



# Descoberta do Conhecimento





# Descoberta do Conhecimento

---

Cleilton Lima Rocha

Universidade 7 de Setembro  
Fortaleza - CE, Brasil



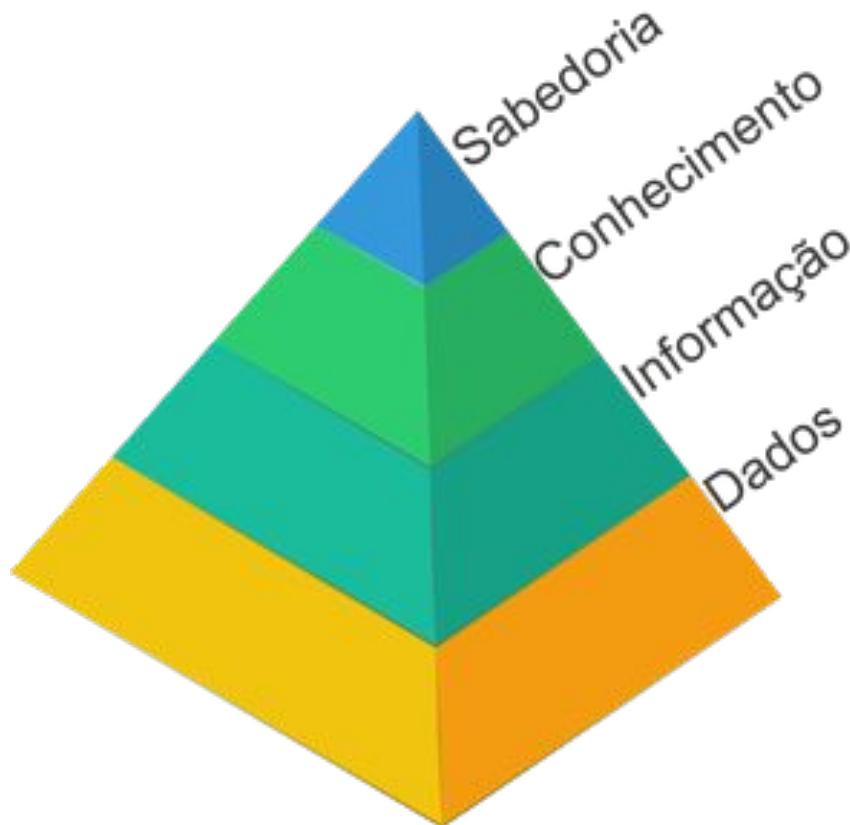


# Agenda

- ◊ Introdução ao Processo de Descoberta de Conhecimento e Data Science
  - ◊ Exploração de Dados
  - ◊ Feature engineering:
    - Pré-processamento de dados
      - Modelagem dos dados
      - Seleção de Features ...
  - ◊ Modelos de aprendizagem supervisionada e não supervisionada
  - ◊ Análise do *bias variance threshold*
  - ◊ Introdução a Aprendizagem por reforço e Deep Learning
  - ◊ Projeto prático aplicado à Data Science.
- 



# Processo de Descoberta de Conhecimento





*“O KDD pode ser visto como o processo de descoberta de padrões e tendências por análise de grandes conjuntos de dados, tendo como principal etapa o processo de mineração, consistindo na execução prática de análise e de algoritmos específicos que, sob limitações de eficiência computacionais aceitáveis, produz uma relação particular de padrões a partir de dados FAYYAD et al (1996).”*



*“Informação é o resultado do processamento de dados num formato que tem significado para o usuário respectivo e que tem valor real ou potencial nas decisões presentes ou prospectivas DAVIS (1974).”*

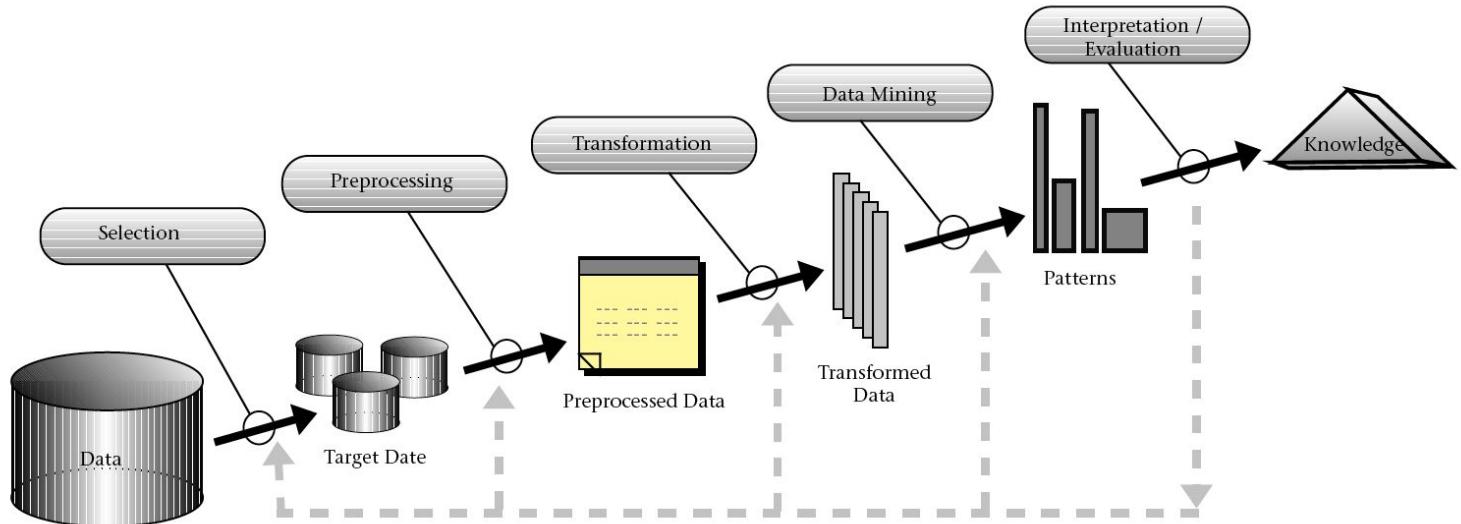


*“Segundo DAVENPORT e PRUSAK (1998), a GC pode ser vista como uma série de ações gerenciais constantes e sistemáticas que facilitam os processos de criação, registro e compartilhamento do conhecimento nas organizações.”*



*“O conhecimento necessário para se decidir e/ou avaliar torna-se disponível por meio de informações SANCHES (1997).”*

# Fases do KDD





## Data Mining e seus métodos

- ◊ Aprendizagem supervisionada
- ◊ Aprendizagem não supervisionada
- ◊ Modelos de regras de associação
- ◊ Modelos de relacionamento entre variáveis



# ATD

## Apoio à tomada de decisão





## 2017 This Is What Happens In An Internet Minute



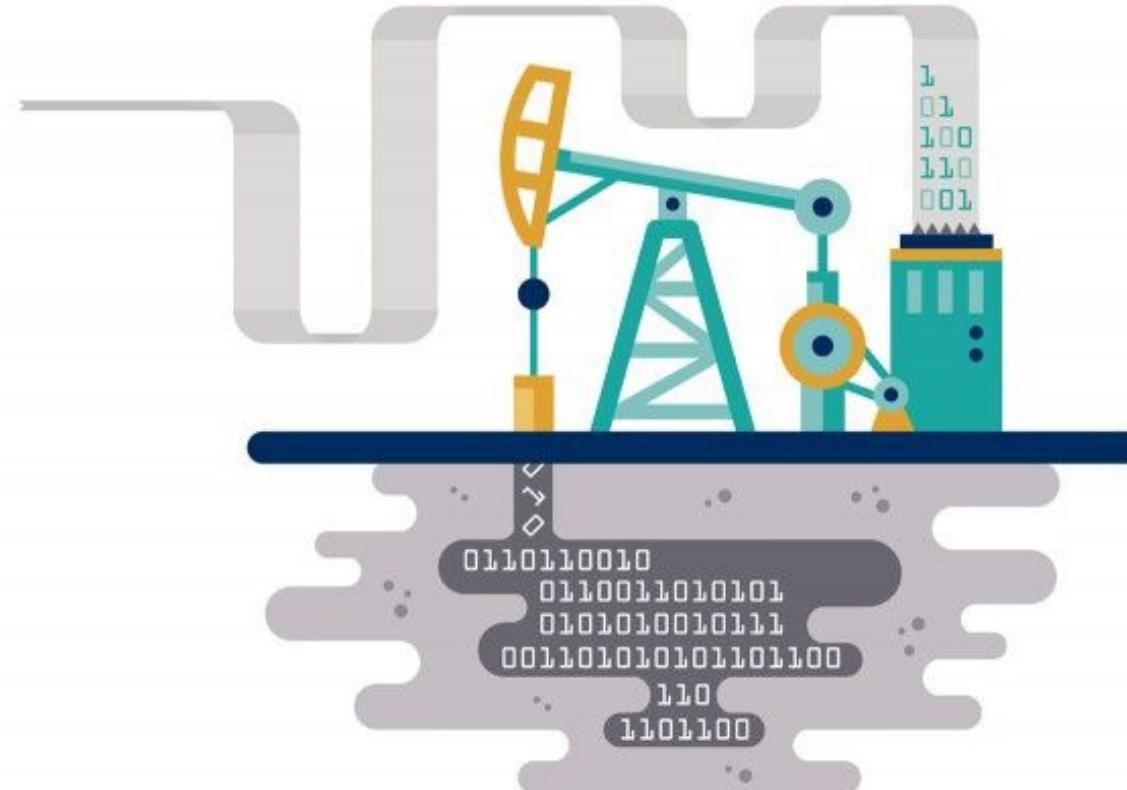
## 2018 This Is What Happens In An Internet Minute



Volume de dados gerados em um minuto, nos anos de 2017 e 2018



## Riqueza dos Dados





# Interesse em Data Science

● data science  
Termino de pesquisa

● big data  
Termino de pesquisa

● machine learning  
Termino de pesquisa

+ Adicionar comparação

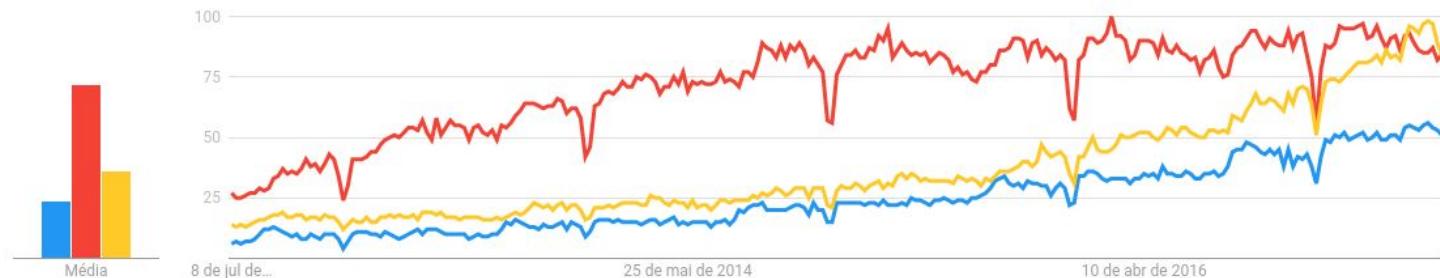
Todo o mundo ▾

Nos últimos 5 anos ▾

Todas as categorias ▾

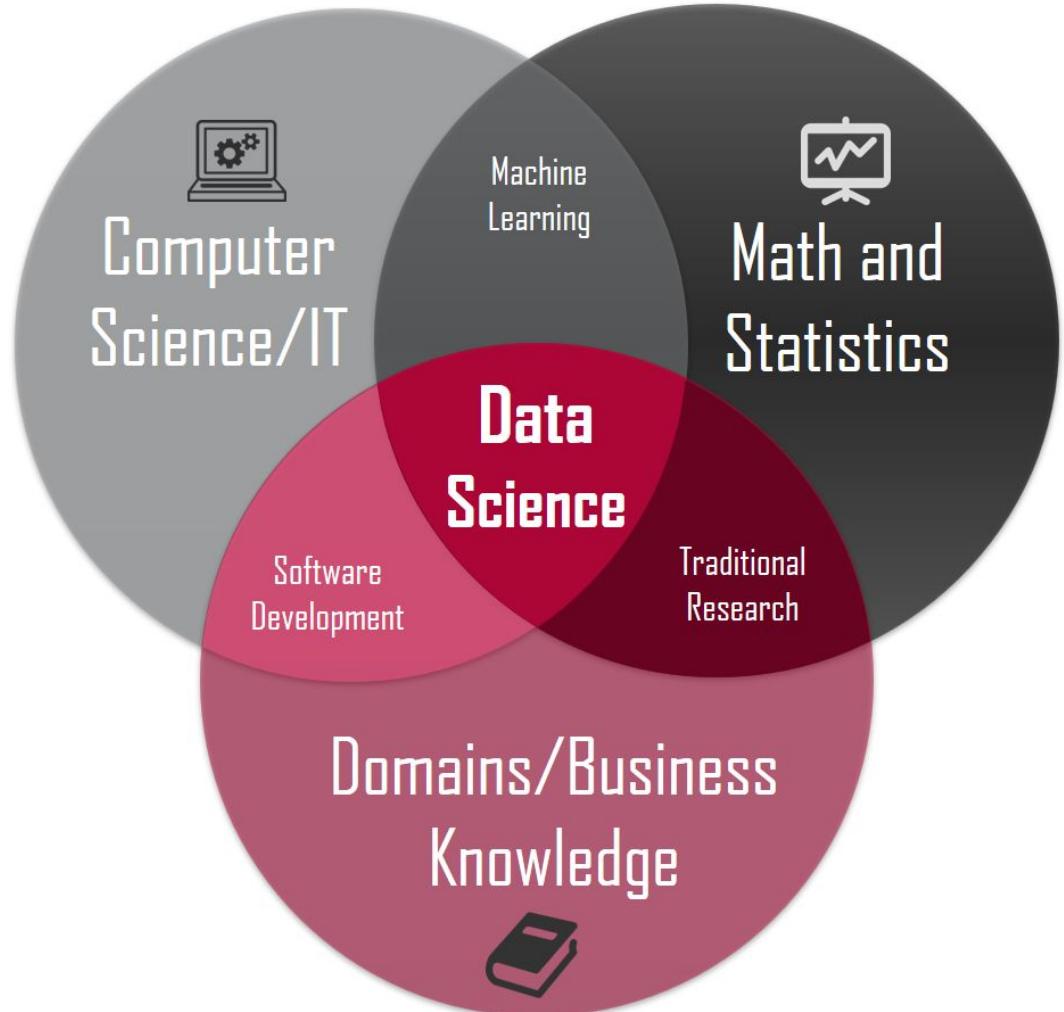
Pesquisa na Web ▾

Interesse ao longo do tempo ?



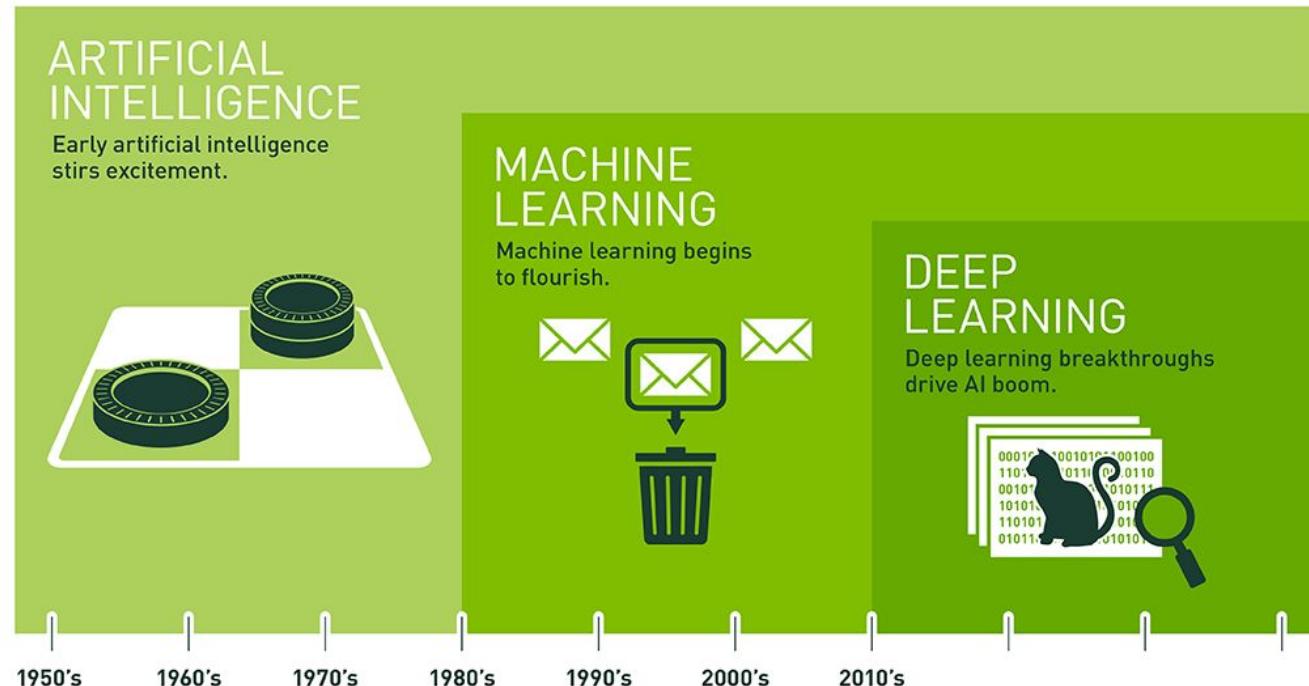


Data Science – uma ciência  
interdisciplinar





# Machine Learning Overview



Since an early flush of optimism in the 1950s, smaller subsets of artificial intelligence – first machine learning, then deep learning, a subset of machine learning – have created ever larger disruptions.



# Data Science Overview

**data** **quantitative** **statistical** **inference** **models** **statistics**

research coefficient regression learning generalized linear bayesian probability modeling maximum research coefficient regression learning generalized linear bayesian probability modeling maximum

analytics expectation likelihood trend management spatial visualization methods predictive normal parameter time function causal equation simulation workshops consulting covariate duration variance distribution graphical standard



## Exemplos



Inteligência em Saúde



Recommendation Systems



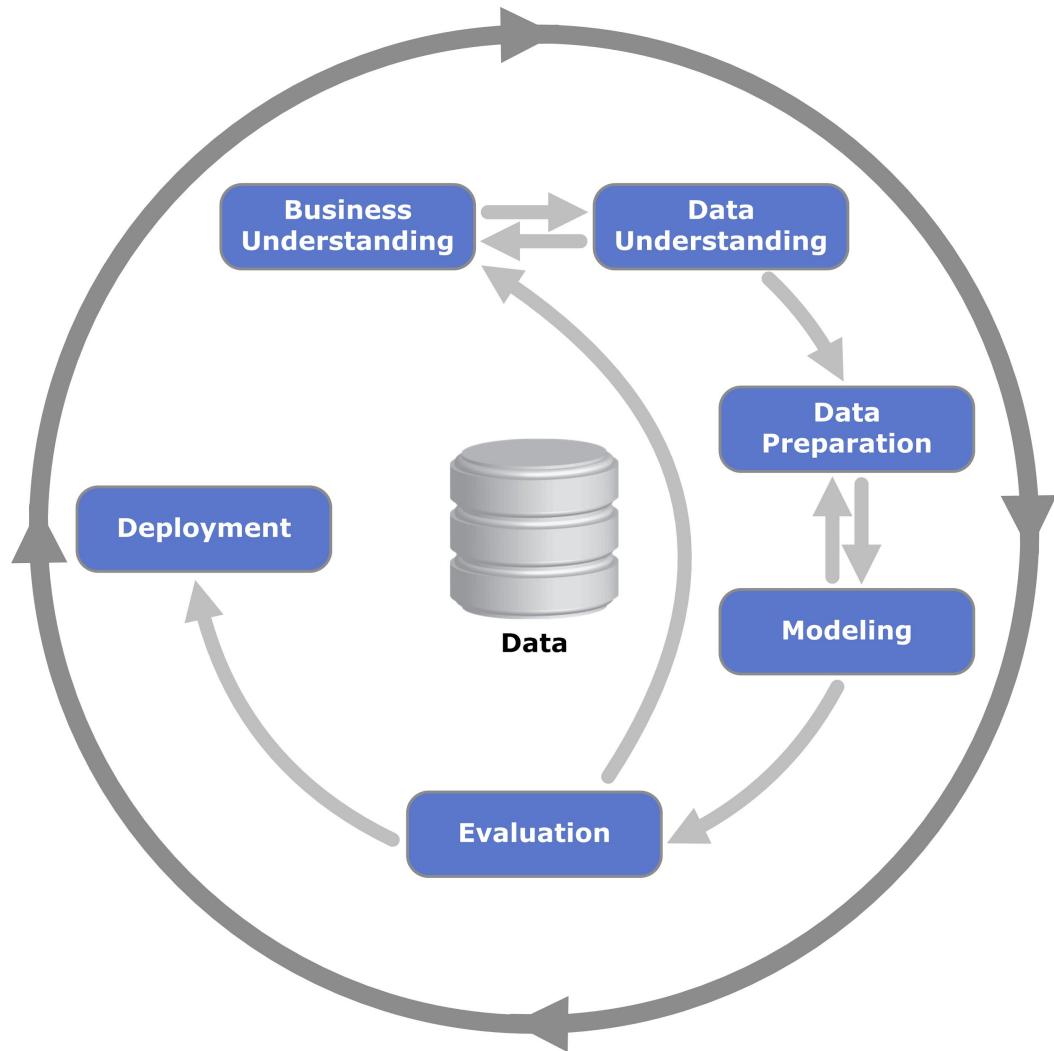
Inventory planning



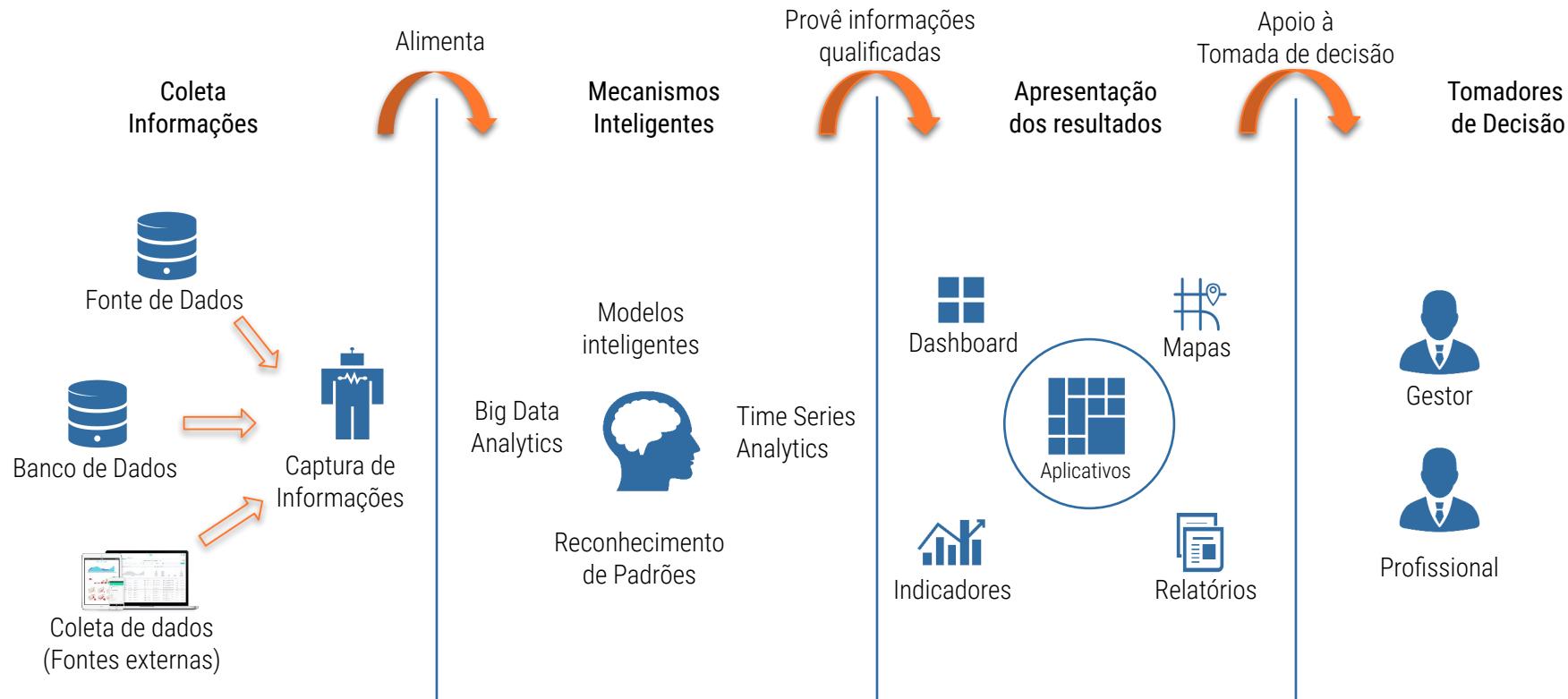
Dynamic  
pricing



## CRISP-DM

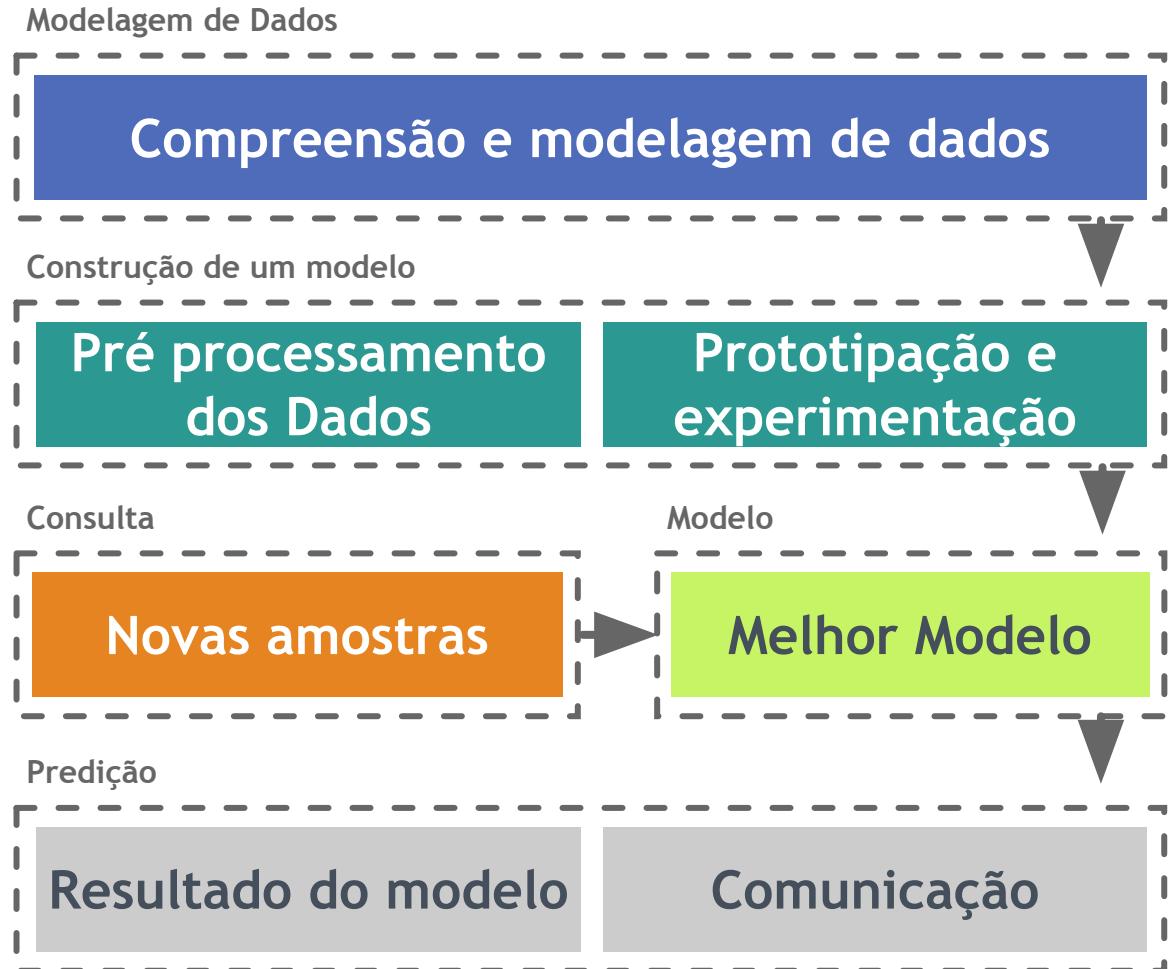


# Metodologia geral adotada





# Metodologia



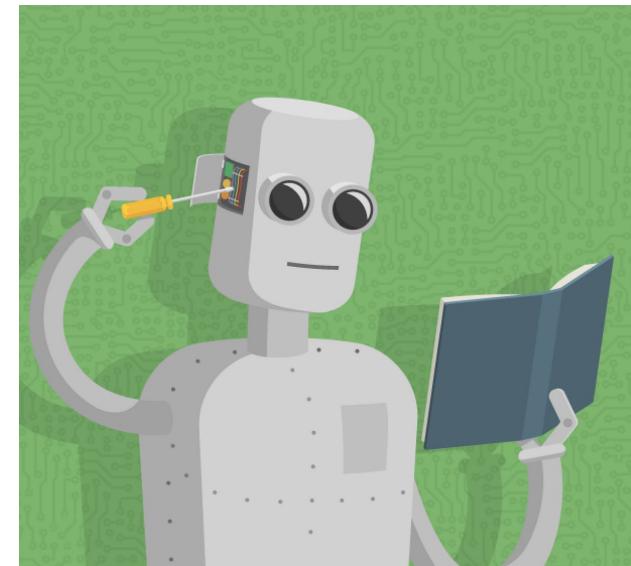
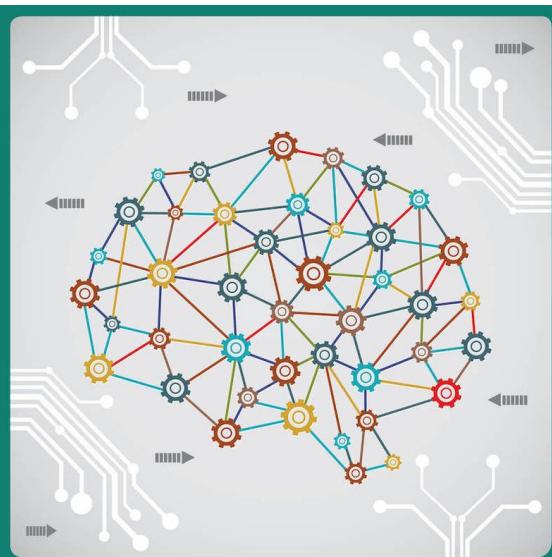


## Exemplo de uma solução end-to-end





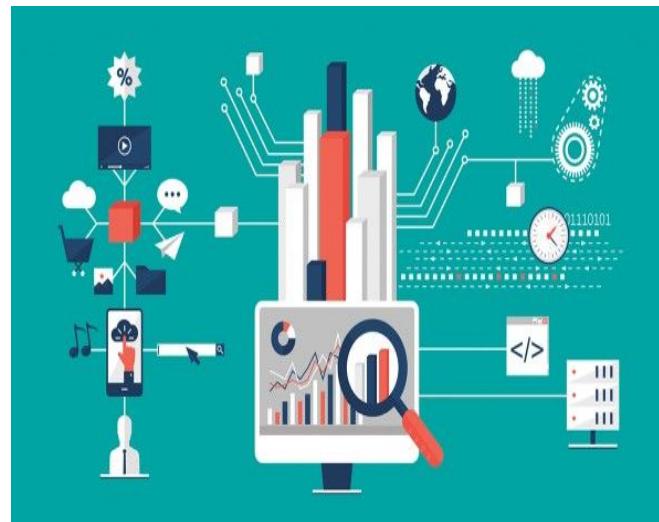
## Conhecimentos desejáveis





Big Data

## Conhecimentos desejáveis



Processamento de stream e  
séries temporais



Processamento de Linguagem  
Natural



# O que um cientista de dados faz?



## Conhecimentos desejáveis



# Cientista de Dados



## MODERN DATA SCIENTIST

Data Scientist, the sexiest job of 21th century requires a mixture of multidisciplinary skills ranging from an intersection of mathematics, statistics, computer science, communication and business. Finding a data scientist is hard. Finding people who understand who a data scientist is, is equally hard. So here is a little cheat sheet on who the modern data scientist really is.

### MATH & STATISTICS

- ★ Machine learning
- ★ Statistical modeling
- ★ Experiment design
- ★ Bayesian inference
- ★ Supervised learning: decision trees, random forests, logistic regression
- ★ Unsupervised learning: clustering, dimensionality reduction
- ★ Optimization: gradient descent and variants

### DOMAIN KNOWLEDGE & SOFT SKILLS

- ★ Passionate about the business
- ★ Curious about data
- ★ Influence without authority
- ★ Hacker mindset
- ★ Problem solver
- ★ Strategic, proactive, creative, innovative and collaborative



### PROGRAMMING & DATABASE

- ★ Computer science fundamentals
- ★ Scripting language e.g. Python
- ★ Statistical computing package e.g. R
- ★ Databases SQL and NoSQL
- ★ Relational algebra
- ★ Parallel databases and parallel query processing
- ★ MapReduce concepts
- ★ Hadoop and Hive/Pig
- ★ Custom reducers
- ★ Experience with xaaS like AWS

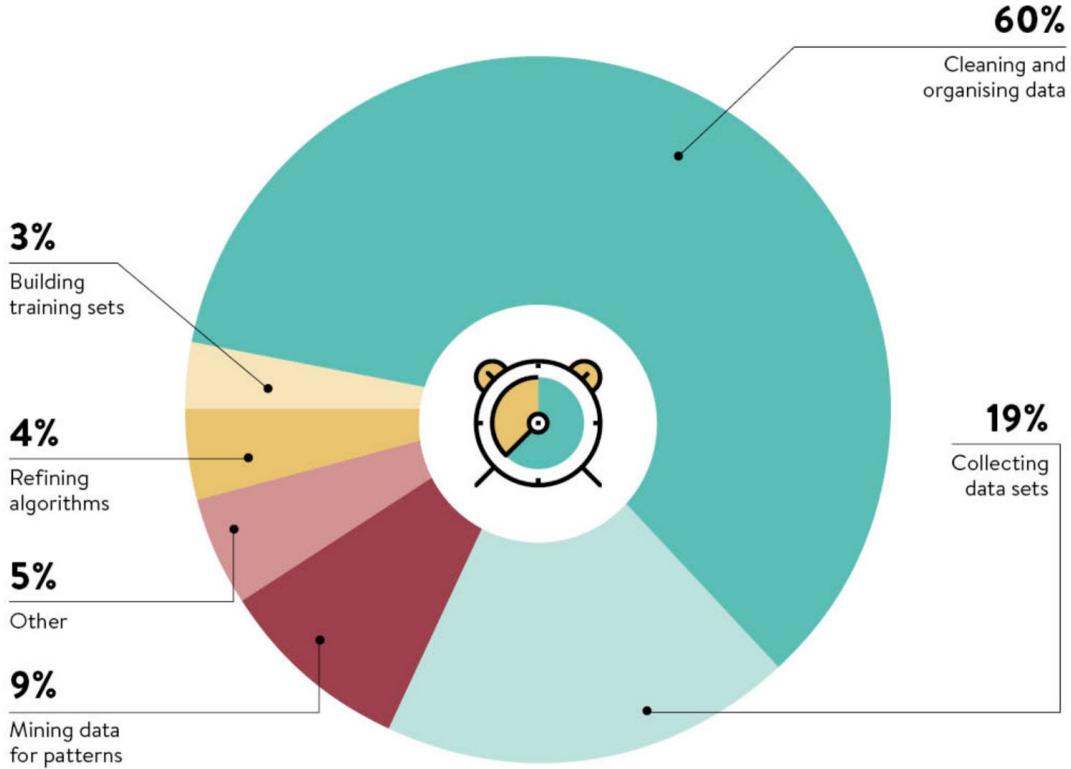
### COMMUNICATION & VISUALIZATION

- ★ Able to engage with senior management
- ★ Story telling skills
- ★ Translate data-driven insights into decisions and actions
- ★ Visual art design
- ★ R packages like ggplot or lattice
- ★ Knowledge of any of visualization tools e.g. Flare, D3.js, Tableau

# Cientista de Dados



## WHAT DATA SCIENTISTS SPEND THE MOST TIME DOING



Source: CrowdFlower 2016



# Definição de Especialistas

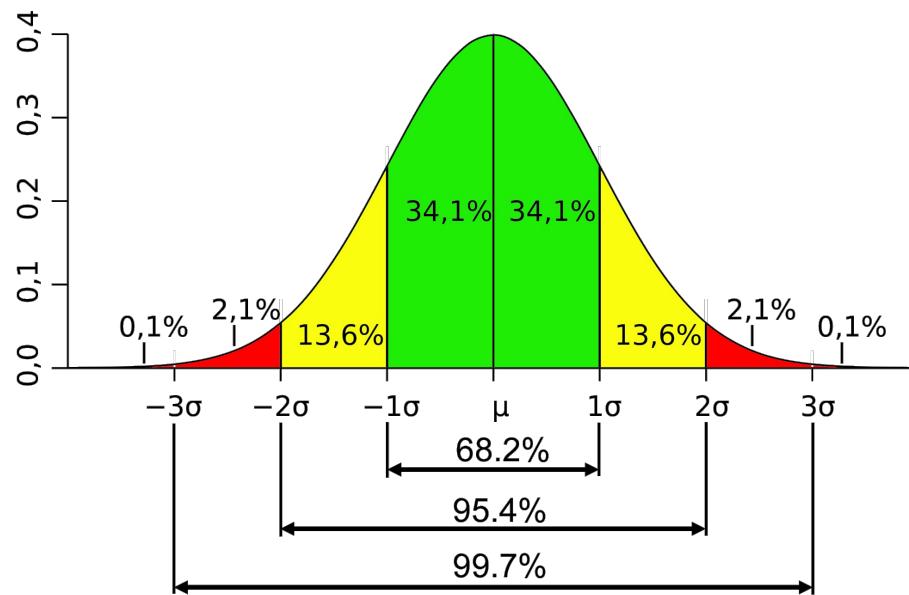
## Data Scientist vs Data Analyst vs Data Engineer



What are the differences?

# Distribuição Normal

Com a curva normal definida, temos informações importantes sobre a distribuição dos nossos dados:



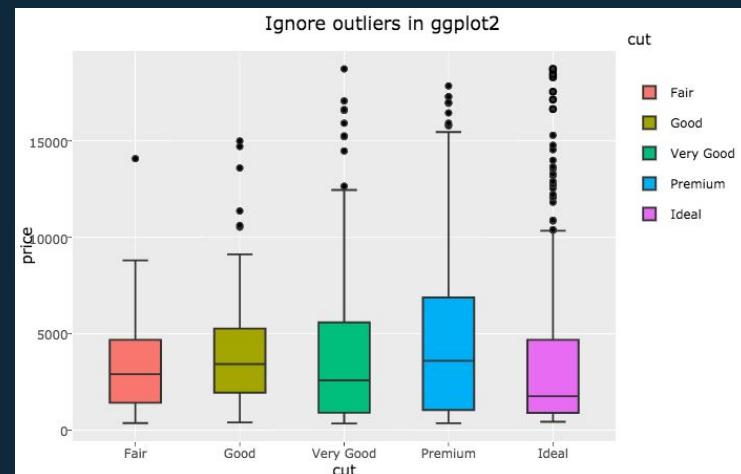
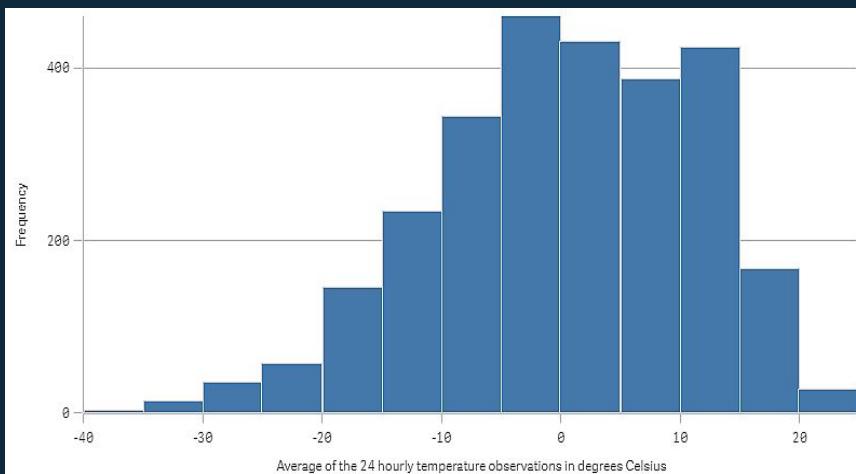
Intervalo	Proporção
$\mu \pm 1\sigma$	68,2%
$\mu \pm 2\sigma$	95,4%
$\mu \pm 3\sigma$	99,7%



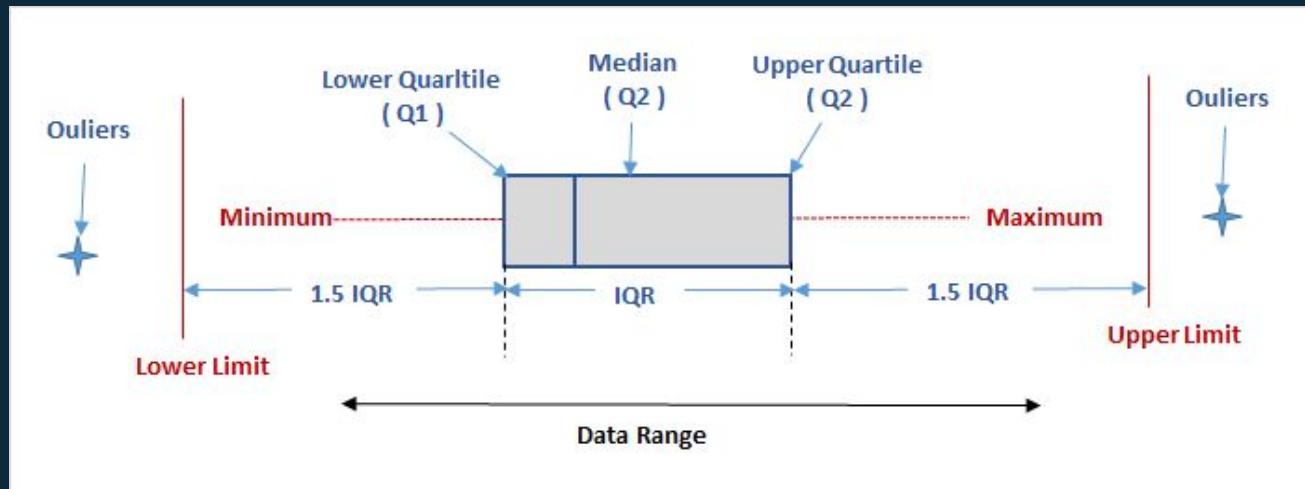
# Exploração de Dados (EDA)



# Exploração do dado



# Box Plot



# Hands-On



A decorative graphic on the left side of the slide features a large teal arrow pointing right. Inside the arrow are several white and dark blue gears of different sizes. Above the arrow, there are several overlapping hexagons in shades of teal, light blue, and dark blue.

# Feature engineering

A decorative graphic on the bottom right corner of the slide consists of several overlapping hexagons in shades of teal, light blue, and dark blue, similar in style to the one on the left.

# Escalas de Dados

- ◊ **Nominal:** nessa escala os valores **valores são não numéricos e não ordenados**. Por exemplo, cor, marca de carro, etc.
- ◊ **Ordinal:** Nessa escala os valores não são numéricos, mas são **ordenados**. Uma amostra pode apresentar um valor comparativamente maior do que uma outra. Ex: Função no trabalho

# Escalas de Dados

- ◆ **Intervalar:** escala onde valores são numéricos, existindo uma ordem entre os valores e uma diferença entre esses valores. O zero é relativo.
- ◆ **Proporcional:** nessa escala de valores numéricos, além da diferença, tem sentido calcular a proporção entre valores.

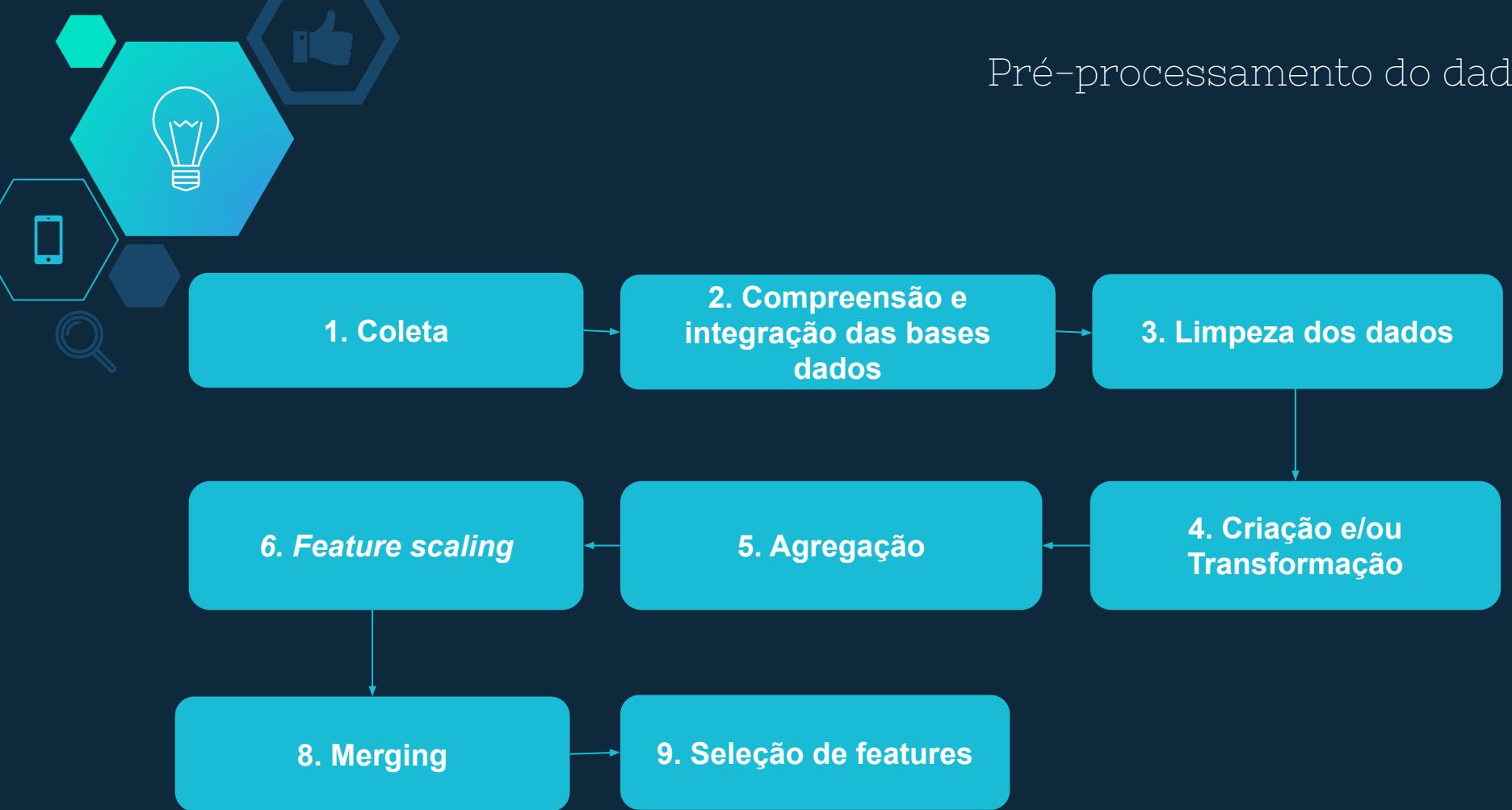
# Os atributos podem ser:

- ◊ **Qualitativo**:  
escalas  
nominais ou  
ordinais
  - Variáveis Discretas
  - Binárias

- ◊ **Quantitativo**:  
escalas  
intervalar ou  
proporcional
  - Variáveis **contínuas**

Ausentes ou  
inaplicáveis

# Pré-processamento do dado

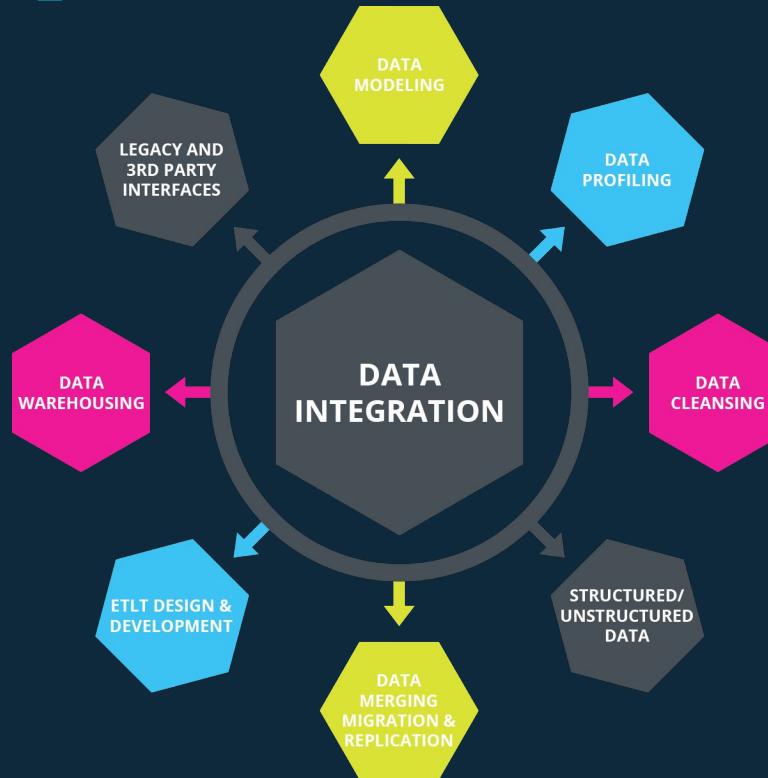




# Coletar o dado

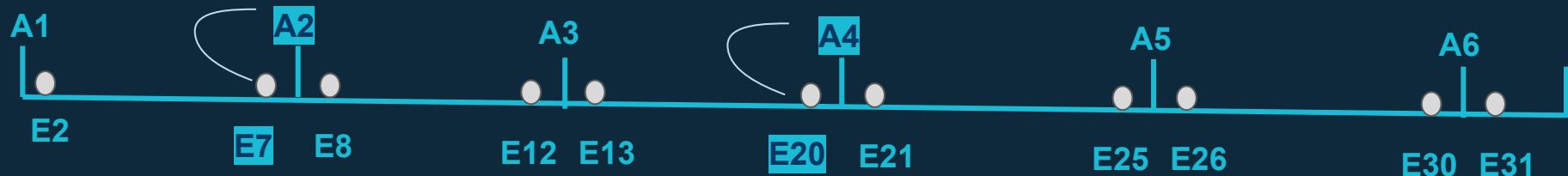
- ◊ Dados Públicos
- ◊ Dados no DBpedia
- ◊ Plataformas de ensino (e.g. Kaggle, UCI)
- ◊ Crowler
- ◊ REST API
- ◊ Acesso direto as fontes de dados
- ◊ Dado estruturado
- ◊ Dado não estruturado

# Compreender e Integrar





# Modelagem da amostra



# Limpeza dos dados

- ◊ Preencher dados ausentes
  - Como preencher valores numéricos?
  - Como preencher valores nominais?
  - Aplicar ML
- ◊ Remover dados ausentes
  - Quando eliminar uma amostra?
  - Quando eliminar uma coluna?
- ◊ Identificar outlier
  - Qual a melhor fórmula?

# Criação de features

- ◊ Aplicar Fórmula (e.g.: Faixa salarial)
- ◊ Valores proporcionais (e.g.: IMC)
- ◊ Opcional: Eliminar features originais

*“É necessário para obter os dados em uma forma apropriada para a aplicar data science com machine learning”*

# Transformação

- ◊ Label encoding
  - Sexo (F, M) → Sexo (F: 0), (M: 1)
- ◊ One Hot Encoding
  - Resulta em uma matriz esparsa

	Idade	Sexo_M	Sexo_F
Amastra _1	10	1	0
Amastra _2	30	0	1

# Agregação

- ◊ Combinar dois ou mais atributos (ou objetos) em apenas um atributo (ou objeto), e.g.: cargo e função)
- ◊ Objetivo:
  - Reduzir o número de atributos ou amostras
  - Mudar escala (e.g: cidade em estado)
  - Possuir dados mais estáveis devido a menor variabilidade

# Feature scaling

Normalização: o propósito é **minimizar os problemas oriundos do uso de unidades e dispersões distintas entre as variáveis.**

Normalização segundo a amplitude: unidades diferentes ou dispersões muito heterogêneas.

◆ Min e max norm:

$$Y = \frac{X - \text{min}}{\text{Max} - \text{Min}}$$

◆ Média norma.:

$$Y = \frac{X - \text{media}}{\text{Max} - \text{Min}}$$

◆ Standardization

$$Y = \frac{X - \text{media}}{\text{std}}$$

# Feature scaling

Normalização distribucional: é interessante nas situações em que há distorção nos valores aberrantes, obtenção de simetria etc. Por exemplo: salário dos brasileiros

Exemplo mais comum:  
◊ Log X  
Salários (1000, 10000)



Pré-processamento do dado

# Merging



# MERGE



# Seleção de features



# Importância

- ◊ Otimizar modelo
- ◊ Facilitar a interpretação
- ◊ Obter *insights*
- ◊ Eliminar atributos insignificantes
- ◊ ...



# Filter method



- ◊ É independente do modelo de aprendizagem
  - ◊ Pode ser feito com base no conhecimento do negócio
- Exemplos:
- ◊ Seleção manual
  - ◊ Correlação de Pearson
  - ◊ Chi Square

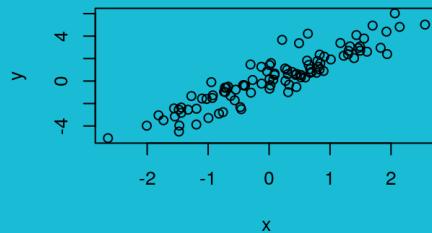
# Filter method

Feature/Response	Contínua	Categórica
Contínua	Correlação de Pearson	LDA
Categórica	Anova	Chi-Square

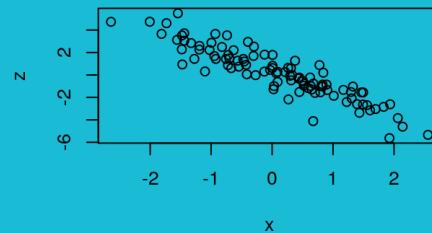
O que fazer com variáveis que são  
fortemente correlacionadas?

# Correlação de pearson

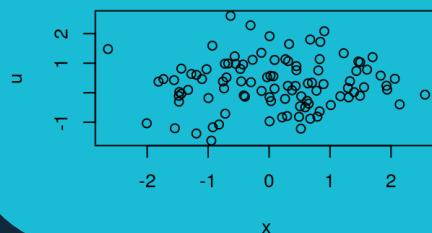
Relação linear positiva



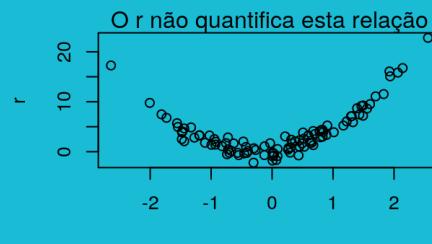
Relação linear negativa



Ausência de relação

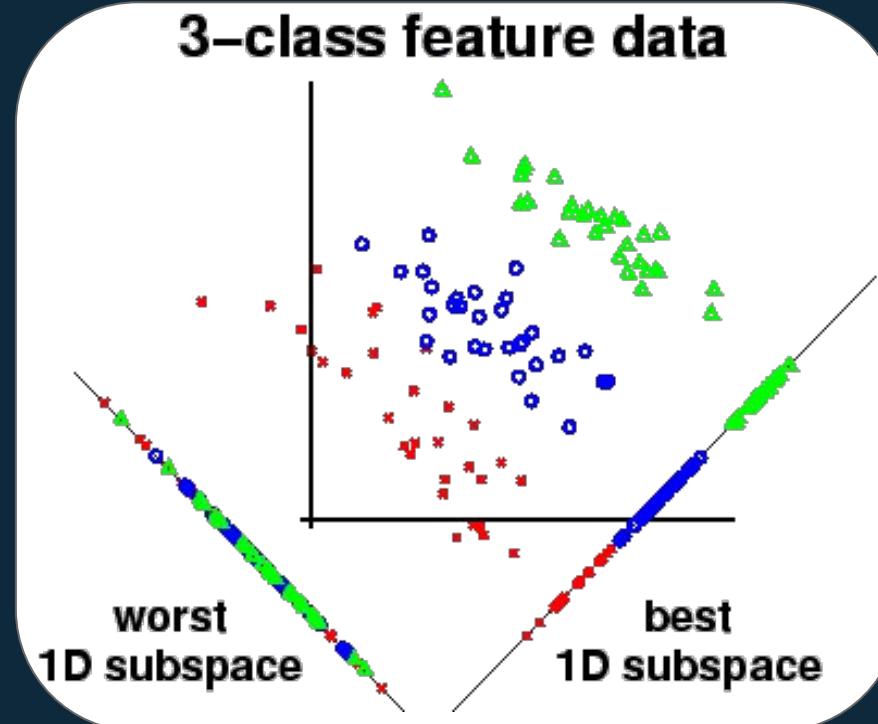


Relação não-linear

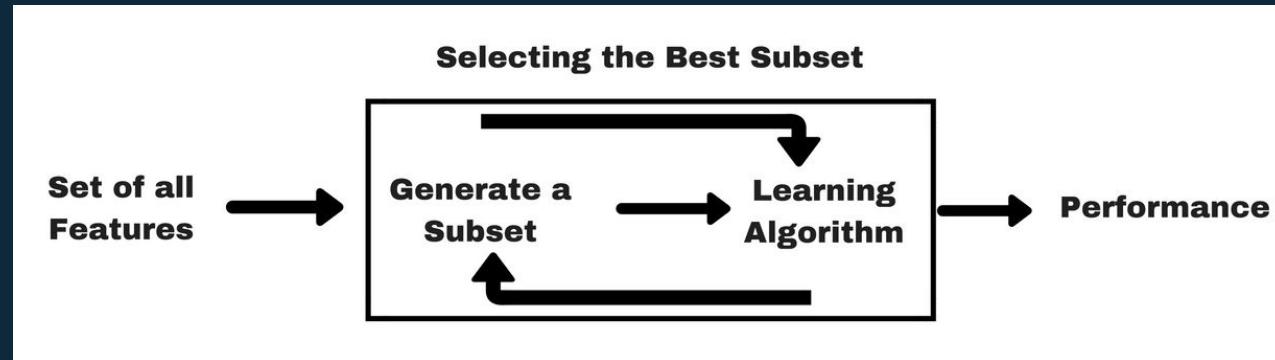


Filter method

# LDA - Análise Discriminante Linear



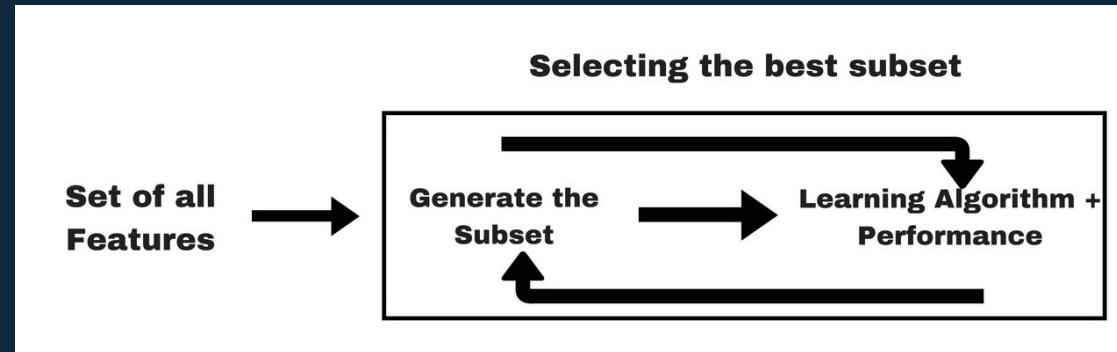
# Wrapper method



- ◊ Forward Selection
- ◊ Backward Elimination
- ◊ Recursive Feature elimination

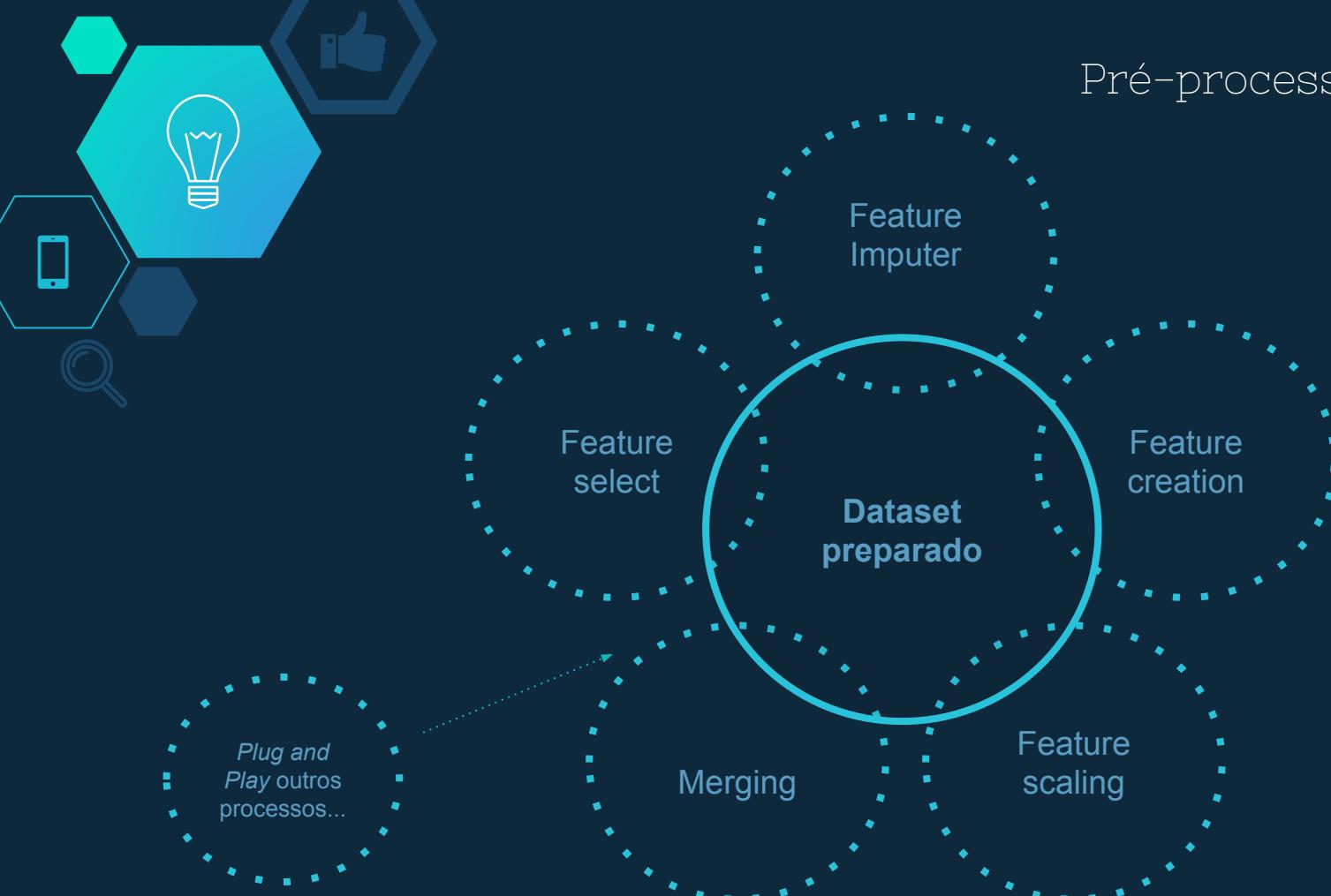
Qual a complexidade de todos os testes possíveis?

# Embedded method



- ◊ Ganho da informação
  - Modelos baseado em árvore
- ◊ Lasso regression performs L1
- ◊ Ridge regression performs L2

## Pré-processamento do dado





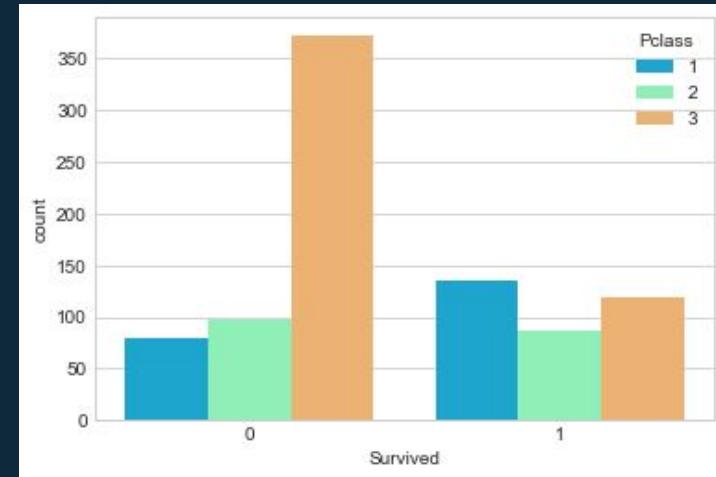
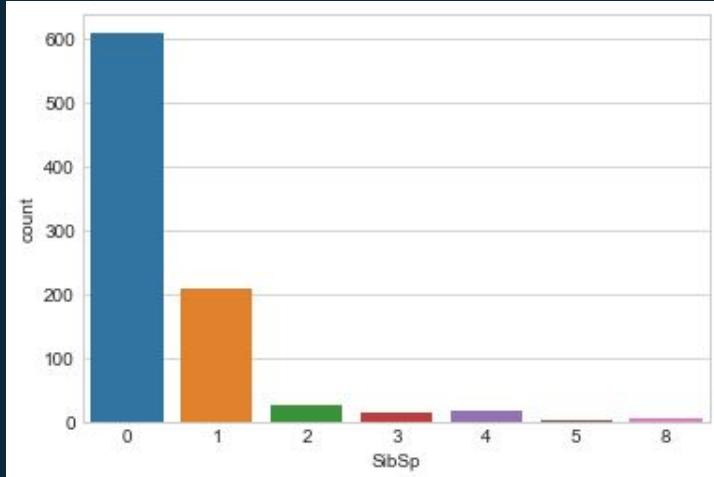
Qual o melhor  
pré-processamento  
do dado?



# Hands on

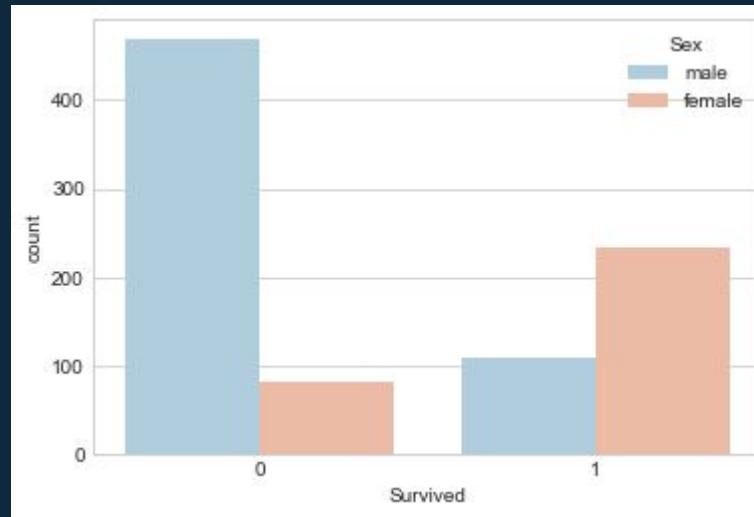


# Insights (e.g.: ...)





# Insights(e.g.: ...)





# Aprendizagem Supervisionada





“São apresentadas ao computador exemplos de entradas e saídas desejadas, fornecidas por um “professor”. O objetivo é aprender uma regra geral que mapeia as entradas para as saídas.”

# Conceitos

Algoritmo treinado sobre dados rotulados:

- ◊ Algoritmos de Classificação (e.g. tumor maligno e/ou benigno)
  - Prediz Valores discretos
  - Algoritmos: Árvore de Decisão, Regressão Logística, KNN, etc.
- ◊ Algoritmos Regressão linear (e.g. prever o preço de um imóvel)
  - Prediz Valores contínuos
  - Algoritmos: Regressão Linear e Polinomial, SVM Regressor, Árvore de Decisão

## + Exemplos

- ◊ Identificação de fraudes
- ◊ Detecção de epidemias
- ◊ Precisão de tratamentos
- ◊ Análise de sentimento
- ◊ Filtros de spam
- ◊ Cálculo de empréstimo



# Árvores de Decisão



## Árvore de Decisão



## Árvore de Decisão



## Conceitos

- ◊ “Árvores de decisão são métodos de aprendizado de máquinas supervisionado não-paramétricos, muito utilizados em tarefas de classificação e regressão.”
  
- ◊ “A construção de uma árvore de decisão é guiada pelo objetivo de diminuir a entropia, dificuldade de previsão, da variável alvo.”

# Conceitos

- ◊ **Entropia:** A entropia da informação, no caso do aprendizado de máquina, mede a impureza de um determinado conjunto de dados. Em outras palavras mede a dificuldade que se tem para saber qual a classificação de cada amostra dentro do meu conjunto de dados.

# Entropia

$$E(X) = - \sum_{i=1}^n p_i * \log_2 p_i$$

- ◊ **X:** é o atributo;
- ◊ **n:** quantidade de classes no conjunto de dados.
- ◊ **Pi:** probabilidade de cada uma delas acontecer para um dado atributo

“A entropia de um atributo é definida pela soma ponderada das entropias de suas partições.”

# Conceitos

- ◊ **Ganho de Informação:** O ganho de informação ao contrário da entropia mede a pureza de um determinado conjunto de dados, essa definição nada mais é do que a eficácia do atributo testado ao tentar classificar a base de dados.

$$GI(x) = E(\text{Classe}) - E(x)$$

## Conceitos

$$\diamond \quad GI(x) = E(Classe) - E(x)$$

- X: é o atributo
- E(Classe): é a entropia da classe no dataset
- E(x): é a entropia do atributo.

***Definida pela soma ponderada das entropias de suas partições.***

# Information Gain

ESCOLA	IDADE	LABEL(Bolsa?)
Part_bolsa	>18	Não
Particular	<=18	Não
Part_bolsa	<=18	Sim
Particular	>18	Não
Pública	<=18	Sim
Pública	>18	Sim
Part_bolsa	>18	Não
Part_bolsa	<=18	Sim



# Entropia da classe

## ◊ Entropia da classe:

- $E(\text{Bolsa}) = -\sum P_i \log_2 P_i$ 
  - ◊  $P_1(\text{bolsa=sim}) = 4/8 = 0,5$       ◊ 4 bolsistas
  - ◊  $P_2(\text{bolsa=sim}) = 4/8 = 0,5$       ◊ 4 não bolsistas
- $E(\text{Bolsa}) = -P_1 \log_2 P_1 - P_2 \log_2 P_2$
- $E(\text{Bolsa}) = -0,5 \log_2 0,5 - 0,5 \log_2 0,5$
- $E(\text{Bolsa}) = 1$



Information gain

# Entropia do atributo

<u>ESCOLA</u>	IDADE	LABEL(Bolsa?)
Part_bolsa	>18	Não
Particular	<=18	Não
Part_bolsa	<=18	Sim
Particular	>18	Não
Pública	<=18	Sim
Pública	>18	Sim
Part_bolsa	>18	Não
Part_bolsa	<=18	Sim

Bolsa?	PU	PA	PB
Não	0	2	2
Sim	2	0	2

# Information Gain

- ◊ Entropia do atributo (Escola):
  - $E_{esc}(Esc=PU) = -\sum P_i \cdot \log_2 P_i$ 
    - $P_1(\text{bolsa=sim}|Esc=PU) = 2/2 = 1$
    - $P_2(\text{bolsa=nao}|Esc=PU) = 0/2 = 0$
  - $E_{esc}(Esc=PU) = -P_1 \log_2 P_1 - P_2 \log_2 P_2$
  - $E_{esc}(Esc=PU) = -1 \log_2 1 - 0 \log_2 0$
  - **$E_{esc}(Esc=PU)=0$**

Bolsa?	PU	PB	PA
Não	0	2	2
Sim	2	2	0

# Information Gain

- ◊ Entropia do atributo (Escola):
  - $E_{esc}(Esc=PB) = -\sum P_i \cdot \log_2 P_i$ 
    - $P_1(\text{bolsa=sim}|Esc=PB) = 2/4 = 1/2$
    - $P_2(\text{bolsa=nao}|Esc=PB) = 2/4 = 1/2$
  - $E_{esc}(Esc=PB) = -P_1 \log_2 P_1 - P_2 \log_2 P_2$
  - $E_{esc}(Esc=PB) = -0,5 \log_2 0,5$   
 $-0,5 \log_2 0,5$
  - **$E_{esc}(Esc=PB)=1$**

Bolsa?	PU	PB	PA
Não	0	2	2
Sim	2	2	0

# Information Gain

◇ Entropia do atributo (Escola):

- $E_{esc}(Esc=PA) = -\sum P_i \cdot \log_2 P_i$ 
  - $P_1(\text{bolsa=sim}|Esc=PA) = 0/2 = 0$
  - $P_2(\text{bolsa=nao}|Esc=PA) = 2/2 = 1$
- $E_{esc}(Esc=PA) = -P_1 \log_2 P_1 - P_2 \log_2 P_2$
- $E_{esc}(Esc=PA) = -0 \log_2 0 - 1 \log_2 1$
- **$E_{esc}(Esc=PA)=0$**

Bolsa?	PU	PB	PA
Não	0	2	2
Sim	2	2	0

# Information Gain

“A entropia de um atributo é definida pela soma ponderada das entropias de suas partições.”

- ◊ Entropia do atributo (Escola)  $\Rightarrow E(\text{Escola}) = \sum P_i * E_{\text{esc}}(i)$ 
  - $E(\text{Esc=PU}) = 0$
  - $E(\text{Esc=PB}) = 1$
  - $E(\text{Esc=PA}) = 0$
- ◊  $E(\text{Esc}) = P(\text{Esc=PU}) * E(\text{Esc=PU}) + P(\text{Esc=PB}) * E(\text{Esc=PB}) + P(\text{Esc=PA}) * E(\text{Esc=PA})$
- ◊  $E(\text{Esc}) = (2/4) * 0 + (4/8) * 1 + (2/8) * 0$
- ◊  $E(\text{Esc}) = 0,5$

**Fazendo todos esses cálculos para a idade temos:**

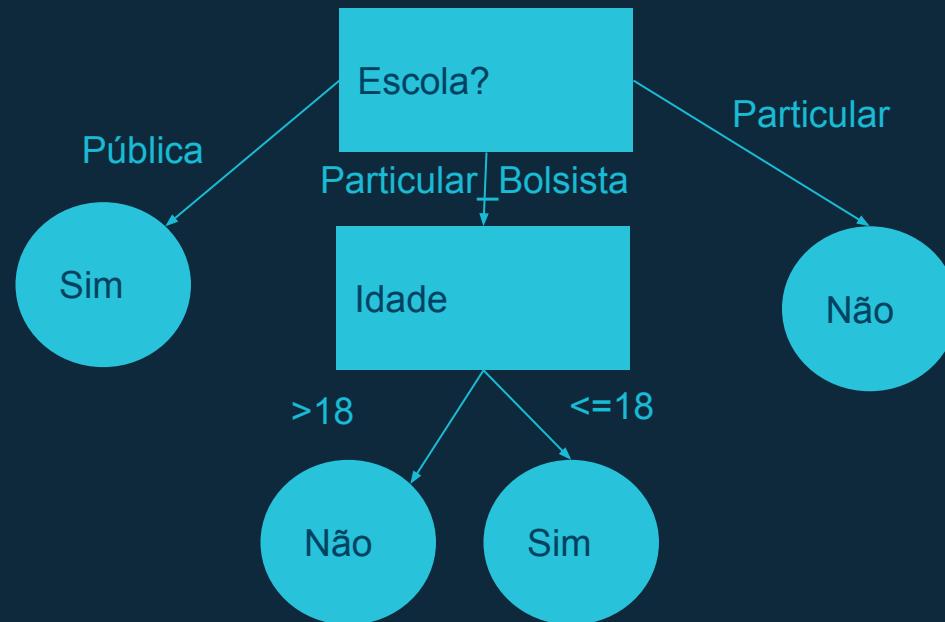
- ◊  $E(\text{Idade}) = 0,81$

# Information Gain

- ◊  $E(Esc) = 0,5$
- ◊  $E(Idade) = 0,81$
- ◊  $GI(Esc) = E(Bolsa) - E(Esc) = 1 - 0,5 = 0,5$
- ◊  $GI(Idade) = E(Bolsa) - E(Idade) = 1 - 0,811 = 0,189$

Qual o melhor atributo?

# Indução da árvore



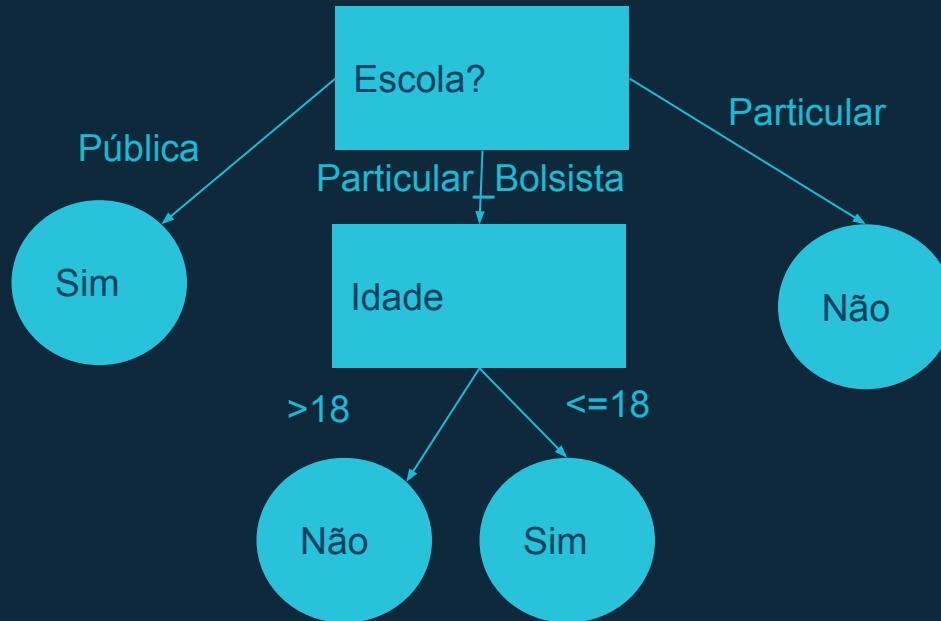
# Algoritmo de Indução

Algoritmo de indução:

- 1) Escolher uma feature
- 2) Estender a árvore adicionando um ramo para cada valor do atributo
- 3) Filtrar as amostras de acordo com o valor do atributo e enviar as amostras para a folha
- 4) Para cada folha:
  - a) Se as amostras forem da mesma classe, associar a folha.
  - i) Senão, repetir os passos de 1 até 4

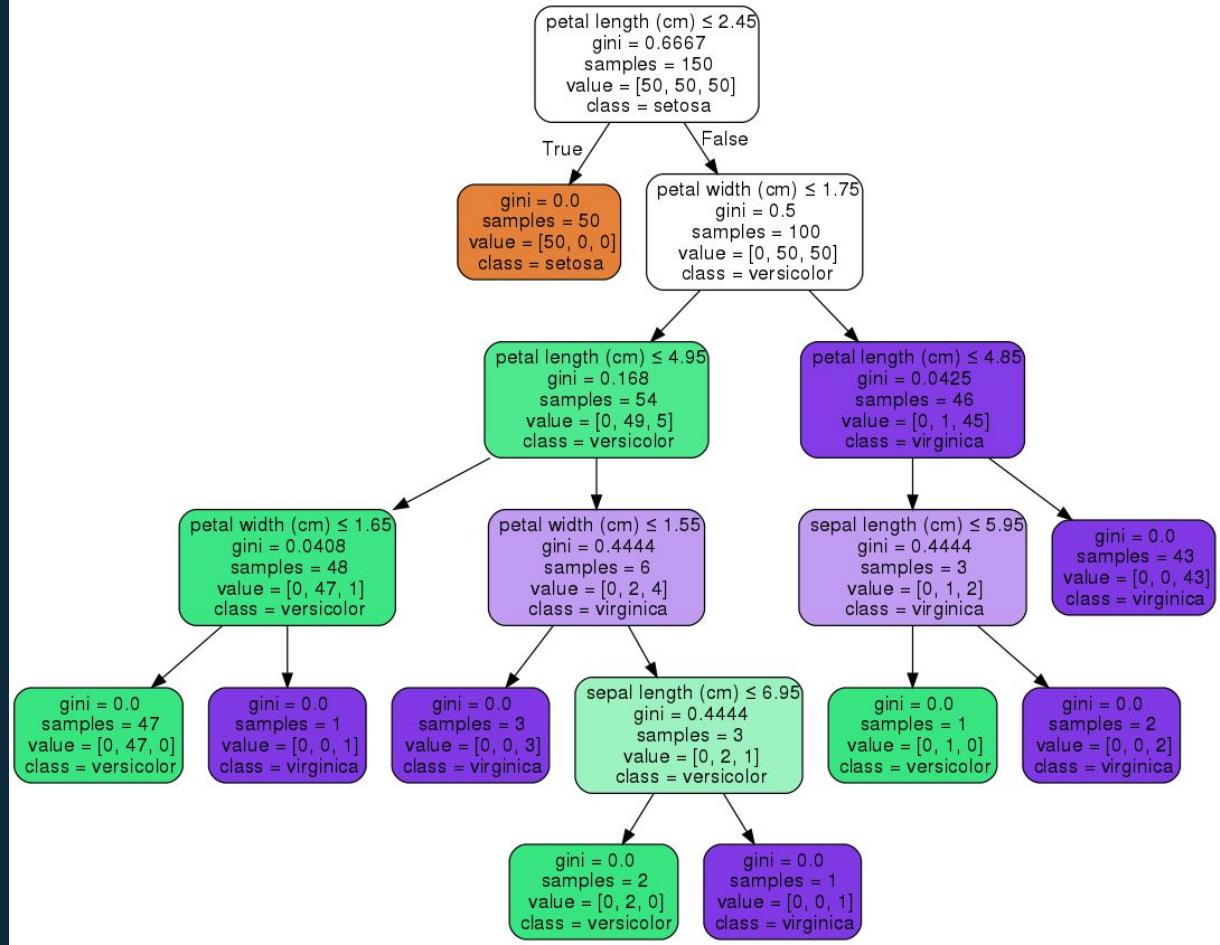
Qual o melhor atributo?

# Indução da árvore



# Árvore de Decisão

Exemplo de uma árvore  
de decisão para o  
problema de classificação  
de flores



# Considerações

- ◊ GI tem um bias que favorece a escolha de atributos com muitos valores;
- ◊ Para minimizar o *overfitting* deve-se:
  - Aplicar procedimentos de poda
  - Definir bem os hiperparâmetros
  - Selecionar atributos a priori, etc.



# Random Forest

- ◆ “Floresta Aleatória (Random Forest) é um algoritmo de aprendizagem de máquina flexível e fácil de usar que produz excelentes resultados a maioria das vezes, mesmo sem ajuste de hiperparâmetros. É também um dos algoritmos mais utilizados, devido à sua simplicidade e o fato de que pode ser utilizado para tarefas de classificação e também de regressão.”<sup>1</sup>

[1] - <https://medium.com/machina-sapiens/o-algoritmo-da-floresta-aleat%C3%B3ria-3545f6babdf8>



# Random Forest

- ◊ A “floresta” criada é uma combinação (ensemble) de árvores de decisão
- ◊ Treinados com o método de bagging, (amostras diferentes da base de dados que são usadas para aprender hipóteses diferentes)
- ◊ Busca a melhor característica em um subconjunto aleatório de todas as características

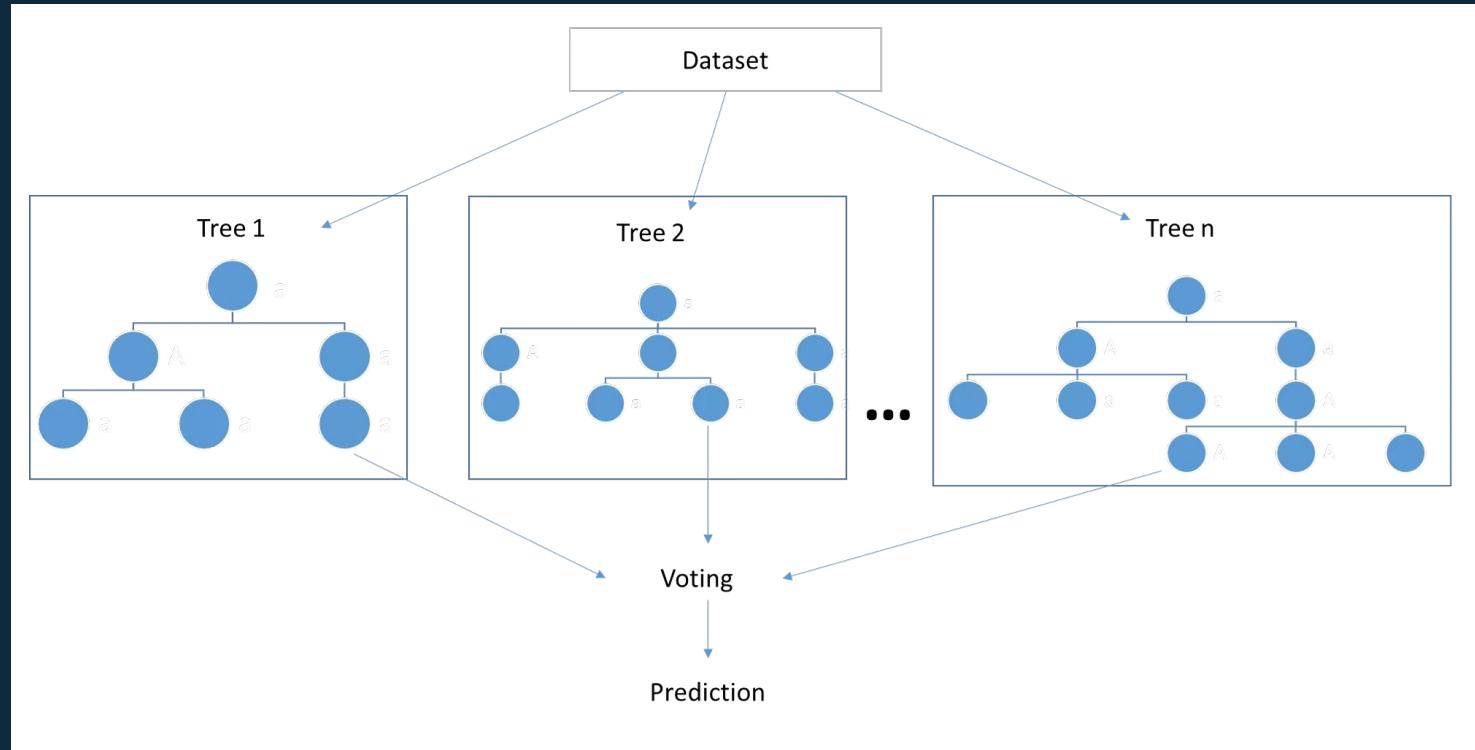


# Random Forest

- ◆ A previsão final para um exemplo de teste é a média da previsão de cada hipótese
- ◆ Cria diversidade, o que geralmente leva a geração de modelos melhores.
- ◆ Muito bom para se medir a importância relativa de cada característica (feature) para a predição



# Random Forest



# Considerações

## ◆ Vantagens:

- Poder ser utilizado tanto para regressão quanto para classificação
- É fácil visualizar a importância relativa que ele atribui para cada característica na suas entradas
- O número de hiperparâmetros não é tão grande e são fáceis de serem compreendidos.
- Diminui o overfitting se comparado a árvore de decisão

# Considerações

## ◆ Desvantagens:

- Uma quantidade grande de árvores pode tornar o algoritmo lento e ineficiente para previsões em tempo real.
- Muito lento para fazer previsões depois de treinados (São rápidos para treinar)
- Uma previsão com mais acurácia requer mais árvores, o que faz o modelo ficar mais lento



# Métricas de avaliação



# Matriz de confusão

		Valor Observado (valor verdadeiro)	
		Label. Pos. ( $Y=1$ )	Label. Neg. ( $Y=0$ )
Valor Preditivo	Pred. Pos. ( $Y=1$ )	VP (verdadeiro positivo)	FP (falso positivo)
	Pred. Neg. ( $Y=0$ )	FN (falso negativo)	VN (verdadeiro negativo)



# Acurácia

- ◊ A proporção de previsões corretas, sem levar em consideração o que é positivo e o que é negativo. .
- ◊ 
$$\text{ACC} = \frac{\text{Total de Acertos}}{\text{Total de dados no conjunto}}$$
$$= \frac{(\text{VP} + \text{VN})}{(\text{VP} + \text{FP} + \text{VN} + \text{FN})}$$

	Label. Pos.	Label. Neg.
Pred. Pos.	VP	FP
Pred. Neg.	FN	VN



# Precisão

- ◊ É a fração de instâncias **recuperadas** que são relevantes.
- ◊  $\text{PREC} = \text{ACERTOS POSITIVOS} / \text{TOTAL DE ACERTOS PREDITOS COMO POSITIVOS}$
- ◊  $\text{PREC} = \text{VP} / (\text{VP} + \text{FP})$

	Label. Pos.	Label. Neg.
Pred. Pos.	VP	FP
Pred. Neg.	FN	VN

# Recall (Sensibilidade)

- ◆ A capacidade do sistema em predizer corretamente a condição para casos que realmente a têm (é a proporção de verdadeiros positivos). É a fração de instâncias relevantes que são recuperadas
- ◆ Também conhecida como sensibilidade, revocação ou true positive rate (TPR)
- ◆ REC = ACERTOS POSITIVOS / TOTAL DE POSITIVOS

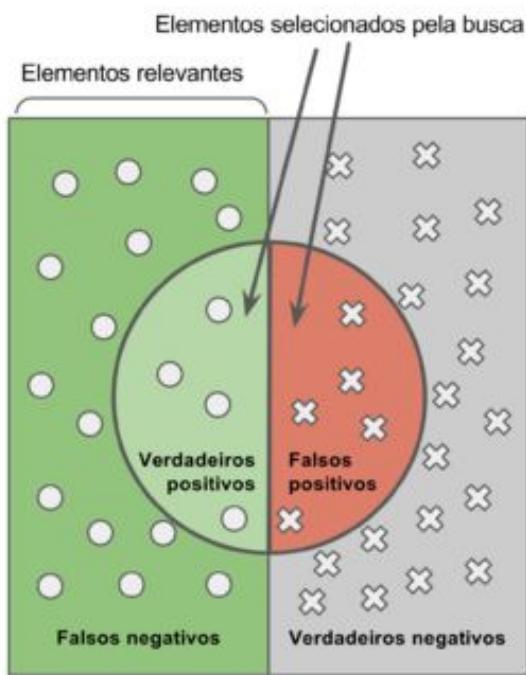
$$= VP / (VP + FN)$$

	Label. Pos.	Label. Neg.
Pred. Pos.	VP	FP
Pred. Neg.	FN	VN



# Precisão vs Recall

	Label. Pos. (Y=1)	Label. Neg. (Y=0)
Pred. Pos. (Y'=1)	20	10
Pred. Neg. (Y'=0)	40	30



$$\text{Precisão} = \frac{\text{Verdadeiros positivos}}{\text{Elementos selecionados}}$$

"Quantos elementos selecionados são relevantes?"

$$\text{Revocação} = \frac{\text{Verdadeiros positivos}}{\text{Elementos relevantes}}$$

"Quantos elementos relevantes foram selecionados?"

# F1-Score

- ◊ Essa métrica combina precisão e recall de modo a trazer um valor único que indique a qualidade geral do modelo
- ◊ Trabalha bem mesmo com conjuntos de dados que possuem classes desproporcionais.
- ◊ 
$$\text{F1-Score} = \frac{2 * \text{PRECISAO} * \text{RECALL}}{\text{PRECISAO} + \text{RECALL}}$$

# Especificidade

- ◆ A proporção de verdadeiros negativos: a capacidade do sistema em predizer corretamente a ausência da condição para casos que realmente não a têm.
- ◆ Também conhecida como true negative rate (TNR)
- ◆  $SPEC = ACERTOS\ NEGATIVOS / TOTAL\ DE\ NEGATIVOS$

$$= VN / (VN + FP)$$

	Label. Pos.	Label. Neg.
Pred. Pos.	VP	FP
Pred. Neg.	FN	VN

# Exemplo

Após executar um classificador, que classifica os clientes da Cartoes&CIA entre bons e maus pagadores, sobre 100 amostras (55 como bons pagadores e 45 como maus pagadores), nós obtivemos o seguinte resultado:

Dos 55 **bons pagadores** apenas 40 foram preditos corretamente, e dos 45 **maus pagadores** apenas 35 foram preditos corretamente. Construa a matriz de confusão e calcule as seguintes métricas Precisão, Recall, Acurácia, F1-Score e Especificidade.

# Exemplo

Matriz de confusão.

	$Y = BP = 1$	$Y = MP = 0$
$Y' = BP = 1$	40	10
$Y' = MP = 0$	15	35

# Curva ROC

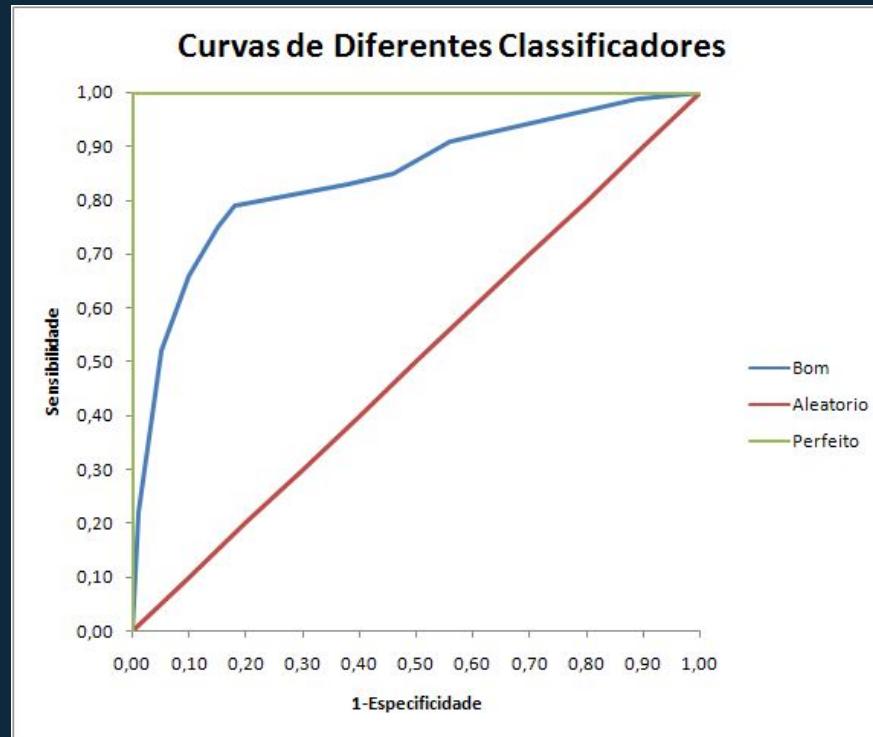
- ◊ Criada por engenheiros elétricos e de sistemas de radar durante a Segunda Guerra Mundial para detectar objetos inimigos em campos de batalha
- ◊ Os algoritmos de classificação produzem um valor situado dentro de um determinado intervalo contínuo, como  $[0;1]$ , é necessário definir um ponto de corte, ou um limiar de decisão, para se classificar e contabilizar o número de previsões positivas e negativas.

# Curva ROC

- ◊ Este limiar pode ser selecionado arbitrariamente, a melhor prática para se comparar o desempenho de diversos sistemas é estudar o efeito de seleção de diversos limiares sobre o resultado das previsões.



# Área Sob Curva ROC



# Hands-On





# Análise do Overfitting e Underfitting

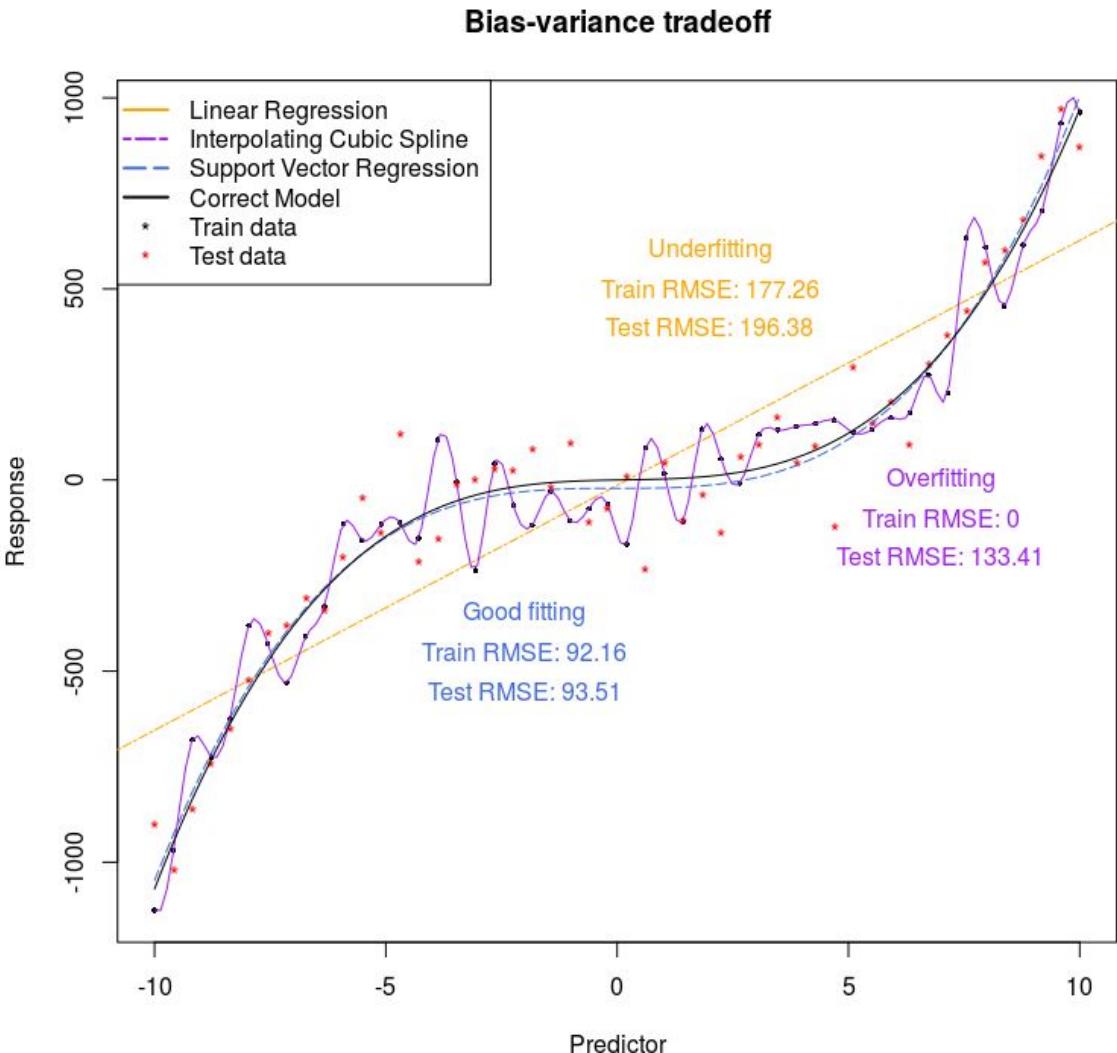


# Definições

- ◆ **Underfitting** - Utilizar um modelo simples que é bem generalizável, **mas não reduz consideravelmente** o erro de previsão no train set. Nesse caso estamos optando por um modelo com viés mais alto, mas variância baixa.
- ◆ **Overfitting** - Utilizar um modelo complexo que é capaz de **reduzir consideravelmente** o erro de previsão no train set, mas ao mesmo tempo **não é tão generalizável a ponto de apresentar um bom resultado no test set**. Nesse caso estamos optando por estimar um modelo com viés baixo e variância alta.



# Bias/ Variance tradeoff



# Bias vs Variance

$$E[(y - \hat{f}(x))^2] = Bias[\hat{f}(x)]^2 + Var[\hat{f}(x)] + \sigma^2$$

$$Bias[\hat{f}(x)] = E[\hat{f}(x) - f(x)],$$

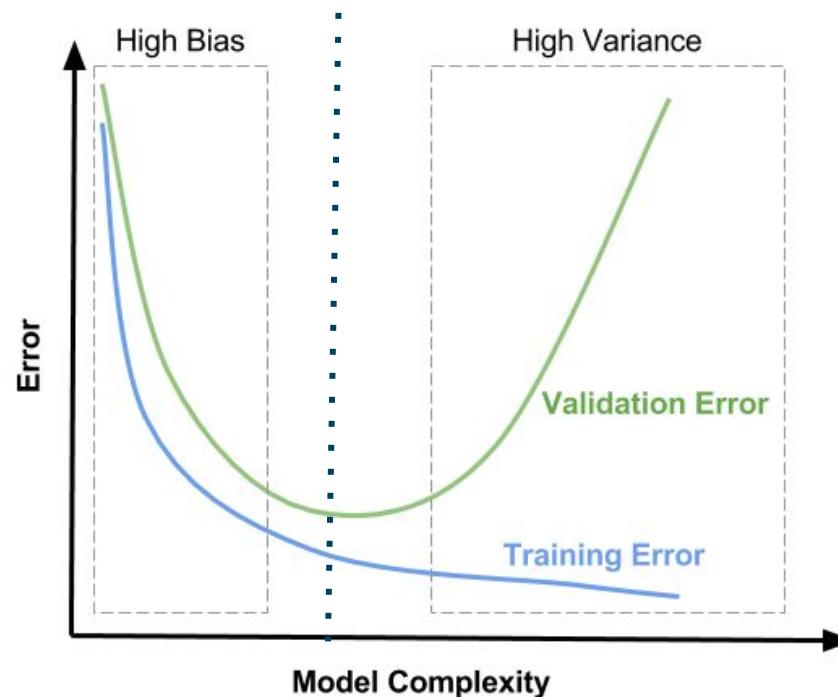
$$Var[\hat{f}(x)] = E[\hat{f}(x)^2] - E[\hat{f}(x)]^2.$$

# Bias vs Variance

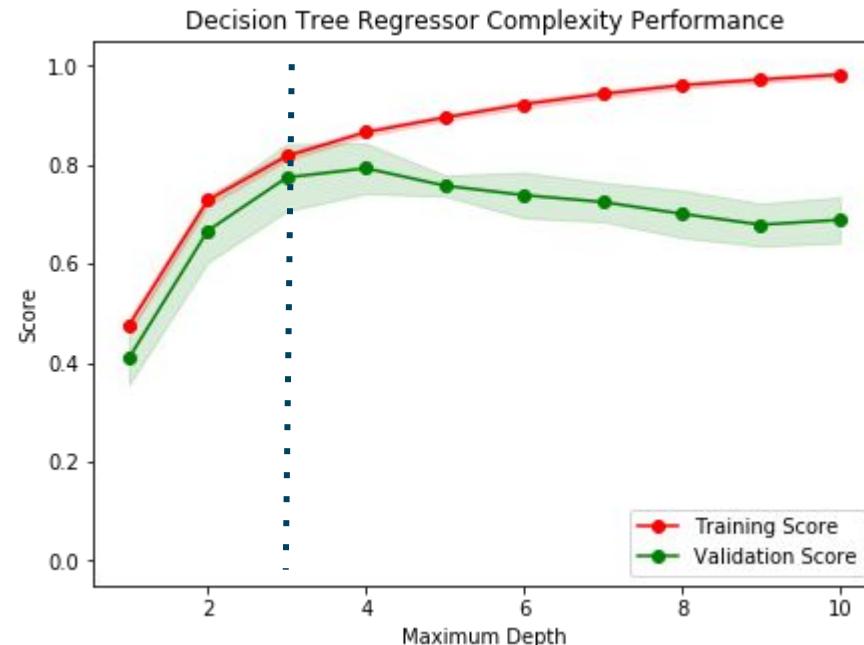
- ◊ O objetivo é escolher a  $f(x)$ , um modelo, próxima do ideal, visto que tanto o viés quanto a variância aumentam o erro de previsão.
- ◊ Obviamente a escolha entre bias e variance é um tradeoff, e o ideal é permanecermos em um meio termo entre um modelo complexo e um bem generalizável.



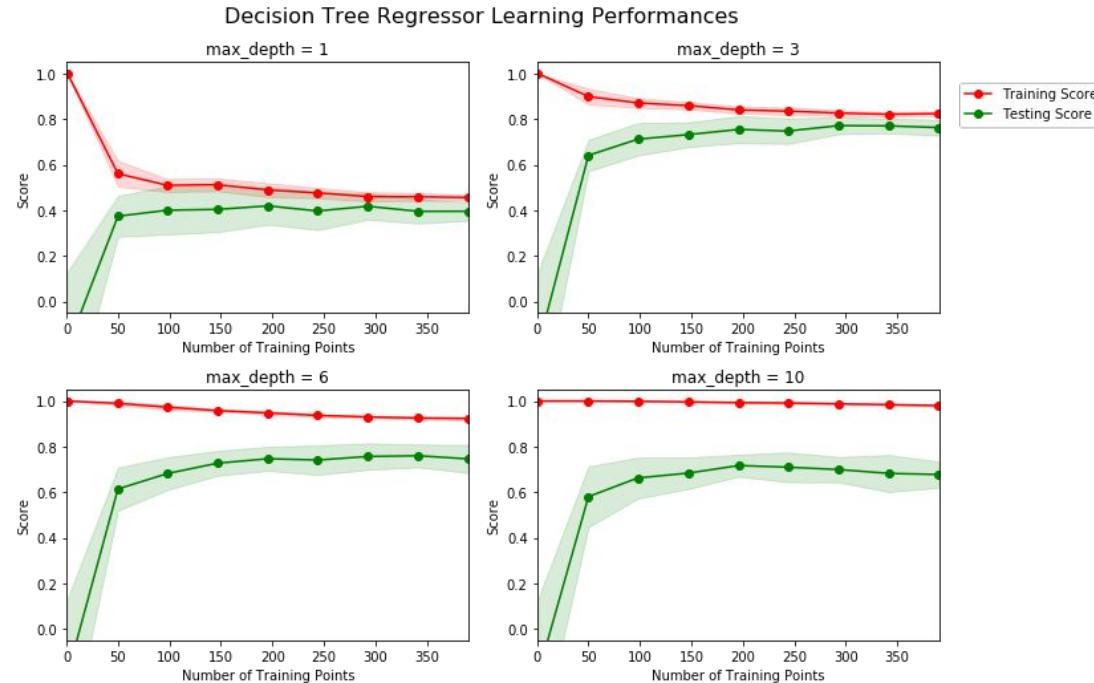
# Complexidade do modelo



# Overfitting vs Underfitting



# Overfitting vs Underfitting



# Grid Search

- ◆ Tem por objetivo identificar qual o melhor hiperparâmetro do algoritmo de aprendizagem de máquina que pode ser utilizado durante a construção do modelo, considerando a métrica de avaliação utilizada para validar o quanto bom um modelo é.
  
- ◆ Para ser utilizada basta que você especifique quais valores, ou range, de hiperparâmetros que você deseja testar.

# Grid Search

- ◊ Sempre que formos aplicar esta técnica devemos ter em mente que este processo é **custoso**, pois **quanto mais valores a serem analisados mais demorado será**.

## MODEL SELECTION

Finding the machine learning algorithm and its hyperparameter values that produce the best model.

Chris Albon



# Exemplo

```
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import GridSearchCV
# Create the parameter grid based on the results of random search
param_grid = {
    'max_depth': [80, 90, 100, 110],
    'max_features': [2, 3],
    'min_samples_leaf': [3, 4, 5],
    'min_samples_split': [8, 10, 12],
    'n_estimators': [100, 200, 300, 1000]
}
# Create a based model
rf = RandomForestRegressor()
# Instantiate the grid search model
grid_search = GridSearchCV(estimator = rf, param_grid = param_grid, scoring='accuracy')
```

# Hands On

