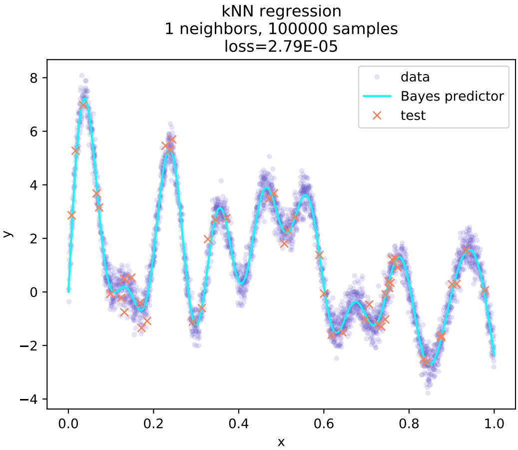
Fondamentaux théoriques du machine learning



General information

Repository of the course :
`https://github.com/nlehir/FTML_PTML`.
This repo will contain :

- lecture slides (FTML)
- practical sessions guidelines and solutions (PTML)
- projects descriptions

Organisation of the course :

- 12×2 hours lectures.
- $12 \times 1,5$ hour practical sessions.

General information

- Lectures : you can ask questions during the lectures.
- We will do short exercices during the lectures (do not hesitate to bring some paper and pen to search).

Overview of lecture 1

Definitions

We will start by the question : **What are AI and ML ?**

Definitions

We will start by the question : **What are AI and ML ?**
First remark : ML is **easier** to define than AI. Several definitions of AI exists, and the term can be used in different contexts.

Examples

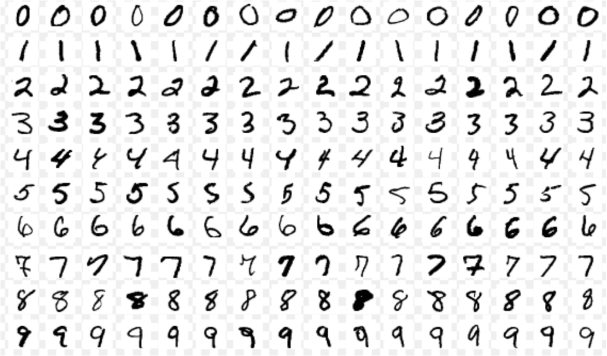


Figure – MNIST database [LeCun and Cortes, 2010]

Introduction examples

- Boston Dynamics robots (video)
- https://www.youtube.com/watch?v=LikxFZZO2sk
- https://www.youtube.com/watch?v=g0TaYhjp0fo
- https://www.youtube.com/watch?v=tF4DML7FIWk&t=57s
- https://www.youtube.com/watch?v=EezdinoG4mk

Introduction examples

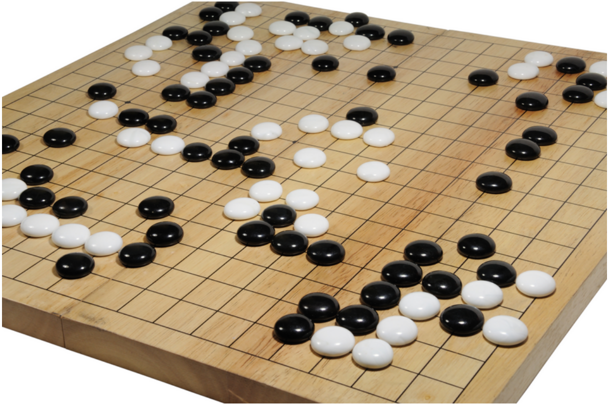


Figure – Go game, won by AlphaGo in 2017 [Silver et al., 2016]

Introduction examples

Figure – Coffee machine (https ://www.stockresto.com/fr/machine-a-cafe/83-machine-a-cafe-conti-cc100-2-groupes.html)

Introduction examples

-
  - Boston Dynamics robot
  - MINST classification
  - AlphaGo
  - Coffee Machine
- All do different things but could be "reasonably" gathered under the term "AI".

Definition

- Let us pick a definition of AI : "the theory and techniques aiming at emulating intelligence".
- Ok, but what is intelligence then ?

History

- It is reasonable to say that the problem of AI was born at the same time as that of **Computer Science**. One of the founders of Computer Science is Alan Turing (1912-1954).

Turing Test

- In 1950 and the article **"Computing machinery and intelligence"** [Turing, 2009], Turing introduces the **Turing test**.

Turing Test

- In 1950 and the article **"Computing machinery and intelligence"** [Turing, 2009], Turing introduces the **Turing test**.
- On of the forms of a Turing Test is a game in which a computer tries to behave like a human by answering questions.

Turing test

Turing, A.M. (1950). Computing machinery and intelligence. Mind, 59, 433-460.

COMPUTING MACHINERY AND INTELLIGENCE

By A. M. Turing

1. The Imitation Game

I propose to consider the question, "Can machines think?" This should begin with definitions of the meaning of the terms "machine" and "think." The definitions might be framed so as to reflect so far as possible the normal use of the words, but this attitude is dangerous, If the meaning of the words "machine" and "think" are to be found by examining how they are commonly used it is difficult to escape the conclusion that the meaning and the answer to the question, "Can machines think?" is to be sought in a statistical survey such as a Gallup poll. But this is absurd. Instead of attempting such a definition I shall replace the question by another, which is closely related to it and is expressed in relatively unambiguous words.

The new form of the problem can be described in terms of a game which we call the 'imitation game." It is played with three people, a man (A), a woman (B), and an interrogator (C) who may be of either sex. The interrogator stays in

Turing test

- Importantly, Turing stresses that the terms "think", and "machines" are far from being **unambiguous**.
- Related to that, many questions regarding our own "intelligence" and "consciousness" as humans remain open today.
- In that respect, it is tricky to give a indisputable definition of AI.

Moravec Paradox

- It is harder to emulate or simulate simple sensorimotor capacities than level abstract reasoning.
- ie : the sensorimotor system of an ant is harder for us to emulate than beating the world champion at chess (Deep Blue, 1997).

Strong AI, Weak AI

- There exists a distinction between **Weak AI**, and a **hypothetical** AI called **Strong AI**.

Strong AI, Weak AI

- There exists a distinction between **Weak AI**, and a **hypothetical** AI called **Strong AI**.
- **Weak AI** (also known as **narrow AI**) designed and trained for a particular task (e.g. Siri).

Strong AI, Weak AI

- There exists a distinction between **Weak AI**, and a **hypothetical** AI called **Strong AI**.
- **Weak AI** (also known as **narrow AI**) designed and trained for a particular task (e.g. Siri).
- **Strong AI** (also known as Artificial General Intelligence AGI) : able to find a solution, faced with an **unfamiliar task**. This is **hypothetical**, it does not exist yet, and may never exist.

Conscious reasoning

Broadly speaking, the first phases of research in AI were more dedicated to emulating the **conscious brain.**

- logic programming (Prolog)
- expert systems

History : AI Winter (1970's)

- In 1974 fundings for AI reseach started being cut by goverments because of unsuccessful projects.
- there were actually several so-called "AI winters".

Famous AI : Deep Blue (1997)

- Beat the world chess champion Gary Kasparov in 1997.
- 

Figure – Deep Blue

- Despite its name, Deep Blue is **not related** to Deep learning !

Decision trees

One of the underlying mechanisms of Deep Blue were **Decision Trees** / minimax algorithm.

Limitations of the method

- Let p be the **branching factor** of the tree. (Here, the average number of children at each node)
- Let d be the **depth** of the tree.

Limitations of the method

Exercice 1 : **Computation time**

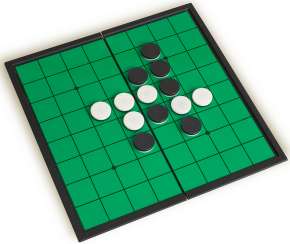
- Let p be the **branching factor** of the tree of actions. (Here, the average number of children at each node)
- Let d be the **depth** of the tree.
- What is the order of magnitude of the number of nodes in the tree ?

Limitations of the method

- Let p be the **branching factor** of the tree of actions. (Here, the average number of children at each node)
- Let d be the **depth** of the tree.
- What is the order of magnitude of the number of nodes in the tree ?
- So in order to run the minimax algorithm needs to perform p^d evaluations.

Exercise

- In the **Othello game**, the average number of actions in each state is around 8.
- We assume that the evaluation for one leaf node takes 1×10^{-6} seconds.

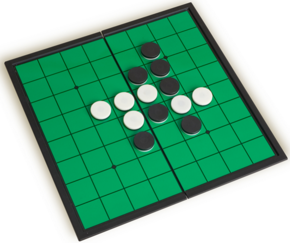


-

Exercise

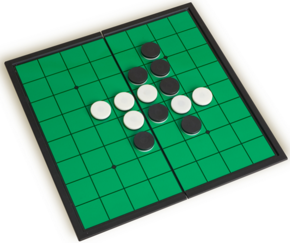
Exercice 2 : **Computation time**

- In the **Othello game**, the average number of actions in each state is around 8.
- We assume that the evaluation for one leaf node takes 1×10^{-6} seconds.
- How long would be the search of the minimax if we look 10 actions ahead ?



Exercise

- In the **Othello game**, the average number of actions in each state is around 8.
- We assume that the evaluation for one leaf node takes 1×10^{-6} seconds.
- This duration is too long to be used.



▶

2000's and 2010's

- Around the 2000's, there has been a shift in the focus of AI.
- Machine learning and data science have become prominent research and industrial topics.
- We could argue that emulating the **unconscious brain** is more of a focus today.

2000's and 2010's

- Around the 2000's, there has been a shift in the focus of AI.
- Machine learning and data science have become prominent research and industrial topics.
- We could argue that emulating the **unconscious brain** is more of a focus today.
- An important contemporary and future trend in research might be to connect research on emulating the conscious brain and research emulating the unconscioue brain.

Machine learning

- **Proposed definition :** "artificial intelligence which can learn and model some phenomena without being explicitely programmed".
- In Machine learning, some parameters are learned in an *automatic way* in order to solve a problem or to optimize a solution.

Machine learning

Machine learning has received attention and funding because it has reached state-of-the-art efficiency on several problems, such as :

- computer vision
- spam classification
- machine translation
- speech recognition
- self-driving cars

Machine learning

Machine learning has received attention and funding because it has reached state-of-the-art efficiency on several problems such as :

- computer vision
- spam classification
- machine translation
- speech recognition
- self-driving cars

Deep learning is involved in several of these setups.

Example 1 : ImageNet

- A database of images (more than 15M), hand-annotate in order to indicate what objects are present in the image. More than 20000 categories of images.
- Contest : ImageNet Large Scale Visual Recognition Challenge.
- The best classification went from 25% in 2011 to $\simeq$ 15.3% in 2012.

Example 1 : ImageNet

- A database of images (more than 15M), hand-annotate in order to indicate what objects are present in the image. More than 20000 categories of images.
- Contest : ImageNet Large Scale Visual Recognition Challenge.
- The best classification (top-5 score) went from 25% in 2011 to $\simeq$ 15.3% in 2012.
- The technology used was deep learning, exploiting GPUs (AlexNet).

Example 2 : AlphaGo

- In 2015 : beats a professionnal player. In 2017 : beats the world champion.
- Uses several technologies : among them **Deep reinforcement learning.**
- Improvements : AlphaGo Zero, trained without a database of played games. In 20217, AlphaZero beats AlphaGo Zero after 3 days of learning.

Example 3 : AlphaFold

- Goal : to predict the spatial configuration of proteins, from their DNA sequence.
- Achieves a breakthgough performance on the CASP challenge :
  - 2018 : more than 50% GDT (Global distance test), whereas it was $\leq 40\%$ before then.
  - 2020 : $92,4\%$ GDT. At a $\geq 90\%$ score, the method is considered competitive with experimental methods..
- https://alphafold.ebi.ac.uk/
- Also based on Deep learning.

Deep learning

In this course, we will not focus much on deep learning, but rather on more elementary/fundamentals tools.

AI and Machine Learning

In the four first examples, according to you which one are NOT a Machine Learning system ?

AI and Machine Learning

- Alpha Go : Machine Learning (Reinforcement Learning)

AI and Machine Learning

- Alpha Go : Machine Learning (Reinforcement Learning)
- MNIST : Machine Learning (Supervised Learning)

AI and Machine Learning

- Alpha Go : Machine Learning (Reinforcement Learning)
- MNIST : Machine Learning (Supervised Learning)
- Coffee Machine : no Machine Learning (Automation)

AI and Machine Learning

- Alpha Go : Machine Learning (Reinforcement Learning)
- MNIST : Machine Learning (Supervised Learning)
- Coffee Machine : no Machine Learning (Automation)
- Boston Dynamics : no Machine Learning (Robotics)

AI and Machine Learning

- Alpha Go : Machine Learning (Reinforcement Learning)
- MNIST : Machine Learning (Supervised Learning)
- Coffee Machine : no Machine Learning (Automation)
- Boston Dynamics : no Machine Learning (Robotics)
- **edit :** while the robots on the video have no Machine Learning, Boston Dynamics seems to now show interest in the topic. ML technologies might start being implemented in their robots, especially for artificial vision.

AI and science

- People doing "AI" today can actually come from rather different scientific communities. Which ones according to you ?

AI and science

- People doing "AI" today can actually come from rather different scientific communities :
  - Statistics and probabilities

AI and science

- People doing "AI" today can actually come from rather different scientific communities :
  - Statistics and probabilities
  - Optimization

AI and science

- People doing "AI" today can actually come from rather different scientific communities :
  - Statistics and probabilities
  - Optimization
  - Other mathematical fields :

AI and science

- People doing "AI" today can actually come from rather different scientific communities :
  - Statistics and probabilities
  - Optimization
  - Other mathematical fields : graph theory, combinatorics, information theory

AI and science

- People doing "AI" today can actually come from rather different scientific communities :
  - Statistics and probabilities
  - Optimization
  - Other mathematical fields : graph theory, combinatorics
  - Statistical physics

AI and science

- People doing "AI" today can actually come from rather different scientific communities :
  - Statistics and probabilities
  - Optimization
  - Other mathematical fields : graph theory, combinatorics
  - Statistical physics
  - Robotics

AI and science

- People doing "AI" today can actually come from rather different scientific communities :
  - Statistics and probabilities
  - Optimization
  - Other mathematical fields : graph theory, combinatorics
  - Statistical physics
  - Robotics
  - Cognitive sciences / neuroscience / psychology

AI and science

- People doing "AI" today can actually come from rather different scientific communities :
  - Statistics and probabilities
  - Optimization
  - Other mathematic fields : graph theory, combinatorics
  - Statistical physics
  - Robotics
  - Cognitive sciences / neuroscience / psychology
- For instance, *data science* is mostly a mix between statistics, optimization, graph theory

ML revolution

- Technical progress in the computing and storage capacities
- Increase in amount of available data. According to IBM, 10^{18} bytes are created each day.
- Progress in algorithmic methods to analyze the data.

Learning paradigms

- Let us now study **machine learning paradigms**.

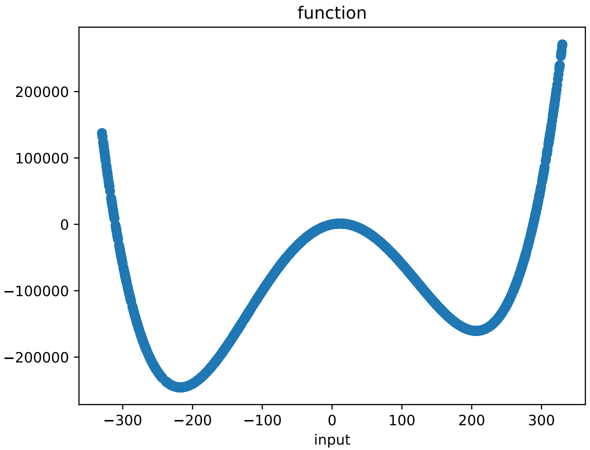
Machine learning and Optimization

- Let us now study **Machine Learning paradigms**.
- We said earlier that in Machine Learning, some parameters are learned in an automatic way to solve a problem.

Machine Learning and Optimization

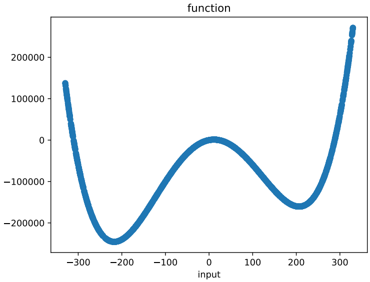
- Let us now study **Machine Learning paradigms**.
- We said earlier that in Machine Learning, some parameters are learned in an automatic way to solve a problem.
- But how are the parameters learned ?

Notion of optimization



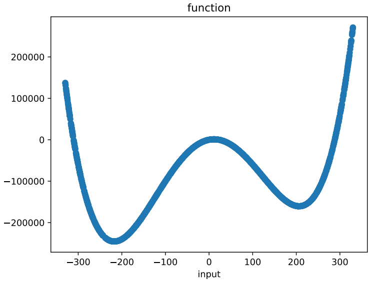
Notion of optimization

Often, the **quality** of a set of parameters can be measured with a real number. This measure helps us in finding the best parameters possible.



Notion of optimization

Often, the **quality** of a set of parameters can be measured with a real number. This measure helps us in finding the best parameters possible.
The question is then the choice of this measure, often called a **loss function**, or **loss function**.



Supervised Learning : The problem

- For a certain input x, you want to **predict** an output y : for instance,
  - x : contains the age, and the height of a person, so here x is a **vector** containing **two features**.
  - y : best record on a 100 meters track
- To do so, you learn from a number of **labeled examples** (x_i, y_i)
- In the case where what you want to predict is a **class**, it is a **classification problem**
- In the case where what you want to predict is a general function $y = f(x)$, it is a **regression problem** (example : 100 meters track time)

Supervised learning

- To do so, you learn from a number of **labeled examples** (x_i, y_i)
- In the case where what you want to predict is a **class**, it is a **classification problem** : $y \in \mathbb{N}$. (example : MNIST)
- In the case where what you want to predict is a general function $y = f(x)$, it is a **regression problem** : $y \in \mathbb{R}$.
- **Objective :** find a good estimation $\tilde{f}$, of f.

Choices and modelisation

- What subset of functions should $\tilde{f}$ belong to ?
- What **loss function** whould we use ?

Lost function

- What **loss function** whould we use ?
- It should be a measure of the discrepancy between our prediction and the correct label.

Loss function

- What **loss function** whould we use ?
- It should be a measure of the discrepancy between our prediction and the correct label.
- Let us take the example of a regression problem, when the label is a real number.

Loss function

- What **loss function** whould we use ?
- It should be a measure of the discrepancy between our prediction and the correct label.
- Let us take the example of a regression problem, when the label is a real number.
- For an individual sample, **a** discrepancy is the least-square loss

$$(f(x_i) - y_i)^2 \tag{1}$$

Loss function

- Taking into account the whole dataset, the **loss function** writes :

$$\sum_{i=1}^{n} (f(x_i) - y_i)^2 \tag{2}$$

- Several other loss functions are possible :

$$\sum_{i=1}^{n} |f(x_i) - y_i| \tag{3}$$

Loss function

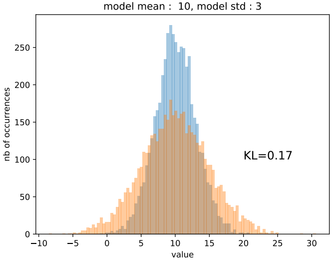
- The loss function is a **real number** measuring the relevance of a **collection** of parameters.
- The number of parameters depends on the situation, and varies between 1 (e.g. for a simple linear model) and millions (e.g. for deep neural networks).

Unsupervised Learning

- From a number of samples x_i, you want to retrieve information on their structure : **modelisation**.
-
  - density estimation
  - clustering
  - dimensionality reduction

Unsupervised Learning

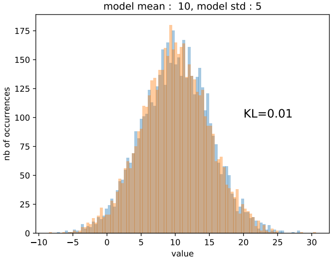
- From a large number of samples x_i, you want to retrieve information on their structure : **modelisation**.
- density estimation :



-

Unsupervised Learning

- From a large number of samples x_i, you want to retrieve information on their **structure**
- density estimation :



-

Unsupervised Learning

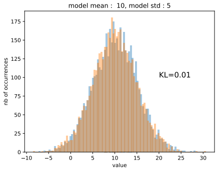
- From a large number of samples x_i, you want to retrieve information on their **structure**
- density estimation.
- Question : how do you constrain your distribution to fit your data ?
  - Parametric model
  - Non-parametric model

Loss function

- It is a little bit harder to think of a loss function in that situation.

Loss function

- It is a little bit harder to think of a loss function in that situation.
- In the case of fitting a distribution, the **Kullbach-Leibler divergence** is an example of loss function.



Reinforcement Learning

- A **more general paradigm** that describes an **agent** in a **world**.
- The standard formalization was the one proposed by Richard Sutton [Andrew, 1998] (pdf available online).

Reinforcement Learning

- A **more general paradigm** that describes an **agent** in a **world**.
- The standard formalization was the one proposed by Richard Sutton [Andrew, 1998]
- At each time, the world is in a state s. An agent perfoms an **action** a according to a **policy** π. When performing an action, the agent receives an **reward** r.

Reinforcement Learning

- A **more general paradigm** that describes an **agent** in a **world**.
- The standard formalization was the one proposed by Richard Sutton [Andrew, 1998]
- At each time, the world is in a state s. An agent perfoms an **action** a according to a **policy** π. When performing an action, the agent receives an **reward** r.

Reinforcement Learning applications

- Example : some Chessplayers, AlphaGo, automatic vacuum cleaner, self-driving cars (trajectory optimization, parking, etc.)
- healthcare : https://arxiv.org/pdf/1908.08796.pdf
- https://deepmind.com/blog/article/alphago-zero-starting-scratch

Reinforcement Learning

- At each time, the world is in a state s. An agent perfoms an **action** a according to a **policy** π. When performing an action, the agent receives an **reward** r.
- The agent wants to learn an **optimal policy**, meaning the policy that maximises its reward.

This paradigm has many variants

- State s, action a, policy π, reward r.
- Is the policy **deterministic** ? Is it **stochastic** ?
- Does the agent have a **model** of its environment ?
- How many steps ahead whould the agent look ?

This paradigm has many variants

- State s, action a, policy π, reward r.
- Is the policy **deterministic** ? Is it **stochastic** ?
- Does the agent have a **model** of its environment ?
- How many steps ahead should the agent look ?
- All these conditions change the way the problem should be addressed and solved. The **Bellman equations** rule the updates of the optimal policy.

Example problem

Exercice 3 : **Exploration and exploitation**
Should I explore my environment more or exploit what I have learnt so far ?

Example problem

Concept ϵ-greedy policy.

Final remark

- These paradigms can be mixed

Final remark

- These paradigms wan be mixed
- Mosty, this means that
  - unsupervised learning can be used in a supervised learning problem (semi supervised learning)
  - unsupervised learning and supervised learning can be used in a reinforcement learning problem

Example I

Predict the winning team of an NBA game at half-time.

- Dataset : 15 years of games (comments, text) : approximately 17000 games.
- The dataset is preprocessed to have as an input a time-series : each time contains the score **and** 10 technical features (rebounds, etc.). So for each time the dimension is 11. Each game is a matrix of size 1440×11, reorganized as a line vector.
- Output : Receiving team wins or looses (classification)
- Evaluation metric : classification error ("0-1" loss).

Example II

Predict the quantity of oil in a rock.

- Input : tomographic image of a rock.
- Output : material of the rock, average presence of residual oil in the rock (regression).

Example III

Detect issues in wind farms.

- Input : sensors on the wind turbine (wind direction, air temperature, electric tension, rotation speed, component temperature, etc.) as a time series. Each step represents 10 minutes (several years).
- Output : Power generated by the turbine (regression)
- Evaluation metric : MAE (mean absolute error).

Empirical risk minimization

- We consider a **regression problem**, from $\mathbb{R}$ to $\mathbb{R}$.
- The dataset D_n is a collection of n samples $\{(x_i, y_i)\}_{1 \leq i \leq n}$, that are **independent and identically distributed** draws of a joint random variable (X, Y).
- the law of (X, Y) is unknown, we can note it ρ. We assume there exists an unknown function f that relates X and Y (not necessary deterministic).
- we look for an estimator $\tilde{f}_n$ of f. n refers to the fact that we have n samples.

Risks

Let l be a loss.
The **risk** (or **statistical risk, generalization error, test error)** of estimator f writes

$$E_{(X,Y)\sim\rho}[l(Y, f(X))]$$

The **empirical risk (ER)** of an estimator f writes

$$R_n(f) = \frac{1}{n}\sum_{i=1}^{n} l(y_i, f(x_i))$$

We emphasize that the risks depend on the loss l.

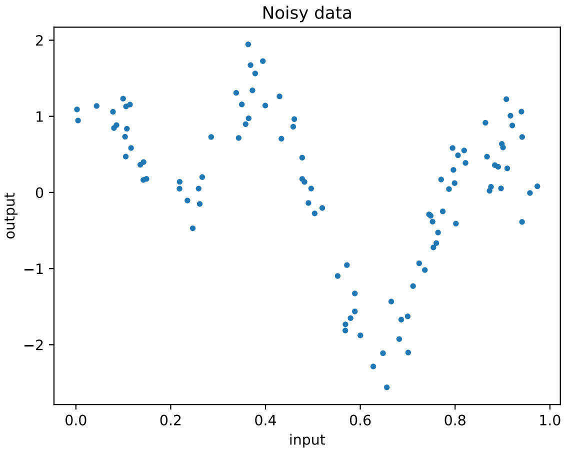
Empirical risk minimization

- **Empirical risk minimization** (ERM) consists in searching for an estimator $\tilde{f}_n$ that minimizes R_n.
- The question is : does ERM provide an estimator that minimizes the generalization error R ? (which **is** what we are interested in)

Hypothesis space

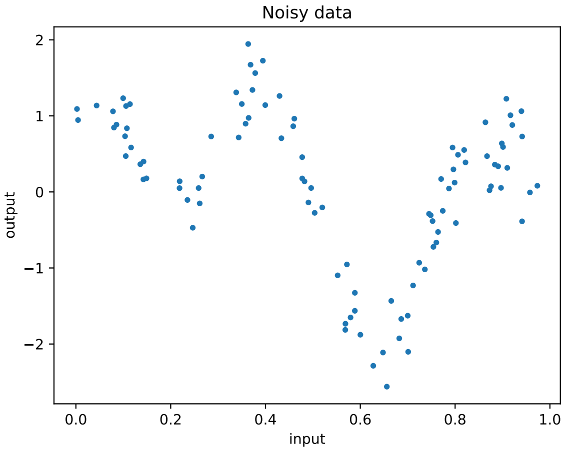
The **hypothesis space** is the subset of functions in which $\tilde{f}_n$ lies..

Polynomial regression

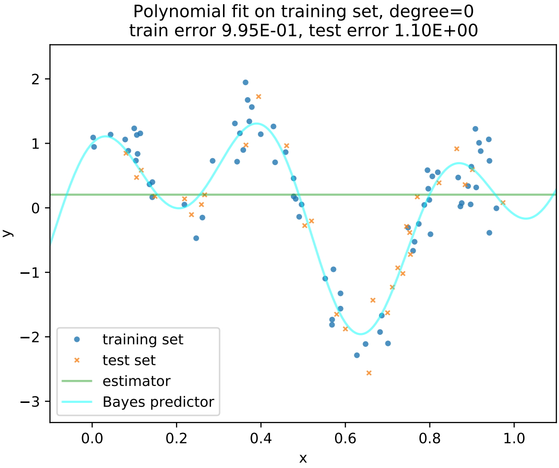


Polynomial regression

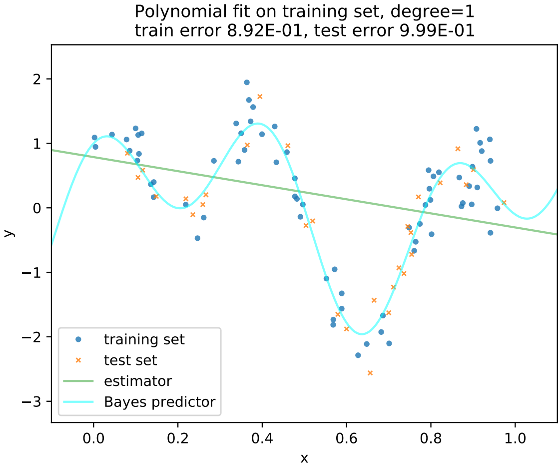
Exercice 4 : **ERM** : is there a polynom that interpolates the data ?



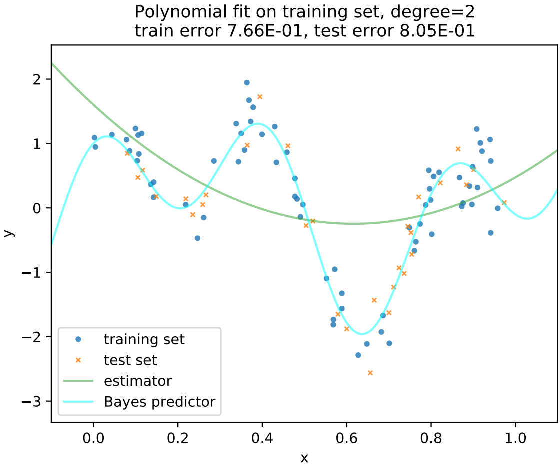
Polynomial regression



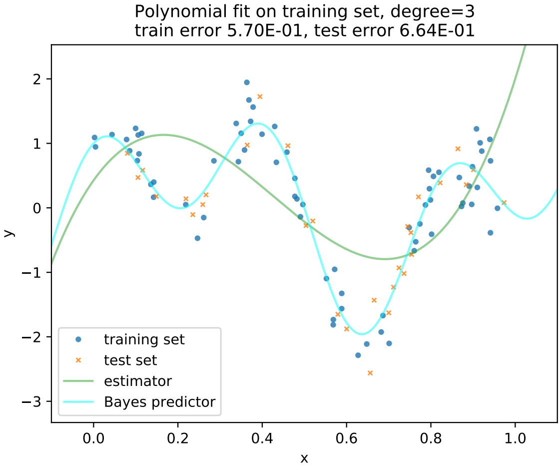
Polynomial regression



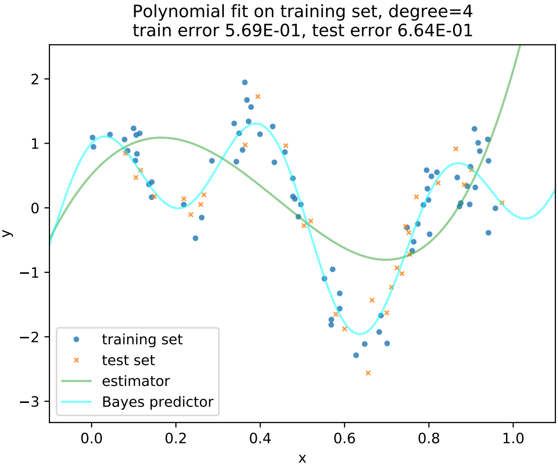
Polynomial regression



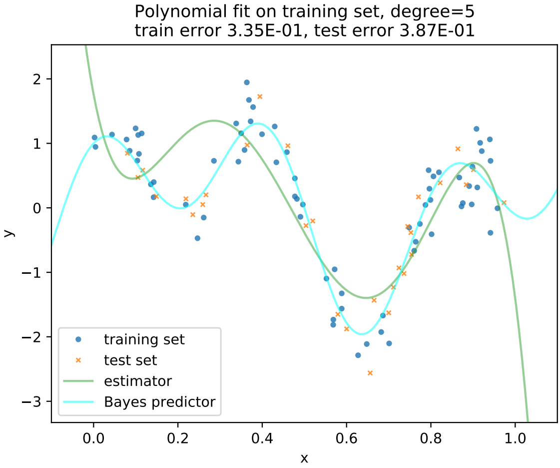
Polynomial regression



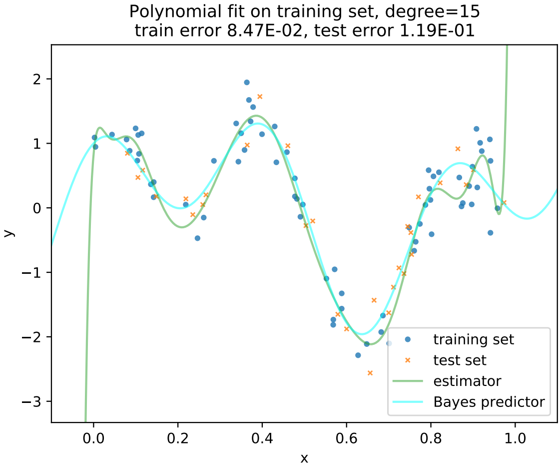
Polynomial regression



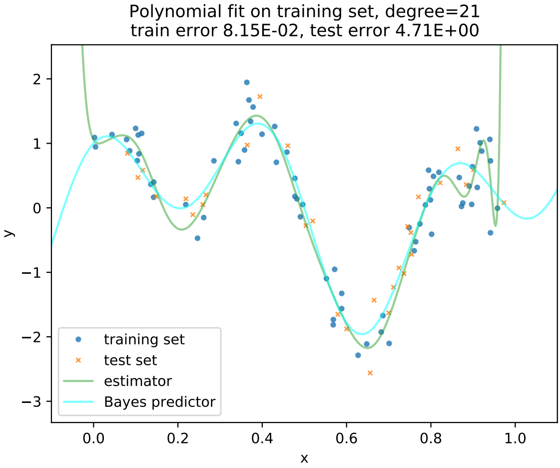
Polynomial regression



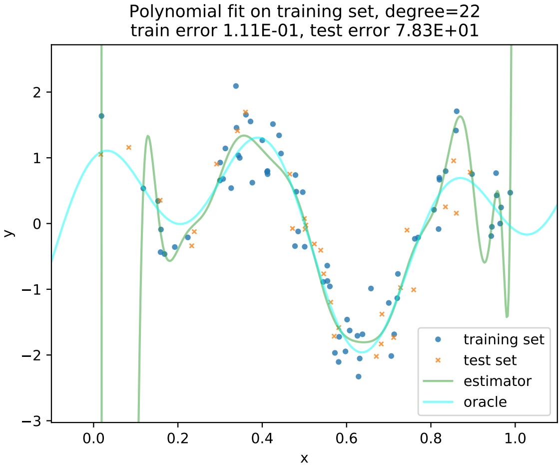
Polynomial regression



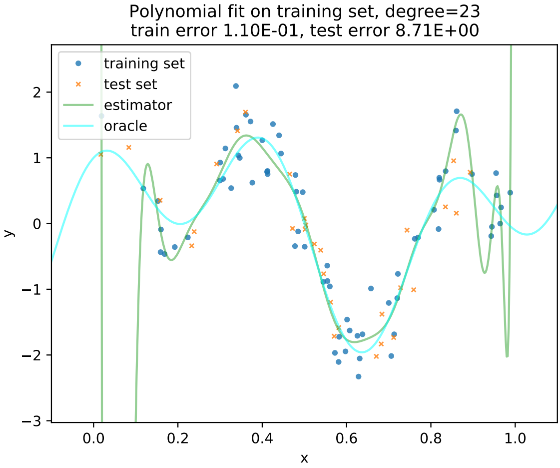
Polynomial regression

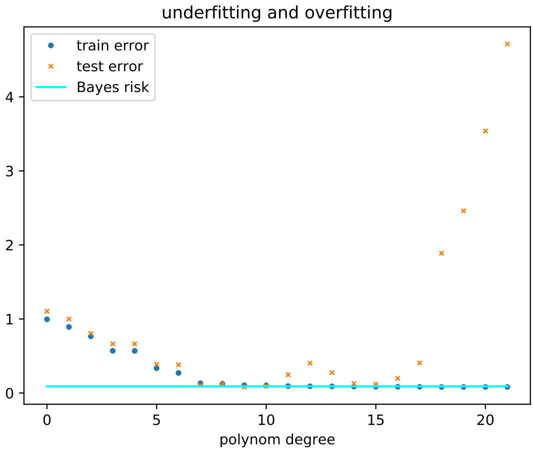


Polynomial regression



Polynomial regression




underfitting and overfitting
train error
test error
Bayes risk
0
1
2
3
4
0
5
10
15
20
polynom degree

Curse of dimensionality

Two numbers are important in machine learning :

- n : number of samples
- d : dimension (number of features) of a unique sample

Both can be large and prohibitive for some algorithms.

Curse of dimensionality

Two numbers are important in machine learning :

- n : number of samples
- d : dimension (number of features) of a unique sample

Both can be large and prohibitive for some algorithms.

Exercice 5 : **Example :** what is the algorithmic complexity of the inversion of an invertible matrix $A \in \mathbb{R}^{d,d}$?

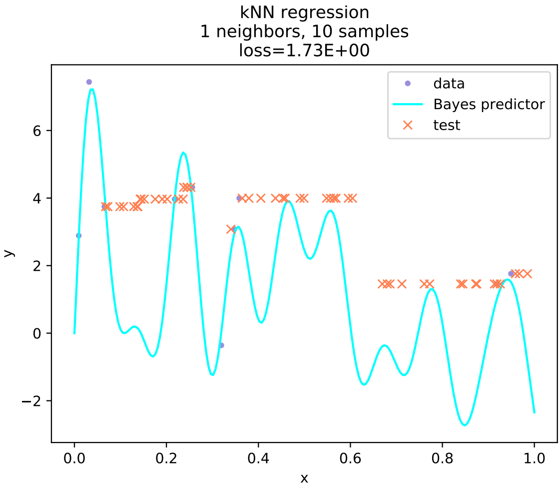
Curse of dimensionality

- n is large when the dataset has many samples.
- d is large is each sample has many features :
  - image
  - DNA sequence
  - text
  - audio/video file

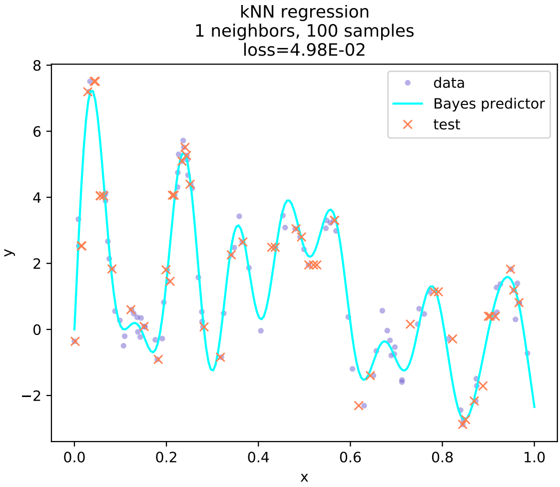
Local averaging

The curse of dimensionality is the reason why **local averaging** does not work in large dimensions.

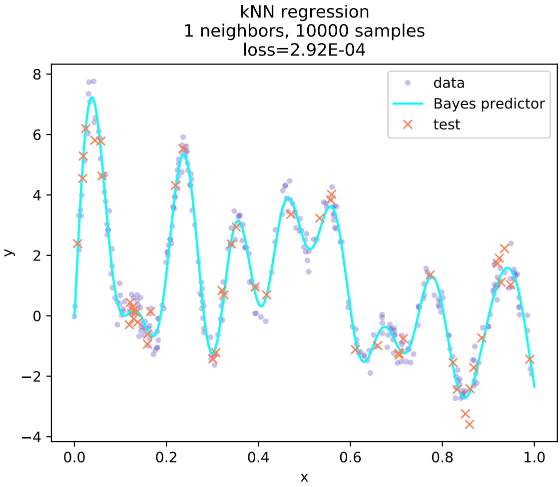
Local averaging



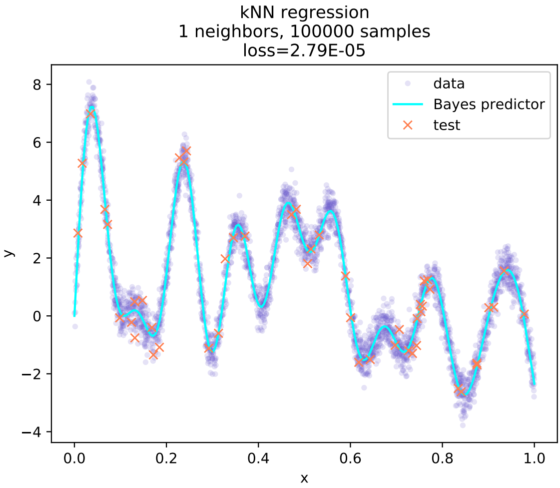
Local averaging



Local averaging



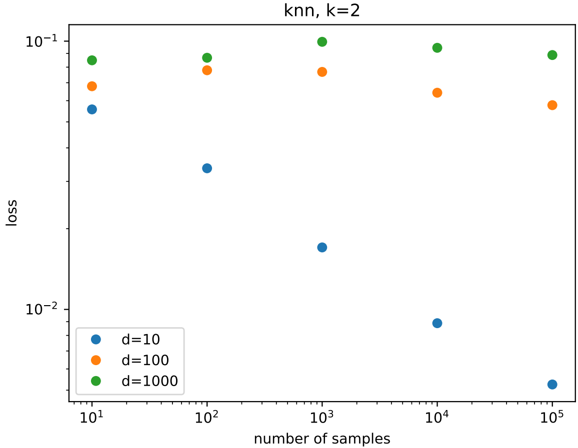
Local averaging



d dimensions

Exercice 6 : **Local averaging in d dimensions**. In d dimensions, with a regular mesh of b samples, with a uniformly Lipshitz function, what is the maximum of the predictor ?

In d dimensions



Optimization

We often have to optimize functions that are **hard** to optimize.

- non convexity
- non linearity
- instability of estimators

References I

Andrew, A. M. (1998).
Reinforcement Learning : An Introduction.

LeCun, Y. and Cortes, C. (2010).
{MNIST} handwritten digit database.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., and Hassabis, D. (2016).
Mastering the Game of {Go} with Deep Neural Networks and Tree Search.
Nature, 529(7587) :484–489.

References II

Turing, A. M. (2009).
Computing machinery and intelligence.
In *Parsing the Turing Test : Philosophical and Methodological Issues in the Quest for the Thinking Computer.*