

---

ANNO ACCADEMICO 2025/2026

---

# Apprendimento Automatico

---

## Teoria

Altair's Notes



**UNIVERSITÀ  
DI TORINO**



---

DIPARTIMENTO DI INFORMATICA

---



**CAPITOLO 1****INTRODUZIONE****PAGINA 5**

1.1 Che Cos'è l'Apprendimento Automatico?

5

Terminologia — 5 • Tasks — 7 • Spazio di Ipotesi — 7



# Premessa

## Licenza

Questi appunti sono rilasciati sotto licenza Creative Commons Attribuzione 4.0 Internazionale (per maggiori informazioni consultare il link: <https://creativecommons.org/licenses/by/4.0/>).



## Formato utilizzato

Box di "Concetto sbagliato":

### Concetto sbagliato 0.1: Testo del concetto sbagliato

Testo contenente il concetto giusto.

Box di "Corollario":

### Corollario 0.0.1 Nome del corollario

Testo del corollario. Per corollario si intende una definizione minore, legata a un'altra definizione.

Box di "Definizione":

### Definizione 0.0.1: Nome delle definizioni

Testo della definizione.

Box di "Domanda":

### Domanda 0.1

Testo della domanda. Le domande sono spesso utilizzate per far riflettere sulle definizioni o sui concetti.

Box di "Esempio":

### Esempio 0.0.1 (Nome dell'esempio)

Testo dell'esempio. Gli esempi sono tratti dalle slides del corso.

**Box di "Note":**

**Note:-**

Testo della nota. Le note sono spesso utilizzate per chiarire concetti o per dare informazioni aggiuntive.

**Box di "Osservazioni":**

**Osservazioni 0.0.1**

Testo delle osservazioni. Le osservazioni sono spesso utilizzate per chiarire concetti o per dare informazioni aggiuntive. A differenza delle note le osservazioni sono più specifiche.



# 1

## Introduzione

### 1.1 Che Cos'è l'Apprendimento Automatico?

#### Definizione 1.1.1: Machine Learning

Un programma informatico apprende dall'esperienza  $E$  rispetto a una classe di task  $T$  e una performance  $P$ , se la sua performance nel task  $T$ , misurata da  $P$ , aumenta con l'esperienza  $E$ .

Sostanzialmente:

- L'esperienza viene data sotto forma di esempi "risolti" al computer.
- Un task (compito) da risolvere.
- Con un modo per valutare la risoluzione (performance).

#### 1.1.1 Terminologia

Attributes/Features				Label/Class   $y$
ID	Color	Root	Sound	Ripe
Training set	1	green	curly	muffled
	2	dark	curly	muffled
	3	green	straight	crisp
	4	dark	slightly curly	dull
Test set		1	green	curly
Sample/Instance   $x_i$				?
Attribute value/Feature value   $x_{i1}$				

Figure 1.1: Terminologia.

- Attributi (Features): le colonne.
- Etichetta (Classe): elemento che indica come risolvere un task.
- Istanza (Sample): una riga.
- Valori: le celle.
- Set di Training: insieme su cui si va a dedurre una regola per classificare.
- Test Set: insieme per vedere quanto si sarà accurati su insiemi futuri.

**Note:-**

Si usa un set diverso per il training e il test perché se si usasse lo stesso il modello farebbe risultati elevati essendo addestrato su quello.

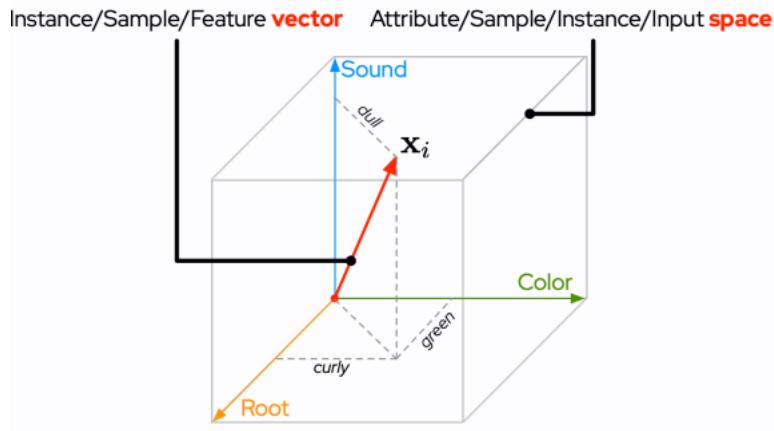


Figure 1.2: Terminologia 2.

Immaginando gli oggetti in un qualche campo euclideo:

- *Features vector*: ogni esempio corrisponde a un vettore.
- *Attribute space*: l'insieme di tutti gli esempi.

Symbol	Meaning
$\mathcal{X}$	instance space, in many cases $\mathcal{X} = \mathbb{R}^d$
$\mathcal{Y}$	label space, e.g., $\mathcal{Y} = \{0, 1\}$ for binary classification
$\mathbf{x}_i \in \mathcal{X}$	$i$ -th instance/sample, $\mathbf{x}_i = [x_{i1}, x_{i2}, \dots, x_{id}]^\top$
$d$	dimensionality of the instance space
$y_i \in \mathcal{Y}$	label of the $i$ -th instance, e.g., $y_i = 1$ if the watermelon is ripe, $y_i = 0$ otherwise
$D$	Dataset, $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}$
$n$	number of instances in the dataset
$h$	model/hypothesis, a function $h : \mathcal{X} \mapsto \mathcal{Y}$ that maps instances to labels
$\mathcal{L}$	Learning algorithm $\mathcal{L} : \mathcal{P}(\mathcal{X} \times \mathcal{Y}) \rightarrow \mathcal{H}$ , where $\mathcal{P}$ is the power set and $\mathcal{H}$ is the hypothesis space
$\mathbb{I}$	Indicator function, $\mathbb{I}(P)$ (sometime denoted as $\mathbb{I}[P]$ ) is 1 if $P$ is true, 0 otherwise

Figure 1.3: Tabella di riferimento.

**Definizione 1.1.2: Learning (Training)**

Il learning è un processo in cui si usano algoritmi di apprendimento automatico per costruire dei modelli.

- I dati utilizzati in questo processo sono detti training data.
- Ogni istanza è un training example.
- L'insieme di tutti i training example è il training set.

**Note:-**

Un modello addestrato corrisponde a una serie di regole sui dati, quindi si chiama anche *ipotesi* e le regole sono i *fatti* (grounded-truth).

**1.1.2 Tasks****Definizione 1.1.3: Tasks predittivi**

Un task predittivo è focalizzato sul prevedere una variabile sulla base degli esempi. Si parte da problemi vecchi per trovare la soluzione a nuovi problemi.

I tasks predittivi possono essere:

- *Binari e Multi-classe*: di categorizzazione.
- *Regressivi*: con un target numerico.
- *Clustering*: un target sconosciuto.

**Definizione 1.1.4: Tasks descrittivi**

Un task descrittivo si concentra sul fornire regolarità nel dataset.

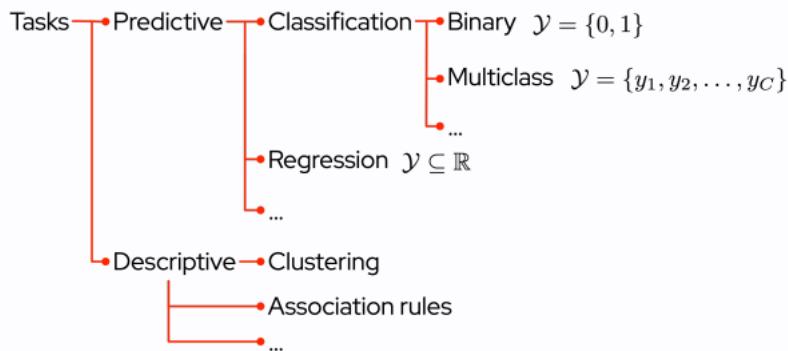


Figure 1.4: Vari tasks.

**Assunzione:**

- Si assume che i dati siano *indipendenti*.
- Si assume che i dati siano *identicamente distribuiti*.

**1.1.3 Spazio di Ipotesi**

Nei sistemi *assiomatici* il processo di derivare un teorema da assiomi è detto *deduzione*. Questo è un processo corretto se si assume che gli assiomi siano veri. L'*induzione* è il processo opposto ed è il principio su cui si basa tutto l'apprendimento automatico.

**Note:-**

ATTENZIONE: l'induzione come intesa in questo corso non è l'induzione matematica.

**Osservazioni 1.1.1 Deduzione vs. Induzione**

- La deduzione è valida: assumere le premesse vere garantisce che le conclusioni siano vere.
- L'induzione non è valida come forma di ragionamento: non garantisce che la conclusione sia vera anche se tutte le osservazioni sono corrette.

**Definizione 1.1.5: Boolean Concept Learning**

L'obiettivo è quello di apprendere una funzione booleana  $h : \mathcal{X} \mapsto \{0, 1\}$

**Note:-**

Siamo nel campo del *symbolic concept learning*, studiato nel campo della *inductive logic programming*.

**Definizione 1.1.6: Spazio di Ipotesi**

Lo spazio di Ipotesi è l'insieme di tutte le possibili ipotesi che possono essere imparate da un algoritmo di apprendimento.

**Note:-**

Il Machine Learning è la ricerca attraverso lo spazio delle ipotesi per trovare l'insieme di tutte le ipotesi che sono consistenti con i training data e selezionare i migliori secondo un qualche criterio.



