

---

# Inteligência Artificial

— Introdução —

---

# Conteúdo - Aula 1

- Conhecer o perfil de todos e conversar sobre a disciplina;
- Introdução ao conceito de aprendizado de máquina;
- Tipos de algoritmo aprendizado de máquina;
- Fluxo de uma aplicação em aprendizado de máquina;
- Estratégias para separação de dados e validação dos modelos;

# Fernando Vieira da Silva

- Formação:

- Tec. Processamento de Dados (FATEC)
- Mestre Ciência da Computação (IC/UNICAMP)
- Doutorado Ciência da Computação (IC/UNICAMP)

- Atuação:

- Processamento de Linguagem Natural
- Pesquisador Líder na n2vec

# Quem são vocês?

- Formação
- O que espera da Pós graduação?
- Quais são suas expectativas quanto às aplicações de inteligência artificial?
- Qual sua experiência com Python? Quais bibliotecas conhece?
- Já tem ideia do que fará como TCC?

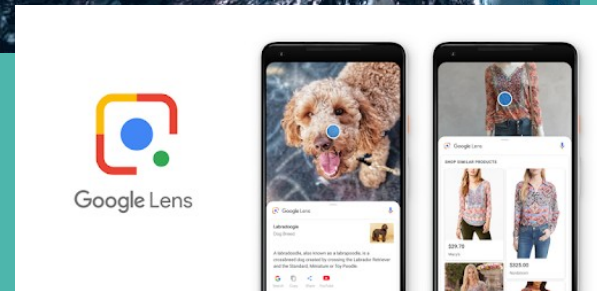
# Avaliação

- 4 Encontros:

- 6,0: Apresentação de atividades (2,0 x 3)
- 4,0: Atividade no último encontro

Ao final de cada encontro, será discutido um problema que deverá ser estudado e testado até o próximo encontro. Os resultados encontrados serão apresentados no começo do encontro seguinte, em formato de pitch (até 5 minutos).

# O que é Inteligência artificial?





The diagram consists of three concentric circles. The outermost circle is blue and represents Artificial Intelligence. Inside it is a teal circle representing Machine Learning. The innermost circle is orange and represents Deep Learning. This visualizes that Deep Learning is a subset of Machine Learning, which is a subset of Artificial Intelligence.

# **ARTIFICIAL INTELLIGENCE**

Programs with the ability to learn and reason like humans

## **MACHINE LEARNING**

Algorithms with the ability to learn without being explicitly programmed

## **DEEP LEARNING**

Subset of machine learning in which artificial neural networks adapt and learn from vast amounts of data

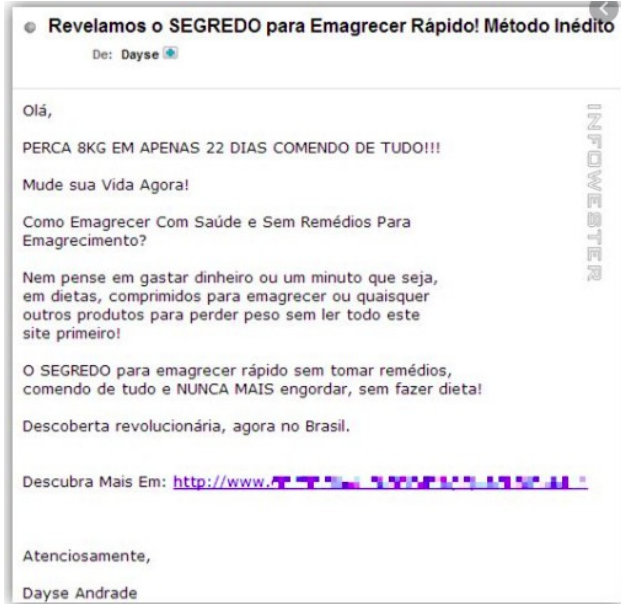
# Definição

- Field of study that gives computers the ability to learn without being explicitly programmed. (Arthur Samuel, 1959)
- A computer program is said to learn from experience **E** with respect to some task **T** and some performance measure **P**, if its performance on **T**, as measured by **P**, improves with experience **E**. (Tom Mitchell, 1997)

**Habilitar computadores a aprenderem através de dados**



# Tarefa



ENC: Segundo aviso de fatura em atraso # 879748475

Entrada x



VIVO

Adic. a circ.



VIVO <cobrancaviqrba4380@terra.com.br>

para mim

00:05 (Há 55 minutos)



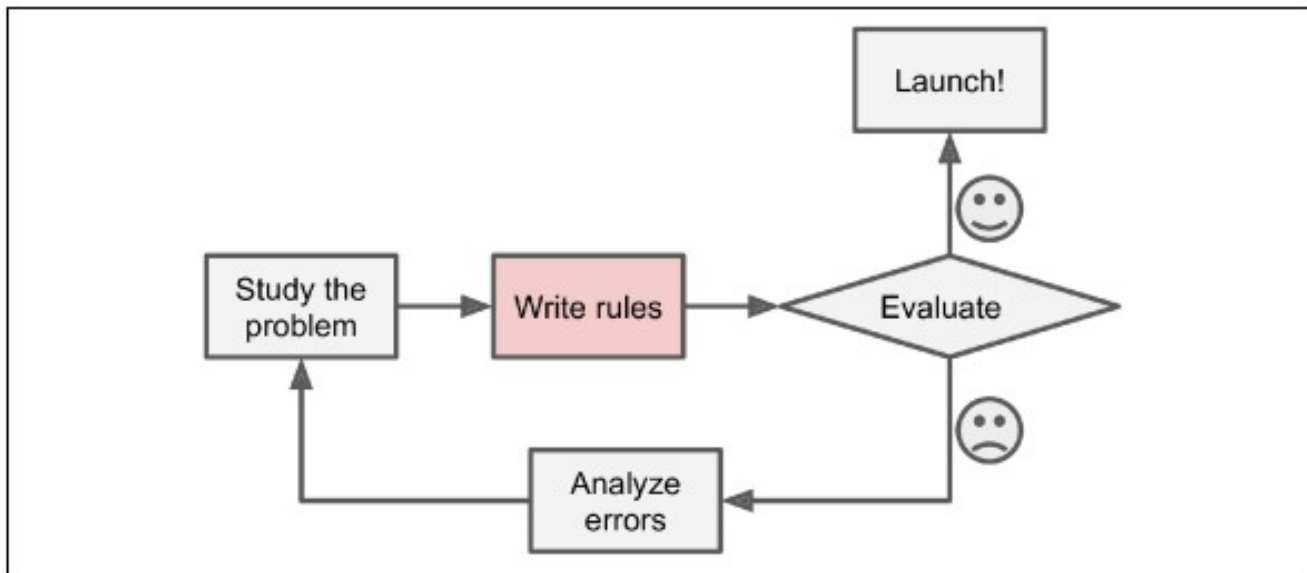
Mostrar detalhes

Prezado Cliente, Informamos que será debitado em 07/05/2016 do debito automático de sua conta, o valor de R\$ 228,14 referente a um debito que consta em seu nome em nosso sistema. Para mais detalhes consulte fatura 231566818 detalhada em anexo.



# Caso: Classificador de SPAM

Abordagem convencional



Fonte: Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow

# E se?

## **Novos padrões de e-mails SPAM foram criados?**

1. Extrair novos padrões do conteúdo de e-mails classificados como SPAM
2. Adicionar uma nova regra para classificar um e-mail como SPAM
3. Manter uma quantidade cada vez maior de regras de classificação

# A abordagem utilizando aprendizado de máquina

Dados os exemplos de e-mails (Indicados manualmente) HAM e SPAM, o sistema é capaz de aprender a identificar novos e-mails como SPAM, onde:

- Os dados são um conjunto de e-mails
- O sistema deve aprender diferenciar os e-mails SPAM dos e-mails HAM
- Deve ser adotada uma métrica para avaliar o sistema

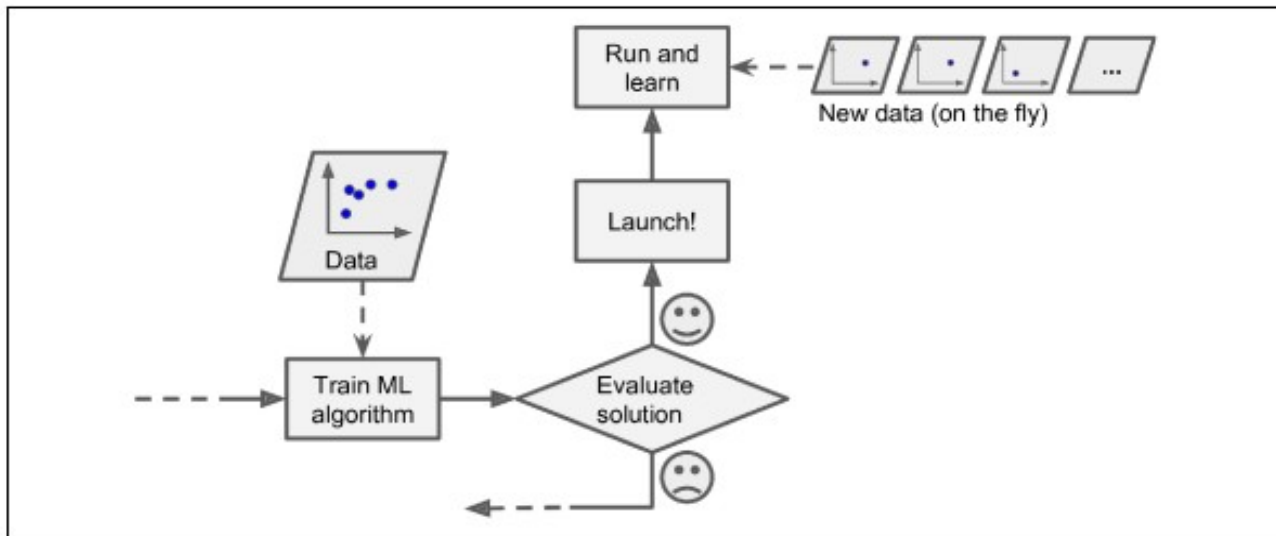
Dados de treinamento

HAM e-mails

SPAM e-mails

# Por que utilizar aprendizado de máquina?

Abordagem com aprendizado de máquina



Fonte: Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow

# Tipos de sistemas de aprendizado de máquina

# Aprendizado supervisionado

Os dados de treinamento são fornecidos junto com a saída esperada (**Label / Target variable**)

- Classificação: Classificar os dados de entrada, entre 2 ou mais categorias



- Regressão: Tem como objetivo, através de características de entrada, prever um valor discreto (numérico)

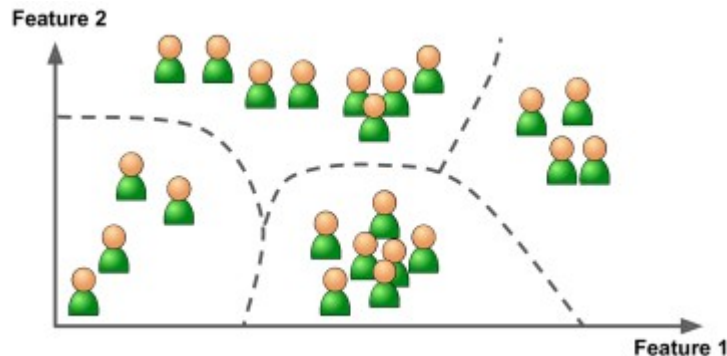
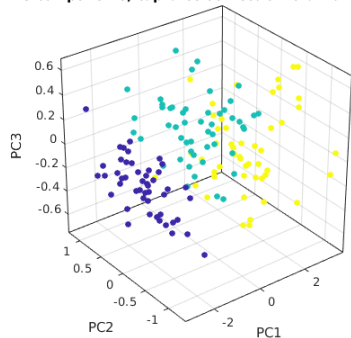


# Aprendizado não-supervisionado

Os dados de treinamento fornecidos ao sistema não contêm a resposta desejada do sistema

- Agrupamento
- Redução de dimensões
- Detecção de anomalias

3 components, captures 99.48% of total variation



Fonte: Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow



# Outros tipos de aprendizado

- Aprendizado semi-supervisionado: Ex.: Google Photos
- Aprendizado por reforço: Ex.: Robótica móvel



Fonte: Laboratório de robótica móvel ICM-USP

# Aprendizado baseado em exemplo vs modelo

- Exemplo

- Sistema carrega os dados e utiliza para classificar novos dados através de medidas de similaridade (Ex.: K-nn)

- Modelo

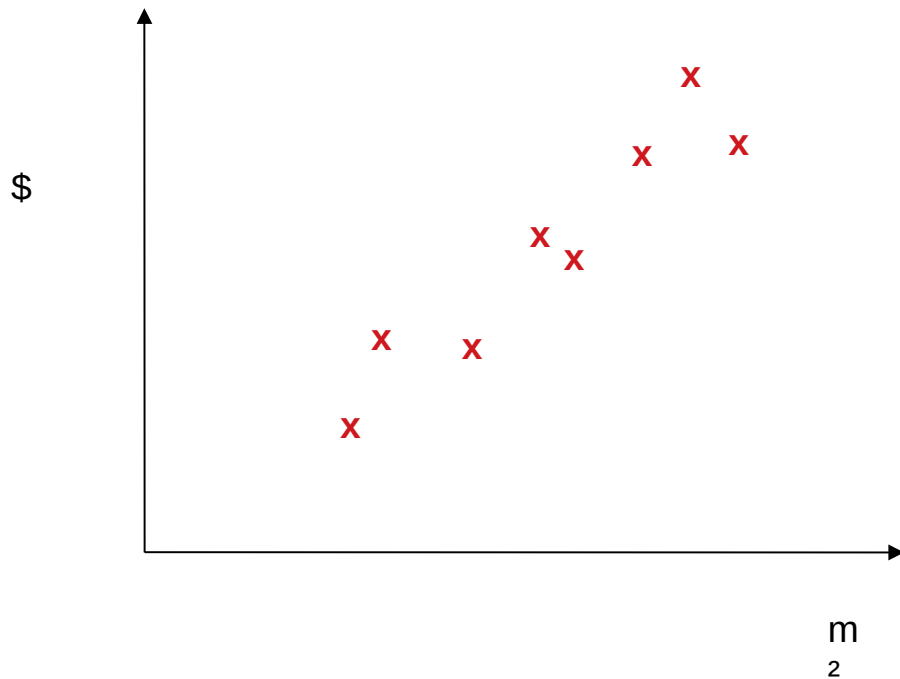
- Através dos dados de treinamento é extraído um modelo que é utilizado para efetuar novas predições (Ex.: Linear Regression)

# **Desmitificando Inteligência Artificial**

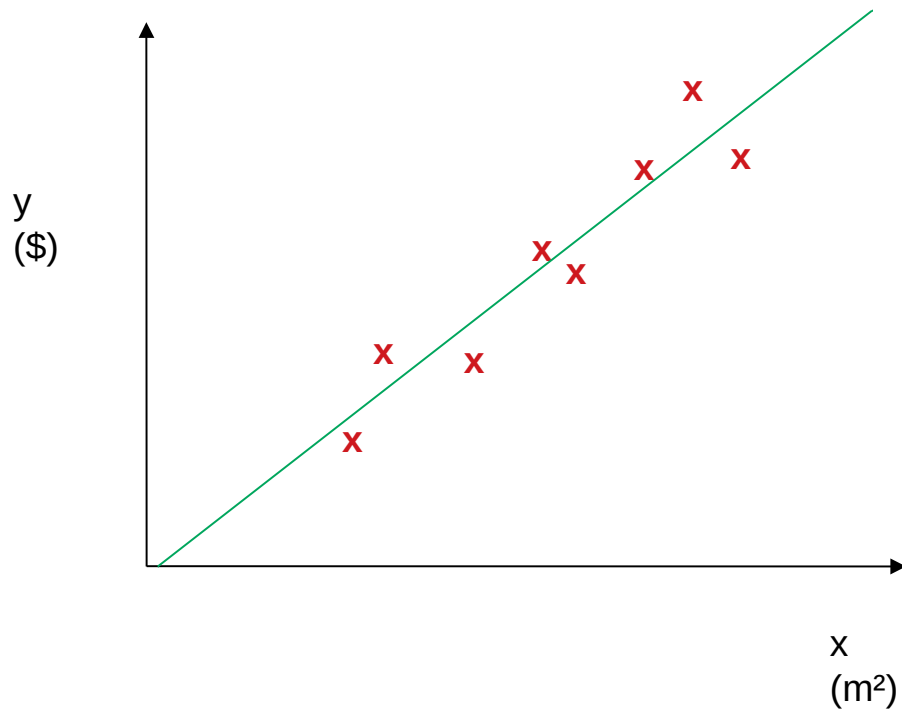
# COMO A MÁQUINA APRENDE?

Exemplo: Algoritmo que prevê o valor de uma casa, com base na metragem

# COMO A MÁQUINA APRENDE?

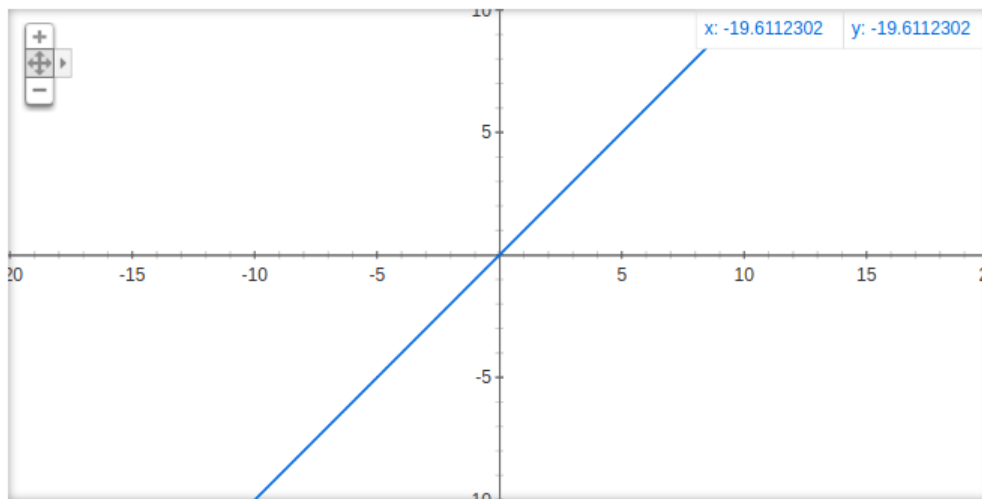


# COMO A MÁQUINA APRENDE?



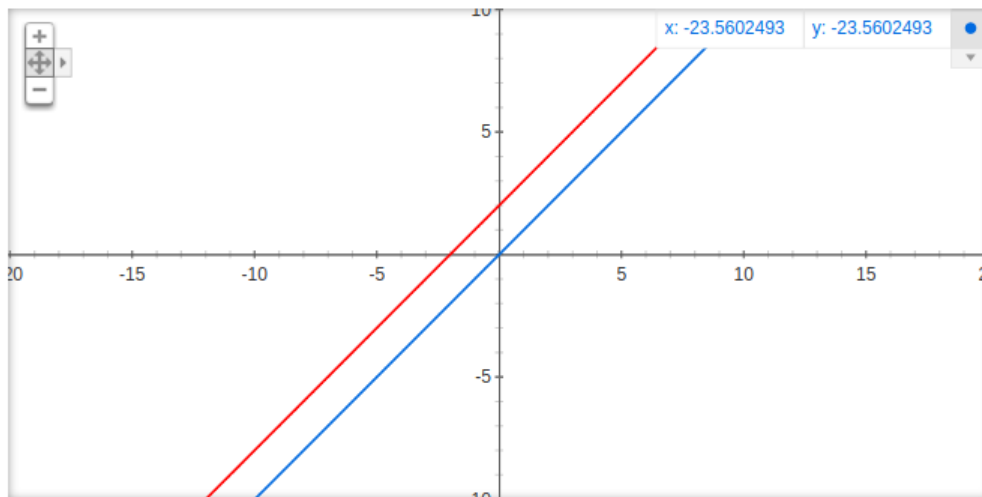
Abordagem simples:  
Descobrir qual função  
 $f(x)$   
permite chegar  
ao valor de  $y$ ? (preço)

# ENTENDENDO A FUNÇÃO DE UMA RETA



$$y = x$$

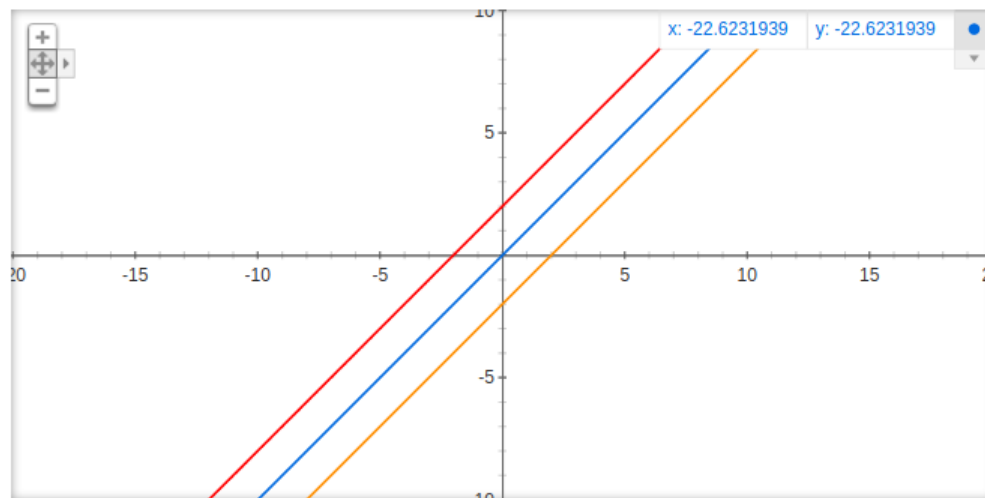
# ENTENDENDO A FUNÇÃO DE UMA RETA



$$y = x, y = x + 2$$

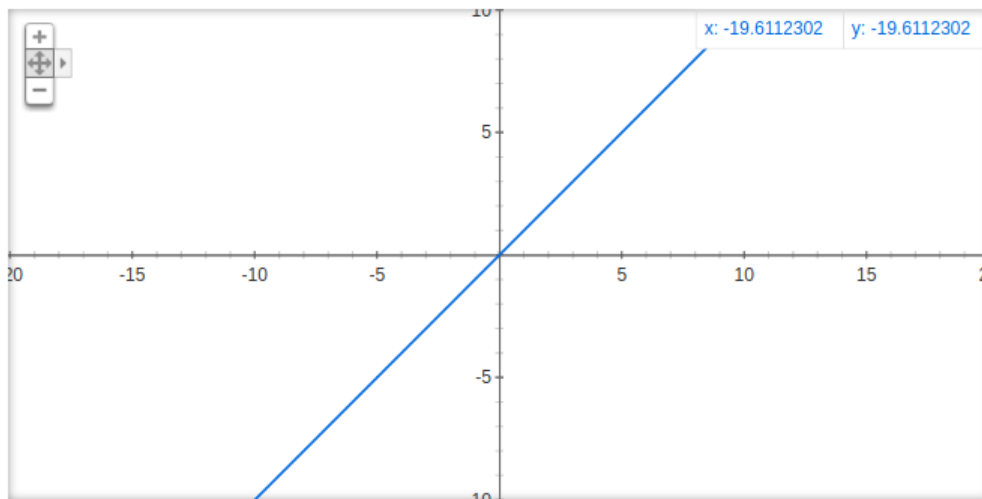


# ENTENDENDO A FUNÇÃO DE UMA RETA



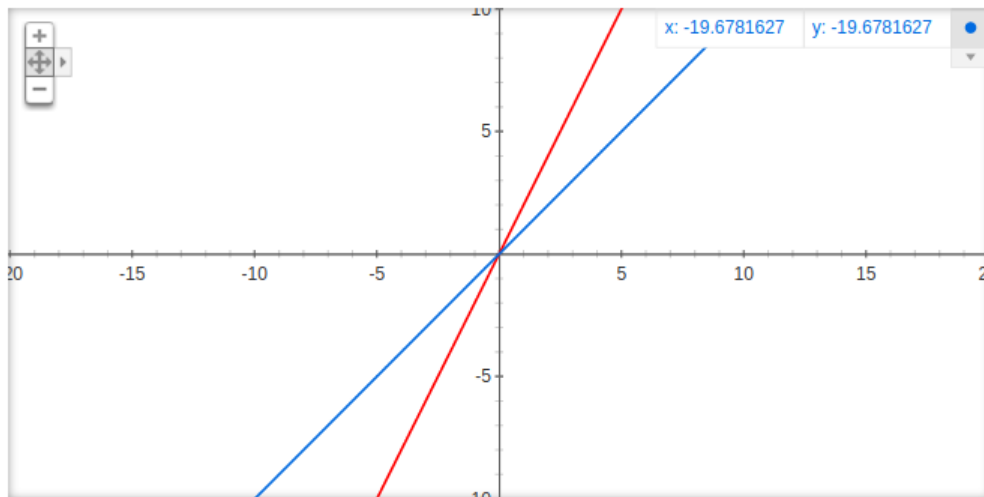
$$y = x, y = x + 2, y = x - 2$$

# ENTENDENDO A FUNÇÃO DE UMA RETA



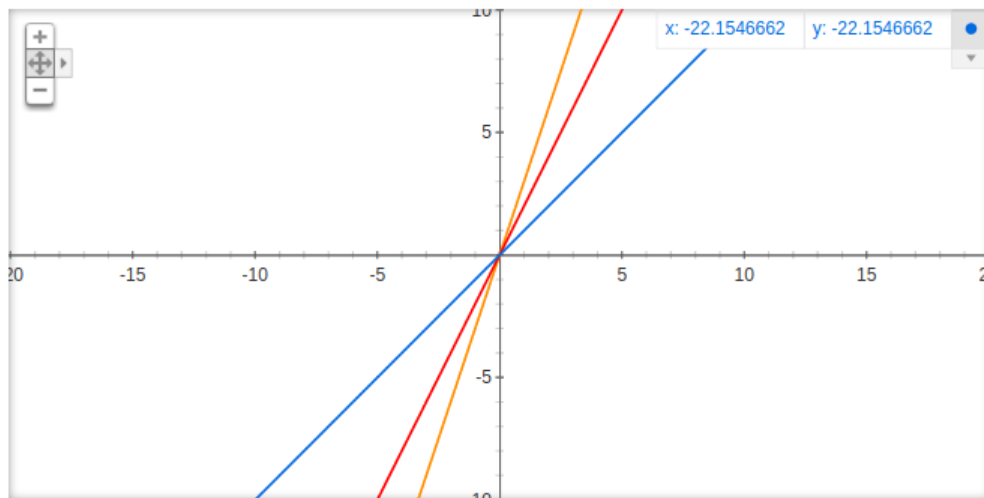
$$y = x$$

# ENTENDENDO A FUNÇÃO DE UMA RETA



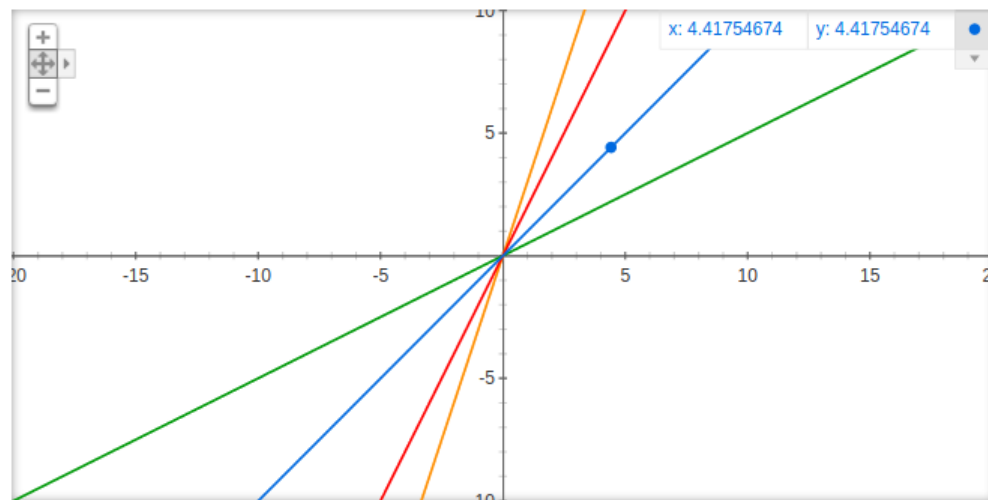
$$y = x, y = 2x$$

# ENTENDENDO A FUNÇÃO DE UMA RETA



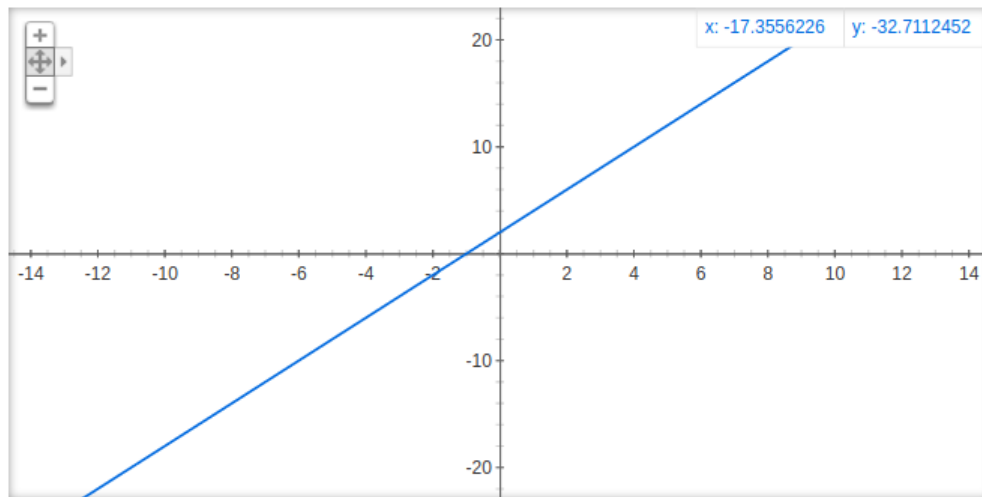
$$y = x, y = 2x, y = 3x$$

# ENTENDENDO A FUNÇÃO DE UMA RETA



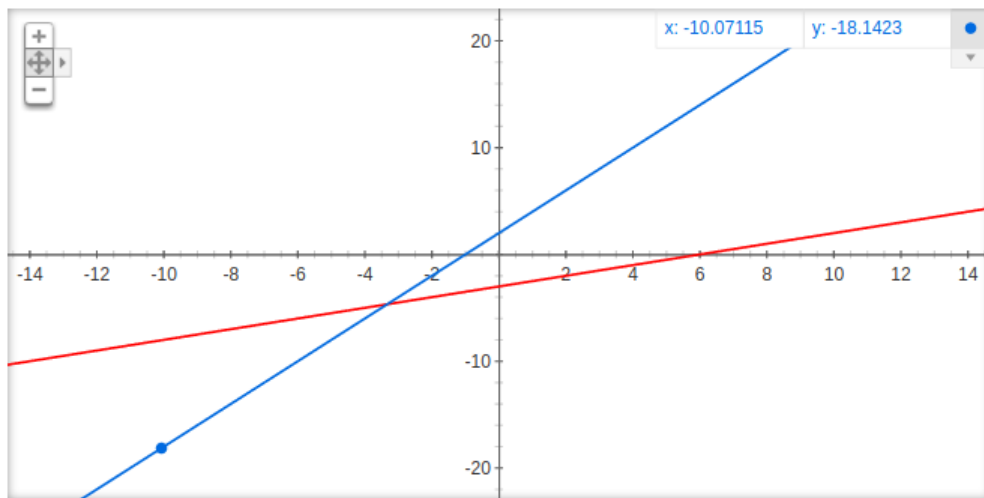
$$y = x, y = 2x, y = 3x, y = 0.5x$$

# ENTENDENDO A FUNÇÃO DE UMA RETA



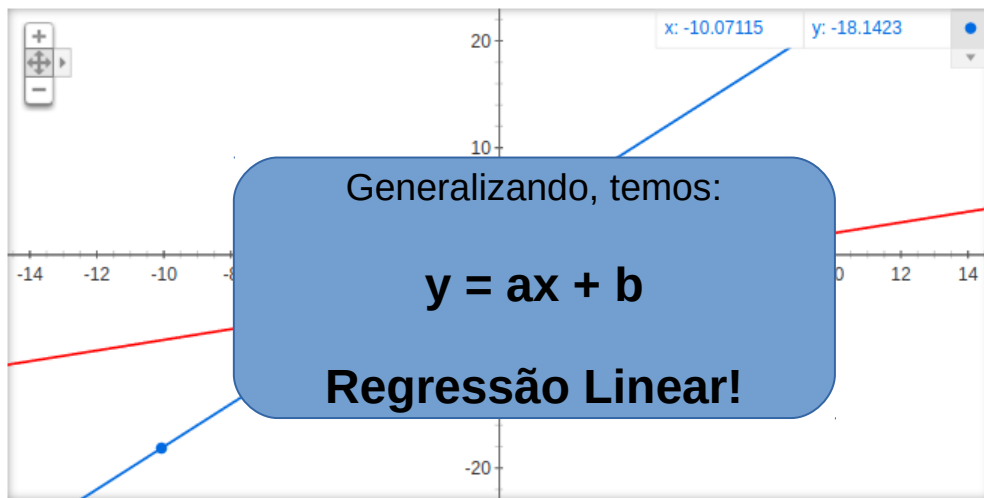
$$y = 2x + 2$$

# ENTENDENDO A FUNÇÃO DE UMA RETA



$$y = 2x + 2, y = 0.5x - 3$$

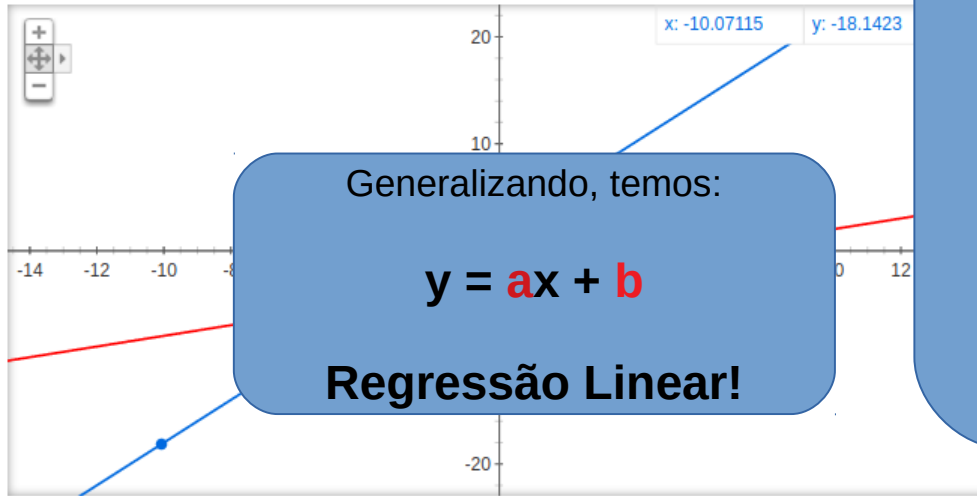
# ENTENDENDO A FUNÇÃO DE UMA RETA



$$y = 2x + 2, y = 0.5x - 3$$



# O QUE O ALGORITMO APRENDE?



O que o algoritmo aprende?

**a e b**

tal que o erro seja mínimo!

$$y = 2x + 2, y = 0.5x - 3$$

# EXERCÍCIO 1

Escreva um algoritmo que aprende os melhores valores de  $a$  e  $b$ , tentando combinações diferentes. Os melhores valores são aqueles que permitem obter o melhor resultado final.

# COMO MEDIMOS O ERRO?

Mean Squared Error: Média da soma das diferenças quadráticas

●  $MSE = \frac{1}{2} \sum_{i=0}^n (y_i - (ax + b))^2$

Valor real

Valor previsto pelo algoritmo

# COMO ESCOLHER OS VALORES DE A E B?

Vamos escolher o a e b onde MSE for o menor possível!

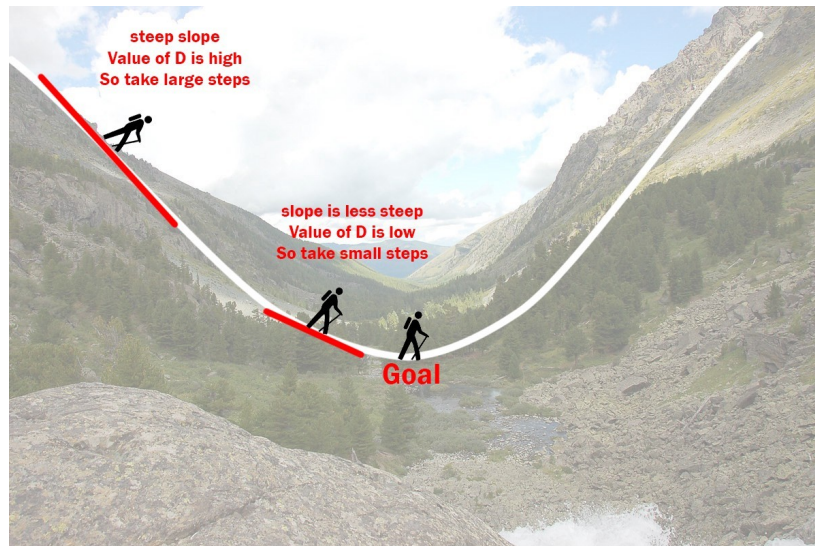
●  $MSE = \frac{1}{2} \sum_{i=0}^n (\mathbf{y_i} - (\mathbf{ax + b}))^2$

Valor real

Valor previsto pelo algoritmo

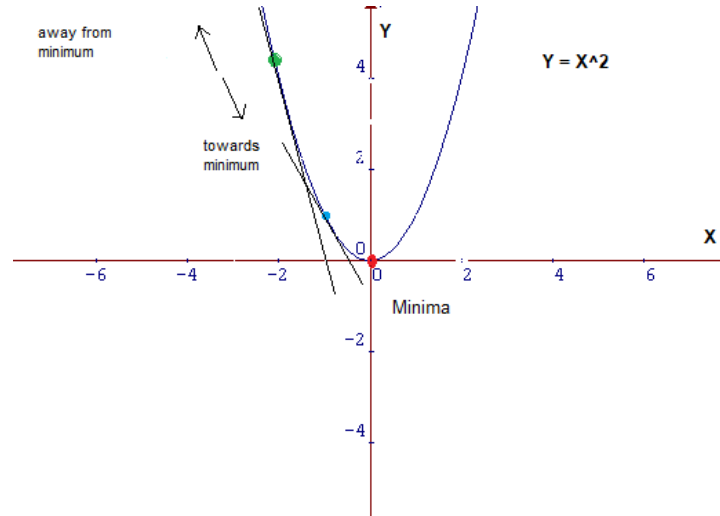
# DESCIDA DO GRADIENTE

Nosso alpinista não sabe onde está o fundo do vale! Precisa andar até encontrar...



# DESCIDA DO GRADIENTE

O “vale” na verdade é a função MSE (ou os erros obtidos)



# DESCIDA DO GRADIENTE

O “vale” na verdade é a função MSE (ou os erros obtidos)

O aprendizado funciona assim:

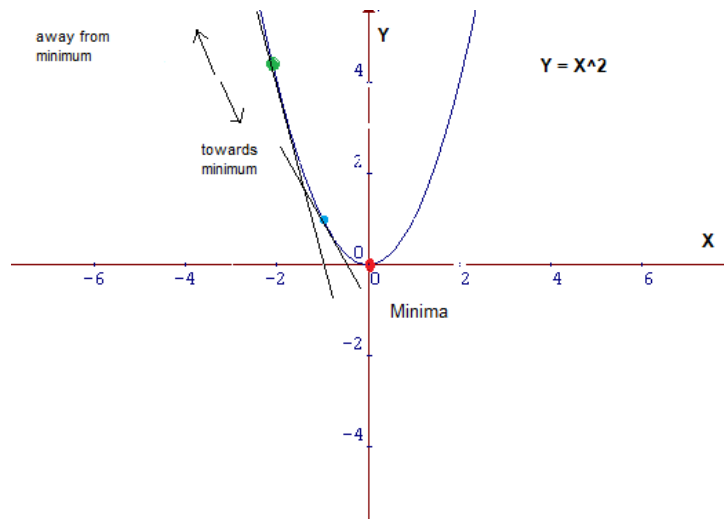
1 – Definimos valores iniciais para  $a$  e  $b$

2 – Atualizamos  $a$  e  $b$  a cada iteração, para tentar chegar no fundo do “vale”

**Mas como podemos atualizar  $a$  e  $b$ ?**

# DESCIDA DO GRADIENTE

Ao seguirmos a reta tangente à curva em um determinado ponto (aka. Ponto = resultado obtido no MSE), andamos em direção ao mínimo





# DESCIDA DO GRADIENTE

Ao seguirmos a reta tangente à curva em um determinado ponto (aka. Ponto = resultado obtido no MSE), andamos em direção ao mínimo

**Reta tangente lembra alguma coisa?**

Usamos a derivada da função MSE para obter a angulação da reta tangente!

# DESCIDA DO GRADIENTE

Obtemos as derivadas da função MSE, usando a e b como variáveis (o restante como constantes), e substituímos nas fórmulas abaixo (L = taxa de aprendizado):

$$a = a - L * \text{MSE}'_a$$

$$b = b - L * \text{MSE}'_b$$

# DESCIDA DO GRADIENTE

Obtemos as derivadas da função MSE, usando a e b como variáveis (o restante como constantes), e substituímos nas fórmulas abaixo (L = taxa de aprendizado):

$$a = a - L * ((-2/n) * \text{sum}(X * (Y - Y_{\text{pred}})))$$

$$b = b - L * ((-2/n) * \text{sum}(Y - Y_{\text{pred}}))$$

Don't worry about  
it if you don't  
understand

- Andrew Ng



# Exerício 2

Agora, vamos usar a descida do gradiente para atualizar os valores de  $a$  e  $b$  durante o aprendizado.

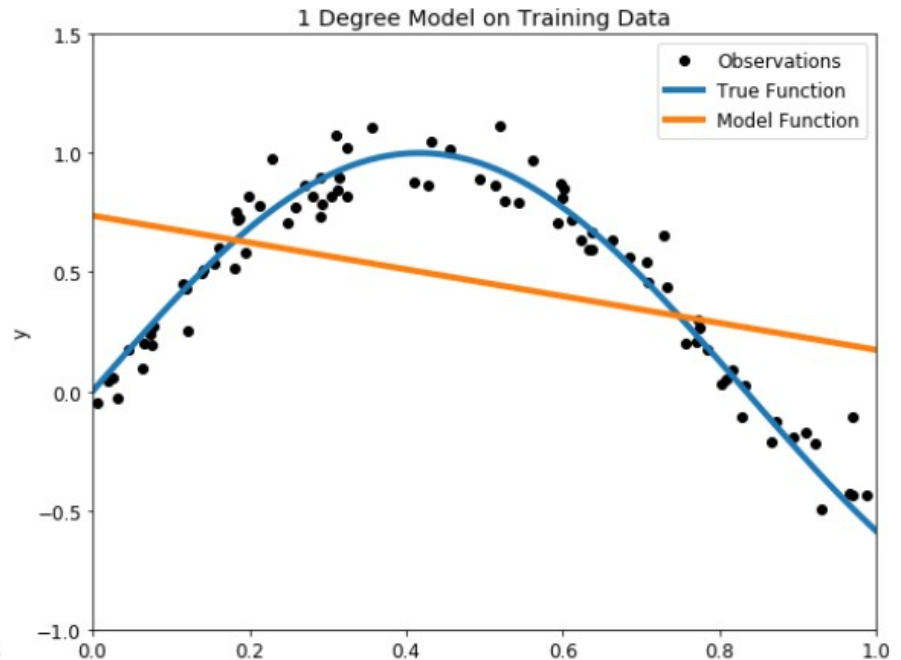
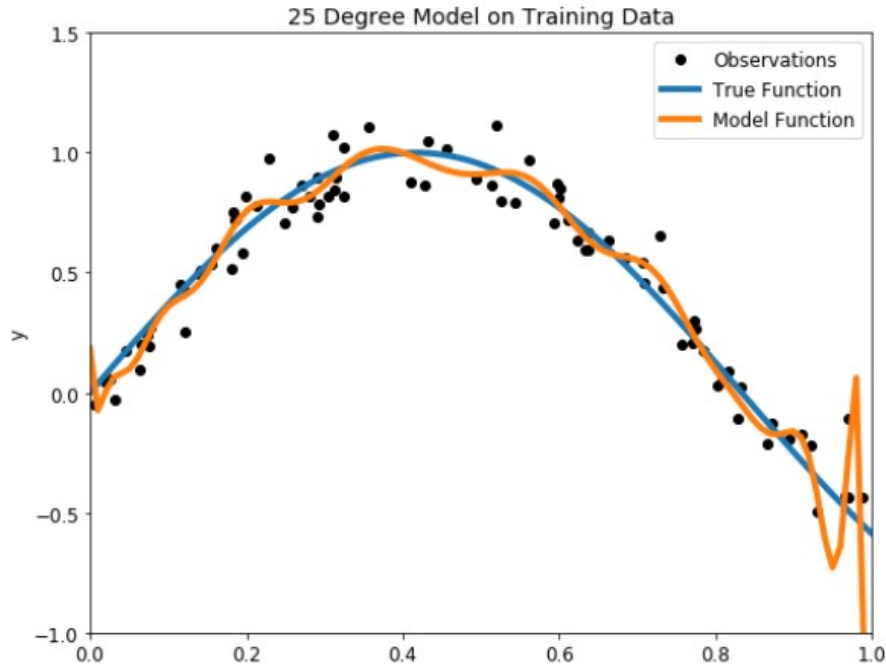
# Overfitting e Underfitting

Dois conceitos muito importantes quando se discute aprendizado em inteligência artificial são **overfitting** e **underfitting**, respectivamente, **sobreajustamento** e **subajustamento**

**Sobreajustar** um modelo implica em viciar o modelo nos dados de treino e torná-lo incapaz de generalizar para os dados de teste.

**Subajustar** significa não ter desempenho aceitável nem mesmo no conjunto de treino.

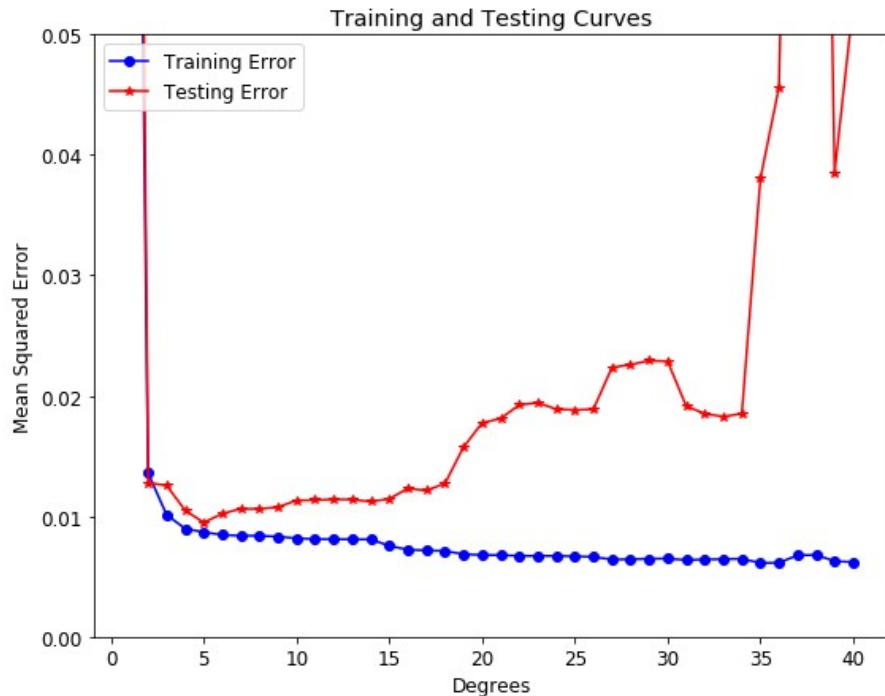
# Overfitting e Underfitting



Fonte: [Towards Data Science - Overfitting vs. Underfitting: A Complete Example](#)

# Overfitting e Underfitting

Esses fenômenos são mais facilmente observáveis em uma **curva de aprendizado**, analisando o **erro** de um modelo a medida que mais amostras de treino são dadas e o resultado é confrontado com o conjunto de teste fixo durante todo o processo.



Fonte: [Towards Data Science - Overfitting vs. Underfitting: A Complete Example](#)

# Overfitting e Underfitting - O que fazer?

Overfitting

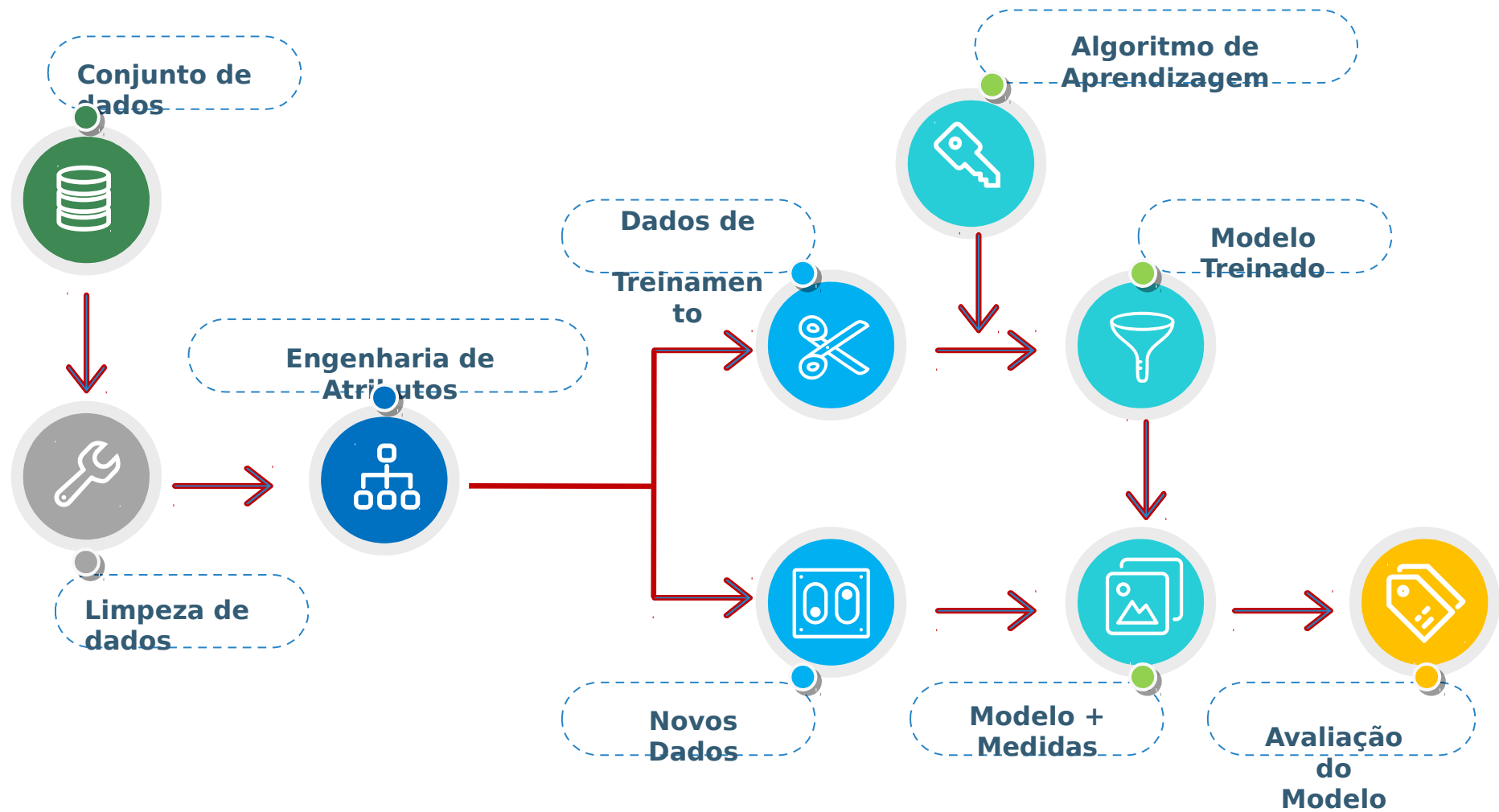
- Simplificar o modelo
- Coletar mais dados

Underfitting

- Selecionar um modelo mais complexo
- Melhorar os dados utilizados

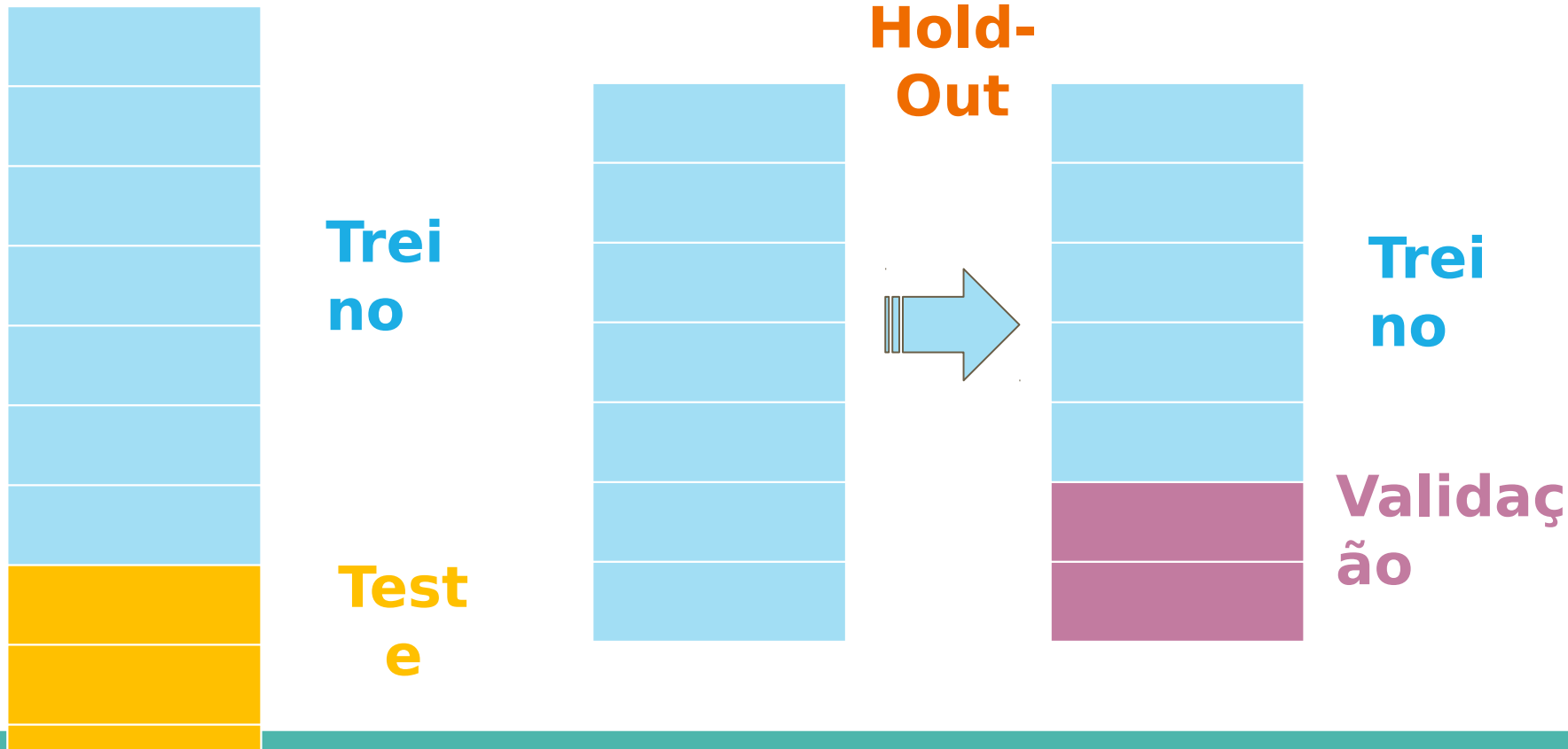


# Fluxo de aplicações de AM



# Estratégias para separação de dados e validação de modelos

# Treino e Teste e Validação



# Cross Validation ( $k = 3$ )



# Estratégias de validação

As estratégias de validação são extraídas do campo da estatística e estão associadas a tornar o protocolo experimental robusto, minimizando que os resultados não tenham sido encontrados por acaso.

Enquanto o Hold-Out é executado uma única vez, mas corre o risco de amostras simples caírem no conjunto de teste, **o Cross Validation é executado várias vezes, e o resultado final é a média com desvio padrão, se assim fizer sentido. Em alguns cenários, Cross-Validation é proibitivo devido o tempo que demandaria.**

# Leave-One-Out

A estratégia Leave-One-Out é vista como o protocolo experimental mais robusto, e mais caro também computacionalmente. Implica em treinar com  $n - 1$  amostras e testar a amostra isolada, repetindo o processo  $n$  vezes até que todas as amostras tenham sido expostas ao teste.