



**UNIVERSITÀ
DEGLI STUDI
DI TRIESTE**

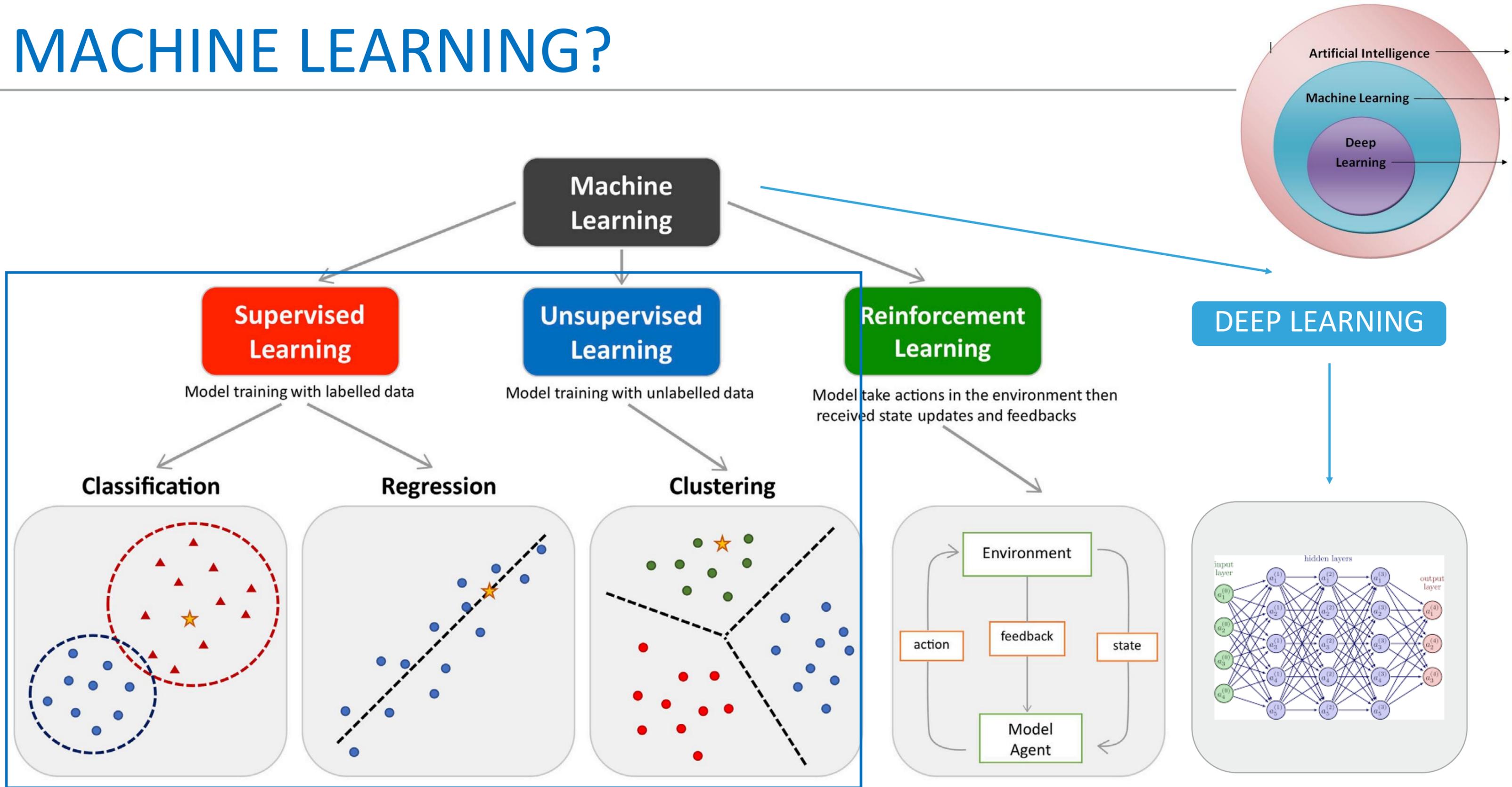
MODULO 3: Machine learning (II) e accenni di teoria della complessità

Prof.ssa Giulia Cisotto

giulia.cisotto@units.it

Trieste, 4 giugno 2025

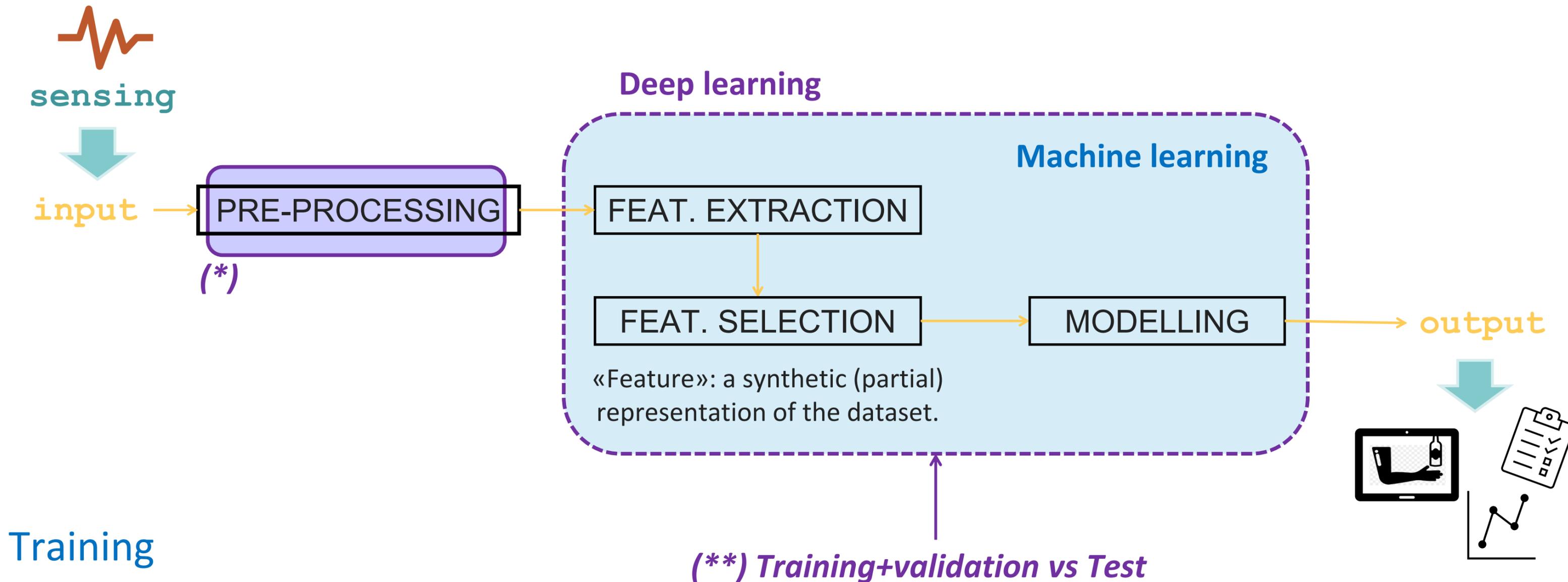
MACHINE LEARNING?



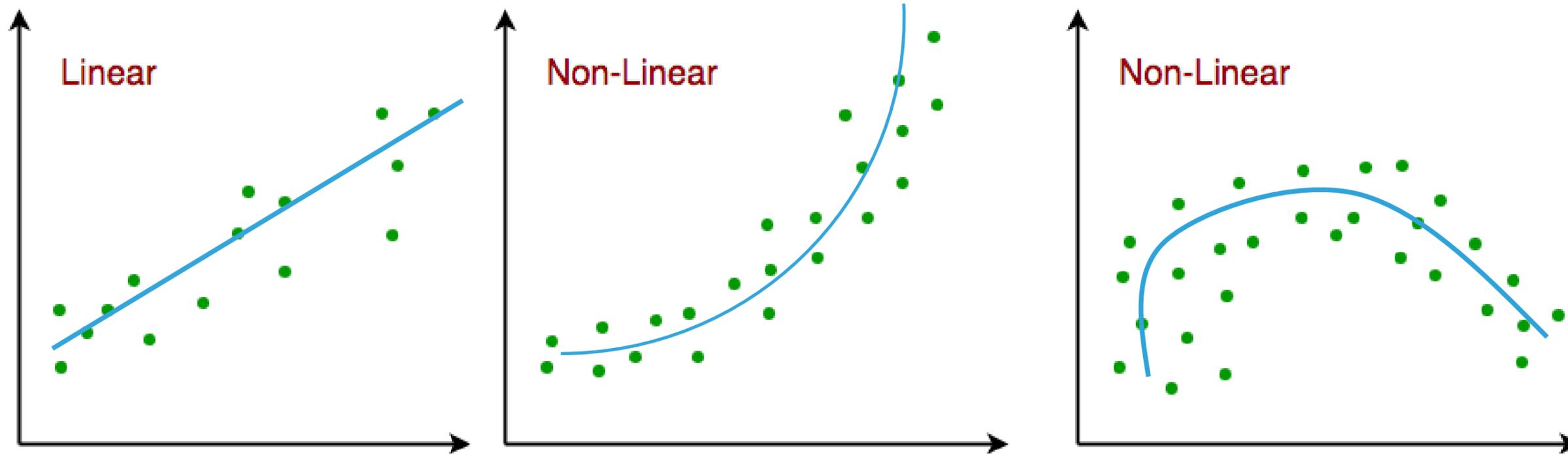
Peng, J., Jury, E. C., Dönnies, P., & Ciurtin, C. (2021). *Frontiers in pharmacology*, 12, 720694.

Questo tipo di AI si chiama anche «**discriminativa**» (in contrapposizione con LLM che sono tipicamente usati come **AI generativa**)

PIPELINE GENERICA DI ELABORAZIONE DEI DATI

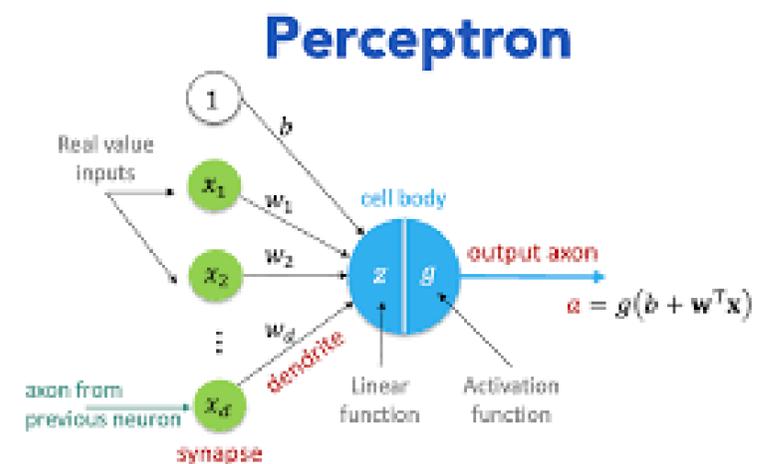
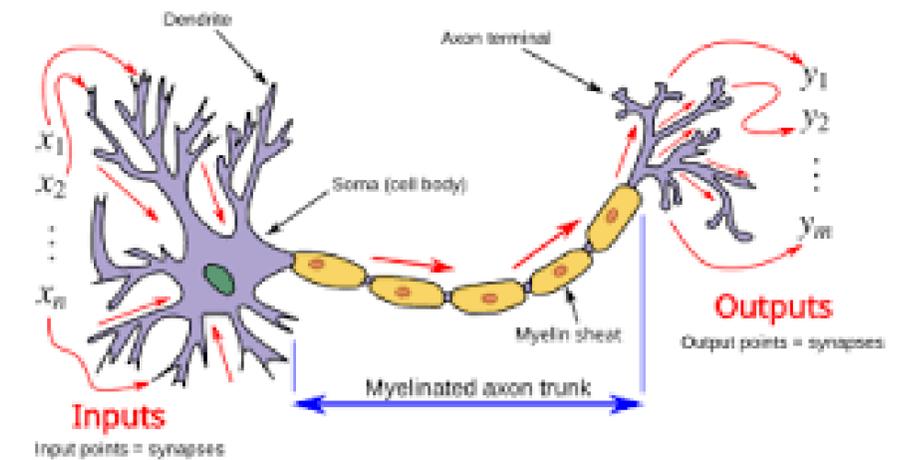
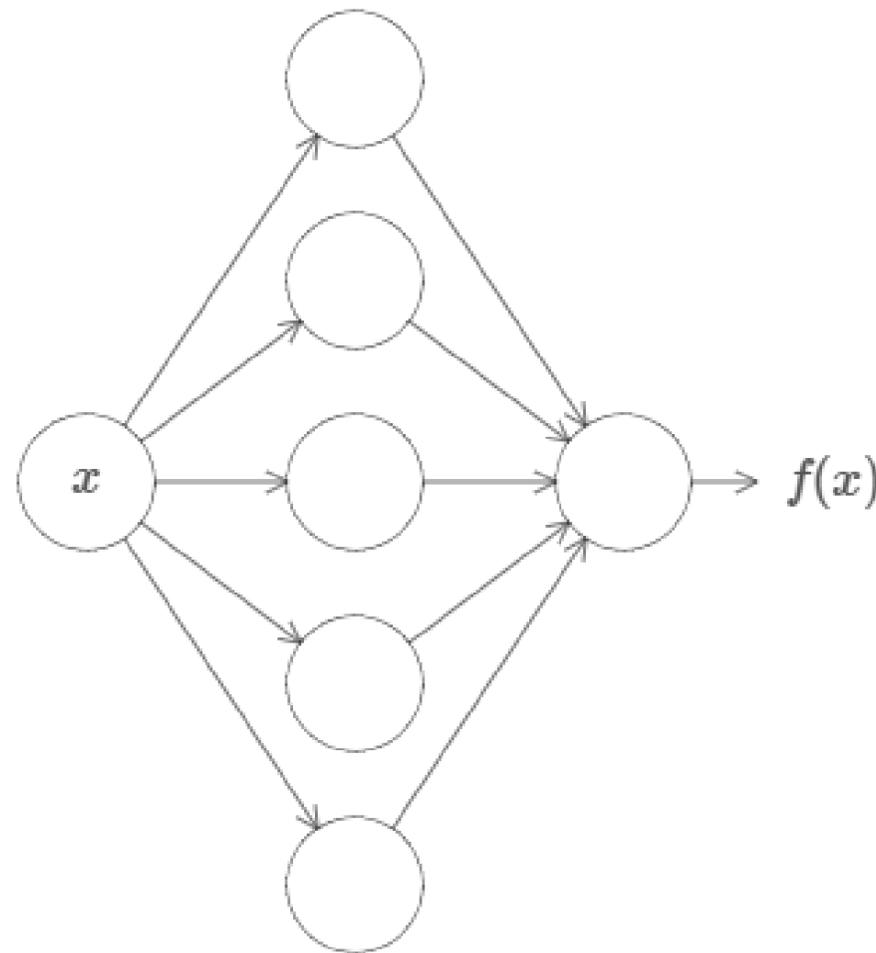
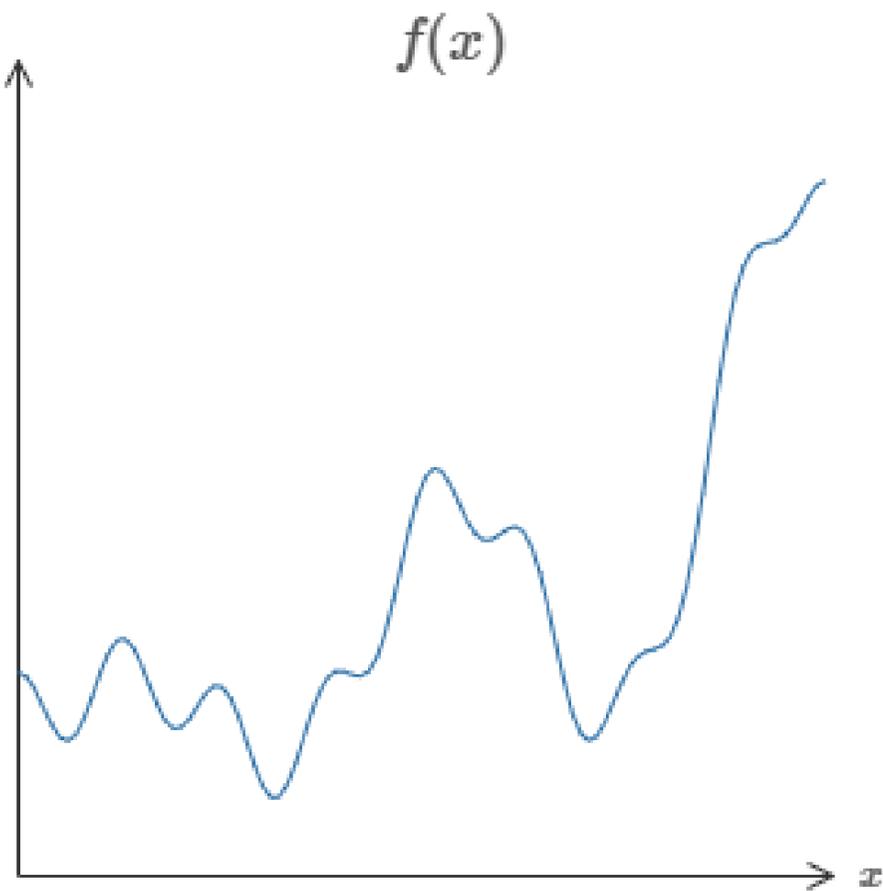


MODELLO SUI DATI



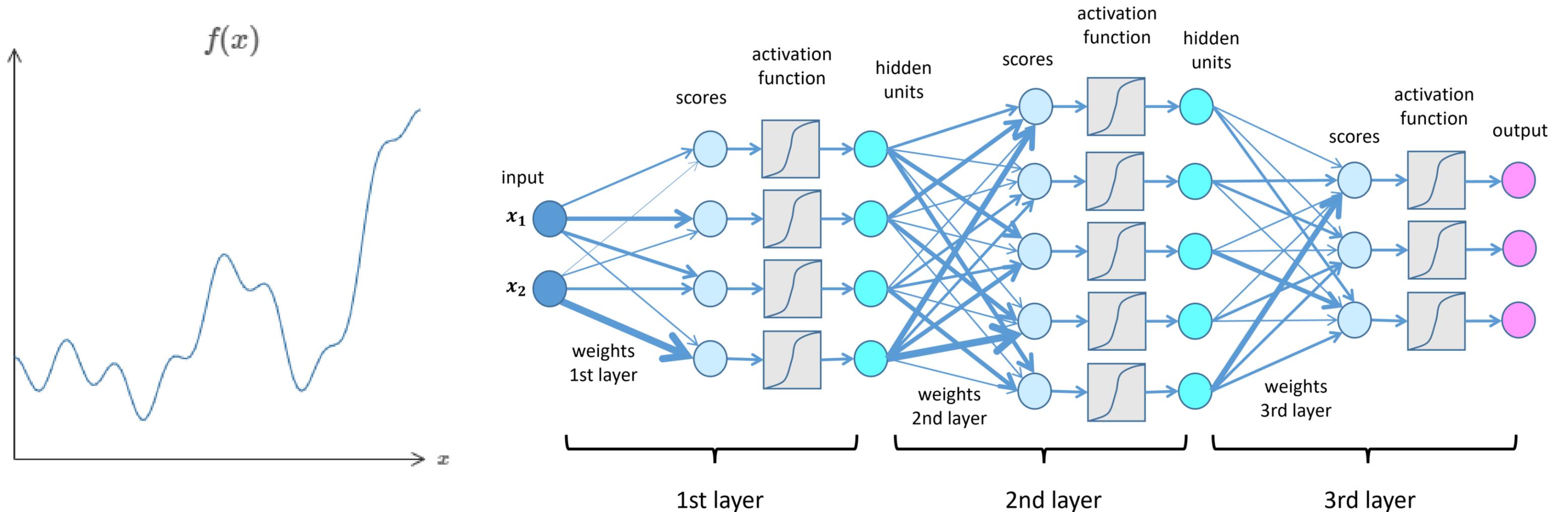
APPROSSIMATORE UNIVERSALE

No matter what the function, there is guaranteed to be a neural network so that for every possible input x , the value $f(x)$ (or some close approximation) is output from the network.



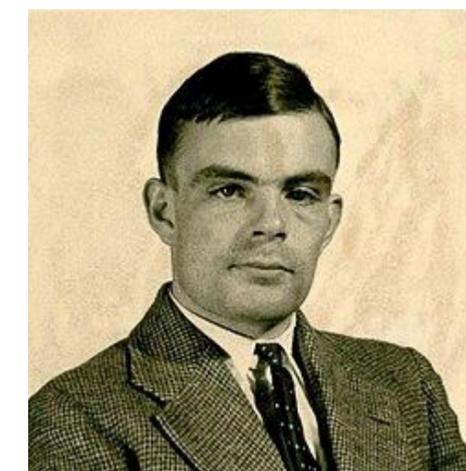
APPROSSIMATORE UNIVERSALE

No matter what the function, there is guaranteed to be a neural network so that for every possible input x , the value $f(x)$ (or some close approximation) is output from the network.

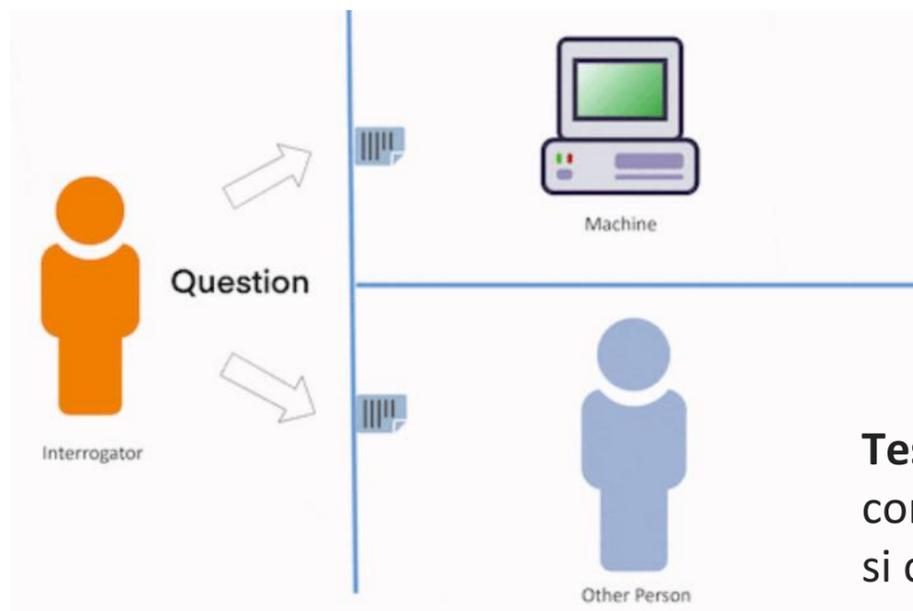


📖 Origini e primi sviluppi (1930–1956)

- **1936** – *Alan Turing* introduce il concetto di macchina universale (una macchina immaginaria che fosse capace di eseguire ogni tipo di calcolo su numeri e simboli).
- **1950** – Turing propone il *Test di Turing* per valutare l'intelligenza di una macchina.
- **1956** – Conferenza di **Dartmouth**: nasce ufficialmente il termine "Artificial Intelligence" (McCarthy, Minsky, Simon, ecc.).



Alan Mathison Turing (1912-1954), matematico, logico, crittografo, filosofo britannico. Considerato padre dell'informatica.



Test di Turing. Se l'interrogatore attribuisce al computer almeno il 30% delle risposte il test di Turing si considera superato dalla macchina.

1956 Dartmouth Conference: The Founding Fathers of AI



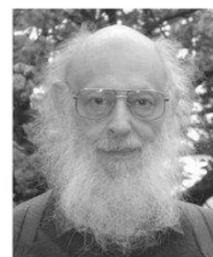
John McCarthy



Marvin Minsky



Claude Shannon



Ray Solomonoff



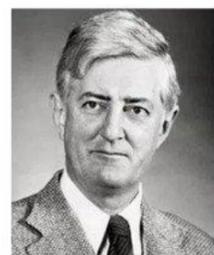
Alan Newell



Herbert Simon



Arthur Samuel



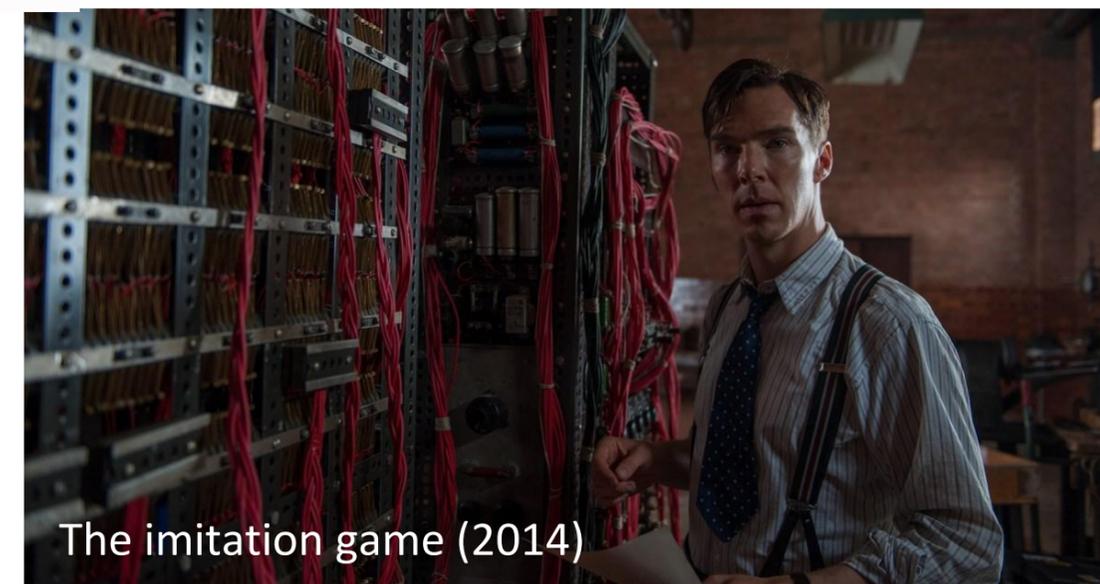
Oliver Selfridge



Nathaniel Rochester



Trenchard More



The imitation game (2014)

☀️ Prima estate dell'AI (1956–1974)

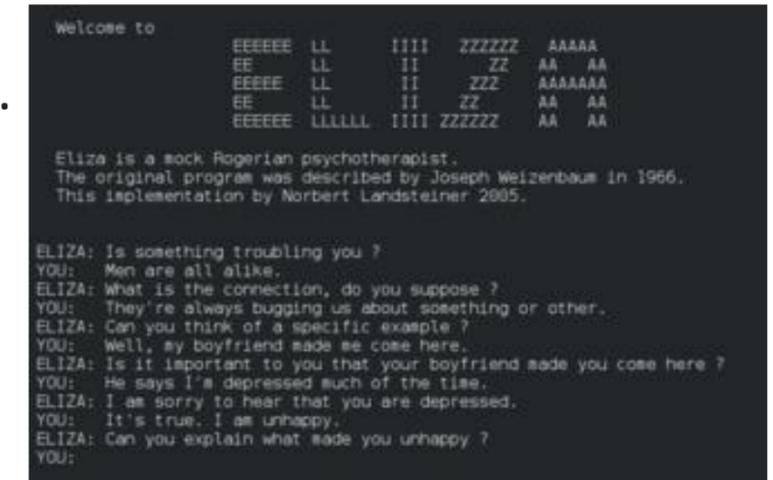
- Ottimismo: i pionieri credono che l'intelligenza umana possa essere replicata entro pochi decenni.
- **Sistemi simbolici**: logica formale, problem solving, giochi (es. scacchi), primi chatbot (ELIZA).
- **Finanziamenti generosi, soprattutto dalla Defense Advanced Research Projects Agency (DARPA).**

ELIZA procedeva analizzando e sostituendo semplici parole chiave in frasi preconfezionate. A seconda delle parole che l'utente immetteva nel programma, l'illusione di un interlocutore umano veniva smascherata o poteva continuare per diverse battute.

ELIZA rimane una pietra miliare semplicemente perché fu la **prima** volta che un programmatore sviluppò un'**interazione uomo-macchina** con l'obiettivo di creare l'illusione (seppur breve) di un dialogo uomo-uomo.

❄️ Primo inverno dell'AI (1974–1980)

- **Cause**:
 - Fallimenti pratici: sistemi fragili e inadatti a problemi reali.
 - Limiti hardware e mancanza di dati.
 - Critiche (es. *Lighthill Report* nel Regno Unito).
 - **Taglio ai finanziamenti.**



[Chi era Joseph Weizenbaum, l'inventore del primo chatbot](#)



[Lighthill, James \(1973\). "Artificial Intelligence: A General Survey". Artificial Intelligence: A paper symposium. UK: Science Research Council.](#)

🌞 Seconda estate: era degli esperti (1980–1987)

- Nascita dei **sistemi esperti** (es. MYCIN): regole + basi di conoscenza.
- Applicazioni industriali e mediche.
- **Investimenti crescono** (Giappone, USA, Europa).

❄️ Secondo inverno dell'AI (1987–1993)

- **Cause:**
 - Sistemi rigidi, non scalabili.
 - **Elevato costo di manutenzione.**
 - **Manca di apprendimento automatico.**
 - Crollo del mercato dei Lisp machines e fallimento delle “AI startup”.

Lisp (List Processor) è una famiglia di linguaggi di programmazione con implementazioni sia compilate sia interpretate, associata nel passato ai progetti di intelligenza artificiale. È stato ideato nel 1958 da John McCarthy come linguaggio formale per studiare la computabilità di funzioni ricorsive su espressioni simboliche.

🌞 Rinascita e boom del machine learning (1993–2011)

- Ritorno del **neural network** (backpropagation).
- Crescita del **machine learning** statistico.
- Algoritmi più robusti (SVM, ensemble methods).
- Data mining e Internet generano più dati.
- **Vincita di Deep Blue su Kasparov (1997) = evento simbolico.**

🌻 Deep learning revolution (2012–2022)

- **2012 – AlexNet domina ImageNet***: svolta epocale.
- Espansione di deep learning in visione, NLP, robotica.
- Nascono framework open-source (TensorFlow, PyTorch).
- Grandi aziende investono: Google, Meta, OpenAI, ecc.
- **Introduzione dei Transformer (2017)**
- **GPT-2 (2019), GPT-3 (2020), ChatGPT (2022).**

AlexNet ha vinto la competizione ImageNet con un **margin** enorme rispetto agli altri, **riducendo l'errore top-5 dal 26% al 15%**, grazie all'uso efficace di **deep learning** (reti neurali profonde), **GPU** per l'addestramento e tecniche come **ReLU**, **dropout** e **data augmentation**.

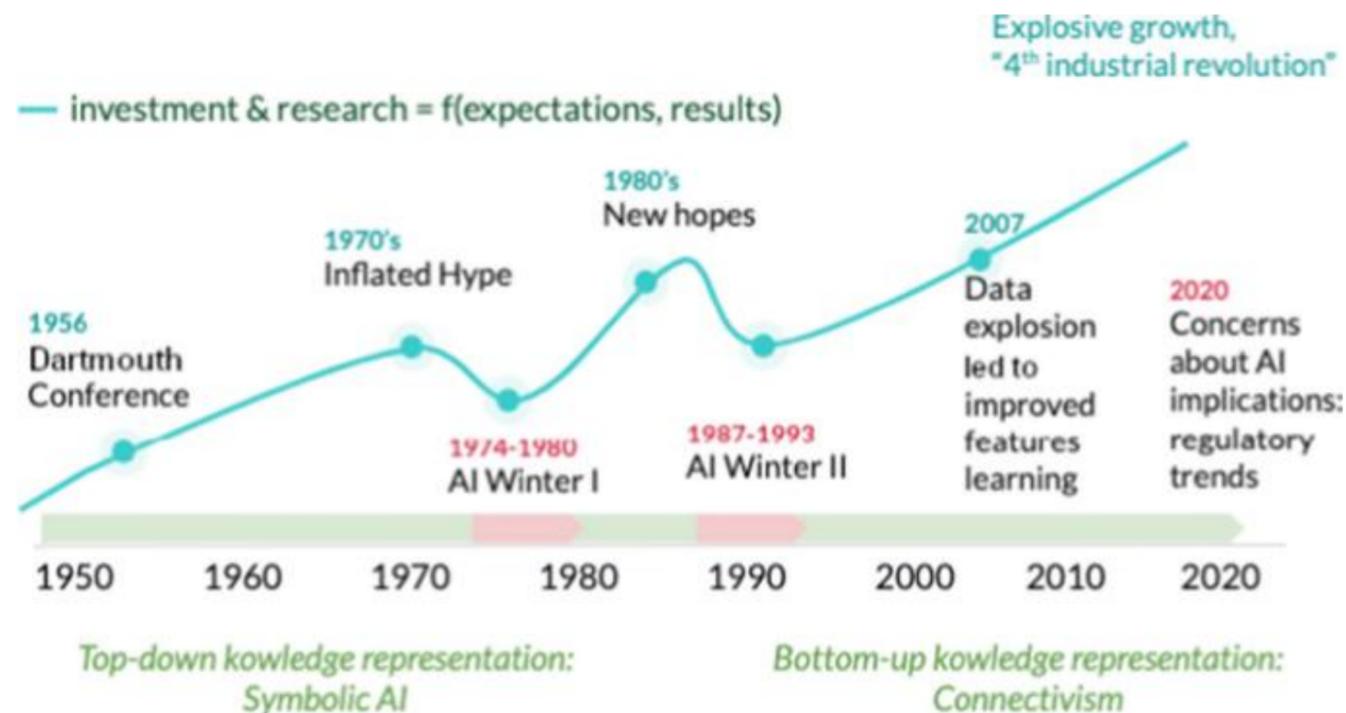
Ha **dimostrato per la prima volta** che le reti profonde, se ben progettate e allenate con molti dati, possono **superare di gran lunga** i metodi classici in compiti complessi di visione artificiale.

*ImageNet = enorme database di immagini organizzato gerarchicamente



🌞 Deep learning revolution (2012–2022)

- **2012** – AlexNet domina ImageNet*: svolta epocale.
- Espansione di deep learning in visione, NLP, robotica.
- Nascono framework open-source (TensorFlow, PyTorch).
- Grandi aziende investono: Google, Meta, OpenAI, ecc.
- **Introduzione dei Transformer (2017)**
- **GPT-2 (2019), GPT-3 (2020), ChatGPT (2022).**



❄️ Mini-inverno etico e regolativo (2023–2024)

• Cause:

- Preoccupazioni su bias, trasparenza, copyright e impatto occupazionale.
- Diffidenza pubblica e necessità di regolamentazione (es. **AI Act UE**, moratorie).

*ImageNet = enorme database di immagini organizzato gerarchicamente

- Saturazione nel hype mediatico.
- Tuttavia, sviluppo tecnico continua: foundation models, RLHF, multimodalità.

🌞 AI Generativa e nuova stabilità (2024–2025)

• Diffusione di modelli multimodali (GPT-4, GPT-4o).

- Integrazione in: educazione, medicina, programmazione, design.

• Aumenta il focus su:

- Efficienza computazionale.
- Sicurezza (AI alignment).
- **AI personalizzate** (local models, on-device AI).

- Cresce anche la ricerca sull'**AGI** e sulla **neurosymbolic AI** (usa anche ragionamento simbolico (logica, regole, concetti) per creare un «modo di pensare» in modo strutturato e spiegabile).

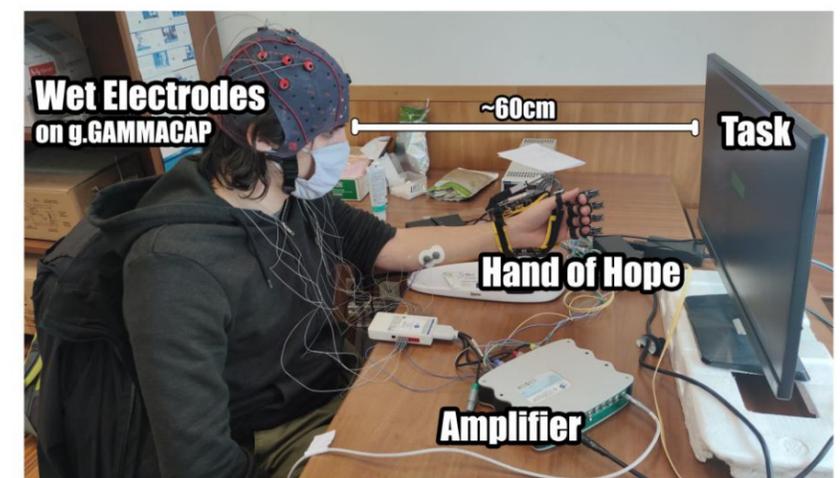
PROBLEMI APERTI

- Quantità e rappresentatività del dataset (training, generalization)
- Overfitting (generalizzazione su nuovi dati)
- Black-box problem
- Complessità computazionale → problemi di diversa COMPLESSITA'

Esempio: deep learning per la classificazione del movimento immaginato per brain-computer

interface (BCI)

<https://sites.google.com/view/giulia-cisotto/home-page/publications?authuser=0>



TEORIA DELLA COMPLESSITÀ (1/2)

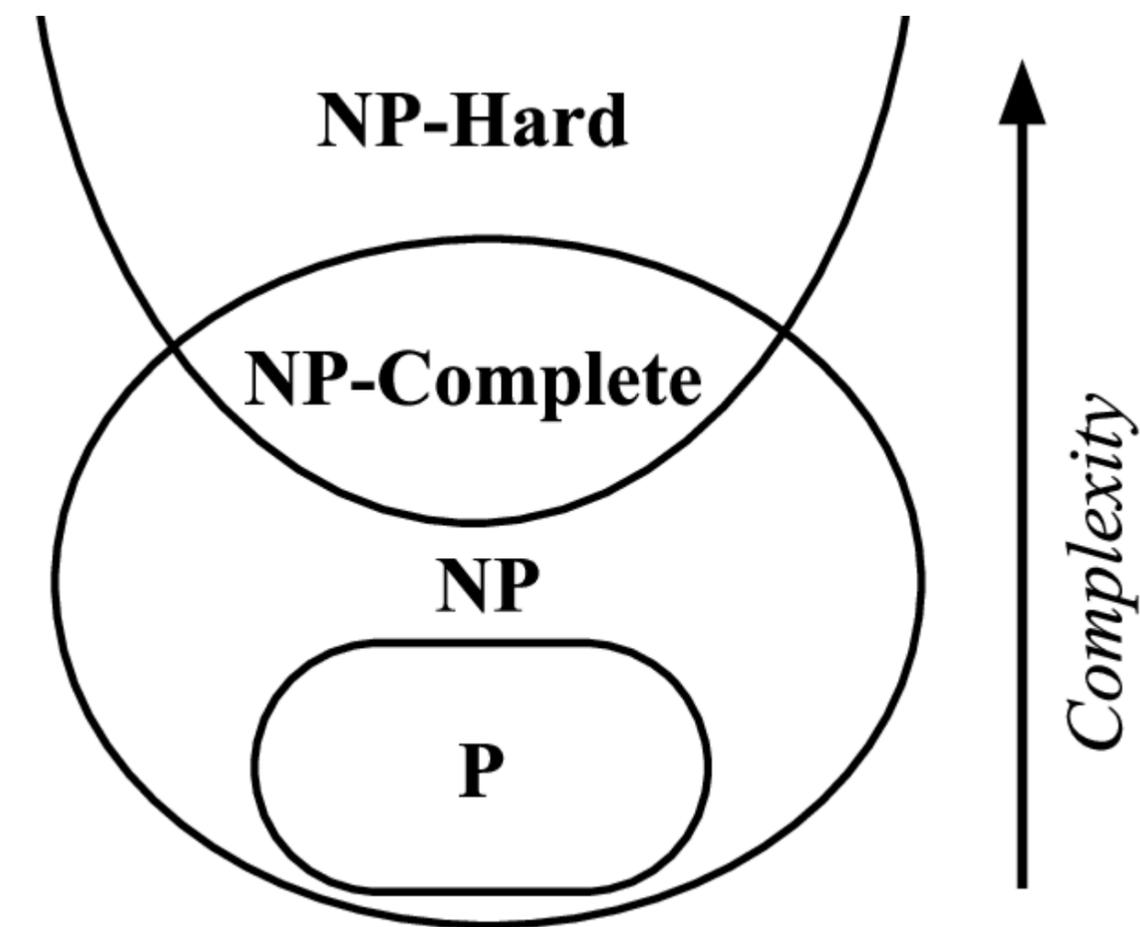
Problemi di diversa complessità in base al tempo/spazio che servono per risolverli o per verificare che una soluzione sia effettivamente risolutiva.

Problemi P: risolvibili in tempo polinomiale con algoritmi deterministici. *Es. ordinamento di un array con bubblesort.*

Problemi NP: posso verificare che una soluzione data è corretta in tempo polinomiale, ma non posso risolvere il problema in tempo polinomiale con un algoritmo deterministico.

Esempio: SAT (Satisfiability Problem), ovvero «Data una formula logica booleana (con variabili, AND, OR, NOT), esiste un'assegnazione di vero/falso alle variabili che rende la formula vera?»

Se ci sono n variabili, bisogna provare 2^n possibili assegnazioni da testare ($O(2^n)$).



TEORIA DELLA COMPLESSITÀ (2/2)

Problemi NP-completo (NP-C): se si trovasse un algoritmo in grado di risolvere "velocemente" (nel senso di utilizzare tempo polinomiale) un qualsiasi problema NP-completo, allora si potrebbe usarlo per risolvere "velocemente" *ogni* problema in NP.

Esempio: problema del commesso viaggiatore (versione decisionale), ovvero «Esiste un giro che visita tutte le città e costa al massimo K?»

Problemi NP-hard: Problemi più difficili di quelli NP. *Esempio: problema del commesso viaggiatore (versione di ottimizzazione), ovvero «qual è il giro più breve possibile che visita tutte le città una sola volta e torna al punto di partenza?»*



Il Problema
del
Commesso
Viaggiatore

INFORMATICA TEORICA

RICERCA OPERATIVA (OTTIMIZZAZIONE)

INTELLIGENZA ARTIFICIALE



**UNIVERSITÀ
DEGLI STUDI
DI TRIESTE**

