

# Árvores de decisão

Victor Lobo

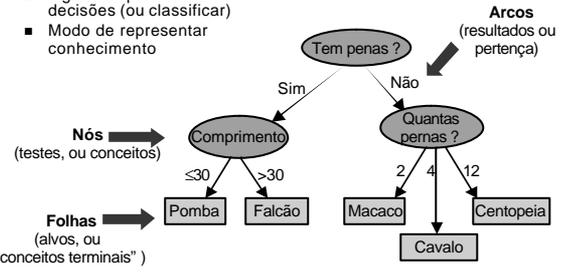
## O que é uma árvore de decisão ?

- Algoritmo para tomar decisões (ou classificar)
- Modo de representar conhecimento

**Nós** (testes, ou conceitos)

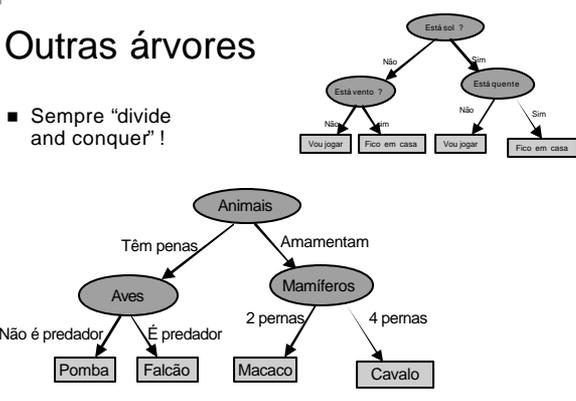
**Folhas** (alvos, ou "conceitos terminais")

**Arcos** (resultados ou pertinça)



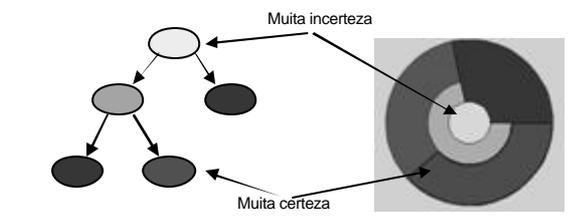
## Outras árvores

- Sempre "divide and conquer" !



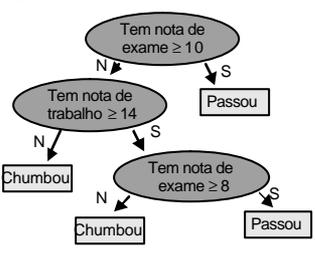
## Outra maneira de ver árvores

- Vista de cima...
  - Permite ver também o número de dados abrangidos, e o poder discriminante da pergunta



## Extracção de regras (a partir de árvores)

- Cada arco
  - acrescenta uma conjunção
- Cada folha "repetida"
  - acrescenta uma disjunção



Passou  $\Leftrightarrow$  (Exame  $\geq 10$ )  $\vee$  ( (Exame  $< 10$ )  $\wedge$  (Trabalho  $\geq 14$ )  $\wedge$  (Exame  $\geq 8$ ) )

Chumbou  $\Leftrightarrow$  ( (Exame  $< 10$ )  $\wedge$  (Trabalho  $< 14$ ) )  $\vee$  ( (Exame  $< 10$ )  $\wedge$  (Trabalho  $\geq 14$ )  $\wedge$  (Exame  $< 8$ ) )

## Vantagens das árvores (1)

- Interpretação
  - Percebe-se a razão da decisão
- Facilidade em lidar com diversos tipos de informação
  - Real, nominal, ordinal, etc
  - Não é necessário definir "importância relativa"
- Insensível a factores de escala

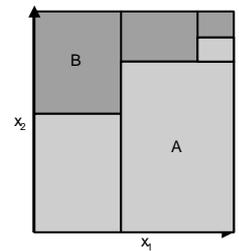
**Importantíssimo !**  
**A razão !**

## Vantagens das árvores (2)

- Escolha automática dos atributos mais relevantes em cada caso
  - Atributos mais relevantes aparecem mais acima na árvore
- Adaptável também a problemas de regressão
  - Modelos locais lineares como folhas

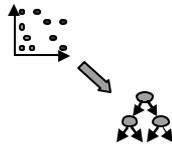
## Desvantagens de árvores

- Fronteiras lineares e perpendiculares aos eixos (CART & Cia.)
- Sensibilidade a pequenas perturbações no conjunto de treino (geram redes muito diferentes)



## Indução de árvores de decisão

- A partir de um conjunto de treino, construir uma árvore
- Problemas:
  - Que pergunta fazer ?
    - Que variável interrogar ?
    - Qual o valor de corte ?
    - Qual o nó a "partir" ?
  - Quantos ramos pôr em cada nó ?
  - Quando parar ?



## Algoritmo básico de indução de árvores de decisão

- Em cada nível divide o conjunto de exemplos em partições alternativas.
  - Utilizando uma medida da *QUALIDADE* da partição selecciona a melhor partição.
- Para a partição seleccionada, volta a repetir o processo para cada um dos elementos da partição.
- Parar quando algum critério for atingido

## Algoritmos mais usados

- ID3, C4.5 e C5 [Quinlan 86,93]
  - Iterative Dichotomizer 3
- CART [Breiman 84]
  - Classification and regression trees
- CHAID [Hartigan 75]
  - *Chi-Squared Automatic Interaction Detection*
  - Usado pelo SPSS e SAS...
- Muitas (mesmo muitas) outras variantes...
  - Em SAS: possibilidade de seleccionar os diferentes parâmetros para a construção da árvore

## Algoritmo DDT (devisive decision tree - Hunt 62)

- Assume-se que existe uma atributo especial a "*Classe*" e que os exemplos foram previamente classificados.
- Cada nó especifica um único atributo, que é usado como teste, designado por atributo mais discriminante.
- $N$  – o nó  $N$
- ASET – Attribute Set – Conjunto de atributos
- ISET – Instance Set – Conjunto de exemplos

## DDT(N, ASET, ISET)

Se o conjunto *ISET* é vazio então o nó terminal *N* é da classe *desconhecida* senão

Se todas os exemplos de *ISET* são da mesma classe então o nó terminal *N* tem o nome da classe senão

Para cada atributo *A* do conjunto de atributos *ASET*

Avalia *A* de acordo com a capacidade de discriminar a classe

Seleciona o atributo *B* que tem o melhor valor discriminante

Para cada valor *V* do melhor atributo *B*

Cria um novo filho *C* do nó *N*

Coloca o par atributo valor (*B, V*) em *C*

Seja *JSET* o conjunto de exemplos de *ISET* com o valor *V* em *B*

Seja *KSET* o conjunto de atributos de *ASET* com *B* removido

DDT(*C, KSET, JSET*)

## Pesquisa

- É uma pesquisa “gulosa”.
- Não tem “backtracking”.
- Pode ficar presa num mínimo local.

*O “bias” desta aproximação indutiva é que as árvores mais pequenas são preferíveis às árvores grandes.*

*(Occam’s razor: prefere a hipótese mais simples que justifica os dados - 1320)*

## Exemplo (Análise de células [Langley 96])

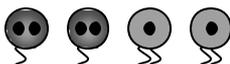
Lethargia



Burpoma



Saudável



## Forma tabelar

# Núcleos	# Caudas	Cor	Membrana	Classe
1	1	Clara	Fina	Lethargia
2	1	Clara	Fina	Lethargia
1	1	Clara	Grossa	Lethargia
1	1	Escura	Fina	Lethargia
1	1	Escura	Grossa	Lethargia
2	2	Clara	Fina	Burpoma
2	2	Escura	Fina	Burpoma
2	2	Escura	Grossa	Burpoma
2	1	Escura	Fina	Saudável
2	1	Escura	Grossa	Saudável
1	2	Clara	Fina	Saudável
1	2	Clara	Grossa	Saudável

## Métrica de qualidade

- Seja a medida de discriminação do atributo

$$f(A) = \frac{1}{n} \sum_{i=1}^{|A|} C_i$$

em que *n* é o número total de exemplos e *C<sub>i</sub>* é o número de exemplos correctamente classificados pela classe mais frequente.

É uma medida da “dominância” ou “pureza”

## Tabelas

Se fizermos a partição pelo *nº* de núcleos...

# núcleos	1	2
Lethargia	4	1
Burpoma	0	3
Saudável	2	2

Poder discriminante:

$$(4 + 3) / 12 = 0.58$$

# Núcleos	# Caudas	Cor	Membrana	Classe
1	1	Clara	Fina	Lethargia
2	1	Clara	Fina	Lethargia
1	1	Clara	Grossa	Lethargia
1	1	Escura	Fina	Lethargia
1	1	Escura	Grossa	Lethargia
2	2	Clara	Fina	Burpoma
2	2	Escura	Fina	Burpoma
2	2	Escura	Grossa	Burpoma
2	1	Escura	Fina	Saudável
2	1	Escura	Grossa	Saudável
1	2	Clara	Fina	Saudável
1	2	Clara	Grossa	Saudável

### Tabelas

Se fizemos a partição pelo nº de caudas...

# caudas	1	2
Lethargia	5	0
Burpoma	0	3
Saudável	2	2

Poder discriminante:  
 $(5 + 3) / 12 = 0.67$

# Núcleos	# Caudas	Cor	Membrana	Classe
1	1	Clara	Fina	Lethargia
2	1	Clara	Fina	Lethargia
1	1	Clara	Grossa	Lethargia
1	1	Escura	Fina	Lethargia
1	1	Escura	Grossa	Lethargia
2	2	Clara	Fina	Burpoma
2	2	Escura	Fina	Burpoma
2	2	Escura	Grossa	Burpoma
2	1	Escura	Fina	Saudável
2	1	Escura	Grossa	Saudável
1	2	Clara	Fina	Saudável
1	2	Clara	Grossa	Saudável

### Tabelas

Se fizemos a partição peça côr ...

Cor	Clara	Escura
Lethargia	3	2
Burpoma	1	2
Saudável	2	2

Poder discriminante:  
 $(3 + 2) / 12 = 0.41$

# Núcleos	# Caudas	Cor	Membrana	Classe
1	1	Clara	Fina	Lethargia
2	1	Clara	Fina	Lethargia
1	1	Clara	Grossa	Lethargia
1	1	Escura	Fina	Lethargia
1	1	Escura	Grossa	Lethargia
2	2	Clara	Fina	Burpoma
2	2	Escura	Fina	Burpoma
2	2	Escura	Grossa	Burpoma
2	1	Escura	Fina	Saudável
2	1	Escura	Grossa	Saudável
1	2	Clara	Fina	Saudável
1	2	Clara	Grossa	Saudável

### Tabelas

Se fizemos a partição pelo tipo de membrana...

Membrana	Fina	Grossa
Lethargia	3	2
Burpoma	2	1
Saudável	3	1

Poder discriminante:  
 $(3 + 2) / 12 = 0.41$

# Núcleos	# Caudas	Cor	Membrana	Classe
1	1	Clara	Fina	Lethargia
2	1	Clara	Fina	Lethargia
1	1	Clara	Grossa	Lethargia
1	1	Escura	Fina	Lethargia
1	1	Escura	Grossa	Lethargia
2	2	Clara	Fina	Burpoma
2	2	Escura	Fina	Burpoma
2	2	Escura	Grossa	Burpoma
2	1	Escura	Fina	Saudável
2	1	Escura	Grossa	Saudável
1	2	Clara	Fina	Saudável
1	2	Clara	Grossa	Saudável

### Tabelas

# núcleos	1	2
Lethargia	4	1
Burpoma	0	3
Saudável	2	2

0.58

Cor	Clara	Escura
Lethargia	3	2
Burpoma	1	2
Saudável	2	2

0.41

# caudas	1	2
Lethargia	5	0
Burpoma	0	3
Saudável	2	2

0.67

Membrana	Fina	Grossa
Lethargia	3	2
Burpoma	2	1
Saudável	3	1

0.41

**Escolha: # Caudas**

### A árvore inicial

uma

# Núcleos	Cor	Membrana	Classe
1	Clara	Fina	Lethargia
2	Clara	Fina	Lethargia
1	Clara	Grossa	Lethargia
1	Escura	Fina	Lethargia
1	Escura	Grossa	Lethargia
2	Escura	Fina	Saudável
2	Escura	Grossa	Saudável

duas

# Núcleos	Cor	Membrana	Classe
2	Clara	Fina	Burpoma
2	Escura	Fina	Burpoma
2	Escura	Grossa	Burpoma
1	Clara	Fina	Saudável
1	Clara	Grossa	Saudável

### Tabelas sabendo #caudas = 1

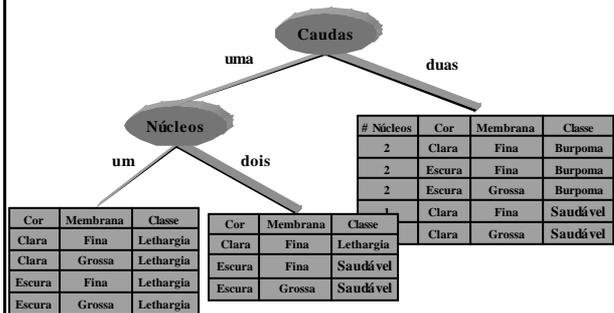
# núcleos	1	2
Lethargia	4	1
Burpoma	0	0
Saudável	0	2

# Núcleos	Cor	Membrana	Classe
1	Clara	Fina	Lethargia
2	Clara	Fina	Lethargia
1	Clara	Grossa	Lethargia
1	Escura	Fina	Lethargia
1	Escura	Grossa	Lethargia
2	Escura	Fina	Saudável
2	Escura	Grossa	Saudável

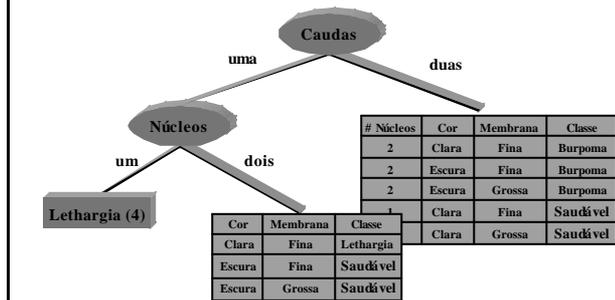
Cor	Clara	Escura
Lethargia	3	2
Burpoma	0	0
Saudável	0	2

Membrana	Fina	Grossa
Lethargia	3	2
Burpoma	0	0
Saudável	0	2

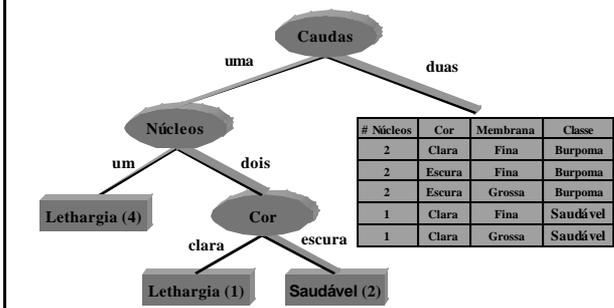
### A árvore (continuação 1)



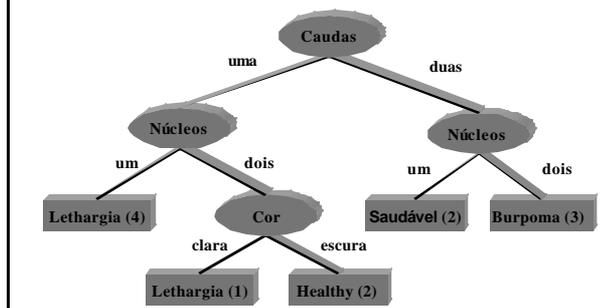
### A árvore (continuação 2)



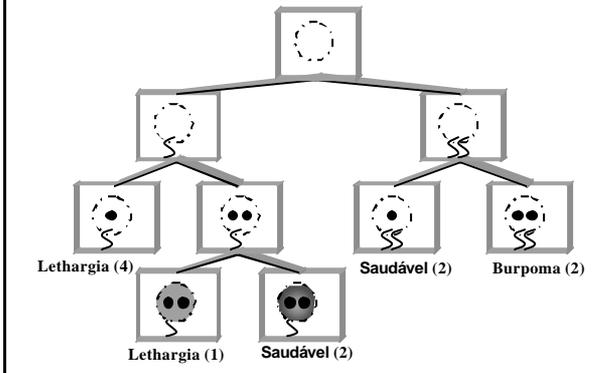
### A árvore (continuação 3)



### A árvore final



### Outra interpretação



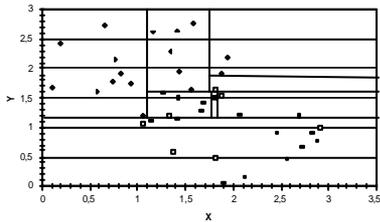
### Descrição das três classes

$$(caudas = uma) \wedge (núcleos = um) \\ \vee \\ (caudas = uma) \wedge (núcleos = dois) \wedge (cor = clara) \rightarrow Lethargia$$

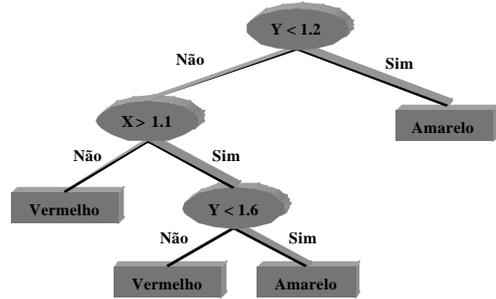
$$(caudas = duas) \wedge (núcleos = um) \\ \vee \\ (caudas = uma) \wedge (núcleos = 2) \wedge (cor = escura) \rightarrow Saudável$$

$$(caudas = duas) \wedge (núcleos = dois) \rightarrow Burpoma$$

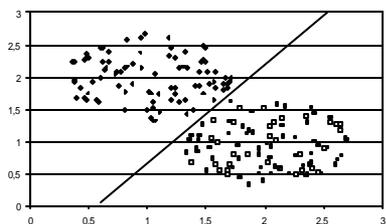
## Outro exemplo



## A árvore do exemplo anterior



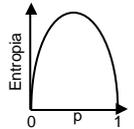
## Qual a melhor solução?



## Crítérios para escolher as partições

### Entropia

- Ideia base: maximizar a informação
- Mede a "pureza" de um nó pela entropia que é definida como sendo
  - $E = -p \log_2(p)$ 
    - onde p é a probabilidade dos exemplos terem uma dada classe



- Entropia de uma partição:

$$Ent(S) = \sum_{i=1}^{\#C} -p_i \log_2(p_i)$$

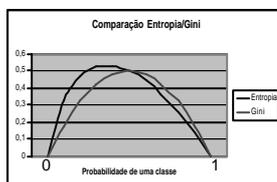
- Ganho da escolha do atributo A

$$Ganho(S, A) = Ent(S) - \sum_{v \in \text{Valores}(A)} \frac{\#S_v}{\#S} Ent(S_v)$$

## Crítérios para escolher as partições (2)

### Gini

- Parecido com a entropia, mas evita o cálculo do logaritmo usando apenas  $G = p(1-p)$
- No caso mais geral  $G = \sum p_i$



### $\chi^2$

- Mede a significância da diferença entre os erros obtidos apenas com o "nó mãe" e com os "nós filhos"

## Estratégias para evitar sobreaprendizagem

### Pruning de árvores (pós-pruning)

- Eliminar as folhas que provocam erros no conjunto de treino
- Eliminar folhas até que o erro no treino seja semelhante ao erro no teste
- Evitar crescimento demasiado
  - Avaliar significância dos nós
- Fazer backtracking, usar sempre todos os dados, etc, etc...

## Bibliografia

- Mitchell, TM: 1997, *Machine Learning*, McGraw-Hill
- Langley, P: 1996, *Elements of Machine Learning*, Morgan and Kaufmann Publishers.
- Breiman, L., J. H. Friedman, R. A. Olsen and C. J. Stone (1984). Classification and Regression Trees, Chapman & Hall, pp 358.