Árvores de Decisão

Huei Diana Lee

Inteligência Artificial
CECE/UNIOESTE-FOZ

Exemplo

- Um cientista está pesquisando a audição das formigas
- A formiga está parada e o cientista dá um grito. A formiga sai correndo
- Ele então arranca uma das pernas da formiga, e dá outro grito, da mesma intensidade que o primeiro. A formiga corre, mas não tão depressa como anteriormente
- O cientista então arranca as outras pernas e dá outro grito. A formiga não corre. Então ele conclui que as formigas ouvem pelas pernas
- A conclusão, baseada no experimento do cientista, **não** é válida **porque ele escolheu mal as características relevantes** na determinação da audição das formigas

Árvores de Decisão

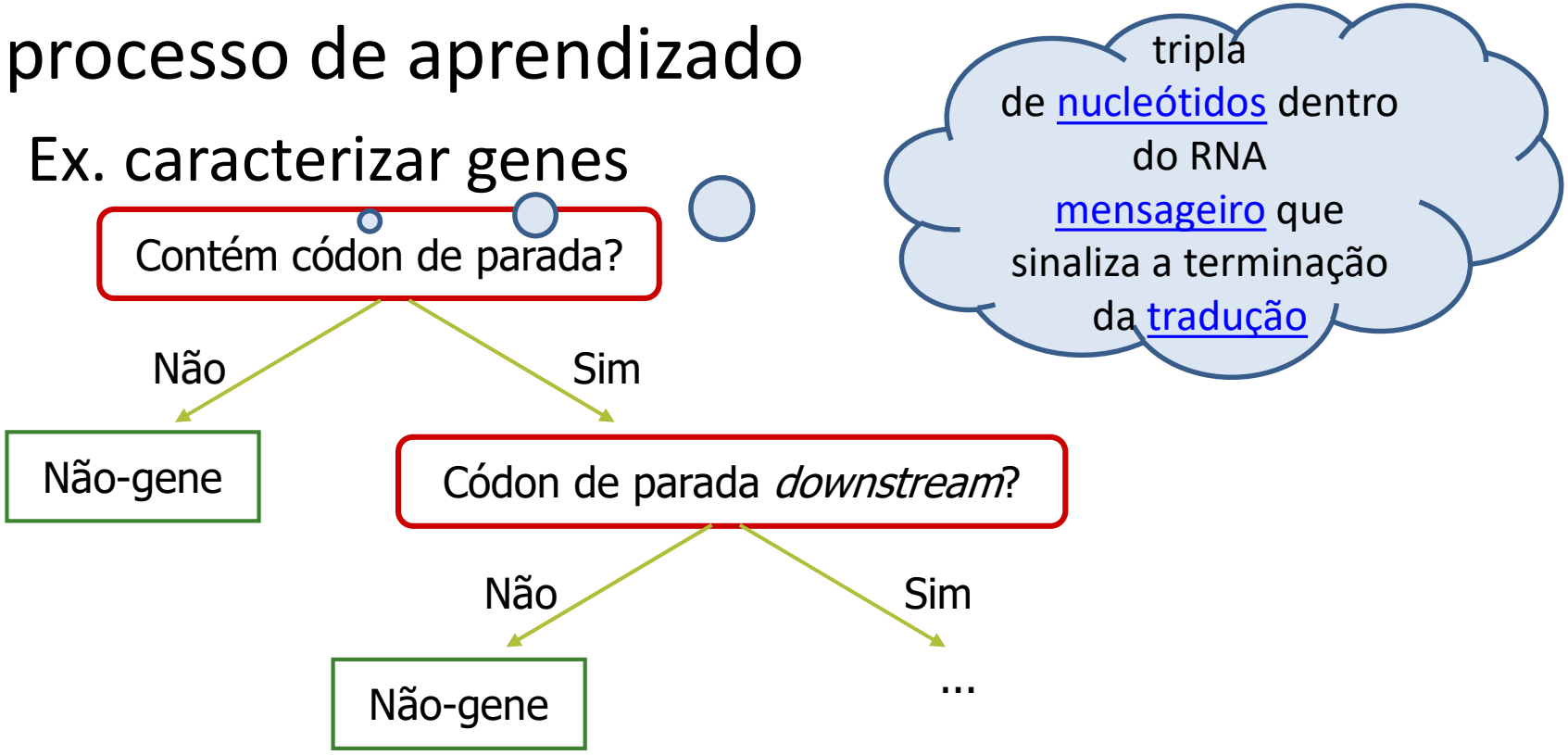
- **Árvore de Decisão** (AD): usa estratégia dividir para conquistar para resolver problema de decisão
  - Problema complexo é dividido em problemas mais simples, aos quais a mesma estratégia é aplicada
  - Soluções dos subproblemas são então combinadas na forma de uma **árvore**

Em problemas de regressão são denominadas **Árvores de Regressão**, mas, dadas suas semelhanças, usaremos o termo Árvore de Decisão de maneira genérica

Árvores de Decisão

- **Estrutura da árvore** é determinada por processo de aprendizado

  Ex. caracterizar genes



Árvore de Decisão

Formalmente: grafo direcionado acíclico em que cada nó é:

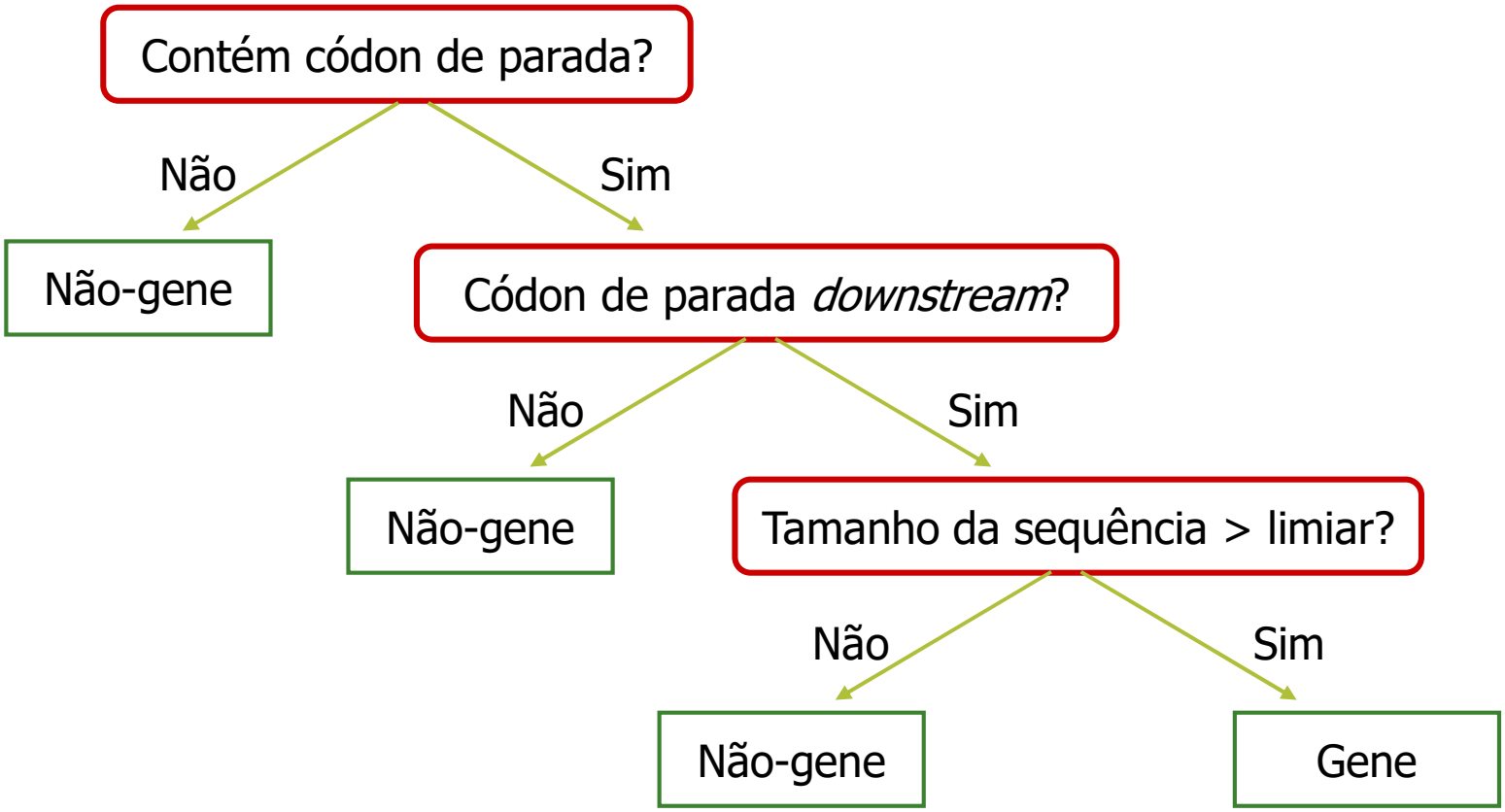
Nó de divisão:

- Possui dois ou mais sucessores
- Contém *teste condicional* baseado nos valores de atributos
- **Padrão**: testes univariados e um atributo
- *Ex: Idade > 18, Profissão* $\in$ *{professor, estudante},* $0,3 + 0,2\,\boldsymbol{x}^1 - 0,5\,\boldsymbol{x}^2 \leq 0$

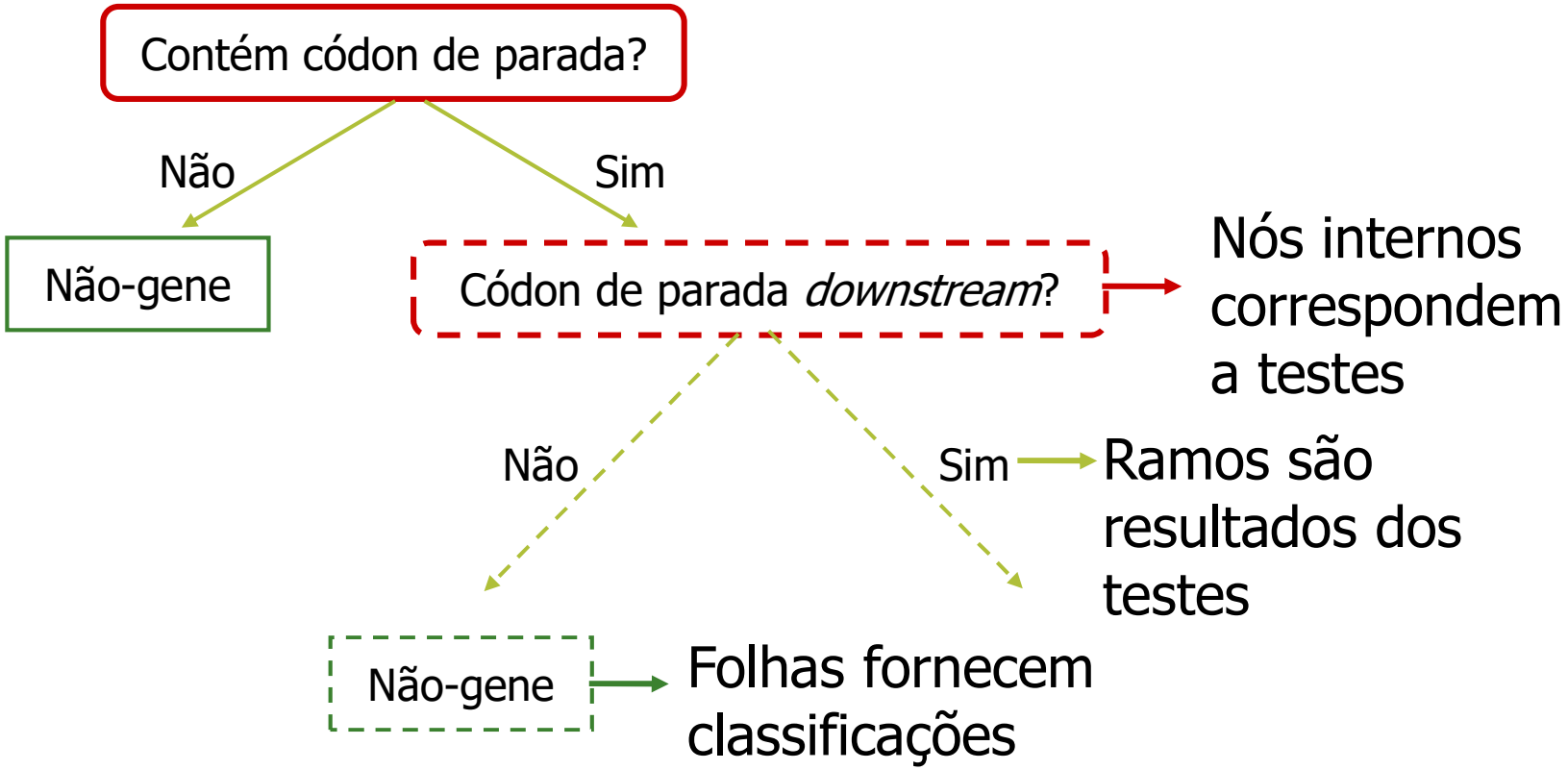
Nó folha:

- É rotulado com uma *função* que considera valores da variável alvo dos exemplos que chegam na folha
- Classificação: **moda**
- Regressão: **média**

Exemplo: determinar gene

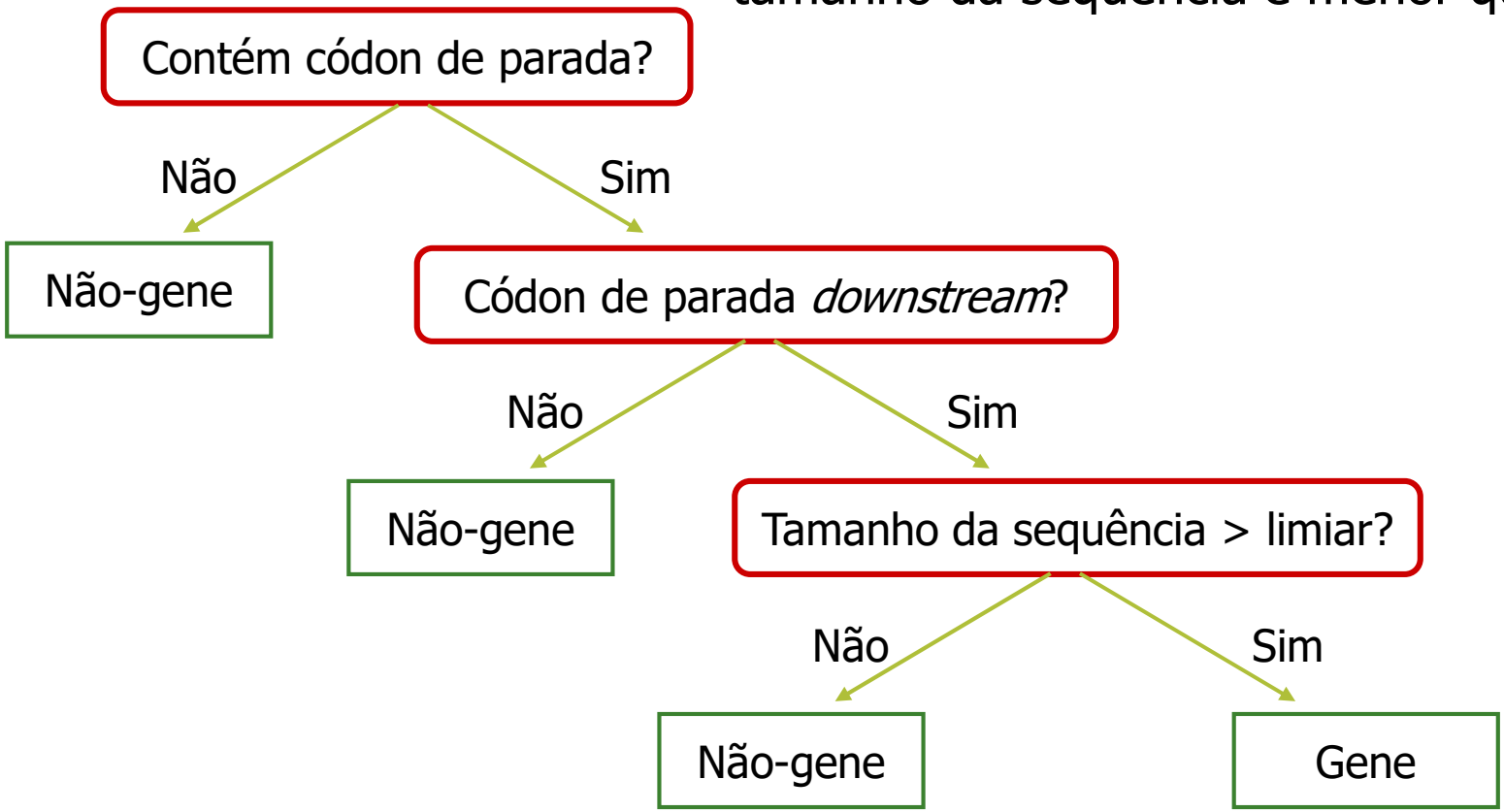


Exemplo: determinar gene



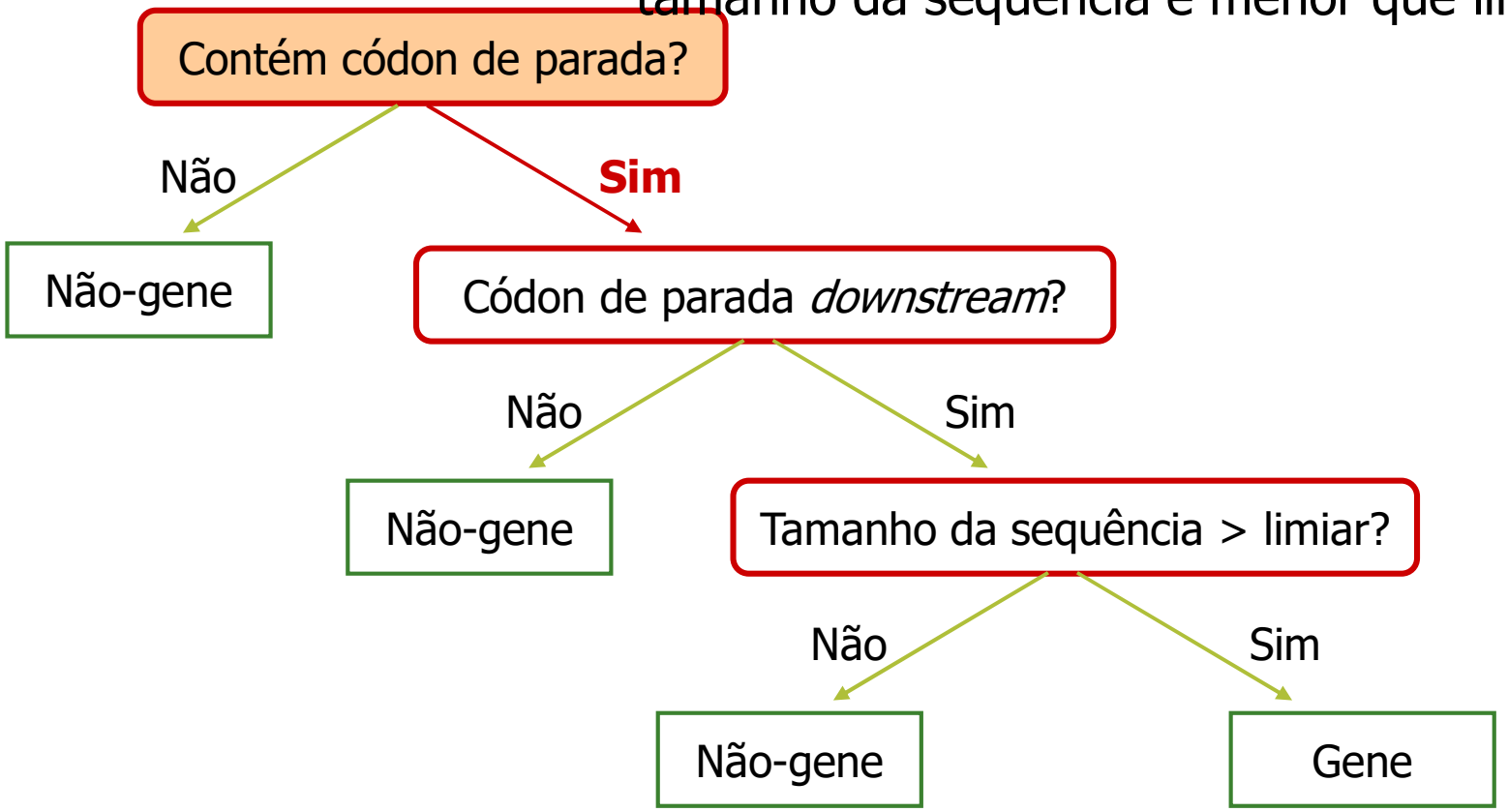
Exemplo: determinar gene

Novo caso: Contém códon de parada dowstream e tamanho da sequência é menor que limiar



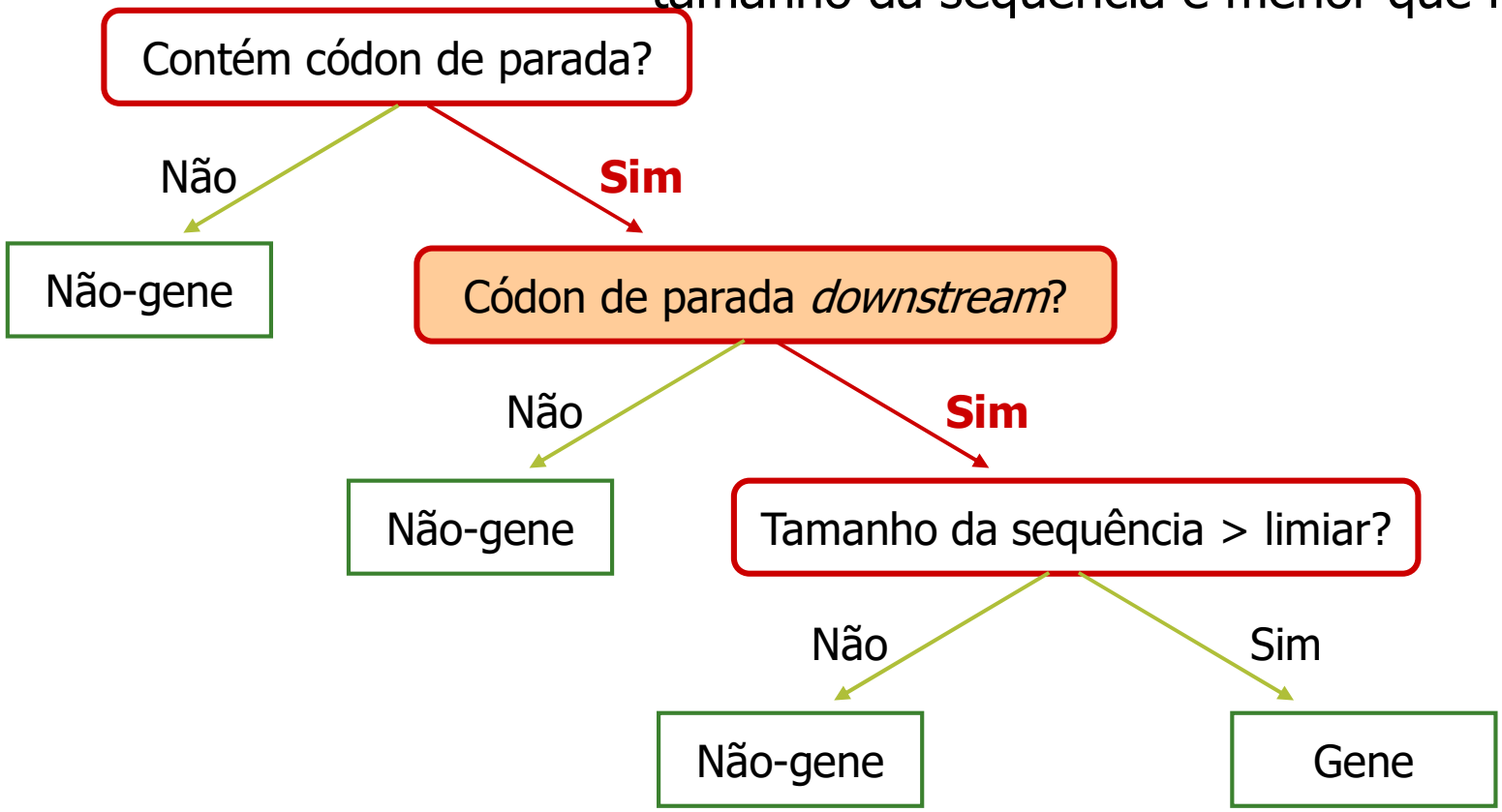
Exemplo: determinar gene

Novo caso: Contém códon de parada dowstream e tamanho da sequência é menor que limiar



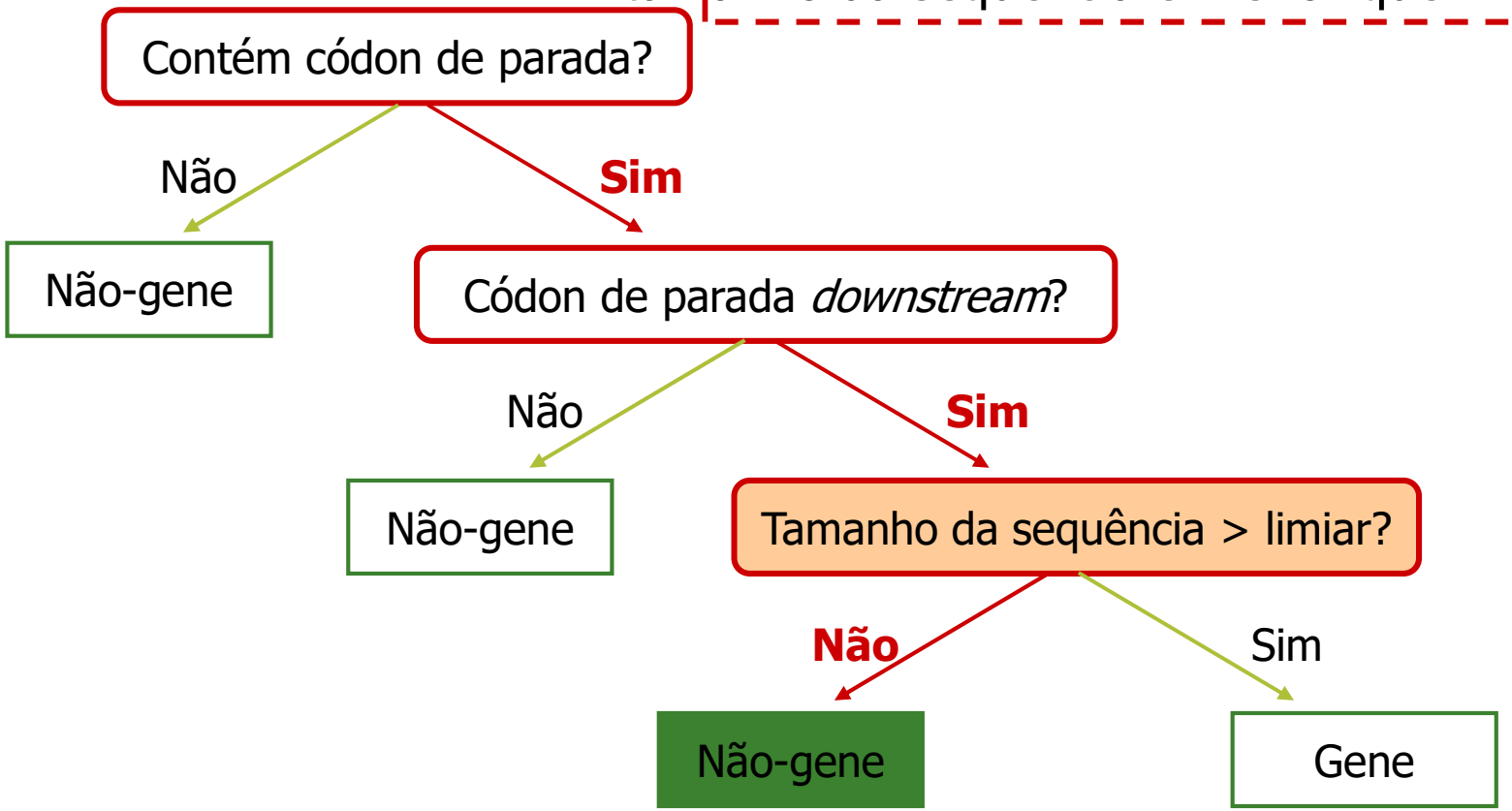
Exemplo: determinar gene

Novo caso: Contém códon de parada dowstream e tamanho da sequência é menor que limiar



Exemplo: determinar gene

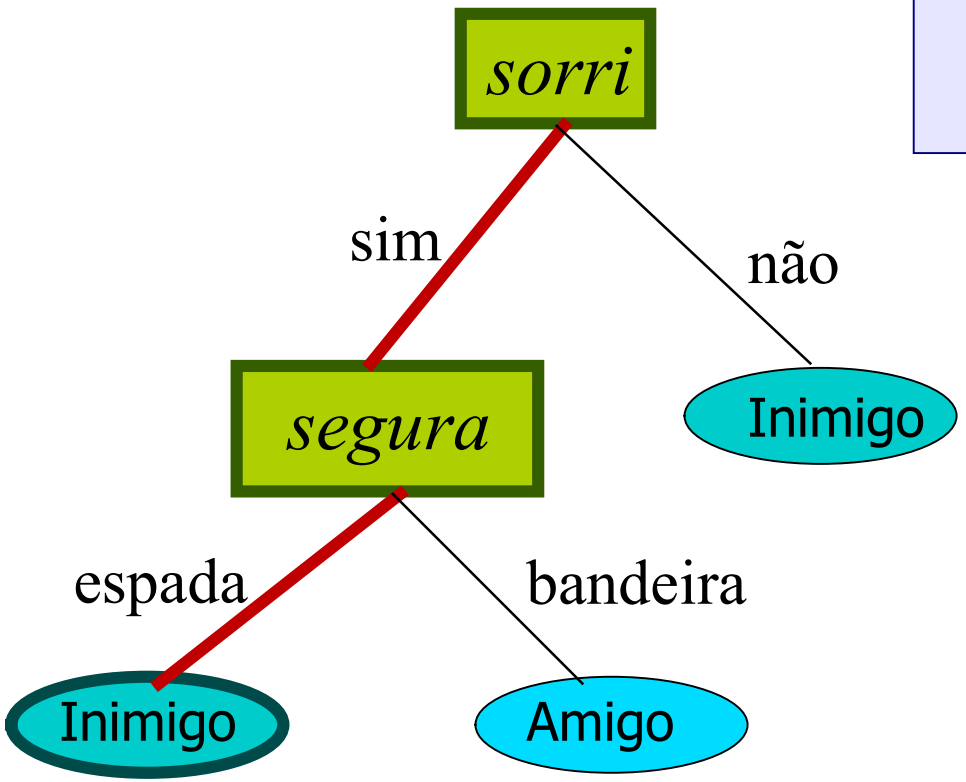
Novo caso: Contém códon de parada dowstream e tamanho da sequência é menor que limiar



Árvore de Decisão

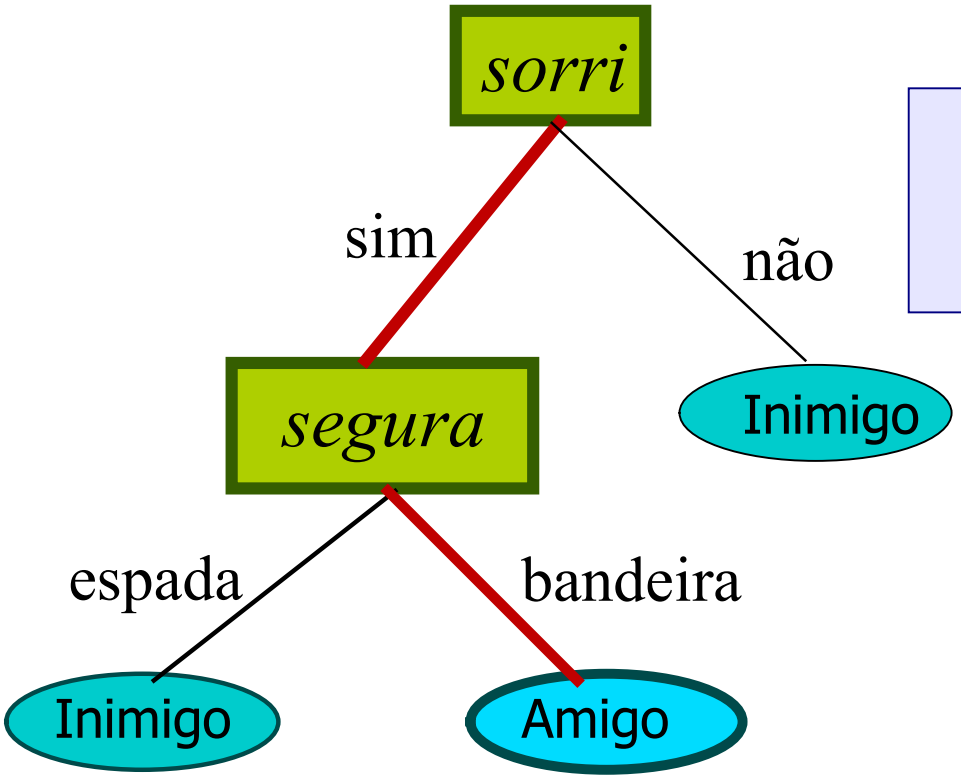
- Espaço de hipóteses de AD enquadra-se no formalismo **Forma Normal Disjuntiva** (FND)
  - Classificador codifica uma FND para cada classe
  - Percurso de raiz a folha (ramo): conjunções de condições
  - Ramos individuais: disjunções
  - Cada ramo forma uma regra com uma parte condicional e uma conclusão

Representação



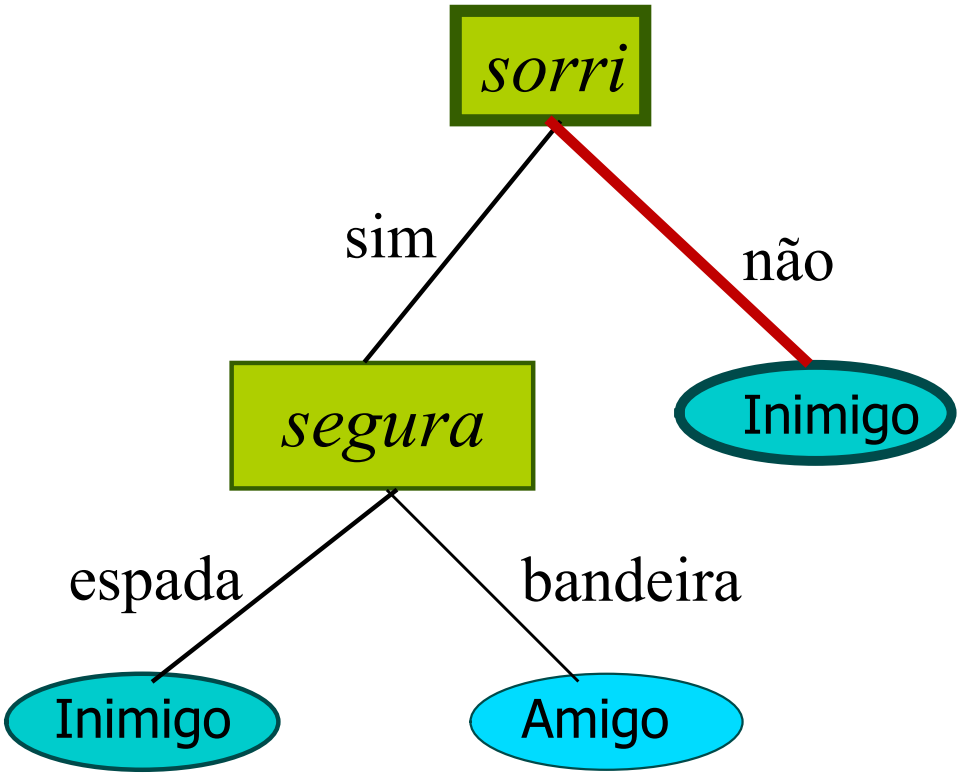
Se sorri **e** segura espada
Então é inimigo

Representação



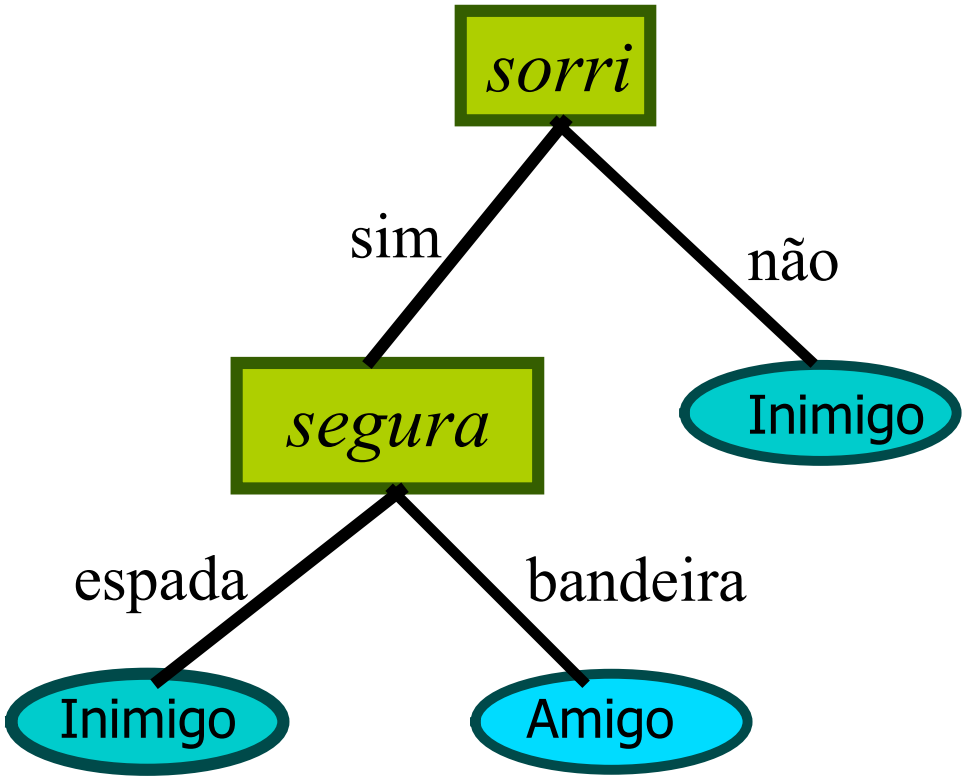
Se sorri **e** segura bandeira
Então é amigo

Representação



Se não sorri
Então é inimigo

Representação



Se sorri **e** segura espada
Então é inimigo
Ou
Se sorri **e** segura bandeira
Então é amigo
Ou
Se não sorri
Então é inimigo

Indução de Árvore de Decisão

Construir árvore a partir de dados – existem vários algoritmos:

- Algoritmo de Hunt
  - Um dos primeiros
  - Base de vários algoritmos atuais
- CART
- CHAID
- ID3, C4.5
- SLIQ, SPRINT

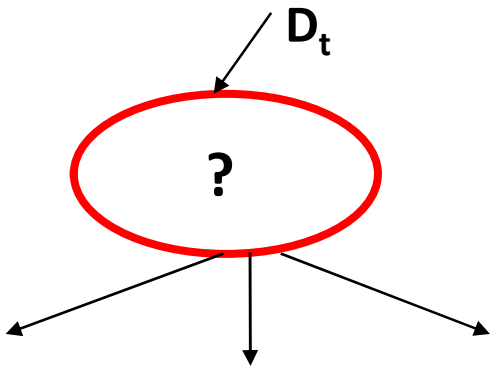
Algoritmo de Hunt

Seja D_t o conjunto de objetos de treinamento que atingem o nó t

Se todos os objetos de $D_t \in$ mesma classe y_t
    Então t é um nó folha rotulado como y_t
Se $D_t = \varnothing$
    Então t é um nó folha rotulado pela classe *default*, y_d
Se os objetos de $D_t \in$ mais de uma classe
    Então dividir dados em subconjuntos com um atributo teste
    Aplicar procedimento a cada subconjunto gerado

Algoritmo de Hunt

Emprego	Estado	Renda	Crédito
Sim	Solteiro	9500	Sim
Não	Casado	8000	Não
Não	Solteiro	7000	Não
Sim	Solteiro	12000	Sim
Não	Divorciado	9000	Sim
Não	Casado	6000	Não
Não	Divorciado	4000	Não
Não	Solteiro	8500	Sim
Não	Casado	7500	Não
Não	Divorciado	8000	Não



Algoritmo de Hunt

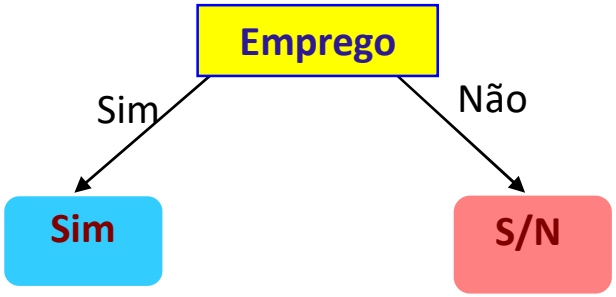
Emprego	Estado	Renda	Crédito
Sim	Solteiro	9500	Sim
Não	Casado	8000	Não
Não	Solteiro	7000	Não
Sim	Casado	12000	Sim
Não	Divorciado	9000	Sim
Não	Casado	6000	Não
Sim	Divorciado	4000	Sim
Não	Solteiro	8500	Sim
Não	Casado	7500	Não
Não	Divorciado	8000	Não

Não

Classe *default*

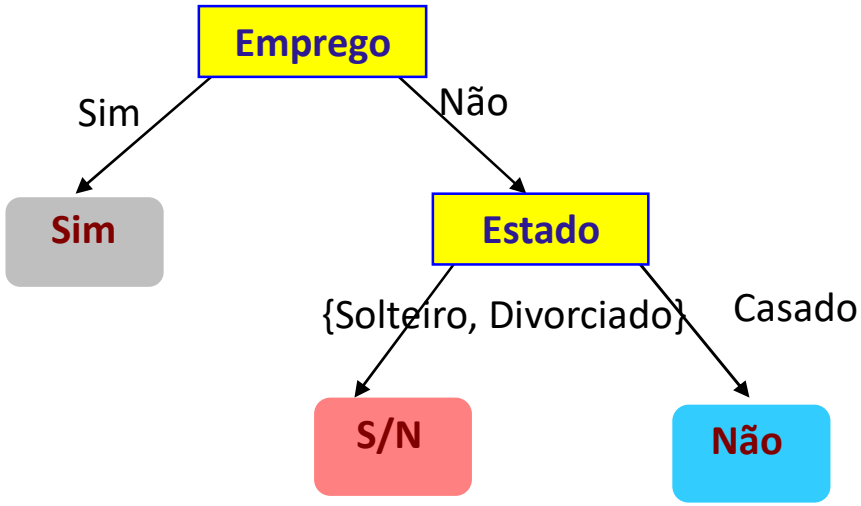
Algoritmo de Hunt

Emprego	Estado	Renda	Crédito
Sim	Solteiro	9500	Sim
Não	Casado	8000	Não
Não	Solteiro	7000	Não
Sim	Casado	12000	Sim
Não	Divorciado	9000	Sim
Não	Casado	6000	Não
Sim	Divorciado	4000	Sim
Não	Solteiro	8500	Sim
Não	Casado	7500	Não
Não	Divorciado	8000	Não



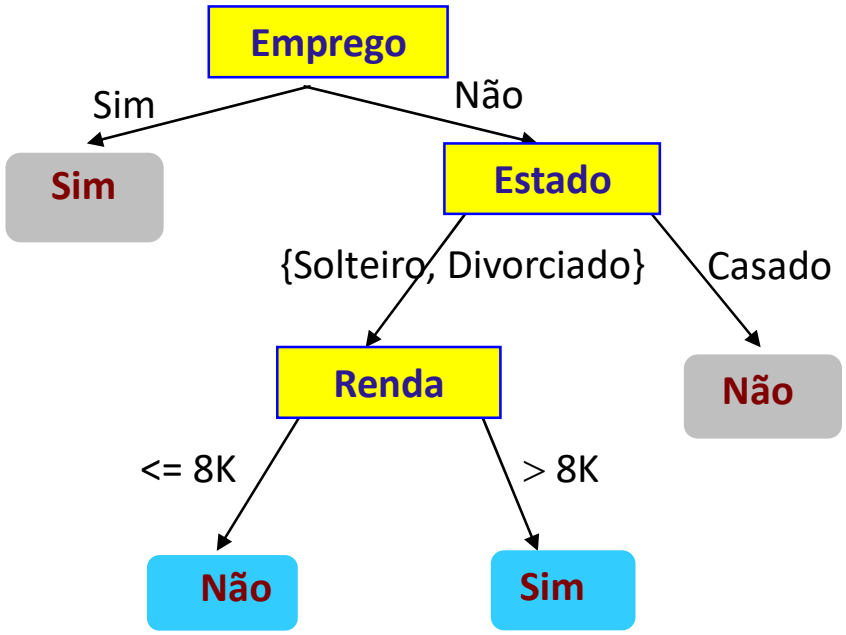
Algoritmo de Hunt

Emprego	Estado	Renda	Crédito
Sim	Solteiro	9500	Sim
Não	Casado	8000	Não
Não	Solteiro	7000	Não
Sim	Casado	12000	Sim
Não	Divorciado	9000	Sim
Não	Casado	6000	Não
Sim	Divorciado	4000	Sim
Não	Solteiro	8500	Sim
Não	Casado	7500	Não
Não	Divorciado	8000	Não



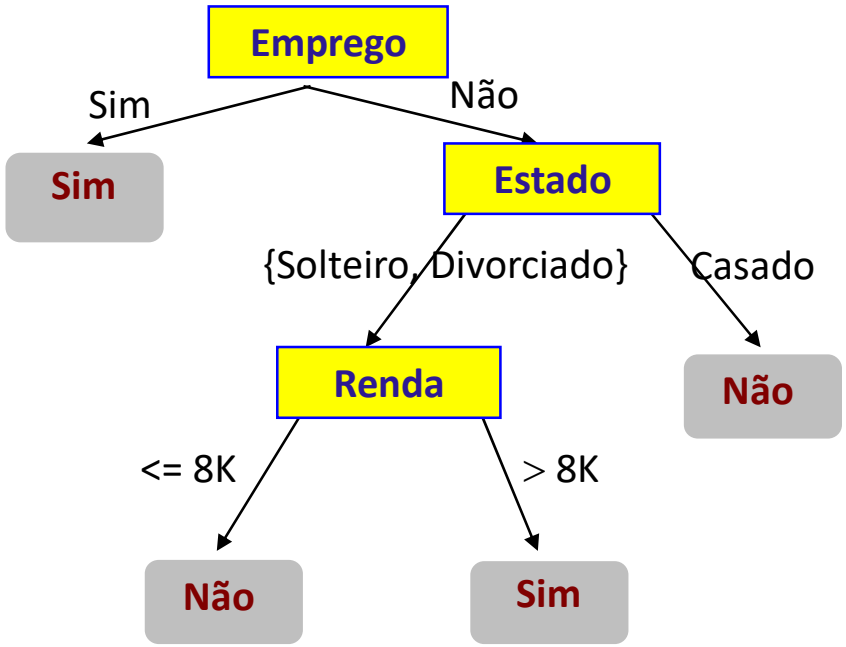
Algoritmo de Hunt

Emprego	Estado	Renda	Crédito
Sim	Solteiro	9500	Sim
Não	Casado	8000	Não
Não	**Solteiro**	7000	Não
Sim	Casado	12000	Sim
Não	**Divorciado**	9000	Sim
Não	Casado	6000	Não
Sim	Divorciado	4000	Sim
Não	**Solteiro**	8500	Sim
Não	Casado	7500	Não
Não	**Divorciado**	8000	Não



Algoritmo de Hunt

Emprego	Estado	Renda	Crédito
Sim	Solteiro	9500	Sim
Não	Casado	8000	Não
Não	**Solteiro**	7000	Não
Sim	Casado	12000	Sim
Não	**Divorciado**	9000	Sim
Não	Casado	6000	Não
Sim	Divorciado	4000	Sim
Não	**Solteiro**	8500	Sim
Não	Casado	7500	Não
Não	**Divorciado**	8000	Não



Indução de Árvore de Decisão

Algoritmo genérico:

GeraÁrvore(D) - **Algoritmo** para construção de AD

Entrada: conjunto de treinamento **D** = $\{(\mathbf{x}_i, y_i), i=1,...,n\}$

Saída: AD

/* Função ***GeraÁrvore(D)*** */

Se critério de parada **então**

  Retorna nó folha com rótulo que maximiza função de custo

Escolha o atributo que maximiza o critério de divisão em **D**

Para cada partição $\mathbf{D}_i$ baseada nos valores do atributo **faça**

  Induzir subárvore $Árvore_i$ = $GeraÁrvore(\mathbf{D}_i)$

Retorna *Árvore* com nó de decisão baseado no atributo escolhido e descendentes $Árvore_i$

Indução de Árvore de Decisão

- Construir AD mínima (número mínimo de nós) condizente com conjunto de dados é problema NP-completo

- Algoritmos usualmente usam heurísticas que olham um passo à frente:
  - Estratégia gulosa
  - Suscetível a encontrar ótimo local
  - Mas permite construção de AD em tempo linear

Indução de Árvore de Decisão

Decisões importantes:

- Como dividir os objetos?

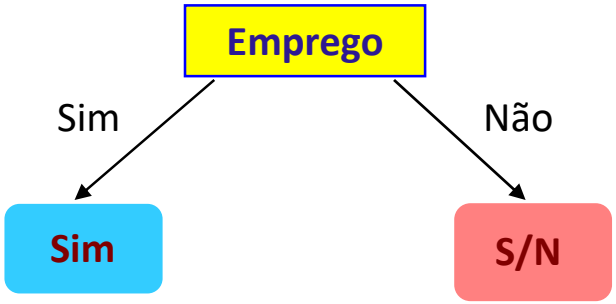
  Método para escolha do atributo de teste / medida para avaliar qualidade de atributo escolhido

- Quando parar de dividir os objetos?

Como dividir os atributos?

- Valores de atributos particionam os objetos
- Como divisão é feita depende:
  - Do tipo do atributo
  - Do número de divisões suportada pelo algoritmo

Emprego	Estado	Renda	Crédito
Sim	Solteiro	9500	Sim
Não	Casado	8000	Não
Não	Solteiro	7000	Não
Sim	Casado	12000	Sim
Não	Divorciado	9000	Sim
Não	Casado	6000	Não
Sim	Divorciado	4000	Sim
Não	Solteiro	8500	Sim
Não	Casado	7500	Não
Não	Solteiro	9000	Sim



Como dividir os atributos?

- **Qualitativos**: usualmente

  #ramos = #possíveis valores

- **Quantitativos**: usualmente

  Comparação (A < valor)

  - Escolher posição (valor) que gera melhor partição
  - Ponto de referência

Que atributo escolher para divisão?

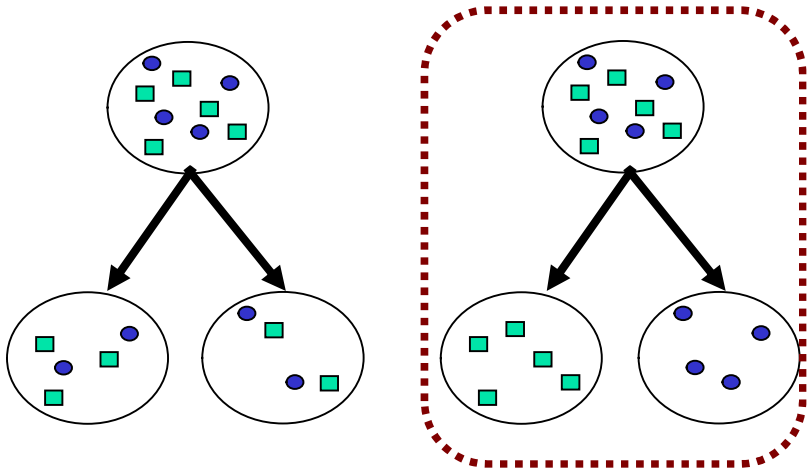
Estratégias para escolha do atributo:

- Aleatória
- Menos valores
- Mais valores
- Ganho de informação
- Ganho máximo
- Razão de ganho
- Índice Gini

Que atributo escolher para divisão?

Regras de divisão para **classificação**:

- Guiada por medida de ***goodness*** *of split*
  - Indica quão bem um atributo discrimina as classes
  - Selecionar atributo que maximiza a medida
- Funciona como heurística que olha um passo para frente



Qualidades Desejadas para uma Medida de Pureza

- Quando um nó é puro, a medida deve ser ZERO
- Quando a impureza é máxima (todas as classes são igualmente prováveis), a medida deve máxima
- Ideia de pureza:

Para duas classes, um bom atributo divide os exemplos em subconjuntos que idealmente são "todos positivos" ou "todos negativos"

Computando Informação

A informação é computada em bits:

- Dada uma distribuição de probabilidade, a informação requerida para predizer um evento é a **entropia** da distribuição
- Entropia é importante para calcular a medida de Ganho de Informação
- Entropia provê a informação requerida em bits (isso pode envolver frações de bits!)

Trabalho 4

1. Pesquisar sobre as medidas:
   - Ganho máximo
   - Razão de ganho
   - Índice Gini
2. Fazer um breve relato (contendo identificação e referências bibliográficas) de até 3 páginas.
3. Trabalho Individual
4. Data da Entrega: até 12hs de 31/05/21

Entropia

Entropia (desordem, impureza) de um conjunto de exemplos, S, relativa a classificação binária é:

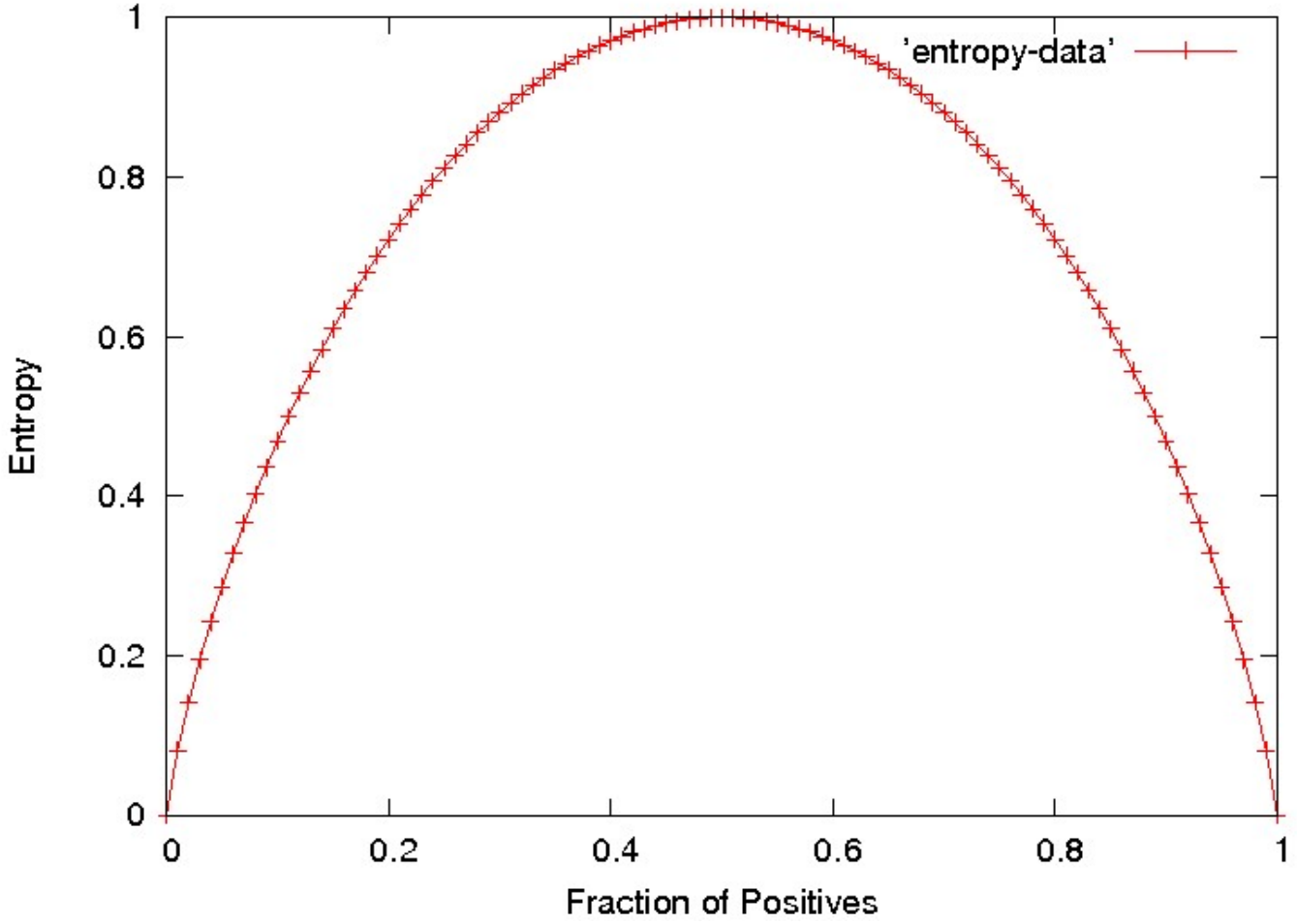
$$Entropy(S) = -p_1 \log_2(p_1) - p_0 \log_2(p_0)$$

onde

p_1 é a fração de exemplos positivos em S e

p_0 é a fração de negativos em S

Entropia para Classificação Binária



Entropia em AD

Usada como medida de impureza para medir a **aleatoriedade** (dificuldade para predizer) do **atributo alvo**

- Para caso binário:
  - A entropia é 0 se todos elementos pertencem à mesma classe, ou seja, pureza máxima
  - A entropia é 1 quanto a coleção contém número igual de exemplos positivos e negativos
- A cada nó de decisão, o atributo que mais reduz a aleatoriedade do alvo é escolhido para divisão

Entropia em AD

Sejam p e q o número de objetos de duas classes diferentes em um conjunto de dados D

$$H(D) = -\frac{p}{p+q}\log\left(\frac{p}{p+q}\right) - \frac{q}{p+q}\log\left(\frac{q}{p+q}\right)$$

Probabilidade é computada a partir do conjunto de treinamento D

Entropia em AD

Entropia pode ser usada em problemas com mais que duas classes (k classes):

$$H(D) = \sum_{i=1}^{k} - p_i \log_2 (p_i)$$

Entropia em AD

Se atributo A com v valores é selecionado, a árvore resultante tem um conteúdo de informação esperado de:

$$H(A,D) = \sum_{i=1}^{v} \frac{p_i + q_i}{p+q} H(D_i)$$

onde p_i e q_i: números de objetos em cada classe na partição D_i

Ganho de informação

Ganho de informação alcançado selecionando A para divisão:

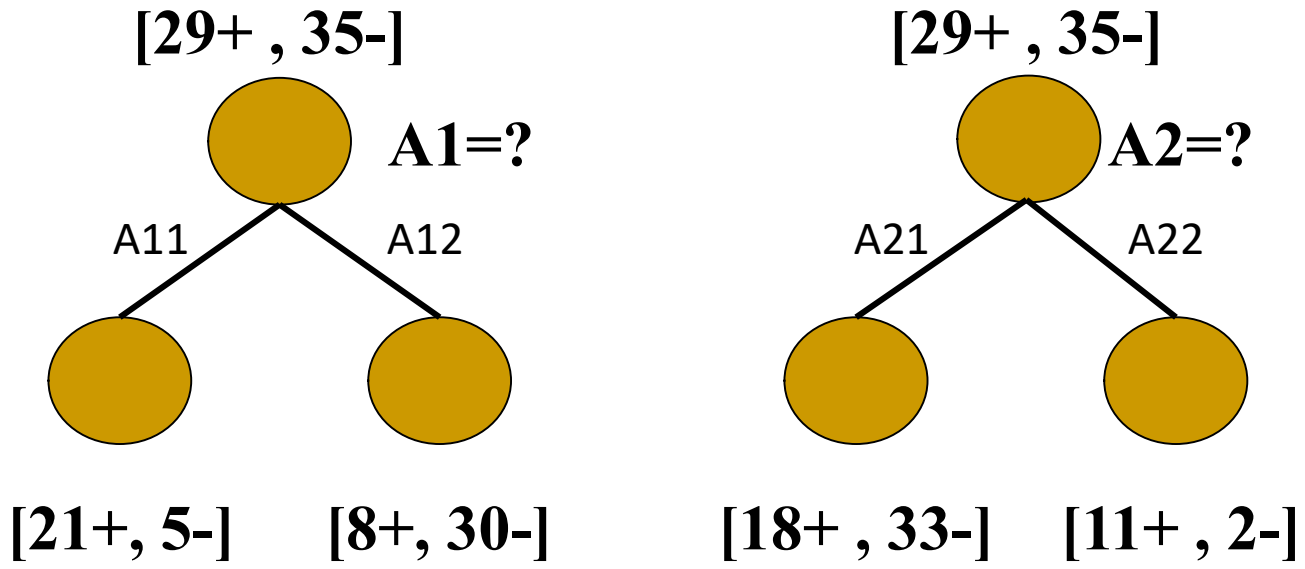
$$IG(A,D) = H(D) - H(A,D)$$

$IG(A,D)$ = redução esperada da entropia devido à "classificação" de acordo com o atributo A

$$IG(A, D) \equiv H(D) - \sum_{v \in Valores(A)} \frac{|D_v|}{|D|} H(D_v)$$

Exemplo: que atributo escolher?

Usar o critério de ganho de informação para decidir!



Exemplo: que atributo escolher?

$$H(D) = -\frac{p}{p+q}\log\left(\frac{p}{p+q}\right) - \frac{q}{p+q}\log\left(\frac{q}{p+q}\right)$$

$$H(A,D) = \sum_{i=1}^{v} \frac{p_i + q_i}{p+q} H(D_i)$$

$$IG(A,D) = H(D) - H(A,D)$$

Ganho de informação:

- D = {29 +, 35 -}
  - H(D)= -(29/64)*$\log_2$(29/64) -(35/64)*$\log_2$(35/64) = 0,994
- De acordo com A1:
  - D_{A11} = {21 +, 5 -}
    - H(D_{A11})= -(21/26)*$\log_2$(21/26) -(5/26)*$\log_2$(5/26) = 0,706
  - D_{A12} = {8 +, 30 -}
    - H(D_{A12})= -(8/38)*$\log_2$(8/38) -(30/38)*$\log_2$(30/38) = 0,742
  - IG(A1,D) = 0,994 – ((26/64)*0,706+(38/64)*0,742) = 0,266

Exemplo: que atributo escolher?

Ganho de informação:

- De acordo com A2:
  - D_{A21} = {18 +, 33 -}
    - $H(D_{21})$= -(18/51)*$\log_2$(18/51) -(33/51)*$\log_2$(33/51) = 0,937
  - D_{A22} = {11 +, 2 -}
    - $H(D_{A22})$= -(11/13)*$\log_2$(11/13) -(2/13)*$\log_2$(2/13) = 0,619
  - IG(A2,D) = 0,994 – ((51/64)*0,937+(13/64)*0,619) = 0,121

A1 traz maior ganho de informação (0,266) do que A2 (0,121).
Assim, A1 é escolhido.

Exemplo ilustrativo

Conjunto de dados play tênis

Decidir quando jogar dadas condições de tempo

Tempo	Temperatura	Umidade	Vento	Joga
Chuvoso	71	91	Sim	Não
Ensolarado	69	70	Não	Sim
Ensolarado	80	90	Sim	Não
Nublado	83	86	Não	Sim
Chuvoso	70	96	Não	Sim
Chuvoso	65	70	Sim	Não
Nublado	64	65	Sim	Sim
Nublado	72	90	Sim	Sim
Ensolarado	75	70	Sim	Sim
Chuvoso	68	80	Não	Sim
Nublado	81	75	Não	Sim
Ensolarado	85	85	Não	Não
Ensolarado	72	95	Não	Não
Chuvoso	75	80	Não	Sim

Exemplo ilustrativo

Conjunto de dados play tênis

Tempo	Temperatura	Umidade	Vento	Joga
Chuvoso	71	91	Sim	Não
Ensolarado	69	70	Não	Sim
Ensolarado	80	90	Sim	Não
Nublado	83	86	Não	Sim
Chuvoso	70	96	Não	Sim
Chuvoso	65	70	Sim	Não
Nublado	64	65	Sim	Sim
Nublado	72	90	Sim	Sim
Ensolarado	75	70	Sim	Sim
Chuvoso	68	80	Não	Sim
Nublado	81	75	Não	Sim
Ensolarado	85	85	Não	Não
Ensolarado	72	95	Não	Não
Chuvoso	75	80	Não	Sim

Entropia da classe para todo o conjunto de exemplos:

p(Joga = Sim) = 9/14 = 0,640
p(Joga = Não) = 5/14 = 0,360

$H(Joga) = -9/14\log_2(9/14) -5/14\log_2(5/14) = 0,940$

Exemplo ilustrativo

Conjunto de dados `play`

Tempo	Temperatura	Umidade	Vento	Joga
Chuvoso	71	91	Sim	Não
Ensolarado	69	70	Não	Sim
Ensolarado	80	90	Sim	Não
Nublado	83	86	Não	Sim
Chuvoso	70	96	Não	Sim
Chuvoso	65	70	Sim	Não
Nublado	64	65	Sim	Sim
Nublado	72	90	Sim	Sim
Ensolarado	75	70	Sim	Sim
Chuvoso	68	80	Não	Sim
Nublado	81	75	Não	Sim
Ensolarado	85	85	Não	Não
Ensolarado	72	95	Não	Não
Chuvoso	75	80	Não	Sim

Joga	Ensolarado	Nublado	Chuvoso
Sim	2	4	3
Não	3	0	2

IG para atributos nominais:
Ex. atributo **Tempo**: três partições
1º passo: estimar probabilidades de observar classes dado cada valor

$p(Joga|Ensolarado) = 2/5$
$p(\neg Joga|Ensolarado) = 3/5$
$H(Joga|Ensolarado) = -2/5*\log_2(2/5) \; -3/5*\log_2(3/5) = 0{,}971$

$p(Joga|Nublado) = 4/4$
$p(\neg Joga|Nublado) = 0/4$
$H(Joga|Nublado) = 0{,}00$

$p(Joga|Chuvoso) = 3/5$
$p(\neg Joga|Chuvoso) = 2/5$
$H(Joga|Chuvoso) = 0{,}971$

Exemplo ilustrativo

Conjunto de dados `play`

Tempo	Temperatura	Umidade	Vento	Joga
Chuvoso	71	91	Sim	Não
Ensolarado	69	70	Não	Sim
Ensolarado	80	90	Sim	Não
Nublado	83	86	Não	Sim
Chuvoso	70	96	Não	Sim
Chuvoso	65	70	Sim	Não
Nublado	64	65	Sim	Sim
Nublado	72	90	Sim	Sim
Ensolarado	75	70	Sim	Sim
Chuvoso	68	80	Não	Sim
Nublado	81	75	Não	Sim
Ensolarado	85	85	Não	Não
Ensolarado	72	95	Não	Não
Chuvoso	75	80	Não	Sim

IG para atributos nominais:
Ex. atributo **Tempo**: três partições

2º passo: calcular a entropia ponderada para o atributo Tempo
H(Joga,Tempo) = 5/14 * 0,971 + 4/14 * 0 + 5/14 * 0,971 = 0,693

3º passo: calcular o ganho de informação em dividir o conjunto de acordo com os valores do atributo Tempo
IG(Tempo) = H(Joga) – H(Joga,Tempo)
= 0,940 – 0,693 = **0,247**
⇒ *Conhecendo o valor do atributo Tempo, precisamos de menos informação para codificar o valor do atributo alvo*

Exemplo ilustrativo

Conjunto de dados `play`

Tempo	Temperatura	Umidade	Vento	Joga
Nublado	64	65	Sim	Sim
Chuvoso	65	70	Sim	Não
Chuvoso	68	80	Não	Sim
Ensolarado	69	70	Não	Sim
Chuvoso	70	96	Não	Sim
Chuvoso	71	91	Sim	Não
Nublado	72	90	Sim	Sim
Ensolarado	72	95	Não	Não
Chuvoso	75	80	Não	Sim
Ensolarado	75	70	Sim	Sim
Ensolarado	80	90	Sim	Não
Nublado	81	75	Não	Sim
Nublado	83	86	Não	Sim
Ensolarado	85	85	Não	Não

IG para atributos contínuos:
Buscar partição binária dos valores
- Atributo $\leq$ valor
- Atributo $>$ valor

E aplicar as equações a essas partições

Ex. atributo **Temperatura**

1º passo: definir ponto de corte
- Ordena-se os valores do atributo
- Toma-se a média de dois valores consecutivos: *candidato* a ponto de corte
- Avalia mérito (ex. IG) do ponto de corte
- Escolhe ponto que maximiza mérito

No exemplo, 1º ponto de corte = 64,5 e último ponto de corte = 84

Exemplo ilustrativo

Conjunto de dados `play`

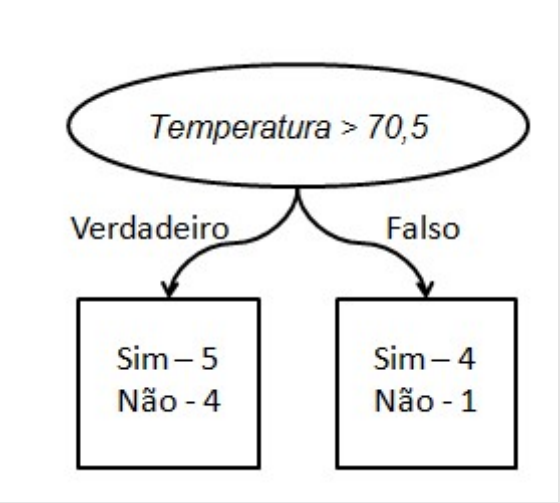
Tempo	Temperatura	Umidade	Vento	Joga
Nublado	64	65	Sim	Sim
Chuvoso	65	70	Sim	Não
Chuvoso	68	80	Não	Sim
Ensolarado	69	70	Não	Sim
Chuvoso	70	96	Não	Sim
Chuvoso	71	91	Sim	Não
Nublado	72	90	Sim	Sim
Ensolarado	72	95	Não	Não
Chuvoso	75	80	Não	Sim
Ensolarado	75	70	Sim	Sim
Ensolarado	80	90	Sim	Não
Nublado	81	75	Não	Sim
Nublado	83	86	Não	Sim
Ensolarado	85	85	Não	Não

IG para atributos contínuos:

Ex. atributo **Temperatura**

2º passo: Escolhido ponto de corte, fazer cálculos de IG correspondentes

Considerando o ponto 70,5:



Exemplo ilustrativo

Conjunto de dados `play`

Tempo	Temperatura	Umidade	Vento	Joga
Nublado	64	65	Sim	Sim
Chuvoso	65	70	Sim	Não
Chuvoso	68	80	Não	Sim
Ensolarado	69	70	Não	Sim
Chuvoso	70	96	Não	Sim
Chuvoso	71	91	Sim	Não
Nublado	72	90	Sim	Sim
Ensolarado	72	95	Não	Não
Chuvoso	75	80	Não	Sim
Ensolarado	75	70	Sim	Sim
Ensolarado	80	90	Sim	Não
Nublado	81	75	Não	Sim
Nublado	83	86	Não	Sim
Ensolarado	85	85	Não	Não

IG para atributos contínuos:
Ex. atributo **Temperatura**
2º passo: considerando o ponto 70,5:

$p(\text{Joga} \mid \text{Temperatura} \leq 70{,}5) = 4/5$
$p(\neg\text{Joga} \mid \text{Temperatura} \leq 70{,}5) = 1/5$

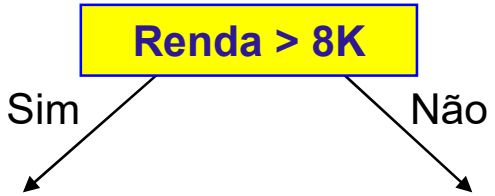
$p(\text{Joga} \mid \text{Temperatura} > 70{,}5) = 5/9$
$p(\neg\text{Joga} \mid \text{Temperatura} > 70{,}5) = 4/9$

$H(\text{Joga} \mid \text{Temperatura} \leq 70{,}5) = -4/5\log_2(4/5) - 1/5\log_2(1/5) = 0{,}721$
$H(\text{Joga} \mid \text{Temperatura} > 70{,}5) = -5/9\log_2(5/9) - 4/9\log_2(4/9) = 0{,}991$
$H(\text{Temperatura}) = 5/14 * 0{,}721 + 9/14 * 0{,}991 = 0{,}895$
$IG(\text{Temperatura}) = 0{,}940 - 0{,}895 = 0{,}045$

Divisão de atributos contínuos

Exemplo:

Emprego	Estado	Renda	Crédito
Sim	Solteiro	9500	Sim
Não	Casado	10000	Não
Não	Solteiro	7000	Não
Não	Casado	12000	Sim
Sim	Divorciado	9000	Sim
Não	Casado	6000	Não
Não	Divorciado	22000	Sim
Sim	Solteiro	8500	Sim
Não	Casado	7500	Não
Não	Solteiro	12500	Sim



Exemplo

Atributo Renda

Emprego	Não		Não		Não		Sim		Sim		Sim		Não		Não		Não		Não	
	Renda																			
Valores ordenados →	60		70		75		85		90		95		100		120		125		220	
Posições de divisão →	55		65		72		80		87		92		97		110		122		172		230	
	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>
Sim	0	3	0	3	0	3	0	3	1	2	2	1	3	0	3	0	3	0	3	0	3	0
Não	0	7	1	6	2	5	3	4	3	4	3	4	3	4	4	3	5	2	6	1	7	0
H	0.881		0.826		0.764		0.690		0.876		0.846		*0.600*		0.690		0.764		0.826		0.881	

Matriz de contagem

Exemplo

Atributo Renda

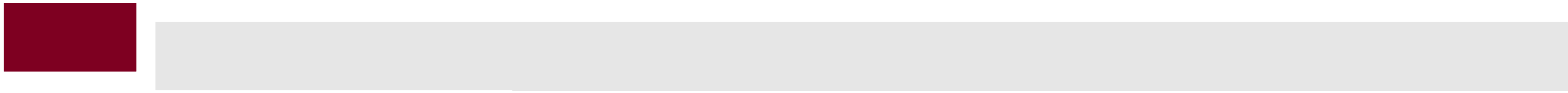
Primeiro candidato: x = 55	Segundo candidato: x = 65
≤ 55	≤ 65
Classe sim: 0	Classe sim: 0
Classe não: 0	Classe não: 1
H = 0	H = 0
> 55	> 65
Classe sim: 3	Classe sim: 3
Classe não: 7	Classe não: 6
H = 0,881	H = 0,918
H = 0x0 + 1x0,881 = 0,881	H = 0,1x0 + 0,9x0,918 = 0,826

Divisão de atributos contínuos

Método pode ser acelerado

Considerar apenas pontos entre dois objetos adjacentes com classes diferentes

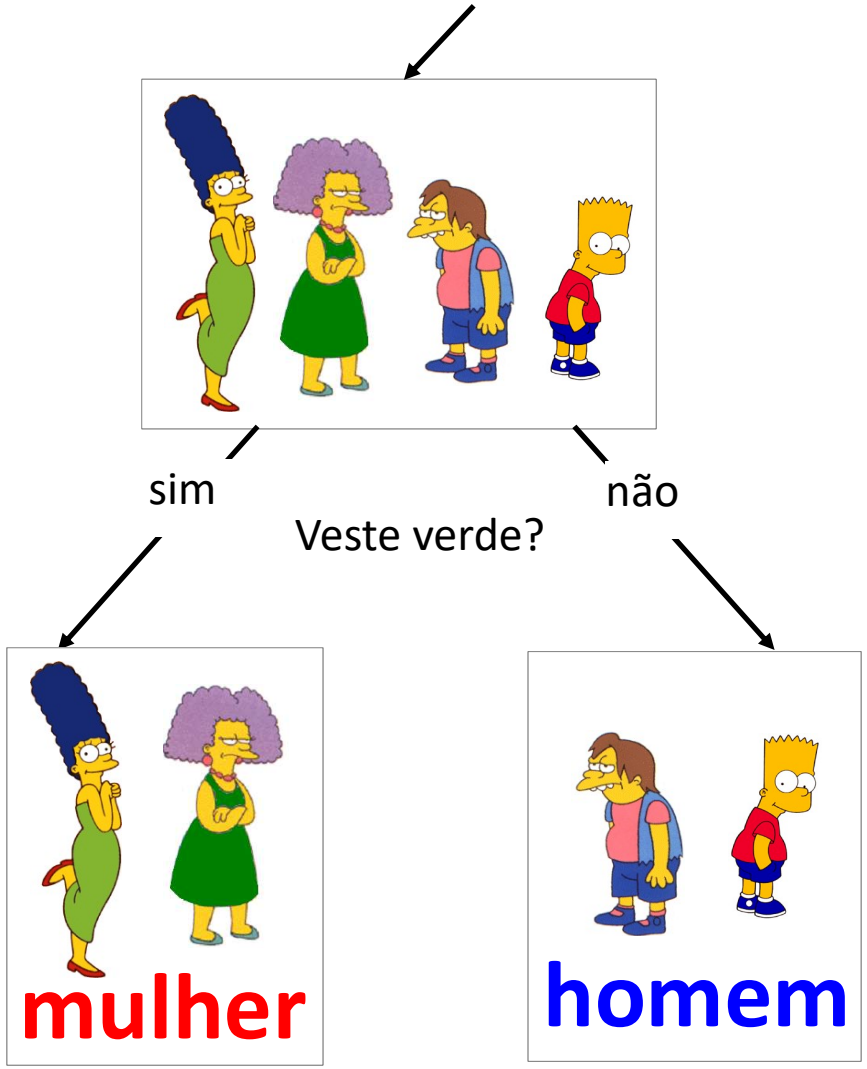
- Não – Sim ou Sim - Não
- Reduz de de 11 para 2 o número de pontos de corte candidatos no exemplo anterior



Exemplo de Superajuste

- Quando se tem poucos dados, há muitas regras de divisão que classificarão perfeitamente os dados, mas que não generalizarão o conhecimento para futuros conjuntos de dados

- Por exemplo, a regra “Veste verde?” classifica perfeitamente os dados, assim como “O nome da mãe é Jacqueline?”, assim como “Tem sapatos azuis”…

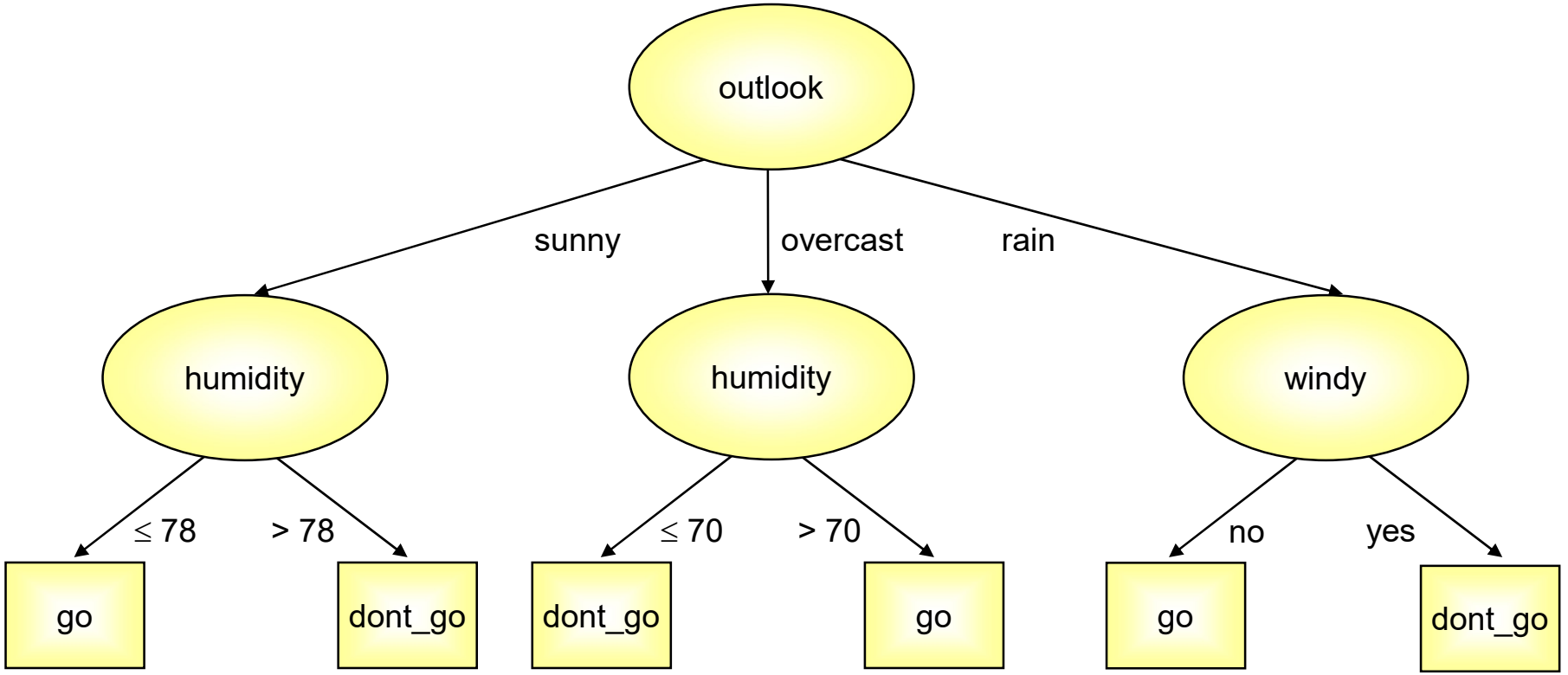


Métodos de Prevenção de Superajuste (Poda)

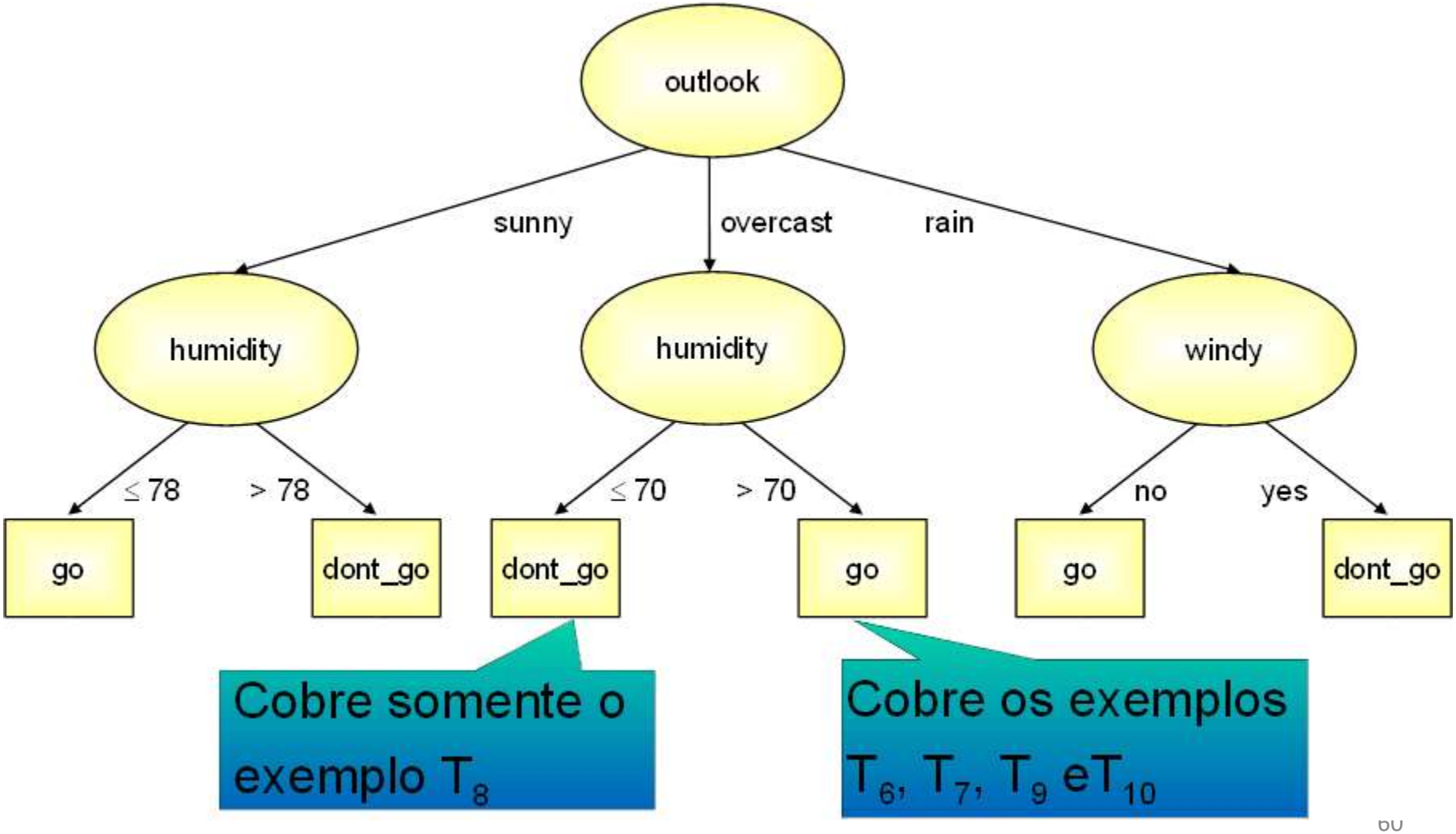
- Duas abordagens básicas para árvores de decisão:
  - **Pré-poda:** para de crescer a árvore quando não há mais dados suficientes para fazer previsões confiáveis
  - **Pós-poda:** constrói a árvore toda, depois as sub-árvores para as quais há não evidência suficiente são removidas
- A folha resultante da poda deve ser marcada com a classe majoritária ou uma distribuição de probabilidade de cada classe
- Todos mantêm ponto de equilíbrio entre tamanho da árvore e estimativa de erro

Exemplo de Poda de AD

Construindo uma AD para o dataset Voyage



Podando a AD

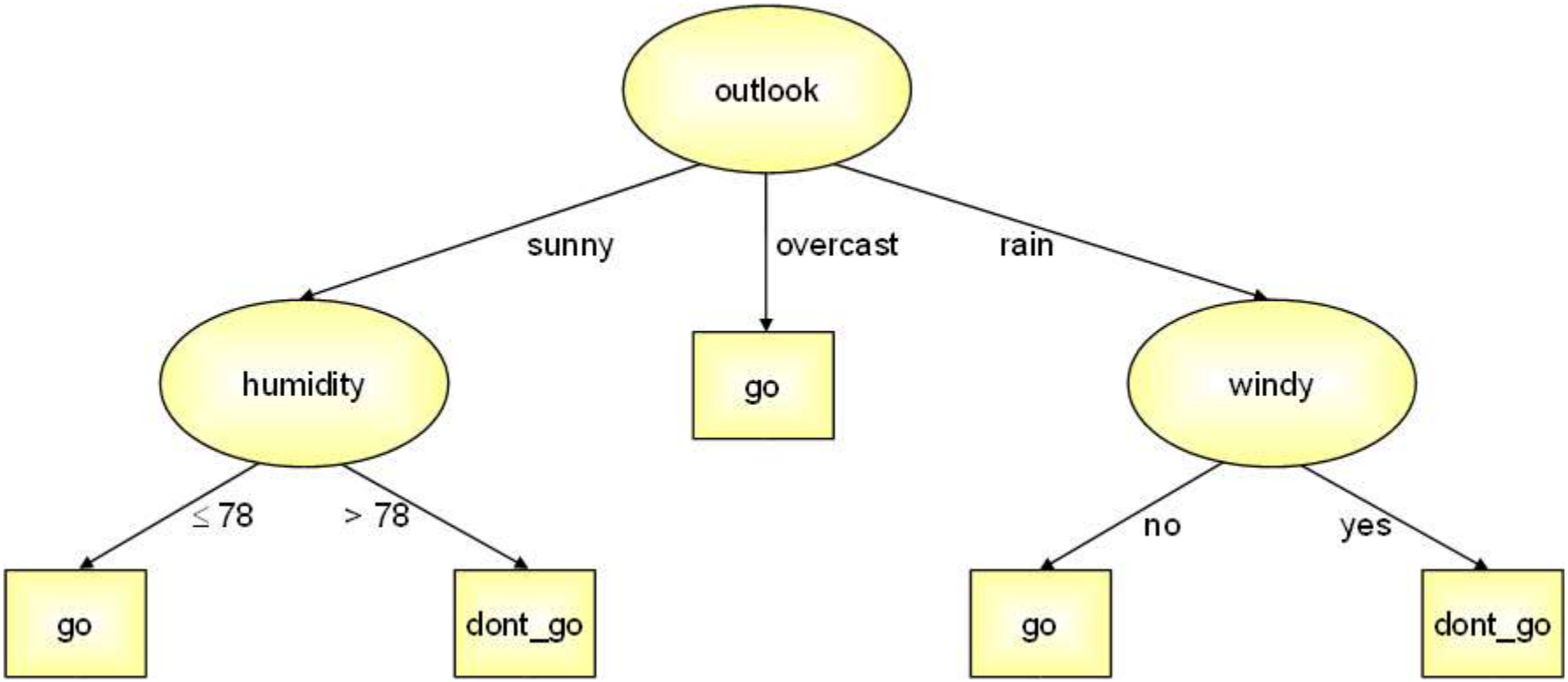

outlook
sunny
overcast
rain
humidity
humidity
windy
≤ 78
> 78
≤ 70
> 70
no
yes
go
dont_go
dont_go
go
go
dont_go
Cobre somente o exemplo T_8
Cobre os exemplos T_6, T_7, T_9 e T_{10}

Podando a AD

- `if outlook = rain and humidity > 70` cobre 4 casos
- `if outlook = rain and humidity <= 70` cobre apenas 1 caso

- Isso pode indicar um *overfitting* sobre os dados
- O indutor pode "podar" este ramo

AD “podada”


outlook
sunny
overcast
rain
humidity
go
windy
≤ 78
> 78
no
yes
go
dont_go
go
dont_go

Exemplo de Construção de Árvore de Decisão com Interpretação Geométrica

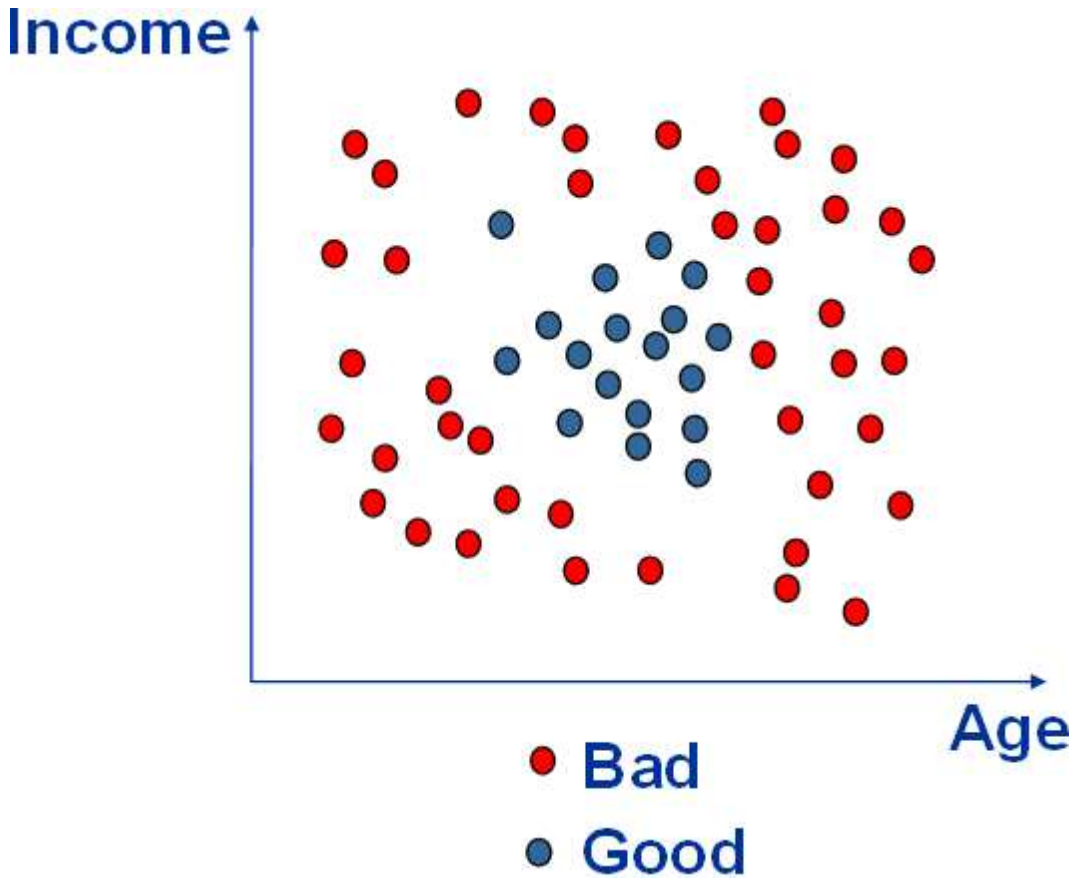
Interpretação Geométrica

- Consideramos os exemplos como um vetor de $\mathbf{m}$ atributos
- Cada vetor corresponde a um ponto em um espaço m-dimensional
- A AD corresponde a uma divisão do espaço em regiões (hiperplanos), cada região rotulada como uma classe

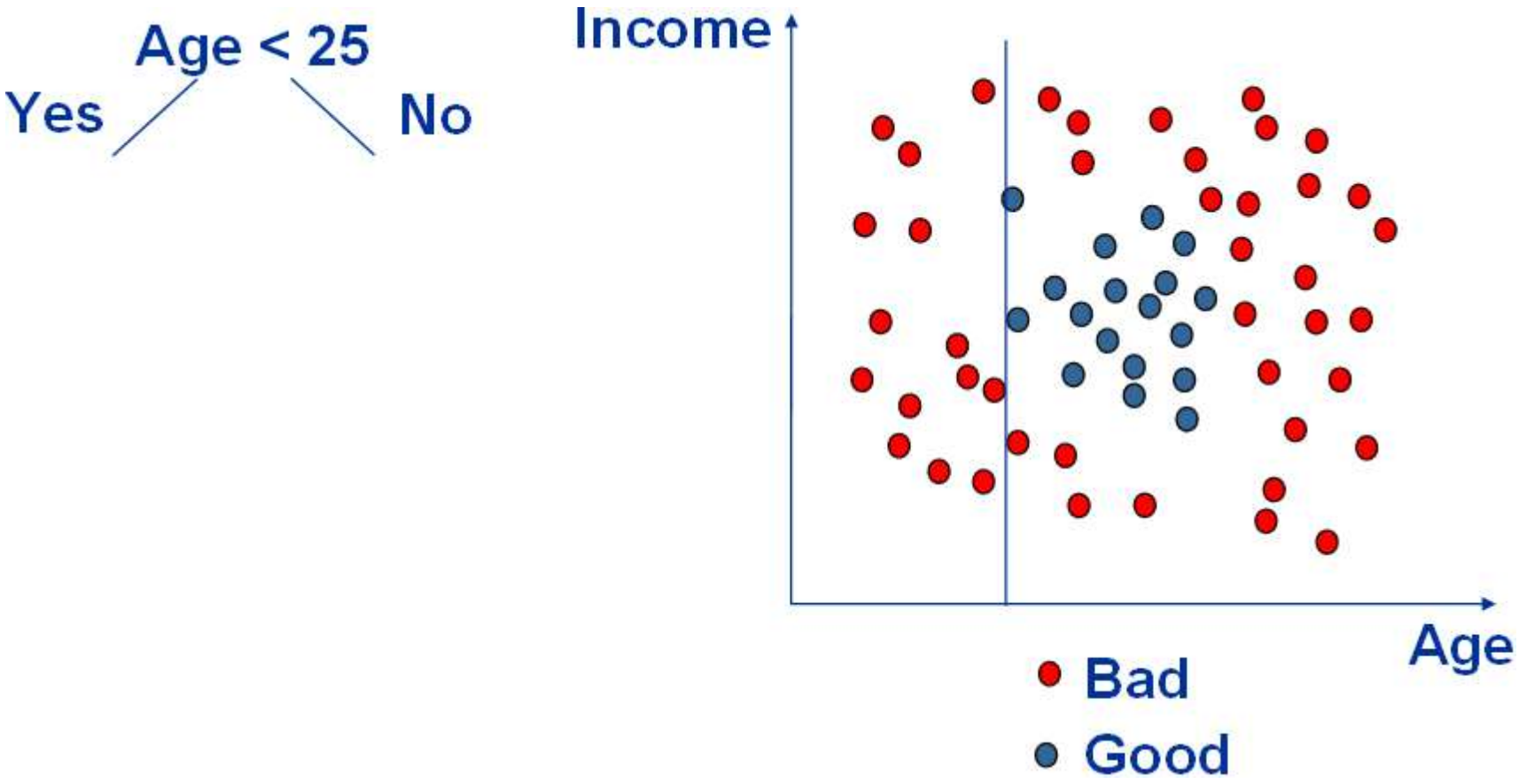
Exemplo – Árvore de Decisão

Age	Income	Class
20	2000	Bad
30	5100	Good
60	5000	Bad
40	6000	Good
...	...	...

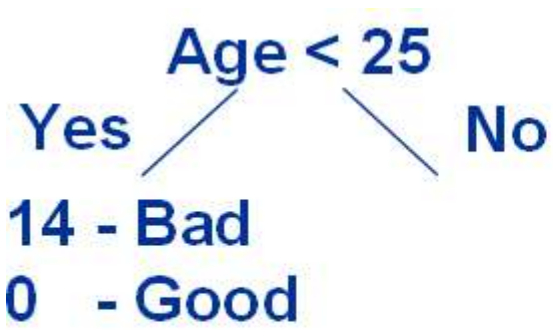
Exemplo – Árvore de Decisão

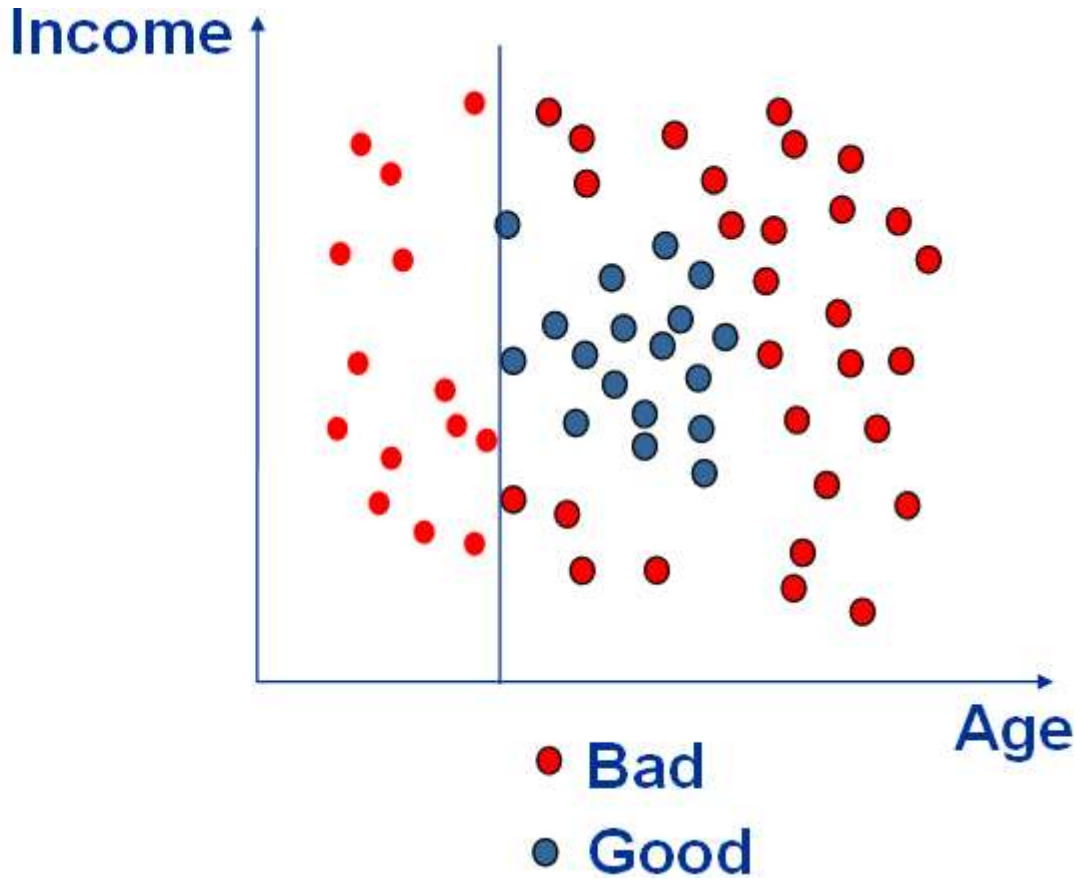


Exemplo – Árvore de Decisão

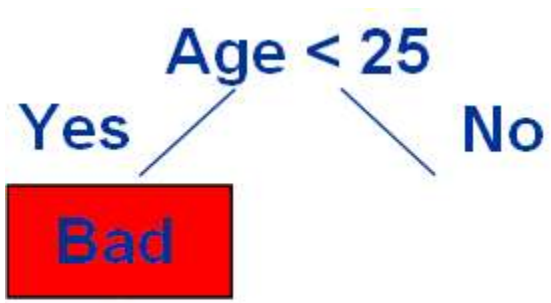


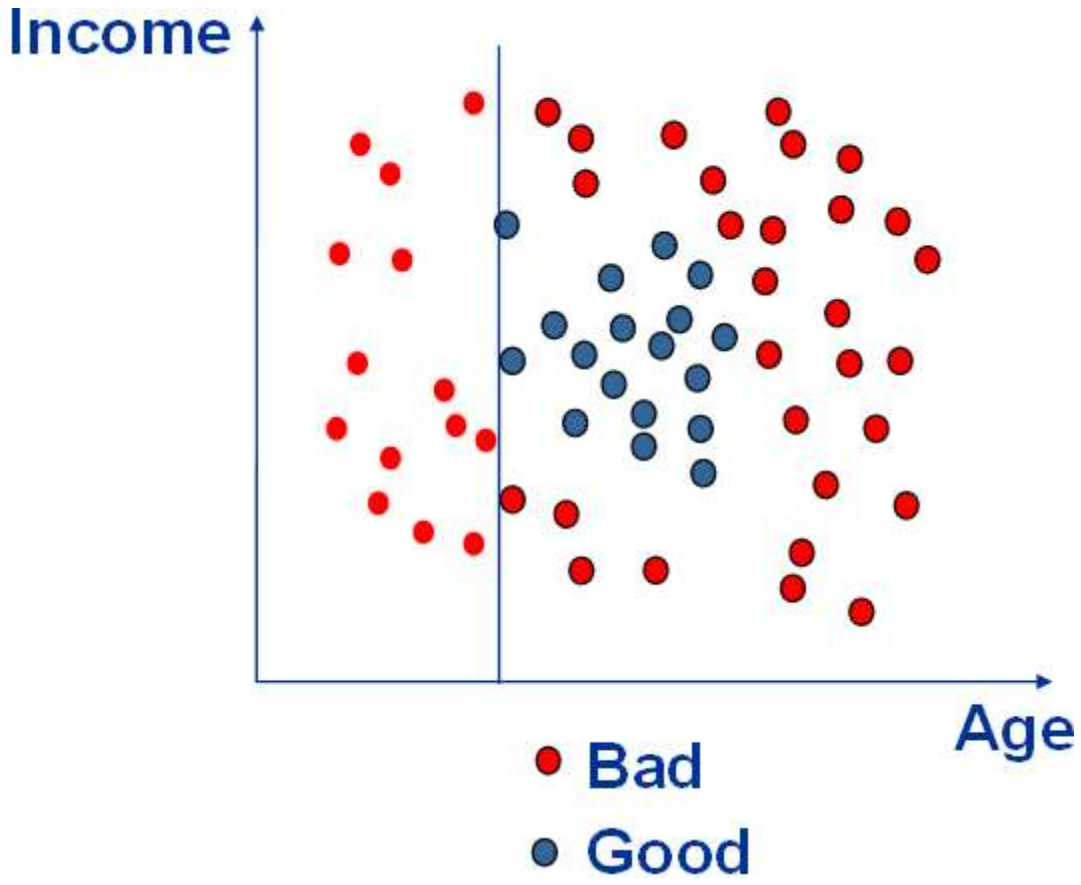
Exemplo – Árvore de Decisão



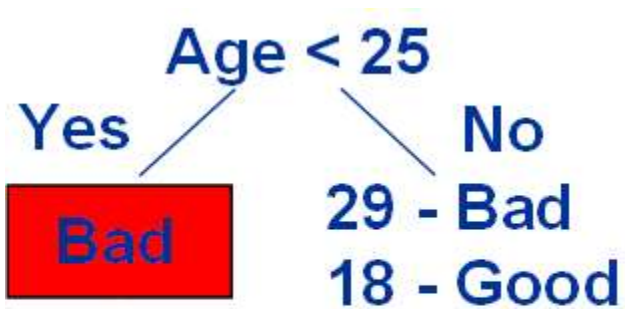


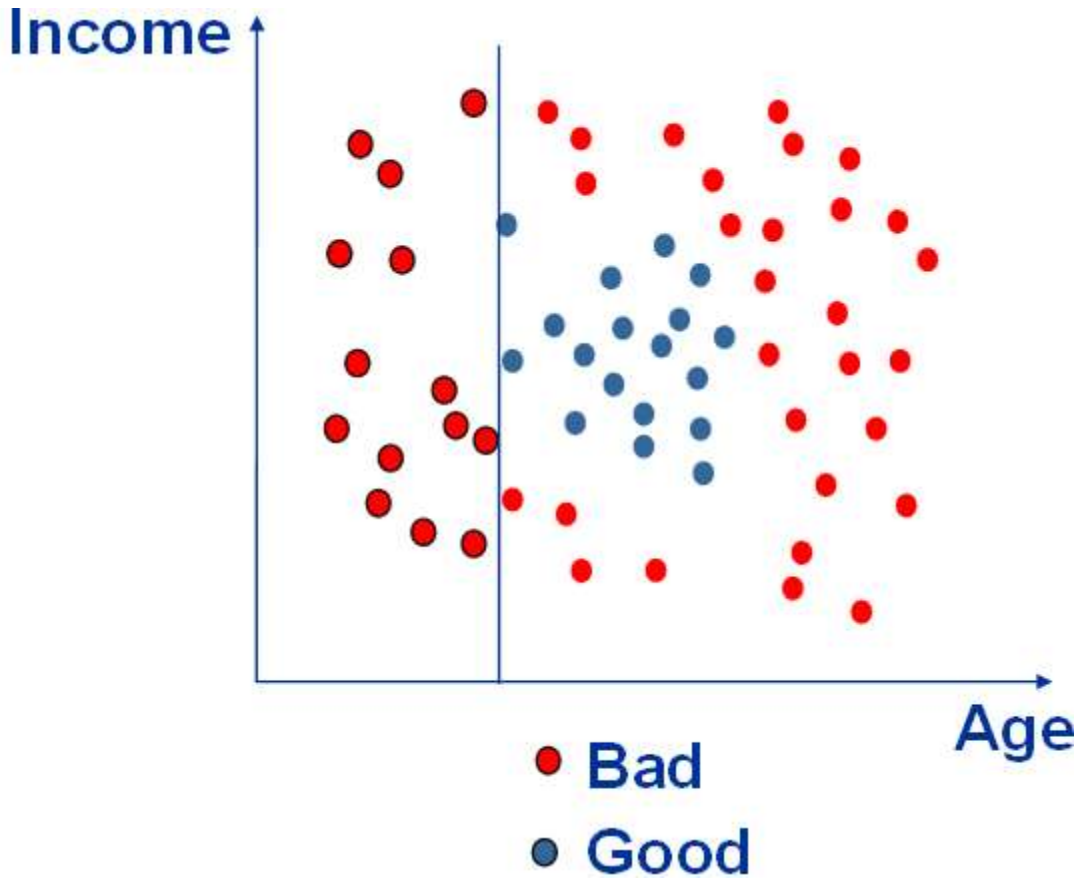
Exemplo – Árvore de Decisão



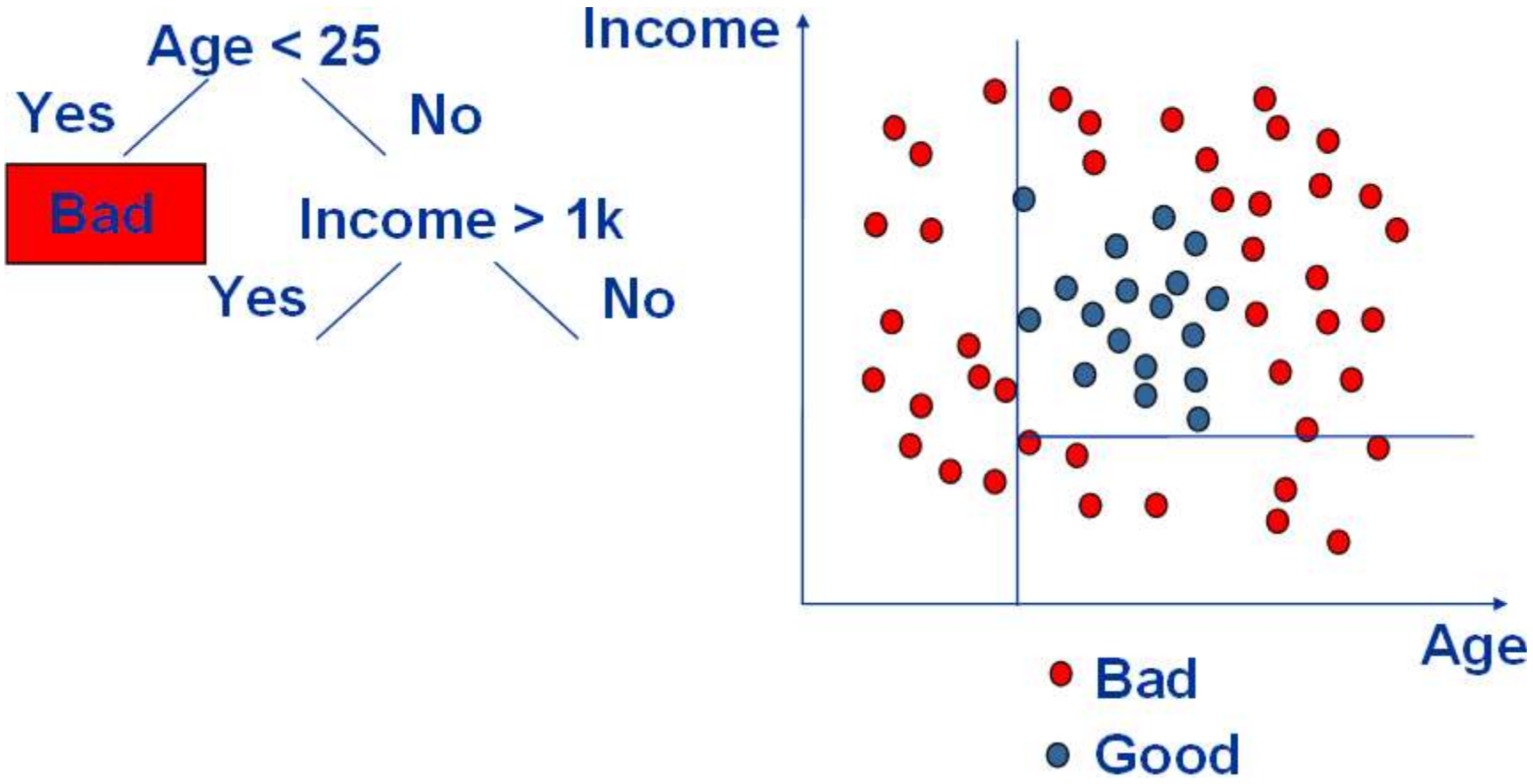


Exemplo – Árvore de Decisão

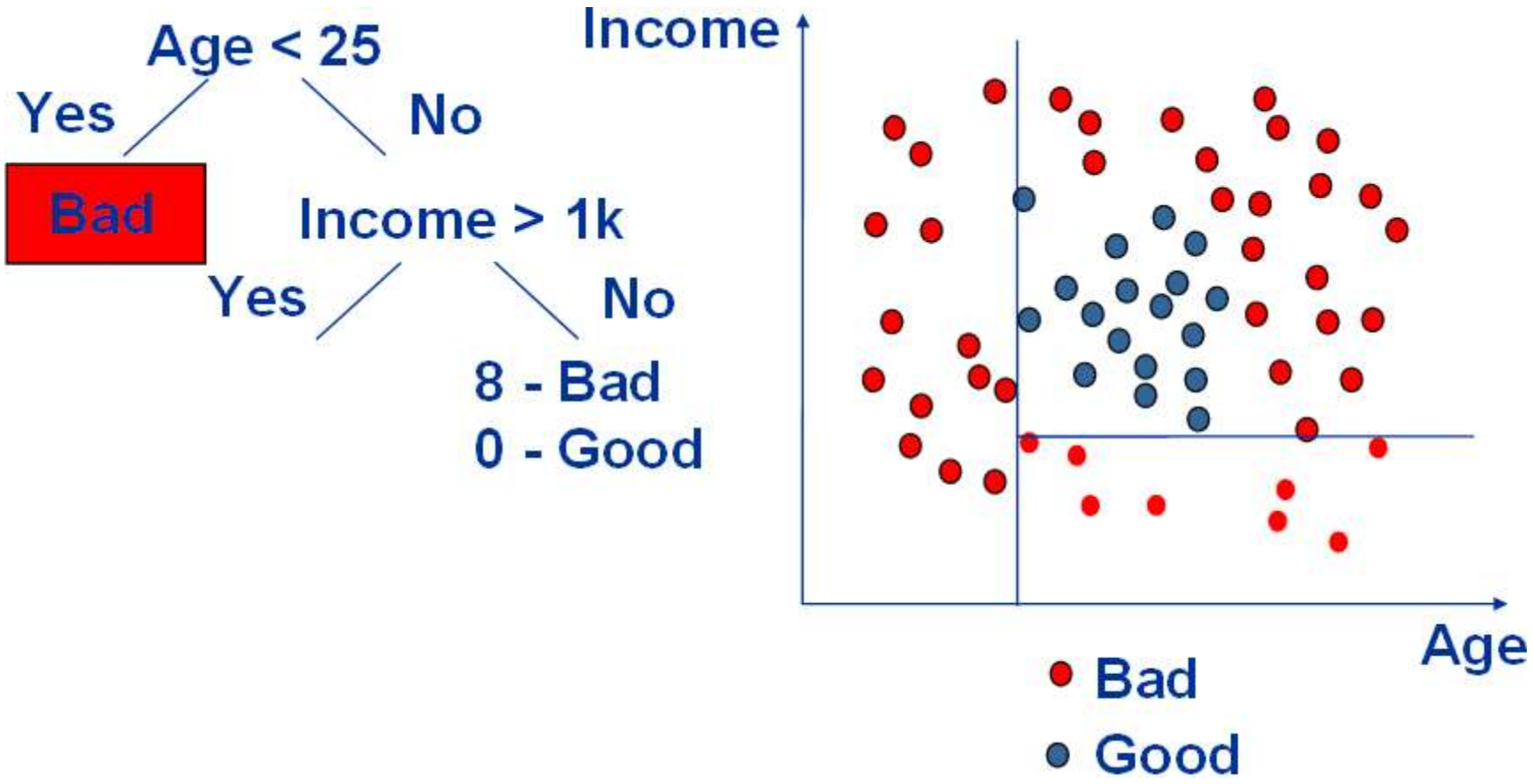




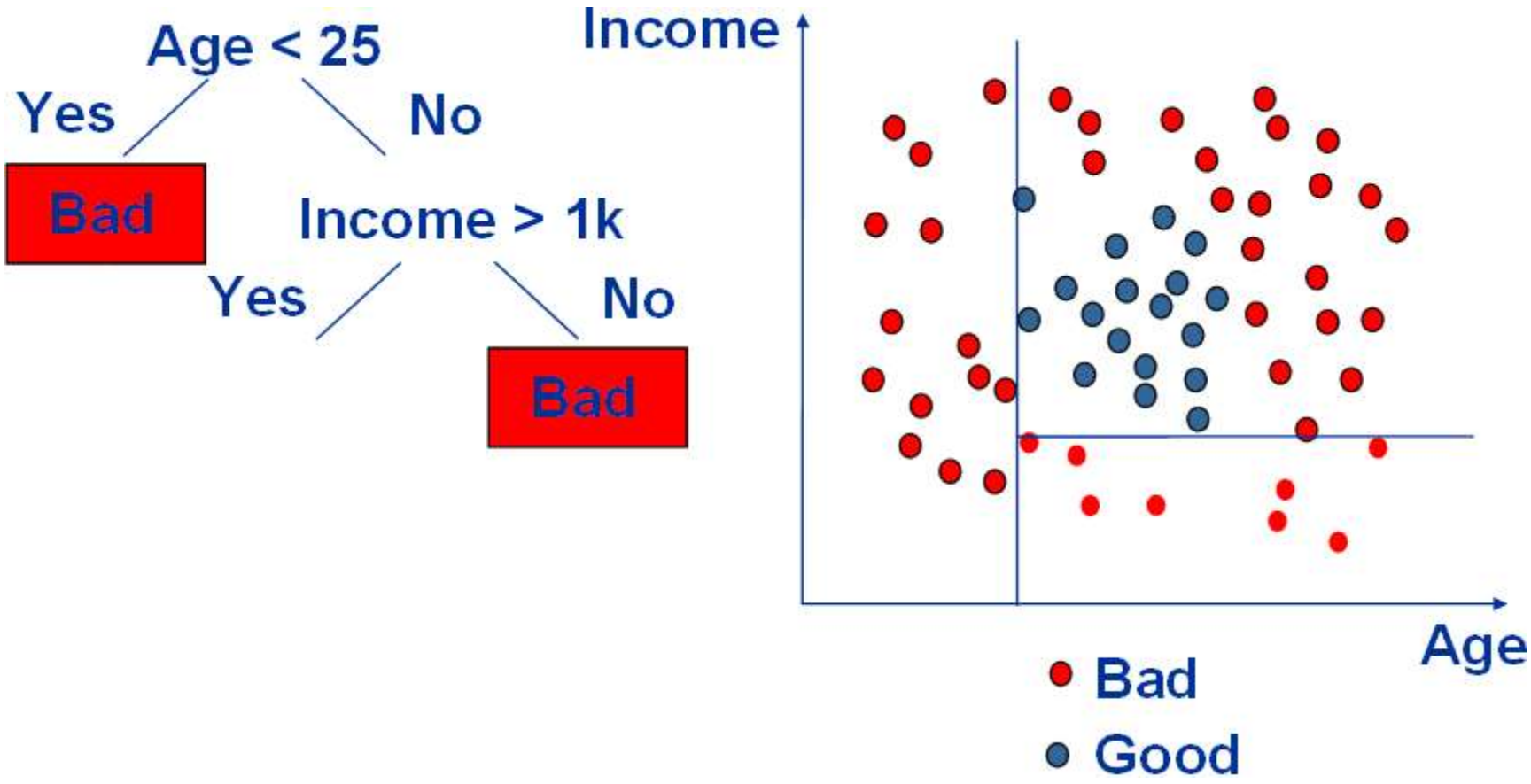
Exemplo – Árvore de Decisão



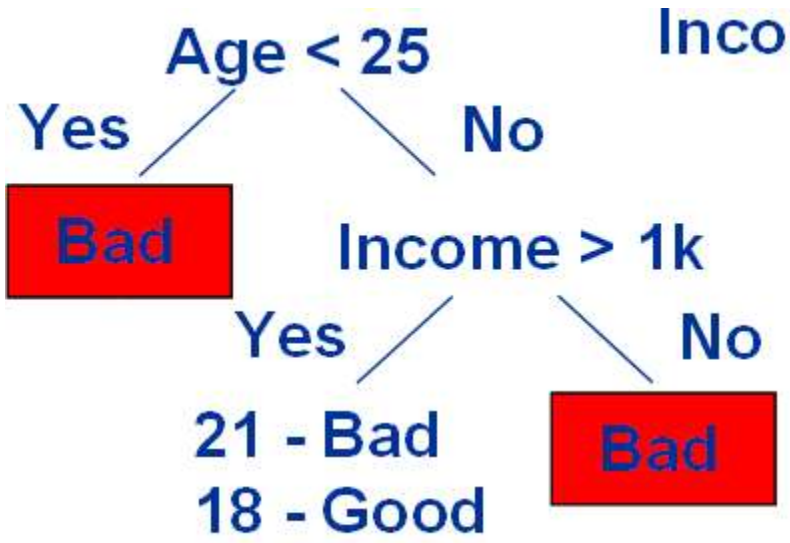
Exemplo – Árvore de Decisão

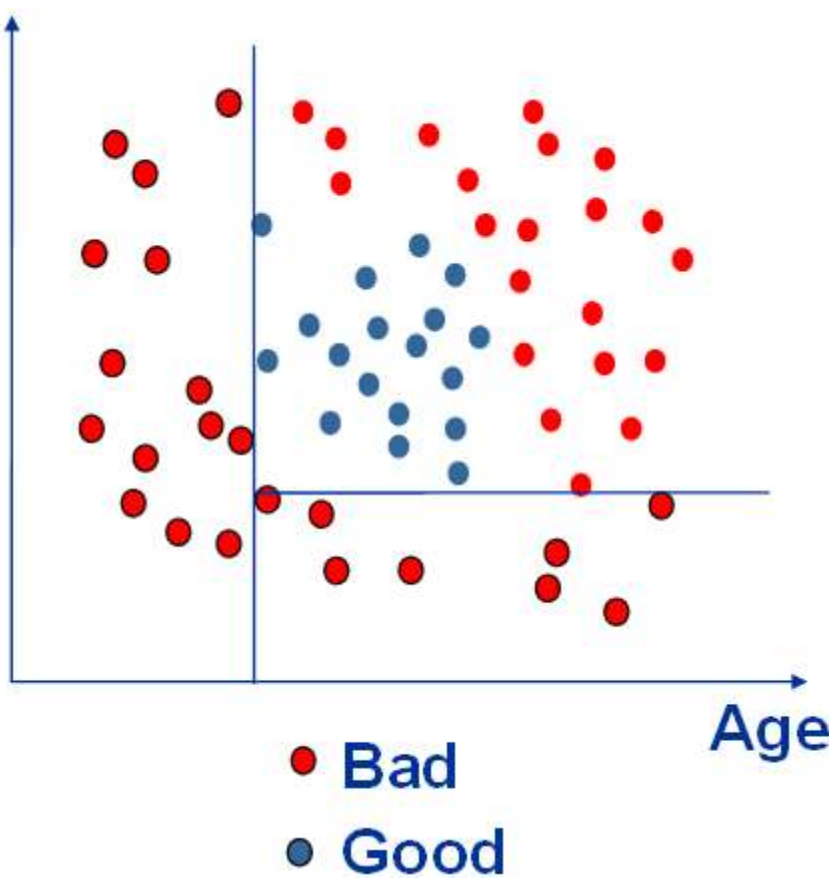


Exemplo – Árvore de Decisão

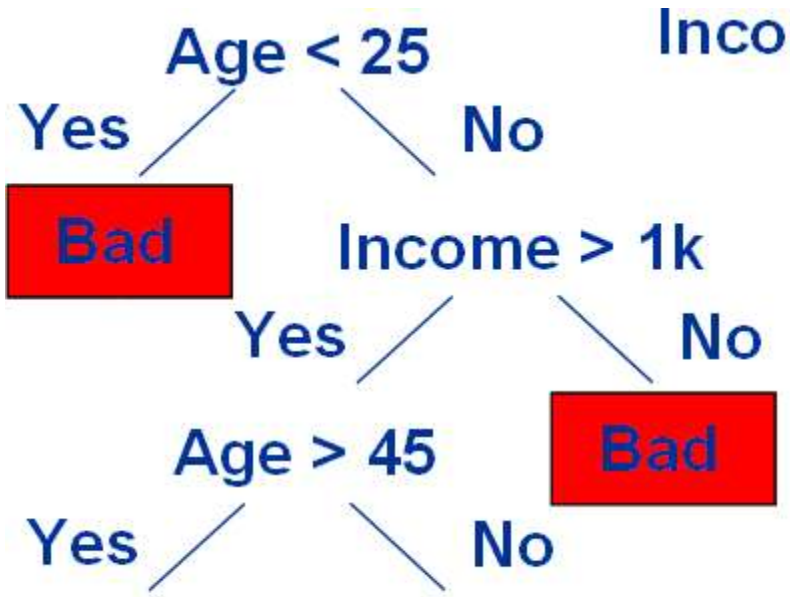


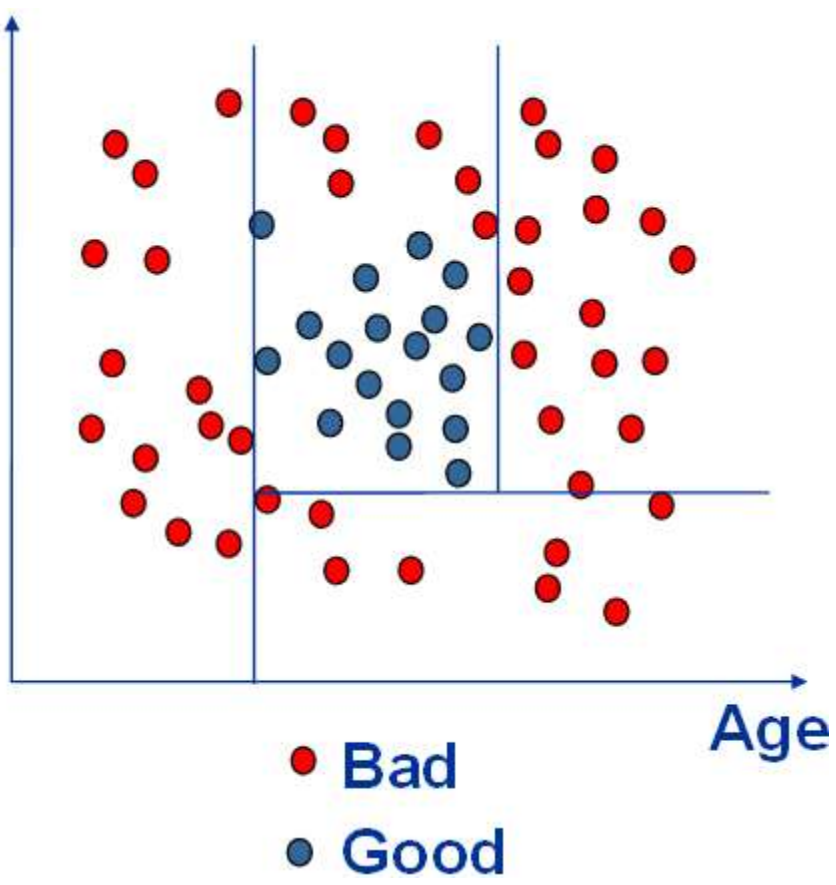
Exemplo – Árvore de Decisão



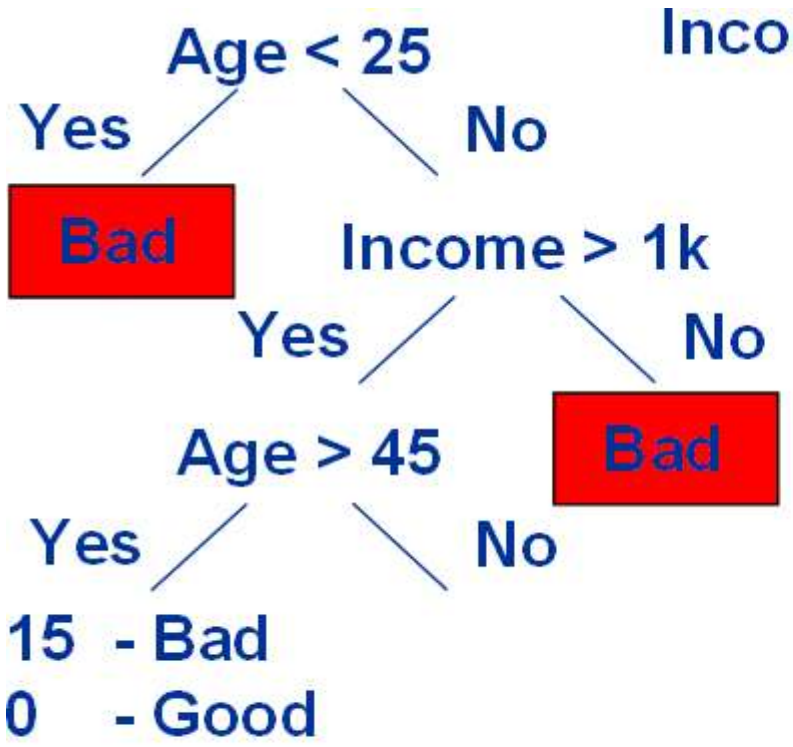


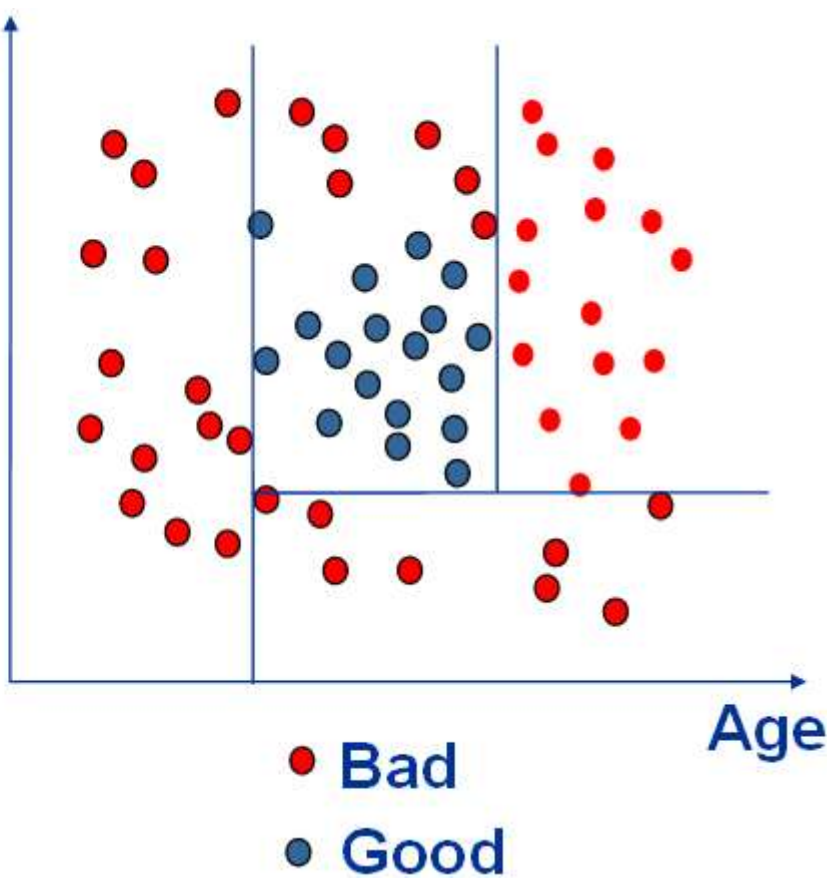
Exemplo – Árvore de Decisão



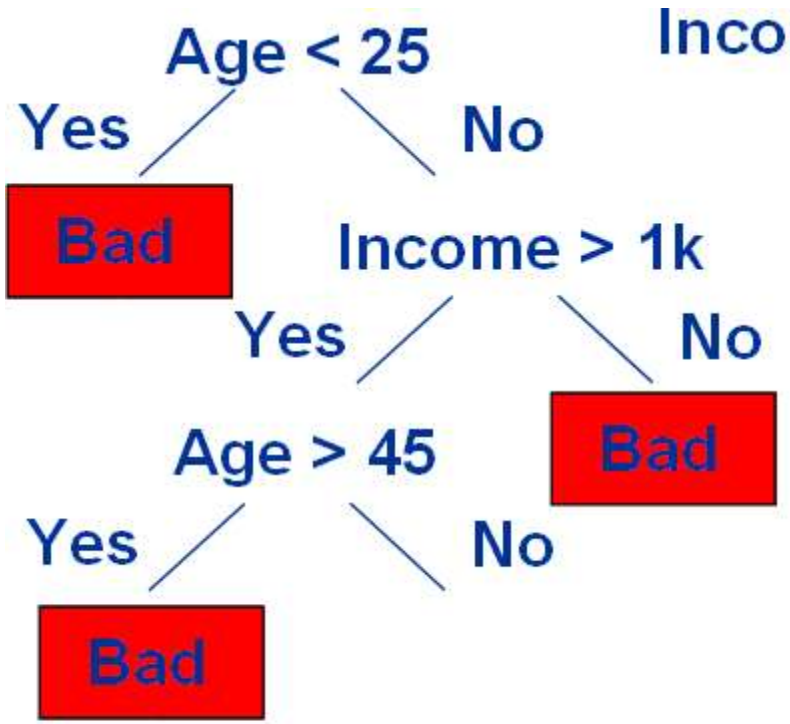


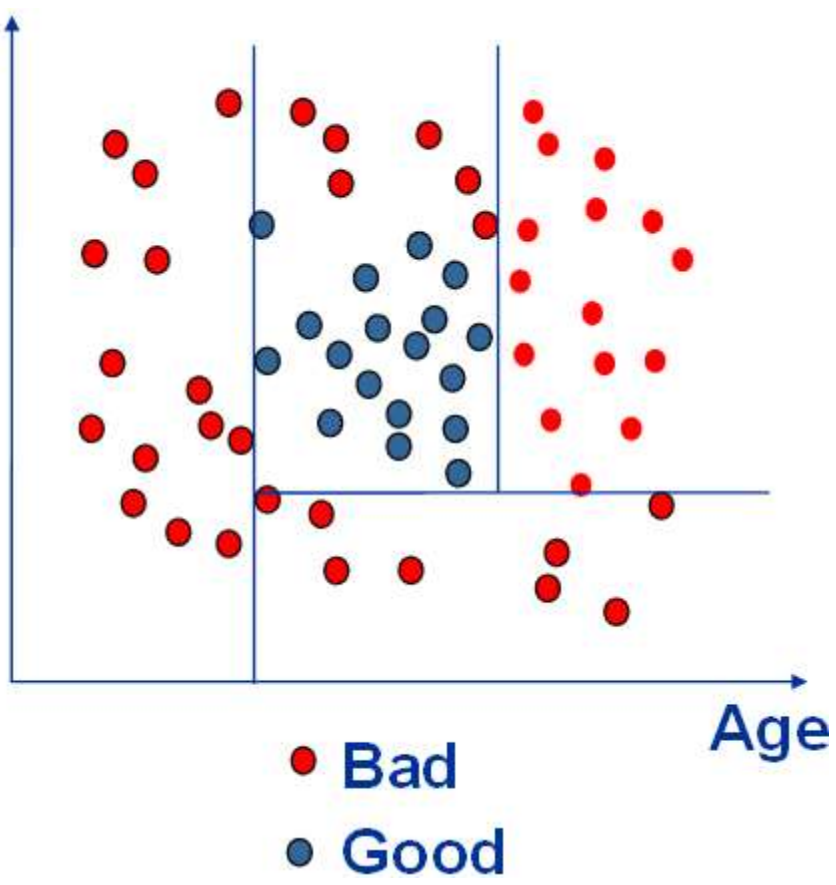
Exemplo – Árvore de Decisão



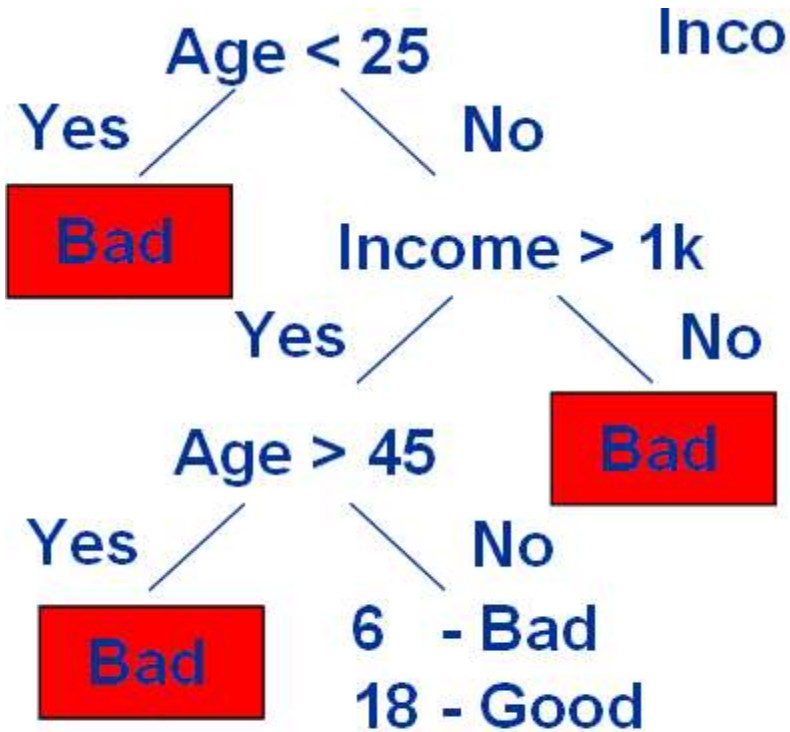


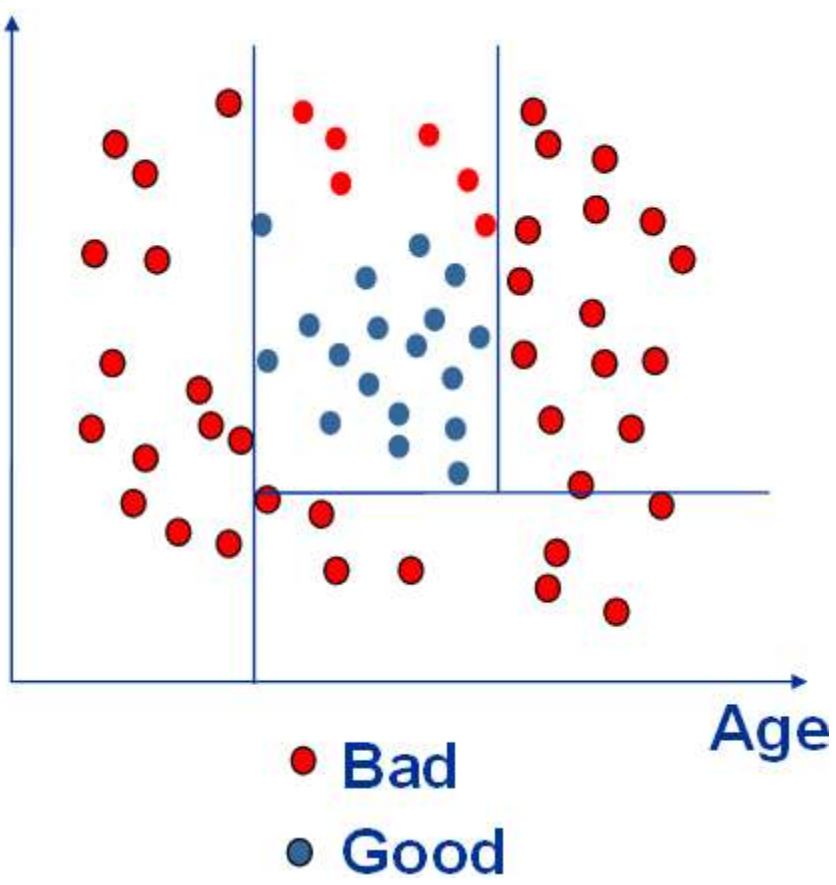
Exemplo – Árvore de Decisão



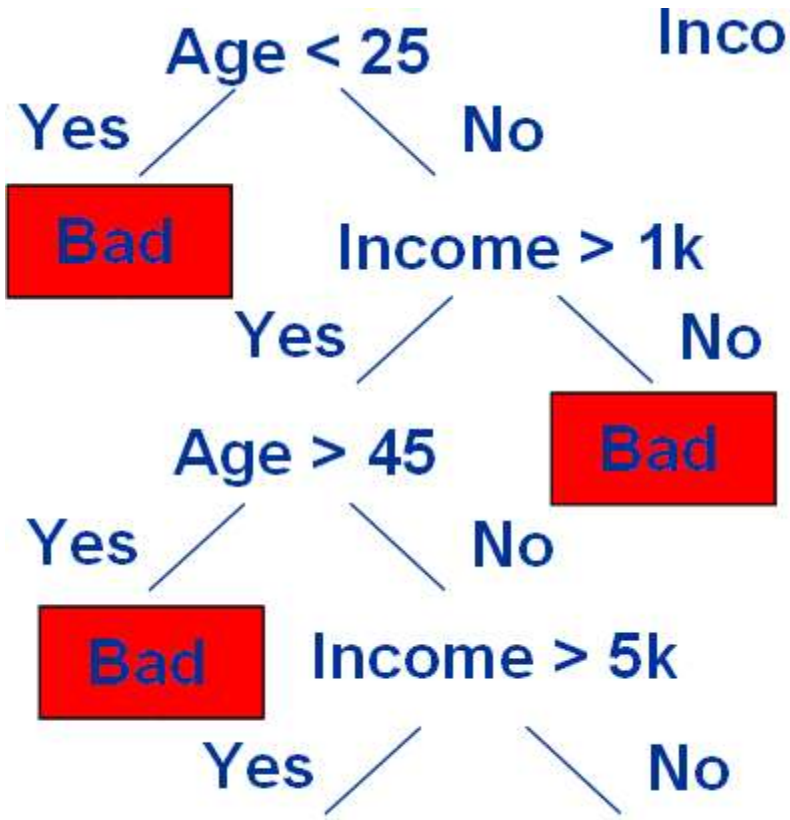


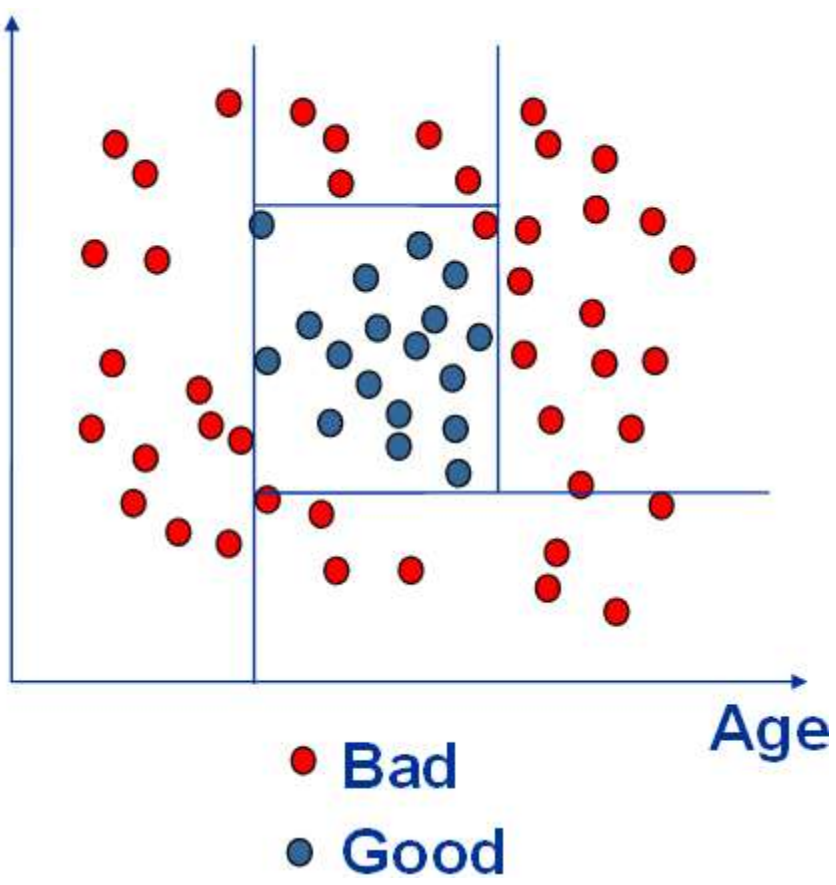
Exemplo – Árvore de Decisão



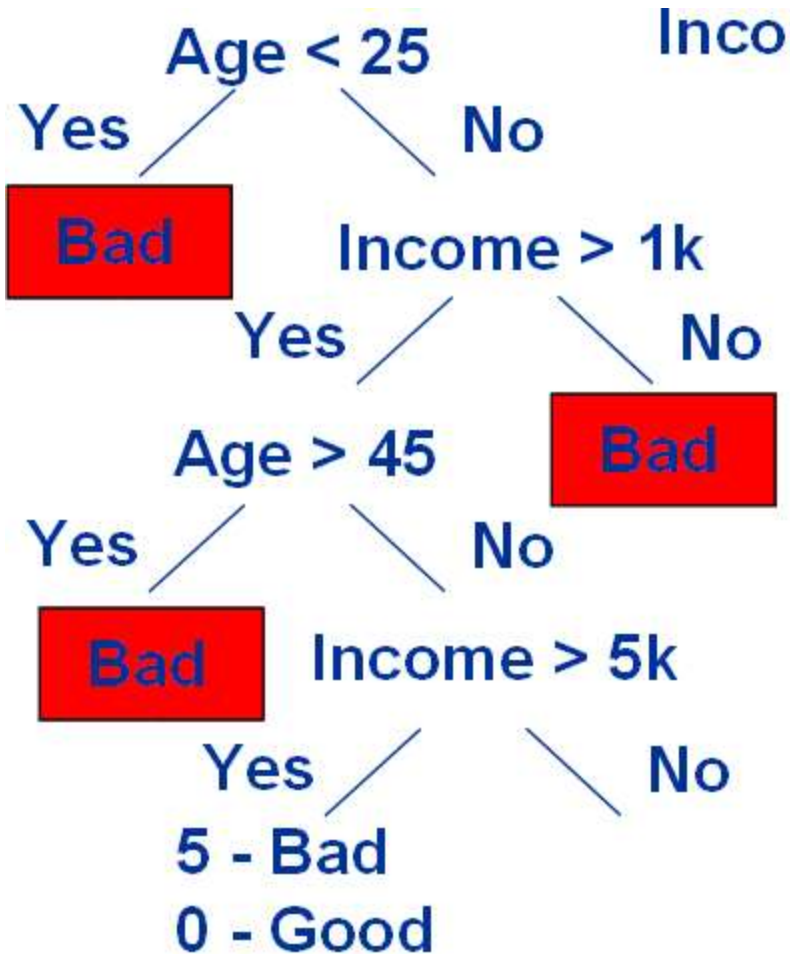


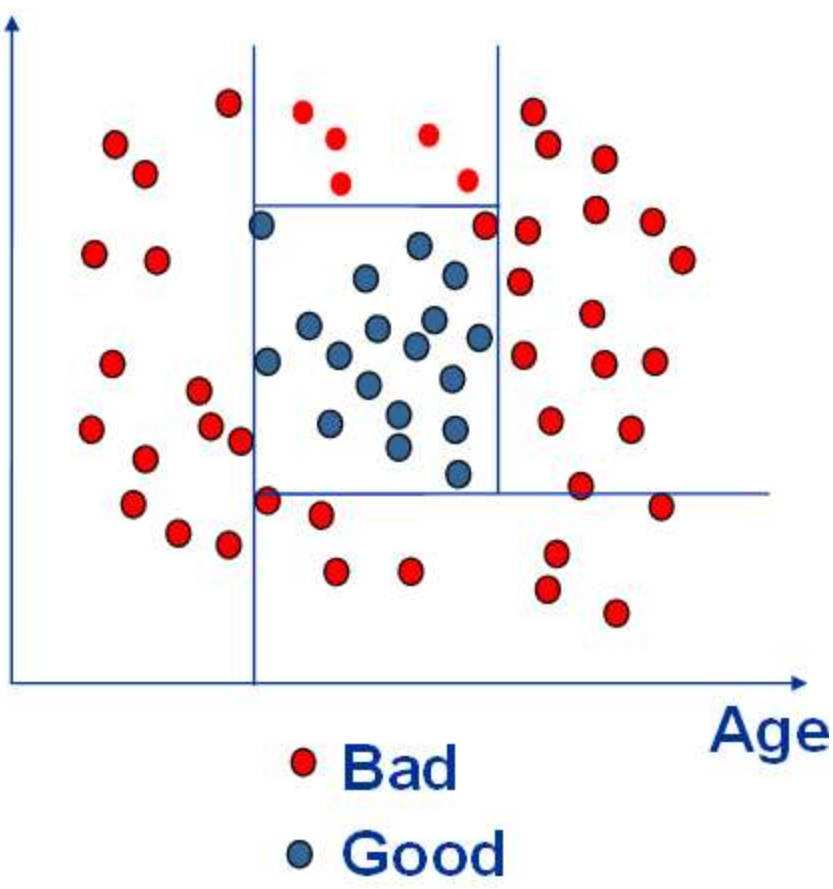
Exemplo – Árvore de Decisão



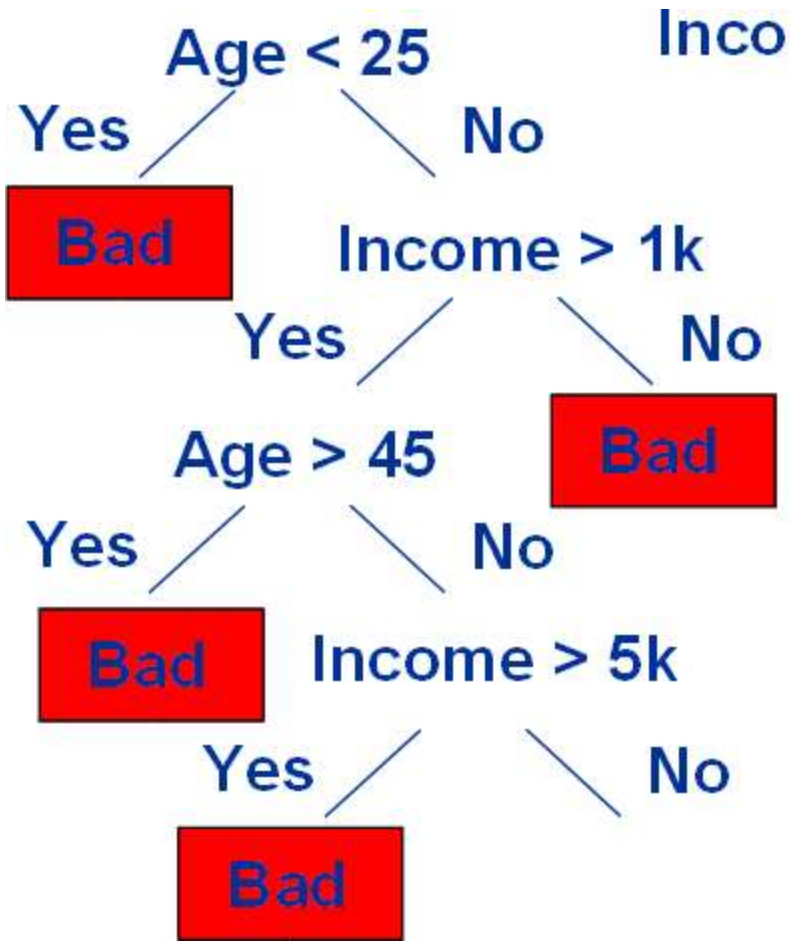


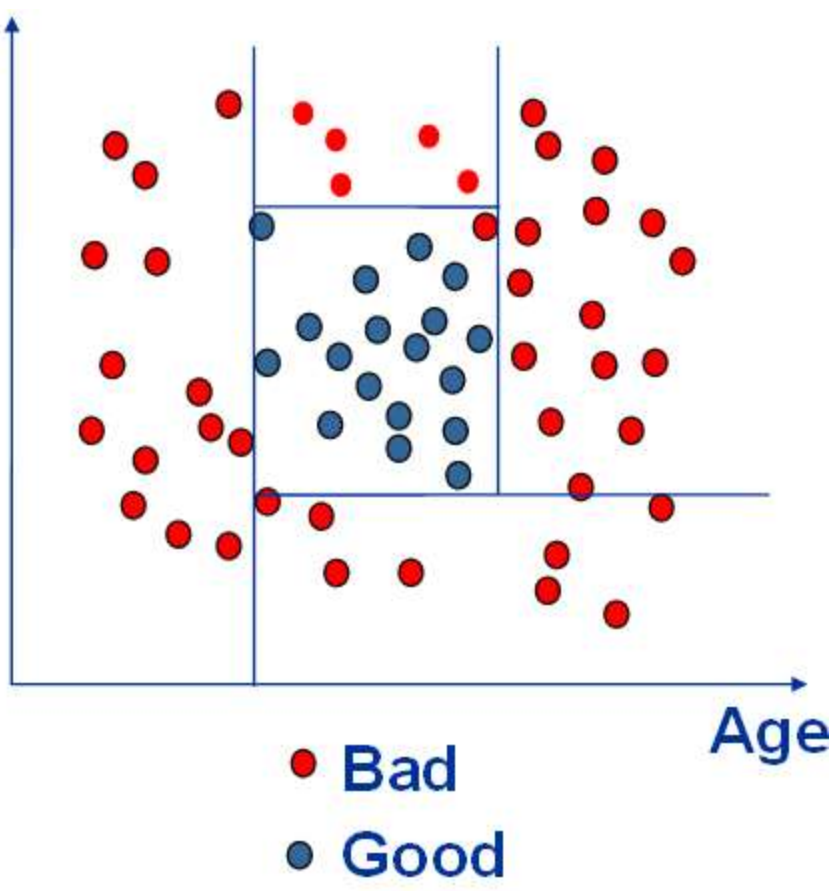
Exemplo – Árvore de Decisão



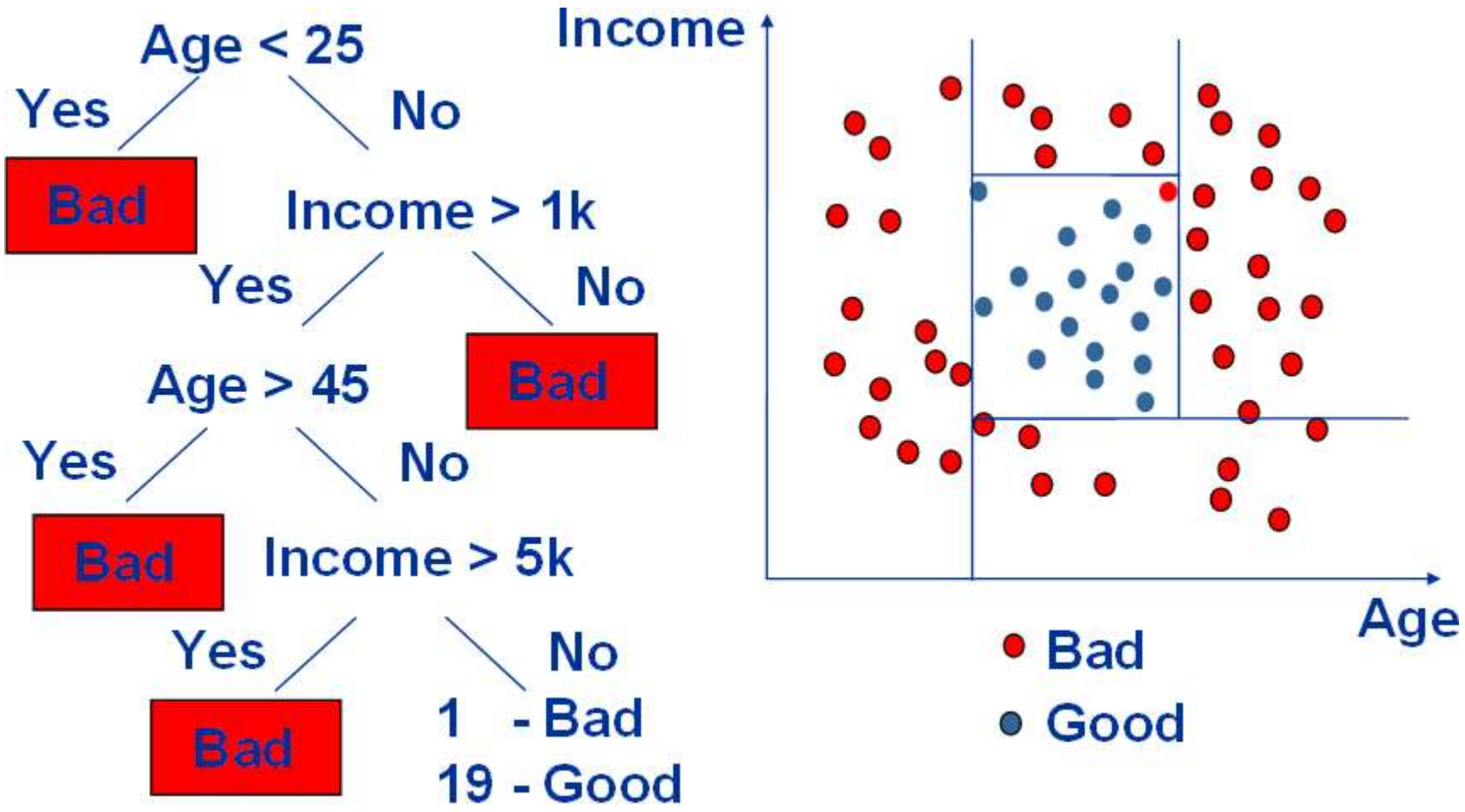


Exemplo – Árvore de Decisão

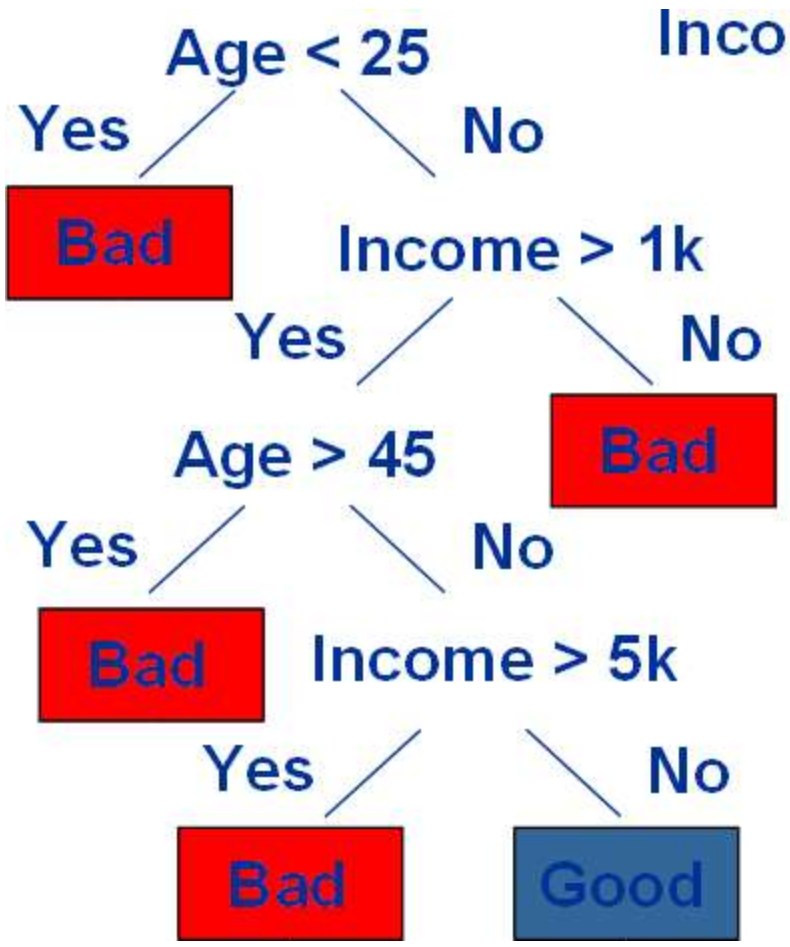


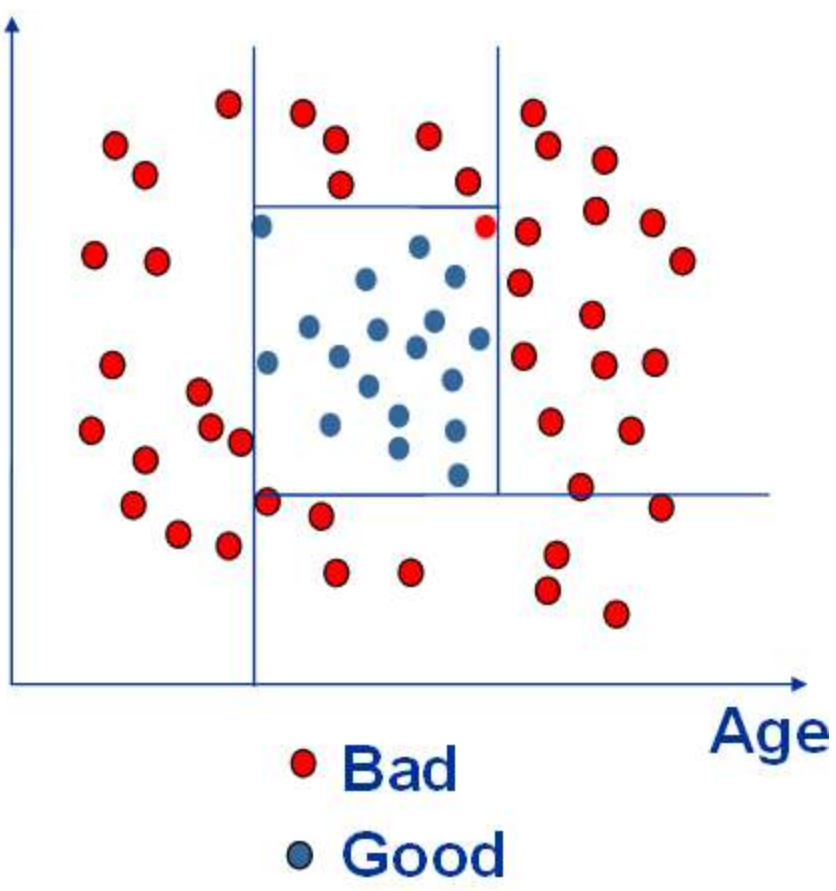


Exemplo – Árvore de Decisão



Exemplo – Árvore de Decisão





Vantagens ADs

- Flexibilidade e robustas
  - São invariantes a transformações monótonas de variáveis de entrada
  - Ex. usar x e $\log x$ produz mesma árvore
- Seleção de atributos embutida
  - Seleciona atributos mais relevantes em sua construção
  - Robustas a atributos irrelevantes e redundantes

Vantagens ADs

- Interpretabilidade
- Eficiência
  - Algoritmo guloso top-down, com estratégia dividir-para-conquistar
  - Complexidade linear no número de exemplos

Desvantagens ADs

- Atributos contínuos
  - Operação de ordenação consome muito tempo
  - Alguns autores recomendam discretização prévia
- Instabilidade
  - Pequenas variações no conjunto de treinamento podem produzir grandes variações na árvore final

Referências

- Slides de:
  - Profa. Ana Carolina Lorena, UNIFESP
  - Prof. André C. P. L. F. Carvalho, ICMC-USP
  - Prof. Marcilio C. P. Souto, UFPE
  - Prof. José Augusto Baranaukas, FFCLRP-USP
  - Profa. Huei Diana Lee

  - Livro Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina - Katti Faceli, Ana Carolina Lorena, João Gama, André C.P.L.F. de Carvalho, Editora LTC, 2011 (Capítulo 6)