ELEC2870 - Machine learning: regression and dimensionality reduction

Introduction

Michel Verleysen

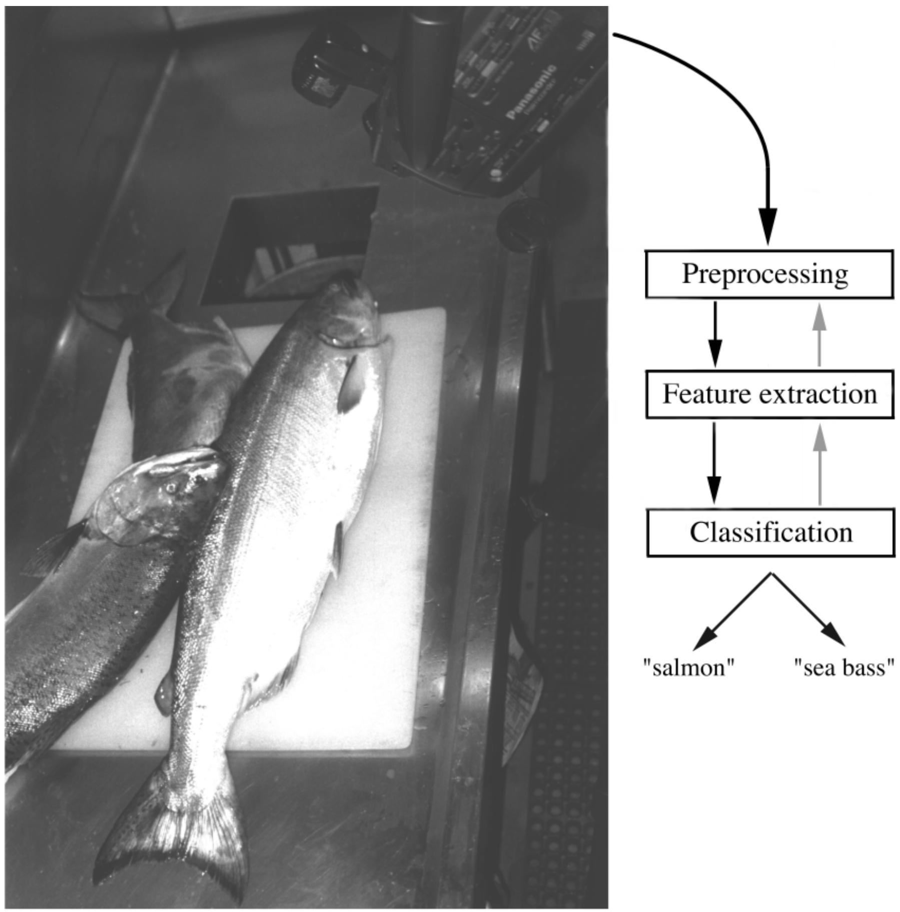
Machine Learning Group
Université catholique de Louvain
Louvain-la-Neuve, Belgium
michel.verleysen@uclouvain.be

Outline

- Machine learning
- Artificial neural networks
- Overfitting
- The curse of dimensionality
- Machine learning tasks

Machine learning

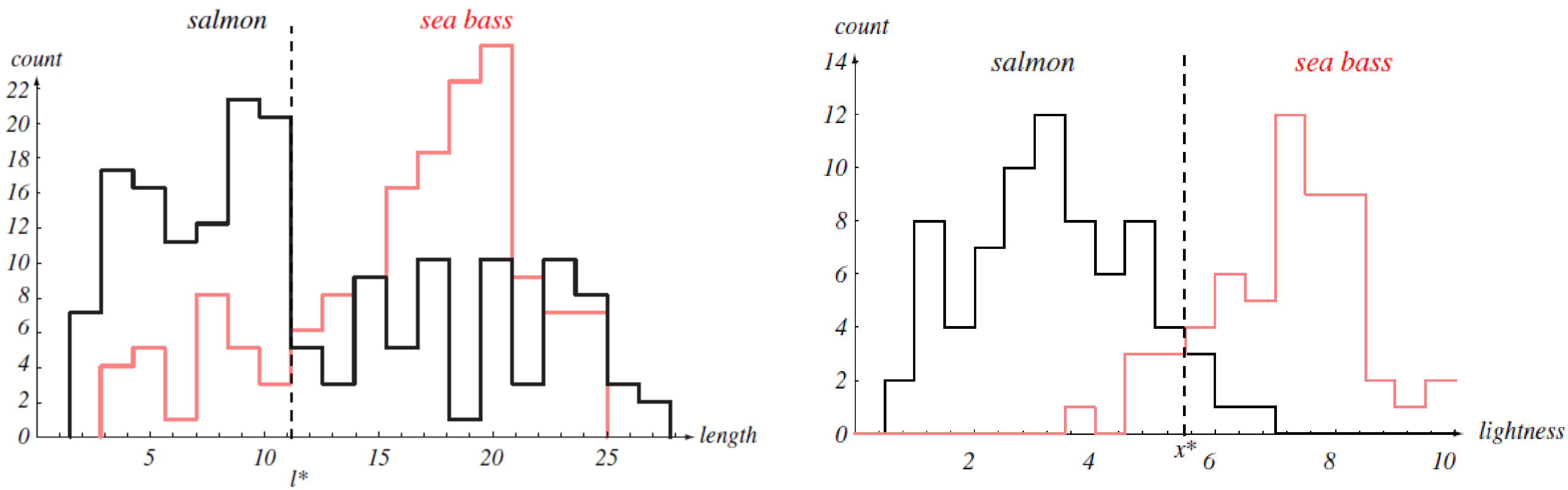
- An example: discriminating between images of fishes



From: Duda et al., Pattern Classification, 2nd ed., Wiley, 2001

Features

- Features are fundamental elements in machine learning
- Fishes: features may be length, lightness, etc.

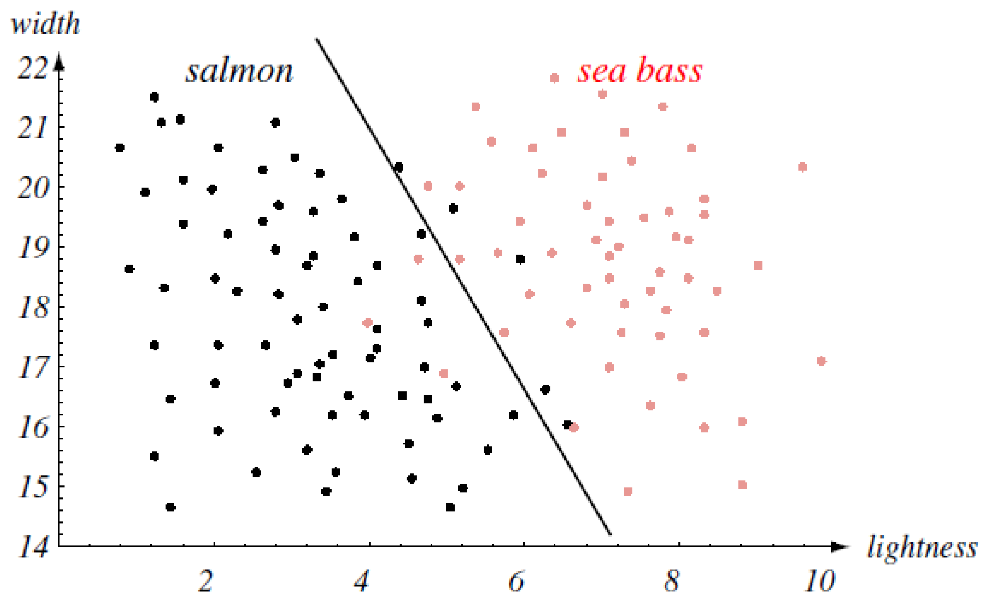


- No single feature is *discriminant* !

From: Duda et al., Pattern Classification, 2nd ed., Wiley, 2001

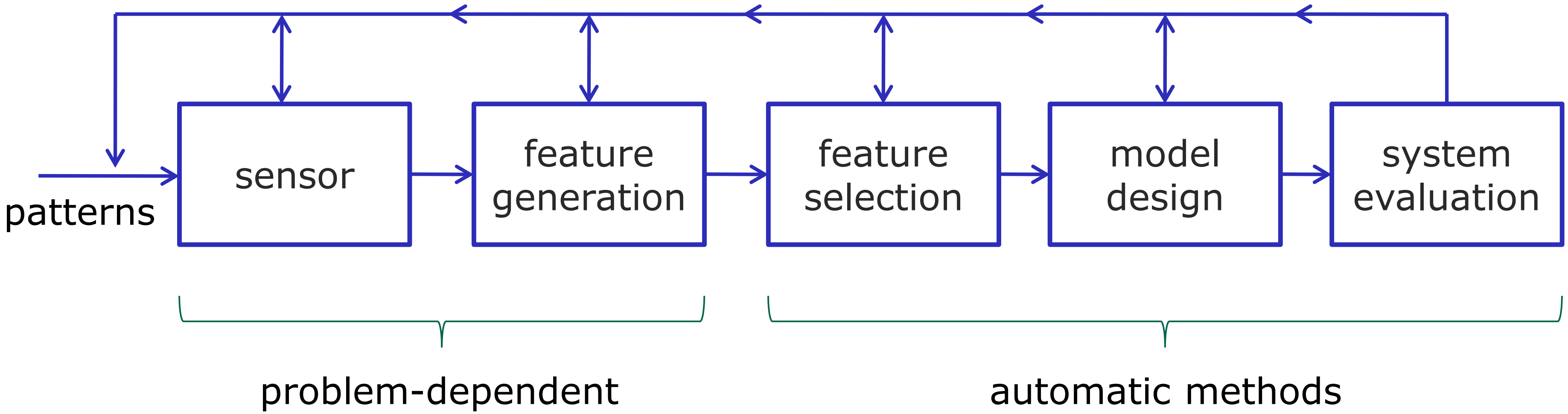
Features

- Solution: working with *several features*



From: Duda et al., Pattern Classification, 2nd ed., Wiley, 2001

Data analysis system

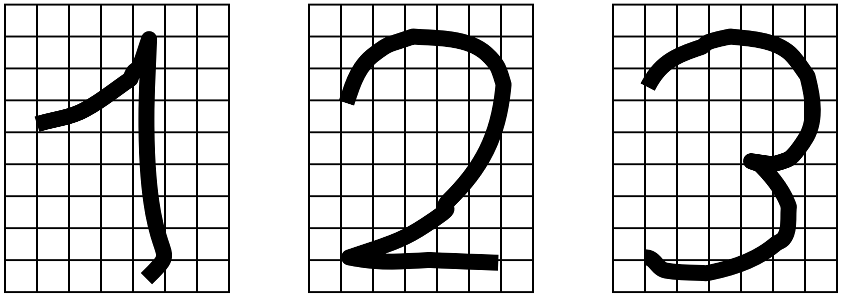


Adapted from: Theodoridis et al., Pattern Recognition, 4th ed., Academix Press, 2009

Why machine learning ?

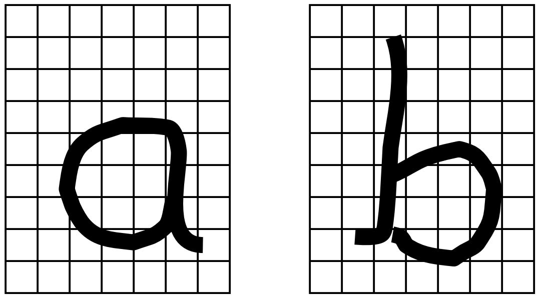
- Hypothetical problem of Optical Character Recognition:
  - If: - image 256 x 256 pixels
    - 8-bits pixel values (256 gray levels)

    then: - 10^{158000} different images
  - Necessity to work with *features* (you cannot build and store a complete truth table)

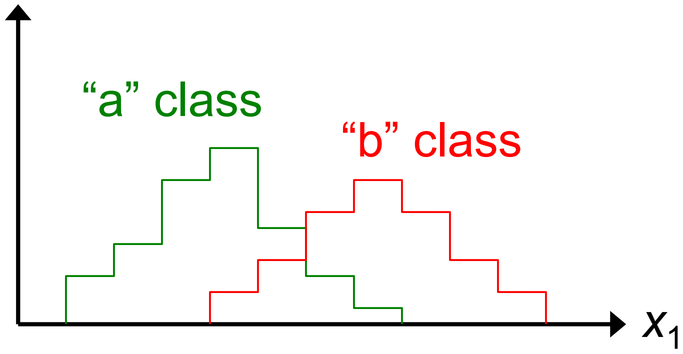


Feature Extraction

- Example of feature in OCR:
  ratio height/width (x_1) of the character

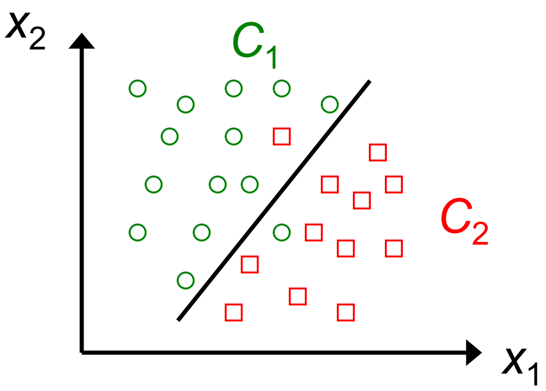


- Histogram of feature value



Multi-dimensional Features

- Necessity to classify according to *several*, but *not too much* features



  - Several: because of the overlap in histogram with 1 feature
  - Not too much: because classification methods are easier in small dimensional spaces

Machine learning: a discipline?

- Machine learning is at the cross-road of many disciplines:
  - artificial neural networks
  - statistics
  - applied mathematics
  - computer science
- It consists in
  - algorithms
  - for analyzing data, signals, etc.
  - to extract relevant information
  - from real data
  - with an engineering view

Outline

- Machine learning
- Artificial neural networks
- Overfitting
- The curse of dimensionality
- Machine learning tasks

Artificial neural networks

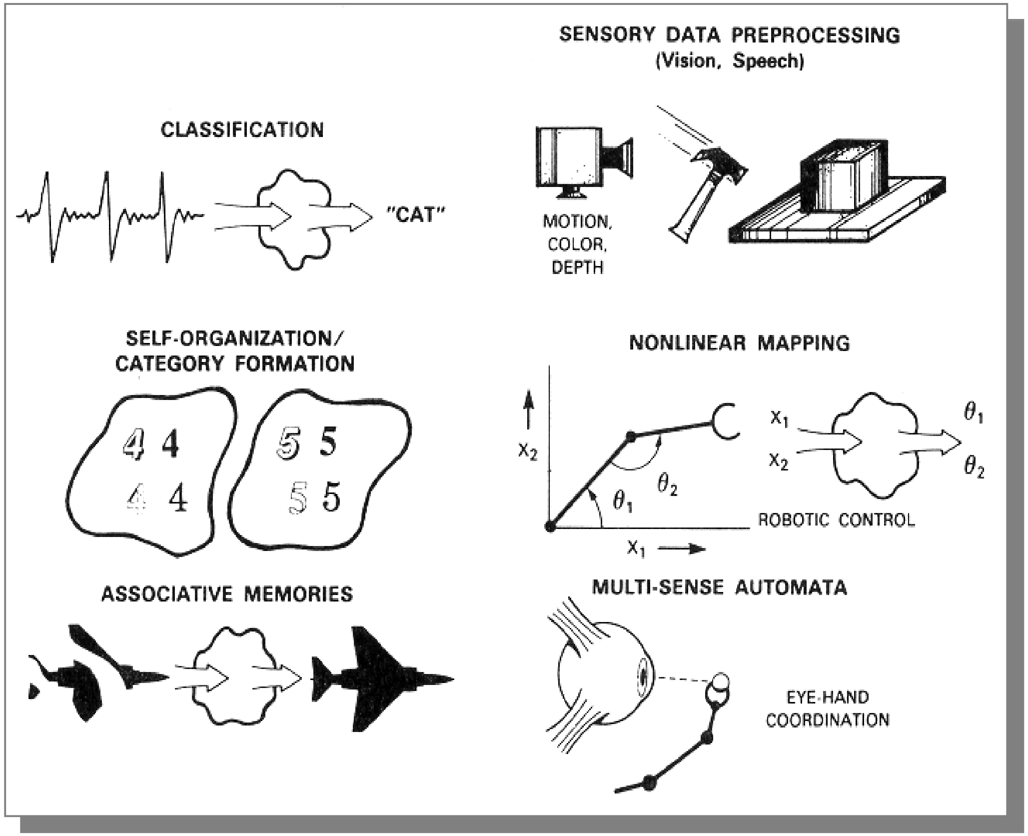
Von Neumann's computer	(Human) brain
determinism	fuzzy behaviour
sequence of instructions	parallelism
high speed	slow speed
repetitive tasks	adaptation to situations
programming	learning
uniqueness of solutions	different solutions
ex: matrix product	ex: face recognition

- Need for other computing paradigms!

From: some book?

Perceptive tasks

- Some examples of tasks that make use of « perception »:
  - face recognition
  - time-series prediction
  - process identification
  - process control
  - optical character recognition
  - adaptive filtering
  - etc.



From "DARPA Neural Network Study", 1988

Historical background

- Some (not necessarily well-chosen…) milestones

1940 – 1965	Hebb	biological learning rule
	McCulloch & Pitts	binary decision units
	Rosenblatt	Perceptron - learning
1969	Minsky & Papert	limits to Perceptrons
1974	Werbos	back-propagation
1980s	Hopfield	recurrent networks
	Parker, LeCun, Rumelhart, McClelland	back-propagation
	Kohonen	Self-Organizing Maps
1990s	Vapnik	Support Vector Machines
	Many…	Feature selection, model selection, evaluation,…

Artificial Neural Networks

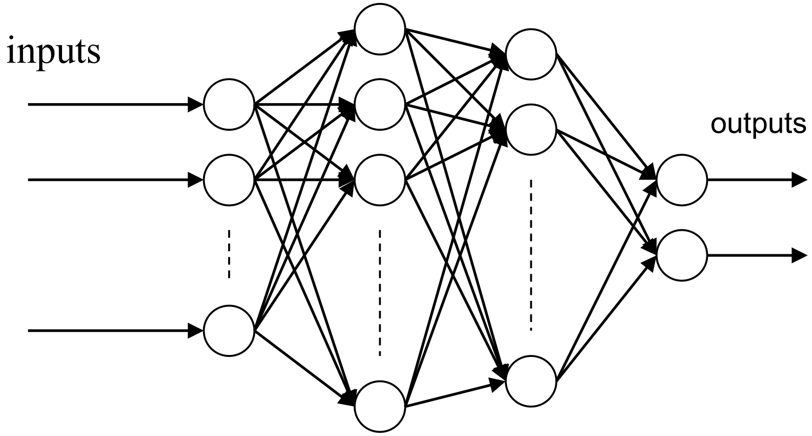
- Artificial neural networks ARE NOT :
  - an attempt to understand biological systems
  - an attempt to reproduce the behavior of a biological system

- Artificial neural networks ARE :
  - a set of tools aimed to solve "perceptive" problems ("perceptive" <-> "rule-based")

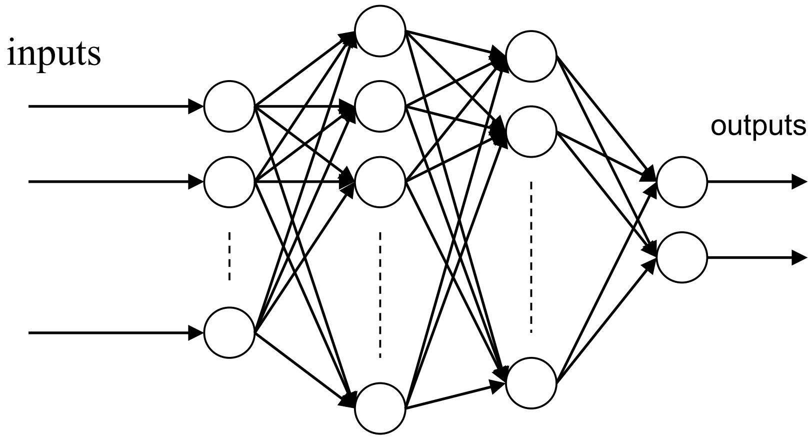
At least in the framework of this lecture, i.e. "Neural Computation"…

Why “Artificial neural networks” ?

- Structure: many computation units in parallel (McCulloch & Pitts 1943)
- Learning rule (adaptation of parameters): sometimes similar to Hebb’s rule (1949)
- And that’s all !



What does « learning » mean?



- Give - many - examples (*input-output pairs*, or *training samples*)
- Compute the parameters to fit the examples
- Test the result!

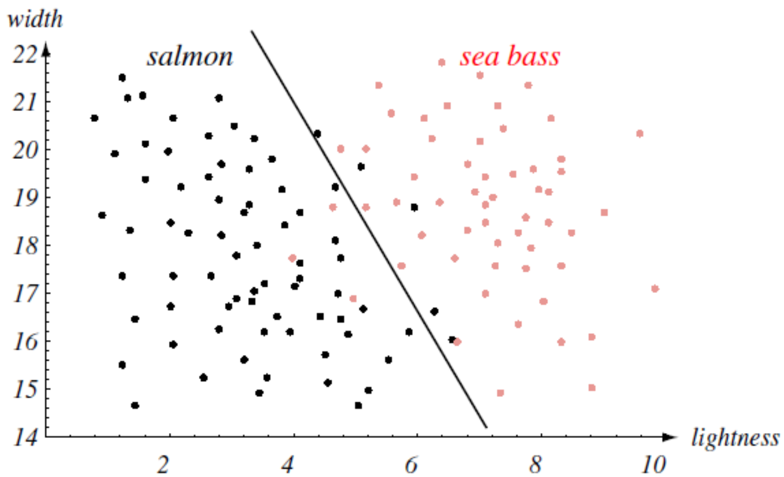
Outline

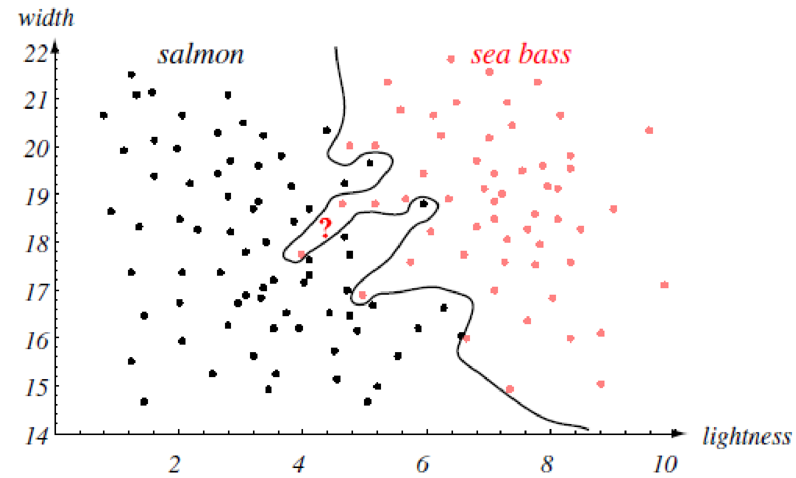
- Machine learning
- Artificial neural networks
- Overfitting
- The curse of dimensionality
- Machine learning tasks

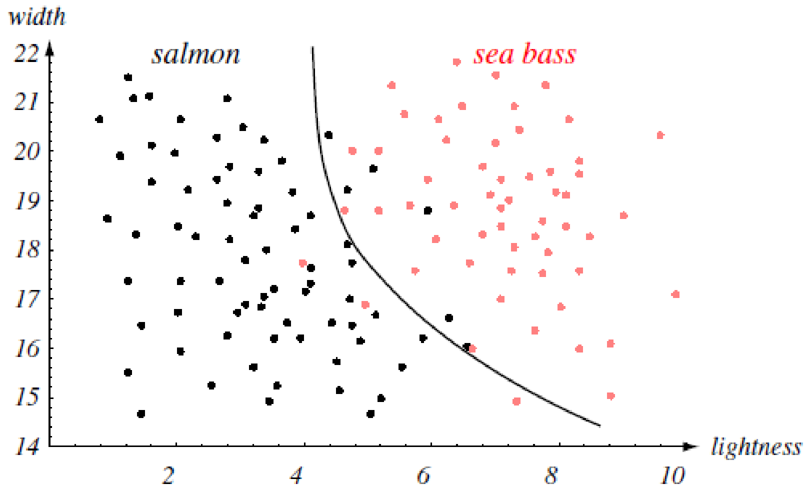
Overfitting

- Overfitting is learning (a finite number of) data so well, that the model is not useful anymore
- Overfitting can occur in classification and regression
- It is one of the main important concerns in learning!

Overfitting in classification



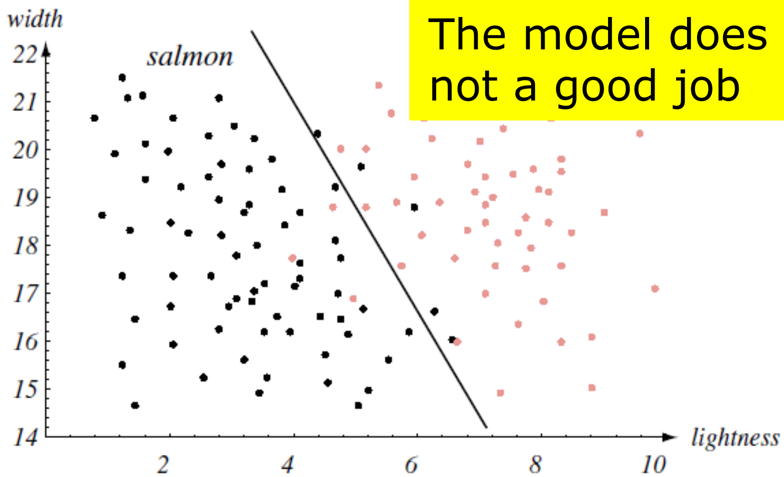




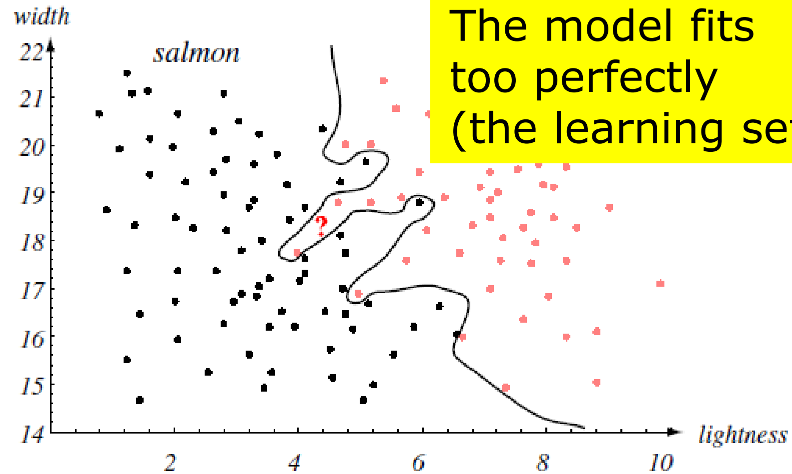
- Which of the 3 models is « best »?

From: Duda et al., Pattern Classification, 2nd ed., Wiley, 2001

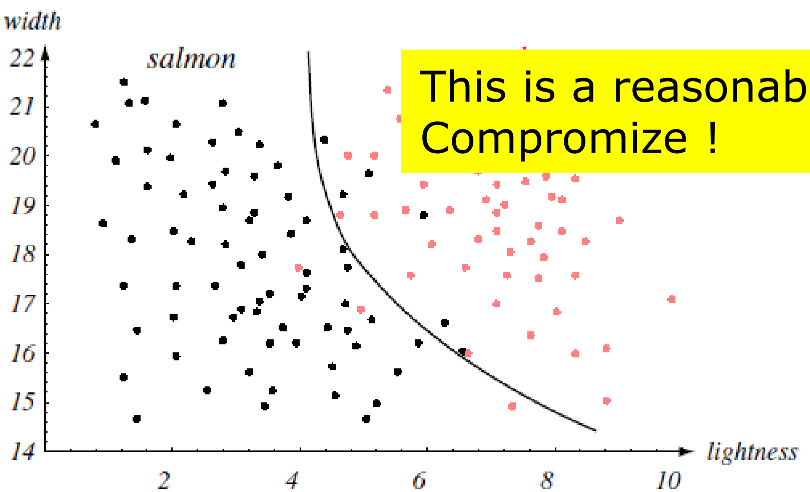
Overfitting in classification



The model does not a good job



The model fits too perfectly (the learning set)



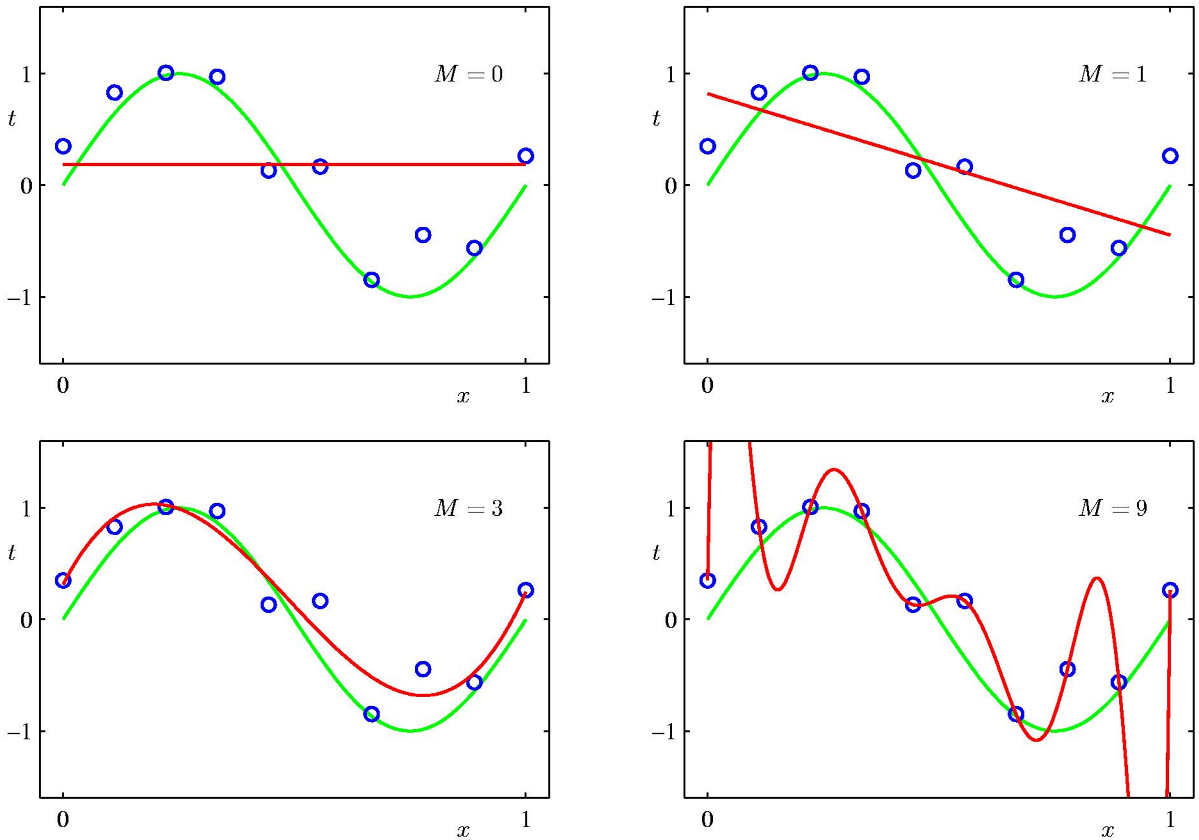
This is a reasonable Compromize !

- Which of the 3 models is « best »?

From: Duda et al., Pattern Classification, 2nd ed., Wiley, 2001

Overfitting in regression

- The following example is about regression with polynomials of order *M* (but the overfitting risk is the same, whatever is the type of model)



Models with *M=0 and M*=1 are poor,

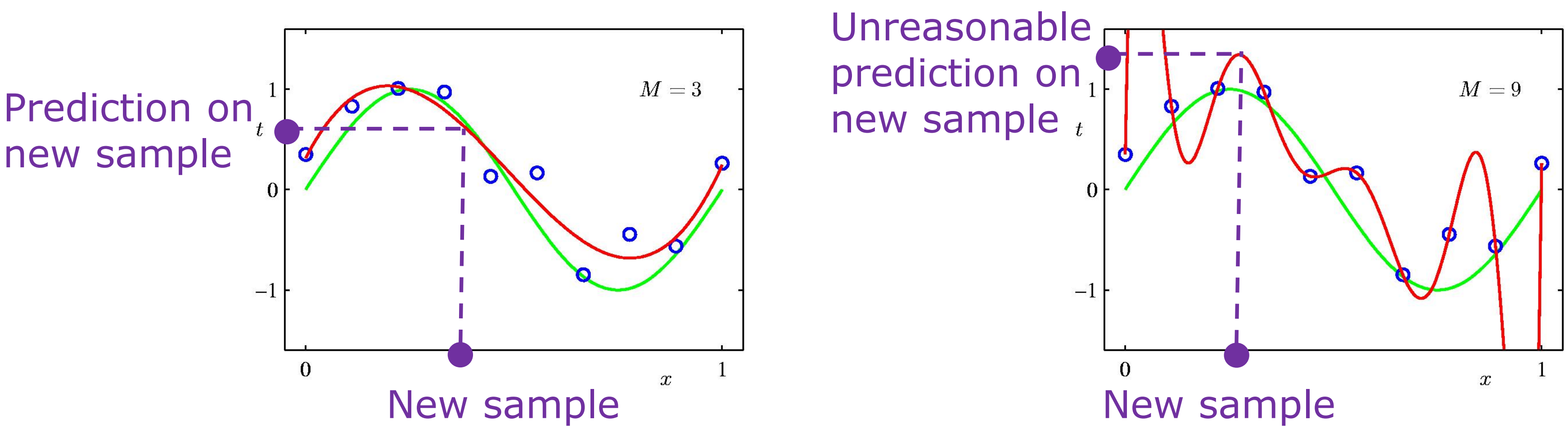
model with *M*=9 overfits,

model with *M*=3 is a good compromize

From: C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

Overfitting and generalization

- Why fitting « as well as possible » is not a good idea?
  - Because nobody cares about having a model that reproduces the output on known examples (= training set)
  - The real goal is to get a model that performs well on *new* samples!



- Don't forget that only the samples (blue dots) are known!

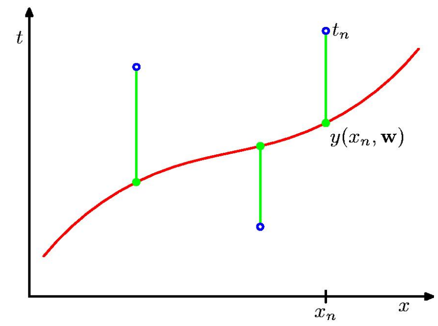
Adapted from: C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

Overfitting and generalization

- Overfitting occurs when the model is *too complex* (with respect to…)

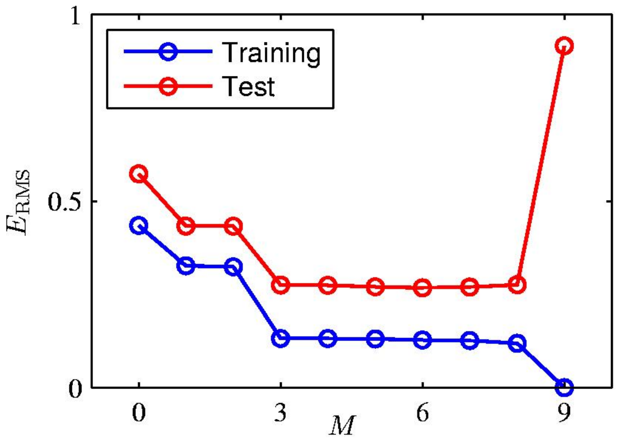
- Define the sum-of-squares error function

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2$$



- The errors on the
  - training set
  - test set

  may be very different!



From: C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

Overfitting and generalization

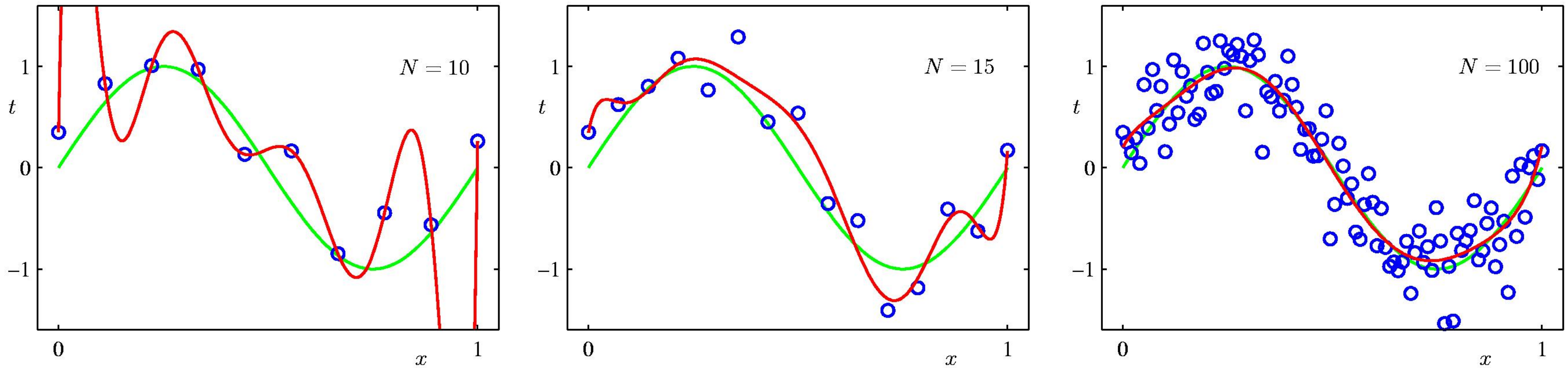
- Overfitting often results in very large model parameters (they « compensate » eachother)

	$M = 0$	$M = 1$	$M = 3$	$M = 9$
$w_0^\star$	0.19	0.82	0.31	0.35
$w_1^\star$		-1.27	7.99	232.37
$w_2^\star$			-25.43	-5321.83
$w_3^\star$			17.37	48568.31
$w_4^\star$				-231639.30
$w_5^\star$				640042.26
$w_6^\star$				-1061800.52
$w_7^\star$				1042400.18
$w_8^\star$				-557682.99
$w_9^\star$				125201.43

From: C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

Overfitting and generalization

- Overfitting decreases with the number of samples available for learning
- Polynomial model of order M=9:

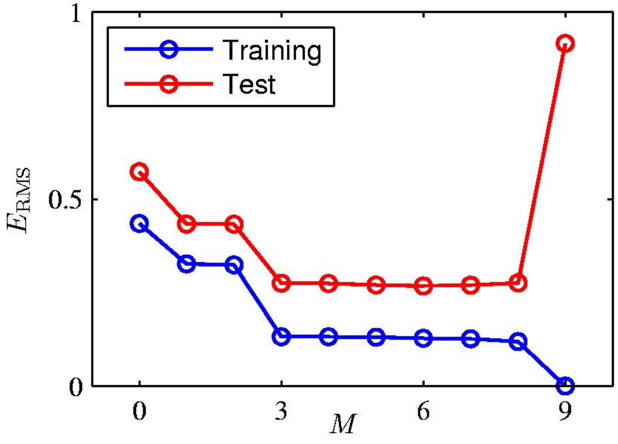


- The risk of overfitting
  - increases with the complexity of the model
  - decreases with the size of the learning set

From: C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

Avoiding overvitting

- Two main directions for avoiding overfitting:



1. Limit the model complexity (here *M*)
   - How? By trial and error, with several values of *M*
   - Requires a *validation* set (test: wrong name!)

2. Using *regularization*:

$$\widetilde{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n,\mathbf{w}) - t_n\}^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2$$

   - λ penalizes the error by some complexity measure
   - Other complexity measures are possible
   - Requires a *validation* set to fix λ

- See chapter on model selection!

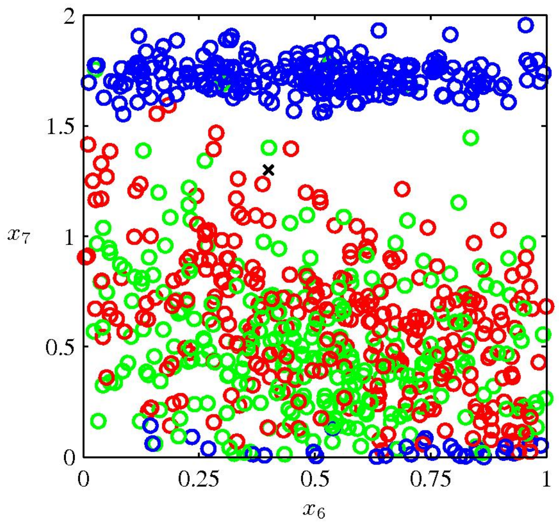
From: C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

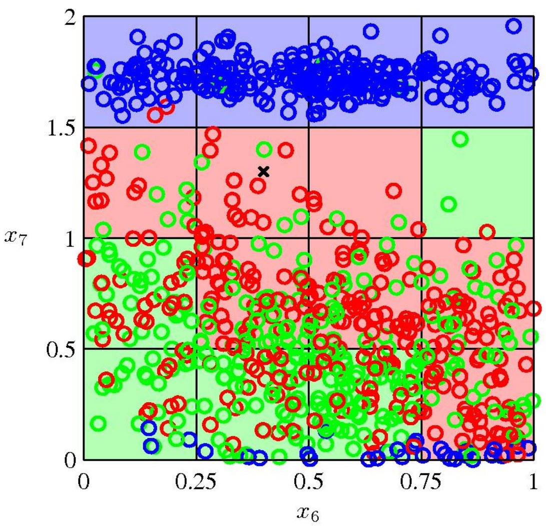
Outline

- Machine learning
- Artificial neural networks
- Overfitting
- The curse of dimensionality
- Machine learning tasks

The curse of dimensionality

- A simple, 2-*D* classification problem
- A simple algorithm:
  - Cut the space in small boxes
  - Attribute a class to each box according to the majority of samples

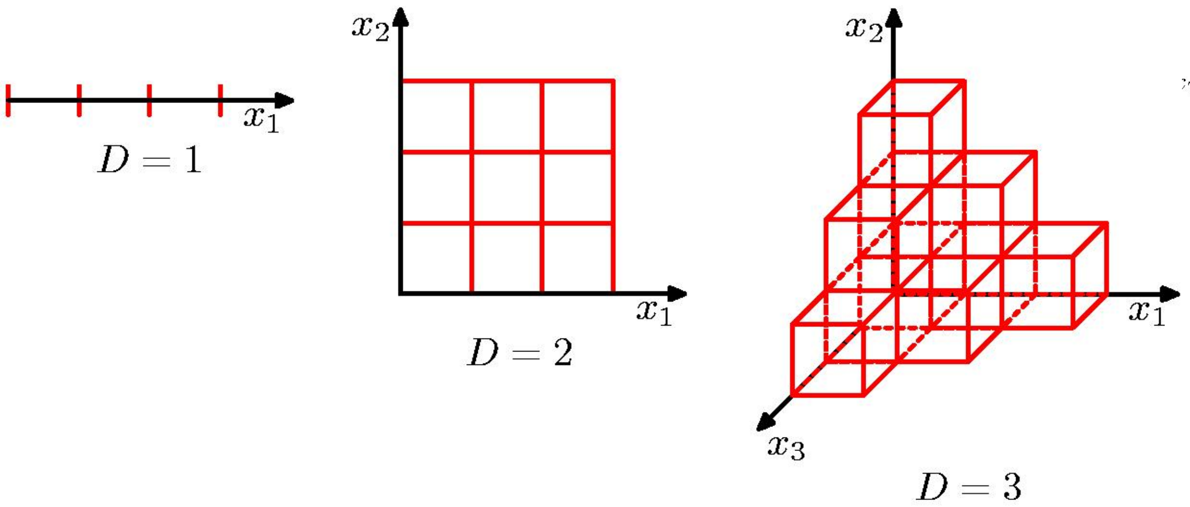


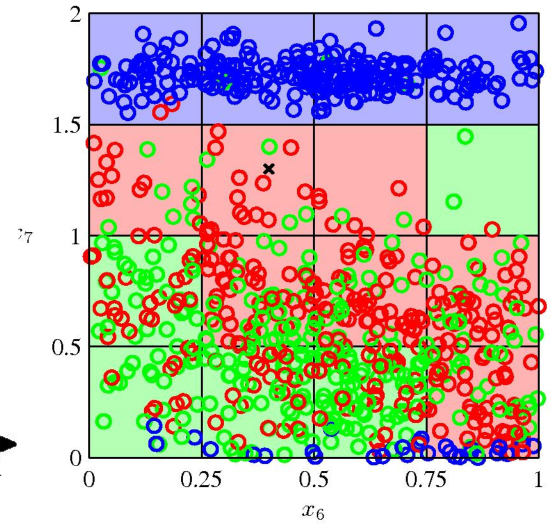


From: C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

The curse of dimensionality

- *The* problem: the number of boxes grows exponentially with the dimension of the space!





- In practical settings we *never* have enough data…

- See chapter on feature selection

From: C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

Multivariate analysis

- Never forget that machine learning is used for analyzing

  high-dimensional data

  (this is multivariate analysis)

- For (very) low-dimensional data, other techniques may be more appropriate: polynoms, splines, etc.

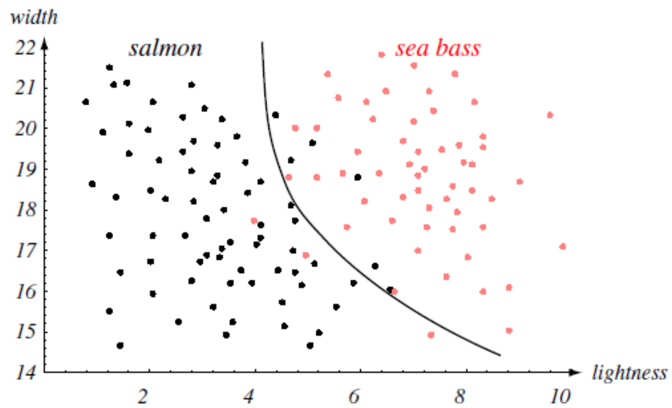
- High-dimensional often means « more than 3 »…

Outline

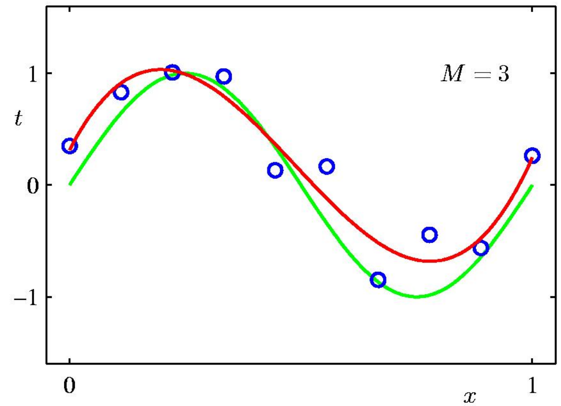
- Machine learning
- Artificial neural networks
- Overfitting
- The curse of dimensionality
- Machine learning tasks

Machine learning tasks

- Classification



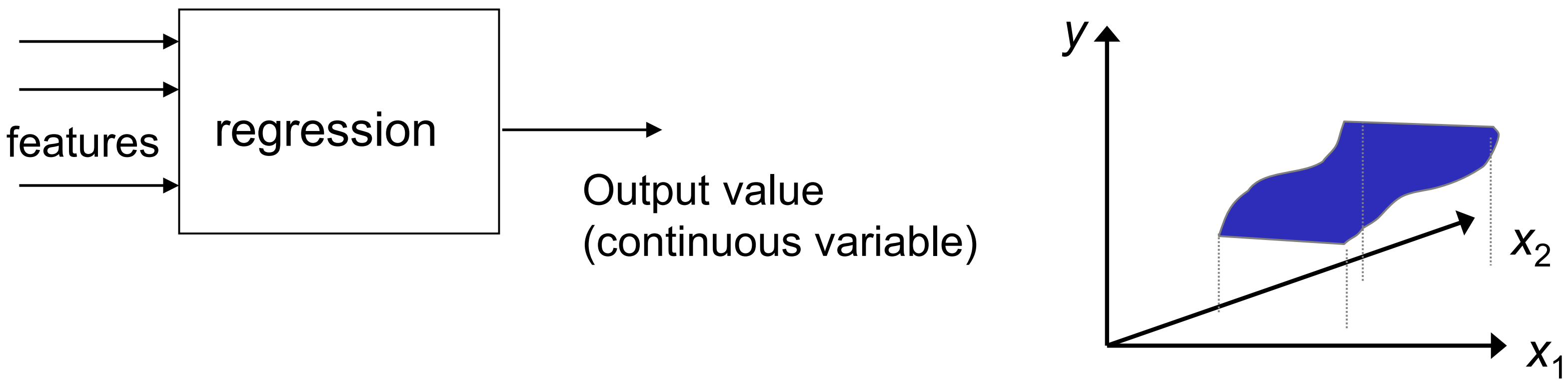
- Regression



From: previously cited books
(same figures)

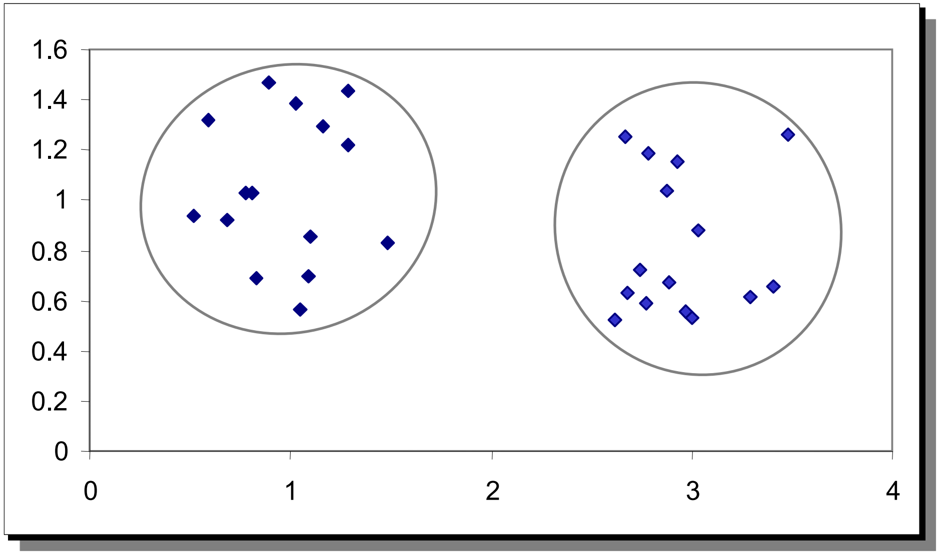
Machine learning tasks

- Classification and regression are not so different:



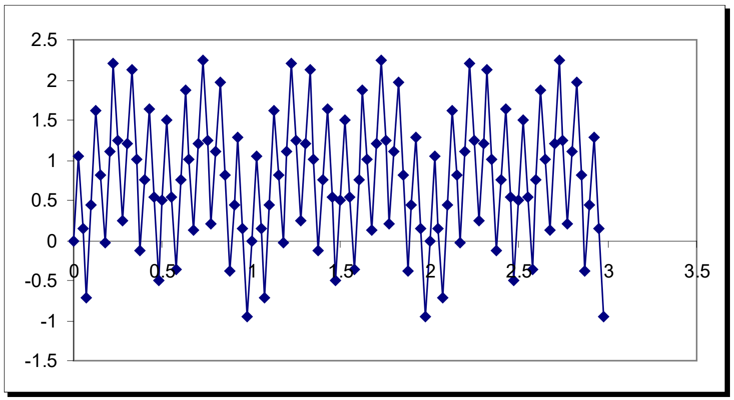
Machine learning tasks

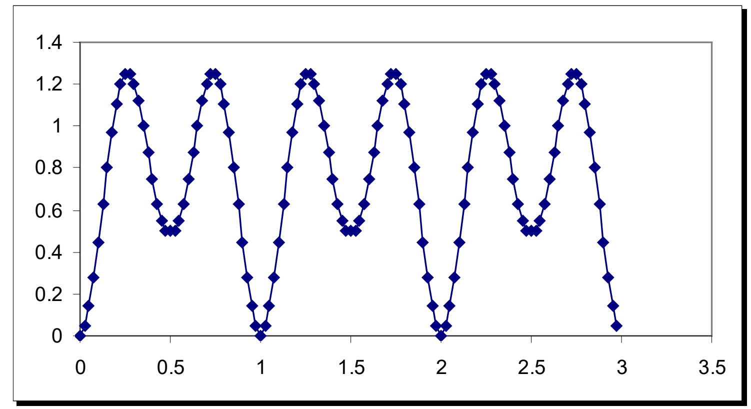
- Clustering



is about finding natural *groups (=clusters)* in data, without class label information

- Adaptive filtering

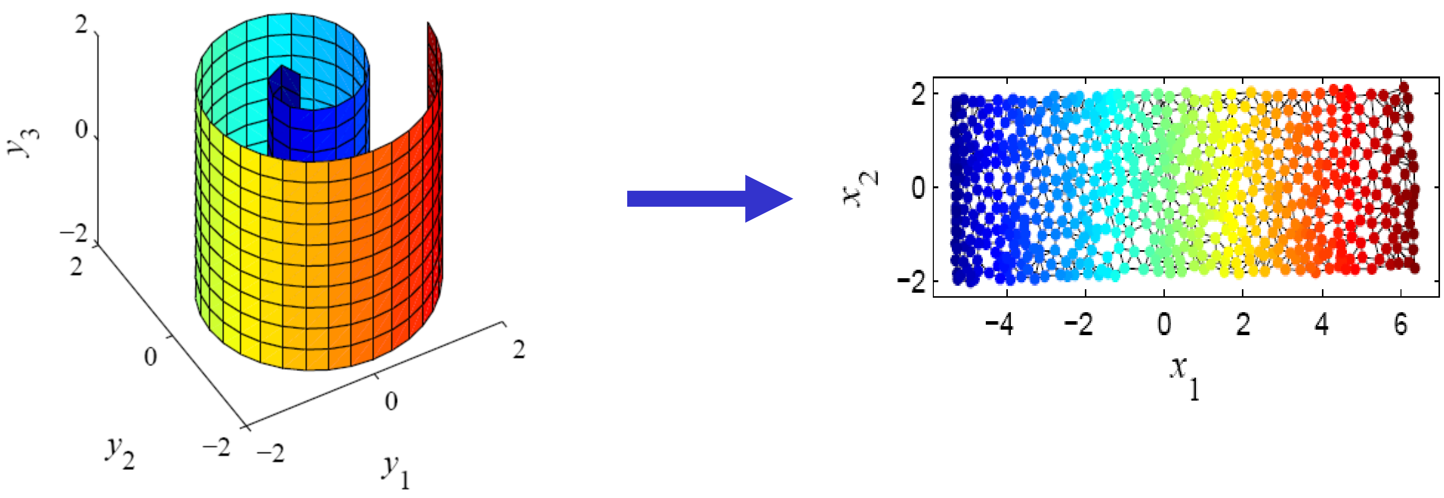




Often assimilated to signal processing, it is also about finding a *relevant information* in data, but in terms of *signals, sources,* etc.

Machine learning tasks

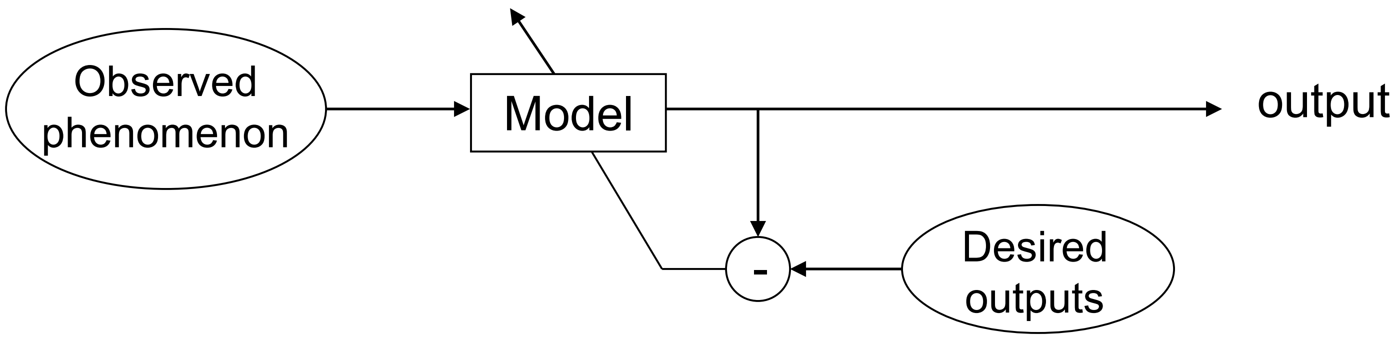
- Projection - visualization



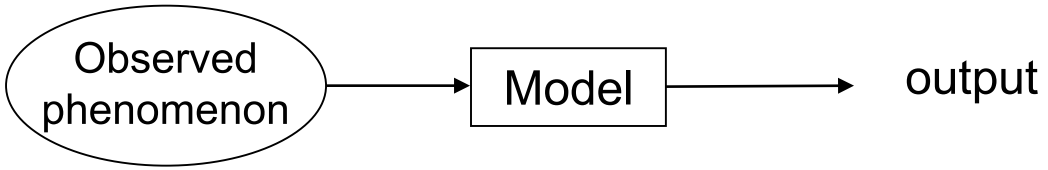
From: J. Lee, M. Verleysen, Nonlinear Dimensionality Reduction, Springer, 2007.

Supervised – unsupervised models

- Supervised learning: building an in-out relation thanks to data with know *desired output* (label, value, etc.)

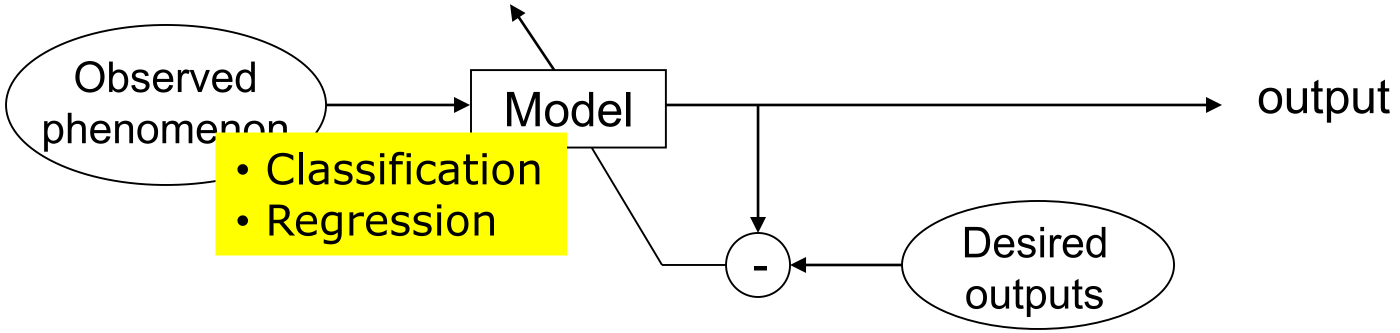


- Unsupervised learning: extracting some useful information from data without supplementary knowledge

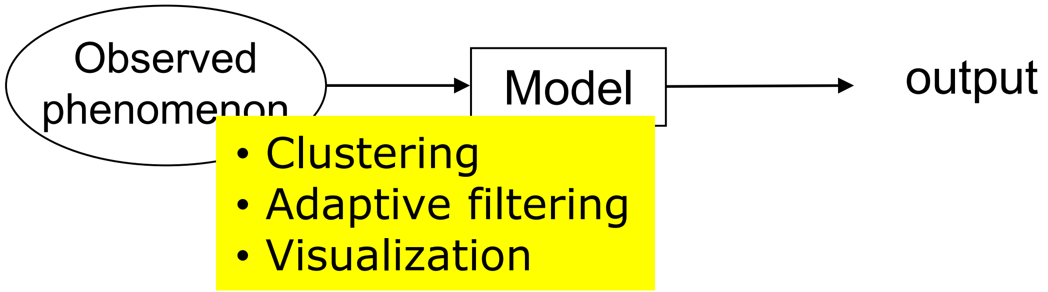


Supervised – unsupervised models

- Supervised learning: building an in-out relation thanks to data with know *desired output* (label, value, etc.)



- Unsupervised learning: extracting some useful information from data without supplementary knowledge



Sources and references

- The four books used as general references for this course
  - C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
  - R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, 2nd ed., Wiley, 2001
  - S. Theodoridis, K. Koutroumbas, Pattern Recognition, 4th ed., Academic Press, 2009
  - S. Haykin, Neural networks – a Comprehensive Foundation, 2nd ed., Prenctice Hall, 1999