Scuola estiva virtuale Climate Change AI

Introduzione all'IA

Tejumade Afonja

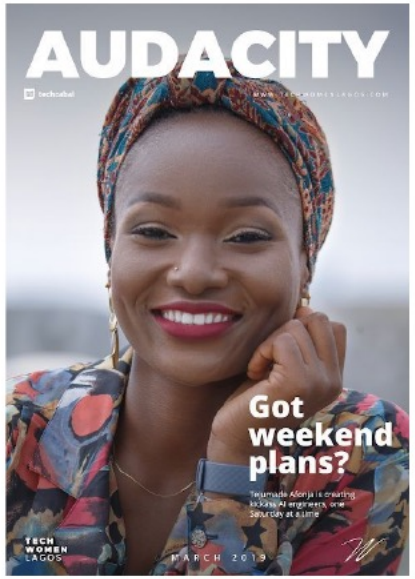
Slide adattate dal workshop pilota dell'AI Climate Institute

23 luglio 2026

Avvertenza

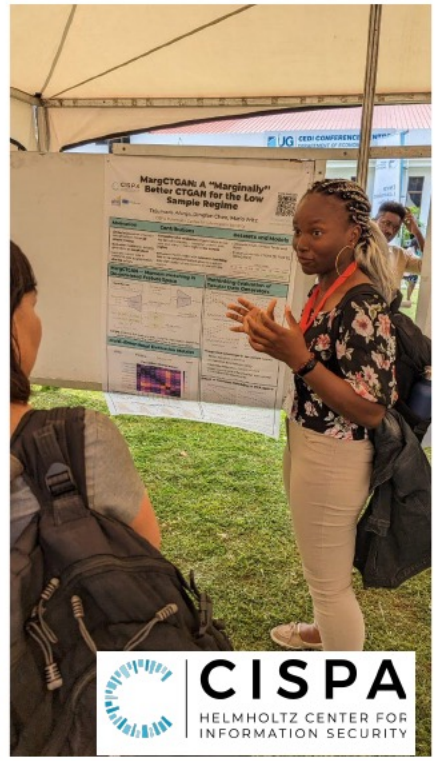
- Queste slide sono adattate dal workshop pilota dell'AI Climate Institute (Pilot Workshop).
- Le ho modificate leggermente per includere esercizi in classe e contenuti aggiuntivi sull'algoritmo di regressione lineare.
- Trovo eccellente la presentazione originale e ho scelto di usarne il materiale. Citate sempre le slide originali.
- Sono una ricercatrice di machine learning, ma non (ancora) di clima: la maggior parte degli esempi climatici non rientra nelle mie competenze, a differenza dei fondamenti del ML.

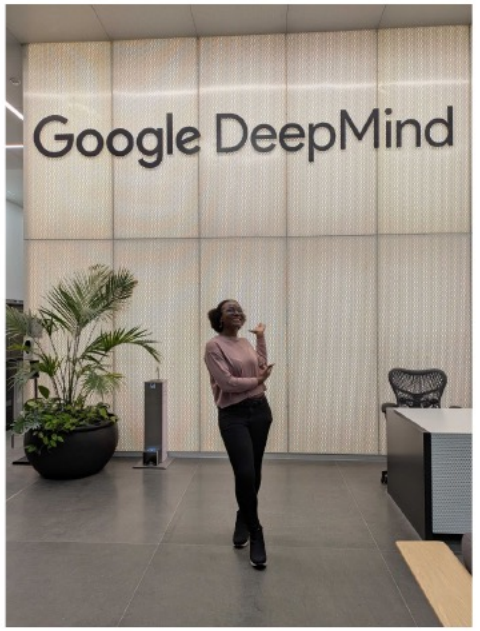
Chi sono









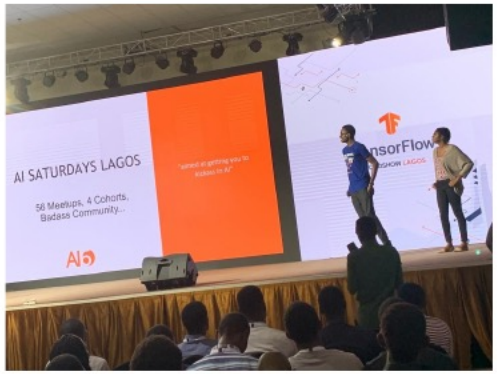












- Docente e ricercatrice
  - Cofondatrice, TRI AI & AI Saturdays Lagos (oggi TRI AI Saturdays)
  - Dottoranda, CISPA Helmholtz Center for Information Security
- Presidente generale, Deep Learning Indaba 2025; ex studentessa ricercatrice, Google DeepMind; borsista DSSG 2022

Sito web: tri-ai.org

Indice della lezione

Che cos’è il machine learning?

Tipi di machine learning

Dati

Capire l’algoritmo di regressione lineare

Domande e risposte

Indice della lezione

Che cos'è il machine learning?

Il ML rispetto ad altri approcci

Punti di forza e limiti

Tipi di machine learning

Dati

Capire l'algoritmo di regressione lineare

Domande e risposte

Che cos’è l’IA?

Intelligenza artificiale (IA): qualsiasi algoritmo che consente a una macchina di svolgere un compito complesso (es. compiti associati all’intelligenza umana)

L’IA indica lo sviluppo di sistemi informatici capaci di emulare l’intelligenza umana. Questi sistemi possono svolgere compiti che di norma richiedono capacità cognitive umane, come risolvere problemi, imparare dai dati e prendere decisioni.

https://runsystem.net/en/news/learn-about-ai-terminology-2024-ani-agi-and-asi

Che cos’è l’IA?

Intelligenza artificiale (IA): qualsiasi algoritmo che consente a una macchina di svolgere un compito complesso (es. compiti associati all’intelligenza umana)

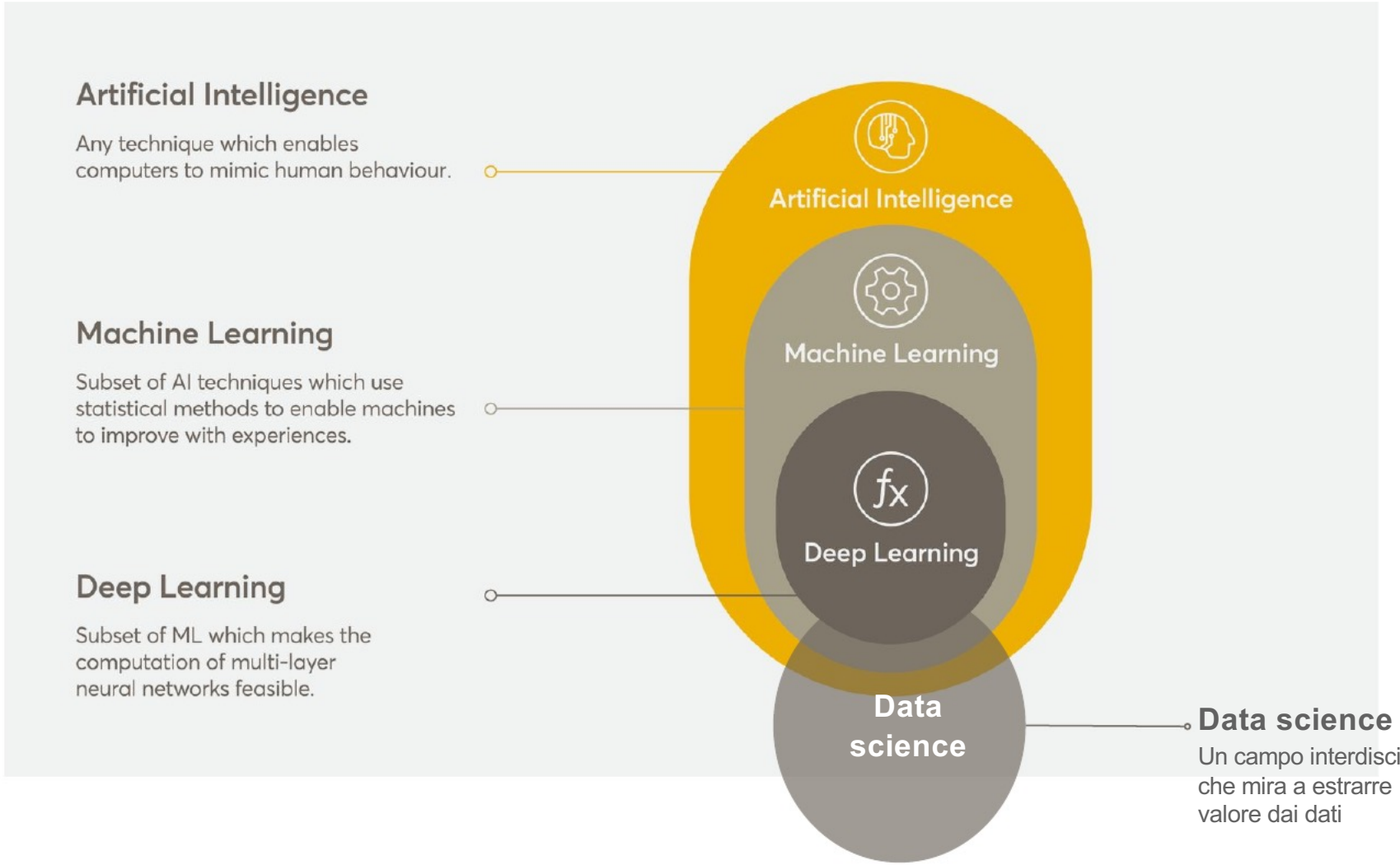
Sistemi basati su regole:
tecniche che ragionano automaticamente su un insieme esplicito di regole



Machine learning (ML):
tecniche che estraggono automaticamente schemi dai dati



Estrarre automaticamente schemi dai dati



https://en.idei.club/27516-iot-artificial-intelligence.html

Esercizio: sistemi basati su regole



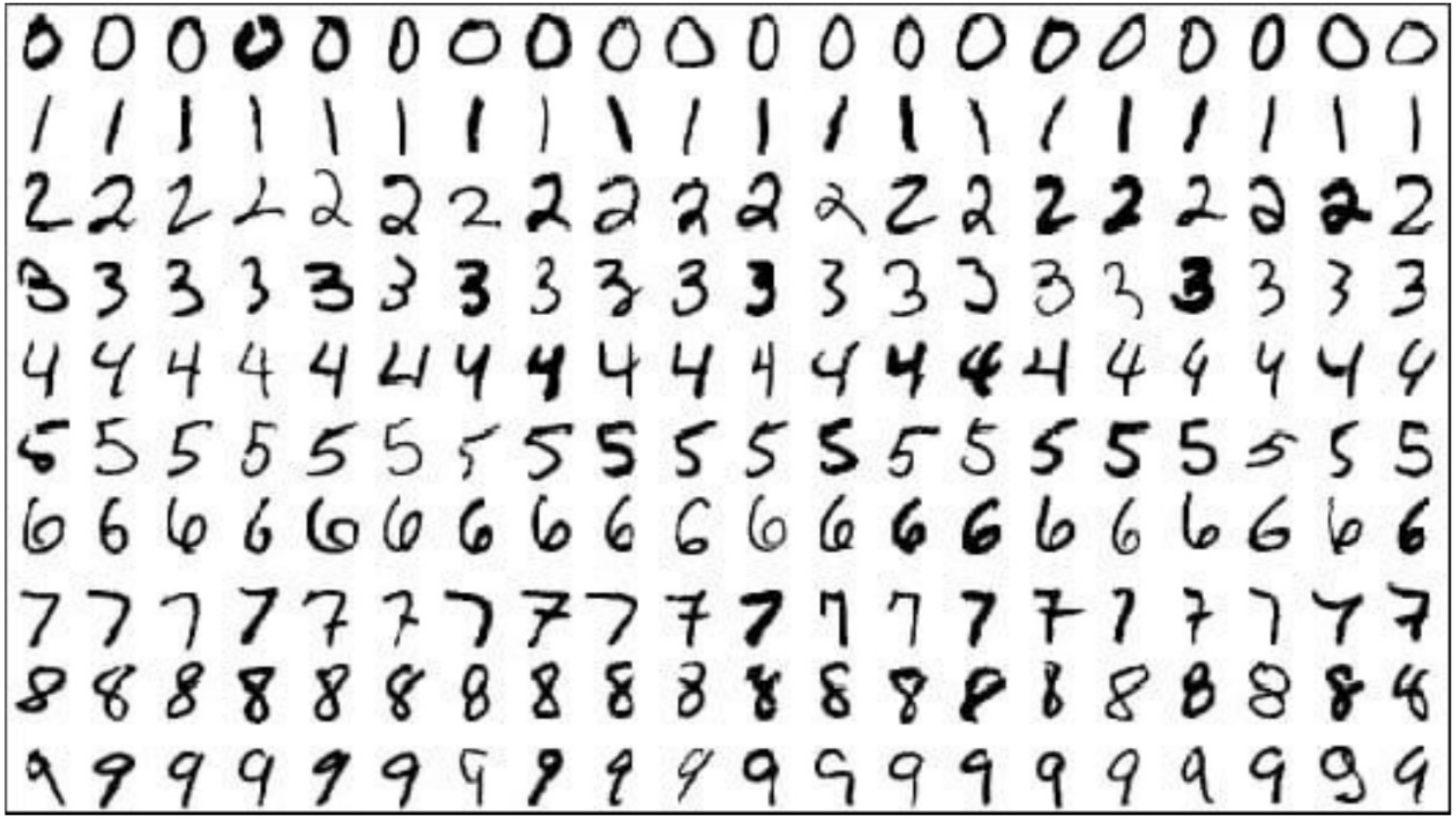
Scrivete un programma semplice per riconoscere la cifra 9 in un’immagine

Esercizio: sistemi basati su regole



il vostro programma funziona ancora? Se no, pensate a come modificarlo

Machine learning: imparare schemi generali dai dati



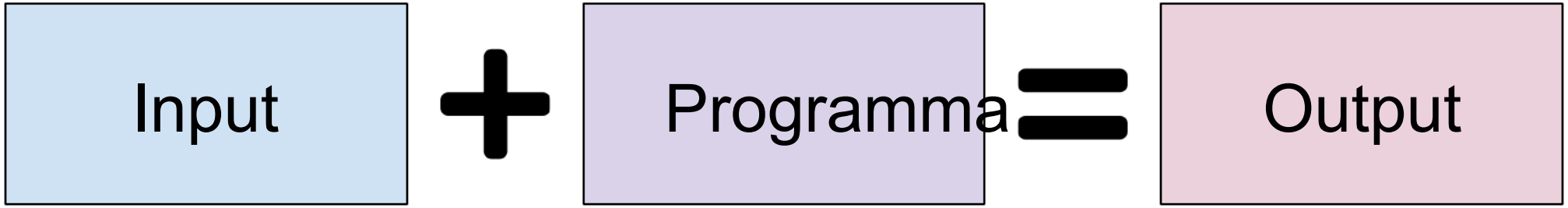
Che cos’è il machine learning (ML)?

«Il machine learning è un campo di studio che dà ai computer la capacità di imparare senza essere programmati esplicitamente»

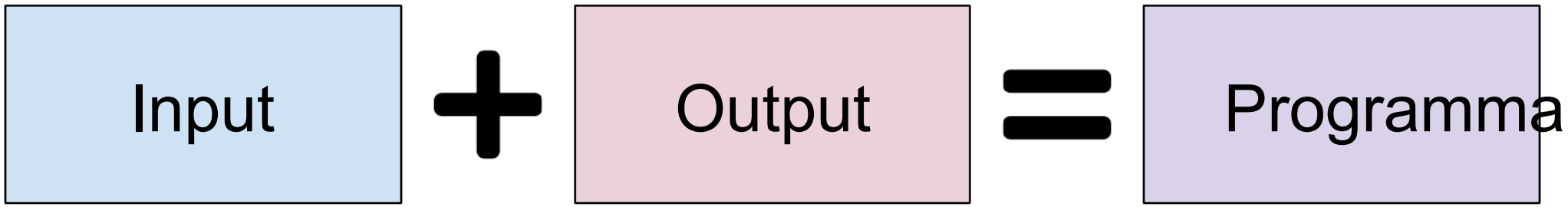
— Arthur Samuel (1959)

Programmazione tradizionale e ML

Programmazione tradizionale



Machine learning (ML)



Alcuni punti di forza e limiti del ML

PUNTI DI FORZA

- Sfruttare gli schemi nei dati
- Svolgere compiti su larga scala
- Ottimizzare sistemi complessi
- Capacità di essere «sempre attivo»
- Combinare input eterogenei
- Integrarsi con altri metodi

LIMITI

- È uno strumento: non adatto a ogni compito
- Può sbagliare (con eccessiva sicurezza)
- Può richiedere molti dati e/o calcolo
- «Garbage in, garbage out»
- Eredita i bias di dati/progettazione/uso
- Presuppone che regole/schemi siano persistenti

Quando usare il ML?

- Non sempre: non tutti i problemi possono o devono essere risolti con il machine learning!
  - Usate il ML quando ha più senso dedurre le regole dagli esempi che scriverle
- Quando disponete di dati che «corrispondono» al compito
  - Risoluzione, copertura, campione rappresentativo di una distribuzione adeguati, ecc.
- Quando le stime sono utili e gli errori ammissibili
  - «Tutti i modelli sono sbagliati, alcuni sono utili» – George Box
- Quando è una soluzione conveniente
  - I costi includono (i) creazione ed elaborazione del dataset e (ii) sviluppo, implementazione e manutenzione del modello

Machine learning

Machine learning (ML): estrarre automaticamente schemi dai dati

MODELLI

Reti neurali (deep learning)
Macchine a vettori di supporto
Alberi decisionali
Modelli bayesiani
…

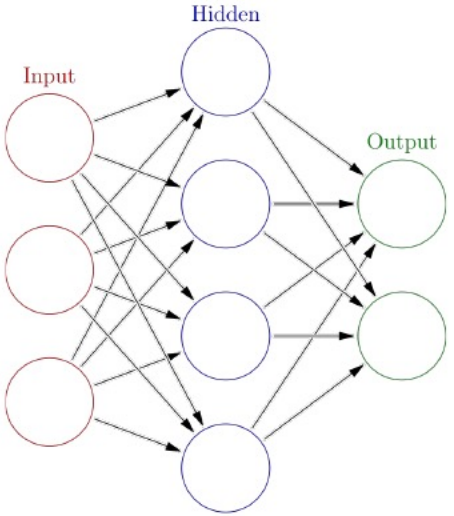
TIPI DI DATI / CASI D'USO

Computer vision
Elaborazione del linguaggio naturale
Robotica
Elaborazione dei segnali
…

PARADIGMI

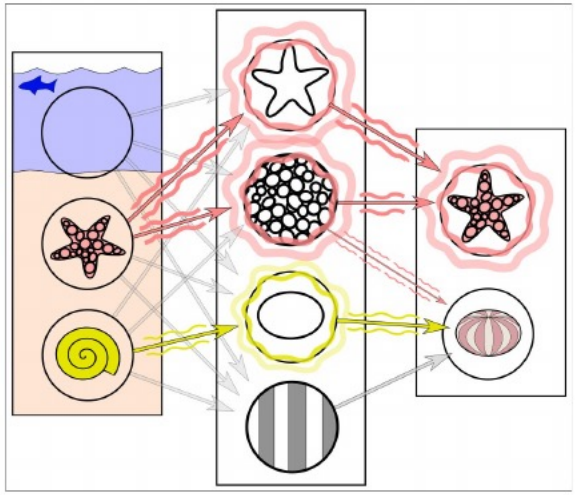
Modelli generativi vs discriminativi
Informati dalla fisica e robusti
Interpretabili
Modelli specifici vs generalisti
…

Che cos’è un modello di ML?



Il modello è costituito da parametri e da una struttura che insieme trasformano input in output

La parte di «apprendimento» significa che questi parametri vengono regolati con l’addestramento per produrre un «modello appreso» capace di svolgere un compito



Modelli «grandi» e «piccoli»

- **Ampio spettro di modelli per numero di parametri**
- I modelli piccoli girano in pochi secondi sulla maggior parte dei portatili
- I modelli più grandi possono richiedere un server (es. una GPU potente per mezza giornata)
- Quelli molto grandi possono richiedere 10.000+ GPU per diversi mesi

```
from sklearn . datasets import load_iris
from sklearn . ensemble import RandomForestClassifier

# Caricare i dati
X , y = load_iris ( return_X_y = True )

# Creare, addestrare e prevedere
model = RandomForestClassifier ()
model . fit ( X , y )

predictions = model . predict ( X [: 5 ])
print ( predictions )
```

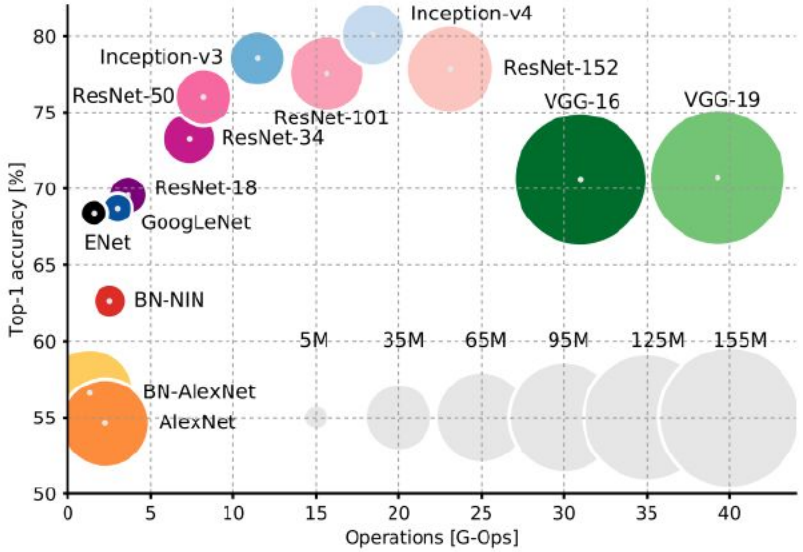


Image: https://arxiv.org/pdf/1605.07678

Modelli specifici e modelli generalisti

Modello specifico

- Addestrato per un solo compito
- Spesso addestrato su una sola modalità di dati
- Esempio: segmentazione per tipo di uso del suolo su immagini satellitari

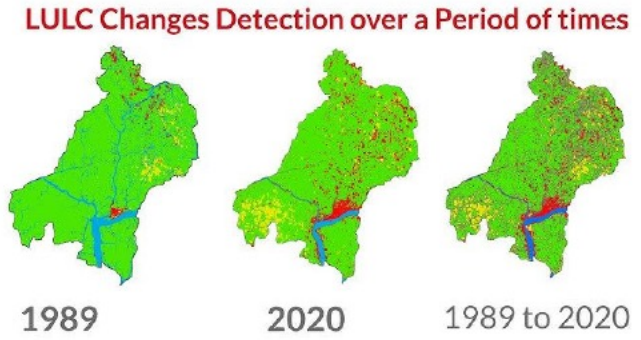


Modelli generalisti

- Mirano a gestire molti tipi di compiti in domini diversi
- Addestrati su dati ampi e diversificati
- Esempio: modelli fondazionali geospaziali per la segmentazione per uso del suolo, rilevamento dei cambiamenti d'uso, ecc.

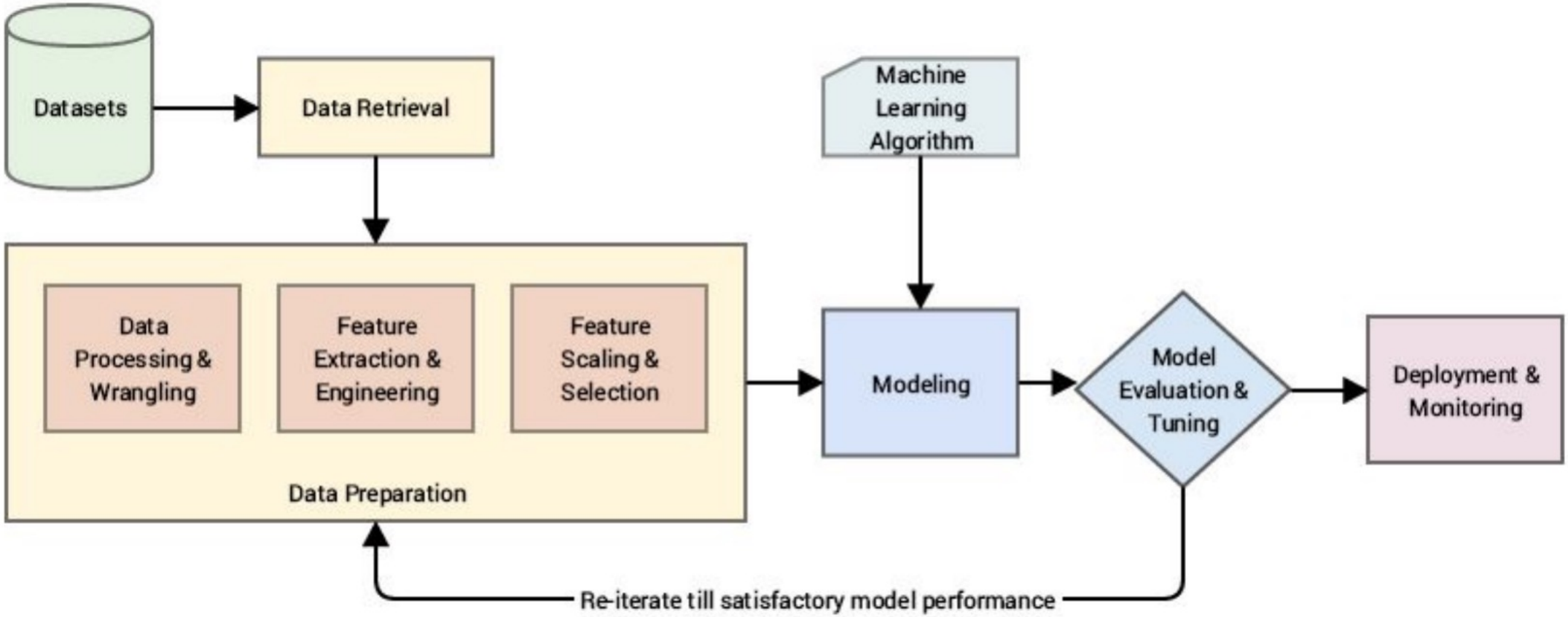


+



+ ...

Il ML in pratica



Practical Machine Learning with Python. Bali, Sarkar, and Sharma.

Sviluppo → addestramento → inferenza

Sviluppo

Predisporre l'infrastruttura della pipeline di dati

Decidere come valutare le prestazioni del modello

Provare diversi algoritmi e iperparametri

Scegliere il modello da usare per l'addestramento finale

Addestramento

«Adattare» i parametri del modello al compito

Scegliere dati di addestramento rappresentativi del contesto di inferenza

Può durare da minuti a mesi

Va ripetuto se la distribuzione dei dati cambia

Inferenza

Produrre previsioni per input «mai visti»

Usa i parametri «congelati» dell'addestramento

Presuppone che gli input «mai visti» abbiano distribuzione simile ai dati di addestramento

Di solito molto più rapida dell'addestramento (da millisecondi a secondi)

Esercizio: flusso di sviluppo nel ML

Secondo voi, quale di queste fasi richiede più tempo in un vero progetto?

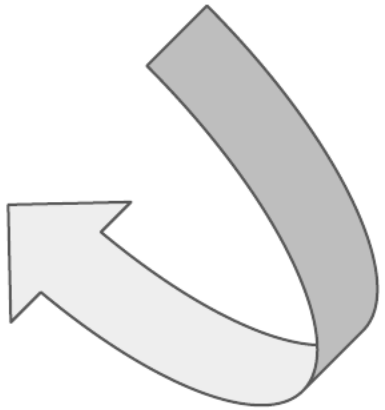
Sviluppo

Predisporre l'infrastruttura della pipeline di dati

Decidere come valutare le prestazioni del modello

Provare diversi algoritmi e iperparametri

Scegliere il modello da usare per l'addestramento finale



https://partici.fi/53603205

Indice della lezione

Che cos’è il machine learning?

Tipi di machine learning

Dati

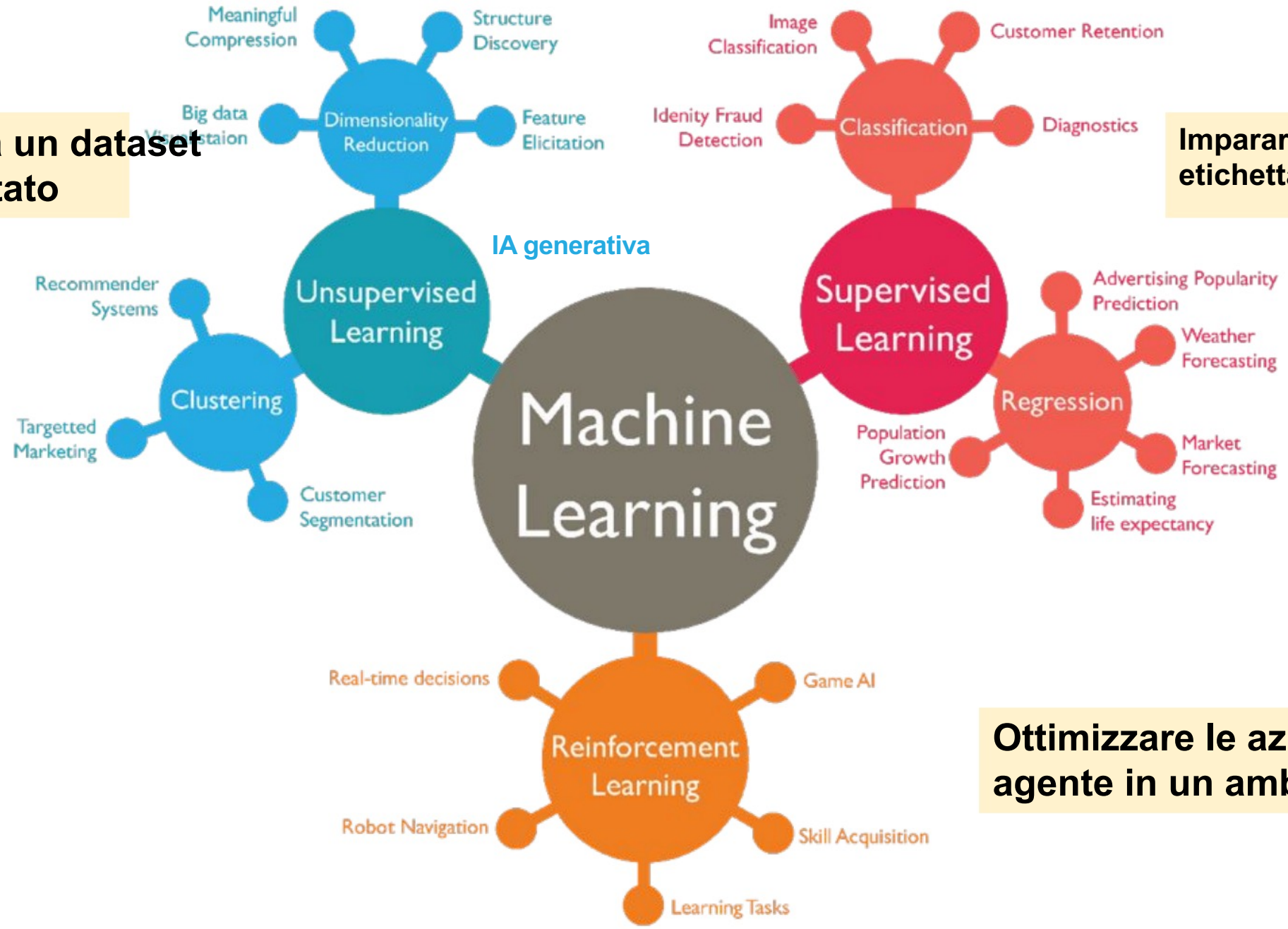
Capire l’algoritmo di regressione lineare

Domande e risposte

Principali paradigmi

Sottocampi del ML rilevanti per il clima

Imparare da un dataset non etichettato

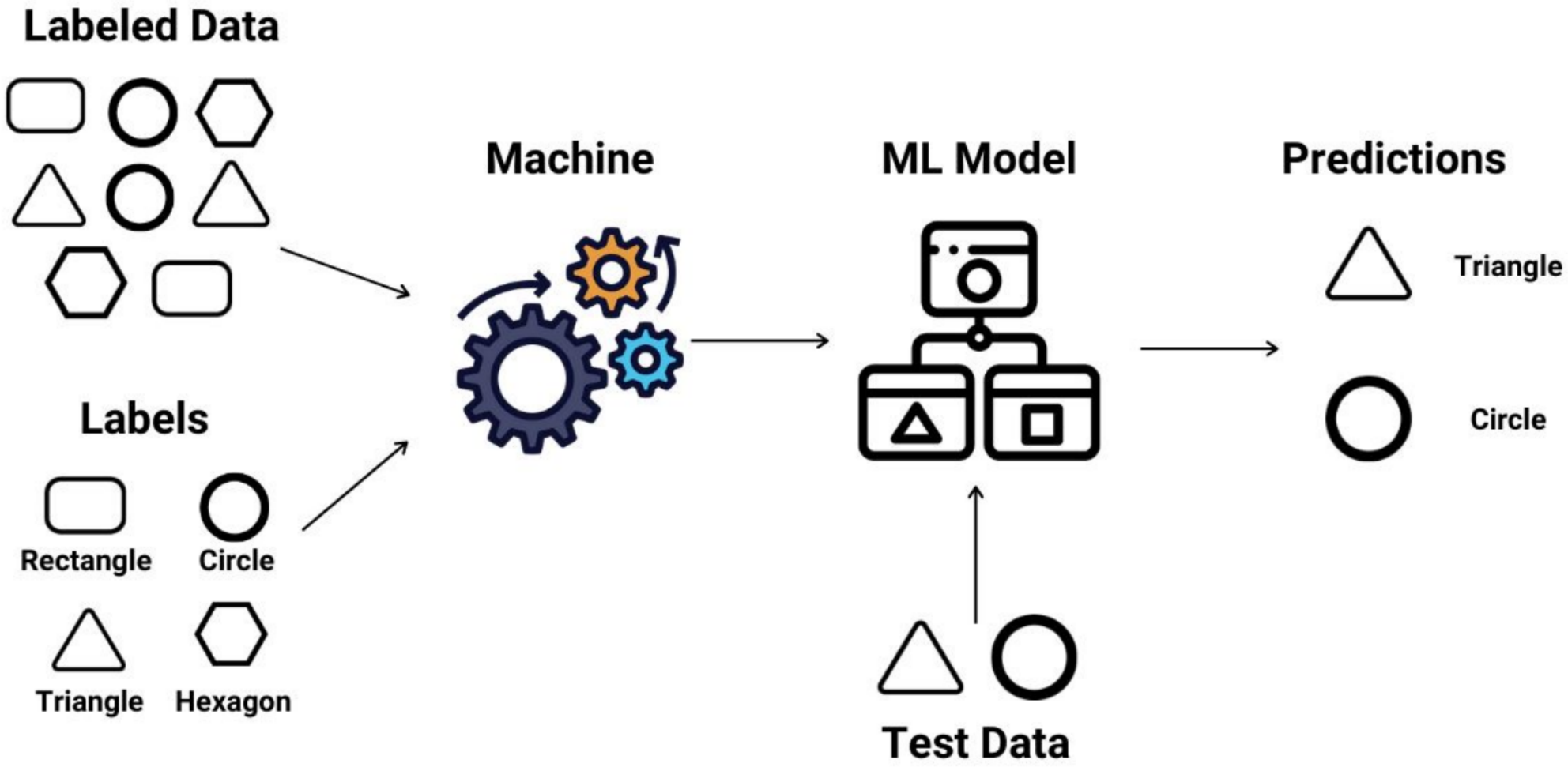


IA generativa

Imparare da un dataset etichettato

Ottimizzare le azioni di un agente in un ambiente

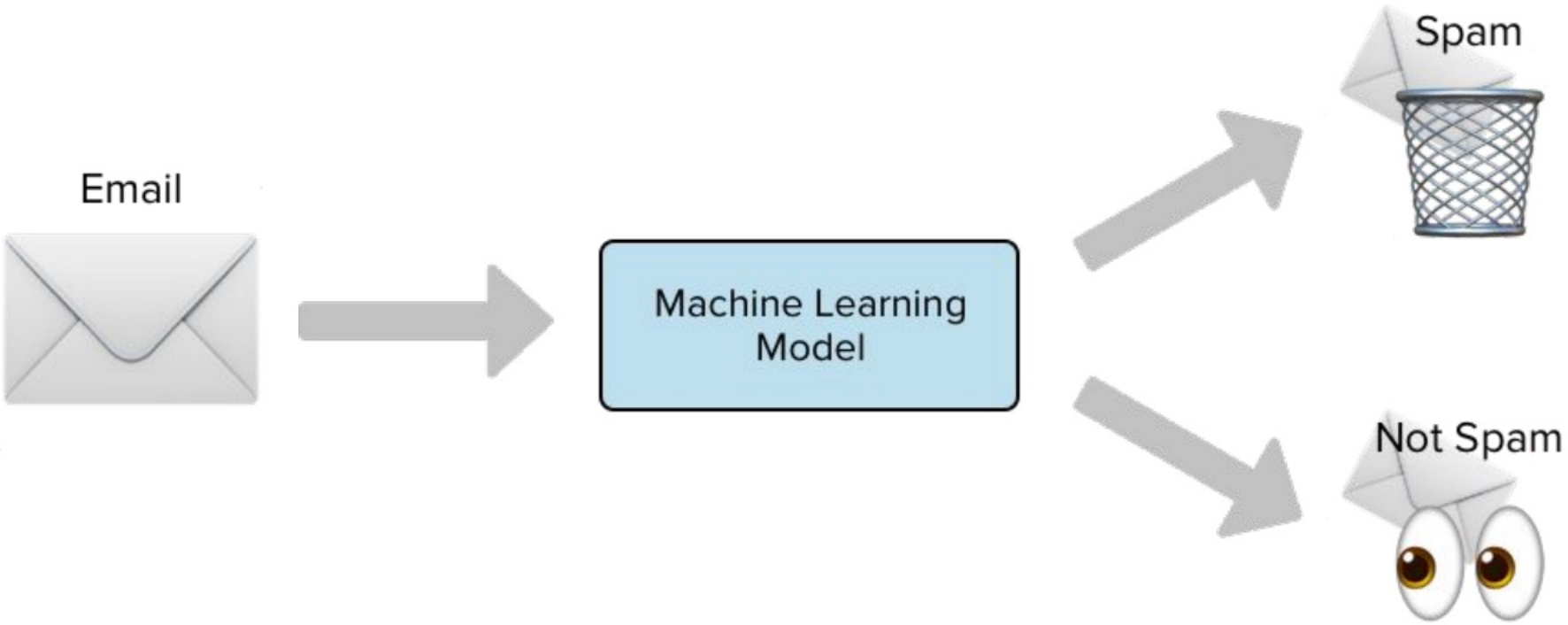
Esercizio: è un compito di regressione o di classificazione?



Esercizio: è un compito di regressione o di classificazione?

Distanza dal mercato più vicino	Presenza di un pozzo	Prezzo della casa (₦)
5 km (vicino al mercato Balogun, Lagos)	Sì	₦75, 000, 000
15 km (periferia di Abuja)	No	₦30, 000, 000

Esercizio: è un compito di regressione o di classificazione?



Come scegliere un approccio di apprendimento?



Dati di input:

- Possono comprendere input e, se noti, output target (comunemente detti etichette)
  un dataset non etichettato è un dataset di cui non conosciamo a priori gli output target

Output del modello:

- Categorico/numerico, discreto/continuo, singolo/multiplo

Approccio di apprendimento:

Apprendimento supervisionato

- **Guidato dal compito**
- **Dati etichettati**
- **Riscontro sulle prestazioni**

Apprendimento non supervisionato

- **Guidato dai dati**
- **Dati non etichettati**

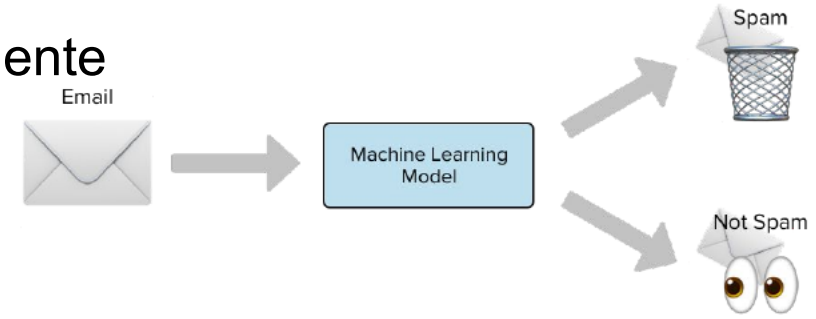
Apprendimento per rinforzo

- **Guidato dalla ricompensa**
- **Dati di stato/azione raccolti dinamicamente**
- **Riscontro sulle prestazioni**

Terminologia dell'apprendimento supervisionato

- Etichetta: è ciò che vogliamo prevedere. Spesso è indicata con y. Detta anche variabile dipendente o target

- **Feature: una variabile di input usata per il** compito. Spesso indicata con x. Detta anche variabile indipendente



- **Modello: codifica la relazione tra le feature** e l'etichetta

- **Addestramento: il processo con cui il modello impara a funzionare bene** sui dati; si dice anche apprendere il modello

- **Inferenza: applicare il modello addestrato a dati mai visti,** detti anche dati di test

Apprendimento supervisionato

Dato un dataset etichettato, cioè input X con etichette y:

$$\{(X^{(1)}, y^{(1)}), (X^{(2)}, y^{(2)}), \ldots, (X^{(n)}, y^{(n)})\}$$

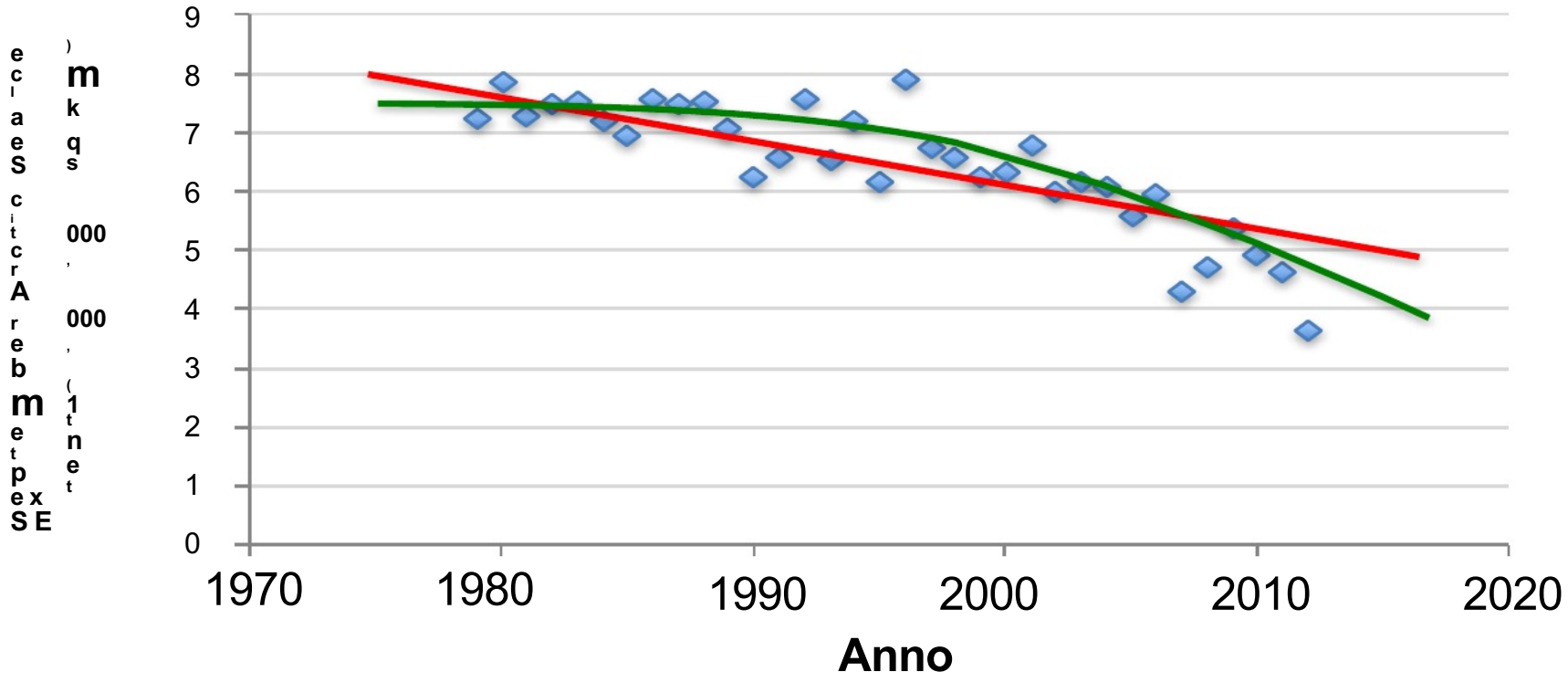
apprendere una funzione f che preveda y a partire da X, cioè $f(X) \cong y$.

- **X può avere più elementi, ciascuno dei quali rappresenta un attributo («feature»)**
- **Regressione: y è a valori reali**
- **Classificazione: y è categorica**

Apprendimento supervisionato: regressione

Apprendere una funzione f che preveda y a partire da X, cioè f(X) $\cong$ y .

- *Regressione: y è a valori reali*

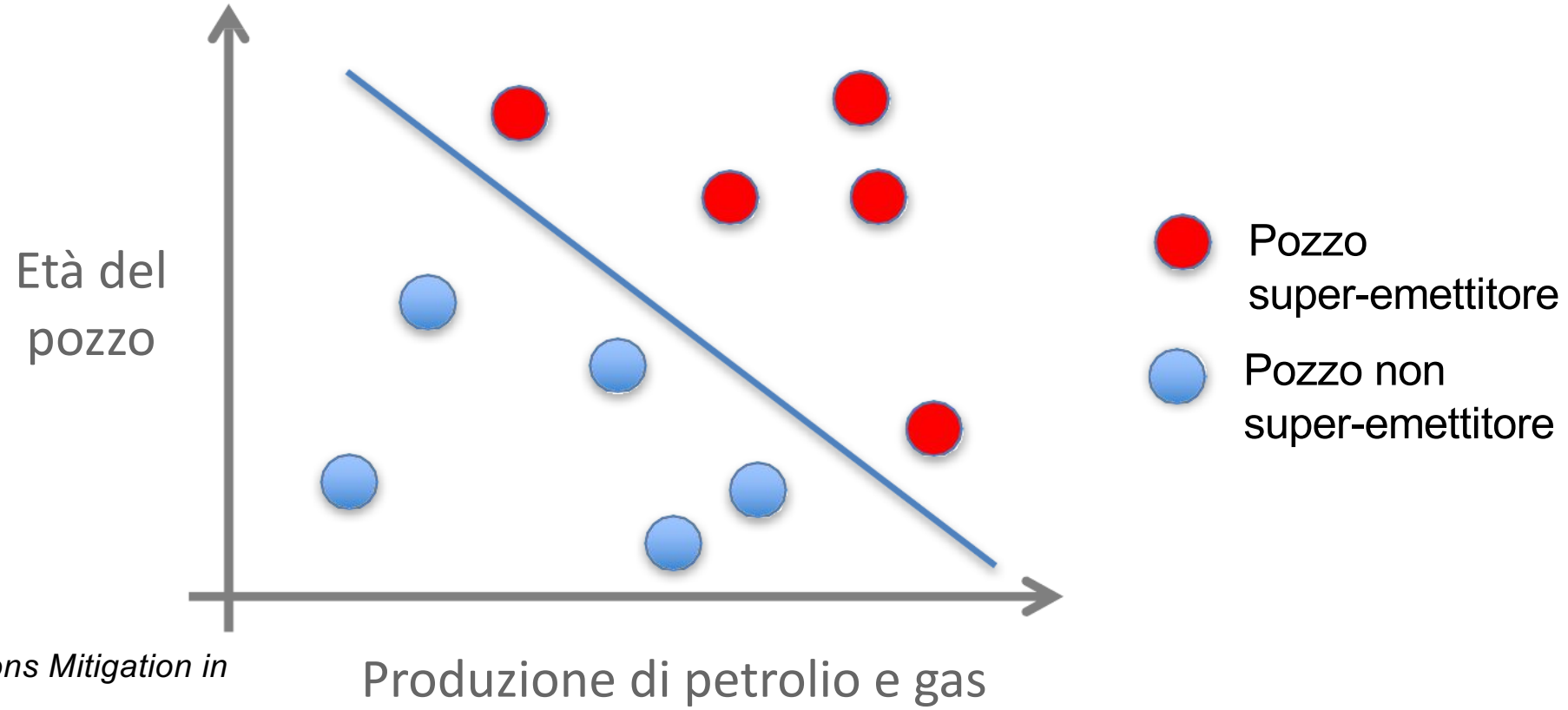


Dati di G. Witt. Journal of Statistics Education, volume 21, numero 1 (2013)

Apprendimento supervisionato: classificazione

Apprendere una funzione f che preveda y a partire da X, cioè f(X) $\cong$ y .

- *Classificazione: y è categorica*



A ML Approach to Methane Emissions Mitigation in the O&G Industry. Wang et al. 2020.

Apprendimento supervisionato: modelli/metodi comuni

- **Classificazione: macchine a vettori di supporto, random forest, regressione logistica, naive Bayes, reti neurali**

- Regressione: regressione lineare, regressione ridge, processi gaussiani, reti neurali

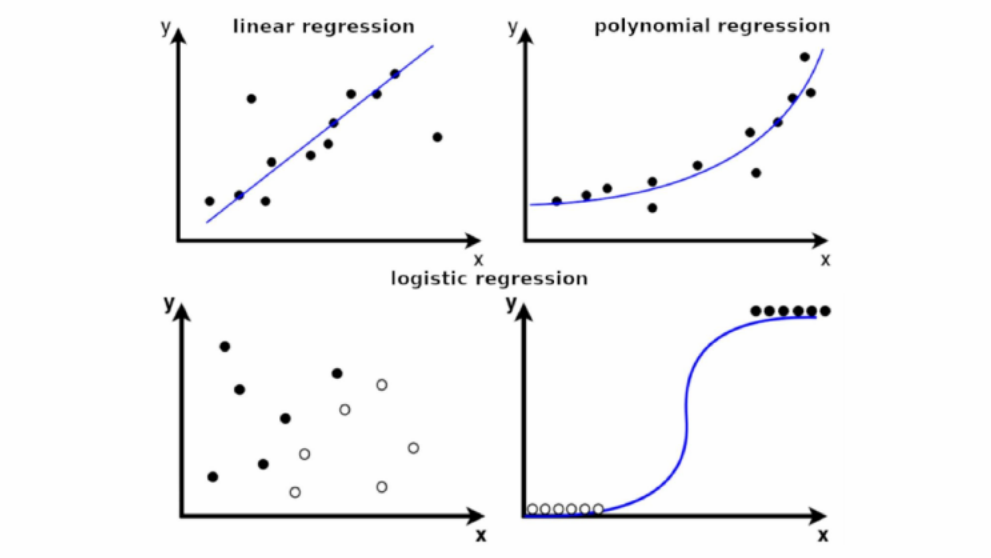


Immagine: Analytics Vidhya

Apprendimento supervisionato

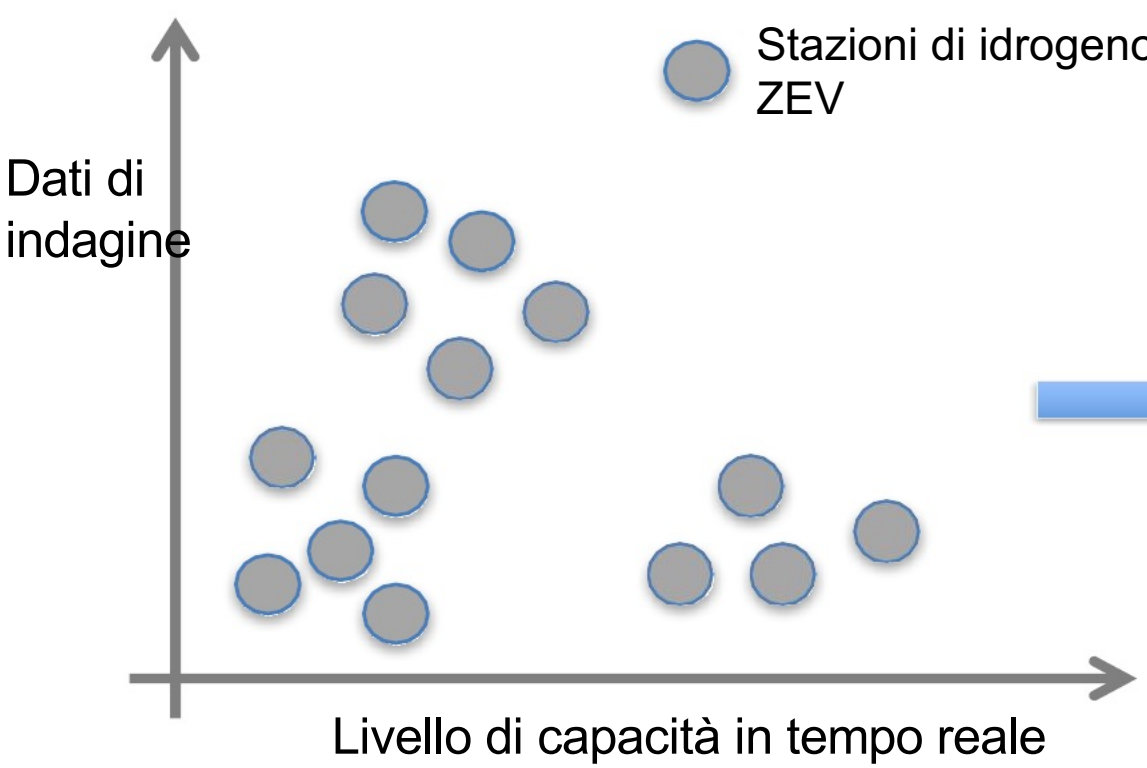
Criteri per scegliere un algoritmo:

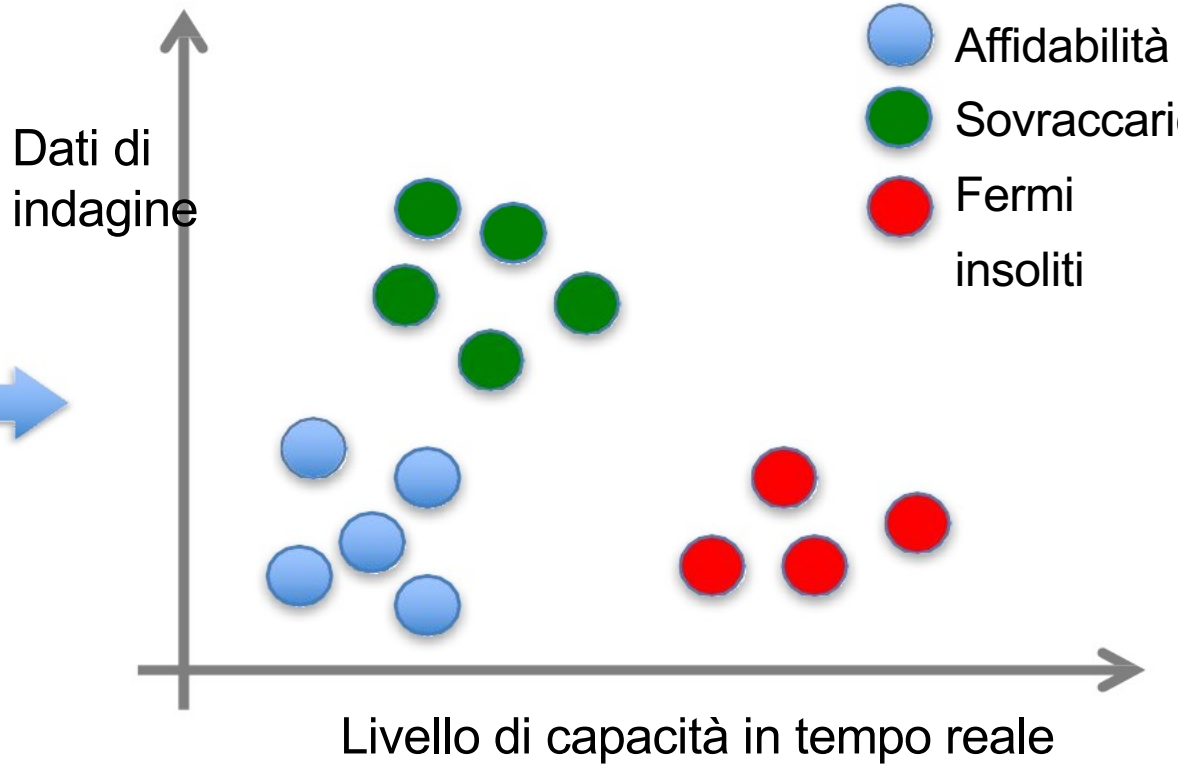
- Tipo di dati di input e di output
- Distribuzione e qualità dei dati, correlazioni tra feature
- Prestazioni desiderate e garanzie di robustezza

Esempio: classificare un pozzo come super-emettitore o no in base all'età e alla produzione di petrolio e gas, con etichette di super-emettitore disponibili per l'addestramento

Apprendimento non supervisionato

Dato un dataset non etichettato, cioè input X senza etichette, apprendere la struttura nascosta dei dati.





Unsupervised Temporal Clustering to Monitor the Performance of Alternative Fueling Infrastructure. Ramea 2020.

Apprendimento non supervisionato: modelli/metodi comuni

- **Apprendere schemi nei dati: analisi delle componenti principali (PCA), clustering, algoritmi di rilevamento delle anomalie**
- Modellazione generativa (generare dati sintetici): normalizing flow, autoencoder variazionali, reti generative avversarie
- Alcuni compiti di transfer learning: es. allineamento di dominio

Apprendimento non supervisionato

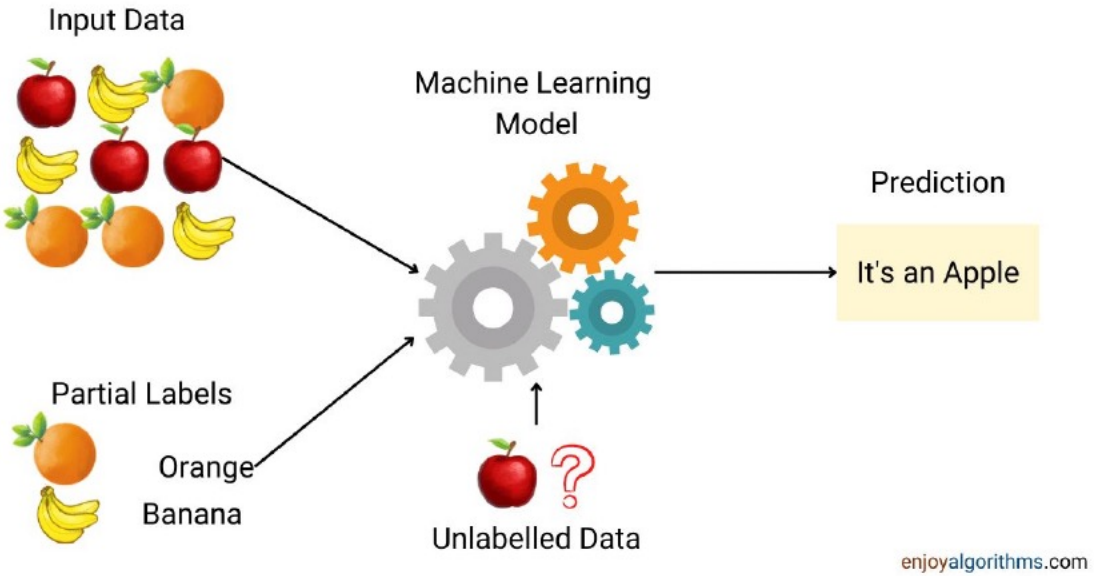
Criteri per scegliere un algoritmo/modello:

- Qual è l'obiettivo principale?
  - Clustering?
  - Generazione di dati sintetici?
  - Adattamento di dominio?
- Distribuzione e qualità dei dati, correlazioni tra feature

Esempio: generazione sintetica di profili di carico energetico degli edifici

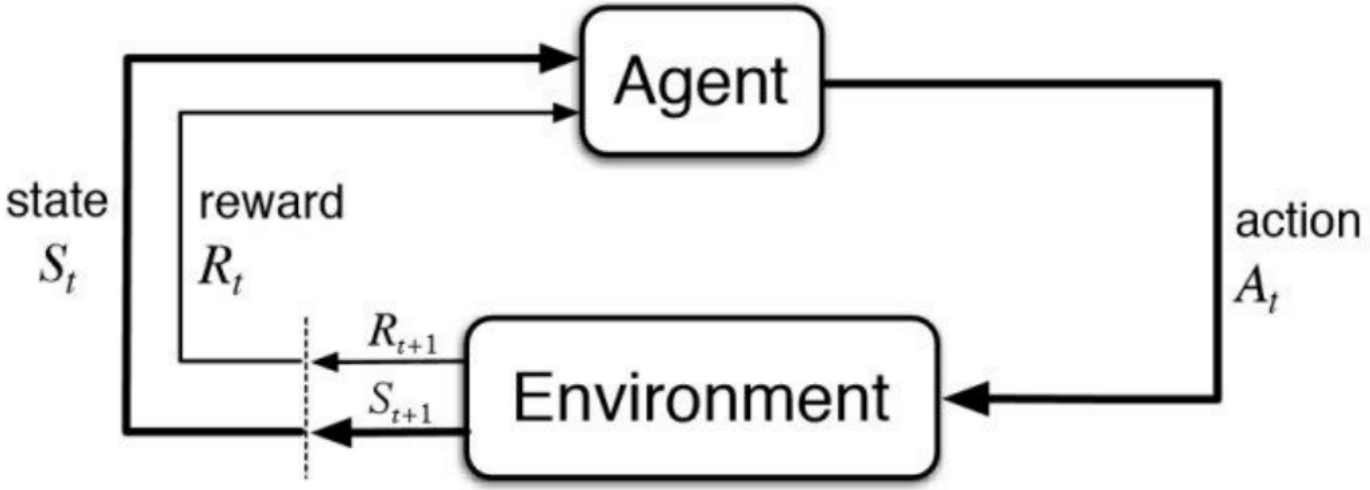
Apprendimento semi-supervisionato

L’apprendimento semi-supervisionato si colloca tra quello supervisionato e quello non supervisionato. Il dataset contiene punti etichettati e non etichettati



https://www.enjoyalgorithms.com/blogs/supervised-unsupervised-and-semisupervised-learning

Apprendimento per rinforzo



Obiettivo: trovare la migliore politica che massimizzi la ricompensa totale attesa

Il RL è spesso usato in contesti di controllo

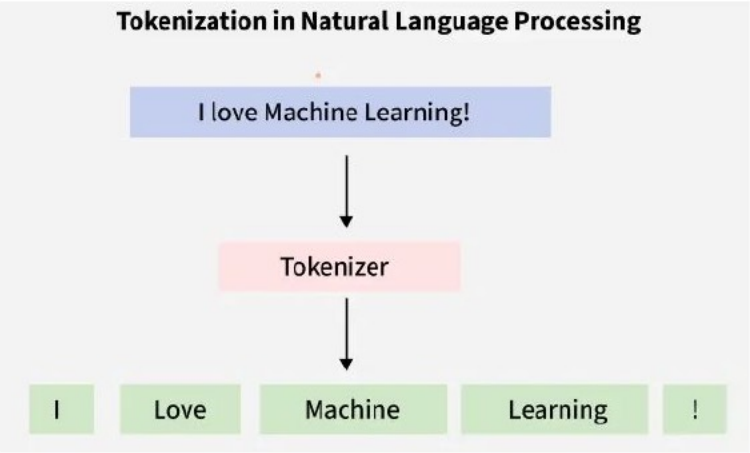
Esempi:

- Giochi
- Ottimizzazione energetica di un edificio
- Panoramica e zoom su immagini satellitari

Figura da «5 Things You Need to Know about Reinforcement Learning»

Alcuni approcci di ML associati alle modalità di dati

Elaborazione del linguaggio naturale (NLP)

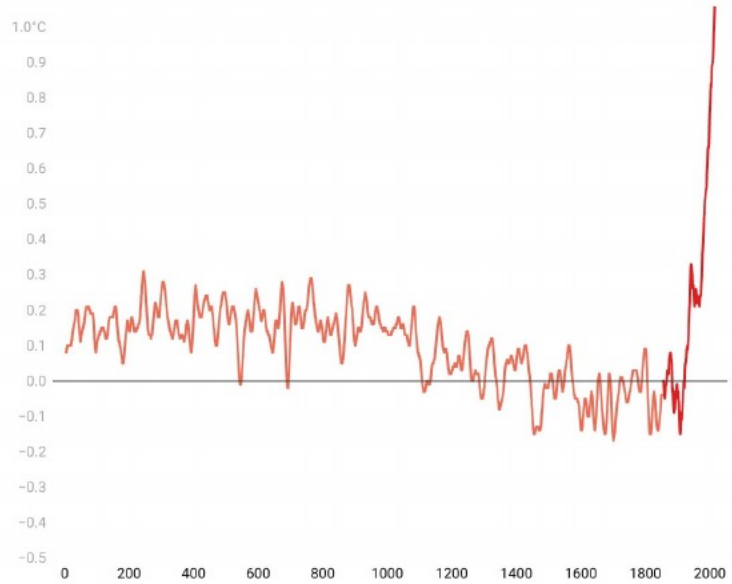


Es. reti neurali ricorrenti (RNN), modelli di Markov nascosti (HMM), grandi modelli linguistici (LLM)

Computer vision



Analisi delle serie temporali

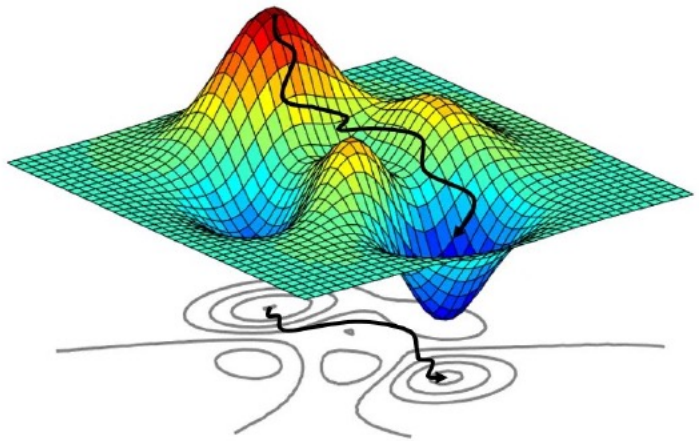


Immagini: Geeksforgeeks, Medium, TIME

Il ML per la modellazione surrogata

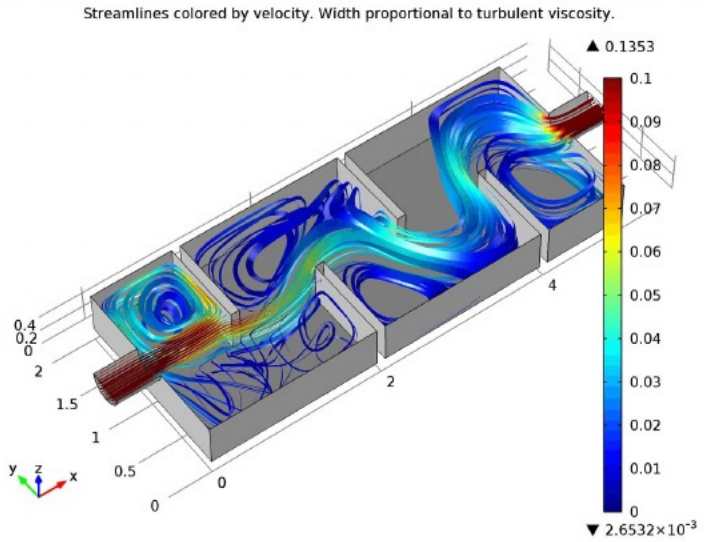
Modelli di ottimizzazione

Approssimare funzioni obiettivo e vincoli costosi per accelerare la ricerca di soluzioni ottimali

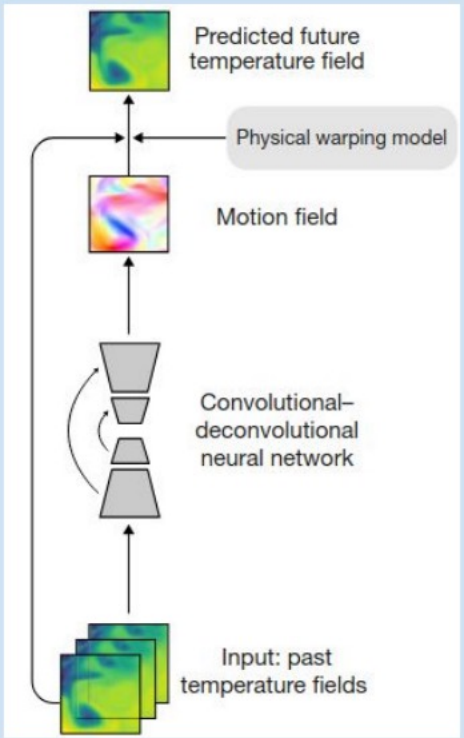


Modelli fisici

Emulare simulazioni fisiche complesse (fluidodinamica o meccanica strutturale)



Modelli informati dalla fisica



Immagini: Medium, Comsol, Reichstein et al. 2019

ML interpretabile e spiegabile

- L'IA spiegabile motiva le sue decisioni a posteriori, anche se il suo funzionamento interno è complesso od opaco.
- L'IA interpretabile ha una logica interna trasparente che le persone possono capire esaminando come funziona il modello stesso.

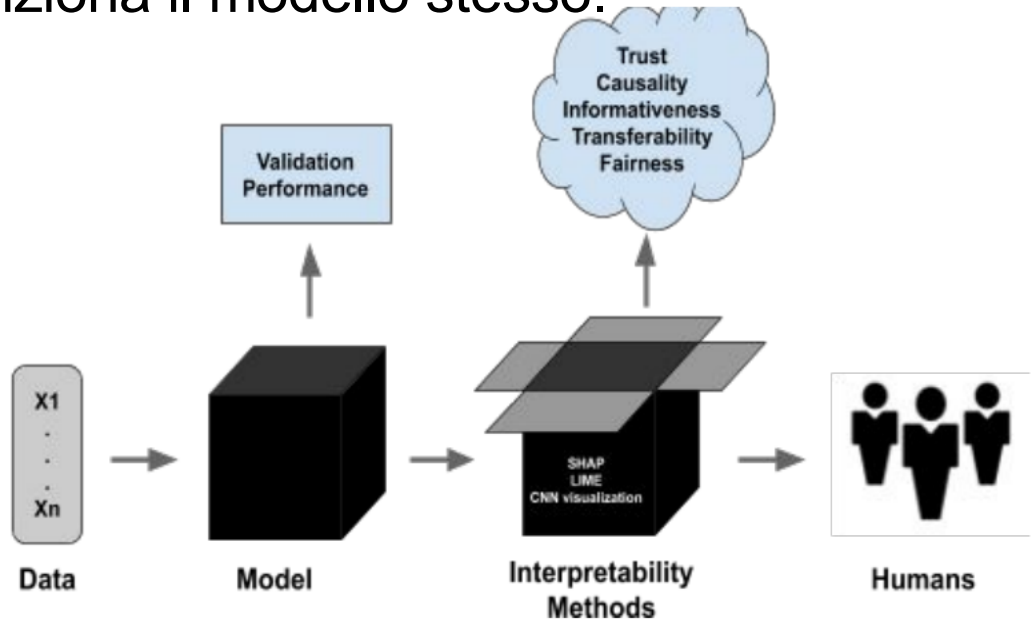
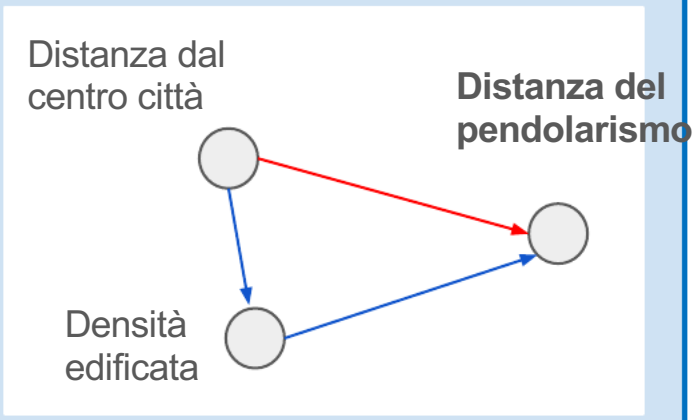


Immagine: ML CMU

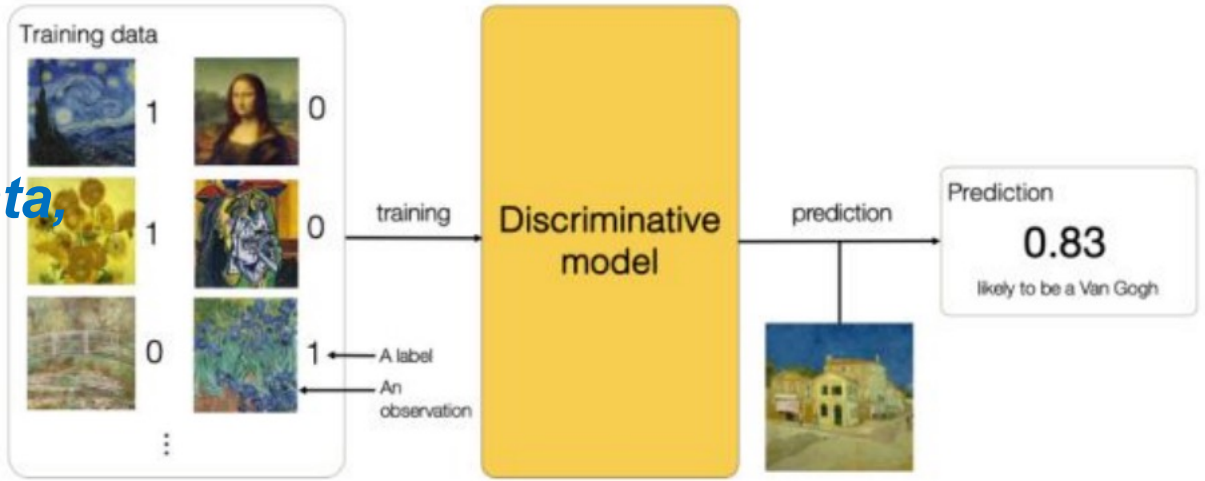
ML causale

- Individuare e quantificare relazioni di causa ed effetto (anziché correlazioni)
- Consente di prevedere gli esiti degli interventi e i controfattuali

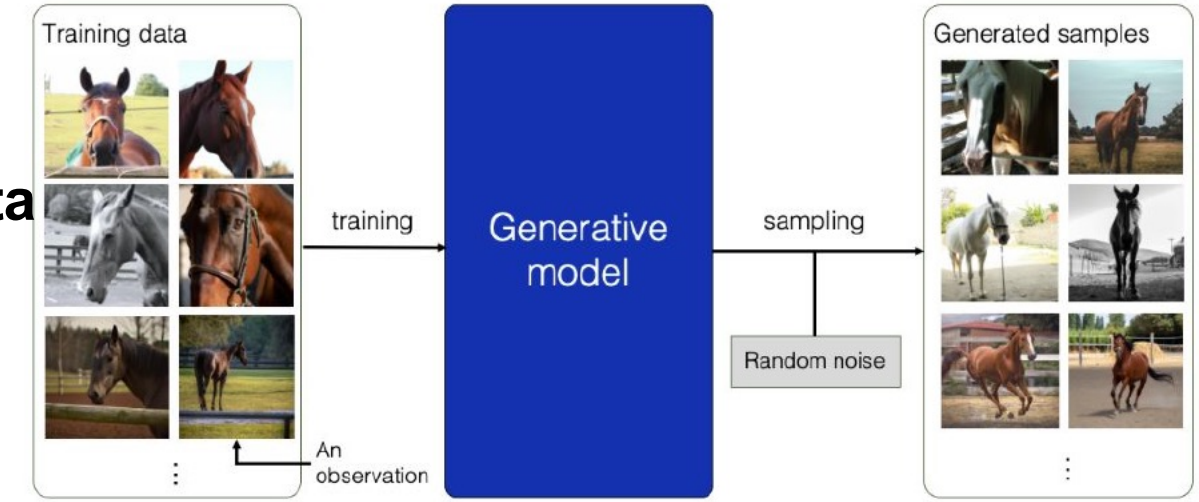


ML generativo e discriminativo

L'IA discriminativa modella la *distribuzione di probabilità condizionata,* imparando a passare direttamente dagli input agli output.



L'IA generativa («GenAI») modella esplicitamente la distribuzione congiunta dei dati, imparando come feature ed etichette si presentano insieme.

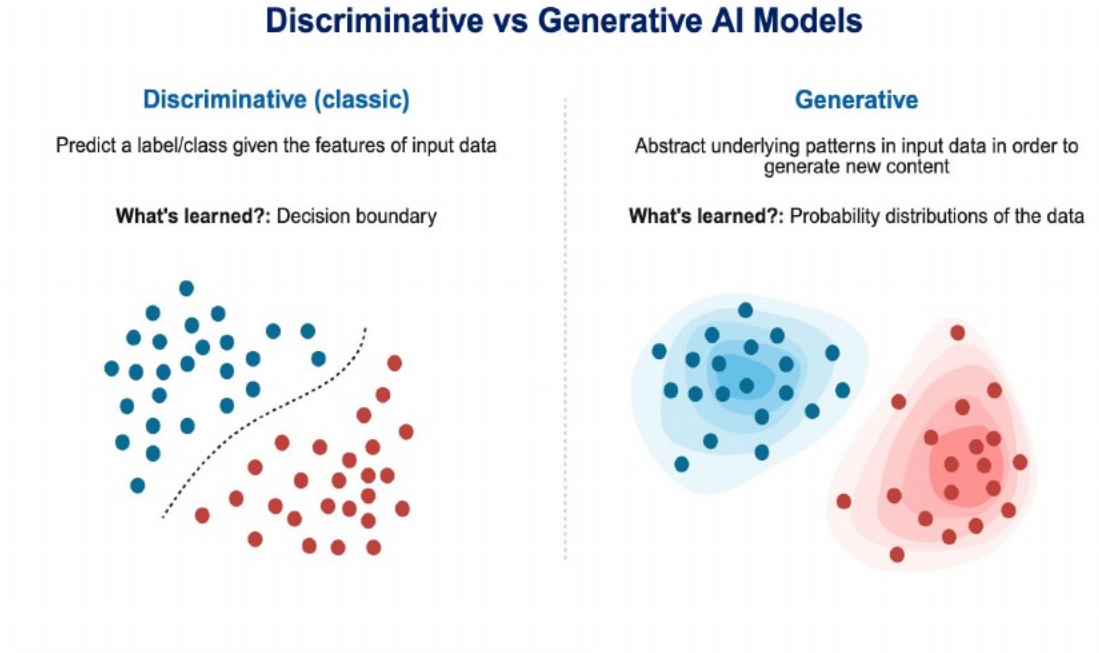


Immagini: Analytics yogi

Modelli discriminativi e generativi

Modelli discriminativi

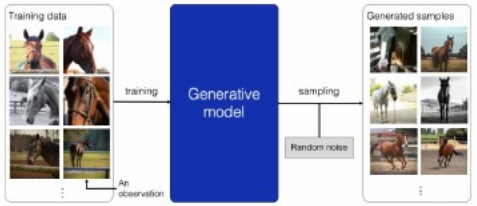
- Servono a separare («discriminare» tra) output diversi in base al valore dell'input.
  - Apprendono la distribuzione condizionata P(Y|X) dei target Y dati gli input X
- Es. prevedono un'etichetta o una categoria (spam o non spam)



Modelli generativi

- Apprendono la distribuzione congiunta P(X,Y) degli input X e del target Y
- Imparano a creare («generare») nuovi contenuti campionati dalla distribuzione appresa: testo, immagini, audio, codice





Esempio: Generative Pre-trained Transformer (GPT)

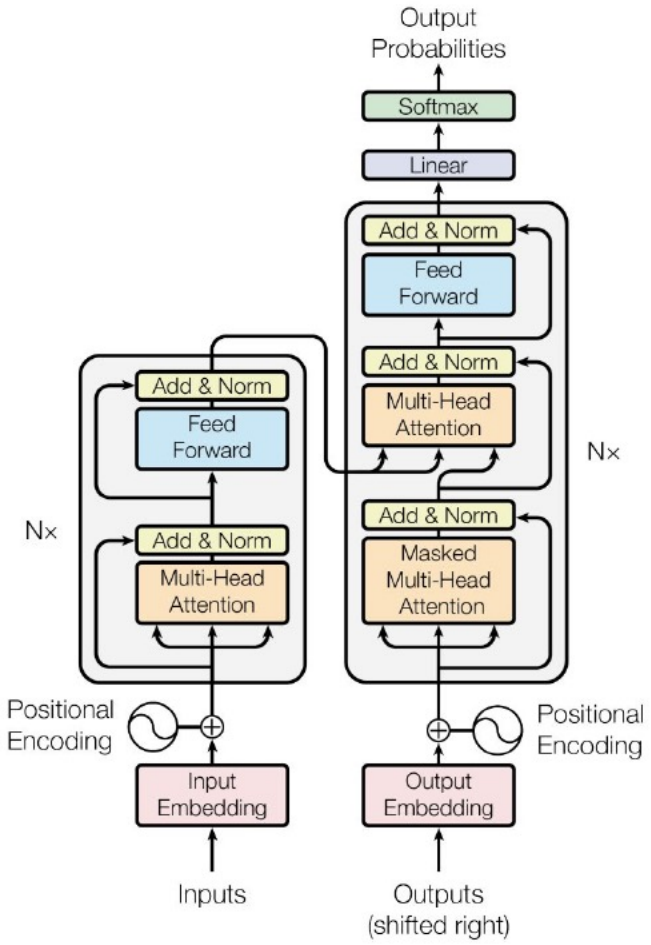
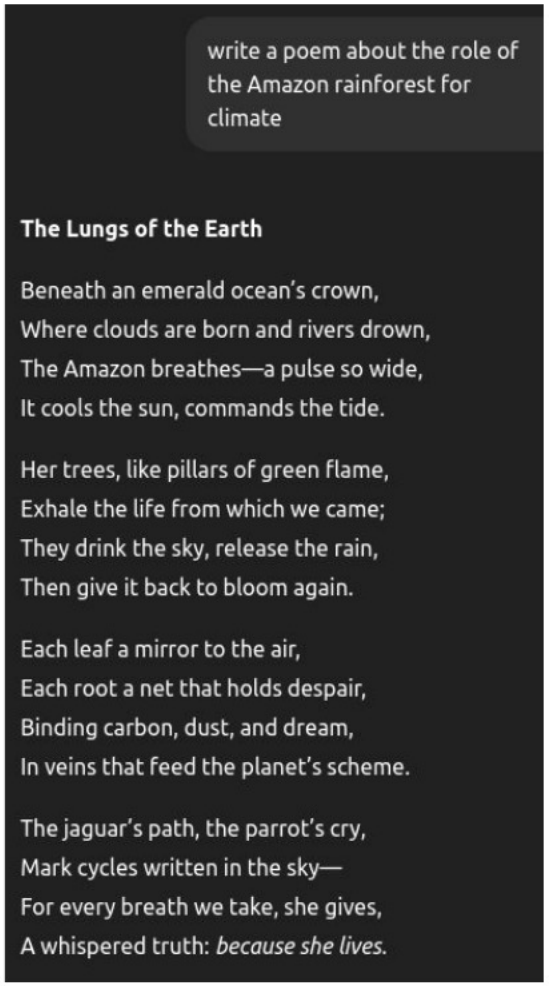


Figure 1: The Transformer - model architecture.



Fonte della figura: Vaswani et al. (2017)

Idea centrale: prevede il token successivo dai precedenti (non supervisionato)

Punti di forza: analizza schemi in dati diversi su larga scala, spesso con risultati notevoli

Limiti:

- Richiede enormi quantità di dati e calcolo
- Eredita bias e imprecisioni dai dati di addestramento
- Può avere allucinazioni senza stime di incertezza

Caratteristiche chiave:

- Interpola schemi
- Riconoscimento di schemi e non recupero diretto di fatti

Discussione: generazione di dati sintetici

Se un dataset è sintetico (generato artificialmente), lo considerereste «sicuro» da condividere pubblicamente? Perché?

https://partici.fi/53603205

Indice della lezione

Che cos’è il machine learning?

Tipi di machine learning

Dati

- Raccolta dei dati
- Preparazione dei dati per il ML

Capire l’algoritmo di regressione lineare

Domande e risposte

L’importanza dei dati

- I dati sono il cuore del machine learning
- La qualità del modello dipende
  dalla qualità dei dati
- Quanti dati bastano?
  - Dipende dal problema
  - Esistono alcune euristiche discrete, ma
    quasi sempre conta ciò che
    funziona empiricamente



Figura da «What is Big Data? A Quick Introduction for Analytics and Data Engineering Beginners»

Raccolta e preparazione dei dati

1. Definire il compito
2. Selezionare dati coerenti con il compito
    a. I dati sono etichettati? Se no, potete etichettarli?
3. Esplorare e pulire i dati
    a. Gestire i valori mancanti (escludere o imputare)
    b. Verificare la qualità delle etichette
4. Suddividere i dati in addestramento, validazione e test

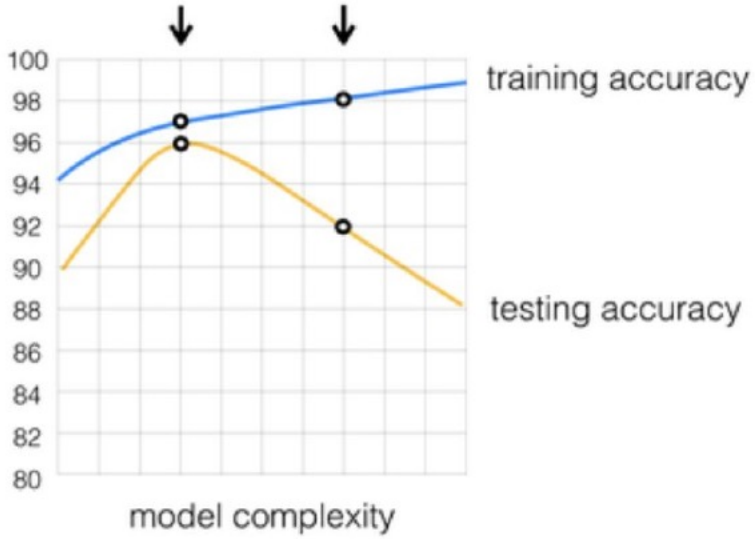
Suddividere il dataset

- Perché suddividere il dataset?
  - Per evitare l'overfitting e valutare le prestazioni del modello su dati nuovi



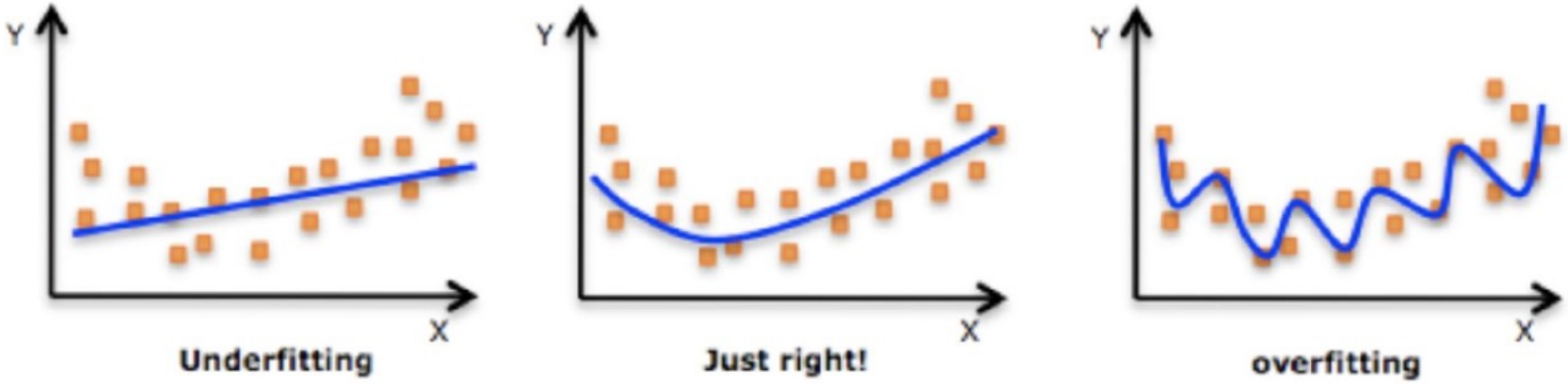
- Addestramento: apprendere i parametri del modello
- Validazione: scegliere gli iperparametri e confrontare i modelli
- Test: valutare il modello migliore per misurarne la generalizzazione su dati mai visti

Overfitting e generalizzazione



Soluzione: k-fold cross-validation

Valutare come si comporterà un modello predittivo su dati «mai visti» suddividendo ripetutamente il dataset in k sottoinsiemi di addestramento e test.



Definire il compito

1. Delineare i requisiti del problema da risolvere:
   - Individuare i vincoli temporali e spaziali del problema (che danno valore alla soluzione di ML)

     es. le previsioni meteo possono servire a ottimizzare la produzione rinnovabile, sia per ottimizzare infrastrutture esistenti (breve periodo) sia per scegliere dove collocare nuovi impianti (lungo periodo)
2. Stabilire le specifiche che il modello di ML deve rispettare:
   - Quali sono le specifiche per l'addestramento? E per l'implementazione?
3. Impostare il compito di ML, per esempio:

Classificazione | Regressione

Clustering | Segmentazione di immagini | Data augmentation

Rilevamento di anomalie | Ottimizzazione online

Selezione dei dati

- Selezionare dati coerenti con il compito di ML
  - Esaminando le applicazioni più avanzate
  - Consultando esperti del settore (ricercatori/professionisti del campo)
  - Valutando con attenzione i requisiti di dati del compito di ML

    Nota: potrebbe essere necessario rivedere/allentare alcune specifiche in base ai dati effettivamente disponibili.
- Il dataset è etichettato? Se no, potete etichettarlo? Vale la pena annotarlo per risolvere il problema, considerando approcci alternativi?

Esplorare e pulire i dati

1. Rimuovere osservazioni duplicate o irrilevanti
2. Correggere gli errori strutturali
3. Filtrare gli outlier indesiderati
4. Gestire i dati mancanti
5. Validare e garantire la qualità

Da «Data Cleaning: Definition, Benefits, And How-To» – Tableau

Discussione: proprietà dei dati

A chi dovrebbero appartenere i dati usati per addestrare i modelli di IA: all'utente, all'azienda o a qualcun altro?

https://partici.fi/53603205

Riepilogo finora

📝 Il machine learning è utile per determinati problemi.

- Non è una soluzione miracolosa.

📝 Esistono molti tipi diversi di machine learning.

- Approcci: supervisionato, non supervisionato, per rinforzo
- Modalità di dati: NLP, computer vision, serie temporali
- Requisiti: es. interpretabile e spiegabile, informato dalla fisica
- Altre scelte: generativo vs discriminativo, grande vs piccolo, generalista vs specifico

📝 I dati sono il cuore del machine learning!

Esercizio: sì o no

Dal punto di vista tecnico, l’intelligenza artificiale (IA) indica soprattutto i grandi modelli generativi generalisti come i grandi modelli linguistici.



https://partici.fi/53603205

Indice della lezione

Che cos’è il machine learning?

Tipi di machine learning

Dati

Capire l’algoritmo di regressione lineare

- Equazione di una retta
- Funzione di perdita
- Ottimizzare i parametri

Domande e risposte

Equazione di una retta

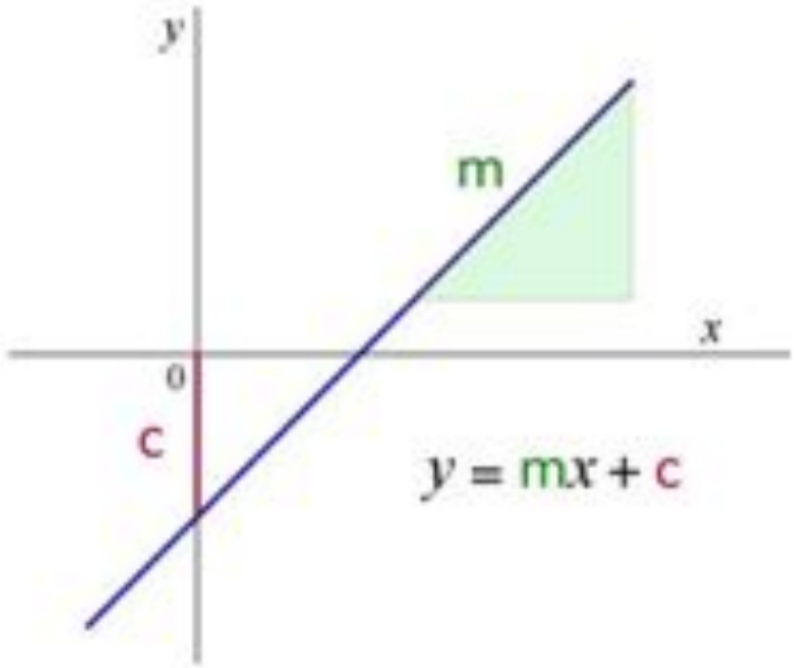
$$y = mx + c$$

Pendenza **Intercetta**

Equazione di una retta

$$y = mx + c$$

Pendenza **Intercetta**

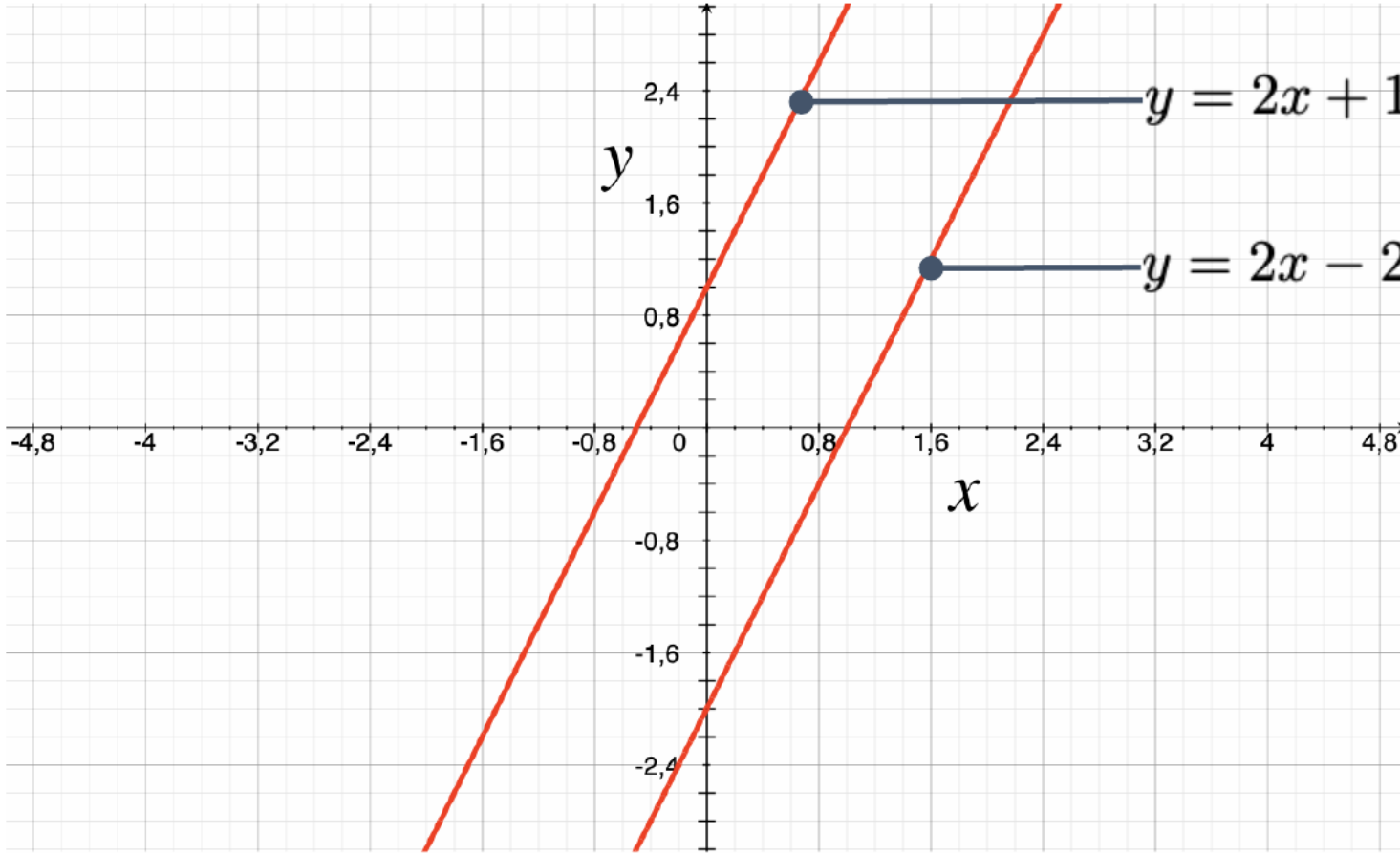


https://www.easycalculation.com/maths-dictionary/equation_of_a_straight_line.html

Equazione di una retta

$$y = mx + c$$

Pendenza **Intercetta**

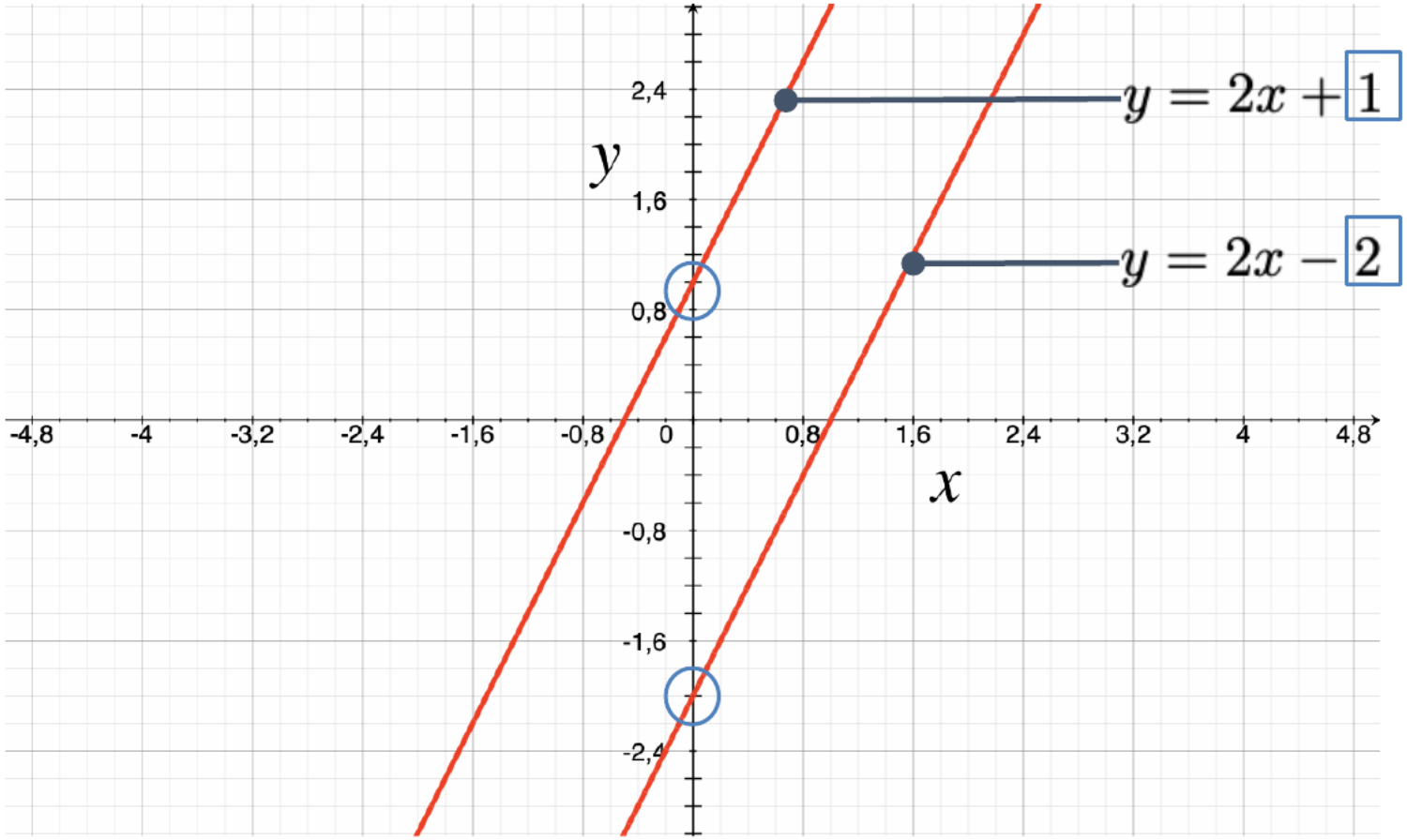


https://www.easycalculation.com/maths-dictionary/equation_of_a_straight_line.html

Equazione di una retta

$$y = mx + c$$

Intercetta

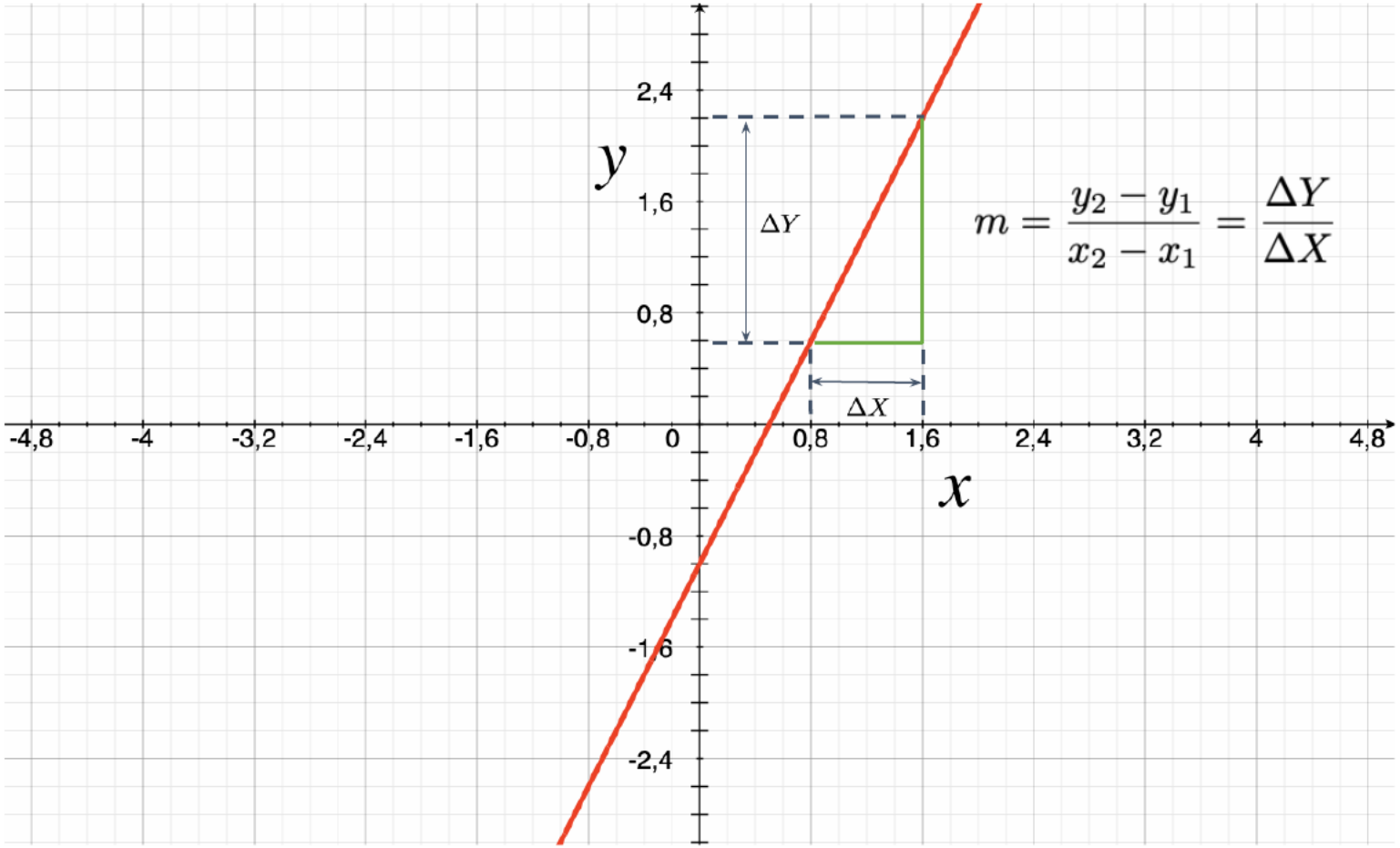


https://www.easycalculation.com/maths-dictionary/equation_of_a_straight_line.html

Equazione di una retta

$$y = mx + c$$

Pendenza

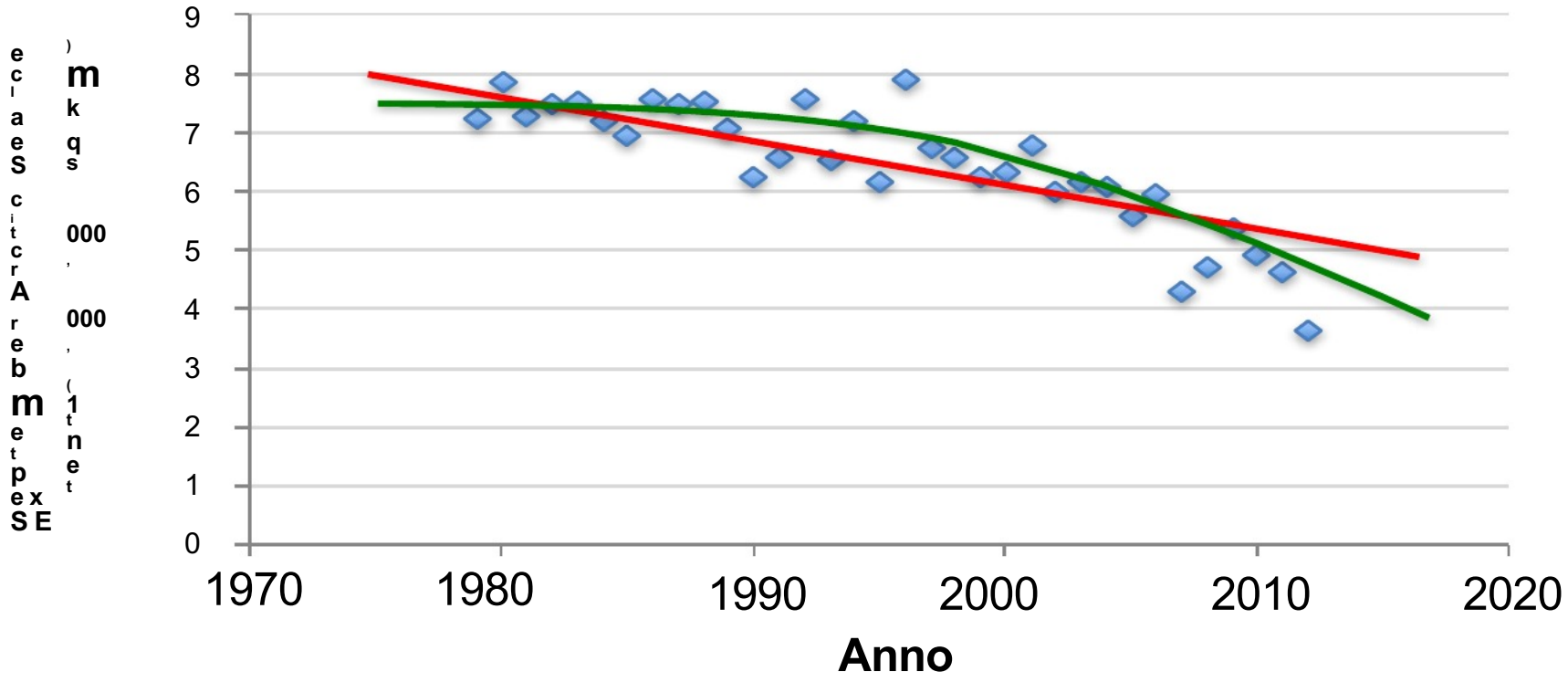


https://www.easycalculation.com/maths-dictionary/equation_of_a_straight_line.html

Prevedere l'estensione del ghiaccio marino artico

Apprendere una funzione f che preveda y a partire da X, cioè f(X) $\cong$ y .

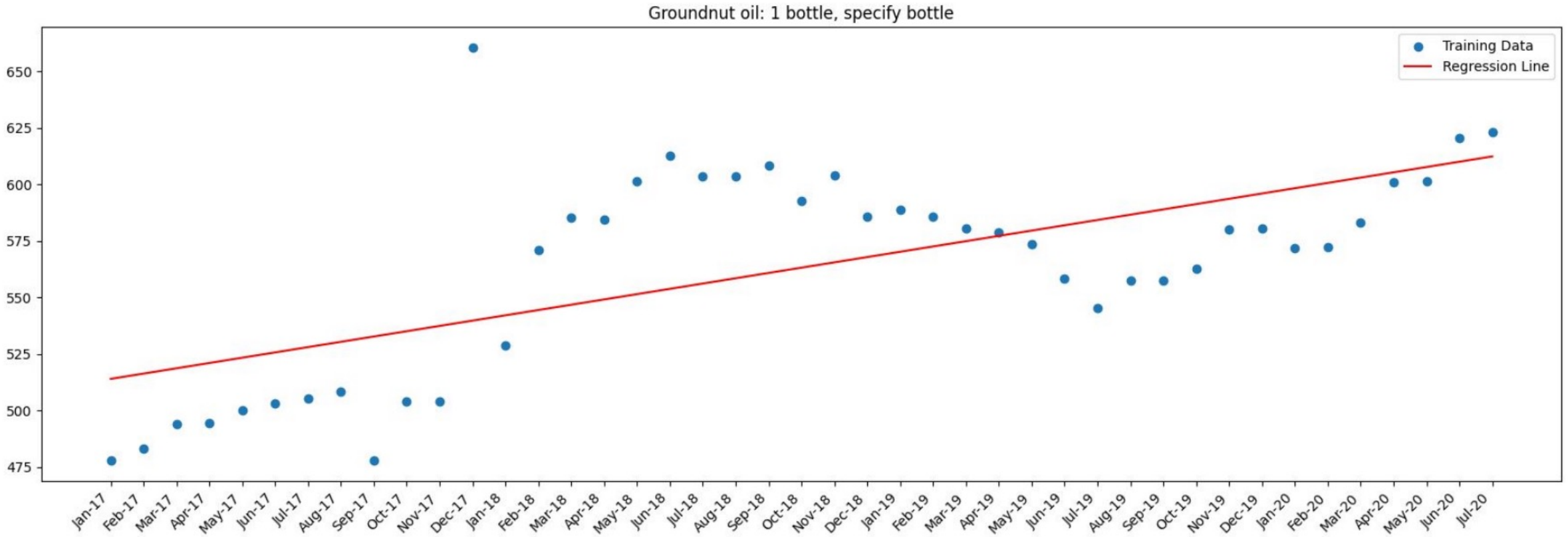
- *Regressione: y è a valori reali*



Dati di G. Witt. Journal of Statistics Education, volume 21, numero 1 (2013)

Prevedere il prezzo dell’olio di arachidi

eba



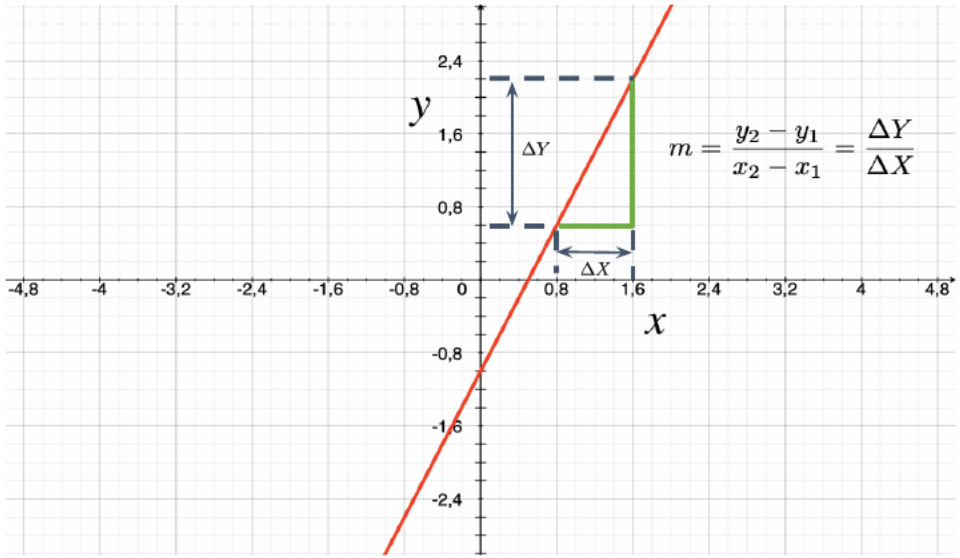
feature

Notebook di esercitazione: https://colab.research.google.com/drive/1UQQ_JrBYSugUZtLSzYOIad5rUNZKsw8K?usp=sharing

Regressione lineare

La regressione lineare presuppone una relazione lineare tra le feature e l'etichetta numerica.

$$y = mx + c$$



Nel ML di solito scriviamo l'equazione così:

$$y = b + w_1 x_1$$

dove:

y è l'etichetta

b è il bias (intercetta), talvolta scritto come w_0

w_1 è il peso della feature x_1

Modello di regressione lineare

Il modello di regressione lineare prevede un output $\hat{y}$ in base alle feature di input $x_1, x_2, \cdots, x_n$

$$\hat{y} = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_n x_n$$

dove:

- $\hat{y}$ è la previsione
- w_0 è l'intercetta (bias)
- $w_1, w_2, \cdots, w_n$ sono i pesi (coefficienti)
- $x_1, x_2, \cdots, x_n$ sono le feature di input

Modello di regressione lineare: funzione di perdita

Per misurare la differenza tra le previsioni e i valori reali $\hat{y}$ y (etichette)

usiamo l'errore quadratico medio (MSE)

$$\mathcal{L}(w) = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2$$

dove

m è il numero di esempi di addestramento

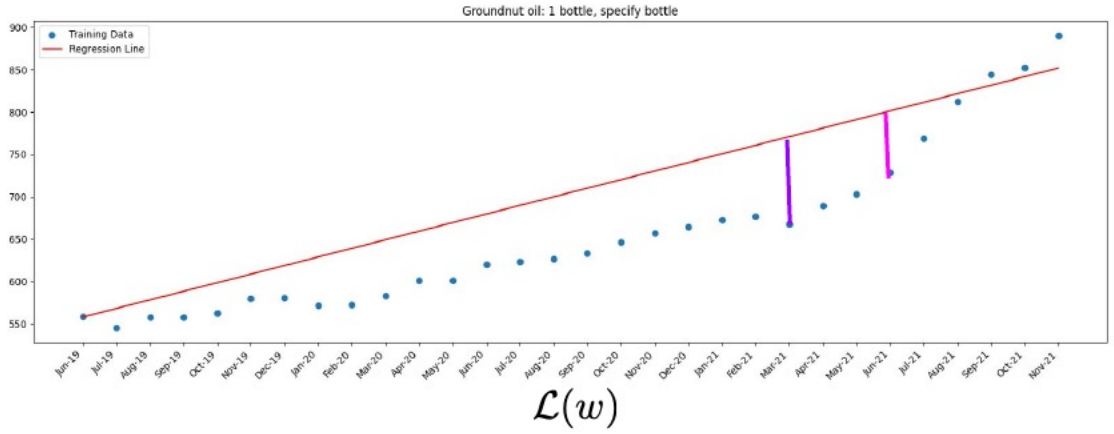
$\mathcal{L}(w)$ è la funzione di perdita

Modello di regressione lineare: funzione di perdita

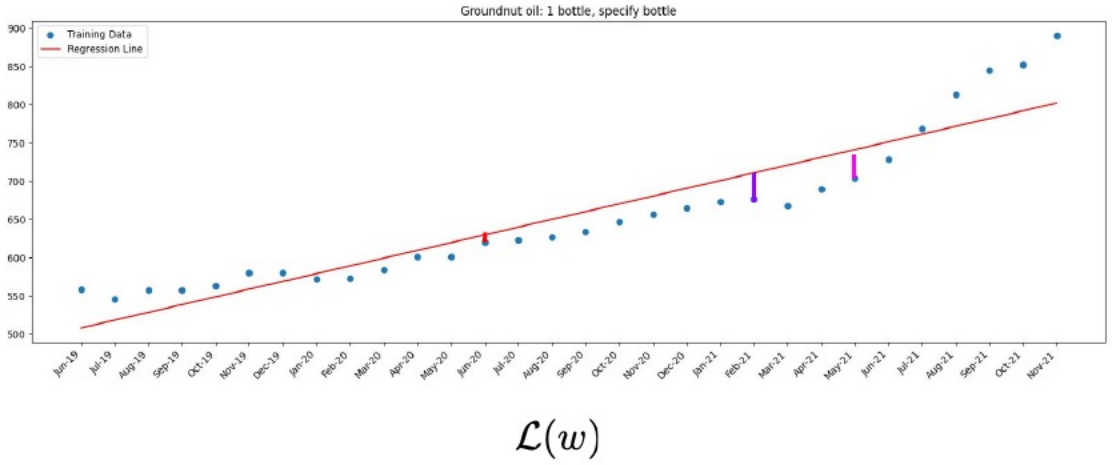
- La perdita è una penalità per una cattiva previsione: è un numero che indica quanto la previsione del modello sia sbagliata su un singolo dato. Se la previsione è perfetta, la perdita è zero, altrimenti è diversa da zero

$$\mathcal{L}(w) = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2$$

- Addestrare un modello di regressione lineare significa apprendere, da esempi etichettati, valori dei pesi che riducono la perdita



$\mathcal{L}(w)$



$\mathcal{L}(w)$

Illustrazione interattiva: https://developers.google.com/machine-learning/crash-course/linear-regression/parameters-exercise

Modello di regressione lineare: peso ottimale (minimizzare la perdita)

$$\mathcal{L}(w) = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2$$

$$\hat{y} = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_n x_n$$

sostituire

$$\mathcal{L}(w) = \frac{1}{m}\sum_{i=1}^{m}(y_i - (w_0 + w_1 x_{i1} + w_2 x_{i2} + \cdots + w_n x_{in}))^2$$

L'obiettivo è trovare $w_0, w_1, w_2, \cdots, w_n$ che minimizzino $\mathcal{L}(w)$

Modello di regressione lineare: peso ottimale (minimizzare la perdita)

Possiamo ottenerlo in due modi:

- Soluzione in forma chiusa (equazione normale)

- Ottimizzazione con discesa del gradiente

Ottimizzare i parametri: equazione normale

Se X è la matrice di progetto (con il termine di bias) e y è il vettore target, i pesi ottimali sono dati da

$$w = (X^T X)^{-1} X^T y$$

Funziona solo quando $X^T X$ è invertibile e per piccoli dataset

Ottimizzare i parametri: discesa del gradiente

I pesi vengono aggiornati iterativamente usando il gradiente della funzione di perdita

$$w_j := w_j - \alpha \frac{\partial \mathcal{L}}{\partial w_j}$$

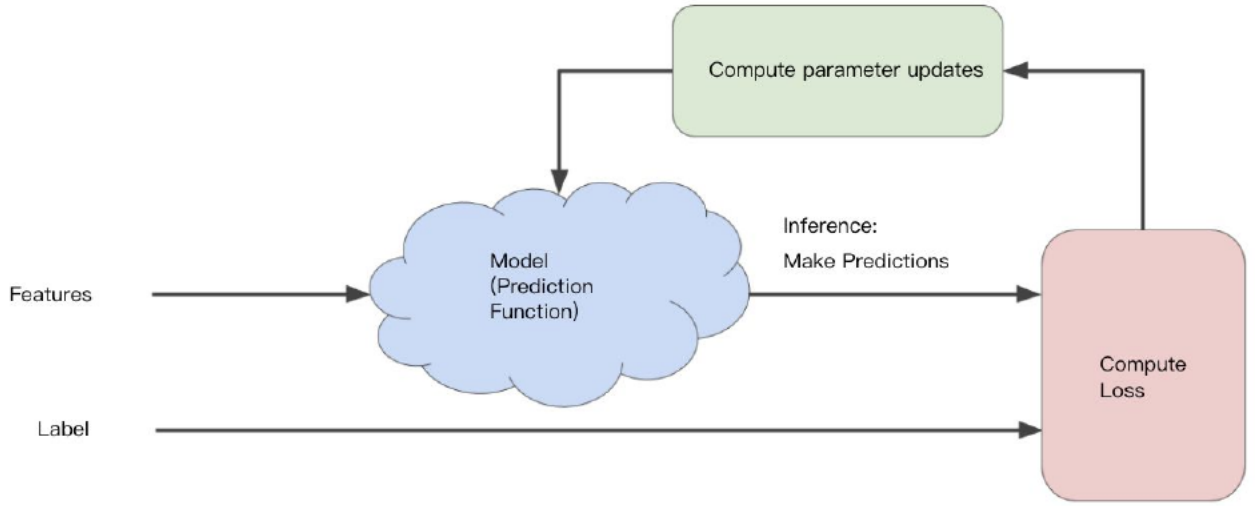
dove è α il tasso di apprendimento e

$$\frac{\partial \mathcal{L}}{\partial w_j} = -\frac{2}{m} \sum_{i=1}^{m} x_{ij}(y_i - \hat{y}_i)$$

Questo metodo si usa per grandi dataset

Trovare il minimo di una funzione

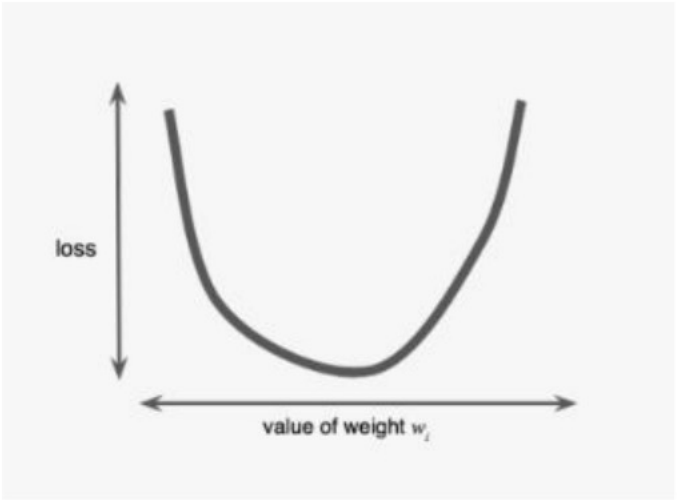
Iniziamo l'addestramento con una stima iniziale di pesi e bias e regoliamo iterativamente i valori fino a ottenere pesi e bias con la perdita più bassa

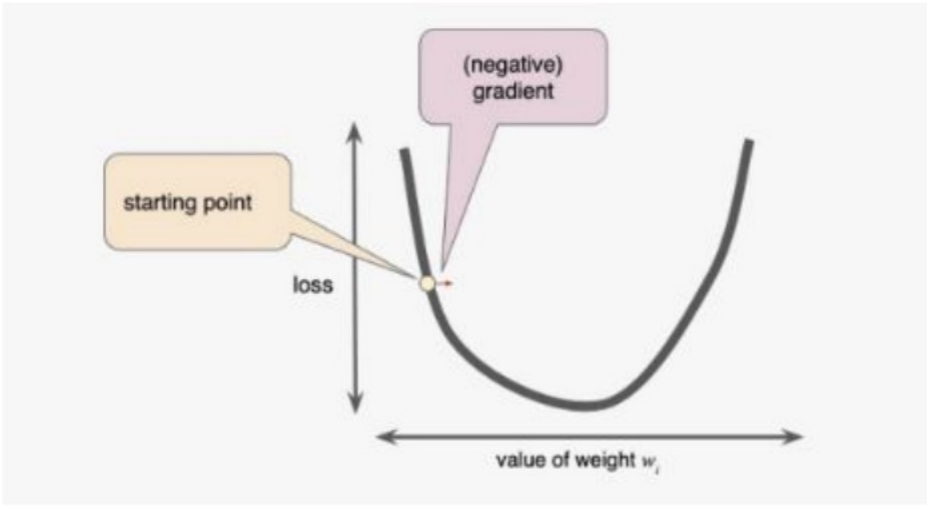


Alla maniera della discesa del gradiente!

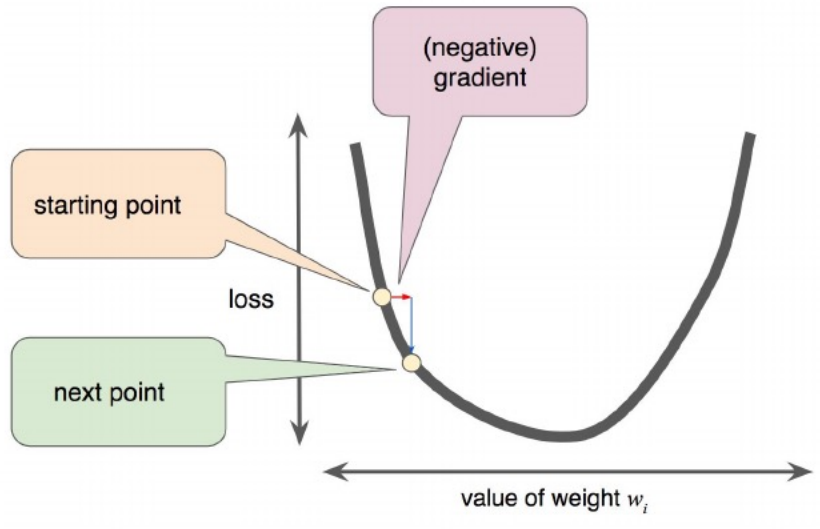
https://github.com/litaotao/machine-learning-crash-course

Trovare il minimo di una funzione





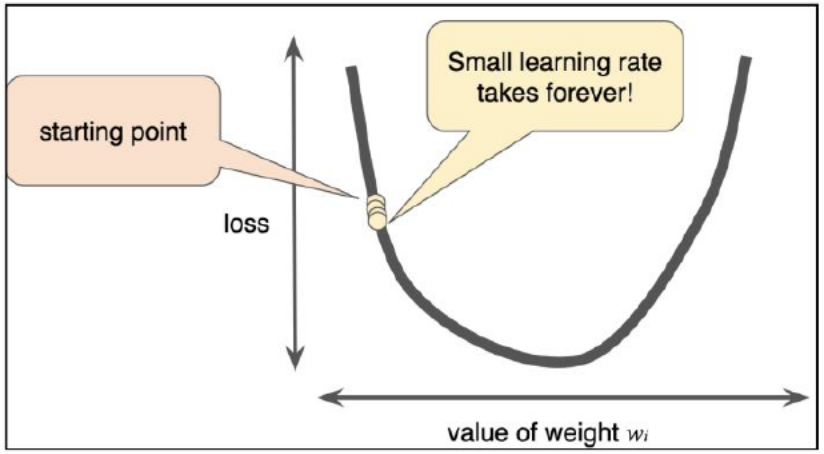
Alla maniera della discesa del gradiente!

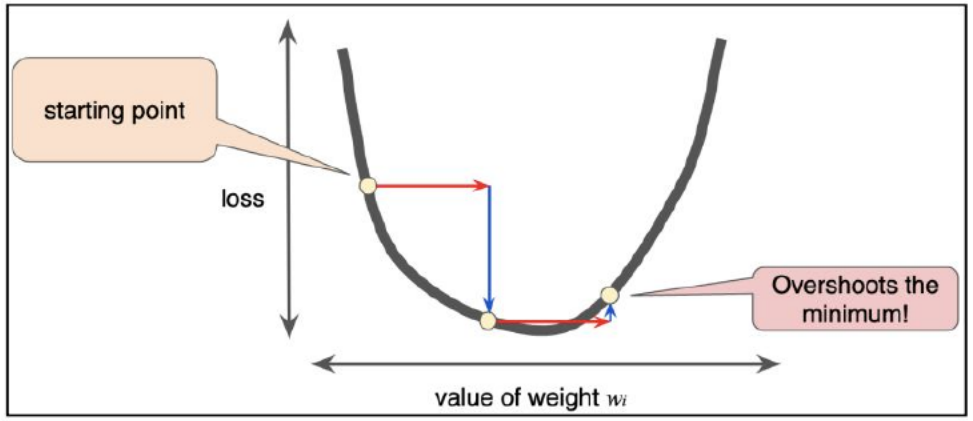


$$w_j := w_j - \alpha \frac{\partial \mathcal{L}}{\partial w_j}$$

Il tasso di apprendimento conta

Il tasso di apprendimento è un iperparametro che determina l'ampiezza dei passi a ogni iterazione quando si aggiornano i parametri del modello durante l'addestramento

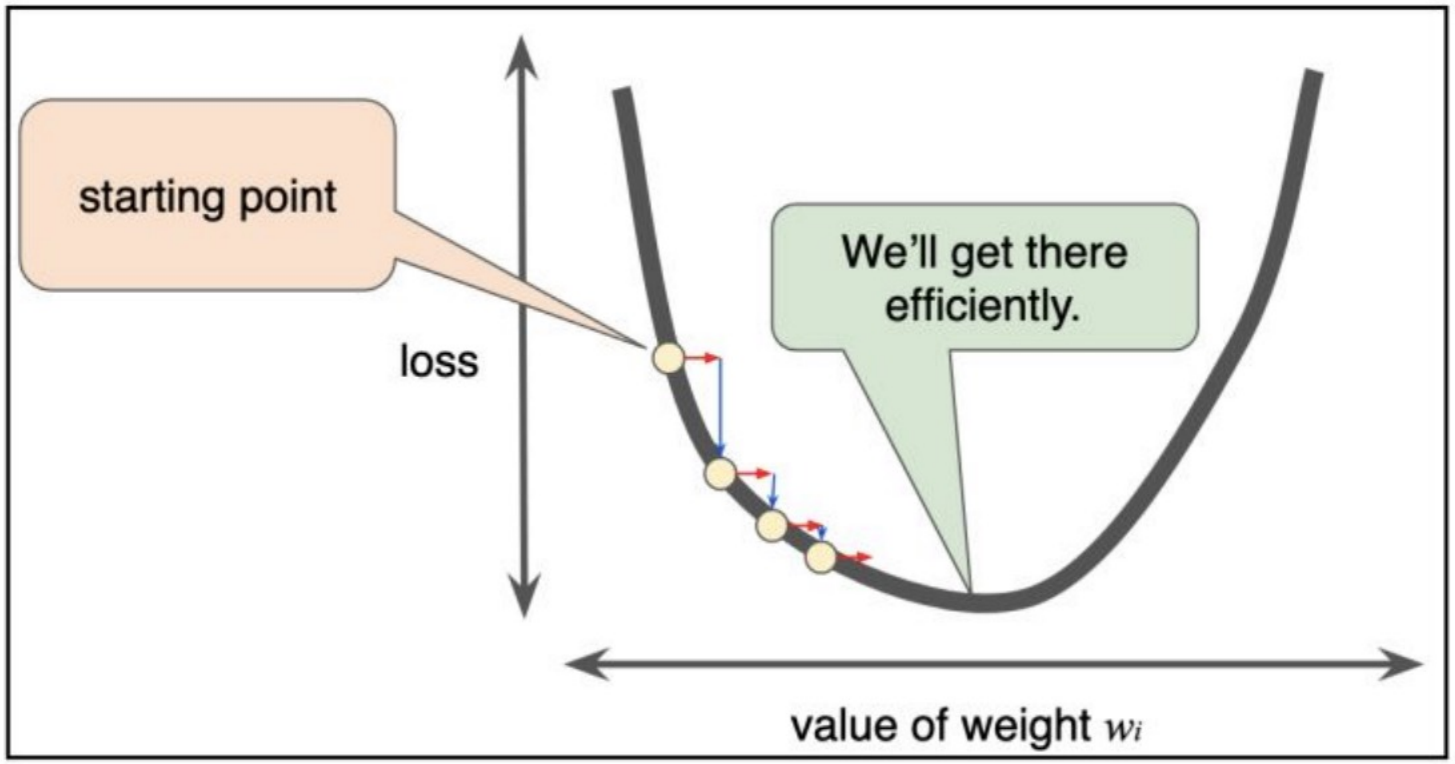




$$w_j := w_j - \alpha \frac{\partial \mathcal{L}}{\partial w_j}$$

https://cedar.buffalo.edu/~srihari/CSE676/8.3%20BasicOptimizn.pdf

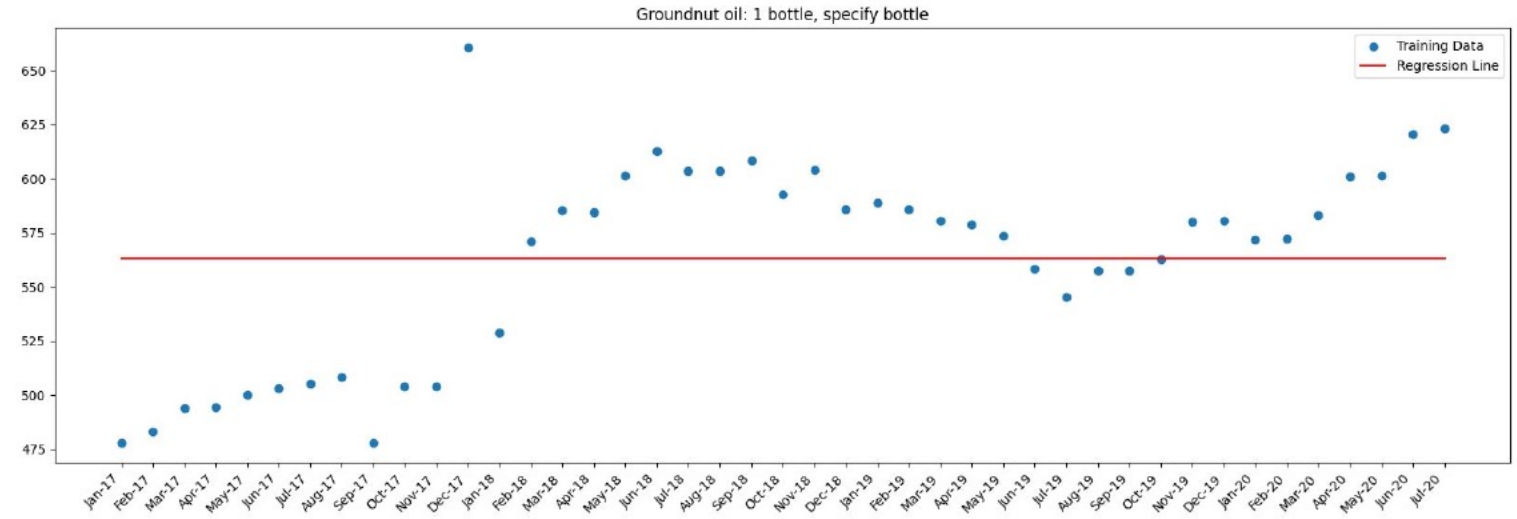
Il tasso di apprendimento conta



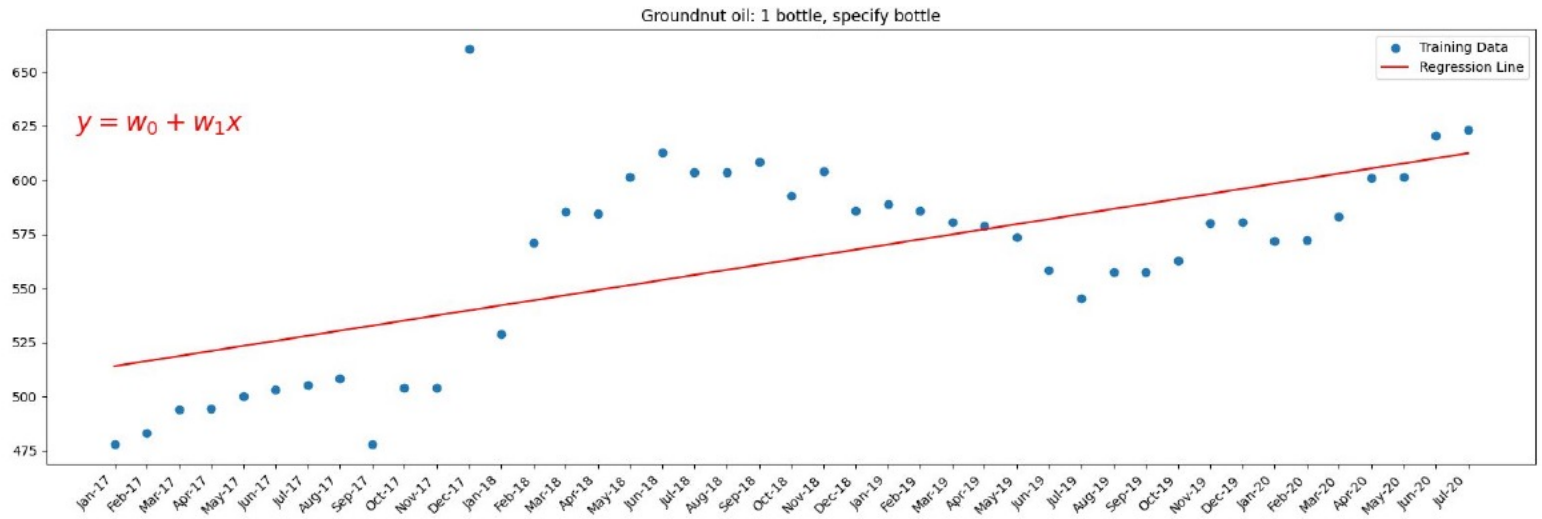
https://cedar.buffalo.edu/~srihari/CSE676/8.3%20BasicOptimizn.pdf

Adattare una retta ai dati sull'olio di arachidi

Quale modello si adatta meglio?

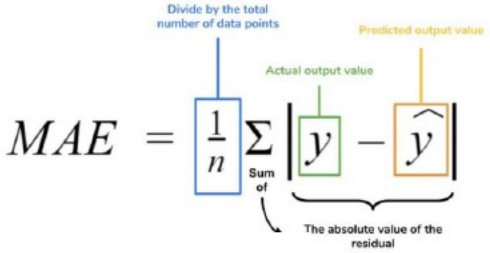


(A)

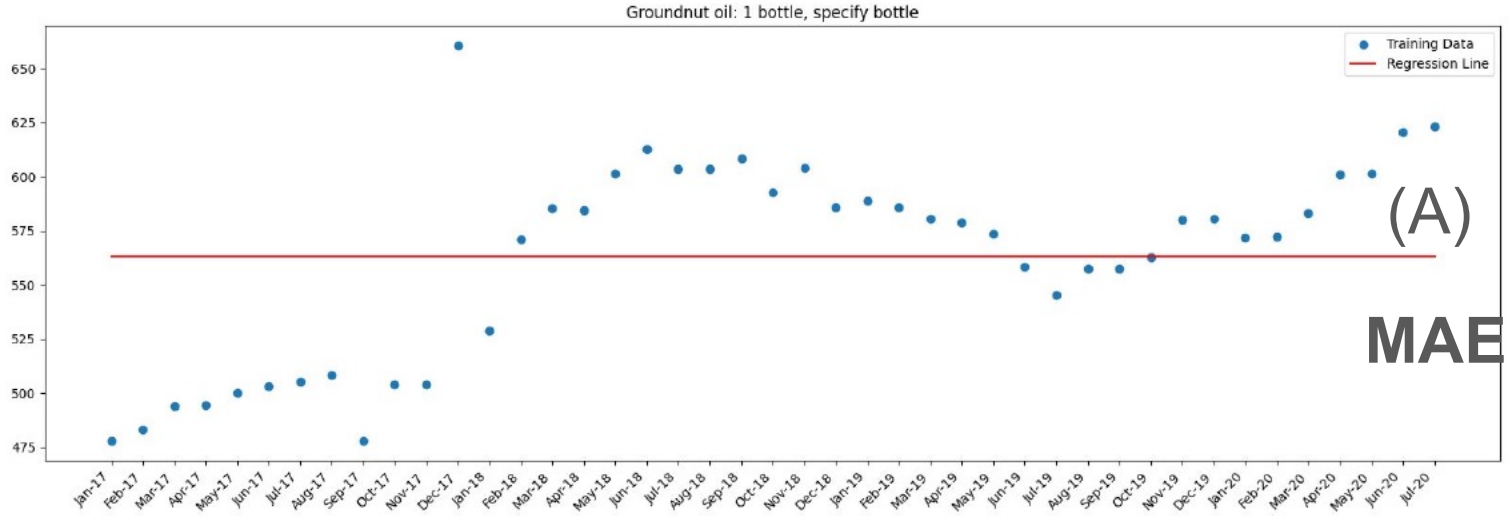


(B)

Valutare il modello: MAE

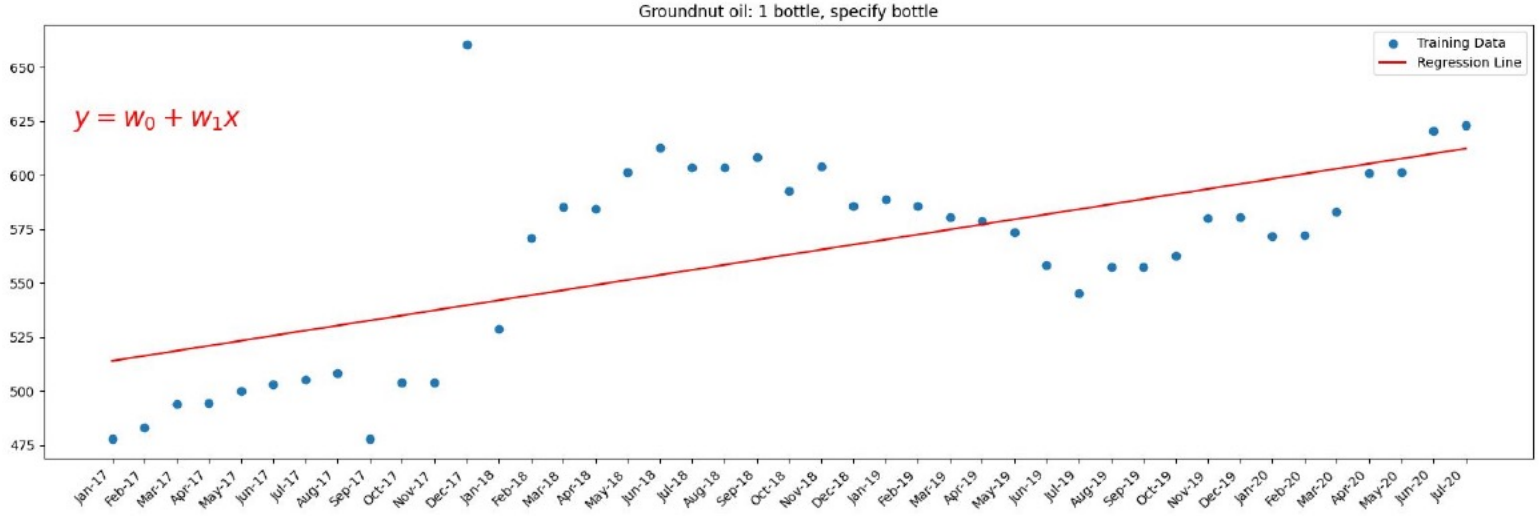


Quale modello si adatta meglio?



(A)

MAE = 47,20

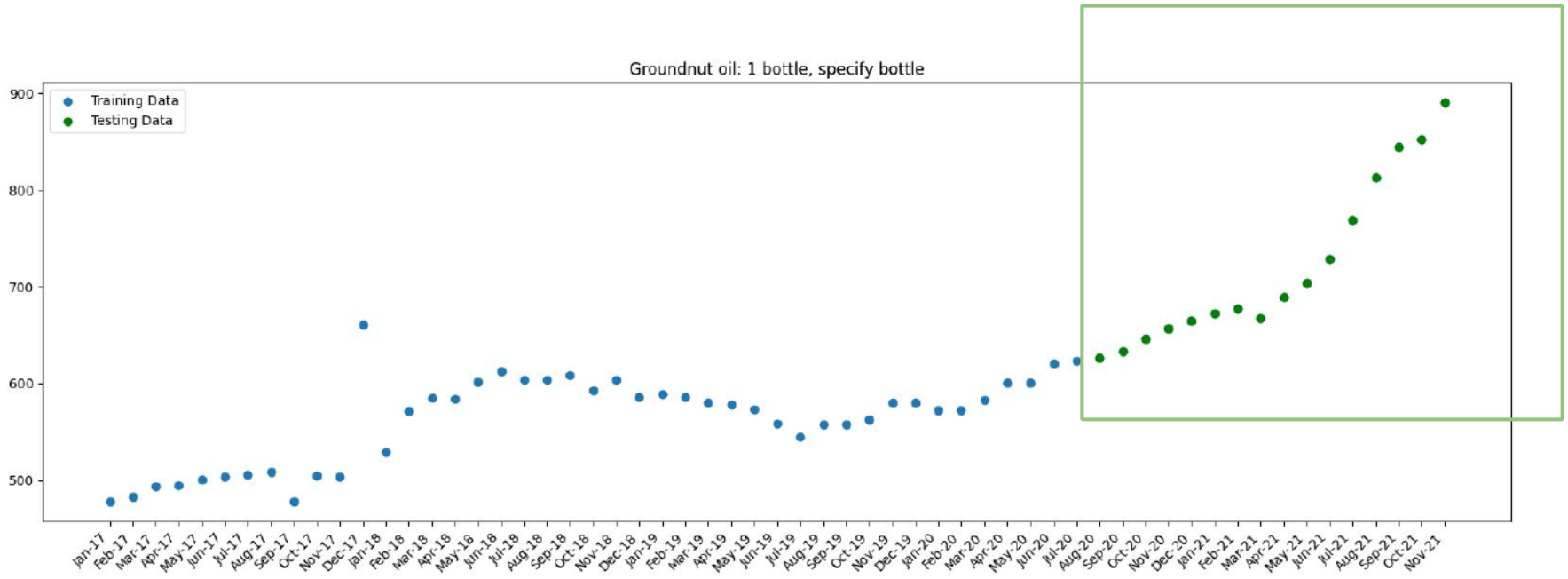


(B)

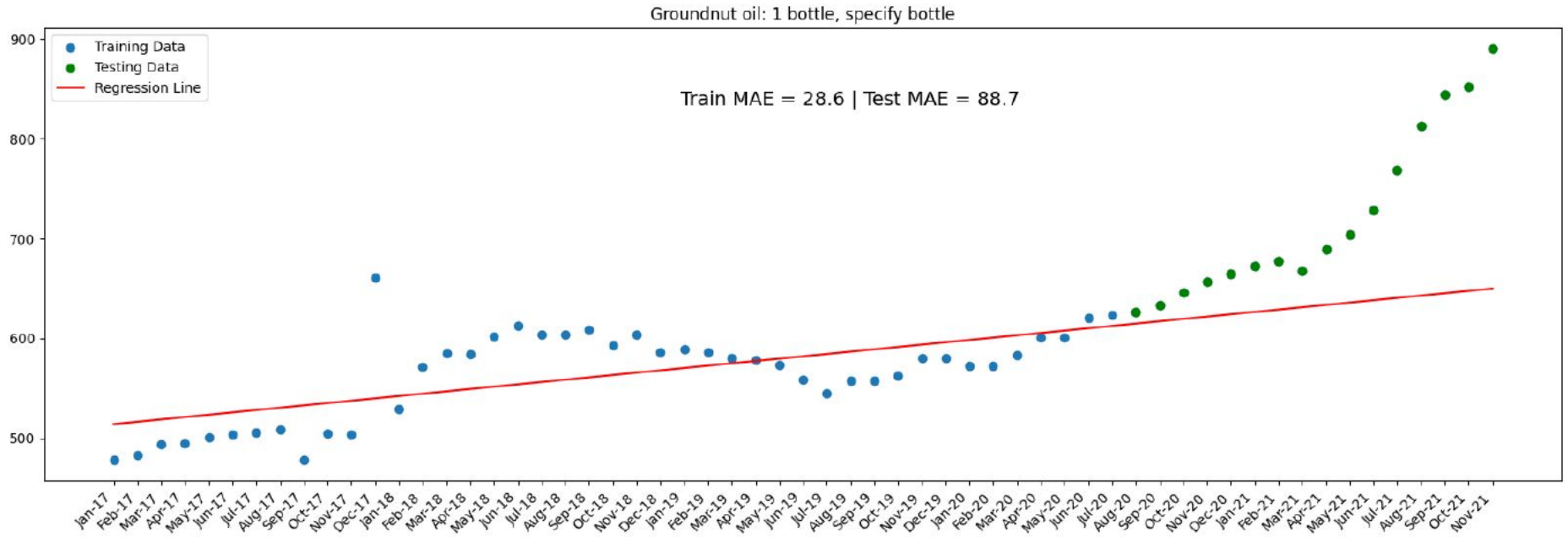
MAE = 28,63

Ricordate: la vera valutazione si fa su dati «mai visti»

Dovete valutare come si comporta il vostro modello su dati «mai visti».



Adattare una retta ai dati sull'olio di arachidi: valutazione sui dati di test



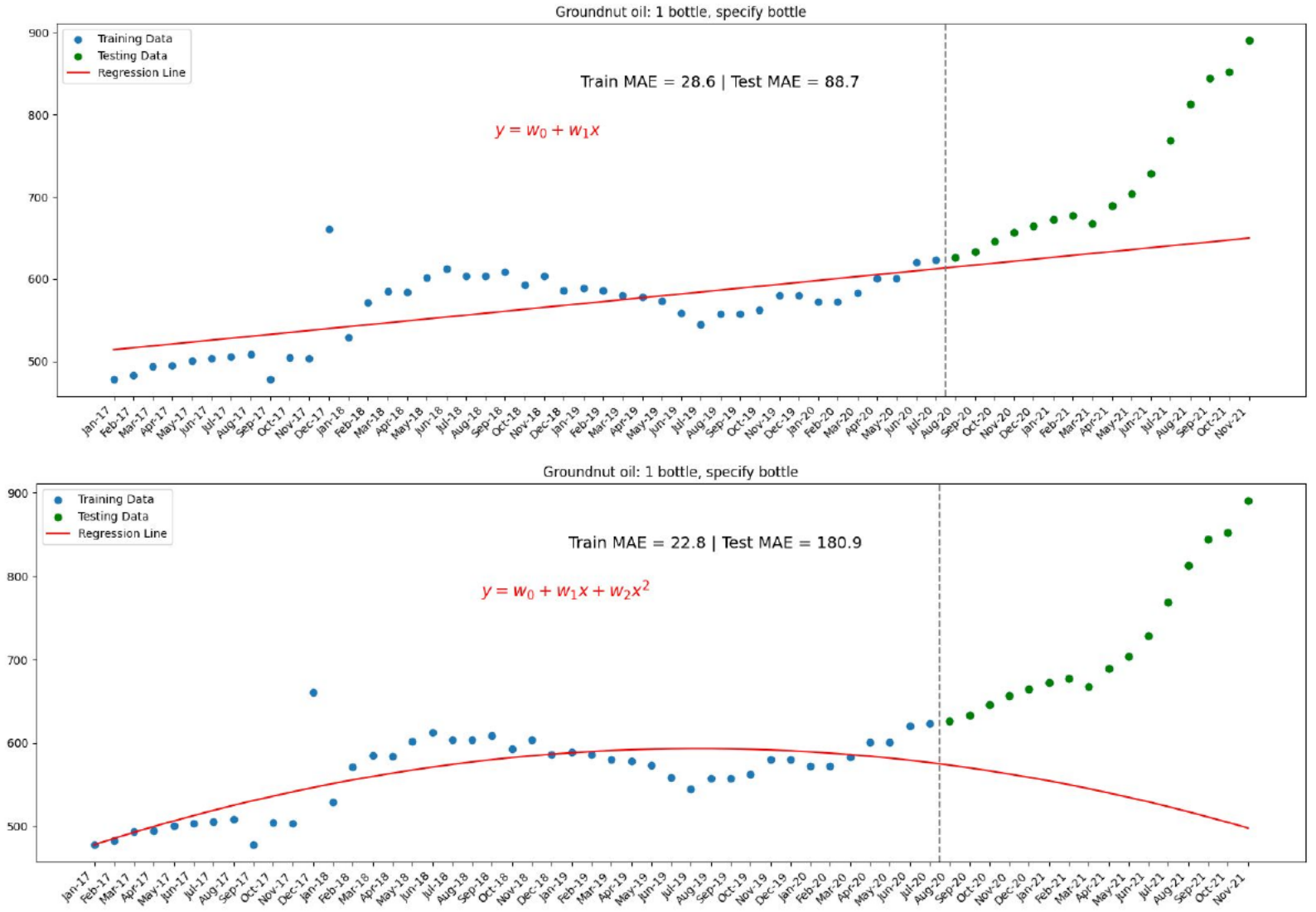
È il meglio che possiamo fare?

Feature engineering: modellare relazioni non lineari

Il feature engineering polinomiale consiste nel fornire a un modello lineare potenze (e prodotti) degli input originali

$$x^2, x^3, x_1 x_2 \cdots$$

così che un modello che traccia solo rette possa adattarsi a relazioni curve, poiché resta lineare nei pesi anche se ora gli input sono curvi.

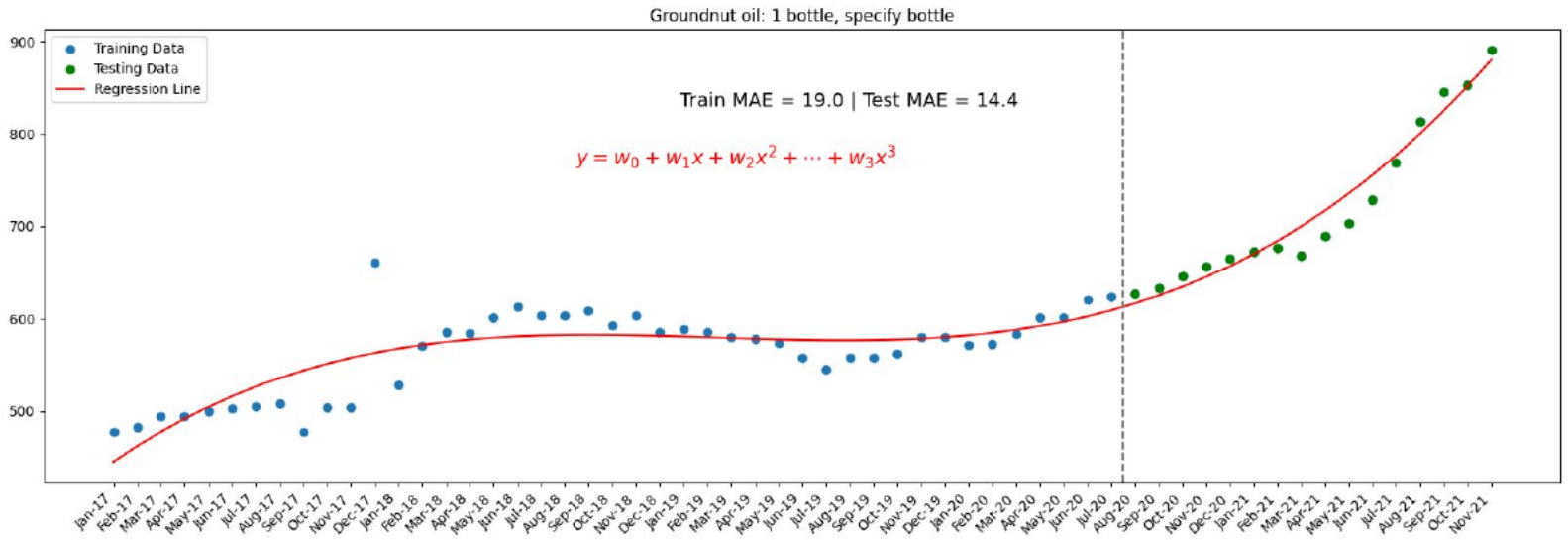


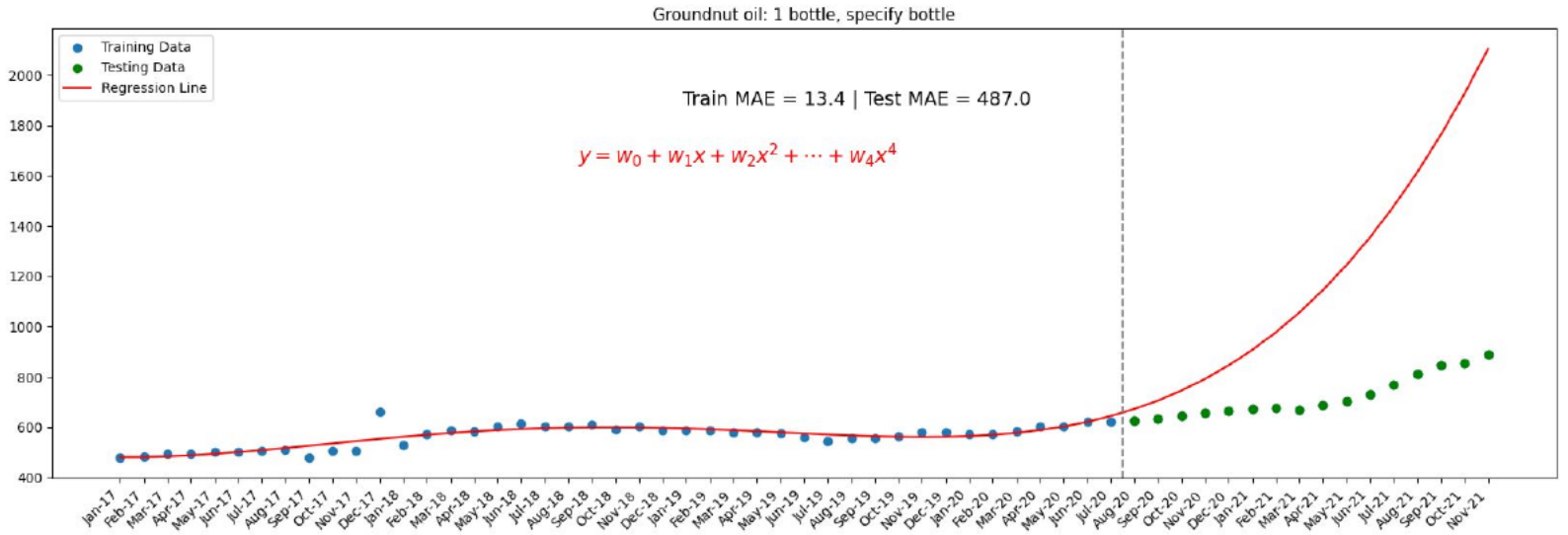
Feature engineering: feature polinomiali

Il feature engineering polinomiale consiste nel fornire a un modello lineare potenze (e prodotti) degli input originali

$$x^2, x^3, x_1 x_2 \cdots$$

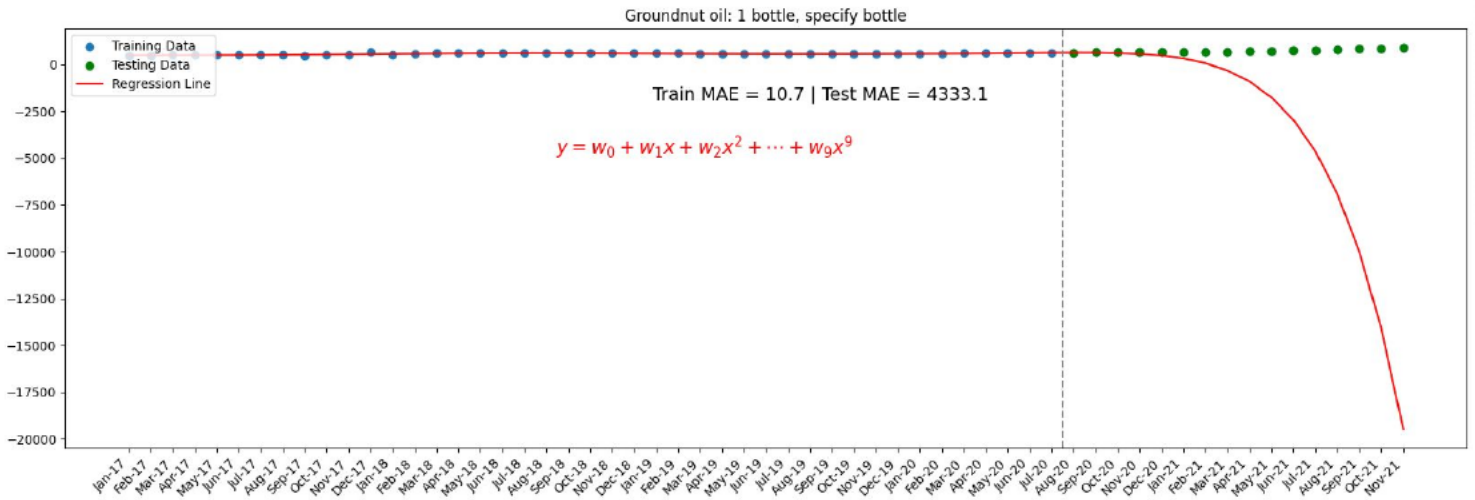
così che un modello che traccia solo rette possa adattarsi a relazioni curve, poiché resta lineare nei pesi anche se ora gli input sono curvi.





Esercizio: overfitting e generalizzazione

Un modello di machine learning ottiene un errore assoluto medio di 10 naira sul training set ma di 4.333 naira sul test set. Qual è la spiegazione più probabile?



https://partici.fi/53603205

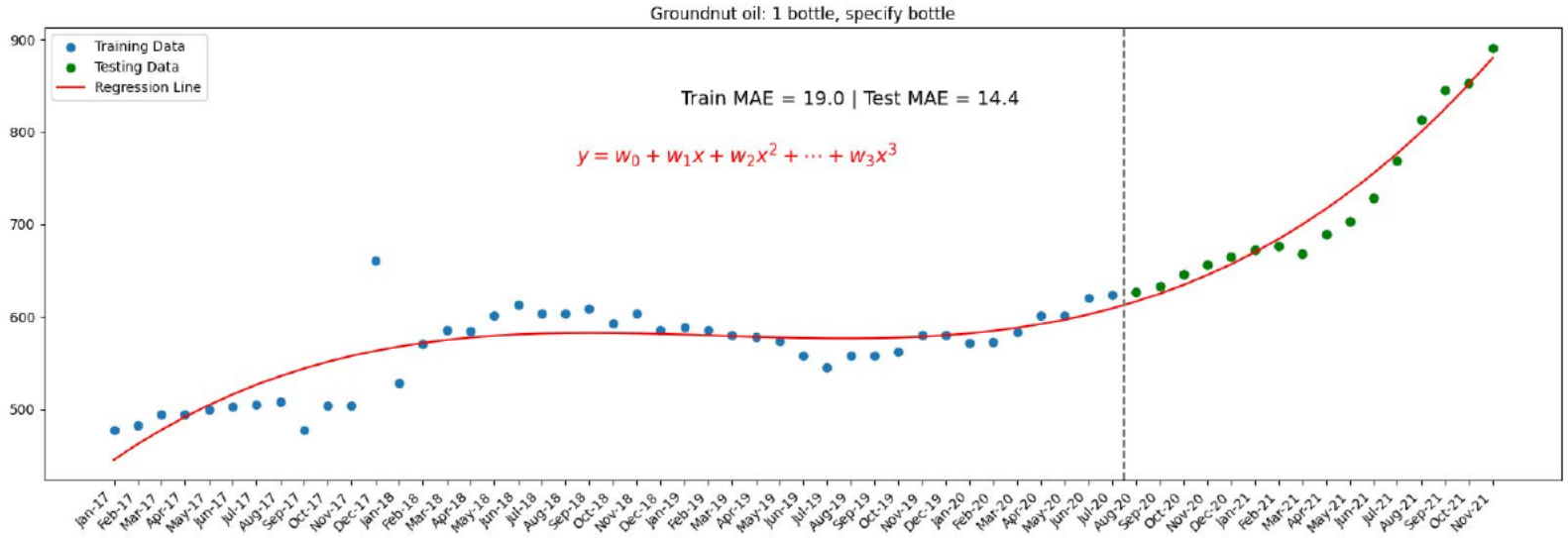
Ipotesi chiave della regressione lineare

Linearità: la relazione reale tra feature ed etichetta è approssimativamente lineare

Indipendenza: le osservazioni sono indipendenti tra loro

Omoschedasticità: la varianza della differenza tra etichetta reale e prevista (residuo) resta costante per tutti i valori delle feature

Assenza di multicollinearità: le feature di input non sono fortemente correlate



Per oggi ci fermiamo qui!
Grazie

Potete contattarmi:

- E-mail: tejumade@tri-ai.org
- Google Scholar, LinkedIn: tejumade Afonja
- X, Bluesky, GitHub: tejuafonja
- Sito web: tejuafonja.com
- Sito della mia ONG: tri-ai.org

Notebook di esercitazione:
https://colab.research.google.com/drive/1UQQ_JrBYSugUZtLSzYOIad5rUNZKsw8K?usp=sharing

Domande e risposte

Potete contattarmi:

- E-mail: tejumade@tri-ai.org
- Google Scholar, LinkedIn: tejumade Afonja
- X, Bluesky, GitHub: tejuafonja
- Sito web: tejuafonja.com
- Sito della mia ONG: tri-ai.org

Notebook di esercitazione:
https://colab.research.google.com/drive/1UQQ_JrBYSugUZtLSzYOIad5rUNZKsw8K?usp=sharing