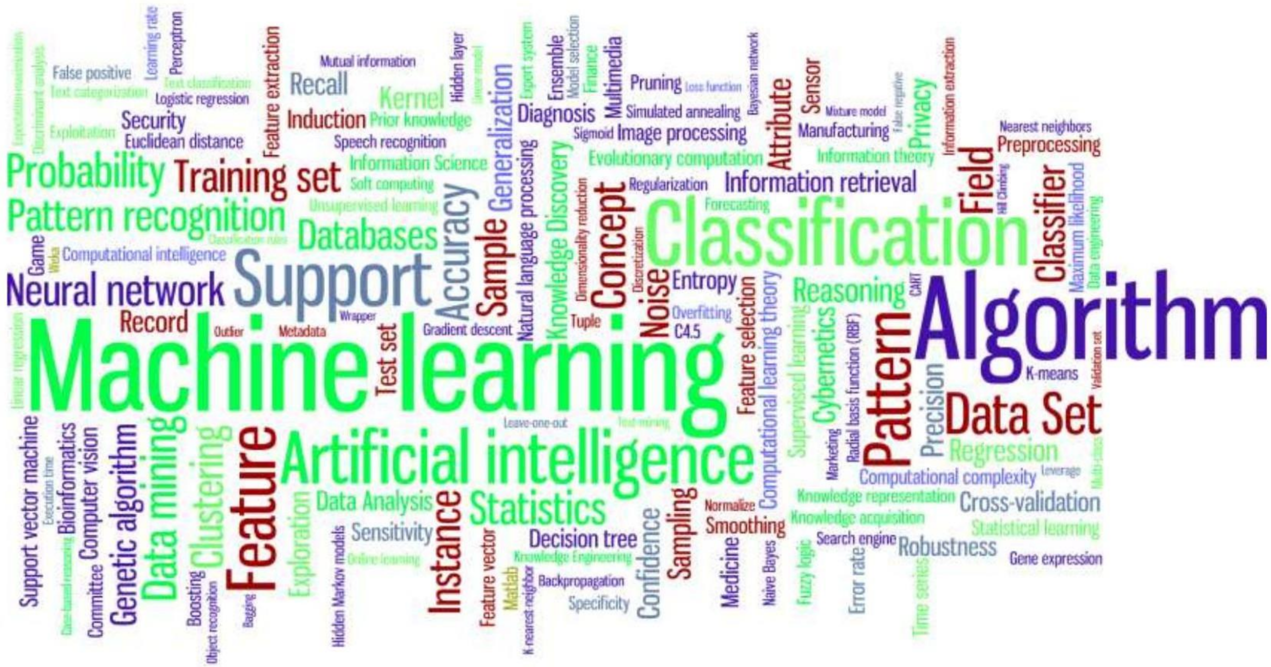
Uma introdução à *Machine Learning*



Aprendizado de Máquinas

- **Herbert Alexander Simon**:
  "Aprendizado é qualquer processo pelo qual um sistema melhora sua *performance* pela experiência."

- "Machine Learning está preocupado com programas de computador que automaticamente melhoram sua *performance* pela experiência. "

- Economista / Matemática 1913 - 2001
  - Simulaçao computacional da Cognição humana (~1954)

Herbert Simon
Turing Award 1975
Nobel Prize in Economics 1978

1913 - 2001

Por que *Machine Learning*?

- Desenvolver sistemas que podem automaticamente se adaptar e se customizar para usuários individuais.
  - Notícias personalizadas OU Filtro de email

- Descobrir novo conhecimento a partir / usando grandes bases de dados (*data mining*).
  - Análise de carrinho de supermer. (e.g. fraldas e cervejas)

Por que *Machine Learning*?

- Habilidade de imitar humanos e substituí-los em certas tarefas monótonas - que exigem alguma inteligência.
  - Como o reconhecimento de caracteres manuscritos

- Desenvolver sistemas que são muito difíceis / caros para construir manualmente porque eles requerem habilidades ou conhecimento detalhados específicos ajustados para uma tarefa específica (gargalo de engenharia do conhecimento).

Por que *AGORA*?

- Inundação de dados disponíveis
  - especialmente com o advento da internet).
- Incremento de força computacional
- Progresso crescente de:
  - algoritmos disponíveis
  - teoria desenvolvida por pesquisadores
- Aumento no suporte/apoio das indústrias

Aplicações em ML

O Conceito de Aprendizado

- **Aprendizado** = **Melhoria** com **experiência** em alguma **tarefa**
  - Melhoria sobre a tarefa **T**
  - Com respeito a medida de desempenho **D**
  - Baseado na experiência **E**

Motivação - Filtro de SPAM

- **Example**: *Spam Filtering*

  Spam (*Sending and Posting Advertisement in Mass*)
  - é todo email que o usuário não queria receber e não autorizou o recebimento

  T: Identificar emails SPAM

  D:

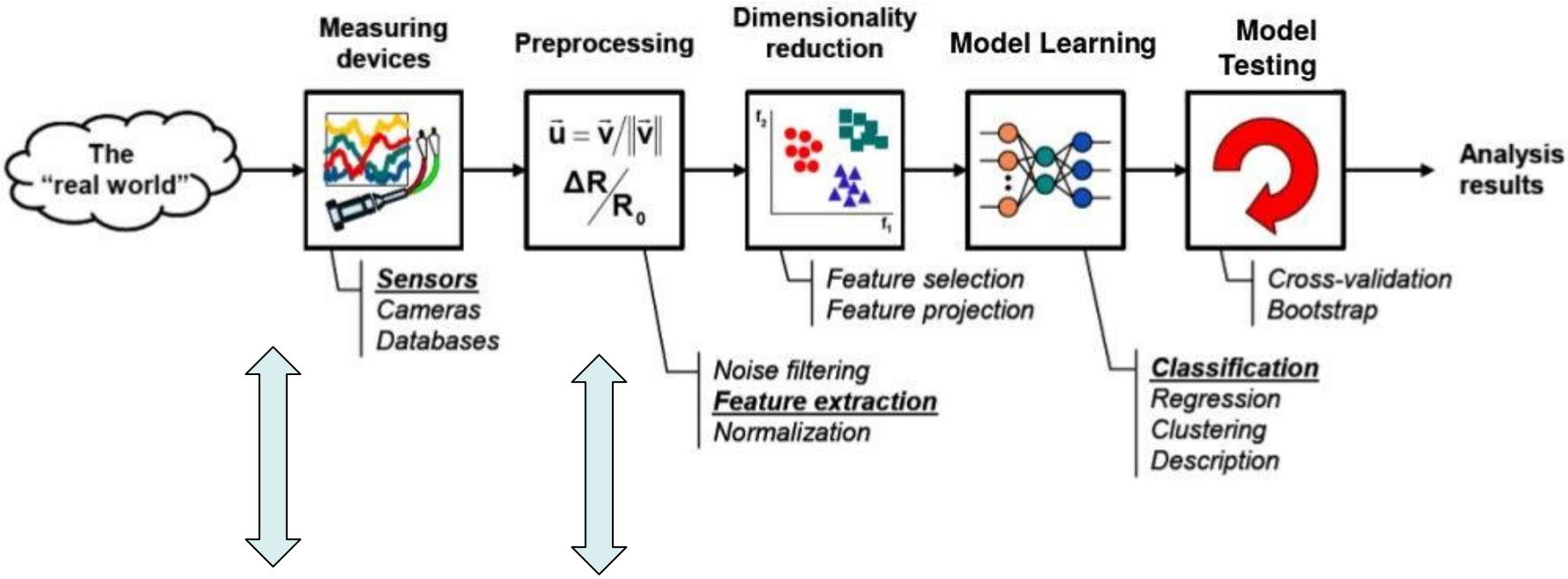
  % de emails spam que foram filtrados

  % de emails non-spam (ham) que foram incorretamente filtrados

  E: uma base de dados de emails que foram rotulados pelos usuários

O Processo de Aprendizado

O Processo de Aprendizado



Servidor de Emails

- Número de destinatários
- Tamanho da Mensagem
- Número de anexos
- Número de "re's" no assunto

A Base de Dados (*Data Set*)

Atributos

Atributo Meta

Número de novos destinatários	Tamanho do Email (kb)	País (IP)	Tipo de Cliente	Tipo de Email
0	2	Brasil	Ouro	Ok
1	4	Brasil	Prata	Ok
5	2	Argentina	Bronze	Spam
2	4	Russia	Bronze	Spam
3	4	Brasil	Bronze	Ok
0	1	EUA	Prata	Ok
4	2	EUA	Prata	Spam

Instâncias

Numéricos

Nominal

Ordinal

Aprendizado do Modelo

Base de Dados

Conjunto de Aprendizado (Treinamento)

Indutor

Algoritmo de Indução

Classificador

Modelo de Classificação

Avaliação do Modelo

Base de Dados

Conjunto de Avaliação (Teste)

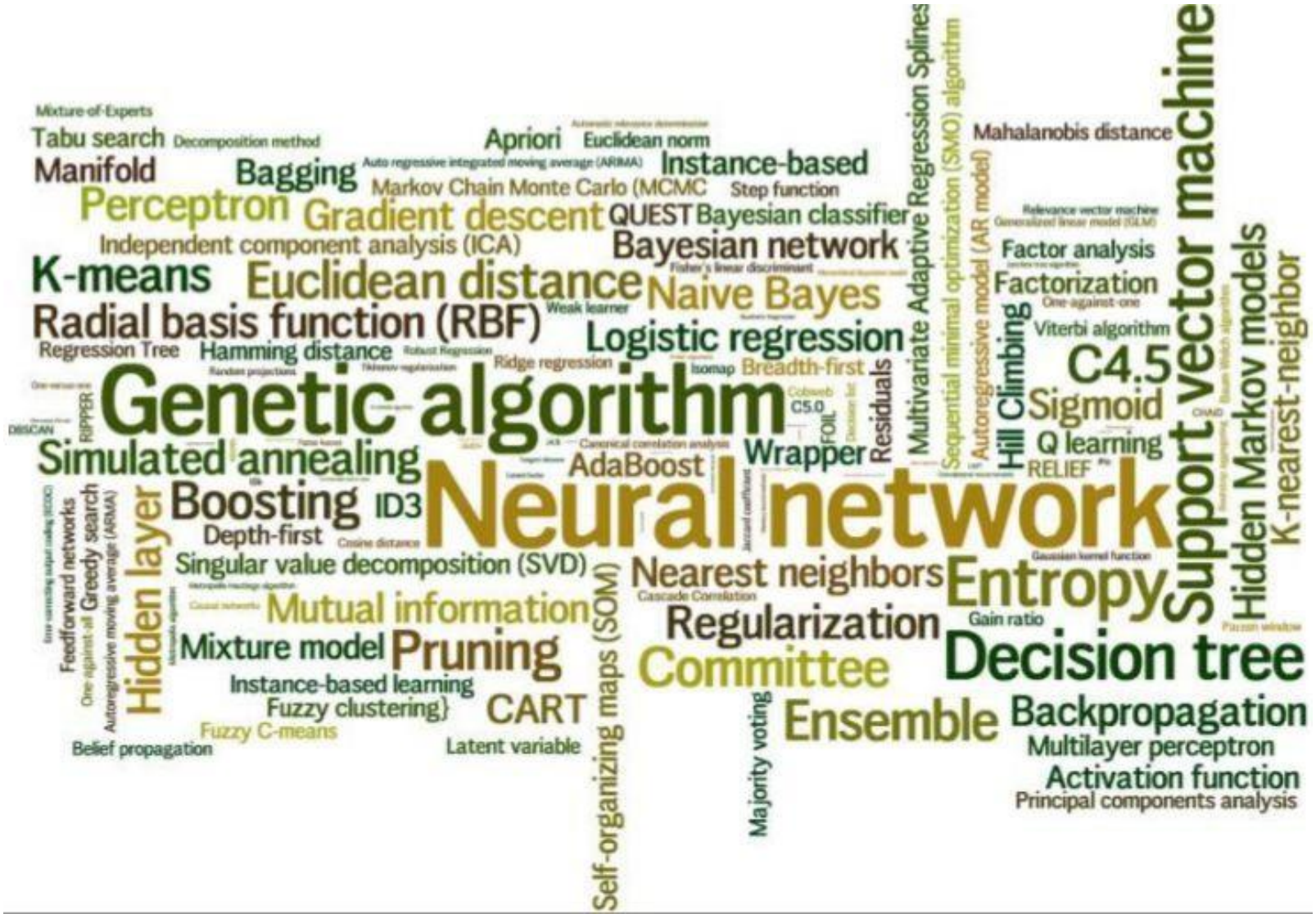
Indutor

Algoritmo de Indução

Classificador

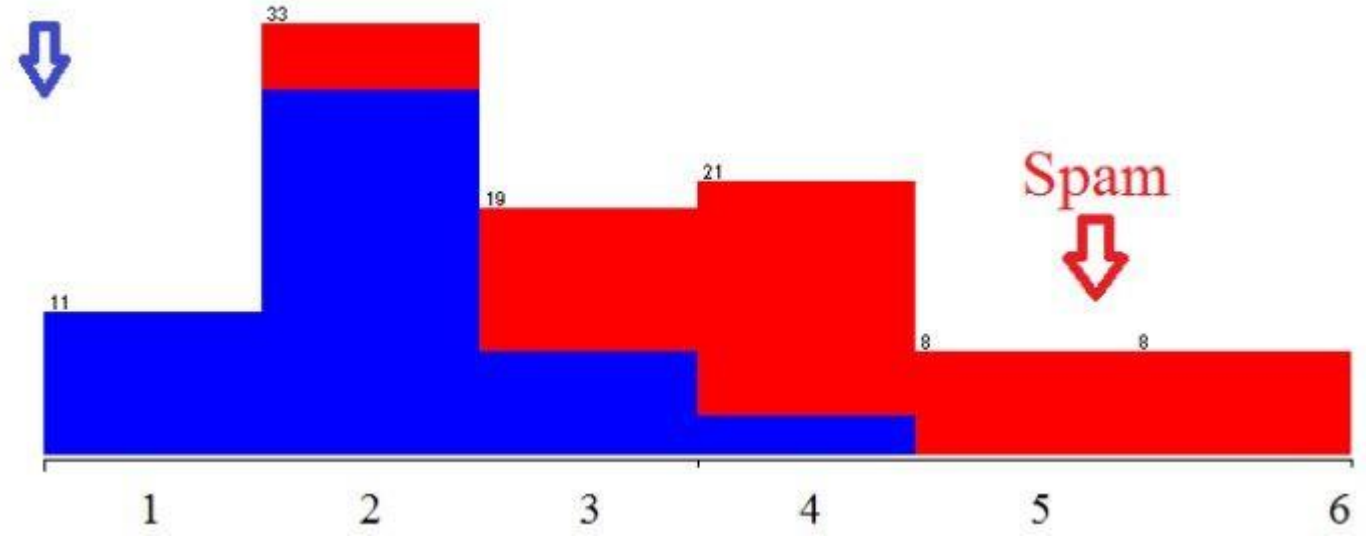
Modelo de Classificação

Algoritmos de Aprendizado



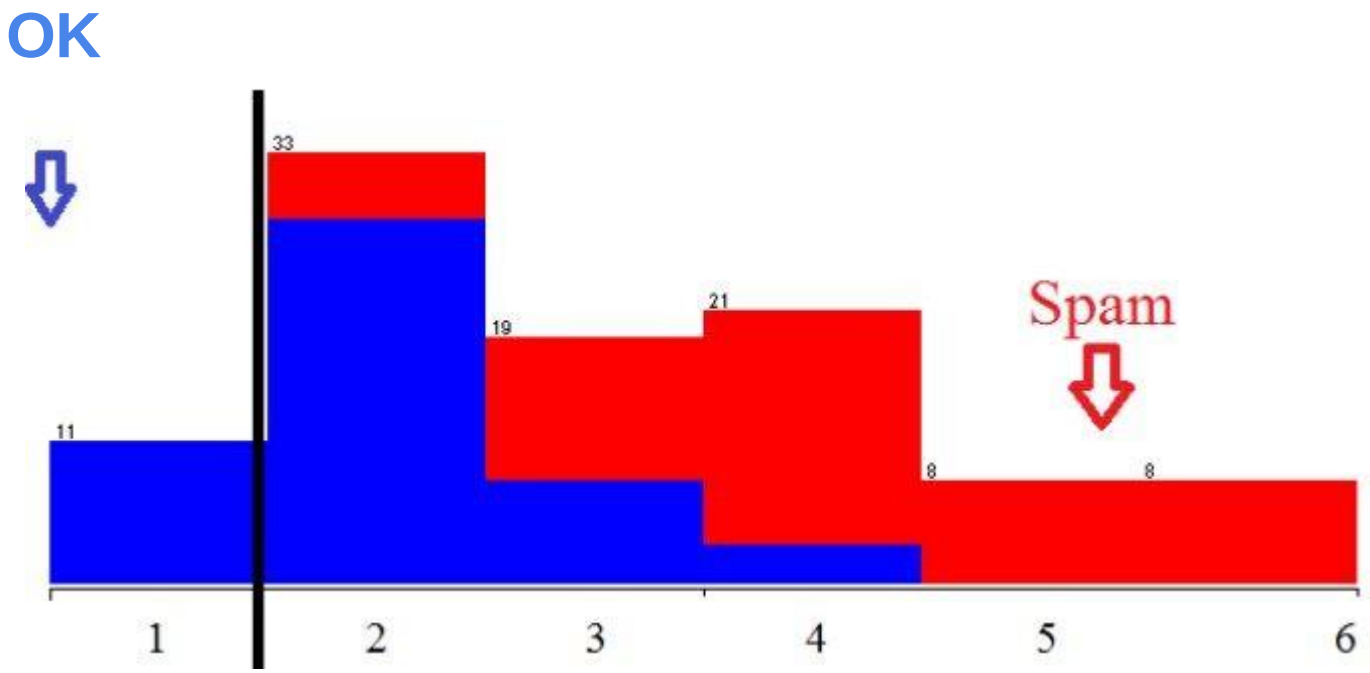
Análise da Classificação

OK



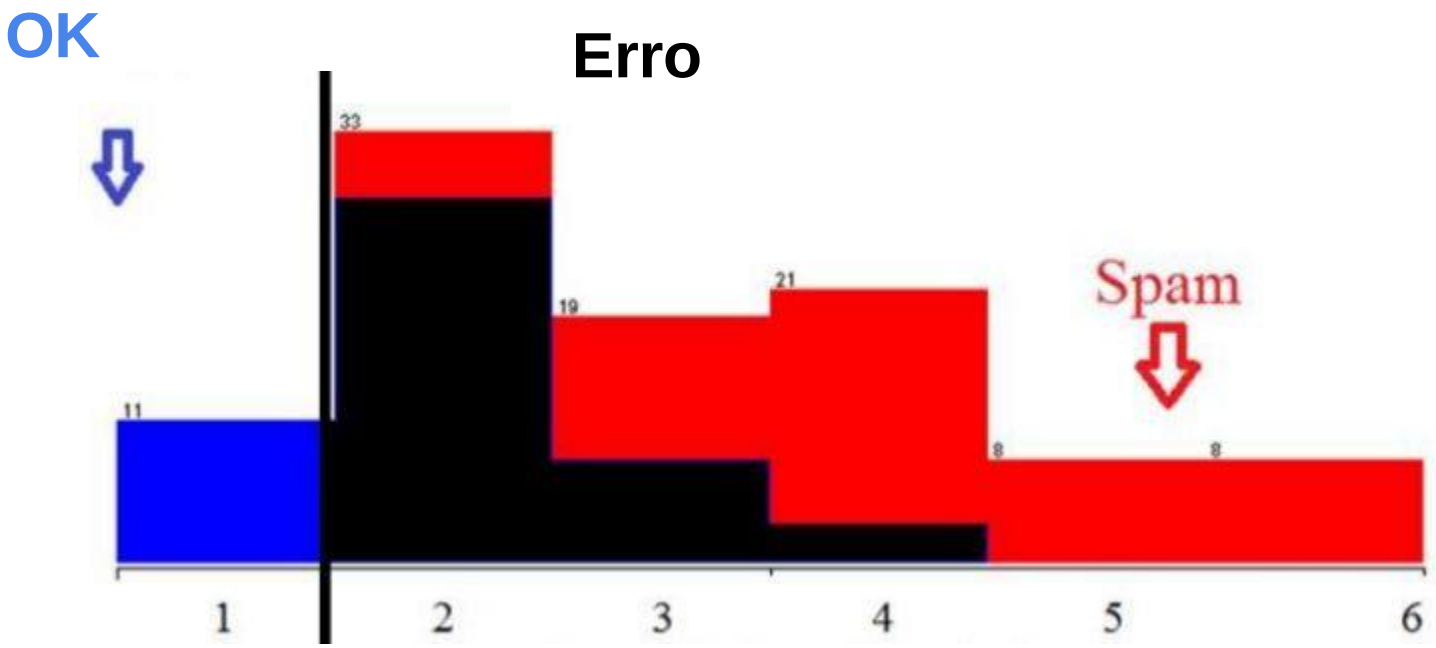
Número de novos destinatários

Análise da Classificação



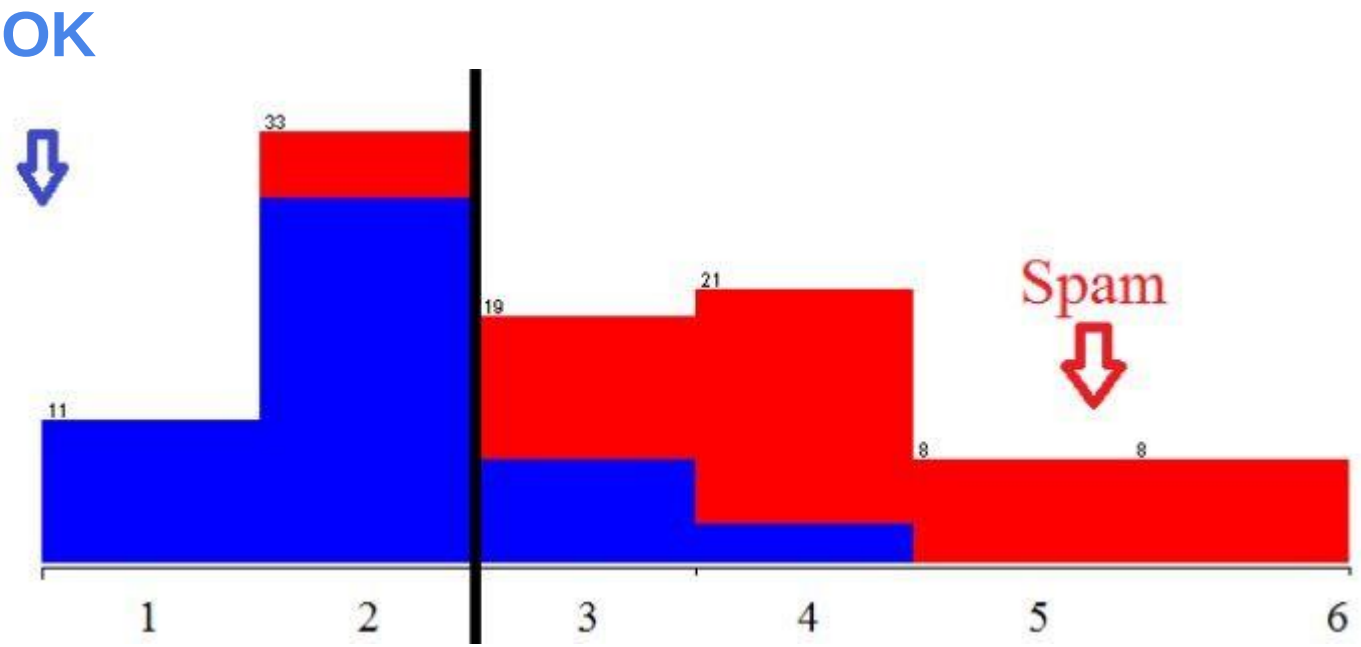
Número de novos destinatários

Análise da Classificação



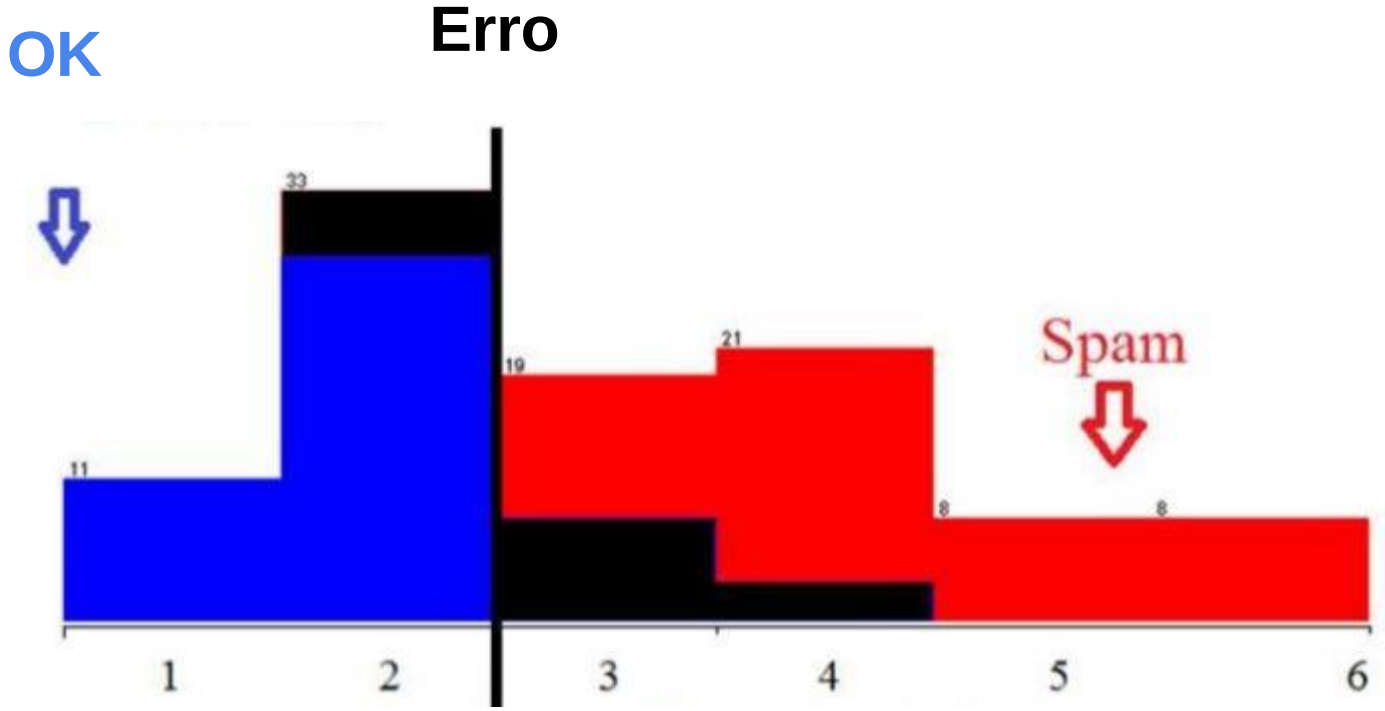
Número de novos destinatários

Análise da Classificação



Número de novos destinatários

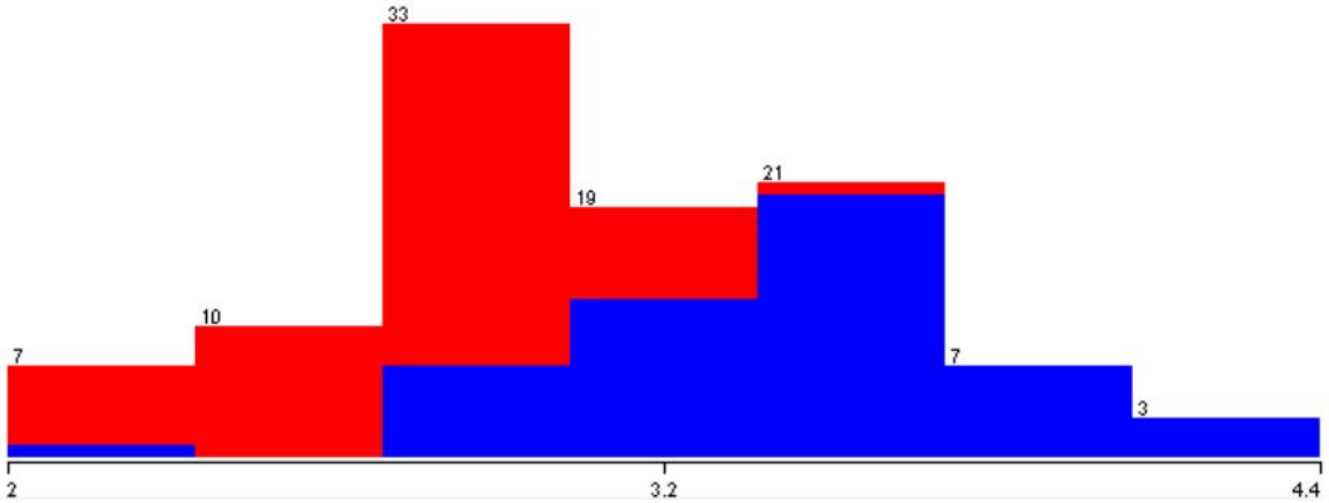
Análise da Classificação



Número de novos destinatários

Análise da Classificação

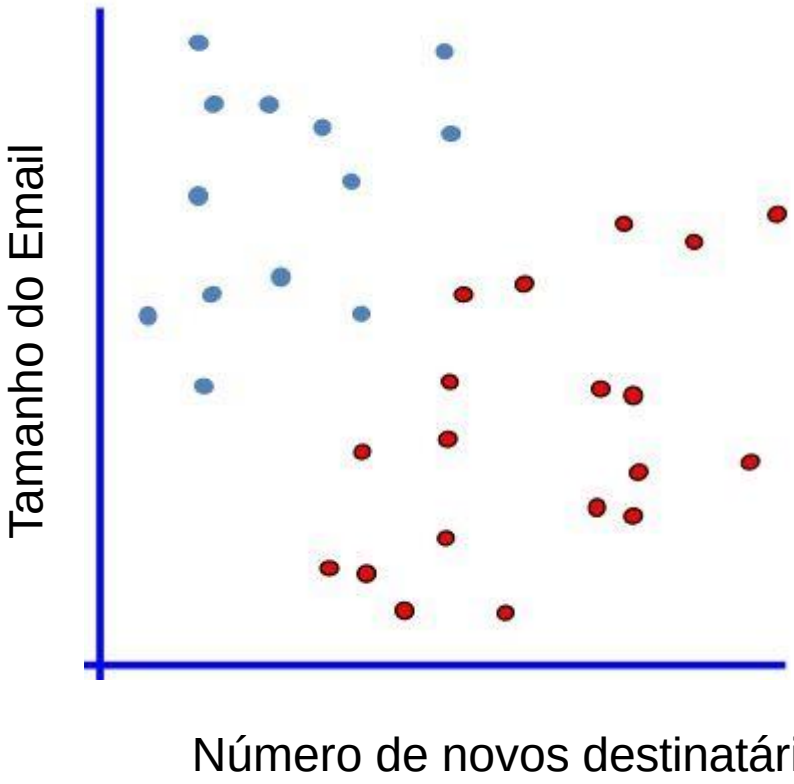
- Outro **atributo** - confusão?



Tamanho do Email (kbytes)

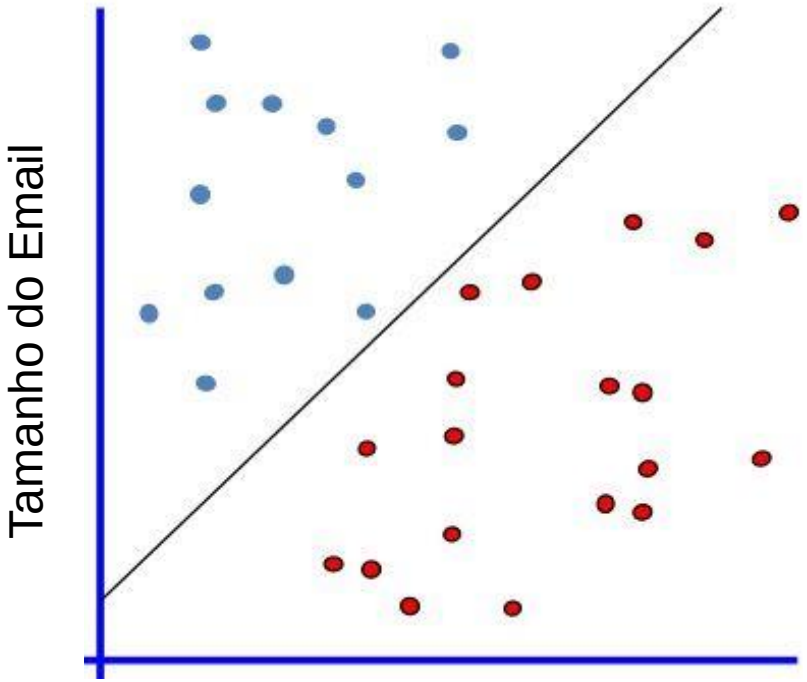
Classificadores Lineares

- Como você classificaria estes dados?



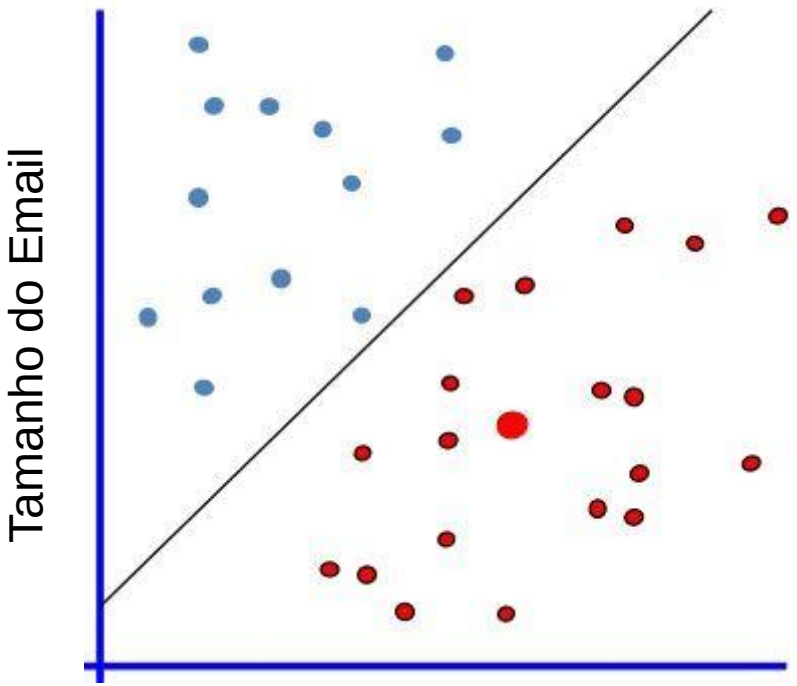
Classificadores Lineares

- Como você classificaria estes dados?



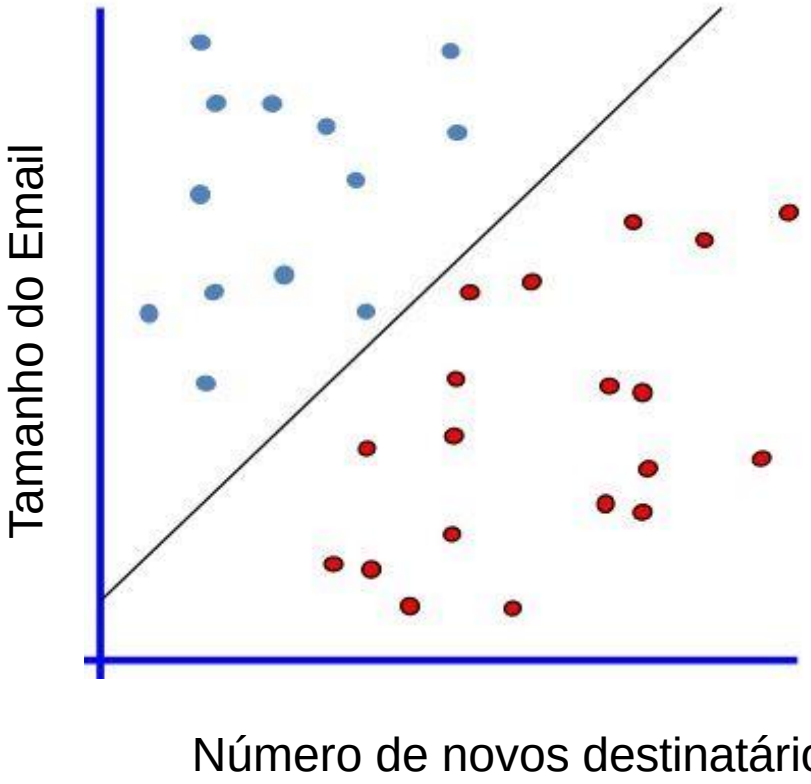
Quando um novo email é enviado

1. Coloca-se o novo email no espaço
2. Classifica-o de acordo com o subespaço no qual ele “reside”



Classificadores Lineares

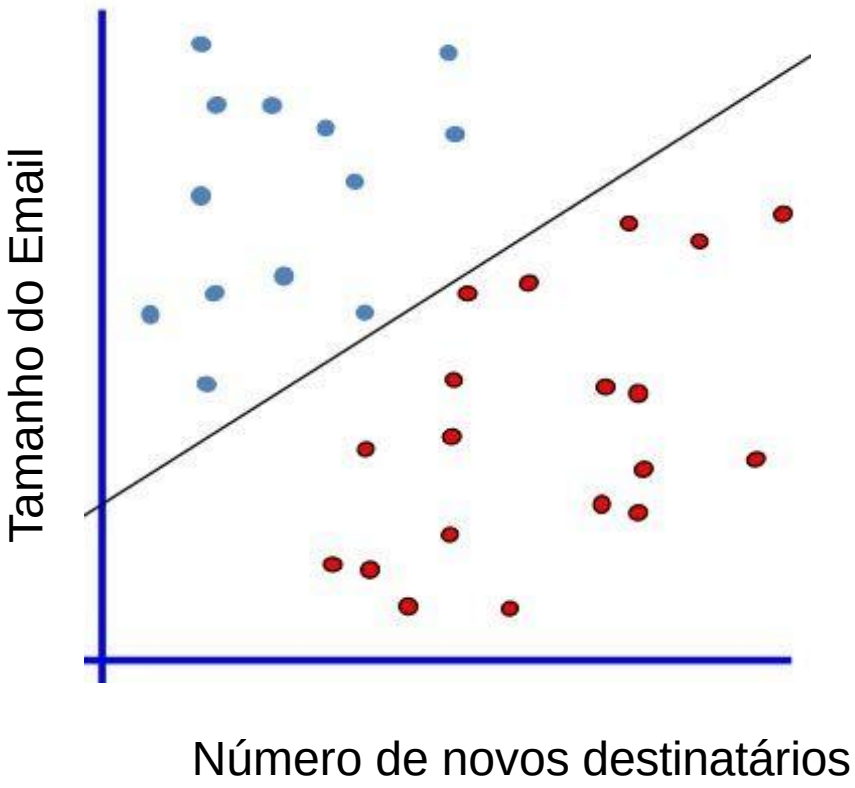
Como você classificaria estes dados?



- Várias separações são possíveis

Classificadores Lineares

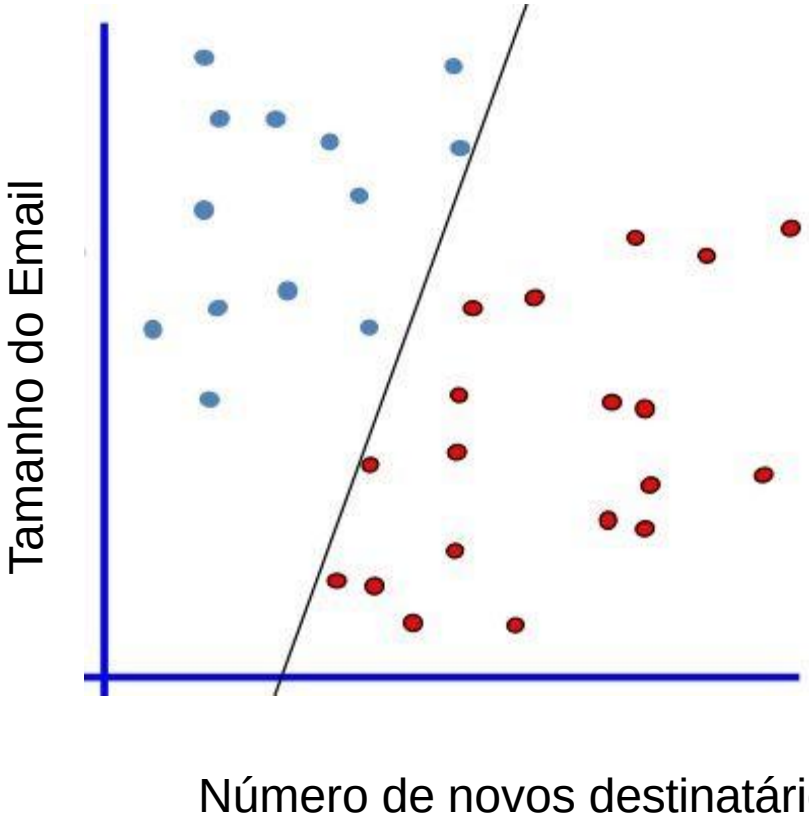
Como você classificaria estes dados?



- Várias separações são possíveis - Tamanho do Email?

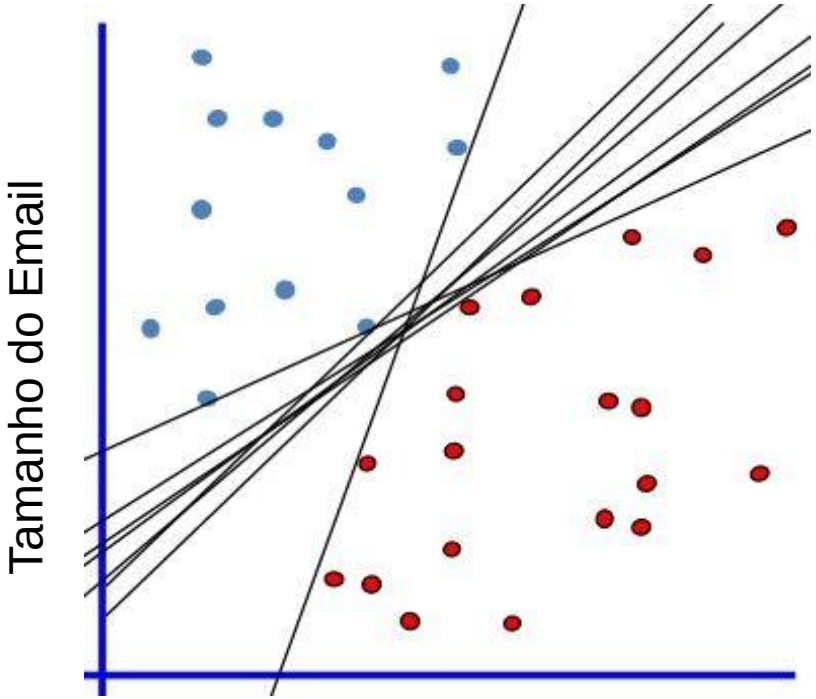
Classificadores Lineares

Como você classificaria estes dados?



- Várias separações são possíveis - Número de novos dest.?

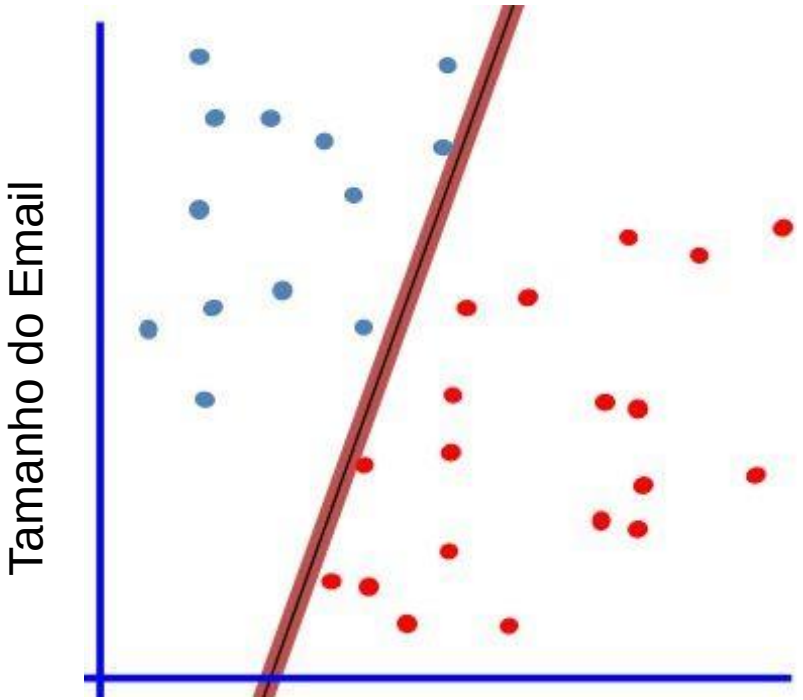
Classificadores Lineares



Qualquer uma delas seria uma boa escolha …

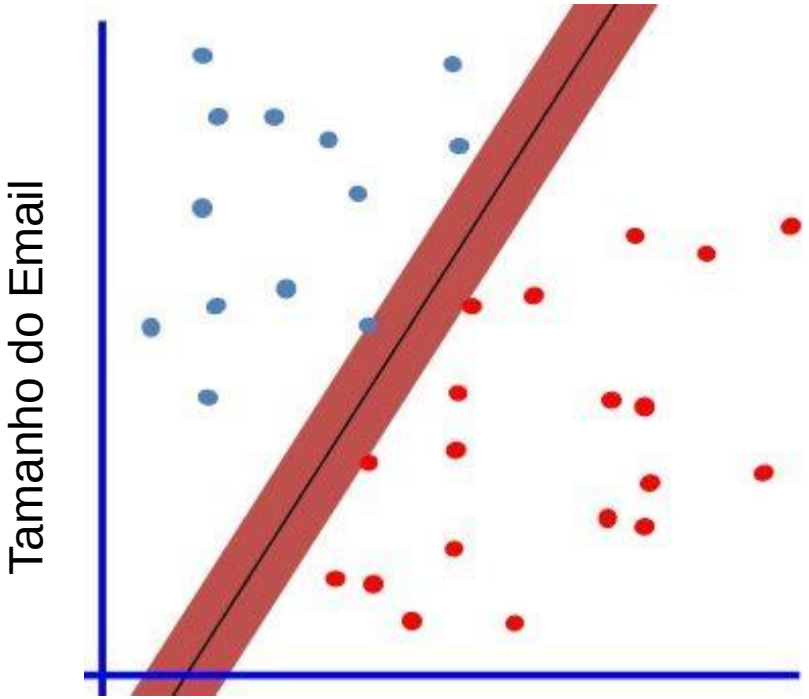
… mas qual é a melhor?

Margem Classificadora



Definir a margem de um classificador linear como a **largura** que o limite da margem pode ser aumentado antes de atingir/acertar um ponto

Margem Máxima



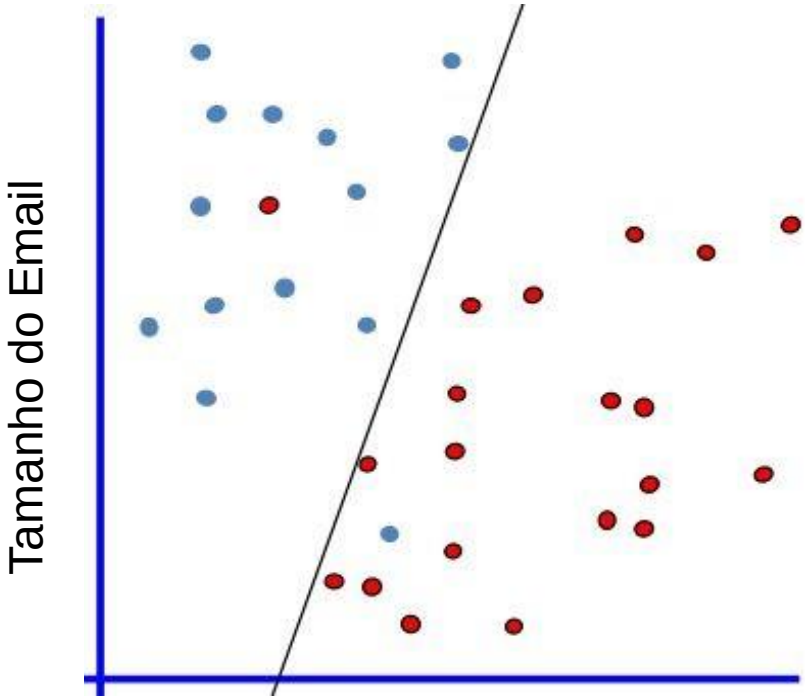
A margem máxima do classificador linear é o classificador linear com a margem máxima.

Este é o classificador mais simples do tipo SVM (Support Vector Machines) chamado de **LSVM**

Linear SVM

Nenhum Classificador Linear pode cobrir todas as instâncias

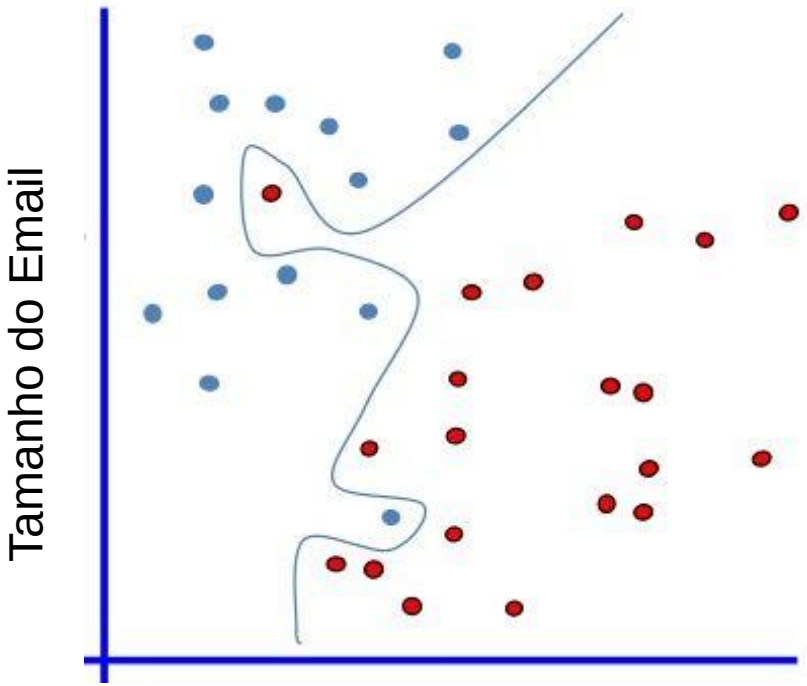
Como você classificaria estes dados?



Nenhum Classificador Linear

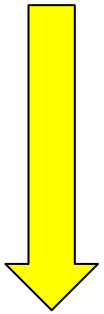
- Idealmente, a melhor fronteira de decisão deveria ser aquela que provê um desempenho ótimo tal como ...

Nenhum Classificador Linear pode cobrir todas as instâncias



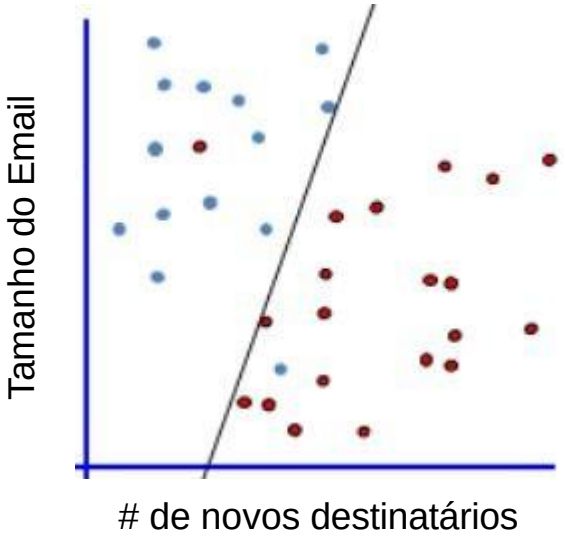
Nenhum Classificador Linear

- No evento, a satisfação imediata é prematura porque o objetivo central do desenvolvimento de um classificador é o de classificar corretamente uma nova entrada

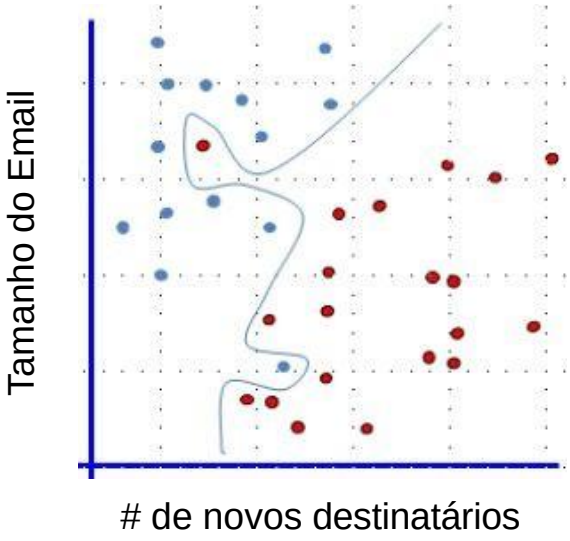


Problema de Generalização

Qual delas?

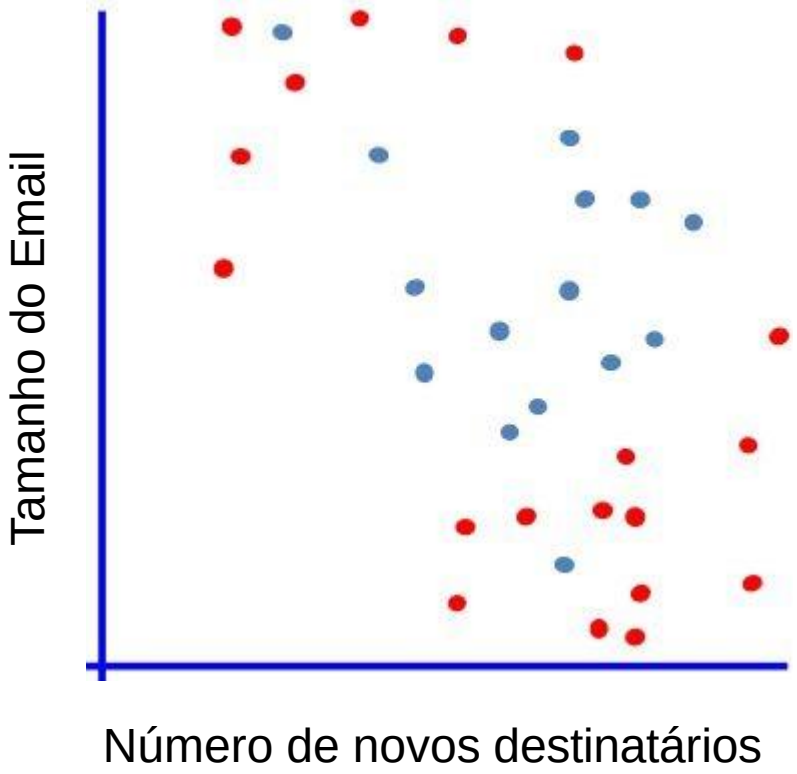


2 Erros
Modelo Simples

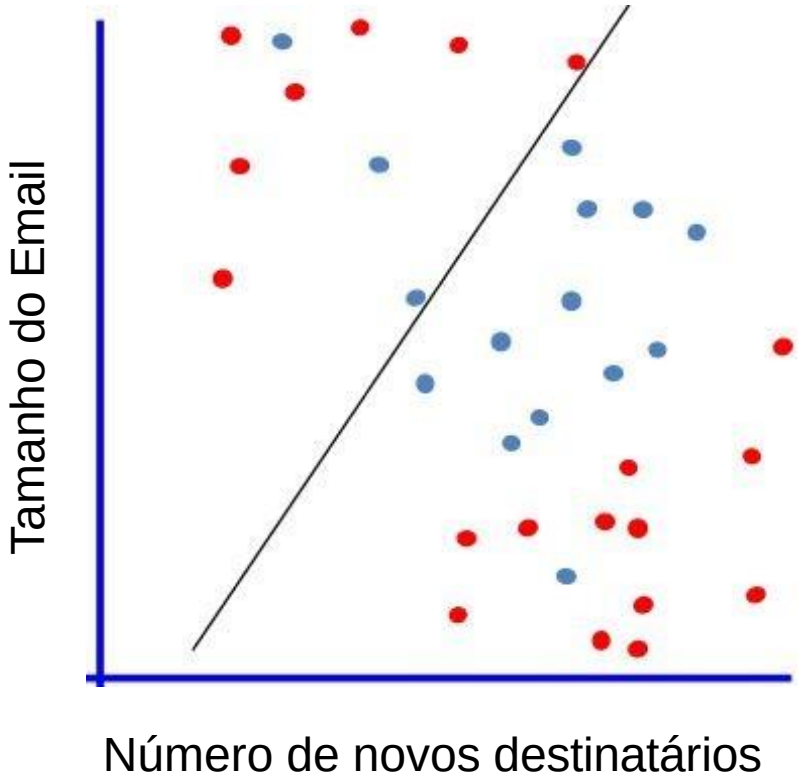


0 Erro
Modelo Complexo

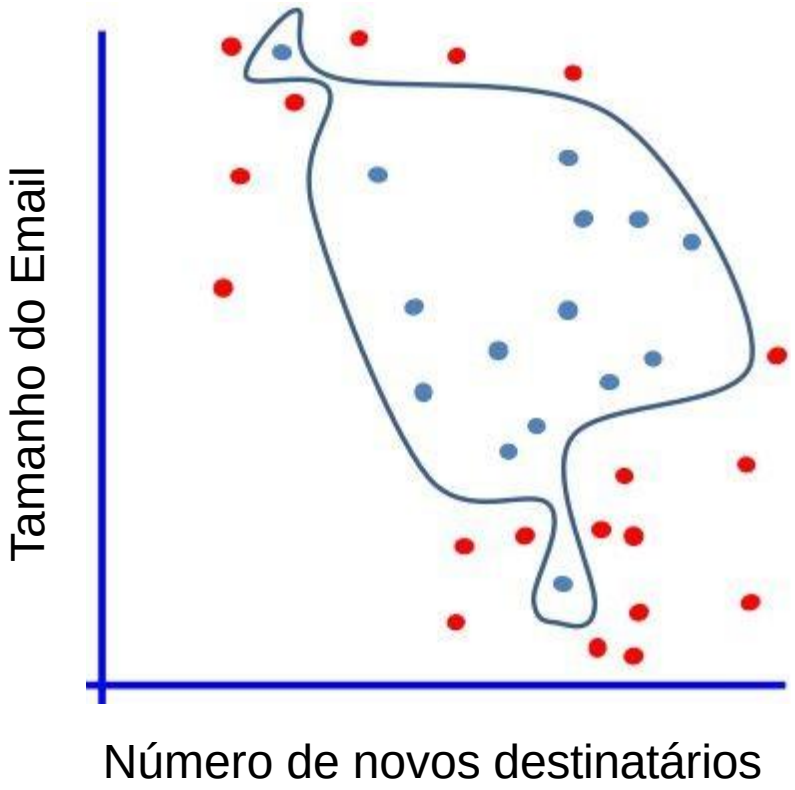
O Caso Não-linearmente Separável



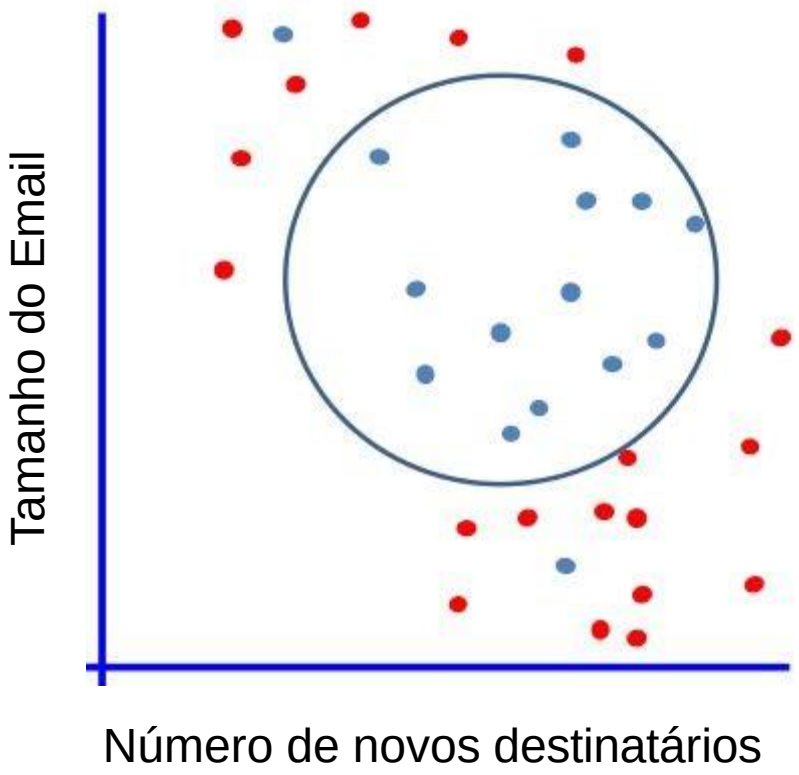
O Caso Não-linearmente Separável



O Caso Não-linearmente Separável

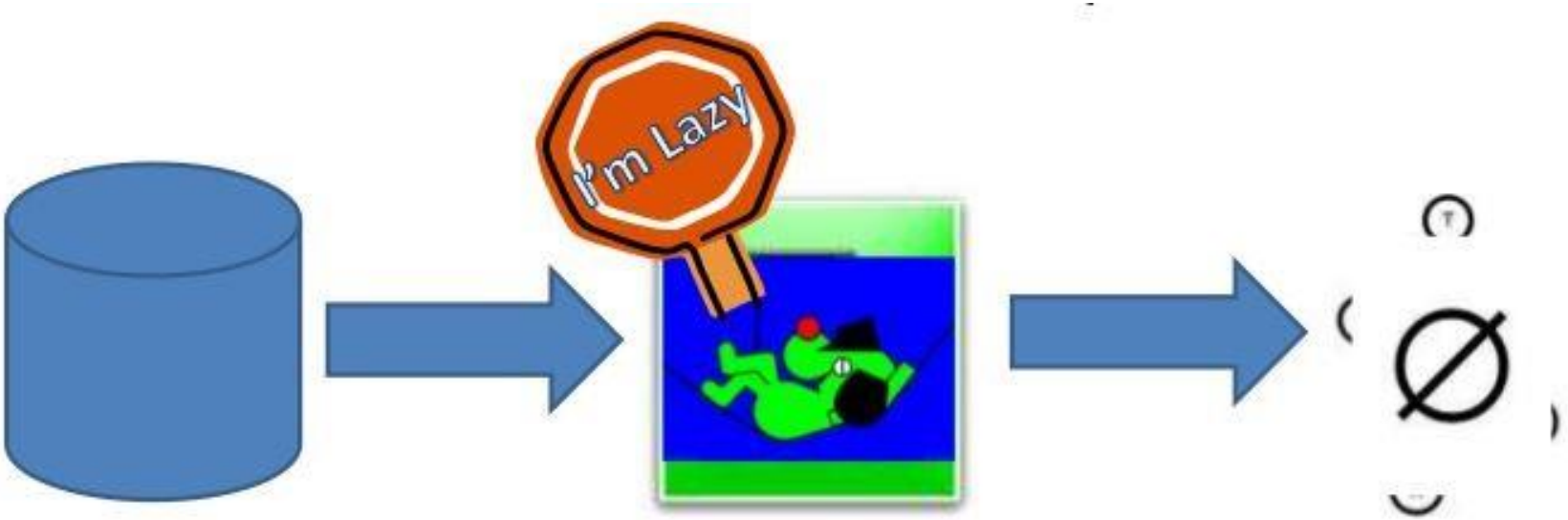


O Caso Não-linearmente Separável



Aprendizado *Lazy* (Preguiçoso)

- A generalização além do dados de treinamento é postergada até que uma nova instância é fornecida ao sistema

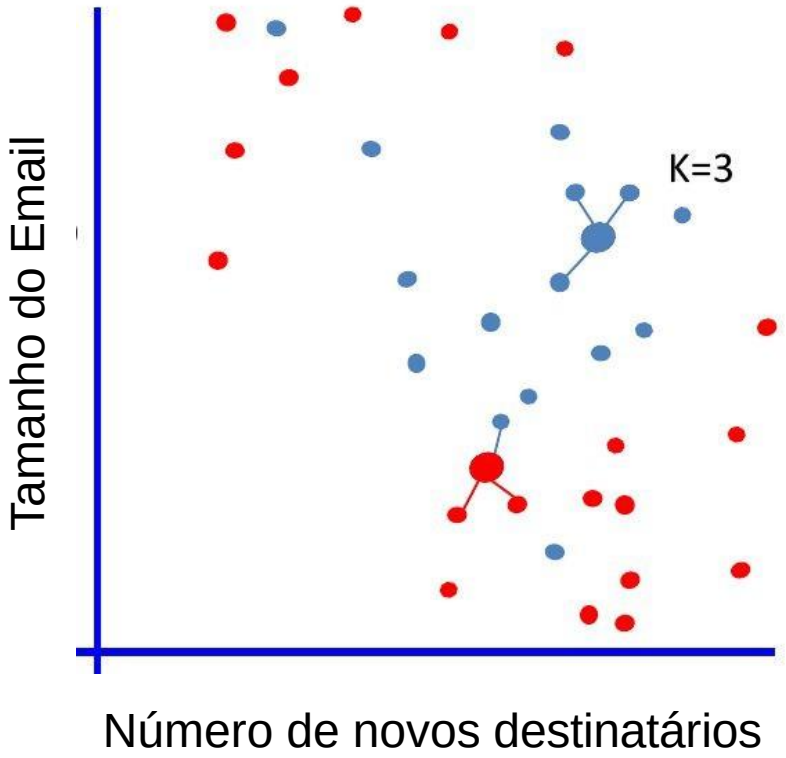


Aprendizado *Lazy*

Instance-based learning

Aprendizado *Lazy*

K-Nearest Neighbors K-Vizinhos Mais Próximos

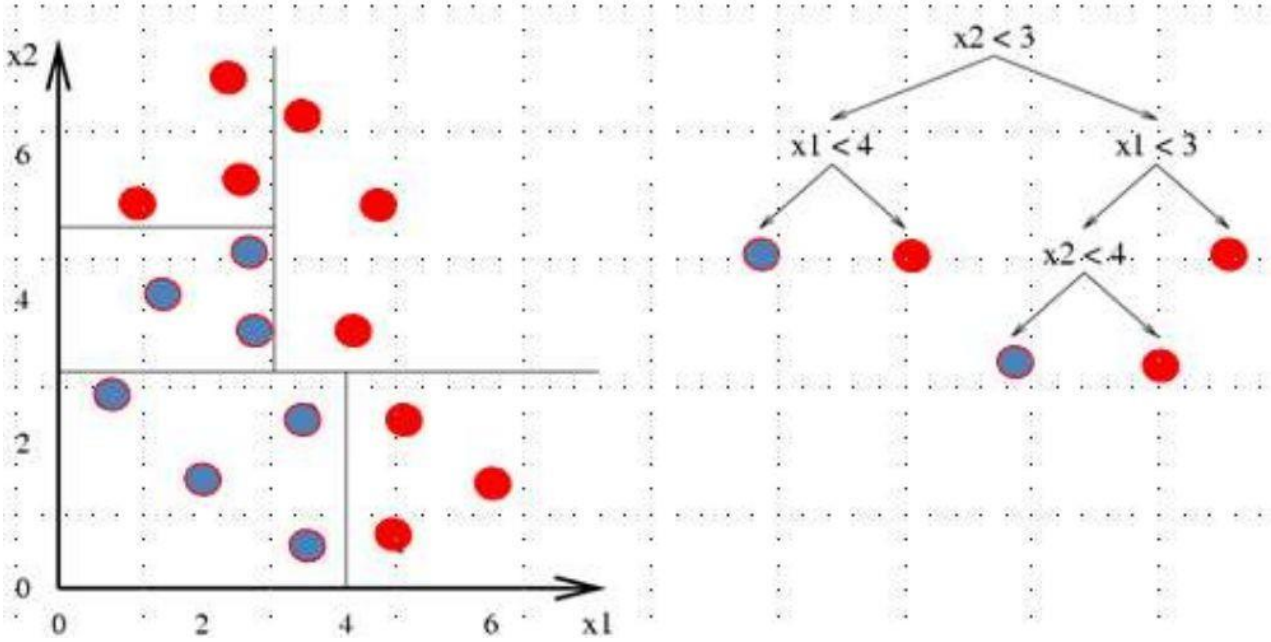


- Qual deve ser o K?
- Qual medida de distância deve ser usada?

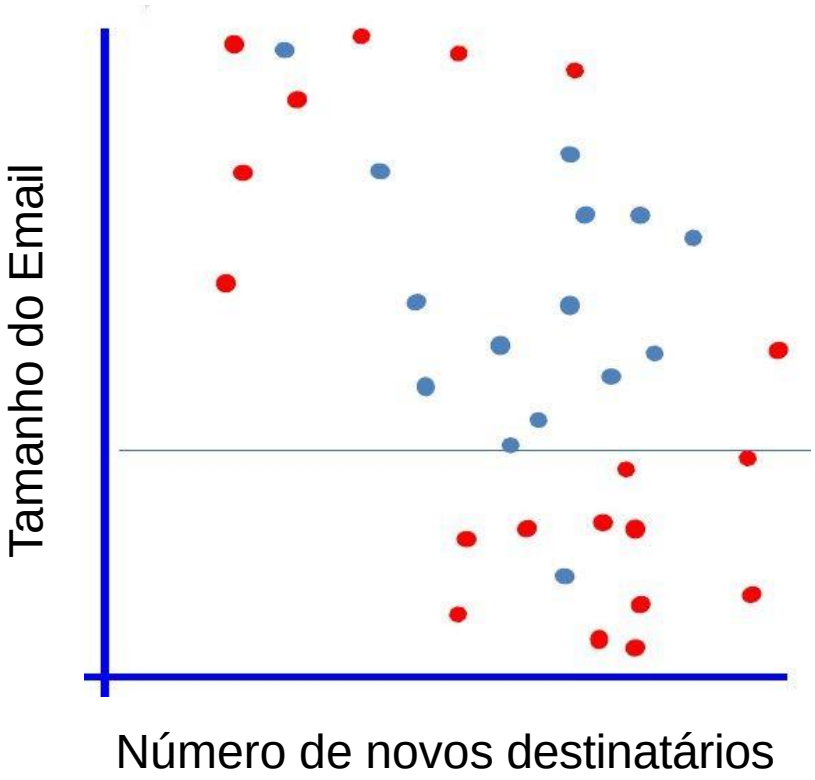
Árvores de Decisão

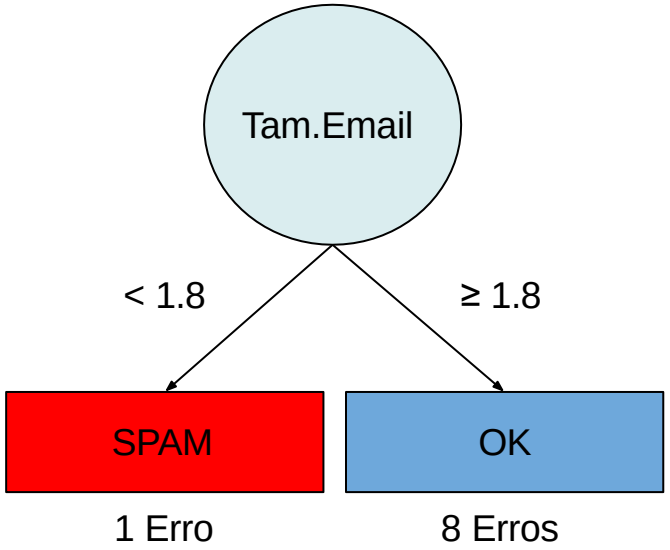
- Uma estrutura de árvore do tipo fluxograma
- Nós internos denotam uma avaliação em UM atributo
- Cada galho representa um resultado da avaliação
- Nós-folha representam uma classe/rótulo/meta

Árvores de decisão dividem o espaço de características em
eixos paralelos retangulares e rotulam cada retângulo com uma classe



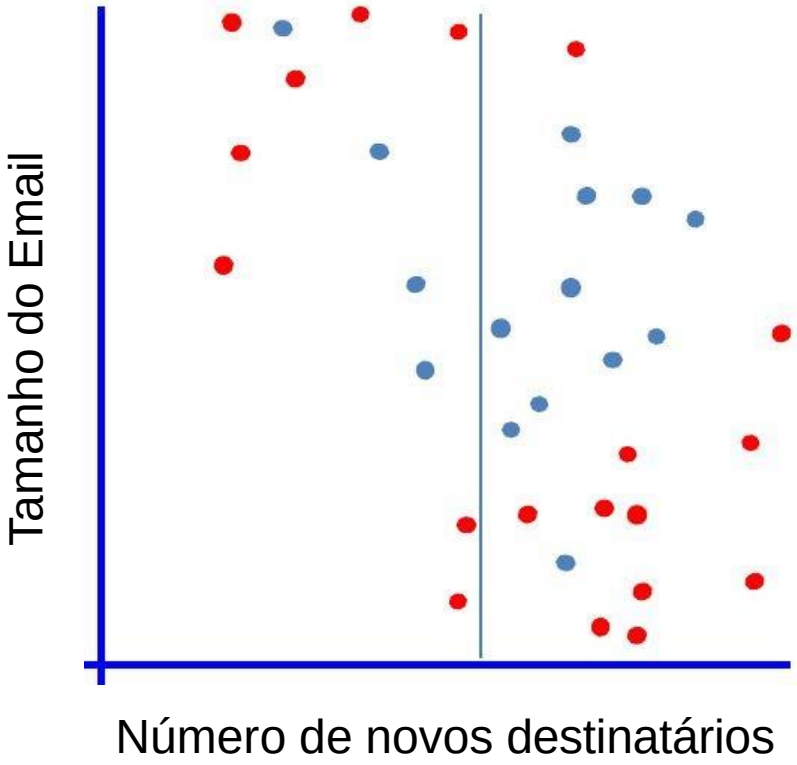
Indução *Top Down* de Árvores de Decisão

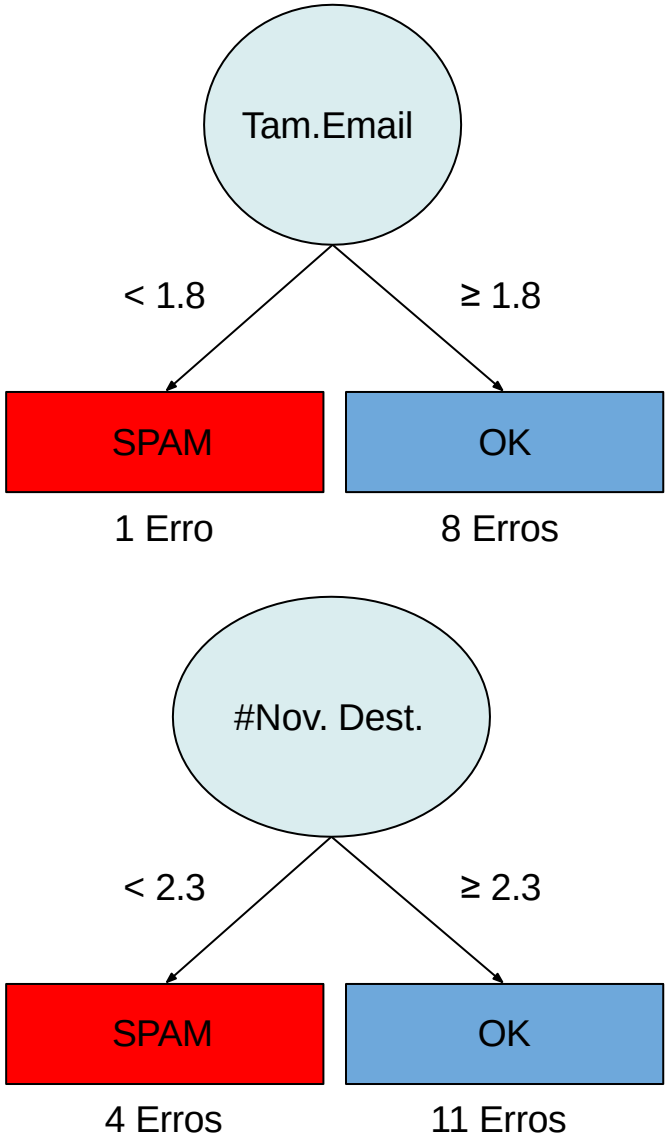




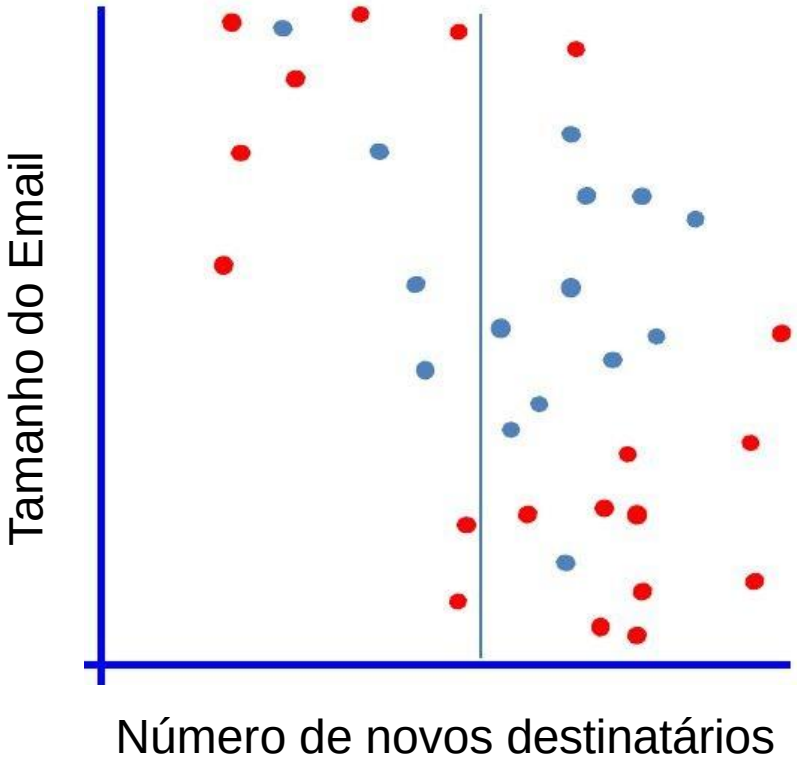
Uma árvore de decisão de um único nível é também conhecida como um Cepo/Toco-Decisão (*decision stump*)

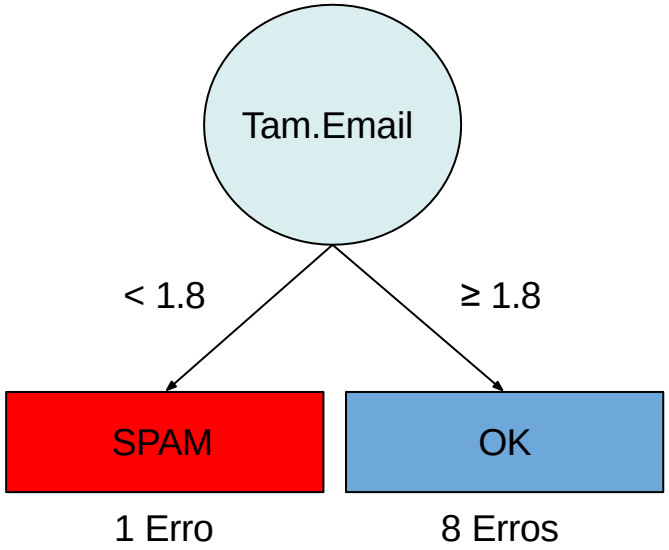
Indução *Top Down* de Árvores de Decisão

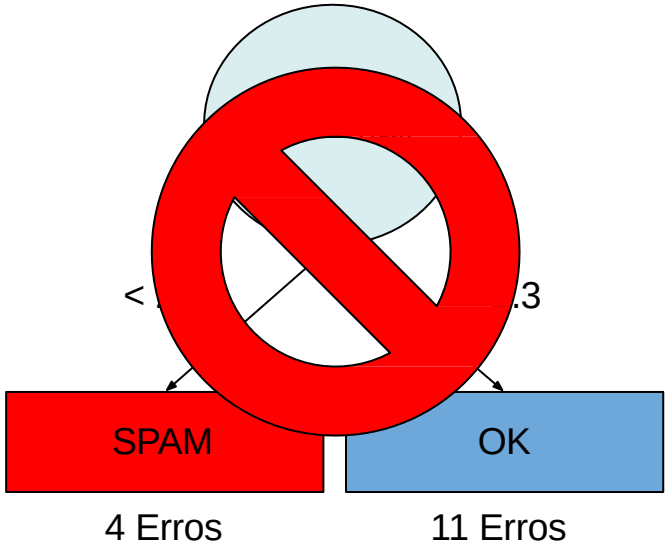




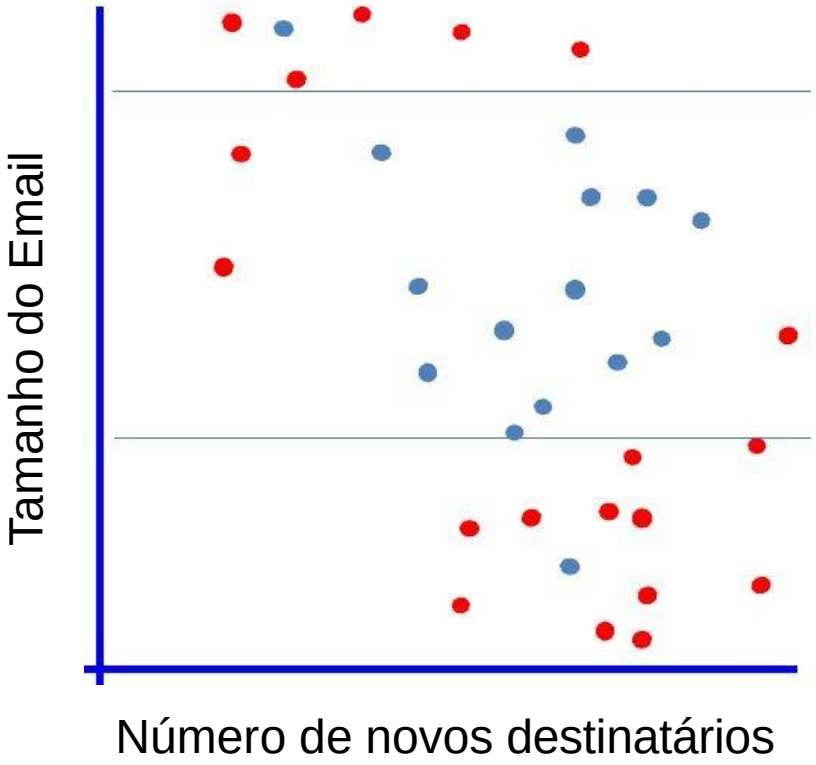
Indução *Top Down* de Árvores de Decisão

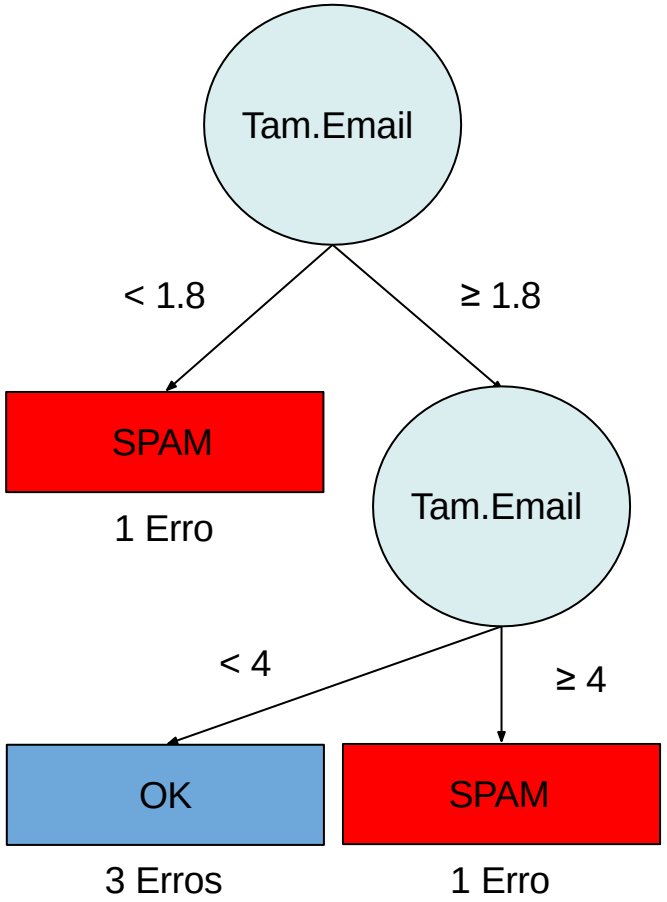




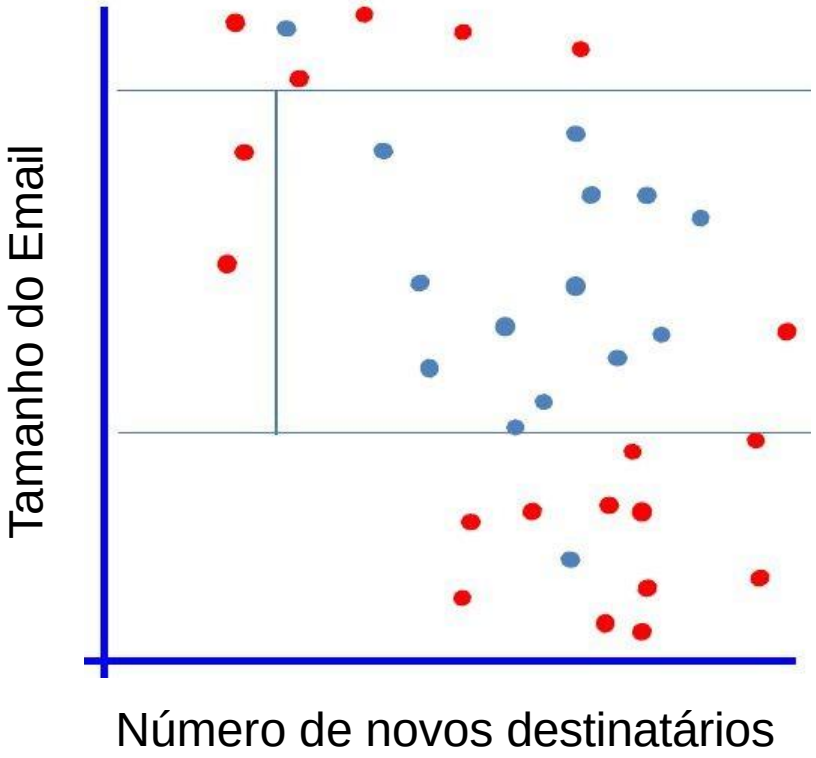


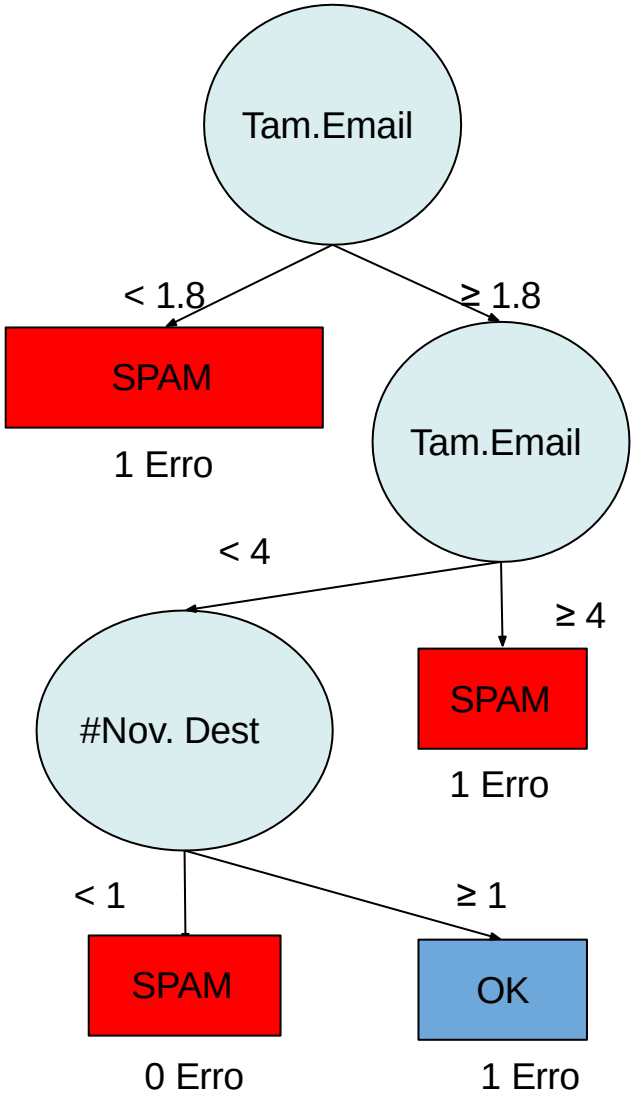
Indução *Top Down* de Árvores de Decisão



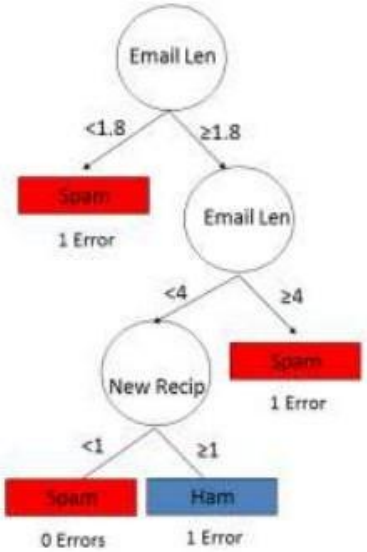


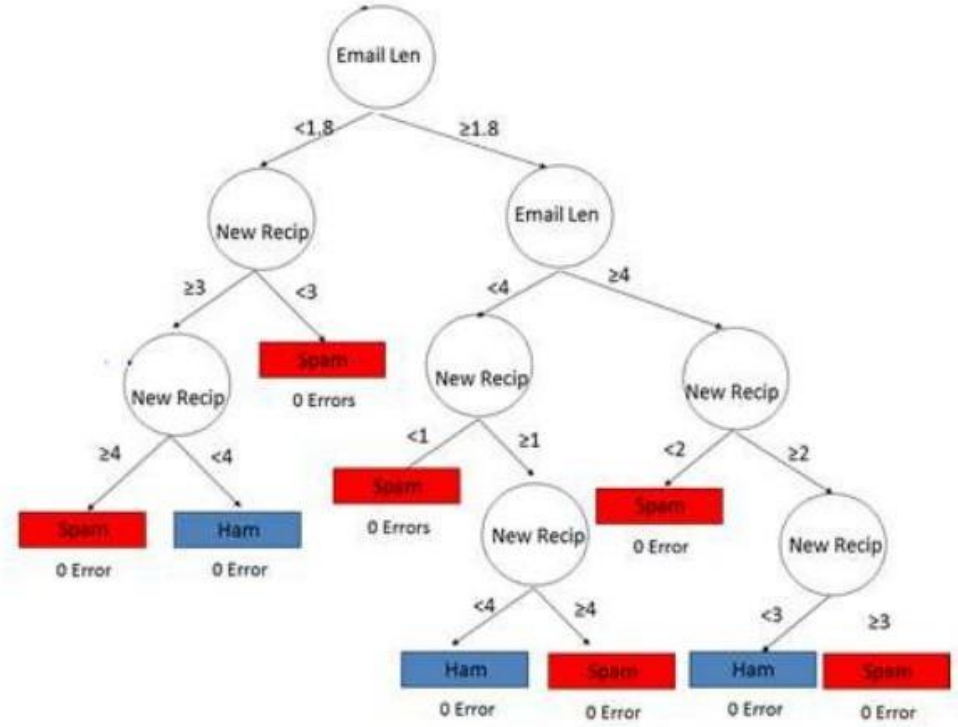
Indução *Top Down* de Árvores de Decisão

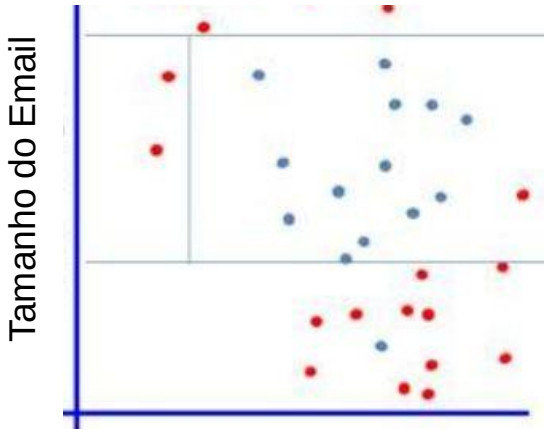


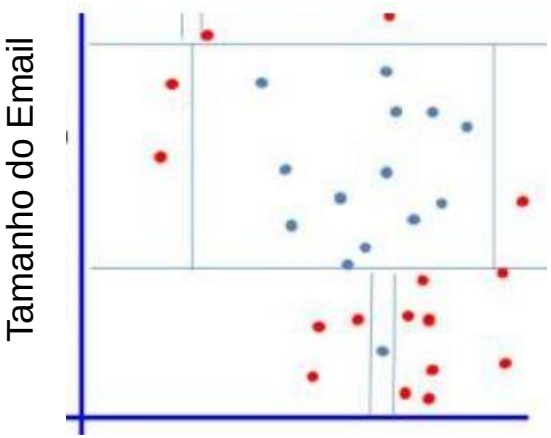


Qual Árvore?


Email Len
<1.8
≥1.8
Spam
1 Error
Email Len
<4
≥4
New Recip
Spam
1 Error
<1
≥1
Spam
0 Errors
Ham
1 Error


Email Len
<1.8
≥1.8
New Recip
Email Len
≥3
<3
<4
≥4
New Recip
Spam
0 Errors
New Recip
New Recip
≥4
<4
<1
≥1
<2
≥2
Spam
0 Error
Ham
0 Error
Spam
0 Errors
New Recip
Spam
0 Error
New Recip
<4
≥4
<3
≥3
Ham
0 Error
Spam
0 Error
Ham
0 Error
Spam
0 Error


Tamanho do Email
de novos destinatários


Tamanho do Email
de novos destinatários


Árvores de Decisão

Roteiro da Apresentação:

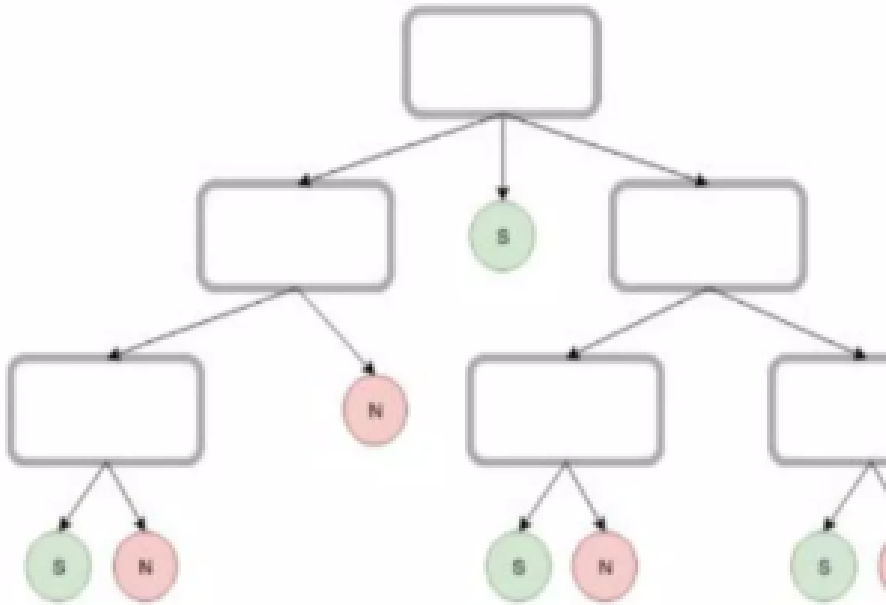
- Conceito de Árvores de Decisão;
- Representação de Árvores de Decisão;
- Características Básicas para o Uso de Árvores de Decisão;
- Cálculo da Entropia;
- Ganho de Informação;
- Razão de Ganho;
- Critério de Parada;

Árvores de Decisão

O que é?

Árvores de decisão (Decision Tree) são modelos estatísticos que utilizam um treinamento supervisionado para a classificação e previsão de dados.

Estes modelos utilizam a estratégia de dividir para conquistar (Gama, 2004).



Árvores de Decisão

- Árvores de decisão são ferramentas que podem ser utilizadas para dar ao agente a capacidade de aprender, bem como para tomar decisões.

- Os elementos do agente são usados também, para aumentar a capacidade do agente de agir no futuro;

- O aprendizado ocorre na medida em que o agente observa suas interações com o mundo e seu processo interno de tomada de decisões.

- Aprendizado de árvores de decisão é um exemplo de aprendizado indutivo;

- Árvores de decisão também podem ser representadas como um conjunto de regras SE- ENTÃO (*if-then)*;

- As árvores de decisão tomam como entrada uma situação descrita por um conjunto de atributos e retorna uma decisão.

Características Básicas para o Uso de Árvores de Decisão

- As instâncias (exemplos) são representadas por pares atributo-valor;
- A função objetivo assume apenas valores discretos;
- O conjunto de dados do treinamento pode conter erros ou valores de atributos faltando.

Representação

- Cada nó de decisão representa um ponto de decisão que irá testar atributo.
- Cada ramo descendente corresponde a um possível valor deste atributo;
- Cada nó folha está associado a uma classe;
- Cada percurso na árvore (do nó raiz a um nó folha) corresponde a uma regra de classificação.

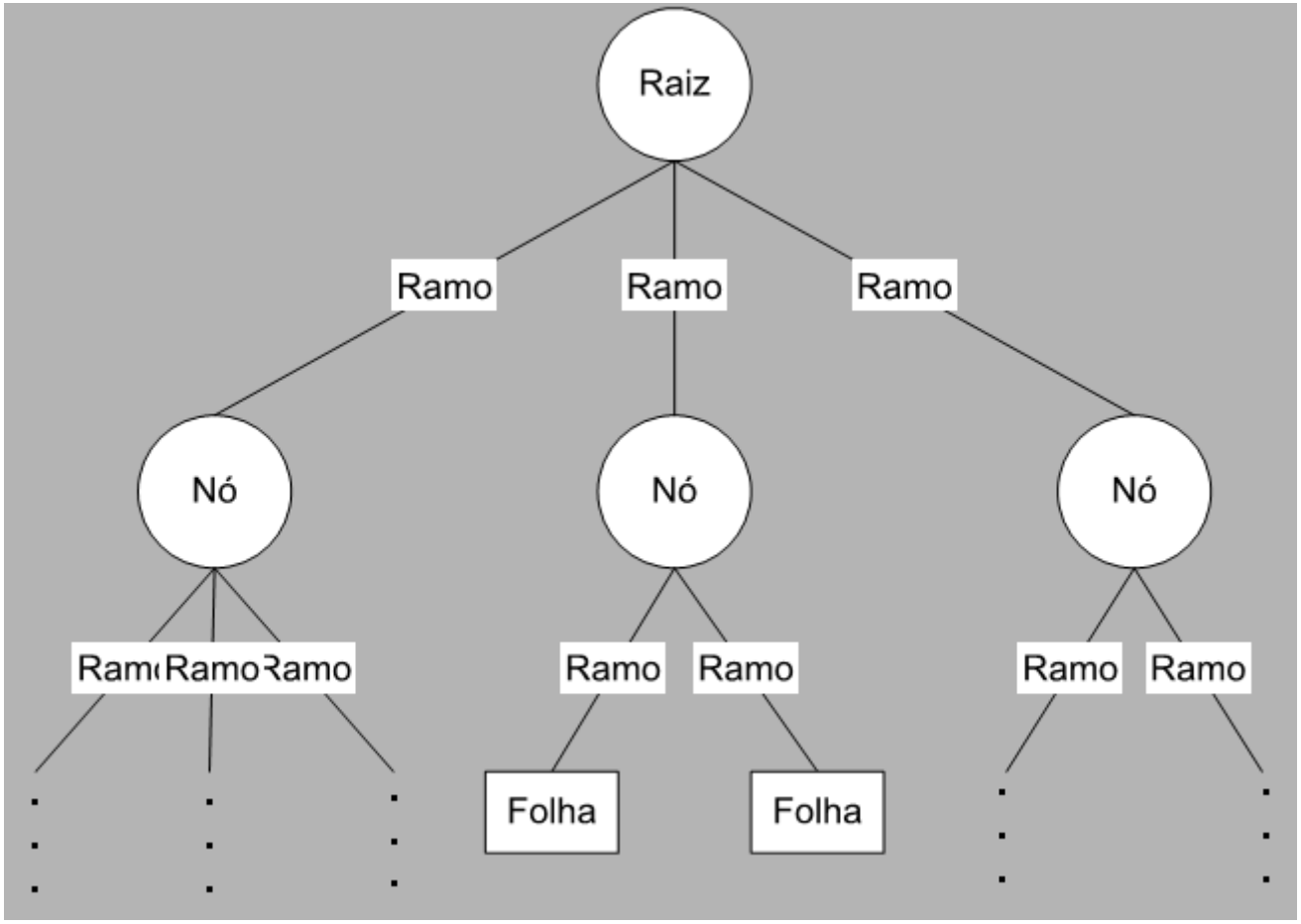


Figura 1: Representação de uma árvore de decisão.

Exemplos de Aplicações para Árvores de Decisão

- Diagnóstico Médico;
- Diagnóstico de Equipamentos;
- Análise de Crédito;
- Análise de Compra e Venda.

Características

- **Permite** comparar possíveis ações com base em seus custos, probabilidades ou benefícios;
- **Adquirem** conhecimento simbólico a partir dos dados de treinamento;
- **Retorna** valores discretos;
- **Podem** ser representadas como conjuntos de regras IF-THEN.

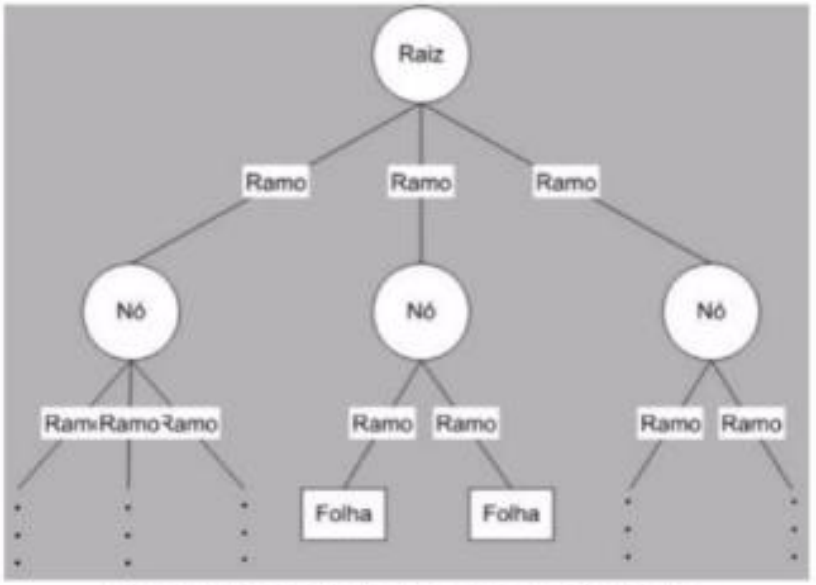


Figura 2: Representação de uma árvore de decisão

Vantagens

- **Fácil** interpretação. Percebe-se a razão da decisão;
- **Atributos** mais relevantes aparecem mais na parte superior da árvore (Entropia e Ganho de Informação);
- **Adaptável** também a problemas de regressão (Árvores de Regressão).

Desvantagens

- **Podem** se tornar complexas (verificar condição de parada);
- **Sensibilidade** a pequenas perturbações no conjunto de treino (necessário de árvore).

Entropia

A Entropia é uma medida que caracteriza a aleatoriedade (impureza) de uma coleção arbitrária de exemplos.

$$entropia\ (X) = -\sum_i p_i * \log_2 p_i$$

- A entropia tem máximo ($\log_2$ i) se $p_i = p_j$ para qualquer i <> j
- A entropia(x) = 0 se existe um *i* tal que $p_i = 1$
  - É assumido que $0 * \log_2 0 = 0$

Dado uma coleção S de exemplos + e – de um conceito alvo, a entropia de S relativa a esta classificação booleana é:

$$\text{Entropia}(S) = - p_{\oplus} \log_2 p_{\oplus} - p_{\ominus} \log_2 p_{\ominus}$$

- $p_{\oplus}$ é a proporção de exemplos positivos em S
- $p_{\ominus}$ é a proporção de exemplos negativos em S

A função Entropia relativa a uma classificação booleana, como a proporção, $p_{\oplus}$ de exemplos positivos varia entre 0 e 1.

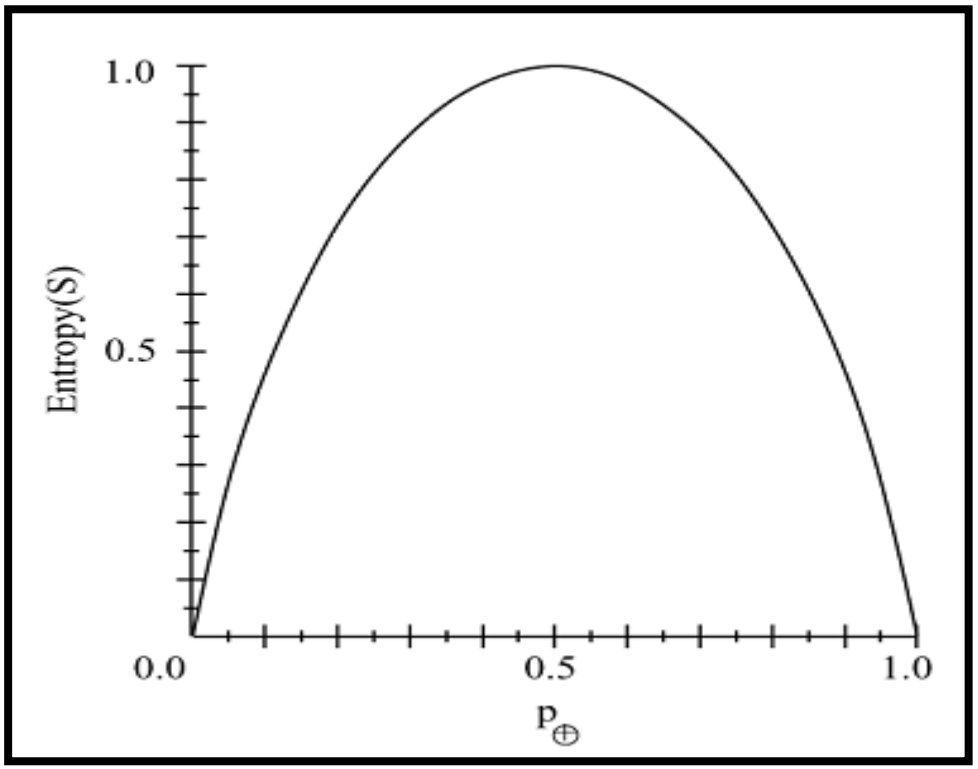
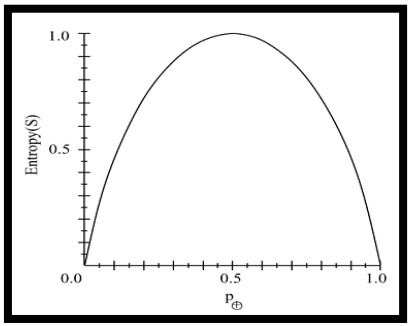


Figura 2: Representação gráfica da função Entropia.

Exemplo do Jogo do Tênis

Uma coleção S com 14 exemplos sendo 9 positivos (sim) e 5 negativos (não) [9+, 5-] o valor da entropia é:

P(+) = 9/14
P(-) = 5/14



$$Entropia\ ([9+,5-]) = -(\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14})$$
$$= 0.940$$

Conceito: “qto maior a Entropia, Maior a Desordem!!!”

Ganho de Informação

- A construção de uma árvore de decisão é guiada pelo objetivo de diminuir a entropia, ou seja, a aleatoriedade - dificuldade de previsão - da variável que define as classes.

- Ganho de Informação é a redução esperada na entropia causada pela partição dos exemplos de acordo com o teste no atributo A.

Peso = nº Amostras filho / nº Amostras pai

$$Gain\,(S,A) \equiv Entropia\ (S) - \sum_{v \in Valores\ (A)} \frac{|S_v|}{|S|} Entropia\ (S_v)$$

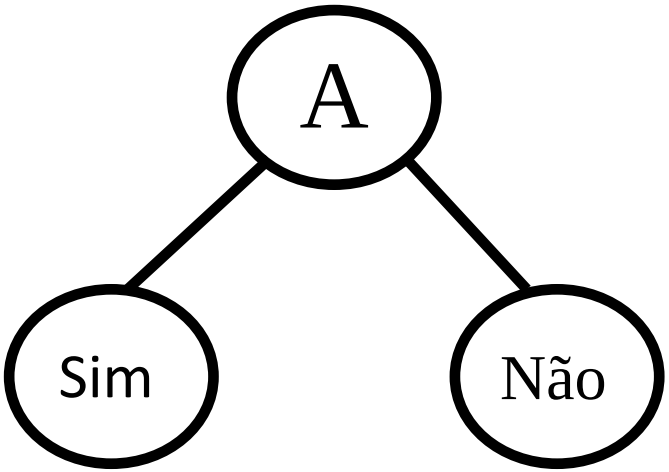


Figura 3: Exemplo de um teste no atributo A.

Para facilitar o entendimento dos conceitos que serão vistos durante a apresentação, usaremos o exemplo do Jogo de Tênis para ilustrar.

Árvore de Decisão para o **Exemplo do Jogo de Tênis**

Árvore criada para auxiliar na decisão de jogar ou não jogar tênis.

Conjunto de Atributos
Tempo
Temperatura
Umidade
Vento

Base de Treinamento do Jogo de Tênis:

Tempo	Temperatura	Umidade	Vento	Joga
Sol	Alta	Media	Não	Não
Sol	Alta	Alta	Sim	Não
Nublado	Alta	Alta	Não	Sim
Chuva	Baixa	Alta	Não	Sim
Chuva	Baixa	Media	Não	Sim
Chuva	Baixa	Baixa	Sim	Não
Nublado	Baixa	Baixa	Sim	Sim
Sol	Media	Alta	Não	Não
Sol	Baixa	Baixa	Não	Sim
Chuva	Media	Media	Não	Sim
Sol	Media	Baixa	Sim	Sim
Nublado	Media	Alta	Sim	Sim
Nublado	Alta	Baixa	Não	Sim
Chuva	Baixa	Alta	Sim	Não

Exemplo do Jogo de Tênis

Calculando o valor do Ganho de Informação para o atributo **Tempo = Sol**

Tempo	Temperatura	Umidade	Vento	Joga
Sol	Alta	Media	Não	Não
Sol	Alta	Alta	Sim	Não
Sol	Media	Alta	Não	Não
Sol	Baixa	Baixa	Não	Sim
Sol	Media	Baixa	Sim	Sim

Exemplo do Jogo de Tênis

Calculando o valor do Ganho de Informação para o atributo **Tempo = Sol**

- p (sim | tempo = sol) = 2/5
- p (não | tempo = sol) = 3/5
- Entropia (joga | tempo = sol) =

$= - (2/5) * \log_2 (2/5) - (3/5) * \log_2 (3/5) = 0.971$

Base de Treinamento do Jogo de Tênis:

Tempo	Temperatura	Umidade	Vento	Joga
Sol	Alta	Media	Não	Não
Sol	Alta	Alta	Sim	Não
Nublado	Alta	Alta	Não	Sim
Chuva	Baixa	Alta	Não	Sim
Chuva	Baixa	Media	Não	Sim
Chuva	Baixa	Baixa	Sim	Não
Nublado	Baixa	Baixa	Sim	Sim
Sol	Media	Alta	Não	Não
Sol	Baixa	Baixa	Não	Sim
Chuva	Media	Media	Não	Sim
Sol	Media	Baixa	Sim	Sim
Nublado	Media	Alta	Sim	Sim
Nublado	Alta	Baixa	Não	Sim
Chuva	Baixa	Alta	Sim	Não

Exemplo do Jogo de Tênis

Calculando o valor do Ganho de Informação para o atributo **Tempo = Nublado**

Tempo	Temperatura	Umidade	Vento	Joga
Nublado	Alta	Alta	Não	Sim
Nublado	Baixa	Baixa	Sim	Sim
Nublado	Media	Alta	Sim	Sim
Nublado	Alta	Baixa	Não	Sim

Exemplo do Jogo de Tênis

Calculando o valor do Ganho de Informação para o atributo **Tempo = Nublado**:

- p (sim | tempo = Nublado) = 1
- p (não | tempo = Nublado) = 0
- Entropia (joga | tempo = nublado) =

$= - (1/5) * \log_2 (1/5) - 0 * \log_2 (0) = 0$

Base de Treinamento do Jogo de Tênis:

Tempo	Temperatura	Umidade	Vento	Joga
Sol	Alta	Media	Não	Não
Sol	Alta	Alta	Sim	Não
Nublado	Alta	Alta	Não	Sim
Chuva	Baixa	Alta	Não	Sim
Chuva	Baixa	Media	Não	Sim
Chuva	Baixa	Baixa	Sim	Não
Nublado	Baixa	Baixa	Sim	Sim
Sol	Media	Alta	Não	Não
Sol	Baixa	Baixa	Não	Sim
Chuva	Media	Media	Não	Sim
Sol	Media	Baixa	Sim	Sim
Nublado	Media	Alta	Sim	Sim
Nublado	Alta	Baixa	Não	Sim
Chuva	Baixa	Alta	Sim	Não

Exemplo do Jogo de Tênis

Calculando o valor do Ganho de Informação para o atributo **Tempo = Chuva**

Tempo	Temperatura	Umidade	Vento	Joga
Chuva	Baixa	Alta	Não	Sim
Chuva	Baixa	Media	Não	Sim
Chuva	Baixa	Baixa	Sim	Não
Chuva	Media	Media	Não	Sim
Chuva	Baixa	Alta	Sim	Não

Exemplo do Jogo de Tênis

Calculando o valor do Ganho de Informação para o atributo **Tempo = Chuva**:

- p (sim | tempo = Chuva) = 3/5
- p (não | tempo = Chuva) = 2/5
- Entropia (joga | tempo = chuva) =

$= - (3/5) * \log_2 (3/5) - (2/5) * \log_2 (2/5) = 0{,}971$

Calculo Peso

Calculando o valor do peso

Tempo = Sol

- peso sol = 5/14
- peso nublado = 4/14
- peso chuva = 5/14

Exemplo do Jogo de Tênis

Ganho de Informação obtida neste atributo:

- Informação (tempo) =
  = 5/14 * 0.971 + **4/14 * 0** + 5/14 * 0.971 = 0.693
- Ganho (S, Tempo) = Entropia (joga) – Informação (tempo)
- **Ganho (S, Tempo) = 0.940 – 0.693 = 0.247**

Peso = nº Amostras filho / nº Amostras pai

$$Gain(S,A) \equiv Entropia\ (S) - \sum_{v \in Valores\ (A)} \frac{|S_v|}{|S|} Entropia\ (S_v)$$

Exemplo do Jogo do Tênis - Façam os demais calculos

Calculando o valor de Ganho de Informação para todos os outros atributos temos:

- Ganho (S, Umidade) = 0,057
- Ganho (S, Vento) = 0,048
- Ganho (S, Temperatura) = 0,029
- Ganho (S, Tempo) = 0,247

Implicações do Cálculo de Ganho de Informação

- O critério de ganho seleciona como atributo-teste aquele que maximiza o ganho de informação;

- O grande problema ao se utilizar o ganho de informação é que ele dá preferência a atributos com muitos valores possíveis;

- Um exemplo claro desse problema ocorreria ao utilizar um atributo totalmente irrelevante;

- Nesse caso, seria criado um nó para cada valor possível, e o número de nós seria igual ao número de identificadores;

- Essa divisão geraria um ganho máximo, embora seja totalmente inútil;

Razão de Ganho

$$\text{Razão de Ganho} = \frac{\text{Ganho}}{\text{Entropia (nó)}}$$

- A Razão de Ganho é o ganho de informação relativo (ponderado) como critério de avaliação;

- A razão não é definida quando o denominador é igual a zero, ou seja, quando o valor da entropia do nó é zero;

- Além disso, a razão de ganho favorece atributos cujo o valor da entropia é pequeno.

Exemplo do Jogo do Tênis

Calculando o valor da Razão de Ganho para o atributo do tempo temos:

$$\text{RazãoGanho (Tempo)} = \frac{\text{Ganho (S, Tempo)}}{\text{Entropia}}$$

$$= \frac{0{,}247}{0{,}940}$$

$$= 0{,}263$$

Exemplo do Jogo do Tênis

Calculando o valor de Ganho de Informação para todos os outros atributos temos:

Razão de Ganho (tempo) = 0,263
Razão de Ganho (umidade) = 0,06
Razão de Ganho (vento) = 0,051
Razão de Ganho (temperatura) = 0,031

Exemplo do Jogo do Tênis

Atributo	Ganho de Informação	Razão de Ganho
Tempo	0,247	0,263
Temperatura	0,029	0,031
Umidade	0,057	0,06
Vento	0,048	0,051

Qual é o melhor atributo para ser selecionado como a raiz da árvore?

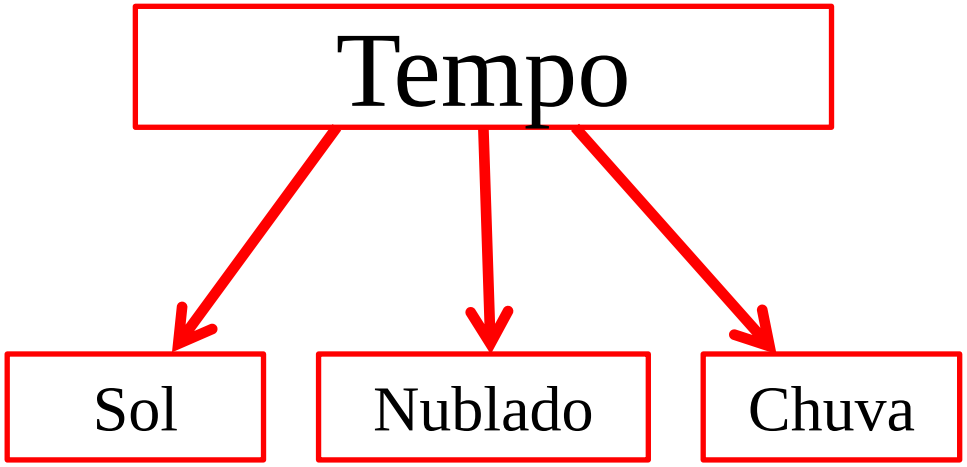
Exemplo do Jogo do Tênis

S = [9+, 5-]

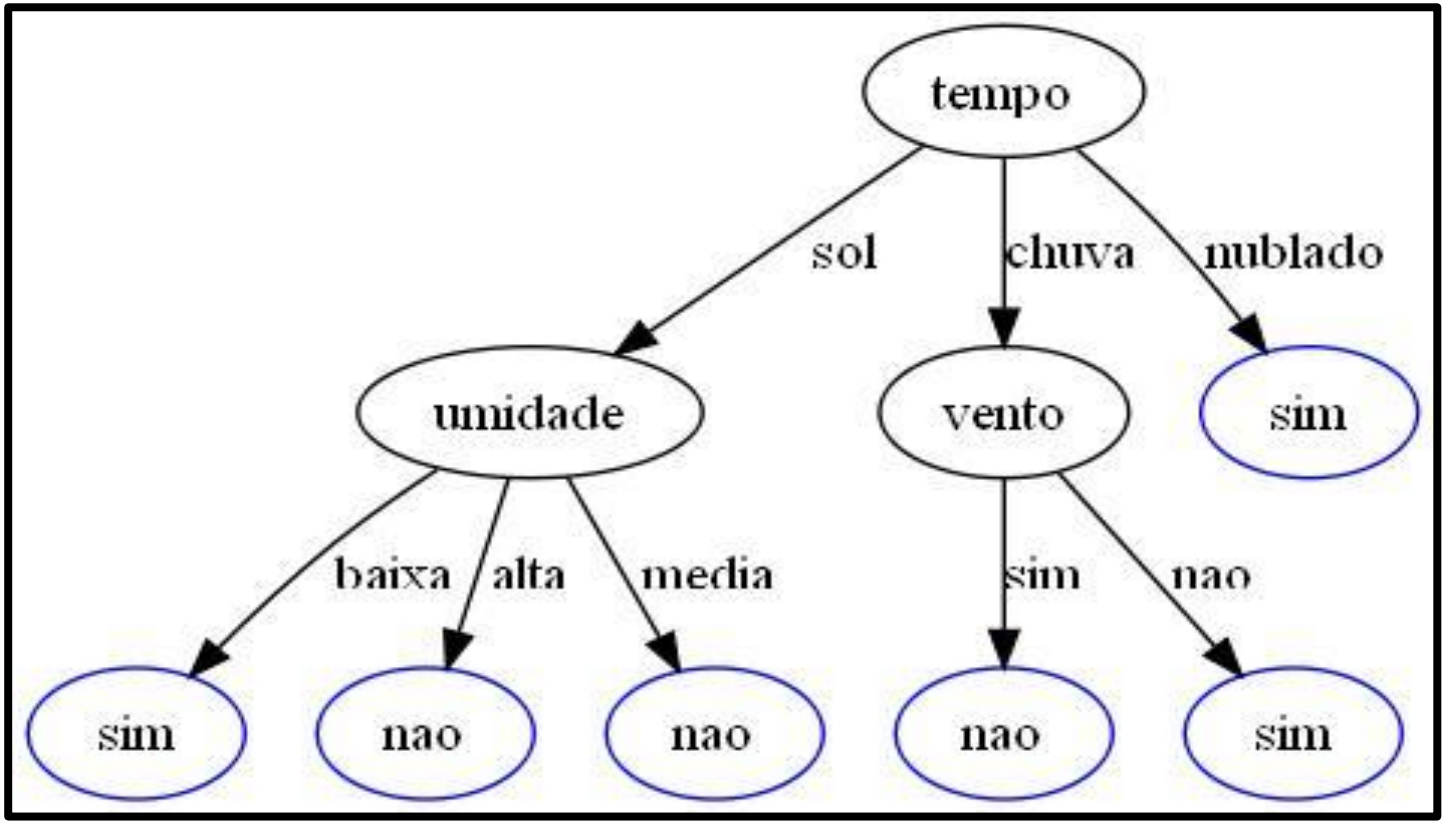
E = 0,940

Ganho (S, Tempo) = 0,247

Razão de Ganho = 0,263



Árvore gerada para o exemplo do jogo de Tênis.



Critério de Parada

Quando parar as divisões?

- Quando todos os exemplos pertencem a mesma classe;
- Quando todos os exemplos têm os mesmos valores dos atributos (mas diferentes classes);
- Quando o número de exemplos é inferior a certo limite;
- Quando o mérito de todos os possíveis testes de partição dos exemplos é muito baixo.

Exemplo 2

Referências Bibliográficas

[1] POZZER, C. T.. *Aprendizado Por Árvores de Decisão*. Departamento de Eletrônica e Computação, Universidade Federal de Santa Maria, 2006. Apostila.

[2] MITCHELL, T.. *Machine Learning*. Mc-Graw Hill, 1997.

[3] QUINLAN, J. R.. Induction of Decision Trees. In: SHAVLIK J. W., DIETTERICH, T. G.. *Readings in Machine Learning*. 1 ed. Morgan Kaufman, 1986. P. 81-106.

[4] ZUBEN, F. J. V., ATTUX, R. R. F.. *Árvores de Decisão*. Universidade de Campinas, 2004. Apostila.

[5] QUINLAN, J. R.. *C4.5: Programs for Machine Learning*. Morgan Kaufman, 1993.

[6] BREIMAN, L. et al. *Classification and Regression Trees*. Belmont, California: Wadsworth, 1984.