UNIVERSIDADE FEDERAL DO CEARÁ

DEPARTAMENTO DE COMPUTAÇÃO

LogIA

Aprendizagem de Máquina

César Lincoln Cavalcante Mattos

2025

Agenda

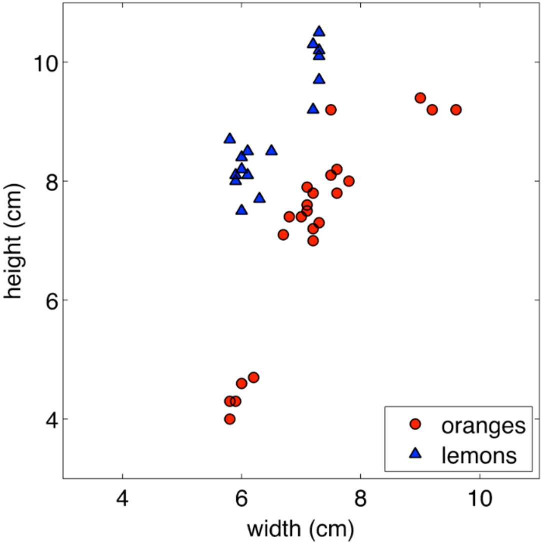
1. Árvores de decisão
2. Tópicos adicionais sobre árvores de decisão
3. Referências

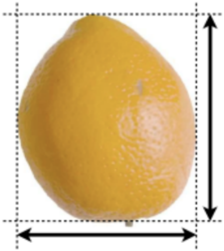
Árvores de decisão

- **Problema**: Como diferenciar laranjas de limões?

Árvores de decisão

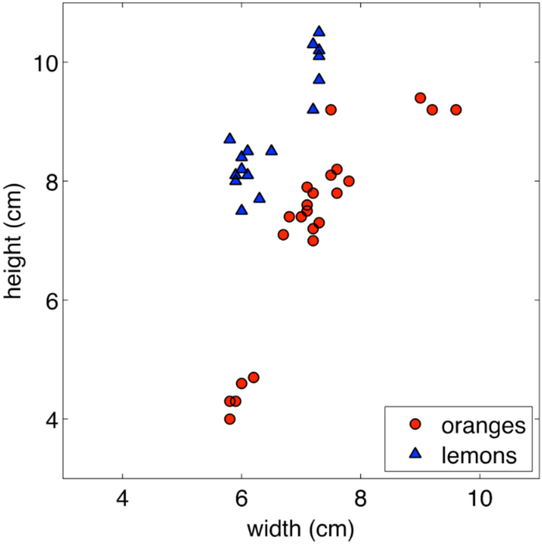
- **Ideia**: Mapeamos largura (*width*) e altura (*height*) das frutas.

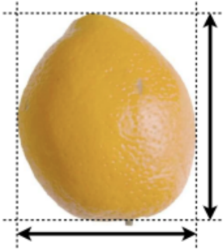




Árvores de decisão

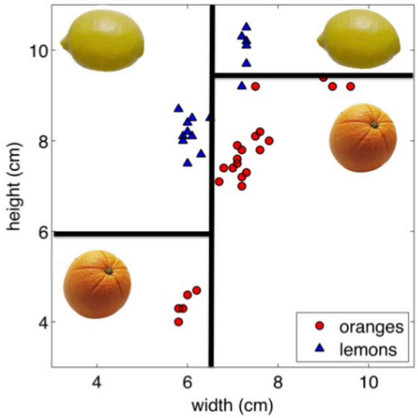
- **Ideia**: Usamos regras lógicas (se-então) para separar as frutas.

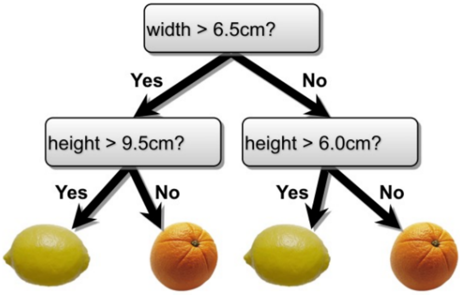




Árvores de decisão

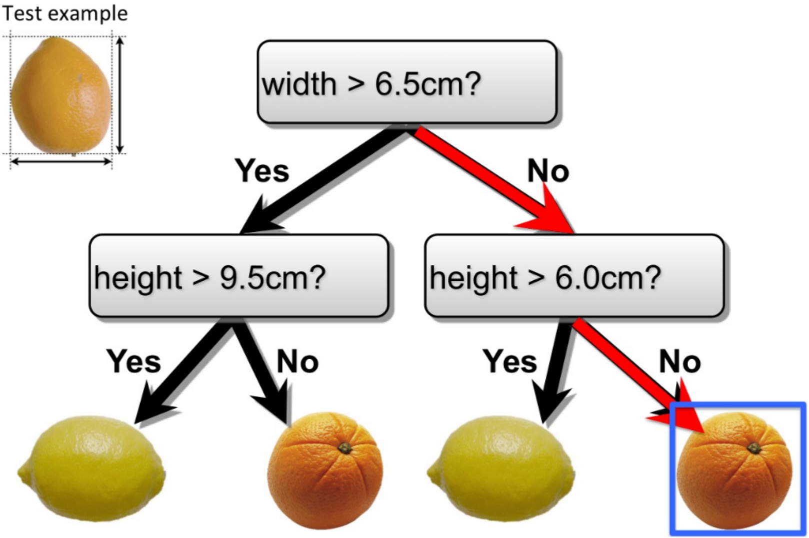
- **Ideia**: Usamos regras lógicas (se-então) para separar as frutas.



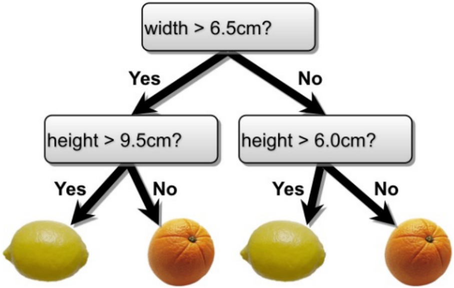


Árvores de decisão

- **Ideia**: Usamos regras lógicas (se-então) para separar as frutas.



Árvores de decisão



- **Nós internos** verificam valores de atributos.
- Cada ramificação é feita de acordo com o **limiar (threshold)** escolhido.
- **Nós terminais (folhas)** estão associados a uma classe específica.

Árvores de decisão

Predições usando árvores de decisão

Dada uma árvore de decisão já existente e um padrão de teste:

1. Inicie no nó mais superior (raiz da árvore).
2. Considere o atributo do nó em questão.
3. Verifique o limiar do nó atual e siga um dos ramos existentes.
4. Caso chegue em um nó terminal (folha), retorne a saída associada. Caso contrário, desça para o próximo nó interno e continue.

Árvores de decisão

- Cada caminho na árvore define uma região (partição) $\mathcal{R}_k$ do espaço de entrada.
- Sejam os padrões $\mathcal{D}_k = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N_k} \in \mathcal{R}_k$ os exemplos de treinamento que alcançam a região $\mathcal{R}_k$.

Árvores de decisão

- Cada caminho na árvore define uma região (partição) $\mathcal{R}_k$ do espaço de entrada.
- Sejam os padrões $\mathcal{D}_k = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N_k} \in \mathcal{R}_k$ os exemplos de treinamento que alcançam a região $\mathcal{R}_k$.

Árvores de decisão para classificação

A saída associada à folha k é a classe mais comum em $\mathcal{D}_k$.

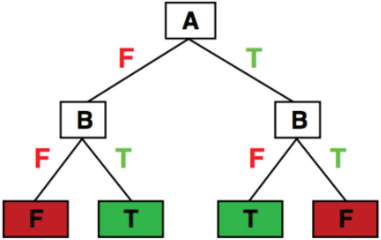
Árvores de decisão para regressão

A saída associada à folha k é a média das saídas contínuas em $\mathcal{D}_k$.

Árvores de decisão

- Para dados (entrada e saída) discretos, árvores de decisão podem expressar qualquer função dos atributos de entrada.

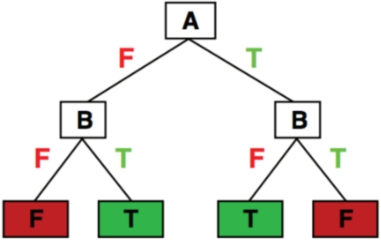
A	B	A xor B
F	F	F
F	T	T
T	F	T
T	T	F



Árvores de decisão

- Para dados (entrada e saída) discretos, árvores de decisão podem expressar qualquer função dos atributos de entrada.

A	B	A xor B
F	F	F
F	T	T
T	F	T
T	T	F

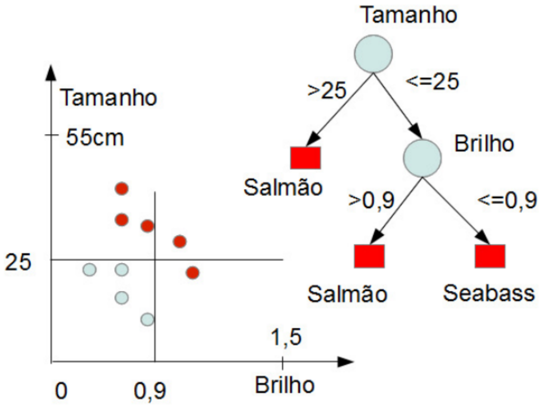


- No caso de dados contínuos, árvores podem aproximas funções com erros arbitrariamente pequenos.

Árvores de decisão

- **Problema**: Classificação de peixe: salmão ou seabass (robalo)?

Brilho	**Tamanho**	**Classe**
1.2	23	Salmão
1.1	30	Salmão
0.9	36	Salmão
0.8	45	Salmão
0.8	38	Salmão
0.9	15	Seabass
0.8	20	Seabass
0.8	25	Seabass
0.7	25	Seabass



Árvores de decisão

- **Problema**: Como obter a árvore de decisão automaticamente a partir dos dados de treinamento?

Árvores de decisão

- **Problema**: Como obter a árvore de decisão automaticamente a partir dos dados de treinamento?
- **Problema**: Construir a menor árvore (mais concisa) é um problema NP completo.

Árvores de decisão

- **Problema**: Como obter a árvore de decisão automaticamente a partir dos dados de treinamento?
- **Problema**: Construir a menor árvore (mais concisa) é um problema NP completo.
- **Ideia**: Seguir uma abordagem heurística gulosa (*greedy*):
  1. Comece de uma árvore vazia;
  2. Encontre o melhor atributo para realizar uma divisão;
  3. Repita recursivamente o passo anterior para o próximo nó até encontrar uma folha.

Árvores de decisão

- **Problema**: Como encontrar o melhor atributo para realizar a divisão?

Árvores de decisão

- **Problema**: Como encontrar o melhor atributo para realizar a divisão?
- **Ideia**: Usar índices de **pureza**.
  - $\rightarrow$ **Pureza máxima**: Somente exemplos de uma mesma classe em uma folha.

Árvores de decisão

- **Problema**: Como encontrar o melhor atributo para realizar a divisão?
- **Ideia**: Usar índices de **pureza**.
  - → **Pureza máxima**: Somente exemplos de uma mesma classe em uma folha.
  - → **Pureza mínima**: Quantidades iguais de todas as classes em uma folha.

Árvores de decisão

- **Problema**: Como encontrar o melhor atributo para realizar a divisão?
- **Ideia**: Usar índices de **pureza**.
  - → **Pureza máxima**: Somente exemplos de uma mesma classe em uma folha.
  - → **Pureza mínima**: Quantidades iguais de todas as classes em uma folha.
  - → Distribuições intermediárias são quantificadas por um índice.
  - → A qualidade da divisão é dada pela média dos índices de pureza das folhas geradas ponderada pelas proporções de padrões.

Árvores de decisão

Entropia (teoria da informação)

- Taxa de informação gerada por uma fonte de dados.
- Dados improváveis fornecem mais informação (mais “surpresa”).

Árvores de decisão

Entropia (teoria da informação)

- Taxa de informação gerada por uma fonte de dados.
- Dados improváveis fornecem mais informação (mais "surpresa").
- Maior a pureza, menor a entropia, sendo quantificada por:

$$H = -\sum_{k} P(C_k) \log_2 P(C_k)$$

- Para $\log_2$, temos **bits**, para $\log$ natural, temos **nats**.

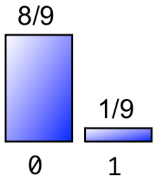
Árvores de decisão

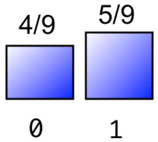
Entropia (teoria da informação)

- Taxa de informação gerada por uma fonte de dados.
- Dados improváveis fornecem mais informação (mais “surpresa”).
- Maior a pureza, menor a entropia, sendo quantificada por:

$$H = -\sum_{k} P(C_k) \log_2 P(C_k)$$

- Para $\log_2$, temos **bits**, para $\log$ natural, temos **nats**.





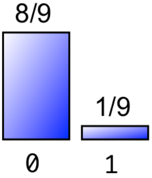
Árvores de decisão

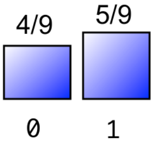
Entropia (teoria da informação)

- Taxa de informação gerada por uma fonte de dados.
- Dados improváveis fornecem mais informação (mais "surpresa").
- Maior a pureza, menor a entropia, sendo quantificada por:

$$H = -\sum_k P(C_k) \log_2 P(C_k)$$

- Para $\log_2$, temos **bits**, para $\log$ natural, temos **nats**.





$$-\frac{8}{9}\log_2\frac{8}{9} - \frac{1}{9}\log_2\frac{1}{9} \approx 0.5 \qquad -\frac{4}{9}\log_2\frac{4}{9} - \frac{5}{9}\log_2\frac{5}{9} \approx 0.99$$

Árvores de decisão

Índice (ou impureza de) Gini

- Frequência em que um exemplo aleatório é incorretamente classificado.
- Pode ser quantificado por:

$$G = \sum_k P(C_k)(1 - P(C_k)) = 1 - \sum_k P(C_k)^2$$

Árvores de decisão

Índice (ou impureza de) Gini

- Frequência em que um exemplo aleatório é incorretamente classificado.
- Pode ser quantificado por:

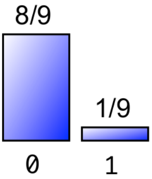
$$G = \sum_k P(C_k)(1 - P(C_k)) = 1 - \sum_k P(C_k)^2$$

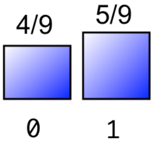
Árvores de decisão

Índice (ou impureza de) Gini

- Frequência em que um exemplo aleatório é incorretamente classificado.
- Pode ser quantificado por:

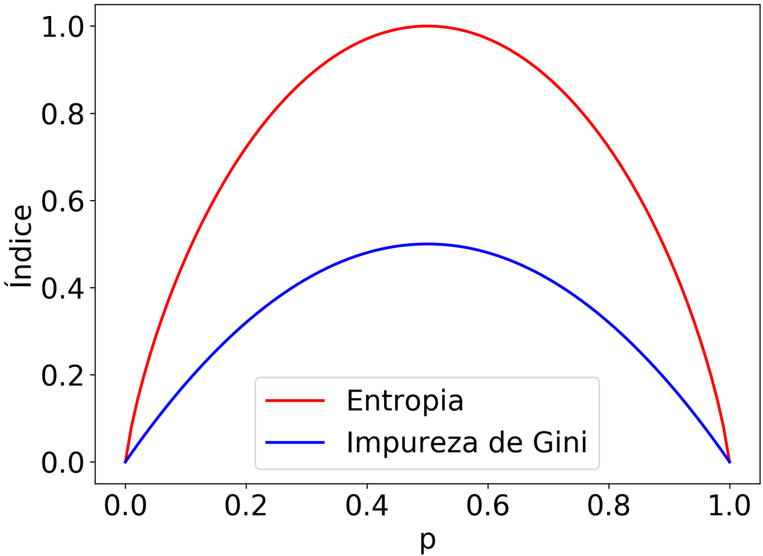
$$G = \sum_k P(C_k)(1 - P(C_k)) = 1 - \sum_k P(C_k)^2$$





$$1 - \left(\frac{8}{9}\right)^2 - \left(\frac{1}{9}\right)^2 \approx 0.2 \qquad 1 - \left(\frac{4}{9}\right)^2 - \left(\frac{5}{9}\right)^2 \approx 0.49$$

Comparação entre entropia e impureza de Gini



Árvores de decisão

- Vamos aplicar o índice Gini na divisão dos exemplos de peixes.

Brilho	Tamanho	Classe
1.2	23	Salmão
1.1	30	Salmão
0.9	36	Salmão
0.8	45	Salmão
0.8	38	Salmão
0.9	15	Seabass
0.8	20	Seabass
0.8	25	Seabass
0.7	25	Seabass

- Gini original (5 Salmão e 4 Seabass):

$$G = 1 - \left(\frac{5}{9}\right)^2 - \left(\frac{4}{9}\right)^2 \approx 0.49$$

Árvores de decisão

- Vamos aplicar o índice Gini na divisão dos exemplos de peixes.

Brilho	**Tamanho**	**Classe**
1.2	23	Salmão
1.1	30	Salmão
0.9	36	Salmão
0.8	45	Salmão
0.8	38	Salmão
0.9	15	Seabass
0.8	20	Seabass
0.8	25	Seabass
0.7	25	Seabass

- Gini original (5 Salmão e 4 Seabass):

$$G = 1 - \left(\frac{5}{9}\right)^2 - \left(\frac{4}{9}\right)^2 \approx 0.49$$

- Escolhendo Brilho > 0.7:
  $\rightarrow$ ✗ 1 Seabass e 0 Salmão:
  $G_1 = 1 - \left(\frac{1}{1}\right)^2 - \left(\frac{0}{1}\right)^2 = 0$

Árvores de decisão

- Vamos aplicar o índice Gini na divisão dos exemplos de peixes.

Brilho	**Tamanho**	**Classe**
1.2	23	Salmão
1.1	30	Salmão
0.9	36	Salmão
0.8	45	Salmão
0.8	38	Salmão
0.9	15	Seabass
0.8	20	Seabass
0.8	25	Seabass
0.7	25	Seabass

- Gini original (5 Salmão e 4 Seabass):

$$G = 1 - \left(\frac{5}{9}\right)^2 - \left(\frac{4}{9}\right)^2 \approx 0.49$$

- Escolhendo Brilho > 0.7:
  - $\rightarrow$ ✗ 1 Seabass e 0 Salmão: $G_1 = 1 - \left(\frac{1}{1}\right)^2 - \left(\frac{0}{1}\right)^2 = 0$
  - $\rightarrow$ ✓ 3 Seabass e 5 Salmão: $G_2 = 1 - \left(\frac{3}{8}\right)^2 - \left(\frac{5}{8}\right)^2 \approx 0.47$

Árvores de decisão

- Vamos aplicar o índice Gini na divisão dos exemplos de peixes.

Brilho	Tamanho	Classe
1.2	23	Salmão
1.1	30	Salmão
0.9	36	Salmão
0.8	45	Salmão
0.8	38	Salmão
0.9	15	Seabass
0.8	20	Seabass
0.8	25	Seabass
0.7	25	Seabass

- Gini original (5 Salmão e 4 Seabass):

$$G = 1 - \left(\frac{5}{9}\right)^2 - \left(\frac{4}{9}\right)^2 \approx 0.49$$

- Escolhendo Brilho > 0.7:
  - $\rightarrow$ ✗ 1 Seabass e 0 Salmão: $G_1 = 1 - \left(\frac{1}{1}\right)^2 - \left(\frac{0}{1}\right)^2 = 0$
  - $\rightarrow$ ✓ 3 Seabass e 5 Salmão: $G_2 = 1 - \left(\frac{3}{8}\right)^2 - \left(\frac{5}{8}\right)^2 \approx 0.47$
  - $\rightarrow$ Gini médio das ramificações: $G_B = \frac{1}{9}G_1 + \frac{8}{9}G_2 \approx 0.42$

Árvores de decisão

- Vamos aplicar o índice Gini na divisão dos exemplos de peixes.

Brilho	Tamanho	Classe
1.2	23	Salmão
1.1	30	Salmão
0.9	36	Salmão
0.8	45	Salmão
0.8	38	Salmão
0.9	15	Seabass
0.8	20	Seabass
0.8	25	Seabass
0.7	25	Seabass

- Gini original (5 Salmão e 4 Seabass):

$$G = 1 - \left(\frac{5}{9}\right)^2 - \left(\frac{4}{9}\right)^2 \approx 0.49$$

Árvores de decisão

- Vamos aplicar o índice Gini na divisão dos exemplos de peixes.

Brilho	Tamanho	Classe
1.2	23	Salmão
1.1	30	Salmão
0.9	36	Salmão
0.8	45	Salmão
0.8	38	Salmão
0.9	15	Seabass
0.8	20	Seabass
0.8	25	Seabass
0.7	25	Seabass

- Gini original (5 Salmão e 4 Seabass):

$$G = 1 - \left(\frac{5}{9}\right)^2 - \left(\frac{4}{9}\right)^2 \approx 0.49$$

- Escolhendo Tamanho > 25:

  $\rightarrow$ ✗ 4 Seabass e 1 Salmão:
  $G_1 = 1 - \left(\frac{4}{5}\right)^2 - \left(\frac{1}{5}\right)^2 \approx 0.32$

Árvores de decisão

- Vamos aplicar o índice Gini na divisão dos exemplos de peixes.

Brilho	**Tamanho**	**Classe**
1.2	23	Salmão
1.1	30	Salmão
0.9	36	Salmão
0.8	45	Salmão
0.8	38	Salmão
0.9	15	Seabass
0.8	20	Seabass
0.8	25	Seabass
0.7	25	Seabass

- Gini original (5 Salmão e 4 Seabass):

$$G = 1 - \left(\frac{5}{9}\right)^2 - \left(\frac{4}{9}\right)^2 \approx 0.49$$

- Escolhendo Tamanho > 25:
  - $\rightarrow$ ✗ 4 Seabass e 1 Salmão: $G_1 = 1 - \left(\frac{4}{5}\right)^2 - \left(\frac{1}{5}\right)^2 \approx 0.32$
  - $\rightarrow$ ✓ 0 Seabass e 4 Salmão: $G_2 = 1 - \left(\frac{0}{4}\right)^2 - \left(\frac{4}{4}\right)^2 = 0$

Árvores de decisão

- Vamos aplicar o índice Gini na divisão dos exemplos de peixes.

Brilho	**Tamanho**	**Classe**
1.2	23	Salmão
1.1	30	Salmão
0.9	36	Salmão
0.8	45	Salmão
0.8	38	Salmão
0.9	15	Seabass
0.8	20	Seabass
0.8	25	Seabass
0.7	25	Seabass

- Gini original (5 Salmão e 4 Seabass):

$$G = 1 - \left(\frac{5}{9}\right)^2 - \left(\frac{4}{9}\right)^2 \approx 0.49$$

- Escolhendo Tamanho > 25:
  - $\rightarrow$ ✗ 4 Seabass e 1 Salmão:
    $G_1 = 1 - \left(\frac{4}{5}\right)^2 - \left(\frac{1}{5}\right)^2 \approx 0.32$
  - $\rightarrow$ ✓ 0 Seabass e 4 Salmão:
    $G_2 = 1 - \left(\frac{0}{4}\right)^2 - \left(\frac{4}{4}\right)^2 = 0$
  - $\rightarrow$ Gini médio das ramificações:
    $G_T = \frac{5}{9}G_1 + \frac{4}{9}G_2 \approx 0.18$

Árvores de decisão

- Vamos aplicar o índice Gini na divisão dos exemplos de peixes.

Brilho	**Tamanho**	**Classe**
1.2	23	Salmão
1.1	30	Salmão
0.9	36	Salmão
0.8	45	Salmão
0.8	38	Salmão
0.9	15	Seabass
0.8	20	Seabass
0.8	25	Seabass
0.7	25	Seabass

- Gini original (5 Salmão e 4 Seabass):

$$G = 1 - \left(\frac{5}{9}\right)^2 - \left(\frac{4}{9}\right)^2 \approx 0.49$$

- Opções de ramificação:

$$\text{Brilho} > 0.7 \rightarrow G_B \approx 0.42$$
$$\text{Tamanho} > 25 \rightarrow G_T \approx 0.18$$

Árvores de decisão

- Vamos aplicar o índice Gini na divisão dos exemplos de peixes.

Brilho	**Tamanho**	**Classe**
1.2	23	Salmão
1.1	30	Salmão
0.9	36	Salmão
0.8	45	Salmão
0.8	38	Salmão
0.9	15	Seabass
0.8	20	Seabass
0.8	25	Seabass
0.7	25	Seabass

- Gini original (5 Salmão e 4 Seabass):

$$G = 1 - \left(\frac{5}{9}\right)^2 - \left(\frac{4}{9}\right)^2 \approx 0.49$$

- Opções de ramificação:

$$\text{Brilho} > 0.7 \rightarrow G_B \approx 0.42$$
$$\text{Tamanho} > 25 \rightarrow G_T \approx 0.18$$

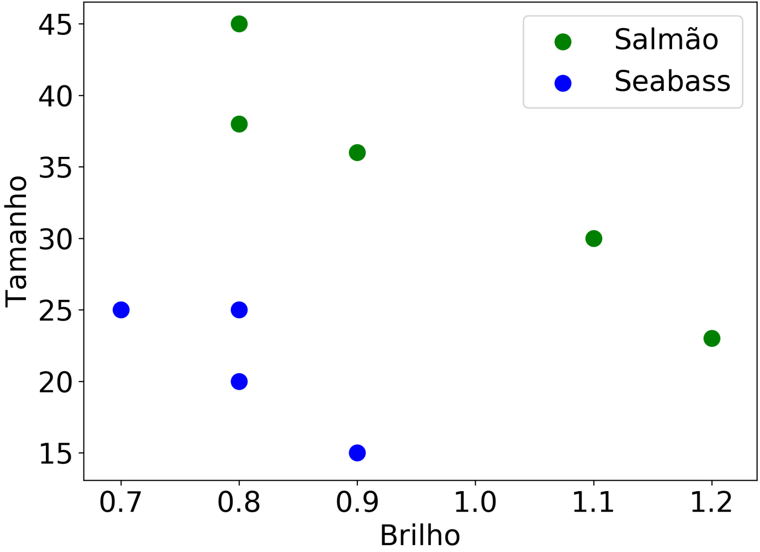
- Escolhemos a opção (Tamanho > 25) que apresenta a maior queda de impureza Gini em relação ao nó pai.

Árvores de decisão

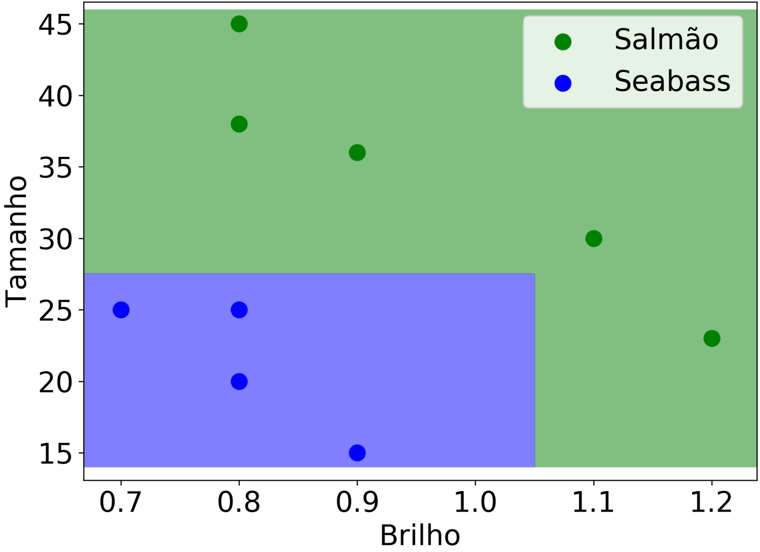
Treinamento guloso (*greedy*) de árvores de decisão

1. Calcule o índice de pureza/impureza do nó atual (nó pai);
2. Crie ramificações a partir de um atributo e um limiar candidatos;
3. Escolha a ramificação com maior queda de impureza (maior pureza) em relação ao nó pai;
4. Para cada nó criado pela ramificação escolhida:
    - Se não houver exemplos de treinamento, retorne a classe mais comum no nó pai.
    - Se todos os exemplos são de uma mesma classe, retorne-a.
    - Caso contrário, retorne ao primeiro passo.

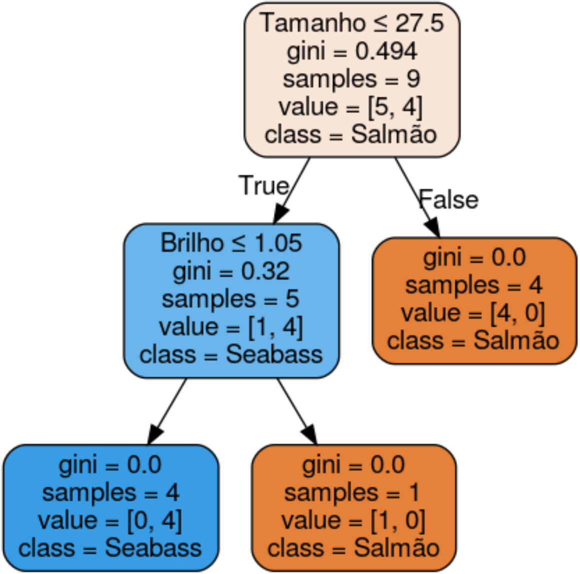
Arvore de decisão para classificação de peixes



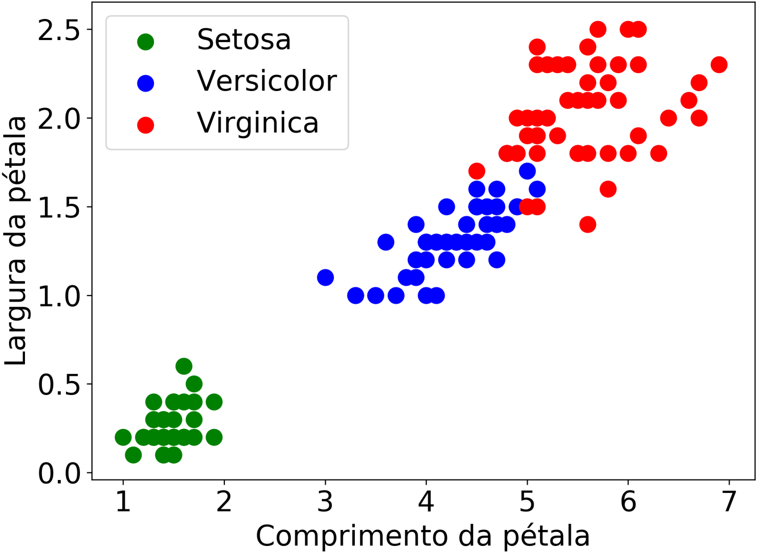
Arvore de decisão para classificação de peixes



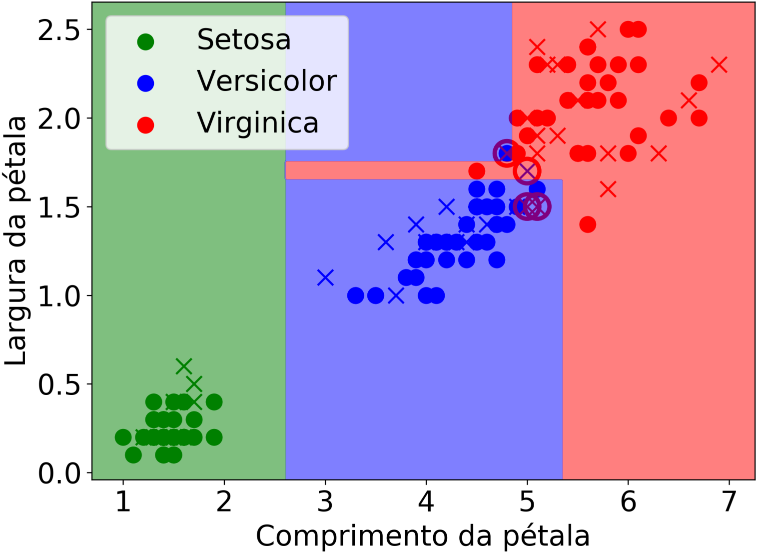
Arvore de decisão para classificação de peixes



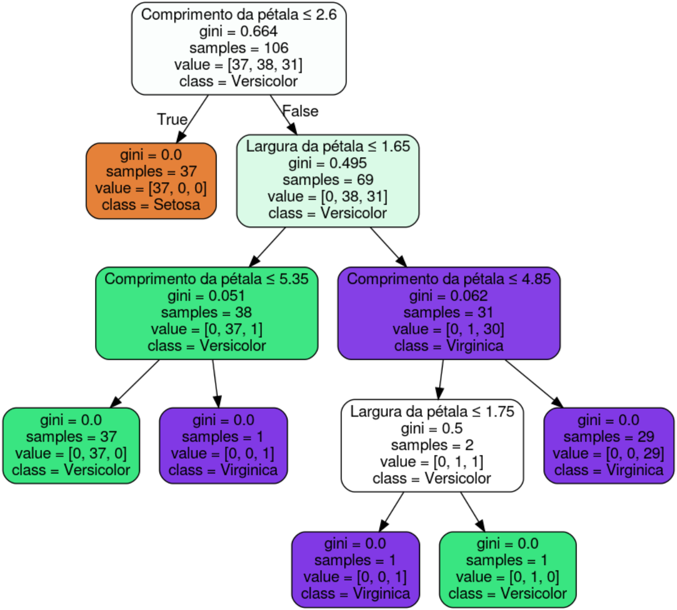
Arvore de decisão para classificação das flores íris



Arvore de decisão para classificação das flores íris



Arvore de decisão para classificação das flores íris



Árvores de decisão

- Podemos formalizar o procedimento detalhado até aqui.
- Seja $\mathcal{D}_n = \{(\boldsymbol{x}_i, y_i) \in N_n\}$ os dados no n-ésimo nó da árvore.

Árvores de decisão

- Podemos formalizar o procedimento detalhado até aqui.
- Seja $\mathcal{D}_n = \{(\boldsymbol{x}_i, y_i) \in N_n\}$ os dados no n-ésimo nó da árvore.
- Seja $\mathcal{D}_n^L(d, t) = \{(\boldsymbol{x}_i, y_i) \in N_n : x_{id} \leq t\}$ e $\mathcal{D}_n^R(d, t) = \{(\boldsymbol{x}_i, y_i) \in N_n : x_{id} > t\}$ uma bipartição de $\mathcal{D}_n$.
    - No caso de atributos categóricos, teríamos $\mathcal{D}_n^L(d, t) = \{(\boldsymbol{x}_i, y_i) \in N_n : x_{id} = t\}$ e $\mathcal{D}_n^R(d, t) = \{(\boldsymbol{x}_i, y_i) \in N_n : x_{id} \neq t\}$.

Árvores de decisão

- Podemos formalizar o procedimento detalhado até aqui.
- Seja $\mathcal{D}_n = \{(\boldsymbol{x}_i, y_i) \in N_n\}$ os dados no n-ésimo nó da árvore.
- Seja $\mathcal{D}_n^L(d,t) = \{(\boldsymbol{x}_i, y_i) \in N_n : x_{id} \leq t\}$ e $\mathcal{D}_n^R(d,t) = \{(\boldsymbol{x}_i, y_i) \in N_n : x_{id} > t\}$ uma bipartição de $\mathcal{D}_n$.
  - No caso de atributos categóricos, teríamos $\mathcal{D}_n^L(d,t) = \{(\boldsymbol{x}_i, y_i) \in N_n : x_{id} = t\}$ e $\mathcal{D}_n^R(d,t) = \{(\boldsymbol{x}_i, y_i) \in N_n : x_{id} \neq t\}$.
- A escolha do melhor atributo d_n e do melhor limiar t_n será dado pela otimização abaixo:

$$(d_n, t_n) = \arg \min_{d \in \{1,\ldots,D\}} \min_{t \in \mathcal{T}_d} \frac{|\mathcal{D}_n^L(d,t)|}{|\mathcal{D}_n|} c(\mathcal{D}_n^L(d,t)) + \frac{|\mathcal{D}_n^R(d,t)|}{|\mathcal{D}_n|} c(\mathcal{D}_n^R(d,t)),$$

em que $c(\cdot)$ é a função custo (e.g. índice Gini, entropia ou MSE) e $\mathcal{T}_d$ é o conjunto de possíveis limiares para o atributo d.

Árvores de decisão

- **Importante**: Sempre é possível criar uma árvore em que todos os exemplos de treinamento são perfeitamente separados, desconsiderando o ruído.

Árvores de decisão

- **Importante**: Sempre é possível criar uma árvore em que todos os exemplos de treinamento são perfeitamente separados, desconsiderando o ruído.
- **Questão**: Isso prejudica a generalização do modelo?

Árvores de decisão

- **Importante**: Sempre é possível criar uma árvore em que todos os exemplos de treinamento são perfeitamente separados, desconsiderando o ruído.
- **Questão**: Isso prejudica a generalização do modelo? Sim (overfitting)!

Árvores de decisão

- **Importante**: Sempre é possível criar uma árvore em que todos os exemplos de treinamento são perfeitamente separados, desconsiderando o ruído.
- **Questão**: Isso prejudica a generalização do modelo? Sim (overfitting)!
- **Ideias**:
  - $\rightarrow$ Evitar árvores muito grandes fixando uma variação mínima de pureza para executar uma ramificação.
  - $\rightarrow$ Podar a árvore gerada (remover e/ou unir nós) usando um conjunto de validação.

Árvores de decisão

Algoritmos para treinamento de árvores de decisão

- **ID3 (Iterative Dichotomizer)**: Um dos primeiros e mais simples algoritmos de árvore de decisão. Normalmente usa a entropia para escolher novas ramificações.
- **C4.5**: Versão mais avançada do algoritmo ID3, com suporte a poda e dados discretos, contínuos, faltantes.
- **CART (Classification And Regression Tree)**: Similar ao algoritmo C4.5. Normalmente usa a impureza de Gini para escolher novas ramificações.

Árvores de decisão

Vantagens

- Facilmente interpretáveis, pois geram regras de decisão.
- São escaláveis.
- Seleção automática de atributos importantes.
- Podem lidar com dados faltosos.

Árvores de decisão

Vantagens

- Facilmente interpretáveis, pois geram regras de decisão.
- São escaláveis.
- Seleção automática de atributos importantes.
- Podem lidar com dados faltosos.

Desvantagens

- Tendência ao overfitting.
- Pequenas variações no conjunto de treinamento resultam em árvores diferentes.

Árvores de decisão como bases adaptativas

- De uma maneira geral, a saída de um modelo de árvore de decisão pode ser escrita como:

$$f(\boldsymbol{x}) = \sum_{k=1}^{K} w_k \mathbb{I}(\boldsymbol{x} \in \mathcal{R}_k), \quad w_k = \frac{\sum_i y_i \mathbb{I}(\boldsymbol{x}_i \in \mathcal{R}_k)}{\sum_i \mathbb{I}(\boldsymbol{x}_i \in \mathcal{R}_k)},$$

  em que $\mathcal{R}_k$ denota a k-ésima partição, w_k indica a "resposta média" dos exemplos da partição $\mathcal{R}_k$.

Árvores de decisão como bases adaptativas

- De uma maneira geral, a saída de um modelo de árvore de decisão pode ser escrita como:

$$f(\boldsymbol{x}) = \sum_{k=1}^{K} w_k \mathbb{I}(\boldsymbol{x} \in \mathcal{R}_k), \quad w_k = \frac{\sum_i y_i \mathbb{I}(\boldsymbol{x}_i \in \mathcal{R}_k)}{\sum_i \mathbb{I}(\boldsymbol{x}_i \in \mathcal{R}_k)},$$

  em que $\mathcal{R}_k$ denota a k-ésima partição, w_k indica a "resposta média" dos exemplos da partição $\mathcal{R}_k$.
- Considerando uma função $\phi(\cdot)$ responsável por criar as partições:

$$f(\boldsymbol{x}) = \sum_{k=1}^{K} w_k \phi(\boldsymbol{x}, \boldsymbol{\theta}_k) = \boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{\Theta}), \quad \boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_K],$$

  em que $\boldsymbol{\theta}_k$ reúne os parâmetros (atributos e limiares) que definem a partição $\mathcal{R}_k$.

Árvores de decisão como bases adaptativas

- De uma maneira geral, a saída de um modelo de árvore de decisão pode ser escrita como:

$$f(\boldsymbol{x}) = \sum_{k=1}^{K} w_k \mathbb{I}(\boldsymbol{x} \in \mathcal{R}_k), \quad w_k = \frac{\sum_i y_i \mathbb{I}(\boldsymbol{x}_i \in \mathcal{R}_k)}{\sum_i \mathbb{I}(\boldsymbol{x}_i \in \mathcal{R}_k)},$$

  em que $\mathcal{R}_k$ denota a k-ésima partição, w_k indica a "resposta média" dos exemplos da partição $\mathcal{R}_k$.
- Considerando uma função $\phi(\cdot)$ responsável por criar as partições:

$$f(\boldsymbol{x}) = \sum_{k=1}^{K} w_k \phi(\boldsymbol{x}, \boldsymbol{\theta}_k) = \boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{\Theta}), \quad \boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K],$$

  em que $\boldsymbol{\theta}_k$ reúne os parâmetros (atributos e limiares) que definem a partição $\mathcal{R}_k$.
- Portanto, uma árvore de decisão pode ser entendida como um **modelo de bases adaptativas**.

Agenda

Tópicos adicionais sobre árvores de decisão

- Poda (prunning) de árvores.
- Modelos de mistura em árvores.
- Árvores aditivas: Bayesian additive regression trees (BART).
- Bagging e boosting de árvores de decisão (**ainda veremos!**)

Agenda

1. Árvores de decisão
2. Tópicos adicionais sobre árvores de decisão
3. Referências

Referências bibliográficas

- **Caps. 5 e 16** - MURPHY, Kevin P. **Machine learning: a probabilistic perspective**, 2012.
- **Cap. 18** - MURPHY, Kevin P. **Probabilistic Machine Learning: An Introduction**, 2021.
- **Cap. 14** - BISHOP, C. **Pattern recognition and machine learning**, 2006.