Linguaggi di Programmazione per la Matematica

B. Manca

A.a. 2025/26

Contents of the Lectures

- Introduction on Machine Learning
- Classification algorithms:
  - First examples of trivial classifiers;
  - Support Vector Machines;
  - Multiclass Classification;
- Clutering algorithms:
  - K-means clustering;
  - Spectral clustering;
  - Density based clustering;
- Regression
- Dimensionality reduction

Bibliographic References

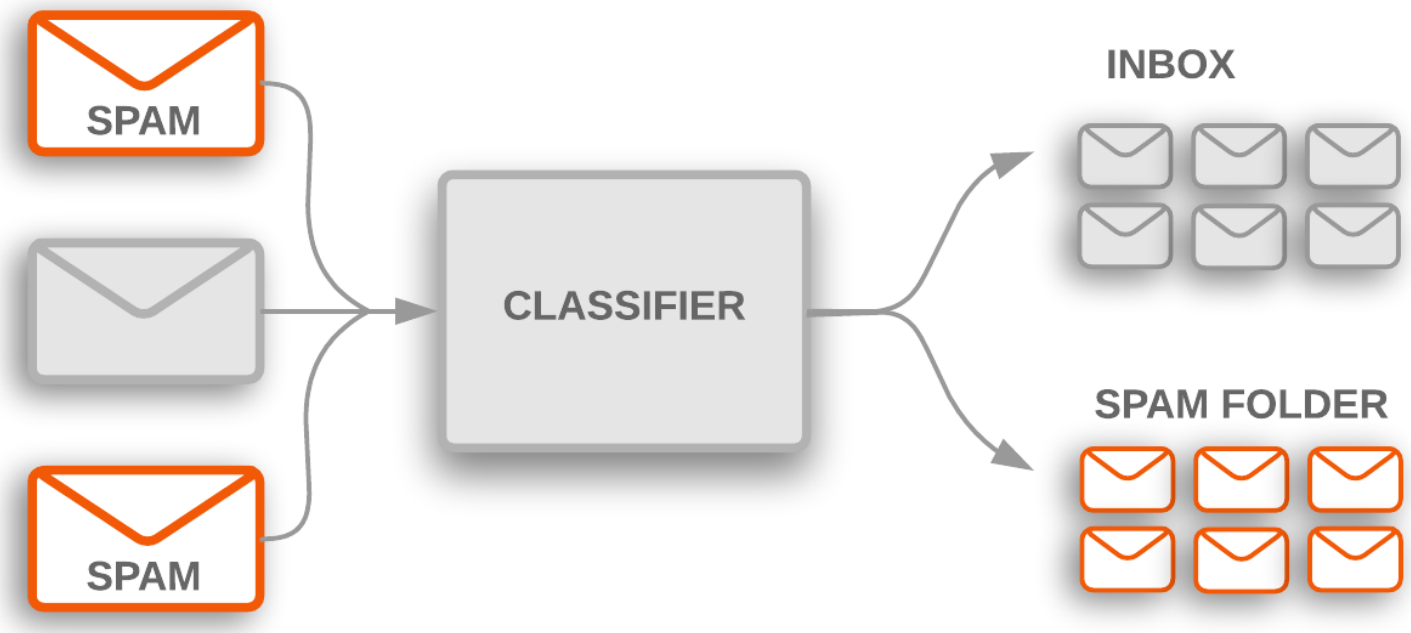
- The main bibliographic references are available online for free
  - **Foundations of Machine Learning** - M. Mohri, A. Rostamizadeh, A. Talwalkar
  - **Understanding Machine Learning, From Theory to Algorithms** - S. S. Shwartz, S. Ben-David

- For the implementation we will use:
  - **Python3**
  - **CvxPy** as modeling language (Python3 package)
  - **Mosek** Optimization solver (free academic license available)

What is Machine Learning

- With Learning we denote the process that convert experience into knowledge
  - The experience is represented by the input given by the user, i.e., training data
  - The knowledge can be, for example, a program that performs a certain action
- How can we evaluate the quality of the program obtained after the training phase?

Example: Spam and not Spam emails



Example: Rats learning how to avoid poisonous food

Example: Pigeon's superstition

The need of knowledge a priori

- The difference between the pigeon's superstition and the rat's learning is given by the a priori knowledge that the rats have on the consequences of the food
- It is necessary to insert a certain amount of such knowledge in the training phase so that the algorithm doesn't arrive to nonsense conclusions
- The more knowledge we give to the algorithm the easier it is to training it
- However, having too much knowledge implies a "rigid" training phase, not able to generalize

Ingredients of a Machine Learning algorithm

- **Sample** X: objects or data used to train or evaluate the algorithm (collection of emails)
- **Features**: attributes associated to the sample, often represented as elements of a vector space (object, length of the email, keywords ...)
- **Labels** Y: categories associated to the samples (Spam or not Spam)
- **Hyper-parameters**: free parameters not determined by the algorithm but specified as input
- **Training sample**: $X_{\text{tr}} \subseteq X$ used only for the training phase (subset of the emails together with the corresponding labels)

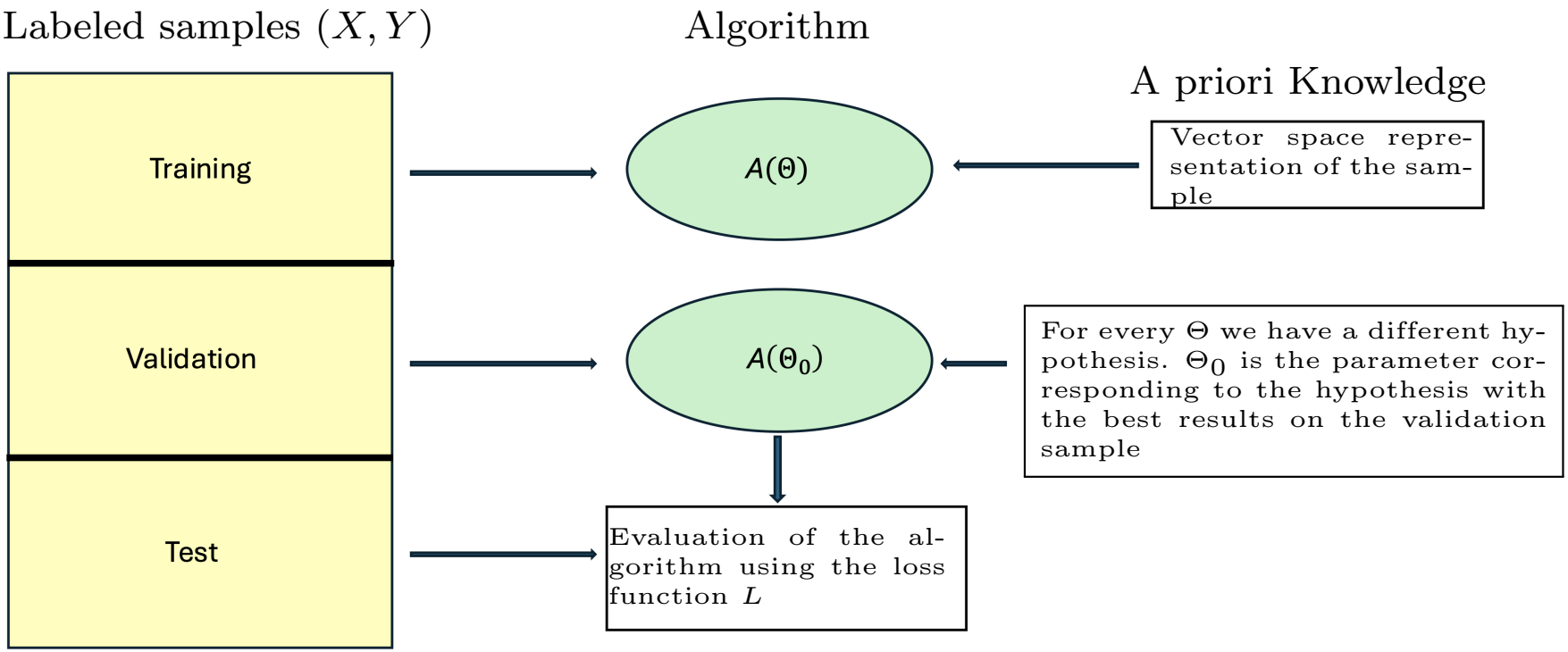
Ingredients of a Machine Learning algorithm

- **Validation sample**: $X_{\text{val}} \subseteq X$ used to estimate the best hyper-parameters
- **Test sample**: $X_{\text{tes}} \subseteq X$ used to evaluate the results of the algorithm and not used in the training phase
- **Loss function**: function that measures the difference between a label predicted by the algorithm and the "true" label. If $\tilde{Y}$ is the set of predicted labels, the loss function has the form

$$L : Y \times \tilde{Y} \to \mathbb{R}_+$$

- **Hypothesis set**: set of functions that map the features of the data in Y

Learning steps



Different type of learning

- **Supervised**: There are all the labels of the input data and therefore a ground truth to compare with
- **Non supervised**: There are no labels in input. It is more difficult to evaluate the algorithm
- **Semi-supervised**: There are labels only for some of the input data
- **Reinforced**: The training and testing phase are combined. In order to collect information, the algorithm performs some action and gets a rewards for each of them. The goal is to maximize the total reward.

Generalization

The generalization is a fundamental aspects of Machine Learning: we want the algorithm to be able to generalize the informations obtained from the input on data that it never saw before

- An Hypothesis set too complex can yield to a classification function too close to the training data and unable to generalize (overfitting)
- An Hypothesis set too simple can yield to a large error on training data (underfitting)