Introdução ao Machine Learning

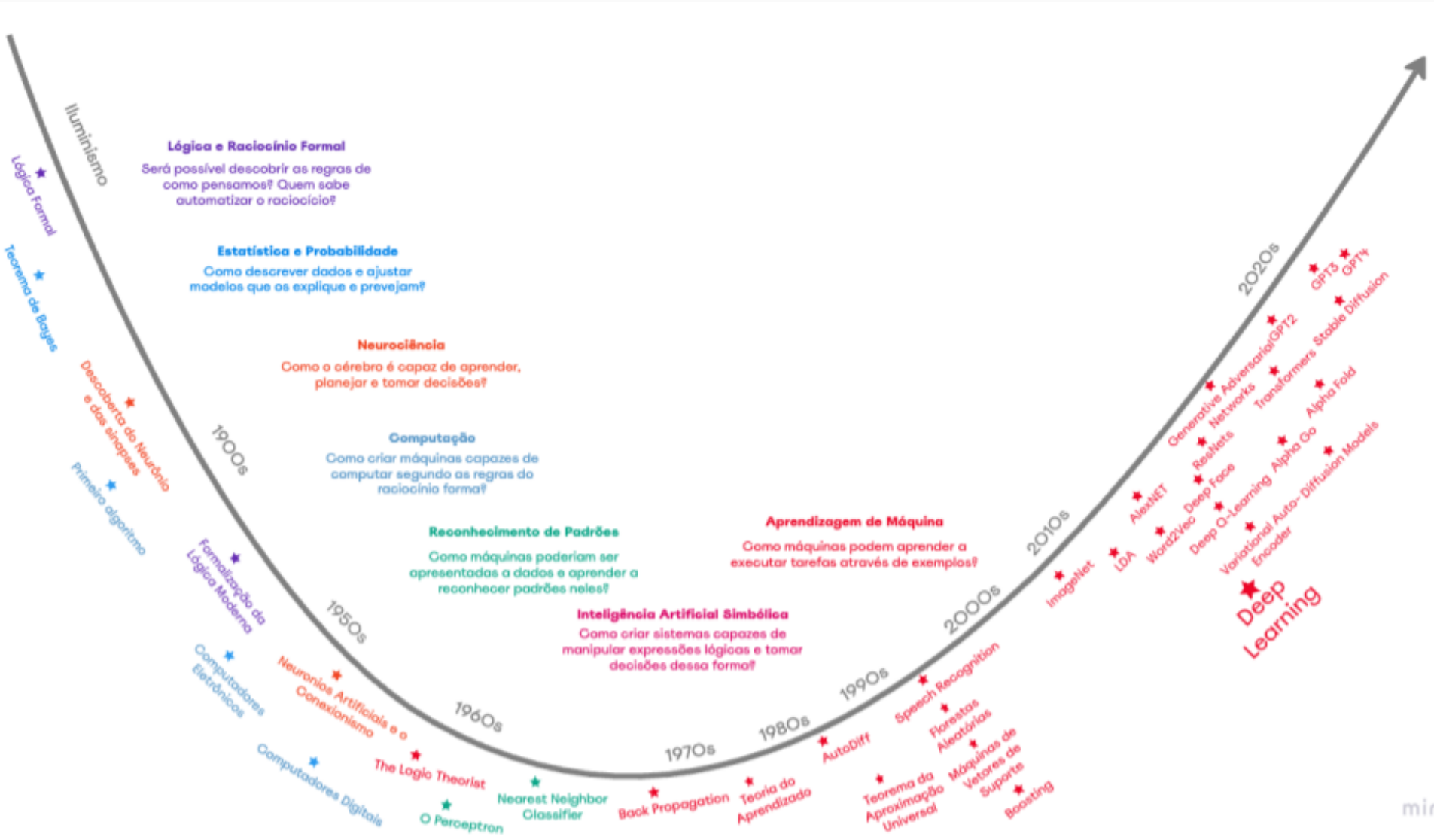
Rafael S. Calsaverini

April 22, 2024

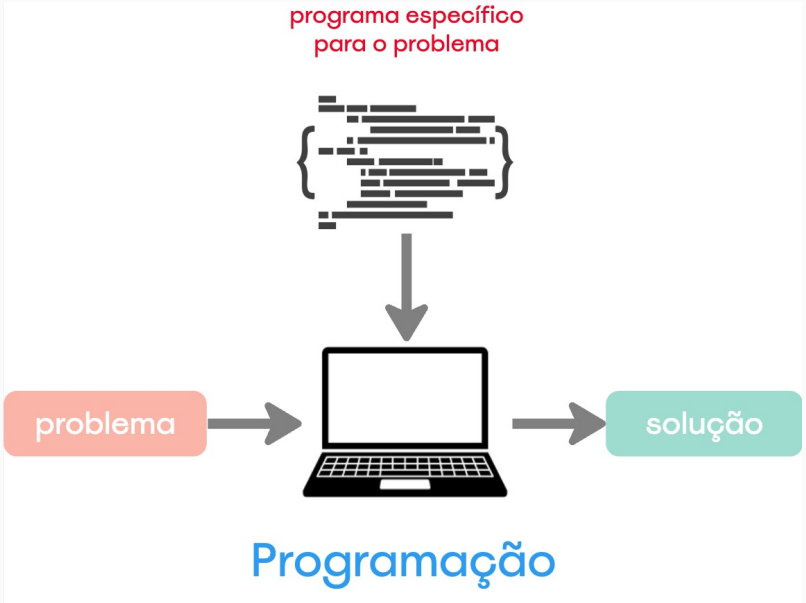
Grupo de Estudos de ML

Notas Históricas


Inteligência Computacional
Como computadores podem resolver problemas de forma autônoma?
Ciências Cognitivas
Como funciona o cérebro? Como ele é capaz de aprender, planejar e tomar decisões?
Reconhecimento de Padrões
Mineração de Dados
Aprendizado Computacional
Aprendizado por Reforço
Conexionismo
Aprendizagem de Máquina
Como computadores podem "aprender" a executar uma tarefa através de exemplos?
Ciência da Computação
Compressão e Complexidade
Teoria de Informação
Teoria de Decisão
Inferência Bayesiana
Aprendizado Estatístico
Modelos Paramétricos
Máxima Verossimilhança
Sistemas Complexos
Mecânica Estatística
Descida de Gradiente Estocástico
Processos de Decisão de Markov
Estatística
Como descrever dados e ajustar sobre eles modelos descritivos, gerativos ou causais?
Telecomunicações
Teoria de Probabilidades
Processos Estocásticos
Teoria de Otimização
miro

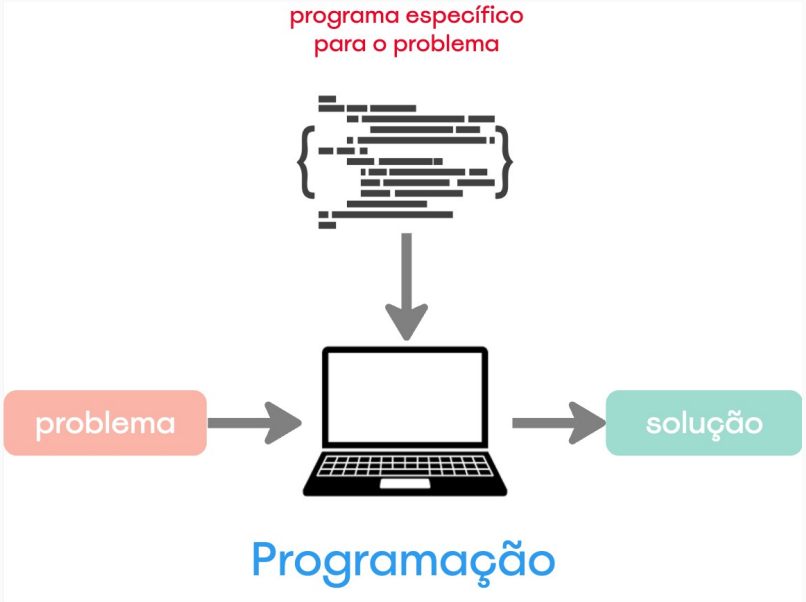

Iluminismo
1900s
1950s
1960s
1970s
1980s
1990s
2000s
2010s
2020s
Lógica e Raciocínio Formal
Será possível descobrir as regras de como pensamos? Quem sabe automatizar o raciocício?
Estatística e Probabilidade
Como descrever dados e ajustar modelos que os explique e prevejam?
Neurociência
Como o cérebro é capaz de aprender, planejar e tomar decisões?
Computação
Como criar máquinas capazes de computar segundo as regras do raciocínio forma?
Reconhecimento de Padrões
Como máquinas poderiam ser apresentadas a dados e aprender a reconhecer padrões neles?
Inteligência Artificial Simbólica
Como criar sistemas capazes de manipular expressões lógicas e tomar decisões dessa forma?
Aprendizagem de Máquina
Como máquinas podem aprender a executar tarefas através de exemplos?
Lógica Formal
Teorema de Bayes
Descoberta do Neurônio e das sinapses
Primeiro algoritmo
Formalização da Lógica Moderna
Computadores Eletrônicos
Neuronios Artificiais e o Conexionismo
Computadores Digitais
The Logic Theorist
O Perceptron
Nearest Neighbor Classifier
Back Propagation
Teoria do Aprendizado
AutoDiff
Teorema da Aproximação Universal
Speech Recognition
Florestas Aleatórias
Máquinas de Vetores de Suporte
Boosting
ImageNet
LDA
AlexNET
Word2Vec
ResNets
Deep Face
Deep Q-Learning
Generative Adversarial Networks
Alpha Go
Variational Auto-Encoder
Transformers
Diffusion Models
Alpha Fold
GPT3
GPT4
Stable Diffusion
Deep Learning

O que é aprendizado de máquina?



Programação
Cria um programa com uma série de instruções especificamente criadas para para resolver um problema.

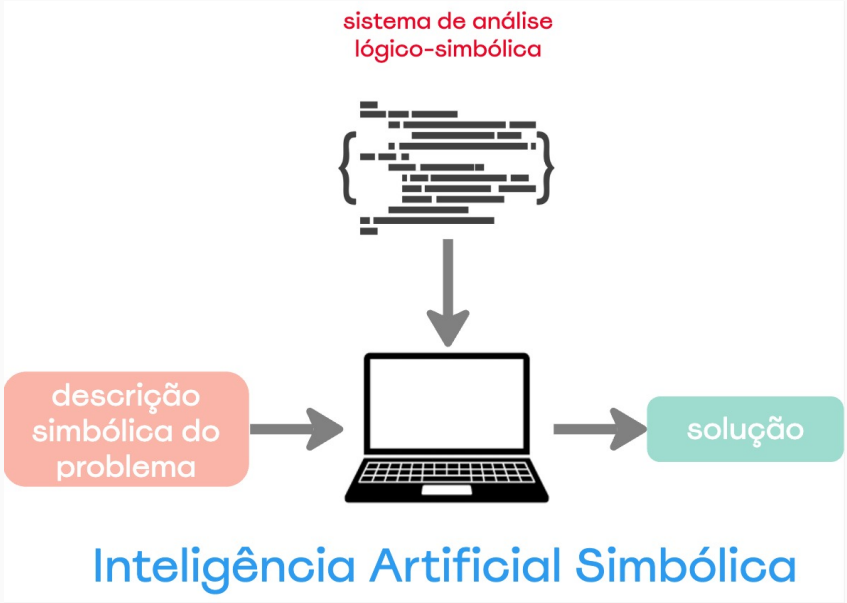
O que é aprendizado de máquina?



Programação

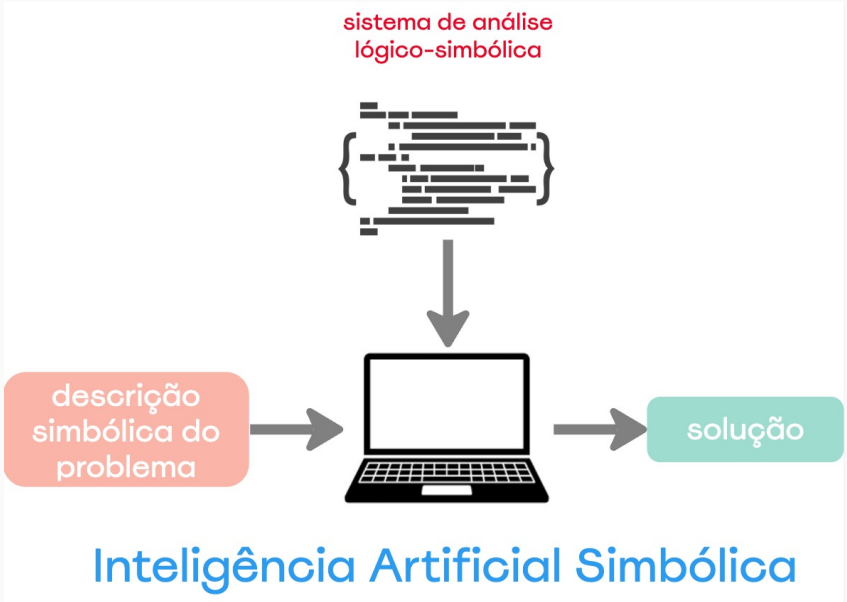
- Problema: prever a posição de um planeta daqui a 150 anos.
- Solução: implementa um programa para executar o método numérico para resolver a segunda lei de Newton com as forças envolvidas e as condições iniciais adequadas.

O que é aprendizado de máquina?



IA Simbólica / GOFAI
Usa algoritmos de manipulação de expressões lógico-simbólicas para encontrar uma solução dada uma descrição formal do problema.

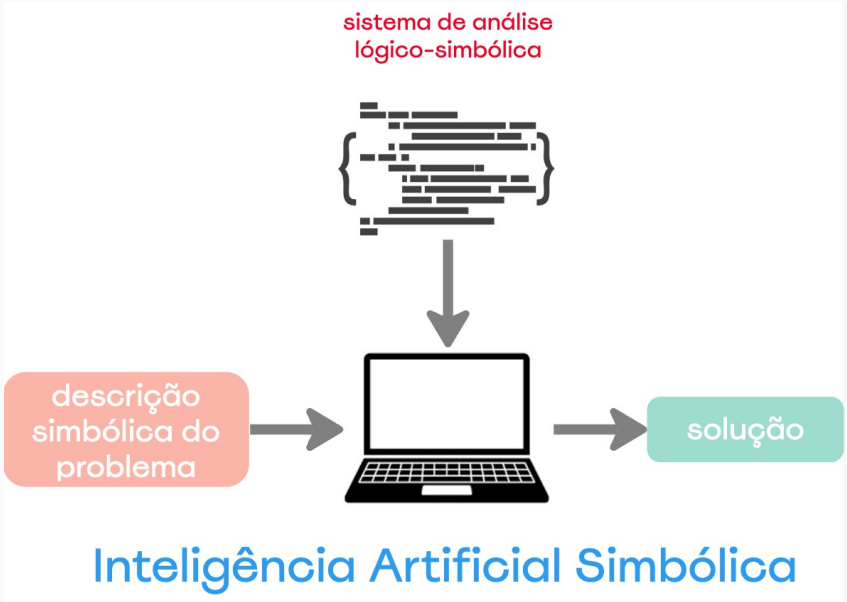
O que é aprendizado de máquina?



IA Simbólica / GOFAI

- Problema: provar um teorema.
- Solução: um sistema capaz de manipular expressões simbólicas correspondendo a uma série de axiomas e lemas tenta encontrar uma demonstração adequada.

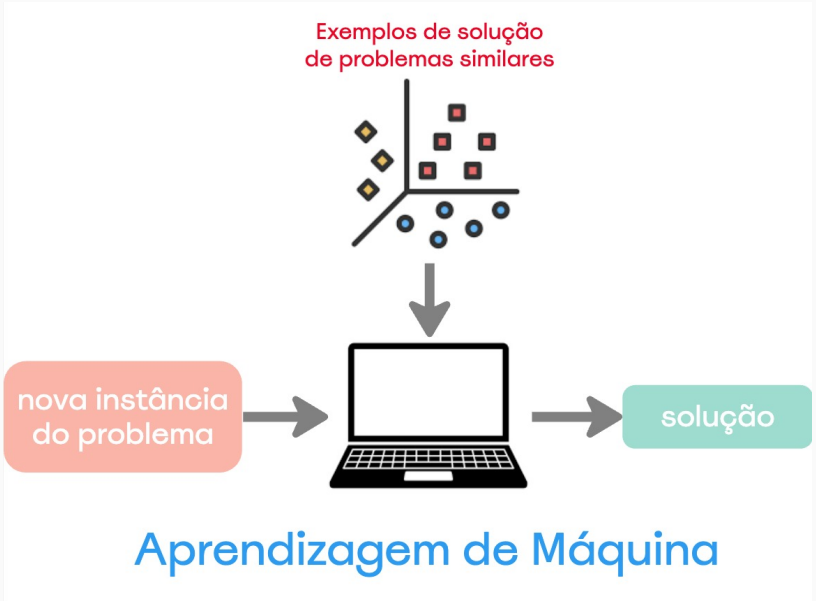
O que é aprendizado de máquina?



IA Simbólica / GOFAI

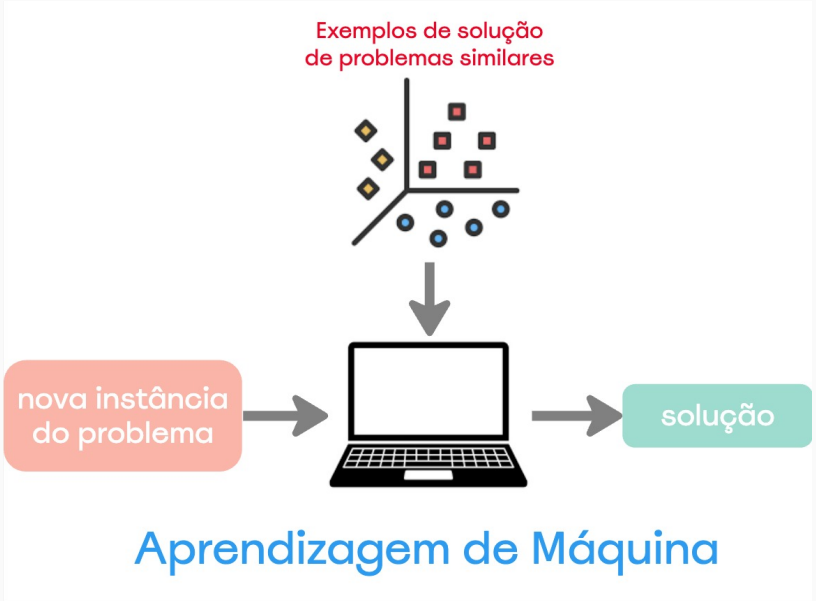
- Problema: provar um teorema.
- Solução: um sistema capaz de manipular expressões simbólicas correspondendo a uma série de axiomas e lemas tenta encontrar uma demonstração adequada.

O que é aprendizado de máquina?



Aprendizado de Máquina
dado um conjunto de dados, um algoritmo deriva uma solução que pode ser aplicada para novos dados não vistos. Ex.:

O que é aprendizado de máquina?



Aprendizado de Máquina

- Problema: transcrever áudios para texto.
- Solução: dado um conjunto de áudios com transcrições conhecidas treinar um algoritmo que é capaz de aprender a fazer a transcrição, e aplicá-lo em novos áudios com transcrição desconhecida.

Tipos de Aprendizado – Aprendizado Supervisionado

Dada uma série de tarefas passadas e suas soluções corretas, o algoritmo deve aprender a resolver a tarefa corretamente em uma situação futura.
Ex.: classificação, regressão, learn-to-rank, etc.

Tipos de Aprendizado – Aprendizado Supervisionado

Dada uma série de tarefas passadas e suas soluções corretas, o algoritmo deve aprender a resolver a tarefa corretamente em uma situação futura.
Ex.: classificação, regressão, learn-to-rank, etc.

Tipos de Aprendizado – outros

Aprendizado por reforço:

dado um cenário em que se podem fazer ações e receber recompensas, aprender a fazer as ações que acumulam mais recompensa ao longo das rodadas. Ex.: Alpha Go.

Aprendizado Gerativo:

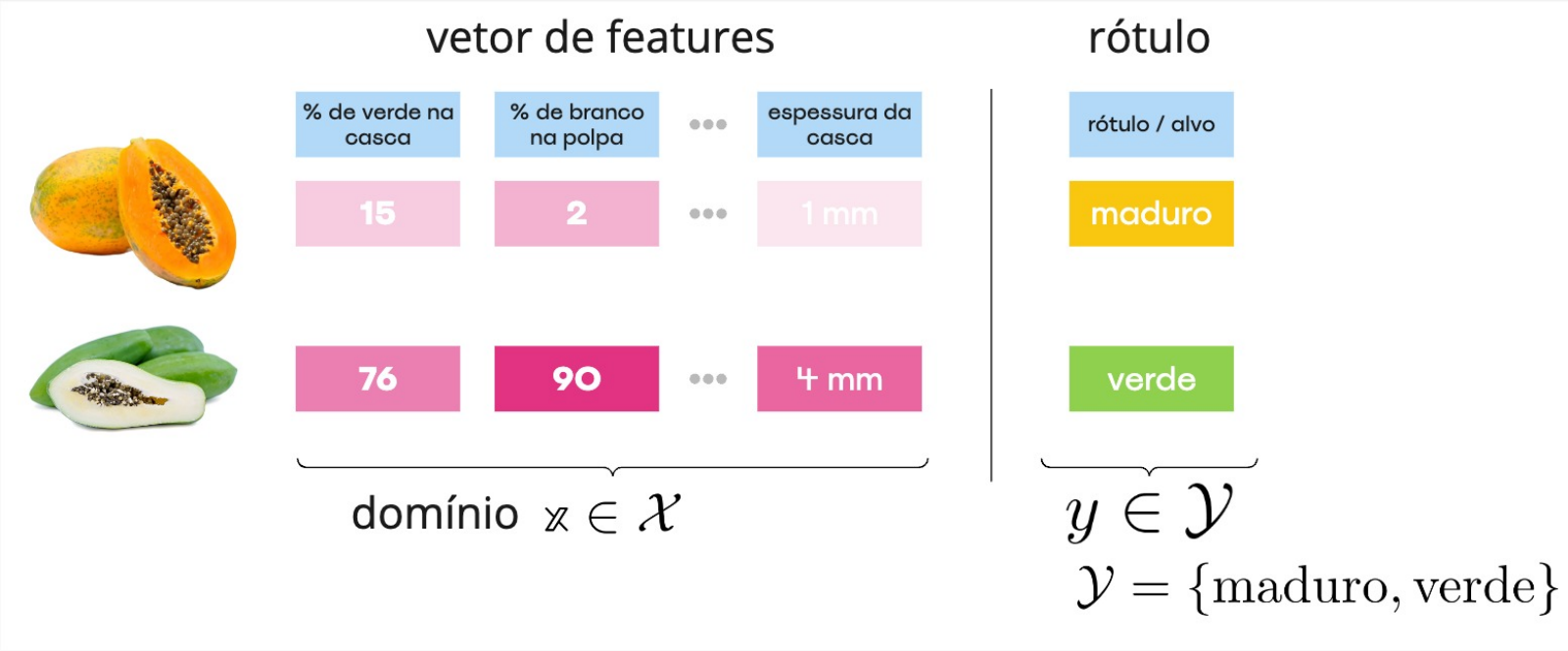
dado um conjunto de exemplos, aprender a simular a itens da mesma distribuição. Ex.: ChatGPT, Stable Diffusion

...e muitos outros:
aprendizado ativo, aprendizado semi-supervisionado, aprendizado de representações, aprendizado auto-supervisionado, aprendizado estruturado, etc...

Teoria – Aprendizado e o Dilema Viés-Variância

Modelo formal de aprendizado

Cenário
queremos um algoritmo para classificar papaias entre verdes e maduras[1]



Modelo formal de aprendizado

Dados
Coletamos dados de muitas papaias colhidas no passado. Para cada uma criamos um **vetor de características** $\mathbb{x} \in \mathcal{X}$ e um **rótulo** $y \in \{\text{maduro}, \text{verde}\}$.

Preditor
Queremos encontrar uma função que, para cada $\mathbb{x}$, retorne uma previsão para o rótulo y:

$$h : \mathcal{X} \to \{\text{maduro}, \text{verde}\}$$

Modelo formal de aprendizado

Distribuição dos dados
Suponha que os dados coletados são descritos por uma distribuição de probabilidades D:

$$(\mathbb{x}, y) \sim D$$

Probabilidade de erro
Podemos avaliar o preditor pela probabilidade de que ele cometa um erro:

$$\varepsilon[h] = \Pr_{(\mathbb{x},y)\sim D}\{h(\mathbb{x}) \neq y\}$$

Um bom preditor tem esse erro o menor possível.

Problema
Não é possível calcular $\varepsilon[h]$ de antemão. Não conhecemos D!!!

Modelo formal de aprendizado

Mas podemos coletar dados e aproximar $\varepsilon[h]$
Suponha que temos um monte de **exemplos** coletados:

$$S = \{(\not{x}_k, y_k) \sim D \mid k = 1, 2, \ldots, N\}$$

Podemos fazer uma estimativa do **erro empírico**:

$$\hat{\varepsilon}_S[h] = \frac{1}{N} \sum_{k=1}^{N} \mathbb{1}\left[h(\not{x}_k) \neq y_k\right]$$

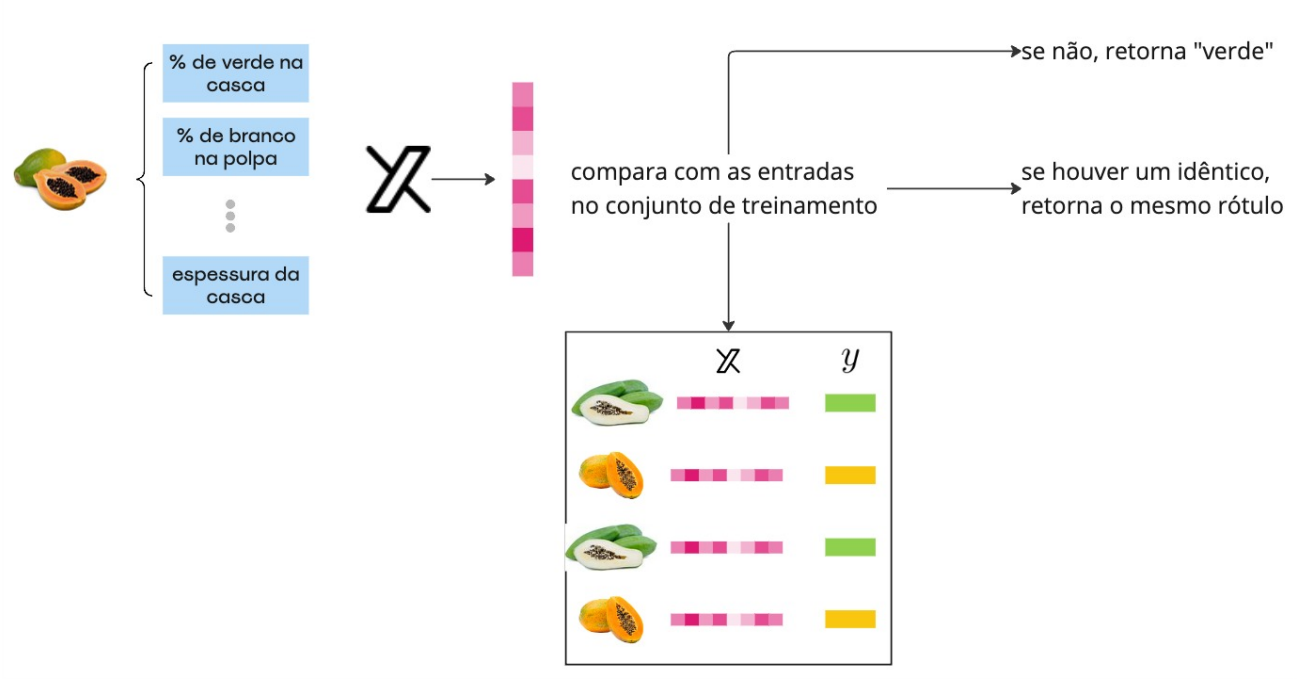
Dessa forma, um possível preditor "aprendido" através dos exemplos seria através da **minimização do erro empírico**.

$$h^\star = \arg\min_{h \in \mathcal{H}} \hat{\varepsilon}[h]$$

Problema: Generalização e Overfitting

Suponha o seguinte preditor

$$h(\mathbb{x}) = \begin{cases} y_k, & \text{if } \mathbb{x} = \mathbb{x}_k \\ \text{verde}, & \text{otherwise} \end{cases}$$



Qual é o erro empirico desse preditor?

Problema: Generalização e Overfitting

Suponha o seguinte preditor

$$h(\mathbb{x}) = \begin{cases} y_k, & \text{if } \mathbb{x} = \mathbb{x}_k \\ \text{verde}, & \text{otherwise} \end{cases}$$

O preditor tem erro empírico zero!!! Mas... é um bom preditor?

Overfitting
O preditor não comete nenhum erro no conjunto de dados usado para treiná-lo, mas não é capaz de **generalizar** para novos casos. A resposta para qualquer novo caso é verde.

Erro de treinamento e erro de generalização

Para determinar quão bem um preditor é capaz de classificar nossas papaias, é importante que **o erro empírico seja calculado em um conjunto não usado para treiná-lo**! Definimos o conjunto de treinamento T e o conjunto hold-out H fazendo uma **partição aleatória** dos dados coletados em S.[1]

Erro de treinamento:

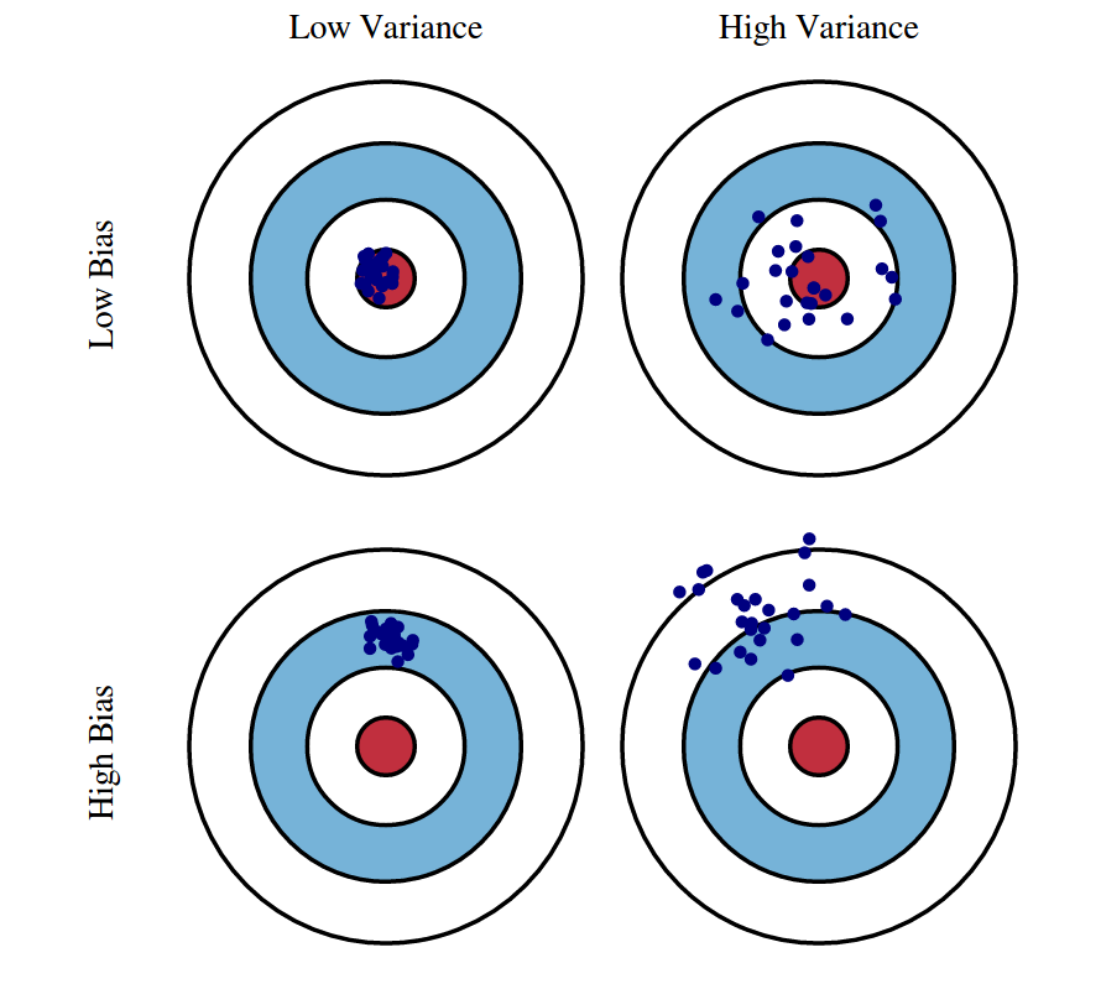
$$\varepsilon_{\text{train}}[h] = \hat{\varepsilon}_T[h]$$

Erro de generalização:

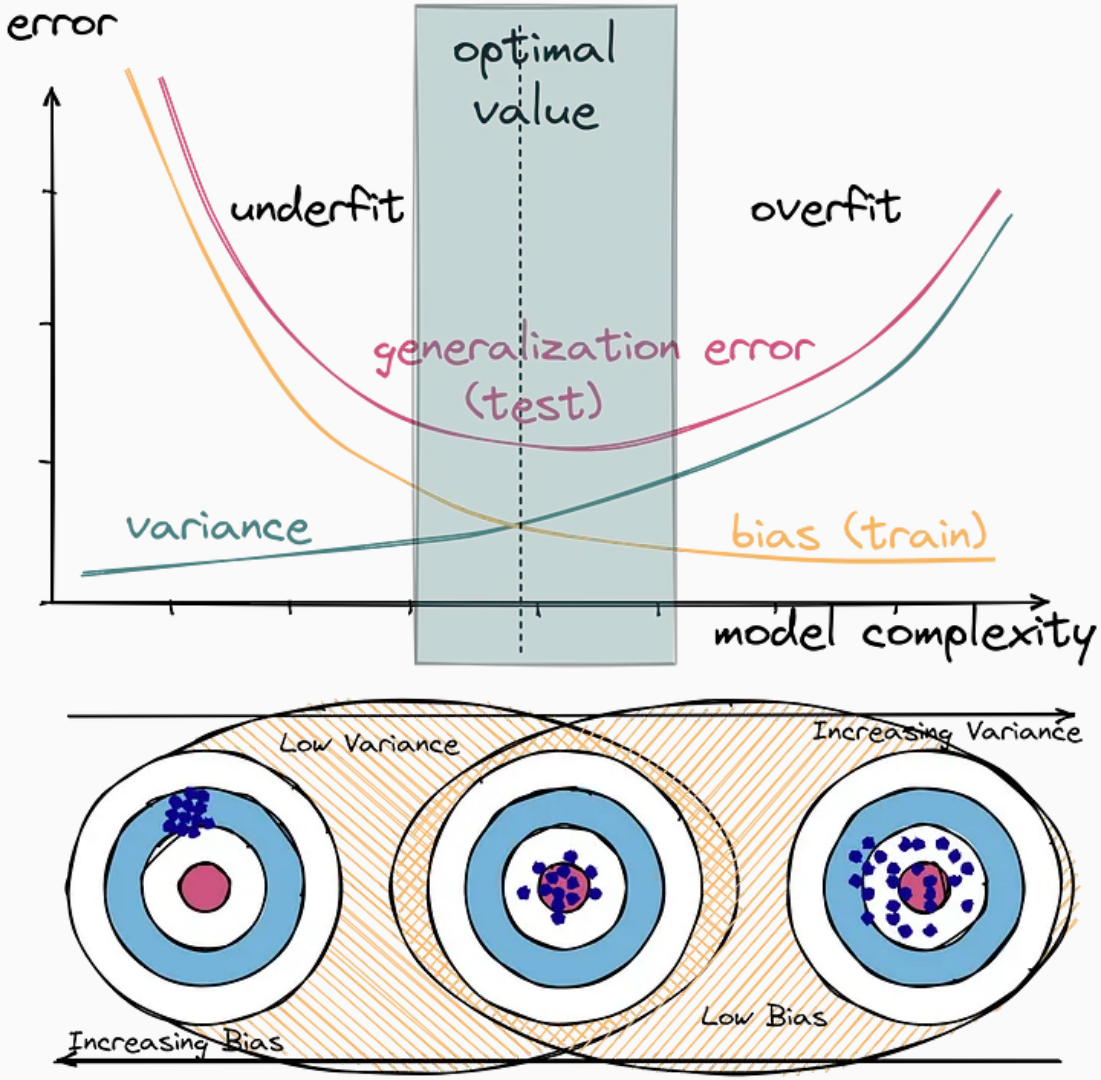
$$\varepsilon_{\text{gen.}}[h] = \hat{\varepsilon}_H[h]$$

[1] Por exemplo, colocando um exemplo em T com probabilidade 0.8 e em H com probabilidade 0.2

Dilema Viés-Variância



Dilema Viés-Variância



References

S. Shalev-Shwartz and S. Ben-David. **Understanding Machine Learning: From Theory to Algorithms.** Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press, 2014. ISBN: 9781107057135. URL: https://books.google.com.br/books?id=ttJkAwAAQBAJ.