

FIAP

FIAP GRADUAÇÃO

TDS - FIAP

INTELIGÊNCIA ARTIFICIAL E CHATBOT

Profª . Miguel Bozer da Silva
profmiguel.silva@fiap.com.br

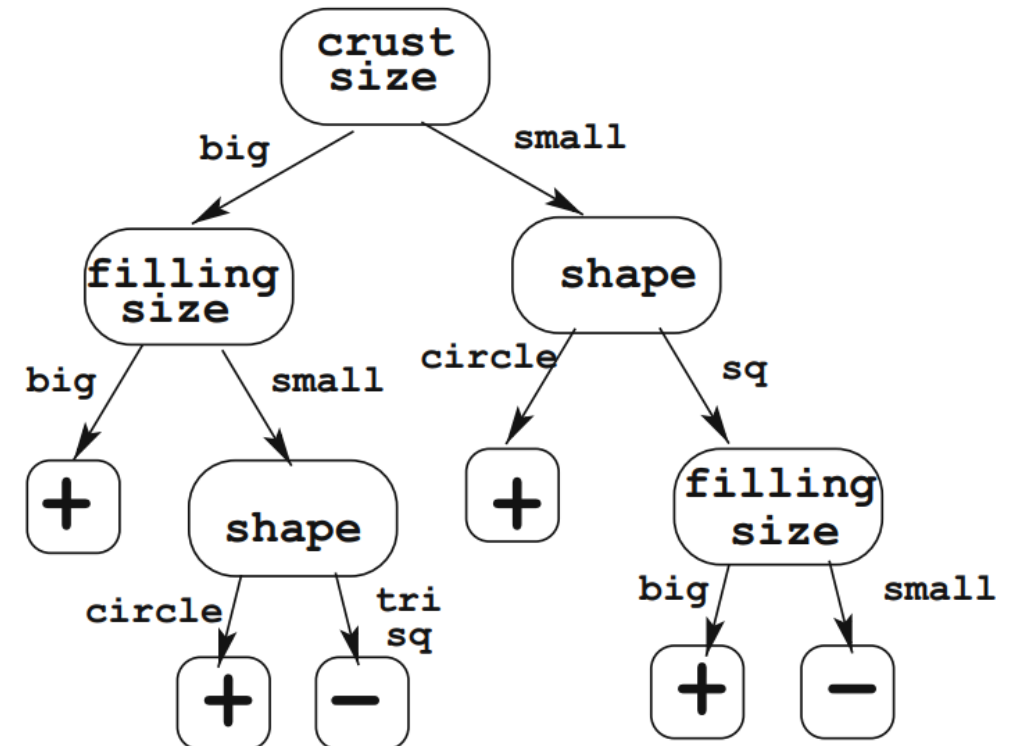
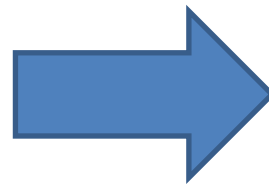
ÁRVORE DE DECISÃO

Profª . Miguel Bozer da Silva
 profmiguel.silva@fiap.com.br

Árvores de Decisão

- Árvores de decisão é um tipo de classificador que as decisões são baseadas nas condições dos atributos

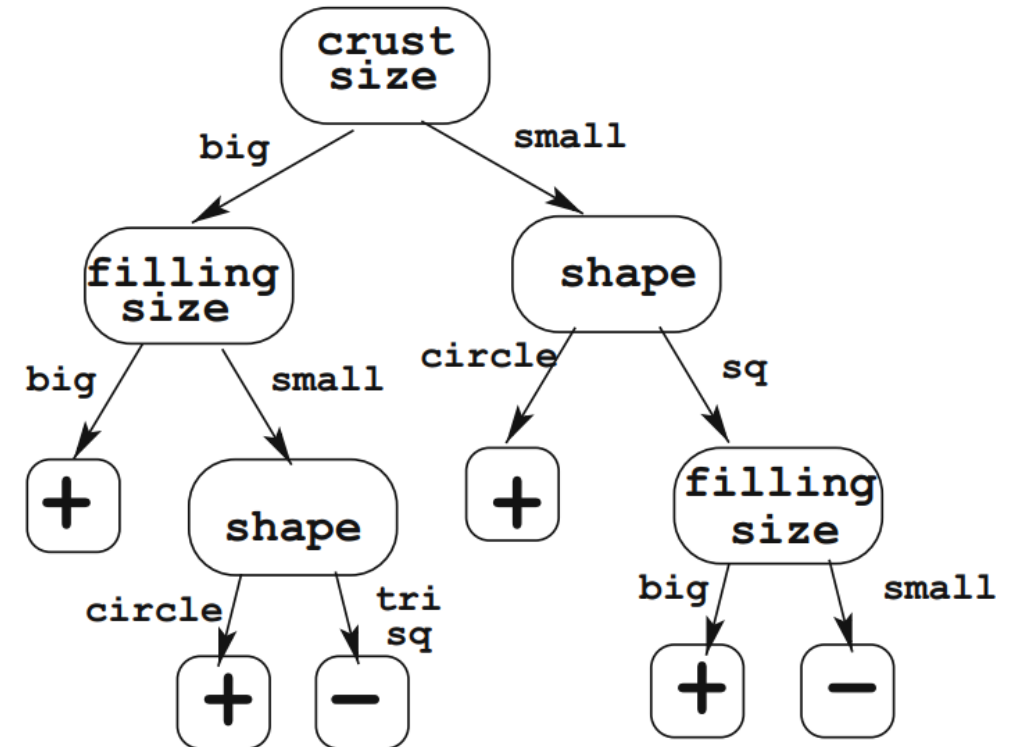
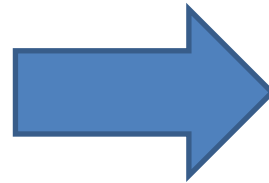
Example	crust size	shape	filling size	Class
e1	big	circle	small	pos
e2	small	circle	small	pos
e3	big	square	small	neg
e4	big	triangle	small	neg
e5	big	square	big	pos
e6	small	square	small	neg
e7	small	square	big	pos
e8	big	circle	big	pos



Árvores de Decisão

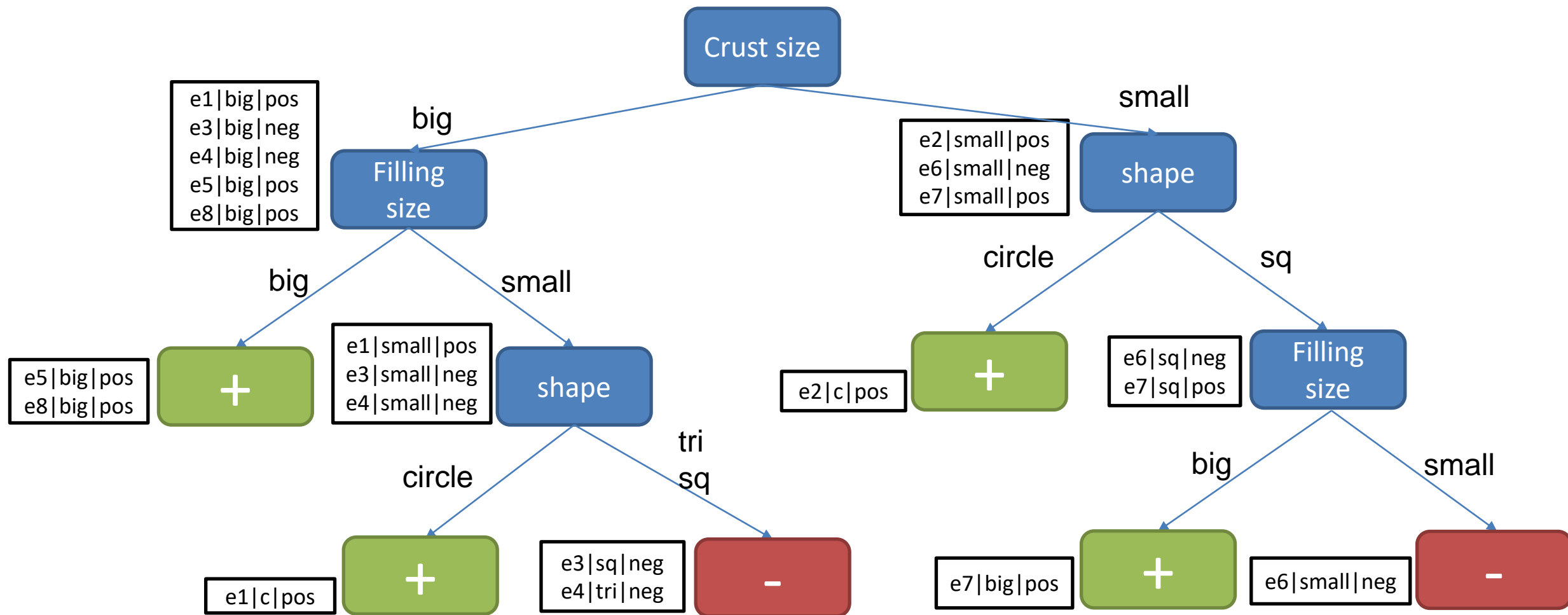
- Os nós da árvore representam condições e as folhas são as classificações

Example	crust size	shape	filling size	Class
e1	big	circle	small	pos
e2	small	circle	small	pos
e3	big	square	small	neg
e4	big	triangle	small	neg
e5	big	square	big	pos
e6	small	square	small	neg
e7	small	square	big	pos
e8	big	circle	big	pos



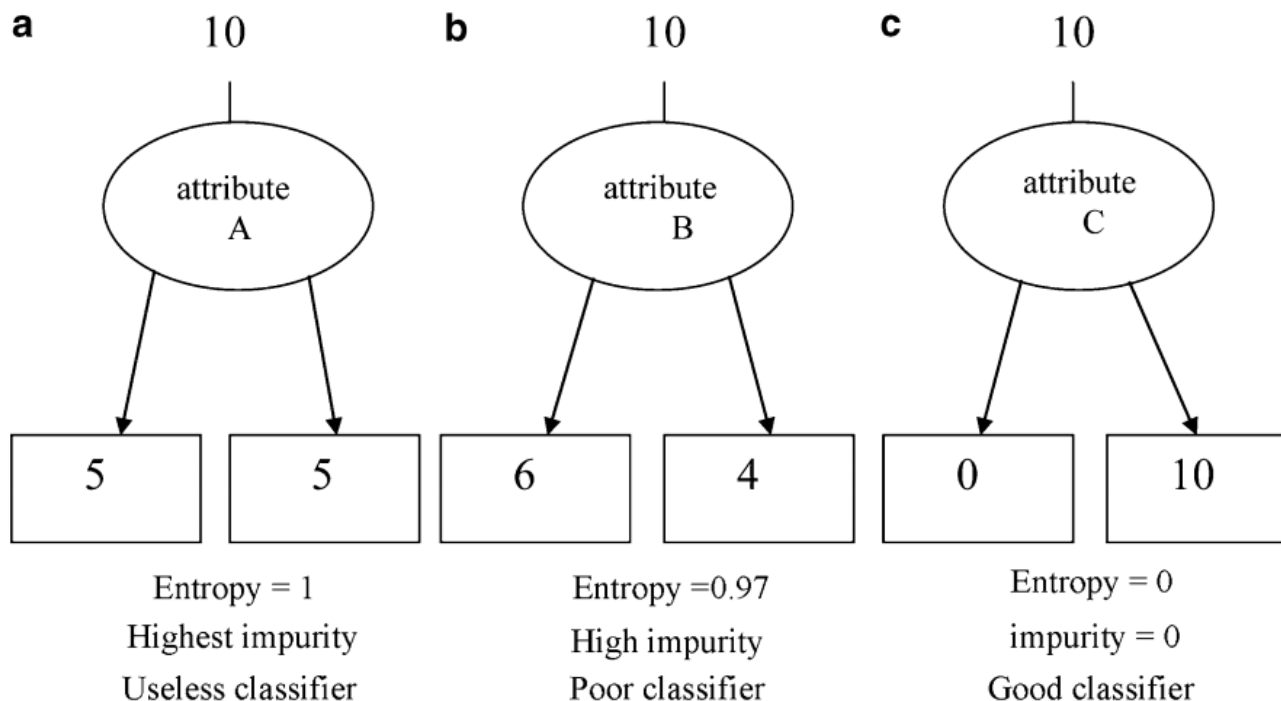
Árvores de Decisão

- Dessa forma a classificação é realizada de acordo com certas condições que os nossos dados possuem



Árvores de Decisão

- A pergunta que podemos ter: **Como decidir em quais perguntas e escolher a sequencia correta de perguntas?**
 - **Para isso usamos a entropia que nos ajuda a checar quais atributos melhor dividem os nossos dados:**



Entropia:

$$H(p) = - \sum_{i=1}^c p_i \log_2 p_i$$

Entropia nesse caso nos fornece a ideia da quantidade de impurezas que temos após a saída do nosso nó

Quanto menor a entropia, melhor a divisão dos dados. Logo melhor para a classificação

- Entretanto, ainda não temos como decidir qual o melhor atributo que podemos escolher em cada nó de nossa árvore.
 - Para medir isso o algoritmo de árvores de decisão usa o ganho de informação.
 - O Ganho significa o quanto removemos de impurezas ao escolher um dado atributo para dividir meus dados
 - $\text{Ganho}(S,A)$: Ganho de um atributo A com relação a todos os dados S

$$\text{Gain}(S,A) = \text{Impurity}(S) - \sum_{i=1}^k \frac{|S_{vi}|}{|S|} \text{Impurity}(S_{vi})$$

Exemplos com cada atributo após a divisão

Entropia para cada atributo após a divisão

Entropia antes da divisão dos dados

Total de exemplos

- Exemplo:

Examples	Weather	Parents visiting?	Money	Decision (category)
1	Sunny	Yes	Rich	Cinema
2	Sunny	No	Rich	Tennis
3	Windy	Yes	Rich	Cinema
4	Rainy	Yes	Poor	Cinema
5	Rainy	No	Rich	Stay in
6	Rainy	Yes	Poor	Cinema
7	Windy	No	Poor	Cinema
8	Windy	No	Rich	Shopping
9	Windy	Yes	Rich	Cinema
10	Sunny	No	Rich	Tennis

Cinema: 6x
Tennis: 2x
Stay in: 1x
Shopping: 1x

Entropia:

$$H(S) = -0,6 \times \log_2 0,6 - 0,2 \times \log_2 0,2 - 2 \times 0,1 \times \log_2 0,1 = 1,571$$

- Determinando o melhor ganho - $G(S, \text{parents})$; $G(S, \text{weather})$; $G(S, \text{money})$:

Examples	Weather	Parents visiting?	Money	Decision (category)	
1	Sunny	Yes	Rich	Cinema	Yes: Cinema: 5x Tennis: 0 Stay in: 0 Shopping: 0
2	Sunny	No	Rich	Tennis	
3	Windy	Yes	Rich	Cinema	
4	Rainy	Yes	Poor	Cinema	
5	Rainy	No	Rich	Stay in	No: Cinema: 1x Tennis: 2x Stay in: 1x Shopping: 1x
6	Rainy	Yes	Poor	Cinema	
7	Windy	No	Poor	Cinema	
8	Windy	No	Rich	Shopping	
9	Windy	Yes	Rich	Cinema	
10	Sunny	No	Rich	Tennis	

$$\begin{aligned} \text{Gain}(S, \text{parents}) &= 1.571 - (|S_{\text{yes}}|/10) \times \text{Entropy}(S_{\text{yes}}) - (|S_{\text{no}}|/10) \\ &\quad \times \text{Entropy}(S_{\text{no}}) = 1.571 - (0.5) \times 0 - (0.5) \times (1.922) = 0.61 \end{aligned}$$

Árvores de Decisão

- Determinando o melhor ganho - $G(S, \text{parents})$; $G(S, \text{weather})$; $G(S, \text{money})$:

Examples	Weather	Parents visiting?	Money	Decision (category)
1	Sunny	Yes	Rich	Cinema
2	Sunny	No	Rich	Tennis
3	Windy	Yes	Rich	Cinema
4	Rainy	Yes	Poor	Cinema
5	Rainy	No	Rich	Stay in
6	Rainy	Yes	Poor	Cinema
7	Windy	No	Poor	Cinema
8	Windy	No	Rich	Shopping
9	Windy	Yes	Rich	Cinema
10	Sunny	No	Rich	Tennis

Sunny:
Cinema: 1x
Tennis: 2x
Stay in: 0
Shopping: 0

Windy:
Cinema: 3x
Tennis: 0
Stay in: 0
Shopping: 1x

Rainy:
Cinema: 2x
Tennis: 0
Stay in: 1x
Shopping: 0

$$\begin{aligned}\text{Gain}(S, \text{weather}) &= 1.571 - (|S_{\text{sunny}}|/10) \times \text{Entropy}(S_{\text{sunny}}) - (|S_{\text{windy}}|/10) \\ &\quad \times \text{Entropy}(S_{\text{windy}}) - (|S_{\text{rainy}}|/10) \times \text{Entropy}(S_{\text{rainy}}) \\ &= 1.571 - (0.3) \times (0.918) - (0.4) \times (0.8113) - (0.3) \\ &\quad \times (0.918) = 0.70\end{aligned}$$

- Determinando o melhor ganho - $G(S, \text{parents})$; $G(S, \text{weather})$; $G(S, \text{money})$:

Examples	Weather	Parents visiting?	Money	Decision (category)
1	Sunny	Yes	Rich	Cinema
2	Sunny	No	Rich	Tennis
3	Windy	Yes	Rich	Cinema
4	Rainy	Yes	Poor	Cinema
5	Rainy	No	Rich	Stay in
6	Rainy	Yes	Poor	Cinema
7	Windy	No	Poor	Cinema
8	Windy	No	Rich	Shopping
9	Windy	Yes	Rich	Cinema
10	Sunny	No	Rich	Tennis

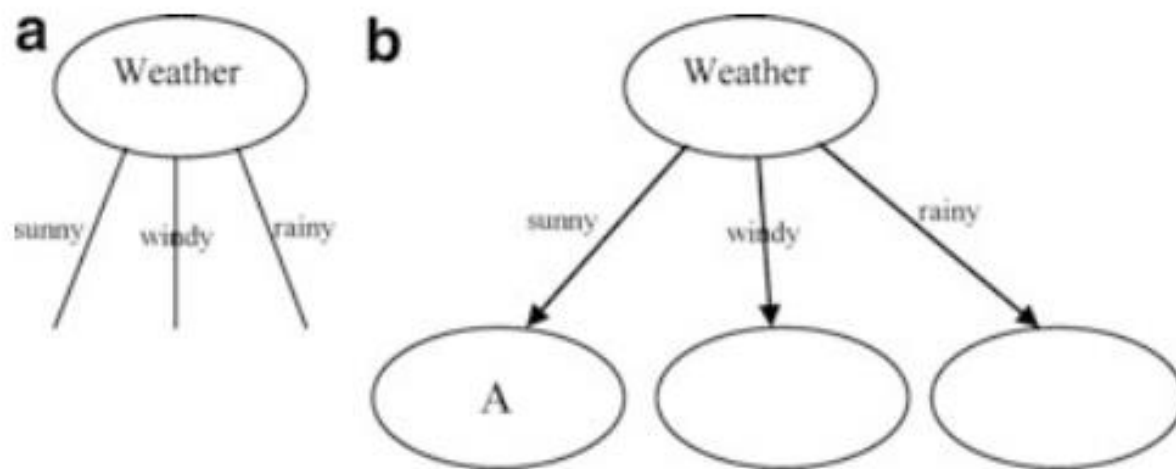
Rich:
 Cinema: 3x
 Tennis: 2x
 Stay in: 1x
 Shopping: 1x

Poor:
 Cinema: 3x
 Tennis: 0
 Stay in: 0
 Shopping: 1x

$$\begin{aligned}
 \text{Gain}(S, \text{money}) &= 1.571 - (|S_{\text{rich}}|/10) \times \text{Entropy}(S_{\text{rich}}) \\
 &\quad - (|S_{\text{poor}}|/10) \times \text{Entropy}(S_{\text{poor}}) \\
 &= 1.571 - (0.7) \times (1.842) - (0.3) \times 0 = 0.2816
 \end{aligned}$$

Árvores de Decisão

- Entre os três primeiros, Weather tem o maior ganho. Logo ele deve ser o primeiro ramo.

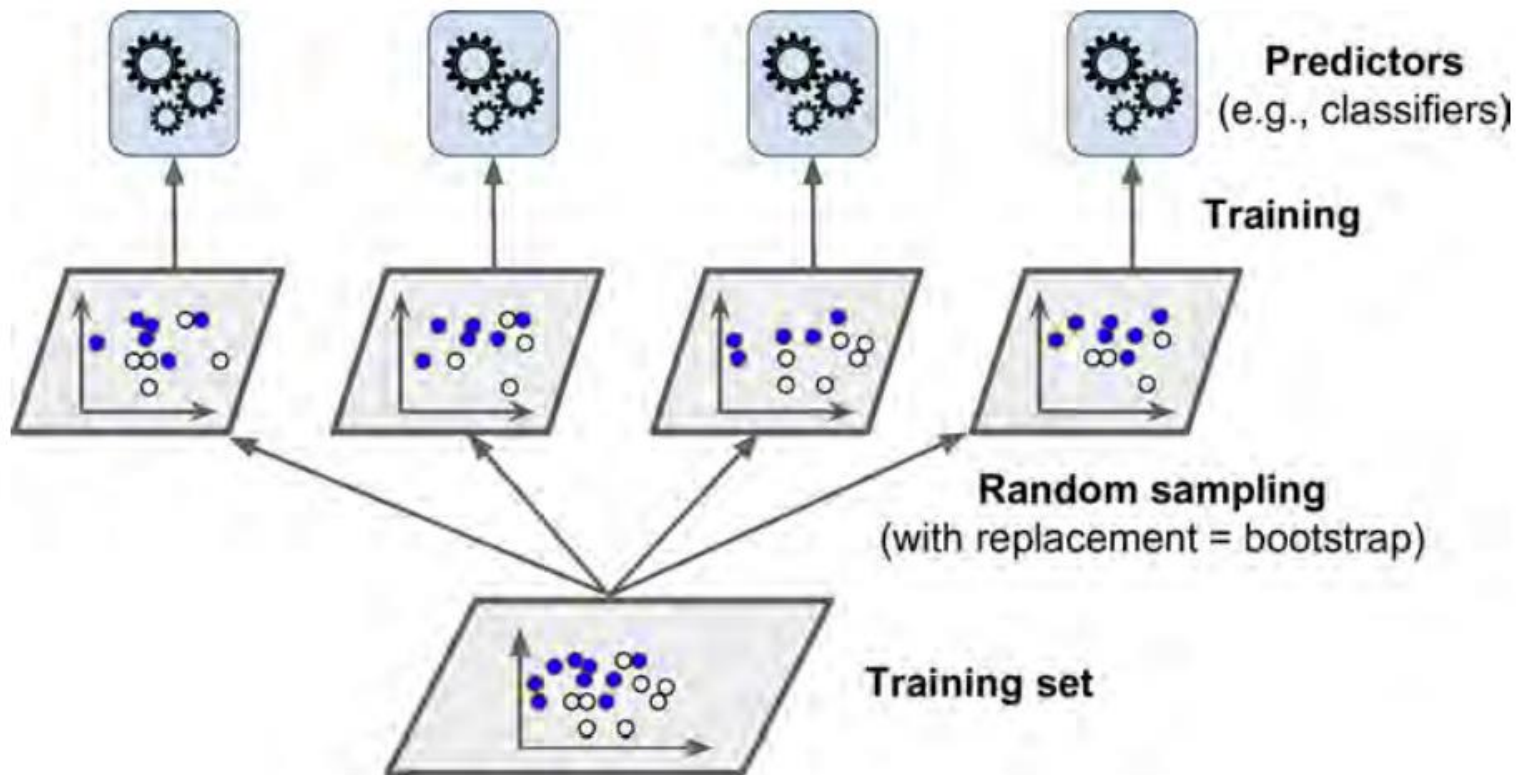


Procedimento se repete agora para cada ramo até classificar todos os itens



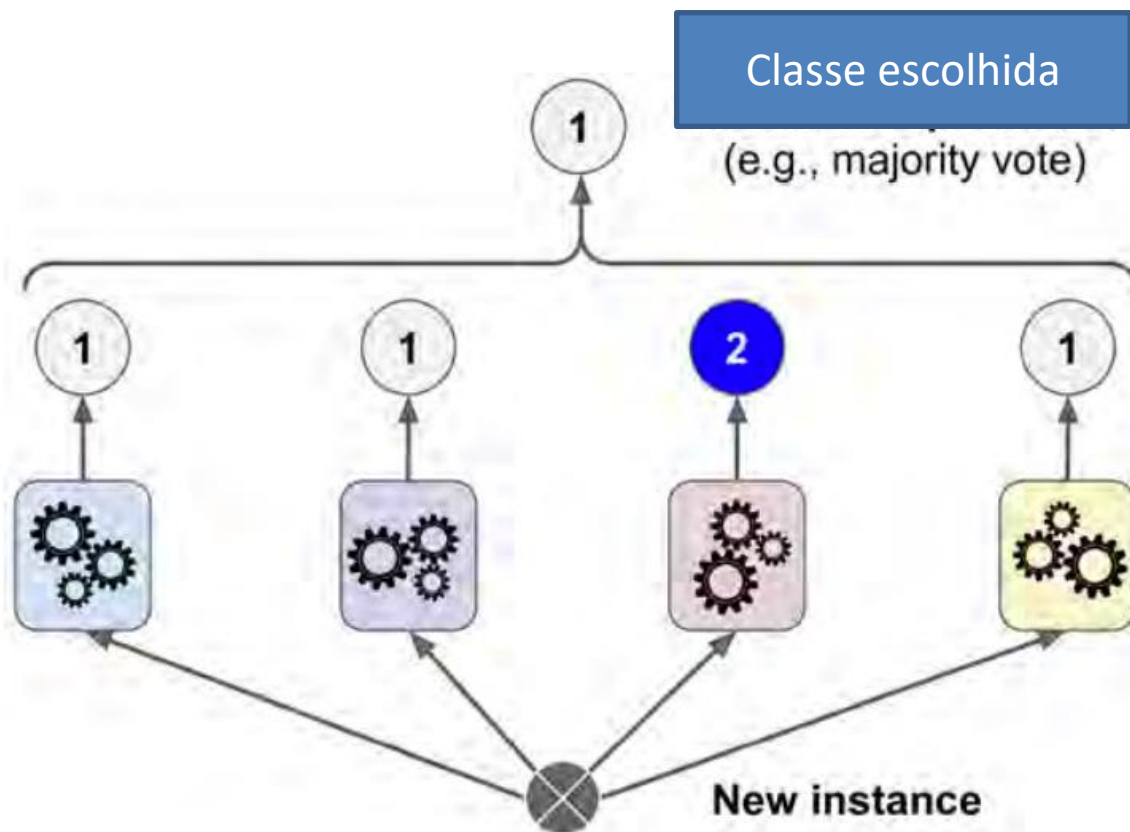
RANDOM FOREST

Random Forest



- Criamos diferentes árvores de decisão para o mesmo conjunto de treinamento;
- Cada árvore é treinada com um subconjunto dos dados de treinamento;
- A saída é determinada a partir da votação de todas as árvores. A mais votada será escolhida

Random Forest



Predictions

Árvores de
Decisão

- A saída é determinada a partir da votação de todas as árvores. A mais votada será escolhida

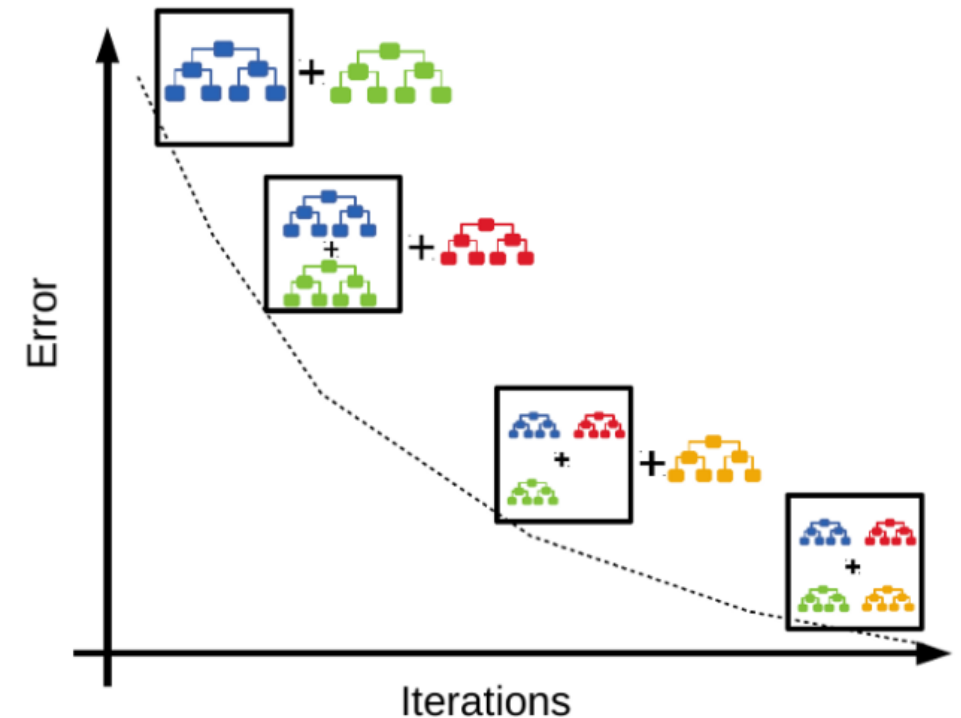


GRADIENT BOOSTING CLASSIFIER

Gradient Boosting Classifier (GBC)

Os classificadores que usam o método de Boosting possuem a ideia de treinar os seus classificadores sequencialmente tentando corrigir os seus predecessores.

O GBC adiciona novos classificadores ao seu modelo tentando reduzir o erro residual dos modelos predecessores



<https://medium.com/swlh/gradient-boosting-trees-for-classification-a-beginners-guide-596b594a14ea>

- DOUGHERTY, Geoff. **Pattern Recognition and Classification: an introduction**. New York: Springer International Publishing, 2013.
- IGUAL, Laura; SEGUÍ, Santi. **Introduction to Data Science: a python approach to concepts, techniques and applications**. Ebook: Springer, 2017. (Undergraduate Topics in Computer Science).
- GÉRON, Aurélien. **Hands-On Machine Learning with Scikit-Learn and TensorFlow**. Sebastopol: O'reilly Media, 2017

Copyright © 2021 Prof. **Miguel Bozer da Silva**

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).