



UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE CIÊNCIAS MATEMÁTICAS E DE COMPUTAÇÃO

Departamento de Ciências de Computação

<http://www.icmc.usp.br>

SCC-630 - Capítulo 9

Indução de Árvores de Decisão

João Luís Garcia Rosa¹

¹Departamento de Ciências de Computação
Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo - São Carlos

joaoluis@icmc.usp.br

2011

Agradecimento

Agradeço à Profa. Maria Carolina Monard, que gentilmente permitiu que eu usasse seus slides [1] para preparação deste capítulo.

Sumário

1 Indução de Árvores de Decisão

Material do Prof. José Augusto Baranauskas

Os próximos 52 slides contêm material do Prof. José Augusto Baranauskas, do Departamento de Física e Matemática - FFCLRP-USP, com atualização da Profa. Maria Carolina Monard.



Indução de Árvores de Decisão



- O objetivo desta aula é fornecer conceitos básicos sobre indução de árvores de decisão

Motivação

- Dados pares $(x, f(x))$, inferir $f(\cdot)$

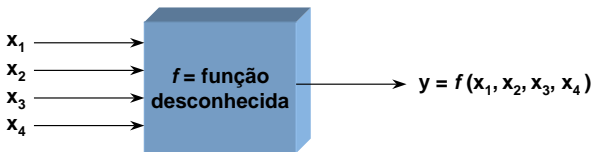
x	f(x)
1	1
2	4
3	9
4	16
5	?

Dada uma amostra finita de exemplos, é freqüentemente impossível determinar a verdadeira função $f(\cdot)$

Abordagem: Encontrar uma **hipótese** (**modelo**) h utilizando os exemplos de treinamento tal que essa **hipótese** seja “boa” para prever exemplos futuros, i.e.

$$h(x) \sim f(x)$$

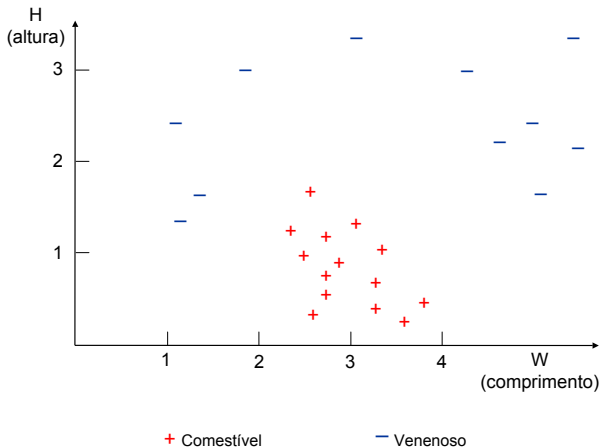
Motivação



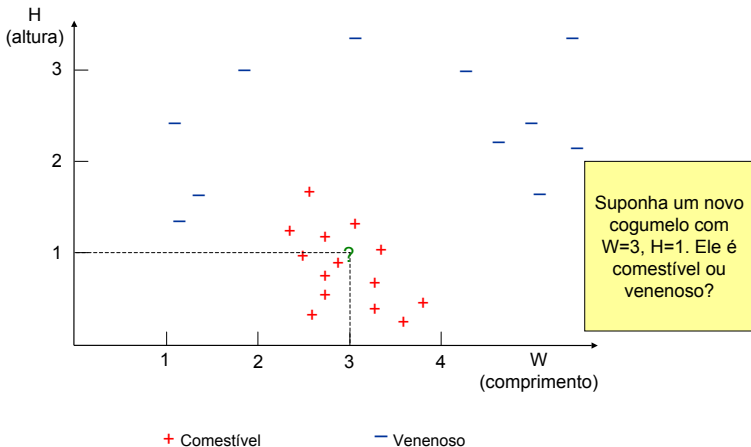
Exemplo	X_1	X_2	X_3	X_4	Y
Z_1	0	1	1	0	0
Z_2	0	0	0	0	0
Z_3	0	0	1	1	1
Z_4	1	0	0	1	1
Z_5	0	1	1	0	0
Z_6	1	1	0	0	0
Z_7	0	1	0	1	0

□ $f: (X_1 \times X_2 \times X_3 \times X_4) \rightarrow Y$

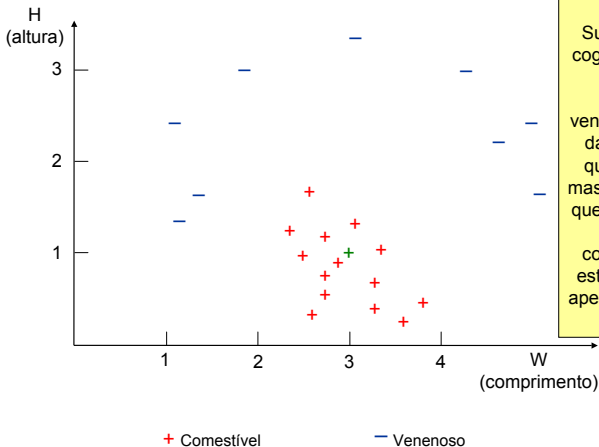
Exemplo: Cogumelos Comestíveis x Venenosos



Exemplo: Cogumelos Comestíveis x Venenosos

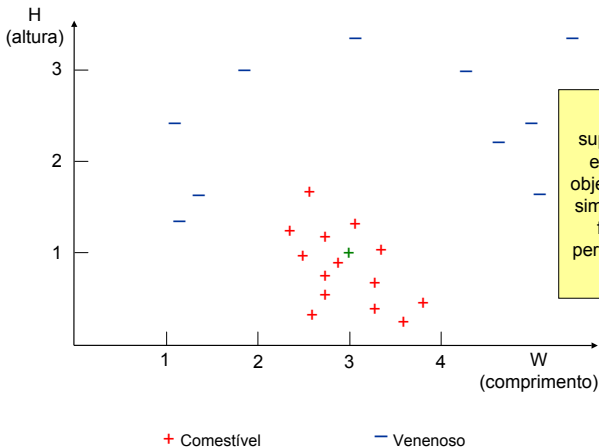


Exemplo: Cogumelos Comestíveis x Venenosos



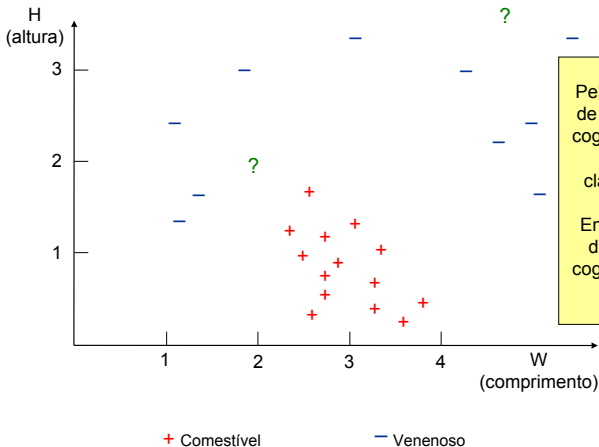
Suponha um novo cogumelo com $W=3$, $H=1$. Ele é comestível ou venenoso? A maioria das pessoas diria que é comestível, mas não há garantias que o cogumelo seja realmente comestível. Assim esta classificação é apenas uma *hipótese*

Exemplo: Cogumelos Comestíveis x Venenosos



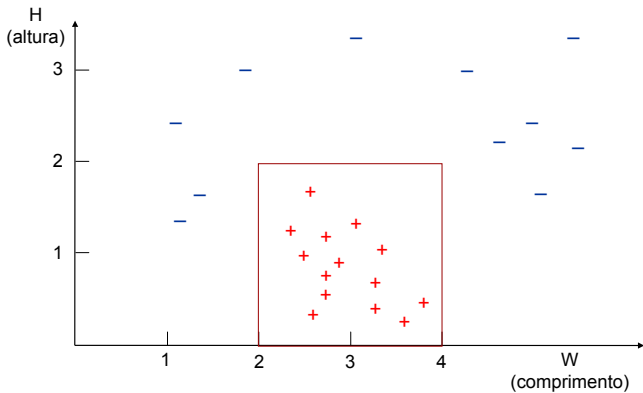
Em geral, a suposição principal em AM é que os objetos que parecem similares de alguma forma também pertencem à mesma classe

Exemplo: Cogumelos Comestíveis x Venenosos



Pelo mesmo motivo de similaridade, um cogumelo com $W=5$, $H=4$ seria classificado como venenoso. Entretanto, é difícil decidir sobre um cogumelo com $W=2$, $H=2$.

Exemplo: Cogumelos Comestíveis x Venenosos



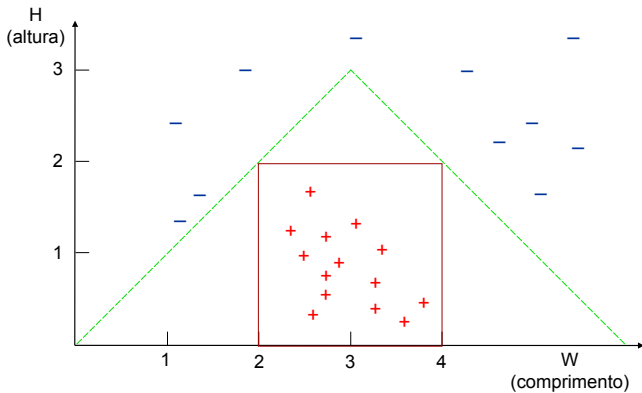
Hipótese 1:

if $2 < W$ and $W < 4$ and $H < 2$
then comestível
else venenoso

+ Comestível

- Venenoso

Exemplo: Cogumelos Comestíveis x Venenosos



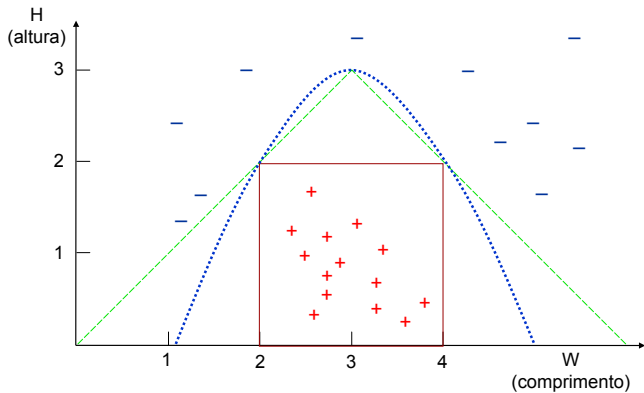
Hipótese 1:
if $2 < W$ and $W < 4$ and $H < 2$
then comestível
else venenoso

Hipótese 2:
if $H > W$
then venenoso
else if $H > 6 - W$
then venenoso
else comestível

+ Comestível

- Venenoso

Exemplo: Cogumelos Comestíveis x Venenosos



Hipótese 1:

if $2 < W$ and $W < 4$ and $H < 2$
then comestível
else venenoso

Hipótese 2:

if $H > W$
then venenoso
else if $H > 6 - W$
then venenoso
else comestível

Hipótese 3:

if $H < 3 - (W - 3)^2$
then comestível
else venenoso

+ Comestível

- Venenoso

Sistemas de Aprendizizado de Máquina – **Árvore de Decisão**

<i>Modos de Aprendizado</i>	<i>Paradigmas de Aprendizado</i>	<i>Linguagens de Descrição</i>	<i>Formas de Aprendizado</i>
<input type="checkbox"/> <u>Supervisionado</u> <input type="checkbox"/> Não Supervisionado <input type="checkbox"/> Semi Supervisionado	<input type="checkbox"/> <u>Simbólico</u> <input type="checkbox"/> Estatístico <input type="checkbox"/> Baseado em Exemplos (<i>Instance-Based</i>) <input type="checkbox"/> Conexionista <input type="checkbox"/> Genético	<input type="checkbox"/> <u>Instâncias ou Exemplos</u> <input type="checkbox"/> <u>Hipótese expressa por uma Árvore de Decisão</u> <input type="checkbox"/> Teoria de Domínio ou Conhecimento de Fundo	<input type="checkbox"/> Incremental <input type="checkbox"/> <u>Não Incremental</u>

Histórico

- ❑ 1960's
 - 1966: Hunt e colegas em psicologia usaram métodos de busca exaustiva em árvores de decisão para modelar o aprendizado de conceitos humanos
- ❑ 1970's
 - 1977: Breiman, Friedman, e colegas em estatística desenvolveram *Classification And Regression Trees* (CART)
 - 1979: Primeiro trabalho de Quinlan com proto-ID3 (*Induction of Decision Trees*)
- ❑ 1980's
 - 1984: primeira publicação em massa do software CART (presente atualmente em vários produtos comerciais)
 - 1986: Artigo de Quinlan sobre ID3
 - Variedade de melhorias: tratamento de ruído, atributos contínuos, atributos com valores desconhecidos, árvores oblíquas (não paralelas aos eixos), etc
- ❑ 1990's
 - 1993: Algoritmo atualizado de Quinlan: C4.5 (release 8)
 - Maior poder, heurísticas de controle de *overfitting* (C5.0, etc.); combinando ADs

TDIDT

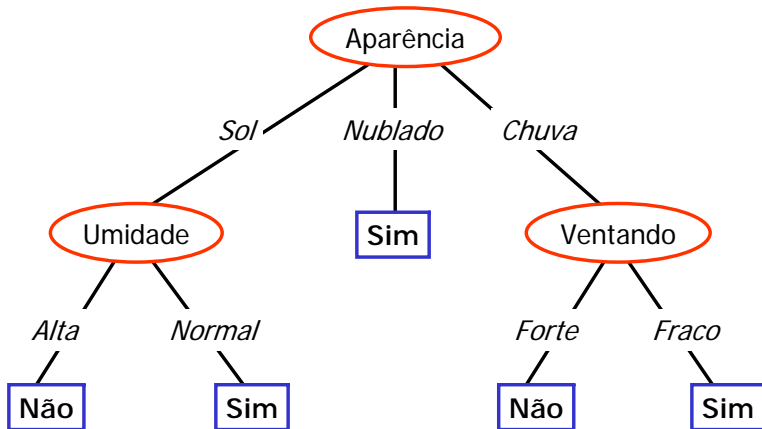
- ❑ Os algoritmos de classificação cujo conhecimento adquirido é representado como Árvore de Decisão (AD) pertencem a família TDIDT (*Top Down Induction of Decision Trees*)
- ❑ Árvore de Decisão: estrutura recursiva definida como:
 - um nó folha que indica uma classe, ou
 - um nó de decisão que contém um teste sobre o valor de um atributo. Cada resultado do teste leva a uma sub-árvore. Cada sub-árvore tem a mesma estrutura da árvore

AD para Jogar Tênis

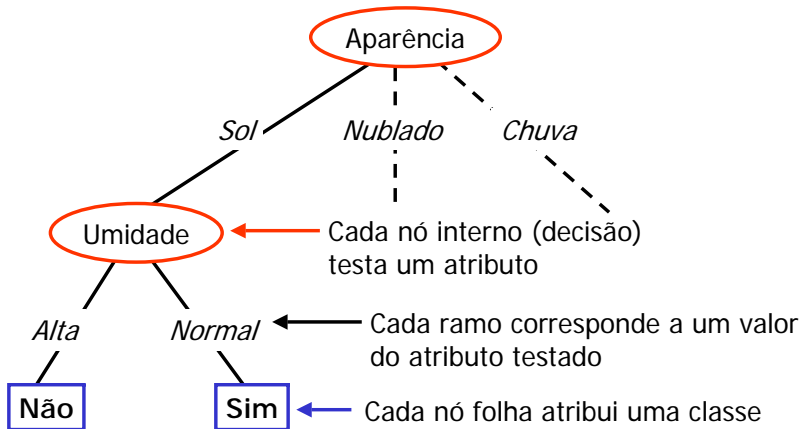
□ Atributos:

- Aparência: *Sol, Nublado, Chuva*
- Umidade: *Alta, Normal*
- Ventando: *Forte, Fraco*
- Temperatura: *Quente, Média, Fria*
- Classe (Conceito Alvo) – jogar tênis: *Sim, Não*

AD para Jogar Tênis



AD para Jogar Tênis



AD para Jogar Tênis

Aparência Temperatura Umidade Ventando Jogar Tênis

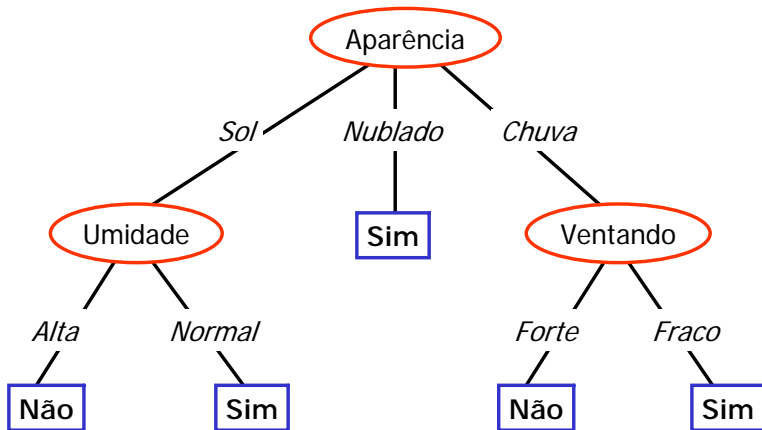
Sol

Quente

Alta

Fraco

?



AD para Jogar Tênis

Aparência Temperatura Umidade Ventando Jogar Tênis

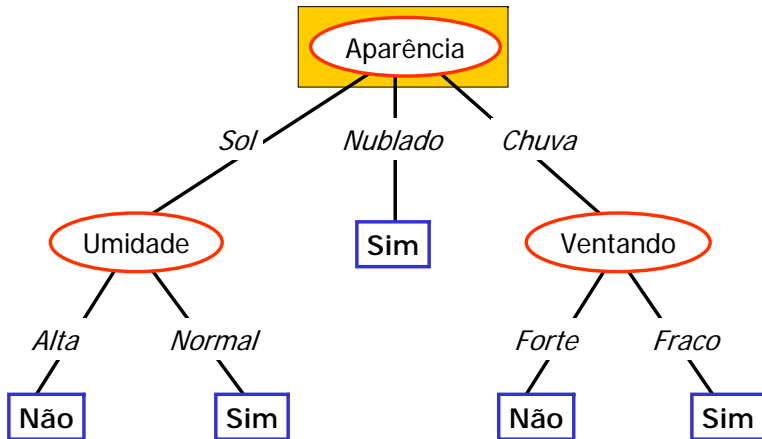
Sol

Quente

Alta

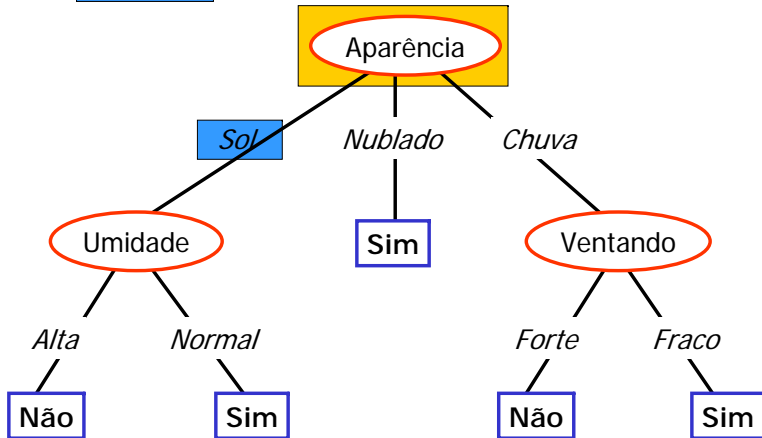
Fraco

?



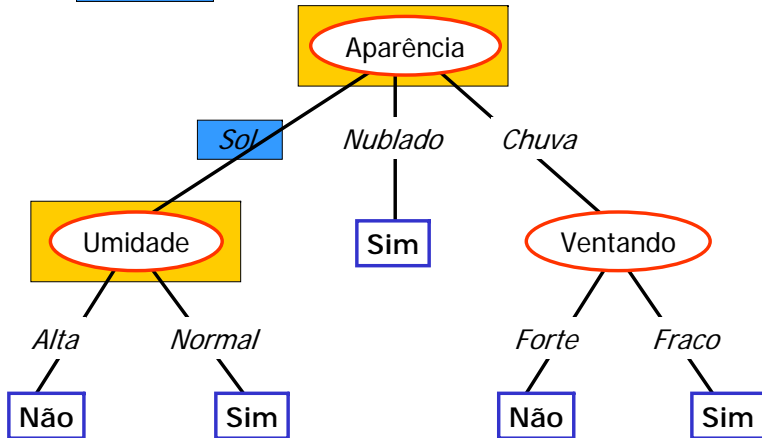
AD para Jogar Tênis

Aparência Temperatura Umidade Ventando Jogar Tênis
Sol Quente Alta Fraco ?



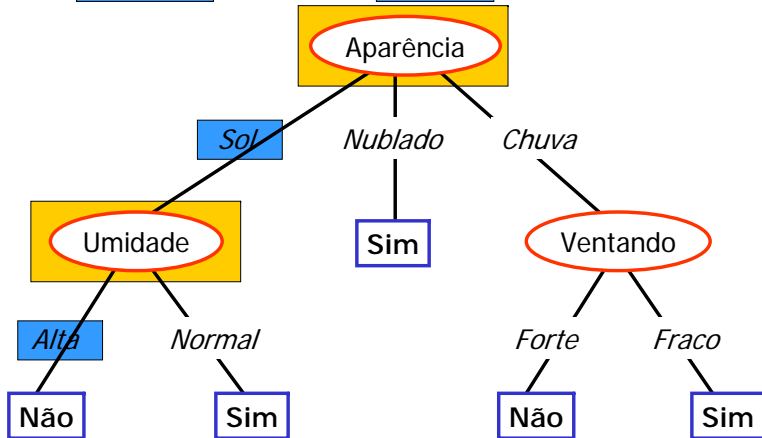
AD para Jogar Tênis

Aparência Temperatura Umidade Ventando Jogar Tênis
Sol Quente Alta Fraco ?



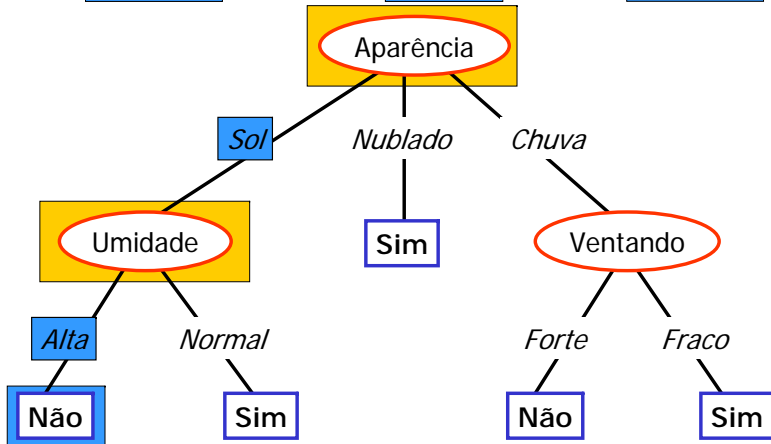
AD para Jogar Tênis

Aparência Temperatura Umidade Ventando Jogar Tênis
Sol Quente Alta Fraco ?

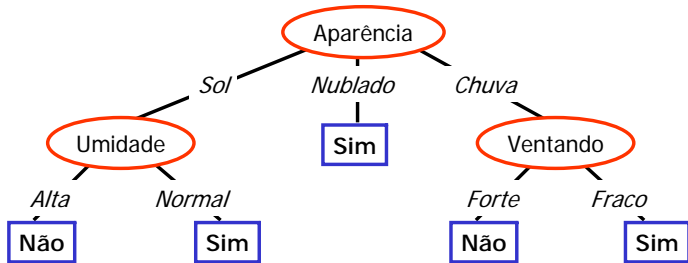


AD para Jogar Tênis

Aparência Temperatura Umidade Ventando Jogar Tênis
Sol Quente Alta Fraco Não



ADs Representam Disjunções de Conjunções



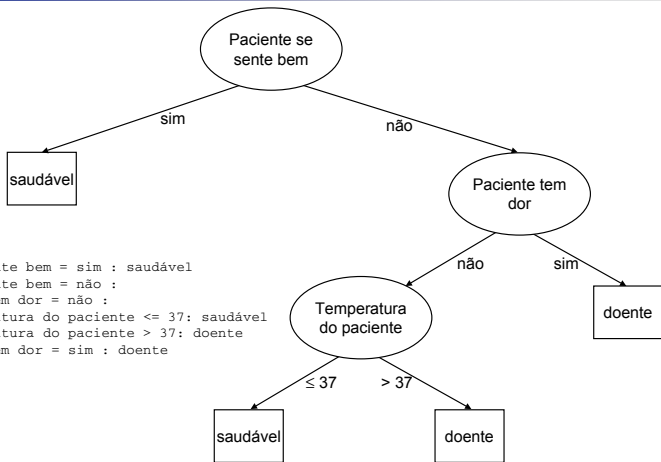
$(\text{Aparência}=\text{Sol} \wedge \text{Umidade}=\text{Normal}) \vee$
 $(\text{Aparência}=\text{Nublado}) \vee$
 $(\text{Aparência}=\text{Chuva} \wedge \text{Ventando}=\text{Fraco})$

Sim

$(\text{Aparência}=\text{Sol} \wedge \text{Umidade}=\text{Alta}) \vee$
 $(\text{Aparência}=\text{Chuva} \wedge \text{Ventando}=\text{Forte})$

Não

Exemplo: Árvore de Decisão

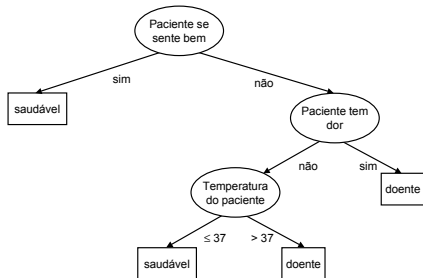


```
Paciente se sente bem = sim : saudável
Paciente se sente bem = não :
...Paciente tem dor = não :
...Temperatura do paciente ≤ 37: saudável
 : Temperatura do paciente > 37: doente
Paciente tem dor = sim : doente
```

Representação da AD como um Conjunto de Regras

- ❑ Uma árvore pode ser representada como um conjunto de regras
- ❑ Cada regra começa na raiz da árvore e caminha para baixo, em direção às folhas
 - Cada nó de decisão acrescenta um teste às premissas (condições) da regra
 - O nó folha representa a conclusão da regra

Representação da AD como um Conjunto de Regras

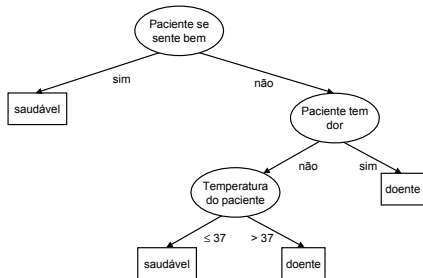


```
if Paciente se sente bem = sim then
  classe = saudável
else
  if Paciente tem dor = não
    if Temperatura do paciente ≤ 37 then
      classe = saudável
    else {Temperatura do Paciente > 37}
      classe = doente
    end if
  else {Paciente tem dor = sim}
    classe = doente
  end if
end if
```

Representação da AD como um Conjunto de Regras Disjuntas

- ❑ As regras representadas por uma árvore de decisão são disjuntas
- ❑ Assim, elas podem ser escritas como regras **separadas**, começando pela raiz, e, conseqüentemente, o *else* não é necessário

Representação da AD como um Conjunto de Regras Disjuntas



```
if Paciente se sente bem = sim then
  classe = saudável
end if
```

```
if Paciente se sente bem = não
  and Paciente tem dor = não
  and Temperatura do paciente ≤ 37 then
  classe = saudável
end if
```

```
if Paciente se sente bem = não
  and Paciente tem dor = não
  and Temperatura do paciente > 37 then
  classe = doente
end if
```

```
if Paciente se sente bem = não
  and Paciente tem dor = sim then
  classe = doente
end if
```

Algoritmo TDIDT

- ❑ Seja T um conjunto de exemplos de treinamento com classes $\{C_1, C_2, \dots, C_k\}$. Há três possibilidades:
 - 1) T contém um ou mais exemplos, todos pertencendo a uma mesma classe C_j ; a árvore de decisão para T é uma folha identificando a classe C_j
 - 2) T não contém exemplos: a árvore de decisão é novamente uma folha, mas a classe associada com a folha deve ser determinada por alguma informação além de T . Por exemplo, a folha pode ser escolhida de acordo com algum conhecimento do domínio, tal como a classe majoritária. C4.5 utiliza a classe mais freqüente do nó pai deste nó (folha)
 - 3) T contém exemplos que pertencem a uma mistura de classes: nesta situação a idéia é refinar T em subconjuntos que são (ou aparentam ser) coleções de exemplos de uma única classe. Um teste é escolhido, baseado em um único atributo, com resultados mutuamente exclusivos. Sejam os possíveis resultados do teste denotados por $\{O_1, O_2, \dots, O_r\}$. T é então particionado em subconjuntos T_1, T_2, \dots, T_r , nos quais cada T_i contém todos os exemplos em T que possuem como resultado daquele teste o valor O_i . A árvore de decisão para T consiste em um nó (interno) identificado pelo teste escolhido e uma aresta para cada um dos resultados possíveis. Para cada partição, pode-se exigir que cada T_i contenha um número mínimo de exemplos, evitando partições com poucos exemplos. O default de C4.5 é de 2 exemplos
- ❑ Os passos 1, 2 e 3 são aplicados recursivamente para cada subconjunto de exemplos de treinamento de forma que, em cada nó, as arestas levam para as sub-árvores construídas a partir do subconjunto de exemplos T_i
- ❑ Após a construção da árvore de decisão, a poda pode ser realizada para melhorar sua capacidade de generalização

Classificando Novos Exemplos

- ❑ Uma AD pode ser usada para classificar novos exemplos (nunca vistos)
- ❑ A partir da raiz basta descer através dos nós de decisão até encontrar um nó folha: a classe correspondente a esse nó folha é a classe do novo exemplo
- ❑ Um exemplo (sem valores desconhecidos) é classificado apenas por uma regra (sub-árvore)

Exemplo (adaptado de Quinlan)

- ❑ Neste exemplo, vamos considerar um conjunto de 15 exemplos que contém medições diárias sobre condições meteorológicas
- ❑ Atributos
 - **aparência**: “sol”, “nublado” ou “chuva”
 - **temperatura**: temperatura em graus Celsius
 - **umidade**: umidade relativa do ar
 - **ventando**: “sim” ou “não”
- ❑ Cada exemplo foi rotulado (*classe*) com “bom” se nas condições meteorológicas daquele dia é aconselhável fazer uma viagem à fazenda e “ruim”, caso contrário

O Conjunto de Dados “Viagem”

Classe

Exemplo	Aparência	Temperatura	Umidade	Ventando	Viajar
E ₁	sol	25	72	sim	bom
E ₂	sol	28	91	sim	ruim
E ₃	sol	22	70	não	bom
E ₄	sol	23	95	não	ruim
E ₅	sol	30	85	não	ruim
E ₆	nublado	23	90	sim	bom
E ₇	nublado	29	78	não	bom
E ₈	nublado	19	65	sim	ruim
E ₉	nublado	26	75	não	bom
E ₁₀	nublado	20	87	sim	bom
E ₁₁	chuva	22	95	não	bom
E ₁₂	chuva	19	70	sim	ruim
E ₁₃	chuva	23	80	sim	ruim
E ₁₄	chuva	25	81	não	bom
E ₁₅	chuva	21	80	não	bom

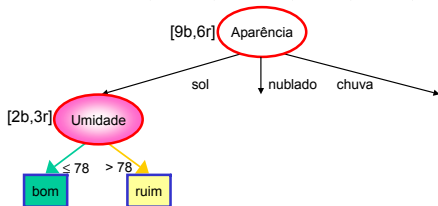
Escolhendo “Aparência” para Particionar

Exemplo		Temperatura	Umidade	Ventando	Viajar
E ₁	sol	25	72	sim	bom
E ₂	sol	28	91	sim	ruim
E ₃	sol	22	70	não	bom
E ₄	sol	23	95	não	ruim
E ₅	sol	30	85	não	ruim
E ₆	nublado	23	90	sim	bom
E ₇	nublado	29	78	não	bom
E ₈	nublado	19	65	sim	ruim
E ₉	nublado	26	75	não	bom
E ₁₀	nublado	20	87	sim	bom
E ₁₁	chuva	22	95	não	bom
E ₁₂	chuva	19	70	sim	ruim
E ₁₃	chuva	23	80	sim	ruim
E ₁₄	chuva	25	81	não	bom
E ₁₅	chuva	21	80	não	bom



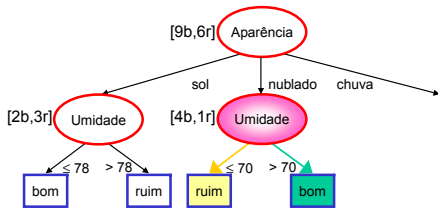
Escolhendo “Umidade” para Particionar “Aparência=sol”

Exemplo	Aparência	Temperatura	Ventando	Viajar
E ₁	sol	25	72	sim bom
E ₂	sol	28	91	sim ruim
E ₃	sol	22	70	não bom
E ₄	sol	23	95	não ruim
E ₅	sol	30	85	não ruim
E ₆	nublado	23	90	sim bom
E ₇	nublado	29	78	não bom
E ₈	nublado	19	65	sim ruim
E ₉	nublado	26	75	não bom
E ₁₀	nublado	20	87	sim bom
E ₁₁	chuva	22	95	não bom
E ₁₂	chuva	19	70	sim ruim
E ₁₃	chuva	23	80	sim ruim
E ₁₄	chuva	25	81	não bom
E ₁₅	chuva	21	80	não bom



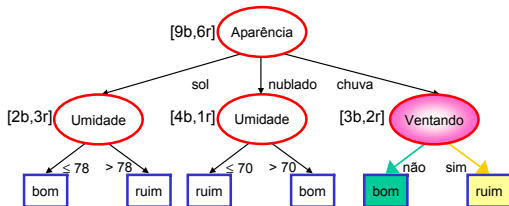
Escolhendo “Umidade” para Particionar “Aparência=nublado”

Exemplo	Aparência	Temperatura	Ventando	Viajar	
E ₁	sol	25	72	sim	bom
E ₂	sol	28	91	sim	ruim
E ₃	sol	22	70	não	bom
E ₄	sol	23	95	não	ruim
E ₅	sol	30	85	não	ruim
E ₆	nublado	23	90	sim	bom
E ₇	nublado	29	78	não	bom
E ₈	nublado	19	65	sim	ruim
E ₉	nublado	26	75	não	bom
E ₁₀	nublado	20	87	sim	bom
E ₁₁	chuva	22	95	não	bom
E ₁₂	chuva	19	70	sim	ruim
E ₁₃	chuva	23	80	sim	ruim
E ₁₄	chuva	25	81	não	bom
E ₁₅	chuva	21	80	não	bom



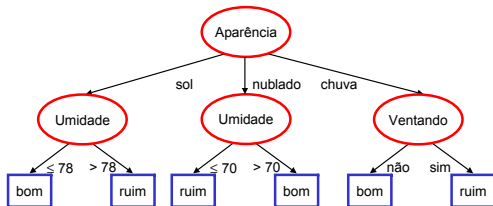
Escolhendo “Ventando” para Particionar “Aparência=chuva”

Exemplo	Aparência	Temperatura	Umidade	Viajar	Viajar
E ₁	sol	25	72	sim	bom
E ₂	sol	28	91	sim	ruim
E ₃	sol	22	70	não	bom
E ₄	sol	23	95	não	ruim
E ₅	sol	30	85	não	ruim
E ₆	nublado	23	90	sim	bom
E ₇	nublado	29	78	não	bom
E ₈	nublado	19	65	sim	ruim
E ₉	nublado	26	75	não	bom
E ₁₀	nublado	20	87	sim	bom
E ₁₁	chuva	22	95	não	bom
E ₁₂	chuva	19	70	sim	ruim
E ₁₃	chuva	23	80	sim	ruim
E ₁₄	chuva	25	81	não	bom
E ₁₅	chuva	21	80	não	bom



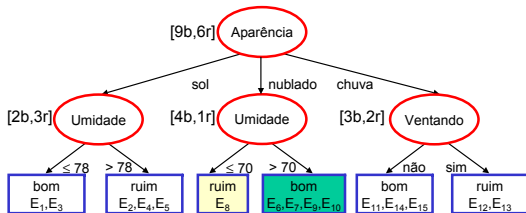
Árvore de Decisão Induzida (sem poda)

Exemplo	Aparência	Temperatura	Umidade	Ventando	Viajar
E ₁	sol	25	72	sim	bom
E ₂	sol	28	91	sim	ruim
E ₃	sol	22	70	não	bom
E ₄	sol	23	95	não	ruim
E ₅	sol	30	85	não	ruim
E ₆	nublado	23	90	sim	bom
E ₇	nublado	29	78	não	bom
E ₈	nublado	19	65	sim	ruim
E ₉	nublado	26	75	não	bom
E ₁₀	nublado	20	87	sim	bom
E ₁₁	chuva	22	95	não	bom
E ₁₂	chuva	19	70	sim	ruim
E ₁₃	chuva	23	80	sim	ruim
E ₁₄	chuva	25	81	não	bom
E ₁₅	chuva	21	80	não	bom



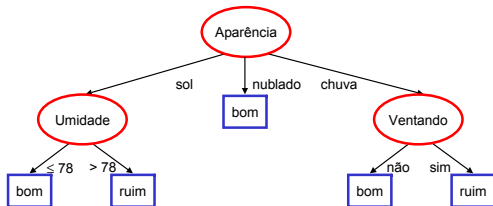
Árvore de Decisão Induzida (sem poda)

Exemplo	Aparência	Temperatura	Umidade	Ventando	Viajar
E ₁	sol	25	72	sim	bom
E ₂	sol	28	91	sim	ruim
E ₃	sol	22	70	não	bom
E ₄	sol	23	95	não	ruim
E ₅	sol	30	85	não	ruim
E ₆	nublado	23	90	sim	bom
E ₇	nublado	29	78	não	bom
E ₈	nublado	19	65	sim	ruim
E ₉	nublado	26	75	não	bom
E ₁₀	nublado	20	87	sim	bom
E ₁₁	chuva	22	95	não	bom
E ₁₂	chuva	19	70	sim	ruim
E ₁₃	chuva	23	80	sim	ruim
E ₁₄	chuva	25	81	não	bom
E ₁₅	chuva	21	80	não	bom



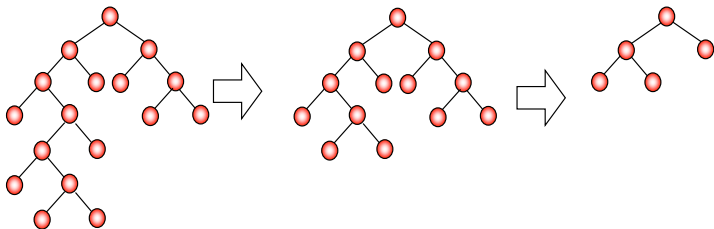
Árvore de Decisão Induzida (podada)

Exemplo	Aparência	Temperatura	Umidade	Ventando	Viajar
E ₁	sol	25	72	sim	bom
E ₂	sol	28	91	sim	ruim
E ₃	sol	22	70	não	bom
E ₄	sol	23	95	não	ruim
E ₅	sol	30	85	não	ruim
E ₆	nublado	23	90	sim	bom
E ₇	nublado	29	78	não	bom
E ₈	nublado	19	65	sim	ruim
E ₉	nublado	26	75	não	bom
E ₁₀	nublado	20	87	sim	bom
E ₁₁	chuva	22	95	não	bom
E ₁₂	chuva	19	70	sim	ruim
E ₁₃	chuva	23	80	sim	ruim
E ₁₄	chuva	25	81	não	bom
E ₁₅	chuva	21	80	não	bom

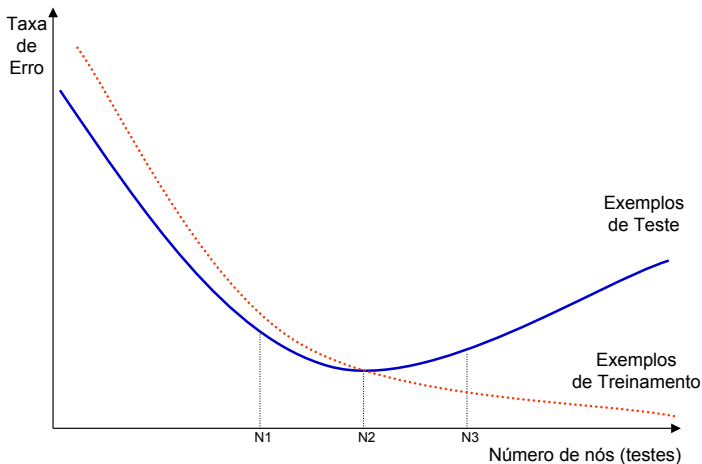


(Pós-)Poda

- Uma árvore maior é induzida de forma a superajustar os exemplos e então ela é podada até obter uma árvore menor (mais simples)
- A poda evita *overfitting*



Relação entre Tamanho da Árvore de Decisão e a Taxa de Erro



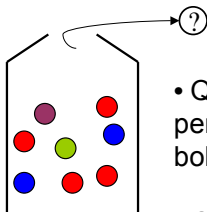
Escolha do Atributo

- ❑ A maioria dos algoritmos de construção de árvores de decisão são sem retrocesso (sem *backtracking*) ou seja, gulosos (*greedy*)
- ❑ Uma vez que um teste foi selecionado para particionar o conjunto atual de exemplos, a escolha é fixada e escolhas alternativas não são exploradas

Escolha do Atributo

- ❑ A chave para o sucesso de um algoritmo de aprendizado no qual h é expressa como uma árvore de decisão depende do critério utilizado para escolher o atributo que particiona o conjunto de exemplos em cada iteração
- ❑ Algumas possibilidades para escolher esse atributo são:
 - aleatória: seleciona qualquer atributo aleatoriamente
 - menos valores: seleciona o atributo com a menor quantidade de valores possíveis
 - mais valores: seleciona o atributo com a maior quantidade de valores possíveis
 - ganho máximo: seleciona o atributo que possui o maior ganho de informação esperado, isto é, seleciona o atributo que resultará no menor tamanho esperado das subárvores, assumindo que a raiz é o nó atual;
 - razão de ganho
 - índice Gini

Exemplo



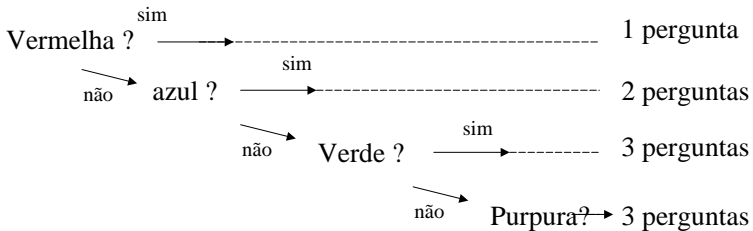
Tira uma bola aleatoriamente –

- Qual a melhor sequencia de perguntas para confirmar a cor dessa bola?
- Qual o número médio de perguntas?

8 bolas: 4 ●, 2 ●, 1 ●, 1 ●

Exemplo

- Melhor sequencia de perguntas (Codigo de Huffman)



O código de Huffman foi desenvolvido por David A. Huffman quando era aluno de PhD no MIT em 1952, e publicado no artigo: *A Method for the Construction of Minimum-Redundancy Codes*.



Professor David A. Huffman
(August 9, 1925 - October 7, 1999)

Exemplo

□ Número médio de perguntas :

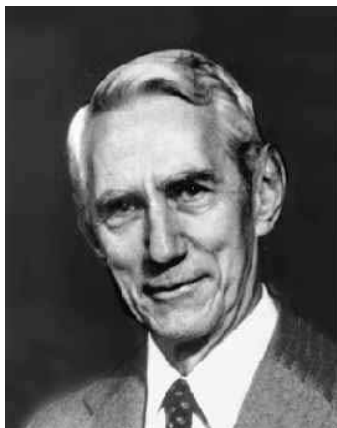
$$P(\bullet) \times 1 + P(\bullet) \times 2 + P(\bullet) \times 3 + P(\bullet) \times 3$$

$$\frac{1}{2} \times 1 + \frac{1}{4} \times 2 + \frac{1}{8} \times 3 + \frac{1}{8} \times 3 = 1.75$$

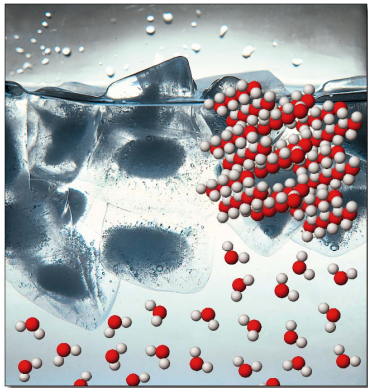
□ Entropia = $-\sum_i p_i \log_2(p_i)$

$$= \frac{1}{2} \times \log_2\left(\frac{1}{2}\right) + \frac{1}{4} \times \log_2\left(\frac{1}{4}\right) + \left(\frac{1}{8}\right) \times \log_2\left(\frac{1}{8}\right) + \left(\frac{1}{8}\right) \times \log_2\left(\frac{1}{8}\right) = 1.75 \text{ bits}$$

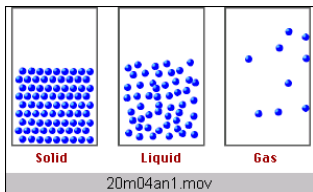
Teoria da Informação



- Claude E. Shannon (1916-2001)
- 1948: *A Mathematical Theory of Communication*
 - *A entropia mede a falta de informação (medida de desordem) de um sistema*



A entropia da
água em estado
líquido é maior
que a entropia
da água em
estado sólido (0°
C)



A entropia de

gases > líquidos > sólidos

Entropia

- ❑ Pode também ser considerada como uma medida da quantidade de informação que uma pessoa necessita para organizar seus conhecimentos e descobrir uma regra
- ❑ Quanto mais alternativas um diagnóstico possui, mais informações são necessárias para aprender ele (maior entropia)
- ❑ Se um diagnóstico não tem alternativas, não é necessária nenhuma informação (entropia = 0)

Referências I



[1] Monard, M. C.

Slides da disciplina SCC630 - Inteligência Artificial. ICMC - USP, 2010.