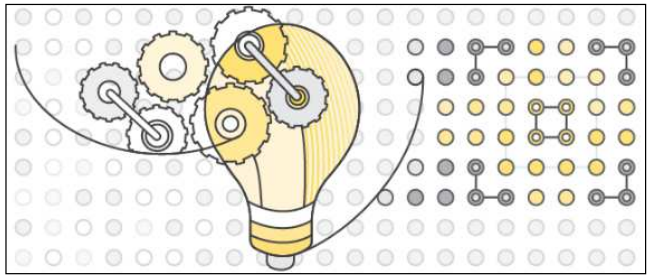
Aprendizado de Máquina: Introdução



Stanley R. M. Oliveira

Resumo da Aula

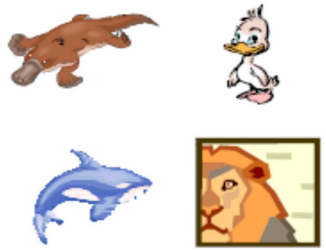
- **Aprendizado de Máquina**:
  - Motivação.
  - Hierarquiza do aprendizado.
  - Paradigmas de aprendizado.
  - Conceitos e definições.
- **Classificação de Dados**:
  - O processo de classificação.
  - Características de um bom classificador.
  - Principais métodos de classificação.
  - Principais algoritmos existentes.
- **Árvores de Decisão**:
  - Conceitos básicos.
  - Algoritmos mais conhecidos.
  - Mecanismos de poda.
  - Escolha do atributo "**split**".

O que é Aprendizado de Máquina ?

- **Machine Learning** ou **Aprendizado de Máquina** é um método de **análise de dados** que automatiza o desenvolvimento de **modelos analíticos**. Utiliza **algoritmos** que aprendem **interativamente** a partir de dados.
- O **objetivo** do **aprendizado de máquina** é programar computadores para **aprender** um determinado **comportamento** ou **padrão** automaticamente a partir de **exemplos** ou **observações**.
- **Não** existe um **único algoritmo** que apresente melhor desempenho para todos problemas.

Motivação – Exemplo 1

- Dado um **conjunto de objetos**, colocar os **objetos** em **grupos** baseados na **similaridade** entre eles.

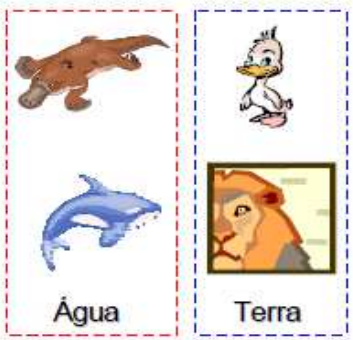


Motivação – Exemplo 1 ...

- Dado um **conjunto de objetos**, colocar os **objetos** em **grupos** baseados na **similaridade** entre eles.

Motivação – Exemplo 1 ...

- Dado um **conjunto de objetos**, colocar os **objetos** em **grupos** baseados na **similaridade** entre eles.



Motivação – Exemplo 1 ...

- Dado um **conjunto de objetos**, colocar os **objetos** em **grupos** baseados na **similaridade** entre eles.

Motivação – Exemplo 2

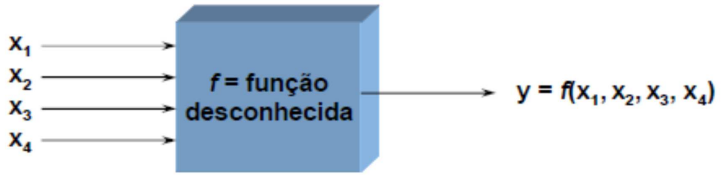
- Dados os pares (x, f(x)), inferir f(x).

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

Dada uma **amostra finita**, é frequentemente **impossível** determinar a verdadeira função f(x).

Abordagem: encontre uma **hipótese** (**modelo**) nos exemplos de treinamento e assuma que a **hipótese** se repita para exemplos futuros também.

Motivação – Exemplo 2 ...

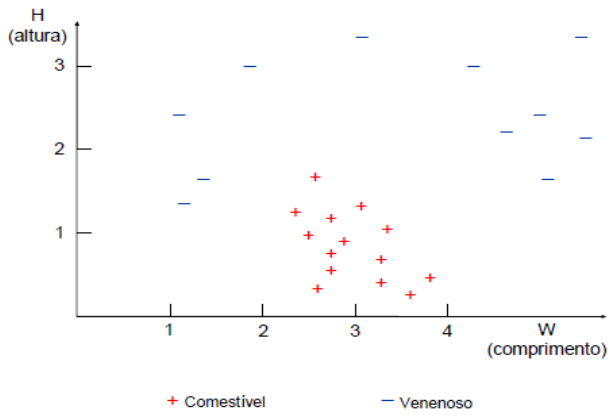


Exemplo	X_1	X_2	X_3	X_4	Y
z_1	0	1	1	0	0
z_2	0	0	0	0	0
z_3	0	0	1	1	1
z_4	1	0	0	1	1
z_5	0	1	1	0	0
z_6	1	1	0	0	0
z_7	0	1	0	1	0

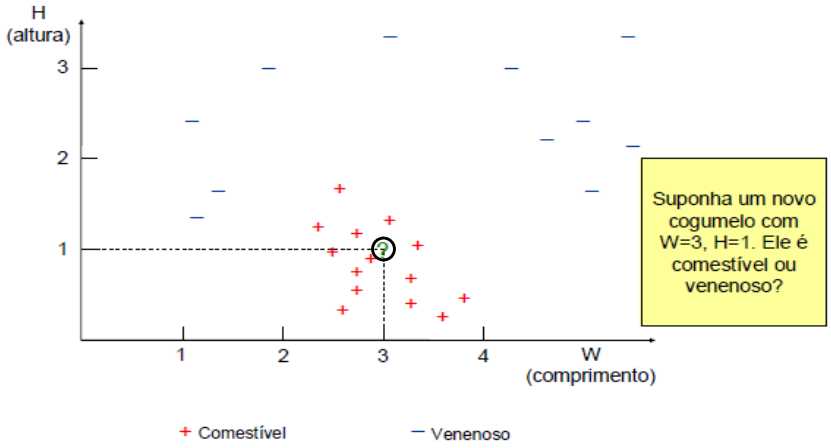
Cogumelos Comestíveis x Venenosos

- Um pesquisador foi a campo e coletou diversos **cogumelos**.
- Ao chegar em seu laboratório, ele mediu o **comprimento** e **altura** de cada **cogumelo**.
- Ele também classificou cada **cogumelo** coletado como **comestível** ou **venenoso**.

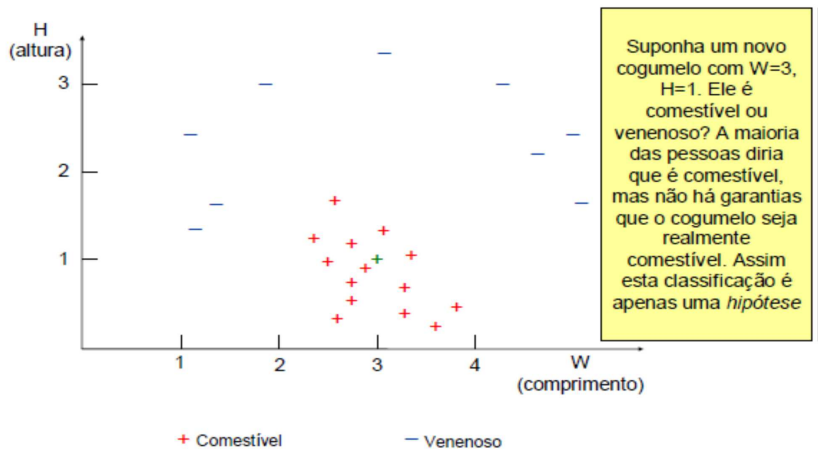
Cogumelos Comestíveis x Venenosos



Cogumelos Comestíveis x Venenosos

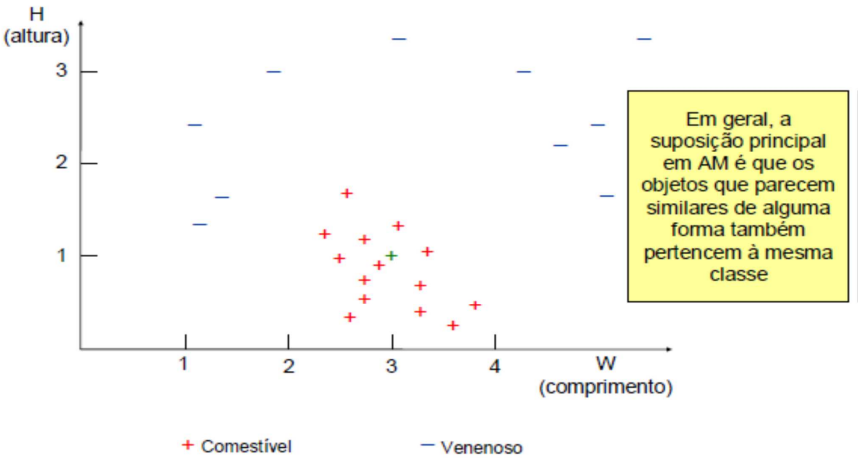


Cogumelos Comestíveis x Venenosos



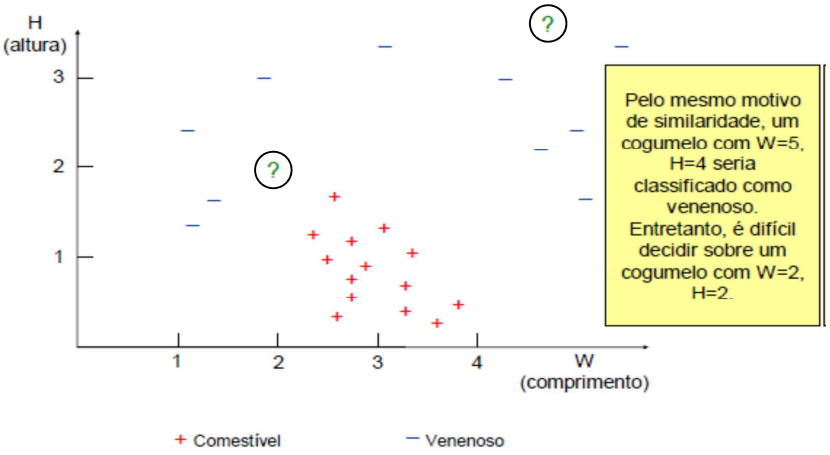
Suponha um novo cogumelo com W=3, H=1. Ele é comestível ou venenoso? A maioria das pessoas diria que é comestível, mas não há garantias que o cogumelo seja realmente comestível. Assim esta classificação é apenas uma *hipótese*

Cogumelos Comestíveis x Venenosos



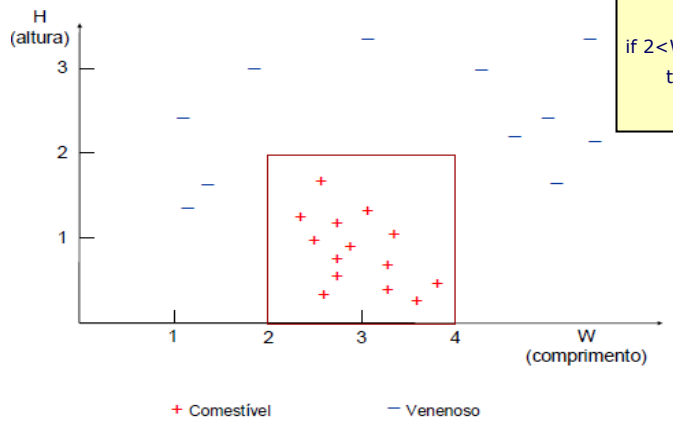
Em geral, a suposição principal em AM é que os objetos que parecem similares de alguma forma também pertencem à mesma classe

Cogumelos Comestíveis x Venenosos



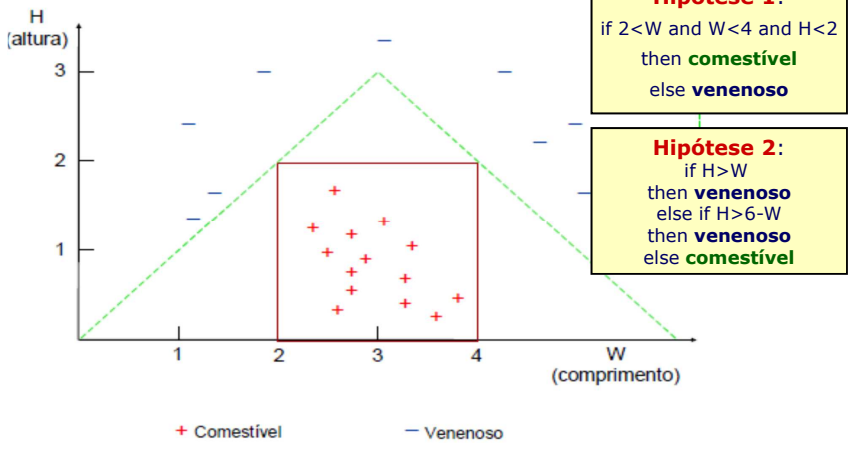
Pelo mesmo motivo de similaridade, um cogumelo com W=5, H=4 seria classificado como venenoso. Entretanto, é difícil decidir sobre um cogumelo com W=2, H=2.

Cogumelos Comestíveis x Venenosos



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

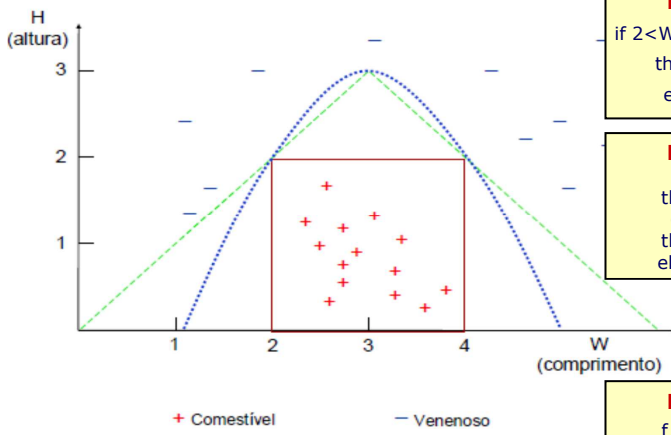
Cogumelos Comestíveis x Venenosos



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**

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:
f **H< 3-(W-3)^2**
then **comestível**
else **venenoso**

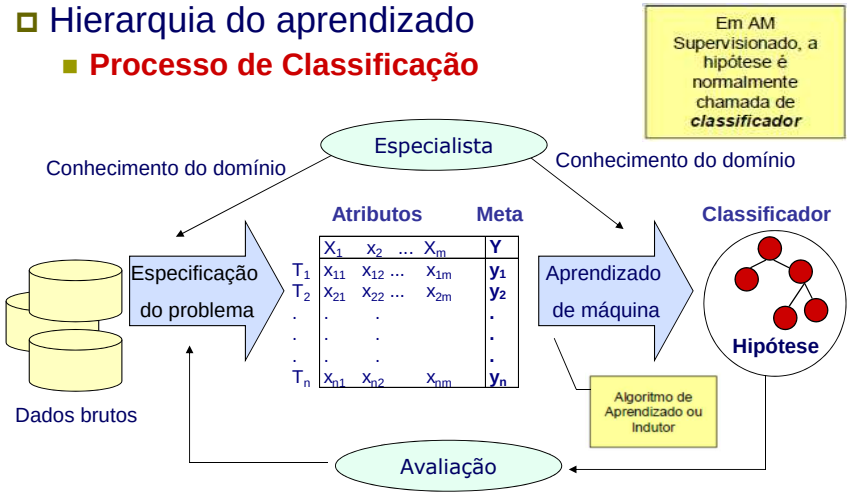
Indução x Dedução

- **Indução:** é a forma de inferência lógica que permite obter conclusões a partir de um conjunto de exemplos.
- Na indução, um conceito é aprendido efetuando-se **inferência indutiva** sobre os **exemplos apresentados** *(cautela na escolha de exemplos)*.
- **Dedução:** Humanos usam raciocínio dedutivo para deduzir nova informação a partir de informação relacionada logicamente.

Hierarquia do Aprendizado

Aprendizado de Máquina

- Hierarquia do aprendizado
  - **Processo de Classificação**



Aprendizado de Máquina ...

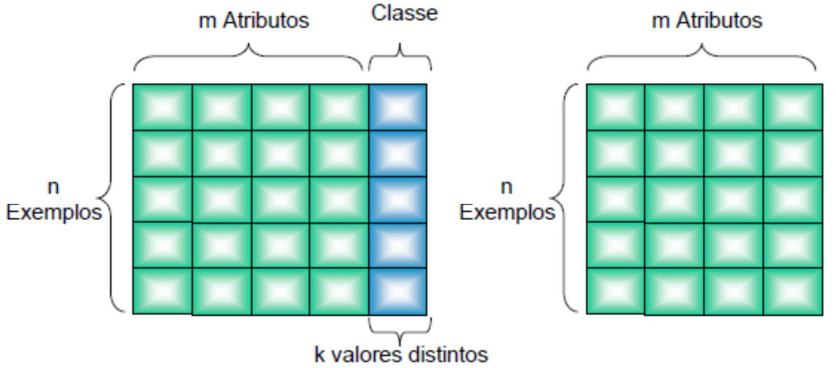
- Paradigmas do aprendizado:
  - **Simbólico:** Buscam aprender construindo representações simbólicas (expressão lógica, **árvores de decisão regras**).
  - **Estatístico:** Buscam métodos estatísticos (**Aprendizado bayesiano**)
  - **Baseado em Exemplos:** Sistemas *lazy* (RBC, ***Nearest Neighboors***).
  - **Conexionista:** Modelos inspirados no modelo biológico do sistema nervoso (**Redes Neurais**).
  - **Evolutivo:** Teoria de Darwin (**Algoritmos Genéticos**).

Categorias de sistemas de aprendizado

- **Não Simbólico ou Caixa-preta**:
  - Não facilmente interpretado por humanos;
  - Desenvolve sua própria representação de conceitos;
  - Não fornece esclarecimento ou explicação sobre o processo de classificação.
- **Simbólico ou Orientado a conhecimento**:
  - Cria estruturas simbólicas que podem ser compreendidas por seres humanos.
  - Os resultados da indução devem ser compreensíveis como simples 'pedaços' de informação, diretamente interpretáveis em linguagem natural.

AM Supervisionado x Não Supervisionado

- No Aprendizado Supervisionado, cada exemplo é rotulado segundo sua classe
- No Aprendizado Não Supervisionado, cada exemplo não possui classe associada



Conjunto de Treinamento x Teste

- Em geral, um **conjunto de exemplos** é **dividido** em dois subconjuntos disjuntos:
  - **Conjunto de treinamento**: é usado para o aprendizado do conceito; e
  - **conjunto de teste**: e usado para medir o grau de efetividade do conceito aprendido.
- Os **subconjuntos** são **disjuntos** para assegurar que as medidas obtidas sejam **estatisticamente válidas**.

O problema de classes desbalanceadas

Prevalência de Classe

Problema com **desbalanceamento** de classes em conjunto de exemplos.

Exemplo: distr (C1, C2) = (99,75%, 0,25%)

Neste exemplo, Classe **Majoritária** (ou **Prevalente**) é C1
Classe **Minoritária** é C2

Classificador que classifique novos exemplos como C1 teria uma precisão de 99,75%.

Se a **Classe C2** fosse, por exemplo, **ocorrência de Geada ...**

Aprendizado de Máquina: Definições

- Algumas Definições em AM
  - **Bias:** qualquer preferência de uma hipótese sobre a outra.
  - **Modo de aprendizado:**
    - todo conjunto de treinamento presente no aprendizado (**não incremental**).
    - quando novos exemplos de treinamento são adicionados (**incremental**).

Aprendizado de Máquina: Definições

Erro (err(h))

Medida de desempenho de um Classificador.

Considerando $||E||$ = { 1 se a expressão for verdadeira; 0, caso contrário

$$err(h) = \frac{1}{n} \sum_{i=1}^{n} || \; y_i \neq h(x_i) ||$$

Acurácia (acc(h))

Complemento da Taxa de Erro, representa a Precisão do Classificador.

$$acc(h) = 1 - err(h)$$

Aprendizado de Máquina: Definições

Distribuição de Classes (distr(Cj))

Para cada Classe Cj, sua distribuição distr(Cj) é calculada como sendo o número de exemplos em T que possuem classe Cj dividido pelo número total de exemplos (*n*), ou seja, a proporção de exemplos em cada classe

$$distr(C_j) = \frac{1}{n} \sum_{i=1}^{n} \| y_i = C_j \|$$

Exemplo: Conjunto com 100 Exemplos: 60 Classe C1
15 Classe C2
25 Classe C3

distribuição (C1, C2, C3) = (60%, 15%, 25%)

Neste exemplo, a **Classe Majoritária** (ou **Prevalente**) é C1; e a Classe **Minoritária** é C2.

Aprendizado de Máquina: Definições

Erro Majoritário (maj-err(T))

Limite Máximo abaixo do qual o erro de um Classificador deve ficar

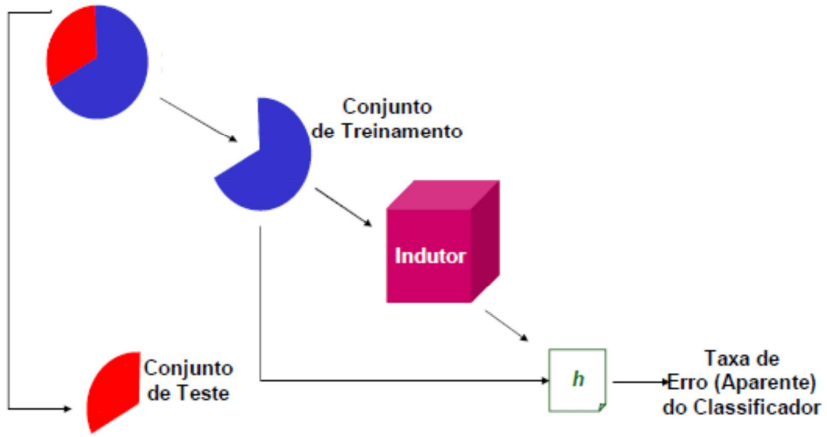
$$maj - err(T) = 1 - \max_{i=1,...k} distr(C_i)$$

No Exemplo anterior: maj-err(T) = 1 - 0,60 = 0,40

Erro Majoritário **INDEPENDE** do algoritmo de aprendizado.

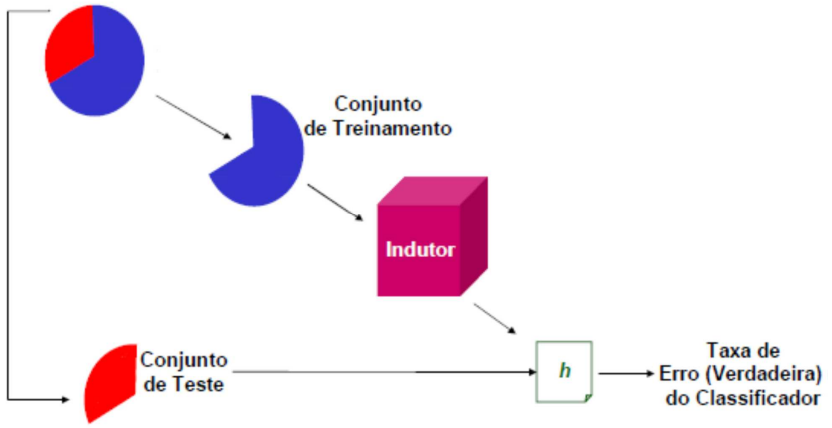
Aprendizado de Máquina: Definições

ERRO APARENTE



Aprendizado de Máquina: Definições

ERRO VERDADEIRO



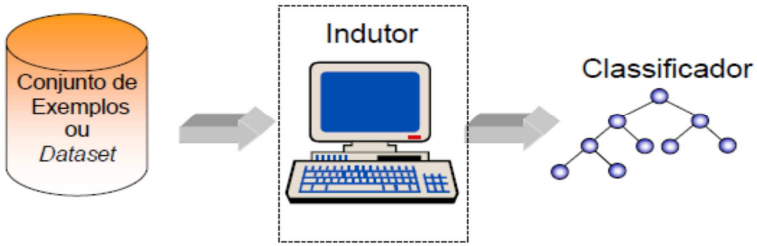
Ruído

- Exemplos imperfeitos que podem ser derivados do processo de aquisição, transformação ou rotulação das classes.
- **Exemplo**: instâncias com os mesmos valores de atributos mas com classes diferentes.

	ATRIBUTOS				CLASSE
Dia	Tempo	Temperatura	Umidade	Vento	Joga-Tenis
1	Sol	Quente	Alta	Fraco	Não
2	Sol	Quente	Alta	Forte	Não
3	Nublado	Quente	Alta	Fraco	Sim
4	Chuva	Moderado	Alta	Fraco	Sim
5	Chuva	Frio	Normal	Forte	Sim
6	Chuva	Frio	Normal	Forte	Não
7	Nublado	Frio	Normal	Forte	Sim
8	Sol	Moderado	Alta	Fraco	Não
9	Sol	Frio	Normal	Fraco	Sim

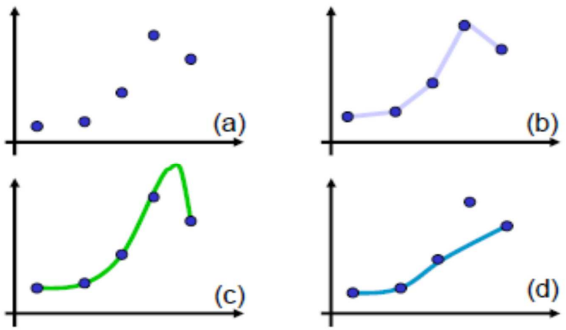
Classificador

- Algumas Definições em AM
  - **Indutor:** programa que gera uma hipótese (**classificador**) a partir de um conjunto de exemplos rotulados.



Exemplos de Hipóteses

- (a) exemplos originais.
- (b), (c), (d) possíveis hipóteses.

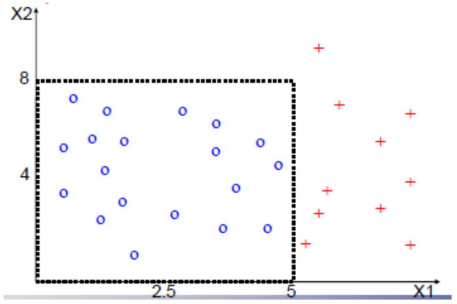


Espaço de Descrição

- ***m* atributos** podem ser vistos como um **vetor**.
- Assim, cada atributo corresponde a uma coordenada em um espaço ***m*-dimensional** denominado **espaço de descrição**.
- No **Aprendizado Supervisionado**, cada ponto no espaço de descrição pode ser rotulado com a classe associada.

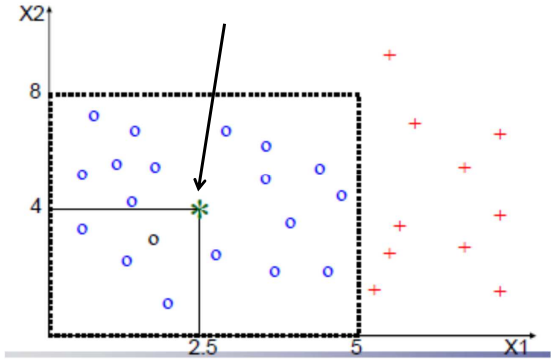
Espaço de Descrição

- Um indutor divide o espaço de descrição em regiões.
- Cada região é rotulada com uma classe.
- **Exemplo**: m = 2 atributos (positivos) e seja o classificador:
  if X1 < 5 **and** X2 < 8 **then** classe = **o** **else** classe = **+**
  divide o espaço bidimensional em duas regiões.



Espaço de Descrição ...

- Para classificar um novo exemplo com **(X1, X2) = (2.5, 4)**, basta verificar em qual região ela se localiza e atribuir a classe associada àquela região (neste caso, classe **o**).

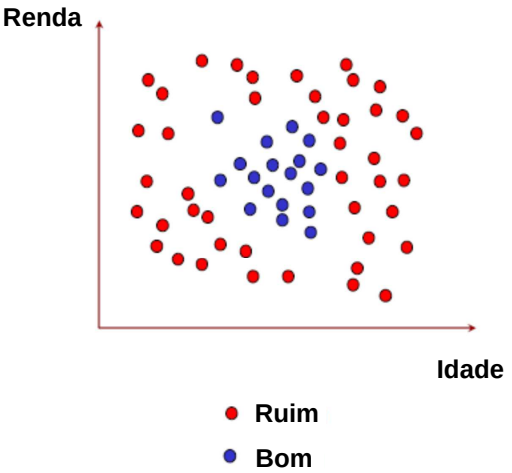


Espaço de Descrição: Exemplo

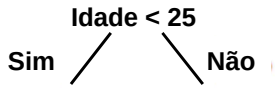
- Assuma o seguinte conjunto de dados sobre exemplos de **crédito bancário**.

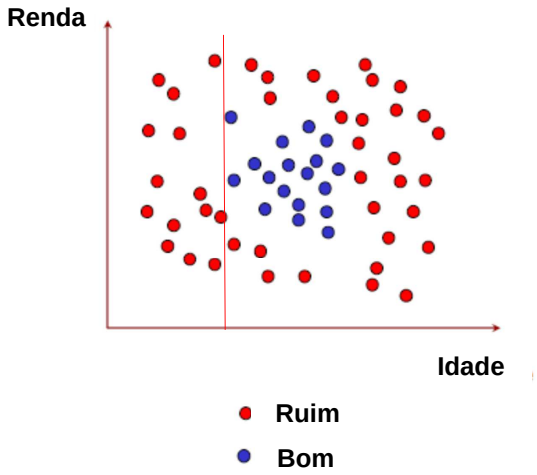
Idade	Renda	Classe
20	2000	Ruim
30	5100	Bom
60	5000	Ruim
40	6000	Bom
...	...	...

Espaço de Descrição: Exemplo

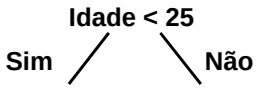


Espaço de Descrição: Exemplo

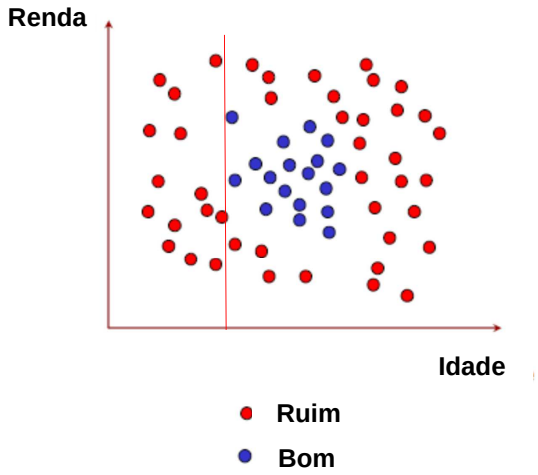




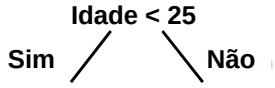
Espaço de Descrição: Exemplo



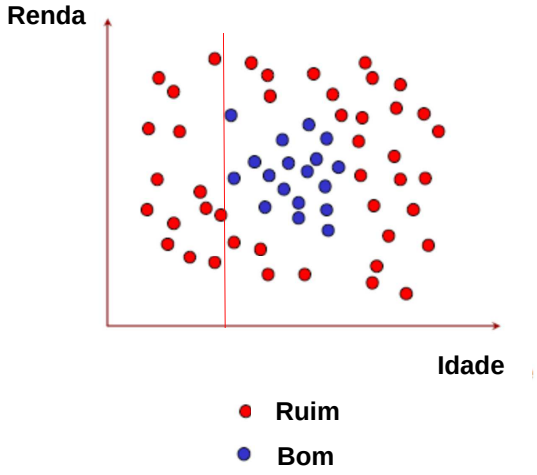
14 – Ruim
0 – Bom



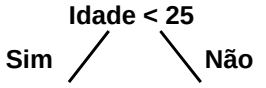
Espaço de Descrição: Exemplo



Ruim

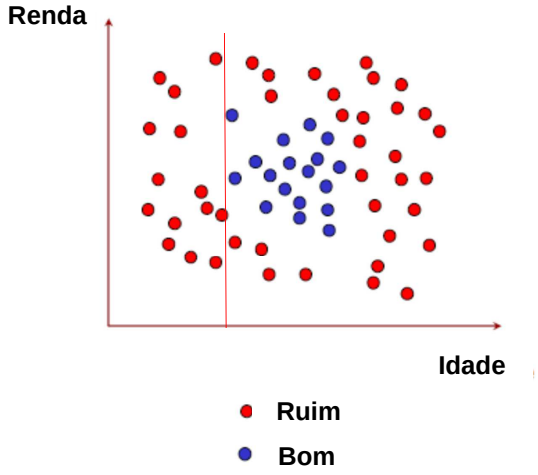


Espaço de Descrição: Exemplo

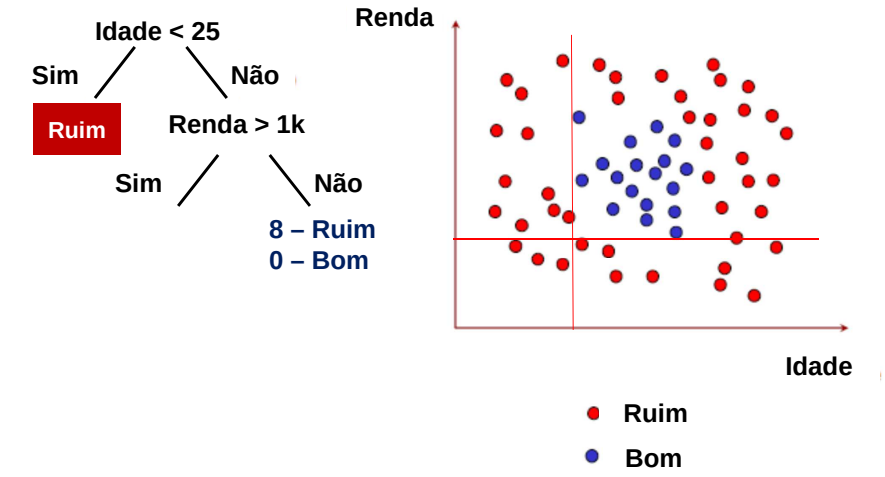


Ruim

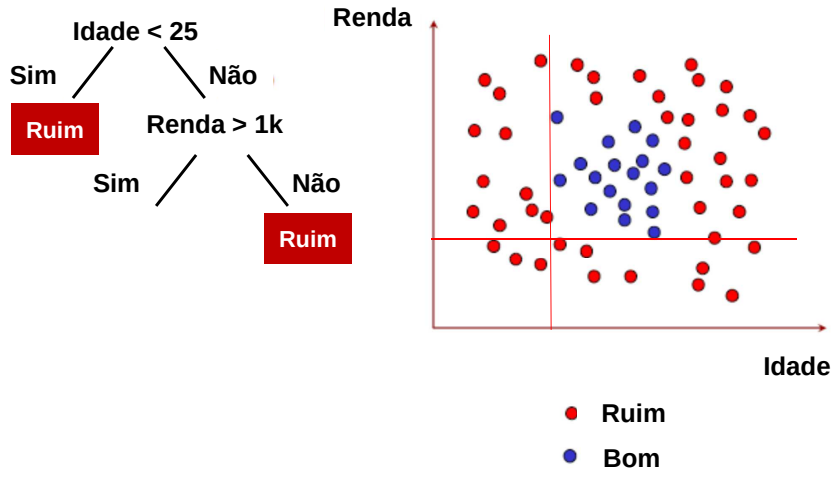
29 – Ruim
19 – Bom



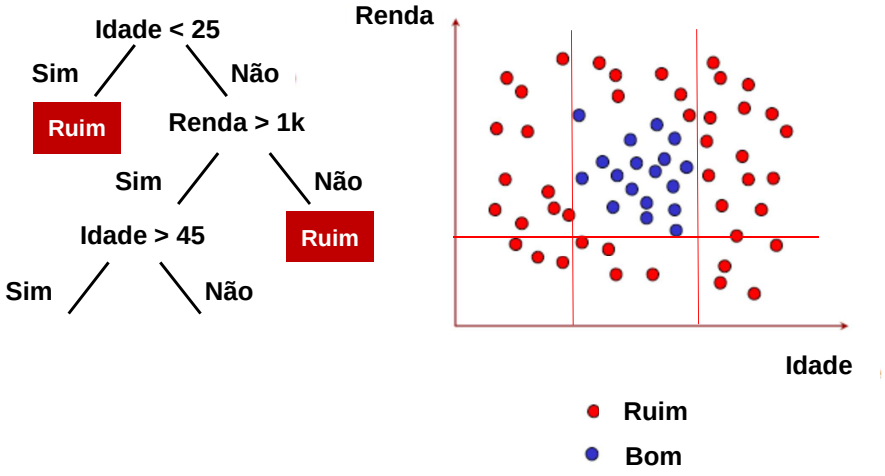
Espaço de Descrição: Exemplo



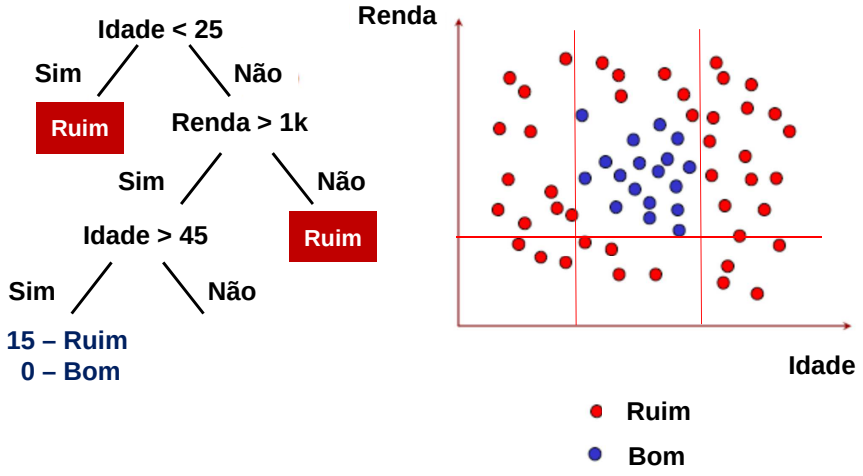
Espaço de Descrição: Exemplo



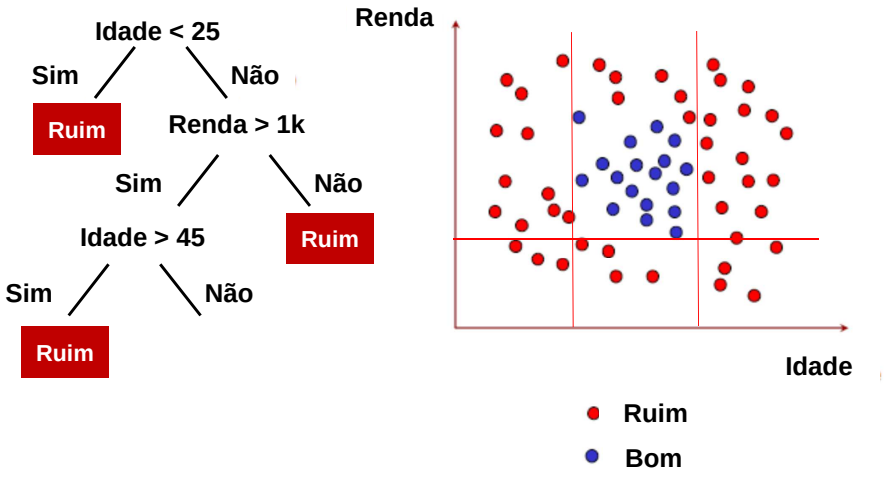
Espaço de Descrição: Exemplo



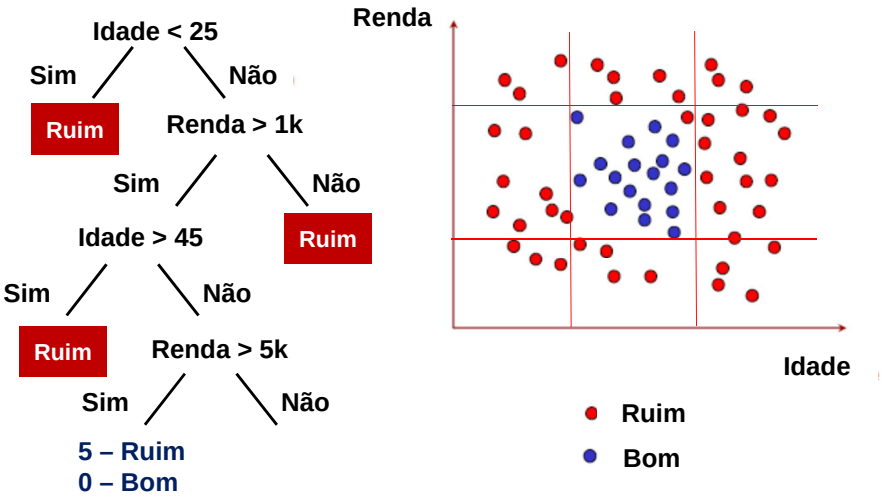
Espaço de Descrição: Exemplo



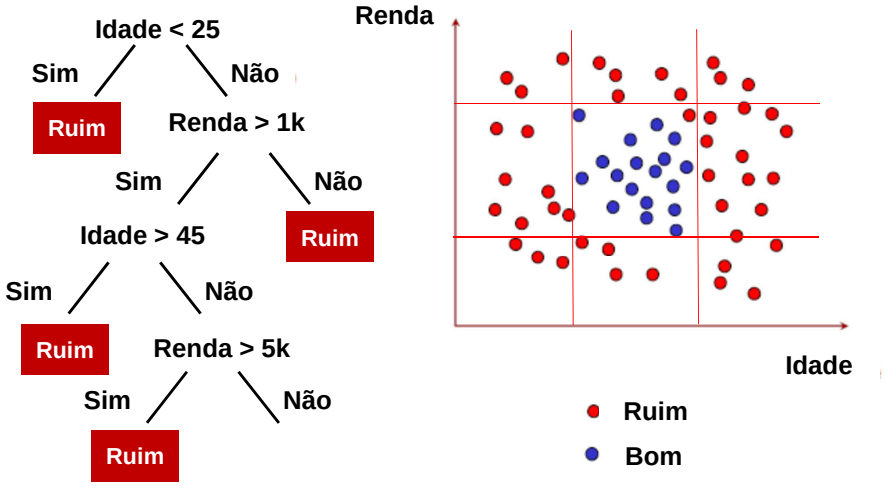
Espaço de Descrição: Exemplo



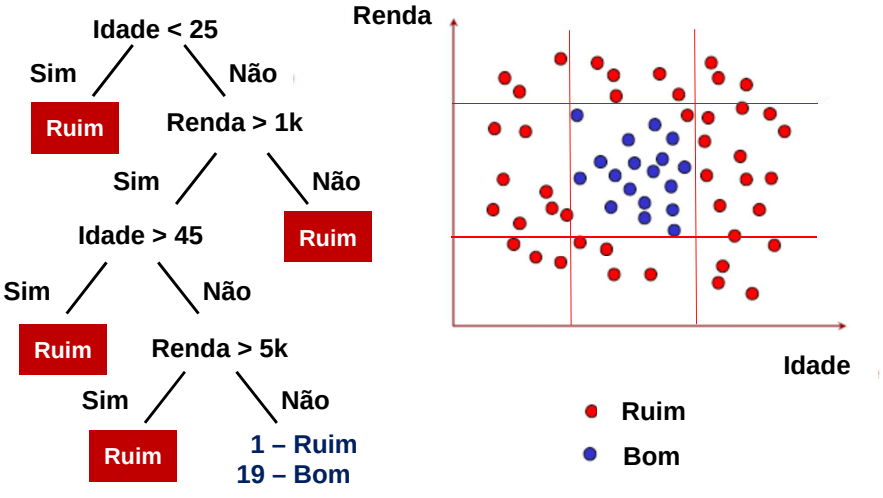
Espaço de Descrição: Exemplo



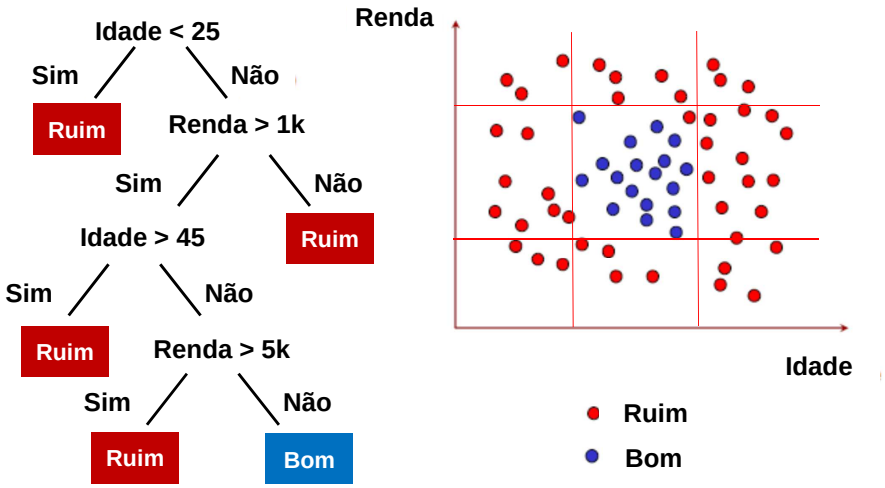
Espaço de Descrição: Exemplo



Espaço de Descrição: Exemplo



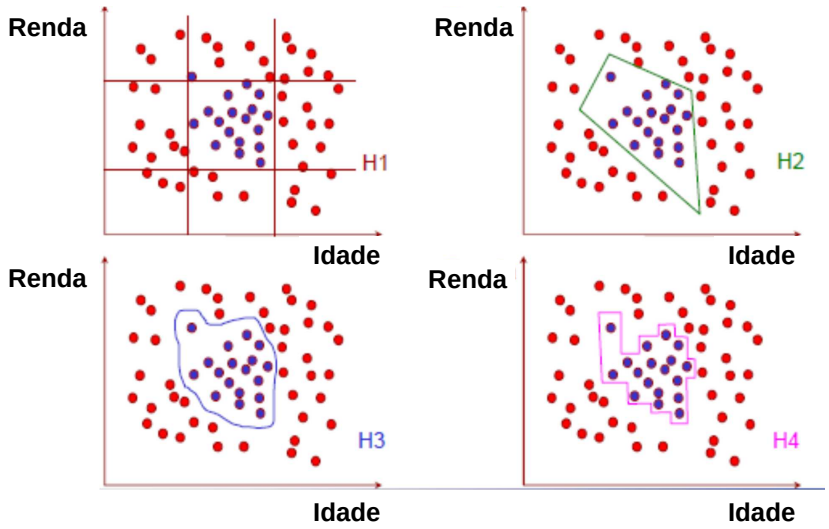
Espaço de Descrição: Exemplo



Erro x Precisão

- Principais fatores de erro:
  - **Qualidade** (**representatividade**) da informação dos atributos.
  - **Adaptação** do algoritmo de aprendizado aos exemplos.
  - **Quantidade** de exemplos.
  - **Distribuição** dos exemplos futuros.

Erro x Possíveis Hipóteses



Overfitting x Underfitting

"Overfitting"

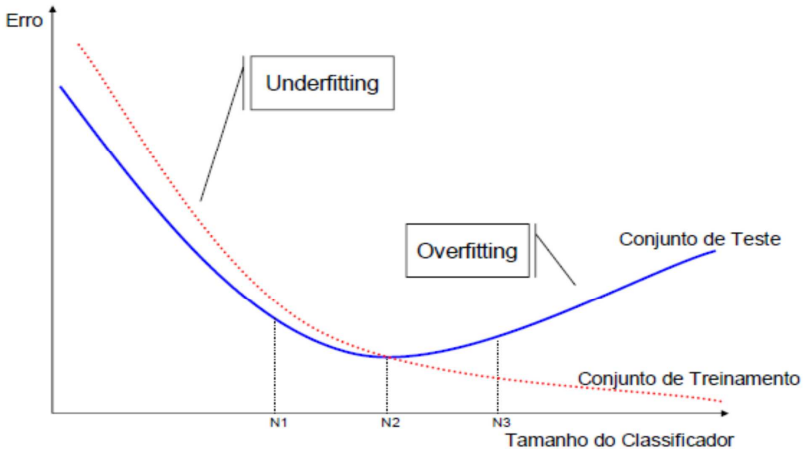
• É possível que o Classificador faça uma indução muito específica para o conjunto de treinamento utilizado (**"Overfitting"**).

• Como este é apenas uma amostra de dados, é possível que a indução tenha bom desempenho no conjunto de treinamento, mas um desempenho ruim em exemplos diferentes daqueles pertencentes ao conjunto de treinamento.

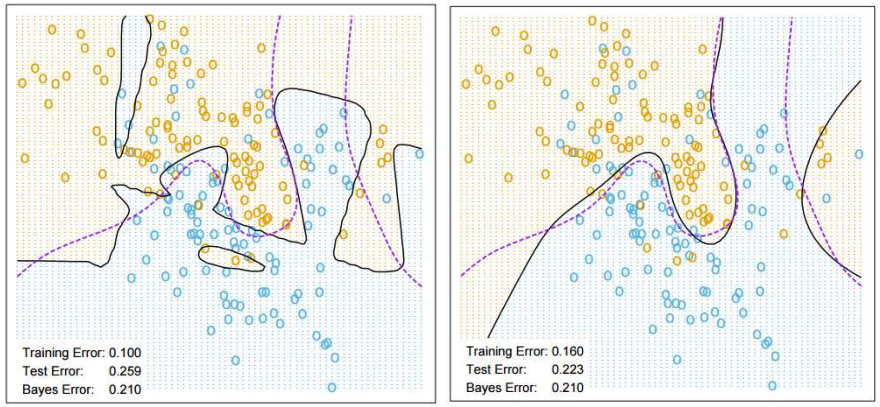
• **Cálculo do Erro** em um conjunto de teste independente evidencia a situação de **"Overfitting"**.

• **Underfitting** e **overfitting**: a hipótese se ajusta **muito pouco** ou **em excesso** ao conjunto de treinamento.

Tamanho do Classificador x Erro

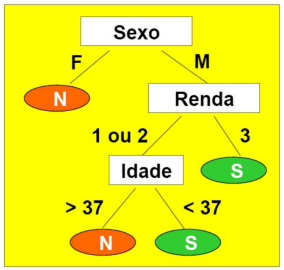


Exemplo de overfitting com RNA



FONTE: Hastie, T., Tibshirani, R., Friedman, J. The elements of statistical learning (2009), pg. 399.

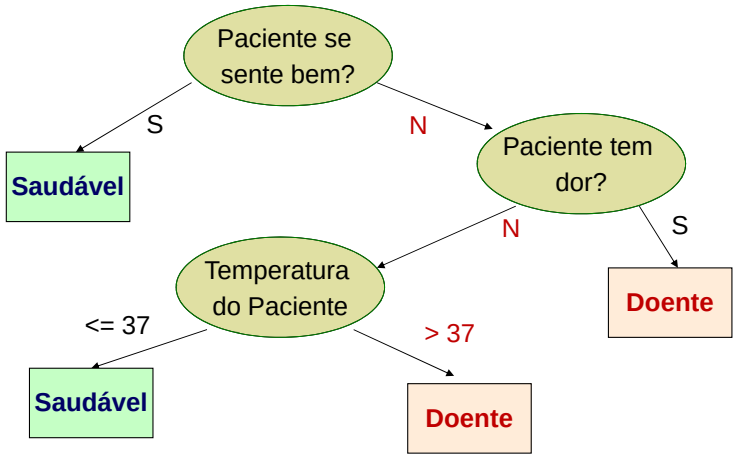
Árvores de Decisão



Árvore de Decisão

- Árvore de decisão
  - Um fluxograma com a estrutura de uma árvore.
  - Nó interno representa um teste sobre um atributo.
  - Cada ramo representa um resultado do teste.
  - Folhas representam as classes.
- A geração de uma árvore consiste de duas fases:
  - **Construção da árvore**
    - Particionamento de atributos (**best fit**).
  - **Fase da poda** (**Tree pruning**).
    - Identifica e remove ramos que refletem ruídos ou outliers.
- **Uso da árvore**: Classificação de amostras desconhecidas
  - Testa os valores dos atributos da amostra "**contra**" a árvore.

Árvore de Decisão – Exemplo



Árvore de Decisão – Exemplo ...

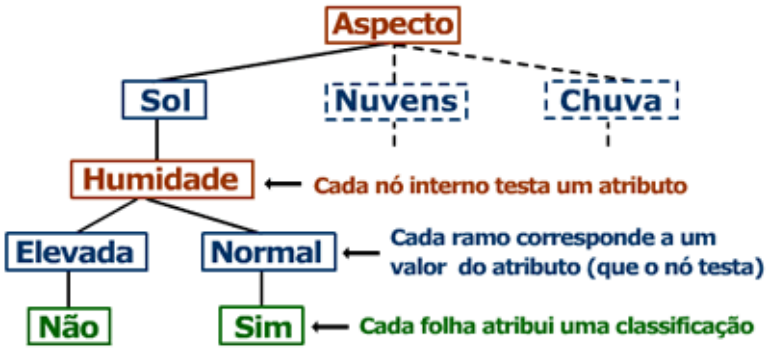
- **Geração de regras**

Se paciente se sente bem = ***sim***
  então classe = ***saudável***
fim se.
Se paciente se sente bem = **não**
  e paciente tem dor = **sim**
  então classe = **doente**
fim se.

...

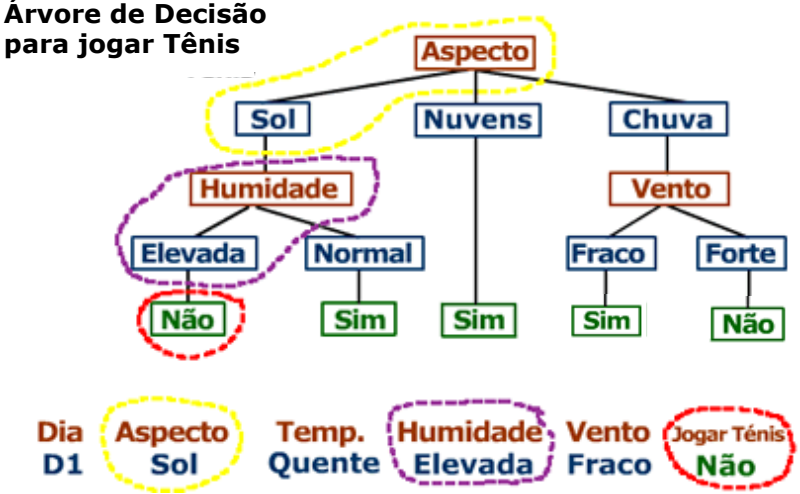
Árvore de Decisão – Exemplo ...

Árvore de Decisão para jogar Tênis



Árvore de Decisão – Exemplo ...

Árvore de Decisão para jogar Tênis



Algoritmos para árvores de decisão

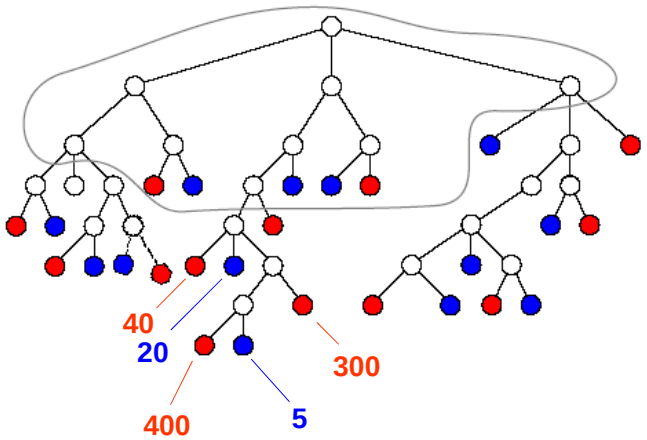
- Algoritmo Básico (**algoritmo guloso**)
  - A árvore é construída recursivamente no sentido **top-down (divisão para conquista).**
  - No início, todas as amostras estão na raiz.
  - Os atributos são nominais (se numéricos, eles são discretizados).
  - Amostras são particionadas recursivamente com base nos atributos selecionados.
  - Atributos "**testes**" são selecionados com base em heurísticas ou medidas estatísticas (ex., **ganho de informação**) **[ID3 / C4.5]**
- Condições de parada do particionamento
  - Todas as amostras de um nó pertencem a mesma classe.
  - Não existem mais atributos para particionamento.
  - Não existem mais amostras no conjunto de treinamento.

Árvore de Decisão: Poda

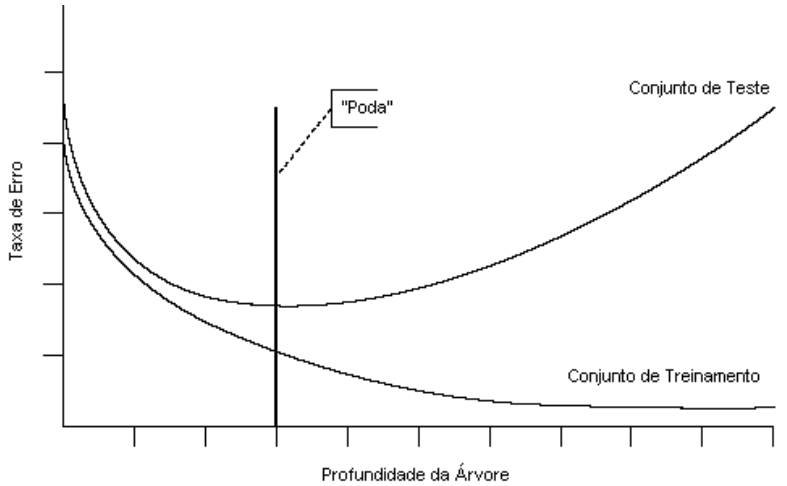
PODA

- Técnica para lidar com ruído e **"Overfitting"**
- **Pré-Poda**: Durante a geração da Hipótese.
  - Alguns exemplos de treinamento são deliberadamente ignorados.
- **Pós-Poda**: Inicialmente, é gerada um Classificador que explique os exemplos.
  - Após isso, elimina-se algumas partes (**cortes em ramos da árvore**) generalizando a Hipótese.

Árvore de Decisão: Poda ...



Árvore de Decisão: Poda ...



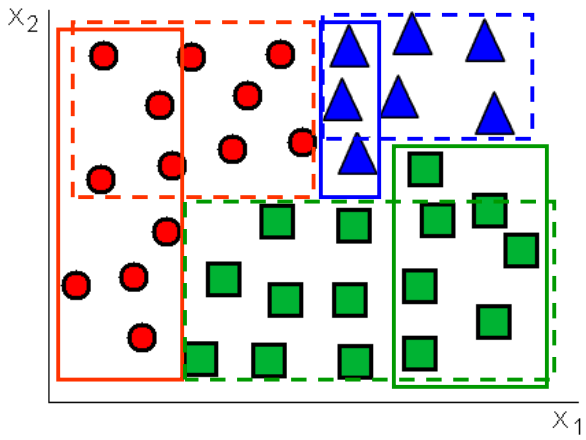
Completude e Consistência

COMPLETUDE E CONSISTÊNCIA

- **COMPLETUDE**: Se a Hipótese gerada pelo Classificador classifica **TODOS** os exemplos.
- **CONSISTÊNCIA**: Se a Hipótese gerada pelo Classificador classifica **CORRETAMENTE** os exemplos.
- Uma Hipótese gerada pelo Classificador pode ser:
  - Completa e Consistente.
  - Incompleta e Consistente.
  - Completa e Inconsistente.
  - Incompleta e Inconsistente.

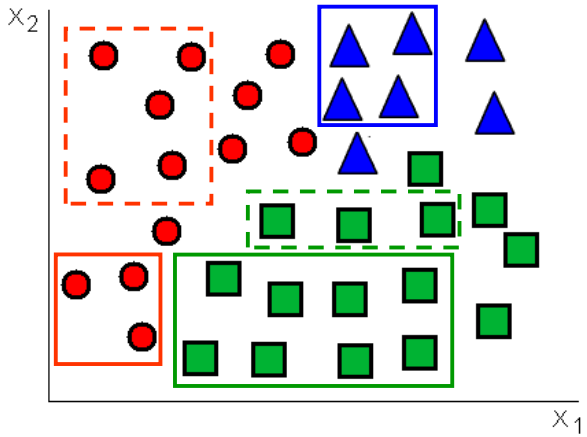
Completude e Consistência ...

COMPLETO e CONSISTENTE



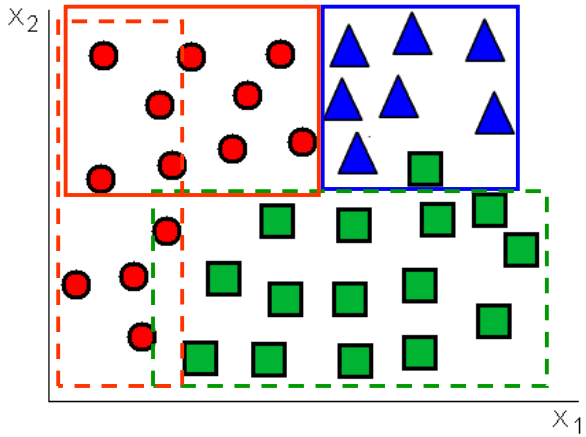
Completude e Consistência ...

INCOMPLETO e CONSISTENTE



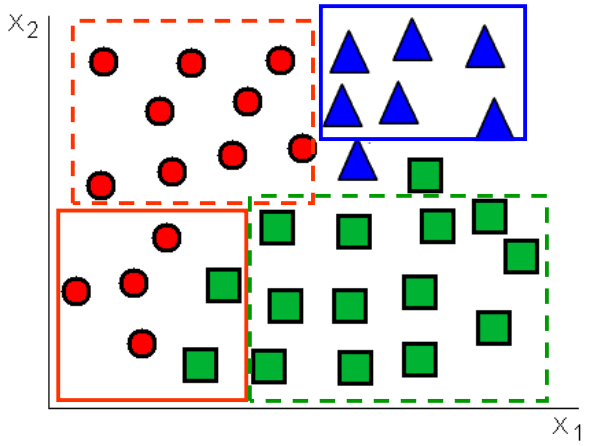
Completude e Consistência ...

COMPLETO e INCONSISTENTE



Completude e Consistência ...

INCOMPLETO e INCONSISTENTE



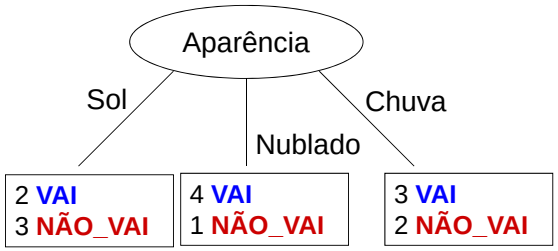
Completude e Consistência: Exemplo

Exemplo	Aparência	Temperatura	Umidade	Ventando	Viajar
T1	sol	25	72	sim	VAI
T2	sol	28	91	sim	NÃO_VAI
T3	sol	22	70	não	VAI
T4	sol	23	95	não	NÃO_VAI
T5	sol	30	85	não	NÃO_VAI
T6	nublado	23	90	sim	VAI
T7	nublado	29	78	não	VAI
T8	nublado	19	65	sim	NÃO_VAI
T9	nublado	26	75	não	VAI
T10	nublado	20	87	sim	VAI
T11	chuva	22	95	não	VAI
T12	chuva	19	70	sim	NÃO_VAI
T13	chuva	23	80	sim	NÃO_VAI
T14	chuva	25	81	não	VAI
T15	chuva	21	80	não	VAI

Completude e Consistência: Exemplo

Exemplo	Aparência	Temperatura	Umidade	Ventando	Viajar
T1	sol	25	72	sim	VAI
T2	sol	28	91	sim	NÃO_VAI
T3	sol	22	70	não	VAI
T4	sol	23	95	não	NÃO_VAI
T5	sol	30	85	não	NÃO_VAI
T6	nublado	23	90	sim	VAI
T7	nublado	29	78	não	VAI
T8	nublado	19	65	sim	NÃO_VAI
T9	nublado	26	75	não	VAI
T10	nublado	20	87	sim	VAI
T11	chuva	22	95	não	VAI
T12	chuva	19	70	sim	NÃO_VAI
T13	chuva	23	80	sim	NÃO_VAI
T14	chuva	25	81	não	VAI
T15	chuva	21	80	não	VAI

Completude e Consistência: Exemplo



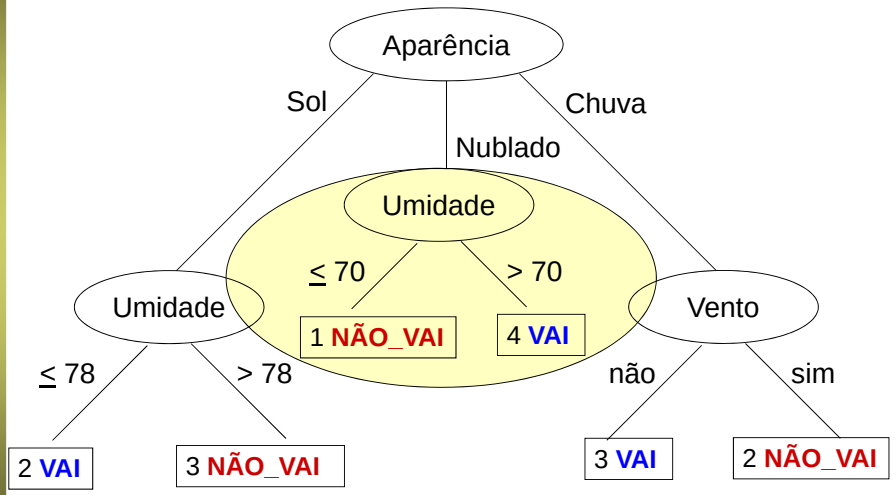
Completude e Consistência: Exemplo

Exemplo	Aparência	Temperatura	Umidade	Ventando	Viajar
T1	sol	25	72	sim	VAI
T2	sol	28	91	sim	NÃO_VAI
T3	sol	22	70	não	VAI
T4	sol	23	95	não	NÃO_VAI
T5	sol	30	85	não	NÃO_VAI
T6	nublado	23	90	sim	VAI
T7	nublado	29	78	não	VAI
T8	nublado	19	65	sim	NÃO_VAI
T9	nublado	26	75	não	VAI
T10	nublado	20	87	sim	VAI
T11	chuva	22	95	não	VAI
T12	chuva	19	70	sim	NÃO_VAI
T13	chuva	23	80	sim	NÃO_VAI
T14	chuva	25	81	não	VAI
T15	chuva	21	80	não	VAI

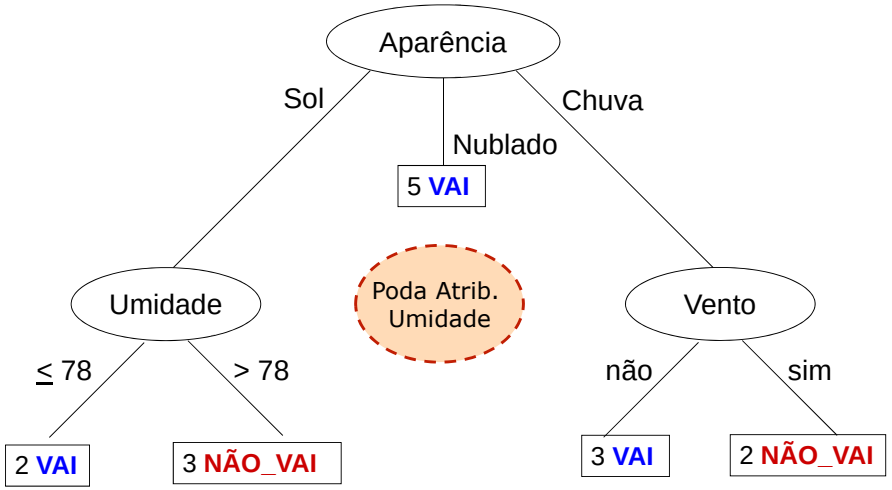
Completude e Consistência: Exemplo

Exemplo	Aparência	Temperatura	Umidade	Ventando	Viajar
T1	sol	25	72	sim	VAI
T2	sol	28	91	sim	NÃO_VAI
T3	sol	22	70	não	VAI
T4	sol	23	95	não	NÃO_VAI
T5	sol	30	85	não	NÃO_VAI
T6	nublado	23	90	sim	VAI
T7	nublado	29	78	não	VAI
T8	nublado	19	65	sim	NÃO_VAI
T9	nublado	26	75	não	VAI
T10	nublado	20	87	sim	VAI
T11	chuva	22	95	não	VAI
T12	chuva	19	70	sim	NÃO_VAI
T13	chuva	23	80	sim	NÃO_VAI
T14	chuva	25	81	não	VAI
T15	chuva	21	80	não	VAI

Completude e Consistência: Exemplo



Completude e Consistência: Exemplo



Algoritmos mais conhecidos

- **ID3** (**Iterative Dichotomiser 3**) (Quilan,1986):
  - Um algoritmo usado para gerar árvores de decisão. Os atributos do conjunto de dados devem ser obrigatoriamente categóricos.
- **C4.5** (**J48 no Weka**) (Quilan, 1993):
  - Um algoritmo para geração de árvores de decisão, sucessor do algoritmo ID3.
  - O algoritmo C4.5 considera atributos numéricos e categóricos.
- **CART** (**C**lassification **A**nd **R**egression **T**rees) (Breiman et al., 1984):
  - Técnica não-paramétrica que produz árvores de classificação ou regressão, dependendo se as variáveis são categóricas ou numéricas, respectivamente.

Como escolher o melhor atributo?

Escolha do melhor atributo "split"

- **Não existe solução computacionalmente viável** para que se obtenha sempre a melhor árvore de decisão possível (problema **NP-completo**: custo de proceder buscas exaustivas da melhor solução cresce a taxas exponenciais à medida que o tamanho do conjunto de treinamento aumenta).
- **Utilização de heurísticas**: soluções baseadas em algum tipo de conhecimento prévio sobre as propriedades dos dados, na procura de uma boa solução (mas não necessariamente a melhor).

Como escolher o melhor atributo?

Exemplo: Conjunto de todas soluções possíveis (**floresta de decisão**).

BUSCA EXAUSTIVA:
Correr todo esse conjunto, comparando cada elemento, até que todos tenham sido avaliados, e selecionar a melhor solução.
SOLUÇÃO ÓTIMA GARANTIDA.

BUSCA HEURÍSTICA:
Procura tendenciosa na floresta, visitando apenas as soluções com mais potencial de serem boas, com base em algumas premissas previamente conhecidas.

A rapidez do processo aumenta, mas é possível que a melhor solução entre todas não tenha sido encontrada, pois eventualmente pode ter ficado fora do trajeto percorrido.

Como escolher o melhor atributo?

Problema: Como definir alguma característica sobre os dados que permita definir um critério para identificação do melhor atributo em cada nível da árvore ?

Abordagem baseada na Teoria da Informação

Boa subdivisão:

Produz grupos mais homogêneos com relação ao atributo categórico.

Idéia → Classificação evidencia as linhas gerais que fazem um elemento pertencer a uma determinada classe, o que é facilitado quando se produz agrupamentos mais organizados.

Melhor atributo "split"

Atributo mais informativo sobre a lógica dos dados num determinado contexto.

Como escolher o melhor atributo?

CASCA	COR	TAMANHO	POLPA	RISCO
aspera	marrom	grande	dura	baixo
aspera	verde	grande	dura	baixo
lisa	vermelho	grande	macia	alto
aspera	verde	grande	macia	baixo
aspera	vermelho	pequena	dura	baixo
lisa	vermelho	pequena	dura	baixo
lisa	marrom	pequena	dura	baixo
aspera	verde	pequena	macia	alto
lisa	verde	pequena	dura	alto
aspera	vermelho	grande	dura	baixo
lisa	marrom	grande	macia	baixo
lisa	verde	pequena	macia	alto
aspera	vermelho	pequena	macia	baixo
lisa	vermelho	grande	dura	alto
lisa	vermelho	pequena	dura	baixo
aspera	verde	pequena	dura	alto

Como escolher o melhor atributo?

Cálculo da Entropia ⟶ $- \log_2 p (c_i | a_j)$

"Quantidade de informação" que a_j tem a oferecer sobre a conclusão c_i

Entropia = $-\sum_{i=1}^{n} p(c_i | a_j) \log_2 p(c_i | a_j)$

Quanto menor a Entropia ⟶ Menor a "dúvida"

↓

Maior a informação que a_j traz sobre C

Melhor atributo "split" ⟶ Subconjuntos mais homogêneos (grupos menos "confusos" com relação à classe).

Conceito de Entropia (**Termodinâmica**):

Inversamente proporcional ao grau de informação (valor entre 0 e 1)

Como escolher o melhor atributo?

$$Entropia(S) = -\sum_{i=1}^{n} p_i \; Log_2(p_i)$$

Onde: ***S*** é a distribuição de probabilidade das ***n*** mensagens possíveis;
p_i é a probabilidade de ocorrência da *i*-ésima mensagem

• Para o caso de um classificador construído para um problema com 2 classes possíveis (A e B), um atributo ***x*** vai permitir dividir os dados em tantos subconjuntos ***S*** quantos forem os seus possíveis valores.

• A entropia de cada um desses subconjuntos S_k seria calculada por:

$$Entropia\,(S_k) = -p_A \; Log_2(p_A) - p_B \; Log_2(p_B)$$

P(A)	P(B)	Entropia
0,50	0,50	1,00
0,67	0,33	0,92
1,00	0,00	0,00

Quanto mais uniforme a distribuição, maior o grau de entropia.

Como escolher o melhor atributo?

Para o caso de um atributo ***x*** que possa assumir três valores (por exemplo, valores inteiros entre 1 e 3), três subconjuntos de ***S*** são formados, cada um com seu próprio grau de entropia.

Pode-se avaliar a entropia em ***S*** quando considerado o atributo ***x***, através da média ponderada dos graus de entropia dos subconjuntos gerados (***S₁***, ***S₂*** e ***S₃***, neste exemplo).

$$Entropia\,(x,S) = \sum_{i=1}^{n} \frac{|S_i|}{|S|} \cdot Entropia\,(S_i)$$

Ganho de Informação (*"**Information Gain**"*) devido a ***x*** na predição da classe é determinada pela redução da entropia original de ***S***.

$$Ganho\ de\ Informação\,(x,S) = Entropia\,(S) - Entropia\,(x,S)$$

Como escolher o melhor atributo?

Entropia de um Atributo A com relação à Classe C

$$-\sum_{j=1}^{m} p(a_j) \sum_{i=1}^{n} p(c_i \mid a_j) \log_2 p(c_i \mid a_j)$$

Atributo com **MENOR** entropia é o **MELHOR** para determinar a Classe

CASCA	COR	TAMANHO	POLPA	RISCO
aspera	marrom	grande	dura	baixo
aspera	verde	grande	dura	baixo
lisa	vermelho	grande	macia	alto
aspera	verde	grande	macia	baixo
aspera	vermelho	pequena	dura	baixo
lisa	vermelho	pequena	dura	baixo
lisa	marrom	pequena	dura	baixo
aspera	verde	pequena	macia	alto
lisa	verde	pequena	dura	alto
aspera	vermelho	grande	dura	baixo
lisa	marrom	grande	macia	baixo
lisa	verde	pequena	macia	alto
aspera	vermelho	pequena	macia	baixo
lisa	vermelho	grande	dura	alto
lisa	vermelho	pequena	dura	baixo
aspera	verde	pequena	dura	alto

Consideremos o Atributo "**Casca**"

p(baixo | aspera) = 6 / 8
p(alto | aspera) = 2 / 8
p(aspera) = 8 / 16

p(baixo | lisa) = 4 / 8
p(alto | lisa) = 4 / 8
p(lisa) = 8 / 16

Entropia para o Atributo "**Casca**"

$$-\frac{8}{16}\left(\frac{6}{8}\log\left(\frac{6}{8}\right)+\frac{2}{8}\log\left(\frac{2}{8}\right)\right)+\frac{8}{16}\left(\frac{4}{8}\log\left(\frac{4}{8}\right)+\frac{4}{8}\log\left(\frac{4}{8}\right)\right)$$

0.90564

CASCA	COR	TAMANHO	POLPA	RISCO
aspera	marrom	grande	dura	baixo
aspera	verde	grande	dura	baixo
lisa	vermelho	grande	macia	alto
aspera	verde	grande	macia	baixo
aspera	vermelho	pequena	dura	baixo
lisa	vermelho	pequena	dura	baixo
lisa	marrom	pequena	dura	baixo
aspera	verde	pequena	macia	alto
lisa	verde	pequena	dura	alto
aspera	vermelho	grande	dura	baixo
lisa	marrom	grande	macia	baixo
lisa	verde	pequena	macia	alto
aspera	vermelho	pequena	macia	baixo
lisa	vermelho	grande	dura	alto
lisa	vermelho	pequena	dura	baixo
aspera	verde	pequena	dura	alto

Consideremos o Atributo "**Cor**"

p(baixo | marrom) = 3 / 3
p(alto | marrom) = 0 / 3
p(marrom) = 3 / 16

p(baixo | verde) = 2 / 6
p(alto | verde) = 4 / 6
p(verde) = 6 / 16

p(baixo | vermelho) = 5 / 7
p(alto | vermelho) = 2 / 7
p(vermelho) = 7 / 16

Entropia para o Atributo "**Cor**"

$$\frac{3}{16}\left(\frac{3}{3}\log\left(\frac{3}{3}\right)+\frac{0}{3}\log\left(\frac{0}{3}\right)\right)+\frac{6}{16}\left(\frac{2}{6}\log\left(\frac{2}{6}\right)+\frac{4}{6}\log\left(\frac{4}{6}\right)\right)+\frac{7}{16}\left(\frac{5}{7}\log\left(\frac{5}{7}\right)+\frac{2}{7}\log\left(\frac{2}{7}\right)\right)$$

0.721976

CASCA	COR	TAMANHO	POLPA	RISCO
aspera	marrom	grande	dura	baixo
aspera	verde	grande	dura	baixo
lisa	vermelho	grande	macia	alto
aspera	verde	grande	macia	baixo
aspera	vermelho	pequena	dura	baixo
lisa	vermelho	pequena	dura	baixo
lisa	marrom	pequena	dura	baixo
aspera	verde	pequena	macia	alto
lisa	verde	pequena	dura	alto
aspera	vermelho	grande	dura	baixo
lisa	marrom	grande	macia	baixo
lisa	verde	pequena	macia	alto
aspera	vermelho	pequena	macia	baixo
lisa	vermelho	grande	dura	alto
lisa	vermelho	pequena	dura	baixo
aspera	verde	pequena	dura	alto

Consideremos o Atributo "**Tamanho**"

p(baixo | grande) = 5 / 7
p(alto | grande) = 2 / 7
p(grande) = 7 / 16

p(baixo | pequeno) = 5 / 9
p(alto | pequeno) = 4 / 9
p(pequeno) = 9 / 16

Entropia para o Atributo "**Tamanho**"

$$\frac{7}{16}\left(\frac{5}{7}\log\left(\frac{5}{7}\right)+\frac{2}{7}\log\left(\frac{2}{7}\right)\right)+\frac{9}{16}\left(\frac{5}{9}\log\left(\frac{5}{9}\right)+\frac{4}{9}\log\left(\frac{4}{9}\right)\right)$$

0.9350955

CASCA	COR	TAMANHO	POLPA	RISCO
aspera	marrom	grande	dura	baixo
aspera	verde	grande	dura	baixo
lisa	vermelho	grande	macia	alto
aspera	verde	grande	macia	baixo
aspera	vermelho	pequena	dura	baixo
lisa	vermelho	pequena	dura	baixo
lisa	marrom	pequena	dura	baixo
aspera	verde	pequena	macia	alto
lisa	verde	pequena	dura	alto
aspera	vermelho	grande	dura	baixo
lisa	marrom	grande	macia	baixo
lisa	verde	pequena	macia	alto
aspera	vermelho	pequena	macia	baixo
lisa	vermelho	grande	dura	alto
lisa	vermelho	pequena	dura	baixo
aspera	verde	pequena	dura	alto

Consideremos o Atributo "**Polpa**"

p(baixo | dura) = 7 / 10
p(alto | dura) = 3 / 10
p(dura) = 10 / 16

p(baixo | macia) = 3 / 6
p(alto | macia) = 3 / 6
p(macia) = 6 / 16

Entropia para o Atributo "**Polpa**"

$$\frac{10}{16}\left(\frac{7}{10}\log\left(\frac{7}{10}\right)+\frac{3}{10}\log\left(\frac{3}{10}\right)\right)+\frac{6}{16}\left(\frac{3}{6}\log\left(\frac{3}{6}\right)+\frac{3}{6}\log\left(\frac{3}{6}\right)\right)$$

0.92581

Resultados do cálculo da entropia

Atributo	Entropia
Casca	0.90564
Cor	0.721976
Tamanho	0.9350955
Polpa	0.92581

Como **"Cor"** tem a **menor entropia**, pode-se afirmar que também tem o maior ganho de informação. Logo deve ser usado como **"atributo split"**.

Análise dos Resultados

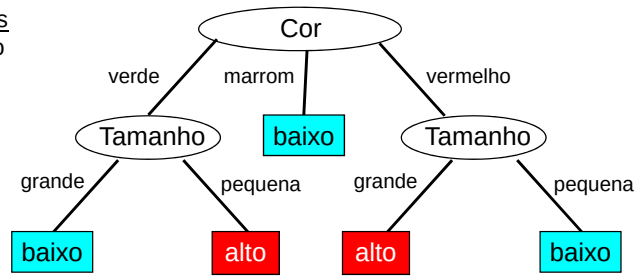
Resultado WEKA

```
=== Confusion Matrix ===
 a b   <-- classified as
 9 1 | a = baixo
 0 6 | b = alto
```

Acurácia: 93.75 %

TP Rate	Class
0.9	baixo
1	alto

```
Cor = marrom: baixo (3.0)
Cor = verde
|   Tamanho = grande: baixo (2.0)
|   Tamanho = pequena: alto (4.0)
Cor = vermelho
|   Tamanho = grande: alto (3.0/1.0)
|   Tamanho = pequena: baixo (4.0)
```



Árvores de decisão: prós e contras

Vantagens

- Custo computacional é baixo.
- Muito rápido para classificar amostras desconhecidas.
- Fácil de interpretar árvores de tamanho pequeno.
- Precisão é semelhante a de outros métodos de classificação, para muitos datasets simples.

Desvantagens

- "**Overfitting**" resulta em árvores de decisão que são mais complexas do que necessárias.
- O treinamento do erro nem sempre produz uma boa estimativa com relação à execução da árvore para amostras desconhecidas.
- Necessita de novas maneiras para estimar erros.