Escola de verão virtual Climate Change AI

Introdução à IA

Tejumade Afonja

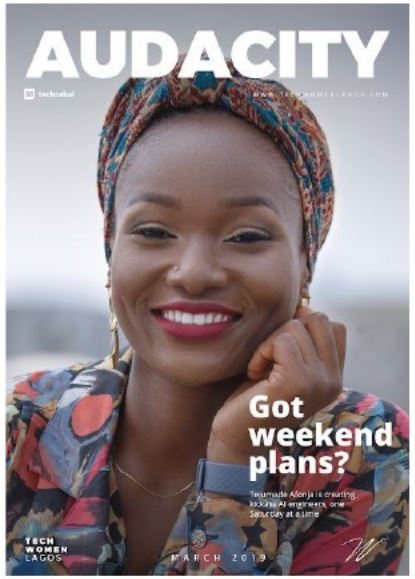
Slides adaptados do workshop piloto do AI Climate Institute

23 de julho de 2026

Aviso

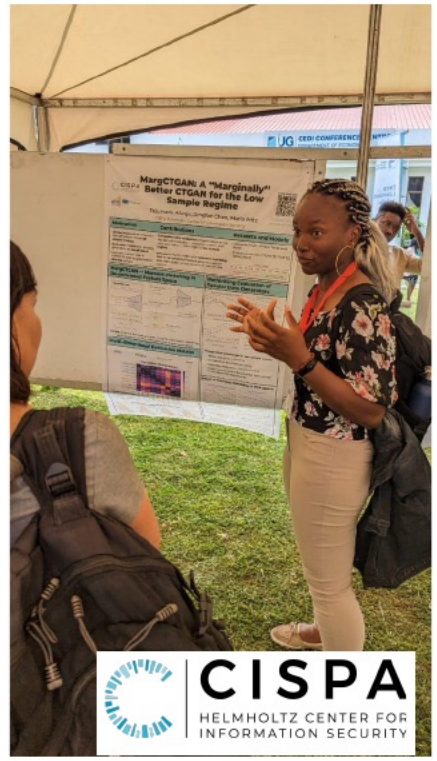
- Estes slides foram adaptados do workshop piloto do AI Climate Institute (Pilot Workshop).
- Modifiquei-os ligeiramente para incluir exercícios em sala e conteúdo adicional sobre o algoritmo de regressão linear.
- Acho a apresentação original excelente e escolhi usar o material. Sempre citem/referenciem os slides originais.
- Sou pesquisadora em aprendizado de máquina, mas (ainda) não em clima: a maioria dos exemplos climáticos não é minha especialidade, mas os fundamentos de AM são meu forte.

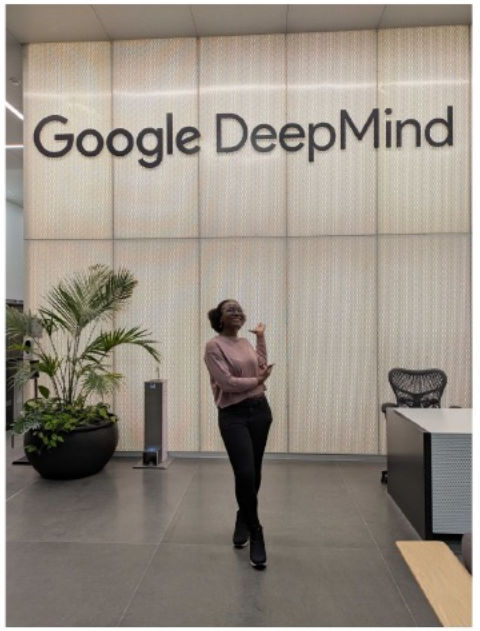
Sobre mim









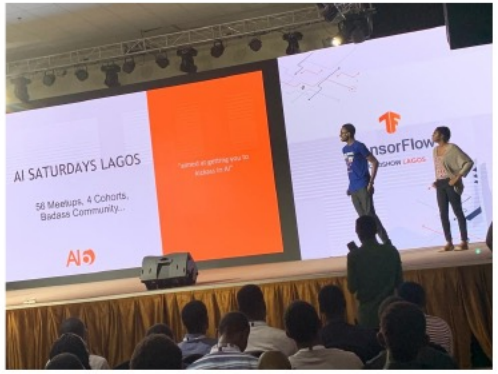












- Educadora e pesquisadora
  - Cofundadora, TRI AI & AI Saturdays Lagos (hoje TRI AI Saturdays)
  - Pesquisadora de doutorado, CISPA Helmholtz Center for Information Security
- Presidente geral, Deep Learning Indaba 2025; ex-pesquisadora estudante, Google DeepMind; bolsista DSSG 2022

Site: tri-ai.org

Roteiro da aula

O que é aprendizado de máquina?

Tipos de aprendizado de máquina

Dados

Entendendo o algoritmo de regressão linear

Perguntas e respostas

Roteiro da aula

O que é aprendizado de máquina?

Tipos de aprendizado de máquina

Dados

Entendendo o algoritmo de regressão linear

Perguntas e respostas

AM vs. outras abordagens

Pontos fortes e limitações

O que é IA?

Inteligência artificial (IA): qualquer algoritmo que permite a uma máquina realizar uma tarefa complexa (p. ex. tarefas associadas à inteligência humana)

A IA representa o desenvolvimento de sistemas computacionais capazes de emular a inteligência humana. Esses sistemas podem realizar tarefas que normalmente exigem capacidades cognitivas humanas, como resolver problemas, aprender com dados e tomar decisões.

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

O que é IA?

Inteligência artificial (IA): qualquer algoritmo que permite a uma máquina realizar uma tarefa complexa (p. ex. tarefas associadas à inteligência humana)

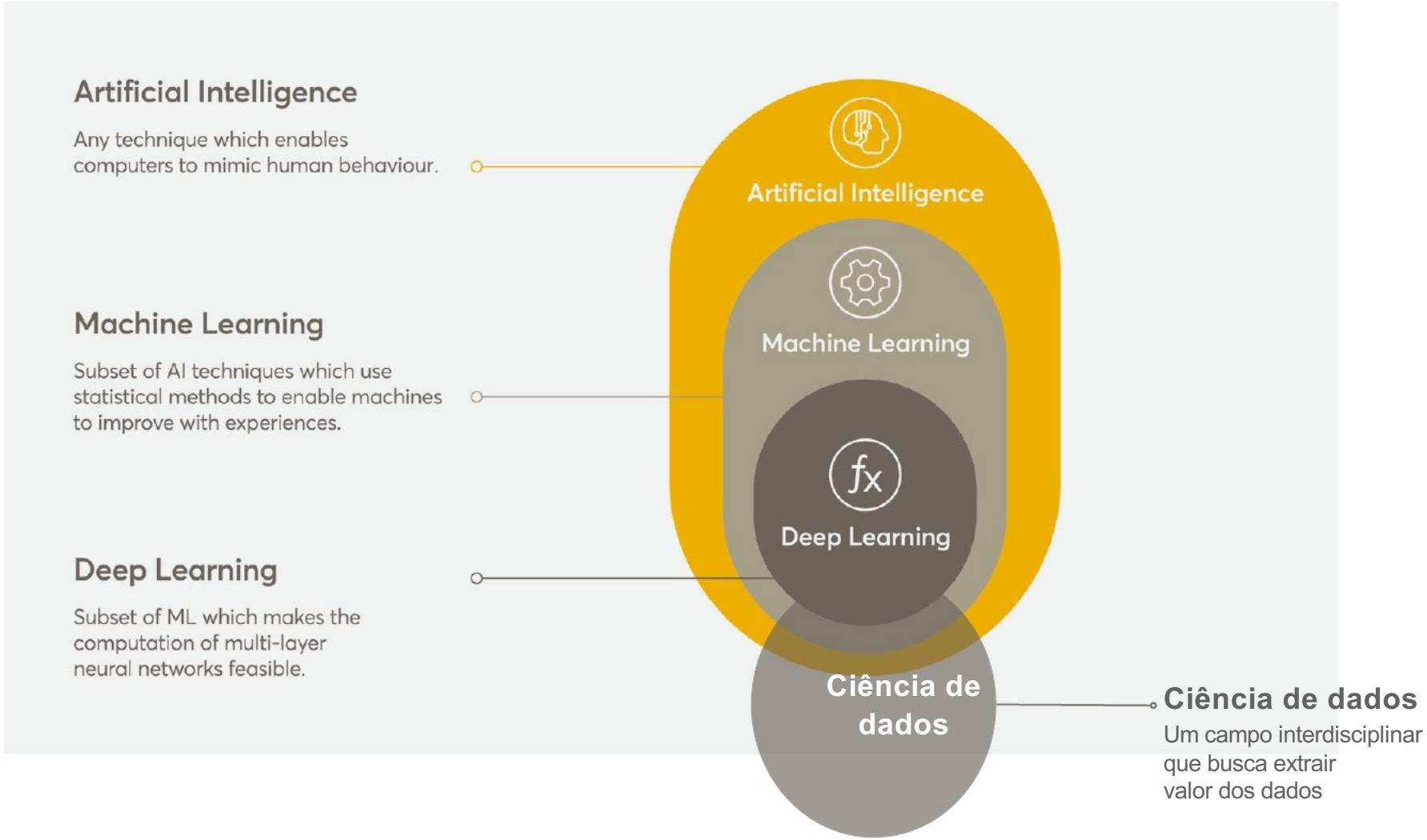
Sistemas baseados em regras:
técnicas que raciocinam automaticamente sobre um conjunto explícito de regras



Aprendizado de máquina (AM):
técnicas que extraem automaticamente padrões dos dados



Extração automática de padrões a partir de dados



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

Exercício: sistemas baseados em regras



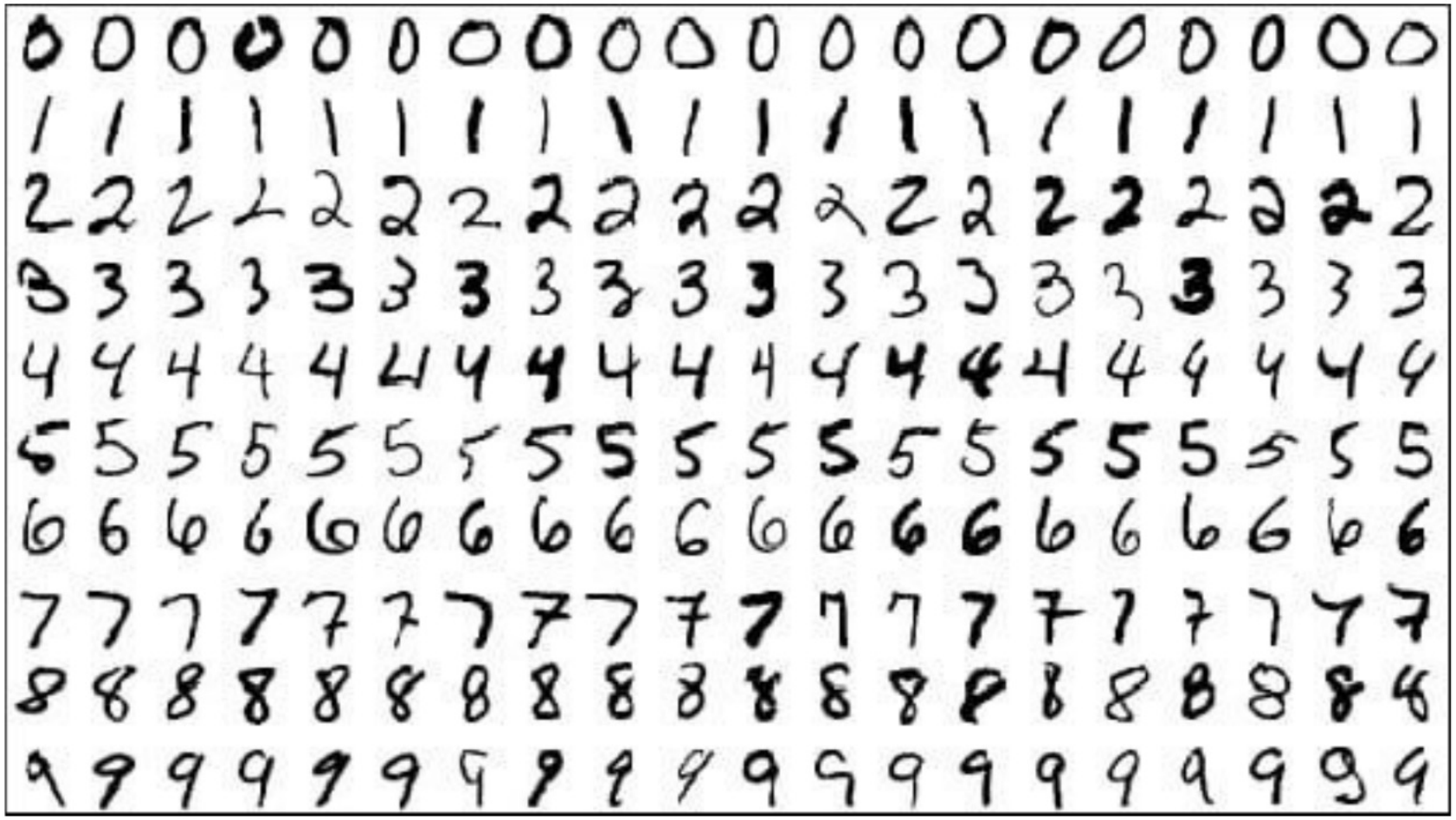
Escreva um programa simples para reconhecer o dígito 9 em uma imagem

Exercício: sistemas baseados em regras



seu
programa ainda
funciona? Se
não, pense
em como
modificá-lo

Aprendizado de máquina: aprender padrões gerais a partir dos dados



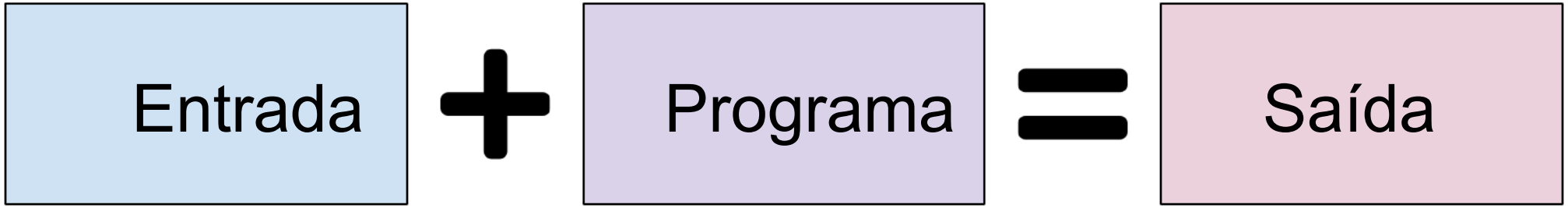
O que é aprendizado de máquina (AM)?

“Aprendizado de máquina é um campo de estudo que dá aos computadores a capacidade de aprender sem serem explicitamente programados”

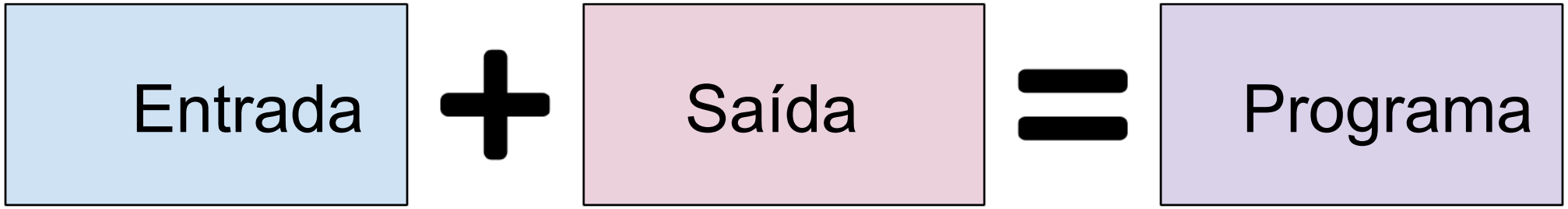
— Arthur Samuel (1959)

Programação tradicional vs. AM

Programação tradicional



Aprendizado de máquina (AM)



Alguns pontos fortes e limitações do AM

PONTOS FORTES

- Aproveitar padrões nos dados
- Executar tarefas em escala
- Otimizar sistemas complexos
- Capacidade de estar “sempre ligado”
- Combinar entradas heterogêneas
- Integrar-se a outros métodos

LIMITAÇÕES

- É uma ferramenta: não serve para toda tarefa
- Pode errar (com excesso de confiança)
- Pode exigir muitos dados e/ou computação
- “Entra lixo, sai lixo”
- Herda vieses dos dados/projeto/uso
- Supõe que regras/padrões persistem

Quando usar AM?

- Nem sempre: nem todo problema pode ou deve ser resolvido com aprendizado de máquina!
  - Use AM quando fizer mais sentido inferir as regras a partir de exemplos do que escrevê-las
- Quando você tem acesso a dados que “correspondem” à tarefa
  - Resolução, cobertura, amostra representativa de uma distribuição adequadas etc.
- Quando estimativas são úteis e erros são admissíveis
  - “Todos os modelos estão errados, alguns são úteis” – George Box
- Quando é uma solução custo-efetiva
  - Os custos incluem (i) criação e processamento do conjunto de dados e (ii) desenvolvimento, implantação e manutenção do modelo

Aprendizado de máquina

Aprendizado de máquina (AM): extrair automaticamente padrões dos dados

MODELOS

Redes neurais (aprendizado profundo)
Máquinas de vetores de suporte
Árvores de decisão
Modelos bayesianos
…

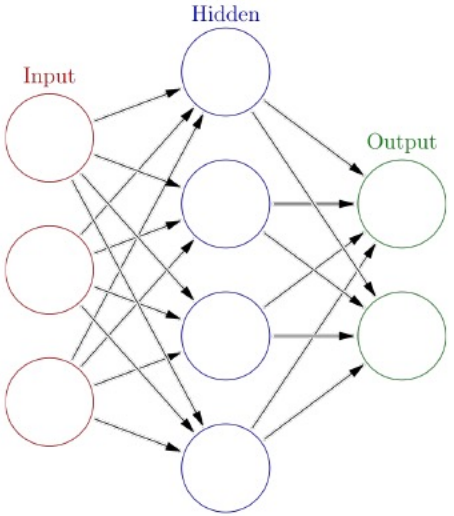
TIPOS DE DADOS / CASOS DE USO

Visão computacional
Processamento de linguagem natural
Robótica
Processamento de sinais
…

PARADIGMAS

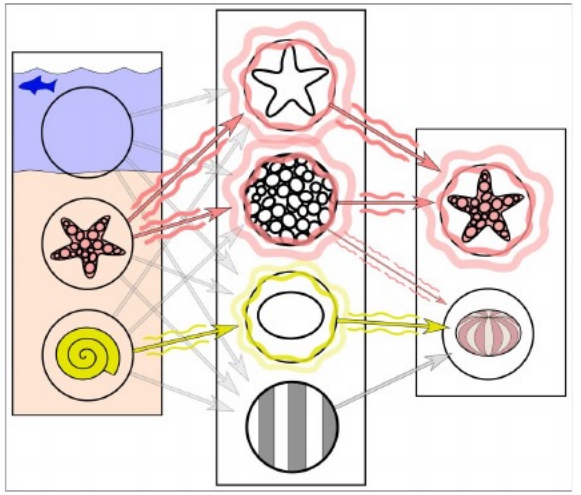
Modelos generativos vs. discriminativos
Informados pela física e robustos
Interpretáveis
Modelos específicos vs. de uso geral
…

O que é um modelo de AM?



O modelo consiste em parâmetros e em uma estrutura que, juntos, transformam entradas em saídas

A parte de “aprendizado” significa que esses parâmetros são ajustados no treinamento para produzir um “modelo aprendido” capaz de realizar uma tarefa



Modelos “grandes” vs. “pequenos”

- **Amplo espectro de modelos em número de parâmetros**
- Modelos pequenos rodam em segundos na maioria dos notebooks
- Modelos maiores podem exigir um servidor (p. ex. uma GPU potente por meio dia)
- Modelos muito grandes podem exigir 10.000+ GPUs por vários meses

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

# Carregar os dados
X , y = load_iris ( return_X_y = True )

# Criar, treinar e prever
model = RandomForestClassifier ()
model . fit ( X , y )

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

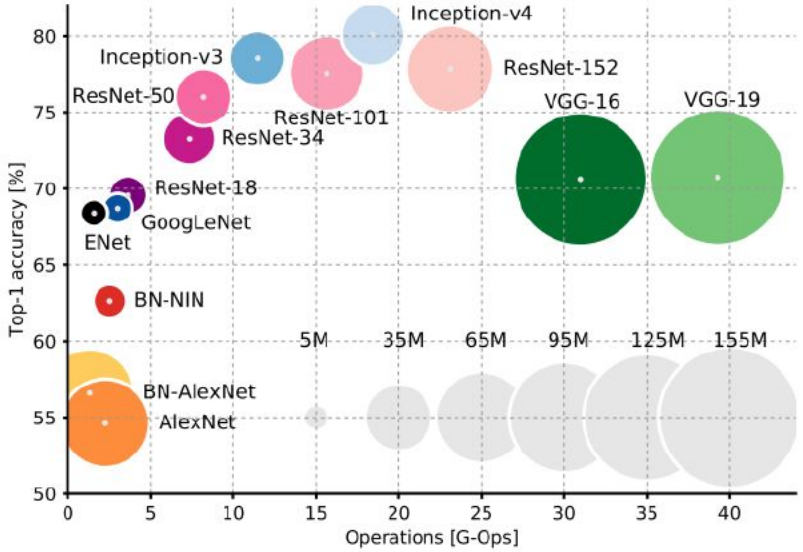
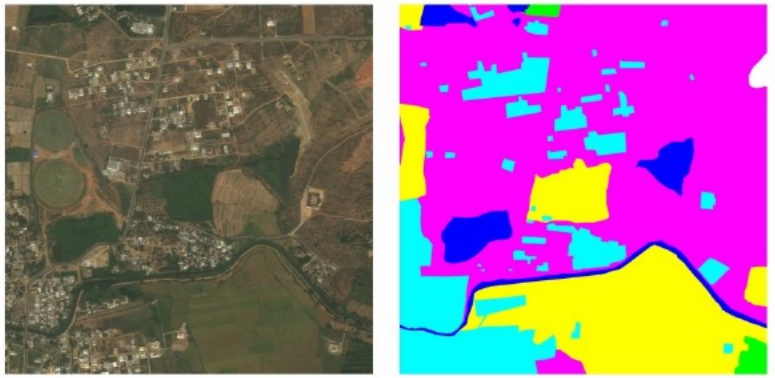


Image: https://arxiv.org/pdf/1605.07678

Modelos específicos vs. de uso geral

Modelo específico

- Treinado para uma única tarefa
- Em geral treinado com uma modalidade de dados
- Exemplo: segmentação por tipo de uso do solo em imagens de satélite

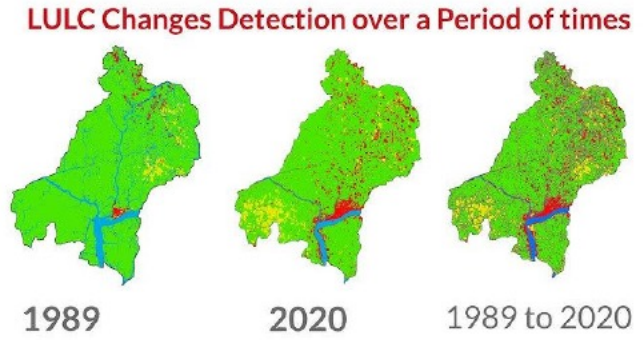


Modelos de uso geral

- Busca lidar com muitos tipos de tarefas em domínios variados
- Treinado com dados amplos e diversos
- Exemplo: modelos de fundação geoespaciais para segmentação por uso do solo, detecção de mudança de uso etc.

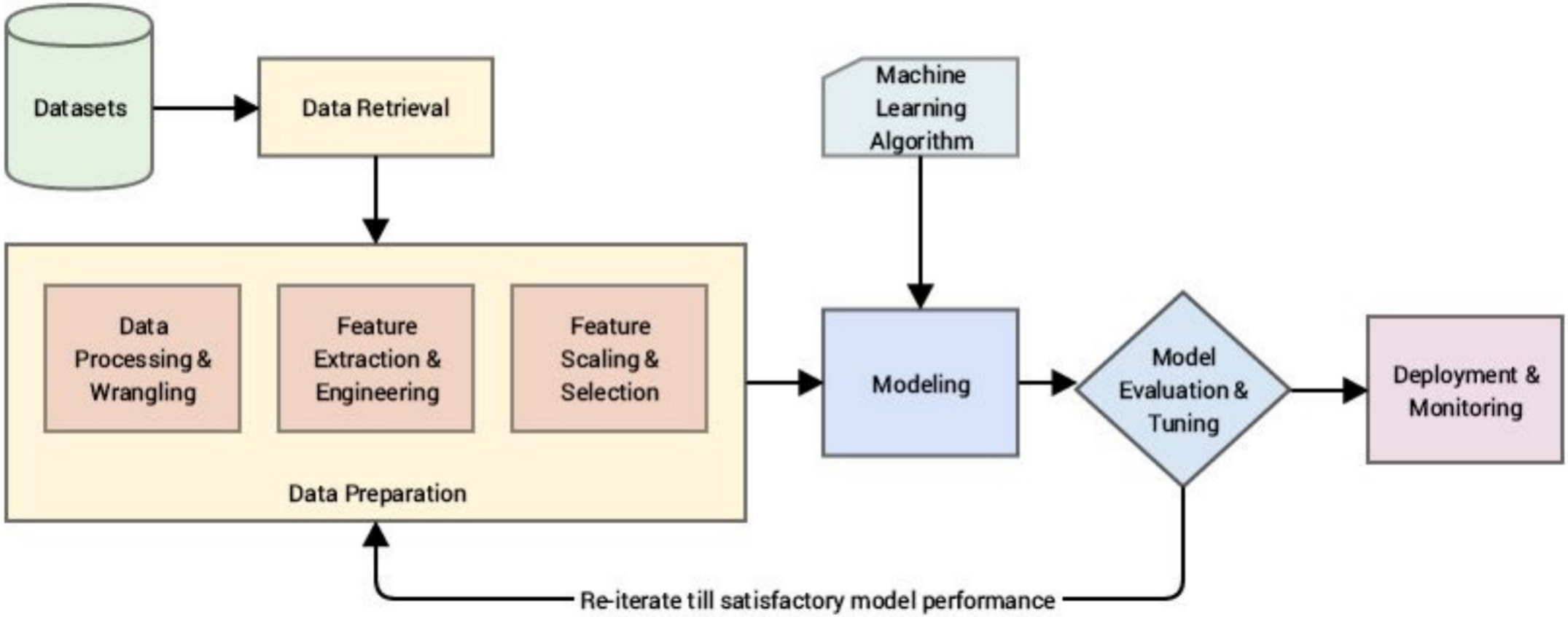


+



+ ...

AM na prática



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

Desenvolvimento → treinamento → inferência

Desenvolvimento

Montar a infraestrutura do pipeline de dados

Decidir como avaliar o desempenho do modelo

Testar diferentes algoritmos e hiperparâmetros

Escolher o modelo usado no treinamento final

Treinamento

“Ajustar” os parâmetros do modelo à tarefa

Escolher dados de treinamento representativos do contexto de inferência

Pode levar de minutos a meses

Precisa ser refeito se a distribuição dos dados mudar

Inferência

Produzir previsões para entradas “não vistas”

Usa os parâmetros “congelados” do treinamento

Supõe que entradas “não vistas” têm distribuição semelhante à dos dados de treinamento

Costuma ser bem mais rápida que o treinamento (de milissegundos a segundos)

Exercício: fluxo de desenvolvimento em AM

Qual destas etapas você acha que leva mais tempo em um projeto real de AM?

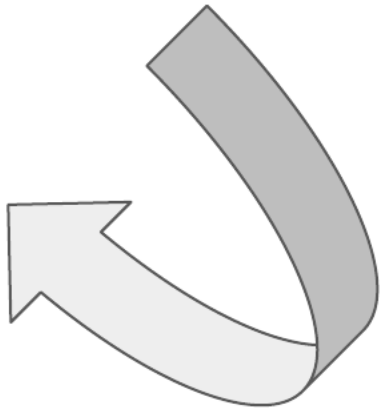
Desenvolvimento

Montar a infraestrutura do pipeline de dados

Decidir como avaliar o desempenho do modelo

Testar diferentes algoritmos e hiperparâmetros

Escolher o modelo usado no treinamento final



https://partici.fi/53603205

Roteiro da aula

O que é aprendizado de máquina?

Tipos de aprendizado de máquina

Dados

Entendendo o algoritmo de regressão linear

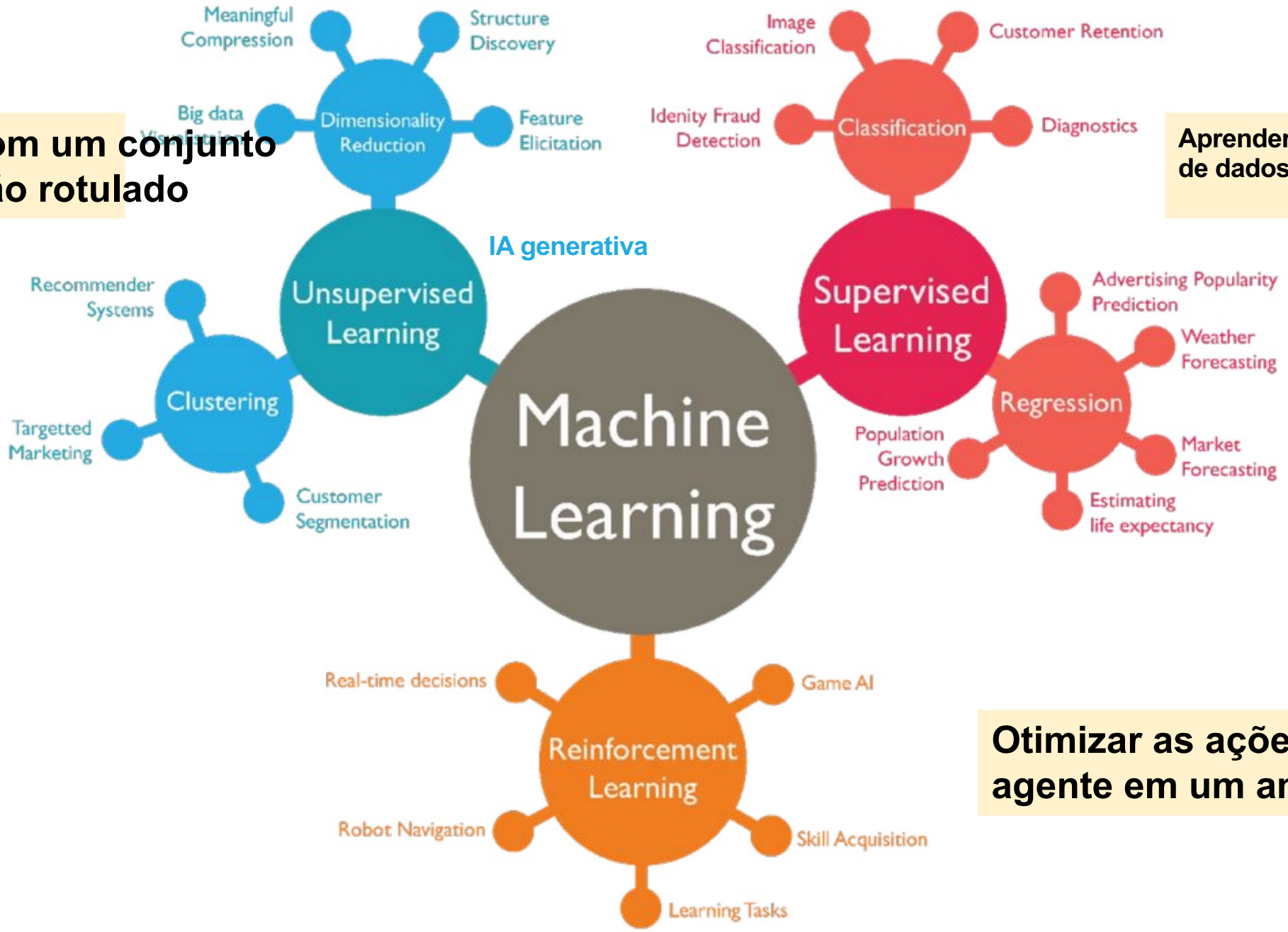
Perguntas e respostas

Principais paradigmas

Subáreas de AM relevantes para o clima

Aprender com um conjunto de dados não rotulado

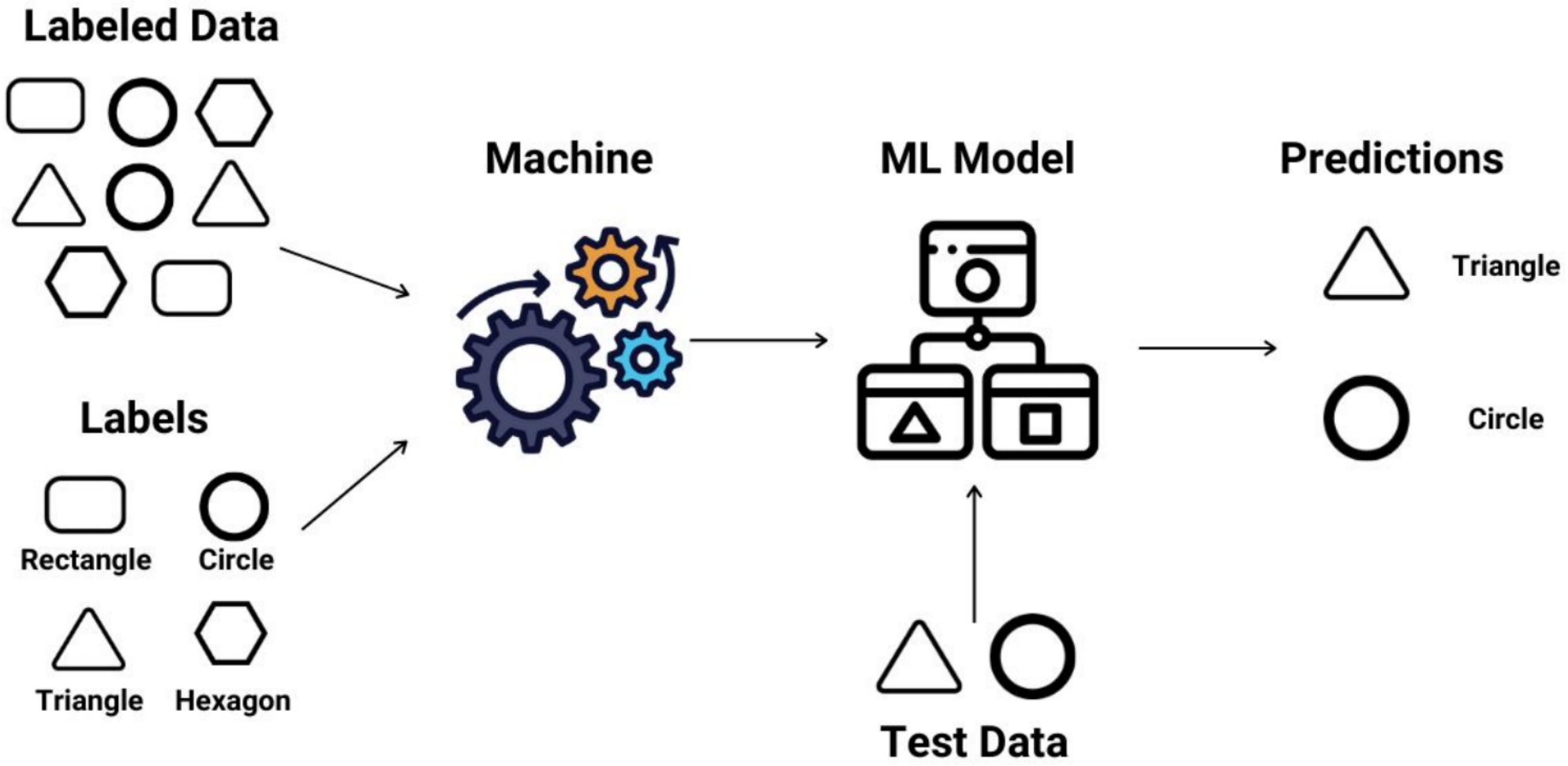
Aprender com um conjunto de dados rotulado



IA generativa

Otimizar as ações de um agente em um ambiente

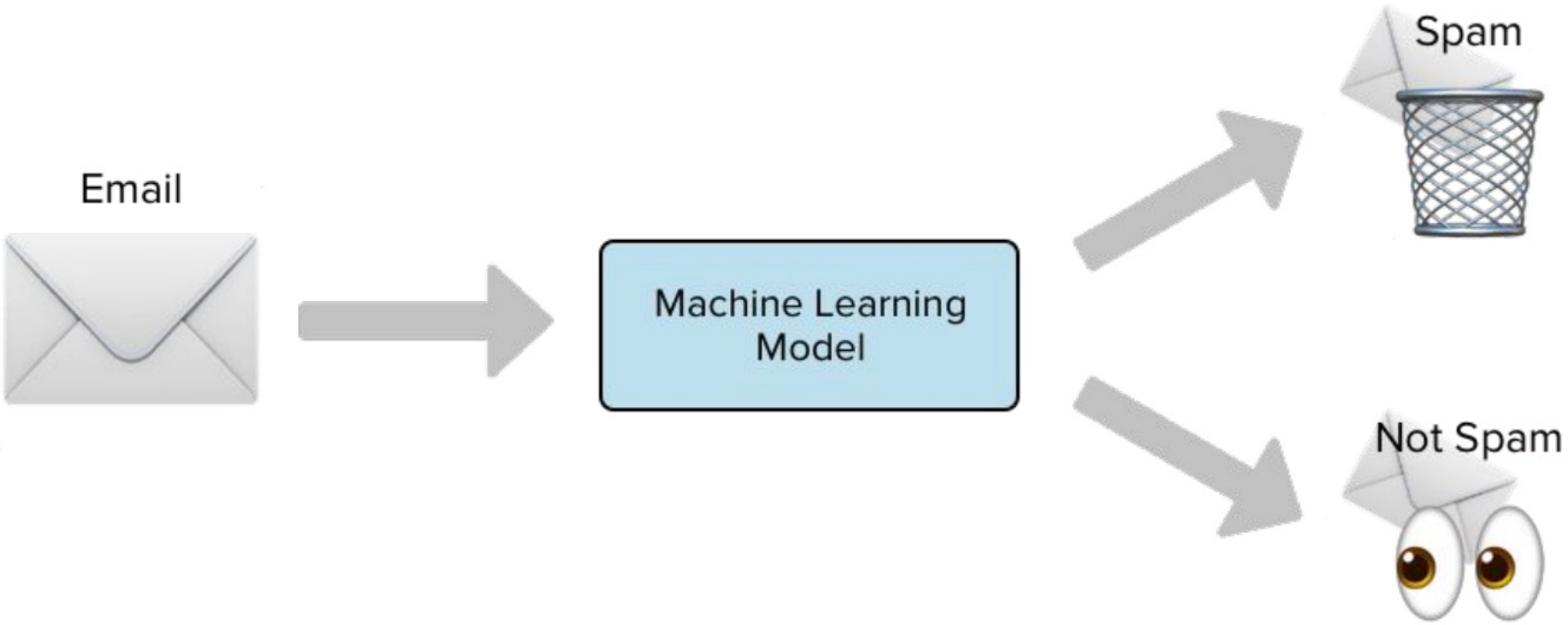
Exercício: esta é uma tarefa de regressão ou classificação?



Exercício: esta é uma tarefa de regressão ou classificação?

Distância até o mercado mais próximo	Presença de poço	Preço da casa (₦)
5 km (perto do mercado Balogun, Lagos)	Sim	₦75, 000, 000
15 km (periferia de Abuja)	Não	₦30, 000, 000

Exercício: esta é uma tarefa de regressão ou classificação?



Como escolher uma abordagem de aprendizado?



Dados de entrada:

- Podem incluir entradas e, quando conhecidas, saídas-alvo (chamadas rótulos)

  um conjunto de dados não rotulado é aquele cujas saídas-alvo não conhecemos a priori

Saída(s) do modelo:

- Categórica/numérica, discreta/contínua, saída única/múltipla

Abordagem de aprendizado:

Aprendizado supervisionado

- **Orientado por tarefa**
- **Dados rotulados**
- **Retorno sobre o desempenho**

Aprendizado não supervisionado

- **Orientado por dados**
- **Dados não rotulados**

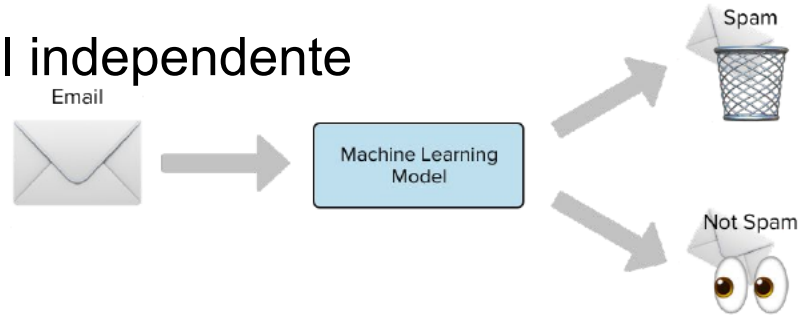
Aprendizado por reforço

- **Orientado por recompensa**
- **Dados de estado/ação coletados dinamicamente**
- **Retorno sobre o desempenho**

Terminologia do aprendizado supervisionado

- Rótulo: é o que estamos prevendo. Costuma ser denotado por y. Também chamado variável dependente ou alvo

- **Atributo: variável de entrada usada na** tarefa. Costuma ser denotado por . Também chamado variável independente



- **Modelo: codifica a relação entre os atributos** e o rótulo

- **Treinamento: o processo de ensinar o modelo a ir bem** nos dados; também dizemos aprender o modelo

- **Inferência: aplicar o modelo treinado a dados não vistos,** também chamados dados de teste

Aprendizado supervisionado

Dado um conjunto de dados rotulado, ou seja, entradas X com rótulos y:

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

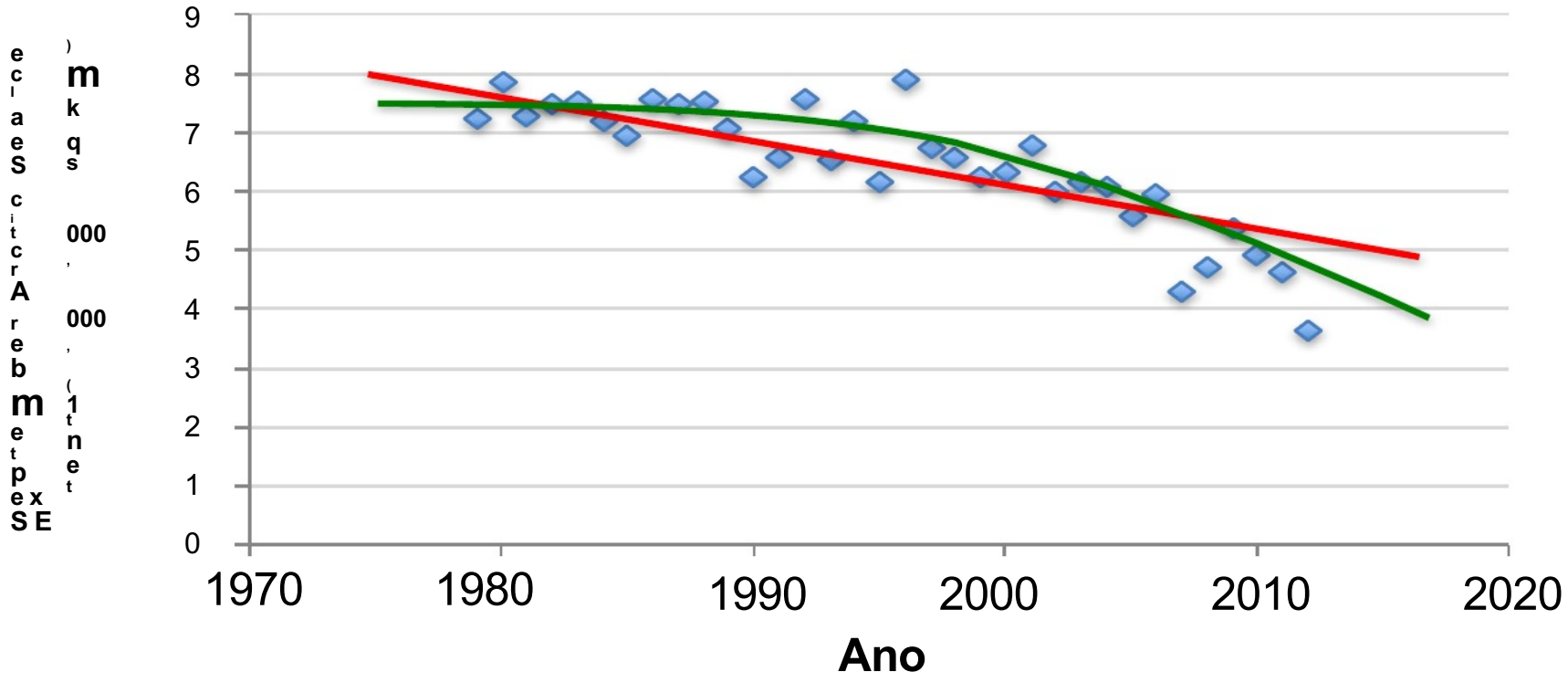
aprender uma função f que preveja y a partir de X, ou seja, $f(X) \cong y$.

- **X pode ter vários elementos, cada um representando um atributo (“feature”)**
- **Regressão: y tem valor real**
- **Classificação: y é categórico**

Aprendizado supervisionado: regressão

Aprender uma função f que preveja y a partir de X, ou seja, f(X) $\cong$ y .

- *Regressão: y tem valor real*

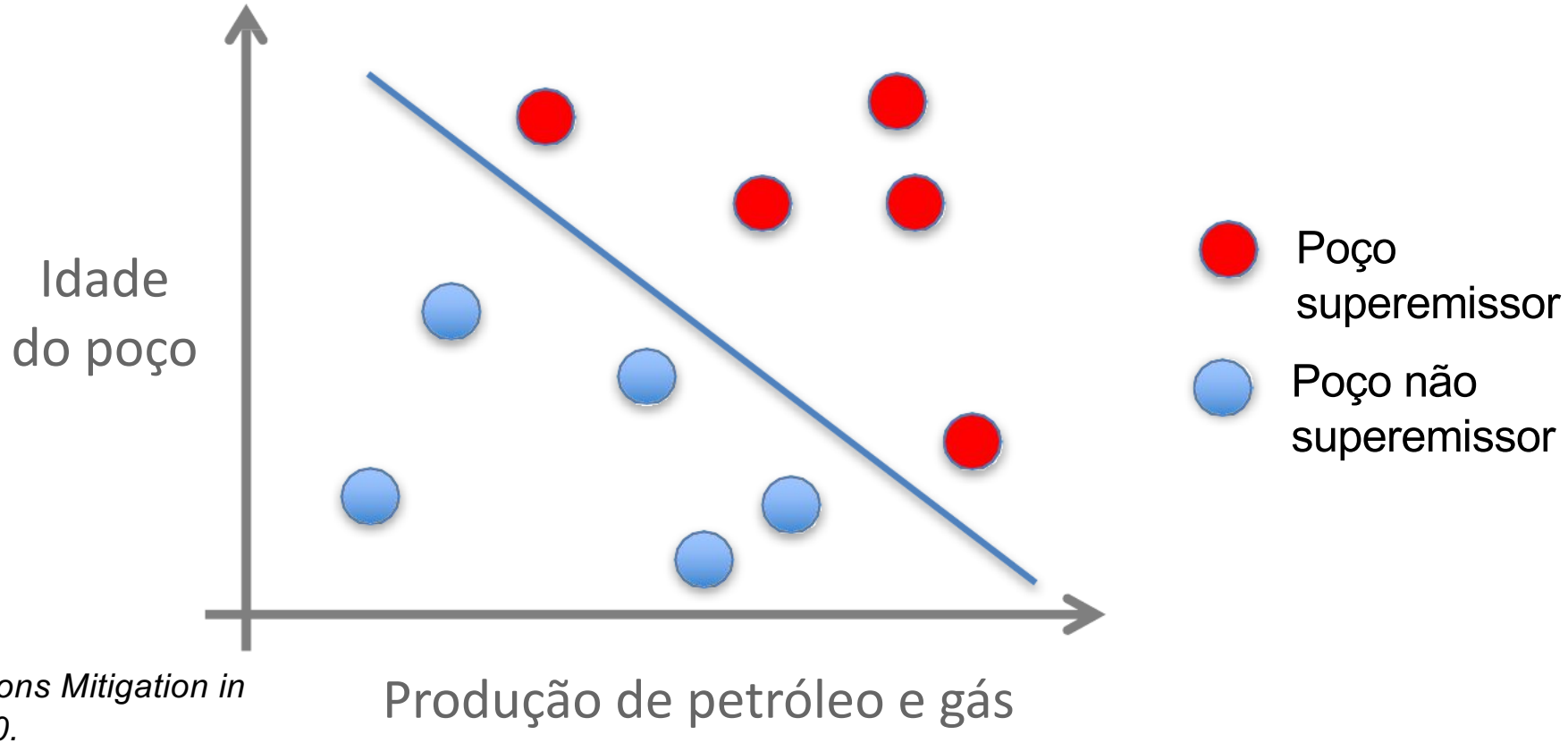


Dados de G. Witt. Journal of Statistics Education, volume 21, número 1 (2013)

Aprendizado supervisionado: classificação

Aprender uma função f que preveja y a partir de X, ou seja, f(X) $\cong$ y .

- *Classificação: y é categórico*



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

Aprendizado supervisionado: modelos/métodos comuns

- **Classificação: máquinas de vetores de suporte, florestas aleatórias, regressão logística, naive Bayes, redes neurais**

- Regressão: regressão linear, regressão ridge, processos gaussianos, redes neurais

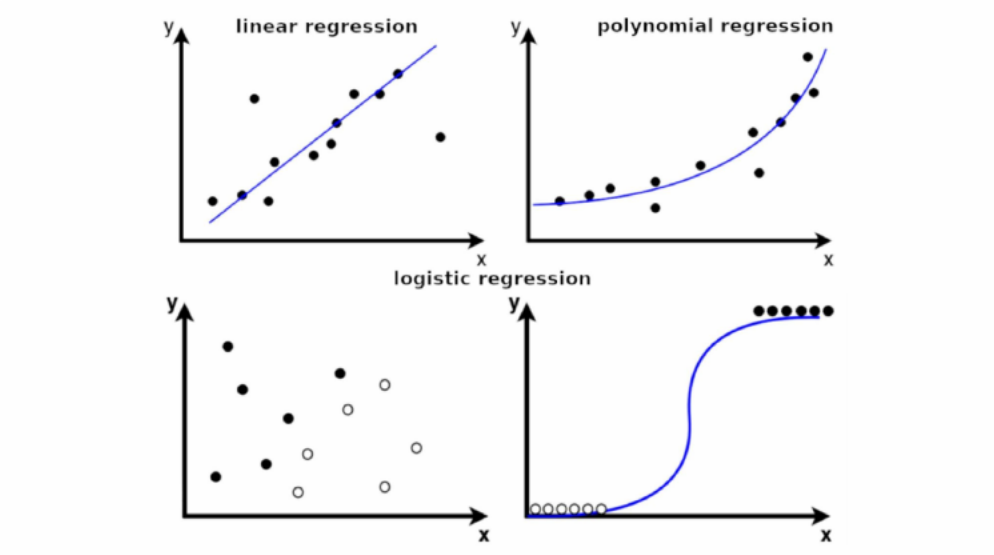


Imagem: Analytics Vidhya

Aprendizado supervisionado

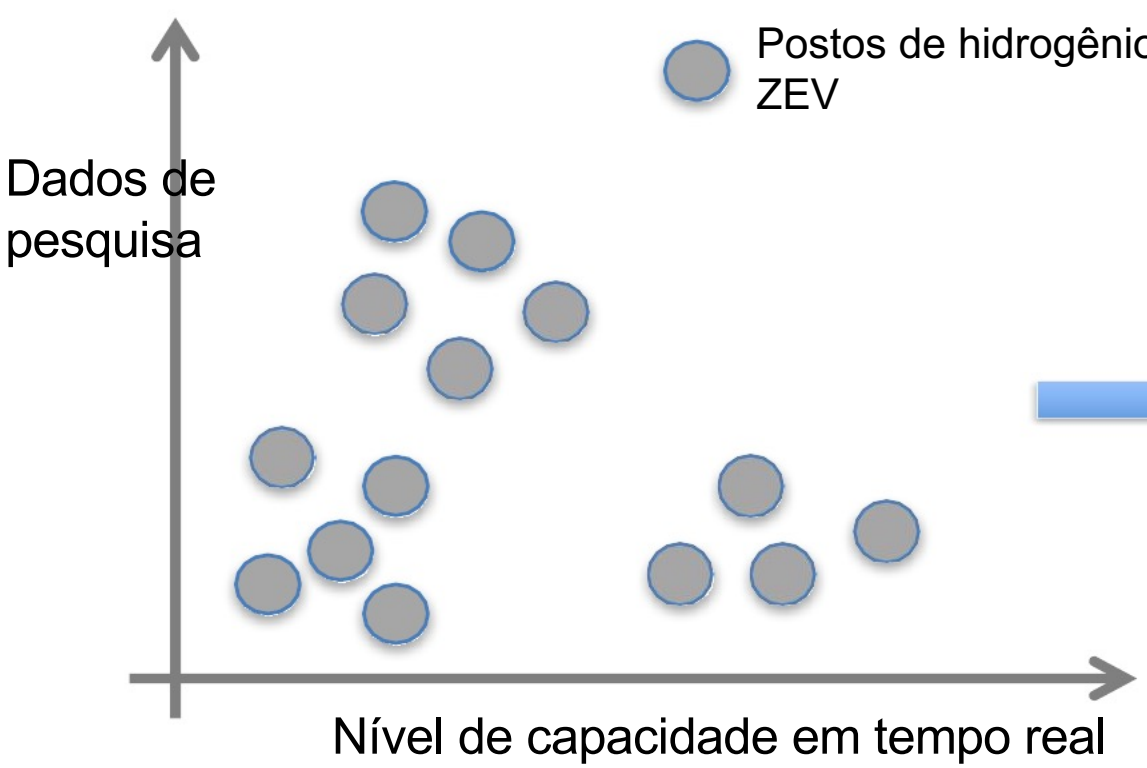
Critérios para escolher um algoritmo:

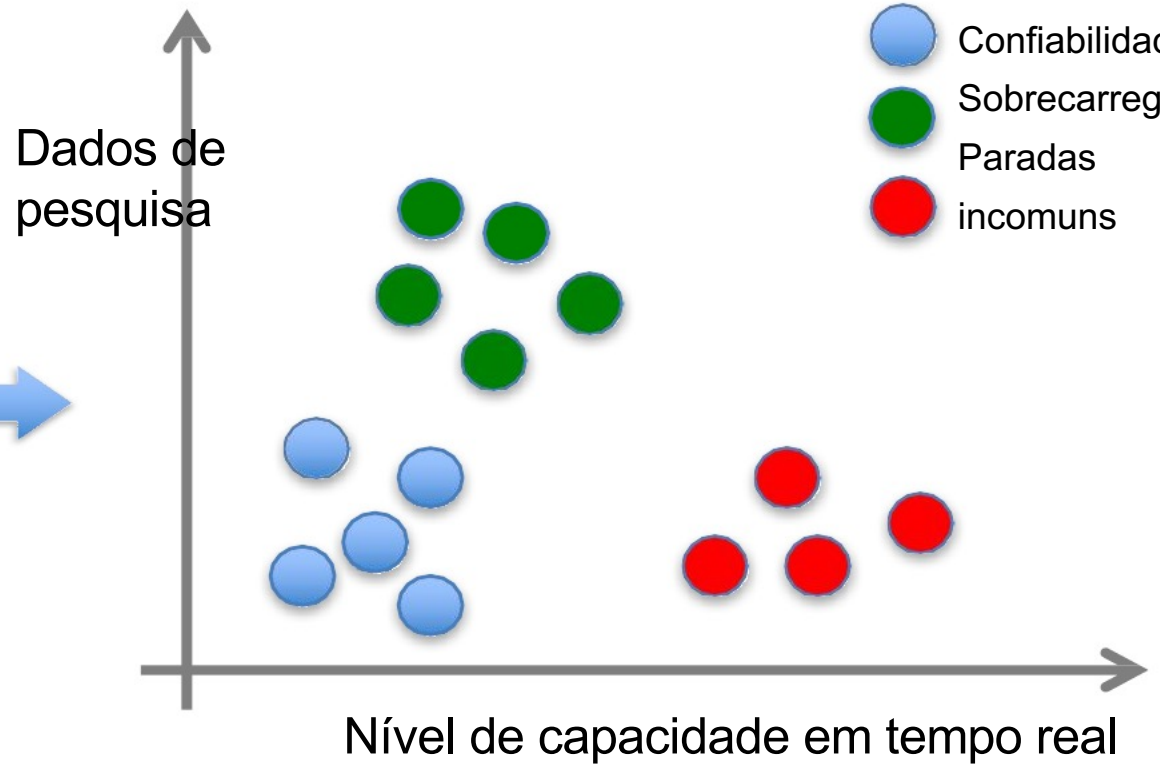
- Tipo de dados de entrada e saída
- Distribuição e qualidade dos dados, correlações entre atributos
- Desempenho desejado e garantias de robustez

Exemplo: classificar um poço como superemissor ou não com base em sua idade e produção de petróleo e gás, com rótulos de superemissor disponíveis para o treinamento

Aprendizado não supervisionado

Dado um conjunto não rotulado, ou seja, entradas X sem rótulos, aprender a estrutura oculta dos dados.





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

Aprendizado não supervisionado: modelos/métodos comuns

- **Aprender padrões nos dados: análise de componentes principais (PCA), agrupamento (clustering), algoritmos de detecção de anomalias**
- Modelagem generativa (gerar dados sintéticos): fluxos normalizadores, autoencoders variacionais, redes adversariais generativas
- Algumas tarefas de aprendizado por transferência: p. ex. alinhamento de domínios

Aprendizado não supervisionado

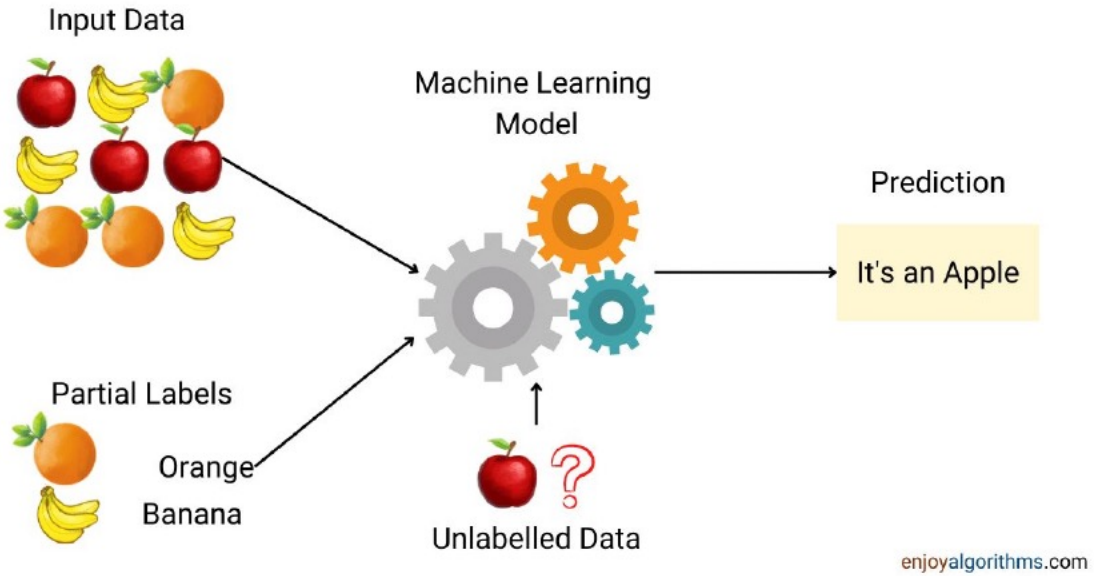
Critérios para escolher um algoritmo/modelo:

- Qual é o objetivo principal?
  - Agrupamento?
  - Geração de dados sintéticos?
  - Adaptação de domínio?
- Distribuição e qualidade dos dados, correlações entre atributos

Exemplo: geração sintética de perfis de carga energética de edifícios

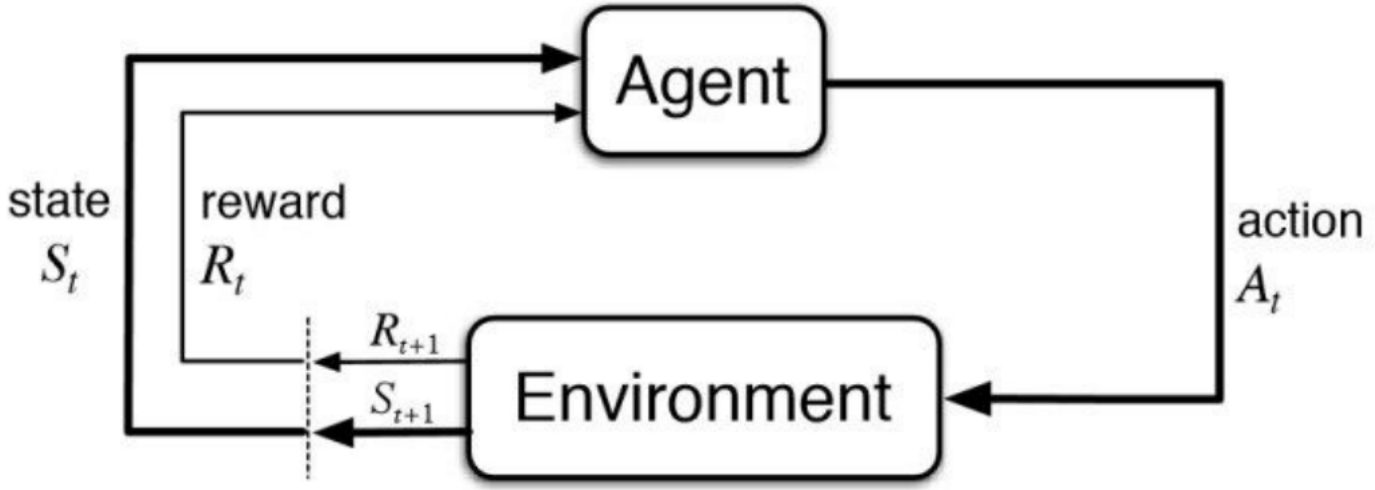
Aprendizado semissupervisionado

O aprendizado semissupervisionado fica entre o supervisionado e o não supervisionado. O conjunto de dados contém pontos rotulados e não rotulados



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

Aprendizado por reforço



Objetivo: encontrar a melhor política que maximize a recompensa total esperada

O RL é usado com frequência em contextos de controle

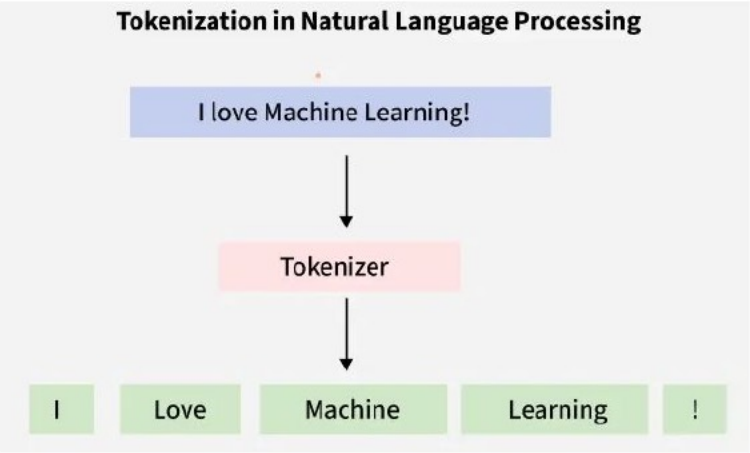
Exemplos:

- Jogos
- Otimização energética de um edifício
- Panorâmica e zoom em imagens de satélite

Figura de "5 Things You Need to Know about Reinforcement Learning"

Algumas abordagens de AM associadas a modalidades de dados

Processamento de linguagem natural (PLN)

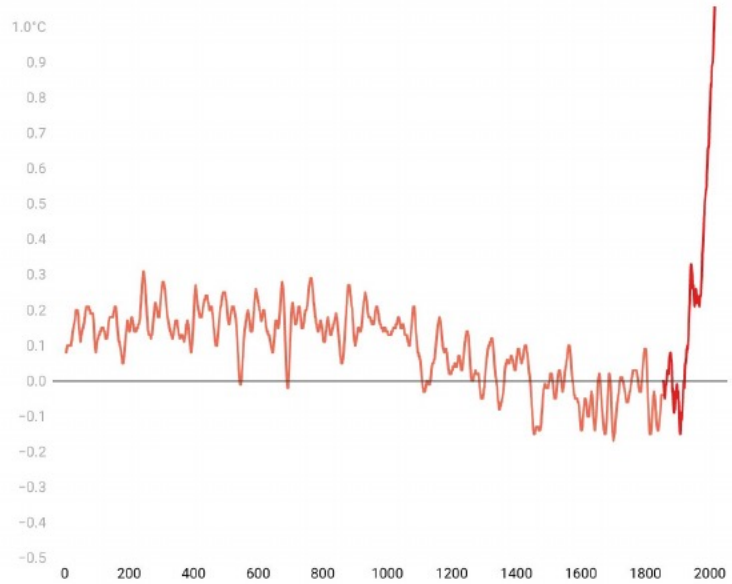


P. ex. redes neurais recorrentes (RNNs), modelos ocultos de Markov (HMMs), grandes modelos de linguagem (LLMs)

Visão computacional



Análise de séries temporais

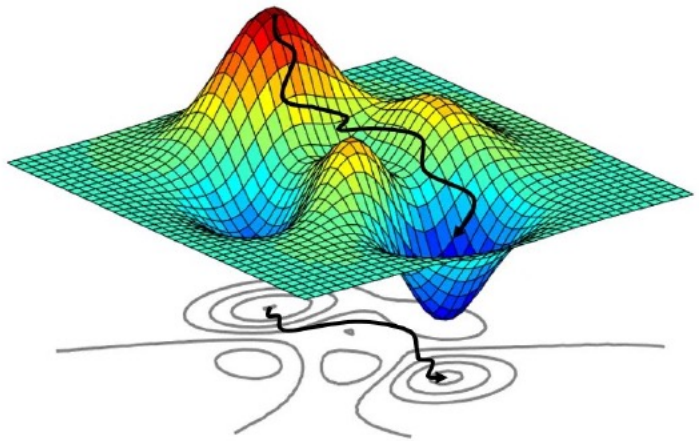


Imagens: Geeksforgeeks, Medium, TIME

AM para modelagem substituta

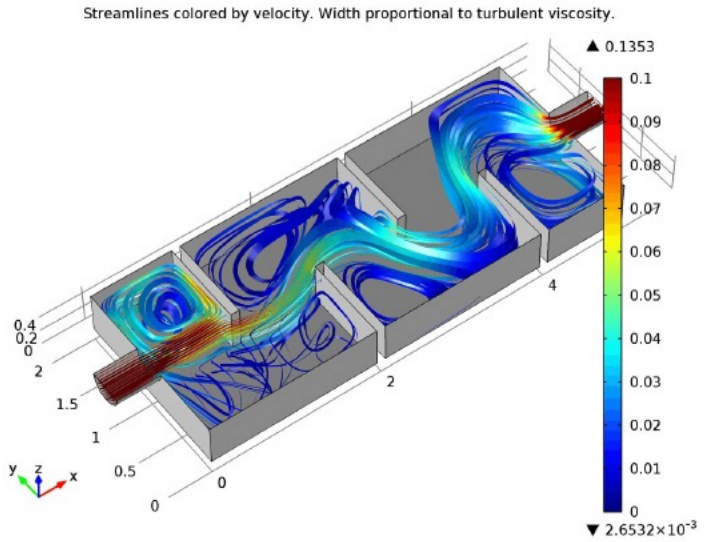
Modelos de otimização

Aproximar funções objetivo e restrições custosas para acelerar a busca por soluções ótimas

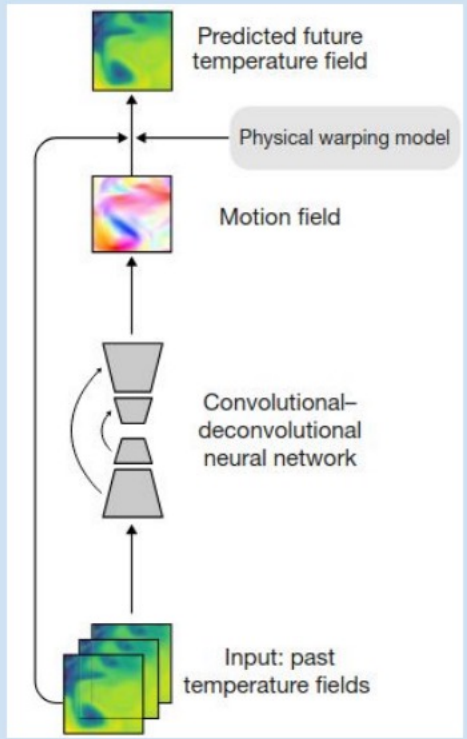


Modelos físicos

Emular simulações físicas complexas (dinâmica de fluidos ou mecânica estrutural)



Modelos informados pela física



Imagens: Medium, Comsol, Reichstein et al. 2019

AM interpretável e explicável

- A IA explicável justifica suas decisões a posteriori, mesmo que seu funcionamento interno seja complexo ou opaco.
- A IA interpretável tem lógica interna transparente que as pessoas podem entender examinando como o próprio modelo funciona.

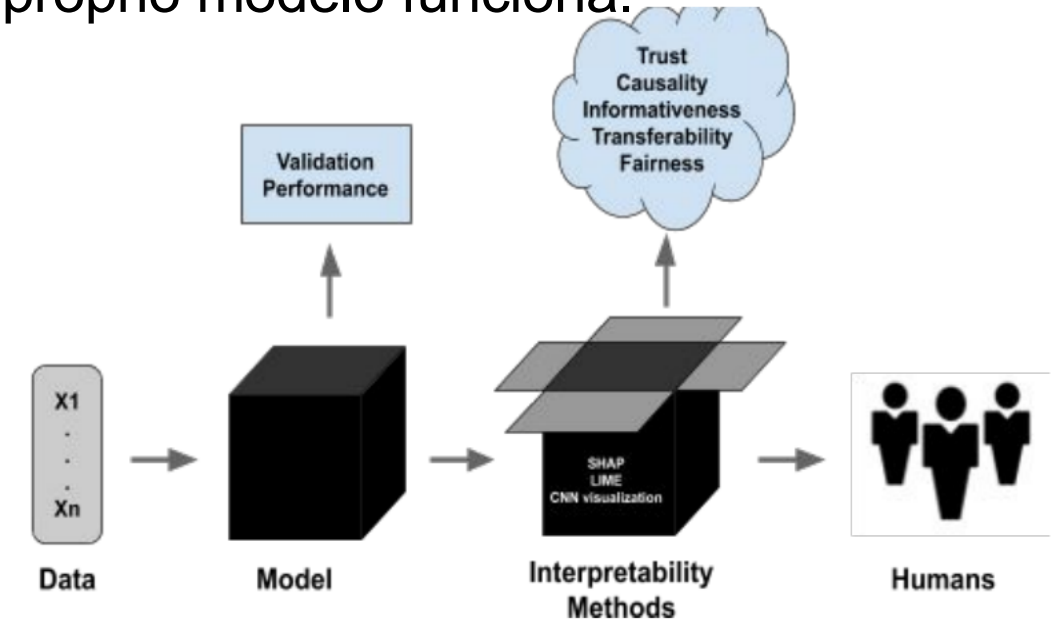
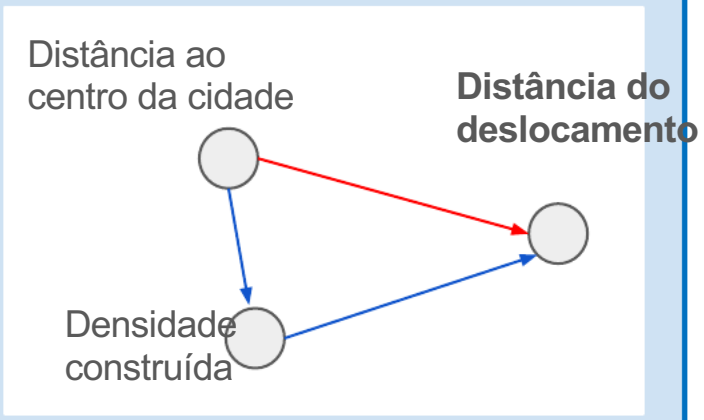


Imagem: ML CMU

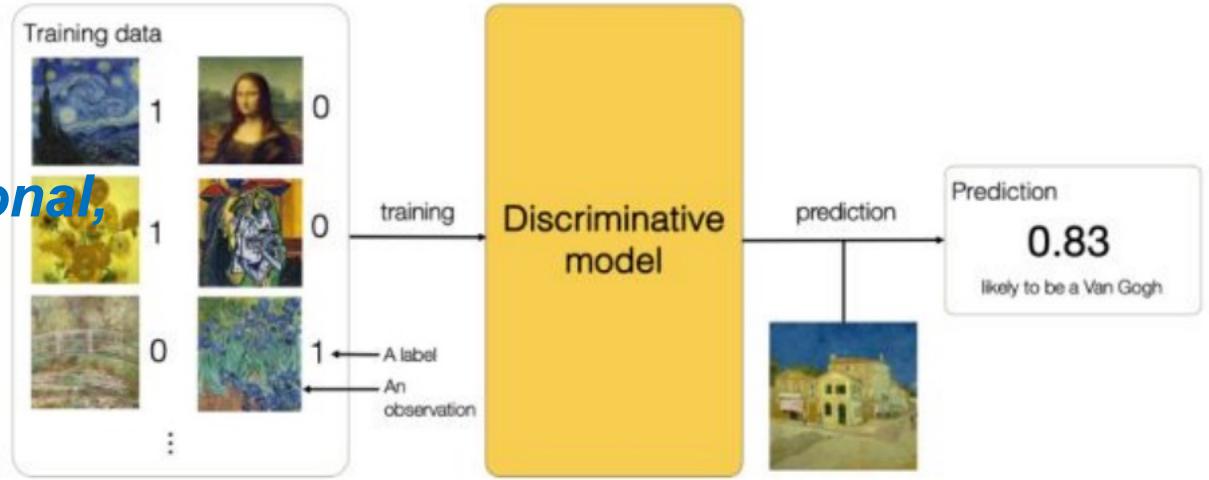
AM causal

- Identificar e quantificar relações de causa e efeito (em vez de correlações)
- Permite prever os resultados de intervenções e contrafactuais

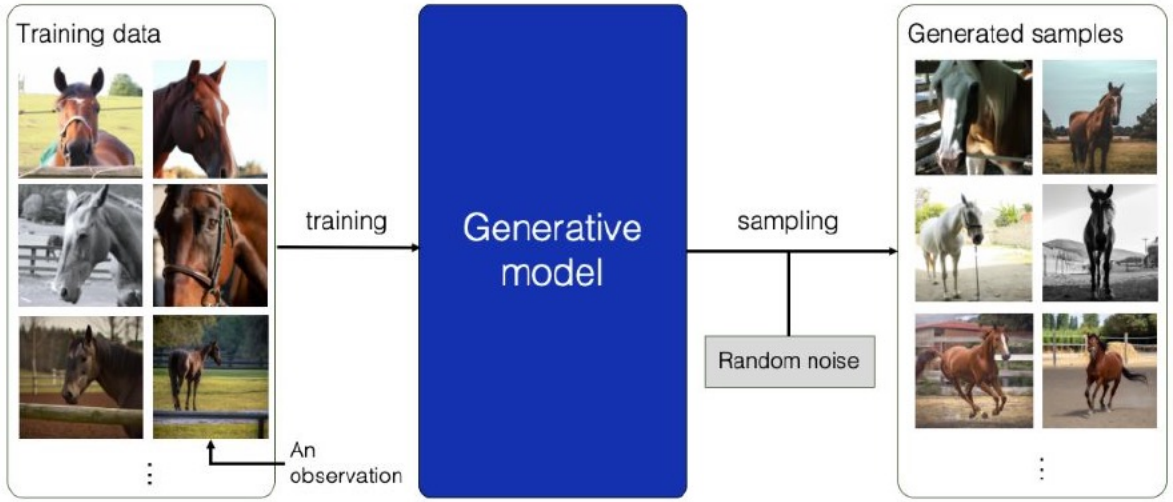


AM generativo vs. discriminativo

A IA discriminativa modela a *distribuição de probabilidade condicional*, aprendendo a mapear diretamente entradas em saídas.



A IA generativa ("GenAI") modela explicitamente a distribuição conjunta dos dados, aprendendo como atributos e rótulos ocorrem juntos.



Imagens: Analytics yogi

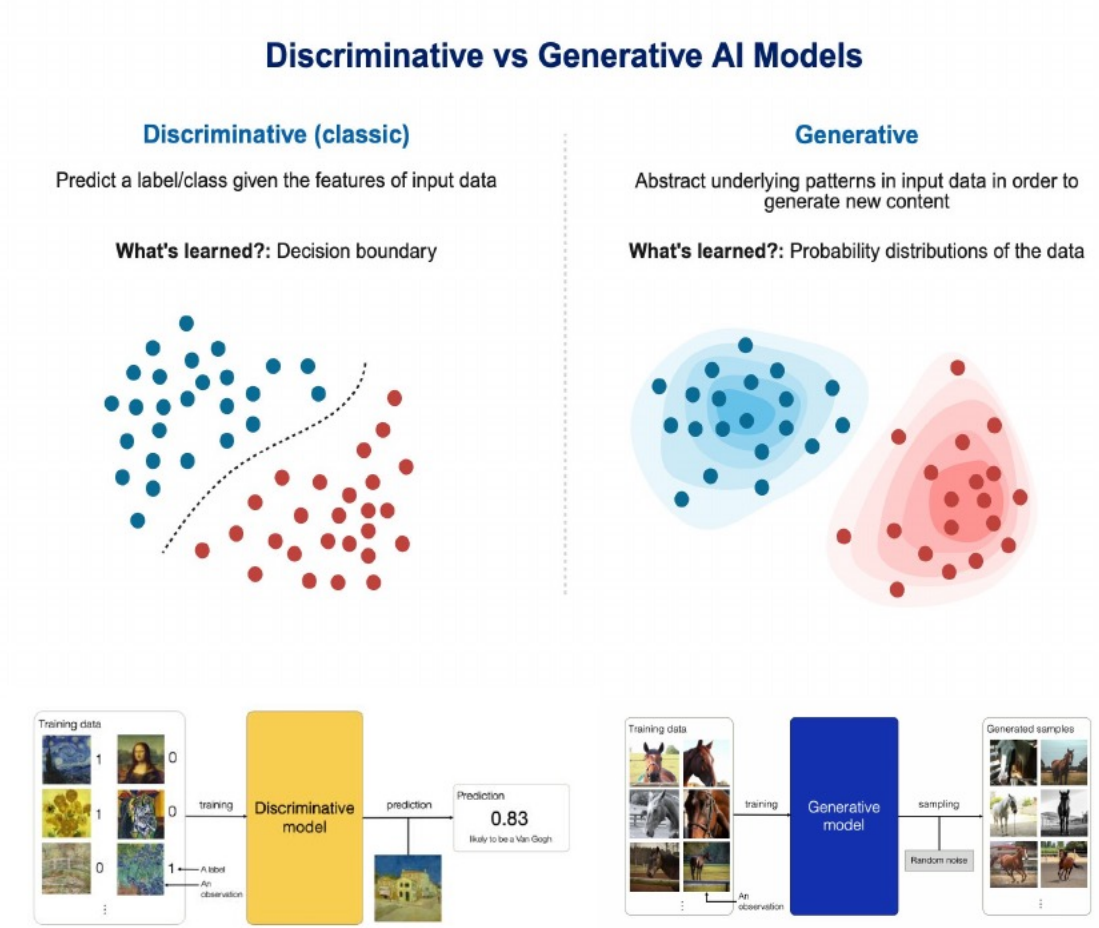
Modelos discriminativos vs. generativos

Modelos discriminativos

- Usados para separar (“discriminar” entre) diferentes saídas com base no valor da entrada.
  - Aprendem a distribuição condicional P(Y|X) dos alvos Y dadas as entradas X
  - P. ex. preveem um rótulo ou categoria (spam ou não spam)

Modelos generativos

- Aprendem a distribuição conjunta P(X,Y) das entradas X e do alvo Y
- Aprendem a criar (“gerar”) novo conteúdo amostrado da distribuição aprendida: texto, imagens, áudio, código



Exemplo: Generative Pre-trained Transformer (GPT)

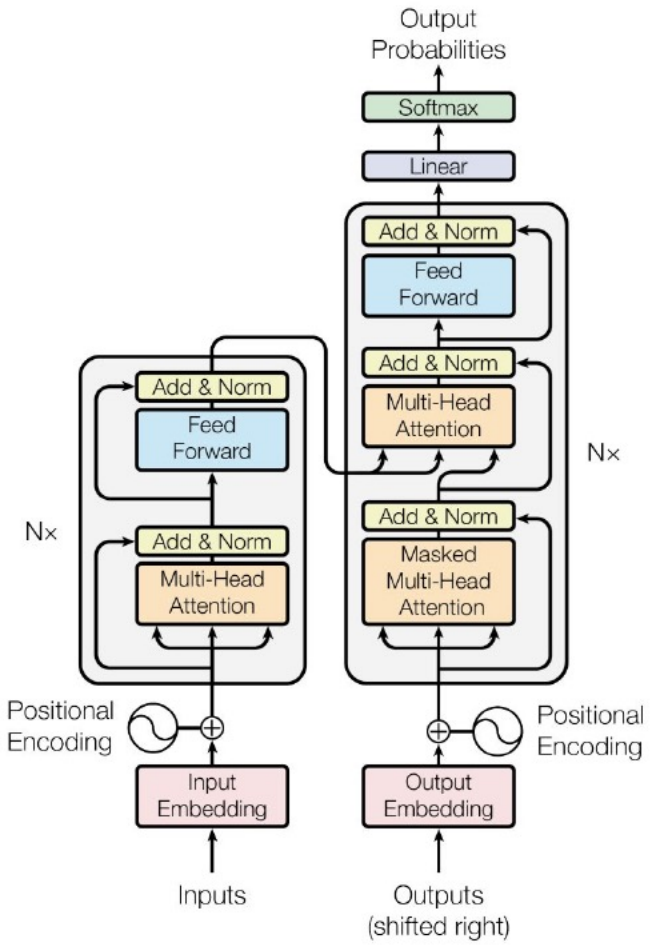
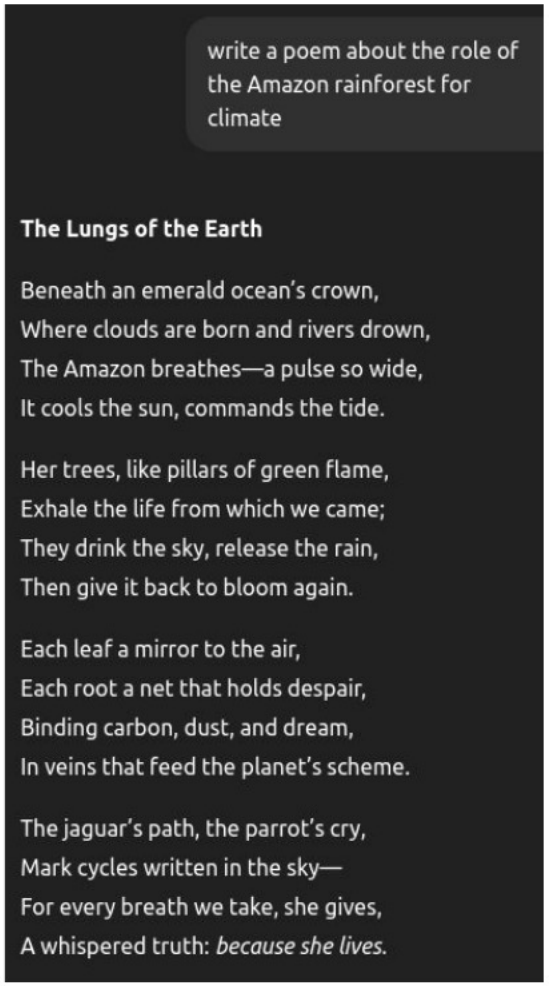


Figure 1: The Transformer - model architecture.

Fonte da figura: Vaswani et al. (2017)



Ideia central: prevê o próximo token a partir dos anteriores (não supervisionado)

Pontos fortes: analisa padrões em dados diversos em escala, muitas vezes com resultados impressionantes

Limitações:

- Exige enormes volumes de dados e computação
- Herda vieses e imprecisões dos dados de treinamento
- Pode alucinar sem estimativas de incerteza

Características principais:

- Interpola padrões
- Correspondência de padrões vs. recuperação direta de fatos
-

Discussão: geração de dados sintéticos

Se um conjunto de dados é sintético (gerado artificialmente), você o consideraria “seguro” para compartilhar publicamente? Por quê?

https://partici.fi/53603205

Roteiro da aula

O que é aprendizado de máquina?

Tipos de aprendizado de máquina

Dados

Entendendo o algoritmo de regressão linear

Perguntas e respostas

Coleta de dados

Preparação de dados para AM

A importância dos dados

- Os dados são o núcleo do aprendizado de máquina
- A qualidade do seu modelo depende da qualidade dos seus dados
- Quantos dados bastam?
  - Depende do problema
  - Existem algumas heurísticas razoáveis, mas quase sempre depende do que funciona empiricamente



Figura de "What is Big Data? A Quick Introduction for Analytics and Data Engineering Beginners"

Coleta e preparação de dados

1. Definir a tarefa
2. Selecionar dados alinhados à tarefa
    a. Os dados são rotulados? Se não, você pode rotulá-los?
3. Explorar e limpar os dados
    a. Tratar valores ausentes (excluir ou imputar)
    b. Verificar a qualidade dos rótulos
4. Dividir os dados em treinamento, validação e teste

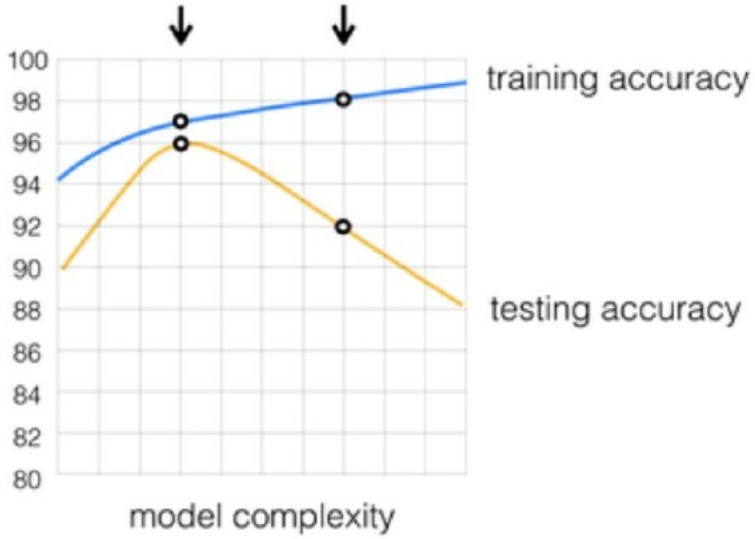
Dividindo o conjunto de dados

- Por que dividir o conjunto de dados?
  - Para evitar sobreajuste e avaliar o desempenho do modelo em dados novos



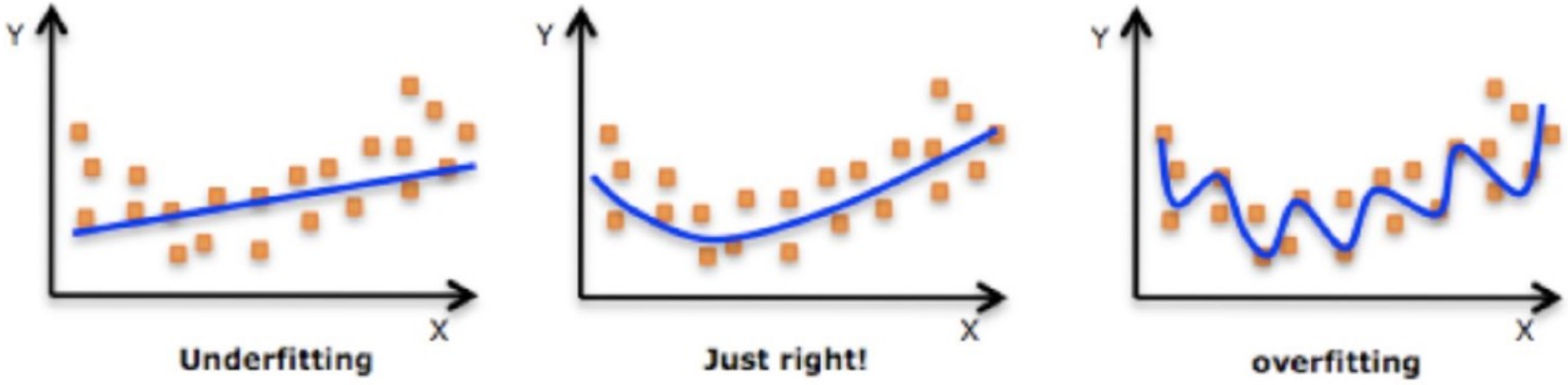
- Treinamento: aprender os parâmetros do modelo
- Validação: escolher hiperparâmetros e comparar modelos
- Teste: avaliar o melhor modelo para medir a generalização em dados não vistos

Sobreajuste vs. generalização



Solução: validação cruzada k-fold

Avaliar como um modelo preditivo vai se sair em dados “não vistos” dividindo repetidamente o conjunto em k subconjuntos de treino e teste.



Definir a tarefa

1. Descrever os requisitos do problema a ser resolvido:
   - Encontrar as restrições temporais e espaciais do problema (que dão valor à solução de AM)

     p. ex. a previsão do tempo pode otimizar a produção renovável, seja para otimizar a infraestrutura existente (curto prazo), seja para definir a localização de novas usinas (longo prazo)
2. Estabelecer as especificações que seu modelo de AM deve cumprir:
   - Quais são as especificações para o treinamento? E para a implantação?
3. Formular a tarefa de AM, por exemplo:

Classificação

Regressão

Agrupamento

Segmentação de imagens

Aumento de dados

Detecção de anomalias

Otimização online

Seleção de dados

- Selecionar dados alinhados à tarefa de AM
  - Revisando aplicações de ponta
  - Consultando especialistas do domínio (pesquisadores/profissionais da área)
  - Avaliando com cuidado os requisitos de dados da tarefa de AM

    Observação: talvez seja preciso revisar/flexibilizar algumas especificações conforme os dados realmente disponíveis.

- O conjunto de dados é rotulado? Se não, você pode rotulá-lo? Vale a pena anotar para resolver o problema, considerando abordagens alternativas?

Explorar e limpar os dados

1. Remover observações duplicadas ou irrelevantes
2. Corrigir erros estruturais
3. Filtrar outliers indesejados
4. Tratar dados ausentes
5. Validar e garantir a qualidade

De “Data Cleaning: Definition, Benefits, And How-To” – Tableau

Discussão: propriedade dos dados

A quem deveriam pertencer os dados usados para treinar modelos de IA: ao usuário, à empresa ou a outra pessoa?

https://partici.fi/53603205

Resumo até aqui

📝 O aprendizado de máquina é útil para problemas específicos.

- Não é uma solução mágica.

📝 Existem muitos tipos diferentes de aprendizado de máquina.

- Abordagens: supervisionado, não supervisionado, por reforço
- Modalidades de dados: PLN, visão computacional, séries temporais
- Requisitos: p. ex. interpretável e explicável, informado pela física
- Outras escolhas: generativo vs. discriminativo, grande vs. pequeno, geral vs. específico

📝 Os dados são o núcleo do aprendizado de máquina!

Exercício: sim ou não

Do ponto de vista técnico, inteligência artificial (IA) refere-se principalmente a grandes modelos generativos de uso geral, como os grandes modelos de linguagem.

https://partici.fi/53603205

Roteiro da aula

O que é aprendizado de máquina?

Tipos de aprendizado de máquina

Dados

Entendendo o algoritmo de regressão linear

Perguntas e respostas

- Equação de uma reta
- Função de perda
- Otimizando os parâmetros

Equação de uma reta

$$y = mx + c$$

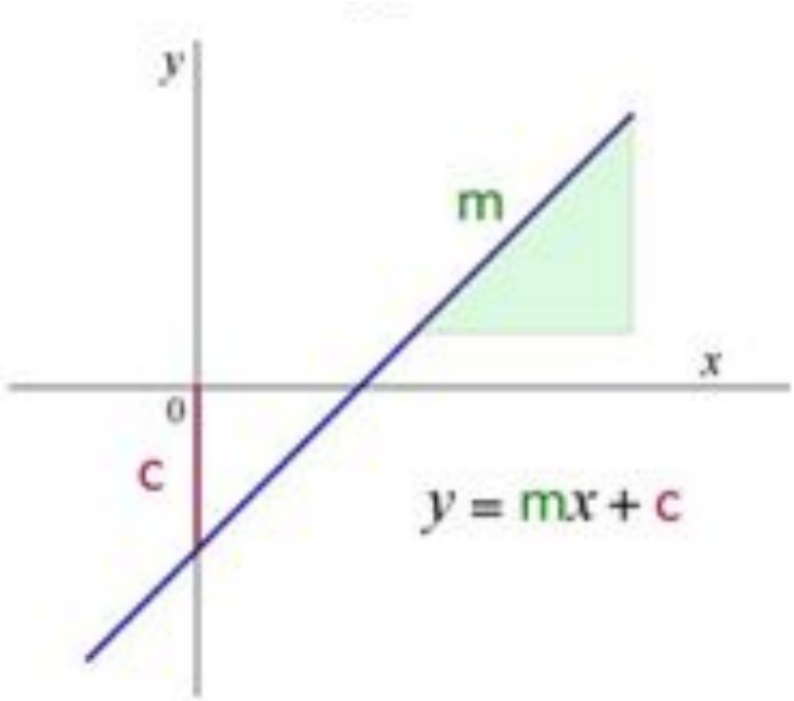
Coeficiente angular

Intercepto em y

Equação de uma reta

$$y = mx + c$$

Coeficiente angular **Intercepto em y**

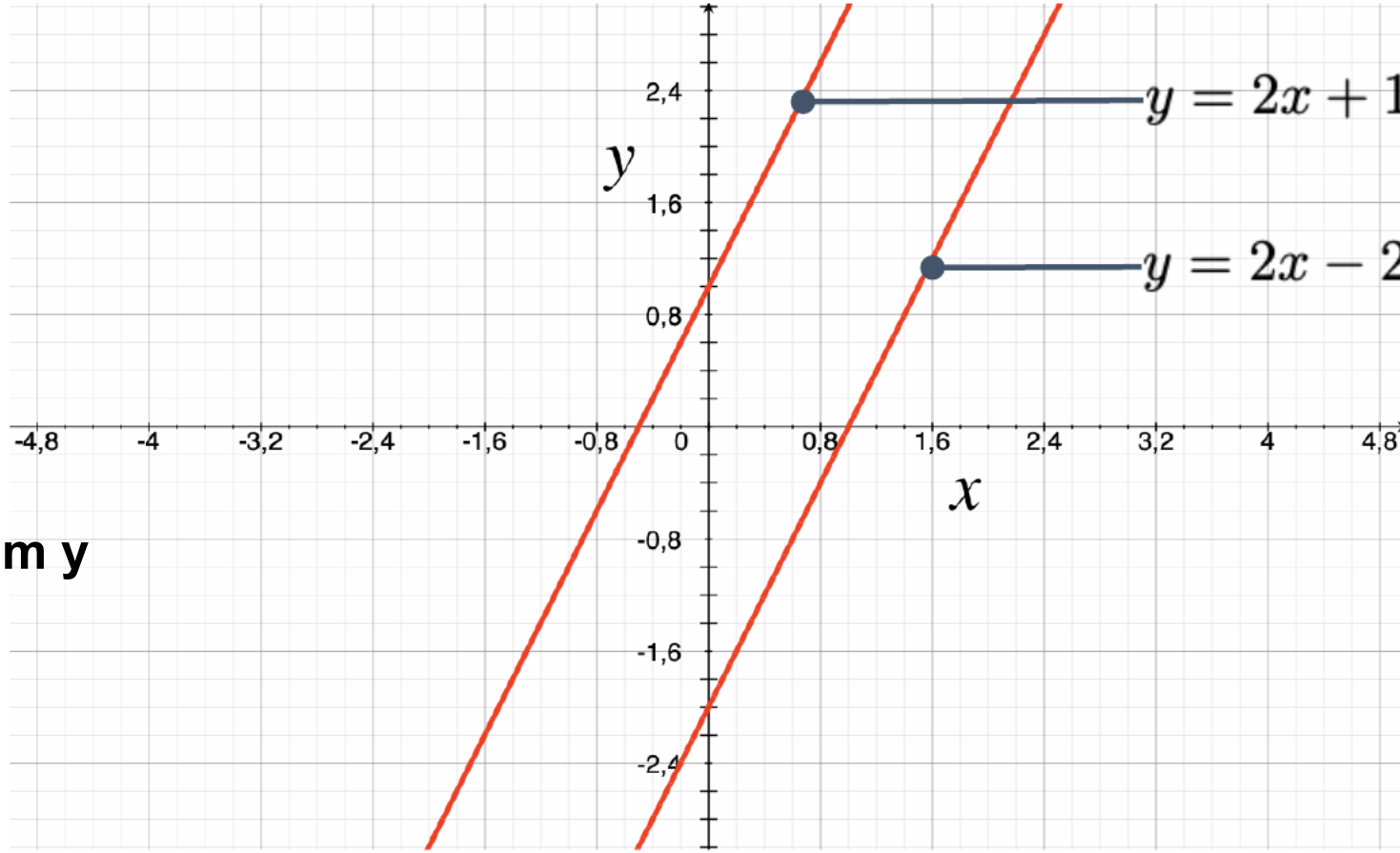


https://www.easycalculation.com/maths-dictionary/equation_of_a_straight_line.html

Equação de uma reta

$$y = mx + c$$

Coeficiente angular **Intercepto em y**

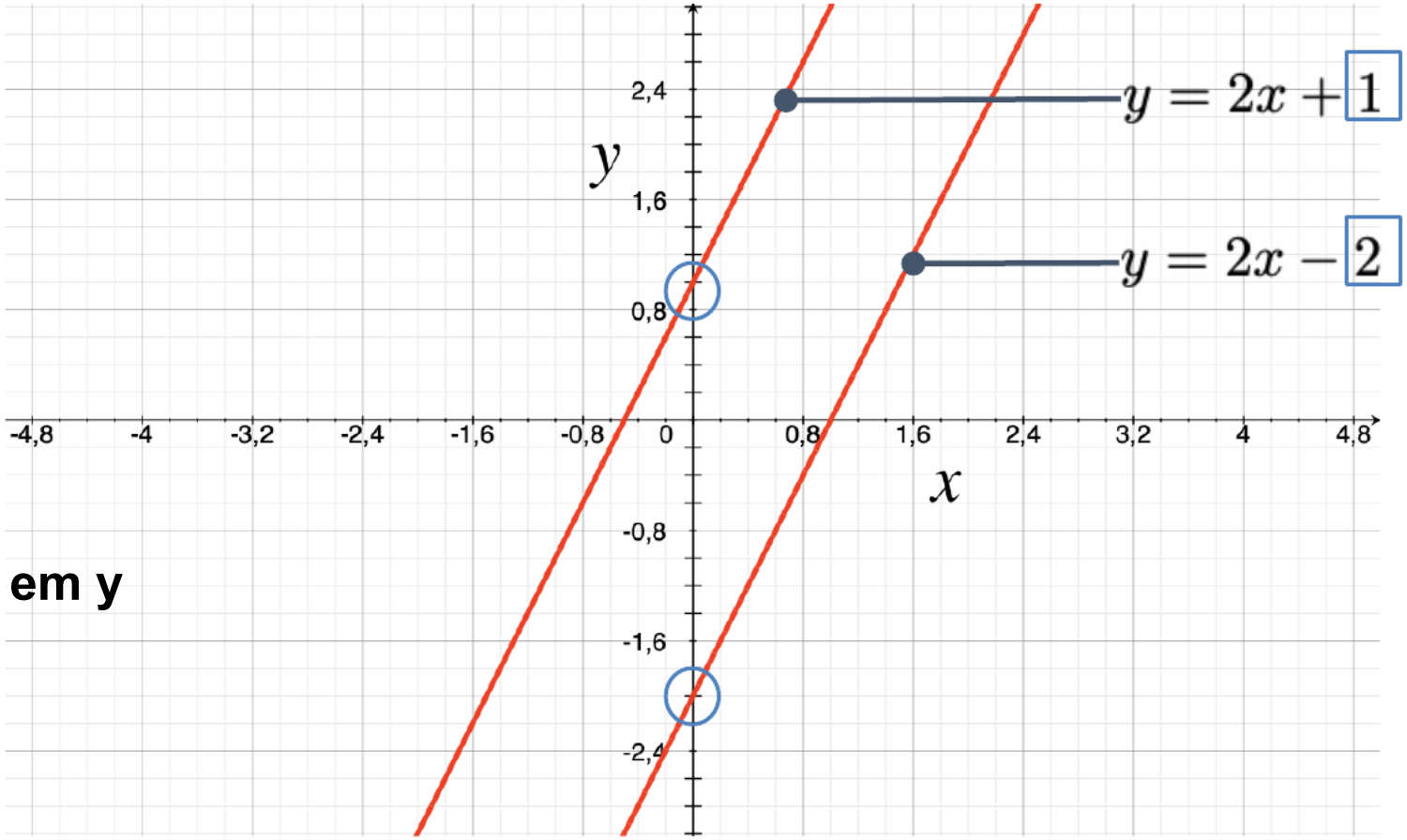


https://www.easycalculation.com/maths-dictionary/equation_of_a_straight_line.html

Equação de uma reta

$$y = mx + c$$

Intercepto em y

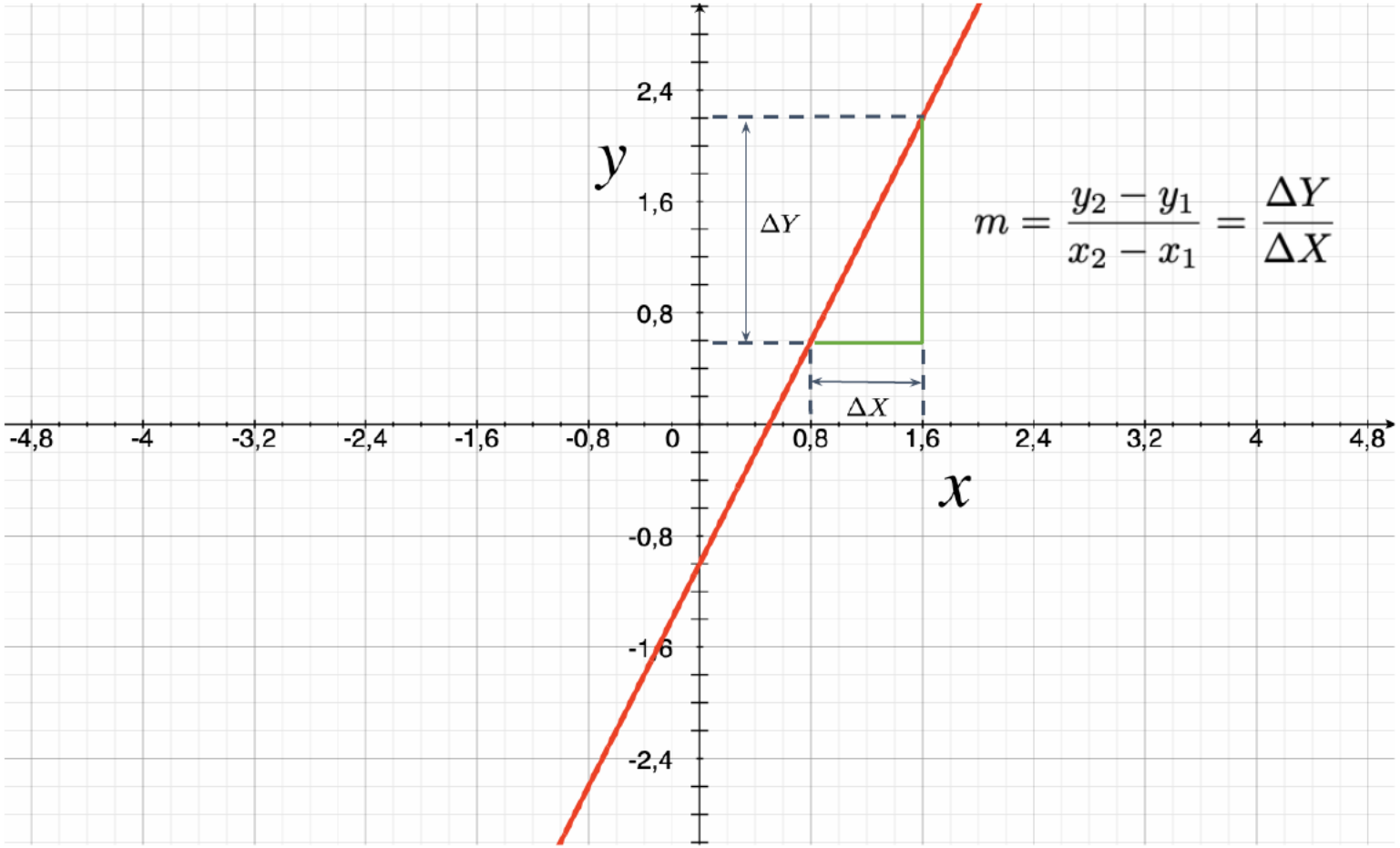


https://www.easycalculation.com/maths-dictionary/equation_of_a_straight_line.html

Equação de uma reta

$$y = mx + c$$

Coeficiente angular

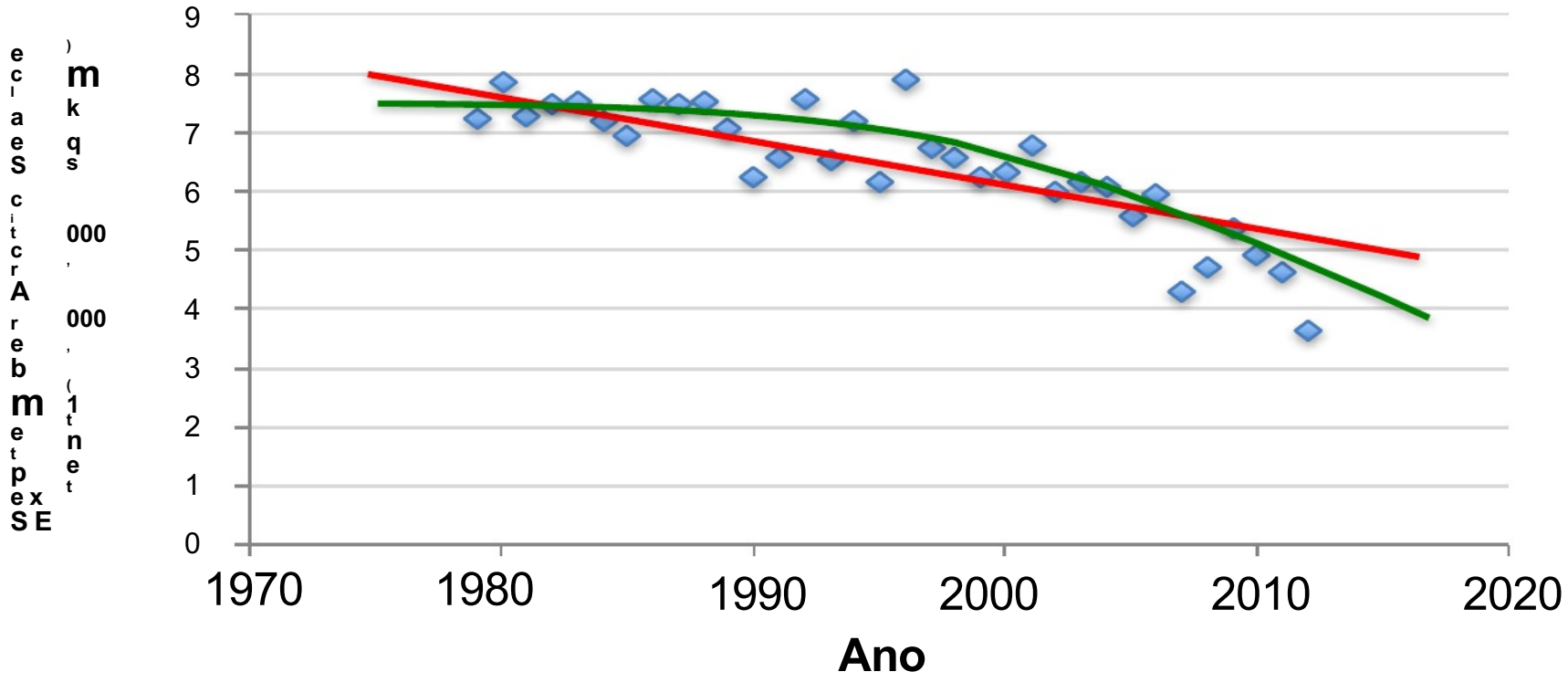


https://www.easycalculation.com/maths-dictionary/equation_of_a_straight_line.html

Prevendo a extensão do gelo marinho do Ártico

Aprender uma função f que preveja y a partir de X, ou seja, f(X) $\cong$ y .

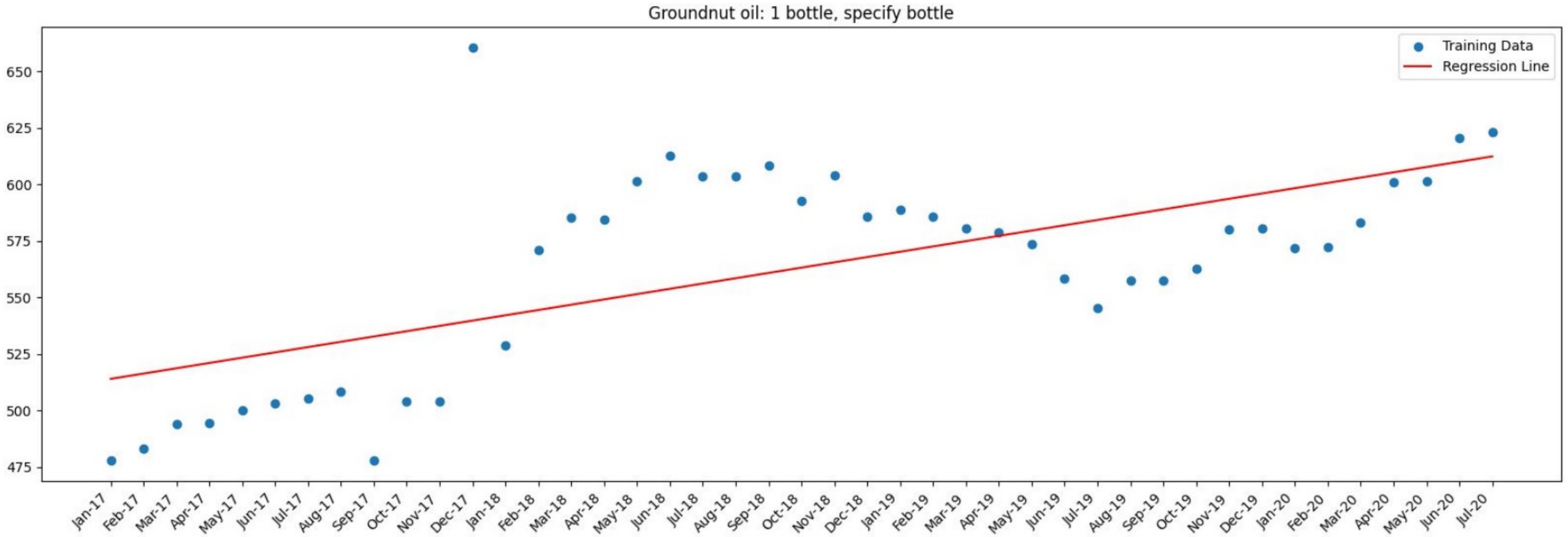
- *Regressão: y tem valor real*



Dados de G. Witt. Journal of Statistics Education, volume 21, número 1 (2013)

Prevendo o preço do óleo de amendoim

eba



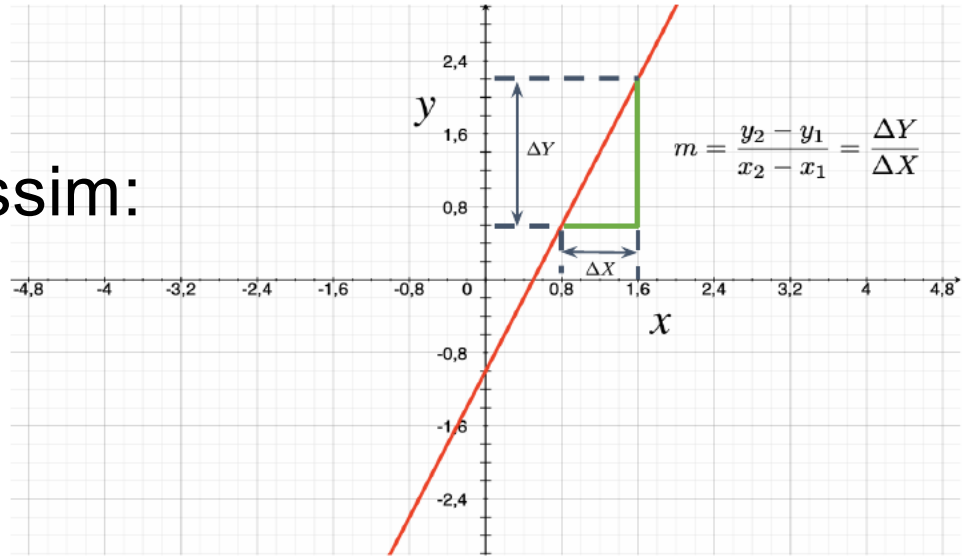
atributo

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

Regressão linear

A regressão linear supõe que existe uma relação linear entre os atributos e o rótulo numérico.

$$y = mx + c$$



Em AM, costumamos escrever a equação assim:

$$y = b + w_1 x_1$$

onde:

é o rótulo

é o viés (intercepto em y), às vezes escrito como w_0

é o peso do atributo x_1

Modelo de regressão linear

O modelo de regressão linear prevê uma saída com base nos atributos de entrada $\hat{y}$

$x_1, x_2, \cdots, x_n$

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

onde:

- $\hat{y}$ é a previsão
- w_0 é o intercepto em y (viés)
- $w_1, w_2, \cdots, w_n$ são os pesos (coeficientes)
- $x_1, x_2, \cdots, x_n$ são os atributos de entrada

Modelo de regressão linear: função de perda

Para medir a diferença entre as previsões e os valores reais $\hat{y}$ y (rótulos)

usamos o erro quadrático médio (EQM)

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

onde

m é o número de exemplos de treinamento

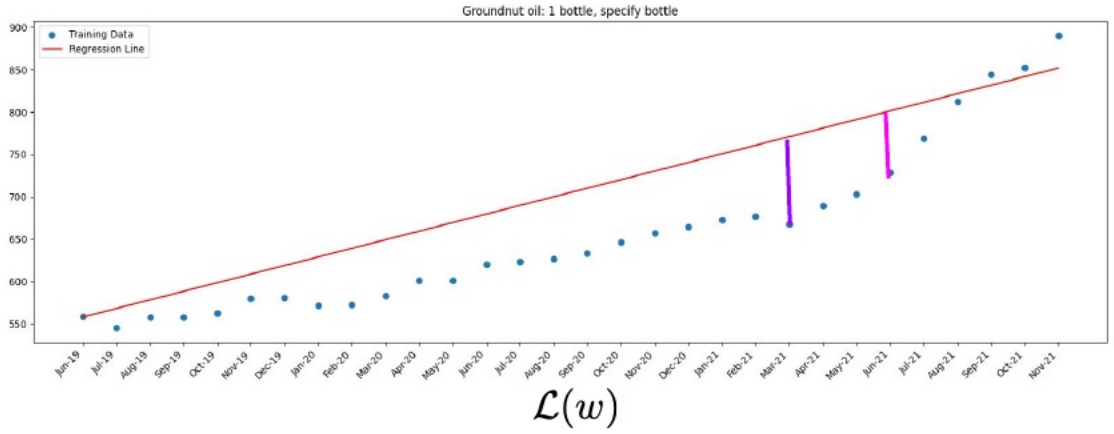
$\mathcal{L}(w)$ é a função de perda

Modelo de regressão linear: função de perda

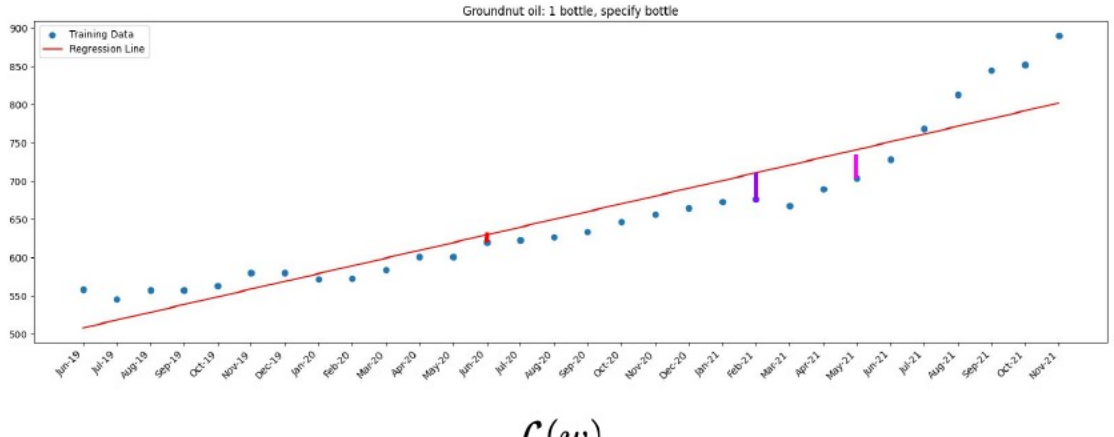
- A perda é uma penalidade por uma previsão ruim: é um número que indica o quanto a previsão do modelo errou em um ponto de dados. Se a previsão for perfeita, a perda é zero; caso contrário, não é zero

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

- Treinar um modelo de regressão linear significa aprender, com exemplos rotulados, valores dos pesos que reduzem a perda



$\mathcal{L}(w)$



$\mathcal{L}(w)$

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

Modelo de regressão linear: peso ótimo (minimizando a perda)

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

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

substituir

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

O objetivo é encontrar $w_0, w_1, w_2, \cdots, w_n$ que minimizem $\mathcal{L}(w)$

Modelo de regressão linear: peso ótimo (minimizando a perda)

Podemos fazer isso de duas formas:

- Solução fechada (equação normal)

- Otimização por gradiente descendente

Otimizando os parâmetros: equação normal

Se X é a matriz de projeto (incluindo o termo de viés) e y é o vetor-alvo, os pesos ótimos são dados por

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

Só funciona quando $X^T X$ é invertível e para pequenos conjuntos de dados

Otimizando os parâmetros: gradiente descendente

Os pesos são atualizados iterativamente usando o gradiente da função de perda

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

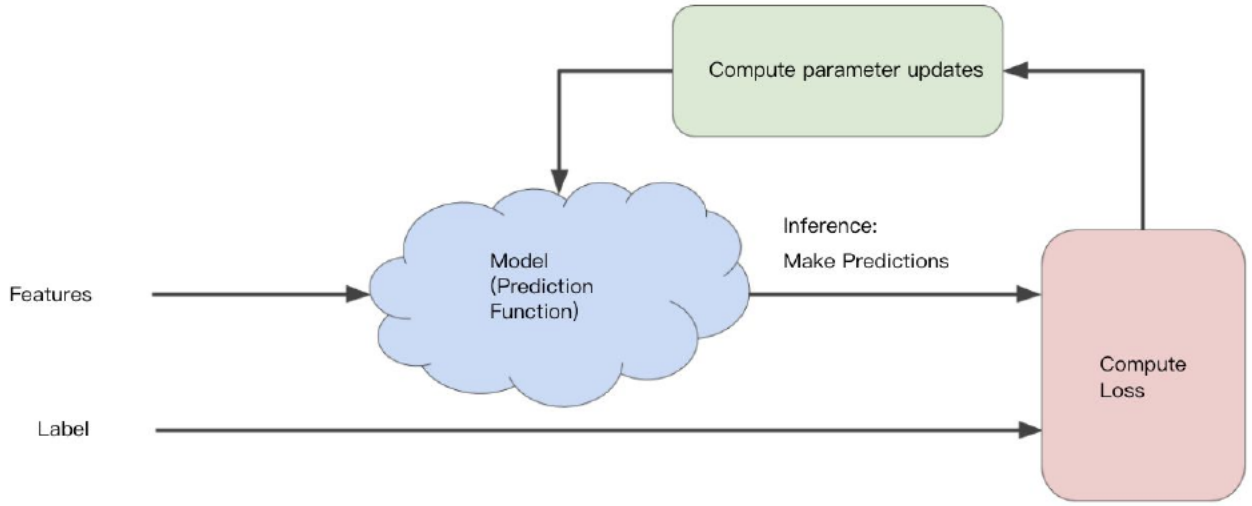
onde é a taxa de aprendizado e α

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

Este método é usado em grandes conjuntos de dados

Encontrando o mínimo de uma função

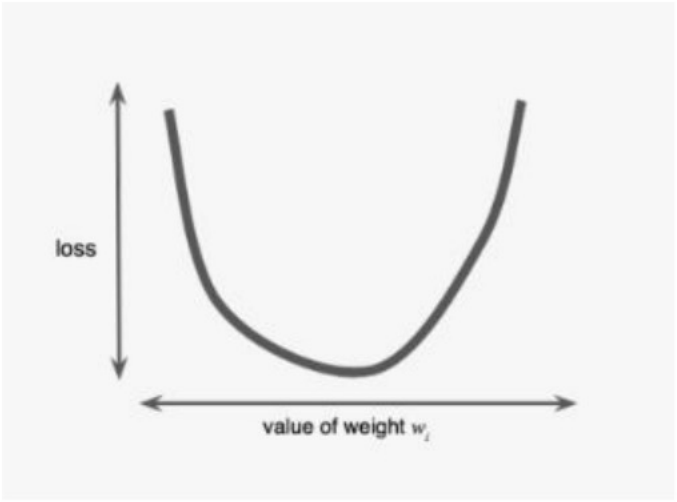
Começamos o treinamento com um palpite inicial para pesos e vieses e ajustamos os valores iterativamente até obter os pesos e o viés com a menor perda

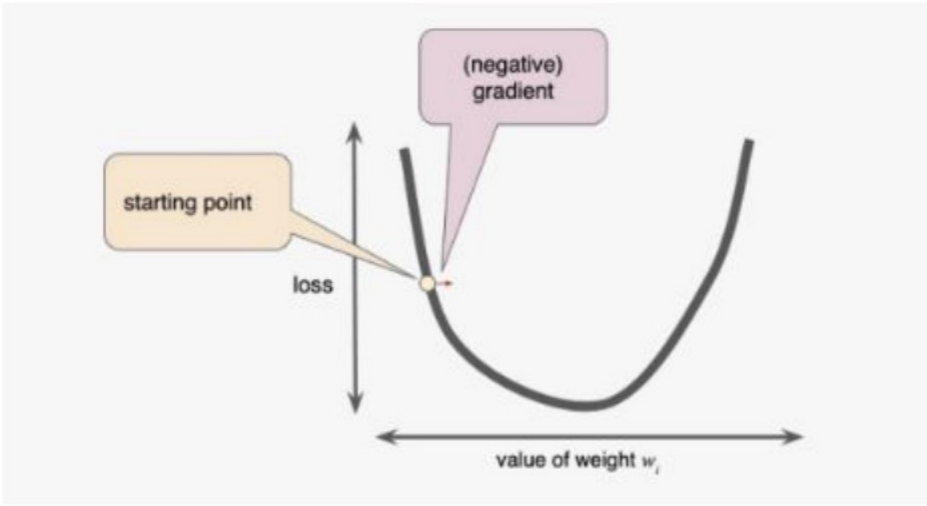


Do jeito do gradiente descendente!

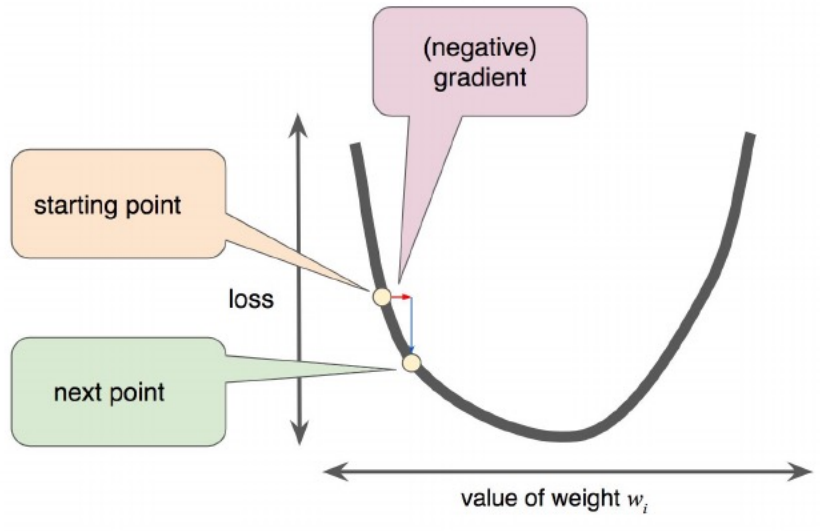
https://github.com/litaotao/machine-learning-crash-course

Encontrando o mínimo de uma função





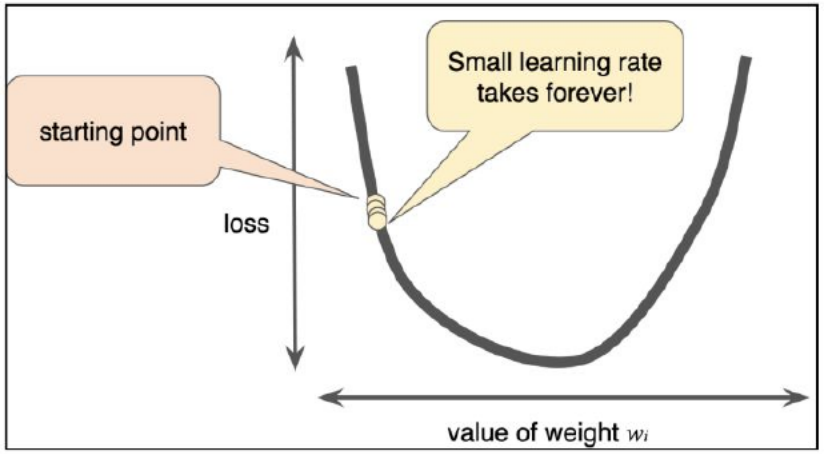
Do jeito do
gradiente descendente!

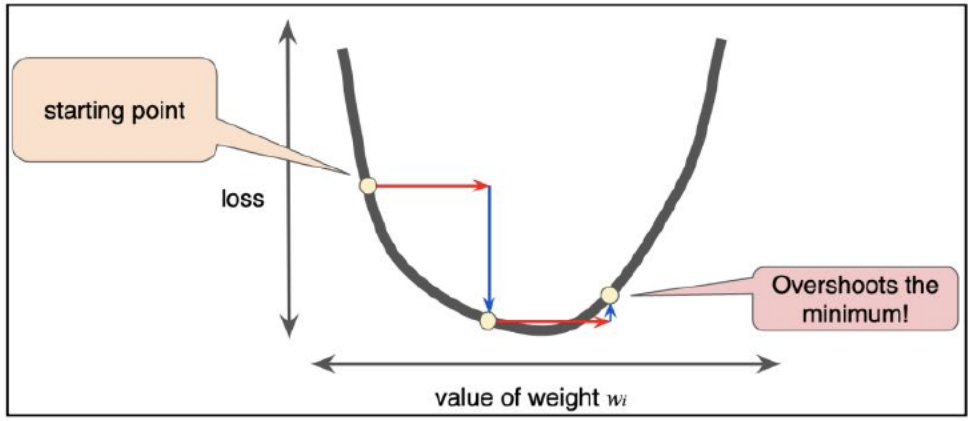


$$w_j := w_j - \alpha \frac{\partial \mathcal{L}}{\partial w_j}$$

A taxa de aprendizado importa

A taxa de aprendizado é um hiperparâmetro que determina o tamanho dos passos dados a cada iteração ao atualizar os parâmetros do modelo durante o treinamento

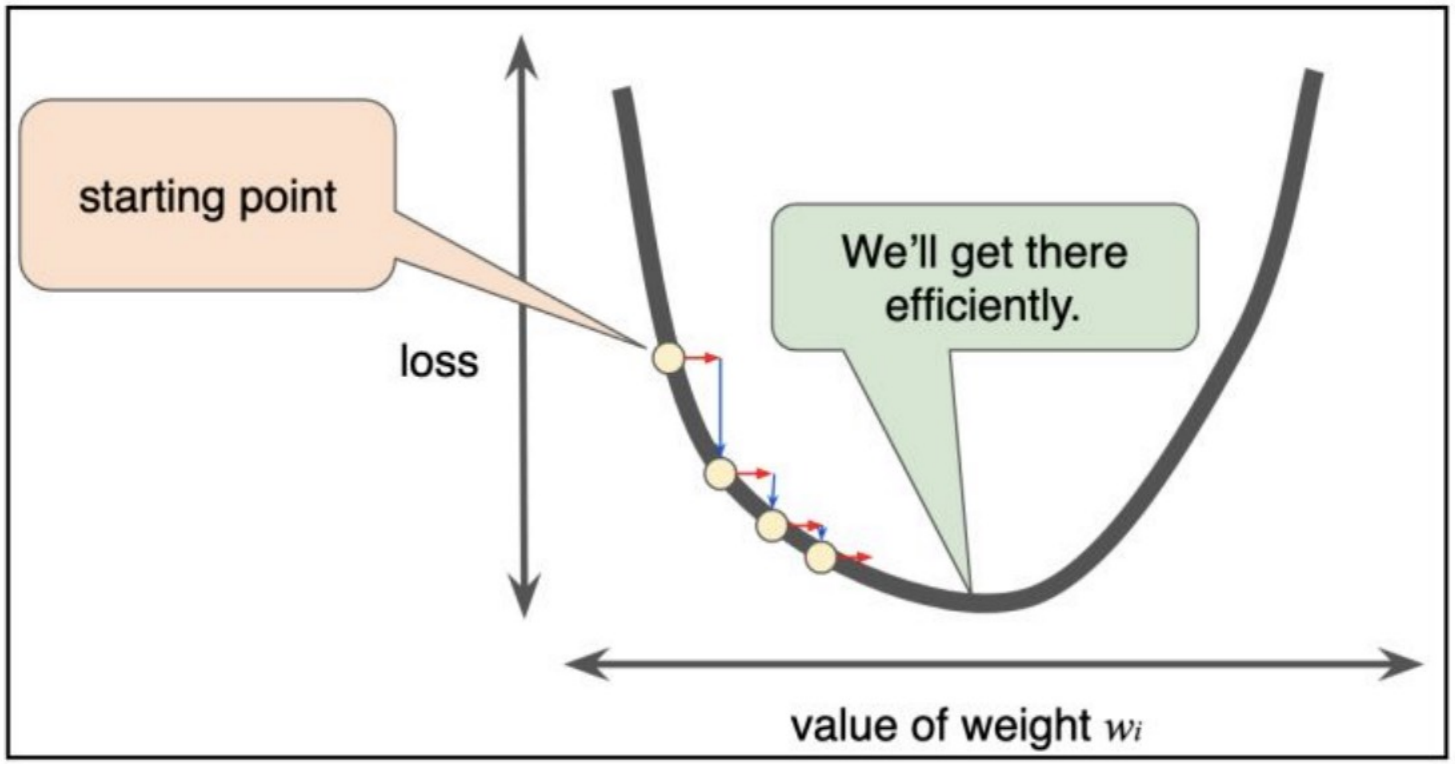




$$w_j := w_j - \alpha \frac{\partial \mathcal{L}}{\partial w_j}$$

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

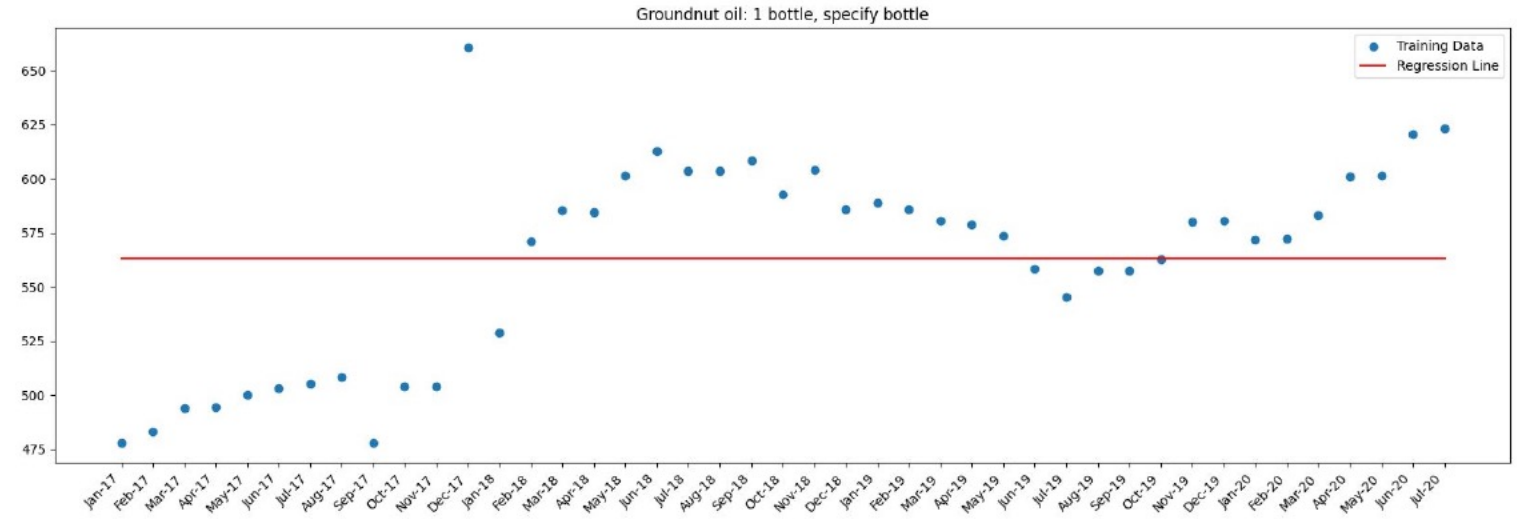
A taxa de aprendizado importa



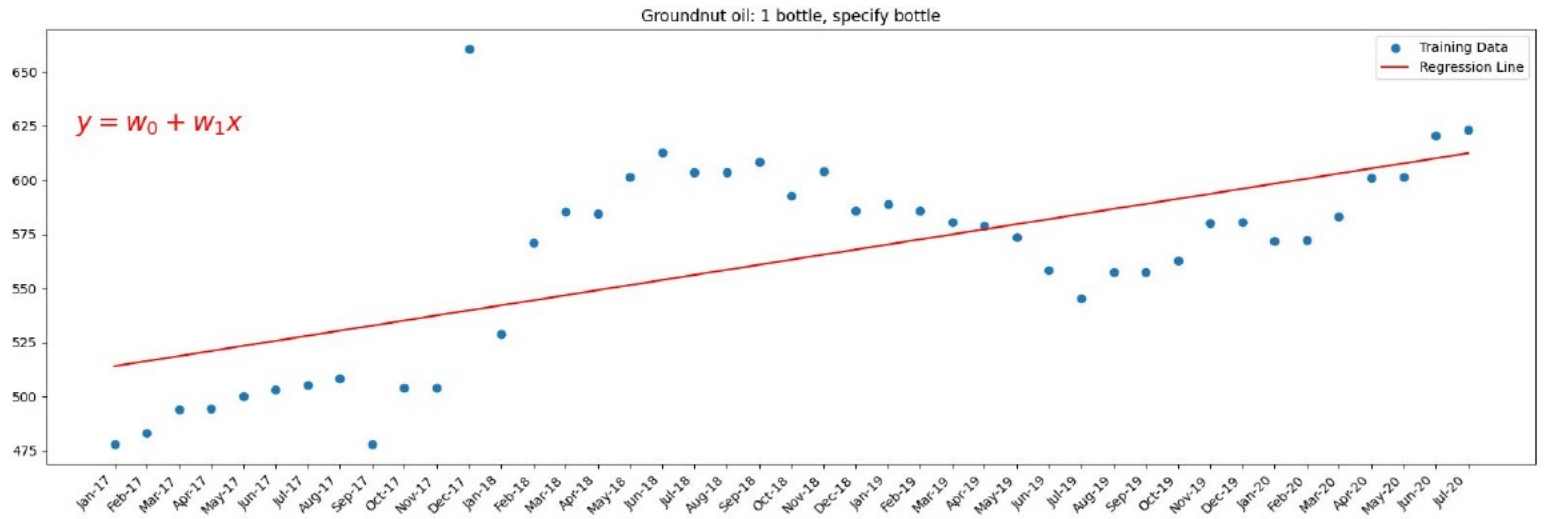
https://cedar.buffalo.edu/~srihari/CSE676/8.3%20BasicOptimizn.pdf

Ajustando uma reta aos dados do óleo de amendoim

Qual modelo se ajusta melhor?



(A)



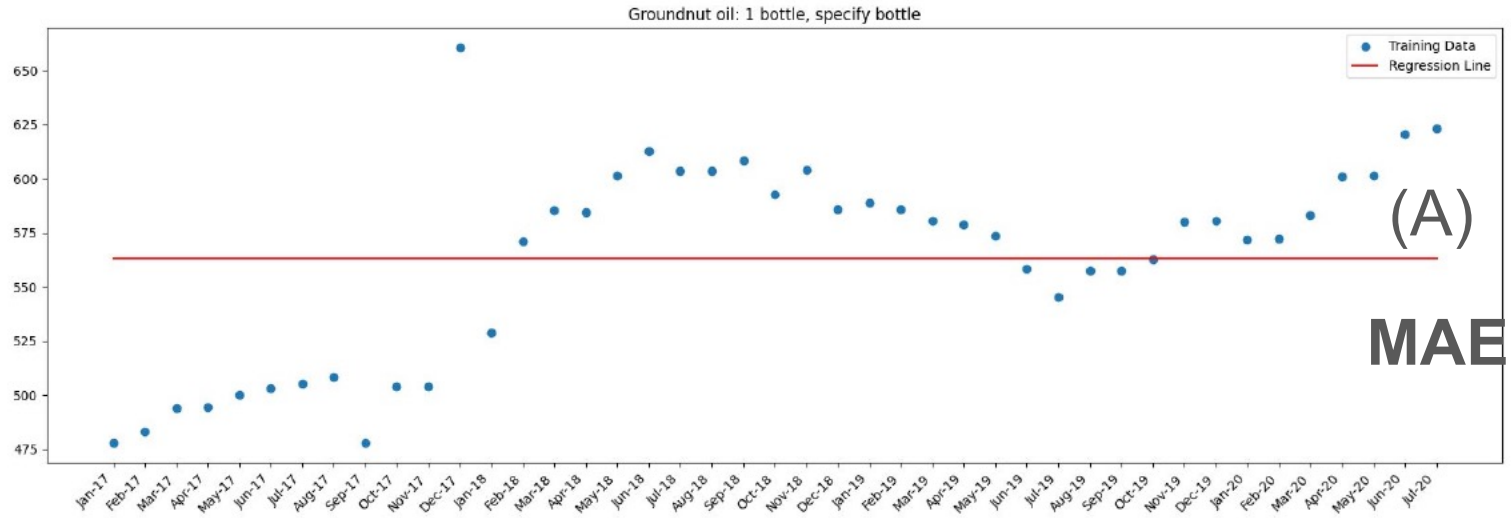
(B)

Avaliando seu modelo: MAE

$$MAE = \frac{1}{n}\sum \left| y - \hat{y} \right|$$

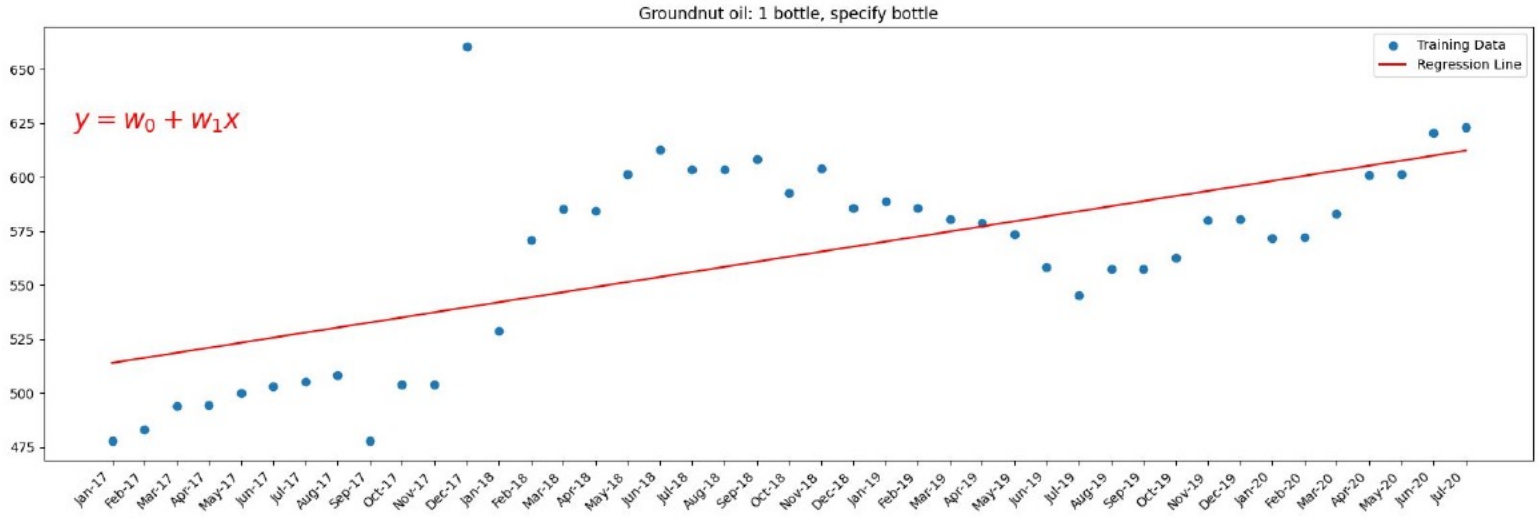
Divide by the total number of data points. Predicted output value. Actual output value. Sum of. The absolute value of the residual.

Qual modelo se ajusta melhor?



(A)

MAE = 47,20

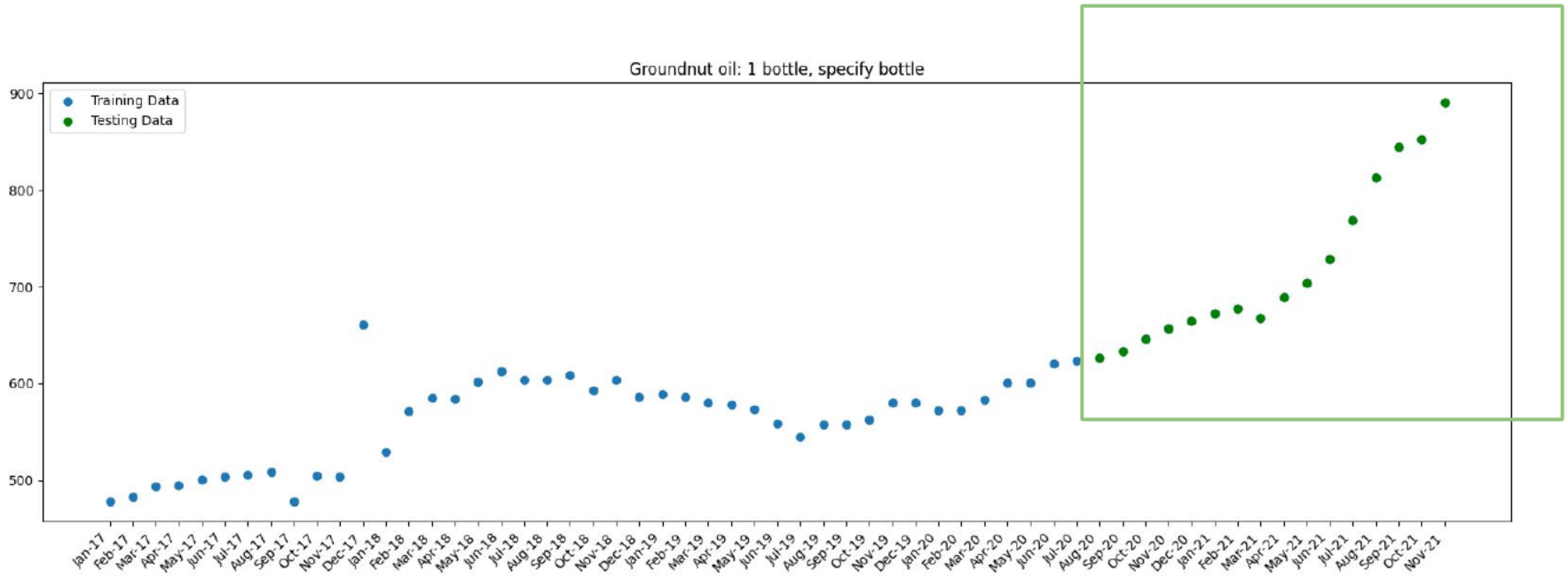


(B)

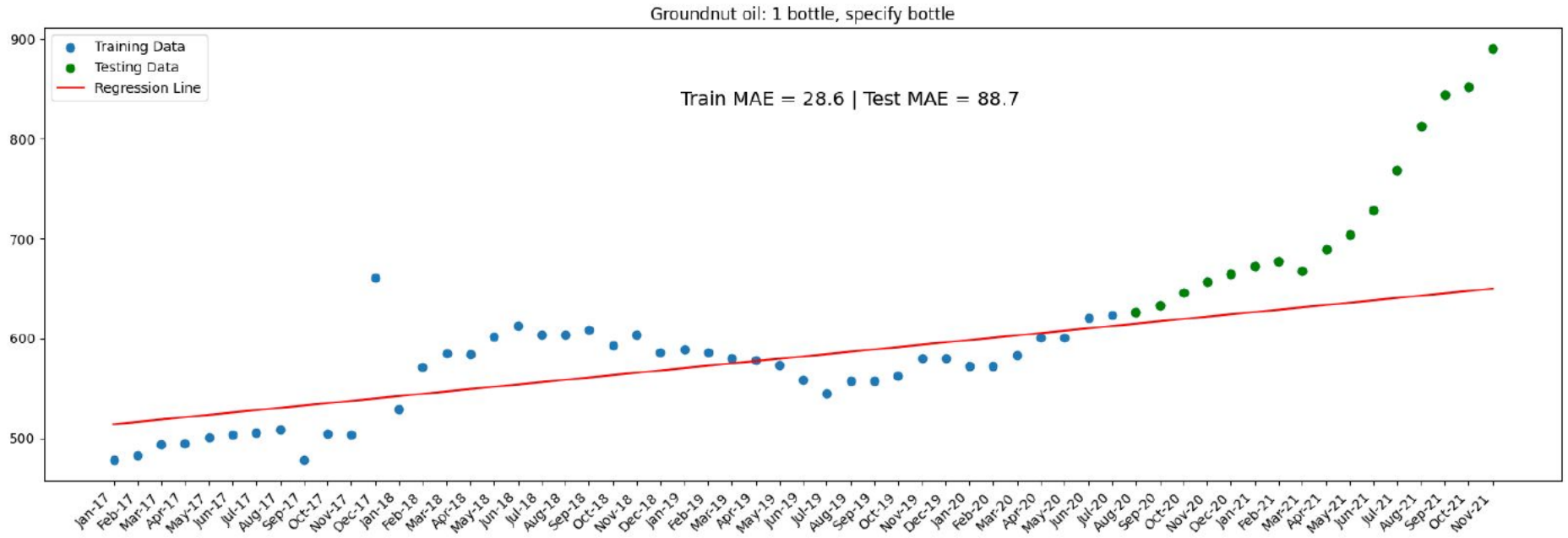
MAE = 28,63

Lembre-se: a avaliação real é feita com dados “não vistos”

Você precisa avaliar o desempenho do seu modelo em dados “não vistos”.



Ajustando uma reta aos dados do óleo de amendoim: avaliação nos dados de teste



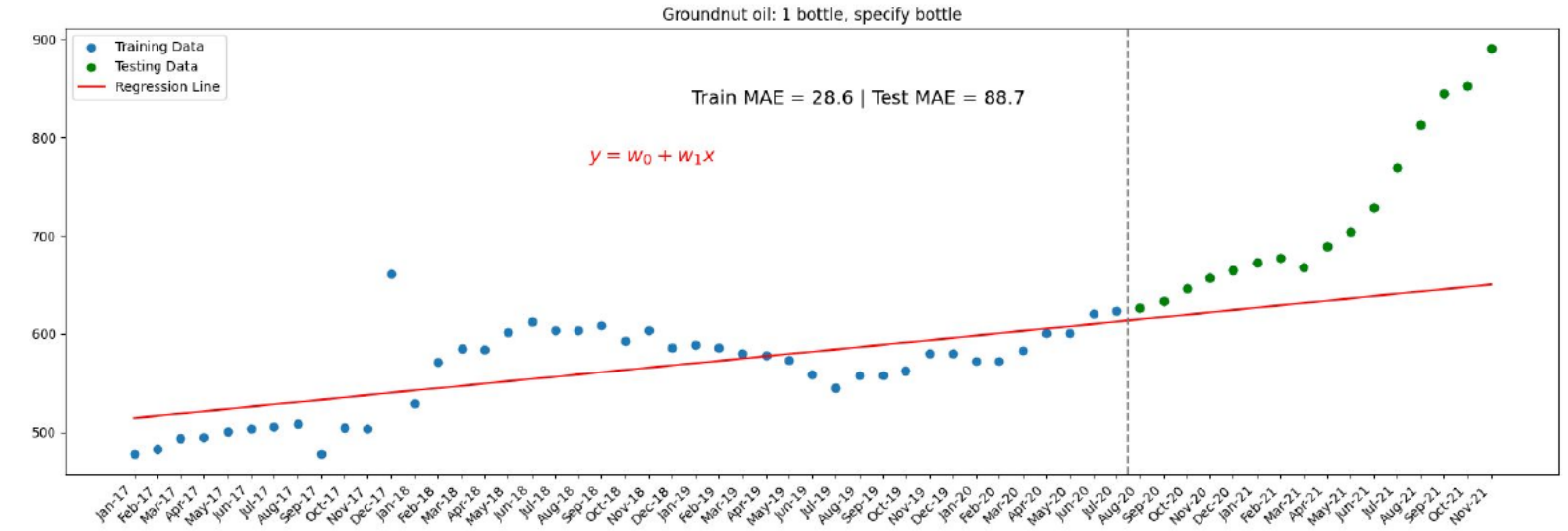
Isso é o melh
que podemos fa

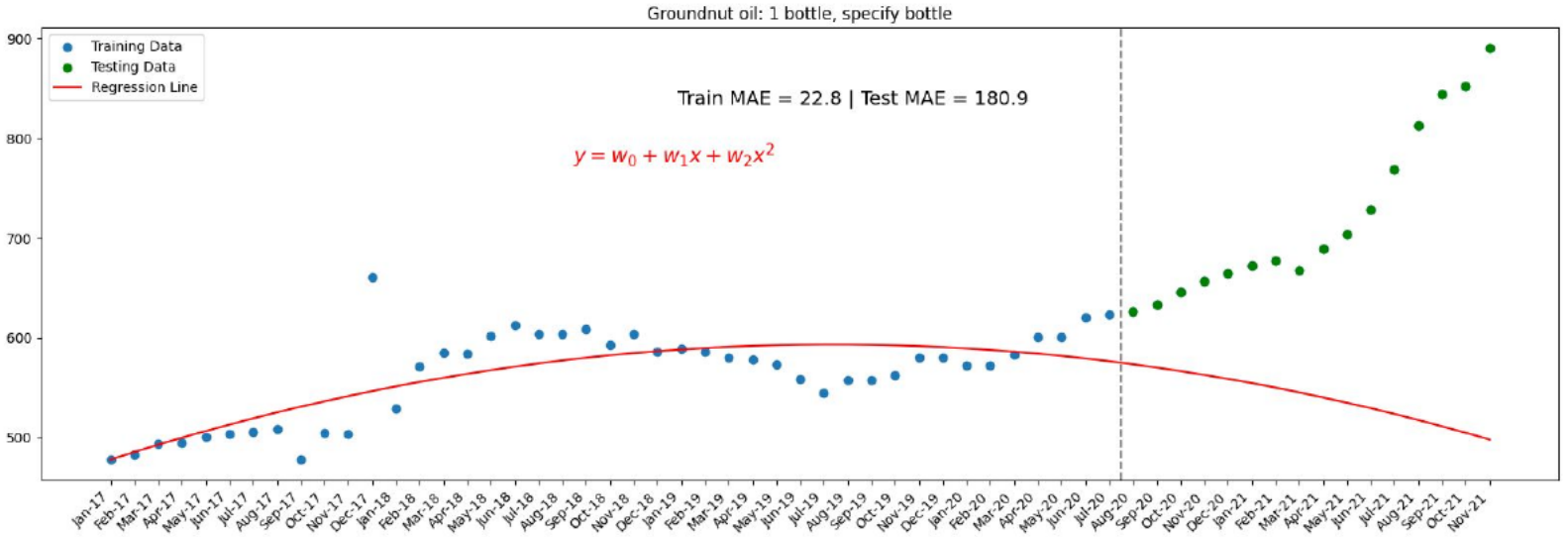
Engenharia de atributos: modelando relações não lineares

A engenharia de atributos polinomiais é o truque de alimentar um modelo linear com potências (e produtos) das entradas originais

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

para que um modelo que só traça retas consiga ajustar relações curvas, já que continua linear nos pesos, embora as entradas agora sejam curvas.



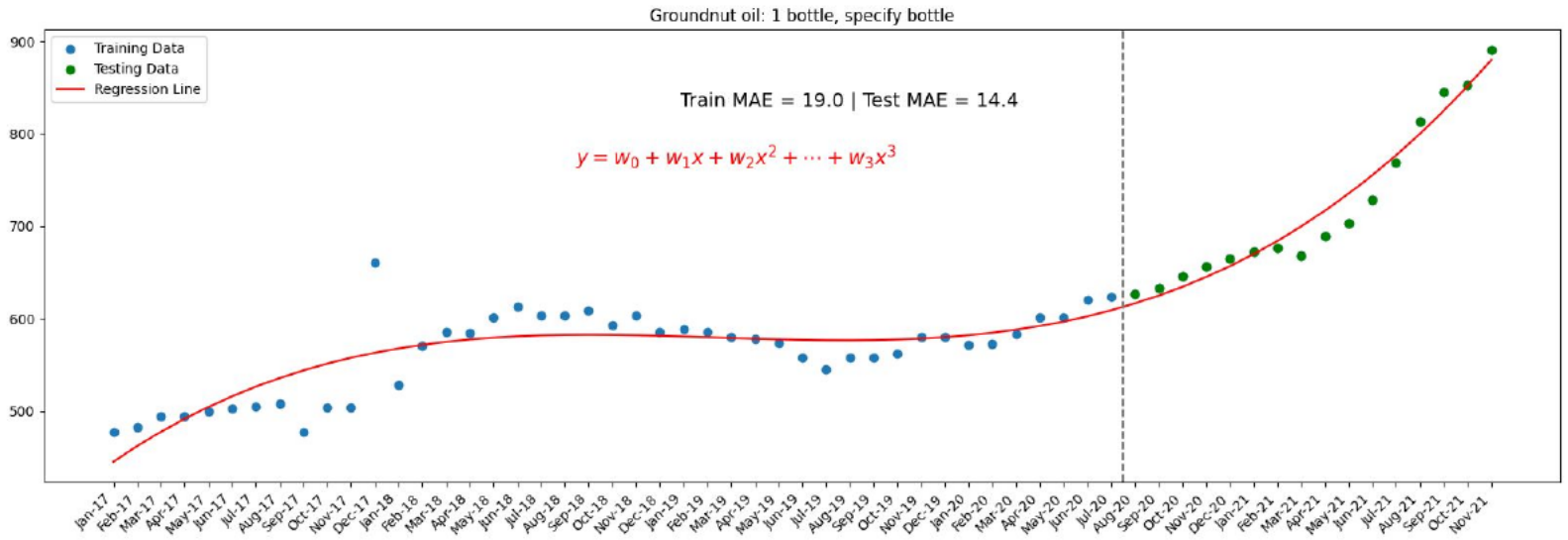


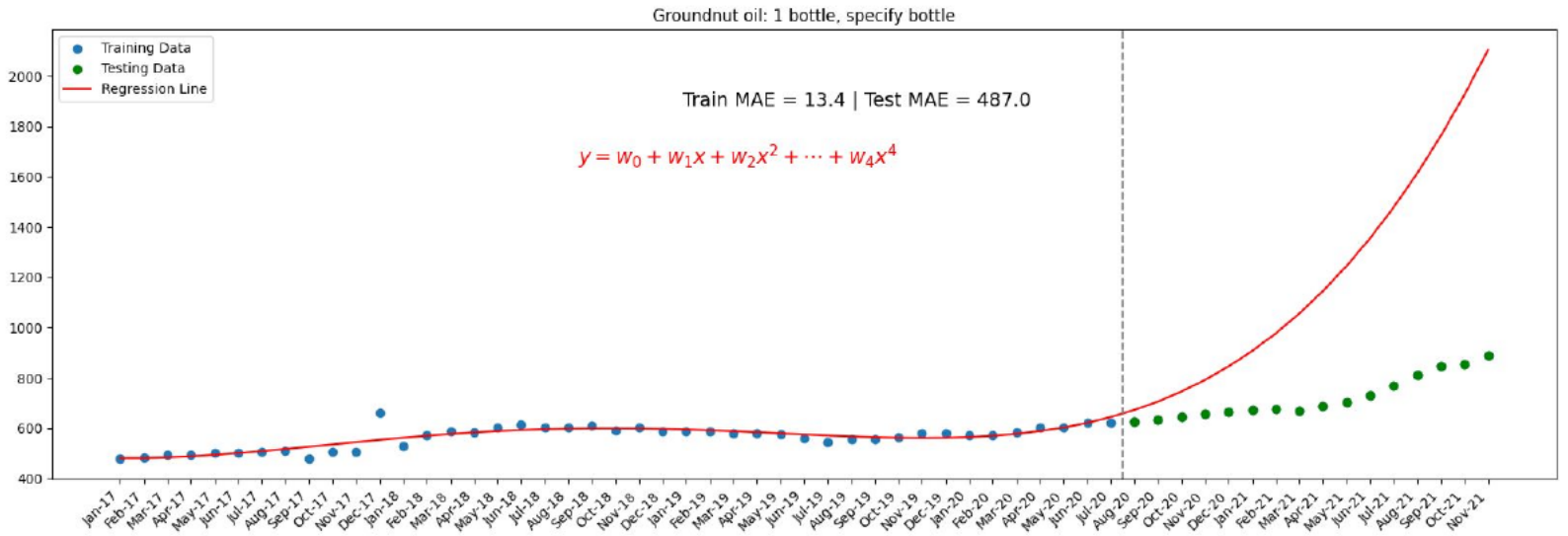
Engenharia de atributos: atributos polinomiais

A engenharia de atributos polinomiais é o truque de alimentar um modelo linear com potências (e produtos) das entradas originais

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

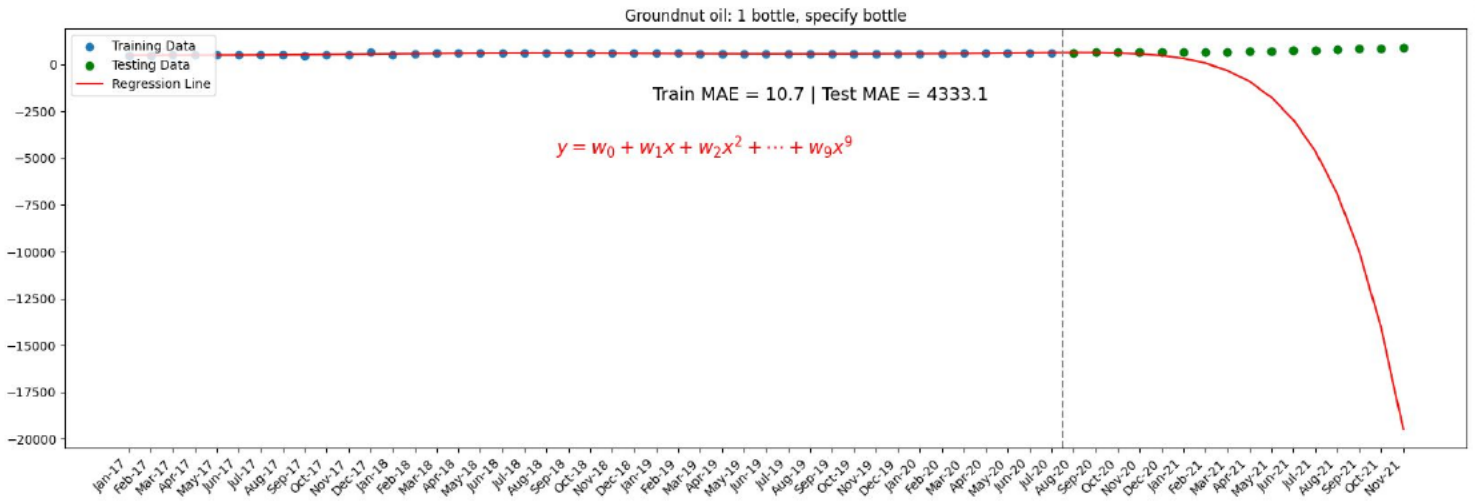
para que um modelo que só traça retas consiga ajustar relações curvas, já que continua linear nos pesos, embora as entradas agora sejam curvas.





Exercício: sobreajuste vs. generalização

Um modelo de aprendizado de máquina obtém erro absoluto médio de 10 nairas no conjunto de treinamento, mas de 4.333 nairas no conjunto de teste. Qual é a explicação mais provável?



https://partici.fi/53603205

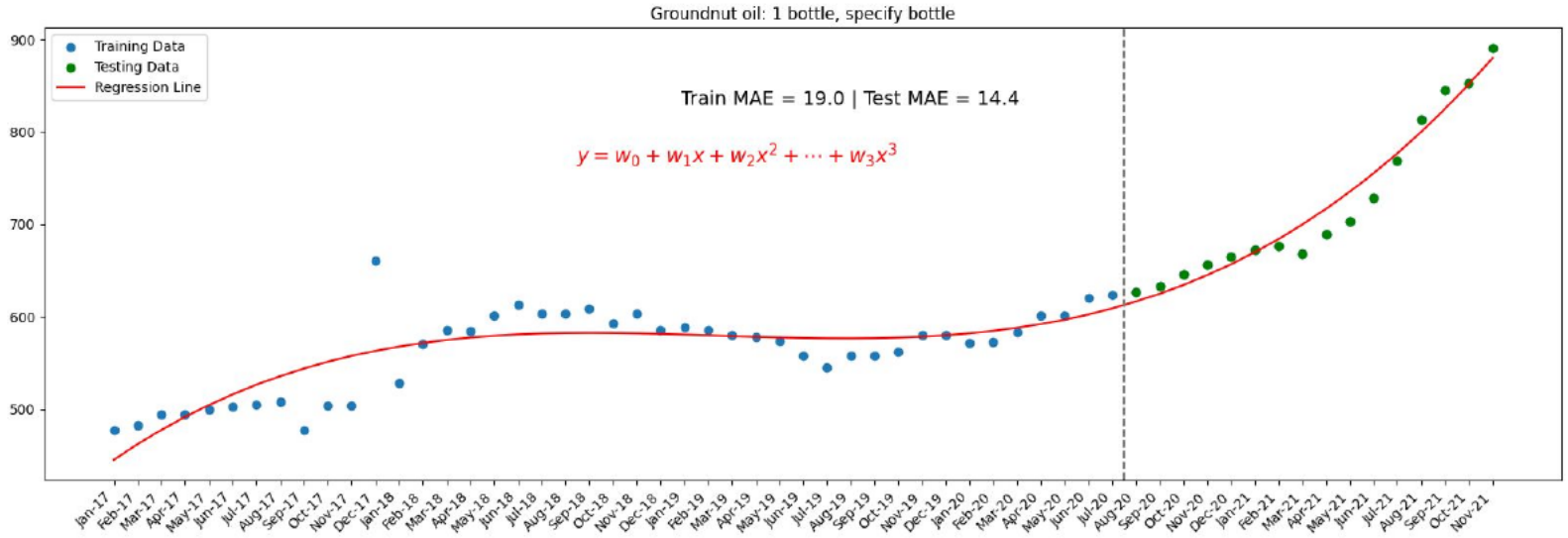
Principais pressupostos da regressão linear

Linearidade: a relação real entre o atributo e o rótulo é aproximadamente linear

Independência: as observações são independentes entre si

Homocedasticidade: a variância da diferença entre o rótulo real e o previsto (resíduo) permanece constante para todos os valores dos atributos

Ausência de multicolinearidade: os atributos de entrada não são fortemente correlacionados



Paramos por aqui hoje!
Obrigada

Você pode me contatar:

- E-mail: tejumade@tri-ai.org
- Google Scholar, LinkedIn: tejumade Afonja
- X, Bluesky, GitHub: tejuafonja
- Site: tejuafonja.com
- Site da minha ONG: tri-ai.org

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

Perguntas e respostas

Você pode me contatar:

- E-mail: tejumade@tri-ai.org
- Google Scholar, LinkedIn: tejumade Afonja
- X, Bluesky, GitHub: tejuafonja
- Site: tejuafonja.com
- Site da minha ONG: tri-ai.org

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