

ICC204 - Aprendizagem de Máquina e Mineração de Dados

Classificação

(parte 2/3)



Prof. Rafael Giusti
rgiusti@icomp.ufam.edu.br

Agenda

- Parte 1/3
 - Definições
 - Teoria das probabilidades
 - Aprendizado Bayesiano e modelos probabilísticos
- Parte 2/3
 - Modelos baseados em árvores
 - Modelos baseados em regras
- Parte 3/3
 - Classificação preguiçosa: k-NN
 - Máquina de vetores de suporte

Agenda

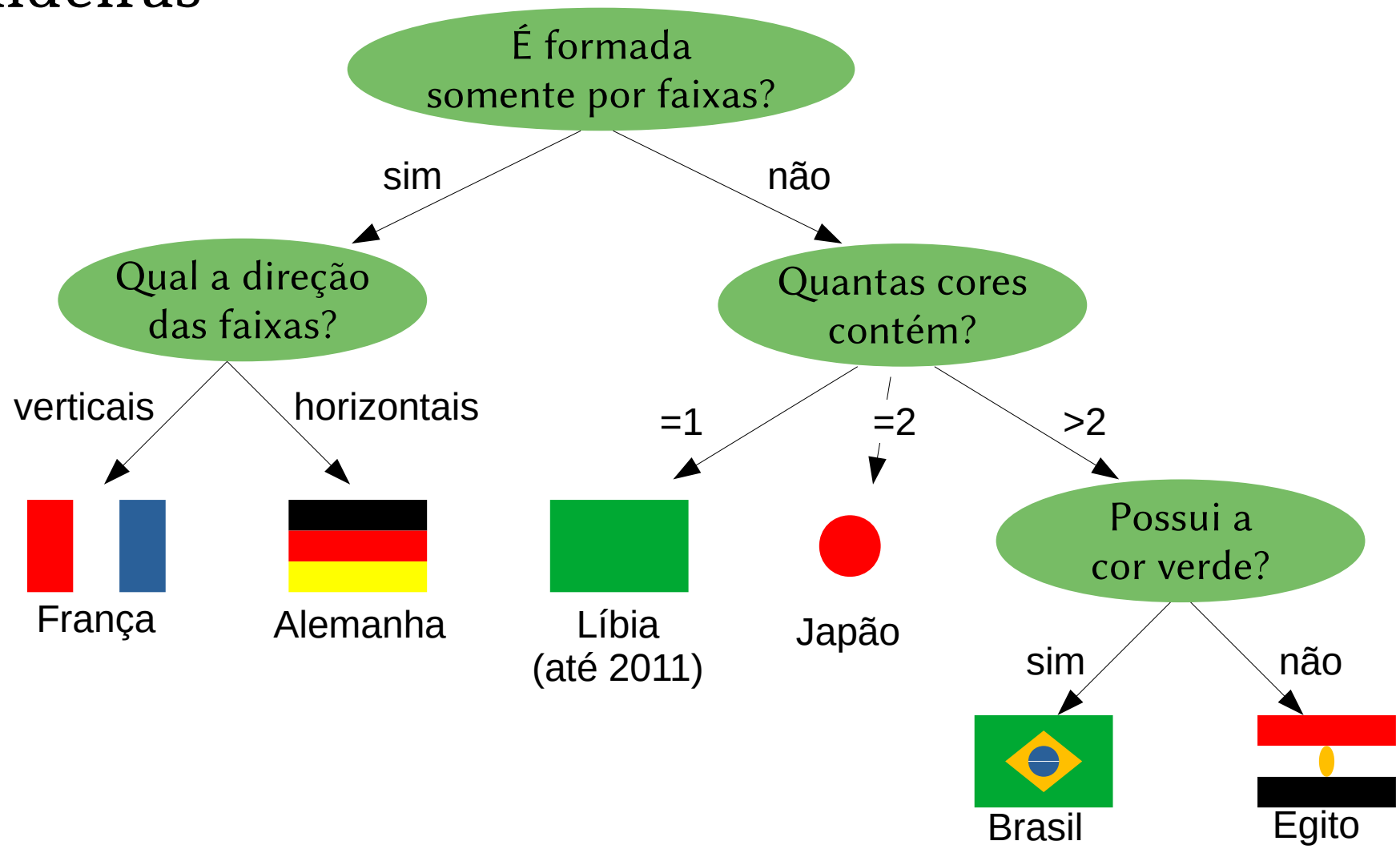
- Definições
- Teoria das probabilidades
- Aprendizado Bayesiano e modelos probabilísticos
- Modelos baseados em árvores
- Modelos baseados em regras
- Classificação preguiçosa: k-NN
- Máquina de vetores de suporte

Linguagem de árvore

- Uma árvore de decisão é um modelo no qual o conhecimento é representado através de uma **árvore n -ária**
- Cada nó **interno** da árvore faz uma pergunta sobre um atributo
- Os nós **folha** estabelecem decisões sobre a qual classe um exemplo pertence

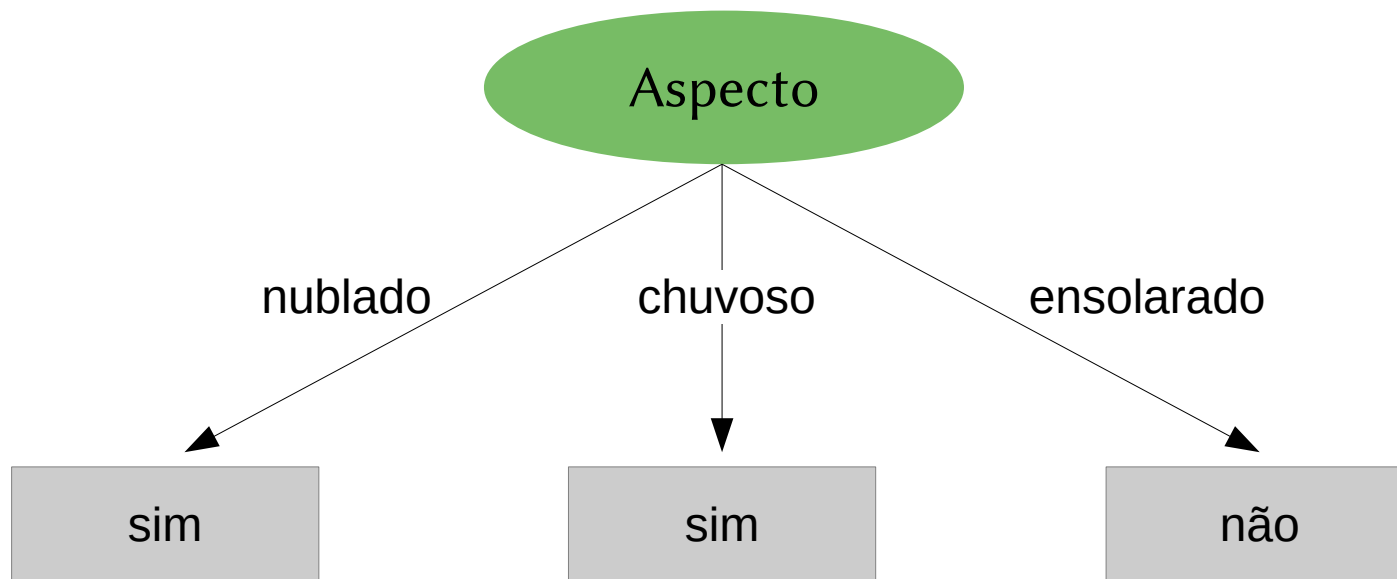
Linguagem de árvore

- Árvore de decisão para um pequeno conjunto de bandeiras



Linguagem de árvore

- Árvore de decisão de apenas dois níveis para o conjunto "jogar-tênis"

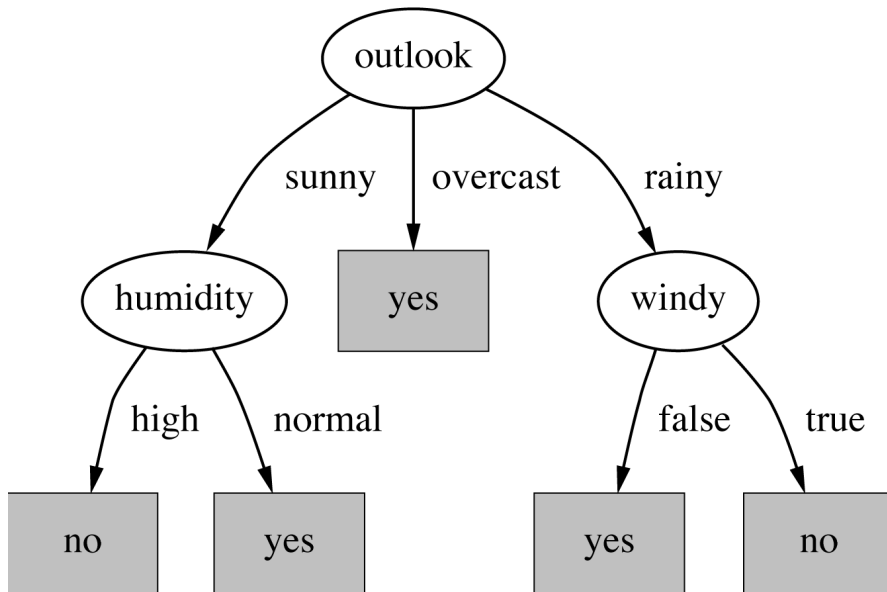


Observe que esta é apenas uma das possíveis árvores para o conjunto. Ela não é necessariamente uma boa árvore e não é a que iremos gerar seguindo o algoritmo.

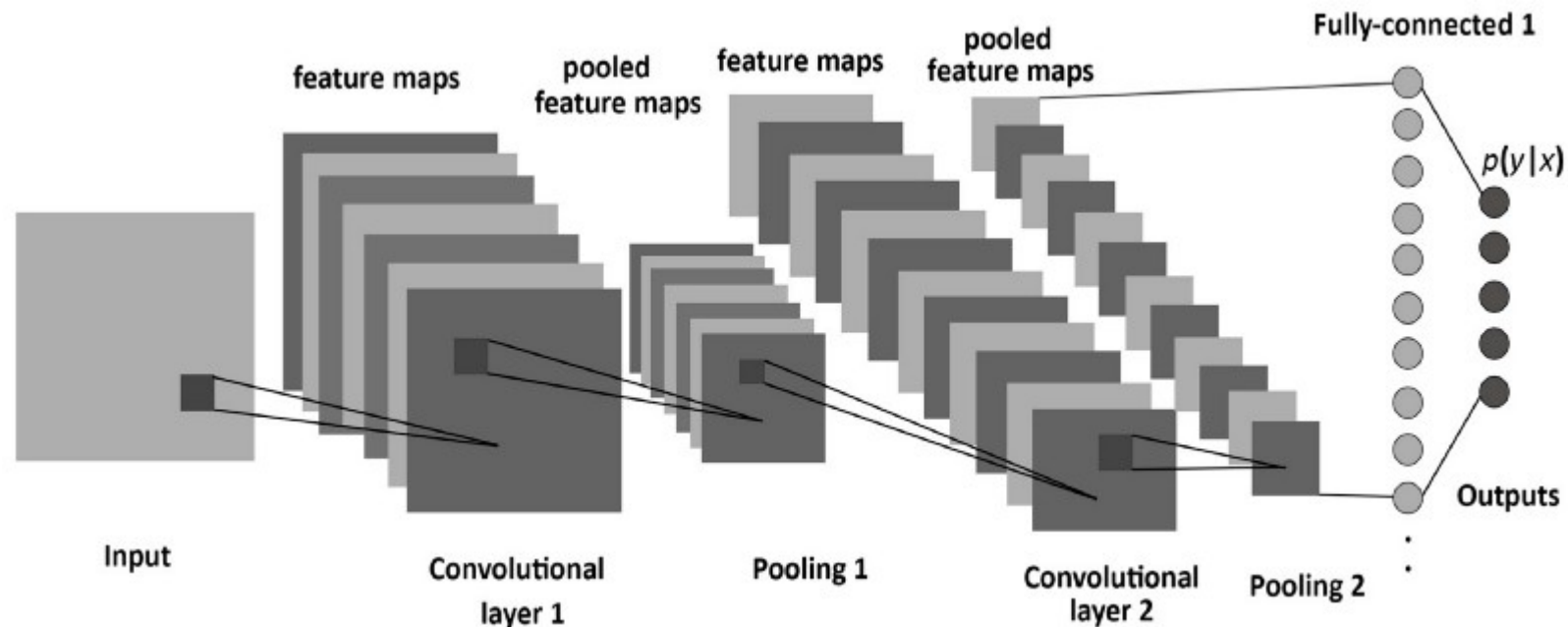
Linguagem de árvore

- A árvore de decisão é uma **representação simbólica** do conhecimento
 - A palavra "simbólica" em expressões como representação simbólica ou modelo simbólico indica que o conhecimento é representado de forma inteligível

Simbólico vs. não simbólico



Autoria: Prof^a Eulanda

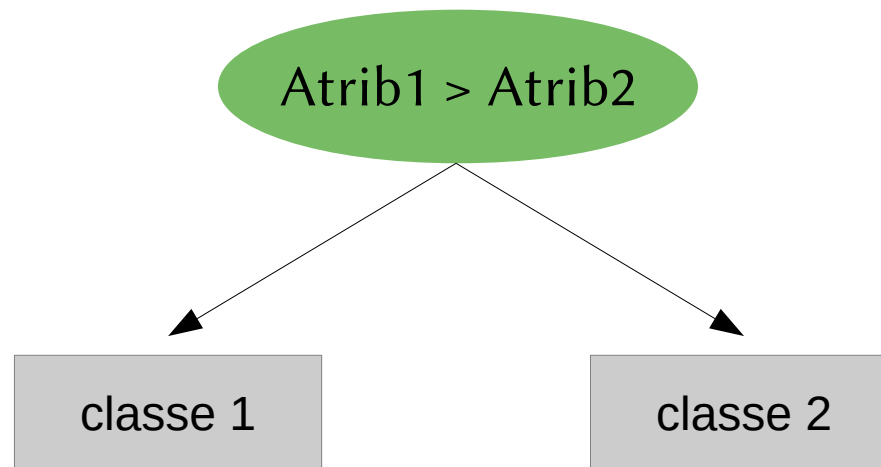


Características de árvores

- Podem ser utilizadas para atributos numéricos e categóricos
- As divisões podem ser binárias ou n -árias
- Admitem valores ausentes
 - Mas precisamos tratá-los
 - O valor ausente pode ser um valor próprio ou podemos empregar uma estratégia de substituição

Características de árvores

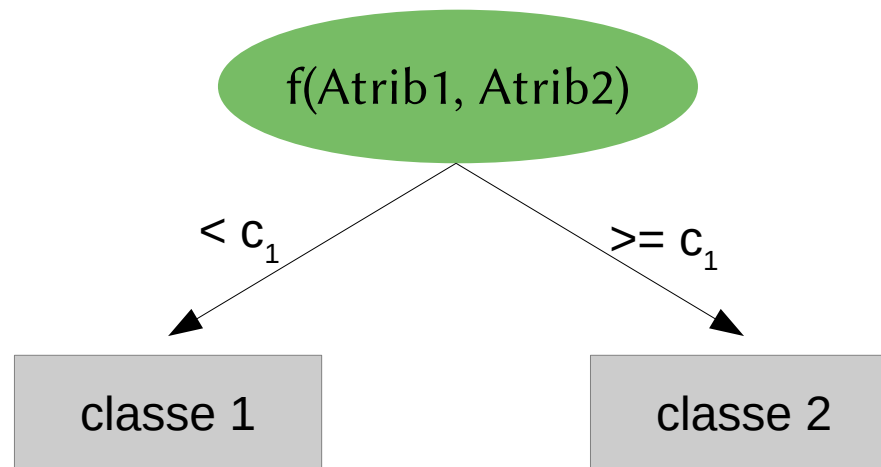
- As decisões não se restringem a comparações de um atributo contra um valor ou intervalo



Uma árvore pode promover comparações entre atributos

Características de árvores

- As decisões não se restringem a comparações de um atributo contra um valor ou intervalo



A árvore pode utilizar funções sobre os atributos

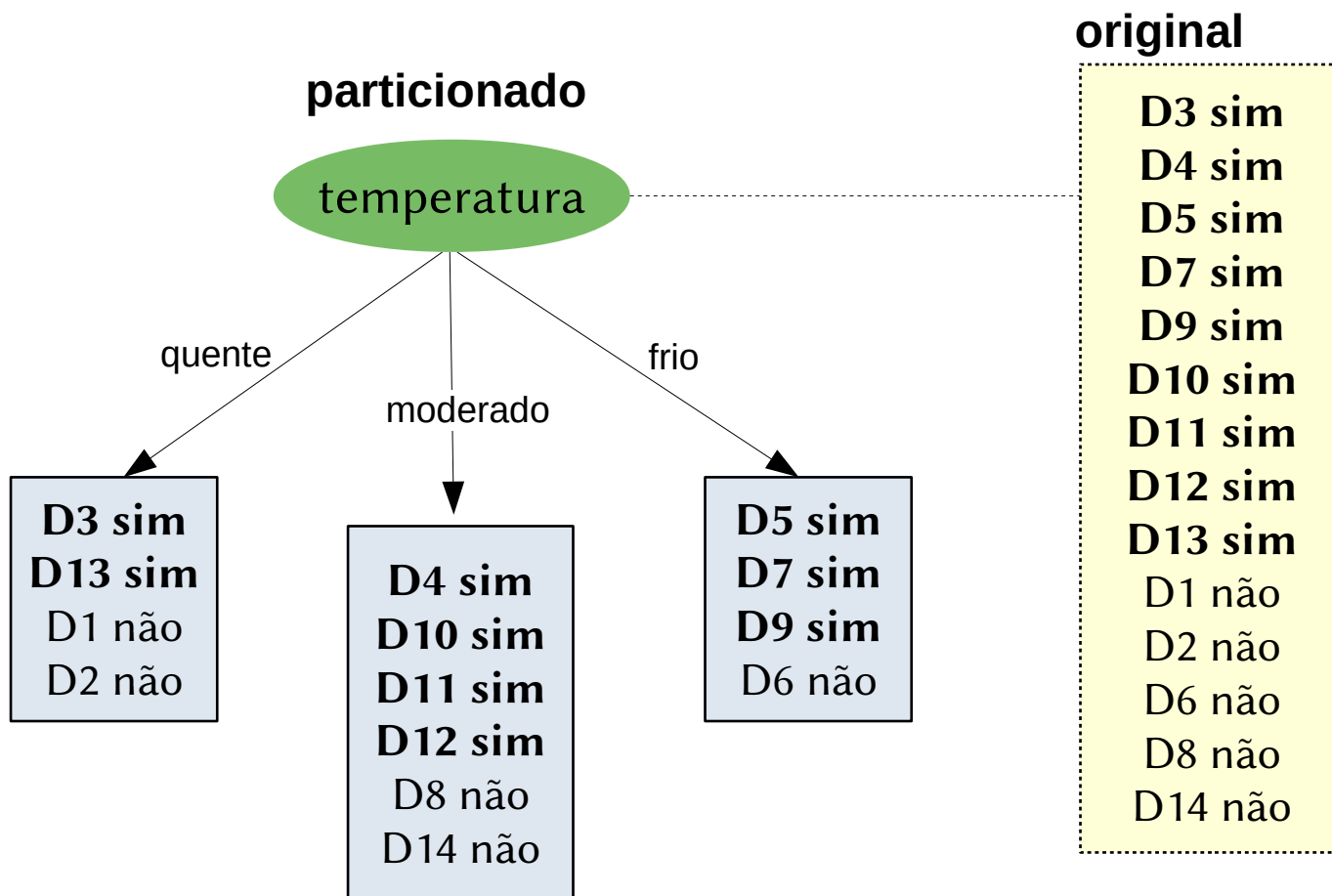
FT-Tree: os nós podem ser regressores logísticos sobre os atributos

Particionamento

- Vamos tratar de árvores que envolvem apenas um atributo por nó interno
 - Se ele for categórico, a árvore seleciona subconjuntos de exemplos que possuem aquele valor
 - Se ele for numérico, a árvore particiona o espaço fazendo cortes longitudinais sobre algum eixo

Particionamento (atributo categórico)

- Exemplo de uma árvore que divide os exemplos de acordo com o atributo temperatura

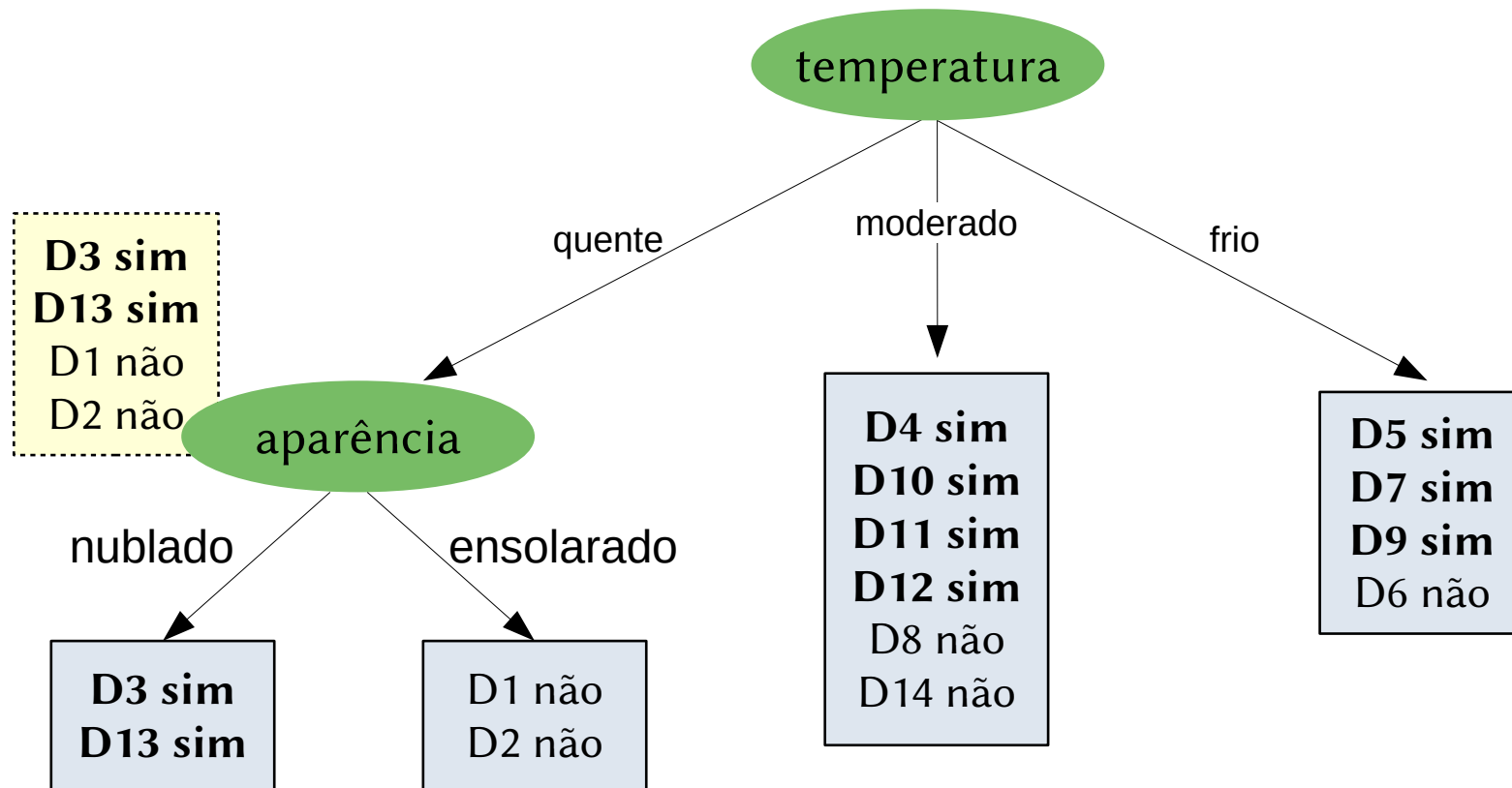


O espaço original possui, no conjunto de treinamento, 14 exemplos; 9 da classe "sim" e 6 da classe "não".

Particionando esse espaço de acordo com o atributo "temperatura", obtemos três sub-espacos. No conjunto de treinamento, esse sub-espaco possui os exemplos ilustrados nas caixas sob a árvore.

Particionamento (atributo categórico)

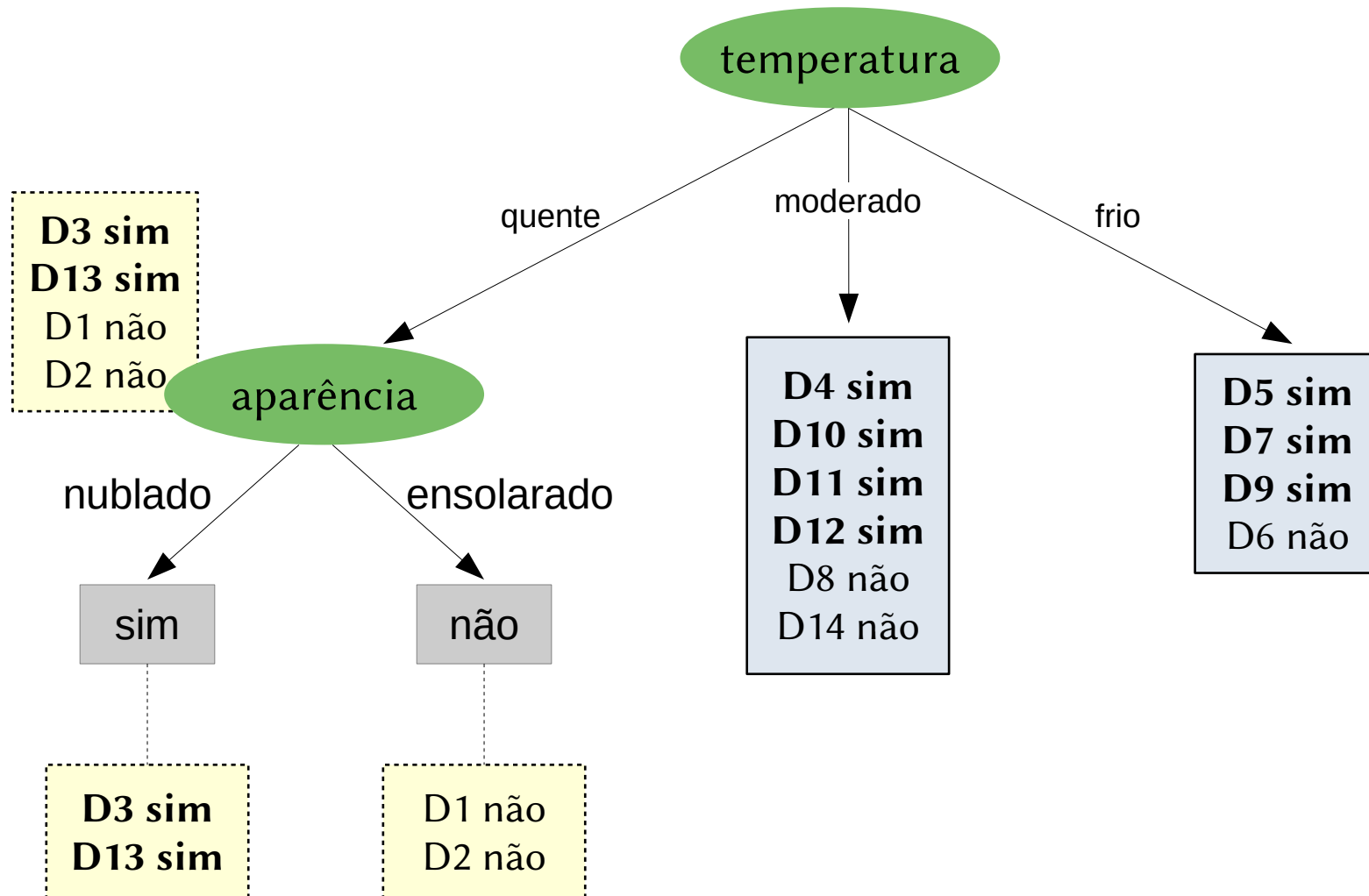
- Particionando o espaço temperatura=quente...



Note que aqui chegamos a uma separação exata das classes; poderíamos associar esses dois sub-espacos a nós folhas

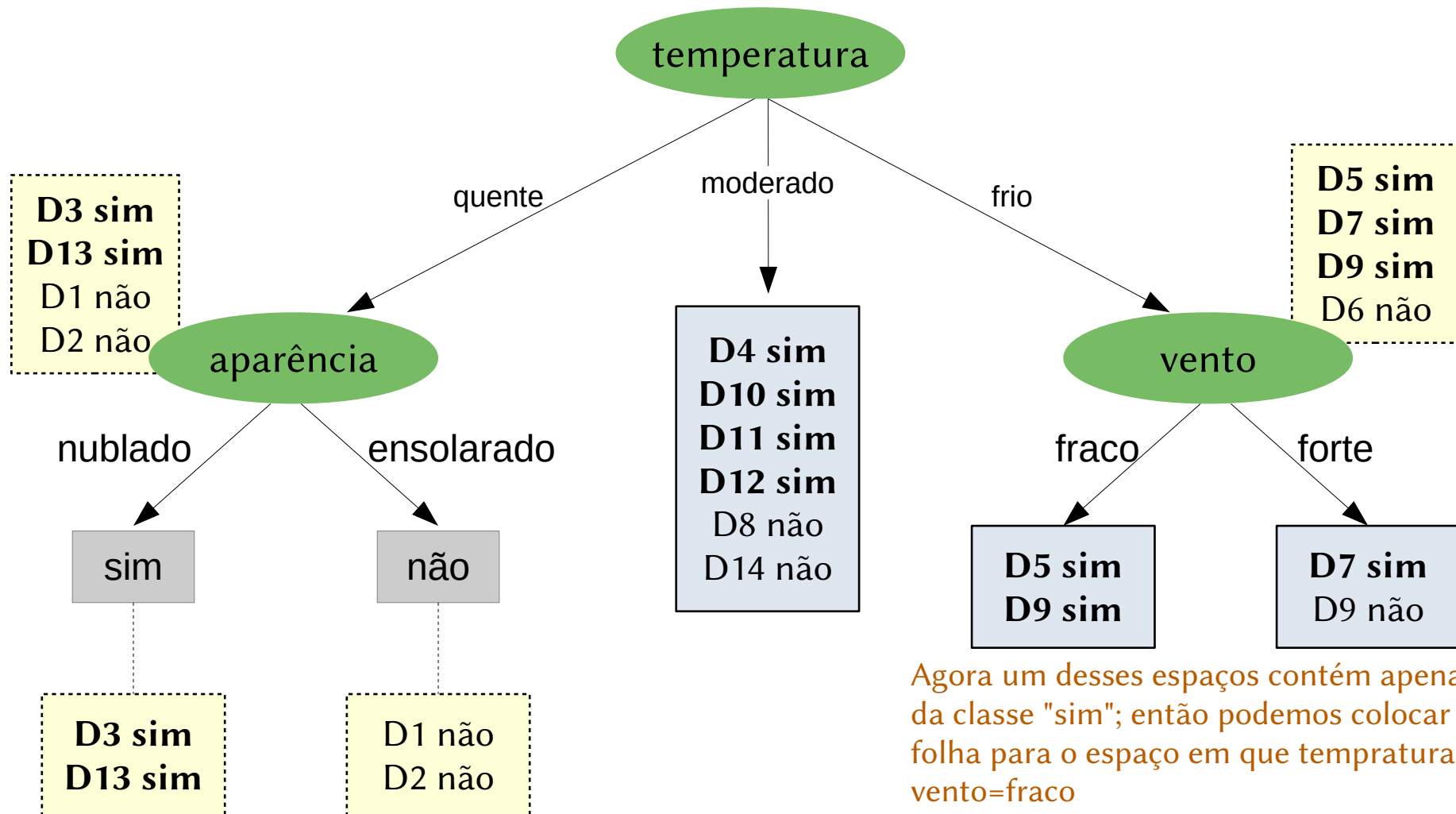
Particionamento (atributo categórico)

- Atribuindo os sub-espacos a nós folhas



Particionamento (atributo categórico)

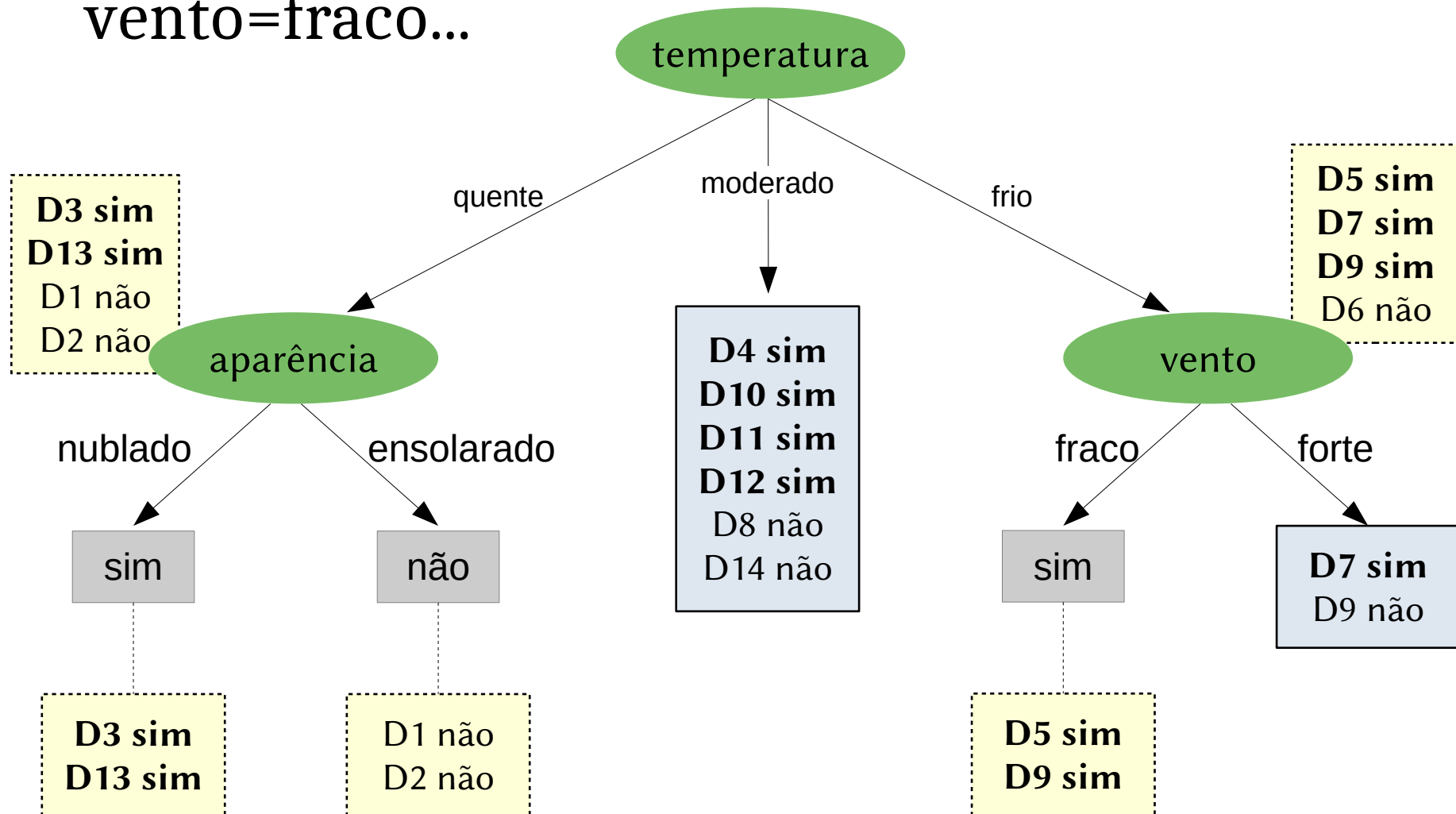
- Particionando o espaço temperatura=frio...



Agora um desses espaços contém apenas exemplos da classe "sim"; então podemos colocar um nó folha para o espaço em que temperatura=frio && vento=fraco

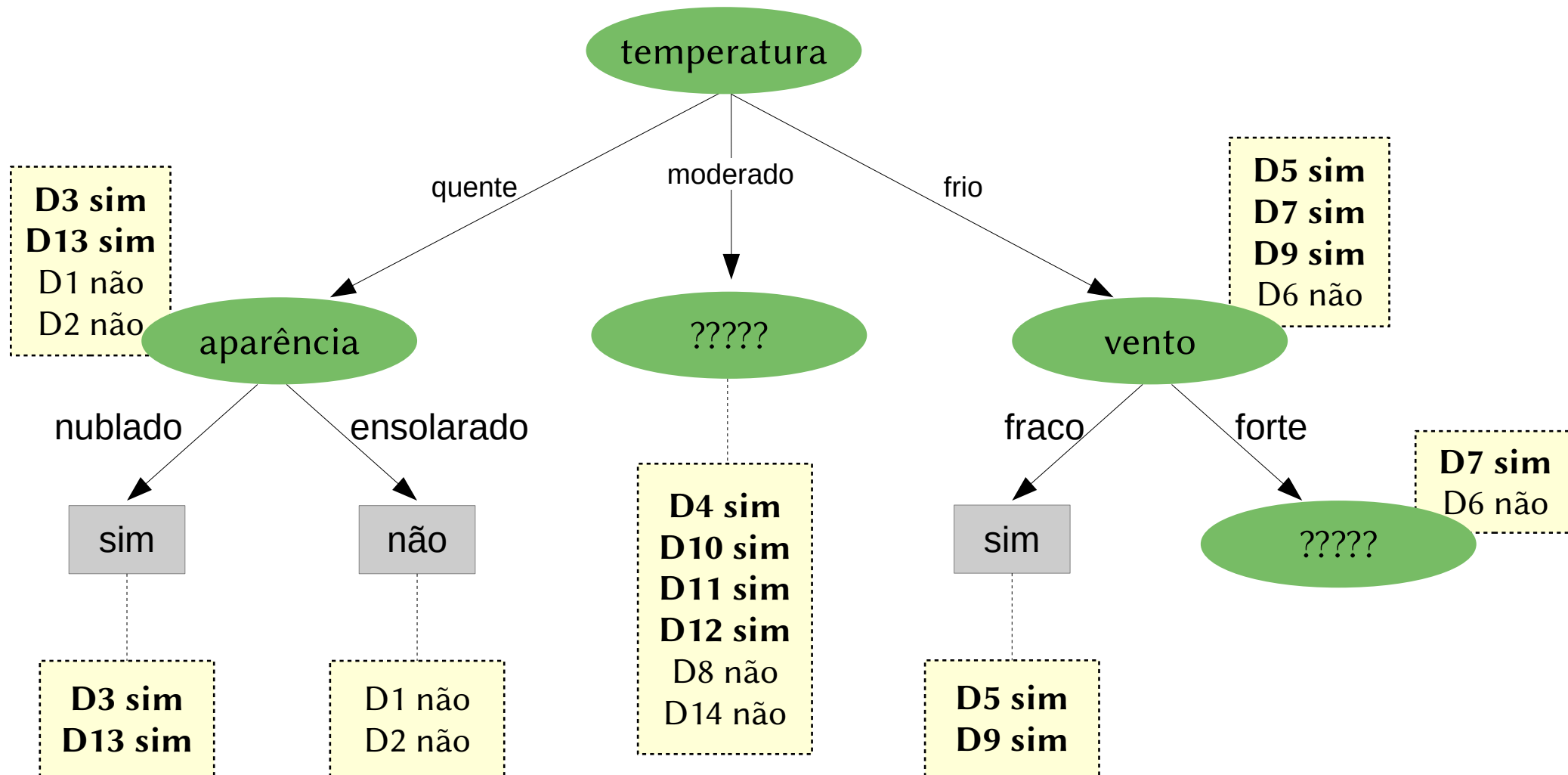
Particionamento (atributo categórico)

- Atribuindo uma classe para o sub-espço
vento=fraco...



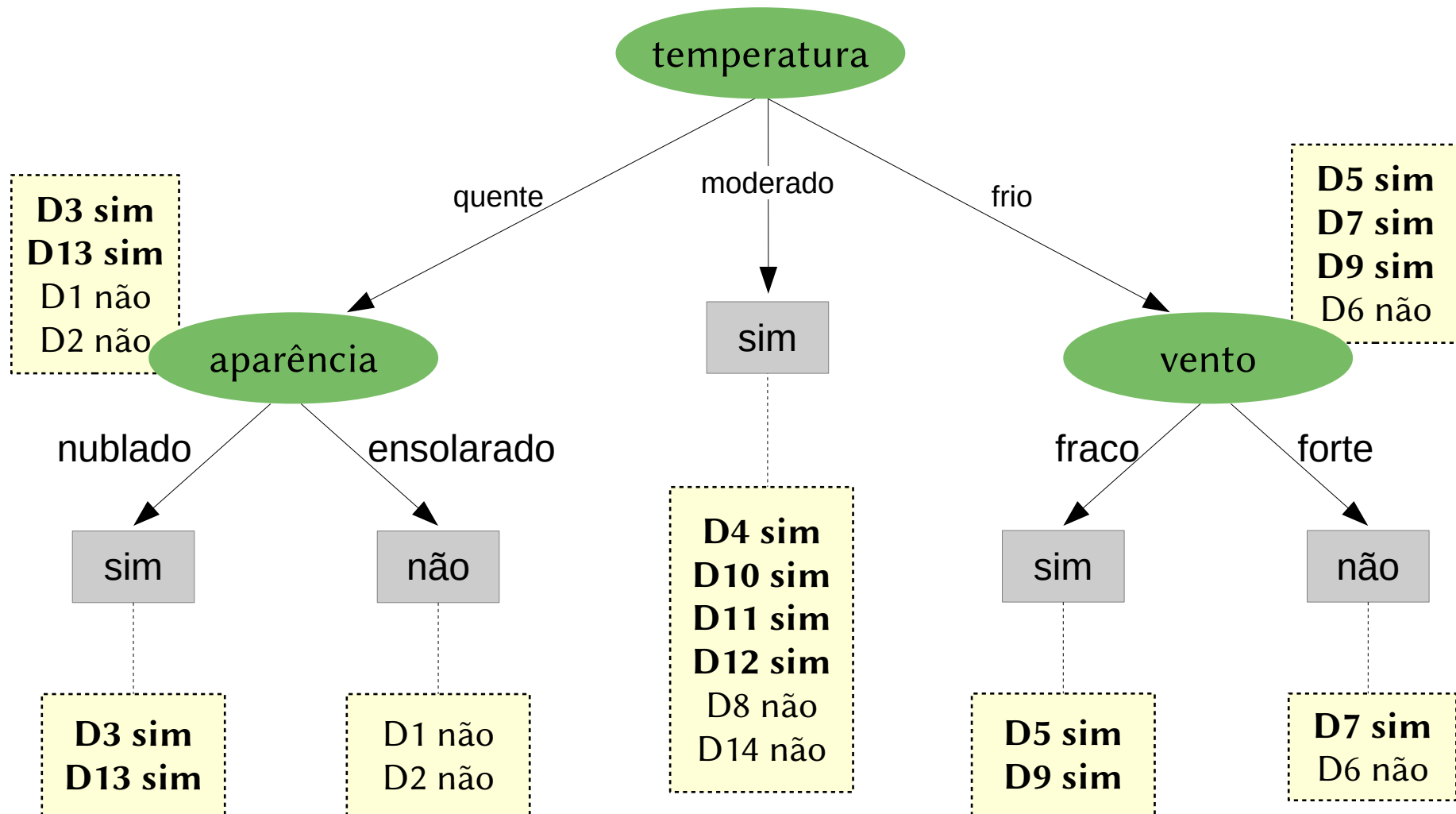
Particionamento (atributo categórico)

- E assim poderíamos seguir dividindo

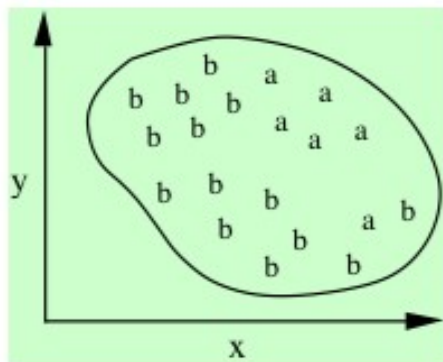


Particionamento (atributo categórico)

- Ou parar onde estamos

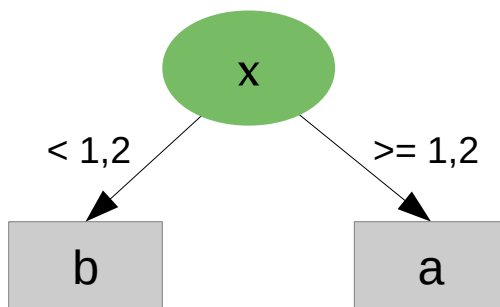
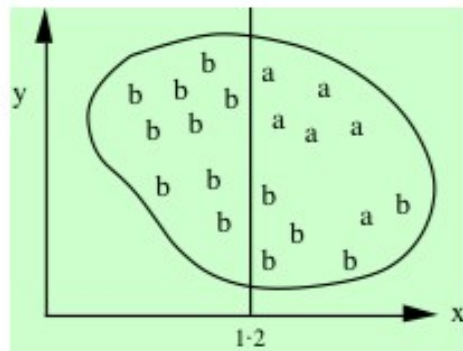


Particionamento (atributo numérico)

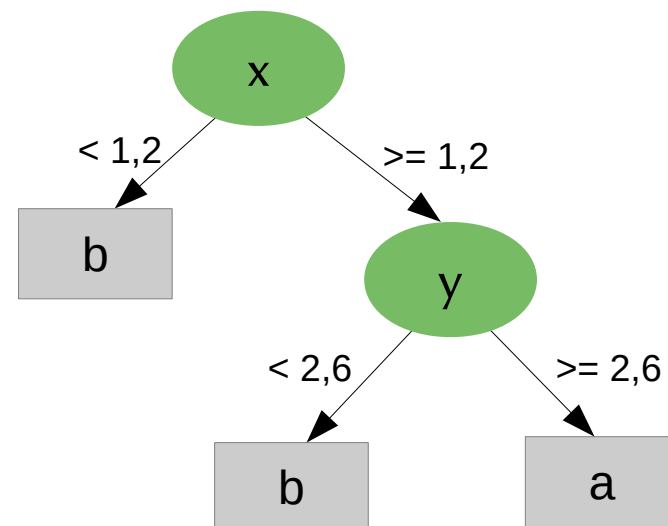
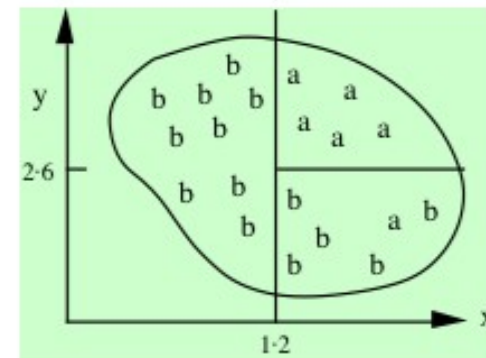


b

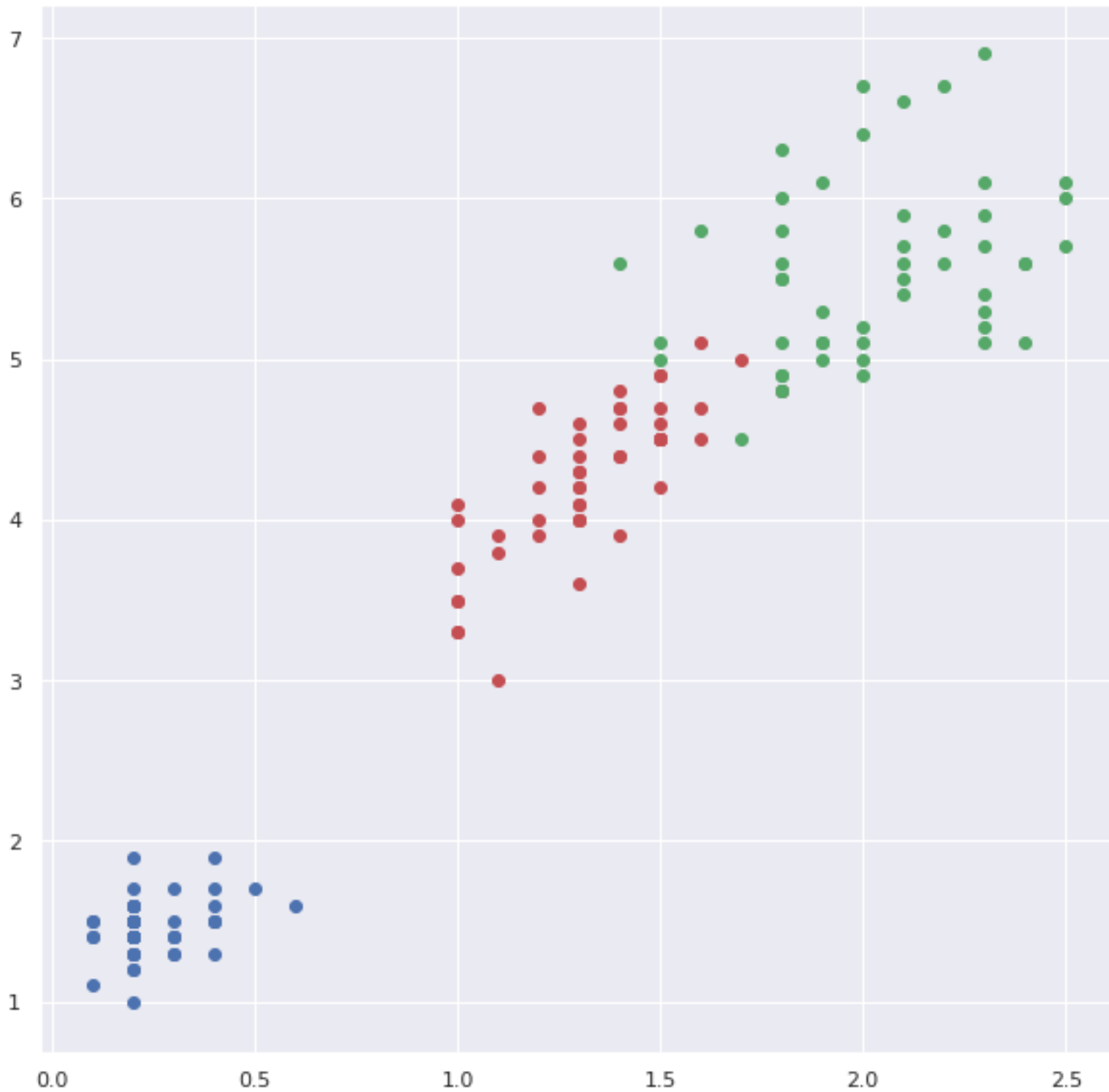
Possível árvore:
nenhuma decisão;
todos os exemplos
são classificados
como pertencentes à
classe majoritária (b)



Possível árvore: um
único nó
intermediário
promovendo um
único corte



Possível árvore: o
particionamento anterior é
refinado por um segundo
corte

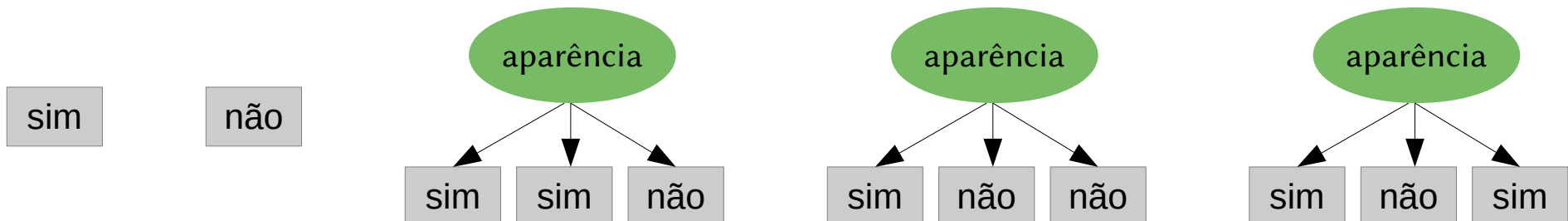


exemplo dado
em aula

exercício: particione
adequadamente com
dois cortes

Como induzir árvores?

- Mesmo para espaços de atributos relativamente pequenos, existe um grande número de árvores no espaço de modelos
 - Por exemplo, para a base *tenis*, só existem duas árvores com apenas um nó folha...
 - ...mas 16 árvores com apenas um nó interno



Um algoritmo de divisão e conquista

- Algoritmo **Induz_Árvore**(S):

T \leftarrow árvore vazia

se todos os exemplos em S são de uma mesma classe c_j :

faça a raiz de **T** ser um nó folha associado a c_j

senão:

X_{best} \leftarrow selecione o melhor atributo para particionar S

faça a raiz de **T** ser um nó interno associado a **X_{best}**

para cada valor x_i do atributo **X_{best}**:

Ss \leftarrow o sub-espaco de S no qual todos os exemplos têm
o valor **X_{best}** = x_i

St \leftarrow Induz_Árvore(Ss)

insira **St** na árvore **T** com uma aresta da raiz para **St**
associada à decisão **X_{best}** = x_i

retorne **T**

Um algoritmo de divisão e conquista

- O algoritmo Induz_Árvore é capaz de encontrar uma árvore que generaliza bem o problema
 - Condições
 - Os exemplos do conjunto de treinamento devem representar bem o espaço de atributos S
 - A função de particionamento deve fazer uma boa escolha do melhor atributo X_{best}

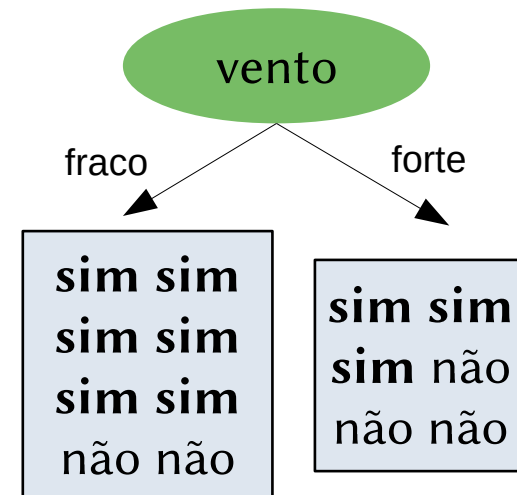
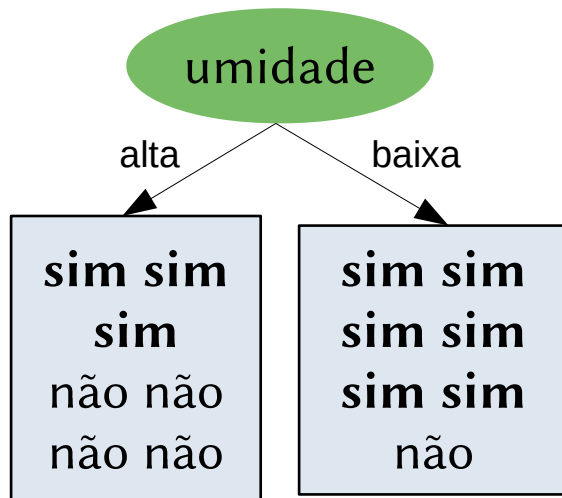
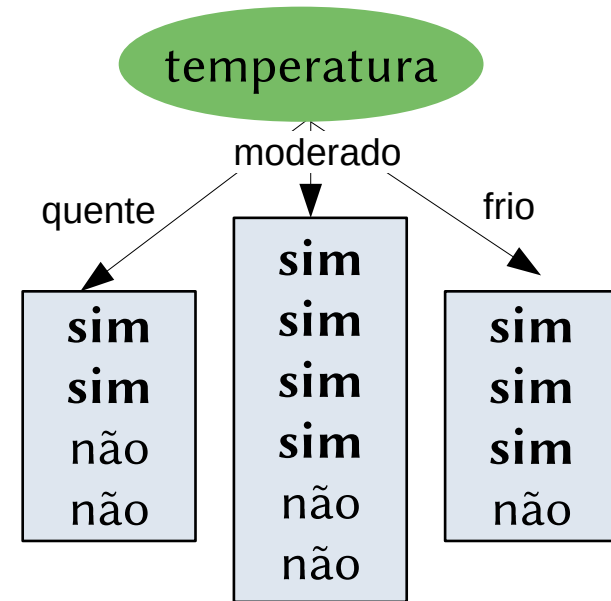
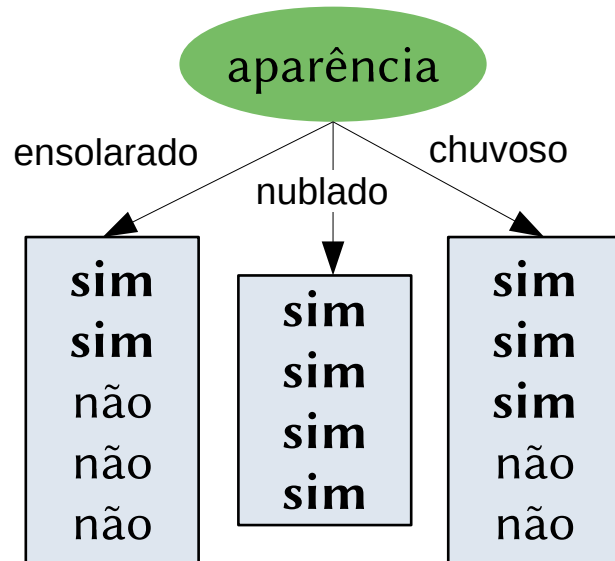
Como escolher X_{best} ?

- Primeiro, observe que o erro empírico das árvores induzidas por Induz_Árvore será sempre zero
 - Então, todas as hipóteses encontradas por esse algoritmo serão igualmente boas no conjunto de treinamento
 - Pelo princípio da navalha de Occam, se tivermos várias explicações igualmente boas para alguma coisa, em geral preferimos a mais simples

Como escolher X_{best} ?

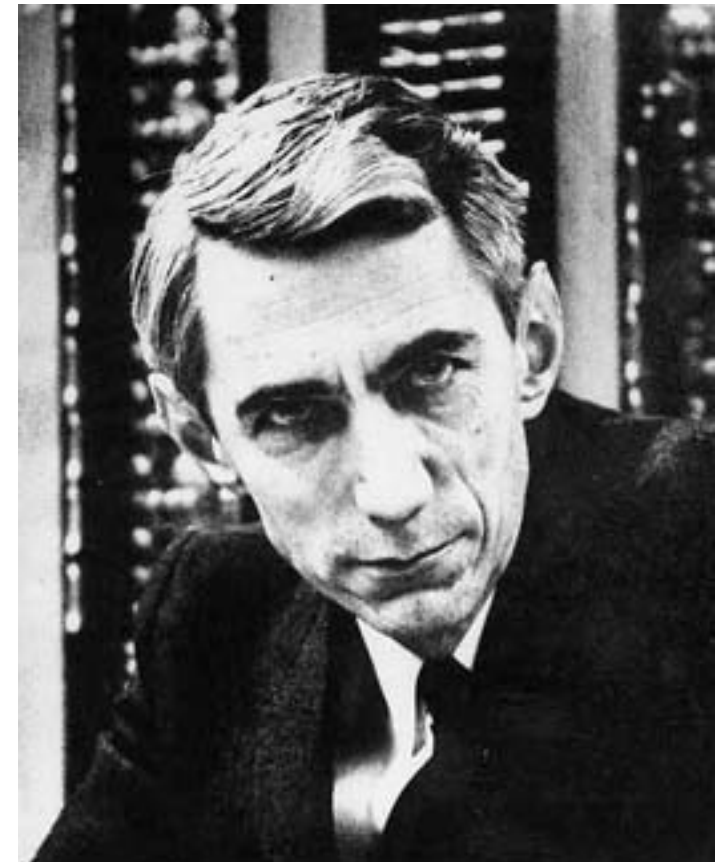
- Queremos, então, árvores com a seguinte característica
 - Altura baixa
 - Folhas homogêneas
- Essa árvore pode ser obtida se fizermos uma decisão gulosa sobre o melhor atributo para a raiz
 - Selecione sempre o atributo que produz os sub-espacos mais homogêneos em cada particionamento

Como escolher X_{best} ?



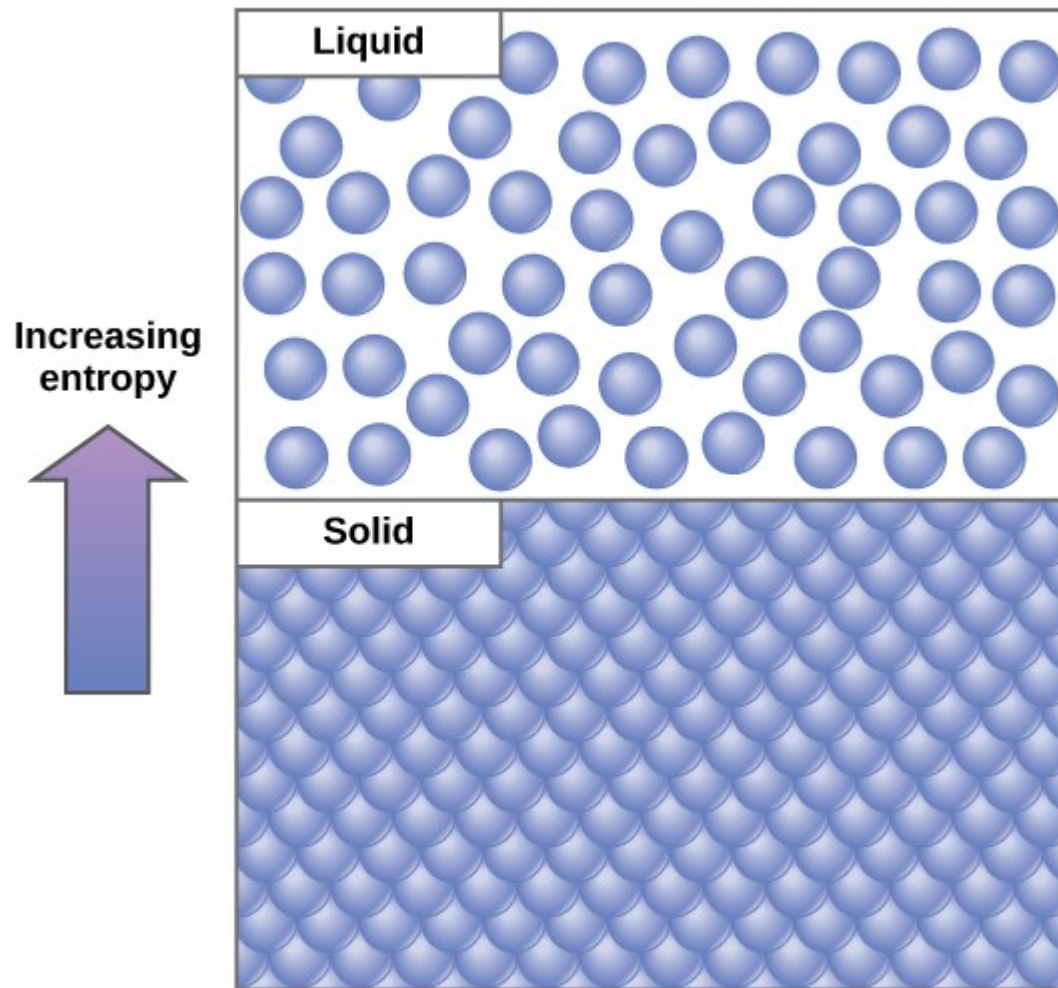
Como escolher X_{best} ?

- O melhor particionamento pode ser encontrado minimizando a **entropia** e maximizando o **ganho de informação**
- Entropia: conceito definido por Claude Shannon, "pai" da teoria da informação
 - Um valor que reflete o grau de incerteza de uma variável aleatória



Entropia (termodinâmica)

- Conceito de "desordem" em sistemas físicos



Entropia (teoria da informação)

- No contexto de teoria da informação, entropia ou, mais especificamente, **entropia de Shannon** é
 - Uma medida da quantidade de **informação média** produzida por uma fonte **aleatória**
 - Qual fonte produz mais informação média
 - Um dado de 6 lados ou uma moeda honesta?
 - Um dado viciado ou uma moeda honesta
 - Depende do quão viciado o dado é

Entropia de Shannon

- Quanto mais informação uma fonte produz, mais informação é necessária para **codificar** o seu resultado como uma **mensagem**
 - O tamanho da mensagem é medido em bits
 - Uma moeda pode ser codificada com mensagens de apenas 1 bit
 - Cara \rightarrow "0"
 - Coroa \rightarrow "1"

Entropia de Shannon

- O resultado de um dado de 4 faces requer pelo menos 2 bits
 - Exemplo de codificação
 - Face 1 \rightarrow "00"
 - Face 2 \rightarrow "01"
 - Face 3 \rightarrow "10"
 - Face 4 \rightarrow "11"

Entropia de Shannon

- Mas e se a moeda fosse viciada?
 - $p(H) = 1, \quad p(T) = 0$
 - Precisaríamos codificar alguma coisa?
- E se o dado fosse viciado?
 - $p(1) = 1/2, \quad p(2) = 1/4, \quad p(3) = 1/4, \quad p(4) = 0$
 - Poderíamos fazer uma codificação mais compacta?

Entropia de Shannon

- A informação de que o dado é viciado pode ser utilizada para definir uma codificação que, na média, utiliza menos bits
 - Por exemplo, não precisamos codificar o resultado "4" porque sabemos que ele nunca irá ocorrer
 - Além disso, o resultado "1" é mais frequente
 - Podemos usar um código que emprega cadeias menores para os símbolos mais frequentes

Entropia de Shannon

- Exemplo de codificação para um dado de 4 lados viciado (com probabilidades $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{4}$ e 0)
 - Face 1 \rightarrow "0"
 - Face 2 \rightarrow "10"
 - Face 3 \rightarrow "11"
- Embora o código das faces 2 e 3 seja mais longo, eles são menos frequentes, portanto na média as mensagens precisarão de menos que 2 bits por evento

Entropia de Shannon

- Qual o tamanho **médio** das mensagens que codificam o resultado do lançamento de um dado viciado de 4 faces com probabilidades $(\frac{1}{2}, \frac{1}{4}, \frac{1}{4}, 0)$?

$$\frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 2 + \frac{1}{4} \cdot 2 = 1,5 \text{ bits}$$

A face "4" existe, mas não precisa se representada, pois nunca será o resultado do dado – sabemos que essa mensagem nunca será transmitida

Entropia de Shannon

- Quanto menor a entropia de uma variável aleatória, menor precisa ser o **tamanho médio** da sequência que codifica seus eventos
 - *Minimum description length* (MDL)
- A entropia de Shannon calcula o **limite inferior** em bits desse tamanho
 - Pode ser, portanto, um valor fracionado
 - Uma unidade alternativa é o *Shannon*

Cálculo da entropia

- Seja uma variável aleatória X que tem os eventos associados às possíveis probabilidades
 - E_1 , com probabilidade $p(X = E_1) = p_1$
 - E_2 , com probabilidade $p(X = E_2) = p_2$
 - ...
 - E_n , com probabilidade $p(X = E_n) = p_n$

Cálculo da entropia

- Então a entropia de X é calculada como a esperança da informação própria de X

$$H(X) = E[I(X)]$$

$$H(X) = \text{info}(p_1, p_2, \dots, p_n)$$

$$= -p_1 \log p_1 - p_2 \log p_2 - \dots - p_n \log p_n$$

Cálculo da entropia

- Exemplo: entropia da moeda honesta
 - Seja X uma V.A. que representa o resultado de uma moeda honesta com $p(H) = 0,5$ e $p(T) = 0,5$

$$\begin{aligned}\text{info} \left(\frac{1}{2}, \frac{1}{2} \right) &= -\frac{1}{2} \log \left(\frac{1}{2} \right) - -\frac{1}{2} \log \left(\frac{1}{2} \right) \\ &= -\log \left(\frac{1}{2} \right) = 1\end{aligned}$$

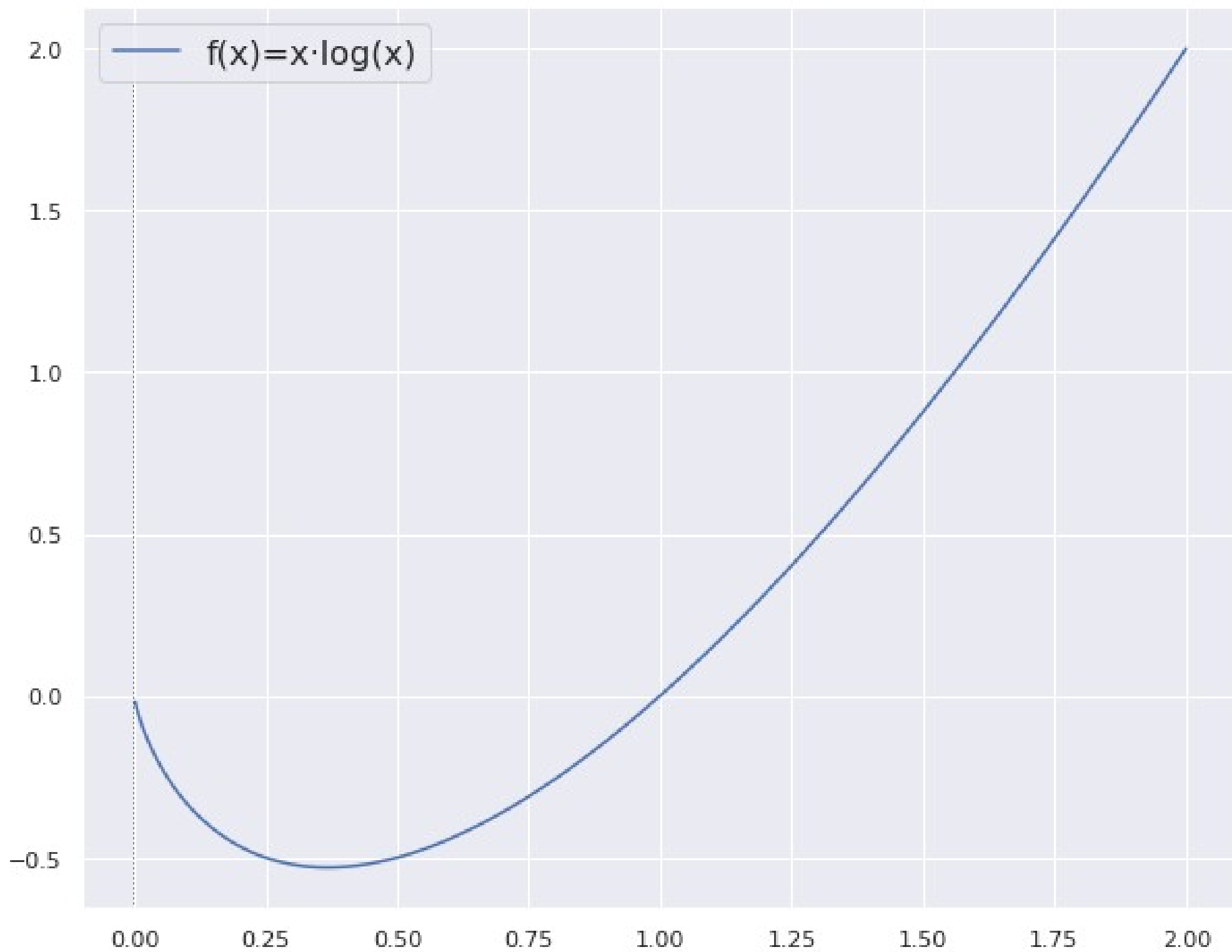
Cálculo da entropia

- Exemplo: entropia do dado viciado
 - Seja X uma V.A. que representa o resultado do nosso dado viciado com $p(1) = 1/2$, $p(2) = p(3) = 1/4$ e $p(4) = 0$

$$\text{info} \left(\frac{1}{2}, \frac{1}{4}, \frac{1}{4}, 0 \right) = -\frac{1}{2} \log \left(\frac{1}{2} \right) - 2 \cdot \frac{1}{4} \log \left(\frac{1}{4} \right) - 0 \log 0$$

Atenção: $\log 0$ não é definido, mas

$$\lim_{x \rightarrow 0!} x \cdot \log(x) = 0$$



Cálculo da entropia

- Exemplo: entropia do dado viciado
 - Seja X uma V.A. que representa o resultado do nosso dado viciado com $p(1) = 1/2$, $p(2) = p(3) = 1/4$ e $p(4) = 0$

$$\begin{aligned}\text{info} \left(\frac{1}{2}, \frac{1}{4}, \frac{1}{4}, 0 \right) &= -\frac{1}{2} \log \left(\frac{1}{2} \right) - 2 \cdot \frac{1}{4} \log \left(\frac{1}{4} \right) - 0 \log 0 \\ &= \frac{1}{2} + 2 \cdot \frac{1}{4} \cdot 2 + 0 = 1,5\end{aligned}$$

Entropia como medida de desordem

- A entropia também pode ser vista como uma medida de incerteza ou desordem
 - No conjunto original da classe *tênis*, temos 9 exemplos da classe positiva e 5 da classe negativa

sim	sim	sim
sim	sim	sim
sim	sim	sim
não	não	não
não	não	

$$p(\text{sim}) = \frac{9}{14}$$

$$p(\text{não}) = \frac{5}{14}$$

$$\text{info}([9, 5]) = \frac{9}{14} \log \left(\frac{9}{14} \right) - \frac{5}{14} \log \left(\frac{5}{14} \right) = 0,94$$

Entropia como medida de desordem

- Se o espaço de classe possuíse apenas exemplos de uma classe, ele seria menos incerto ou desorganizado
 - Teríamos a certeza de que todos os exemplos nesse espaço pertencem a uma classe

sim	sim	sim
sim	sim	sim
sim	sim	sim

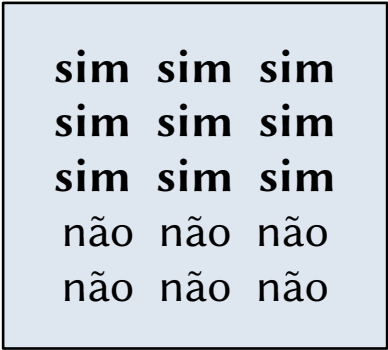
$$p(\text{sim}) = 1$$

$$p(\text{não}) = 0$$

$$\text{info}([9, 0]) = -\log 1 - 0 \cdot \log 0 = 0$$

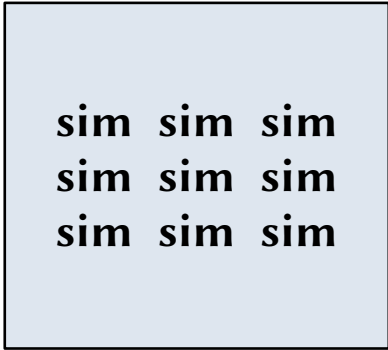
Entropia como medida de desordem

- Dizer que um espaço desorganizado possui maior entropia equivale a dizer que selecionar aleatoriamente uma classe para exemplos desse espaço gera maior incerteza

A light blue square box containing a 4x3 grid of text. The first three rows consist of the word 'sim' repeated three times. The last two rows consist of the word 'não' repeated three times.

sim sim sim
sim sim sim
sim sim sim
não não não
não não não

mais desorganizado
mais incerto

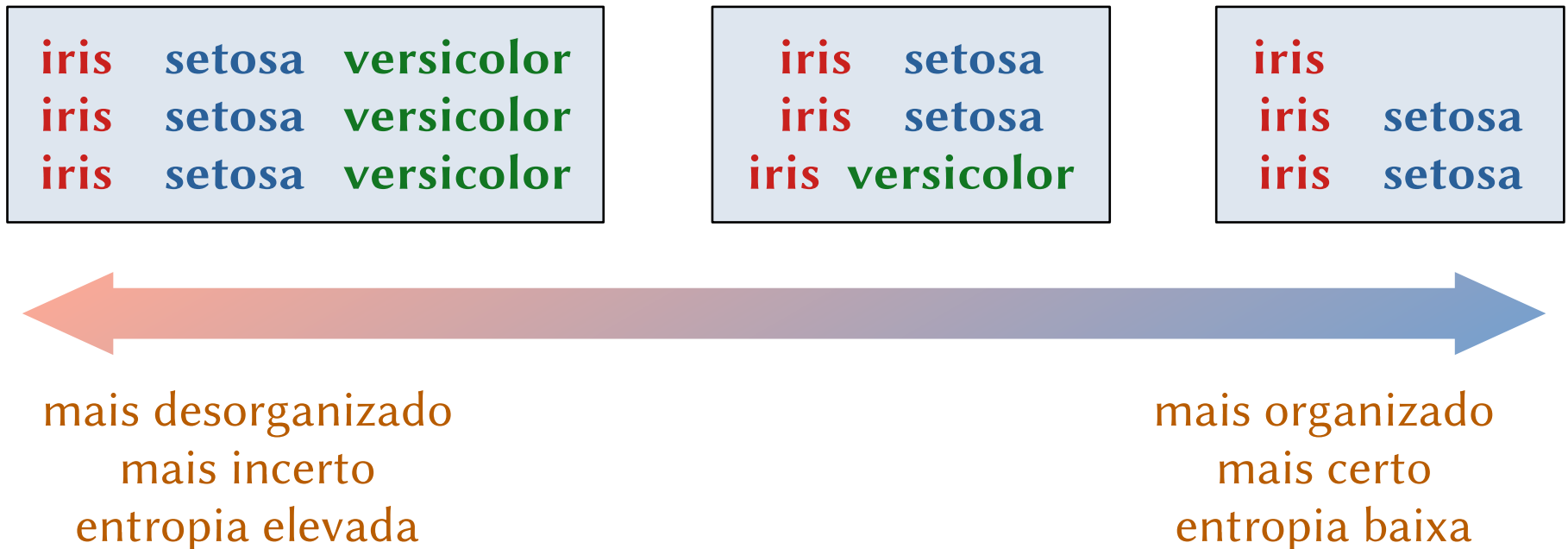
A light blue square box containing a 3x3 grid of text. All six rows consist of the word 'sim' repeated three times.

sim sim sim
sim sim sim
sim sim sim
sim sim sim
sim sim sim
sim sim sim

mais organizado
mais certo

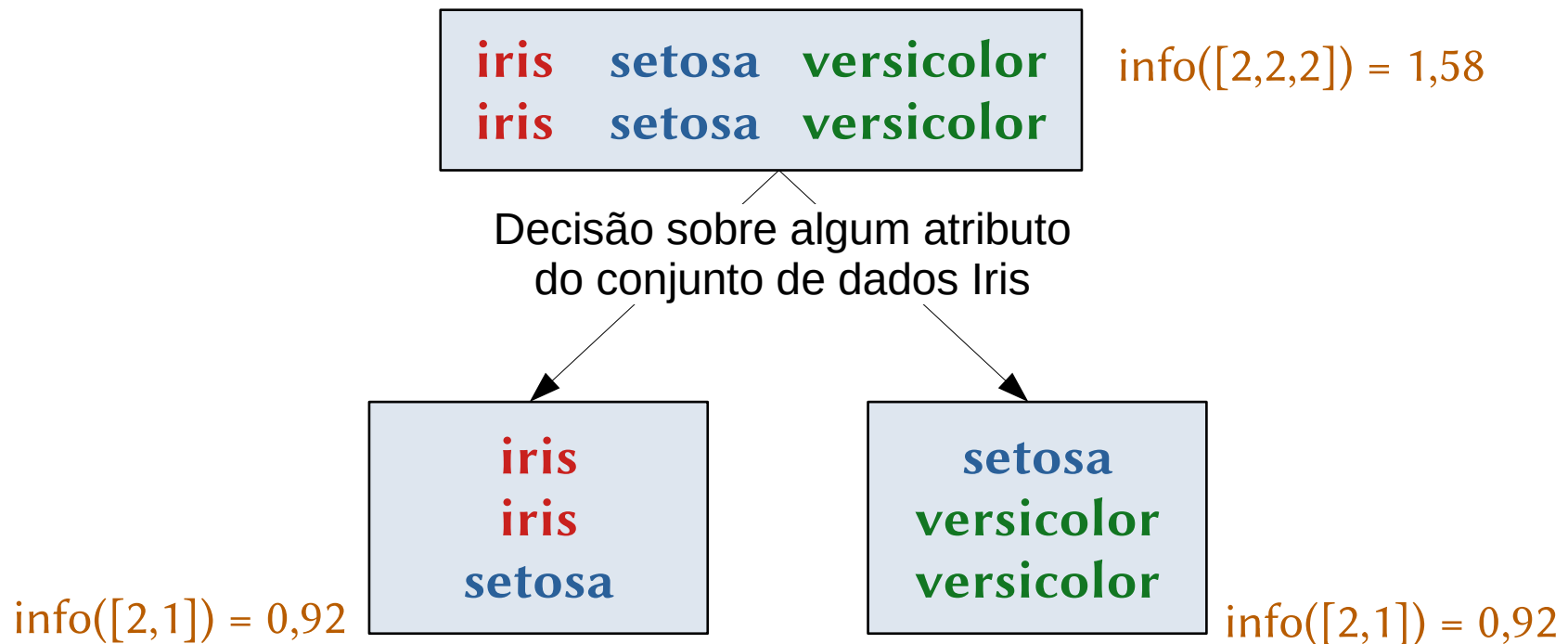
Entropia como medida de desordem

- Dizer que um espaço desorganizado possui maior entropia equivale a dizer que selecionar aleatoriamente uma classe para exemplos desse espaço gera maior incerteza



Entropia de espaço de atributos

- Se o classificador nos "conduz" de um espaço altamente incerto/desordenado para um espaço mais ordenado, então *ganhamos informação*



Ganho de informação

- Diferença entre informação/desordem do espaço original e a informação/desordem média dos sub-espaços obtidos pelo particionamento
 - A média dos sub-espaços deve ser ponderada pela probabilidade de encontrarmos um exemplo nele
 - Quanto mais homogêneos são os sub-espaços gerados pelo particionamento, maior é o ganho de informação

Ganho de Informação

- Como encontrar o atributo X_{best} ?
 - Selecione aquele que nos dá o maior ganho de informação!

Árvore para jogar tênis: informação original

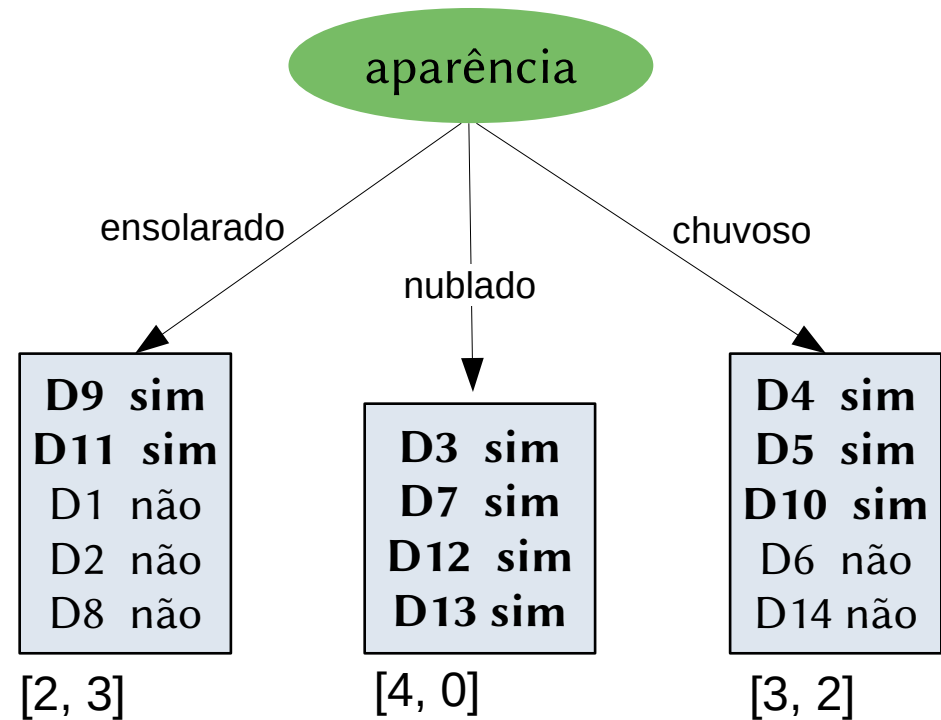
dia	aparência	temperatura	umidade	vento	jogar
D1	ensolarado	quente	alta	fraco	não
D2	ensolarado	quente	alta	forte	não
D3	nublado	quente	alta	fraco	sim
D4	chuvoso	moderado	alta	fraco	sim
D5	chuvoso	frio	baixa	fraco	sim
D6	chuvoso	frio	baixa	forte	não
D7	nublado	frio	baixa	forte	sim
D8	ensolarado	moderado	alta	fraco	não
D9	ensolarado	frio	baixa	fraco	sim
D10	chuvoso	moderado	baixa	fraco	sim
D11	ensolarado	moderado	baixa	forte	sim
D12	nublado	moderado	alta	forte	sim
D13	nublado	quente	baixa	fraco	sim
D14	chuvoso	moderado	alta	forte	não

não: 5 exemplos
sim: 9 exemplos

$$\text{info}([9, 5]) = \frac{9}{14} \log \left(\frac{9}{14} \right) - \frac{5}{14} \log \left(\frac{5}{14} \right) = 0,94$$

Testando aparência como atributo da raiz

dia	aparência	jogar
D1	ensolarado	não
D2	ensolarado	não
D3	nublado	sim
D4	chuvoso	sim
D5	chuvoso	sim
D6	chuvoso	não
D7	nublado	sim
D8	ensolarado	não
D9	ensolarado	sim
D10	chuvoso	sim
D11	ensolarado	sim
D12	nublado	sim
D13	nublado	sim
D14	chuvoso	não



$$\text{info}([2, 3]) = -\frac{2}{5} \log \left(\frac{2}{5} \right) - \frac{3}{5} \log \left(\frac{3}{5} \right) = 0,97$$

$$\text{info}([4, 0]) = -\log 1 - 0 \cdot \log 0 = 0$$

$$\text{info}([3, 2]) = \text{info}([2, 3]) = 0,97$$

Testando aparência como atributo da raiz

- Informação média ponderada de aparência:

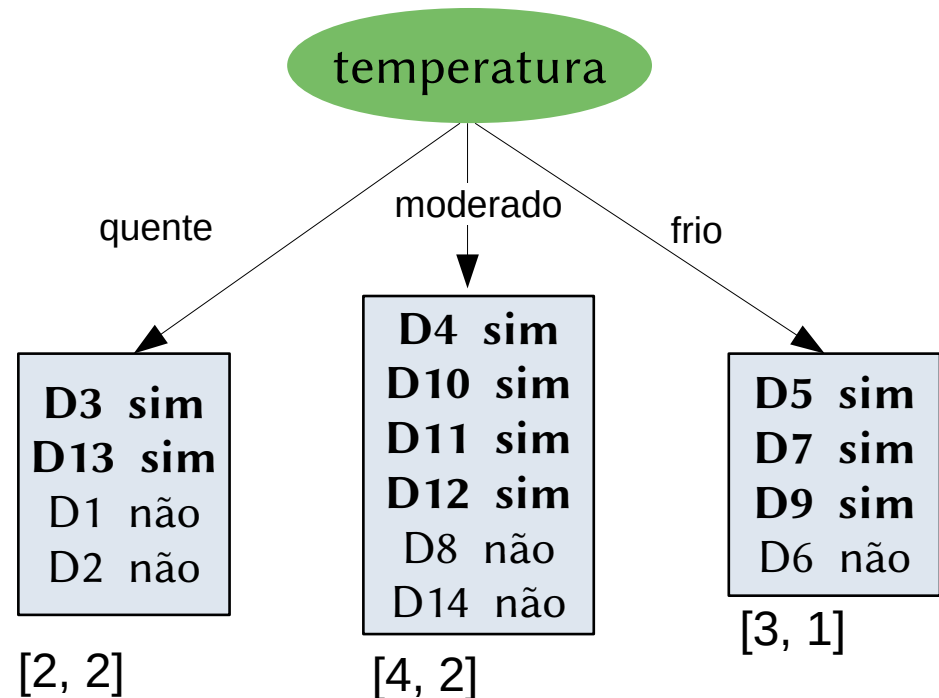
$$\text{info}([2, 3], [4, 0], [3, 2]) = \frac{5}{14} \cdot \text{info}([2, 3]) + \frac{4}{14} \cdot \text{info}([4, 0]) + \frac{5}{14} \cdot \text{info}([3, 2]) = 0,69$$

- Ganho do atributo aparência:

$$\begin{aligned} GI(\text{aparência}) &= \text{info}([9, 5]) - \text{info}([2, 3], [4, 0], [3, 2]) \\ &= 0,94 - 0,69 = 0,25 \end{aligned}$$

Testando temperatura como atributo da raiz

dia	temperatura	jogar
D1	quente	não
D2	quente	não
D3	quente	sim
D4	moderado	sim
D5	frio	sim
D6	frio	não
D7	frio	sim
D8	moderado	não
D9	frio	sim
D10	moderado	sim
D11	moderado	sim
D12	moderado	sim
D13	quente	sim
D14	moderado	não



$$\text{info}([2, 2]) = -\frac{2}{4} \log \left(\frac{2}{4} \right) - \frac{2}{4} \log \left(\frac{2}{4} \right) = 1$$

$$\text{info}([4, 2]) = -\frac{4}{6} \log \left(\frac{4}{6} \right) - \frac{2}{6} \log \left(\frac{2}{6} \right) = 0,92$$

$$\text{info}([3, 1]) = -\frac{3}{4} \log \left(\frac{3}{4} \right) - \frac{1}{4} \log \left(\frac{1}{4} \right) = 0,81$$

Testando temperatura como atributo da raiz

- Informação média ponderada de temperatura:

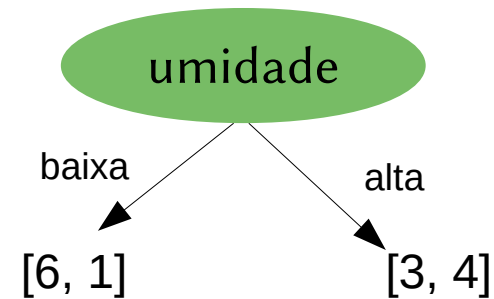
$$\begin{aligned}\text{info}([2, 2], [4, 2], [3, 1]) &= \frac{4}{14} \cdot \text{info}([3, 2]) + \frac{6}{14} \cdot \text{info}([4, 2]) + \frac{4}{14} \cdot \text{info}([3, 1]) \\ &= 0,91\end{aligned}$$

- Ganho do atributo temperatura:

$$\begin{aligned}GI(\text{temperatura}) &= \text{info}([9, 5]) - \text{info}([2, 2], [4, 2], [3, 1]) \\ &= 0,94 - 0,91 = 0,03\end{aligned}$$

Testando umidade como atributo da raiz

dia	umidade	jogar
D1	alta	não
D2	alta	não
D3	alta	sim
D4	alta	sim
D5	baixa	sim
D6	baixa	não
D7	baixa	sim
D8	alta	não
D9	baixa	sim
D10	baixa	sim
D11	baixa	sim
D12	alta	sim
D13	baixa	sim
D14	alta	não



$$\text{info}([6, 1]) = 0,59$$

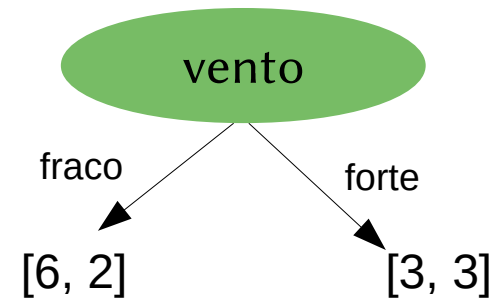
$$\text{info}([3, 4]) = 0,98$$

$$\text{info}([6, 1], [3, 4]) = 0,78$$

$$\text{ganho} = 0,94 - 0,78 = 0,16$$

Testando vento como atributo da raiz

dia	vento	jogar
D1	fraco	não
D2	forte	não
D3	fraco	sim
D4	fraco	sim
D5	fraco	sim
D6	forte	não
D7	forte	sim
D8	fraco	não
D9	fraco	sim
D10	fraco	sim
D11	forte	sim
D12	forte	sim
D13	fraco	sim
D14	forte	não



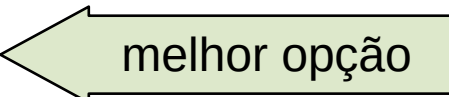
$$\text{info}([6, 2]) = 0,81$$

$$\text{info}([3, 3]) = 1,00$$

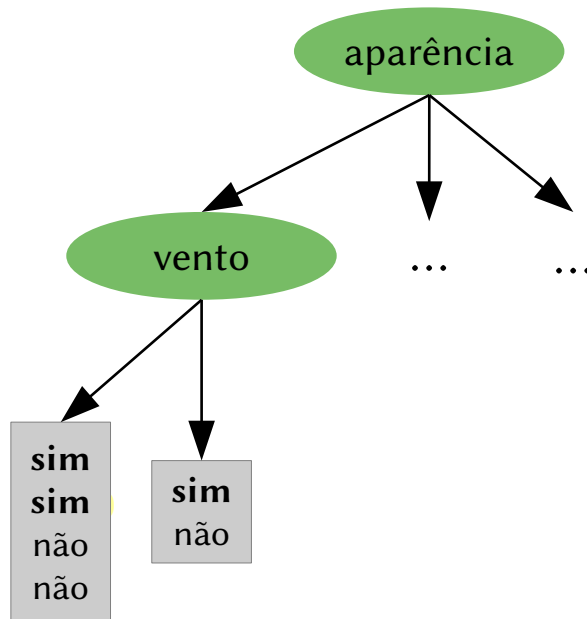
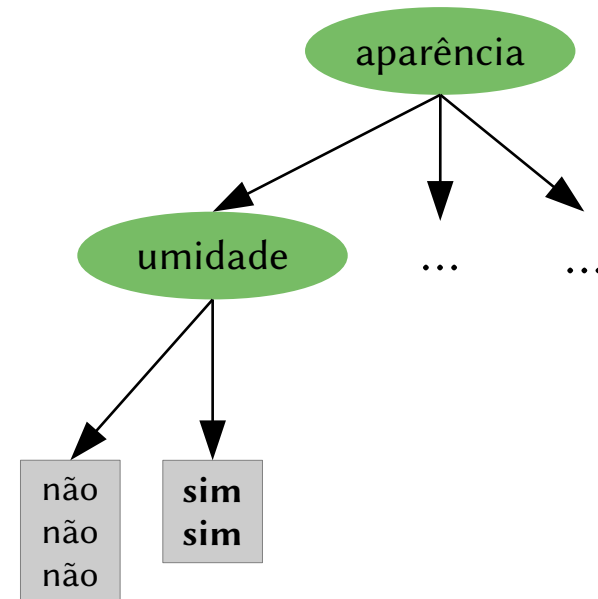
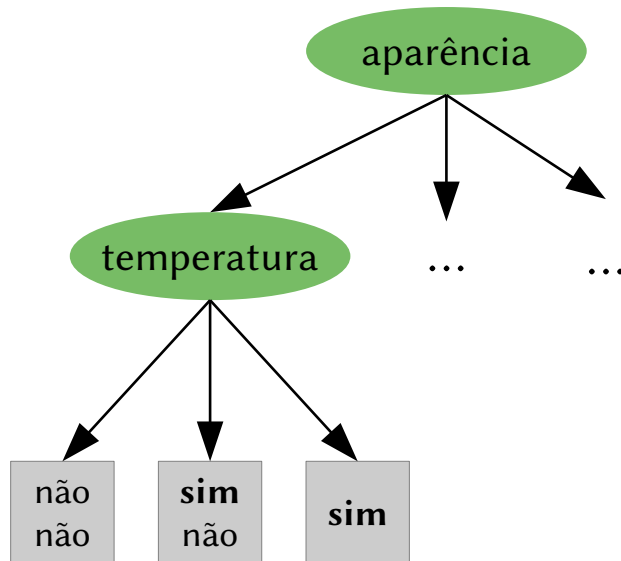
$$\text{info}([6, 2], [3, 3]) = 0,89$$

$$\text{ganho} = 0,94 - 0,89 = 0,05$$

Ganhos verificados para a raiz

- $\text{ganho}(\text{aparência}) = 0,25$ 
- $\text{ganho}(\text{temperatura}) = 0,03$
- $\text{ganho}(\text{umidade}) = 0,16$
- $\text{ganho}(\text{vento}) = 0,05$

Continuando a dividir...



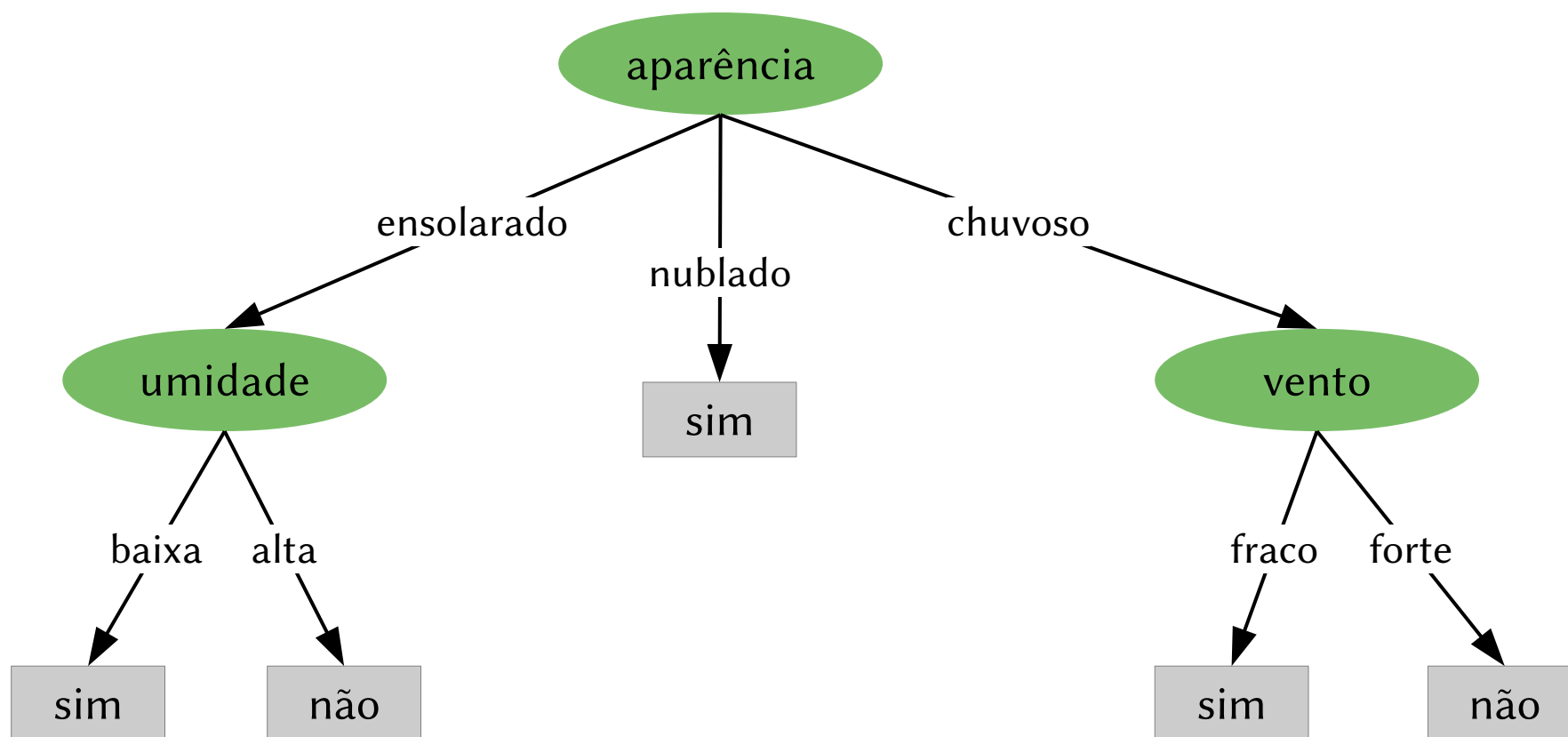
ganho(Temperatura) = 0,57 bits

ganho(Umididade) = 0,97 bits

ganho(Vento) = 0,02 bits

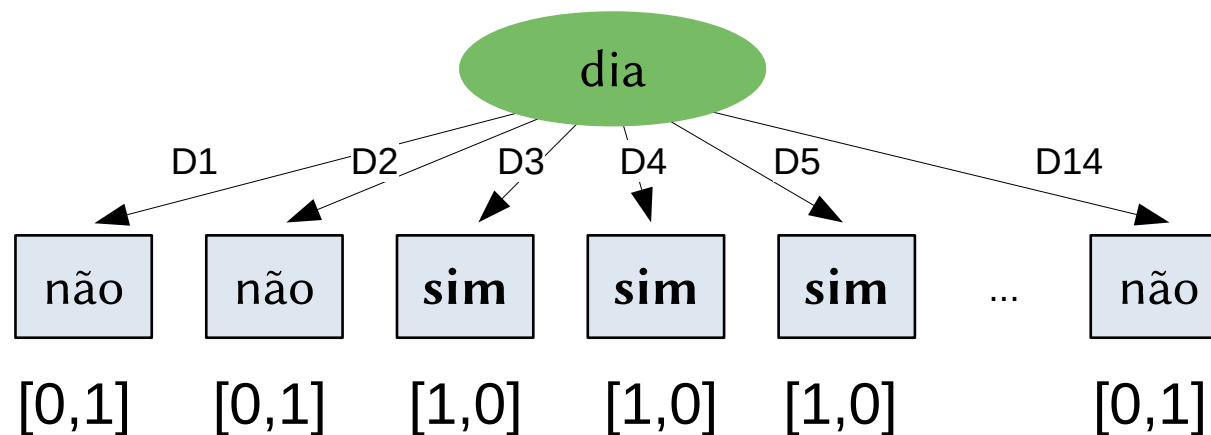
ganho(Aparência) = 0 (por quê?)

Árvore de decisão final



Problema do ganho de informação

- Atributos com muitos valores tendem a dar ganho de informação muito elevado, mas causam *overfitting*
- Caso extremo: identificador
 - O ganho de informação do identificador é máximo



$$\text{info}([0,1], [0,1], [1,0], [1,0], [1,0], \dots, [0,1]) = 0$$

Razão de Ganho

- Embora identificadores sejam um caso extremo, o mesmo problema pode ocorrer em menor grau com atributos válidos
- Uma forma de contornar esse problema é empregar **razão de ganho**
 - A razão de ganho relativiza o ganho de informação pela **informação intrínseca** do atributo

Razão de Ganho

- A **informação intrínseca** do atributo é a quantidade de informação do atributo
 - Quanto informação é necessária para descrever o valor do atributo X_i ?
 - Exemplo: aparência
 - Ensolarado Nublado Chuvoso
5 exemplos 4 exemplos 5 exemplos
- $IV(\text{aparência}) = \text{info}([5, 4, 5])$

Razão de Ganho

- A razão de ganho é a razão entre o ganho de informação e a informação intrínseca do atributo
 - A razão de ganho não tem unidade

$$GR(\text{atributo}) = \frac{GI(\text{atributo})}{IV(\text{atributo})}$$

Razão de Ganho

- Exemplo (id)
 - Ganho:
 - $GI(id) = \text{info}([9, 5]) - \text{info}(1, 1, 1, \dots, 1] = 0,94$
 - Informação intrínseca:
 - $IV(id) = \text{info}([1, 1, 1, \dots, 1])$
$$= 14 \cdot \left[-\frac{1}{14} \log \left(\frac{1}{14} \right) \right] = 3,8$$
 - Razão de ganho
 - $GR(id) = \frac{0,94 \text{ bits}}{3,80 \text{ bits}} = 0,25$

Razão de Ganho

Aparência		
Informação	info([2,3], [4,0], [3,2])	0,69
Ganho	0,94 - 0,69	0,25
Info. intrínseca	info([5, 4, 5])	1,58
Razão de ganho	0,25 / 1,57	0,16

Temperatura		
Informação	info([2,2], [4,2], [3,1])	0,91
Ganho	0,94 - 0,91	0,03
Info. intrínseca	info([4, 6, 4])	1,56
Razão de ganho	0,03 / 1,56	0,02

Temperatura		
Informação	info([6,1], [3,4])	0,78
Ganho	0,94 - 0,78	0,16
Info. intrínseca	info([7, 7])	1,00
Razão de ganho	0,16 / 1,00	0,16

Vento		
Informação	info([6,2], [3,3])	0,89
Ganho	0,94 - 0,89	0,05
Info. intrínseca	info([8, 6])	0,99
Razão de ganho	0,05 / 0,99	0,05

Atributos numéricos

dia	aparencia	temperatura	umidade	vento	jogar
D1	ensolarado	23	58	fraco	nao
D2	ensolarado	24	55	forte	nao
D3	nublado	19	66	fraco	sim
D4	chuvoso	15.1	72	fraco	sim
D5	chuvoso	13.5	65	fraco	sim
D6	chuvoso	9	60	forte	nao
D7	nublado	14.2	45	forte	sim
D8	ensolarado	18	63	fraco	nao
D9	ensolarado	13.2	38	fraco	sim
D10	chuvoso	15.5	50	fraco	sim
D11	ensolarado	15	36	forte	sim
D12	nublado	16.5	68	forte	sim
D13	nublado	20	35	fraco	sim
D14	chuvoso	13	70	forte	nao

Atributos numéricos

- Podemos lidar com atributos numéricos em árvore de duas maneiras
 - Discretizá-los em um certo número de *bins*. Por exemplo, discretizando o atributo **temperatura** em cinco *bins* uniformemente distribuídos...
 - $bin1 = (-\infty; 13,85]$
 - $bin2 = (13,85; 15,3]$
 - $bin3 = (15,3; 18,5]$
 - $bin4 = (18,5; \infty)$

Atributos numéricos

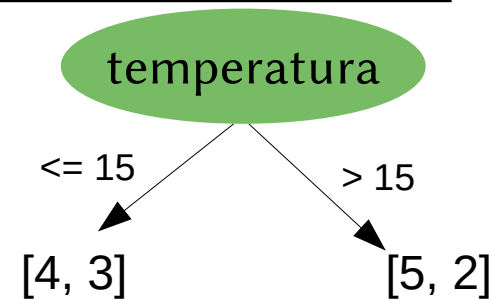
dia	aparencia	temp_discret	umidade	vento	jogar
D1	ensolarado	temp_4	58	fraco	nao
D2	ensolarado	temp_4	55	forte	nao
D3	nublado	temp_4	66	fraco	sim
D4	chuvoso	temp_2	72	fraco	sim
D5	chuvoso	temp_1	65	fraco	sim
D6	chuvoso	temp_1	60	forte	nao
D7	nublado	temp_2	45	forte	sim
D8	ensolarado	temp_3	63	fraco	nao
D9	ensolarado	temp_1	38	fraco	sim
D10	chuvoso	temp_3	50	fraco	sim
D11	ensolarado	temp_2	36	forte	sim
D12	nublado	temp_3	68	forte	sim
D13	nublado	temp_4	35	fraco	sim
D14	chuvoso	temp_1	70	forte	nao

Atributos numéricos

- Podemos lidar com atributos numéricos em árvore de duas maneiras
 - Estabelecer um ponto de corte $c_{atributo}$ e considerar o particionamento
 - Se = sub-espço no qual todos os exemplos possuem valor $atributo \leq c_{atributo}$
 - Sd = todos os exemplos possuem valor $atributo > c_{atributo}$

Atributos numéricos

dia	temperatura	jogar
D1	23	nao
D2	24	nao
D3	19	sim
D4	15.1	sim
D5	13.5	sim
D6	9	nao
D7	14.2	sim
D8	18	nao
D9	13.2	sim
D10	15.5	sim
D11	15	sim
D12	16.5	sim
D13	20	sim
D14	13	nao



$$\text{info}([4, 3]) = 0,98$$

$$\text{info}([5, 2]) = 0,86$$

$$\text{info}([4, 3], [5, 2]) = 0,92$$

$$IG(\text{temp}_{15}) = 0,94 - 0,92 = 0,02$$

$$IV(\text{temp}_{15}) = \text{info}([5, 7]) = 0,98$$

$$GR(\text{temp}_{15}) = \frac{0,02}{0,98} = 0,20$$

Atributos numéricos

- Podemos proceder da seguinte forma:
 - Ordenamos os dados de acordo com o atributo que queremos testar
 - Verificamos os pontos onde há mudança de classe
 - Para cada potencial ponto de corte, calculamos o ganho de informação / a razão de ganho
 - Selecionamos o melhor como o ganho de informação máxima do atributo

Atributos numéricos

dia	temperatura	jogar
D6	9	nao
D14	13	nao
D9	13.2	sim
D5	13.5	sim
D7	14.2	sim
D11	15	sim
D4	15.1	sim
D10	15.5	sim
D12	16.5	sim
D8	18	nao
D3	19	sim
D13	20	sim
D1	23	nao
D2	24	nao

$$c_1 \rightarrow [0,2], [9, 3]$$

$$c_2 \rightarrow [7,2], [2, 3]$$

$$c_3 \rightarrow [7,3], [2, 2]$$

$$c_4 \rightarrow [9,3], [0, 2]$$

Agenda

- Definições
- Teoria das probabilidades
- Aprendizado Bayesiano e modelos probabilísticos
- Modelos baseados em árvores
- Modelos baseados em regras
- Classificação preguiçosa: k-NN
- Máquina de vetores de suporte

Regra de conhecimento

- Modelo no qual o conhecimento é representado através de conjunções de condições
- São descritas como cláusulas de Horn
 - $p_1 \wedge p_2 \wedge p_3 \wedge \dots \wedge p_n \rightarrow u$
 - p_i : condições ou premissas
 - u : fato

Regra de conhecimento

- A regra de conhecimento possui duas partes
 - Corpo (*body*) ou complexo: condições
 - Cabeça (*head*): classe associada ao exemplo

$$R : \underbrace{if \text{ complexo}}_{body \text{ ou } B} \text{ then } \underbrace{class = C_v}_{head \text{ ou } H}$$

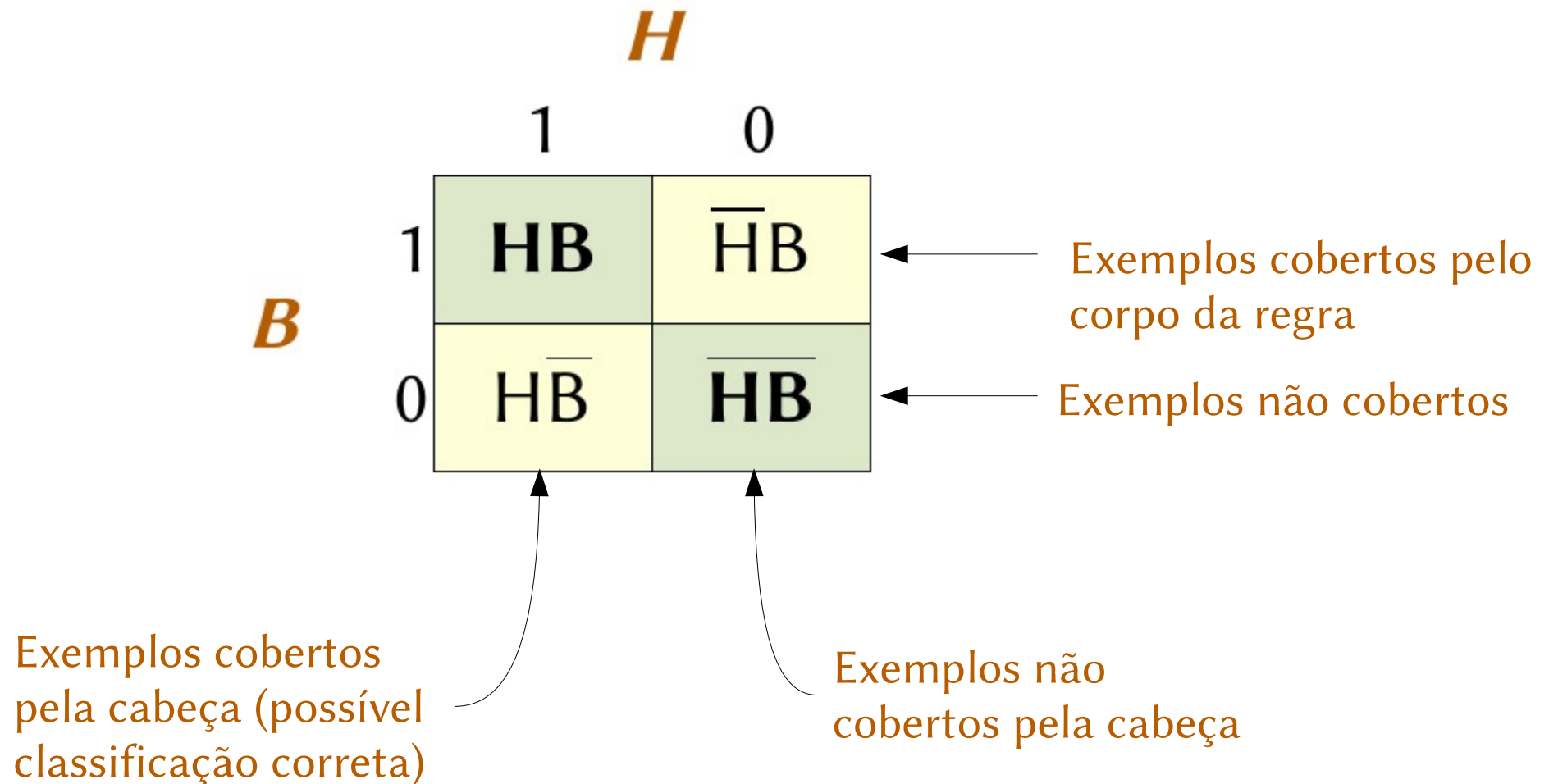
Regra de conhecimento

- Um exemplo que é compatível com as condições da regra é **coberto** pelo corpo da regra
- Um exemplo que é compatível com a cabeça é coberto pela cabeça da regra
 - Se a cabeça da regra é compatível com a classe de um exemplo coberto, então ele **pode** ser corretamente classificado

Matriz de contingência

- **Matriz de contingência** é uma tabela para duas ou mais variáveis aleatórias independentes
 - **Matriz de confusão** é um caso particular de matriz de contingência
 - Aplica-se especificamente a algoritmos de aprendizado de máquina
 - Matrizes de contingência podem ser utilizadas para contabilizar cobertura e precisão de regras

Matriz de contingência



Matriz de contingência

- Existem quatro eventos
 - HB ou f_{HB} : cobertura do corpo e da cabeça
 - $H\bar{B}$ ou $f_{H\bar{B}}$: cobertura da cabeça, mas não do corpo
 - $\bar{H}B$ ou $f_{\bar{H}B}$: cobertura do corpo, mas não da cabeça
 - $\bar{H}\bar{B}$ ou $f_{\bar{H}\bar{B}}$: não cobertura do corpo, nem da cabeça

Zero-rule

- O modelo de regra mais simples é o R_0 (*zero-rule*)
 - Como cláusula de Horn:
 - $V \rightarrow u$
 - Como classificador
 - Um *baseline* que classifica todos os exemplos como pertencentes a uma mesma classe
 - Qual seria o R_0 para o conjunto **tennis**?

Zero-rule

- Indução do R_0
 - Defina o corpo como vazio (cobre todos os exemplos)
 - Associe os exemplos à classe mais frequente c_{moda}

if TRUE then

CLASS = c_{moda}

Medidas de qualidade de regras (1)

- Precisão:
 - Mede o grau de especificidade de uma regra para um problema

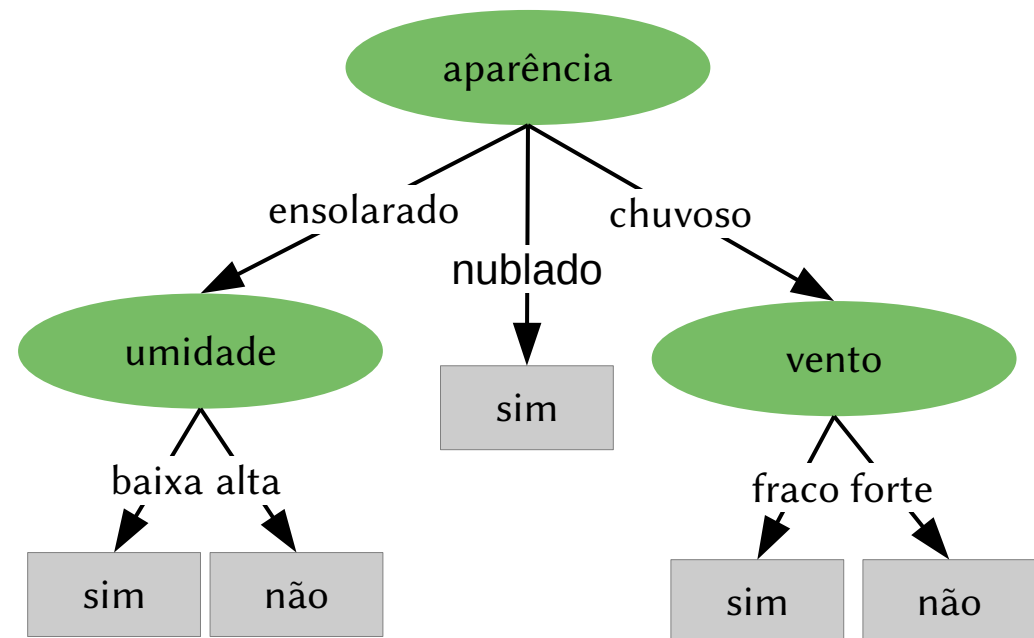
$$Pre(R) = \frac{f_{hb}}{f_b}$$

- Acurácia:
 - Mede o desempenho geral de uma regra

$$Acc(R) = f_{hb} + f_{\overline{hb}}$$

Regras a partir de árvores

- Podemos converter uma árvore em uma coleção de regras de conhecimento
 - Selecione um caminho da árvore e transforme em uma cláusula de Horn que implica na classe associada ao nó folha



Regras a partir de árvores

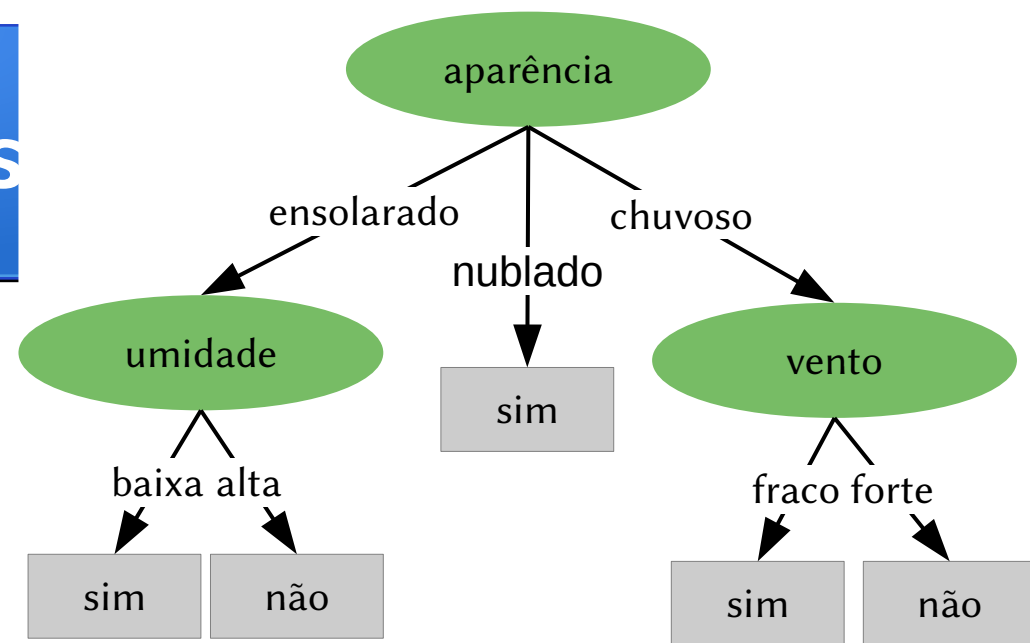
- IF aparência = ensolarado
AND umidade = baixa
THEN class = sim

- IF aparência = ensolarado
AND umidade = alta
THEN class = não

- IF aparência = nublado
THEN class = sim

- IF aparência = chuvoso
AND vento = fraco
THEN class = sim

- IF aparência = chuvoso
AND umidade = forte
THEN class = não



Regras a partir de árvores

- Vantagem
 - Conversão simples e direta
 - Utiliza-se do conceito de ganho de informação para construir regras curtas
- Problema
 - Embora cada regra seja simples, o conjunto de regras é complexo
 - Simplificação não é trivial

Construção direta de regras

- Algoritmo de cobertura/precisão
 - Dado um conjunto de exemplos
 - Para cada classe, encontre uma regra com precisão máxima: $R_{c1}, R_{c2}, \dots, R_{cM}$
 - Selecione a regra R_{ci} com maior precisão
 - Descarte os exemplos cobertos por R_{ci}
 - Repita até $Pre(R_{ci}) = 1$ ou não seja mais possível separar os exemplos

Exemplo: Lentes de contato

Age	Specta prescription	Astigmatism	Tear prod. rate	Lenses
Young	Myope	No	Reduced	None
Young	Myope	No	Normal	Soft
Young	Myope	Yes	Reduced	None
Young	Myope	Yes	Normal	Hard
Young	Hypermetrope	No	Reduced	None
Young	Hypermetrope	No	Normal	Soft
Young	Hypermetrope	Yes	Reduced	None
Young	Hypermetrope	Yes	Normal	hard
Pre-presbyopic	Myope	No	Reduced	None
Pre-presbyopic	Myope	No	Normal	Soft
Pre-presbyopic	Myope	Yes	Reduced	None
Pre-presbyopic	Myope	Yes	Normal	Hard
Pre-presbyopic	Hypermetrope	No	Reduced	None
Pre-presbyopic	Hypermetrope	No	Normal	Soft
Pre-presbyopic	Hypermetrope	Yes	Reduced	None
Pre-presbyopic	Hypermetrope	Yes	Normal	None
Presbyopic	Myope	No	Reduced	None
Presbyopic	Myope	No	Normal	None
Presbyopic	Myope	Yes	Reduced	None
Presbyopic	Myope	Yes	Normal	Hard
Presbyopic	Hypermetrope	No	Reduced	None
Presbyopic	Hypermetrope	No	Normal	Soft
Presbyopic	Hypermetrope	Yes	Reduced	None
Presbyopic	Hypermetrope	Yes	Normal	None

Regra procurada

if ?
then Lenses = hard

89

Age	Specta prescription	Astigmatism	Tear prod. rate	Lenses
Young	Myope	No	Reduced	None
Young	Myope	No	Normal	Soft
Young	Myope	Yes	Reduced	None
Young	Myope	Yes	Normal	Hard
Young	Hypermetrope	No	Reduced	None
Young	Hypermetrope	No	Normal	Soft
Young	Hypermetrope	Yes	Reduced	None
Young	Hypermetrope	Yes	Normal	hard
Pre-presbyopic	Myope	No	Reduced	None
Pre-presbyopic	Myope	No	Normal	Soft
Pre-presbyopic	Myope	Yes	Reduced	None
Pre-presbyopic	Myope	Yes	Normal	Hard
Pre-presbyopic	Hypermetrope	No	Reduced	None
Pre-presbyopic	Hypermetrope	No	Normal	Soft
Pre-presbyopic	Hypermetrope	Yes	Reduced	None
Pre-presbyopic	Hypermetrope	Yes	Normal	None
Presbyopic	Myope	No	Reduced	None
Presbyopic	Myope	No	Normal	None
Presbyopic	Myope	Yes	Reduced	None
Presbyopic	Myope	Yes	Normal	Hard
Presbyopic	Hypermetrope	No	Reduced	None
Presbyopic	Hypermetrope	No	Normal	Soft
Presbyopic	Hypermetrope	Yes	Reduced	None
Presbyopic	Hypermetrope	Yes	Normal	None

Regra procurada

if Age = ?
then Lenses = hard

90

Age	Specta prescription	Astigmatism	Tear prod. rate	Lenses
Young	Myope	No	Reduced	None
Young	Myope	No	Normal	Soft
Young	Myope	Yes	Reduced	None
Young	Myope	Yes	Normal	Hard
Young	Hypermetrope	No	Reduced	None
Young	Hypermetrope	No	Normal	Soft
Young	Hypermetrope	Yes	Reduced	None
Young	Hypermetrope	Yes	Normal	hard
Pre-presbyopic	Myope	No	Reduced	None
Pre-presbyopic	Myope	No	Normal	Soft
Pre-presbyopic	Valor do atributo			Pre(R)
Pre-presbyopic	Age = Young			2/8
Pre-presbyopic	Age = Pre-presbyopic			1/8
Pre-presbyopic	Age = Presbyopic			1/8
Pre-presbyopic	Hypermetrope	Yes	Normal	None
Presbyopic	Myope	No	Reduced	None
Presbyopic	Myope	No	Normal	None
Presbyopic	Myope	Yes	Reduced	None
Presbyopic	Myope	Yes	Normal	Hard
Presbyopic	Hypermetrope	No	Reduced	None
Presbyopic	Hypermetrope	No	Normal	Soft
Presbyopic	Hypermetrope	Yes	Reduced	None
Presbyopic	Hypermetrope	Yes	Normal	None

Exemplo: lentes de contato

- Regra procurada:

```
if Age = ?
then Lenses = hard
```

- Testes possíveis:

Age = Young	2/8
Age = Pre-presbyopic	1/8
Age = Presbyopic	1/8
Spectacle prescription = Myope	3/12
Spectacle prescription = Hypermetrope	1/12
Astigmatism = no	0/12
Astigmatism = yes	4/12
Tear production rate = Reduced	0/12
Tear production rate = Normal	4/12

Exemplo: lentes de contato

- Regra procurada:

```
if Age = ?
then Lenses = hard
```

- Testes possíveis:

Age = Young	2/8
Age = Pre-presbyopic	1/8
Age = Presbyopic	1/8
Spectacle prescription = Myope	3/12
Spectacle prescription = Hypermetrope	1/12
Astigmatism = no	0/12
Astigmatism = yes	4/12
Tear production rate = Reduced	0/12
Tear production rate = Normal	4/12

Regra procurada

if Astigmatism = yes
and ?
then Lenses = hard

93

Age	Specta prescription	Astigmatism	Tear prod. rate	Lenses
Young	Myope	No	Reduced	None
Young	Myope	No	Normal	Soft
Young	Myope	Yes	Reduced	None
Young	Myope	Yes	Normal	Hard
Young	Hypermetrope	No	Reduced	None
Young	Hypermetrope	No	Normal	Soft
Young	Hypermetrope	Yes	Reduced	None
Young	Hypermetrope	Yes	Normal	hard
Pre-presbyopic	Myope	No	Reduced	None
Pre-presbyopic	Myope	No	Normal	Soft
Pre-presbyopic	Myope	Yes	Reduced	None
Pre-presbyopic	Myope	Yes	Normal	Hard
Pre-presbyopic	Hypermetrope	No	Reduced	None
Pre-presbyopic	Hypermetrope	No	Normal	Soft
Pre-presbyopic	Hypermetrope	Yes	Reduced	None
Pre-presbyopic	Hypermetrope	Yes	Normal	None
Presbyopic	Myope	No	Reduced	None
Presbyopic	Myope	No	Normal	None
Presbyopic	Myope	Yes	Reduced	None
Presbyopic	Myope	Yes	Normal	Hard
Presbyopic	Hypermetrope	No	Reduced	None
Presbyopic	Hypermetrope	No	Normal	Soft
Presbyopic	Hypermetrope	Yes	Reduced	None
Presbyopic	Hypermetrope	Yes	Normal	None

Regra modificada

if Astigmatism = yes
then Lenses = hard

Exemplos cobertos

Age	Specta prescription	Astigmatism	Tear prod. rate	Lenses
Young	Myope	Yes	Reduced	None
Young	Myope	Yes	Normal	Hard
Young	Hypermetrope	Yes	Reduced	None
Young	Hypermetrope	Yes	Normal	hard
Pre-presbyopic	Myope	Yes	Reduced	None
Pre-presbyopic	Myope	Yes	Normal	Hard
Pre-presbyopic	Hypermetrope	Yes	Reduced	None
Pre-presbyopic	Hypermetrope	Yes	Normal	None
Presbyopic	Myope	Yes	Reduced	None
Presbyopic	Myope	Yes	Normal	Hard
Presbyopic	Hypermetrope	Yes	Reduced	None
Presbyopic	Hypermetrope	Yes	Normal	None

Regra procurada

if Astigmatism = yes
and ??
then Lenses = hard

Age	Specta prescription	Astigmatism	Tear prod. rate	Lenses
Young	Myope	Yes	Reduced	None
Young	Myope	Yes	Normal	Hard
Young	Hypermetrope	Yes	Reduced	None
Young	Hypermetrope	Yes	Normal	hard
Pre-presbyopic	Myope	Yes	Reduced	None
Pre-presbyopic	Myope	Yes	Normal	Hard
Pre-presbyopic	Hypermetrope	Yes	Reduced	None
Pre-presbyopic	Hypermetrope	Yes	Normal	None
Presbyopic	Myope	Yes	Reduced	None
Presbyopic	Myope	Yes	Normal	Hard
Presbyopic	Hypermetrope	Yes	Reduced	None
Presbyopic	Hypermetrope	Yes	Normal	None

Exemplo: lentes de contato

- Regra procurada:
- Testes possíveis:

```
if Astigmatism = yes
and ??
then Lenses = hard
```

Age = Young	2/4
Age = Pre-presbyopic	1/4
Age = Presbyopic	1/4
Spectacle prescription = Myope	3/6
Spectacle prescription = Hypermetrope	1/6
Tear production rate = Reduced	0/6
Tear production rate = Normal	4/6

Regra modificada

if Astigmatism = yes
and Tear production rate = normal
then Lenses = hard

97

Exemplos cobertos

Age	Specta prescription	Astigmatism	Tear prod. rate	Lenses
Young	Myope	Yes	Normal	Hard
Young	Hypermetrope	Yes	Normal	hard
Pre-presbyopic	Myope	Yes	Normal	Hard
Pre-presbyopic	Hypermetrope	Yes	Normal	None
Presbyopic	Myope	Yes	Normal	Hard
Presbyopic	Hypermetrope	Yes	Normal	None

Regra refinada

```

if Astigmatism = yes
  and Tear production rate = normal
  and ???
then Lenses = hard
  
```

Age	Specta prescription	Astigmatism	Tear prod. rate	Lenses
Young	Myope	Yes	Normal	Hard
Young	Hypermetrope	Yes	Normal	hard
Pre-presbyopic	Myope	Yes	Normal	Hard
Pre-presbyopic	Hypermetrope	Yes	Normal	None
Presbyopic	Myope	Yes	Normal	Hard
Presbyopic	Hypermetrope	Yes	Normal	None

Testes possíveis

Age = Young	2/2
Age = Pre-presbyopic	1/2
Age = Presbyopic	1/2
Spectacle prescription = Myope	3/3
Spectacle prescription = Hypermetrope	1/3

Regra refinada

99

```
if Astigmatism = yes
  and Tear production rate = normal
  and ???
then Lenses = hard
```

Age	Specta prescription	Astigmatism	Tear prod. rate	Lenses
Young	Myope	Yes	Normal	Hard
Young	Hypermetrope	Yes	Normal	hard
Pre-presbyopic	Myope	Yes	Normal	Hard
Pre-presbyopic	Hypermetrope	Yes	Normal	None
Presbyopic	Myope	Yes	Normal	Hard
Presbyopic	Hypermetrope	Yes	Normal	None

Testes possíveis

Age = Young 2/2

Age = Pre-presbyopic 1/2

Age = Presbyopic 1/2

Spectacle prescription = Myope 3/3

Spectacle prescription = Hypermetrope 1/3

maior
cobertura

Exemplo: lentes de contato

- Regra final:

```
if Astigmatism = yes  
and Tear production rate = normal  
and Spectacle prescription = Myope  
then Lenses = hard
```

- Segunda regra para a classe **hard** (gerada a partir dos exemplos não cobertos pela primeira)

```
if Age = young  
and Astigmatism = yes  
and Tear production rate = normal  
then Lenses = hard
```

Algoritmo PRISM

Algoritmo PRISM(X, y)

Regras $\leftarrow \emptyset$

para cada classe c_j em y

$X' \leftarrow X$

$R \leftarrow (\emptyset \rightarrow c_j)$

enquanto for possível melhorar R com X'

para cada X_i não incluído em R

$R_i \leftarrow$ adicione X_i ao corpo de R

$R \leftarrow$ a melhor regra R_i

remova de X' as instâncias cobertas por R

adicione R ao conjunto Regras

Medidas de qualidade de regras (2)

- Cobertura:
 - Mede o grau de cobertura de uma regra

$$Cov(R) = f_b$$

- Sensitividade (também chamada *recall*)
 - Mede a capacidade de classificar corretamente exemplos de uma classe

$$Sens(R) = \frac{f_{hb}}{f_h}$$

Medidas de qualidade de regras (2)

- Novidade:
 - Mede a independência entre corpo e cabeça; quanto maior a novidade, maior a probabilidade de existir uma correlação inesperada entre H e B

$$Nov(R) = 16 \times (f_{hb} - f_h \times f_b)^2$$

Medidas de qualidade de regras (2)

- Laplace:
 - Uma correção da precisão que considera o número de classes distintas

$$LAcc(R) = \frac{f_{hb} + 1}{f_b + N_{Cl}}$$