

Inteligência Artificial

Guilherme Henrique de Souza Nakahata

Universidade Estadual do Paraná - Unespar

02 de Abril de 2024

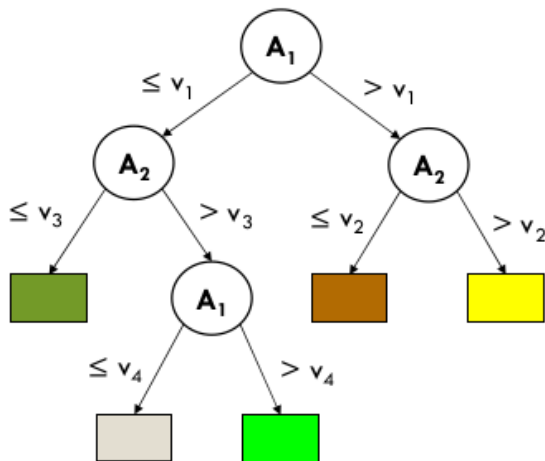
Árvores de decisão

- Modelos simbólicos;
- Utilizam uma linguagem de representação de alto nível;
- Fácil compreensão por seres humanos;
 - Árvores de decisão ou regras;
- Aprendizado;
- Exploração do espaço de hipóteses;
 - Percorre-se o espaço de hipóteses possíveis;
 - Buscando a melhor hipótese;
 - Evita percorrer o espaço de forma exaustiva;
 - Busca heurística;
- Árvores de decisão;
- Regras;

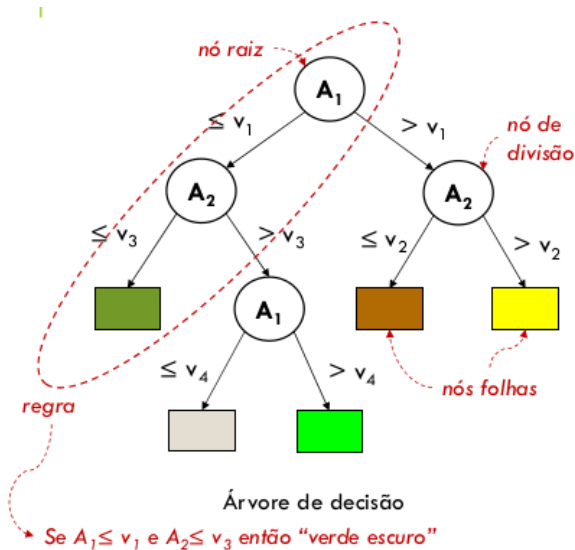
- Simples;
- Bem-sucedida;
- Indução de árvores de decisão;
 - Aproximar funções discretas;
 - Classificação;
 - Adaptadas para funções contínuas;
 - Regressão;
- Estratégia dividir-para-conquistar;
 - Problema complexo é decomposto em subproblemas mais simples;
 - Recursivamente;

- Uma árvore de decisão é um grafo acíclico direcionado;
- Cada nó:
 - Nó de divisão, com dois ou mais filhos;
 - Nó folha;
- Nó folha:
 - Moda dos valores de classe naquela folha;
- Nó de divisão:
 - Teste condicional baseado nos valores do atributo que está sendo testado.

Árvores de decisão



Árvores de decisão



Construção da árvore

- Com construir uma árvore de decisão?
- Consistente com o conjunto de treinamento;
- Espaço de hipóteses;
- Muito grande para uma busca exaustiva;
- Busca heurística;
- Top-down com escolha gulosas;
- Nó raiz e depois recursivamente os próximos nós;
- Nós folhas.

- Múltiplas árvores consistentes no espaço de hipóteses;
- Árvores mais simples diminuem a chance de superajuste;
- "O problema de encontrar a menor árvore de decisão consistente com um conjunto de treinamento é NP-completo" (QUINLAN, 1993)"
- Buscas gulosas minimiza o número de árvores possíveis;
- Não temos garantia de que a árvore encontrar é a melhor árvore possível.

Construção da árvore

- Passos gerais para a indução top-down de uma árvore de uma de decisão:
 - Escolha um atributo e crie um nó;
 - Estenda a árvore adicionando ao nó um ramo para cada valor possível do atributo
 - Divida os exemplos entre os nós filhos tendo em conta o valor do atributo em uso
 - Para cada nó filho:
 - Se todos os exemplos do filho são de uma mesma classe, crie um nó folha e associe essa classe à ela
 - Senão, repita os passos 1 a 4 considerando o subconjunto de exemplos presentes no nó filho

Construção da árvore

- Passos gerais para a indução top-down de uma árvore de uma de decisão:
 - Escolha um atributo e crie um nó;
 - Estenda a árvore adicionando ao nó um ramo para cada valor possível do atributo
 - Divida os exemplos entre os nós filhos tendo em conta o valor do atributo em uso
 - Para cada nó filho:
 - Se todos os exemplos do filho são de uma mesma classe, crie um nó folha e associe essa classe à ela
 - Senão, repita os passos 1 a 4 considerando o subconjunto de exemplos presentes no nó filho

Atributo

Qual atributo escolher no passo 1?

- Árvores consistentes e pequenas;
- Melhor divide os exemplos;
 - Fornece mais informação a respeito das classes;
 - Mais discriminativo;
- Como medir a capacidade de um atributo?
- Várias heurísticas.

- ID3 (anos 80)
 - Proposto para problemas de classificação com atributos categóricos
 - Utiliza o ganho de informação como métrica para a seleção de atributos
- C4.5 (anos 90)
 - É o algoritmo de indução de árvores de decisão mais utilizado até os dias de hoje
 - Incorpora uma série de melhorias ao ID3, como o tratamento de atributos numéricos e poda da árvore
 - Utiliza a razão de ganho como métrica para a seleção de atributos

Exemplo

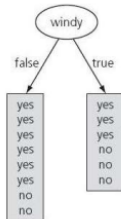
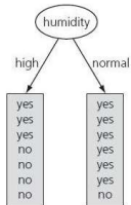
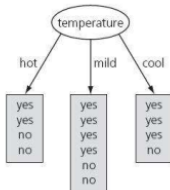
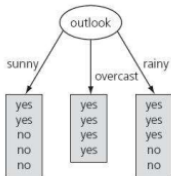
Outlook	Temperature	Humidity	Windy	Play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

classe

14 exemplos
2 classes: yes e no

- 9 exemplos yes
- 5 exemplos no

Exemplo

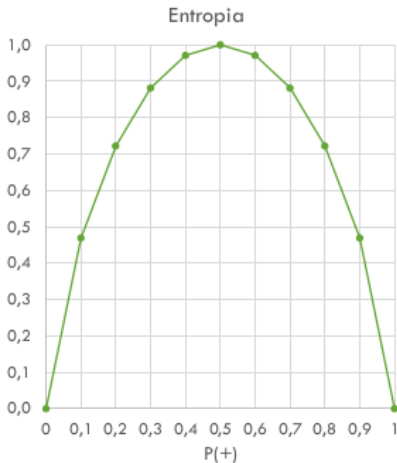


- O algoritmo ID3 seleciona atributos para inserção em ganho de informação (information gain);
- O ganho de informação é calculado com base em uma medida chamada entropia;
- No contexto da teoria da informação, a entropia é uma função que estima a aleatoriedade de um conjunto (impureza);
- Em outras palavras, a entropia indica se o conjunto apresenta tendência para uma ou outra classe;

- Aleatoriedade está relacionada à distribuição do atributo;
- Exemplo para problemas binários:
 - $Entropia(S) = -(P_+ X \log_2 P_+) - (P_- X \log_2 P_-)$
- onde:
 - S é o conjunto de exemplos;
 - $+$ e $-$ são as classes possíveis;
 - P_+ proporção de exemplos positivos;
 - P_- proporção de exemplos negativos;

Entropia

- Se entropia(S) é igual 0, todos os exemplos tem a mesma classe;
- Se entropia(S) é igual 1, os exemplos estão balanceados, P_+ e $P_- = 0,5$.



Exemplo

Outlook	Temperature	Humidity	Windy	Play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

classe

14 exemplos
2 classes: yes e no

- 9 exemplos yes
- 5 exemplos no

Exemplo

- $S = [9_{yes}, 5_{no}]$;
- $P_{yes} = \frac{9}{14}$;
- $P_{no} = \frac{5}{14}$;
- $P_{yes} + P_{no} = 1$;
- $Entropia(S) = -\frac{9}{14} \log_2 \frac{9}{14} - \frac{5}{14} \log_2 \frac{5}{14}$;
- $Entropia(S) = -(0,643 \times (-0,637)) - (0,357 \times (-1,485))$
- $Entropia(S) = -(-0,410) - (-0,531)$
- $Entropia(S) = 0,940$

- O **ganho de informação (IG)** é calculado para um **atributo**;
- Ganho de informação = Entropia(S) - Somatório Entropia(A);
- $\sum \frac{|S_v|}{|S|} \times Entropia(S_v)$

Exemplo

Fazendo os cálculos para o atributo outlook:

$$S_{\text{sunny}} = [2_{\text{yes}}, 3_{\text{no}}] \rightarrow P_{\text{yes}} = \frac{2}{5}, P_{\text{no}} = \frac{3}{5}$$

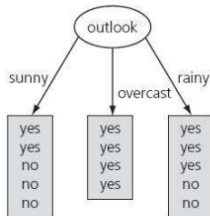
$$\text{Entropia}(S_{\text{sunny}}) = -\frac{2}{5} \log_2 \frac{2}{5} - \frac{3}{5} \log_2 \frac{3}{5} = [0,971]$$

$$S_{\text{overcast}} = [4_{\text{yes}}, 0_{\text{no}}] \rightarrow P_{\text{yes}} = \frac{4}{4}, P_{\text{no}} = \frac{0}{4}$$

$$\text{Entropia}(S_{\text{overcast}}) = -\frac{4}{4} \log_2 \frac{4}{4} - \frac{0}{4} \log_2 \frac{0}{4} = [0]$$

$$S_{\text{rainy}} = [3_{\text{yes}}, 2_{\text{no}}] \rightarrow P_{\text{yes}} = \frac{3}{5}, P_{\text{no}} = \frac{2}{5}$$

$$\text{Entropia}(S_{\text{rainy}}) = -\frac{3}{5} \log_2 \frac{3}{5} - \frac{2}{5} \log_2 \frac{2}{5} = [0,971]$$



$$\begin{aligned} \text{Entropia}(\text{outlook}) &= \frac{5}{14} \times \text{Entropia}(S_{\text{sunny}}) + \frac{4}{14} \times \text{Entropia}(S_{\text{overcast}}) + \frac{5}{14} \times \text{Entropia}(S_{\text{rainy}}) \\ &= \frac{5}{14} \times 0,971 + \frac{4}{14} \times 0 + \frac{5}{14} \times 0,971 = [0,693] \end{aligned}$$

Fazendo os cálculos para o atributo **outlook**:

$$Entropia(S) = -\frac{9}{14} \log_2 \frac{9}{14} - \frac{5}{14} \log_2 \frac{5}{14} = 0,940$$

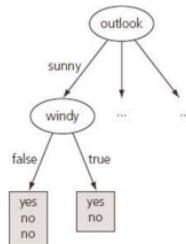
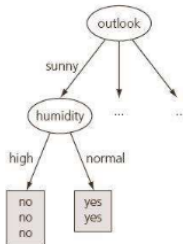
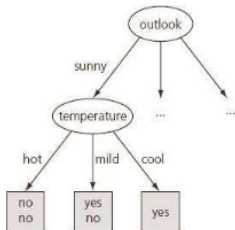
$$Entropia(outlook) = 0,693$$

$$\begin{aligned} IG(S, outlook) &= Entropia(S) - Entropia(outlook) \\ &= 0,940 - 0,693 \\ &= \boxed{0,247} \end{aligned}$$

- Fazendo os cálculos para os outros atributos:
 - Outlook = 0,247
 - Temperature = 0,029
 - Humidity = 0,152
 - Windy = 0,048
- O algoritmo ID3 escolhe o atributo outlook como raiz da árvore;
- O processo continua até que obtenha folhas puras.

Exemplo

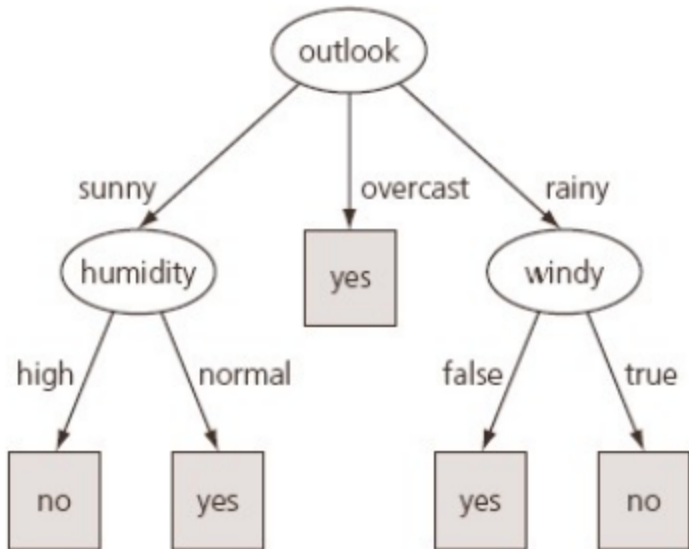
- Para a ramificação sunny qual é o melhor atributo que podemos colocar?
- Entropia de Sunny = 0,971 ($E(s)$);



- Calcule o ganho de informação para Temperature, Humidity e Windy;

- Temperature = 0,571;
- Humidity = **0,971**;
- Windy = 0,020;

Exemplo



- Continua recursivamente até não haver mais ramos a serem particionados;
- Não haver mais atributos a serem testados;
- Termina com folhas puras;
- Nem sempre acontece;
- Presença de ruídos;
- Atributos insuficientes;
- Termina quando os subconjuntos não puderem mais ser particionados;
- Todos os atributos já foram utilizados;

- Implemente um algoritmo que calcule o ganho de informação para um atributo.

Obrigado! Dúvidas?

Guilherme Henrique de Souza Nakahata

guilhermenakahata@gmail.com

<https://github.com/GuilhermeNakahata/UNESPAR-2024>