

data

# Introdução à Ciência de Dados

João Pedro (Dora) Mattos • 11/03/2020  
*@joaopedromattos (Telegram)*

# Motivação

- Machine Learning, de onde vem a necessidade?
  - Pense em todos os problemas de computação que você já resolveu. Todos eles tinham um nível de abstração muito baixo: ordenar um vetor, fazer operações matemáticas, exibir coisas na tela, etc... Imagine, por minuto, se você dispusesse apenas dessas ferramentas para construir um algoritmo que fosse capaz de, por exemplo, precificar uma casa. Tal tarefa aparenta ser absurdamente complexa, uma vez que traduzir a relação de contextos reais, como “número de cômodos”, com o preço final da casa é simplesmente impensável para um computador. Nesse sentido, utilizamos **métodos estatísticos/matemáticos para prever/estimar determinada característica**. A estes métodos, damos o nome de **Machine Learning**.



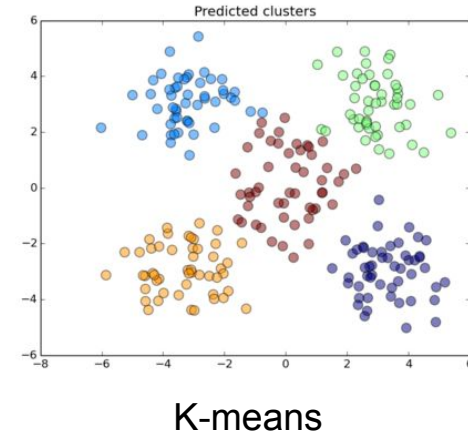
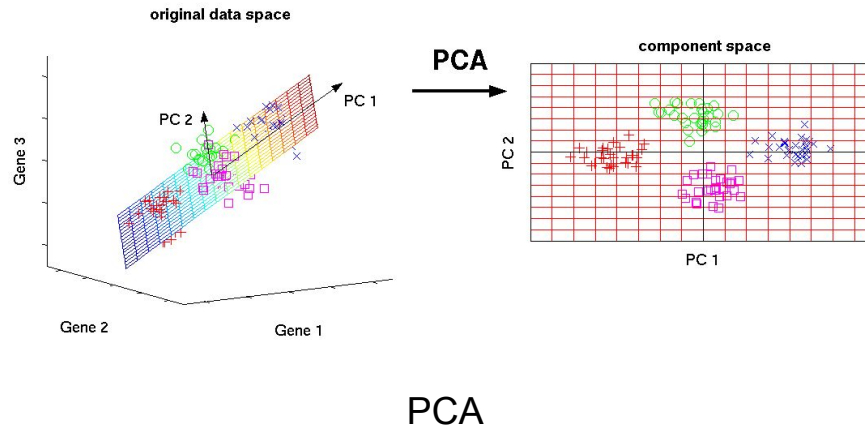
# Vários problemas, vários modelos...

- Você mencionou o preço de uma casa... mas existem outros problemas?
  - SIM, MUITOS. Em geral, existem 3 tipos de modelos de aprendizado de máquina (nome traduzido **Machine Learning**):
- Aprendizado não-supervisionado (Unsupervised Learning)
- Aprendizado por reforço (Reinforcement Learning)
- Aprendizado supervisionado (Supervised Learning)
- Neste curso, **temos enfoque no último tipo**, então não vamos abordar os dois primeiros em detalhes.



# Aprendizado não-supervisionado

- O aprendizado se dá através da própria estrutura dos dados.
- Compressão (PCA) / Agrupamento de dados (K-means)



# Aprendizado por reforço

- Diverge um pouco do convencional. Seu modelo aprende através de estímulos de um ambiente (pode ser virtual ou não).
- Muito em voga atualmente.
- Exemplo: [Redes Neurais jogando Dota 2](#), AlphaGo, etc...



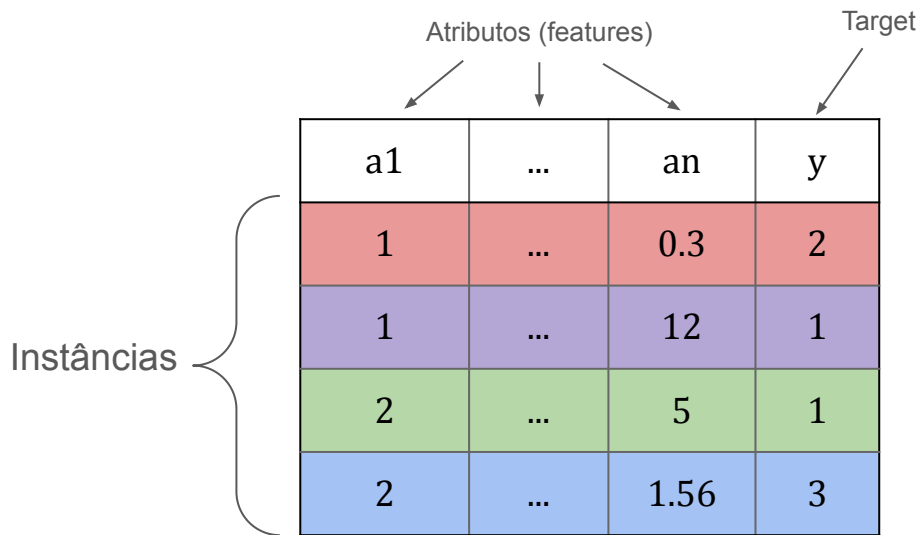
# Aprendizado supervisionado

- Conceito inicial: **Seu modelo aprende através de exemplos.**
  - Temos um conjunto de dados (chamamos de *dataset*). Esse conjunto de dados é como se fosse uma prova, ele tem instâncias (que seriam as questões) e tentamos acertar as respostas corretas (que chamaremos de *target*).

Muito complicado? Veja mais detalhes no próximo slide!



# Exemplo



The diagram shows a table representing a dataset. Above the table, the text 'Atributos (features)' has three arrows pointing to the first three columns (a1, ..., an). The text 'Target' has an arrow pointing to the fourth column (y). To the left of the table, the text 'Instâncias' is next to a large curly bracket that spans the four data rows. The table itself has four columns and five rows. The first row contains the headers 'a1', '...', 'an', and 'y'. The subsequent four rows contain numerical data, each with a unique background color: red, purple, green, and blue.

	a1	...	an	y
Instâncias	1	...	0.3	2
	1	...	12	1
	2	...	5	1
	2	...	1.56	3

- Podemos representar nosso modelo de **Machine Learning** como uma função  $f$ .
- A função  $f$  tem parâmetros  $x$  que tentam mapear os atributos (features) de uma instância (um exemplo do *dataset*). Representaremos o processo de predição/aproximação:
  - $f(x) = \hat{y}$ , sendo  $\hat{y}$  a predição/aproximação do modelo.
- Para que  $f(x)$  consiga aproximar bem os targets, precisamos que ela se ajuste bem a esse conjunto. O processo de se “aprender” o conjunto de dados é chamado de “**fitting**”.



# Tipos de supervisão

- Classificação
  - Seu target são “rótulos”, exemplo: (“Pacientes doentes” / “Saudáveis”), (“1”, “2”, “3”, “4”, “5”)
- Regressão
  - Target será um valor real, uma grandeza em uma determinada escala, exemplo: (“Preço de uma casa”), (“Coordenadas num plano cartesiano”)





# Pipeline

- Existe uma sequência de passos para construir modelos que é padronizada dentro da comunidade de **Machine Learning**. Chamamos esse conjunto de passos de **Pipeline**.
- Os passos são:
  - Obtenção de dados;
  - Análise exploratória (EDA);
  - Pré-processamento;
  - Criação do modelo;
  - Avaliação.



# Obtenção de dados

- Como vimos, nosso objetivo é ter um modelo ( $f(x)$ ) que precisa se ajustar ao *dataset*. Logo, coletar o *dataset* é o primeiro passo dentro do nosso Pipeline.
- Nosso *dataset* pode ter dados **Estruturados** ou **Não-estruturados**:
  - **Estruturados**: Tabelas
  - **Não-estruturados**: Áudio, Vídeo, Texto, Séries Temporais, etc...
- Podemos ter um dataset com dados mistos, exemplo: uma tabela de filmes, em que uma das colunas é um *review*.



# Obtenção de dados

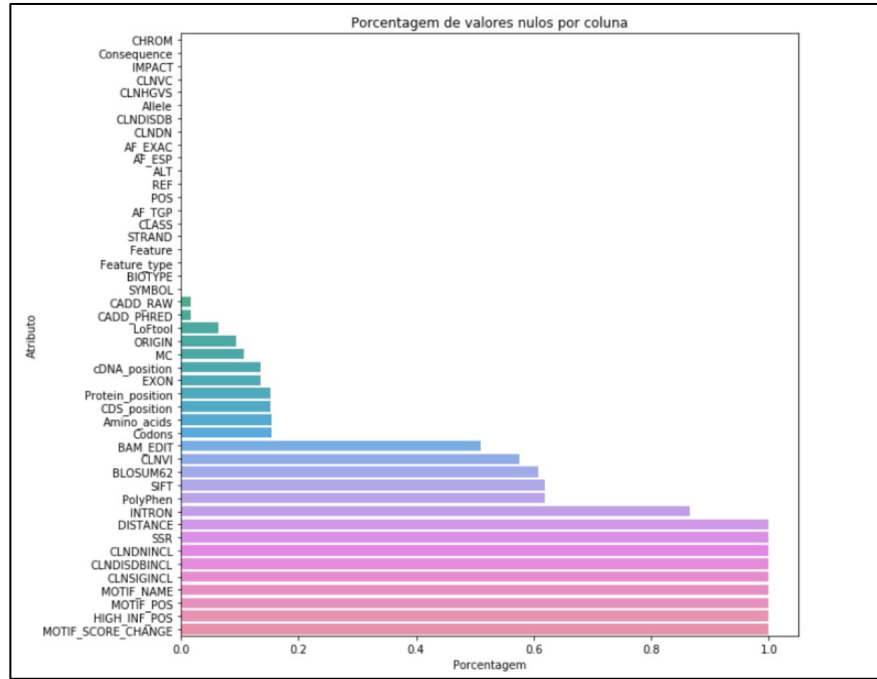
- Por agora, definiremos dois tipos de dados:
  - **Numéricos:** seu dado corresponde a um número, que pode ser inteiro, ponto flutuante, negativo, etc... Exemplo: Número de quartos de uma casa (1, 2, 3, 4).
  - **Catégoricos:** seu dado corresponde a uma categoria, que não possui valor intrínseco. Exemplo: Cor de uma casa (“Azul marinho”, “Cinza”, “Amarelo”).
- **Importante:** os modelos de **Machine Learning** são sensíveis ao tipo de dado. Logo, você deve verificar se os tipos de dado do seu *dataset* são compatíveis com os que seu modelo aceita.



# Análise Exploratória de Dados

- Realizar experimentos estatísticos com as *features* do nosso *dataset*.
  - Calcular média, mediana, testes de aderência, etc...
- Visualização de gráficos
  - *Plotting* de vários gráficos para entender melhor suas distribuições estatísticas.
- Coleta de informações para o **Pré-processamento**.





← Durante a análise exploratória que descobrimos isso :)

Exemplo em base real do Kaggle ([link](#))



# Pré-processamento

- Sua base de dados foi analisada estatisticamente no último passo, **mas ela ainda está suja.**
- O objetivo desta etapa é realizar correções e adições ao dataset que melhorem o desempenho do seu modelo, por exemplo:
  - Correção de tipos de dados para melhor compatibilidade com o modelo;
  - Tratamento de nulos.
  - Tratamento de dados inválidos
  - Criação de **novas features** através da combinação de features já existentes.



# Criação do Modelo

- Nesta etapa se dá a escolha e instanciação do modelo. Existem, em **Machine Learning** diversos modelos para vários tipos de tarefas. Nesse sentido, a sua escolha deve se basear na literatura e na sua experiência como cientista de dados.



# Avaliação

- O nosso objetivo é avaliar nosso modelo para que possamos ajustar nosso conjunto de dados da melhor forma possível, **mas de forma generalizada**.
- Separamos nossa avaliação passa por 3 sessões diferentes
  - Treino: Parte do nosso *dataset* (80%, por exemplo) em que nosso modelo vai tentar modificar seus parâmetros para se ajustar.
  - Validação: Parte restante do *dataset* (os outros 20%) na qual nosso modelo vai ver se “aprendeu” a generalizar os targets.
  - Teste: Etapa de uso do modelo, onde colocamos em produção.
  - **Atenção:** Essas nomenclaturas podem mudar de acordo com a situação ou a fonte. Muitas pessoas usam a palavra “Teste” para se referir à “Validação”.





# Avaliação e suas métricas

- Para verificarmos o aprendizado do modelo, usamos algumas métricas
  - **Acurácia (Classificação):** Obter a % de acertos do seu modelo.
  - **Mean Squared Error/ Erro Quadrado Médio (Regressão):**  $\frac{1}{N} \sum_{t=1}^N (y(t) - \hat{y}(t))^2$
  - E muitas outras...

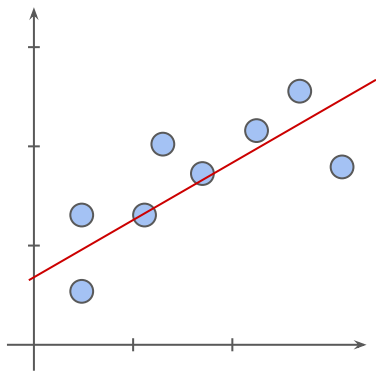


# Avaliação na prática

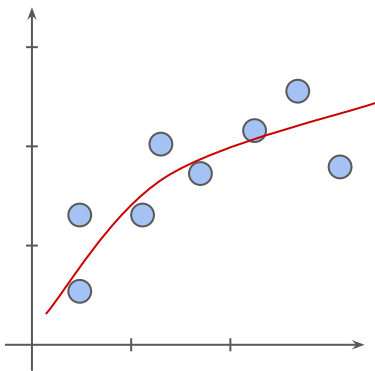
Na prática, independente da métrica que utilizaremos, nosso objetivo é a **generalização do modelo**. Assim, teremos uma métrica para o conjunto de treino, e outra para a validação.

- **Underfitting**: Seu modelo não “treinou” bem, logo terá um desempenho ruim durante o treino e a validação.
- **Fitting**: Treino e validação foram bem feitos, seu modelo generaliza bem.
- **Overfitting**: O modelo “decorou” todos os resultados do treino, mas não consegue generalizar para os exemplos da validação.

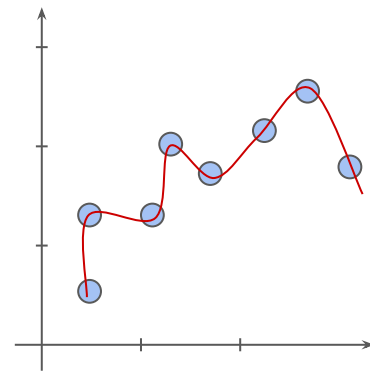




**Underfitting:** O modelo não está bem ajustado aos dados do conjunto de treino.



**Ok:** O modelo está ajustado aos dados de treino na quantidade correta e generaliza bem o suficiente na validação.



**Overfitting:** O modelo está mais ajustado aos dados de treino do que deveria. Provavelmente, não generaliza bem.