

# Sistemas Inteligentes

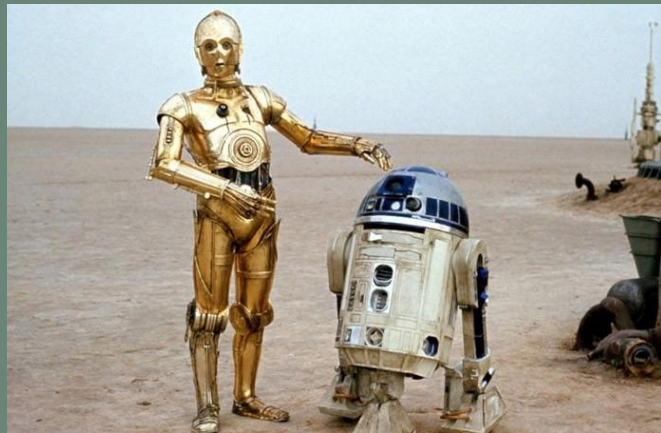
## Background

*Profa: Deborah Magalhães*



**O que é aprendizado de máquina?**

O que as pessoas pensam que é...



O que realmente é ...



“

*Um campo de estudo que confere  
ao computador a habilidade de  
aprender sem ter sido  
explicitamente programado  
para isso*

“

*Um programa de computador aprendeu com uma experiência  $E$ , relacionada à uma classe de tarefas  $T$  e desempenho  $P$ , se o desempenho da tarefa  $T$  medido por  $P$ , melhora a experiência  $E$*

## Como nós conhecemos os dados ...

| ◇ | A     | B        | C        | D        |
|---|-------|----------|----------|----------|
| 1 |       | Column 1 | Column 2 | Column 3 |
| 2 | Row 1 | 2.2      | 2.3      | 1        |
| 3 | Row 2 | 2.3      | 2.6      | 0        |
| 4 | Row 3 | 2.1      | 2        | 1        |
| E |       |          |          |          |

## Perspectiva estatística ...

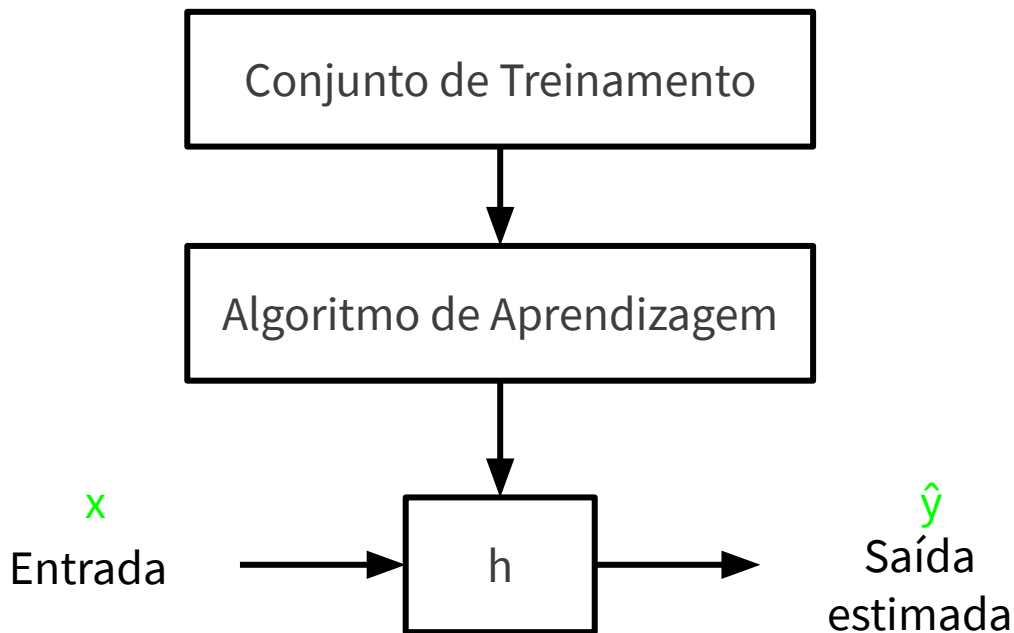
$$OutputVariable = f(InputVariables)$$

| ◇ | A   | B   | C |
|---|-----|-----|---|
| 1 | X1  | X2  | Y |
| 2 | 2.2 | 2.3 | 1 |
| 3 | 2.3 | 2.6 | 0 |
| 4 | 2.1 | 2   | 1 |
| 5 |     |     |   |

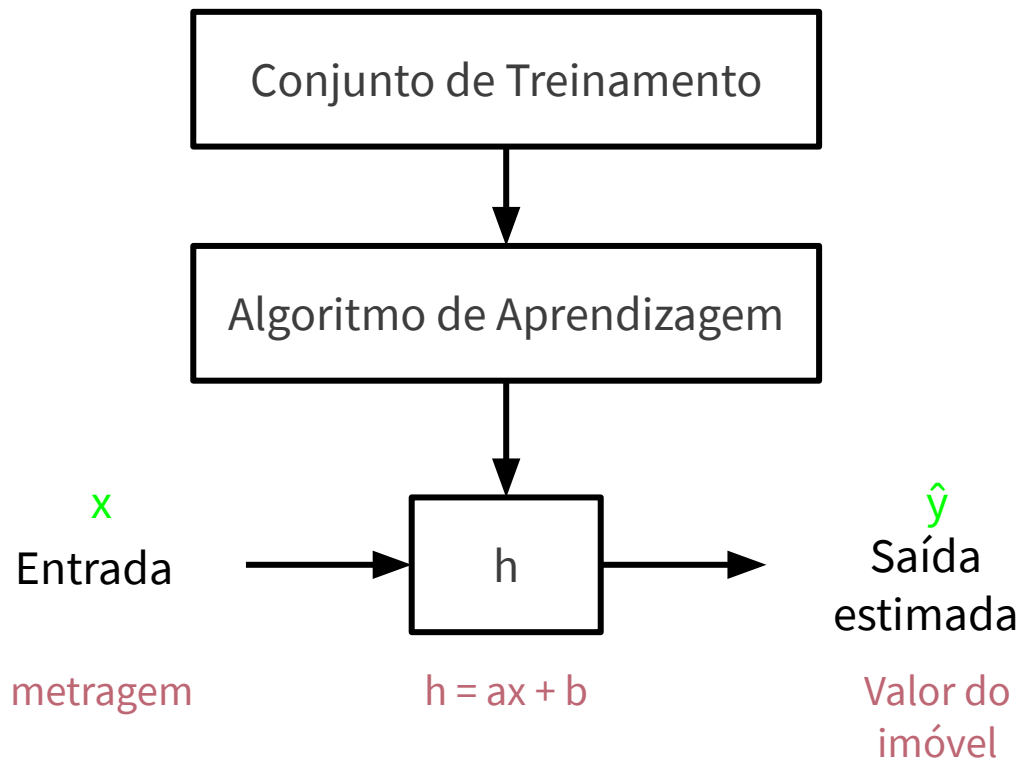


# Hipótese:

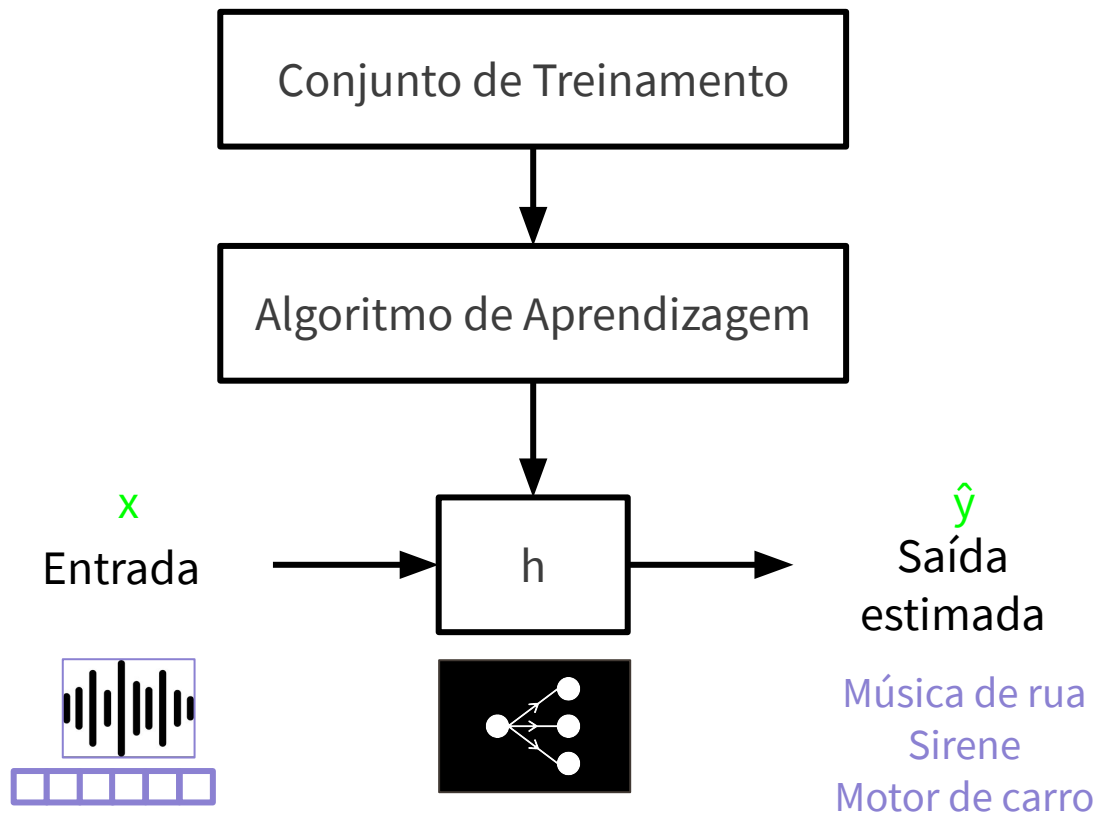
O modelo  $X$  é bom  
o suficiente  
para representar  
os dados?



# Hipótese



# Hipótese



# Como ensinar o computador?

# Algoritmos Paramétricos

1. Seleciona uma determinada função para representar os dados
2. Aprende os coeficientes da função a partir do treinamento

$$h = b_0 + b_1 \cdot x_1 + b_2 \cdot x_2$$

# Algoritmos Paramétricos

1. Seleciona uma determinada função para representar os dados
2. Aprende os coeficientes da função a partir do treinamento

$$y = b_0 + b_1 \cdot x_1 + b_2 \cdot x_2$$

**Coeficientes**

# Vantagens e Desvantagens dos Algoritmos Paramétricos

**Simplicidade**

**Rapidez**

**Menos dados**

**Limitação**

**Baixa complexidade**

**Ajuste ruim**

# Algoritmos Não Paramétricos

- Assumem pouco ou nada sobre a função de mapeamento, eles a **constroem** durante o treinamento
  - Exemplos: Support Vector Machine e Redes Neurais



# Vantagens e Desvantagens dos Algoritmos Não Paramétricos

**Flexível**

**Poderoso**

**Eficiência na  
previsão**

**Mais dados**

**Treinamento lento**

**Overfitting**

# Tipos de aprendizado

**Supervisionado**

**Não supervisionado**

**Semi-supervisionado**

**Aprendizado por  
reforço**

# Tipos de aprendizado

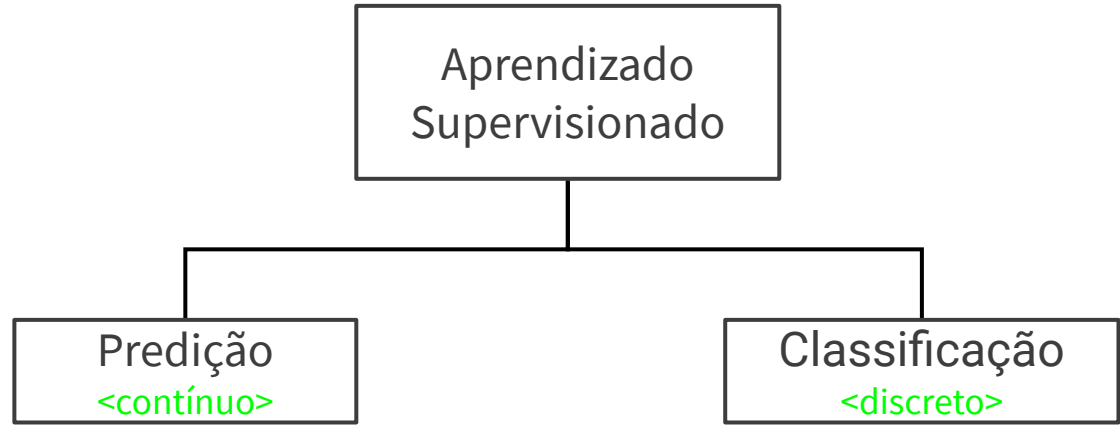
**Supervisionado**

**Não supervisionado**

**Semi-supervisionado**

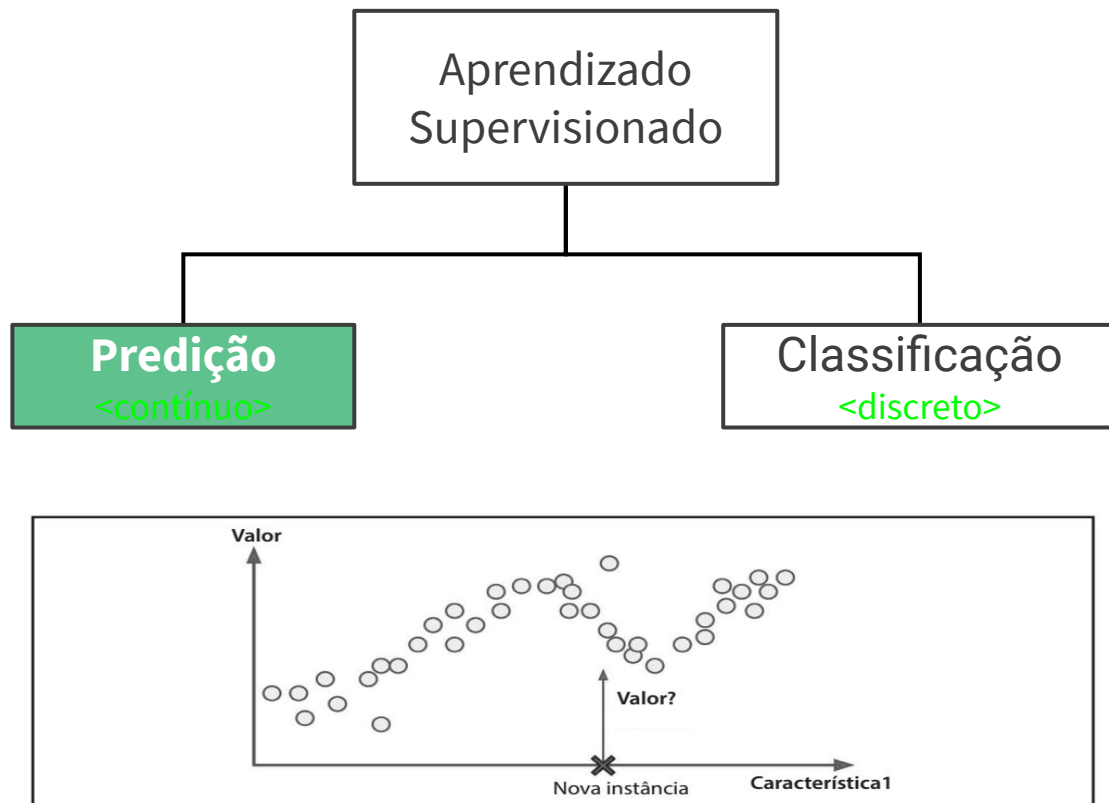
**Aprendizado por  
reforço**

# Aprendizado Supervisionado



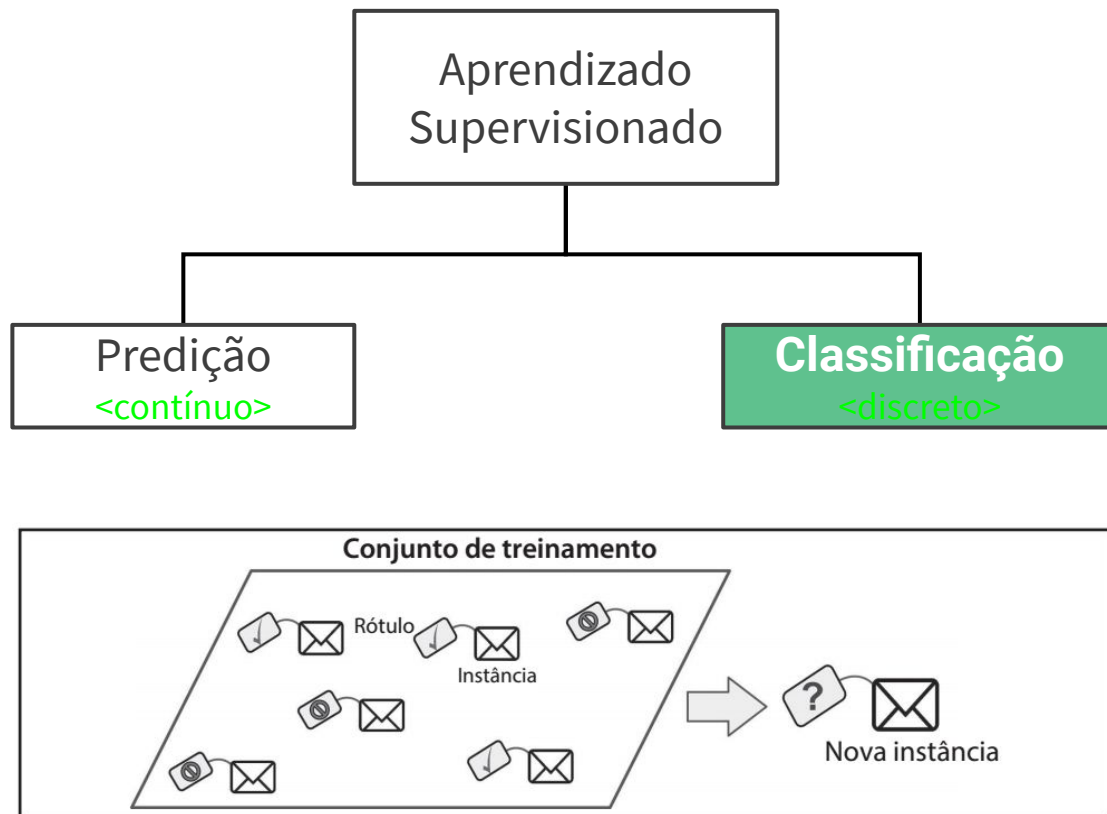
# Aprendizado Supervisionado

## Predição



# Aprendizado Supervisionado

## Classificação



# **Aprendizado Supervisionado**

## **Algoritmos**

- Regressão (Linear e Logística)
- Máquinas de Vetores de Suporte (SVM)
- Florestas Aleatórias (Random Forest)
- Redes Neurais

# Tipos de aprendizado

**Supervisionado**

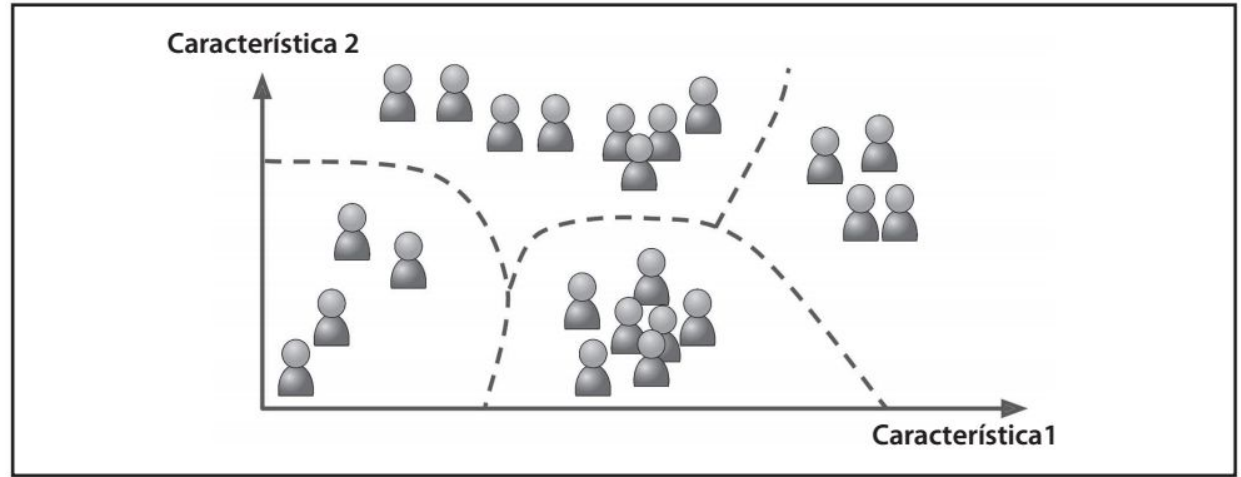
**Não supervisionado**

**Semi-supervisionado**

**Aprendizado por  
reforço**



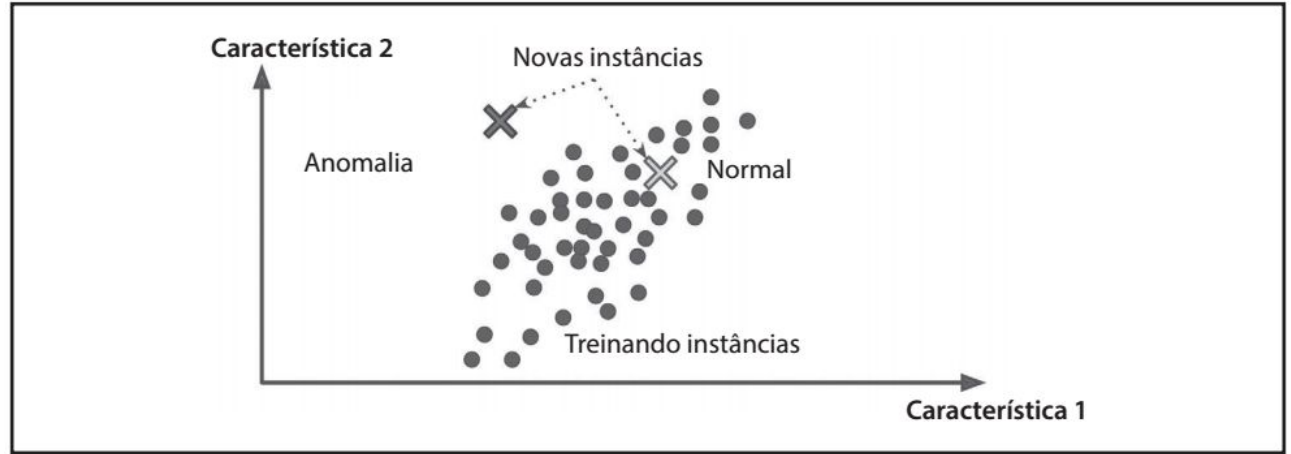
# Aprendizado Não Supervisionado Clustering



*Clustering*

# Aprendizado Não Supervisionado

## Detecção de Anomalia



*Detecção de Anomalia*

# **Aprendizado Não Supervisionado**

## **Algoritmos**

- Clustering
  - k-Means
- Redução de dimensionalidade
  - Análise dos Componentes Principais (PCA)

# Tipos de aprendizado

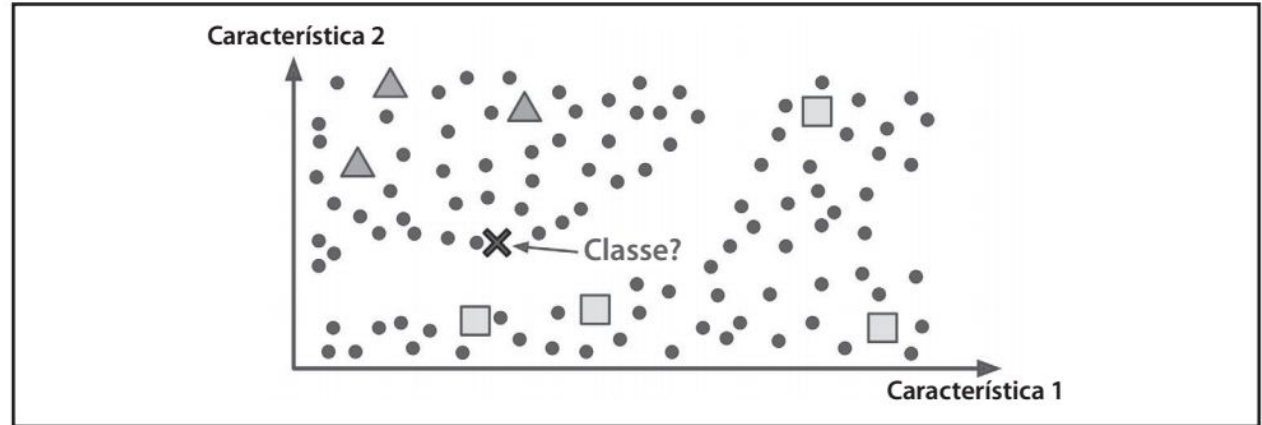
**Supervisionado**

**Não supervisionado**

**Semi-supervisionado**

**Aprendizado por  
reforço**

# Aprendizado Semi-supervisionado



# Tipos de aprendizado

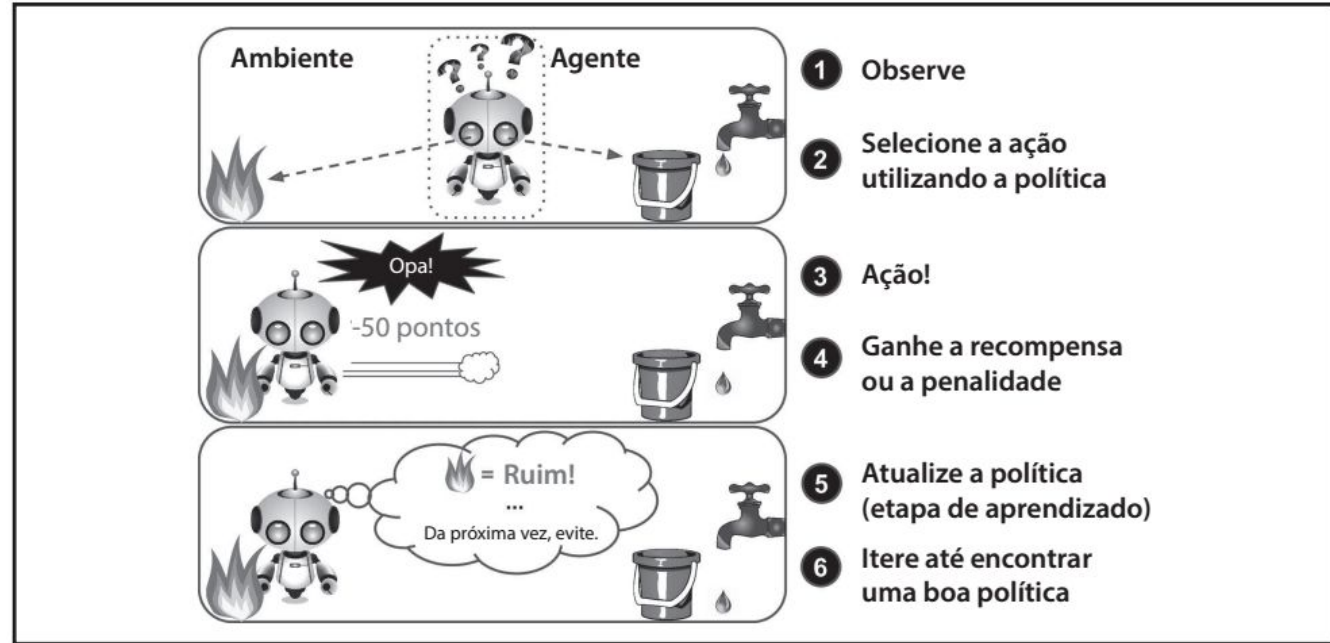
**Supervisionado**

**Não supervisionado**

**Semi-supervisionado**

**Aprendizado por  
reforço**

# Aprendizado por reforço



# Tipos de aprendizado

**Supervisionado**

**Não supervisionado**

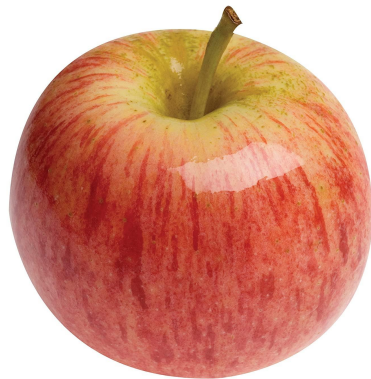
**Semissupervisionado**

**Aprendizado por  
reforço**

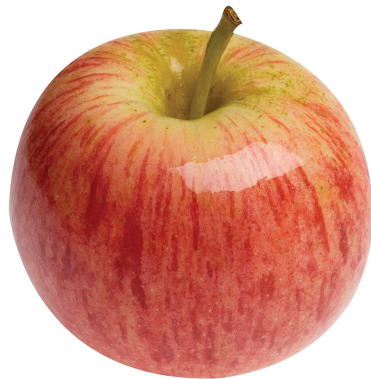


**Quais as dificuldades de ensinar o computador a reconhecer padrões nos dados?**

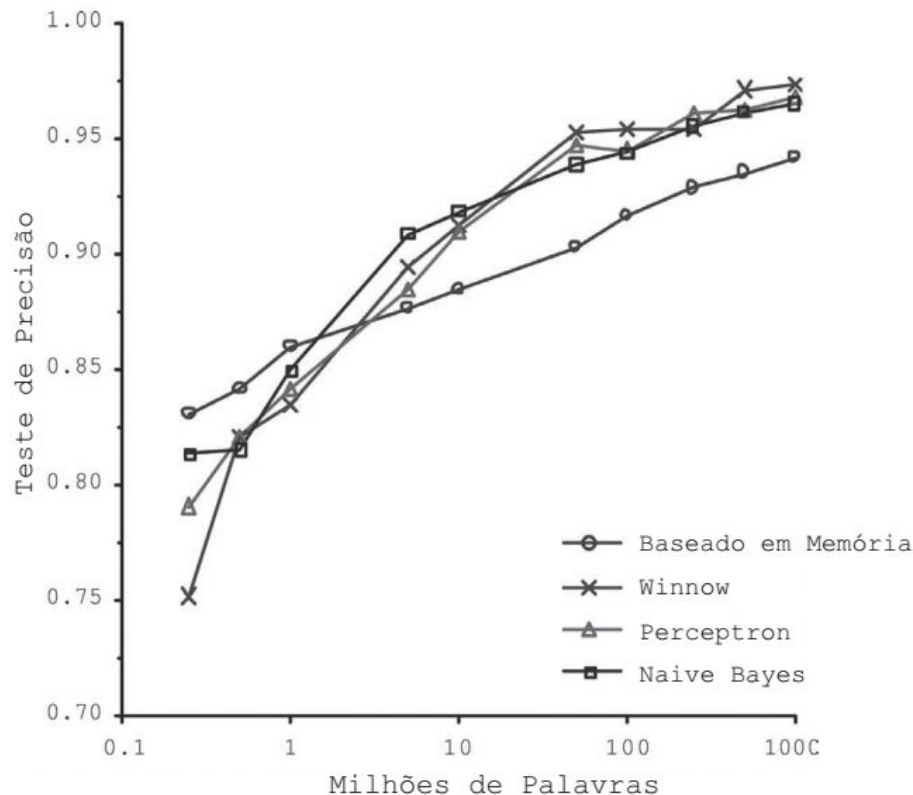
# #1- Poucos dados de treino



# #1- Poucos dados de treino

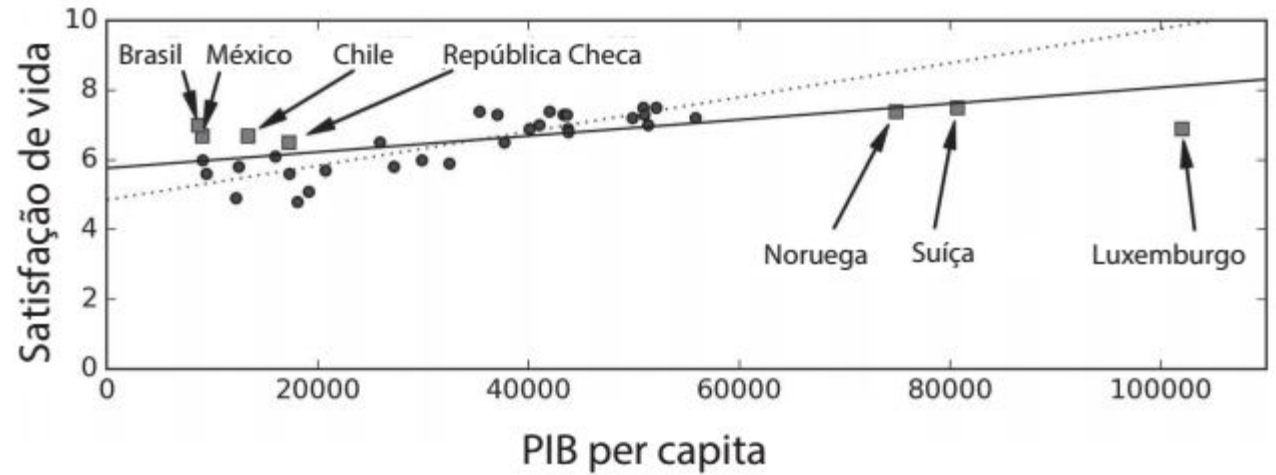


# #1- Poucos dados de treino



BANKO, Michele; BRILL, Eric. Scaling to very very large corpora for natural language disambiguation. In: **Proceedings of the 39th annual meeting of the Association for Computational Linguistics**. 2001. p. 26-33.

## #2- Dados não representativos

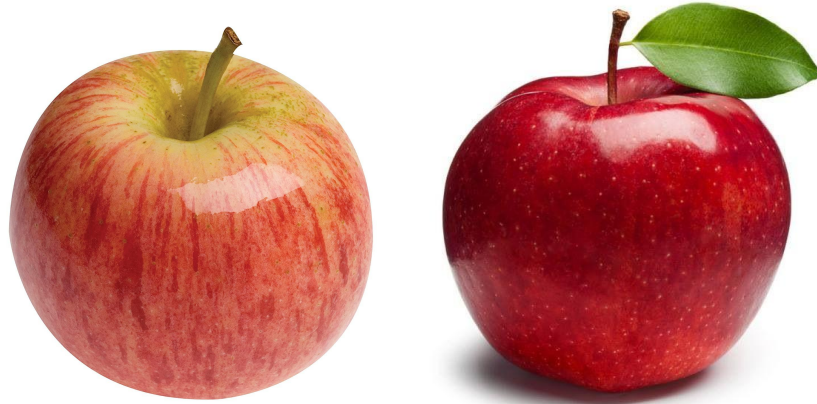


### **#3- Dados de baixa qualidade**





# #4 - Características irrelevantes



O sistema **NÃO** terá um bom funcionamento se o seu conjunto de treinamento for muito **pequeno** ou se os dados forem **não representativos, ruidosos** ou **poluídos** com **características irrelevantes** (entra lixo, sai lixo).

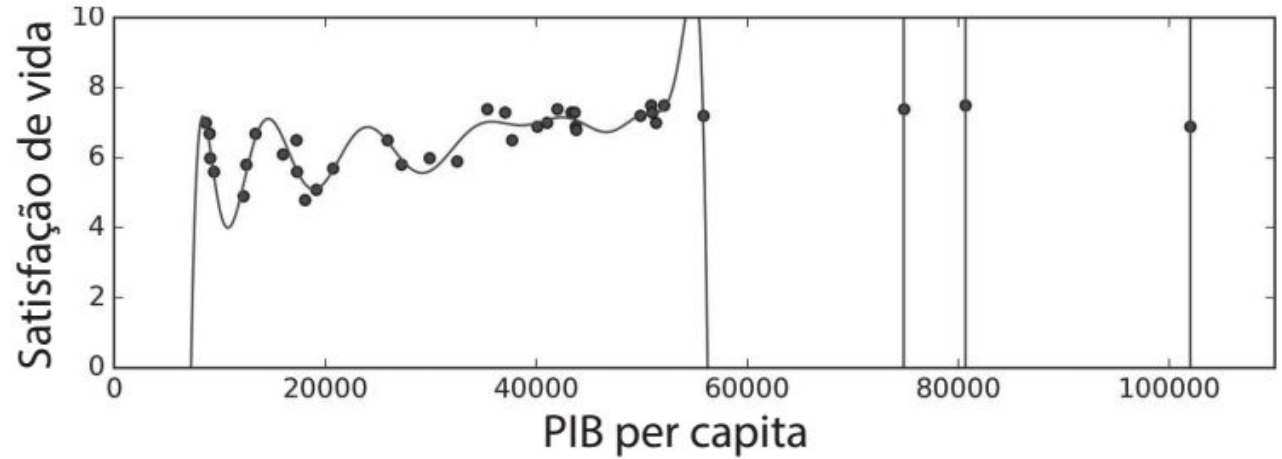


**Até agora falamos dos dados.  
Mas e os algoritmos?**

## #5- Overfitting e Downfitting



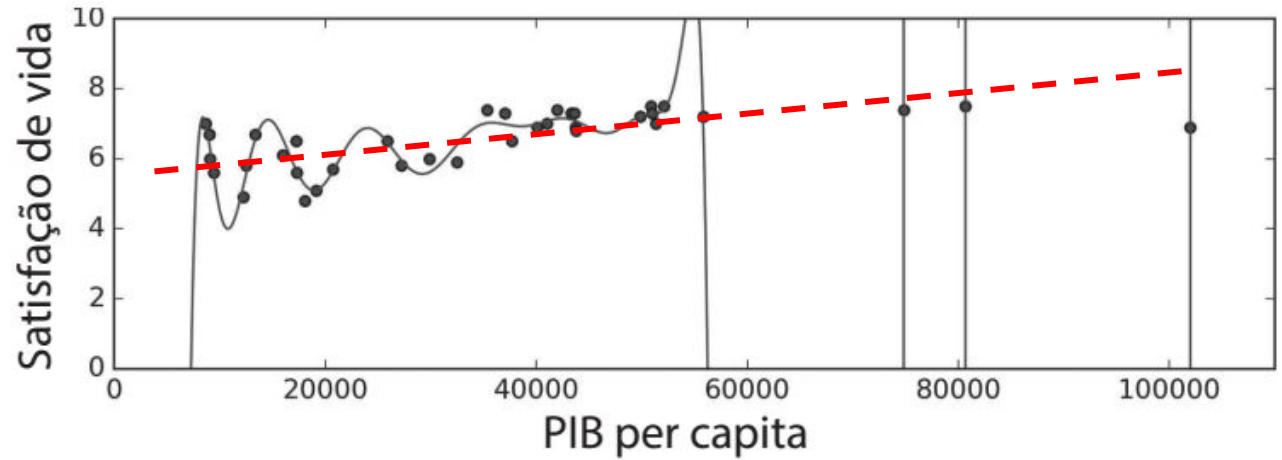
## #5 - Sobreajuste (overfitting)



## #5 - O que fazer quando temos um modelo sobreajustado?

- **Simplificar** o modelo ao selecionar aquele com menos parâmetros
- **Coletar** mais dados de treinamento
- **Reduzir** o ruído nos dados de treinamento (eliminar erros de dados e remover outliers)

## #5 - Subajuste (under/down fitting)



## #5 - O que fazer quando temos um modelo subajustado?

- Selecionar um modelo mais poderoso, com mais parâmetros;
- Alimentar o algoritmo de aprendizado com melhores características



# Muito Obrigada!

Se você tiver qualquer dúvida ou sugestão:

- [deborah.vm@ufpi.edu.br](mailto:deborah.vm@ufpi.edu.br)

