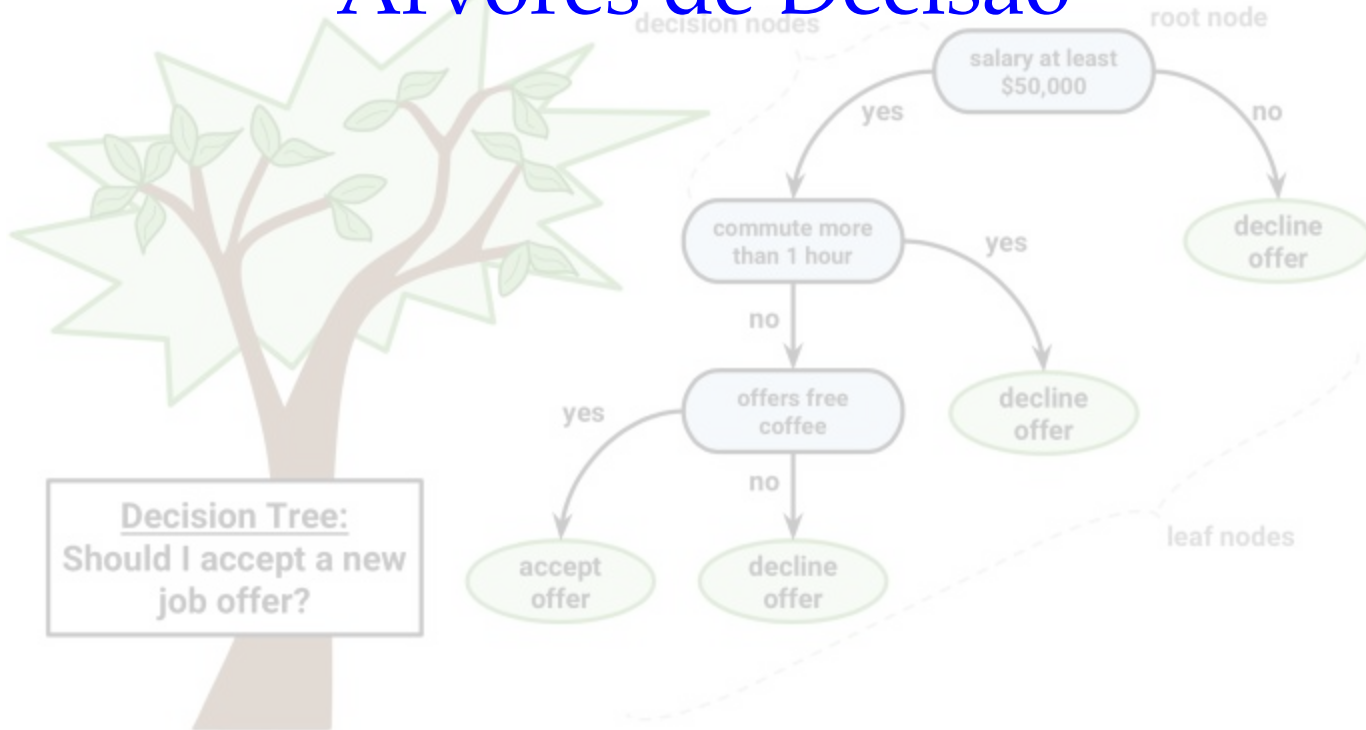


# Árvores de Decisão



David Menotti

[www.inf.ufpr.br/menotti/ci171-182](http://www.inf.ufpr.br/menotti/ci171-182)

# Árvores de Decisão

## Agenda

- Introdução
- Representação
- Quando Usar
- Algoritmo de Aprendizagem
- Resumo

# Árvores de Decisão

- Método prático
- Um dos mais utilizados na aprendizagem indutiva.
- Diferentemente dos métodos de aprendizagem conceitual, são mais robustas à ruídos nos dados (C4.5)
- Aproxima funções alvo de valor **discreto**, em que a função aprendida é representada por uma árvore de decisão.
- Também pode ser representada por um conjunto de regras (IF-THEN)
- *White model* - conhecimento compreensível
  - Diferentemente de uma rede neural que é muitas vezes referenciada como um *Black-box*

# Árvores de Decisão

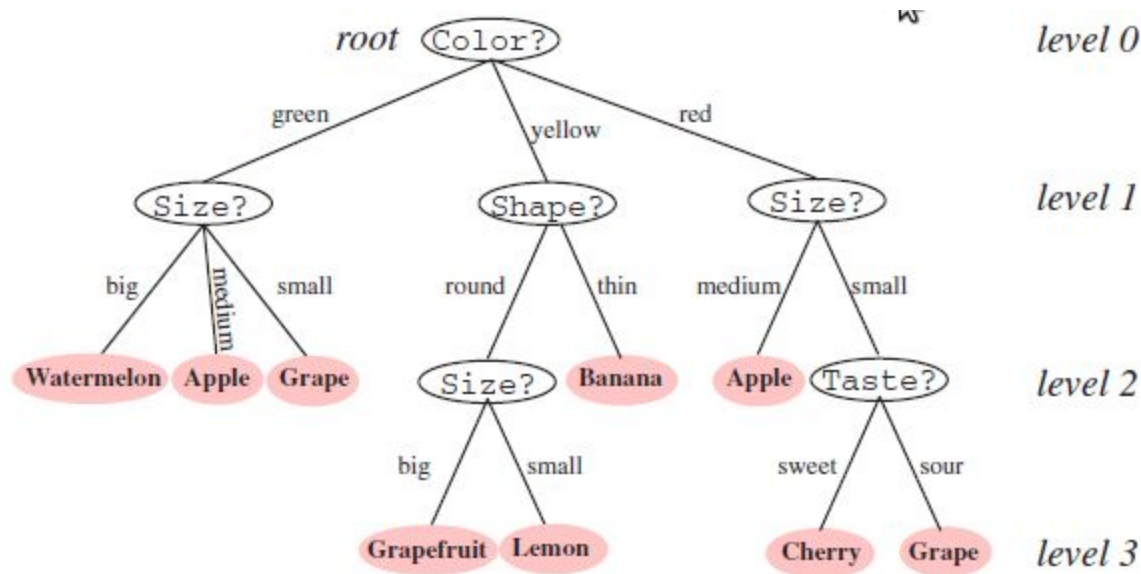
- Um dos métodos de aprendizagem mais conhecidos
- Não necessita de manipulação de dados, como por exemplo, métodos de normalização
- Pode receber tanto dados **numéricos** (C4.5) quanto **simbólicos**.
- Aplicações diversas como auxílio a diagnóstico, análise de risco, etc.

# Árvores de Decisão

Entretanto

- Alguns conceitos são de difícil aprendizagem em árvores de decisão, gerando árvores extremamente grandes, por exemplo, XOR.
- Aprendizagem de uma árvore ótima é conhecida como **NP Completo**.
- Utiliza heurísticas.
- Pode não gerar a melhor árvore.

# Representação



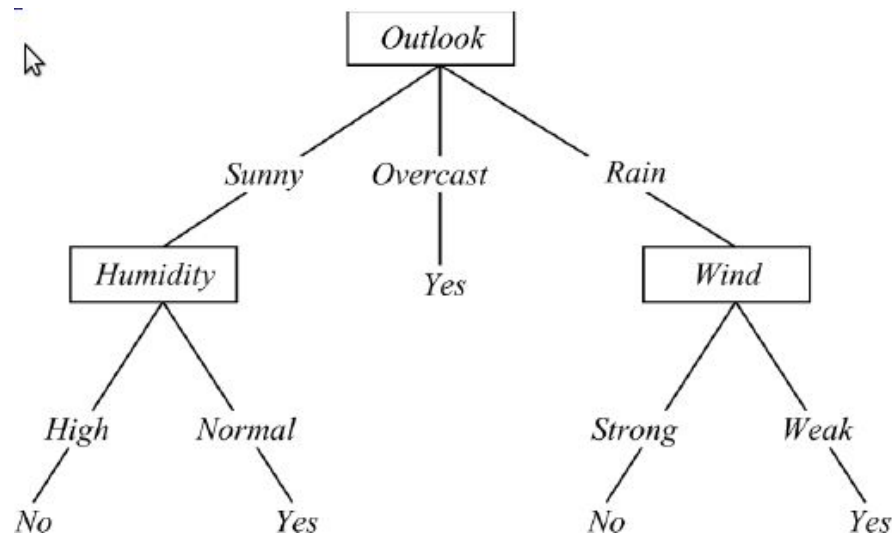
- Árvores de decisão classificam instâncias ordenando-as sub-árvores acima (ou abaixo), a partir da raiz até alguma folha.
- Cada nó da árvore especifica a avaliação de algum atributo da instância.
- Cada ramo partindo de um nó corresponde a um dos valores possíveis dos atributos

# Representação

- Uma instância é classificada inicialmente pelo nó raiz, testando o atributo especificado por este nó
- Em seguida, movendo-se através do ramo correspondendo ao valor do atributo no exemplo dado
- Este processo é repetido para a sub-árvore originada no novo nó

# Representação

Uma árvore de decisão para o conceito *Play Tennis*



- Um exemplo é classificado ordenado-o através da árvore para o nó da folha apropriado.
- Então retorna a classificação associada com esta folha (Yes / No)



# Representação

- Em geral, as árvores de decisão representam uma disjunção de restrições sobre valores dos atributos das instâncias
- Cada caminho entre a raiz da árvore e uma folha corresponde a uma **conjunção (E)** de testes de atributos e a própria árvore corresponde a uma **disjunção (OU)** destas conjunções.

Exemplo:

( **Outlook** = Sunny      **E**    **Humidity** = Normal )  
**OU** ( **Outlook** = Overcast )  
**OU** ( **Outlook** = Rain      **E**    **Wind** = Weak )

# Quando Considerar Árvores de Decisão

- Instâncias descritas por pares atributo-valor.
- Instâncias descritas por um conjunto **fixo** de atributos, por exemplo, `Temperatura` com valores definidos (`Quente`, `Frio`).
- **Classe** tem valores discretos de saída, por exemplo, `Sim` e `Não`.
- Dados de treinamento podem conter erros e valores de atributos faltantes (C4.5).

## Casos de aplicações:

- Diagnóstico ou equipamentos médicos
- Análise de Risco
- Modelagem de preferências em agendamento

# Algoritmo Básico

- A maioria dos algoritmos de aprendizagem de árvores derivam do algoritmo ID3.
  - C4.5 e C5.0 são mais recentes
  - O ID3 aprende a árvore usando uma estratégia *top-down*
- **Questão inicial?**
  - Qual atributo deve ser testado na raiz da árvore?
- Para cada atributo A da base de dados
  - Avalie como A classifica *Train set*
    - Como avaliar?

# Algoritmo Básico

- O melhor atributo é selecionado e usado como **raiz** da árvore.
- Um descendente (**sub-árvore**) do nó raiz é então criado para cada valor possível deste atributo e os exemplos de treinamento são ordenados para o nó descendente apropriado.
- O processo é repetido usando exemplos com cada nó descendente para selecionar o melhor atributo para avaliar naquele ponto da árvore.
- Um algoritmo de busca gulosa ( *greed* ) é utilizado
  - i.e., não recua para reconsiderar escolhas prévias.
- **Ganho de Informação (entropia)** é utilizada como medida quantitativa.

# Entropia

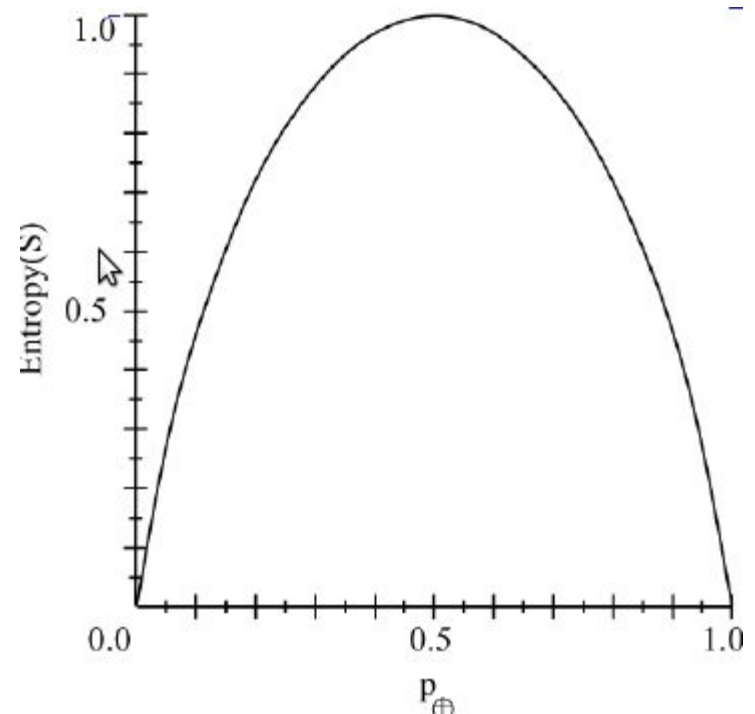
Medida de aleatoriedade de uma variável

- A entropia de uma variável nominal  $X$  que pode tomar  $i$  valores

- $Entropia(X) = - \sum_i p_i \times \log_2 p_i$

- Ex:  $Entropia([9+, 5-]) = - \left( \frac{9}{14} \log_2 \frac{9}{14} + \frac{5}{14} \log_2 \frac{5}{14} \right) = 0.94$

- A entropia tem máximo ( $\log_2 i$ ) se  $p_i = p_j$  para qualquer  $i \neq j$
- A  $Entropia(X) = 0$  se existe um  $i$  tal que  $p_i = 1$
- É assumido que  $0 \times \log_2 0 = 0$



# Ganho de Informação

- No contexto das árvores de decisão a entropia é usada para estimar a aleatoriedade da variável a prever: **classe**.
- Dado um conjunto de exemplos, que **atributo** escolher?
  - Os valores de um atributo definem partições do conjunto de exemplos
  - O ganho de informação mede a redução da entropia causada pela partição dos exemplos de acordo com os valores do atributo.

$$Ganho(S, A) = Entropia(S) - \sum_{v \in Valores(A)} \frac{|S_v|}{|S|} Entropia(S_v)$$

- A construção de uma árvore de decisão é guiada pelo objetivo de diminuir a entropia ou seja a aleatoriedade-dificuldade de previsão da variável objetivo

# Exemplo

- Base de dados para o problema “Play Tennis”

<i>Day</i>	<i>Outlook</i>	<i>Temperature</i>	<i>Humidity</i>	<i>Wind</i>	<i>PlayTennis</i>
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

# Exemplo

- Calcular o Ganho (  $S$  , **Wind** )
  - Valores ( **Wind** ) = { *Weak* , *Strong* }
  - $S = [ 9+ , 5- ]$ 
    - $S_{Weak} = [ 6+ , 2- ]$
    - $S_{Strong} = [ 3+ , 3- ]$

$$Ganho(S, Wind) = E(S) - \sum_{v \in \{Weak, Strong\}} \frac{|S_v|}{|S|} E(S_v)$$

$$Ganho(S, Wind) = E(S) - \left( \frac{8}{14} \right) E(S_{Weak}) - \left( \frac{6}{14} \right) E(S_{Strong})$$

$$Ganho(S, Wind) = 0,940 - \left( \frac{8}{14} \right) 0,811 - \left( \frac{6}{14} \right) 1,000 = 0,048$$



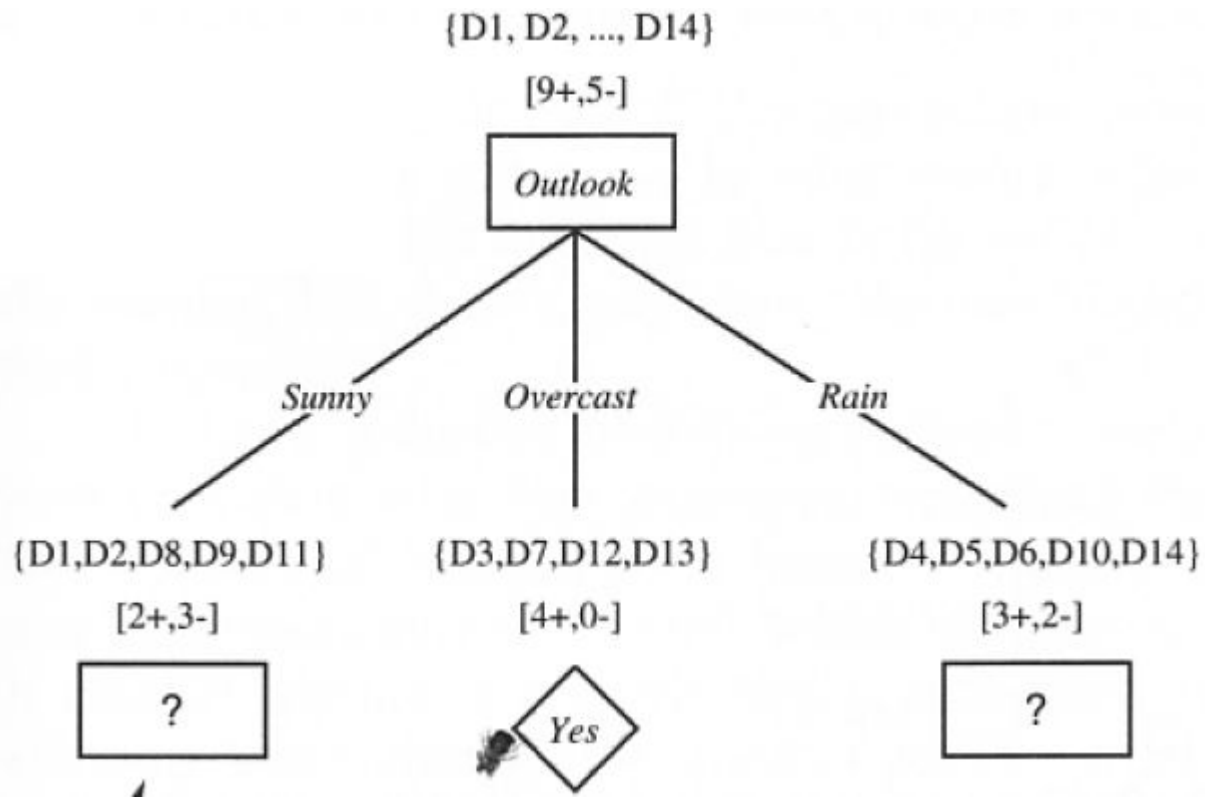
# Exemplo

## Voltando a pergunta inicial

Qual atributo deve ser testado primeiro na árvore?

- Determinar o ganho de informação (Gain) para cada atributo candidato.
- Selecionar aquele cujo o ganho de informação é o mais alto.
  - $\text{Ganho}( S , \textit{Outlook} ) = \mathbf{0,246}$
  - $\text{Ganho}( S , \textit{Humidity} ) = 0,151$
  - $\text{Ganho}( S , \textit{Wind} ) = 0,048$
  - $\text{Ganho}( S , \textit{Temperature} ) = 0,029$

# Exemplo

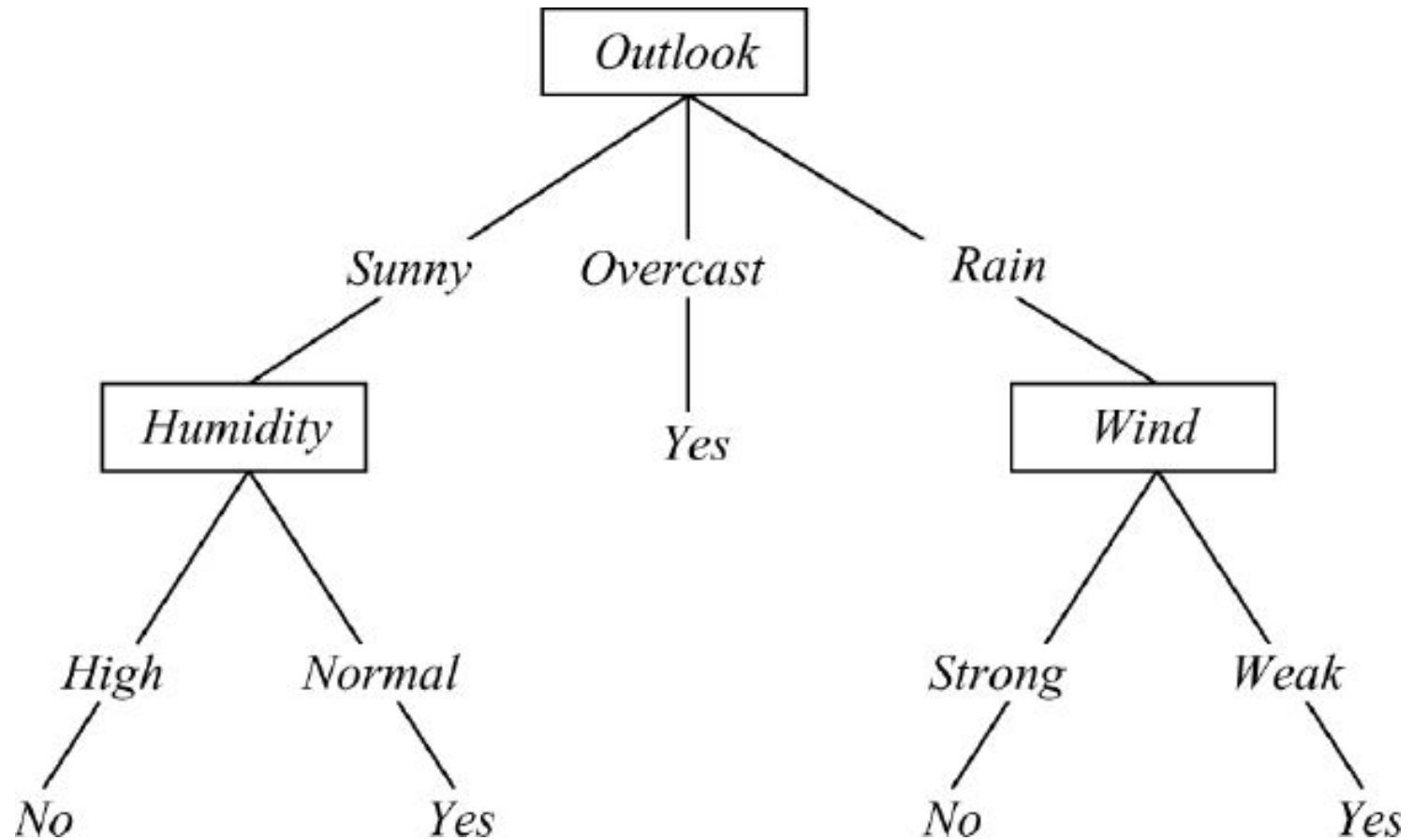


# Exemplo

- O processo para seleccionar um novo atributo e particionar os exemplos de treinamento é repetido para cada nó descendente não terminal.
- São utilizados somente os exemplos de treinamento associados com este nó.
- Atributos que foram incorporados anteriormente à árvore são excluídos. Qualquer atributo\* deve aparecer somente uma vez ao longo de qualquer caminho na árvore.
- Este processo continua até que uma das seguintes condições seja atendida:
  - a. Todos os atributos já estejam incluídos ao longo deste caminho da árvore.
  - b. Os exemplos de treinamento associados com este nó folha tenham todos o mesmo valor de atributo alvo.

# Exemplo

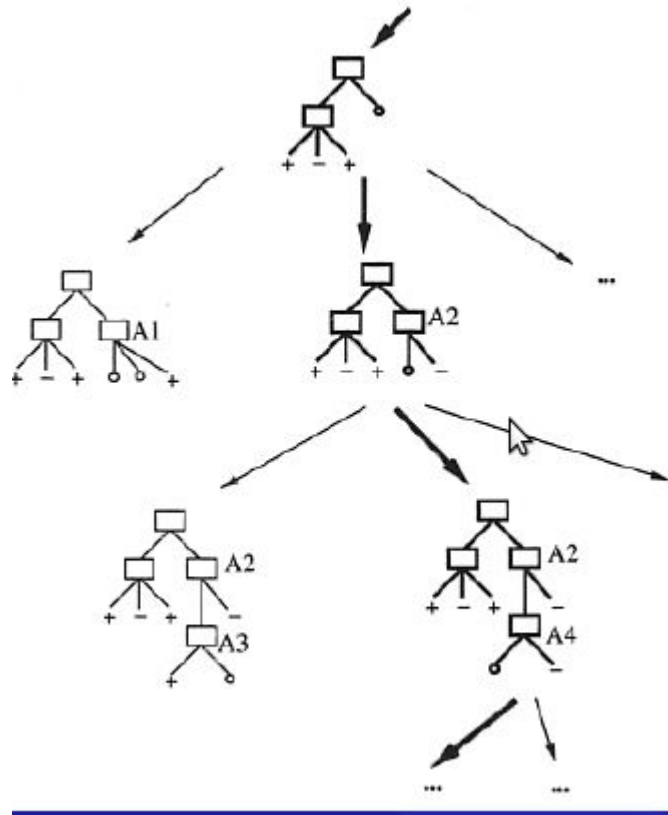
A árvore final



# Busca no Espaço de Hipóteses

- O modelo de aprendizagem ID3 (Iterative Dichotomizer 3) pode ser caracterizado como um método de busca em um espaço de hipótese, por uma hipótese que se ajusta aos exemplos de treinamento.
- O espaço de hipóteses buscado pelo ID3 é o conjunto de árvores de decisão possíveis.
- O ID3 realiza uma busca simples
  - *hill climbing* através do espaço de hipótese começando com uma árvore vazia e considerando progressivamente hipóteses mais elaboradas

# Busca no Espaço de Hipóteses



- Espaço de Hipóteses - ID3 procura possíveis árvores a partir da mais simples aumentando a complexidade, usando para isso o ganho de informação

# *Bias Indutivo*

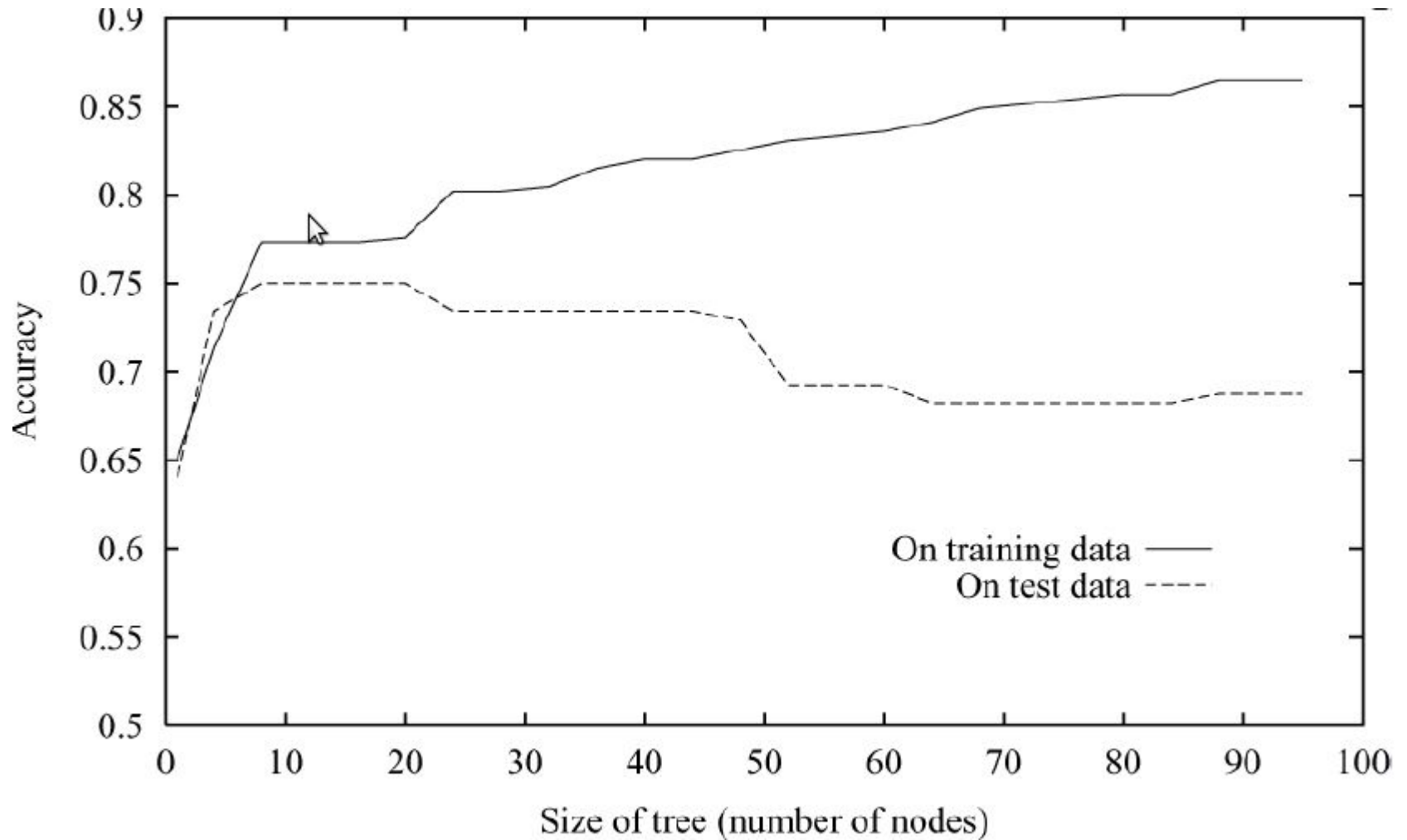
- Dada uma coleção de exemplos de treinamento, geralmente existem várias árvores de decisão consistentes com os exemplos.
- Qual árvore deve ser escolhida?
  - Mecanismo implícito do algoritmo.
- Preferência é por árvores mais curtas e por aquelas com atributos de **alto ganho de informação** próximos da raiz.
  - **Bias**: é uma preferência por algumas hipóteses aos invés de uma restrição do espaço de hipóteses.
  - Modelos menos complexos (árvores menores) são preferíveis (menor risco de *overfitting*).

# Overfitting

- Como detectar *overfitting* em árvores de decisão.
  - Erro da hipótese  $h$  sobre os dados de treinamento:  $err_t(h)$
  - Erro da hipótese  $h$  sobre todos os dados:  $err_{all}(h)$
- Uma hipótese  $h \in H$  tem *overfitting* sobre os dados de treinamento se existir uma hipótese alternativa  $h' \in H$  tal que
  - $err_t(h) < err_t(h')$  E
  - $err_{all}(h) > err_{all}(h')$



# Overfitting



# Evitando o *Overfitting*

- Como o *overfitting* pode ser evitado?
  - Parar o crescimento quando a partição de dados não for **estatisticamente significativa**.
  - Desenvolver uma árvore completa e então fazer uma poda ( *Prunning* ).
    - Pode ser feita diretamente na árvore ou ainda no conjunto de regras geradas pela árvore.
- Como selecionar a melhor árvore?
  - Medida de desempenho sobre um conjunto de dados de validação

# Atributos de Valor Contínuo (c4.5)

- Em alguns casos, os valores dos atributos podem ser apresentados na forma contínua, por exemplo:

Temperatura:	40	48	60	72	80	90
Play Tennis:	No	No	Yes	Yes	Yes	No

- Escolher um limiar **c** que produza o maior ganho de informações.
- Identificar exemplos adjacentes que diferem na classificação do alvo.
- Por exemplo, um limiar poderia ser
  - **c** =  $(48 + 60)/2 = 54$

# Árvores de Decisão - Resumo

- A aprendizagem de árvores de decisão fornece um método prático para a aprendizagem de conceito e para a aprendizagem de outras funções de valor discreto.
- A família de algoritmos ID3 infere árvores de decisão expandindo-as a partir da raiz e descendo, selecionando o próximo melhor atributo para cada novo ramo de decisão adicionado na árvore.
- O **bias indutivo** implícito no ID3 inclui uma preferência por árvores menores.
- **Overfitting** é um aspecto importante na aprendizagem de árvores de decisão.

# Referências

- Luiz E. S Oliviera, **Árvores de Decisão**, DInf / UFPR, 2017.
- João Gama, **Árvores de Decisão**, notas de aula [jgama@ncc.up.pt](mailto:jgama@ncc.up.pt), 2012.