Introdução a Ciências de Dados

Aula 7: Classificação: árvores, ensembles

Francisco A. Rodrigues
ICMC/USP
francisco@icmc.usp.br

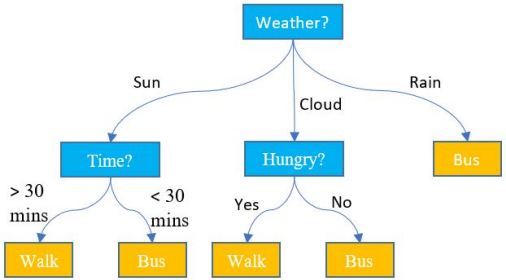
Aula 7: Classificação

- Árvores de Decisão
- Ensemble methods

Árvores de decisão

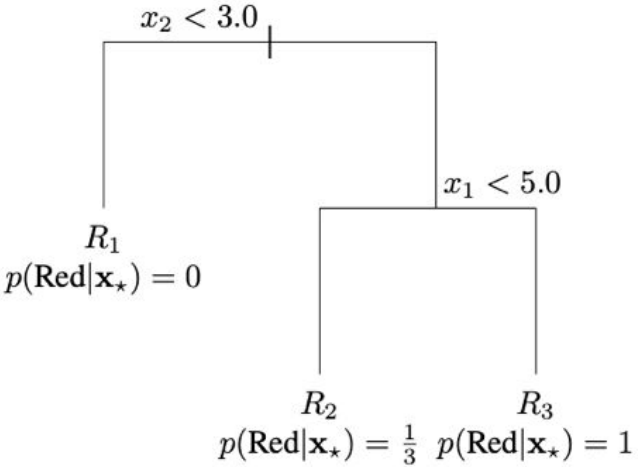
Árvore de decisão

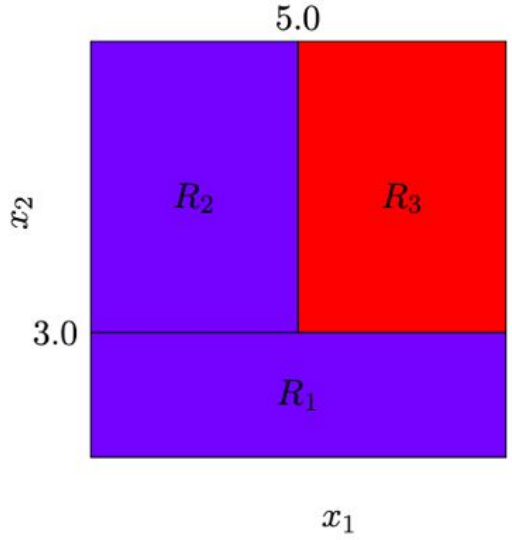
- Métodos baseados em árvores dividem o espaço de atributos em regiões.
- Em cada região, é estimada a função **p(y|x)**.
- As regras para dividir o espaço podem ser resumidas em uma árvore, denominada árvore de decisão.
- Árvores podem ser usadas em problemas de classificação e regressão.



Árvore de decisão

Exemplo:



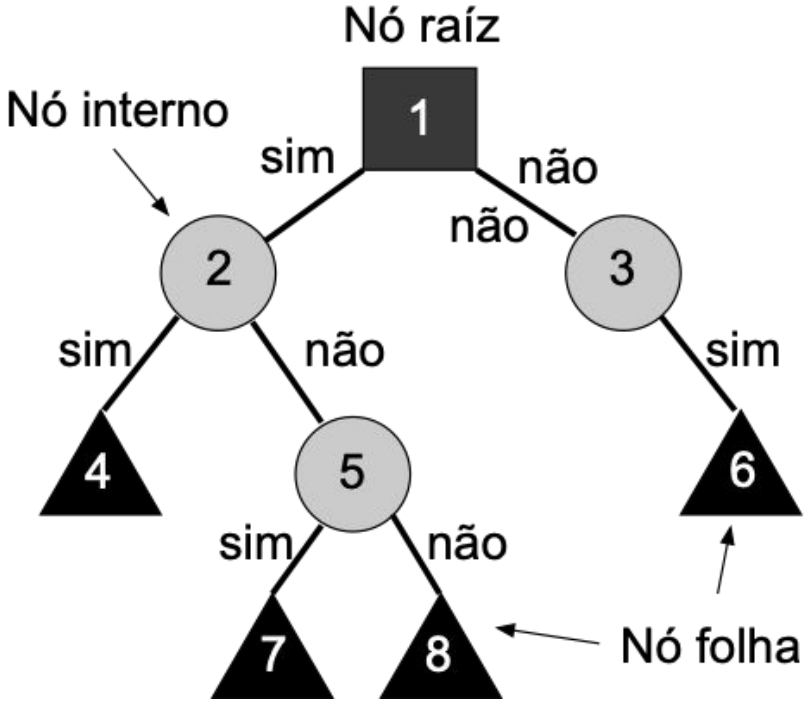


Cada nó corresponde a uma região no espaço de entradas.

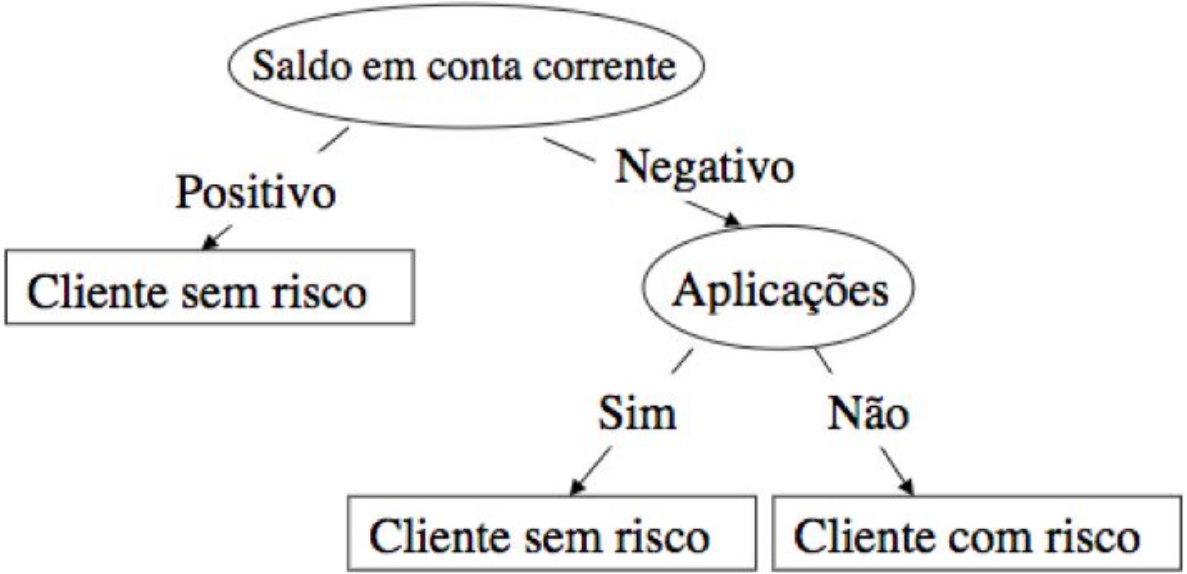
```
if x_2 < 3.0 then
    return p(Red|x)=0
else
    if x_1 < 5.0 then
       return p(Red|x)=1/3
    else
        return p(Red|x)=1
    end
end
```

Árvore de decisão

- **Estrutura da árvore:**
  - **Nós folha (resposta):** contém uma previsão.
  - **Nós internos:** Contém um teste de atributo.
  - Para cada possível valor do atributo, existe um ramo para outra sub-árvore.



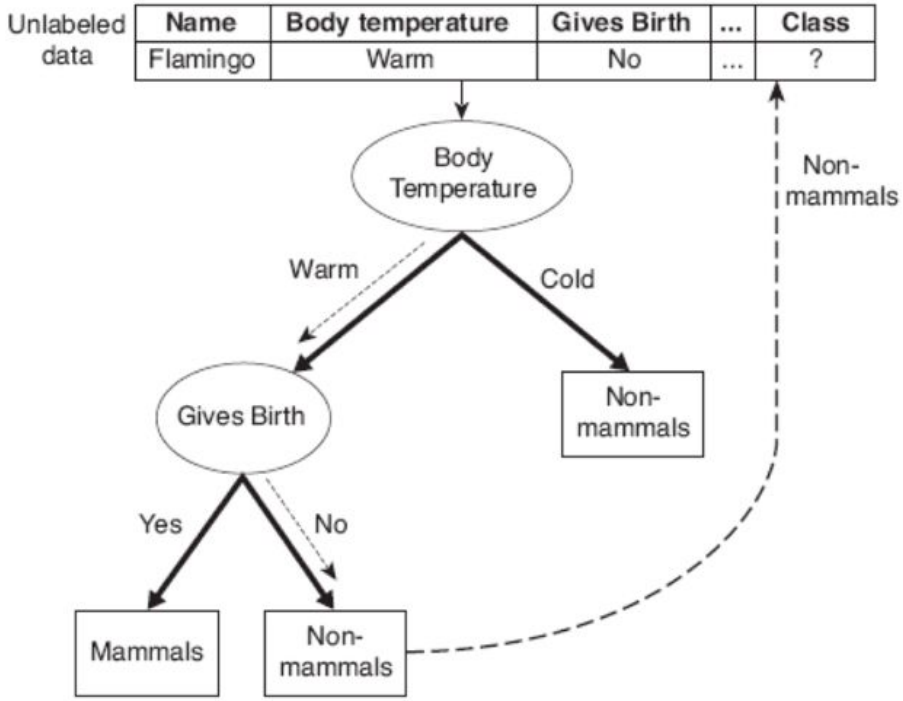
Árvore de decisão



Árvore de decisão

Classificação:

- Depois de construída a árvore, a classificação é feita “navegando" pela árvore até chegar a um nó folha.



Árvore de decisão

- Matematicamente, a árvore modela as probabilidades **p(k|x)** como uma constante $\mathbf{c_{mk}}$ em cada região $\mathbf{R_m}$, para k=1,2,..., K.

$$p(y = k \mid \mathbf{x}) = \sum_{m=1}^{M} c_{mk} I\{\mathbf{x} \in R_m\}, \qquad \sum_{k=1}^{K} c_{mk} = 1$$

- O objetivo é encontrar uma árvore que gere os dados observados, no conjunto de treinamento, com maior probabilidade possível.
- Esse método considera a maximização da verossimilhança.

Árvore de decisão

- A maximização da verossimilhança:

$$-\log \ell(T) = -\log p(\mathbf{y} \mid \mathbf{X}, T)$$

- A maximização dessa função é um problema combinatorial e computacionalmente proibitivo.

- **Solução**: usar um algoritmo de recursão binária.

Árvore de decisão

Como construir a árvore?

- Considerar os dados de treinamento e tentar prevê-los com maior acurácia possível.
- Devemos definir uma métrica para encontrar os ramos e escolher os nós da árvore.
- Há diferente métricas para realizar essa tarefa e o resultado pode depender dessa escolha.

Árvore de decisão

Como construir a árvore?

- Existem vários algoritmos
  - Algoritmo de Hunt
  - CART
  - ID3, C4.5
  - SLIQ, SPRINT

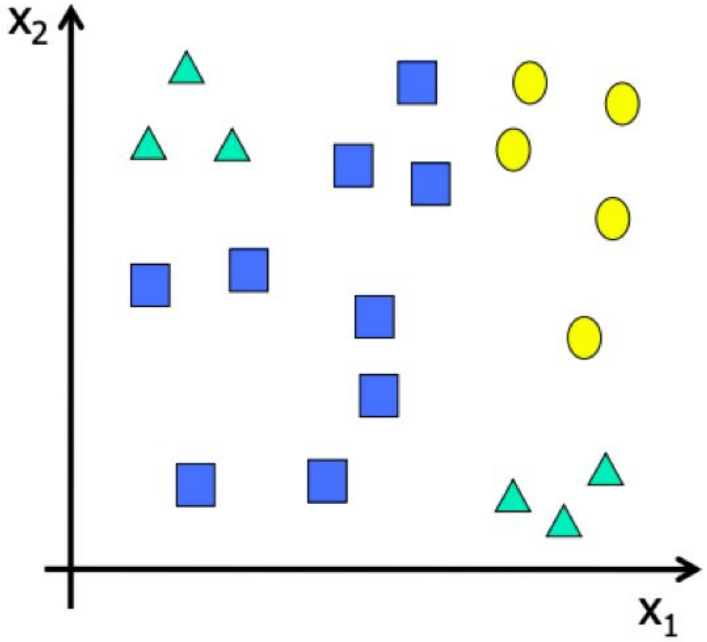
Árvore de decisão

Como construir a árvore?

- **Algoritmo de Hunt:**
  - Seja **Xt** o conjunto de objetos de treinamento que atingem o nó **t**.
  - Se todos os objetos de **Xt** ∈ a mesma classe **yt**
    - Então **t** é um nó folha rotulado como **yt**
  - Caso contrário, se os objetos de Xt ∈ a mais de uma classe
    - Então selecionar um atributo preditivo teste para dividir **Xt**
    - Dividir **Xt** em subconjuntos utilizando esse atributo
  - Aplicar algoritmo a cada subconjunto gerado

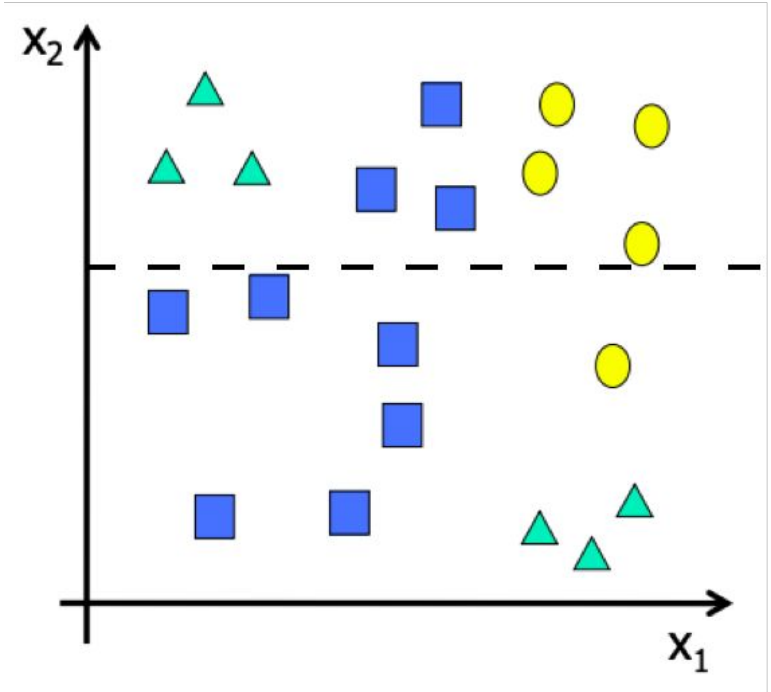
Árvore de decisão

Como construir a árvore?



Árvore de decisão

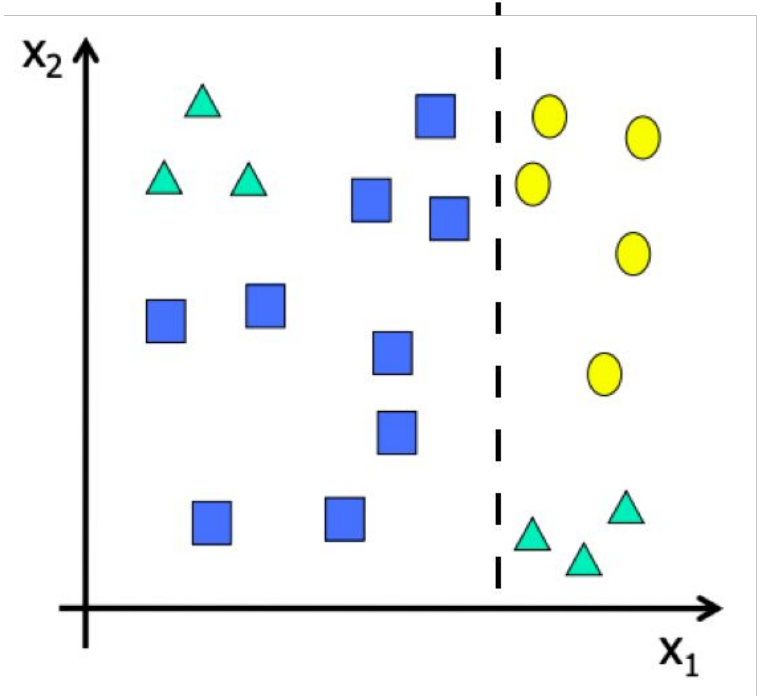
Como construir a árvore?



?

Árvore de decisão

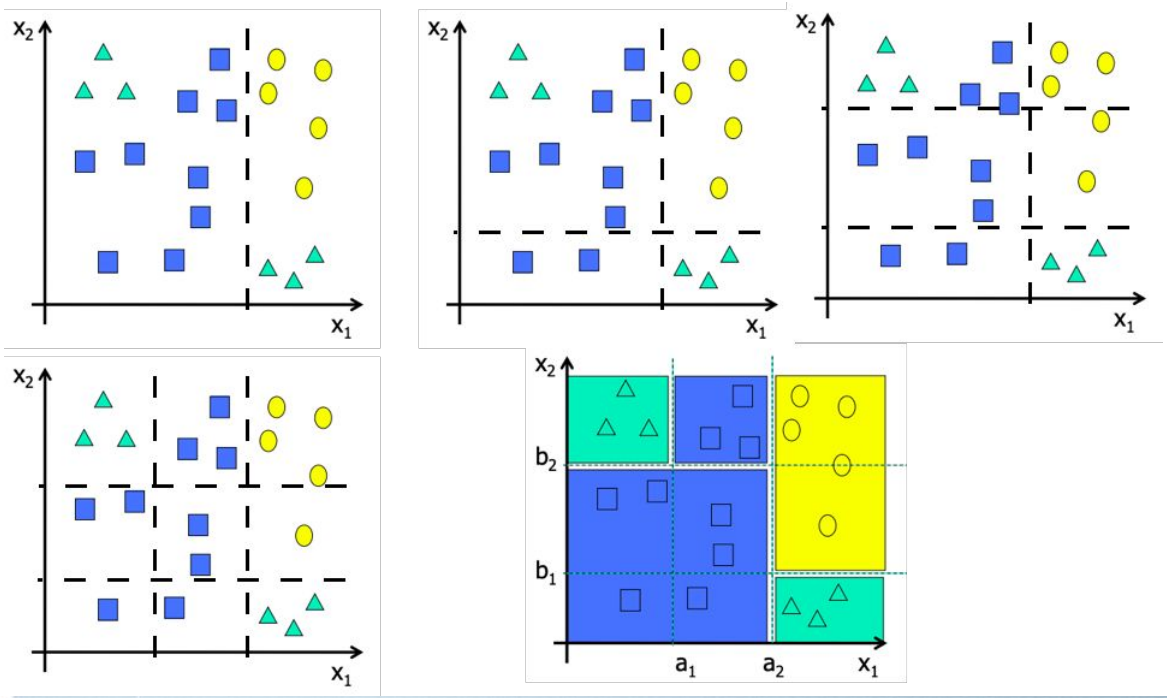
Como construir a árvore?



?

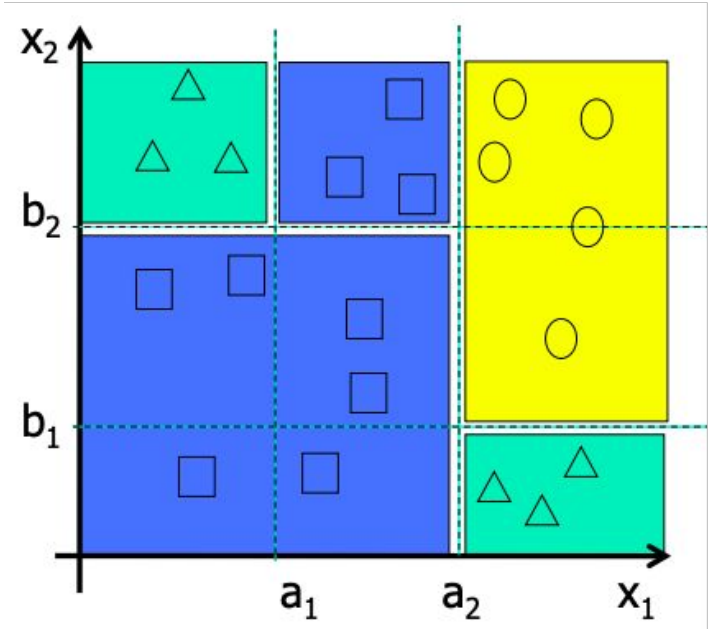
Árvore de decisão

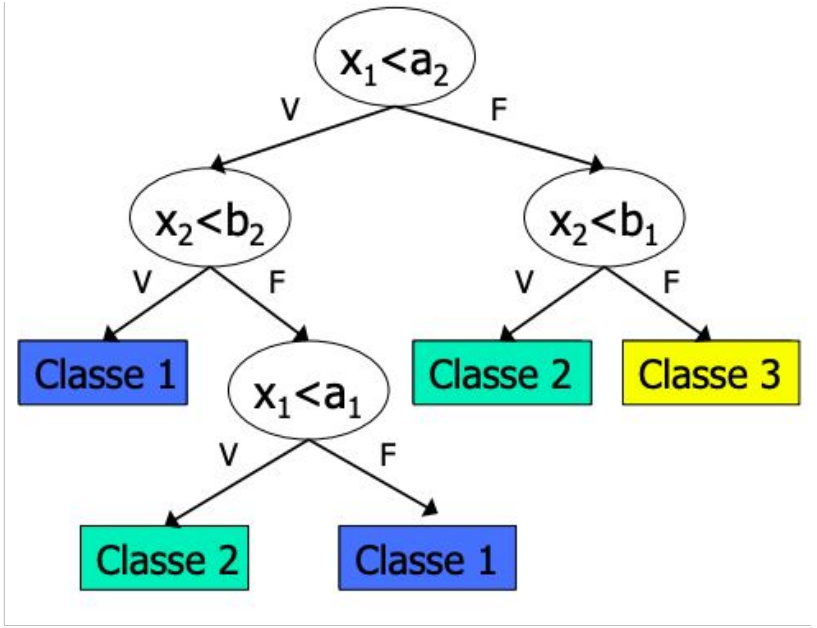
Como construir a árvore?



Árvore de decisão

Como construir a árvore?





Árvores de decisão

Como construir a árvore?

- Construir uma árvore minimal (número mínimo de nós) condizente com conjunto de dados é problema NP-completo.
- Algoritmos usualmente usam heurísticas que olham um passo à frente.
- Estratégia gulosa.
  - Suscetível a encontrar ótimo local.
  - Mas permite construção de AD em tempo linear.

Árvores de decisão

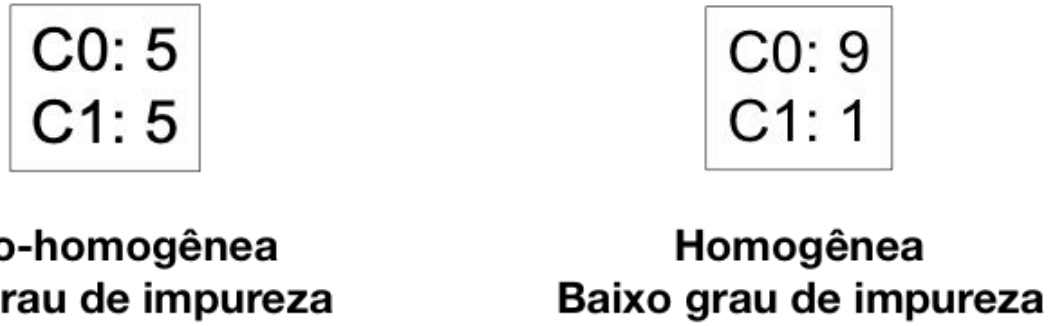
Como construir a árvore?

- Decisões importantes
  - Como dividir os objetos?
  - Medida para avaliar qualidade de atributo escolhido.
  - Quando parar de dividir os objetos.

Árvores de decisão

Como construir a árvore?

- **Estratégia gulosa:**
  - Precisamos de uma medida para selecionarmos entre duas (ou mais) divisões.
  - Medida de pureza: mede o quão homogênea é a distribuição dos elementos das classes com relação à um atributo:



Árvores de decisão

Como construir a árvore?

- Medidas de impureza:
  - Índice de Gini
  - Entropia
  - Erro na classificação

Árvores de decisão

Índice de Gini:

$$\text{Gini}(t) = 1 - \sum_{j=0}^{c-1} [p(j \mid t)]^2$$

- **p(j|t)** representa a frequência de elementos da classe **j** no nó **t**.

C1	**0**
C2	**6**

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1

Gini = 1 – P(C1)² – P(C2)² = 1 – 0 – 1 = 0

C1	**2**
C2	**4**

P(C1) = 2/6 P(C2) = 4/6

Gini = 1 – (2/6)² – (4/6)² = 0.444

Árvores de decisão

Entropia de Shannon:

$$H(t) = -\sum_{j=0}^{c-1} p(j \mid t) \log_2 p(j \mid t)$$

- **p(j|t)** representa a frequência de elementos da classe j no nó t.

C1	**0**
C2	**6**

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1

H = -(0/6*log$_2$(0/6) + 1*log$_2$(1)) = 0

C1	**2**
C2	**4**

P(C1) = 2/6 P(C2) = 4/6

Gini =-(2/6*log$_2$(2/6) + 4/6*log$_2$(4/6)) = 0.92

Árvores de decisão

Erro na classificação:

$$\text{Erro}(t) = 1 - \max_i [p(i \mid t)]$$

- **p(j|t)** representa a frequência de elementos da classe j no nó t.

C1	**0**
C2	**6**

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1

Erro = 1 - 1 = 0

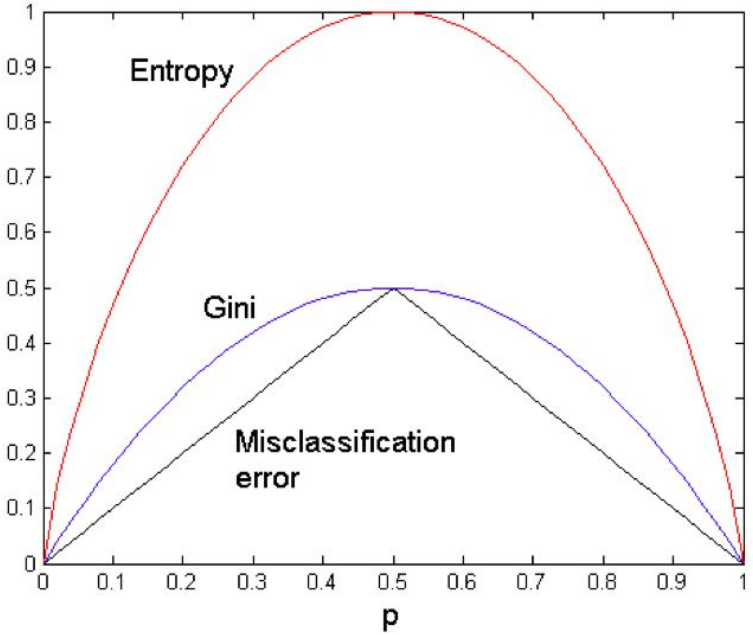
C1	**2**
C2	**4**

P(C1) = 2/6 P(C2) = 4/6

Erro = 1 - 4/6 = 2/6 = 0.33

Árvores de decisão

Erro na classificação:



Árvores de decisão

- Mas como decidir se paramos uma divisão ou continuamos a crescer a árvore?
- Precisamos decidir quando aceitamos um novo ramo ou inserimos um nó folha.
- Para isso, comparamos o grau de impureza do nó pai (antes da divisão) com o do nó filho (após a divisão).
- Quanto maior essa diferença, melhor é a divisão.

Árvores de decisão

A medida de ganho:

$$\Delta = I(\text{pai}) - \sum_{i=1}^{k} \frac{N(v_i)}{N} I(v_i),$$

onde

- I(.) é uma medida de impureza de um dado nó,
- N é o número total de elementos no nó pai
- k é o número de atributos
- N(vi) é o número de elementos associado ao nó filho vi.
- **Quanto maior o ganho, melhor é a divisão.**

Árvores de decisão

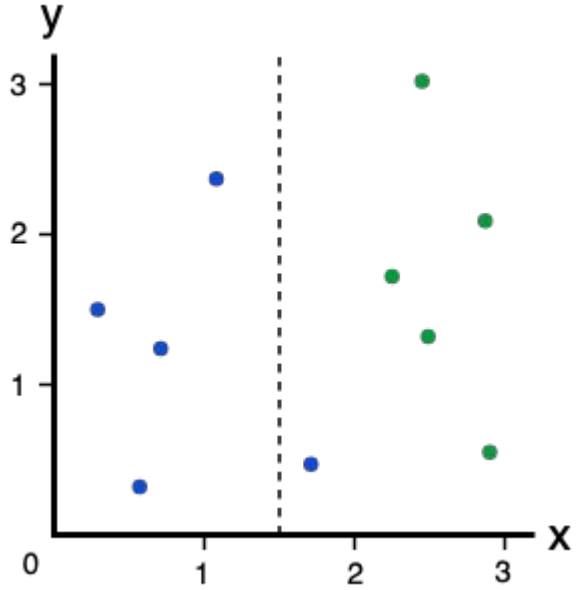
Ganho de informação:

$$\Delta_{\text{info}} = H(\text{pai}) - \sum_{i=1}^{k} \frac{N(v_i)}{N} H(i)$$

- onde
  - I(.) é uma medida de impureza de um dado nó,
  - N é o número total de elementos no nó pai
  - k é o número de atributos
  - N(vj) é o número de elementos associado ao nó filho vj.
- Escolha a partição que resultar no maior ganho.
- Usado nos algoritmos ID3 e C4.5
- Tende a gerar muitas partições, pequenas, mas puras.

Árvores de decisão

Ganho de informação:



- Antes da divisão, temos 5 pontos azuis e 5 verdes:
- H(pai) = $-(0.5\log_2 0.5+0.5\log_2 0.5)=1$

$$\Delta_{\text{info}} = H(\text{pai}) - \sum_{i=1}^{k} \frac{N(v_i)}{N} H(i)$$

$$-\left(\frac{1}{6}\log_2\left(\frac{1}{6}\right) + \frac{5}{6}\log_2\left(\frac{5}{6}\right)\right)$$

$$\Delta_{\text{info}} = 1 - (0{,}4 \times 0 + 0{,}6 \times 0{,}65) = 0{,}61$$

Entropia é nula, pois todos os círculos são azuis.

Árvores de decisão

Ganho de informação:

- Tende a gerar muitas partições, pequenas, mas puras.

Razão de ganho (Gain ratio):

$$G = \frac{\Delta_{info}}{-\sum_{i=1}^{k} \frac{N(v_i)}{N} \log_2 \frac{N(v_i)}{N}}$$

- Usado para solucionar a limitação do ganho de informação.
- Usado no algoritmo C4.5.

Árvores de decisão

Índice de Gini:

- Quando um nó j é dividido em k partições (filhos), a qualidade dessa divisão é calculada por:

$$\text{Gini}_{split} = \sum_{i=1}^{k} \frac{N(v_i)}{N} \text{Gini}(i)$$

- N(vi) = número de entradas no nó filho.
- N = número de entradas no nó pai.
- Se houver redução no índice, aceitamos a divisão.
- Usado nos algoritmos CART, SLIQ, SPRINT.

Árvores de decisão

Propriedades

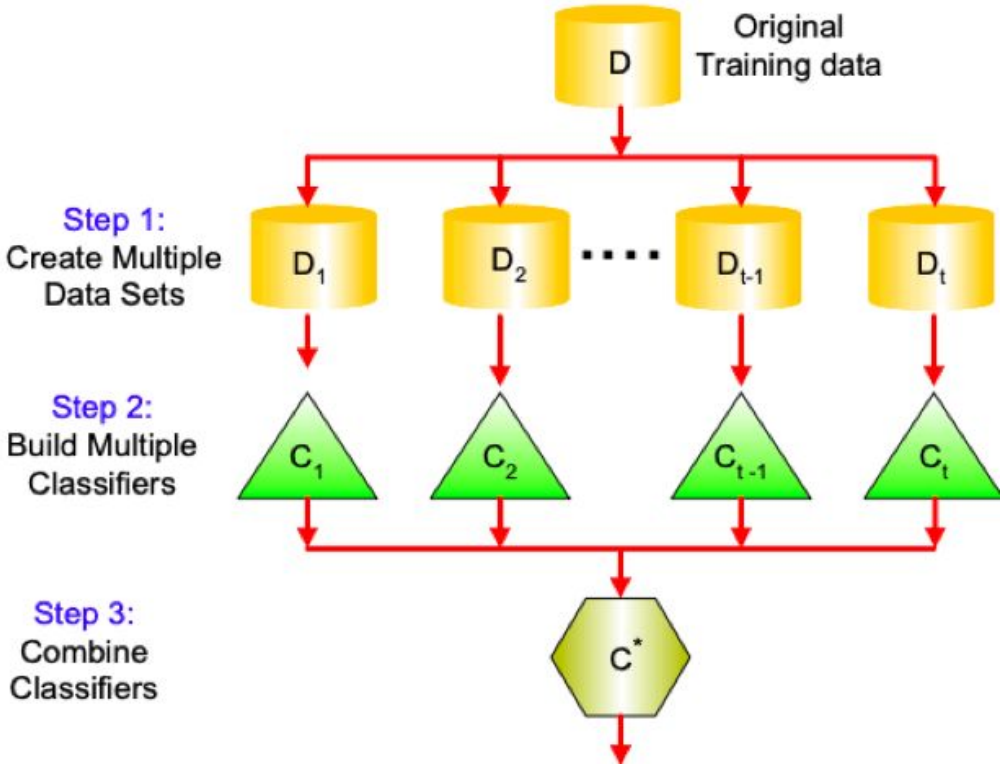
- Simples entendimento e interpretação.
- Não requer normalização dos dados.
- Pode ser usada com dados numéricos e categóricos ao mesmo tempo.
- É um modelo “caixa branca”.
- É um método robusto a outliers.
- Pode ser usado em grandes bancos de dados.
- É um método não-paramétrico.

Ensemble methods

Ensemble methods

- Um classificador ensemble (comitê de learners, mistura de especialistas ou sistema de classificadores múltiplo) é um sistema que combina classificadores treinados individualmente para se chegar a uma solução melhor do que aquela obtida usando classificadores individuais.

Ensemble methods



Ensemble methods

Métodos:

- Bagging
- Boosting
- Random forest

Ensemble methods

Bagging:

- **Pergunta**: Dado um modelo com baixo viés e alta variância, é possível reduzir a variância e preservar o viés?
- Seja $z_1, z_2, \ldots, z_B$ uma coleção de variáveis aleatórias identicamente distribuídas, mas possivelmente dependentes, com esperança $E[z_i]=\mu$ e $Var[zi]=\sigma^2$
- Para essa coleção, pode-se mostrar:

$$\mathbb{E}\left[\frac{1}{B}\sum_{b=1}^{B} z_i\right] = \mu,$$

$$\mathrm{Var}\left[\frac{1}{B}\sum_{b=1}^{B} z_b\right] = \frac{1-\rho}{B}\sigma^2 + \rho\sigma^2.$$

Ensemble methods

Bagging:

- Ou seja, quando temos uma amostra de uma distribuição, a média se mantém, mas a variância diminui se a correlação for menor do que um.

$$\mathbb{E}\left[\frac{1}{B}\sum_{b=1}^{B} z_i\right] = \mu,$$

$$\mathrm{Var}\left[\frac{1}{B}\sum_{b=1}^{B} z_b\right] = \frac{1-\rho}{B}\sigma^2 + \rho\sigma^2.$$

Ensemble methods

Bagging:

- A variância de predições obtidas de modelos identicamente distribuídos, cada um com baixo viés, pode ser reduzida se calcularmos a média dessas previsões.
- **Problema**: como gerar mais dados?

Ensemble methods

Bagging:

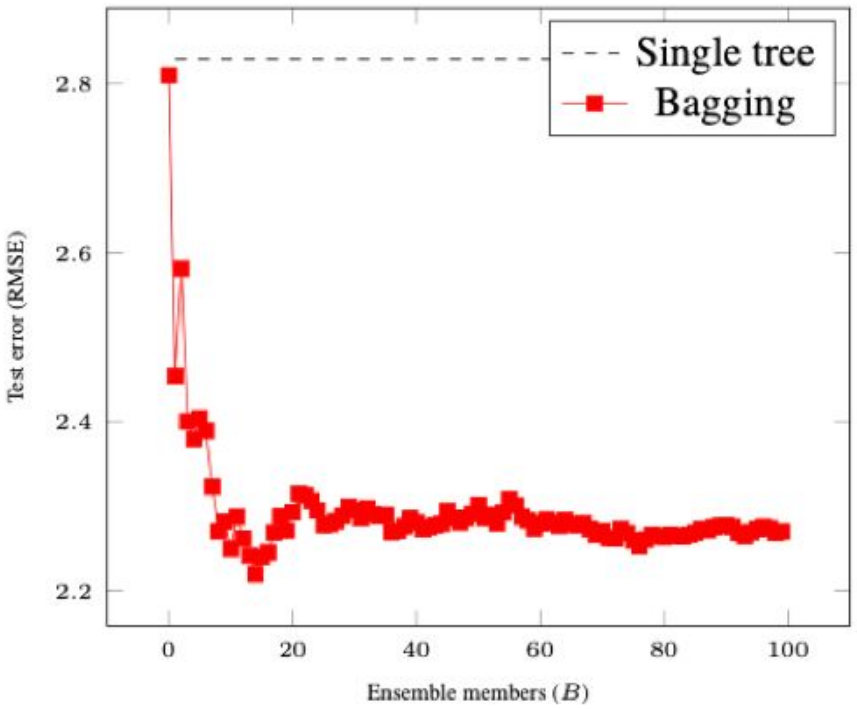
- **Bagging: Bootstrap Aggregating**
- **Ideia**: Selecionar **B** amostras com reposição a partir do conjunto de treinamento. Cada amostra tem o mesmo tamanho que o conjunto de treinamento.
- Usar cada amostra para treinar um classificador.
- **Predição**:

$$\widehat{y}_{\star}^{\text{avg}} = \frac{1}{B}\sum_{b=1}^{B}\widehat{y}_{\star}^{b}.$$

Ensemble methods

Bagging:

- O método de classificação pode ser qualquer modelo preditivo.
- **Exemplo**: para uma árvore de decisão:



Ensemble methods

Boosting:

- É um método iterativo usado para mudar a distribuição dos dados de forma adaptativa .
- Isto é feito modificando o conjunto de treinamento de modo a colocar maior ênfase nas observações que o modelo não conseguiu acertar.
- A ideia é transformar um conjunto de modelos pouco precisos em um método preciso.

Ensemble methods

Boosting:

- **Algoritmo:**
  - Inicialmente, cada observação possui o mesmo peso.
  - A seguir, o algoritmo ajusta os pesos de forma iterativa.
  - Observações classificadas erroneamente recebem maior peso.
  - Quanto maior o peso, maior é a probabilidade de ser escolhido nos passos posteriores.

Original Data	1	2	3	4	5	6	7	8	9	10
Boosting (Round 1)	7	3	2	8	7	9	4	10	6	3
Boosting (Round 2)	5	4	9	4	2	5	1	7	4	2
Boosting (Round 3)	4	4	8	10	4	5	4	6	3	4

- Exemplo 4 é difícil de acertar.

Ensemble methods

AdaBoost:

- O método associa um peso inicial $w_i = 1/N$ a cada observação (x_i, y_i). A seguir, o peso é atualizado por:

$$w_i^{(j+1)} = \frac{w_i^{(j)}}{Z_j} \times \begin{cases} \exp^{-\alpha_j} & \text{if } C_j(\mathbf{x_i}) = y_i \\ \exp^{\alpha_j} & \text{if } C_j(\mathbf{x_i}) \neq y_i \end{cases},$$

onde:

$$\alpha_i = \frac{1}{2} \ln \left(\frac{1 - \epsilon_i}{\epsilon_i} \right). \qquad \epsilon_i = \frac{1}{N} \left[\sum_{j=1}^{N} w_j \, I\Big(C_i(\mathbf{x}_j) \neq y_j \Big) \right],$$

- Z_j é uma constante normalizadora.
- As amostras são sorteadas de acordo com o peso. Quanto maior o peso, maior a chance de ser selecionado.

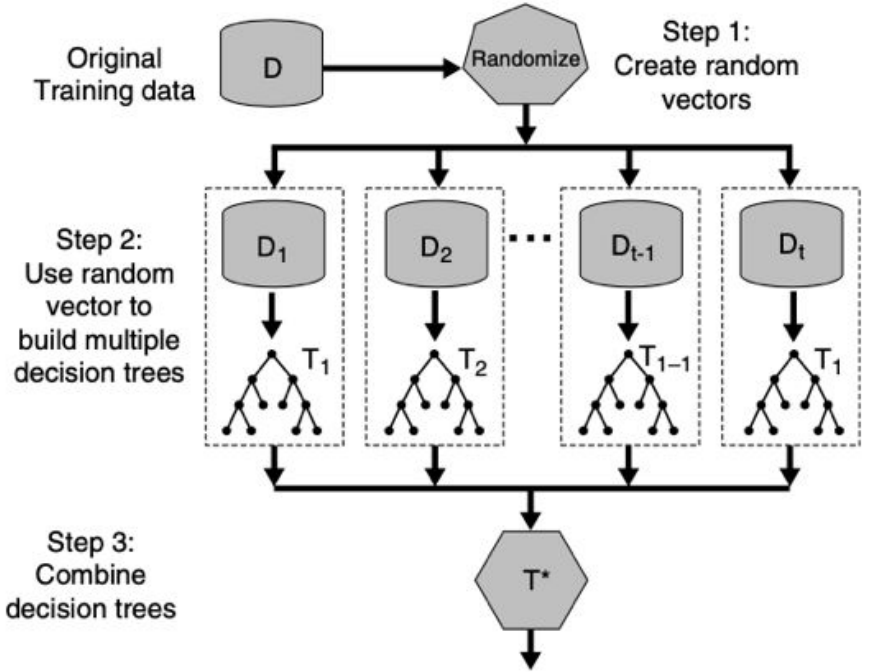
Ensemble methods

- **Propriedades do Boosting:**
- Se os dados apresentarem ruídos, pode ocorrer *overfitting*.
- O algoritmo precisa lidar com duas escolhas importantes:
  - Qual classificador de base será usado (geralmente usa-se árvores de decisão).
  - Quantas interações (B) serão usadas?
- Boosting ajusta o viés e variância.
  - Enquanto que o método bagging necessita de um classificador com baixo viés, mas pode apresentar alta variância, boosting não requer que o classificador tenham baixo viés.
  - No entanto, bagging não produz *overfitting*.

Ensemble methods

Florestas aleatórias

- É parecido com os métodos bagging e boosting, mas também amostra os atributos.

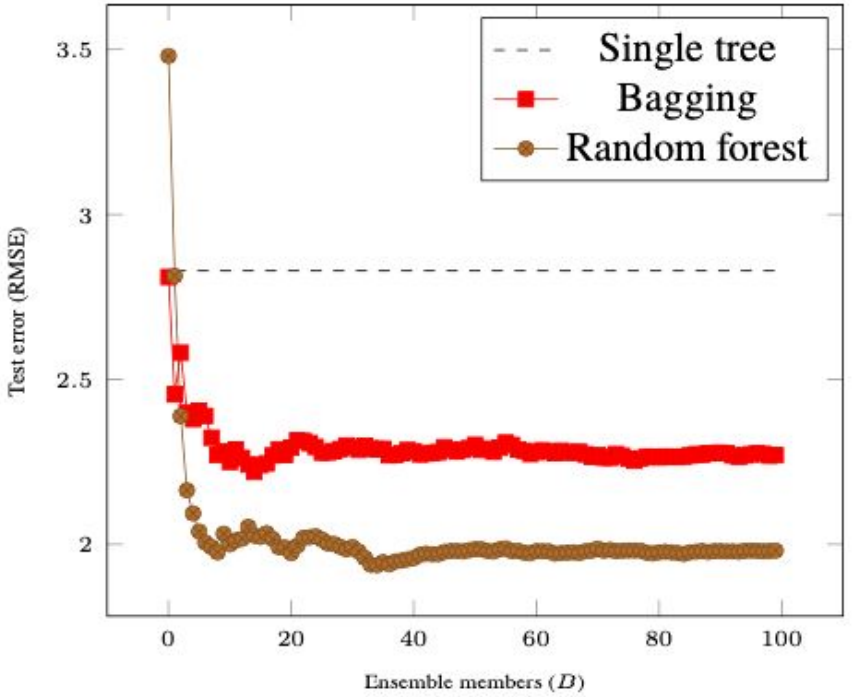


Random forests.

Ensemble methods

Florestas aleatórias

- A amostragem dos atributos permite que as árvores geradas não sejam dominadas por um atributo com alto poder de discriminação.



Sumário

- Árvores de Decisão
- Ensemble methods

Leitura adicional

- Lindholm et al., Supervised Machine Learning, 2019. http://www.it.uu.se/edu/course/homepage/sml/literature/lecture_notes.pdf

- Introduction to Data Mining, Tan, Steinbach, Karpatne, Kumar, Pearson, 2013.