UNIVERSITÀ DEGLI STUDI DI TERAMO

Nuove Tecnologie ICT

Introduzione all'Intelligenza Artificiale

Prof. ssa Romina Eramo
Università degli Studi di Teramo
Dipartimento di Scienze della Comunicazione
reramo@unite.it

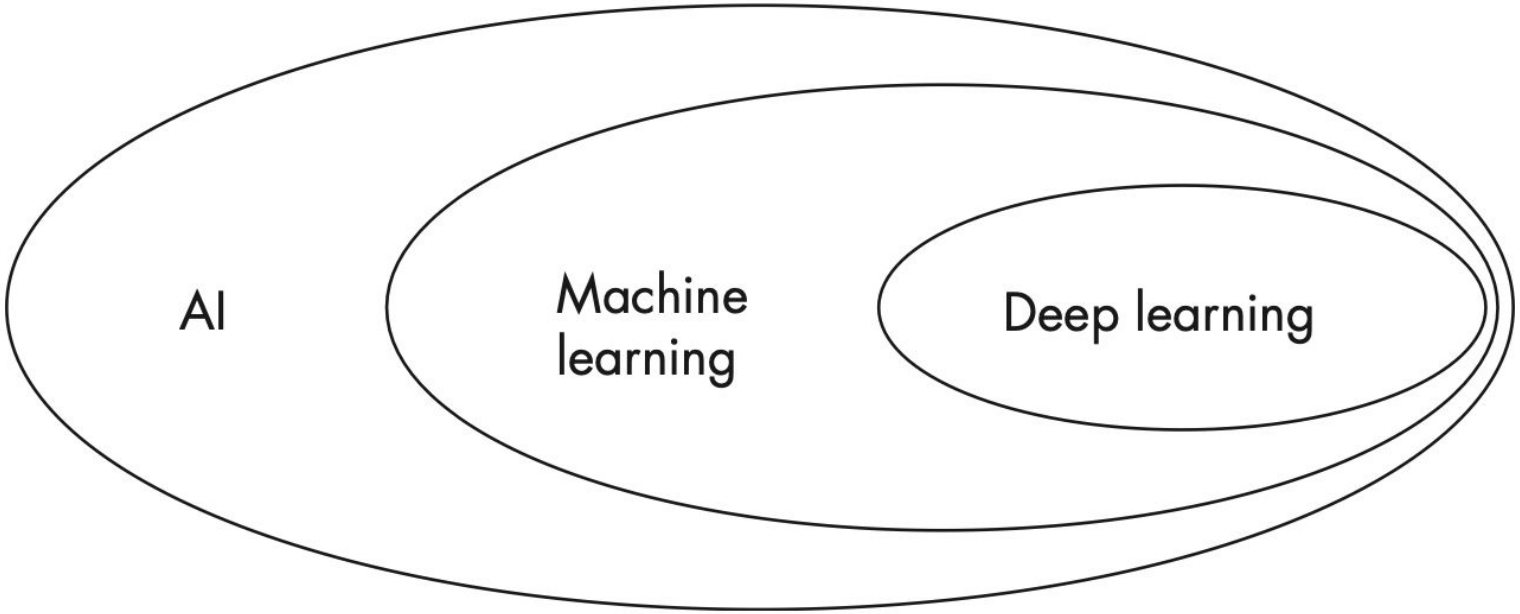
Panoramica sull'IA

L'intelligenza artificiale (IA) tenta di convincere una macchina, in genere un computer, a comportarsi in modi che gli umani giudicano intelligenti. La frase è stata coniata negli anni '50 dal famoso informatico John McCarthy (1927-2011).

Panoramica sull'IA

- I *computer* sono programmati per svolgere un compito specifico, fornendo loro una sequenza di istruzioni, un programma, che incarna un algoritmo, o la ricetta che il programma fa eseguire al computer.
- L'*algoritmo* è un elenco di istruzioni dettagliate, elaborate per svolgere una determinata attività o risolvere un problema specifico.
- Un essere umano concepisce un algoritmo, quindi traduce l'algoritmo in una sequenza di passaggi (un *programma*). La macchina esegue il programma, implementando così l'algoritmo. La macchina non capisce cosa sta facendo; sta semplicemente eseguendo una serie di istruzioni primitive.

Relazione tra Intelligenza Artificiale, Machine Learning e Deep Learning



- Il Deep Learning è un sottocampo del Machine Learning, che è un sottocampo dell'Intelligenza Artificiale
- Questa relazione implica che l'IA implichi concetti che non sono né ML né DL

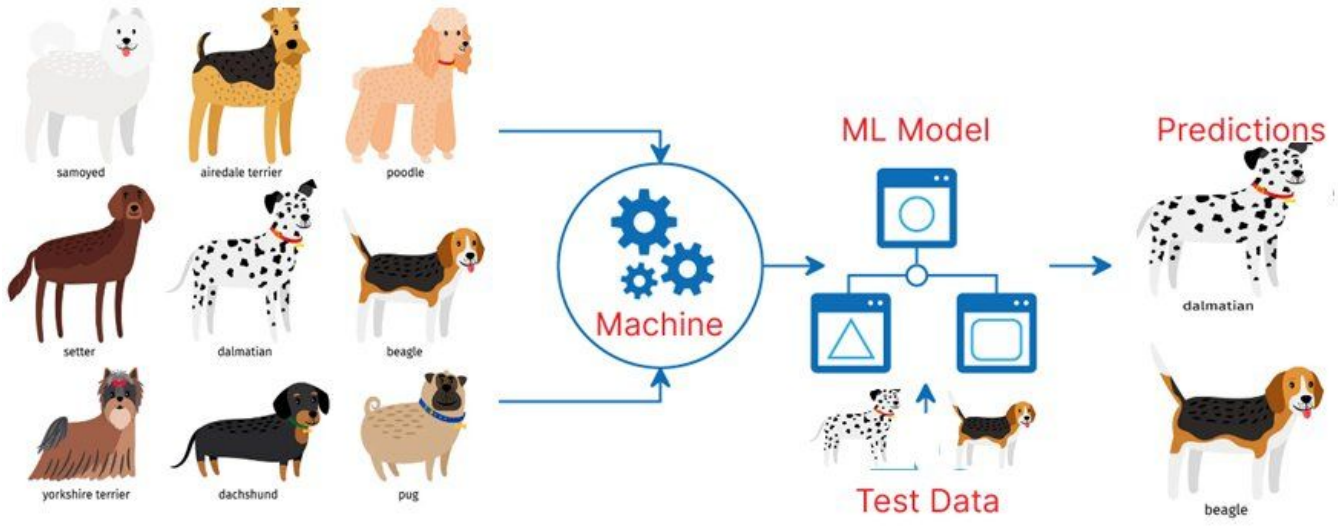
Panoramica sull'IA

Machine Learning (ML) costruisce modelli a partire dai dati.

- In ML, un *modello* è una nozione astratta di qualcosa che accetta input e genera output, dove input e output sono correlati in qualche modo significativo.
- L'obiettivo principale del ML è condizionare un modello utilizzando **dati noti** (*known data*) in modo che il modello produca output significativi quando vengono forniti **dati sconosciuti** (*unknown data*).

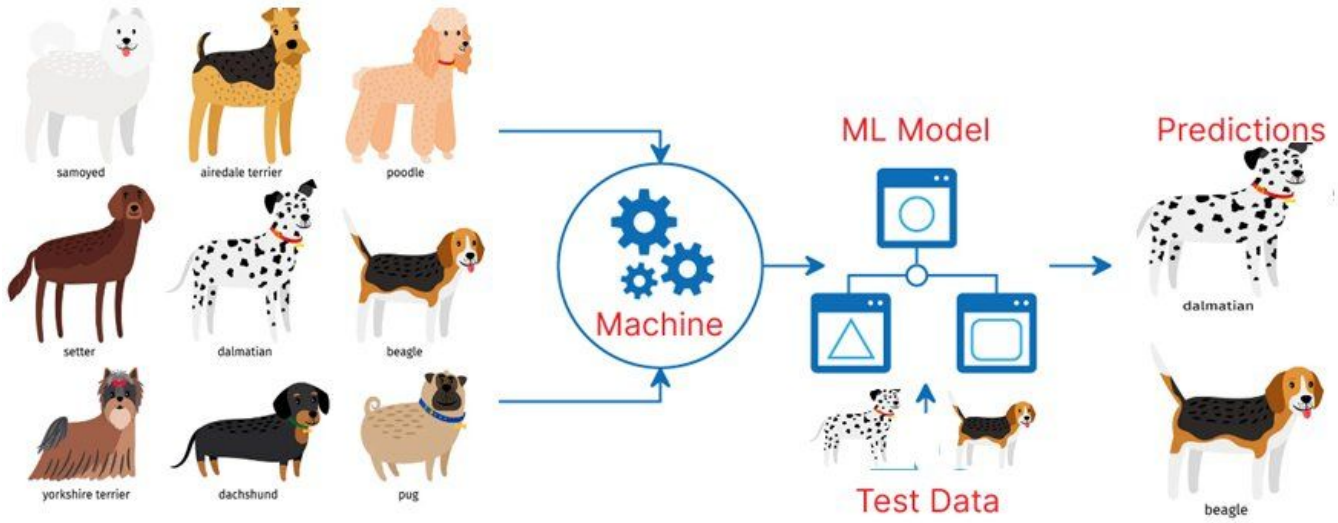
Deep learning si avvale di modelli di grandi dimensioni, che in passato erano troppo grandi per essere utilizzati.

Panoramica sull'IA



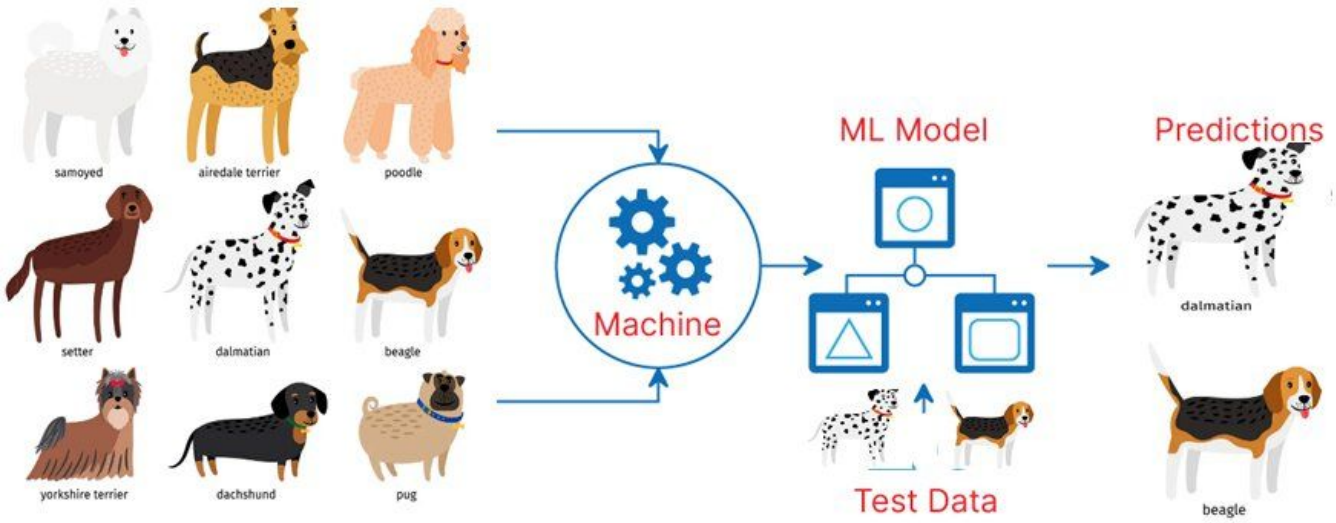
» I dati sono tutto nell'intelligenza artificiale.
  - Il modello è una tabula rasa che i dati devono condizionare per renderlo adatti a un compito.
  - Se i dati sono scadenti, il modello è scadente (da qui dati "buoni" e "cattivi").

Panoramica sull'IA



- Un modello di ML è una scatola nera che
  - accetta un input, solitamente una raccolta di numeri,
  - e produce un output, solitamente un'etichetta come "cane" o "gatto", o un valore continuo come la probabilità di essere un "cane" o il valore di una casa con le caratteristiche fornite al modello (dimensioni, numero di bagni, codice postale e così via).

Panoramica sull'IA



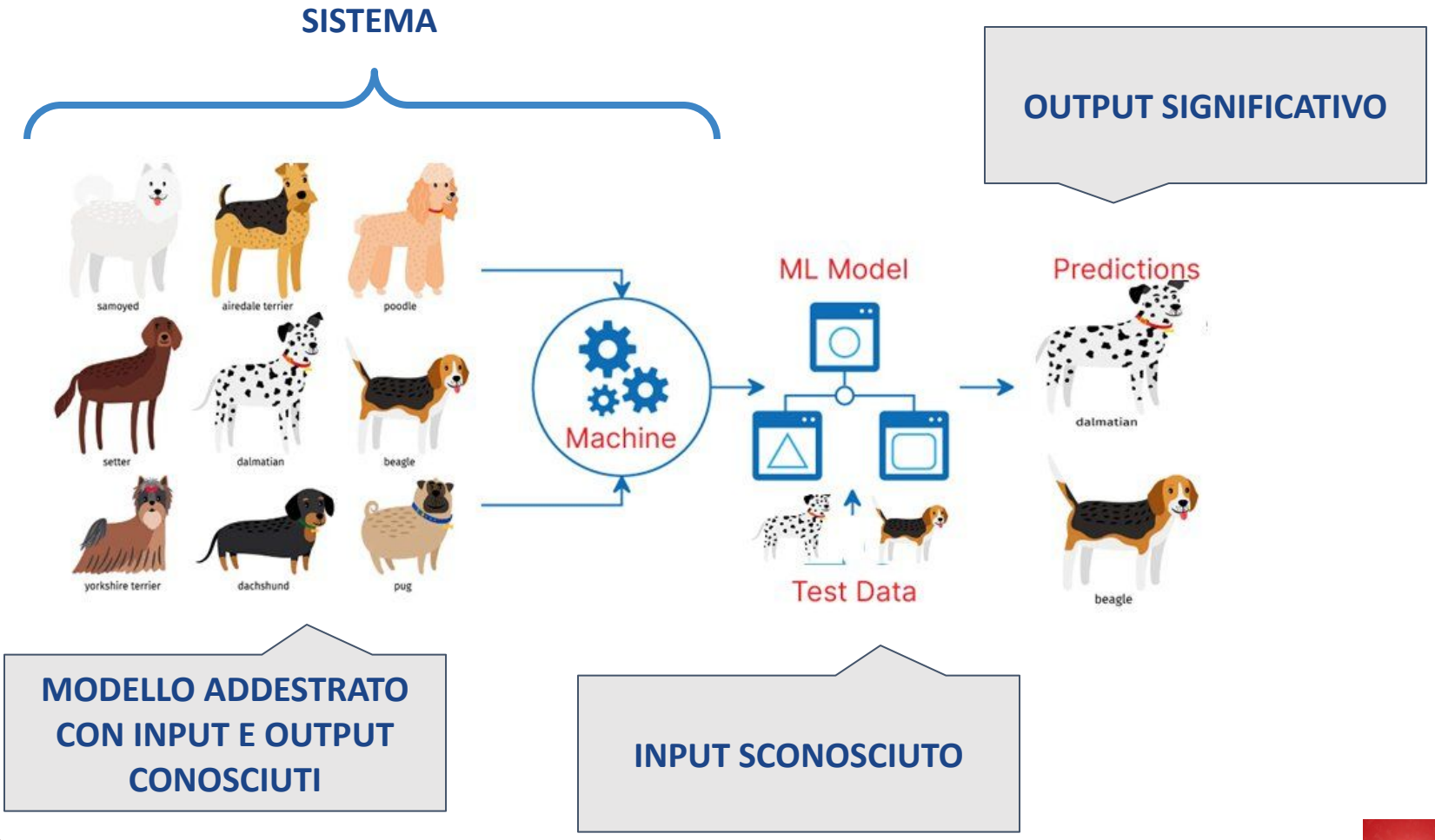
» Il modello ha dei **parametri**, che controllano l'output del modello. Il condizionamento di un modello, noto come addestramento (*training*), cerca di impostare i parametri del modello in modo tale che producano l'output corretto per un dato input.

Panoramica sull'IA

1. Raccogli un set di dati di addestramento costituito da una raccolta di input per il modello e dagli output che ci aspettiamo dal modello per quegli input.
2. Seleziona il tipo di modello che vogliamo addestrare.
3. Addestra il modello presentando gli input di addestramento e regolando i parametri del modello quando sbaglia gli output.
4. Ripeti il passaggio 3 finché non siamo soddisfatti delle prestazioni del modello.
5. Utilizza il modello ora addestrato per produrre output per nuovi input sconosciuti.

Supervised learning: Utilizziamo dati etichettati noti per addestrare il modello. “Supervisioniamo” il modello mentre impara a produrre output corretti.

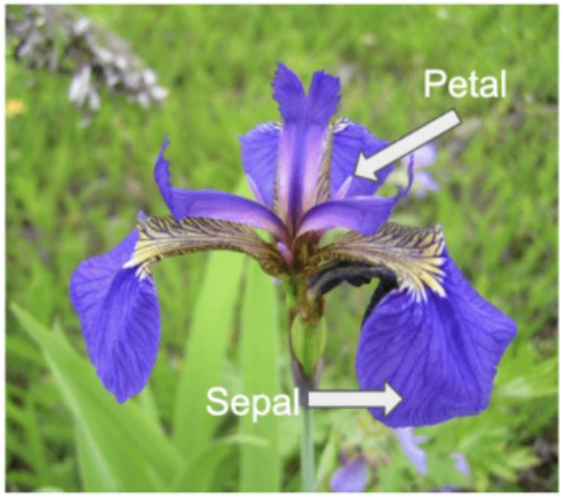
Panoramica sull'IA



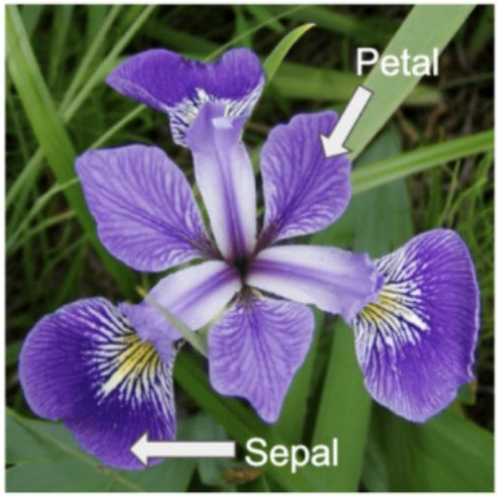
ESEMPIO

Iris Flower Classification

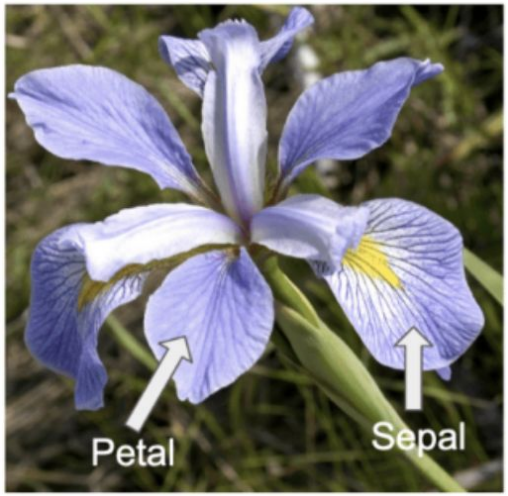
Iris setosa



Iris versicolor



Iris virginica



Problema: Il fiore di iris si divide in tre specie primarie (Iris setosa, Iris versicolor e Iris virginica) che differiscono per lunghezza e larghezza del sepalo, lunghezza e larghezza del petalo.

Obiettivo: Sviluppare un modello di ML in grado di apprendere dalle misurazioni dei fiori di iris e automatizzare il processo di classificazione in base alle caratteristiche distinte di ciascuna specie di iris.

https://github.com/Apaulgithub/oibsip_taskno1

ESEMPIO

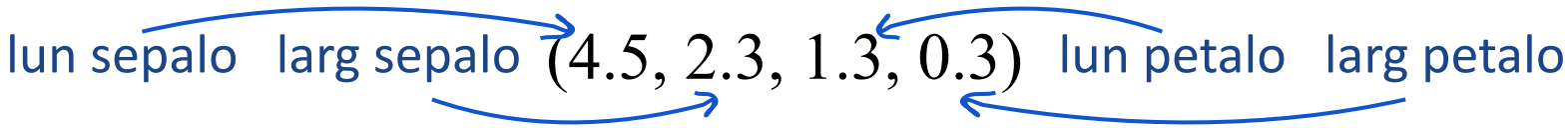
Iris Flower Classification

Vettore: è una stringa di valori trattati come un'unica entità.
Es, possiamo raggruppare le misure di un Iris

$$(4.5, 2.3, 1.3, 0.3)$$

Iris Flower Classification

Vettore: è una stringa di valori trattati come un'unica entità.
Es, possiamo raggruppare le misure di un Iris



Iris Flower Classification

Vettore: è una stringa di valori trattati come un'unica entità. Es, possiamo raggruppare le misure di un Iris

lun sepalo larg sepalo $(4.5, 2.3, 1.3, 0.3)$ lun petalo larg petalo

Matrice: in ML, spesso i set di dati sono rappresentati come matrici, dove le righe sono vettori che rappresentano gli elementi del set di dati, come un fiore di iris, e le colonne sono le misurazioni.

$$\begin{bmatrix} 4.5 & 2.3 & 1.3 & 0.3 \\ 5.6 & 2.9 & 3.6 & 1.3 \\ 5.7 & 4.4 & 1.5 & 0.4 \\ 6.7 & 3.1 & 4.4 & 1.4 \\ 4.6 & 3.1 & 1.5 & 0.2 \end{bmatrix}$$

ESEMPIO

Iris Flower Classification

Vettore: è una stringa di valori trattati come un'unica entità. Es, possiamo raggruppare le misure di un Iris

lun sepalo larg sepalo $(4.5, 2.3, 1.3, 0.3)$ lun petalo larg petalo

Matrice: in ML, spesso i set di dati sono rappresentati come matrici, dove le righe sono vettori che rappresentano gli elementi del set di dati, come un fiore di iris, e le colonne sono le misurazioni.

misurazioni

set di dati

$$\begin{bmatrix} 4.5 & 2.3 & 1.3 & 0.3 \\ 5.6 & 2.9 & 3.6 & 1.3 \\ 5.7 & 4.4 & 1.5 & 0.4 \\ 6.7 & 3.1 & 4.4 & 1.4 \\ 4.6 & 3.1 & 1.5 & 0.2 \end{bmatrix}$$

Iris Flower Classification

Vettore: è una stringa di valori trattati come un'unica entità. Es, possiamo raggruppare le misure di un Iris

lun sepalo larg sepalo $(4.5, 2.3, 1.3, 0.3)$ lun petalo larg petalo

Matrice: in ML, spesso i set di dati sono rappresentati come matrici, dove le righe sono vettori che rappresentano gli elementi del set di dati, come un fiore di iris, e le colonne sono le misurazioni.

- Il numero di elementi in un vettore determina la sua dimensionalità

misurazioni

set di dati

$$\begin{bmatrix} 4.5 & 2.3 & 1.3 & 0.3 \\ 5.6 & 2.9 & 3.6 & 1.3 \\ 5.7 & 4.4 & 1.5 & 0.4 \\ 6.7 & 3.1 & 4.4 & 1.4 \\ 4.6 & 3.1 & 1.5 & 0.2 \end{bmatrix}$$

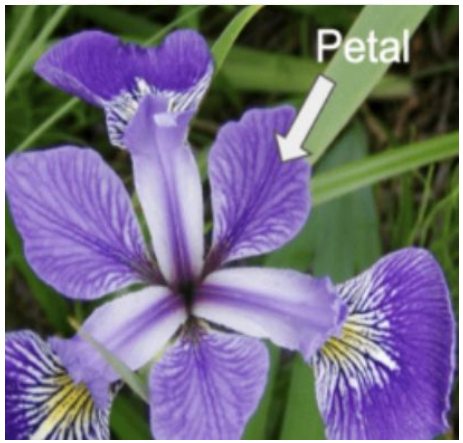
vettore quadridimensionali (quattro misurazioni del fiore)

Iris Flower Classification

Semplificazione

Iris setosa

Iris versicolor



Misurazioni:

- larghezza del petalo
- lunghezza del petalo

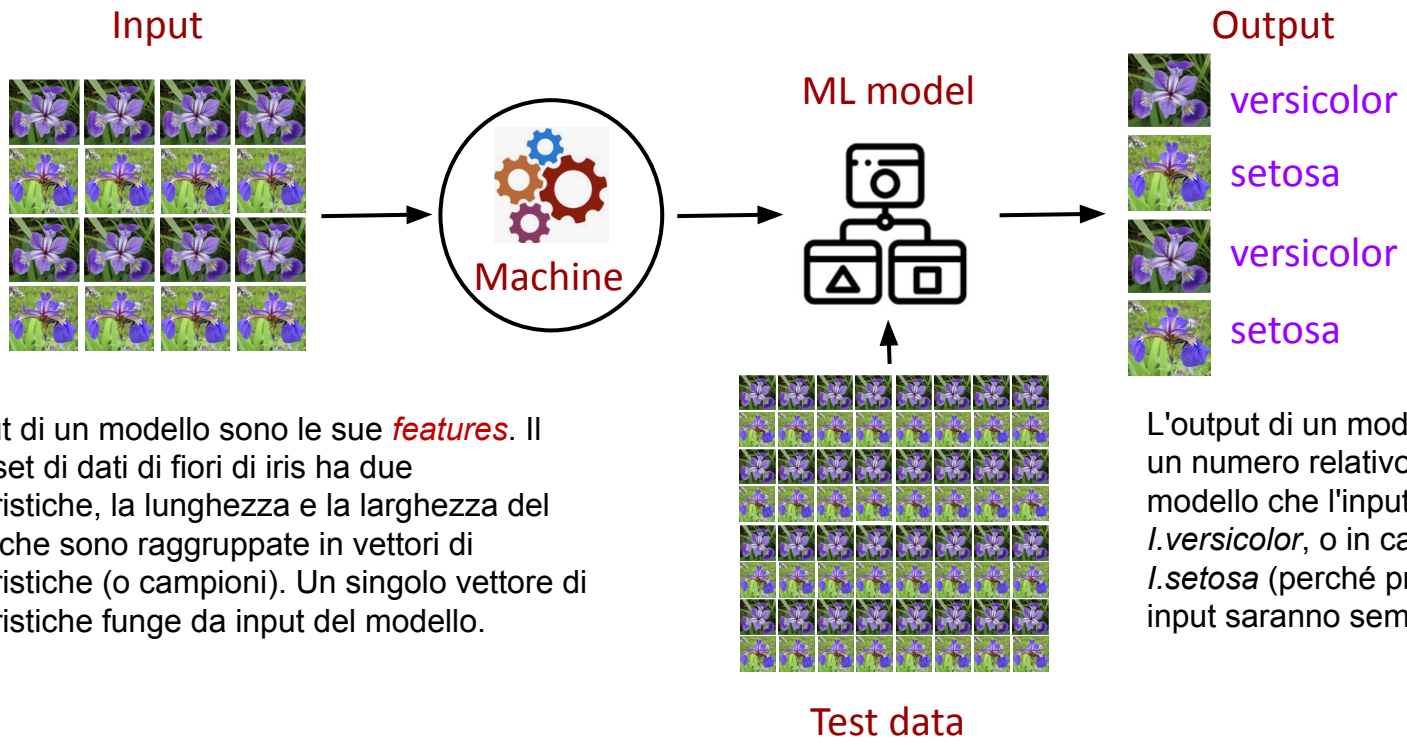
Specie:

- *I.setosa*
- *I.versicolor*

Pertanto, vogliamo che il modello accetti due misurazioni come input e ci fornisca un output che possiamo interpretare come *I.setosa* o *I.versicolor*.

- Modelli binari (come questo) decidono tra due possibili output e sono comunemente usati in IA
- Se il modello decide tra più di due categorie, è un modello multiclasse (multiclass model).

Iris Flower Classification

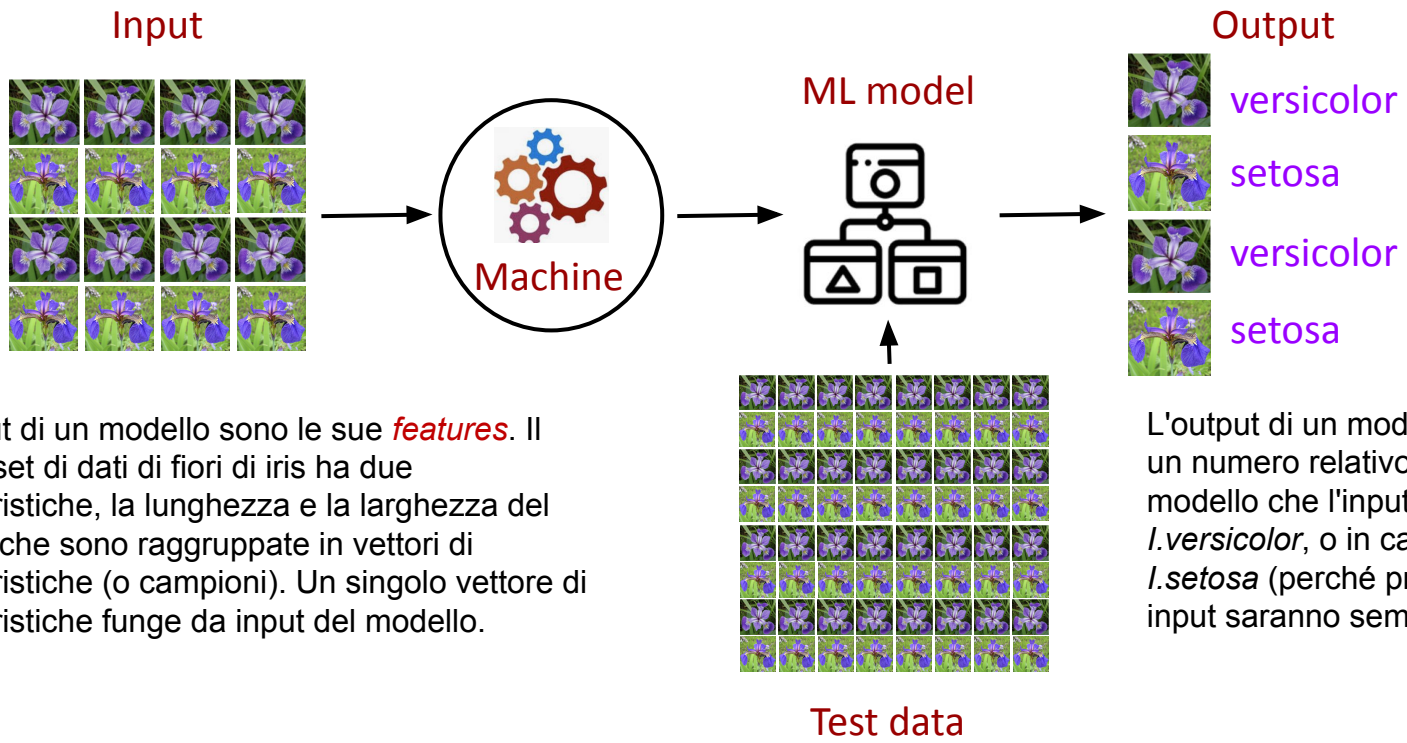


Gli input di un modello sono le sue *features*. Il nostro set di dati di fiori di iris ha due caratteristiche, la lunghezza e la larghezza del petalo, che sono raggruppate in vettori di caratteristiche (o campioni). Un singolo vettore di caratteristiche funge da input del modello.

L'output di un modello binario è in genere un numero relativo alla convinzione del modello che l'input appartenga alla classe *I.versicolor*, o in caso contrario, l'input è *I.setosa* (perché presumiamo che gli input saranno sempre uno o l'altro).

ESEMPIO

Iris Flower Classification



Gli input di un modello sono le sue *features*. Il nostro set di dati di fiori di iris ha due caratteristiche, la lunghezza e la larghezza del petalo, che sono raggruppate in vettori di caratteristiche (o campioni). Un singolo vettore di caratteristiche funge da input del modello.

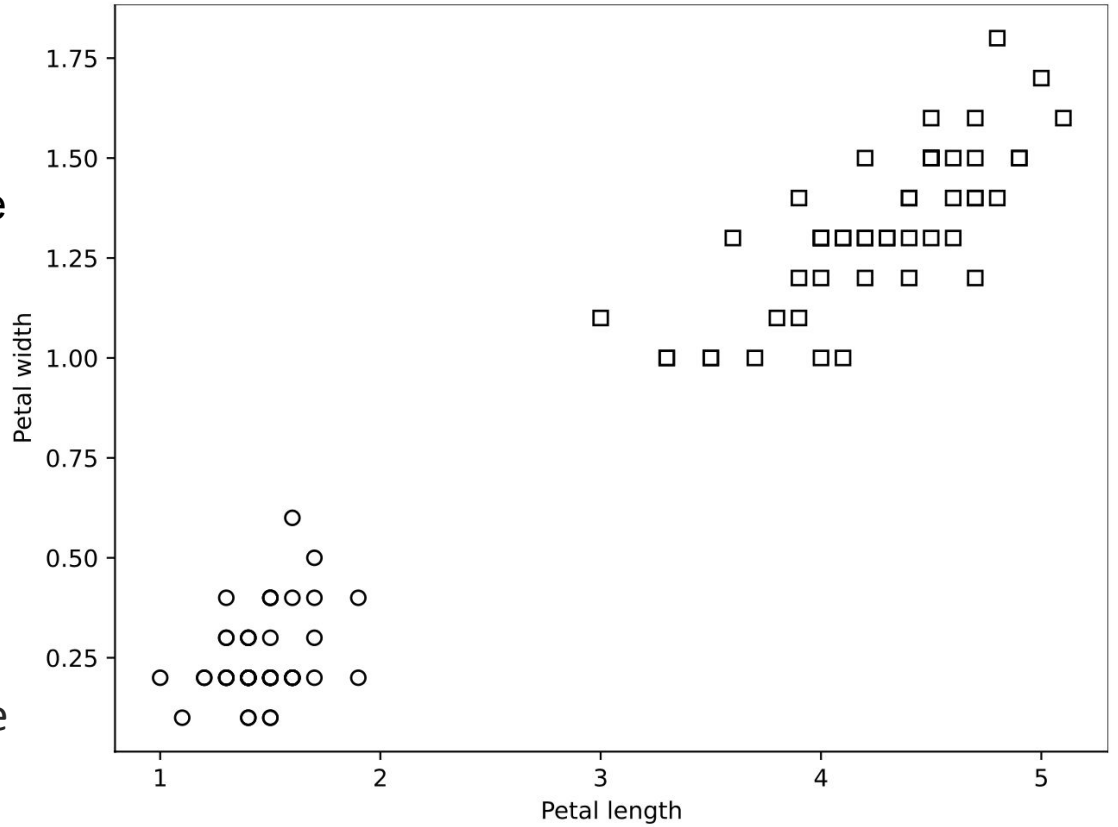
L'output di un modello binario è in genere un numero relativo alla convinzione del modello che l'input appartenga alla classe *I.versicolor*, o in caso contrario, l'input è *I.setosa* (perché presumiamo che gli input saranno sempre uno o l'altro).

Il modo corretto per testare un modello è conservare alcuni dei dati di *training* da utilizzare dopo. Utilizzeremo 80 campioni etichettati per per il training e ne terremo 20 per i test, assicurandoci che sia i set di addestramento che quelli di test contengano un mix approssimativamente uniforme di entrambe le classi (tipi di fiori).

Si vuole un modello che apprenda le caratteristiche generali dei dati di addestramento per generalizzare a nuovi dati. Quindi per il modello, il set di test contiene dati nuovi e invisibili che non ha utilizzato per modificare i suoi parametri. Le prestazioni del modello sul set di test sono un indizio delle sue capacità di generalizzazione

Iris training data (bidimensionali)

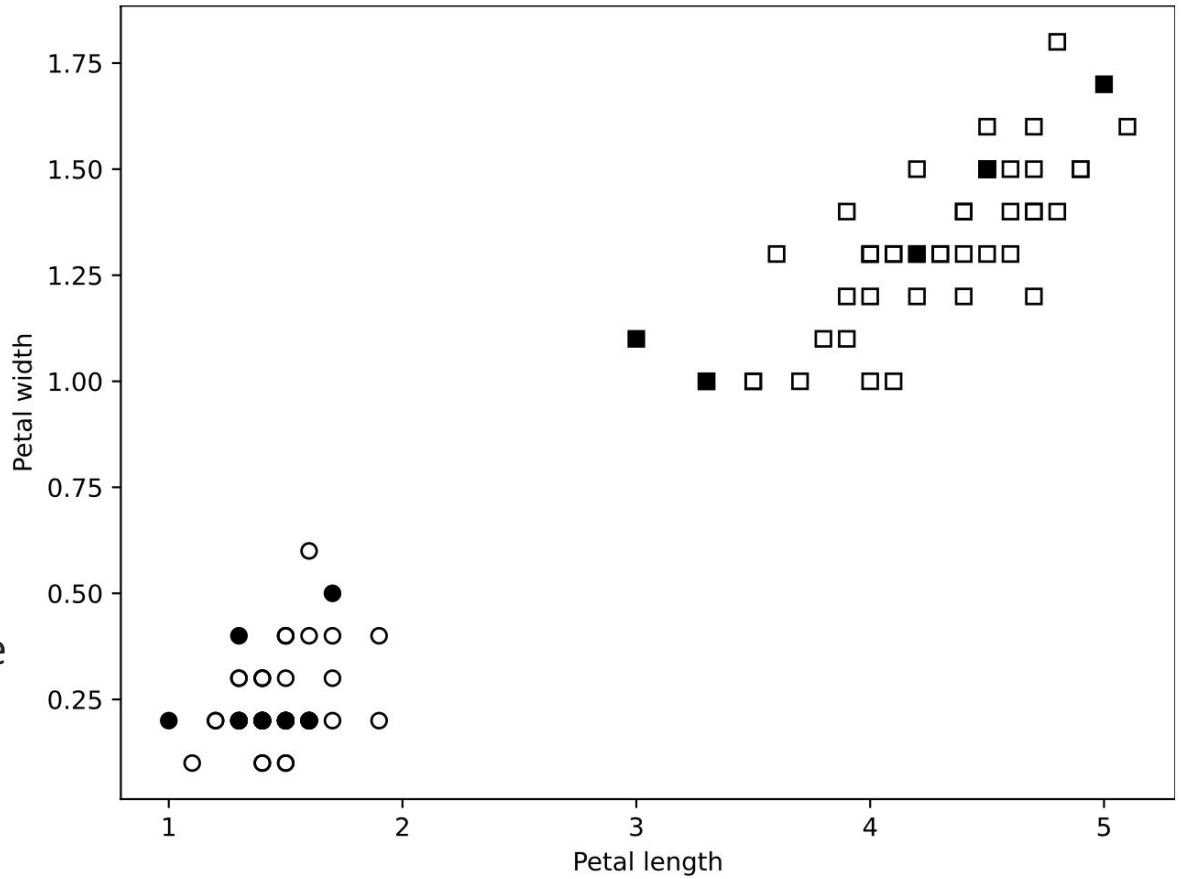
- L'asse x è la lunghezza del petalo e l'asse y è la larghezza del petalo.
- I cerchi corrispondono alle istanze di *I.setosa* e i quadrati a *I.versicolor*.
- Ogni cerchio o quadrato rappresenta un singolo campione di training, la lunghezza e la larghezza del petalo per un fiore specifico.
- Non c'è sovrapposizione tra le classi, sono separate nello spazio delle caratteristiche.



Come classificare? La lunghezza dei petali è inferiore a 2,5 cm? Se la risposta è "sì", allora restituisci classe 0, *I.setosa*; altrimenti, restituisci classe 1, *I.versicolor.*

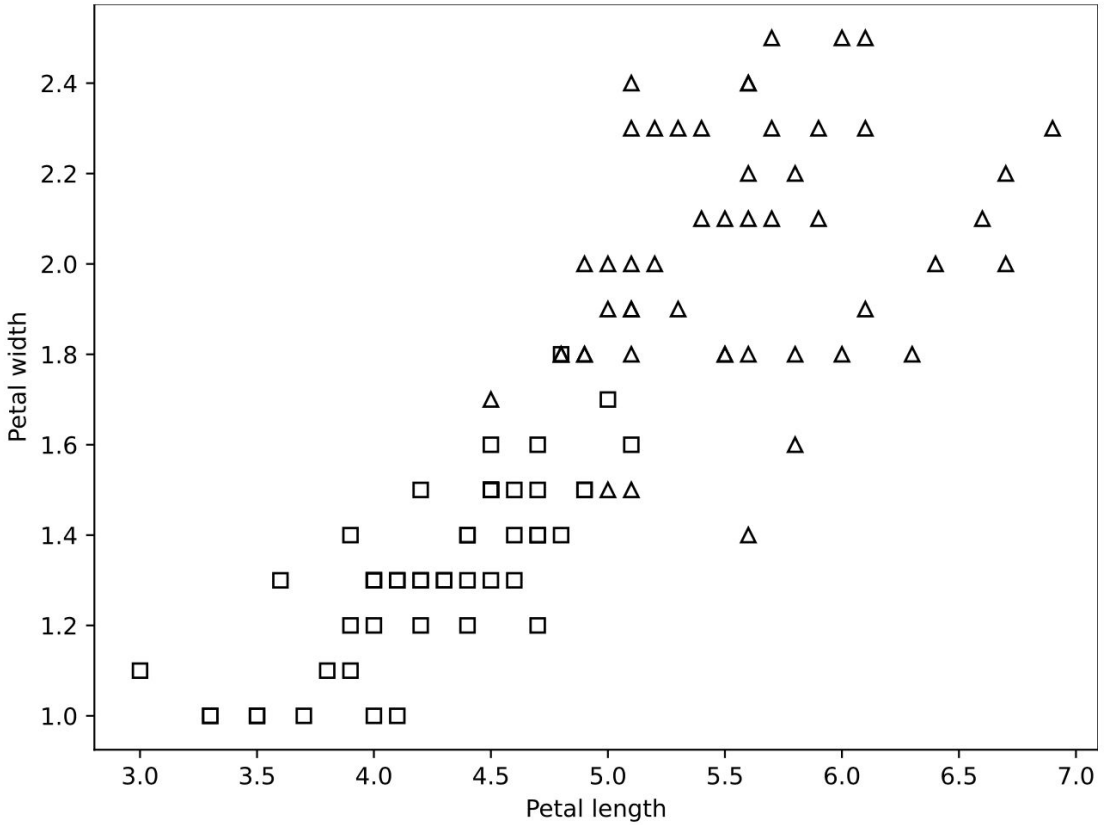
Iris training data

- Si aggiungono i dati di test (cerchi e i quadrati pieni) che non abbiamo utilizzato per creare il classificatore a domanda singola.
- Nessuno dei dati di test viola la nostra regola; otteniamo comunque etichette di classe corrette chiedendo se la lunghezza del petalo è inferiore a 2,5 cm. Pertanto, il nostro modello è perfetto; non commette errori!



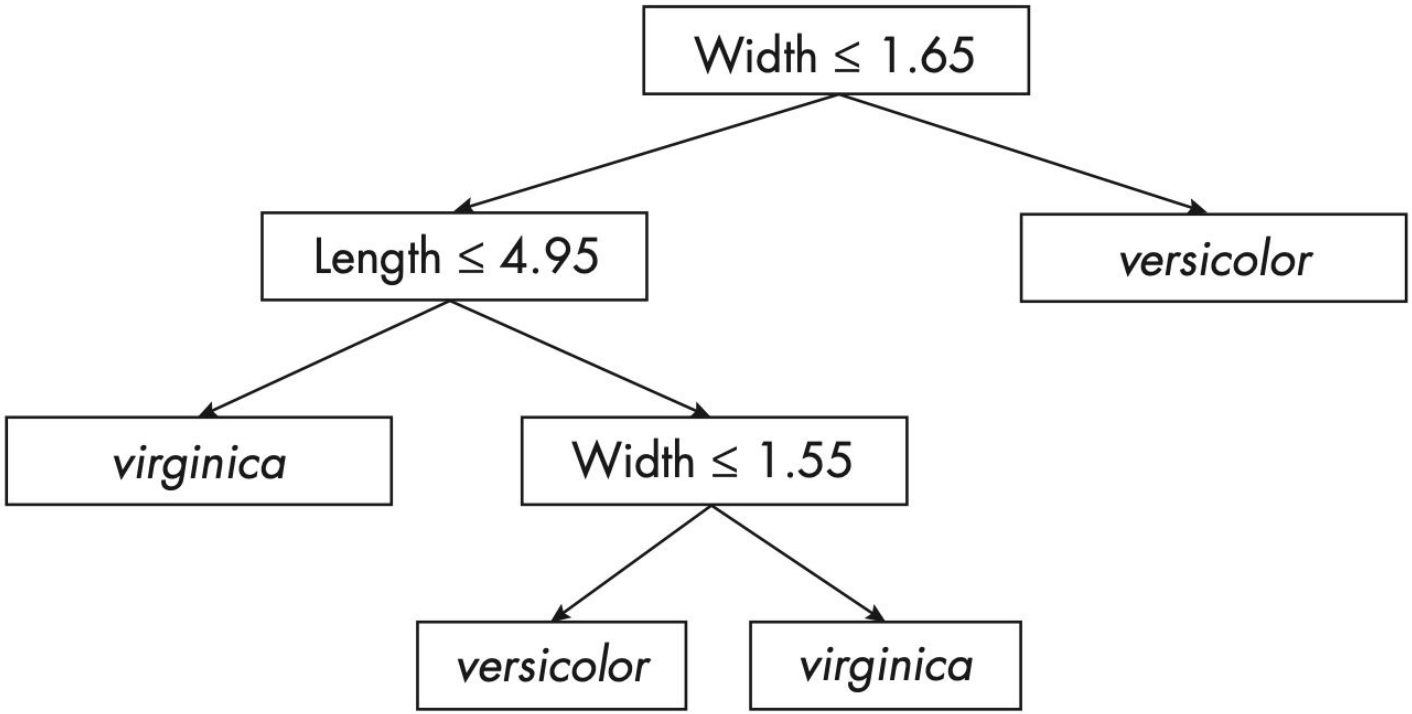
Iris training data

- Ripetiamo l'esercizio, sostituendo *I.setosa* con la specie di iris rimanente, *I.virginica* (triangoli). L'evidente divario tra le classi è scomparso e si sovrappongono.
- Come prima, c'erano 80 campioni per l'addestramento e 20 trattenuti per i test. Questa volta, il modello non era perfetto. Ha etichettato correttamente 18 dei 20 campioni, per una precisione di 9 su 10, o il 90 percento.
- Ciò significa approssimativamente che quando questo modello assegna un fiore a una particolare classe, c'è una probabilità del 90 percento che sia corretto.



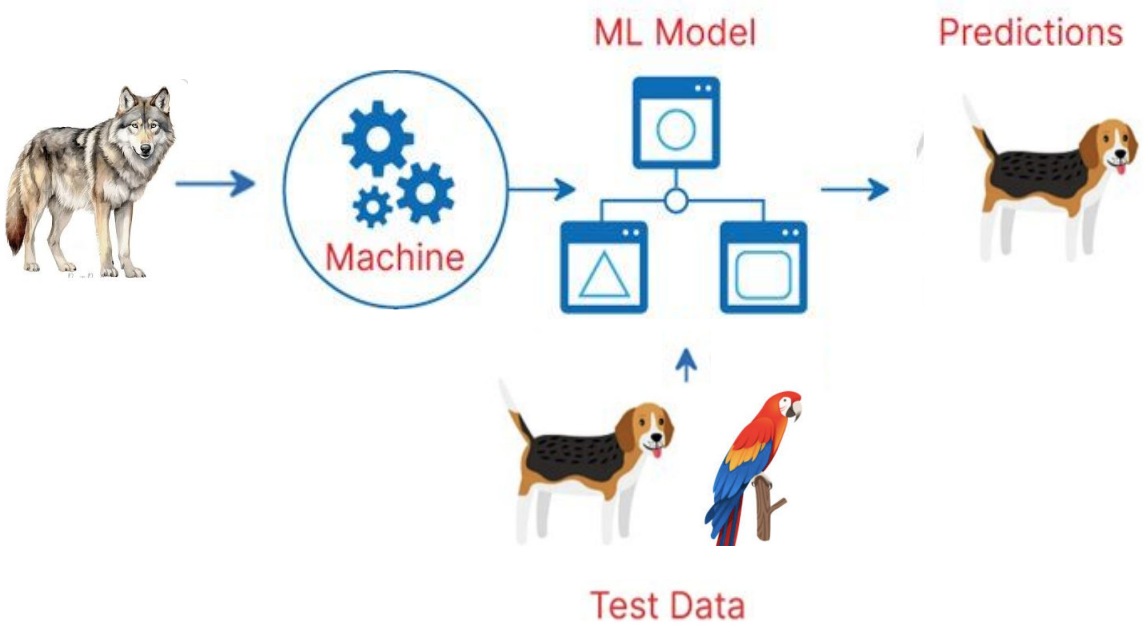
I modelli di apprendimento automatico non sono sempre perfetti; (abbastanza frequentemente) commettono errori.

Iris training data (Albero decisionale)



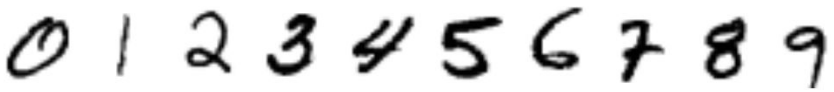
Gamma di input

» Affinché un modello funzioni bene "in natura", ovvero quando viene utilizzato nel mondo reale, i dati utilizzati per addestrare il modello devono coprire l'intera gamma di input che il modello potrebbe incontrare



MNIST dataset

- Dataset di decine di migliaia di piccole immagini contenenti cifre scritte a mano, da 0 a 9.
  - MNIST (Modified NIST) perché derivato da un set di dati costruito dal National Institute of Standards and Technology (NIST)



- **Obiettivo**: Costruire una rete neurale che impari a identificare le cifre 0, 1, 3 e 9.
- Possiamo addestrare Reti Neurali senza sapere come funzionano grazie a potenti toolkit open source (es., scikit-learn)

ESEMPIO

MNIST (Matrice di confusione)

- Le righe della matrice rappresentano le etichette reali per i campioni forniti al modello.
- Le colonne sono le risposte del modello
- I valori nella tabella sono conteggi, il numero di volte in cui si è verificata ogni possibile combinazione di classe di input ed etichetta assegnata dal modello.
- Nel complesso, il modello delle cifre è accurato al 99%!
- **Ma cosa succede se l'input è diverso da 0, 1, 3 o 9?**

Confusion Matrix

	0	1	3	9
0	978	0	1	1
1	2	1,128	3	2
3	5	0	997	8
9	5	1	8	995

MNIST (Cambio di input)

» Dando al modello 982 quattro, si ottiene:

0	1	3	9
48	9	8	917

» E dando sette?

0	1	3	9
19	20	227	762

» Il modello non è stato addestrato a riconoscere i quattro o i sette, quindi ha fatto la cosa migliore e li ha inseriti in una categoria vicina.

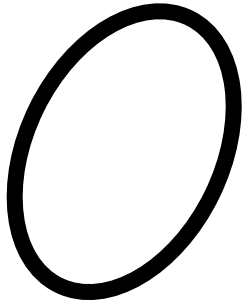
Interpolazione ed estrapolazione

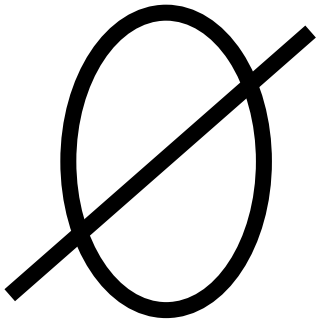
- ***L'interpolazione*** **approssima entro l'intervallo di dati noti e** ***l'estrapolazione*** **va oltre i dati noti.**
- Quando da una legge sperimentale si ricava un valore non misurato, si esegue una:
  - Interpolazione se il valore ricade nell'intervallo dei valori misurati
  - Estrapolazione se il valore ricade al di fuori dell'intervallo dei valori misurati

ESEMPIO

Interpolazione ed estrapolazione

- L'**interpolazione** potrebbe riferirsi all'incontro con uno **zero inclinato** in natura quando nessuno degli zeri nel set di addestramento era particolarmente inclinato
  - il modello deve interpolare, in un certo senso, per rispondere correttamente
- L'**estrapolazione** è più simile alla classificazione di uno **zero con una barra che lo attraversa**
  - che è qualcosa di invisibile durante il periodo di addestramento





ESEMPIO

Interpolazione ed estrapolazione

- Lo stesso processo di generazione dei dati ha creato entrambi.
  - Il modello stava, in un certo senso, interpolando nei primi casi; ma quando abbiamo forzato il modello a prendere decisioni su 4 e 7, stavamo estrapolando facendo in modo che il modello prendesse decisioni su dati che non aveva mai visto durante l'addestramento.
- Va detto che: interpolazione buona, estrapolazione cattiva. Set di dati cattivi portano a modelli cattivi; set di dati buoni portano a modelli buoni, che si comportano male quando sono costretti a estrapolare. E, per buona misura: tutti i modelli sono sbagliati, ma alcuni sono utili.

ESEMPIO

Falsi positivi e falsi negativi

- Nuovo modello: “La cifra di input è un nove?”
  - Il modello è la stessa rete neurale usata in precedenza. Se addestrato su un set di dati in cui ogni immagine è un nove (ovvero, nei dati di addestramento non ci sono quattro o sette), allora il modello è accurato al 99%, come mostra la matrice di confusione:

	Not 9	9
Not 9	9,754	23
9	38	1,362

falsi positivi

falsi negativi

  - In questo caso, la matrice di confusione è piccola perché il modello ha solo due classi: nove o non nove.
  - In altre parole, questo è un modello binario.

ESEMPIO

Falsi positivi e falsi negativi

» Diamo in input gli sconosciuti quattro e sette

	Not 9	9
Not 9	5,014	9,103

Il modello ha etichettato 2/3 come un nove

» Aggiungiamo quattro e sette al set di addestramento (3 percento dei dati di addestramento quattro e sette)

	Not 9	9
Not 9	9,385	3,321

Il modello ha etichettato solo 1/4 come un nove

» Se aumentiamo la proporzione al 18%, il modello classifica erroneamente in meno dell'1% dei casi.

Falsi positivi e falsi negativi

» Poiché i modelli imparano dai dati, **dobbiamo usare set di dati che siano il più completi possibile in modo che i nostri modelli interpolino e non estrapolino.**

Nota:

» Recenti ricerche dimostrano che i moderni modelli di Deep Learning **sono quasi sempre estrapolatori**, ma più gli input sono simili ai dati su cui è stato addestrato il modello, migliori sono le prestazioni!

Explainable AI

- Chiunque cerchi di comprendere, o di lavorare con, l'IA deve prendere a cuore gli avvertimenti sulla **qualità dei dati utilizzati per addestrare** i modelli di IA.
- *Nature Machine Intelligence da Michael Roberts et al., "Common Pitfalls and Recommendations for Using Machine Learning to Detect and Prognosticate for COVID-19 Using Chest Radiographs and TC Scans", 2021.*
  - Gli autori hanno valutato le prestazioni dei modelli di ML progettati per rilevare il COVID-19 nelle radiografie del torace e nelle TC, riducendo il pool iniziale di candidati di oltre 2.000 studi (modelli) a 62 per test rigorosi. Alla fine, gli autori hanno dichiarato che nessuno dei modelli era adatto all'uso clinico a causa di difetti nella costruzione, distorsioni nei set di dati o di entrambi.
- Risultati come questi hanno portato alla creazione di ***explainable AI*** (**IA spiegabile**), un sottocampo che cerca di dare ai modelli la capacità di spiegare se stessi.
  - *Guarda i tuoi dati e cerca di capire, per quanto umanamente possibile, **cosa sta facendo il tuo modello e perché**.*

Risorse

How AI Works, Ronald T. Kneusel, capitolo 1