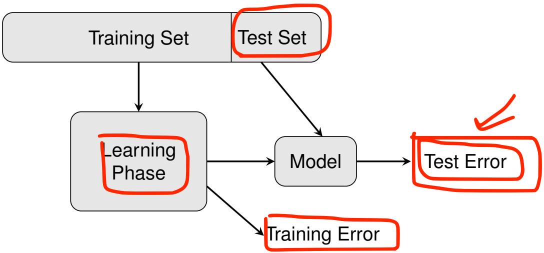
- Training data: used to build a model or learn a classifier for the data.
  - training error: number of wrong classifications by model on training data
- Test data: helps validating it and prevent overfitting.
  - test error: number of wrong classifications by model on test data

Supervised Learning Model



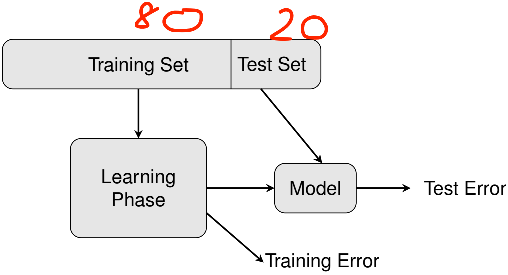
- Training data: used to build a model or learn a classifier for the data.
  - training error: number of wrong classifications by model on training data
- Test data: helps validating it and prevent overfitting.
  - test error: number of wrong classifications by model on test data

Supervised Learning Model



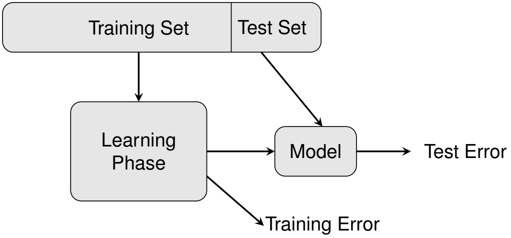
- Training data: used to build a model or learn a classifier for the data.
  - training error: number of wrong classifications by model on training data
- Test data: helps validating it and prevent overfitting.
  - test error: number of wrong classifications by model on test data

Supervised Learning Model



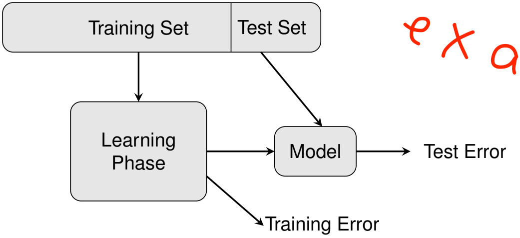
- Training data: used to build a model or learn a classifier for the data.
  - training error: number of wrong classifications by model on training data
- Test data: helps validating it and prevent overfitting.
  - test error: number of wrong classifications by model on test data
- Overfitting:
  - ML algorithms tune their model to the training set and potential noise (and not the general data that the training set represents)
  - rule-of-thumb: split data randomly into 80% training set, 20% test set

Supervised Learning Model



- Training data: used to build a model or learn a classifier for the data.
  - training error: number of wrong classifications by model on training data
- Test data: helps validating it and prevent overfitting.
  - test error: number of wrong classifications by model on test data
- Overfitting:
  - ML algorithms tune their model to the training set and potential noise (and not the general data that the training set represents)
  - rule-of-thumb: split data randomly into 80% training set, 20% test set
- Cross-Validation:

Supervised Learning Model



examples!

- Training data: used to build a model or learn a classifier for the data.
  - training error: number of wrong classifications by model on training data
- Test data: helps validating it and prevent overfitting.
  - test error: number of wrong classifications by model on test data
- Overfitting:
  - ML algorithms tune their model to the training set and potential noise (and not the general data that the training set represents)
  - rule-of-thumb: split data randomly into 80% training set, 20% test set
- Cross-Validation:
  - If data-set small, take average over multiple divisions into training and test set

Outline

Short Bio

Highlights (and Challenges) in AI/ML

What is Machine Learning and Artificial Intelligence?

Content of this Programme

Programme Overview

Introduction / Supervised Learning Nearest-Neighbour Algorithm	Lecture 1 (rec)
Supervised Learning Linear Regression and Polynomial Regression	Lecture 2
Supervised Learning Gradient Descent and Logistic Regression	Lecture 3
Reinforcement Learning Markov Chains	Lecture 4
Reinforcement Learning Hidden Markov Models	Lecture 5
Supervised Learning Decision Trees	Lecture 6
Supervised Learning Perceptron	Lecture 7

Programme Overview

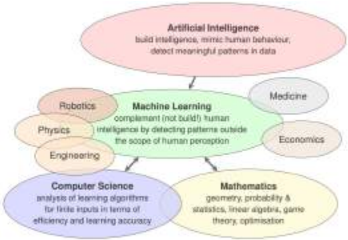
Introduction / Supervised Learning Nearest-Neighbour Algorithm	Lecture 1 (rec)
Supervised Learning Linear Regression and Polynomial Regression	Lecture 2
Supervised Learning Gradient Descent and Logistic Regression	Lecture 3
Reinforcement Learning Markov Chains	Lecture 4
Reinforcement Learning Hidden Markov Models	Lecture 5
Supervised Learning Decision Trees	Lecture 6
Supervised Learning Perceptron	Lecture 7

And we have 3 Supervisions!

0. **Introduction**

Course Introduction and Overview

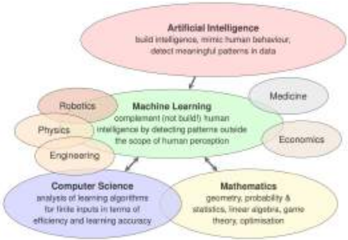
- Recent Developments in ML
- Landscape of ML vs. AI
- Supervised, Unsupervised and Reinforcement Learning
- Course Structure



0. **Introduction**

   Course Introduction and Overview

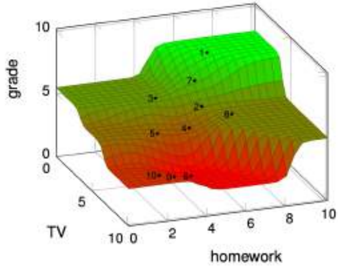
   - Recent Developments in ML
   - Landscape of ML vs. AI
   - Supervised, Unsupervised and Reinforcement Learning
   - Course Structure



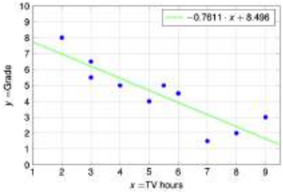
1. **Nearest Neighbour Algorithm**

   Our First ML Algorithm

   - Nearest Neighbour for Classification
   - Number of Neighbours and Distance Function
   - Nearest Neighbour for Regression



Lecture 2

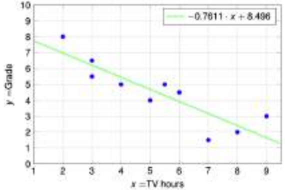


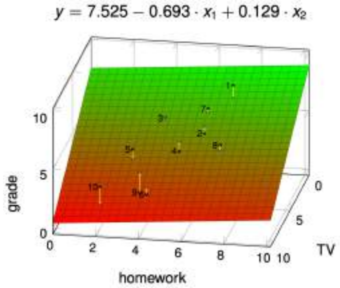
2. **Linear Regression and Beyond**
   Predicting Numerical Values

   - Running Example: Grade Prediction
   - Linear Regression vs. Nearest Neighbour
   - Goodness of Fit $\rightsquigarrow$ R^2-value
   - Explicit Solution in One Dimension

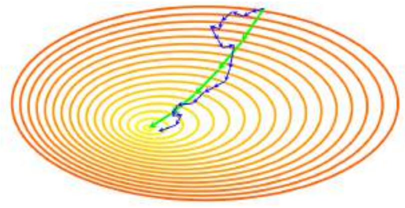
2. **Linear Regression and Beyond**
Predicting Numerical Values

- Running Example: Grade Prediction
- Linear Regression vs. Nearest Neighbour
- Goodness of Fit $\rightsquigarrow$ R^2-value
- Explicit Solution in One Dimension
- Linear Regression in Higher Dimensions
- Outlook on Polynomial Regression
- How to Deal with Categorical Features





Lecture 3

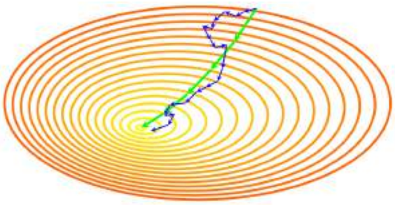


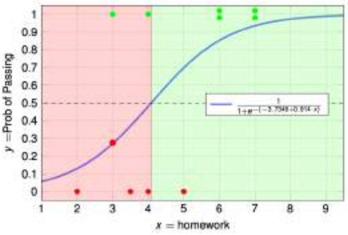
3. **Gradient Descent and Logistic Regression**
   Predicting Probabilities and Classes
   - How to solve General Regression with Gradient Descent
   - Stochastic Gradient Descent

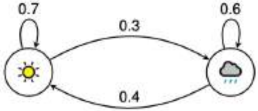
3. **Gradient Descent and Logistic Regression**

   Predicting Probabilities and Classes

   - How to solve General Regression with Gradient Descent
   - Stochastic Gradient Descent
   - Prediction with Logistic Regression
   - Example on Grade Prediction



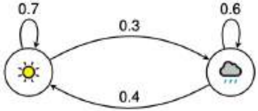




	Day 1	Day 2	Day 3
	1.0	0.7	0.61
	0.0	0.3	0.39

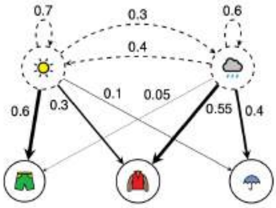
4. **Markov Chains**
   Modelling Random Processes
   - Applications of Markov Chains
   - Transition Matrix
   - Computing Transition Probabilities
   - Stationary Distribution

4. **Markov Chains**
   Modelling Random Processes
   - Applications of Markov Chains
   - Transition Matrix
   - Computing Transition Probabilities
   - Stationary Distribution
   - Examples: Weather Prediction and Website-ranking using PageRank



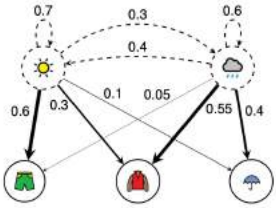
	Day 1	Day 2	Day 3
	1.0	0.7	0.61
	0.0	0.3	0.39

Lecture 5



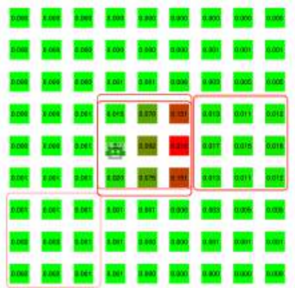
5. **Hidden Markov Models**
   Analysing Temporal Data

   - Gentle Start: inferring weather through observations
   - Maximum Likelihood Principle for Parameter Estimation
   - What is a Hidden Markov Model?



5. **Hidden Markov Models**
   Analysing Temporal Data

   - Gentle Start: inferring weather through observations
   - Maximum Likelihood Principle for Parameter Estimation
   - What is a Hidden Markov Model?
   - Common Tasks in Hidden Markov Models
   - Demo: Robot Localisation using Hidden Markov Models

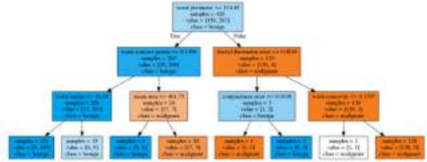


Lecture 6

6. **Decision Trees**

   Tree-Structured Prediction

   - Gentle Start: Decision Stumps in Image Classification
   - How to Build Decision Trees?
   - How to Classify?

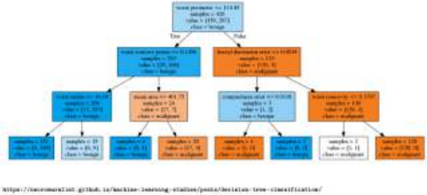


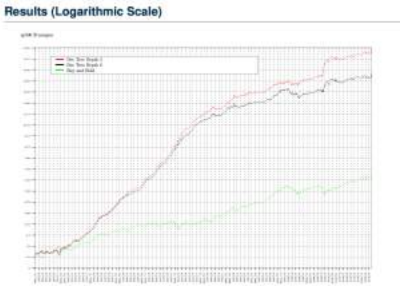
https://necromuralist.github.io/machine-learning-studies/posts/decision-tree-classification/

6. **Decision Trees**

   Tree-Structured Prediction

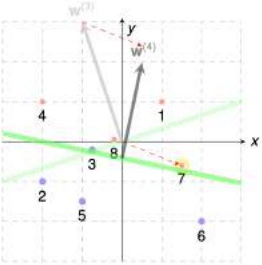
   - Gentle Start: Decision Stumps in Image Classification
   - How to Build Decision Trees?
   - How to Classify?
   - Classification Example: Cancer Detection
   - Regression Example: Stock Price Prediction





7. **Perceptron**
   Linear Classification

   - Inspiration from Neurons
   - Description of the Perceptron Algorithm
   - Detailed Illustration of a Perceptron run
   - Discussion of the Efficiency of Perceptron

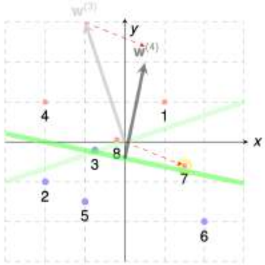


Lecture 7

7. **Perceptron**

   Linear Classification

   - Inspiration from Neurons
   - Description of the Perceptron Algorithm
   - Detailed Illustration of a Perceptron run
   - Discussion of the Efficiency of Perceptron
   - Example: Using Perceptron in Text Classification



i	"and"	"offer"	"the"	"of"	"sale"	y_i
$\mathbf{x}_1$	1	1	0	1	1	+1 pos.
$\mathbf{x}_2$	0	0	1	1	0	−1 neg.
$\mathbf{x}_3$	0	1	1	0	0	+1 pos.
$\mathbf{x}_4$	1	0	0	1	0	−1 neg.
$\mathbf{x}_5$	1	0	1	0	1	+1 pos.
$\mathbf{x}_6$	1	0	1	1	0	−1 neg.

Learning Goals

- become familiar with several machine learning tools

Learning Goals

- become familiar with several machine learning tools
- get experience in analysing data sets

Learning Goals

- become familiar with several machine learning tools
- get experience in analysing data sets
- apply machine learning to a variety of real-life problems:
    - Image Recognition
    - Grade Prediction
    - Weather Forecast
    - Robot Localisation
    - Spam Classification
    - Cancer Detection
    - Stock Price Prediction

    ⋮

Learning Goals

- become familiar with several machine learning tools
- get experience in analysing data sets
- apply machine learning to a variety of real-life problems:
  - Image Recognition
  - Grade Prediction
  - Weather Forecast
  - Robot Localisation
  - Spam Classification
  - Cancer Detection
  - Stock Price Prediction

    ⋮
- many examples: small, intuitive and simple (no need to learn programming or use a specific platform!)

Learning Goals

- become familiar with several machine learning tools
- get experience in analysing data sets
- apply machine learning to a variety of real-life problems:
  - Image Recognition
  - Grade Prediction
  - Weather Forecast
  - Robot Localisation
  - Spam Classification
  - Cancer Detection
  - Stock Price Prediction

    ⋮
- many examples: small, intuitive and simple (no need to learn programming or use a specific platform!)
- Quiz questions and Exercise questions
- . . . and have fun (☺)

Quiz: A Simple(?) Learning Problem (1/2)

example	**label**
train	
197	$+$
128	$-$
30	$-$
72	$-$
133	$-$
109	$+$
213	$+$
84	$+$
3	$-$

(example taken from Robert Schapire)

Quiz: A Simple(?) Learning Problem (1/2)

example	label
train	
197	$+$
128	$-$
30	$-$
72	$-$
133	$-$
109	$+$
213	$+$
84	$+$
3	$-$
test	
200	?
68	?

(example taken from Robert Schapire)

Quiz: A Simple(?) Learning Problem (1/2)

example	**label**
train	
197	$+$
128	$-$
30	$-$
72	$-$
133	$-$
109	$+$
213	$+$
84	$+$
3	$-$
test	
200	?
68	?
111	???

(example taken from Robert Schapire)