



Algoritmi per il Machine Learning

Dott. Antonio Giovanni Lezzi



MACHINE LEARNING

Dott. Antonio Giovanni Lezzi



ALGORITMI DI MACHINE LEARNING

- **Percorso di sviluppo**
- Machine Learning
- Forme di apprendimento
- Algoritmo KNN



PERCORSO DI SVILUPPO

- Per affrontare la creazione di una soluzione con un particolare algoritmo, bisogna essere consapevoli di:
- **responsabilità**: chi è responsabile di un errore fatto dal software/algoritmo?
- **ownership dei dati**: chi è il proprietario del dataset utilizzato?
- **privacy**: i clienti dovrebbero essere a conoscenza dell'utilizzo di AI sui loro dati?
- **comunicazione**: come comunicare rischi e benefici?
- **tecnologia e competenze**: dove reperire le risorse necessarie?
- **trasparenza**: come esegue il compito, come vengono prese le decisioni?



PERCORSO DI SVILUPPO

- **adattabilità:** può adattarsi rapidamente a cambiamenti di contesto?
- **flessibilità:** si possono trasferire le conoscenze da un compito a un altro?
- **scalabilità:** continua a funzionare bene anche se aumenta rapidamente la quantità di dati?
- **emergenza:** cosa fare se il sistema mostra comportamenti anomali? Se si sospende l'uso, quali sono le conseguenze, quali alternative bisogna attivare?
- **dubbio:** cosa deve fornire il sistema se non sa offrire una risposta?



PERCORSO DI SVILUPPO

- Durante il processo di creazione di una soluzione contenente AI bisogna muoversi tra le caratteristiche di **qualità del servizio, il costo di produzione, il tempo di sviluppo.**
- Questa problematica è particolarmente sentita nel machine learning, che **richiede un tempo di addestramento non indifferente, con un elevato impiego di risorse.**



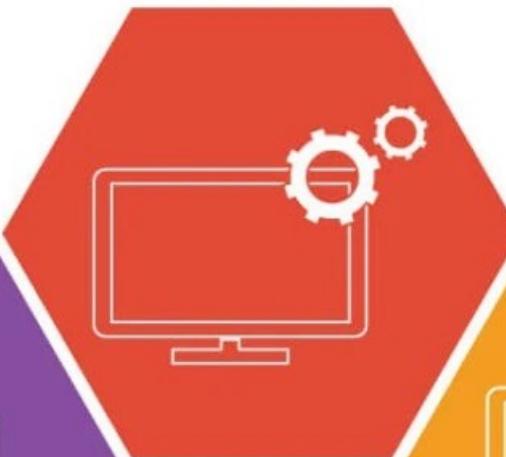
Get Data



1

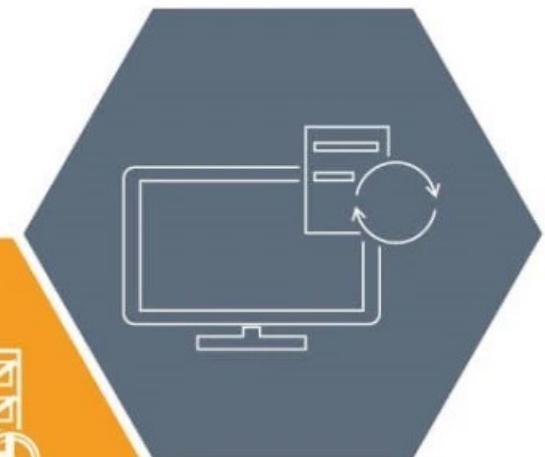
2

Train Model



3

4



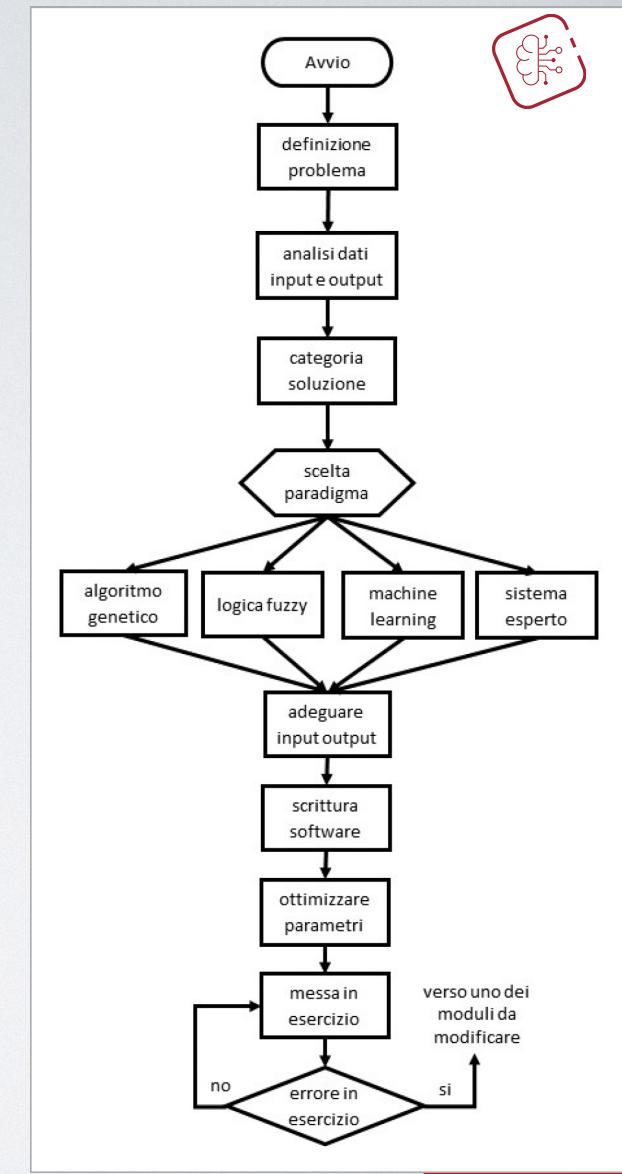
5

Clean, Prepare
& Manipulate Data

Test Data

PERCORSO DI SVILUPPO

- In figura viene indicato il percorso dal problema alla soluzione con un algoritmo basato sulle metodologie di AI
- Si può notare che non esiste la fine della sequenza con una soluzione stabilizzata da non modificare.
- Potrebbero arrivare nuovi dati non previsti nelle analisi precedenti, potrebbero verificarsi guasti, potrebbe essere necessario aumentare o ridurre la soluzione creata.
- Perciò, **il monitoraggio continuo del sistema in esercizio è importante per sapere cosa sta succedendo.**





ADEGUARE INPUT E OUTPUT

- Per cominciare a usare il modello scelto **bisogna adattare i dati al suo metodo computazionale.**
- Per esempio, nel caso delle reti neurali l'input dovrebbe essere ricondotto a valori tra -1 e 1, mentre l'output è un numero da interpretare come una probabilità di appartenenza a una classe, oppure una predizione o altro.
- **Ogni algoritmo scelto ha le sue regole, basta seguirle come previsto.**



OTTIMIZZARE PARAMETRI INPUT E OUTPUT

- Ogni modello ha una serie di parametri che ne regolano il funzionamento,
il cui valore va cambiato con l'obiettivo di ottenere una soluzione migliore in base all'input e all'output da considerare.
- Può essere una fase pesante, lenta, costosa, con varie difficoltà e con la necessità di fare tante prove.
- Ogni algoritmo scelto ha la sua procedura, basta seguirla come previsto.



MESSA IN ESERCIZIO

- In questo modulo **si verifica il modello scelto su dati nuovi, non usati precedentemente, per verificare se tutto va come previsto.**
- **Successivamente, la soluzione viene utilizzata nell'ambito del problema studiato.**
- Vengono usati vari termini, come: messa online, messa in linea, messa in esercizio, deploy.
- Ogni modello di calcolo ha le sue necessità. Per esempio, nel caso di una rete neurale bisogna identificare i parametri creati dall'ottimizzazione e le formule in cui inserirli per ottenere l'output.



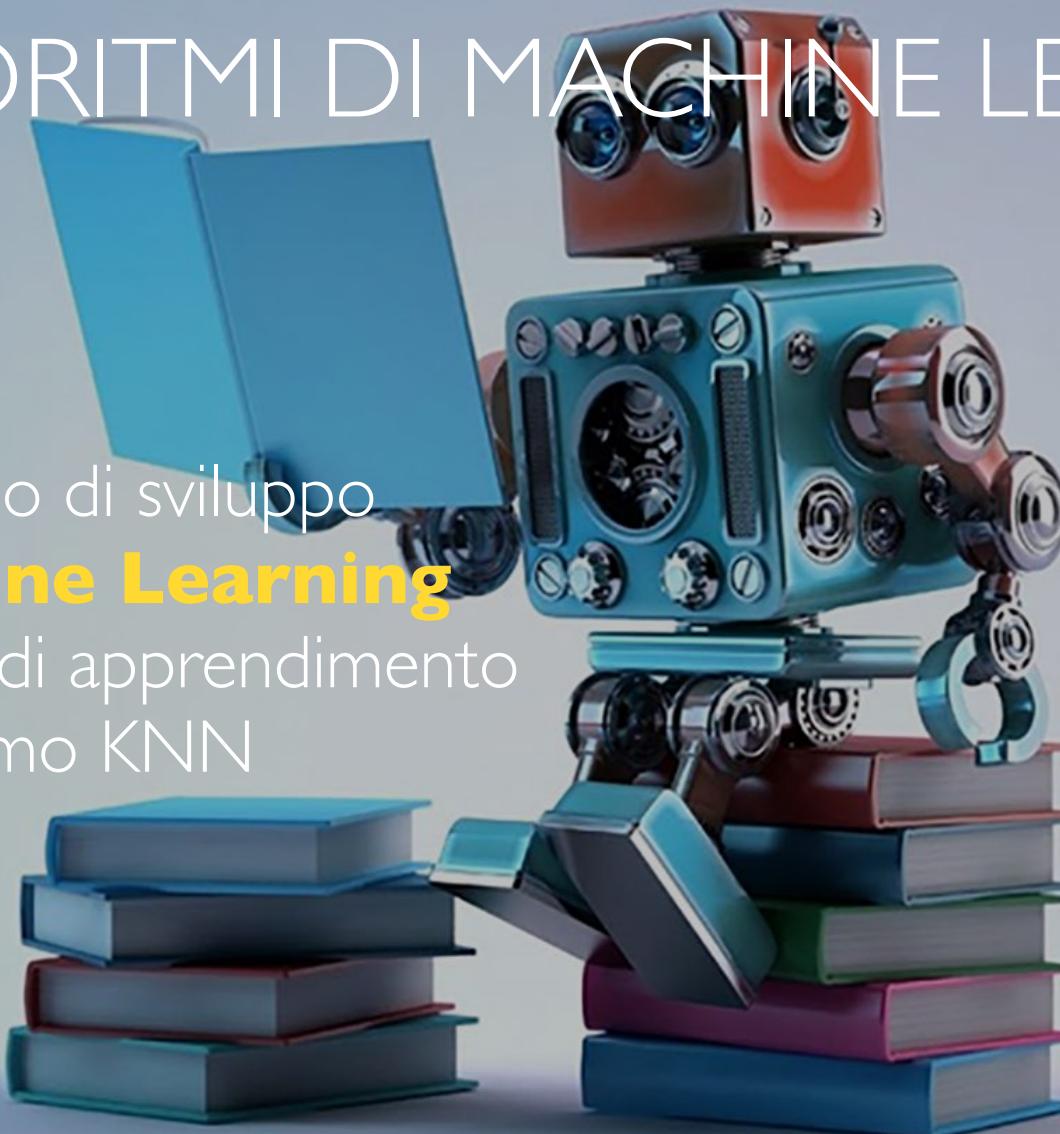
ERRORE IN ESERCIZIO

- **La soluzione ottenuta va provata nel mondo reale**, a stretto contatto con il problema da risolvere, **soprattutto con dati nuovi non considerati precedentemente, lo sviluppo non finisce qui.**
- Per la particolarità dell'AI e dell'ambiente in cui opera **è necessario il monitoraggio continuo per trovare errori, migliorare prestazioni, considerare nuovi dati e scenari.**
- Si parla, infatti, di **learning batch e continuous learning**, per indicare un sistema in grado di imparare continuamente.



ALGORITMI DI MACHINE LEARNING

- Percorso di sviluppo
- **Machine Learning**
- Forme di apprendimento
- Algoritmo KNN



Dott. Antonio Giovanni Lezzi



MACHINE LEARNING

- Il machine learning permette di creare un software che impara come fa un bambino, cioè tramite **'l'induzione di principi generali a partire dall'osservazione dei dati e con la capacità di ottenere nuova conoscenza a partire dall'informazione a disposizione.'**
- Un bambino nasce senza conoscere niente, però sa come studiare per apprendere, quindi va a scuola da un maestro, studia per imparare, si crea un'esperienza, svolge un esame, ottiene un voto per valutare che cosa ha imparato, se il voto è basso ripete l'anno scolastico.
- Nello stesso modo, **'una soluzione di machine learning può studiare, fare esperienza, scoprire le regolarità statistiche che si celano nei dati.'**



MACHINE LEARNING

- In tal modo, consegue l'importante **possibilità di generalizzare, cioè funzionare con la stessa efficacia anche con dati non considerati in precedenza.**
- Un'altra analogia con l'agricoltura può far comprendere il ruolo delle parti coinvolte. Gli algoritmi di apprendimento sono i semi, i dati sono il terreno, i modelli che apprendono sono le piante adulte.
- Lo sviluppatore di machine learning è come l'agricoltore: semina, irriga e concima il terreno, controlla lo stato di salute del raccolto, ma per il resto non interferisce.



MECCANISMO DI FUNZIONAMENTO

- Esistono varie definizioni di machine learning:
- Quelle meno formali indicano il machine learning come la scienza per far apprendere i dati al computer senza che sia stato esplicitamente programmato con delle regole.
- Un altro modo indica il machine learning come un mapping (corrispondenza) tra valori di input e output, che può essere modellato con una funzione non lineare.



MECCANISMO DI FUNZIONAMENTO

- Data la non linearità, conviene impiegare un approccio di machine learning basato su **funzioni $y(x; w)$** che contengono **parametri w** modificabili secondo **i dati x** .
- Il programmatore non deve specificare tutti i parametri w che caratterizzano il dato compito, perché questi vengono trovati tramite il modello.
- **Una definizione formale:** Dato un insieme di addestramento di N esempi definiti come coppie di dati input e output associato $(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$ dove y_i è stato generato da una funzione non nota $y=f(x)$, trovare una funzione h che approssimi la funzione sconosciuta f .



MECCANISMO DI FUNZIONAMENTO

- La funzione h è l'ipotesi, il machine learning cerca la funzione h che meglio spiega i dati in uscita e che meglio opera su dati nuovi non usati nell'addestramento.
- Si tratta, perciò, di apprendimento induttivo basato esclusivamente sull'osservazione dei dati esistenti.
- Il risultato del machine learning è nient'altro che un'equazione matematica complicata con molti parametri da ottimizzare.



ALGORITMI DI MACHINE LEARNING

- Percorso di sviluppo
- Machine Learning
- **Forme di apprendimento**
- Algoritmo KNN



FORME DI APPRENDIMENTO

- Le modalità con cui il machine learning **permette agli algoritmi di fare learning** (apprendimento, a volte indicato come addestramento) **con i dati sono classificate in cinque categorie, caratterizzate dal tipo di feedback** su cui si basa il sistema di apprendimento.
- Si tratta di un aspetto fondamentale nella progettazione, infatti viene subito specificato quando bisogna descrivere la soluzione a un problema.



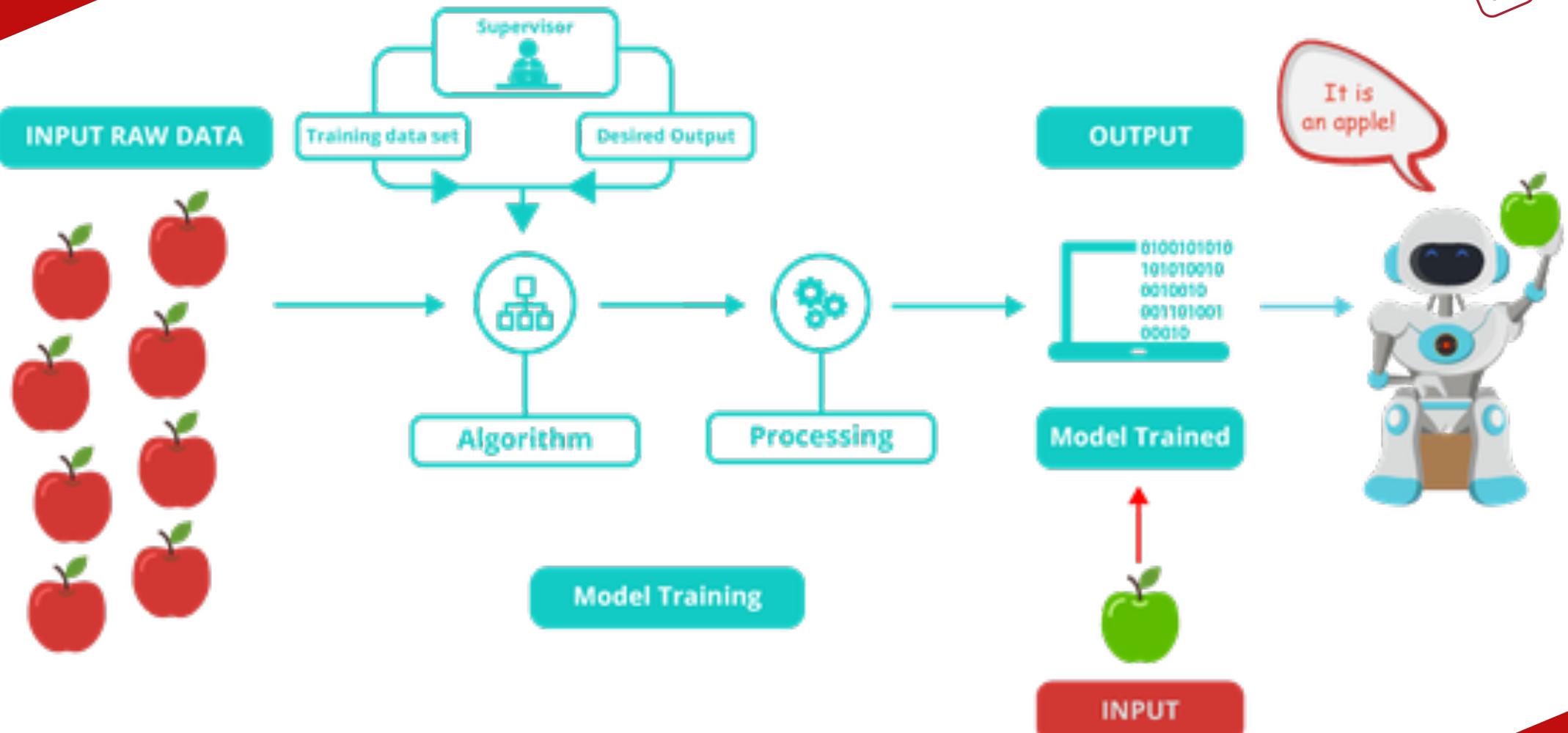
SUPERVISED LEARNING

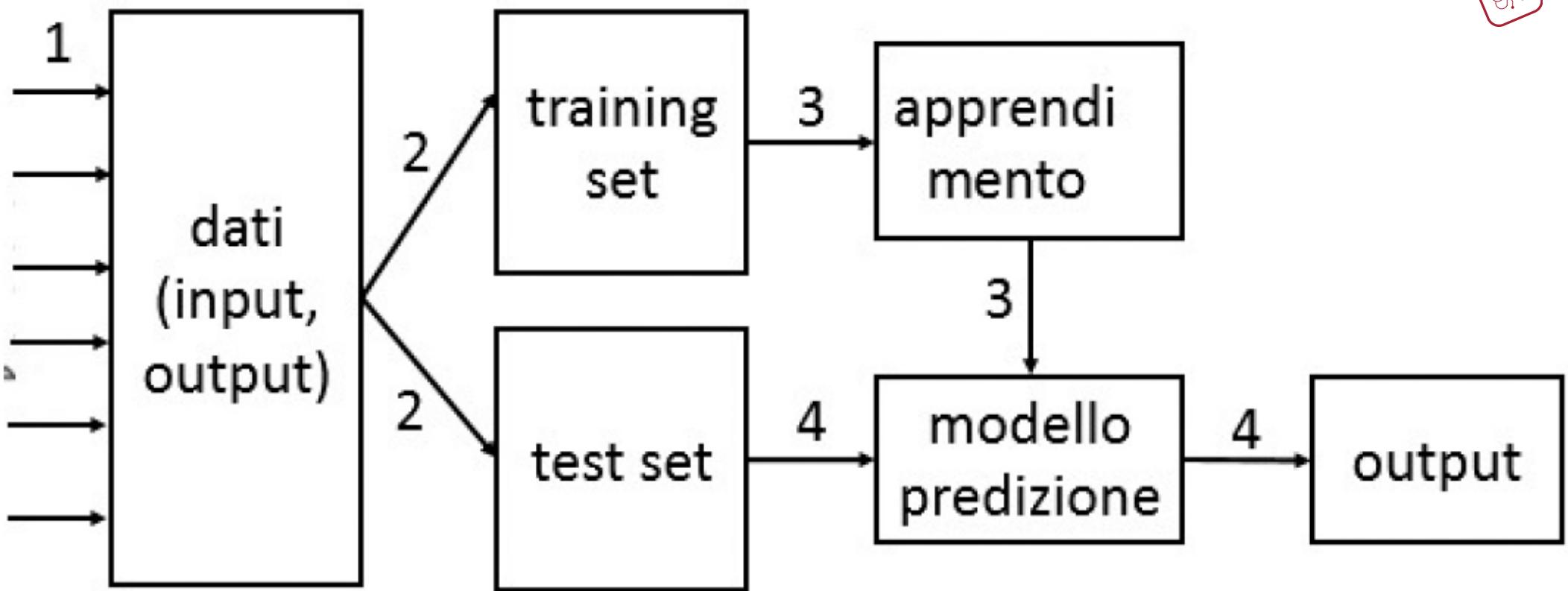
- Nel supervised learning (apprendimento supervisionato) **vengono presentati al modello scelto gli esempi formati dagli input e relativi output desiderati**
- per esempio un'immagine e un'etichetta (label) per descrivere l'oggetto contenuto, con lo scopo di apprendere una regola generale in grado di mappare gli input negli output, **come se ci fosse un maestro che supervisiona lo studente mentre impara dalle sue lezioni.**
- Le coppie di input e output vengono divise in due gruppi distinti: training set formato dall'80% dei dati con cui addestrare l'algoritmo, test set formato dal restante 20% per valutare la prestazione, come un esame finale in cui viene dato un voto su quanto si è bravi.



SUPERVISED LEARNING

- Bisogna stare attenti nell'inserire i dati in questi due insiemi: **tutte le casistiche possibili devono essere coperte in entrambi gli insiemi, non devono esserci sbilanciamenti con molti dati riguardanti una classe e pochi altri riguardanti le restanti classi.**
- Spesso l'assegnazione dei dati a questi due insiemi viene svolta con un'estrazione casuale, per non subire influenze indotte dall'esperto umano, più o meno consapevolmente.
- Permette di affrontare i problemi di classificazione e regressione.





- I numeri indicano il flusso delle informazioni e nel training si segue il flusso da 1 a 4, quando il modello ha terminato e viene messo in esercizio si seguono i numeri 1 2 4.



UNSUPERVISED LEARNING

- Con unsupervised learning (apprendimento non supervisionato, indicato anche come addestramento non supervisionato) **vengono forniti al modello scelto solo gli esempi formati dagli input, senza alcun output atteso, con lo scopo di fargli apprendere in autonomia una qualche struttura nei dati d'ingresso**
- Esempio: come se lo studente potesse studiare da solo senza nessun maestro che lo guidi.



UNSUPERVISED LEARNING

- **I risultati possono essere influenzati dalle decisioni su quali dati esporre all'algoritmo e in quale ordine.**
- Permette di gestire i problemi di clustering trovando i gruppi di dati in base alle caratteristiche simili.
- È utile anche per fare association learning, cioè trovare regole associative, come quelle in cui si determina che se una persona compra il prodotto A allora probabilmente preferisce comprare anche il prodotto B e si collegano i prodotti A e B.

TRAINING SET



Feature vectors



Machine learning
algorithm



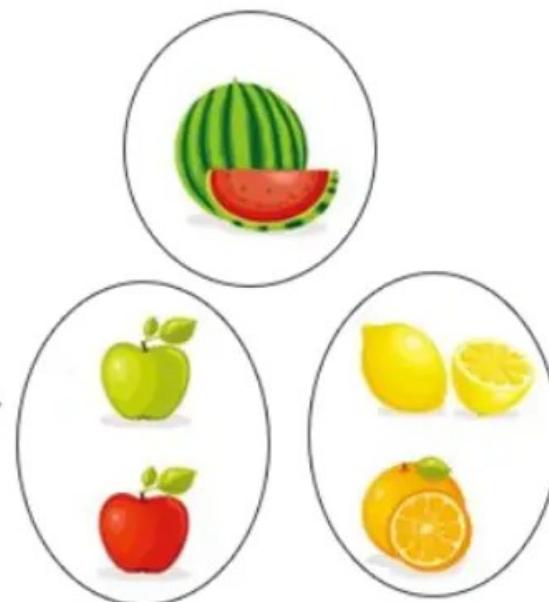
TEST SET



Predictive
Model



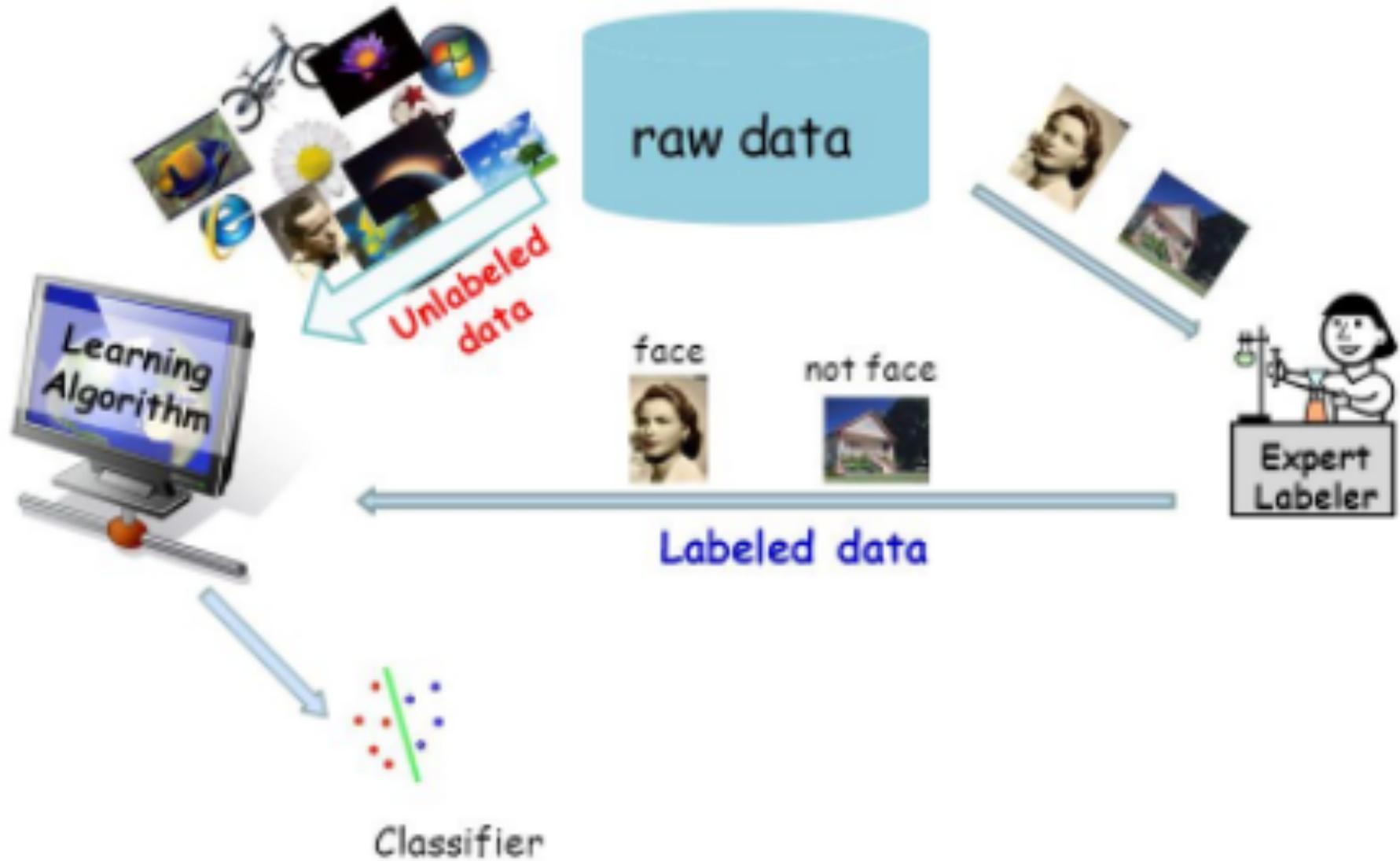
RESULTS





SEMI-SUPERVISED

- Si possono **combinare i due approcci precedenti** con una **prima fase supervised sui dati aventi input e output associato**, e una **successiva fase unsupervised su dati di cui non si conosce l'output associato**.
- Gli input e gli output forniti forniscono il modello generale che si può estrapolare e applicare ai dati rimanenti.





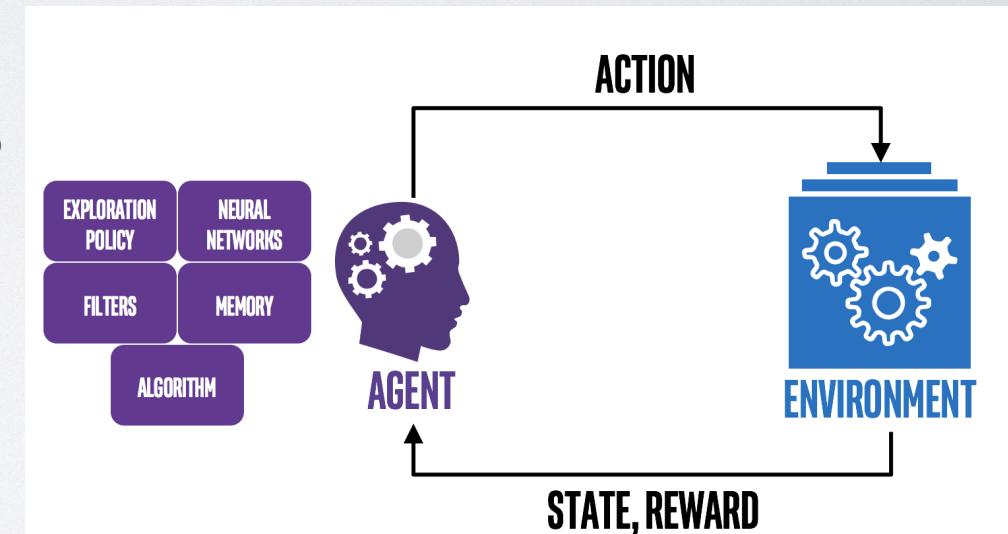
REINFORCEMENT

- Con il reinforcement learning (apprendimento con rinforzo) **si interagisce con un ambiente dinamico in cui raggiungere un certo obiettivo;**
- **mano a mano che si esplora il dominio del problema vengono forniti dei feedback in termini di ricompense o punizioni secondo il comportamento eseguito,** in modo da indirizzare verso la soluzione migliore.



REINFORCEMENT LEARNING

- Il reinforcement learning (apprendimento per rinforzo) **punta a realizzare agenti autonomi in grado di scegliere azioni da compiere per raggiungere obiettivi, tramite l'interazione con l'ambiente in cui sono immersi e una ricompensa che ha lo scopo di incoraggiare comportamenti corretti.**
- Viene usato in problemi di decisioni sequenziali, in cui l'azione da compiere dipende dallo stato attuale del sistema e ne determina quello futuro.
- Per esempio, un robot in cerca di funghi si muove nel bosco sconosciuto cercando di recuperare i funghi verdi (che danno una ricompensa positiva) e di evitare i funghi rossi (che danno una ricompensa negativa).

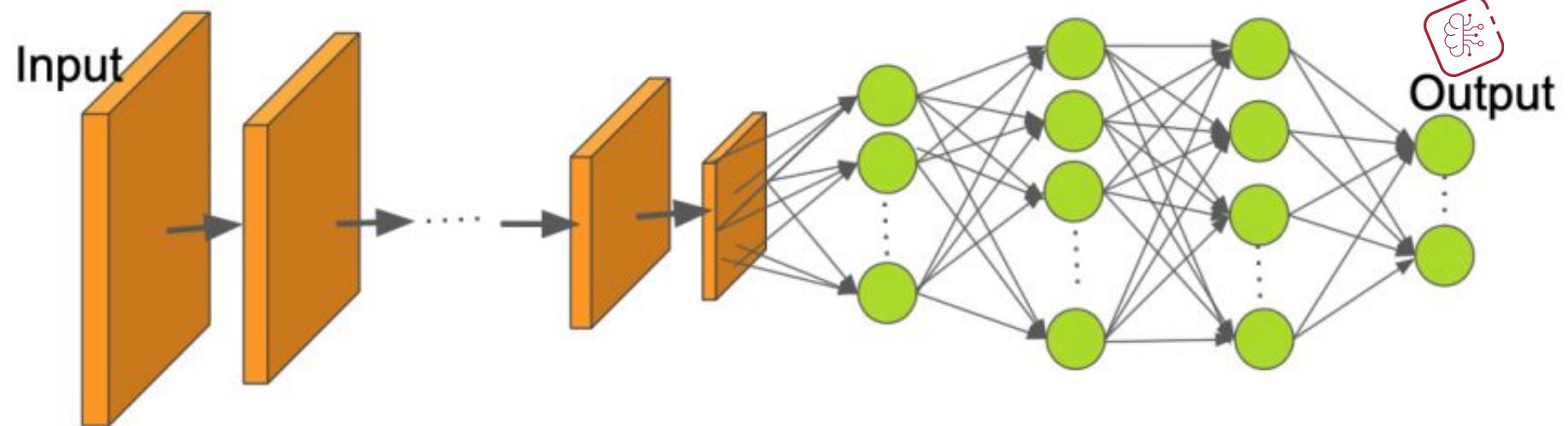




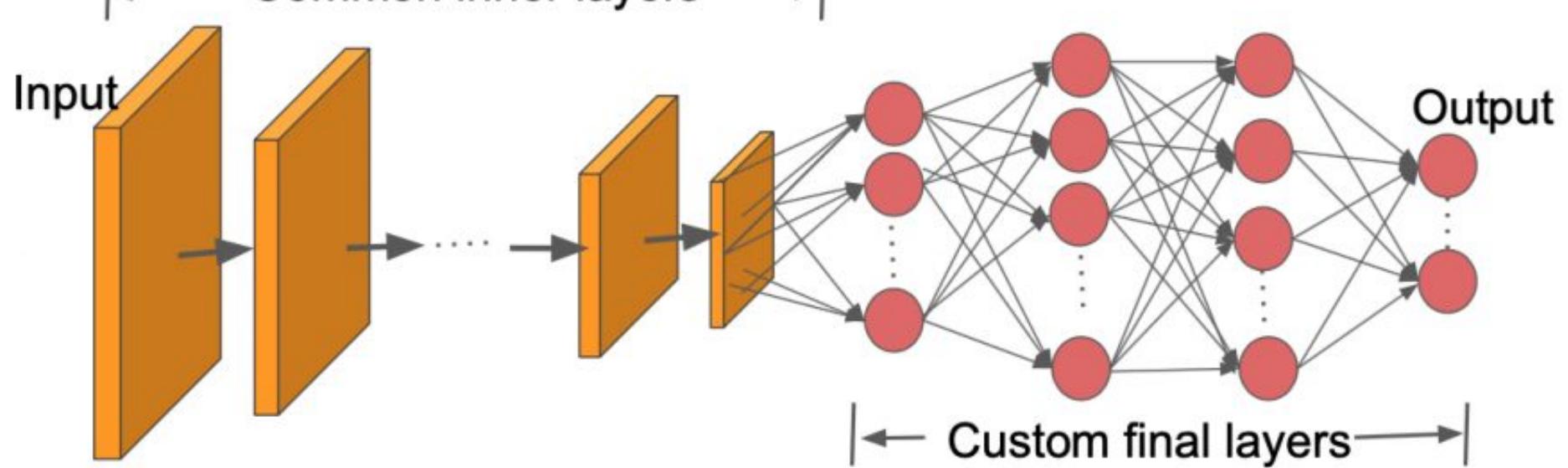
TRANSFER LEARNING

- Il transfer learning (apprendimento con trasferimento) **impara ad affrontare un certo problema generico, poi si prende la conoscenza creata per usarla nell'affrontare un altro problema simile o più specifico.**
- Il grosso vantaggio consiste nel non rifare di nuovo il learning, nel risparmiare tempo, nell'avere una libreria di soluzioni pronte per essere adattate all'esigenza.

Pretrained
Model



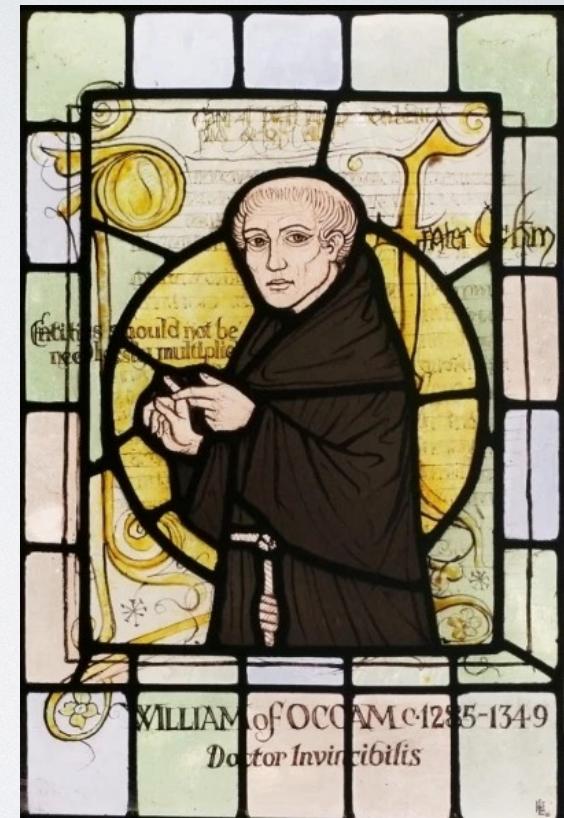
Custom
Model

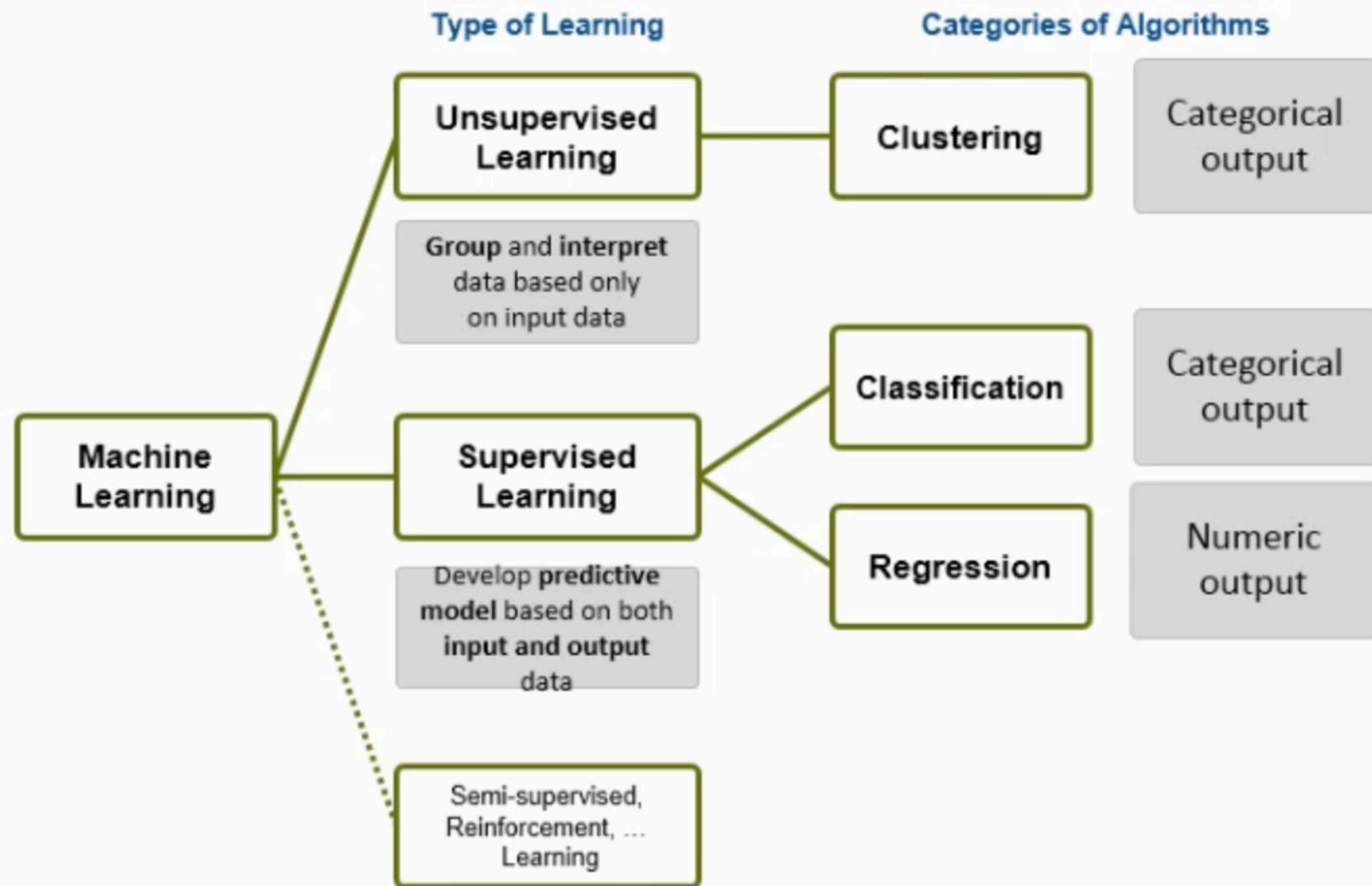




RASOIO DI OCKHAM

- Durante tutto il processo di costruzione della soluzione bisogna prendere varie decisioni, in particolare durante la formulazione del problema, quando si fanno esperimenti per ottimizzare i parametri, quando si analizzano i dati per trovare cosa è più rilevante.
- In questi contesti, una linea guida per cercare di avere la soluzione migliore consiste nell'adottare il cosiddetto rasoio di Ockham, secondo cui, **a parità di fattori, bisogna preferire la spiegazione più semplice, non vi è motivo alcuno per complicare ciò che è semplice.**
- Viene chiamato anche “principio di economia o di parsimonia”, perché suggerisce di fare a meno delle ipotesi superflue quando si cerca di spiegare un fenomeno derivante da un esperimento.





ALGORITMI DI MACHINE LEARNING

- Percorso di sviluppo
- Machine Learning
- Forme di apprendimento
- **Algoritmo KNN**



ALGORITMO KNN

- Uno degli algoritmi più conosciuti nel machine learning è il K-Nearest Neighbors (KNN) che, oltre alla sua **semplicità, produce buoni risultati in un gran numero di domini.**
- ad esempio, per decidere se effettuare la concessione del prestito da parte di istituti creditizi presso clienti, oppure, per prevedere il rating di credito di un nuovo cliente, senza dover eseguire tutti i calcoli; oppure nelle scienze politiche per classificare se un potenziale elettore “voterà” o “non voterà” durante le prossime elezioni.
- Altri esempi avanzati includono il rilevamento della grafia (**come l'OCR**), il **riconoscimento dell'immagine** e persino il riconoscimento video.



ALGORITMO KNN

- **KNN è un algoritmo di apprendimento supervisionato**, il cui scopo è quello di predire una nuova istanza conoscendo i data points che sono separati in diverse classi.
- KNN **potrebbe essere una delle prime scelte per uno studio di classificazione quando c'è poca o nessuna conoscenza precedente sulla distribuzione dei dati.**
- Per capire meglio l'algoritmo, ipotizziamo di dover predire a quale classe tra frutta, verdura e grano appartiene la patata dolce





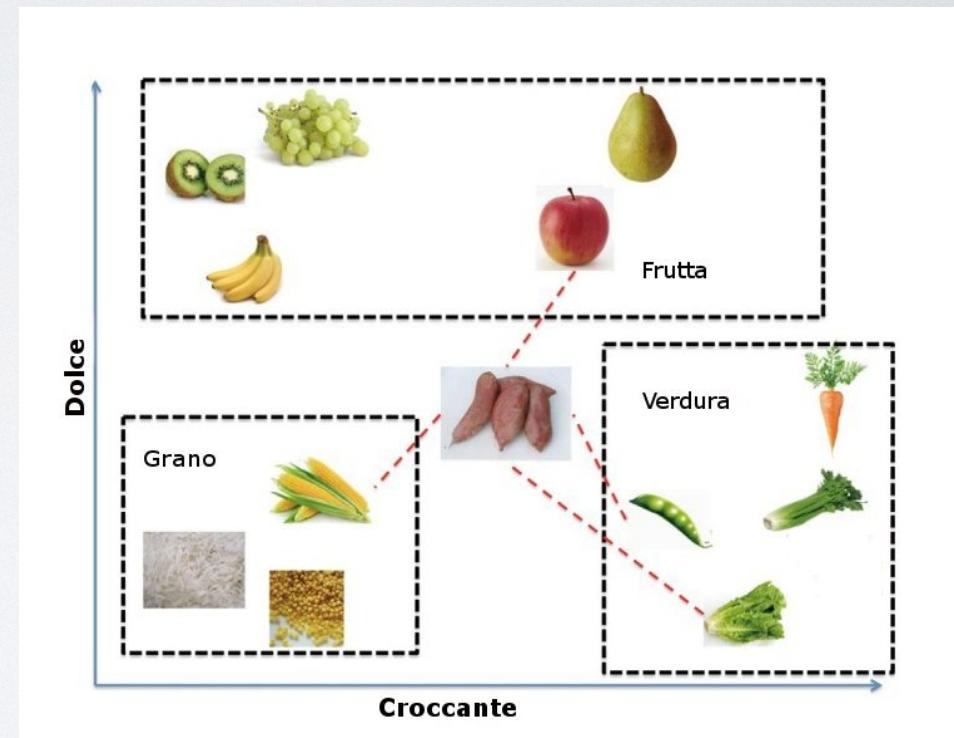
ALGORITMO KNN

- Dentro ad ogni classe vengono associati dei data points o istanze, il cui insieme definisce il set di dati.
- Ad esempio possiamo avere:
 - mela, uva, kiwi, banana, pera dentro la classe frutta;
 - lattuga, carote, sedano, fagiolo verde dentro la classe verdura;
 - mais e farina nella classe grano.



ALGORITMO KNN

- Sappiamo per esperienza che, in generale, i frutti sono più dolci delle verdure, mentre i cereali non sono né croccanti né dolci.
- Possiamo supporre di visualizzare suddette classi su un asse cartesiano dove l'asse delle ordinate rappresenta quanto è dolce un frutto/ortaggio mentre l'asse delle ascisse rappresenta quanto tale frutto/ortaggio è croccante.
- Dolce e croccante rappresentano le caratteristiche del problema (se ne potrebbero usare anche di più di due).





ALGORITMO KNN

- In questo esempio scegliamo quattro tipi di cibo più vicini: mela, fagiolo verde, lattuga e mais. Poiché la classe verdura ha più “voti” (2 contro 1 per la classe grano e 1 per la classe frutta), e poiché vince la classe che ottiene il maggior numero di voti, la patata dolce viene assegnata alla classe verdure.
- Questo semplice esempio mostra la logica di ragionamento che sta dietro l'algoritmo K-Nearest Neighbors.
- **Il suo funzionamento si basa sulla somiglianza delle caratteristiche: più un'istanza è vicina a un data point, più il knn li considererà simili.**



ALGORITMO KNN

- Solitamente la **somiglianza viene calcolata tramite la distanza euclidea** (o un qualche altro tipo di distanza a seconda del problema in esame).
- Minore sarà la distanza e maggiore sarà la somiglianza tra data point e l'istanza da prevedere.
- Oltre alla distanza, **l'algoritmo prevede di fissare un parametro k**, scelto arbitrariamente, **che identifica il numero di data points più vicini**.
- L'algoritmo valuta le **k minime distanze** così ottenute.
- **La classe che ottiene il maggior numero di queste distanze è scelta come previsione.**



COME FUNZIONA IL KNN?

- Il funzionamento dell'algoritmo K-Nearest Neighbors può essere definito tramite i seguenti step:
 1. Scegli un valore k con cui prevedere il nuovo data point;
 2. Nel caso di grandezze non comparabili utilizza tecniche per rendere le misure confrontabili, come la normalizzazione. Altrimenti, nel caso di misure comparabili (metri e metri, Kg e kg, ecc.) passa allo step 3;
 3. Calcola la distanza (ad esempio quella euclidea) tra la nuova istanza e i vari data points;
 4. Ordina le distanze calcolate dalla più piccola alla più grande;
 5. Scegli le prime K: nel caso si stia svolgendo un problema di regressione, si può restituire la media delle etichette K. Se si sta svolgendo un problema di classificazione, si sceglierà la classe che include più valori k trovati precedentemente.



ESEMPIO DI KNN

- Supponiamo di avere l'altezza, il peso e la taglia della maglietta di alcuni clienti e dobbiamo prevedere la taglia della t-shirt di un nuovo cliente.
- In figura sono riportati i dati di altezza, peso e dimensioni della maglietta.

| | A | B | C |
|----|------------------|---------------|----------------|
| 1 | Altezza (in cms) | Peso (in kgs) | T Shirt Taglia |
| 2 | 158 | 58 | M |
| 3 | 158 | 59 | M |
| 4 | 158 | 63 | M |
| 5 | 160 | 59 | M |
| 6 | 160 | 60 | M |
| 7 | 163 | 60 | M |
| 8 | 163 | 61 | M |
| 9 | 160 | 64 | L |
| 10 | 163 | 64 | L |
| 11 | 165 | 61 | L |
| 12 | 165 | 62 | L |
| 13 | 165 | 65 | L |
| 14 | 168 | 62 | L |
| 15 | 168 | 63 | L |
| 16 | 168 | 66 | L |
| 17 | 170 | 63 | L |
| 18 | 170 | 64 | L |
| 19 | 170 | 68 | L |



COME FUNZIONA IL KNN?

- Ipotizziamo:
 - Di voler prevedere la taglia di una nostra cliente Martina, sapendo che essa pesa 58 kg ed è alta 160 cm;
 - Impostiamo $k = 5$.
- Quindi l'algoritmo cerca i 5 clienti più vicini a Martina (nostra previsione). Se 4 di loro avevano “T-shirt medie” e 1 aveva “T-shirt L”, allora l'ipotesi migliore per il cliente è “T-shirt media”.



COME FUNZIONA IL KNN?

- Avendo in questo caso due unità di misura differenti (peso in kg, e altezza in cm) occorre rappresentarle secondo un valore comune per evitare che una grandezza sia più rilevante dell'altra e quindi generi risultati ambigui.
- Per fare questo e cioè per renderle comparabili è necessario ridimensionarle attraverso la seguente formula:
$$X_s = \frac{X - \mu}{\sigma}$$



COME FUNZIONA IL KNN?

- Con σ deviazione standard e calcolabile nel seguente modo:
- $$\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{N - 1}}$$
- Quindi, ad esempio il primo valore può essere così trovato:
- Altezza in cm (cella H2) = $(158 - 164) / 4,32 = - 1,39$



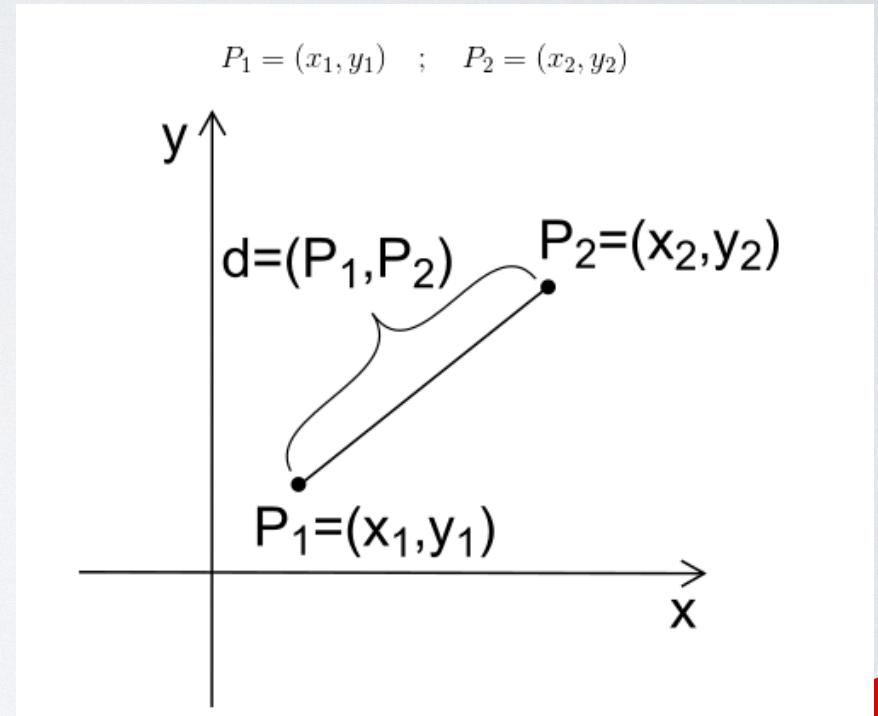
| H | I |
|-------------|----------|
| Altezza st. | Peso st. |
| -1,39 | -1,64 |
| -1,39 | -1,27 |
| -1,39 | 0,25 |
| -0,92 | -1,27 |
| -0,92 | -0,89 |
| -0,23 | -0,89 |
| -0,23 | -0,51 |
| -0,92 | 0,63 |
| -0,23 | 0,63 |
| 0,23 | -0,51 |
| 0,23 | -0,13 |
| 0,23 | 1,01 |
| 0,92 | -0,13 |
| 0,92 | 0,25 |
| 0,92 | 1,39 |
| 1,39 | 0,25 |
| 1,39 | 0,63 |
| 1,39 | 2,15 |
| | |
| -0,92 | -1,64 |

Dott. Antonio Giovanni Lezzi



ESEMPIO DI KNN

- Si può procedere al calcolo della **distanza euclidea**, che ci permette di calcolare quali sono i punti più prossimi all'istanza da prevedere.
- $d(P_1; P_2) = \sqrt{(x_2 - X_1)^2 + (y_2 - y_1)^2}$





| H | I | J | K | L |
|-------------|----------|----------------|-------------------|---|
| Altezza st. | Peso st. | T Shirt Taglia | Distanza Euclidea | |
| -1,39 | -1,64 | M | 0,46 | 2 |
| -1,39 | -1,27 | M | 0,60 | 3 |
| -1,39 | 0,25 | M | 1,95 | |
| -0,92 | -1,27 | M | 0,38 | 1 |
| -0,92 | -0,89 | M | 0,76 | 4 |
| -0,23 | -0,89 | M | 1,03 | 5 |
| -0,23 | -0,51 | M | 1,33 | |
| -0,92 | 0,63 | L | 2,28 | |
| -0,23 | 0,63 | L | 2,38 | |
| 0,23 | -0,51 | L | 1,62 | |
| 0,23 | -0,13 | L | 1,91 | |
| 0,23 | 1,01 | L | 2,90 | |
| 0,92 | -0,13 | L | 2,39 | |
| 0,92 | 0,25 | L | 2,65 | |
| 0,92 | 1,39 | L | 3,56 | |
| 1,39 | 0,25 | L | 2,99 | |
| 1,39 | 0,63 | L | 3,25 | |
| 1,39 | 2,15 | L | 4,44 | |
| | | | | |
| -0,92 | | -1,64 | | |

Quindi a Martina, secondo le previsioni del modello e dei 5-k valori più vicini avrà bisogno di una taglia M

Dott. Antonio Giovanni Lezzi



COME SCEGLIAMO IL FATTORE K?

- Nell'esempio proposto abbiamo utilizzato il valore 5 per k.
- In questo caso, ogni nuovo data point è previsto dai 5 suoi vicini più prossimi nel set di allenamento.

PYTHON

```
from math import sqrt

# calculate the Euclidean distance between two vectors
def euclidean_distance(row1, row2):
    distance = 0.0
    for i in range(len(row1)-1):
        distance += (row1[i] - row2[i])**2
    return sqrt(distance)

# Locate the most similar neighbors
def get_neighbors(train, test_row, num_neighbors):
    distances = []
    for train_row in train:
        dist = euclidean_distance(test_row, train_row)
        distances.append((train_row, dist))
    distances.sort(key=lambda tup: tup[1])
    neighbors = []
    for i in range(num_neighbors):
        neighbors.append(distances[i][0])
    return neighbors
```

PYTHON

```
# Test distance function
dataset = [[2.7810836,2.550537003,0],
           [1.465489372,2.362125076,0],
           [3.396561688,4.400293529,0],
           [1.38807019,1.850220317,0],
           [3.06407232,3.005305973,0],
           [7.627531214,2.759262235,1],
           [5.332441248,2.088626775,1],
           [6.922596716,1.77106367,1],
           [8.675418651,-0.242068655,1],
           [7.673756466,3.508563011,1]]
neighbors = get_neighbors(dataset, dataset[0], 3)
for neighbor in neighbors:
    print(neighbor)
```



ESERCIZI

1. In base ai dati forniti in excel e il codice scritto in Python, verificare di ottenere gli stessi risultati
2. Adattare il codice per leggere il file Iris.csv e indicare che tipo di fiore corrisponde ai seguenti dati: [SepalLen=5.0, SepalWidth=3.6, PetalLen=1.7, PetalWidth=0.4]

(SOLUZIONE)

```
import pandas as pd  
dataset = pd.read_csv('./Iris.csv')
```