

# Data Science for Geosciences

## INTRODUCTION

---

Florent Chatelain <sup>1</sup> Mathieu Fauvel <sup>2</sup>

27-31 January 2020

Toulouse, France

<sup>1</sup>MCF Grenoble INP, GIPSA-lab

<sup>2</sup>CR1 INRA, CESBIO

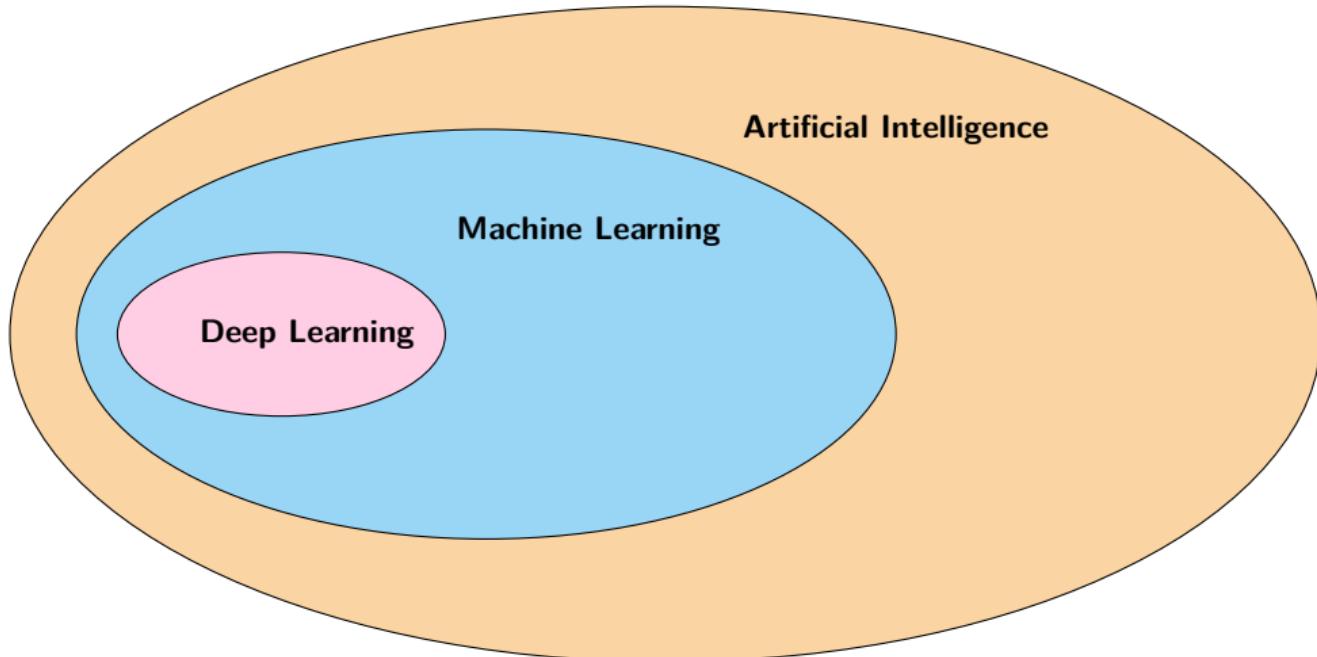
### Florent Chatelain

- 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

### Mathieu Fauvel

- Ph.D. degree in signal and image processing from the National Polytechnic Institute, Grenoble, France, and the University of Iceland, in 2007
- Post-doc position at INRIA - MISTIS Team, 2008-2010
- Assistant Professor (Grenoble, 2007-2008 & Toulouse, 2010-2011)
- Associate Professor at DYNAFOR, National Polytechnic Institute, Toulouse, between 2011-2018.
- Since 2018, Research (CR1) at CESBIO, INRA.
- Research interests are: machine learning for environmental/ecological monitoring

**What is machine learning?**



Taken from [https://www.geospatialworld.net/blogs/difference-between-ai%EF%BB%BF-machine-learning-and-deep-learning/](https://www.geospatialworld.net/blogs/difference-between-ai-machine-learning-and-deep-learning/)

*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**

## 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

## 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**

## 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**

## 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

### Type of data: qualitatives / ordinales / quantitatives variables

- Text: strings
- Speech: time series
- Images/videos: 2/3d dependences
- Networks: graphs
- Games: interaction sequences
- ...

### Big data (volume, velocity, variety, veracity)

Data are available without having decided to collect them!

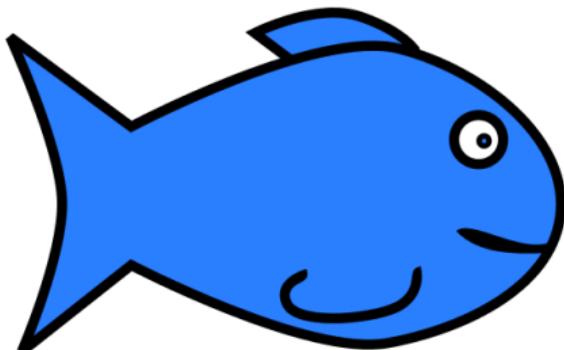
- importance of preprocessings (cleaning up, normalization, coding,...)
- importance of a good representation : from raw data to vectors

## Generalize

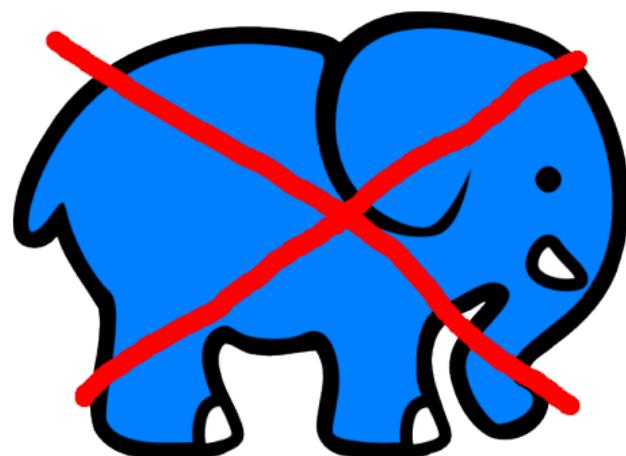
- Perform well (minimize P) on new data (fresh data, i.e. unseen during learning)
- ☞ Derive good ( $P/\text{error rate}$ ) prediction functions

## Generalize

- Perform well (minimize P) on new data (fresh data, i.e. unseen during learning)
- ☞ Derive good ( $P/\text{error rate}$ ) prediction functions



A fish



A fish

## Reference books

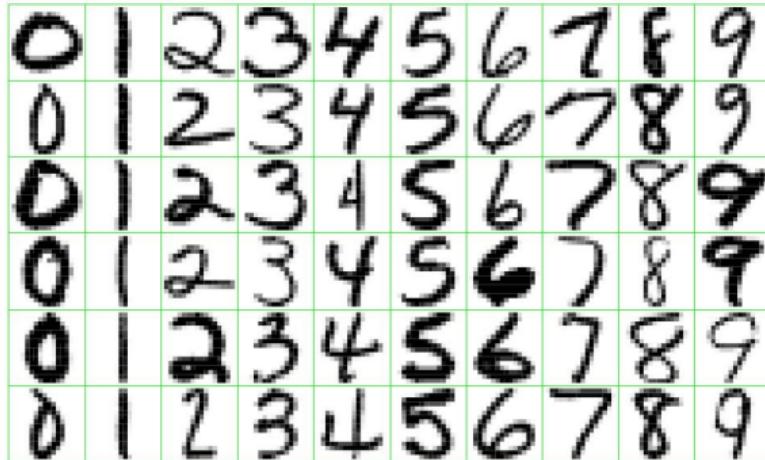
-  Trevor Hastie, Robert Tibshirani et Jerome Friedman (2009), **The Elements of Statistical Learning (2nd Edition)**, *Springer Series in Statistics*
-  Christopher M. Bishop (2007), **Pattern Recognition and Machine Learning**, *Springer*
-  Kevin P. Murphy (2012), **Machine Learning: a Probabilistic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

## Supplementary materials, datasets, online courses, ...

-  <http://www-stat.stanford.edu/~tibs/ElemStatLearn/>
-  <https://www.cs.ubc.ca/~murphyk/MLbook/>
-  <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](https://scikit-learn.org/stable/auto_examples/index.html) *Examples from the sklearn library*

## **Examples**

## Recognition of handwritten digits (US postal envelopes)



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

Supervised classification

## 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

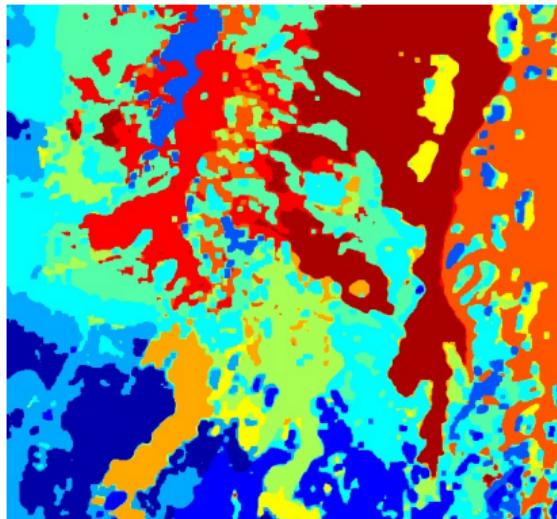
Dear George,  
Could you please send me the report #1248 on  
the project advancement?  
Thanks in advance.

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

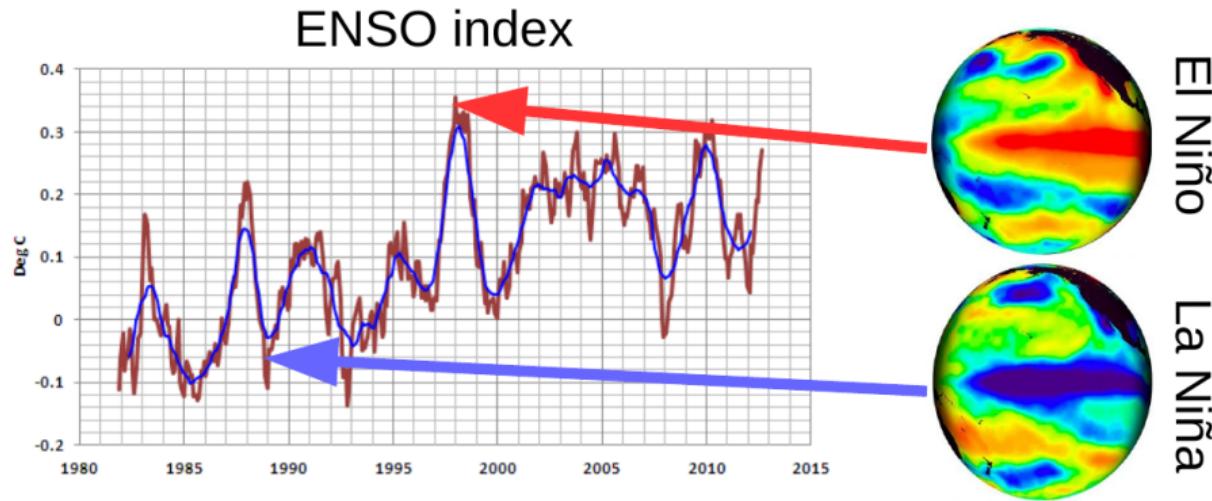
## Recognition of Hekla Volcano landscape, Iceland



- Predict the class of landscape  $\in \{ \text{Lava 1970}, \text{Lava 1980 I}, \text{Lava 1980 II}, \text{Lava 1991 I}, \text{Lava 1991 II}, \text{Lava moss cover}, \text{hyaloclastite formation}, \text{Tephra lava}, \text{Rhyolite}, \text{Scoria}, \text{Firn-glacier ice}, \text{Snow} \}$  from digital remote sensing images

supervised or unsupervised classification

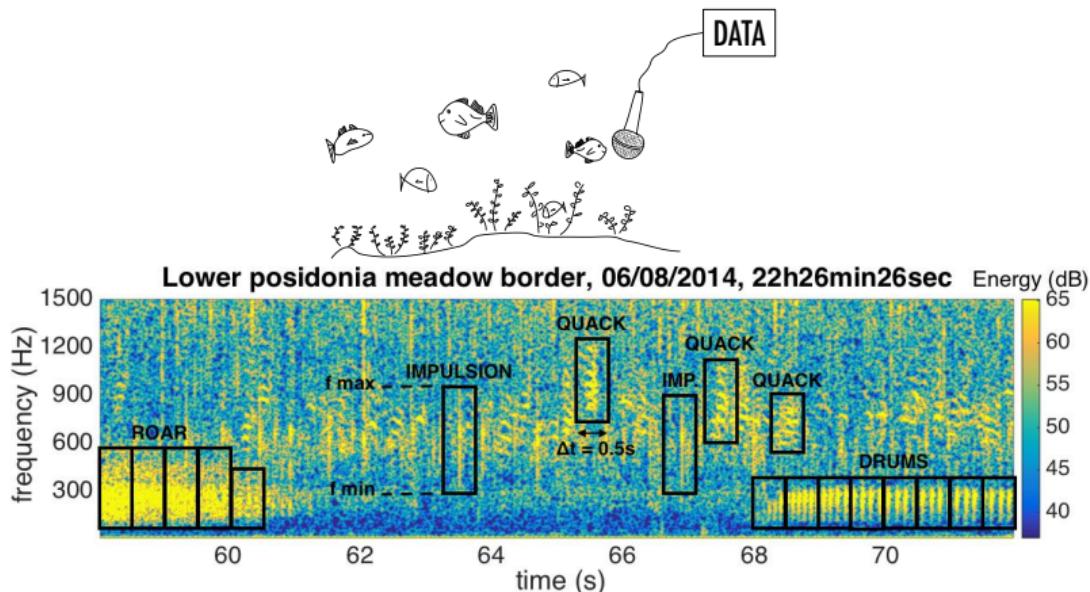
## Prediction of El Niño southern oscillation



- 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

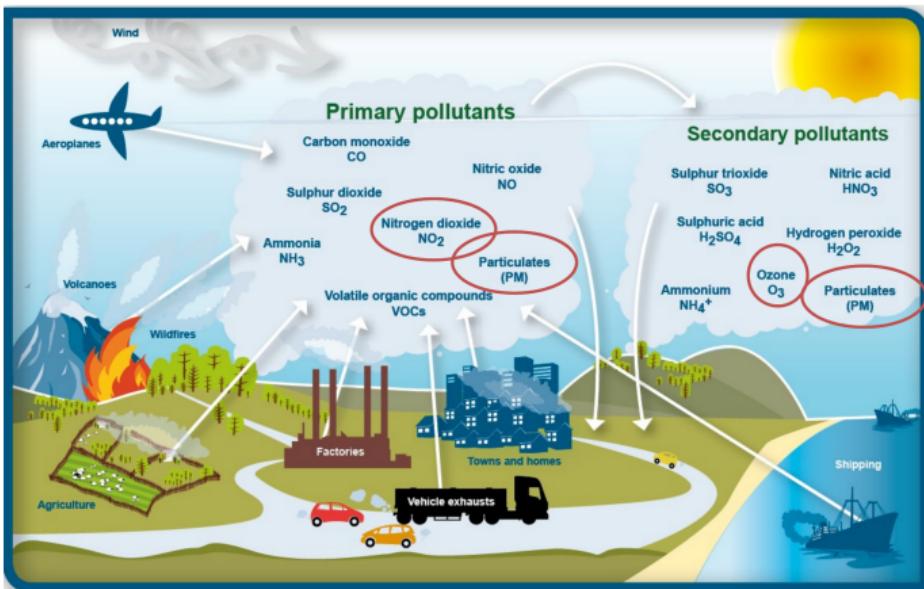
# Recognition of fish sounds



- Predict the class of underwater sounds (roar, quack, drums, impulsion) from times series recorded by hydrophones ( $f_s = 156\text{kHz}$ )

supervised or unsupervised classification

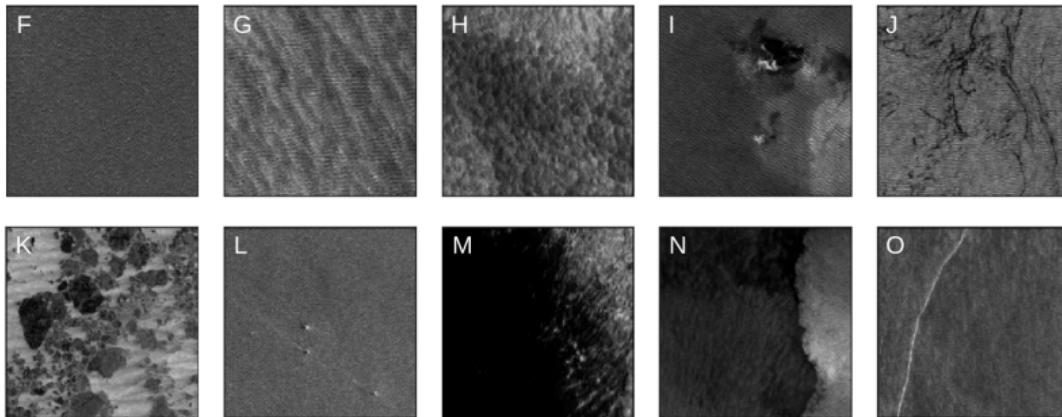
# Prediction of pollutant concentrations



- Predict pollutant concentrations ( $O_3, N_{O_2}, PM10, PM2.5$ ) at time  $D_0+1, +2, +3$  from hourly measures timeseries + weather data + chemistry based forecasting models

supervised regression/classification (pollution alert or not)

## Recognition of climate-ocean events



- ☞ Predict the classes of SAR images of the ocean (convective cells in I, sea ice in K, weather front in N,...) to detect climate-ocean events from water surface roughness

supervised or unsupervised classification

## **Basics**

## Variable terminology

- Observed data referred to as *input* variables, *predictors* or *features*:  $X$
- Data to predict referred to as *output* variables, or *responses*:  $Y$

## Type of prediction problem: regression vs classification

Depending on the type of the *output* variables

- When  $Y$  are **quantitative** data (e.g. ENSO intensity index values): **regression**
- When  $Y$  are **categorical** data (e.g. handwritten digits  $Y \in \{0, \dots, 9\}$ ): **classification**

Two very close problems

## Assumptions

- 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)

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

## Prediction rule

Function of prediction / rule of classification  $\equiv$  function  $\hat{f}: \mathcal{X} \rightarrow \mathcal{Y}$  to get predictions of new elements  $Y$  given  $X$

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

## Supervised or unsupervised learning

---

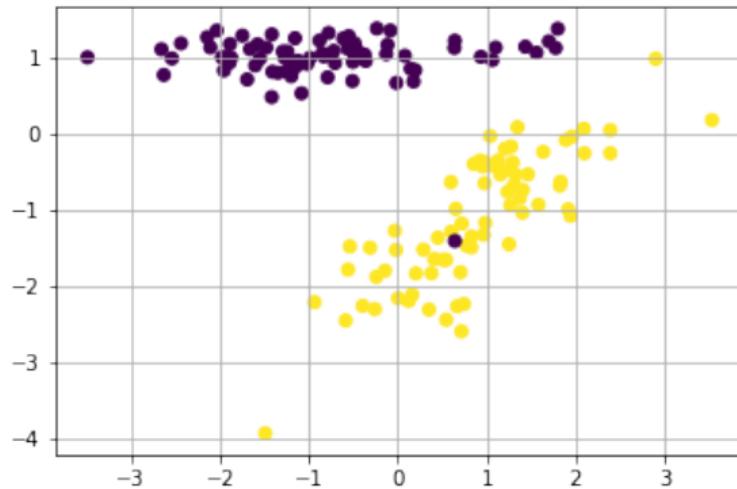
Training set  $\equiv$  available sample  $\mathcal{T}$  to learn the prediction rule  $f$

For a sized  $n$  training set, different cases:

- **Supervised learning:**  $\mathcal{T} \equiv \{(X_1, Y_1), \dots, (X_n, Y_n)\}$  are available
- **Unsupervised learning:**  $\mathcal{T} \equiv (X_1, \dots, X_n)$  are available only
- **Semi-supervised:** mixed scenario (often encountered in practice, but less information than in the supervised case)

## **Toy Example**

## Binary classification



## Simple linear model for classification

We seek a prediction model based on the linear regression of the outputs  $Y \in \{-1, 1\}$  :

$$Y = \beta_1 X_1 + \beta_2 X_2 + \epsilon,$$

where  $\beta = (\beta_1, \beta_2)^T$  is a 2D unknown parameter vector

**Learning problem  $\Leftrightarrow$  Estimation of  $\beta$**

*Least Squares Estimator*  $\hat{\beta} = (\hat{\beta}_1, \hat{\beta}_2)^T$ : minimize the training error rate (quadratic cost sense)

$$RSS(\beta) = \sum_{i=1}^N (Y_i - \beta_1 X_{i,1} - \beta_2 X_{i,2})^2$$

**Classification rule based on least squares regression**

$$f(X) = \begin{cases} 1 & \text{if } \hat{Y} = \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2 \geq 0, \\ -1 & \text{otherwise} \end{cases}$$

Notebook

## Model complexity

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

### Linear classification: model order $p$

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

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

where

$$\begin{aligned} \mathbf{X}_d &= [1, x, x^2, \dots, x^d], \\ \boldsymbol{\beta}_d &= [\beta_0, \beta_1, \beta_2, \dots, \beta_d]^T. \end{aligned}$$

Notebook

### Error rate vs polynomial order $d$

[Notebook](#)

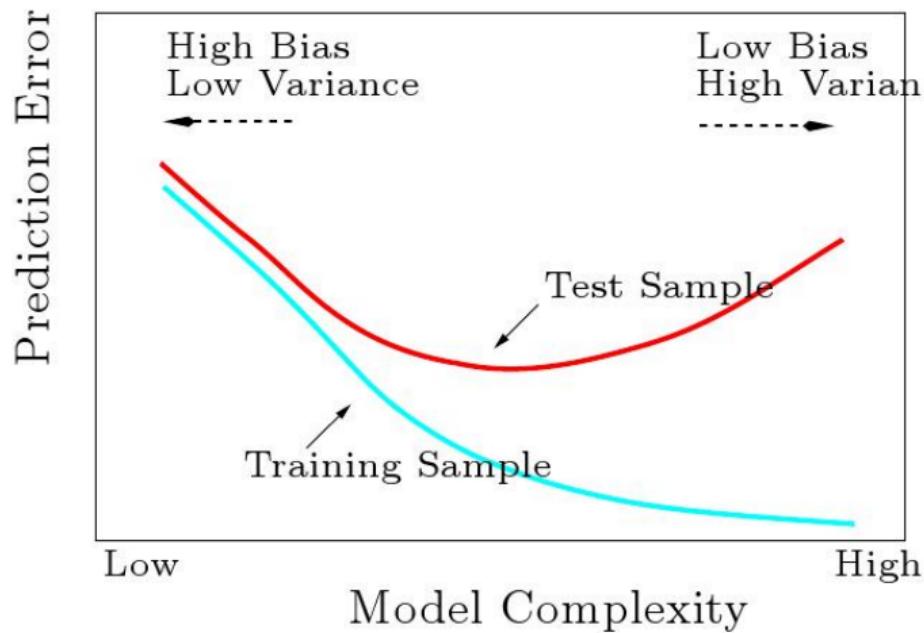
- Training error rate (i.e. error rate for train data used for learning) minimized when  $d = 19$
- True error rate (i.e. error rate for test data not used for learning) minimized when  $d = 5 \dots$

☞ 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) → under-fitting
- Too complex model (high variance) → over-fitting



## Fundamental Bias-Variance trade-off

---

If the true model is

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

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

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

- $\text{Var} [\epsilon]$  is the *irreducible* part
- as the flexibility of  $\hat{f}$  ↗, its variance ↗ and the bias ↘
  - ☞ overfitting/underfitting trade-off

**Thank you for your attention**

This work is licensed under a Creative Commons “Attribution-ShareAlike 4.0 International” license.

