Introduction to Machine Learning

Formation ENSTA ParisTech
Conférence IA

Florent Chatelain* Olivier Michel*

*GIPSA-lab, Univ. Grenoble Alpes,

2-5 février 2021
ENSTA, France

The presenters

Florent Chatelain

- Ph.D. degree in signal processing from the National Polytechnic Institute, Toulouse, France, in 2007
- Post-doc position at INRIA - ARIANA Team, 2007-2008
- Since 2008, Associate Professor at GIPSA-Lab, University of Grenoble, France.
- Research interests are centered around estimation, detection and large scale inference

Olivier J.J. Michel

- Former student ENS-Cachan (ENS Paris-Saclay), aggrégation in applied physics, 1986
- Ph-D degree in signal processing from Univerisity paris 11-Orsay, 1991, Post-Doc at Univ. of Michigan, USA
- Associate prof at Lab. de Physique, ENS-Lyon 1991-1999
- Prof. at Univ. Nice Sophia-Antipolis, Astrophysics lab, 1999-2008
- Prof at GIPSA-Lab, University of Grenoble, France.
- Research interest : random processes, estimation, detection, AI for astro and geo-sciences.

Organization

Volume

- 4 $\times$ 7h lecture and practices sessions

	Monday	Tuesday	Wednesday	Thursday
09h30-12h45	Intro	Classif (Cont'd)	Clustering	Neural Nets: basics, deep
		Lunch		
14h15-17h30	PCA + Classif	Linear models	Random Forests	Recurrent Neural Nets

Objectives

- Understand the theoretical basis of data science/machine learning/AI
- Assess the quality of predictions and inferences
- Implement/apply data science algorithms and models using state-of-the-art frameworks

The material

- Slides (pdf) and notebooks available here : `https://gricad-gitlab.univ-grenoble-alpes.fr/chatelaf/conference-ia/`
- Jupyter notebooks are available to illustrate concepts and methods in Python (.ipynb files)
- Binders are also available to run them remotely and interactively (see README.md)

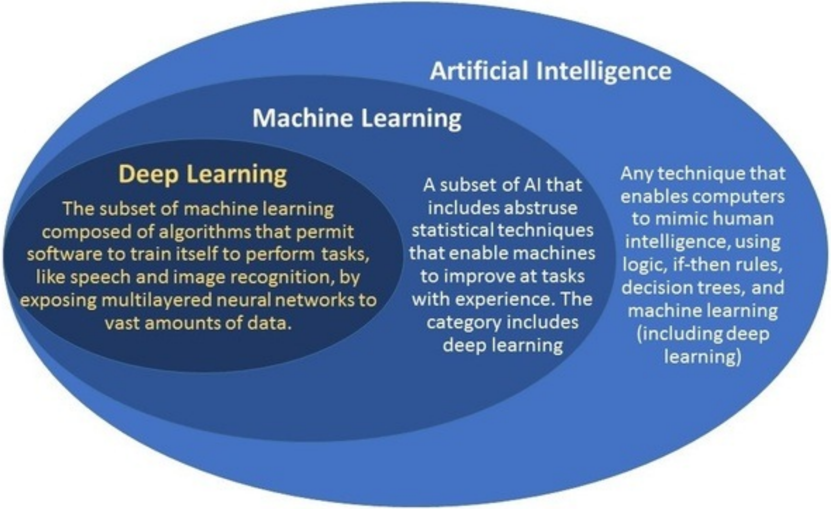
Reference books

- Trevor Hastie, Robert Tibshirani et Jerome Friedman (2009)
  The Elements of Statistical Learning (2nd Edition)
  Springer Series in Statistics
- Christopher M. Bishop (2007)
  Pattern Recognition and Machine Learning *Springer*
- Kevin P. Murphy (2012)
  Machine Learning. A Probabilistic Perspective *MIT Press*

Supplementary materials, datasets, online courses, ...

- http://www-stat.stanford.edu/~tibs/ElemStatLearn/
- https://www.coursera.org/course/ml *very popular MOOC (Andrew Ng)*
- https://work.caltech.edu/telecourse.html *more involved MOOC (Y. Abu-Mostafa)*
- https://scikit-learn.org/stable/auto_examples/index.html *Examples from the sklearn library*

Machine Learning $\subset$ Artificial Intelligence



Data Science Objective

How to extract knowledge or insights from data ?

Learning problems are at the cross-section of several applied fields and science disciplines

- *Machine learning* arose as a subfield of
  - Artificial Intelligence,
  - Computer Science.

  Emphasis on large scale implementations and applications: algorithm centered
- *Statistical learning* arose as a subfield of
  - Statistics,
  - Applied Maths,
  - Signal Processing, . . .

  Emphasizes models and their interpretability: model centered

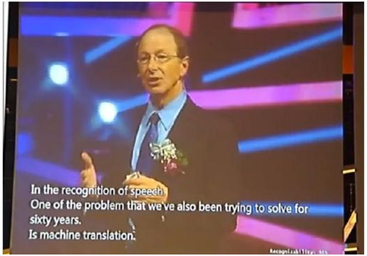
☞ There is much overlap: Data Science

Learning problem

Learning: human vs machine

The learning of a child

- walking: 1 year
- speaking: 2 years
- reasoning: the rest of the time



Definitions of Learning

Machine Learning in Computer Science

Tom Mitchell (The Discipline of Machine Learning, 2006)

A computer program CP is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E

Key points

- Experience E: data and statistics
- Performance measure P: optimization
- tasks T: utility
  - automatic translation
  - playing Go
  - ... doing what human does

Definitions of Learning

Machine Learning in Computer Science

Tom Mitchell (The Discipline of Machine Learning, 2006)

A computer program CP is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E

Key points

- Experience E: data and statistics
- Performance measure P: optimization
- tasks T: utility
  - automatic translation
  - playing Go
  - ... doing what human does

Definitions of Learning

Machine Learning in Computer Science

Tom Mitchell (The Discipline of Machine Learning, 2006)

A computer program CP is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E

Key points

- Experience E: data and statistics
- Performance measure P: optimization
- tasks T: utility
  - automatic translation
  - playing Go
  - ... doing what human does

Definitions of Learning

Machine Learning in Computer Science

Tom Mitchell (The Discipline of Machine Learning, 2006)

A computer program CP is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E

Key points

- Experience E: data and statistics
- Performance measure P: optimization
- tasks T: utility
  - automatic translation
  - playing Go
  - ... doing what human does

Experience E: the data!

Type of data: qualitatives / ordinales / quantitatives variables

- text: strings
- speech: time series
- ages/videos: 2/3d dependences
- networks: graphs
- games: interaction sequences
- ...

Big data (volume, velocity, variety, veracity)

Data are available without having decided to collect them!

- importance of preprocessings (cleaning up, normalization, coding,...)
- importance of a good representation : from raw data to vectors

Objective and performance measures P

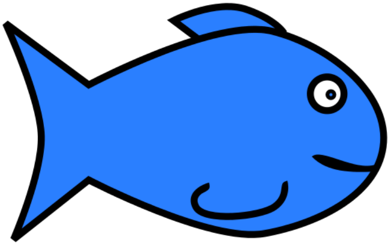
Generalize

- Perform well (minimize P) on new data (fresh data, i.e. unseen during learning)
- "Statistical learning": derive good (P/error rate) prediction functions

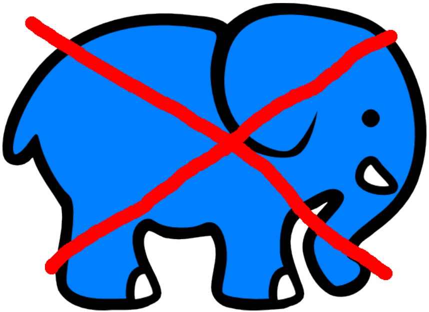
Objective and performance measures P

Generalize

- Perform well (minimize P) on new data (fresh data, i.e. unseen during learning)
- "Statistical learning": derive good (P/error rate) prediction functions



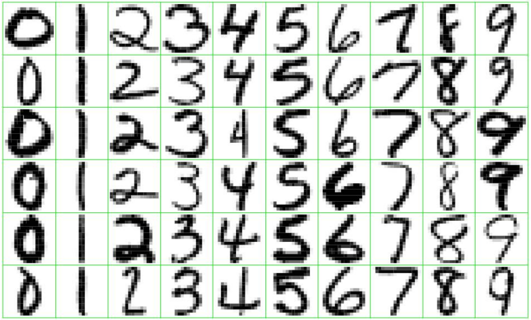
A fish



A fish

Examples of Tasks

Recognition of handwritten digits (US postal envelopes)



- ☞ Predict the class (0,...,9) of each sample from an image of 16×16 pixels, with a pixel intensity coded from 0 to 255
- ▶ Low error rate to avoid wrong allocations of mails!

Supervised classification

Examples of Tasks

Spams Recognition

Spam

```
WINNING NOTIFICATION
We are pleased to inform you of the result
of the Lottery Winners International
programs held on the 30th january 2005.
[...]  You have been approved for a lump sum
pay out of 175,000.00 euros.
CONGRATULATIONS!!!
```

No Spam

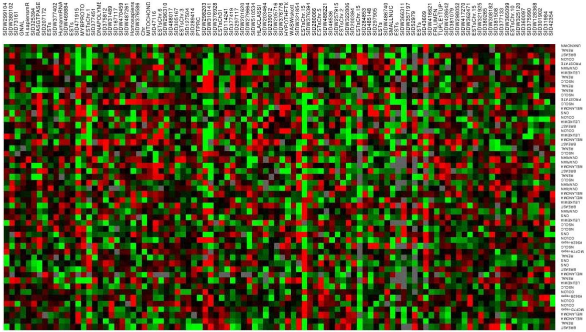
```
Dear George,
Could you please send me the report #1248 on
the project advancement?
Thanks in advance.

Regards,
Cathia
```

- ☞ Define a model to predict whether an email is spam or not
- ▶ Low error rate to avoid deleting useful messages, or filling the mailbox with useless emails

supervised classification

Examples of Tasks

DNA-microarrays

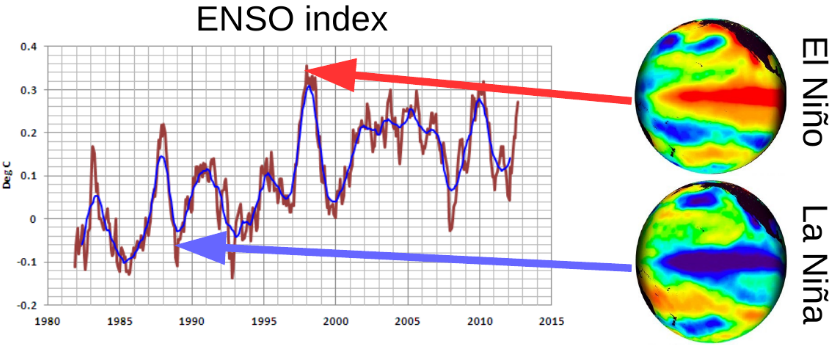


- Genes expression dataset fore several thousand individual genes (columns) and tens of samples (rows)
- ☞ Classification of genes (resp. samples) with similar expression profiles across samples (resp. genes)

unsupervised classification

Examples of Tasks in Geosciences

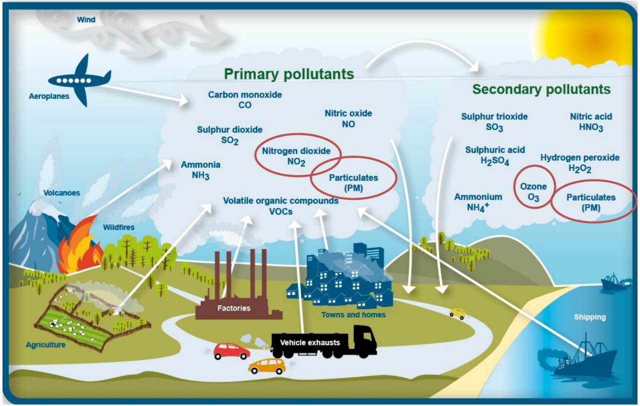
Prediction of El Niño southern oscillation



☞ Predict, 6 months in advance, the intensity of an El Niño Southern Oscillation (ENSO) event from ocean-atmosphere datasets (sea level pressure, surface wind components, sea surface temperature, surface air temperature, cloudiness...)

supervised regression

Prediction of pollutant concentrations



- ☞ Predict pollutant concentrations (0_3,$N0_2$,PM10,PM2.5) at time D_0+1,+2,+3 from hourly measures timeseries + weather data + chemistry based forecasting models

supervised regression (pollutant concentration predicton) / classification (pollution alert or not)

Definitions

Variable terminology

- observed data referred to as *input* variables, *predictors* or *features* $\leftarrow$ usually denoted as X
- data to predict referred to as *output* variables, or *responses* $\leftarrow$ usually denoted as Y

Type of prediction problem: regression vs classification

Depending on the type of the *output* variables

- when Y are quantitative data (continuous variables, e.g. ENSO intensity index values) $\leftarrow$ regression
- when Y are categorical data (discrete qualitative variables, e.g. handwritten digits $Y \in \{0, \dots, 9\}$) $\leftarrow$ classification

Two very close problems.

Prediction problem

Assumptions

- couples of input and output variables (X_i, Y_i) are i.i.d.
- input variables X_i are vectors in $\mathbb{R}^p$:

$$X_i = (X_{i,1}, \dots, X_{i,p})^T \in \mathcal{X} \subset \mathbb{R}^p$$

- output variables Y_i take values:
  - in $\mathcal{Y} \subset \mathbb{R}$ (regression)
  - in a finite set $\mathcal{Y}$ (classification)

Prediction rule

function of prediction / rule of classification $\equiv$ function $\widehat{f}: \ \mathcal{X} \to \mathcal{Y}$ that estimate the true link function f to get predictions of new elements Y given X

$$\widehat{Y} = \widehat{f}(X)$$

Supervised or unsupervised learning

Training set $\equiv$ available sample $\mathcal{T}$ to learn the prediction rule f

For a sized n training set, different cases:

- Supervised learning: $\mathcal{T} \equiv ((X_1, Y_1), \ldots, (X_n, Y_n))$ input/output couples are available to learn the prediction rule f
- Unsupervised learning: $\mathcal{T} \equiv (X_1, \ldots, X_n)$ only the inputs are available
- Semi-supervised: mixed scenario (often encountered in practice, but less information than in the supervised case)

During this course:

- most courses and labs devoted to supervised learning (more interpretable results, abundant literature)
- sessions on unsupervised learning: *dimension reduction (PCA)*, and *clustering*

Model complexity

Most of methods have a complexity related to their *effective* number of parameters

Linear regression: model order p

E.g. dth degree polynomial regression: $p = d + 1$ parameters β_k s.t.

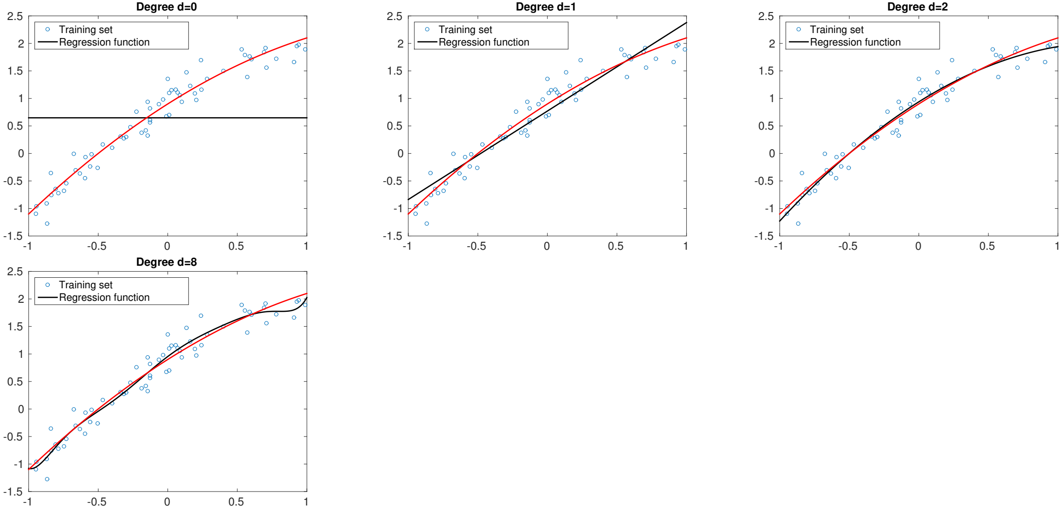
$$\begin{aligned} Y &= \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots + \beta_d x^d + \epsilon, \\ &= \boldsymbol{X}_d \boldsymbol{\beta}_d + \epsilon, \end{aligned}$$

where

$$\boldsymbol{X}_d = \left[1,\ x,\ x^2, \ldots, x^d\right],$$
$$\boldsymbol{\beta}_d = [\beta_0, \beta_1, \beta_2, \ldots, \beta_d]^T.$$

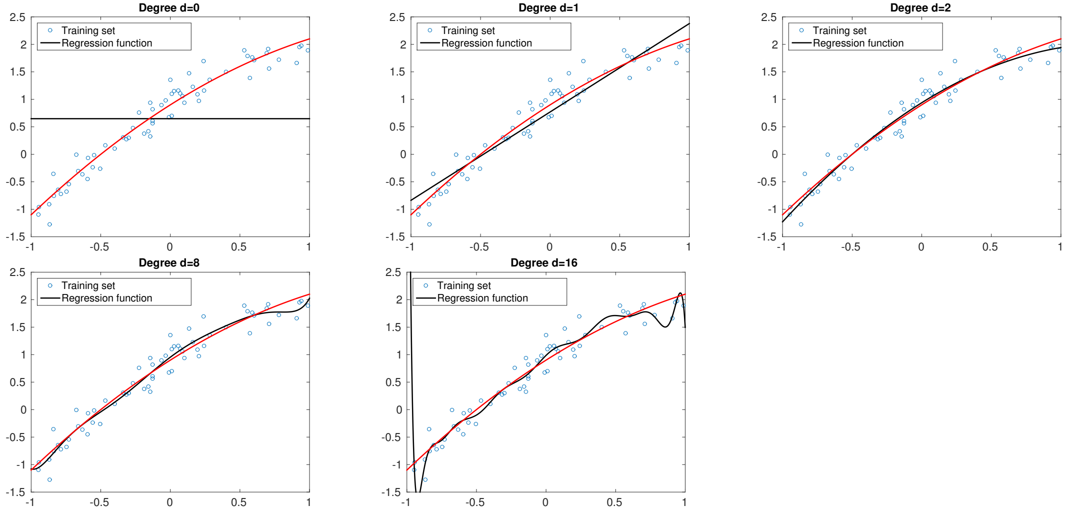
Linear regression: complexity vs stability

Polynomial degree d influence $\leftarrow$ over-fitting issue



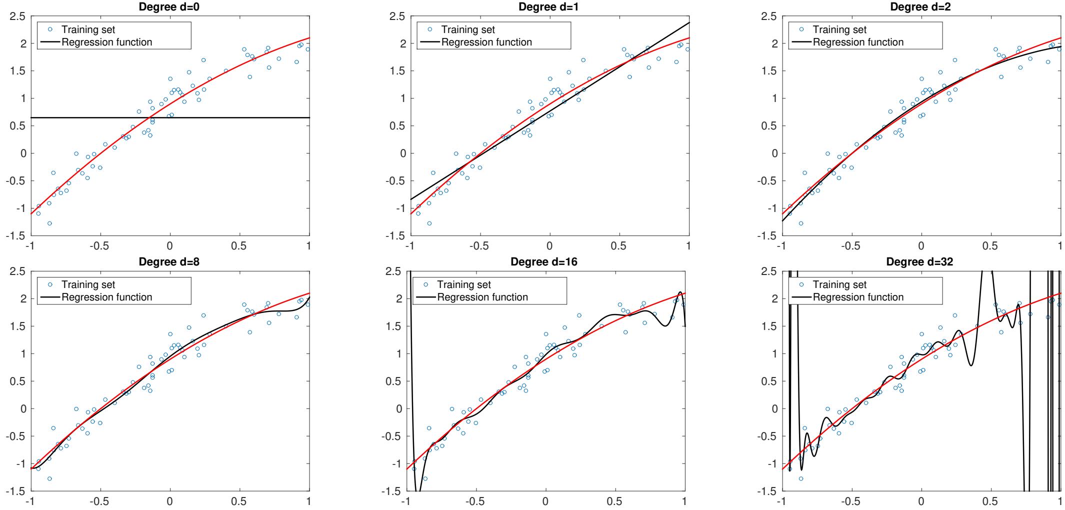
Linear regression: complexity vs stability

Polynomial degree d influence $\leftarrow$ over-fitting issue



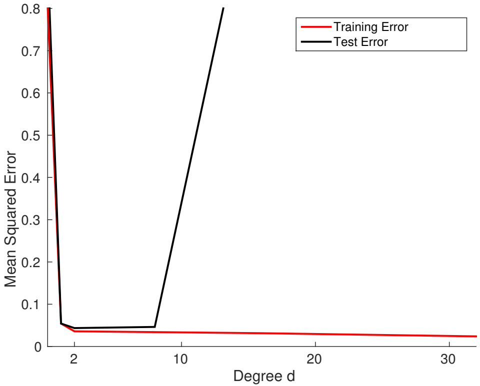
Linear regression: complexity vs stability

Polynomial degree d influence $\leftarrow$ over-fitting issue



Linear Regression: Test error vs Train Error

Error rate vs polynomial order d



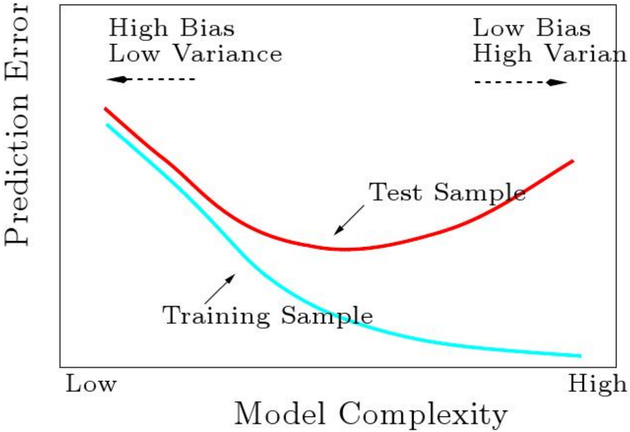
- True error rate (i.e. error rate for test data not used for learning) minimized when $d = 2$ ...
- ... true generative model: order $d = 2$ polynomial (+ white noise)

☞ Training error always decrease with the model complexity. Can't use alone to select the model!

Model Selection

Fundamental trade-off

- too simple model (high bias) $\rightarrow$ under-fitting
- too complex model (high variance) $\rightarrow$ over-fitting



Fundamental Bias-Variance trade-off

if the true model is

$$Y = f(X) + \epsilon,$$

then for any prediction rule $\widehat{f}(X)$, Mean Squared Error (MSE) expresses as

$$E\left[\left(Y - \widehat{f}(x)\right)^2\right] = \mathrm{Var}\left[\widehat{f}(x)\right] + \mathrm{Bias}\left[\widehat{f}(x)\right]^2 + \mathrm{Var}\left[\epsilon\right]$$

- $\mathrm{Var}\left[\epsilon\right]$ is the *irreducible* part
- as the flexibility of $\widehat{f}$ $\nearrow$, its variance $\nearrow$ and the bias $\searrow$

☞ overfitting/underfitting trade-off

Overview of Bias-Variance