

Física 1 – Ciências Moleculares

Caetano R. Miranda

AULA 27 – 28/11/2024

crmiranda@usp.br



sampa



Cronograma

CRONOGRAMA TENTATIVO - FISICA I PARA CIENCIAS MOLECULARES

DATA	aula nº	Segunda (16h - 18h) - 1o Andar Novo Milenio	aula nº	Quintas (09h - 12h) - Sala T34	DATA
05/Aug		Recepção		Recepção	08/Aug
12/Aug	1	Apresentação da Disciplina	2	Experimentação 1 - Escalas	15/Aug
19/Aug	3	Revisão - Bases Matemáticas	4	Revisão - Ferramentas básicas	22/Aug
26/Aug	5	Experimentação 2 - Mov. em 1 D	6	Mov. em 1D	29/Aug
02/Sep	Feriado	Semana da Pátria. Não haverá aula.	Feriado	Semana da Pátria. Não haverá aula.	05/Sep
09/Sep	7	Experimentação 3 - VR & Projéteis	8	Mov. em 2D e 3D	12/Sep
16/Sep	9	Experimentação 4a - Dinâmica & Princípios	10	Princípios da Dinâmica - Leis de Newton	19/Sep
23/Sep	11	Experimentação 4b - Dinâmica & Princípios	12	Energia e Trabalho	26/Sep
30/Sep	13	PROVA 1	14	Energia e Trabalho	03/Oct
07/Oct	15	Experimentação 5 - Energia e Trabalho	16	Projeto	10/Oct
14/Oct	17	Experimentação 6 - Física dos Desenhos Animados	18	Simetria e Conservação	17/Oct
21/Oct	19	Experimentação 7 - Colisões	20	Colisões	24/Oct
28/Oct	Feriado	Consagração ao Funcionário Público. Não haverá aula.	21	Forças de Interação	31/Oct
04/Nov	22	Experimentação 8 - Rotações e Momento Angular	23	Aula Invertida / Projeto	07/Nov
11/Nov		Semana Trabalho		Semana Trabalho	14/Nov
18/Nov	24	Revisão - Monitor	25	PROVA 2	21/Nov
25/Nov	26	Experimentação 9a - Aprendizado de Máquina	27	Experimentação 9b - Aprendizado de Máquina	28/Nov
02/Dec	28	Rotação e Momento Angular	29	Apresentação PROJETO	05/Dec
09/Dec	30	VISTA	31	NOTAS - ENCERRAMENTO	12/Dec

Introdução ao aprendizado de máquina: o caso do rolamento no plano inclinado

Phd. Wagner Wlysses R. Araujo

Ms. Pedro N. O. Junior

Department of Materials Physics and Mechanics

Institute of Physics

University of São Paulo

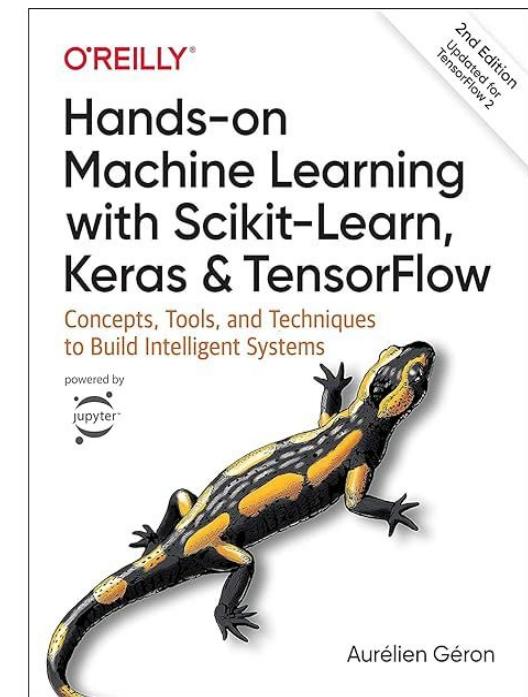
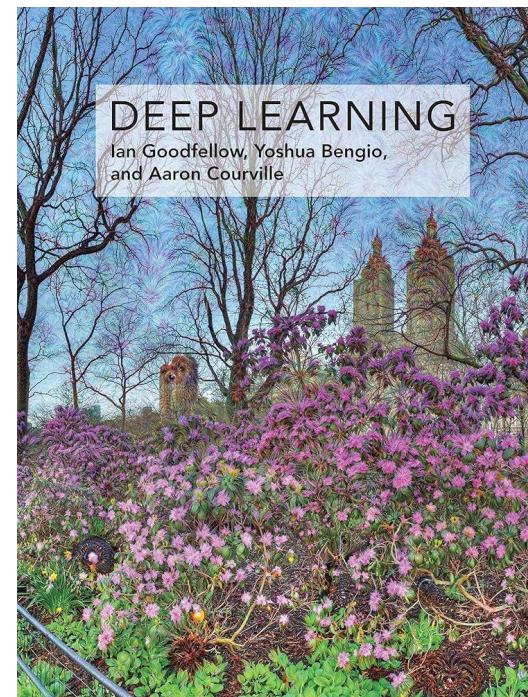
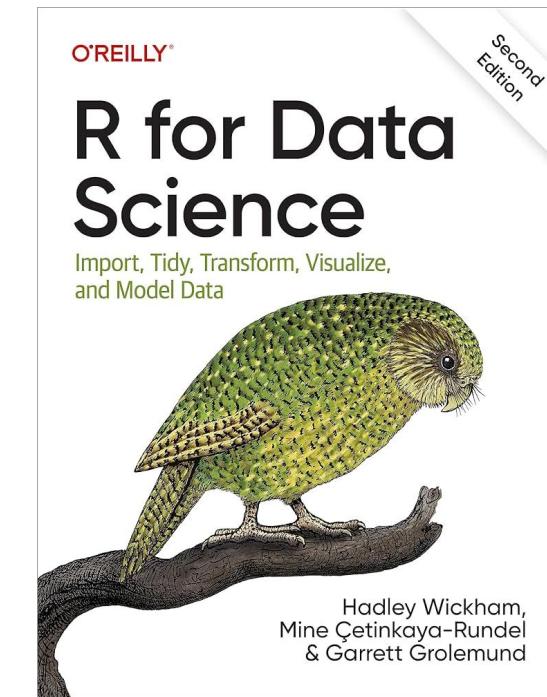
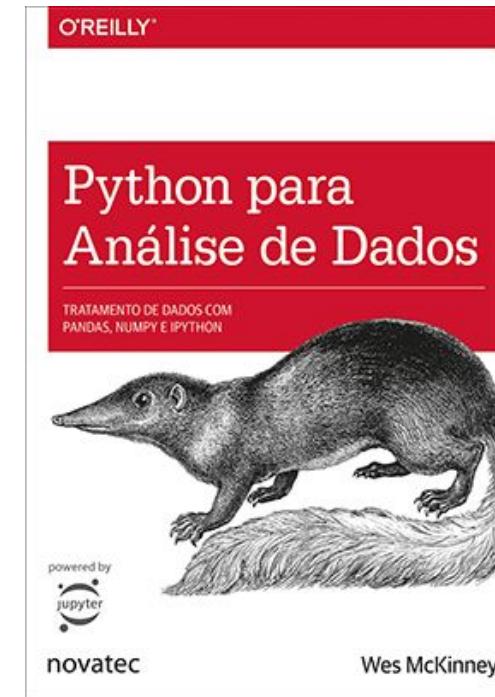
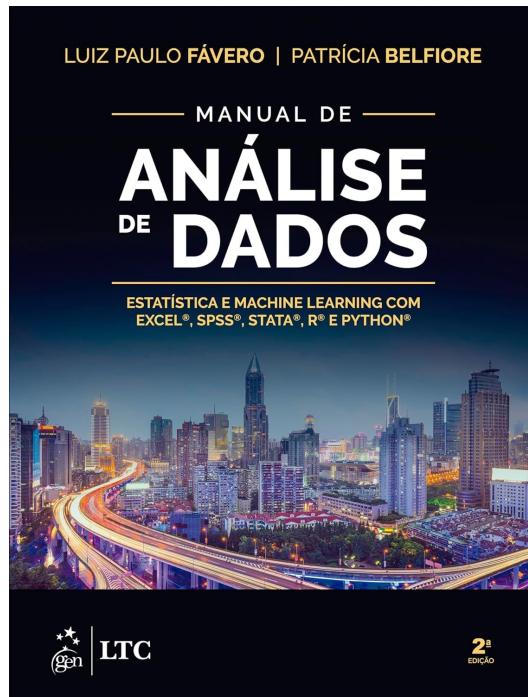
wwlysses@alumni.usp.br

pedronunes@usp.br

Sumário

- 0. Algumas referencias bibliográficas**
 - 1. Introdução**
 - 2. O que é Inteligência artificial?**
 - 3. Tipos de Aprendizado de Máquina**
 - 4. Algumas aplicações em Física**
-

0. Algumas referencias bibliográficas



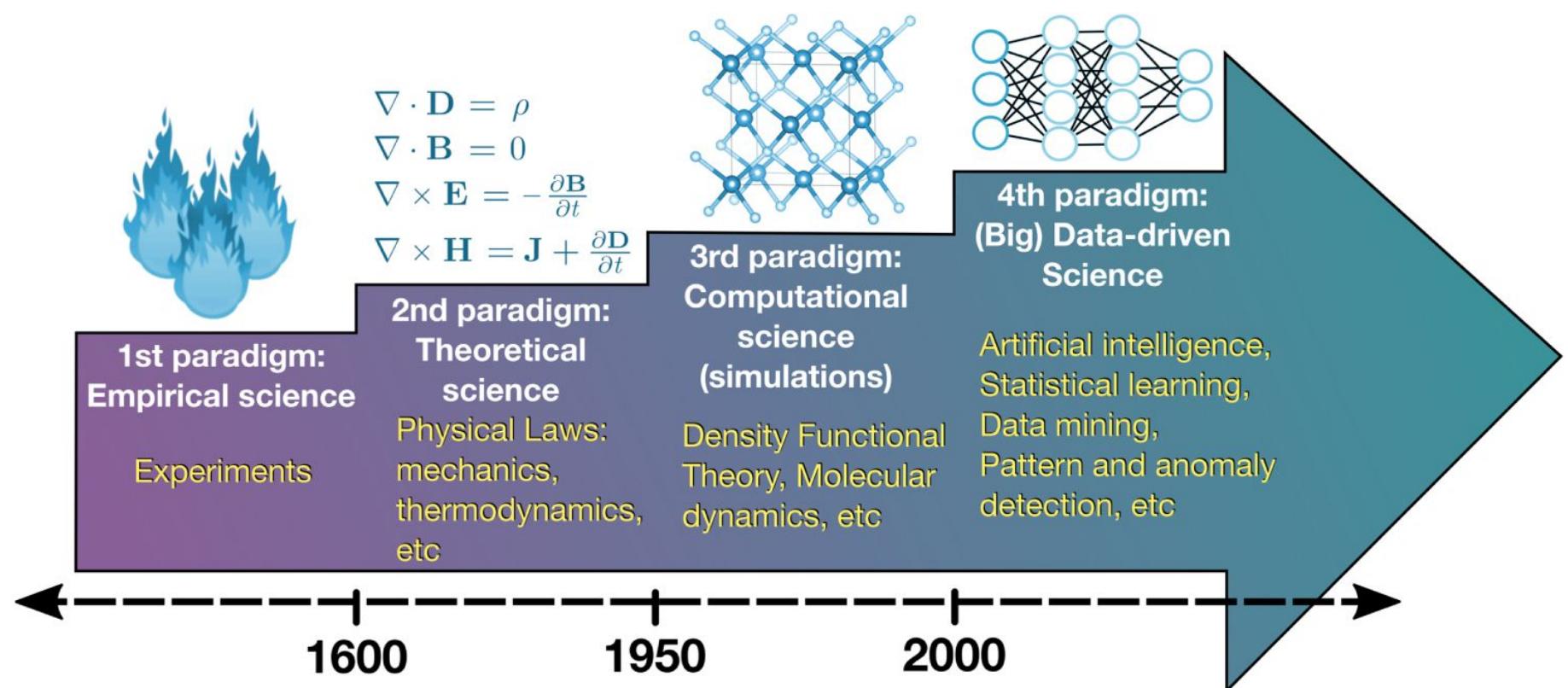
1. Introdução

Os primeiros modelos datam do final da segunda guerra mundial (1944-1946).

Após a segunda guerra, Alan Turing cunha o termo “*Computing Machinery and Intelligence*” (1956).

Em 2012 um modelo de aprendizado de máquina profundo (deep learning) superou todos seus concorrentes.

Em 2017 veio a arquitetura de rede profunda “*transformer*” (GPT, BERT)



Fonte: Schleder et al., Journal of Physics: Materials 2 (3) 2019

2. O que é Inteligência artificial?

Definição mais coloquial:

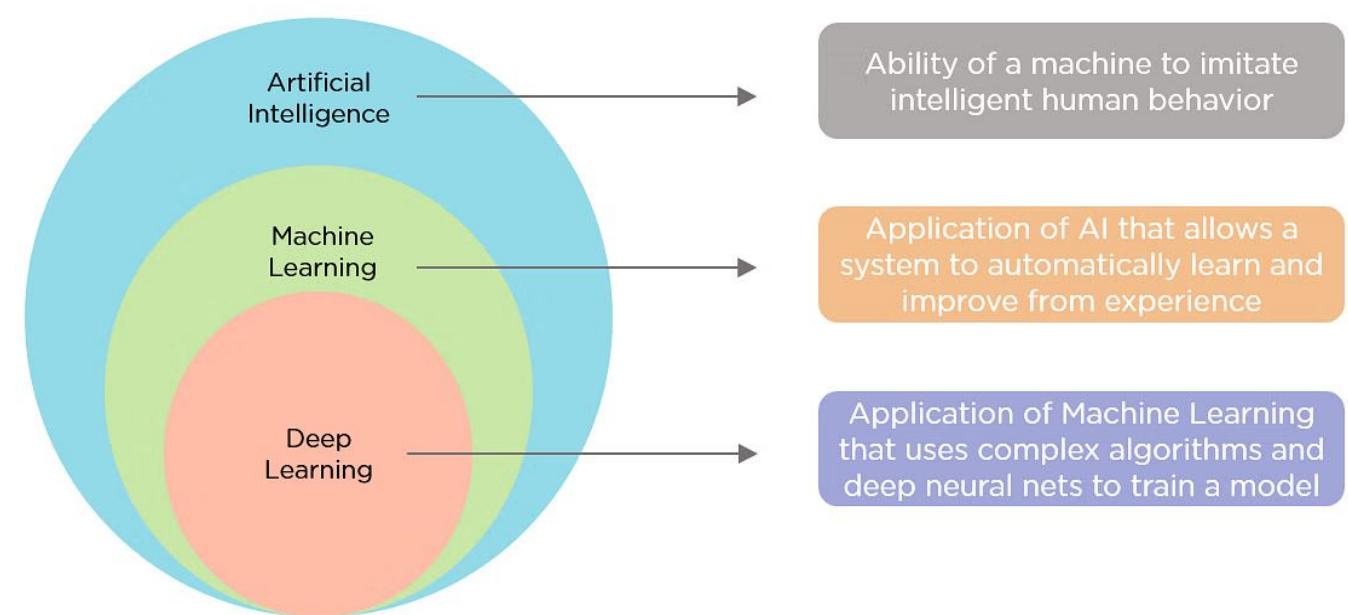
Inteligência artificial é a área que estuda a implementação de atividades realizadas por máquinas, que anteriormente eram realizadas por humanos.

Dicionário Oxford:

“a teoria e o desenvolvimento de sistemas computacionais capazes de executar tarefas que normalmente requerem inteligência humana, como percepção visual, reconhecimento de fala, tomada de decisão e tradução entre idiomas.”

Definição MIT:

“A inteligência artificial é a capacidade dos computadores de imitar funções cognitivas humanas, como aprendizado e solução de problemas. Por meio da IA, um sistema de computador usa matemática e lógica para simular o raciocínio que as pessoas usam para aprender com novas informações e tomar decisões.”



Fonte:

<https://towardsdatascience.com/redefining-data-science-d7f2026a021a>

3. Abordagem programática

Abordagem Programática

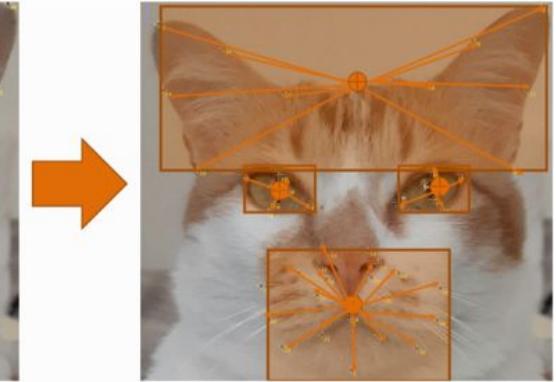
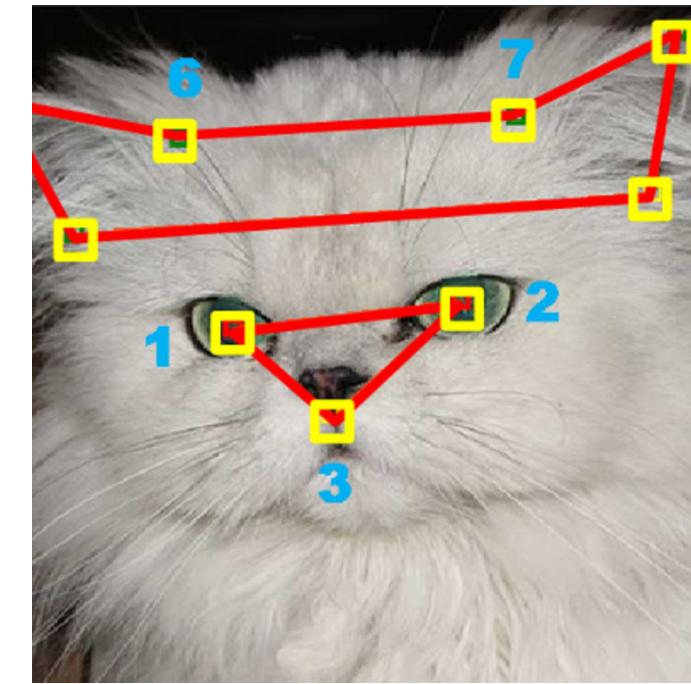
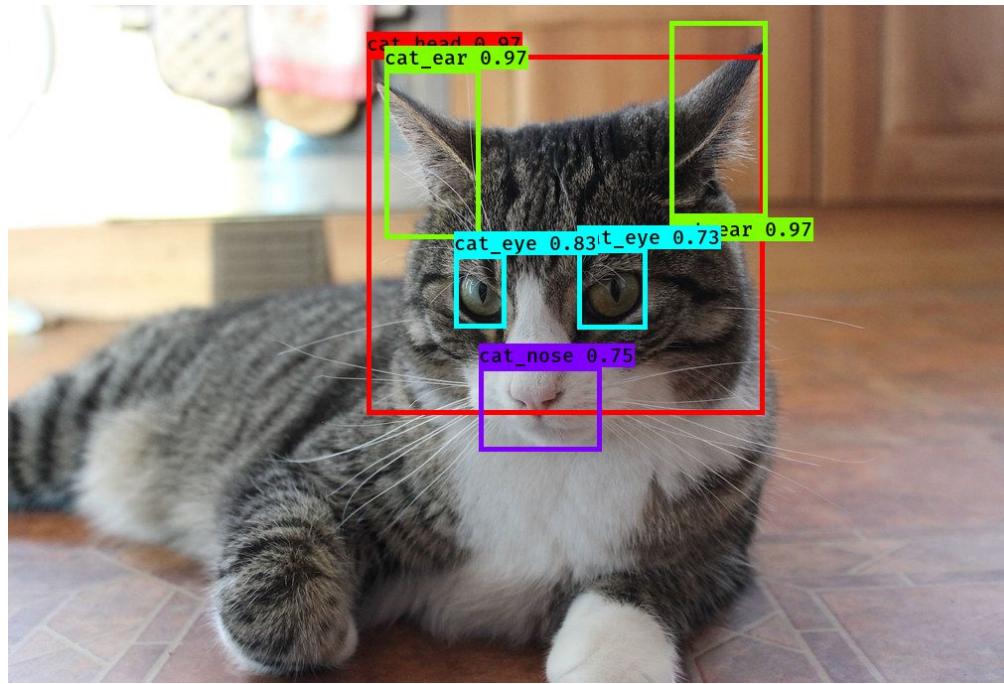
- Envolve criar soluções baseadas em **regras explícitas e pré-definidas**, onde o programador especifica exatamente como o sistema deve processar os dados.
- O comportamento é definido por **lógica condicional, algoritmos específicos ou processos sequenciais** escritos manualmente.

Exemplo:

Para detectar spam em e-mails, você pode definir regras como:

- "Se o e-mail contém as palavras 'dinheiro fácil' ou 'grátis', marque como spam."

3.1 Abordagem programática



3.2 Abordagem programática



Fonte:<https://www.freecodecamp.org/news/chihuahua-or-muffin-my-search-for-the-best-computer-vision-api-cbda4d6b425d/>

4. Abordagem via inteligência artificial

Aprendizado de Máquina

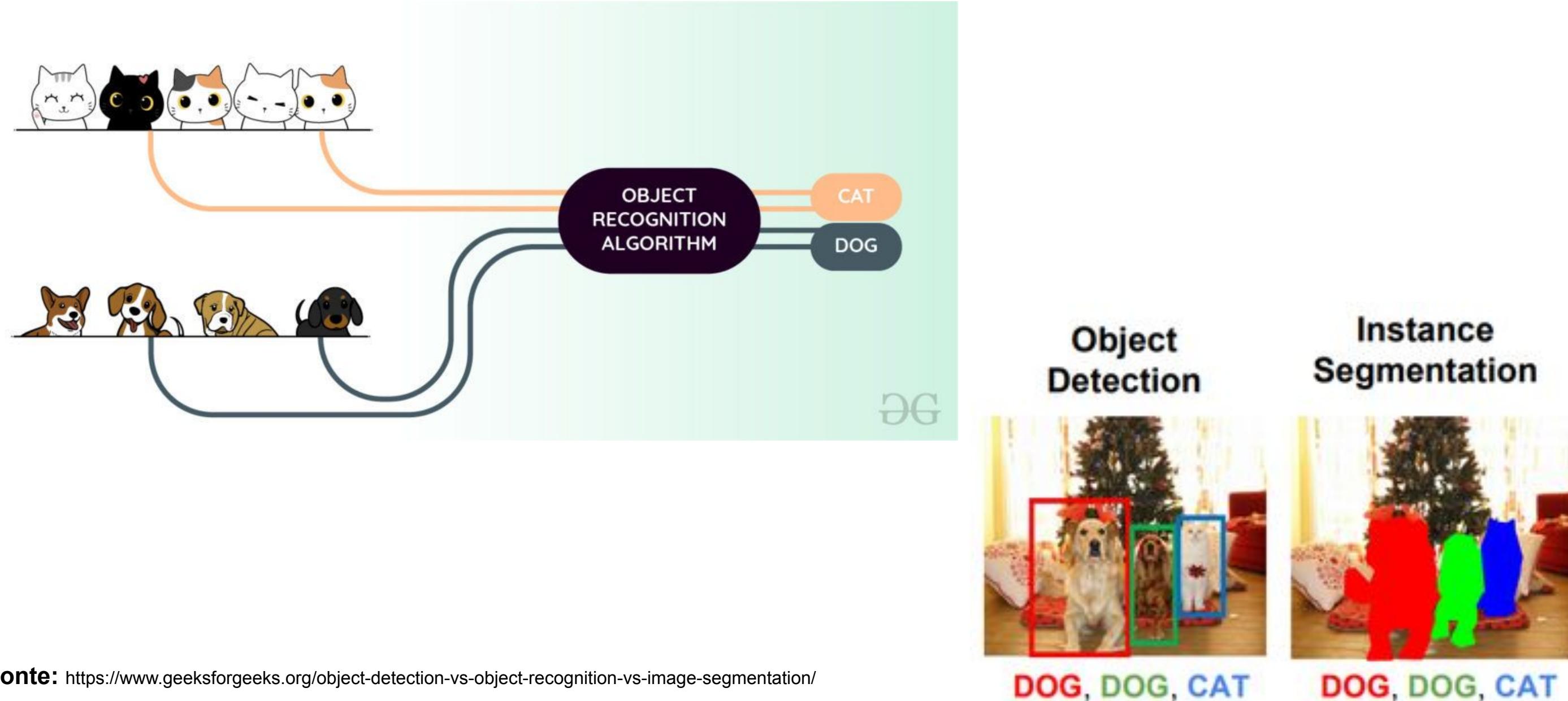
- Em vez de especificar regras explícitas, o modelo aprende padrões a partir de **dados históricos**. O objetivo é criar um sistema que generalize bem para novos dados.
- Os programadores fornecem dados de entrada e o modelo **descobre as regras automaticamente** por meio de otimização matemática.

Exemplo:

Para detectar spam, você fornece ao modelo um conjunto de e-mails rotulados como "spam" ou "não spam". O modelo aprende a identificar características importantes (como palavras, remetentes ou formatos) que ajudam a classificar novos e-mails.

Fonte: <https://www.geeksforgeeks.org/object-detection-vs-object-recognition-vs-image-segmentation/>

4. Abordagem via inteligência artificial



5. Etapas de um fluxo de trabalho em ciência de dados



1. Coleta de Dados:

- Esta etapa envolve a aquisição de dados relevantes para o problema em questão. Isso pode incluir dados estruturados (como tabelas em bancos de dados) ou dados não estruturados (como texto, imagens ou áudio).
- Métodos comuns incluem a extração de dados de fontes online, bancos de dados internos, APIs, entre outros.
- É importante garantir a qualidade e a integridade dos dados coletados, identificando e lidando com problemas como valores ausentes, outliers e inconsistências.



2. Pré-processamento e Limpeza de Dados:

- Nesta etapa, os dados coletados são preparados para análise. Isso pode incluir a limpeza de dados para remover erros, preenchimento de valores ausentes, normalização ou padronização de dados e a seleção de características relevantes.
- Técnicas comuns incluem o uso de bibliotecas como Pandas em Python para manipulação de dados e visualização para entender a distribuição e a relação entre as variáveis.



3. Modelagem:

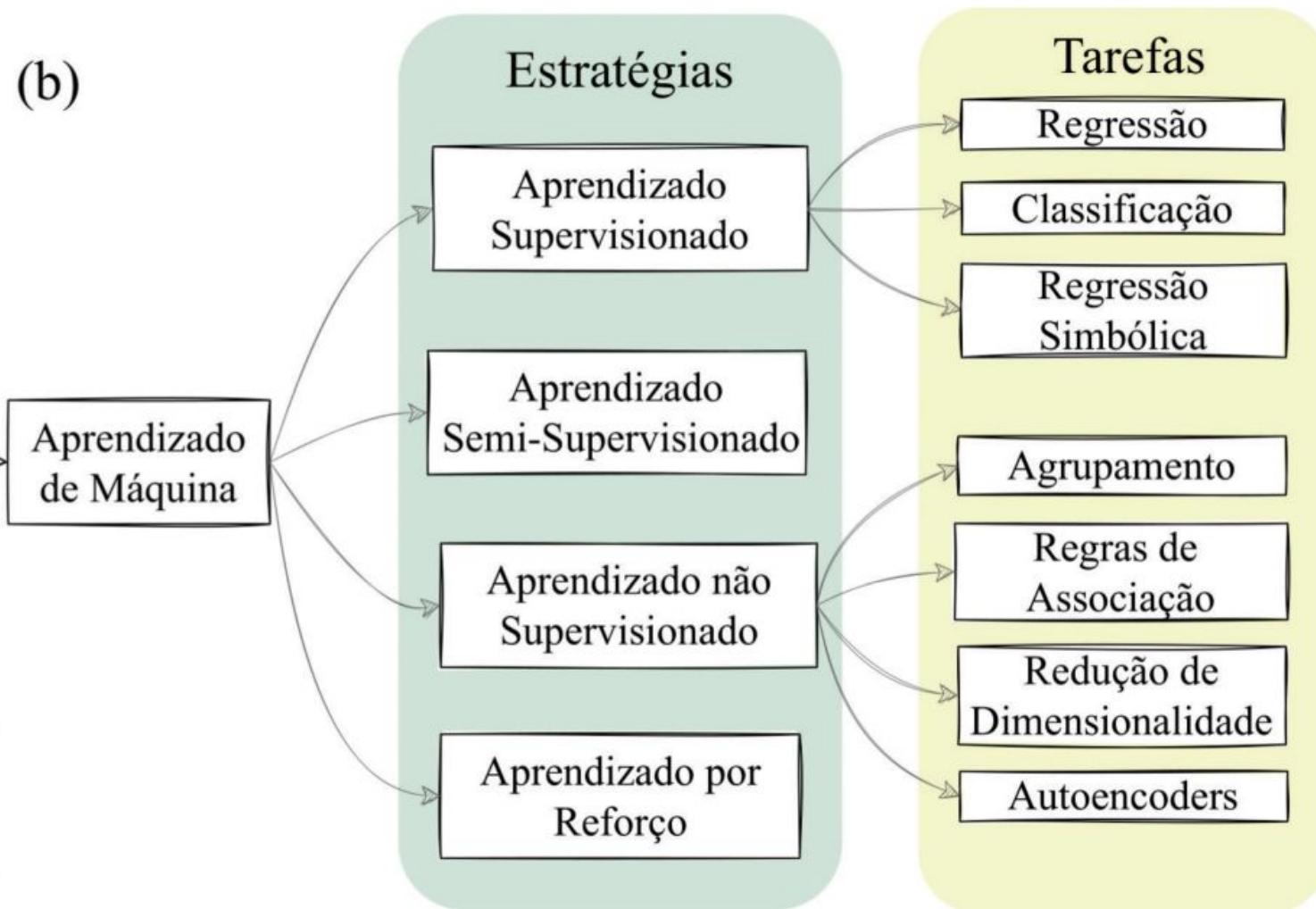
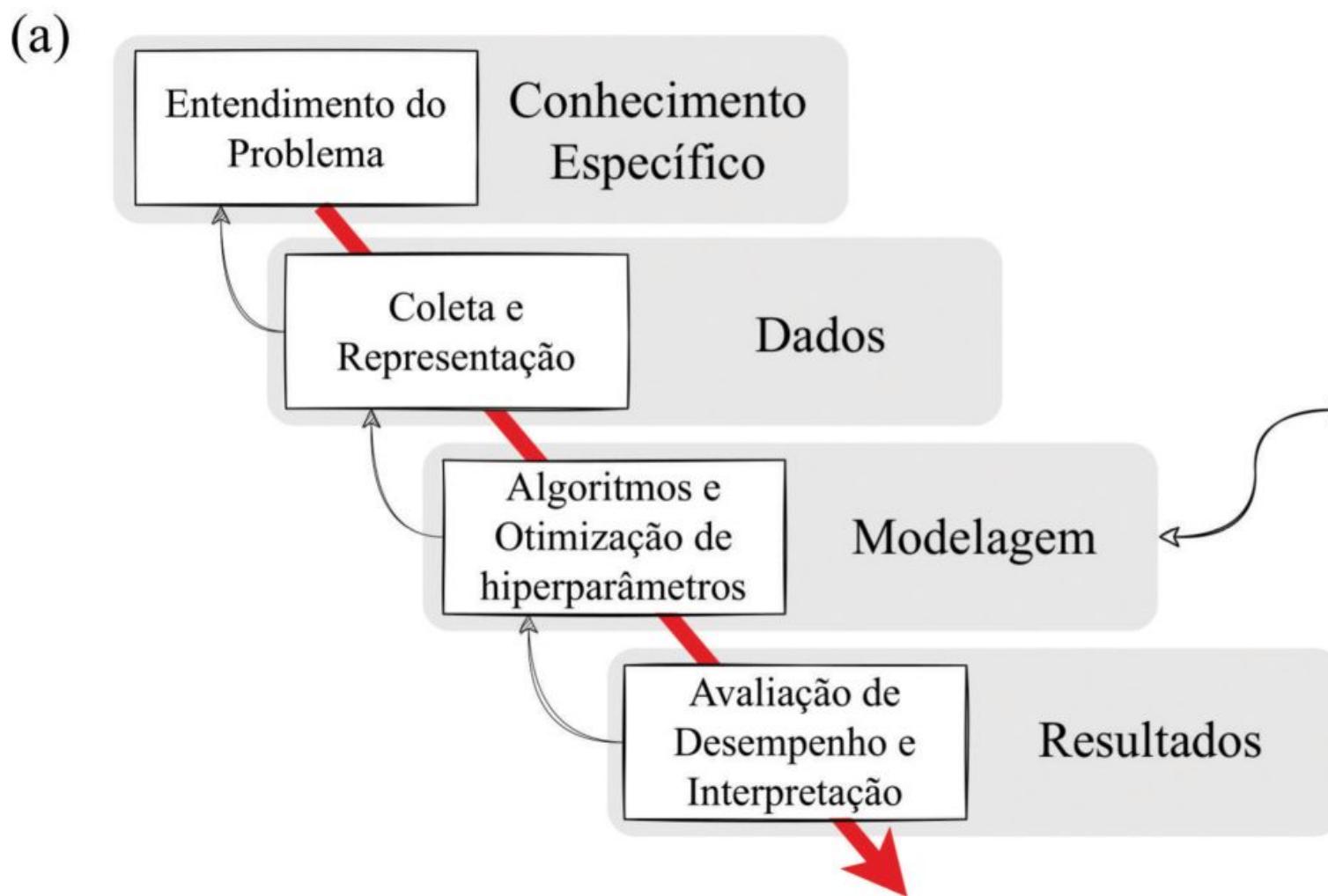
- Aqui, os dados pré-processados são utilizados para construir modelos que possam fornecer insights ou previsões. Dependendo do problema, diferentes técnicas de modelagem podem ser aplicadas, como regressão, classificação, clustering ou aprendizado profundo.
- O ajuste de hiperparâmetros e a validação cruzada são técnicas comuns para otimizar o desempenho do modelo e avaliar sua capacidade de generalização.



4. Avaliação e Implantação:

- Nesta etapa final, os modelos desenvolvidos são avaliados para determinar sua eficácia e utilidade. Isso pode envolver a avaliação de métricas de desempenho específicas para o problema em questão e a comparação com modelos alternativos.
- Uma vez que um modelo satisfatório tenha sido identificado, ele pode ser implantado em produção para uso prático.

6. Modelagem



7. Tipos de Dados

Tipos de dados são utilizados em algoritmos para representar valores e informações de diferentes tipos.

Cada linguagem de programação possui seus próprios tipos de dados, mas alguns dos tipos mais comuns são:

- **Inteiro:** representa números inteiros, positivos ou negativos, sem casas decimais.
- **Real:** representa números reais, com casas decimais.
- **Caractere:** representa um único caractere, como uma letra ou um símbolo.
- **String:** representa uma sequência de caracteres.
- **Booleano:** representa um valor lógico verdadeiro ou falso.

8. Tipos de Dados

1. Dados Estruturados

- **Definição:** São organizados em um formato pré-definido e estruturado, como tabelas, onde os dados são armazenados em linhas e colunas. São fáceis de armazenar, processar e analisar usando sistemas tradicionais, como bancos de dados relacionais (SQL).
- **Exemplos:**
 - Uma tabela com informações de clientes: Nome, Idade, Telefone, Endereço.
 - Planilhas do Excel.
 - Dados em bancos de dados relacionais (MySQL, PostgreSQL, etc.).
- **Características:**
 - Organizados em esquema fixo (como tabelas).
 - Fáceis de consultar com linguagens como SQL.
 - Estrutura rígida, o que facilita análises.

9. Tipos de Dados

2. Dados Não Estruturados

- **Definição:** Não possuem um formato ou estrutura predefinida. Esses dados podem estar em qualquer formato, como texto, imagens, vídeos ou áudios. Eles são difíceis de armazenar e analisar diretamente com ferramentas convencionais.
- **Exemplos:**
 - E-mails com texto e anexos.
 - Arquivos de texto simples (como um livro digital em .txt).
 - Imagens (JPEG, PNG).
 - Vídeos (MP4, AVI).
 - Arquivos de áudio (MP3, WAV).
 - Publicações em redes sociais.
- **Características:**
 - Não seguem uma estrutura rígida.
 - Necessitam de técnicas avançadas (como machine learning ou NLP) para análise.
 - Representam cerca de 80-90% dos dados gerados atualmente.

10. Tipos de variáveis

1. Variáveis Categóricas

- **Definição:** Representam categorias ou grupos qualitativos, ou seja, os dados são não numéricos (embora possam ser representados por números) e dividem os elementos em classes ou categorias.
- **Exemplos:**
 - Gênero: Masculino, Feminino, Não binário.
 - Cores: Azul, Verde, Vermelho.
 - Tipo de veículo: Carro, Moto, Caminhão.
- **Tipos:**
 - **Nominais:** Não possuem ordem ou hierarquia. Ex.: Cores (Azul, Vermelho, Verde).
 - **Ordinais:** Possuem uma ordem ou hierarquia. Ex.: Nível de satisfação (Baixo, Médio, Alto).

11. Tipos de variáveis

2. Variáveis Discretas

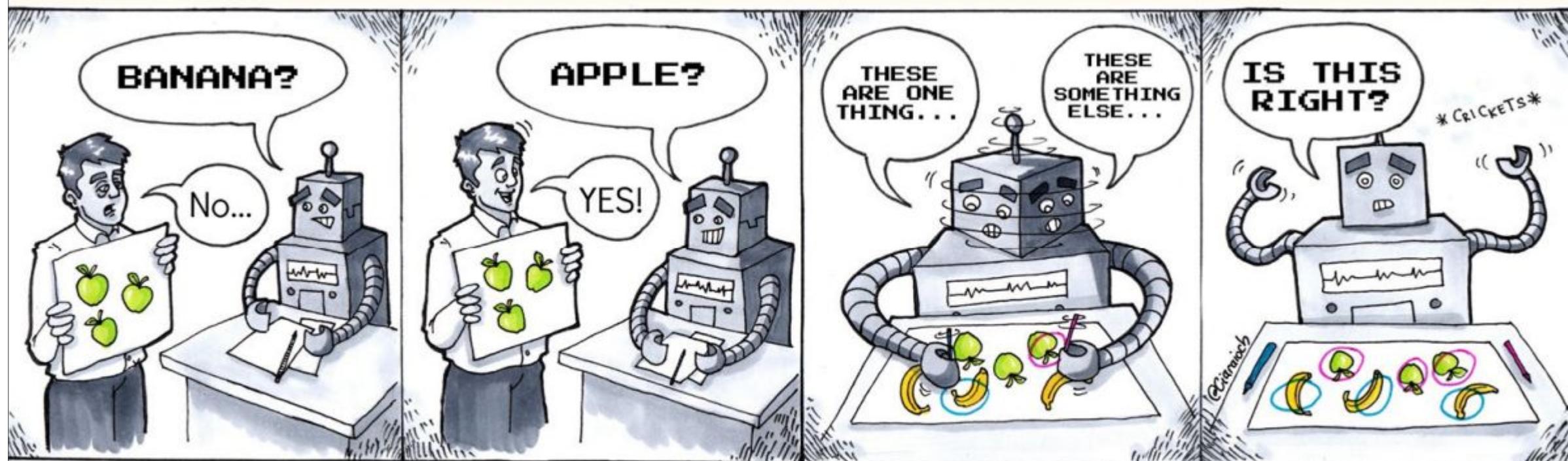
- **Definição:** Representam quantidades contáveis, ou seja, valores inteiros, finitos ou com saltos bem definidos entre eles. Elas não podem assumir valores entre os números inteiros.
- **Exemplos:**
 - Número de filhos em uma família: 0, 1, 2, 3...
 - Número de carros em um estacionamento: 5, 10, 15...
 - Lançamentos de um dado: 1, 2, 3, 4, 5, 6.
- **Características:**
 - Sempre valores inteiros.
 - Não existem frações ou números decimais.

12. Tipos de variáveis

3. Variáveis Contínuas

- **Definição:** Podem assumir qualquer valor em um intervalo contínuo de números reais, incluindo frações e decimais. São geralmente obtidas por medição.
- **Exemplos:**
 - Altura de uma pessoa: 1,75 m, 1,80 m.
 - Peso de um objeto: 3,2 kg, 4,7 kg.
 - Temperatura: 36,5 °C, 100,2 °C.
- **Características:**
 - Valores são infinitamente divisíveis dentro de um intervalo.
 - Podem incluir casas decimais.

13. Tipos de Aprendizado de Máquina



Supervised Learning

Unsupervised Learning

14. Aprendizado de máquina supervisionado

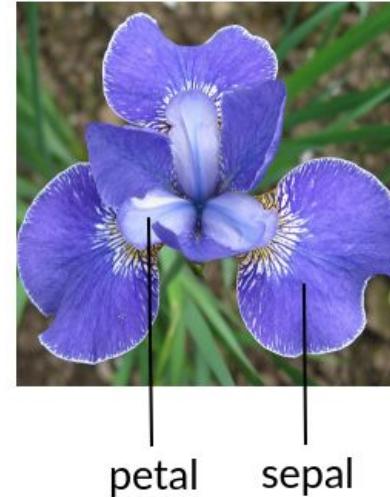
Edgar Anderson coletou os dados para quantificar a variação morfológica das flores de íris de três espécies relacionadas.

O conjunto de dados consiste em 50 amostras de cada uma das três espécies de Íris (Setosa, Virginica e Versicolor).

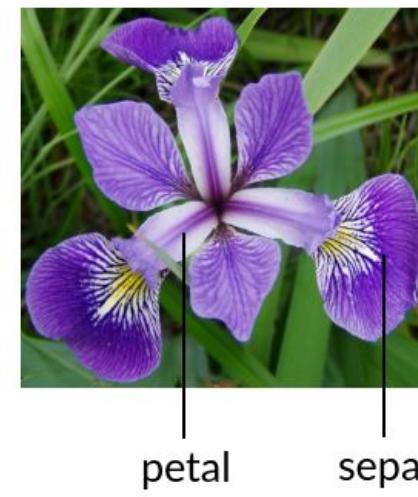
Quatro características foram medidas em cada amostra (cm):

- N=150
- Atributos: 4
 - sepal length (cm) - Cumprimento da Sépala
 - sepal width (cm) - Largura da Sépala
 - petal length (cm) - Cumprimento da Pétala
 - petal width (cm) - Largura da Sépala
- Variável dependente: Tipo de espécie de Iris
 - 0 - Setosa
 - 1 - Virginica
 - 2 - Versicolor

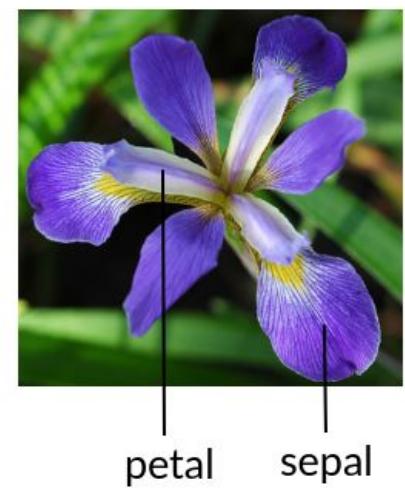
iris setosa



iris versicolor



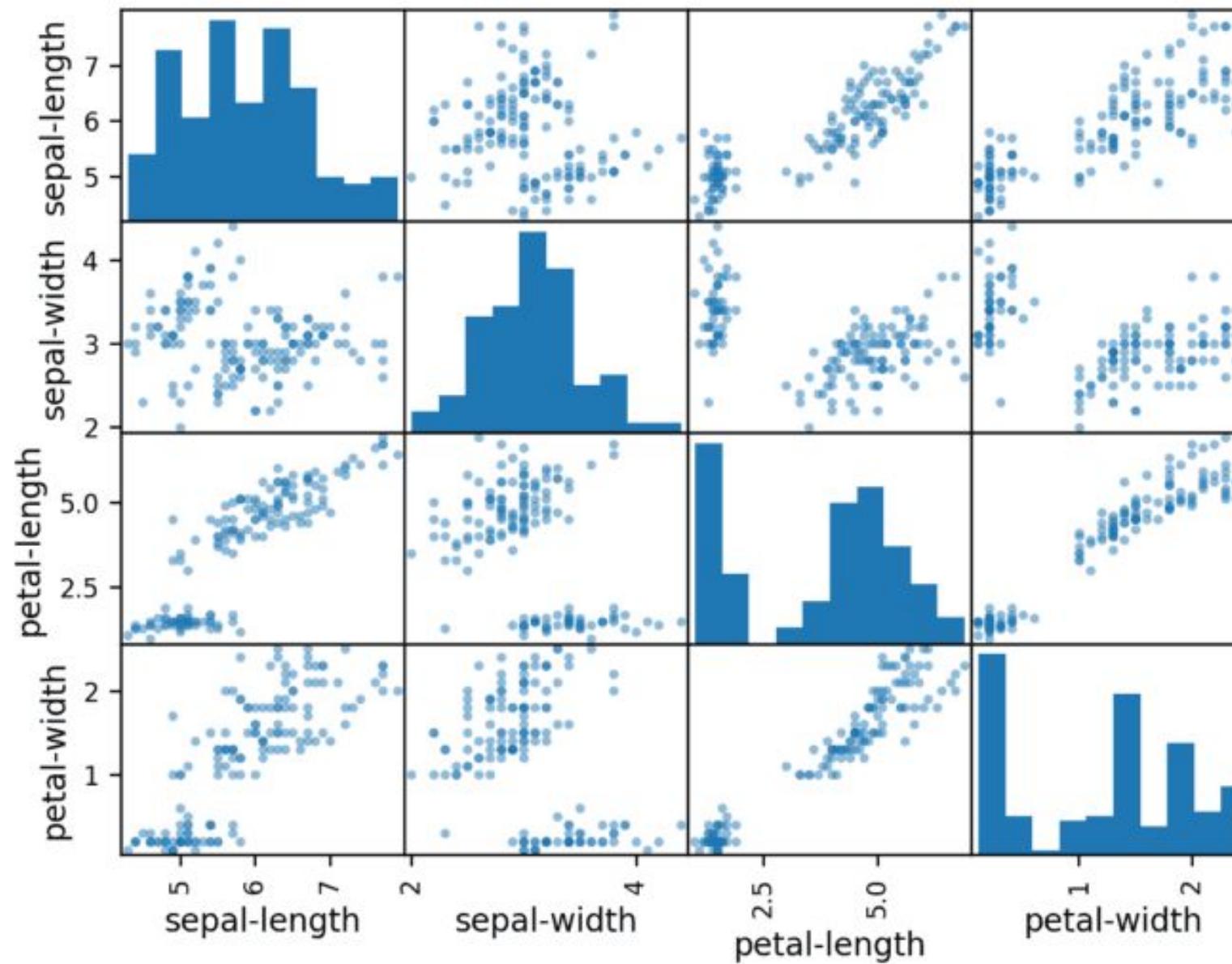
iris virginica



	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	Espécie
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

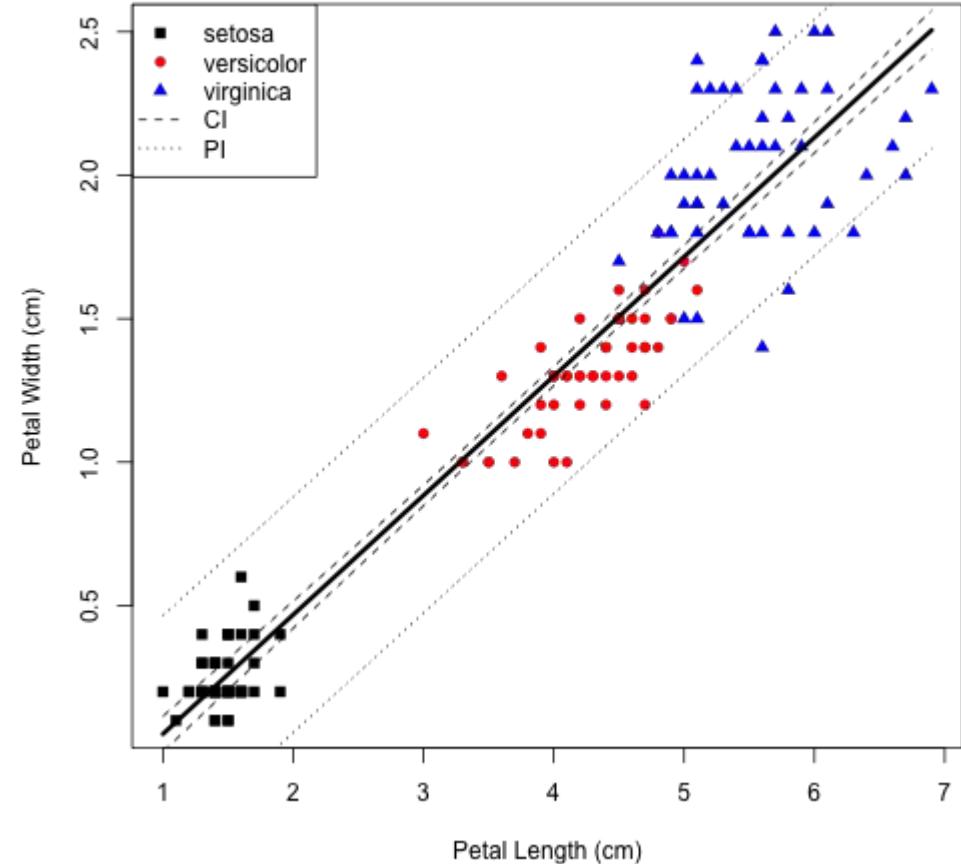
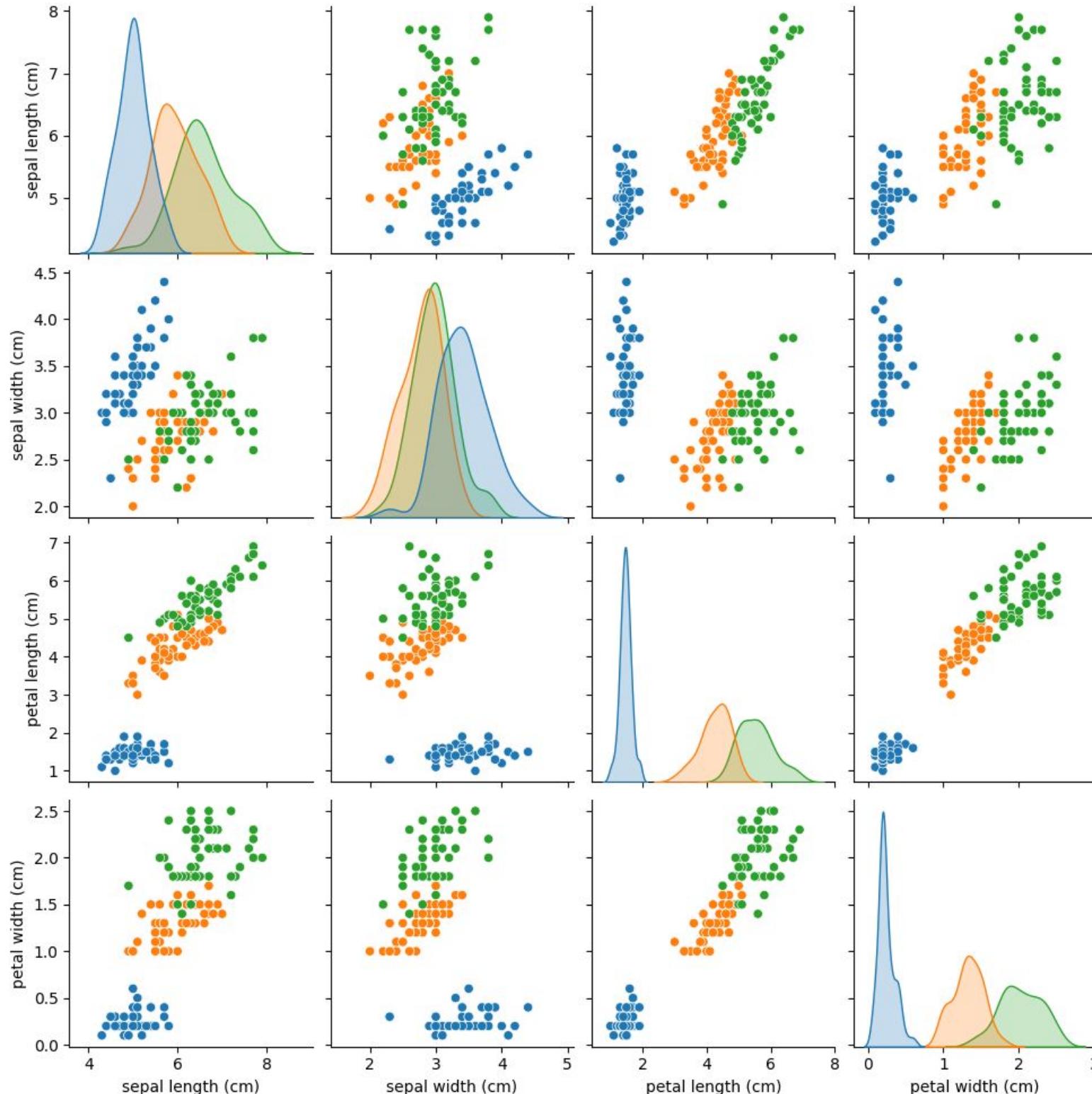
Fonte:https://scikit-learn.org/stable/auto_examples/datasets/plot_iris_dataset.html

15. Aprendizado de máquina supervisionado



Fonte:https://scikit-learn.org/stable/auto_examples/datasets/plot_iris_dataset.html

16. Aprendizado de máquina supervisionado



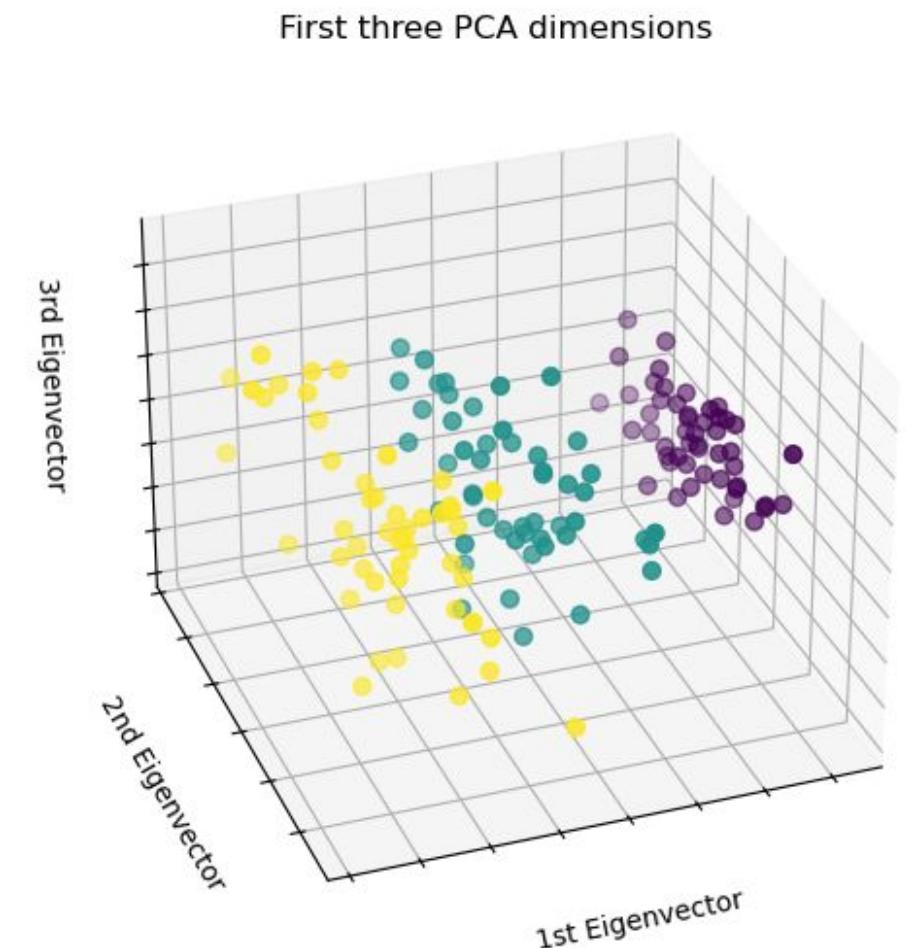
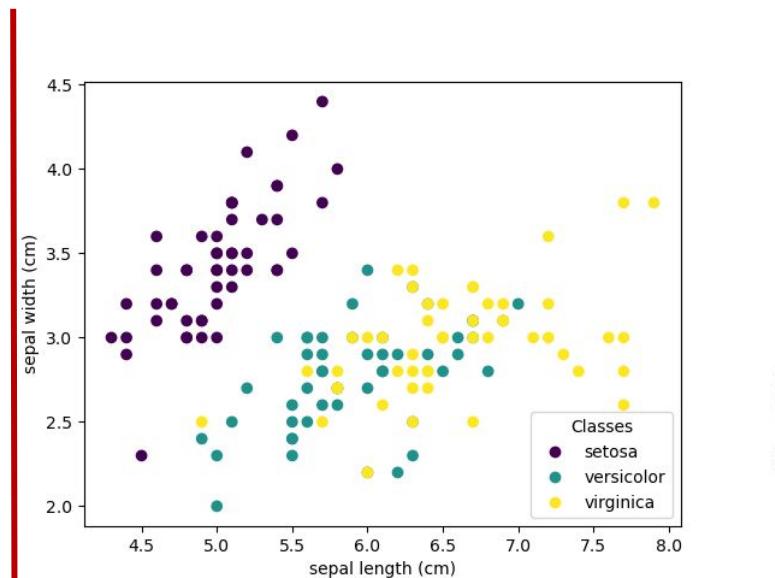
17. Aprendizado de máquina não supervisionado

Agrupamento (Clustering):

- O agrupamento é uma técnica que envolve identificar grupos de dados sem rótulos predefinidos. Os algoritmos de agrupamento tentam encontrar estruturas naturais nos dados, onde pontos de dados semelhantes são agrupados juntos.
- Exemplos de algoritmos de agrupamento incluem K-Means, DBSCAN e Hierarchical Clustering.

Redução de Dimensionalidade:

- A redução de dimensionalidade visa reduzir o número de variáveis ou dimensões nos dados, mantendo o máximo de informação possível. Isso é útil para visualização de dados, remoção de ruído e simplificação de modelos.
- Algoritmos comuns incluem Análise de Componentes Principais (PCA), t-SNE (t-distributed Stochastic Neighbor Embedding) e LDA (Linear Discriminant Analysis).



Fonte:https://scikit-learn.org/stable/auto_examples/datasets/plot_iris_da_taset.html

18. Etapa de treinamento do modelo

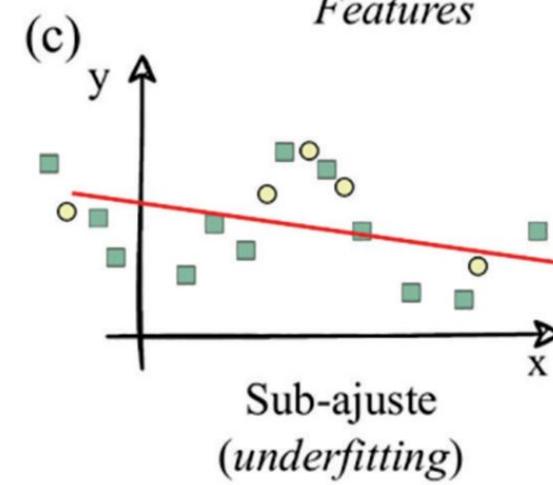
(a)

Dados							
	x_1	x_2	...	x_n	y_1	...	y_m
1	1.2	21	...	0.27	1	...	4.5
2	-4.2	44	...	0.74	0	...	5.0
.	-3.7	30	...	0.11	2	...	3.7
.
$k-1$	7.1	14	...	0.34	2	...	2.8
k	-1.5	42	...	0.55	0	...	4.2

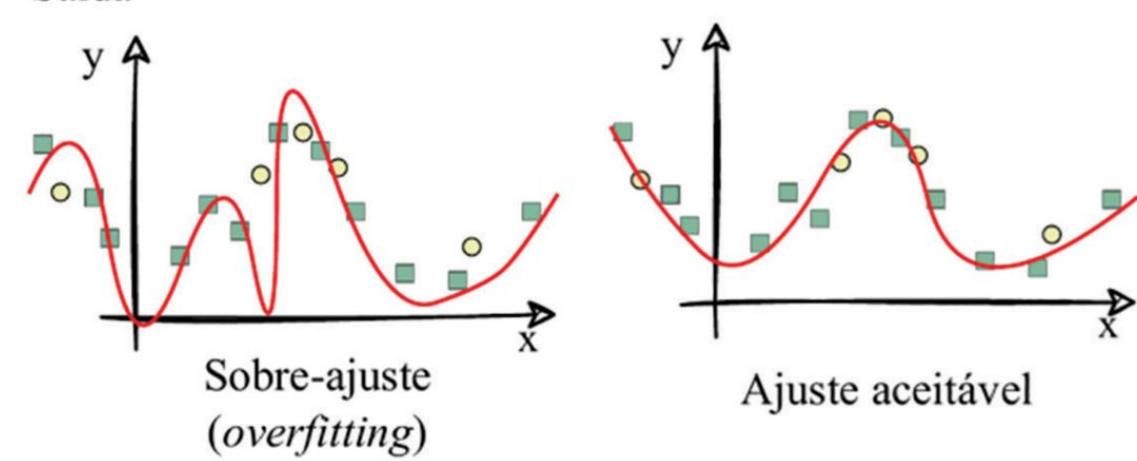
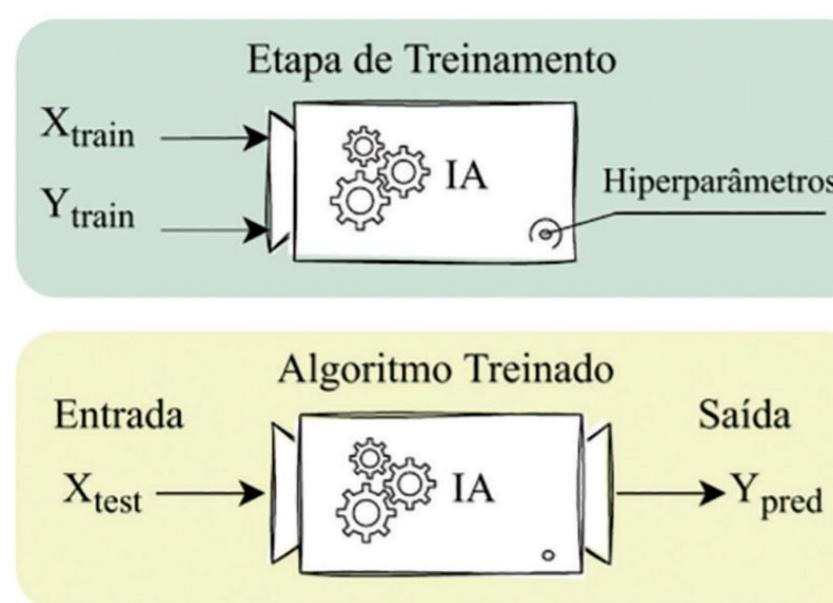
Train

Test

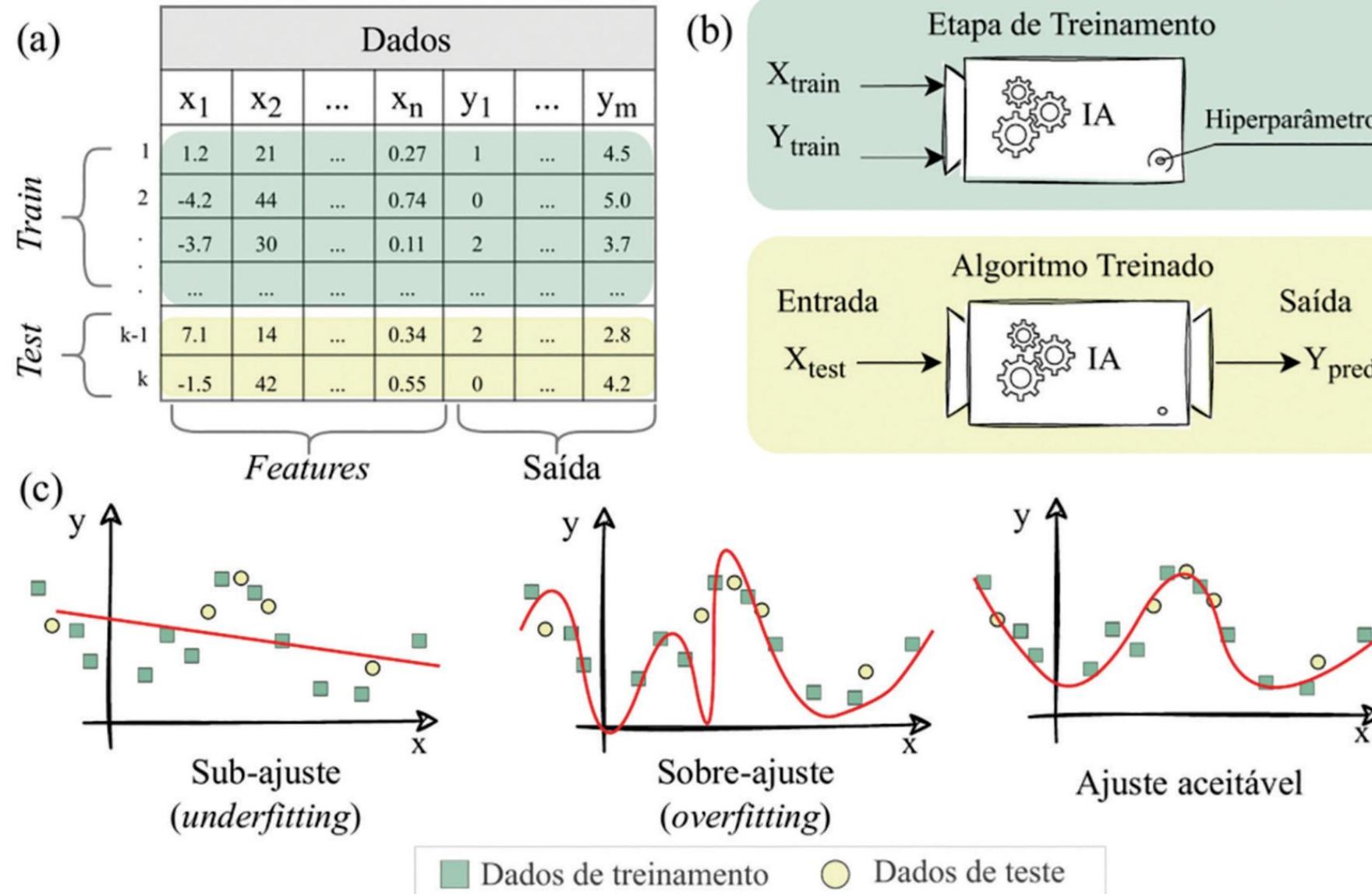
Saída



(b)



19. Etapa de treinamento do modelo



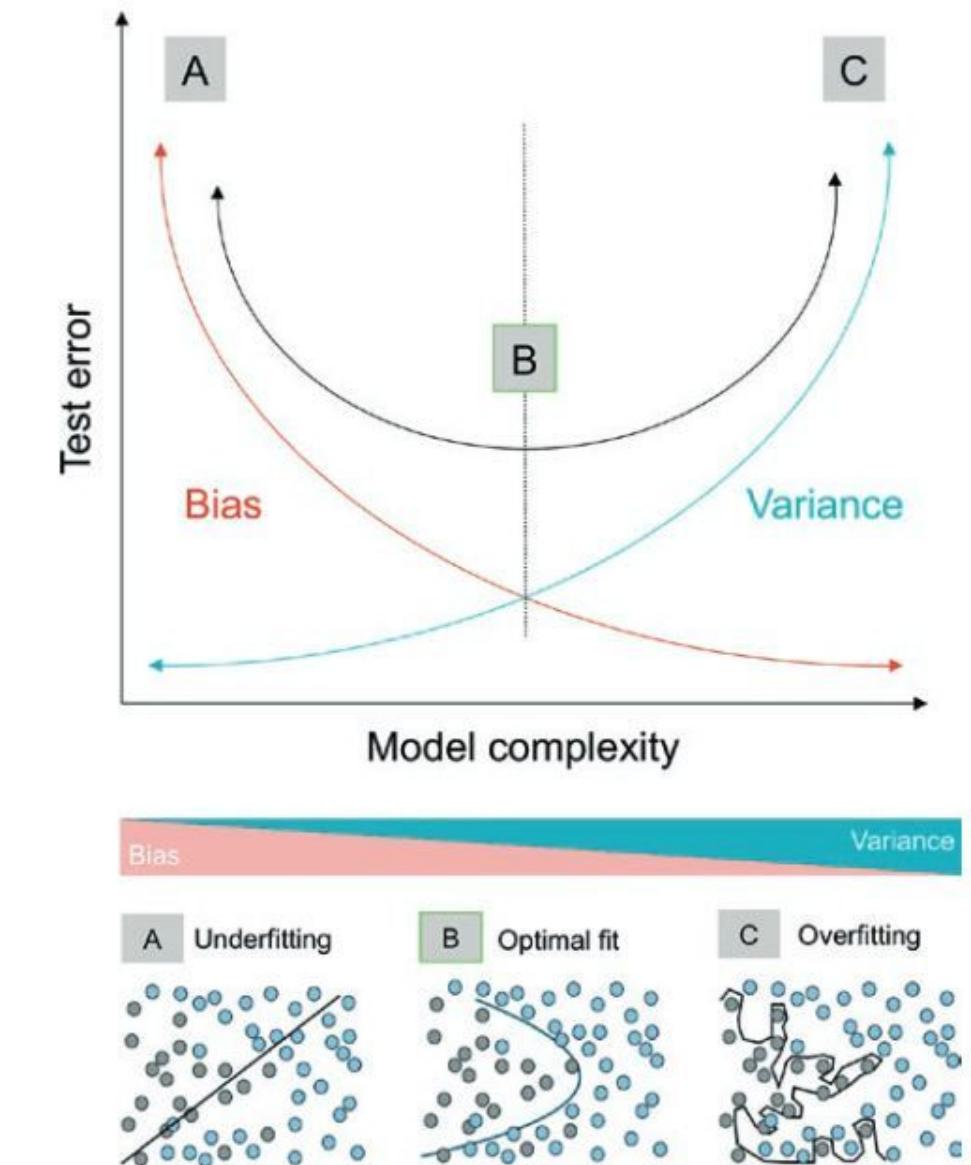
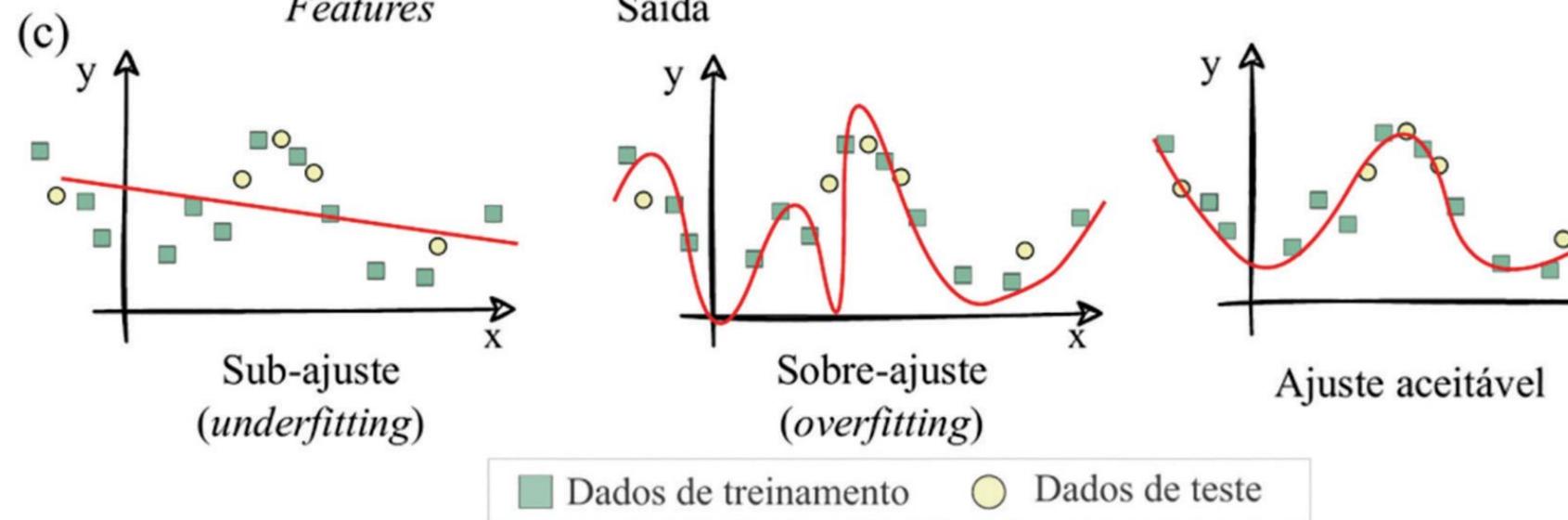
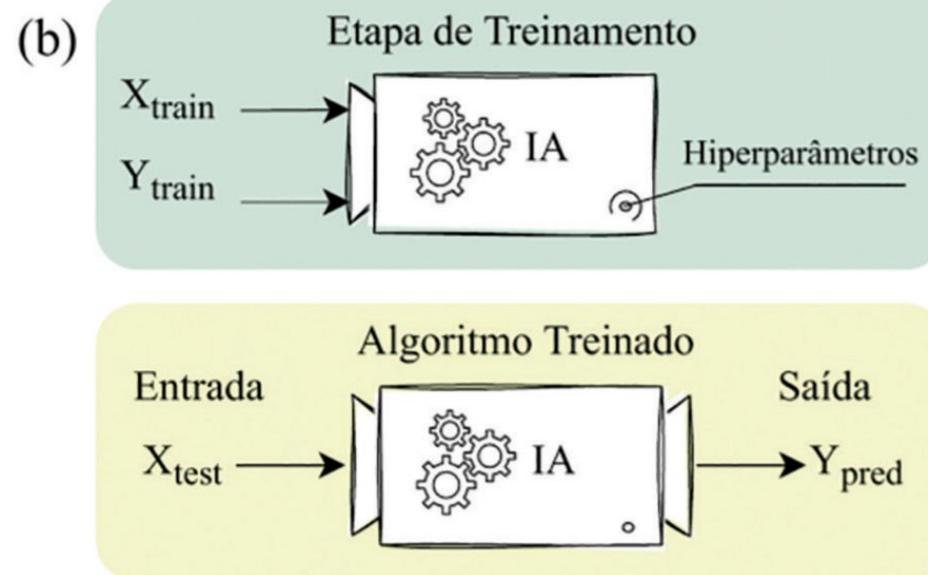
20. Etapa de treinamento do modelo

(a)

Dados							
	x_1	x_2	...	x_n	y_1	...	y_m
1	1.2	21	...	0.27	1	...	4.5
2	-4.2	44	...	0.74	0	...	5.0
.	-3.7	30	...	0.11	2	...	3.7
.
k-1	7.1	14	...	0.34	2	...	2.8
k	-1.5	42	...	0.55	0	...	4.2

Train Test

Features Saída



Fonte 1: https://www.scielo.br/j/rbef/a/J57GjGRH5HmCnqFZSpvFZHK/?lang=pt#ModalFigS1_F3

Fonte 2: <https://viso.ai/computer-vision/what-is-overfitting/>

20. Etapa de treinamento do modelo: Linear

O treino de um modelo linear envolve encontrar os melhores valores para os **pesos (coeficientes)** e o **intercepto** (ou termo de bias), de forma que o modelo possa **minimizar o erro** ao fazer previsões sobre um conjunto de dados. Aqui está uma visão geral passo a passo de como isso é feito:

1. Definição do Problema

Um modelo linear faz previsões usando uma equação do tipo:

- Para regressão linear:

$$y = w_1x_1 + w_2x_2 + \cdots + w_nx_n + b$$

Onde:

- y : saída (variável dependente ou alvo).
- x_1, x_2, \dots, x_n : variáveis de entrada (ou features).
- w_1, w_2, \dots, w_n : coeficientes (pesos) que o modelo ajustará durante o treino.
- b : intercepto (ou bias).
- Para classificação linear (Regressão Logística):

$$\hat{y} = \frac{1}{1 + e^{-(w_1x_1 + w_2x_2 + \cdots + w_nx_n + b)}}$$

Aqui, a saída (\hat{y}) é uma probabilidade, usada para prever categorias.

20. Etapa de treinamento do modelo: Linear

2. Passos para o Treinamento

(a) Preparação dos Dados

1. **Coleta e limpeza:** Obtenha os dados de entrada (X) e os valores reais de saída (y).
2. **Normalização ou padronização:** Escale as features para evitar que alguma variável com valores muito altos domine o modelo.
3. **Divisão do conjunto de dados:** Separe os dados em treino, validação e teste.

20. Etapa de treinamento do modelo: Linear

(b) Definição da Função de Custo

A função de custo mede o erro do modelo e guia o ajuste dos parâmetros (w e b).

- Para regressão linear:
 - Usa-se o Erro Quadrático Médio (MSE):

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

Onde m é o número de exemplos.

- Para classificação (regressão logística):
 - Usa-se a Entropia Cruzada (Log Loss):

$$J(w, b) = -\frac{1}{m} \sum_{i=1}^m [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]$$

20. Etapa de treinamento do modelo: Linear

O objetivo é minimizar a função de custo $J(w, b)$ ajustando os valores dos pesos (w) e do intercepto (b).

1. Escolha de um Algoritmo de Otimização:

- O método mais comum é o **Gradiente Descendente**, que ajusta os pesos com base no gradiente da função de custo:

$$w_j := w_j - \alpha \frac{\partial J(w, b)}{\partial w_j}$$

$$b := b - \alpha \frac{\partial J(w, b)}{\partial b}$$

Onde:

- w_j : peso para a feature j .
- α : taxa de aprendizado, que controla o tamanho dos passos.
- $\frac{\partial J(w, b)}{\partial w_j}$: derivada parcial da função de custo em relação ao peso w_j .

20. Etapa de treinamento do modelo: Linear

2. Iteração (Épocas):

- O processo de ajuste dos pesos é repetido até que:
 - O erro atinja um valor aceitável.
 - Ou o número máximo de iterações (épocas) seja alcançado.

(d) Avaliação do Modelo

Após o treinamento:

1. Use o conjunto de validação ou teste para verificar o desempenho do modelo.

2. Métricas comuns:

- Para regressão: Erro Médio Absoluto (MAE), Raiz do Erro Quadrático Médio (RMSE), ou R^2 .
- Para classificação: Acurácia, Precisão, Recall, F1-Score, ou AUC-ROC.

Introdução: Entendimento do Problema

$$\vec{F} = m\vec{a}_t = \vec{P} - \vec{f} \quad \text{Eq. 1}$$

$$m \cdot a_t = mg \sin(\theta) - f \quad \text{Eq. 2}$$

$$\vec{\tau} = I \cdot \vec{\alpha} = \vec{f} \times \vec{R} \quad \text{Eq. 3}$$

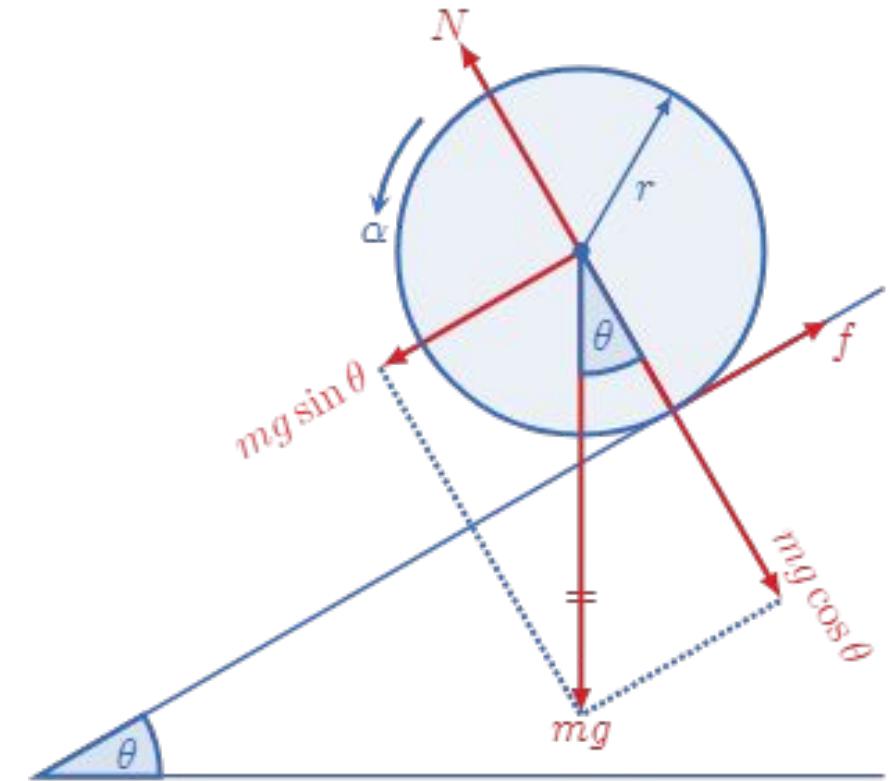
$$I \cdot \alpha = f \cdot R \quad \text{Eq. 4}$$

$$f = I \frac{\alpha}{R} \quad \text{Eq. 5}$$

$$a_t = \alpha \cdot R \quad \text{Eq. 6}$$

→

$$f = I \frac{a_t}{R^2} \quad \text{Eq. 7}$$



Introdução: Entendimento do Problema

$$m \cdot a_t = mgsin(\theta) - f \quad \text{Eq. 2}$$

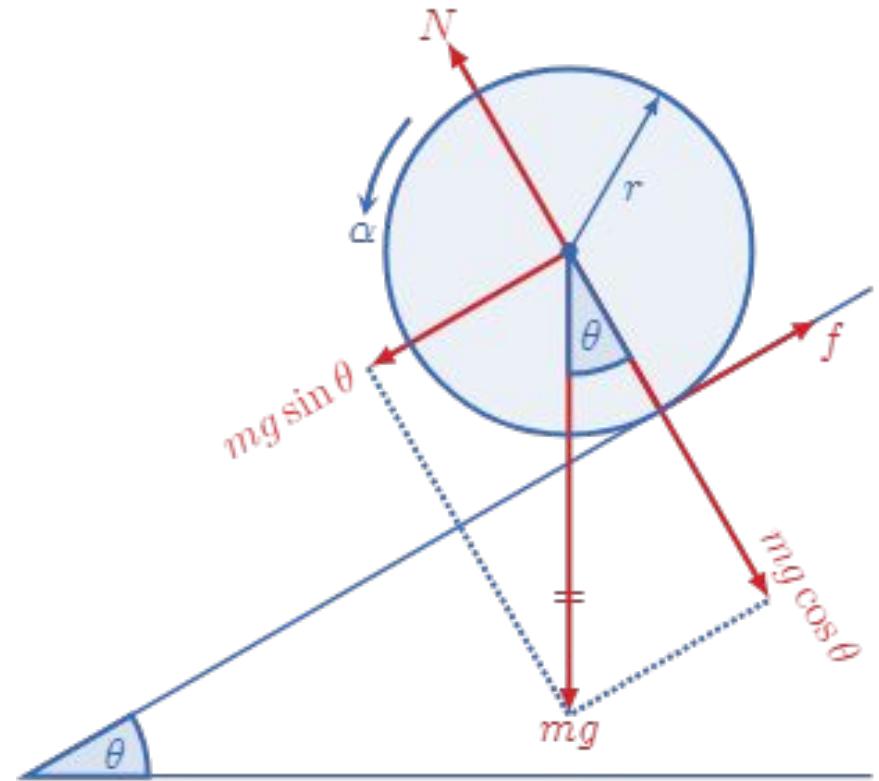
$$f = I \frac{a_t}{R^2} \quad \text{Eq. 7}$$

Substituindo Eq. 7 em Eq. 2

$$m \cdot a_t = mgsin(\theta) - I \frac{a_t}{R^2} \quad \text{Eq. 8}$$

Resolvendo a Eq. 8 para a_t , temos:

$$a_t = \frac{gsin(\theta)}{1 + \frac{I}{mR^2}} \quad \text{Eq. 9}$$



Introdução: Entendimento do Problema

$$a_t = \frac{gsin(\theta)}{1 + \frac{I}{mR^2}}$$

Eq. 9

Usando a relação de Torricelli

$$h = \Delta S \cdot sin(\theta)$$

Eq. 10

$$v^2 = v_0^2 + 2a_t \Delta S$$

Eq. 11

Substituindo Eq. 9 e 10 em Eq. 11

$$v^2 = \frac{2gsin(\theta)\Delta S}{1 + \frac{I}{mR^2}}$$

→

$$v = \sqrt{\frac{2gsin(\theta)\Delta S}{1 + \frac{I}{mR^2}}}$$

→

$$v = \sqrt{\frac{2gh}{1 + \frac{I}{mR^2}}}$$

