Introduzione all'Intelligenza Artificiale (IIA) 2023

Parte 3: Introduzione all'Apprendimento Automatico

Introduction to Machine Learning

Notes of Lect. 1

Alessio Micheli

micheli@di.unipi.it

Dipartimento di Informatica
Università di Pisa - Italy

Computational Intelligence & Machine Learning Group

DRAFT, please do not circulate!

About IIA part 3

Dip. Informatica
University of Pisa

- **IIA Code: 586AA ECTS: 6 Semester: II**

- **IIA part 3: Introduction to Machine Learning**

Lecturer

- **Alessio Micheli:** micheli@di.unipi.it

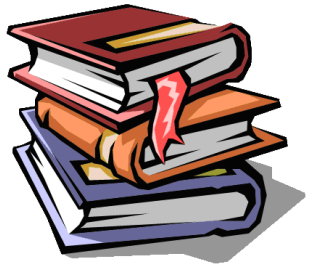
Web page of the course:

- **IIA course E-learning platform**
- **See the "Text Book and material" slide for the access to the slides (not public material)**

Practical premise: what change?

Dip. Informatica
University of Pisa

- This is a *new field* for most of you, with a different methodological approach
  - Take care of the material section at the end of this lecture for the bibliography: *we are not following the AIMA* as in the first part
  - Course attendance is strongly advised!
  - Why English slides? A bridge with the text books.



- Take care of the *background* needed for this part (end of this lecture), it is different from the background used in the first part of the course.

Learning

Dip. Informatica
University of Pisa

Learning : universal principles for living beings, society, machines

> *The problem of **learning** is arguably at the very core of the problem of **intelligence**, both biological and artificial*
>
> Poggio, Shelton, *AI Magazine 1999*

i.e. Learning as a major challenge and a strategic way to provide *intelligence* into the systems

See: AI Spring/«Revolution» in the INTRO lecture

What is ML? First view (I)

Dip. Informatica
University of Pisa

- Learning: a complex aim, a continuously growing research field
- In Computer Science, theoretical and applicative field called:
  - Apprendimento Automatico *in italiano (it)*
  - Machine Learning (ML) *English and literature* (aka learning systems)

- Machine Learning has emerged as an area of research combining the aims of creating *computers that could learn* (IA) and new *powerful adaptive/statistical tools* with rigorous foundation in computational science

- Machines that *learn* by itself. Why? Luxury or necessity?
  - Growing availability and need for analysis of empirical data

  → *Central/methodological role due to changing of paradigm in science: data-driven*
  - Difficult to provide adaptivity/intelligence by programming [see Turing]

  → *learning* as the only choice ...
- DATA + HPC + modern ML → Pave the way to a new AI era... *with you*

What is ML? First view (II)

Dip. Informatica
University of Pisa

Aims include:

- As AI methodology → Build Adaptive Intelligent Systems
  - from search engine to robotics …
- As statistical learning
  → Build powerful predictive system for Intelligent Data Analysis
  - tools for the "data scientist"

- As computer science method for innovative application areas
  → Using models as a tool for complex (interdisciplinary) problems
  - from biological data analysis to image understanding ...

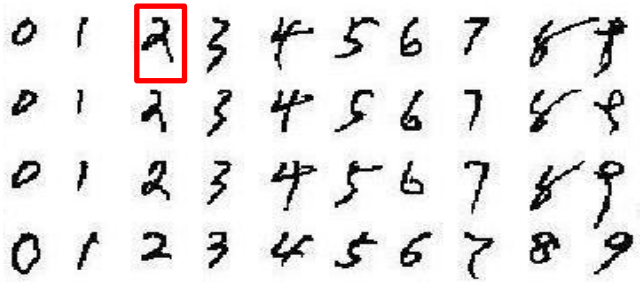
Within a uniform approach to the principles crosswise the different aims (and models/techniques)

Simple concrete examples

Dip. Informatica
University of Pisa

- Automatized learning by the system of the experience (set of examples) to address a computational task

Email spam classification



Character/face/speech recognition

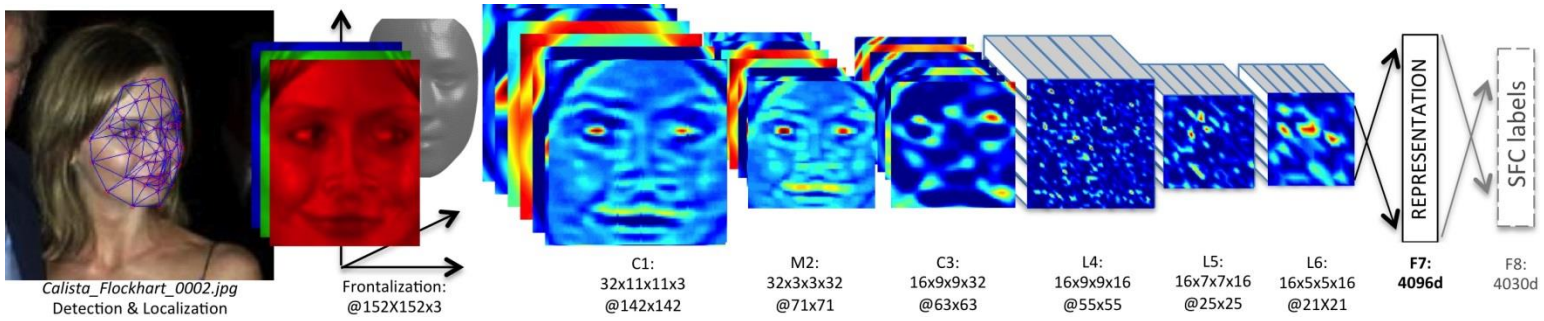
- … no (or poor) prior knowledge/rules for the solution but it is (more) easy to have a source of *training experience* (data with known results)

- Models used in Real-World systems (pervasive*) and in new interdisciplinary area, encompassing:
  - Pattern Recognition, Robotics, Computer Vision, Natural Language Processing, Information Retrieval, Web search engine, Complex Analyses of Data (Med, Bio, Web), Data Mining, Financial forecasting, Adaptive Systems and Filters, Intelligent Sensor Networks (Smart IoT), Personalized components, …

An instance on a "not-recent" result (CVPR, 2014)

Dip. Informatica
University of Pisa

- **Face recognition** combining (deep) Neural Networks and other ML approaches
- Starting form four million facial images belonging to more than 4,000 identities



- Asked whether two photos show the same person, DeepFace answers correctly 97.25% of the time … just a shade behind humans (97.53%).

  On Italian newspaper as: "…un grande cervello artificiale…"

Another instance: Go & Machine Learning

Dip. Informatica
University of Pisa

Nature 529, 445–446 (28 January 2016)
The news: *Deep-learning software defeats human professional for the first time*.

Nature 529, 484–489 (il metodo)
Mastering the game of Go with deep neural networks and tree search

Final results on March 2016 competition:
AlphaGo 4 - Lee Sedol 1.

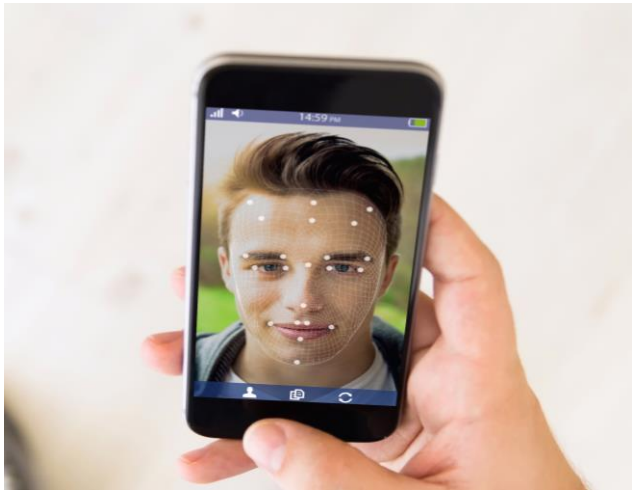
https://deepmind.com/alpha-go

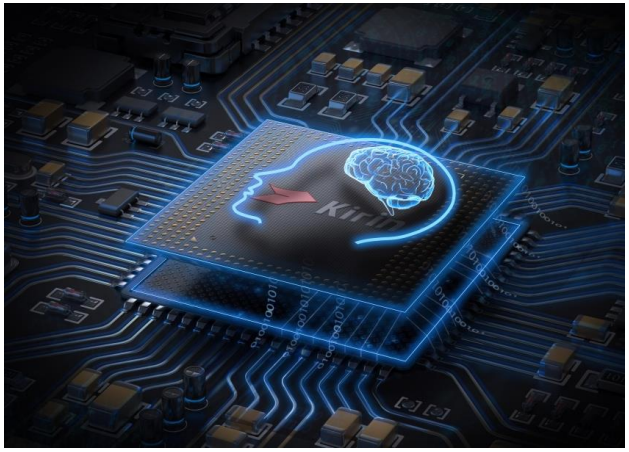
Aggiornamenti di automazione e nuove vittorie…

And "real-world" applications…

In your everyday life…

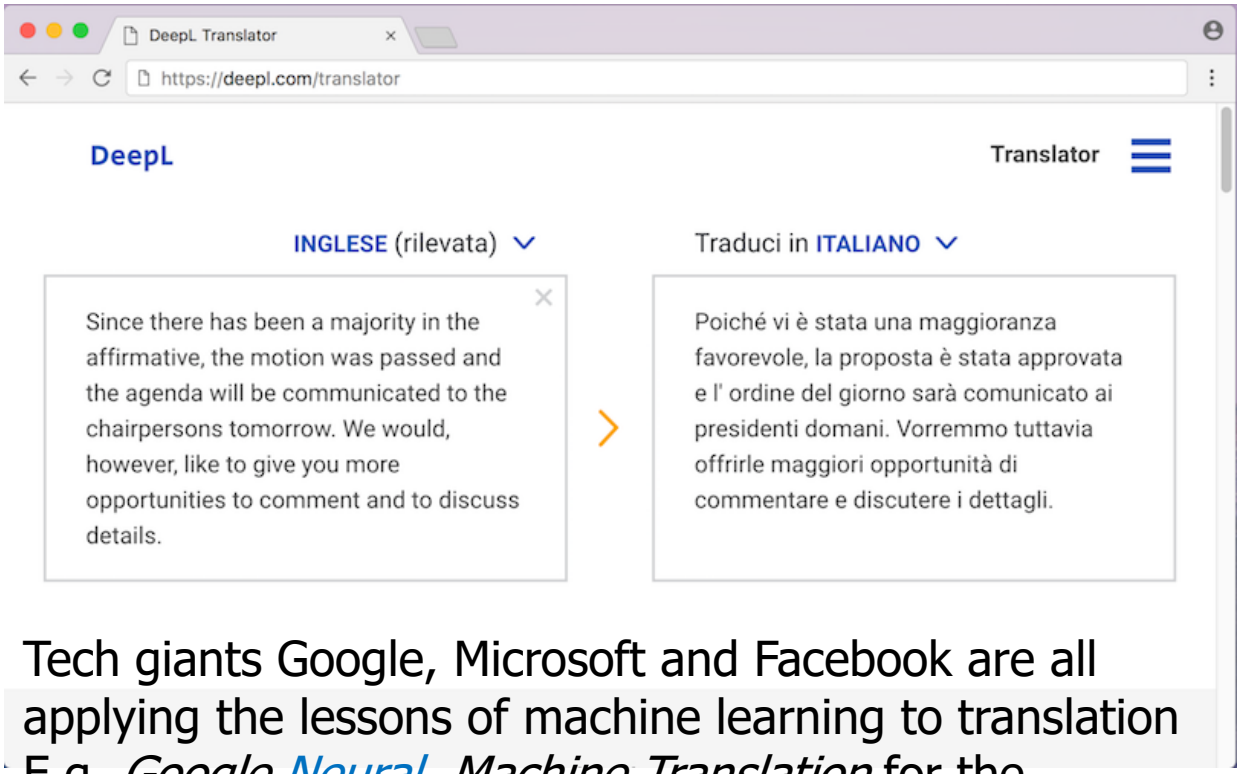
Dip. Informatica
University of Pisa

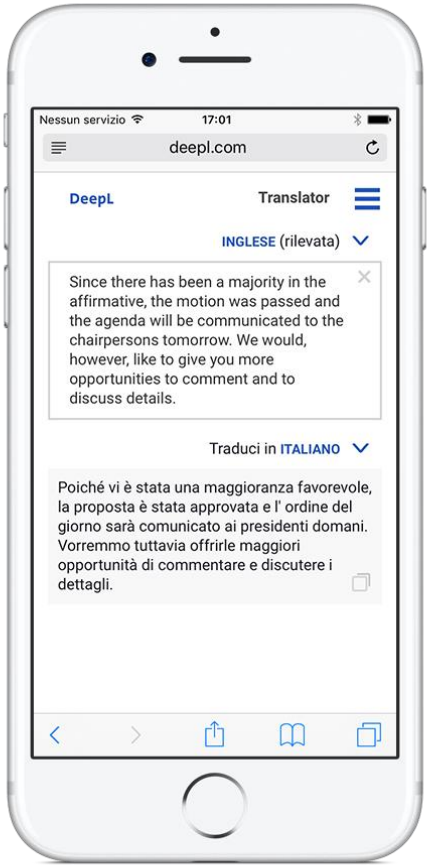




(Automatic) Machine Translation

Dip. Informatica
University of Pisa





Tech giants Google, Microsoft and Facebook are all applying the lessons of machine learning to translation
E.g. *Google Neural Machine Translation* for the <Google Transate> tool since 2016,
but also small companies are making big progresses:

An Example by <DeepL>

Since August 2017

a fully *Neural Network* based system.

Health (Examples)

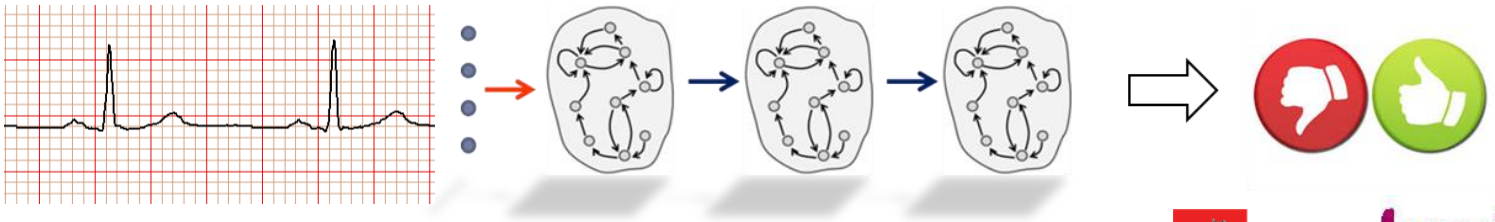
Dip. Informatica
University of Pisa

Diagnosis, therapies, personalized medicine, health monitoring… drug design

Skin Cancer Classification with Deep Learning

- The system (a deep neural network) can learn from 130,000 cases, far more than a doctor can "in many lifetimes"
- It achieves the accuracy of certified dermatologists (Nature 2017)
- It can be implemented with **app**

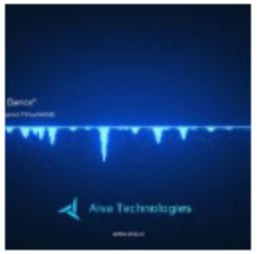
BrAID "Brugada syndrome and Artificial Intelligence applications to Diagnosis" (PRJ @CIML- UniPi – Tuscany region) [2020- 2023]



Regione Toscana braid

Music!

Dip. Informatica
University of Pisa



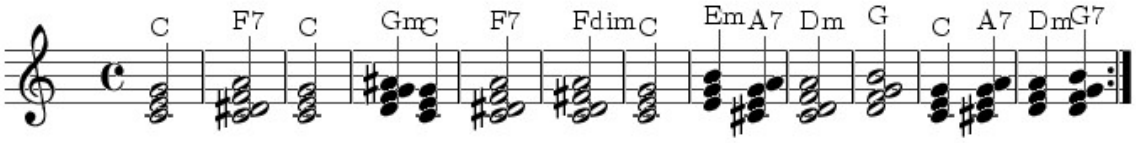
13 APRILE 2018

Il computer-compositore che si ispira a Beethoven

R.it

- «*AIVA prima ha composto un brano per solo piano,[…], poi un intero album, Genesi, per piano e orchestra; infine la musica per la festa nazionale del Lussemburgo; e qualche mese fa, la colonna sonora per uno dei videogame più popolari del mondo, Battle Royale di Fortnite.*»

- «*Come ci riesce è presto detto: al software sono stati fatti conoscere, diciamo così, gli spartiti delle composizioni dei più grandi autori della storia, da Mozart a Beethoven fino a Bach. Ha studiato dai migliori, insomma, con una tecnica che si chiama "deep learning". Da qui AIVA ha ricavato gli schemi ricorrenti di una composizione musical, e a quanto pare è in grado di replicarli adattandosi alla richiesta che viene fatta*»



Turing award 2018

Dip. Informatica
University of Pisa

- On 27-th March 2019, the Association for Computing Machinery, the world's largest society of computing professionals, announced that Drs. Hinton, LeCun and Bengio had won this year's Turing Award for their work on **neural networks**. The Turing Award, which was introduced in 1966, is often called the "Nobel Prize of computing", and it includes a $1 million prize, which the three scientists will share.

- "For conceptual and engineering breakthroughs that have made deep neural networks a critical component of computing"

Y. Bengio

G. Hinton

Y. LeCun

Machine Learning When? (summarizing)

Dip. Informatica
University of Pisa

Opportunity (if useful) and *awareness* (needs and limits)

- Utility of predictive learning models: (in the following cases)
  - **no (or poor) theory** (or knowledge to explain the phenomenon or difficult to be formalized)
  - **uncertain, noisy or incomplete data** (which hinder formalization of solutions)
  - **dynamical environments**, not known in advance (E.g. adapt to *personalized behavior* according to dynamical user preferences)
- Requests:
  - source of training experience (representative data)
  - tolerance on the precision of results *

ML: why?

An informal note from a student question

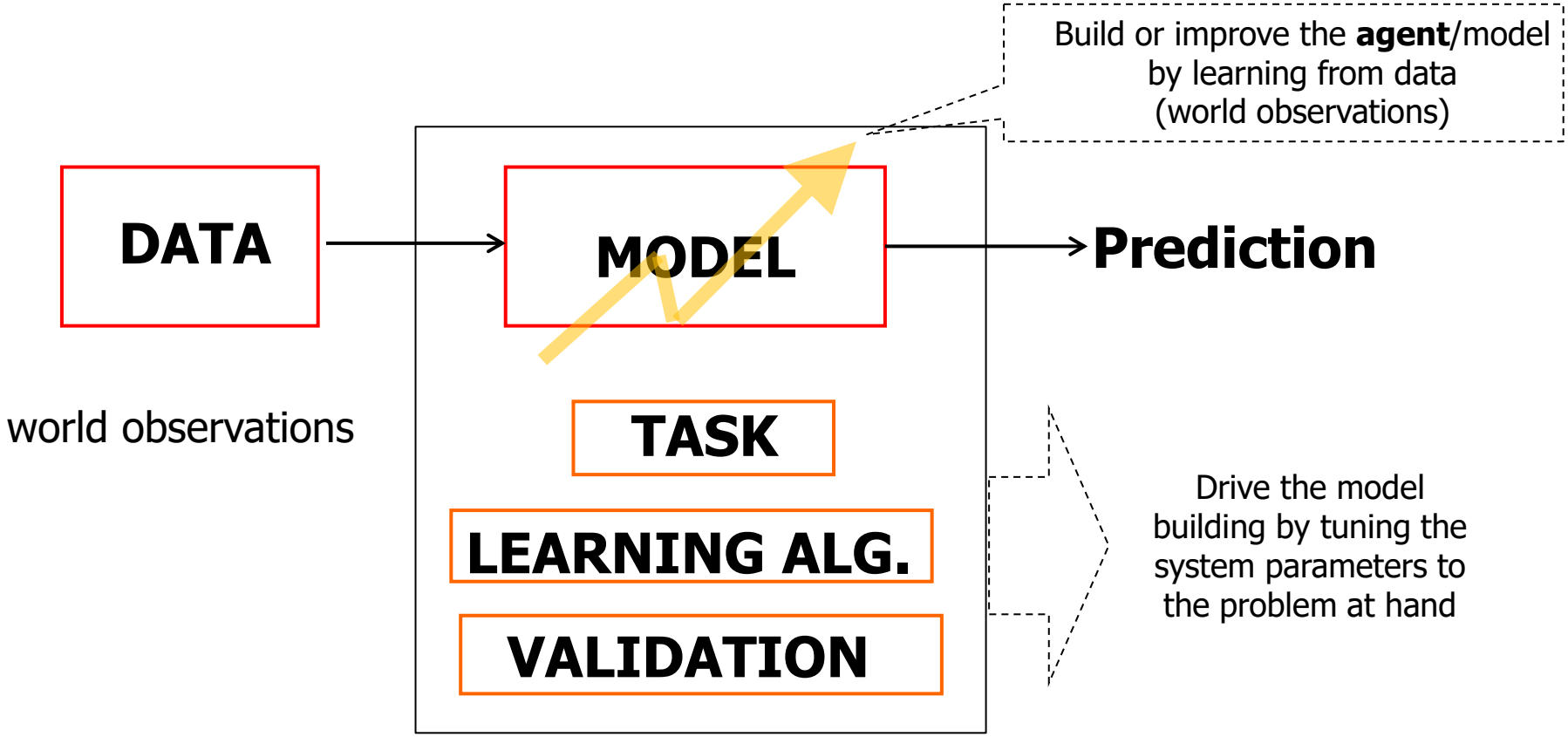
Dip. Informatica
University of Pisa

- An opportunity to know *new computing paradigms*, with an approach different w.r.t. to Standard Programming, algorithmic/classic IA approaches
  - e.g. treatment of uncertainty, tolerance of imprecision, … see previous slide
- Typical of the *soft computing/computational intelligence* area
- To find *approximate solutions* for difficult problems, difficult to be formalized by 'hand-made' algorithm
- To build new robust and wide applicable *intelligent systems*

- But it is *NOT an approximate methodology*!
- It is a *rigorous* approach to find *approximate function* to deal with complex problems (supported by empirical and theoretical results e.g. SLT),
  - soft computing paradigm: open many new opportunities (extend the frontier of CS applications)
  - often bio-inspired (neural) modeling

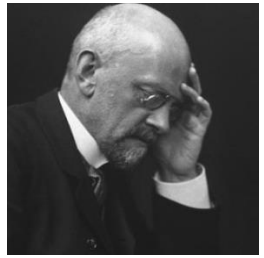
Overview of a ML (predictive) System

Dip. Informatica
University of Pisa



Also as a guide to the **key design choices**

Learning as an approximation of an unknown function from examples



Hilbert spaces

Specific vision but widespread in ML

For us:

- Different tasks seen in uniform framework
- Enables a rigorous formulation

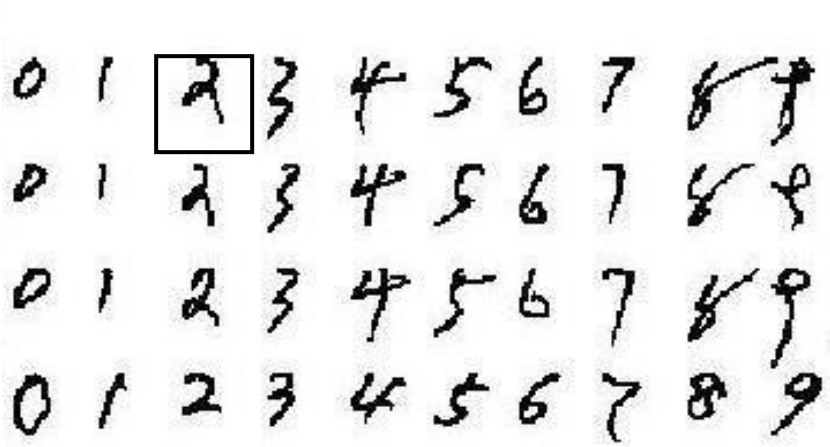
→ **Intro guided by an intuitive example**

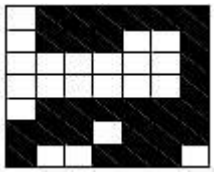
An Example

Dip. Informatica
University of Pisa

- A pilot example: recognition of handwritten digits

- Input: collection of images of handwritten digits (arrays/matrix of values)
- Problem: build model that receives in input an image of handwritten digit and "predict" the digits





8 x 8

Handwritten Digits Recognition

Dip. Informatica
University of Pisa

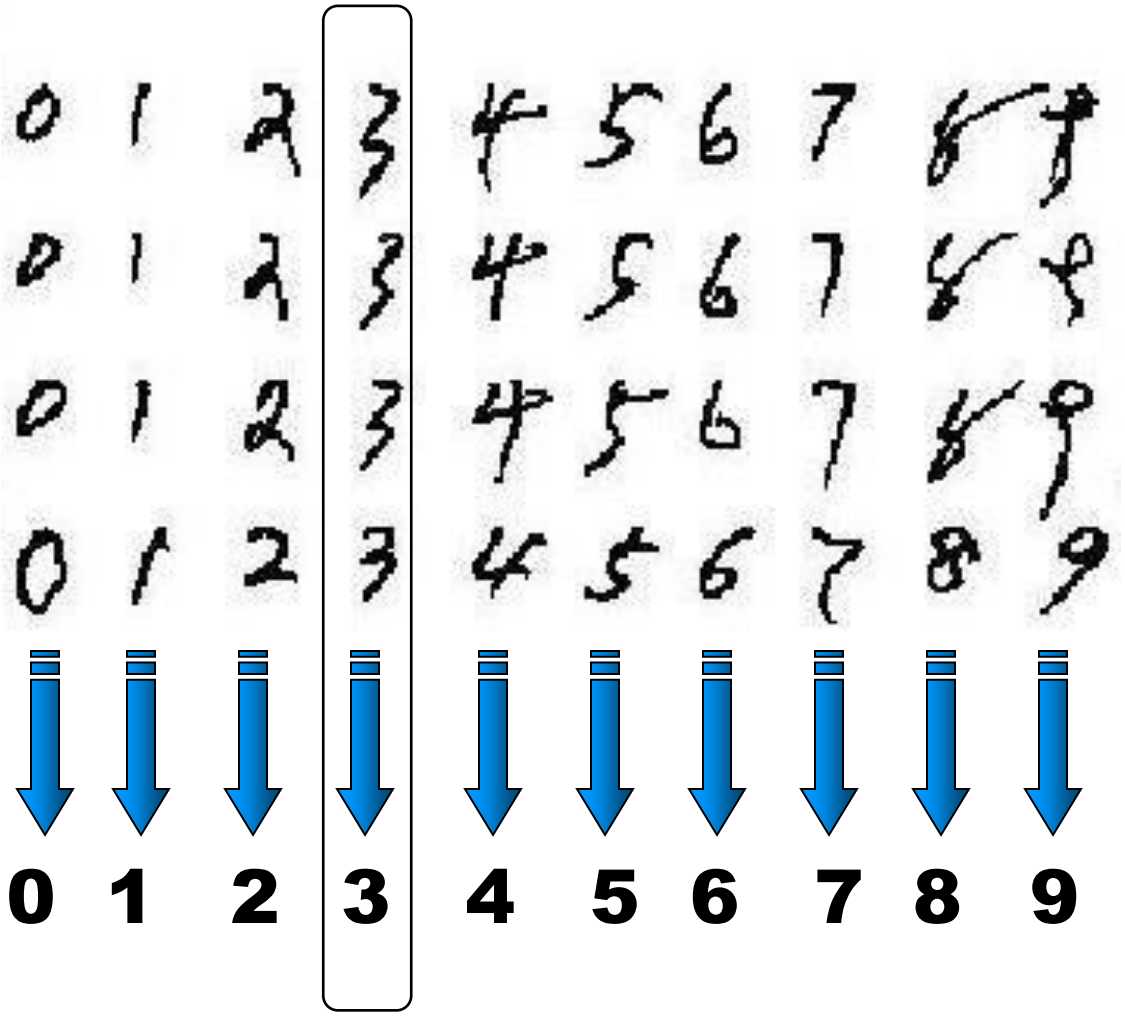
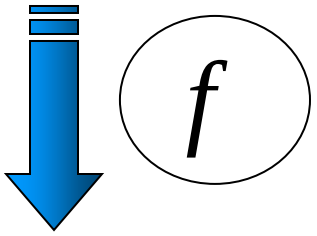


Image
8 x 8



Output class

Handwritten Digits Recognition (II)

Dip. Informatica
University of Pisa

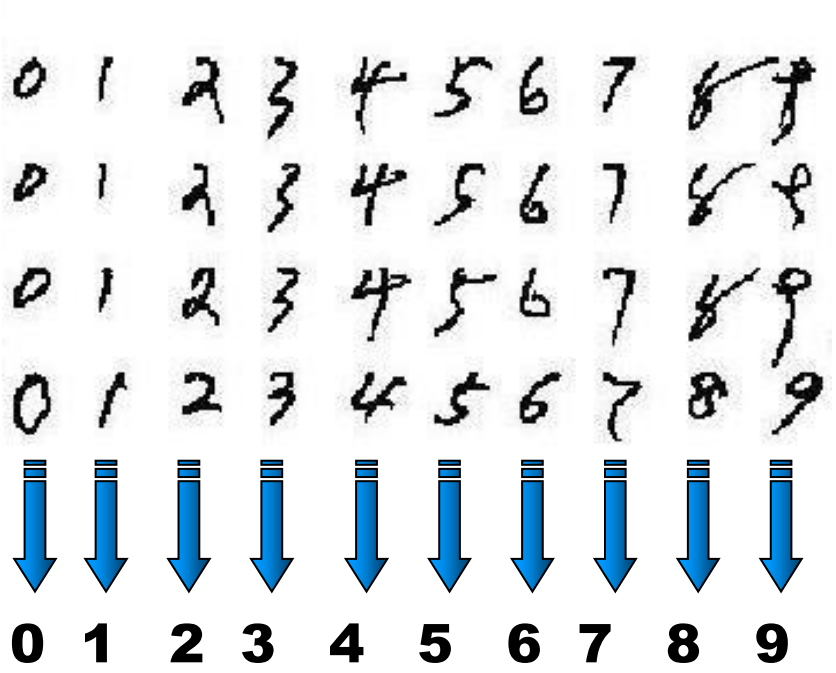
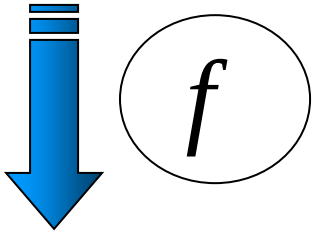


Image
8 x 8



Output class

Classification problem

- Difficult to formalize exactly the solution of the problem: Possible presence of noise and ambiguous data;
- Relatively easy to collect collection of labeled examples

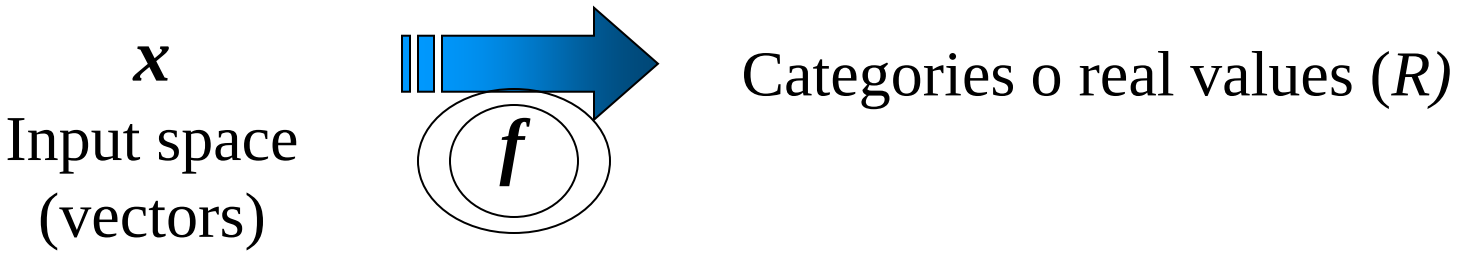
→ Example of successful application of the ML!

A step ahead from the pilot example

Dip. Informatica
University of Pisa

Generalizing the pilot example problem:

- *Supervised learning* (classification, regression)



Build a function from examples

Tasks: Supervised Learning

Dip. Informatica
University of Pisa

- Given: Training examples as $<input,output>=(\boldsymbol{x},d)$ (***labeled* examples**)

  for an unknown function f (known only at the given points of example)

  Def
  - Target value: desiderate value d or t or y … is given by the teacher according to $f(\boldsymbol{x})$ to label the data
- Find: A *good* approximation to f (i.e. a *hypothesis* h that can used for prediction on unseen data $\boldsymbol{x}'$)

- Target d (or t or y): a categorical or numerical *label*
  - ***Classification***: $f(\boldsymbol{x})$ return the (assumed) correct class for $\boldsymbol{x}$

    Def

    $f(\boldsymbol{x})$ is a *discrete-valued function* $\in \{1,2,\ldots,k\}$ *classes*
  - ***Regression***: real continuous output values (approximate a real-valued target function, in R or R^K)

  Both as a *function approximation* task

Examples of x - $f(x)$

Dip. Informatica
University of Pisa

Inferring general functions from know data:

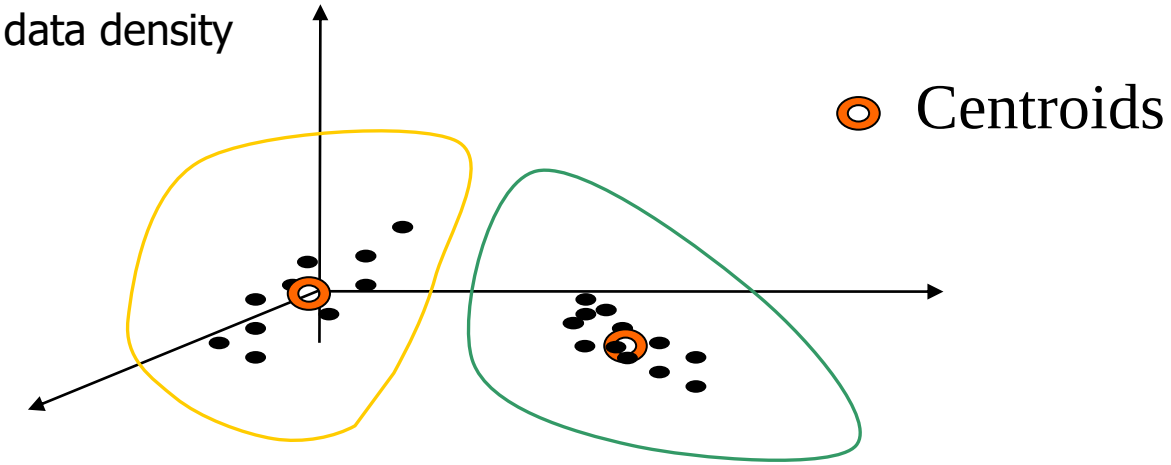
- Handwriting Recognition
  - ***x***: Data from images of the characters.
  - *f(**x**)*: Letter of the alphabet.
- Disease diagnosis (from database of past medical records)
  - ***x*** : Properties of patient (symptoms, lab tests)
  - *f(**x**)*: Disease (or maybe, recommended therapy)
  - TR (training set) <***x***,*f(**x**)*>: database of past medical records
- Face recognition
  - ***x***: Bitmap picture of person's face
  - *f(**x**)*: Name of the person.
- Spam Detection
  - ***x***: Email message
  - *f(**x**)*: Spam or not spam.

Tasks: Unsupervised Learning

Dip. Informatica
University of Pisa

Unsupervised Learning: No teacher!

- TR (Training Set)= set of unlabeled data <x>
- E.g. to find *natural groupings* in a set of data
  - Clustering
  - Dimensionality reduction/ Visualization/Preprocessing
  - Modeling the data density



- Clustering:

Partition of data into clusters (subsets of "similar" data)

Models
and survey of useful concepts

Dip. Informatica
University of Pisa

- MODEL:
  - Aim: to capture/describes the relationships among the data (on the basis of the task)
  - It defines the class of functions that the learning machine can implement (*hypothesis space*)
    - E.g. set of functions $h(\mathbf{x},\mathbf{w})$, where $\mathbf{w}$ are (abstract) parameters

Def

Defs

- **Training example** (superv.): An example of the form ($\mathbf{x}$, $f(\mathbf{x})+noise$)
  $\mathbf{x}$ is usually a vector of features, *(d or t or)* $y=f(\mathbf{x})+noise$ is called the target value
- **Target function**: The true function f
- **Hypothesis**: A proposed function h believed to be similar to f. An expression in a given *language* that describes the relationships among data
- **Hypotheses space**: The space of all hypotheses (specific models) that can, in principle be output by the learning algorithm

Models
Languages for H

Dip. Informatica
University of Pisa

- Fortunatamente già conoscete alcuni **linguaggi** in cui esprimere relazioni che possiamo usare per esprimere modelli di ML (le ipotesi h):

- **Logica** del primo ordine
- **Equazioni numeriche**

} Partiremo proprio da questi per i primi esempi di modelli !

- **Probabilità**: questo verrà reintrodotto per la rappresentazione della conoscenza incerta in AI (e quindi per esprimere modelli di ML)

Models: few simple examples....*

Dip. Informatica
University of Pisa

Just to have **a preview** of different *representation* of hypothesis:

- Linear models (representation of H defines a continuously parameterized space of potential hypothesis);
  each assignment of w is a different hypothesis, e.g:
  - $h_{\mathbf{w}}(x)=w_1x+w_0$ $\quad E.g.\ h_{\mathbf{w}}(x)= 0.232x+246$
- Symbolic Rules: (hypothesis space is based on discrete representations); different rules are possible , e.g:
  - *if* $(x_1=0)$ *and* $(x_2=1)$ *then* $h(\mathbf{x})=1$
  - *else* $h(\mathbf{x})=0$
- Probabilistic models: estimate $p(\mathbf{x},y)$
- Instance based approaches: Predict mean y value of nearest neighbors (memory-based)

Learning Algorithms

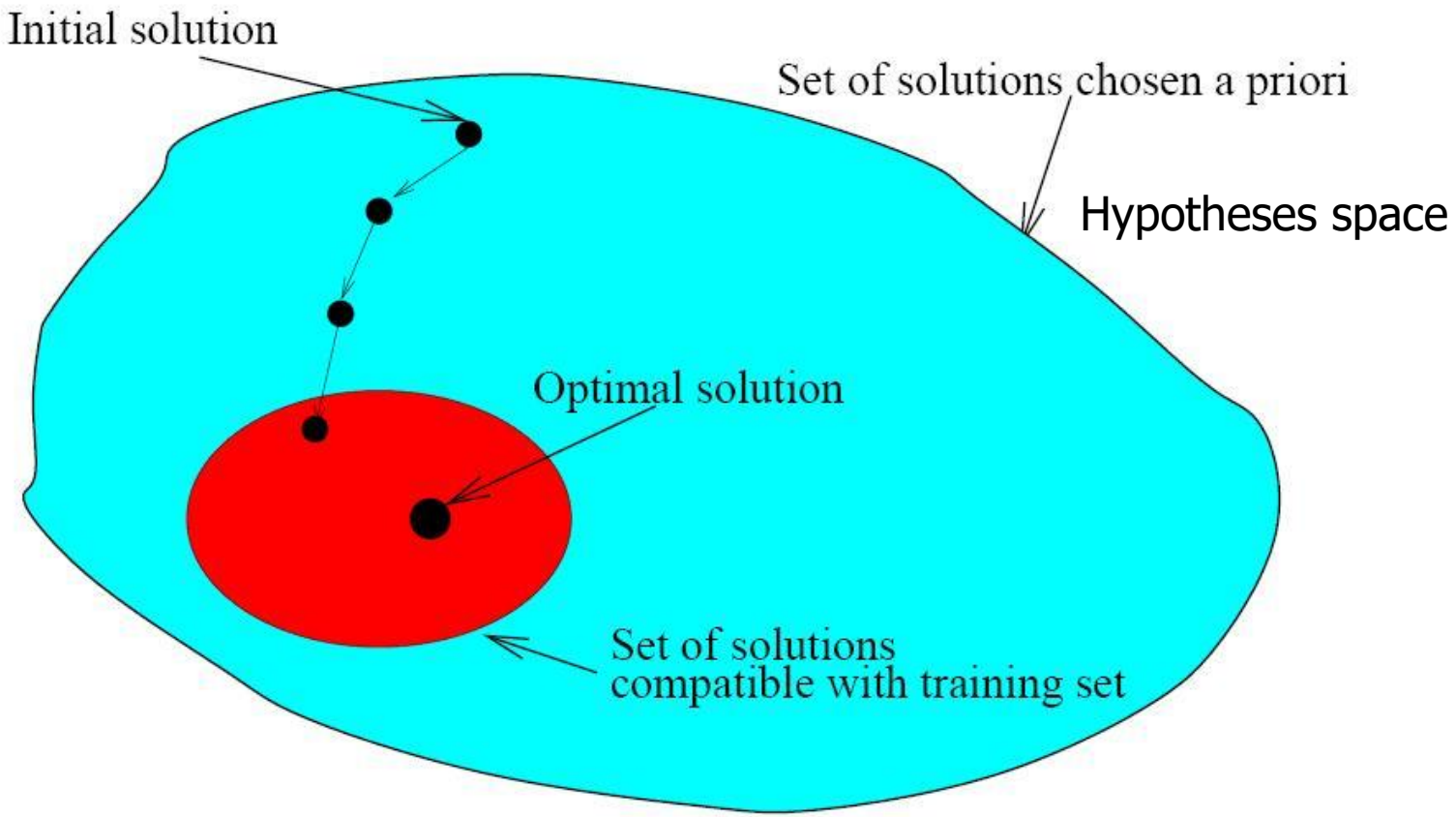
Dip. Informatica
University of Pisa

Def

- LEARNING ALGORITHM
- Basing on data, task and model, learning as a (Heuristic) ***search*** *through the hypothesis space* H of the **best hypothesis**
  - i.e. the best approximation to the (unknown) target function
  - Typically searching for the *h* with the minimum "error"
- E.g. free parameters of the model are *fitted* to the task at hand:
  - Examples: best w in linear models, best rules for symbolic models, ….
- H may not coincide with the set of all possible functions and the search can not be exhaustive: it needs to make assumptions → *we* will see the role of the *Inductive bias*

Learning Algorithms: search

Dip. Informatica
University of Pisa



Typically ***local search*** approaches
(see the first part of the course)

Machine Learning: generalization

Dip. Informatica
University of Pisa

This is a fundamental concept of the course

- *Learning*: search for a *good function* in a function space from known data *(typically minimizing an Error/Loss)*

Def

- Good w.r.t. generalization error: it measures how accurately the model predicts over novel samples of data (*Error/Loss measured over new data)* (low error, high accuracy and vice versa)

Generalization: crucial point of ML!!!

Easy to **use** ML tools *versus* **correct/good use** of ML

Generalization

Dip. Informatica
University of Pisa

- **Learning** phase **(training, fitting)**: build the model from know data – *training data* (and bias)
- **Predictive** phase **(test)**: apply to new examples
  (we take the inputs **x'**; we compute the response by the model; we compare with its target that the model has never seen):

  evaluation of the predictive hypothesis, i.e. of the **generalization capability**

Note: *performance* in ML = *predictive accuracy*

estimated by the error computed on the (Hold out) Test Set

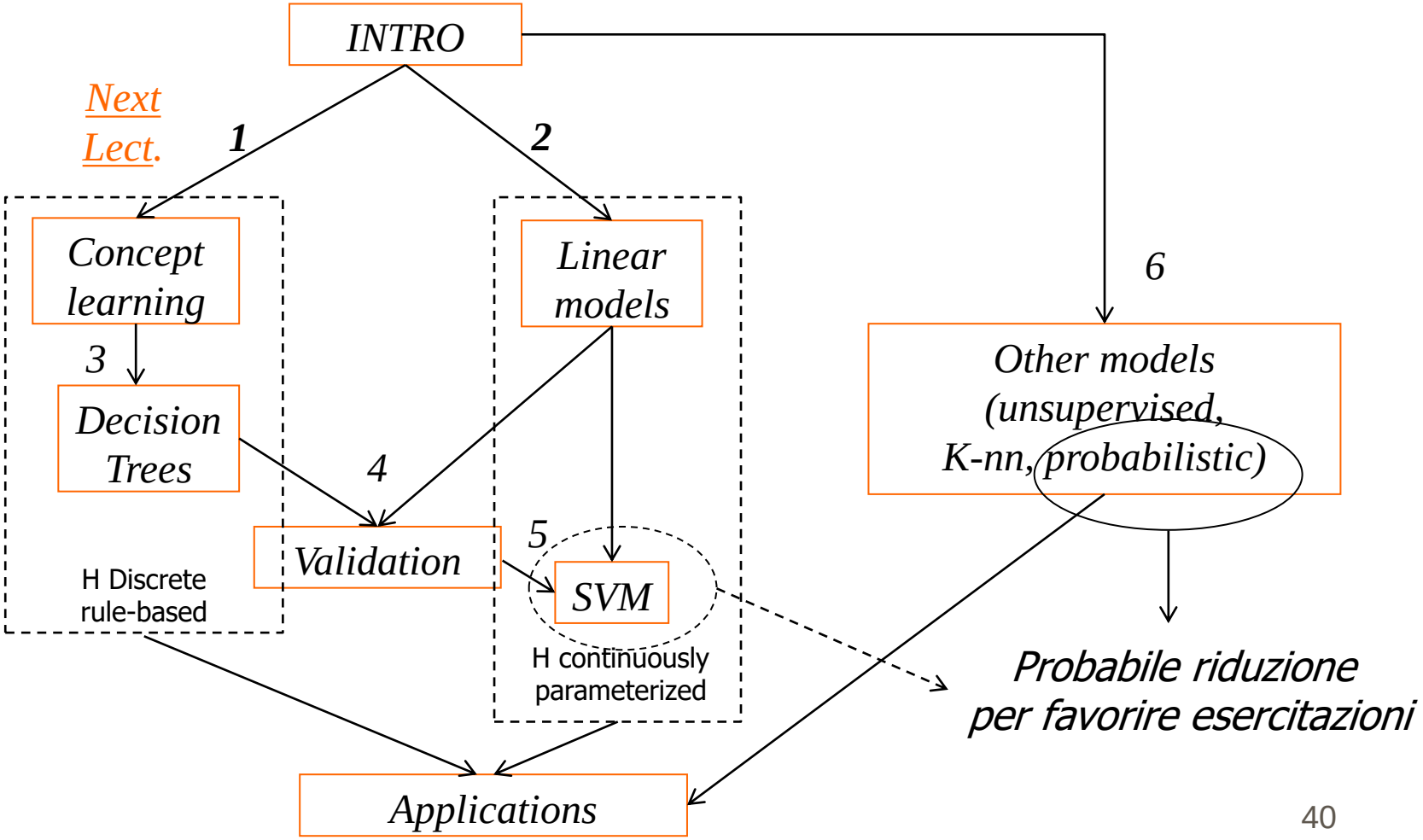
- **Theory**: E.g. Statistical Learning Theory [Vapnik] :
  - *under what (mathematical) conditions is a model able to generalize?* → see next lectures (just basic notions)

ML part structure
and other pratical information

Course structure: preview

Dip. Informatica
University of Pisa

There is a structure in this *intro to ML* (bottom-up: starting from simple cases to view the concrete side of the general principles), collocating each lecture help you.



Objectives details

Dip. Informatica
University of Pisa

- Method: introductory, bottom-up (starting from simple cases to view the concrete side of the general principles) → we have a structure: follow the lectures!!!!
- For deeper view we will use
  - **#ML** to indicate that details will be given at ML master degree course
  - **#Tech** to indicate topic that may be too "technical" for some readers
- Other labels
  - *Def* a definition that is "useful" to know
  - "*Exercise*": something useful to do!
- Detailed Content: **see resume lecture** at the end !!!!
  - We would include an introduction to discrete and linear models, Decisions Trees, validation, SVM, probabilistic approaches and Naive Bayes, basic clustering alg., applications examples…

Textbook and material (1)

Dip. Informatica
University of Pisa

- **Course notes (slides copy)**

  with material from time to time indicated in the classroom:
  bibliographic references at the end of each specific topic via books chapters and/or online material.

  Hint: Course notes are a very useful guide to the selected topics!

Part3: USO personale delle slide elettroniche cosi come delle registrazioni:

- No pswd is needed, however…
- The electronic slides are reserved

(draft material). It is **not** a public material!

(again https://elearning.di.unipi.it)

- Please do not use the link in any web site/social media
- Please do not repost slides in any form

Textbook and material (2)

Dip. Informatica
University of Pisa

- **Main Bibliography for this course (OLTRE LE SLIDES):**

1) (AIMA) S. Russell, P. Norvig: *Artificial Intelligence: A Modern Approach.* Pearson (**3 edition**, 2010)

2) T. M. Mitchell: *Machine learning*, McGraw-Hill, 1997.

3) REPETITA: bibliographic references at the end of each specific topic via books chapters and/or online material.

- 2) and 3) are more relevant for this part

Textbook and material (3)

Dip. Informatica
University of Pisa

- **Other general advanced references for ML (examples):**
  - C.M. **Bishop**: Pattern Recognition and Machine Learning, Springer 2006
  - Hastie, Tibshirani, Friedman: The Elements of Statistical Learning, Springer Verlag, 2001 (new eds. up to 2009) → *pdf on-line!!!*
  - Duda, Hart, Stork: Pattern Classification, 2nd. ed. J. Wiley & Sons, 2001
  - S. Haykin: *Neural Networks: a comprehensive foundation*, IEEE Press; (2nd. Edition, 1998) → OR
    - * S. Haykin: *Neural Networks and Learning Machines,* Prentice Hall; *(3rd Edition, 2008)*
  - * I. Goodfellow, Y. Bengio, A. Courville: ***Deep Learning,*** MIT Press, 2016

* ML master degree course textbooks

Prerequisites

Dip. Informatica
University of Pisa

- No other course in the field of AI is assumed (we start from scratch)

- General prerequisites (typically from a First Cycle «Laurea» Degree Programme in CS/Math/…):
  - elements of mathematical analysis (functions, differential calculus)
  - elements of matrix notation and calculus
  - algorithms
  - elements of probability and statistics

See few slides at the end of this lecture for notation references!

Background

Dip. Informatica
University of Pisa

- Very informal notation resume
- Please, use the AIMA appendix A.2 and A.3 (mathematical background) free available at
- http://aima.cs.berkeley.edu/newchapa.pdf
- Elementary video lectures on basic math (equations, lines, derivative, … in Italian) by Prof. Bombardelli
- Other links in the Moodle platform ("**Prerequisiti**")

- And feel free to ask for any doubt!!!!
- Don't worry: IA/ML are multidisciplinary

fields from the beginning!!!!!

Basic background (references)

Dip. Informatica
University of Pisa

- Multivariable/Multivariate calculus

  Functions with multiple inputs: $f(x_1,x_2)$ *or* $f(\mathbf{x})$, *where* $\mathbf{x}=(x_1,x_2,\ldots)$

  - *Partial derivative, gradient.*

- Probability:
  - *probability density function, mean and variance, Normal random variable*
  - *p(x) or P(X=x), conditional probability: p(x|y), joint probabilities, …*

- Matrix calculations and notations: x *(scalar)*, $\mathbf{x}$ *(vector)*, $\mathbf{X}$ *(matrix)*
  - *inner (dot, scalar) product, inverse, norms, ...*

(it) *Prodotto Scalare*
(en) Inner (dot, scalar) product

Dip. Informatica
University of Pisa

$$\mathbf{a} \cdot \mathbf{b} = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n = \sum_{i=1}^{n} a_i b_i \qquad \mathbf{a} \cdot \mathbf{b} = |\mathbf{a}|\,|\mathbf{b}| \cos\theta$$

- Other notations $\boldsymbol{a} \cdot \boldsymbol{b} = \boldsymbol{a}^T \boldsymbol{b} = \boldsymbol{a}^t \boldsymbol{b} = <\boldsymbol{a}, \boldsymbol{b}>$ and even $\boldsymbol{ab}$ if the context is clear
- Magnitude/size (it '*Modulo*') or length or *Euclidean norm* of a vector $\boldsymbol{x}$

$$\sqrt{\boldsymbol{x}^T \boldsymbol{x}} = \sqrt{\boldsymbol{x}^T \cdot \boldsymbol{x}} = \sqrt{\sum_i {x_i}^2} = d(\boldsymbol{x}, 0) = \|\boldsymbol{x}\|_2 = \|\boldsymbol{x}\| = |\mathbf{x}|$$

Simplified

- Generalization: function that relate a couple of vectors to a number (a scalar): (bilinear symmetric form, by the notation <,>)

$$\langle v, w \rangle = \langle w, v \rangle$$
$$\langle v + w, u \rangle = \langle v, u \rangle + \langle w, u \rangle$$
$$\langle kv, w \rangle = k\langle v, w \rangle$$

- Cauchy-Schwarz inequality $|\langle x, y \rangle| \leq \|x\| \cdot \|y\| \quad \forall x, y \in V$

Partial derivative: calculus

Dip. Informatica
University of Pisa

- The **partial derivative** generalizes the notion of the derivative to higher dimensions. A partial derivative of a multivariable function is a derivative with respect to one variable (e.g. x_1) with all other variables held constant.

- *(it) Come tecnica di calcolo, la derivata parziale di una funzione rispetto a una variabile x_1 (lo stesso discorso può ripetersi per le altre variabili x_2, x_3 ecc.) in un punto si ottiene derivando la funzione nella sola variabile x_1, considerando tutte le tre variabili come se fossero costanti.*

- Hence, if $Df(x) = f'(x) = df/dx$
- for $f(x_1, x_2, x_3)$ we can compute: df/dx_1, df/dx_2, df/dx_3

$$\frac{\partial f}{\partial x_1} \qquad \frac{\partial f}{\partial x_2} \qquad \frac{\partial f}{\partial x_3}$$

Gradient

Dip. Informatica
University of Pisa

- When a function of two variables $f(x_1, x_2)$ have partial derivatives at each point (x_1, x_2) we can associate the vector of the two partial derivatives $\left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}\right)$ called the **gradient** of *f*, often denoted ∇f or *grad f.*

- For *n* variables: the gradient is the **vector** field whose components are the partial derivatives of *f:*

$$grad\ f = \mathbf{e}_1 \frac{\partial f(x_1, \ldots, x_n)}{\partial x_1} + \cdots + \mathbf{e}_n \frac{\partial f(x_1, \ldots, x_n)}{\partial x_n}$$

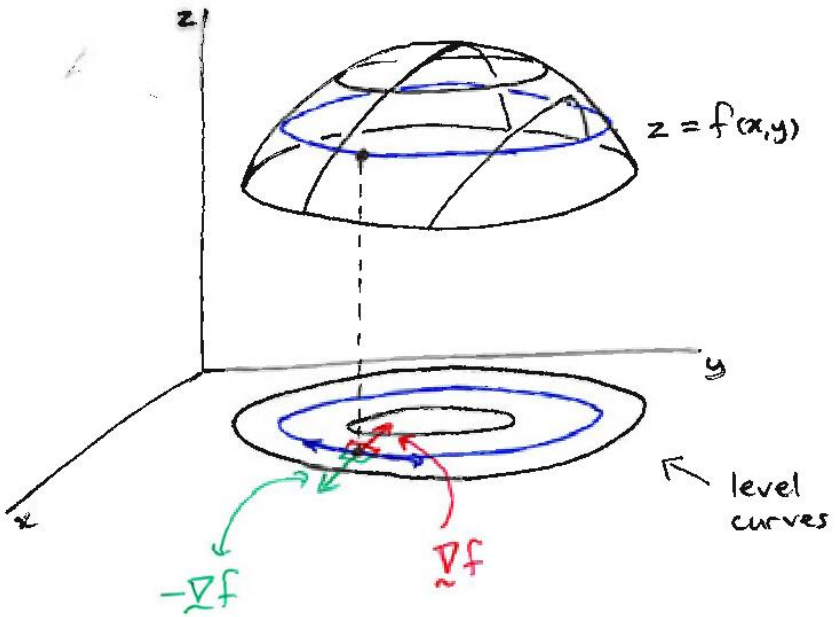
where the $\mathbf{e}_i$ are the orthogonal unit vectors pointing in the coordinate directions *(e.g. 001, 010, 100 in 3D).*

(ri)Vedere lezione su *ricerca locale*
(su spazi continui)

Gradient on a surface

Dip. Informatica
University of Pisa

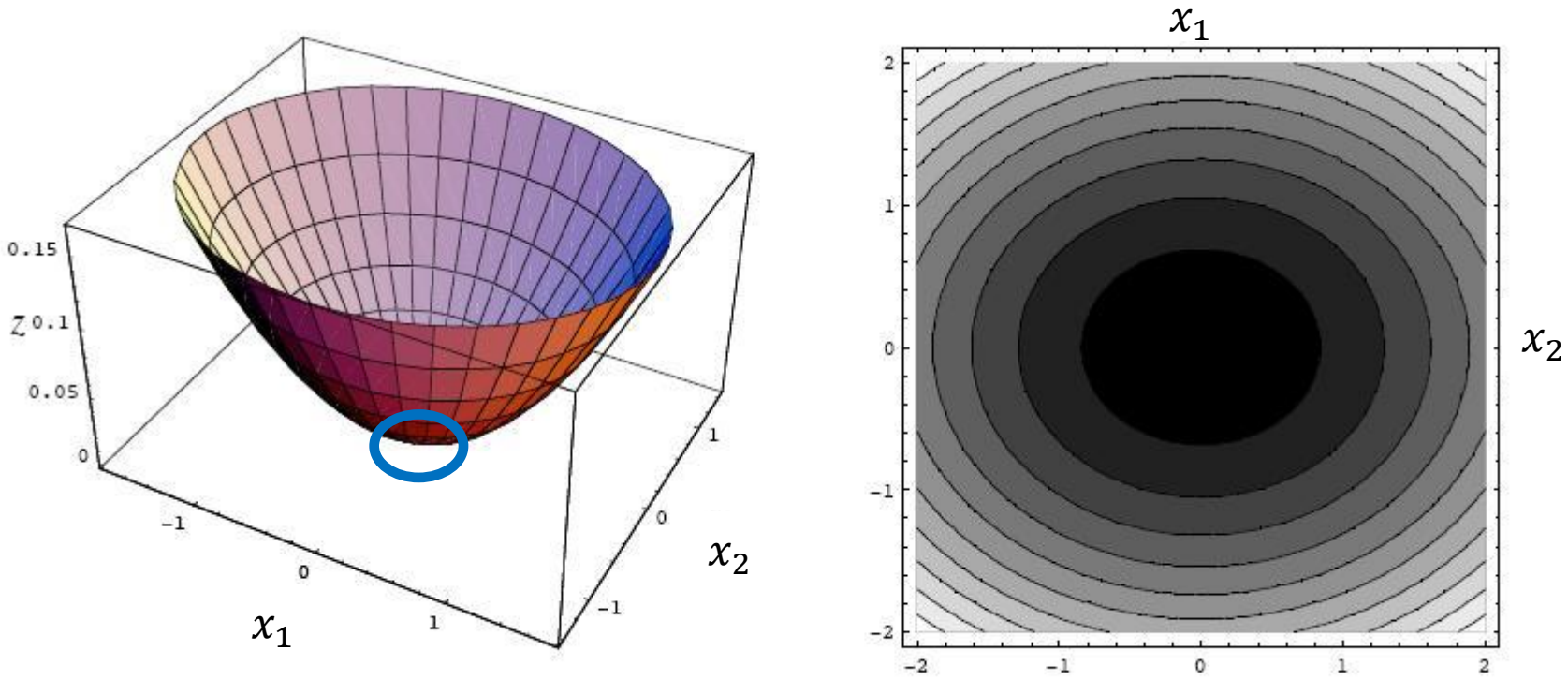
- **The gradient at a point is a vector pointing in the direction of the steepest slope at that point.**
- The steepness of the slope at that point is given by the magnitude of the gradient vector
- Example for a *f* on 2 variables →



→ (ri)Vedere lezione su *ricerca locale* (su spazi continui), con esempio

Local Minimum

Dip. Informatica
University of Pisa



- Example of a *stationary point*: the gradient is null

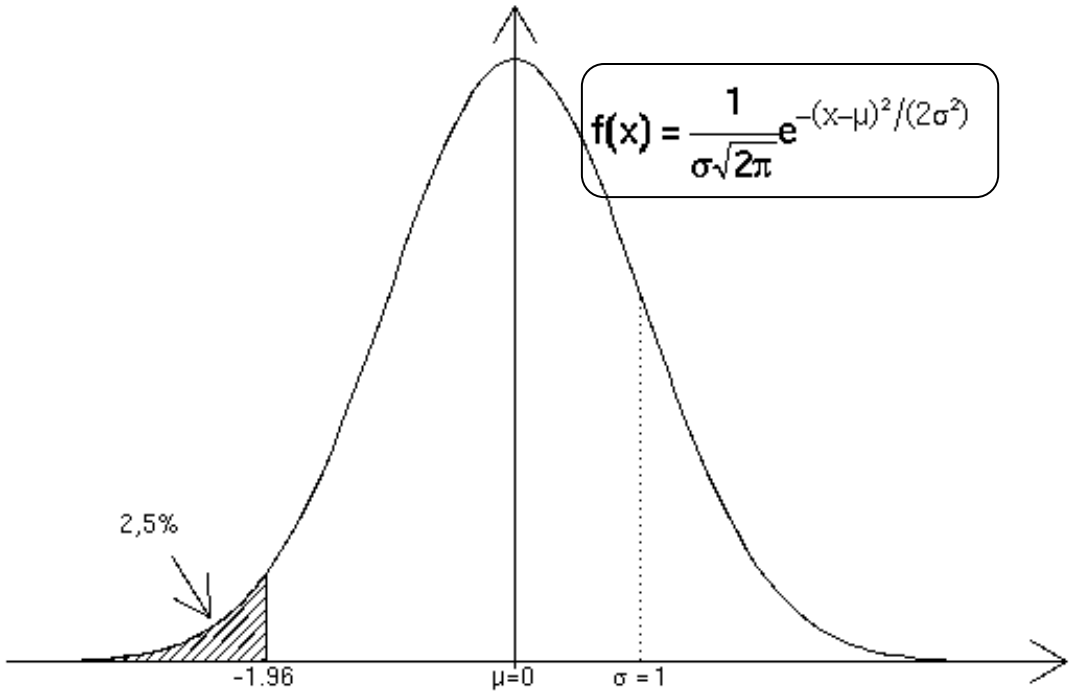
(while are also stationary saddle points or local min/max)

Density (example)

Dip. Informatica
University of Pisa

- The density function of the normal random variable with mean 0 and variance 1 (called standard normal distribution, see the figure) and the analytical expression of the corresponding density in the generic case *(mean μ and variance σ^2).*

The probability density function (pdf) for a **normal distribution:** the "Gaussian function"



DRAFT, please do not circulate!

For information

Alessio Micheli

micheli@di.unipi.it

https://ciml.di.unipi.it/

Dipartimento di Informatica
Università di Pisa - Italy

Computational Intelligence & Machine Learning Group