

# Cenni di AI

## Agenda:

- ❑ Che cos'è l'AI
- ❑ Cenni storici
- ❑ Machine Learning
- ❑ Deep Learning



## Perché?

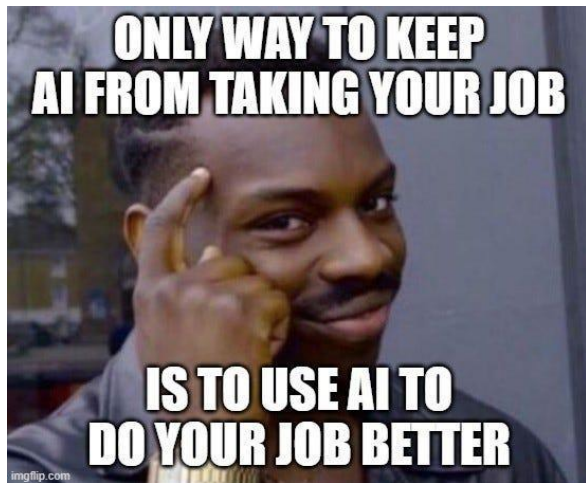
- ❑ L'intelligenza artificiale sta diventando sempre più una realtà applicabile nel day to day personale e lavorativo. Nonostante questo c'è ancora molta confusione su cosa effettivamente sia l'AI e come funzioni.
- ❑ Spesso il lavoro di un data analyst può avere una funzione propedeutica all'addestramento degli algoritmi di intelligenza artificiale: conoscere il funzionamento di questa tecnologia vi permette di aggiungere uno strumento ricercato alla vostra cassetta degli attrezzi.

## Alla fine di questa lezione saprete:

- ❑ Cos'è *davvero* l'intelligenza artificiale
- ❑ Qual è la differenza tra AI, Machine Learning e Deep Learning
- ❑ Come si sviluppa il processo di addestramento di un algoritmo

## Artificial Intelligence

L'Intelligenza Artificiale (IA) si riferisce alla simulazione dell'intelligenza umana nelle macchine che sono programmate per pensare e imparare.



## Origini concettuali

Nel 1950, Alan Turing pubblica il suo famoso articolo "Computing Machinery and Intelligence" in cui introduce il **Test di Turing**.

Il Test di Turing ha stimolato un ampio dibattito sulla possibilità che le macchine potessero pensare, un concetto che ha affascinato scienziati e ricercatori.

Questo interesse ha portato all'organizzazione della **Conferenza di Dartmouth**, nel 1956, dove l'obiettivo era proprio esplorare e definire il campo dell'intelligenza artificiale, che viene spesso considerata come l'inizio formale dell'AI come campo di studi



## Il test di Turing

Test proposto da Alan Turing per **determinare se una macchina possa mostrare comportamenti intelligenti indistinguibili da quelli umani**.

**Metodo:** Un interrogatore interagisce sia con un umano che con una macchina, entrambi nascosti alla vista. Se l'interrogatore non riesce a distinguere in modo affidabile la macchina dall'umano, la macchina è considerata aver superato il test.

## Anni '60-'70: Ricerca Precoce sull'IA e IA Simbolica

La ricerca sull'IA ha preso slancio con i progressi nella scienza informatica. Istituzioni come il MIT, Stanford e Carnegie Mellon sono diventate centri di ricerca sull'IA e danno inizio alla fase della **Symbolic AI: Un approccio all'IA in cui i simboli rappresentano concetti e le regole logiche manipolano questi simboli per eseguire il ragionamento.**

Due progetti chiave:

Logic Theorist (1955): Considerato il primo programma di IA, creato da Allen Newell e Herbert A. Simon, capace di dimostrare teoremi matematici.

ELIZA (1966): Un primo programma di elaborazione del linguaggio naturale di Joseph Weizenbaum che simulava uno psicoterapeuta rogeriano.



## Intelligenza Artificiale Simbolica: Approccio Basato su Regole

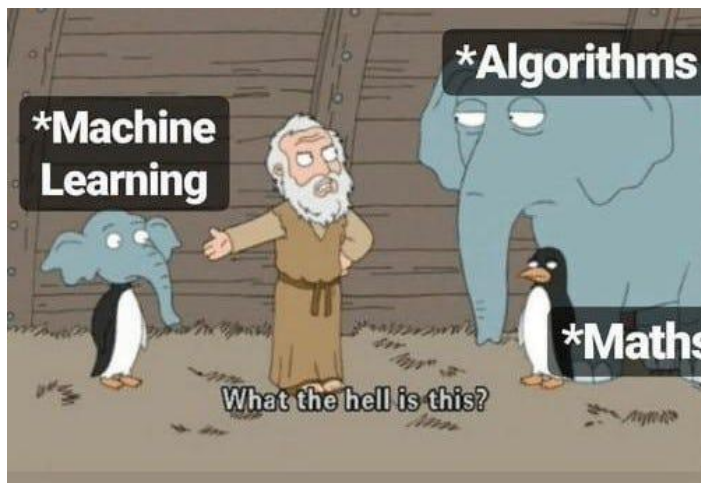
- ❑ **Rappresentazione della Conoscenza:** Nella AI simbolica, la conoscenza è rappresentata da simboli (come parole, numeri o altri caratteri) e da regole formali che descrivono le relazioni tra questi simboli.
- ❑ **Regole di Inferenza:** Queste regole sono utilizzate per fare deduzioni e prendere decisioni. Ad esempio, una regola potrebbe essere "Se X è vero, allora Y è vero".

Ad esempio:

- ❑ **Database di Fatti e Regole:** Un sistema AI simbolico ha un database di fatti (dati noti) e regole (che definiscono come i fatti sono correlati).
- ❑ **Esecuzione delle Regole:** Quando un nuovo fatto viene aggiunto al database, il sistema applica le regole per vedere se può dedurre nuovi fatti o prendere decisioni.
- ❑ **Esempio:** Se una regola dice "Se un animale ha le piume, allora è un uccello" e il sistema conosce il fatto "Tweety ha le piume", può dedurre che "Tweety è un uccello".

## Machine Learning

Il Machine Learning (ML) è una sotto-disciplina dell'IA focalizzata sulla costruzione di sistemi che imparano dai dati per migliorare le loro prestazioni su un compito.



## Supervised Learning

Un tipo di ML in cui il modello viene addestrato su **dati etichettati** (coppie input-output).

Comprende, tra gli altri:

- ❑ **Regressione**: Predizione di un valore continuo (ad esempio, prezzi delle case).
- ❑ **Classificazione**: Predizione di una categoria (ad esempio, rilevamento dello spam via email).

Vediamo nelle prossime slides due algoritmi che fanno parte del gruppo degli algoritmi supervisionati: la **regressione lineare** e gli **alberi decisionali**

## Linear Regression

Un algoritmo di **regressione** che modella la relazione tra una variabile dipendente e una o più variabili indipendenti usando un approccio lineare.

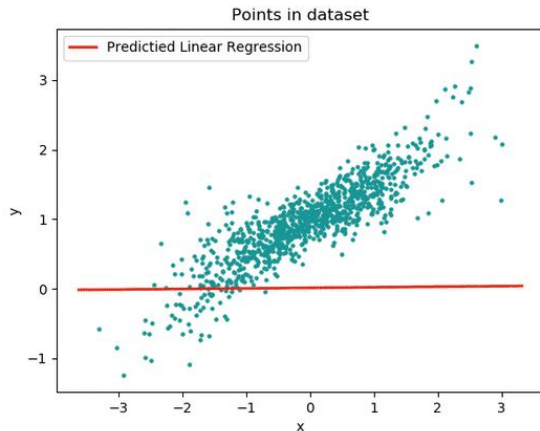
Equazione:  $y=mx+b$

y: Variabile dipendente (obiettivo)

x: Variabile indipendente (caratteristica)

m: Pendenza della retta

b: Intercetta



## Decision Tree

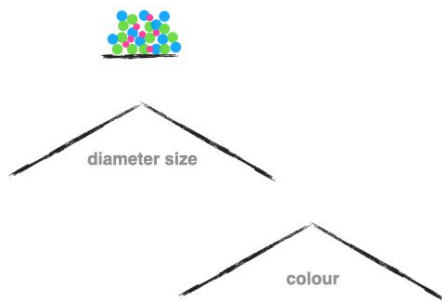
Gli alberi decisionali sono **modelli predittivi strutturati a forma di albero** che rappresentano decisioni e le loro possibili conseguenze, inclusi eventi casuali, costi e benefici.

- ❑ Ogni **nodo** interno dell'albero rappresenta una decisione o un test su un attributo
- ❑ Ogni **ramo** rappresenta l'esito del test o della decisione
- ❑ Ogni **foglia** rappresenta un risultato o una classe.

## Decision Tree

Ad esempio:

- ❑ Problema: Decidere se concedere un prestito a un cliente.
- ❑ Attributi: Età del cliente, reddito annuale, storico creditizio
- ❑ Nodi e Rami:
  - ❑ Nodo 1: "L'età del cliente è maggiore di 30?"
  - ❑ Se sì, procedere al Nodo 2.
  - ❑ Se no, procedere al Nodo 3.
  - ❑ Nodo 2: "Il reddito annuale è maggiore di 50.000€?"
  - ❑ Se sì, concedere il prestito (foglia).
  - ❑ Se no, procedere al Nodo 4.
  - ❑ Nodo 3: "Il cliente ha uno storico creditizio positivo?"
  - ❑ Se sì, concedere il prestito (foglia).
  - ❑ Se no, rifiutare il prestito (foglia).



## Unsupervised Learning

Un tipo di ML in cui il modello viene addestrato su **dati non etichettati**, identificando modelli e strutture senza istruzioni esplicite.

Comprendono:

- ❑ **Clustering**: Raggruppare i punti dati in cluster basati sulla similarità.
- ❑ **Riduzione della Dimensionalità**: Ridurre il numero di caratteristiche preservando al contempo informazioni importanti.

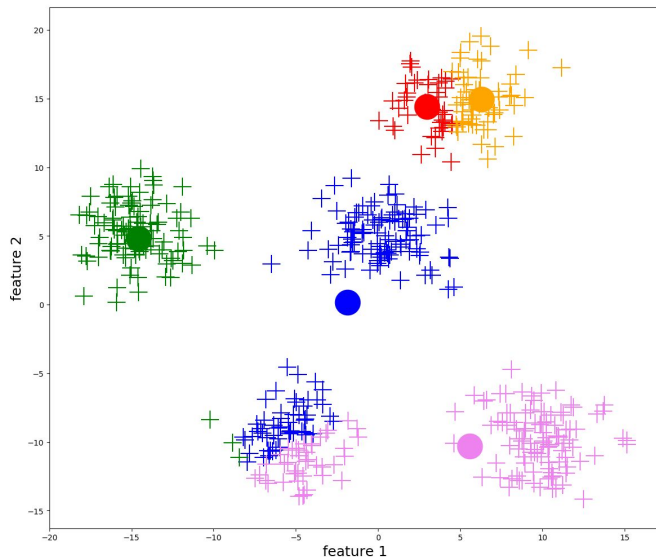
Vediamo come esempio l'algoritmo **K-Means**

## K-Means

K-Means è un **algoritmo di clustering** utilizzato per suddividere un set di dati in K cluster distinti e non sovrapposti.

Processo:

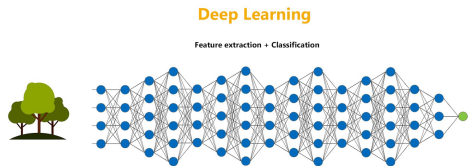
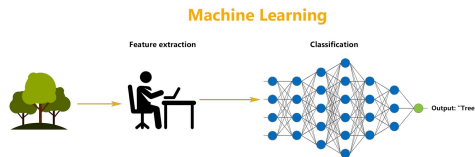
- ❑ **Inizializzazione**: Selezione casuale di K punti (centroidi) come centri di cluster iniziali.
- ❑ **Assegnazione**: Assegnazione di ciascun punto dati al centroide più vicino in base alla distanza euclidea.
- ❑ **Aggiornamento**: Ricalcolo dei centroidi come media di tutti i punti dati assegnati a ciascun cluster.
- ❑ **Ripetizione**: Assegnazione iterativa dei punti dati e aggiornamento dei centroidi fino alla convergenza





## Deep Learning

Il Deep Learning è una sotto-disciplina del ML che utilizza reti neurali con molteplici strati (reti neurali profonde) per modellare modelli complessi in grandi set di dati.



## Componenti fondamentali

Il Deep Learning si basa su **concetti complessi** che richiedono delle solide basi matematiche e statistiche per essere compresi. Diamo però una panoramica di quali sono i componenti fondamentali di un algoritmo di deep learning:

- ❑ **Neuroni**: Unità di base che ricevono, elaborano e trasmettono informazioni.
- ❑ **Strati (layer)**: Strati sovrapposti di neuroni, inclusi strati di input, nascosti e di output.
- ❑ **Funzioni di Attivazione (activation functions)**: Funzioni che introducono non linearità, aiutando la rete a imparare modelli complessi.

Concetti di Addestramento:

- ❑ **Epoca (epochs)**: Un passaggio completo attraverso l'intero set di dati di addestramento. Sono necessarie più epoche per migliorare le prestazioni del modello regolando gradualmente i pesi.
- ❑ **Funzione di Perdita (loss function)**: Una funzione che misura quanto bene le previsioni del modello corrispondano ai valori obiettivo effettivi. L'obiettivo dell'addestramento è minimizzare questa perdita.

## Come funzionano le reti neurali

Riconoscimento dei Pattern:

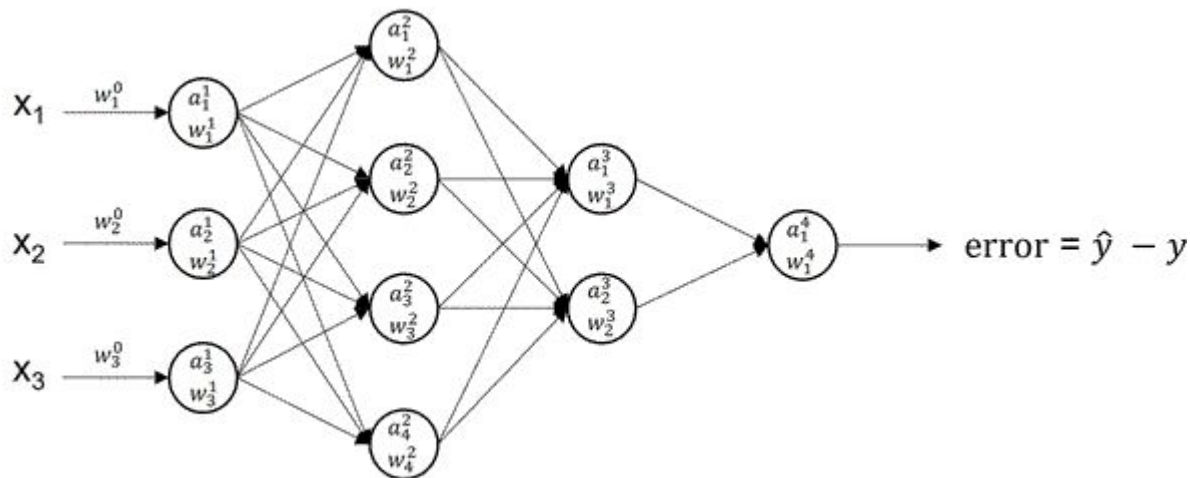
**Pesi e Bias**: Ogni connessione tra neuroni ha un peso associato che determina la forza del segnale trasmesso. I bias regolano l'output insieme ai pesi.

**Propagazione in Avanti (forward propagation)**: I dati di input passano attraverso la rete strato per strato, con ciascun neurone che applica una somma pesata e una funzione di attivazione all'input ricevuto.

**Retropropagazione (back propagation)**:

- ❑ **Calcolo dell'Errore**: La funzione di perdita (loss function) calcola l'errore nell'output.
- ❑ **Discesa del Gradiente**: L'algoritmo calcola i gradienti della funzione di perdita rispetto a ciascun peso e bias.
- ❑ **Aggiornamento dei Pesi**: I pesi vengono regolati per minimizzare l'errore, iterando questo processo per molte epoche fino a quando la rete apprende i pattern nei dati.

## Come funzionano le reti neurali



## Come funzionano le reti neurali

Riassumendo, possiamo dire che l'obiettivo delle reti neurali è di raggiungere la convergenza.

- ❑ Il processo di addestramento coinvolge quindi la **ricerca dei pesi ottimali** per minimizzare la funzione di perdita.
- ❑ L'obiettivo è quello di **ridurre gradualmente l'errore tra le previsioni della rete e i valori target** durante le epoche successive.
- ❑ La rete raggiunge la **convergenza** quando la funzione di perdita (loss function) raggiunge un plateau o smette di diminuire significativamente.
- ❑ Una volta addestrata, la rete viene **valutata su un set di dati di test** per valutarne le prestazioni.

## Le reti neurali come black box

All'aumentare del numero di strati e di parametri diminuisce l'interpretabilità della rete: per questo le grandi reti neurali (come ad esempio gli algoritmi di LLM) vengono descritte come "scatole nere":

- ❑ **Opacità**: Nonostante la loro efficacia nel modellare pattern complessi, i meccanismi interni delle reti neurali possono essere difficili da capire e interpretare.
- ❑ **Complessità**: Con molti strati e migliaia o milioni di parametri, le reti neurali possono catturare relazioni intricate nei dati, ma il ragionamento esatto dietro le loro decisioni potrebbe non essere immediatamente evidente.

Questo fa sorgere delle problematiche rilevanti:

- ❑ **Interpretabilità**: Capire perché una rete neurale prende una specifica previsione o classificazione può essere difficile, specialmente in spazi ad alta dimensione.
- ❑ **Fiducia**: La mancanza di trasparenza può portare a una fiducia ridotta nei sistemi di intelligenza artificiale, in particolare in settori critici come la sanità e la finanza.

## Le reti neurali come black box

- ❑ **Preoccupazioni Etiche**: La natura a scatola nera delle reti neurali solleva questioni etiche riguardanti la responsabilità, l'equità e il bias nei processi decisionali.
- ❑ **Bias e Equità**: Senza trasparenza, è difficile identificare e mitigare i bias codificati nei dati o appresi dal modello, portando a possibili discriminazioni.
- ❑ **Sfide Regolamentari**: Regolatori e legislatori affrontano difficoltà nel valutare la sicurezza, l'affidabilità e la conformità dei sistemi di intelligenza artificiale senza una chiara visibilità dei loro meccanismi interni.
- ❑ **Comprensione dell'Utente**: Gli utenti finali possono avere difficoltà a fidarsi o accettare le decisioni prese dai modelli a scatola nera, soprattutto quando sono in gioco vite umane o transazioni finanziarie significative.
- ❑ **Focus della Ricerca**: Affrontare il problema delle scatole nere è diventato un punto focale nella ricerca sull'IA.

## Dove si inserisce un data analyst in questi processi?

La progettazione dell'infrastruttura è di solito compito del data engineer, la scelta del modello e l'addestramento sono in carico al data scientist. Qual è allora il ruolo di un data analyst in questo mondo?

Alcuni dei ruoli che l'analista può ricoprire sono quelli dello **studio preliminare dei dati, della loro preparazione per il training e anche della parte di feature engineering**.

Vediamoli un po' più nel dettaglio.



## Raccolta e studio preliminare dei dati

- ❑ **Raccolta dei Dati:**  
Identificazione delle Fonti di Dati  
Determinare le fonti di dati rilevanti e disponibili.
- ❑ **Integrazione:**  
Raccogliere i dati da diverse fonti e integrarli in un formato coerente.
- ❑ **Assicurazione della Qualità dei Dati:**  
Verificare l'integrità e la qualità dei dati raccolti.
- ❑ **Documentare il processo** di raccolta dei dati e mantenere un tracciamento accurato delle fonti e delle metodologie utilizzate.

## Pulizia dei dati

- ❑ **Rimozione dei Dati Mancanti:**  
Identificare e rimuovere o imputare i valori mancanti.  
Esempio: Utilizzare la media, la mediana o un metodo avanzato per sostituire i valori mancanti.
- ❑ **Correzione dei Dati Errati:**  
Identificare e correggere valori errati o anomali nei dati.  
Esempio: Correggere errori di battitura, valori fuori scala o inconsistenti.
- ❑ **Rimozione dei Duplicati:**  
Identificare e rimuovere le voci duplicate per evitare bias nei risultati.  
Esempio: Utilizzare funzioni di deduplicazione basate su identificatori unici.
- ❑ **Normalizzazione e Standardizzazione:**  
Normalizzare o standardizzare i dati per garantire che tutte le feature abbiano lo stesso peso.  
Esempio: Ridimensionare le feature in un intervallo specifico (es. 0-1) o utilizzare lo Z-score.

## Feature engineering

Feature Engineering è il **processo di creazione di nuove variabili** (feature) che possono migliorare la performance del modello di intelligenza artificiale.

- ❑ **Creazione di Nuove Feature:**
  - ❑ Generare nuove feature a partire dai dati esistenti.
  - ❑ Esempio: Da una data di nascita, creare una nuova feature "età".
- ❑ **Trasformazione delle Feature:**
  - ❑ Applicare trasformazioni matematiche o logiche alle feature esistenti.
  - ❑ Esempio: Prendere il logaritmo di una feature per ridurre l'impatto degli outlier.

## Divisione del dataset in train-validation-test

Per garantire che il modello di intelligenza artificiale venga valutato su dati non visti durante il training, per assicurarsi che il modello generalizzi bene su nuovi dati, è prassi dividere i dati in tre set distinti:

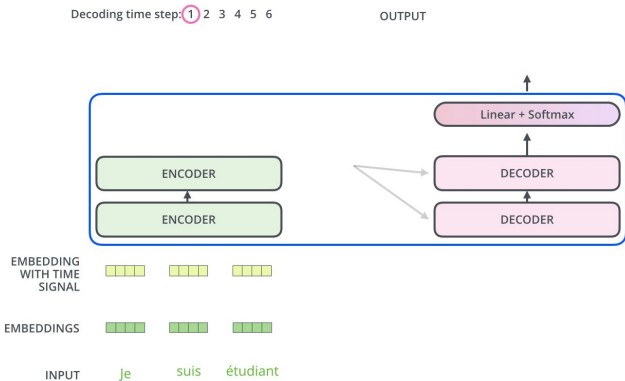
- ❑ **Train Set:**  
Definizione: Sottoinsieme di dati utilizzato per addestrare il modello.  
Scopo: Determinare i parametri del modello.
- ❑ **Validation Set:**  
Definizione: Sottoinsieme di dati utilizzato per valutare il modello durante l'addestramento.  
Scopo: Sintonizzare gli iperparametri e prevenire l'overfitting.
- ❑ **Test Set:**  
Definizione: Sottoinsieme di dati utilizzato per la valutazione finale del modello.  
Scopo: Fornire una misura imparziale delle performance del modello su dati non visti

## Cosa è un Large Language Models (LLM) ?

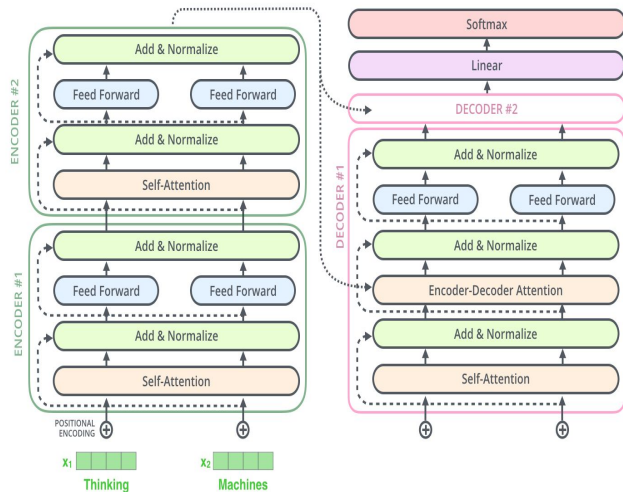
- ❑ Un modello di linguaggio ampio è un tipo di intelligenza artificiale **progettato per comprendere e generare testo umano.**
- ❑ Questi modelli sono addestrati su grandi quantità di dati testuali, che includono libri, articoli, conversazioni e molto altro.
- ❑ Gli LLM sono in grado di comprendere il **contesto delle parole**, le loro relazioni e generare testo che sembra scritto da un umano.
- ❑ Grazie alla loro capacità di elaborare e generare testo, gli LLM trovano applicazione in chatbot, assistenti virtuali, traduzione automatica, generazione di contenuti, analisi del sentiment e molto altro.
- ❑ Questi modelli rappresentano un avanzamento significativo nell'elaborazione del linguaggio naturale (NLP), permettendo **interazioni più naturali ed efficaci tra umani e macchine.**

## Quando nascono?

- ❑ Gli LLM nascono nel 2017 con la pubblicazione del paper **Attention is All You Need** scritto da 8 scienziati di Google.
- ❑ La **tecnologia Transformer** ha rivoluzionato l'elaborazione del linguaggio naturale.
- ❑ I modelli precedenti soffrivano di limitazioni nelle sequenze di testo lunghe.
- ❑ I Transformer, invece, utilizzano **meccanismi di attenzione** che permettono al modello di focalizzarsi su diverse parti della sequenza di input, migliorando l'efficienza e la capacità di catturare relazioni nel testo.
- ❑ Questa innovazione ha aperto la strada allo sviluppo di modelli sempre più grandi e potenti, culminando in LLM come **BERT**, **GPT-2** e **GPT-3**.



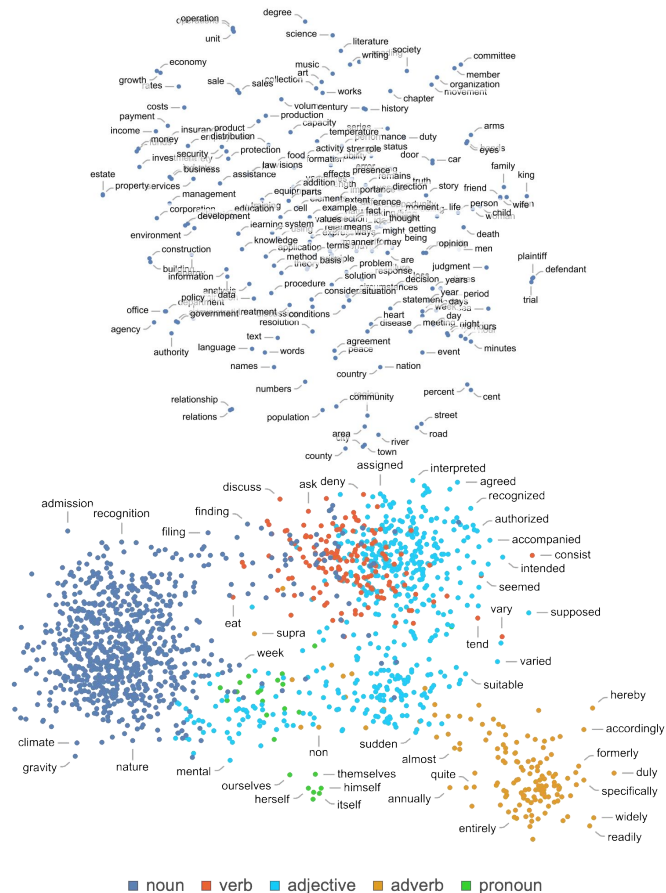
## Architettura di un LLM ed Embeddings



- ❑ Gli **strati di attenzione** permettono al modello di dare più peso a certe parole rispetto ad altre in base alla loro rilevanza nel contesto.
- ❑ Gli **embeddings** sono rappresentazioni numeriche delle parole che catturano il loro «significato semantico».
- ❑ Servono a posizionare le parole in uno spazio multidimensionale dove **la distanza tra i punti riflette la similarità semantica**.
- ❑ Un LLM può avere **milioni o miliardi di parametri**, che gli permette di catturare complessità linguistiche molto sottili.
- ❑ Gli **strati feed-forward** elaborano le informazioni in modo non lineare, permettendo al modello di fare previsioni accurate e generare testo coerente.

### Spazio dei significati e interpretazione semantica

- ❑ In un LLM, ogni parola o frase viene mappata in un vettore di dimensioni fisse, chiamato **embedding**.
- ❑ Questo spazio dei significati permette al modello di catturare la similarità semantica tra le parole: **parole con significati simili sono posizionate vicine nello spazio degli embeddings**.
- ❑ Ad esempio, "gatto" e "felino" avranno vettori simili.
- ❑ Questo approccio permette al modello di comprendere non solo il significato individuale delle parole, ma anche le loro relazioni e il **contesto in cui vengono utilizzate**.



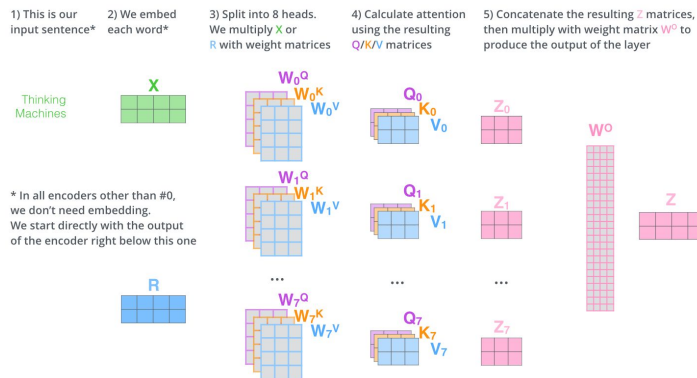


## Come un Data Analyst può sfruttare a suo vantaggio un LLM?

- ❑ Gli LLM possono **automatizzare** l'analisi dei dati testuali, classificare documenti, estrarre informazioni chiave e generare report dettagliati.
- ❑ Possono essere utilizzati per classificare grandi quantità di dati non strutturati in categorie significative, **semplificando la fase di pre-processing**.
- ❑ Utilizzando tecniche di NLP avanzate, un data analyst può anche migliorare la qualità dei dati mediante la **correzione automatica degli errori e la normalizzazione**.
- ❑ Possono assistere nella generazione di insight e raccomandazioni basate sui dati.
- ❑ Gli LLM rappresentano uno strumento potente per **potenziare le capacità analitiche e migliorare l'efficienza del lavoro di un data analyst**.

## Cosa fa ChatGPT e perché funziona?

- ❑ ChatGPT è un modello di linguaggio basato su Transformer **sviluppato da OpenAI**, progettato per comprendere e generare testo in modo naturale.
- ❑ Funziona utilizzando una rete neurale addestrata su un **vasto corpus di dati testuali**.
- ❑ Durante l'addestramento, il modello apprende le strutture grammaticali, il vocabolario e i modelli linguistici dai dati.
- ❑ La sua efficacia deriva dalla **capacità di comprendere il contesto** e generare risposte coerenti e pertinenti.
- ❑ La combinazione di un vasto set di dati di addestramento permette a ChatGPT di fornire risposte che sono spesso indistinguibili da quelle umane.



## Exploratory Data Analysis (EDA) con ChatGPT su un dataset

- ❑ ChatGPT può essere **utilizzato per descrivere i dati, identificare pattern nascosti e suggerire visualizzazioni appropriate.**
- ❑ Un analista può utilizzare ChatGPT per fornire una **sintesi delle principali caratteristiche di un dataset**, evidenziando tendenze e anomalie.
- ❑ ChatGPT può anche **generare codice Python per creare grafici e tabelle.**
- ❑ Può assistere nell'identificazione di **correlazioni tra variabili.**
- ❑ Questo approccio interattivo e intuitivo permette al Data Analyst di concentrarsi sulle decisioni strategiche, lasciando al modello il compito di gestire le operazioni più ripetitive e tecniche.



## Setup

- ❑ Collegarsi al sito <https://chatgpt.com>
- ❑ Personalizzare una **ChatGPT Personas**
- ❑ Usare il **GPTs Data Analyst**
- ❑ Il modello è in continua evoluzione grazie ai nostri feedback
- ❑ La versione gratuita ha un limite consentito di risposte

Personalizza ChatGPT

Istruzioni personalizzate ⓘ

Cosa pensi dovrebbe sapere di te ChatGPT per darti risposte migliori?

- Profession/Role: Data Scientist
- Key Responsibilities: Data analysis, predictive modeling, data visualization.
- Knowledge or Expertise: Statistics, machine learning, data wrangling.
- Typical Challenges: Data quality, algorithmic bias, scalability

532/1500

Come vorresti che rispondesse ChatGPT?

- Tone and Formality: Analytical, insightful, data-centric.
- Level of Detail: Detailed analysis methods, algorithm explanations.
- Preferred References: Scientific papers, data science frameworks.
- Examples or Analogies: Successful data science projects

709/1500

Funzionalità di GPT-4 ⓘ

☒ Navigazione su web

☒ DALL·E

☒ Codice

Abilita per le nuove chat ☒

Annulla Salva

Miglioramento del modello

Migliora il modello per tutti ☒

Accconsenti all'uso dei tuoi contenuti per addestrare i nostri modelli, permettendo così a ChatGPT di essere più utile a te e a tutti i suoi utenti. Adottiamo le misure necessarie per tutelare la vostra privacy. [Scopri di più](#)

Fatto

**Data Analyst**

Di ChatGPT ⓘ

Drop in any files and I can help analyze and visualize your data.

**Other**  
Categoria

**Funzionalità**

☒ Analisi dati

**Valutazioni**

Non ci sono ancora valutazioni sufficienti

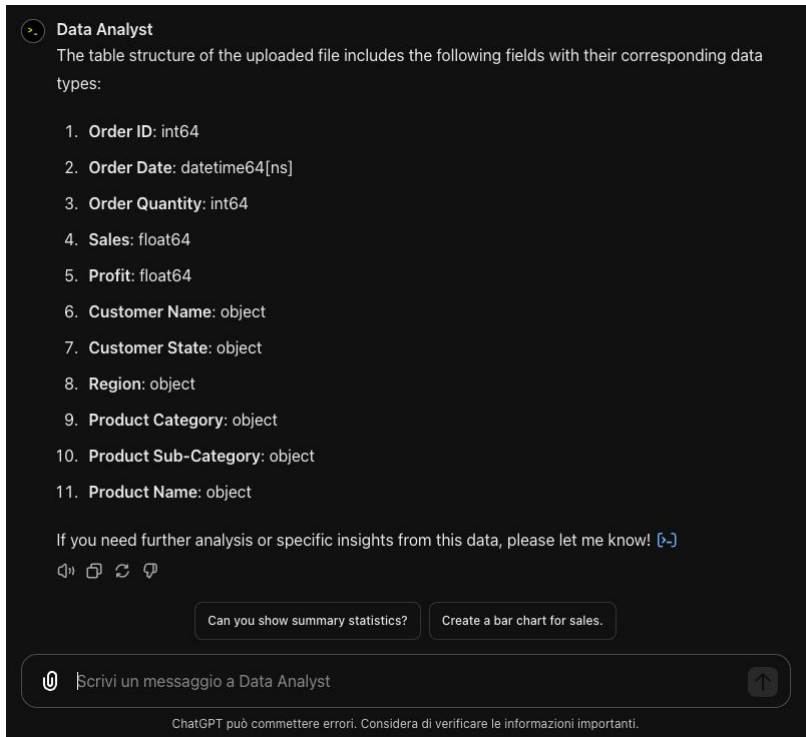
**Altri contenuti di ChatGPT**

**Web Browser**  
Consentì a ChatGPT di accedere al web per aiutarti a raccogliere informazioni o condurre ricerche.

☒ Avvia chat

## Esempio di EDA

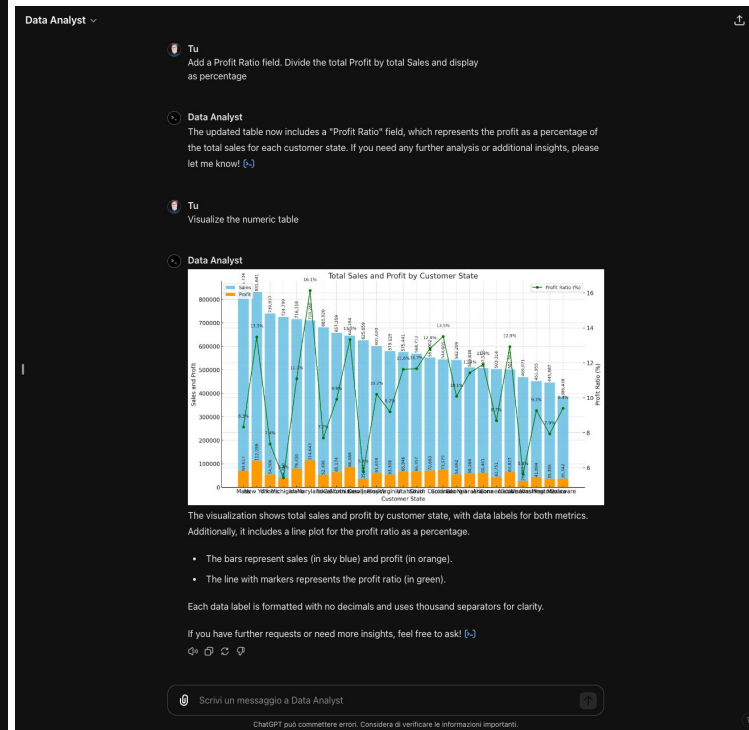
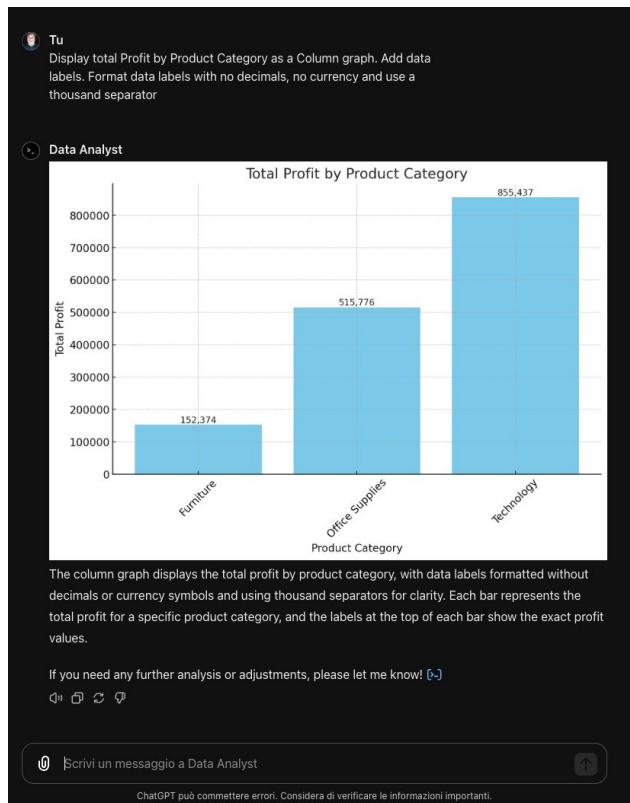
- Carichiamo un dataset su ChatGPT ed iniziamo l'esplorazione dei dati



## PROMPTING



ChatGPT in risposta  
ai nostri prompt  
visualizza grafici più  
o meno complessi



## Analisi dei dati

Ci consente di effettuare delle analisi sui nostri dataset e fornirci una spiegazione...

**Tu**  
How many Customers are in the data set and what is the average Profit

**Data Analyst**  
The data set contains 795 unique customers, and the average profit is approximately 181.40. If you need any further analysis or have any other questions, please let me know! 🙌

**Tu**  
How many Customers have a average Profit over 100

**Data Analyst**  
There are 407 customers with an average profit over 100. If you need any further analysis or have any other questions, please let me know! 🙌

🗨️ 🔄 📄

Show customers' average profit distribution. List customers with lowest average profit.

🗨️ Scrivi un messaggio a Data Analyst

ChatGPT può commettere errori. Considera di verificare le informazioni importanti.

**Tu**  
Explain the calculation

**Data Analyst**  
Here's an explanation of the calculations for each field in the table:

- Total Profit:** This is the sum of all profits for each product sub-category. It represents the overall profit made from all sales in that sub-category.
- Average Profit:** This is the mean profit per transaction for each product sub-category. It is calculated by dividing the total profit by the number of transactions within that sub-category.
- Median Profit:** This represents the middle value of the profit data for each product sub-category when the profits are arranged in ascending order. It is a measure of central tendency that is less affected by extreme values (outliers) compared to the mean.
- Highest Profit:** This is the maximum profit made in a single transaction for each product sub-category. It shows the highest profit value recorded.
- Lowest Profit:** This is the minimum profit made in a single transaction for each product sub-category. It shows the lowest profit value recorded, which can sometimes be negative indicating a loss.
- First Order Date:** This is the earliest date when a product from the sub-category was ordered. It provides a reference point for when the sub-category started being sold in the data set.

These calculations provide a comprehensive summary of the financial performance of each product sub-category, highlighting both overall trends and specific instances of high or low profitability. If you need further details or additional metrics, please let me know!

🗨️ 🔄 📄

Visualize sales and profit data. Segment data by customer region.

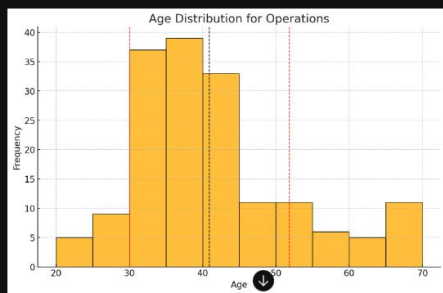
🗨️ Scrivi un messaggio a Data Analyst

ChatGPT può commettere errori. Considera di verificare le informazioni importanti.

**Tu**  
Create a histogram for our Ages. Include standard deviation and to create a separate for each job type. Provide an explanation of the results

**Data Analyst**

Age Distribution For Operations



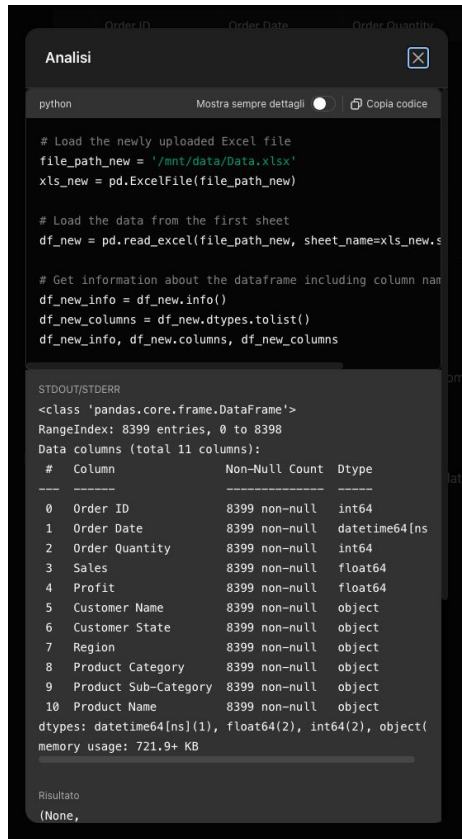
🗨️ 🔄 📄

🗨️ Scrivi un messaggio a Data Analyst

ChatGPT può commettere errori. Considera di verificare le informazioni importanti.

## Codice PYTHON

- ❑ ChatGPT usa il linguaggio Python.
- ❑ Su richiesta possiamo visualizzare il codice generato.
- ❑ Di default ChatGPT usa la libreria Matplotlib, ma potremmo usare anche altre librerie come Seaborn per visualizzare grafici più elaborati.



The screenshot shows a ChatGPT interface with a dark theme. At the top, there are three tabs: 'Order ID', 'Order Date', and 'Order Quantity'. Below the tabs is a header 'Analisi' with a close button. The main area is divided into two sections. The top section is labeled 'python' and contains a code editor with the following Python code:

```
# Load the newly uploaded Excel file
file_path_new = '/mnt/data/Data.xlsx'
xls_new = pd.ExcelFile(file_path_new)

# Load the data from the first sheet
df_new = pd.read_excel(file_path_new, sheet_name=xls_new.s

# Get information about the dataframe including column names
df_new_info = df_new.info()
df_new_columns = df_new.dtypes.tolist()
df_new_info, df_new.columns, df_new_columns
```

The bottom section is labeled 'STDOUT/STDERR' and shows the output of the code:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8399 entries, 0 to 8398
Data columns (total 11 columns):
#   Column                Non-Null Count  Dtype
---  ---
0   Order ID               8399 non-null  int64
1   Order Date             8399 non-null  datetime64[ns]
2   Order Quantity         8399 non-null  int64
3   Sales                  8399 non-null  float64
4   Profit                 8399 non-null  float64
5   Customer Name          8399 non-null  object
6   Customer State         8399 non-null  object
7   Region                 8399 non-null  object
8   Product Category       8399 non-null  object
9   Product Sub-Category   8399 non-null  object
10  Product Name           8399 non-null  object
dtypes: datetime64[ns](1), float64(2), int64(2), object(
memory usage: 721.9+ KB
```

At the bottom, there is a 'Risultato' section showing '(None,'.



## La chiave del Prompting

- ❑ I Large Language Models (LLM) imparano da enormi quantità di testo e non fanno altro che continuare il testo con la parola successiva che ha la maggiore probabilità.
- ❑ Nel momento in cui vado a scrivere un prompt che cosa sto facendo?
- ❑ Sto cercando di allineare la predizione del language model al token successivo.
- ❑ Bisogna proporgli degli esempi e incanalarlo sulle informazioni che vogliamo ci restituisca.
- ❑ Il Language Model non è un essere senziente, è un semplice enorme auto complete e quindi se lo incanali come auto complete funziona molto meglio.
- ❑ In questo senso scrivere dei prompt è un po' come programmare con il linguaggio naturale.
- ❑ Con il linguaggio entriamo all'interno della sua rete neurale e facciamo emergere tutta una serie di pattern latenti e questi pattern latenti ci possono portare dove ci serve:  
questa è la chiave del prompting.



**GRAZIE**  
Epicode