



# Pré-processamento de Dados

Redução,  
Transformação e  
Construção de Atributos  
Detecção de  
Duplicados  
Outliers

*Huei Diana Lee*

Inteligência Artificial  
CECE/UNIOESTE-FOZ

# Redução de Dimensionalidade

Atributos

# Precisão x Acurácia

## **pre·ci·são**

(latim *praecisio*, -nis, corte, golpe, lugar onde algo é cortado, ruína, destruição)  
*substantivo feminino*

1. Falta ou carência de alguma coisa necessária ou útil.
2. Necessidade, urgência.
3. Regularidade ou rigor na execução, funcionamento ou determinação de algo  
. = .EXATIDÃO
4. Concisão, laconismo.
5. Escolha rigorosa de palavras e expressões (ex.: *precisão de linguagem*).
6. Cumprimento rigoroso de horários. = PONTUALIDADE
7. [Antigo] Momento preciso, ocasião inevitável.

## **a·cu·rá·ci·a**

(inglês *accuracy*, do latim *accuratus*, -a, -um, particípio passado de *accuro*, -are, cuidar de)  
*substantivo feminino*

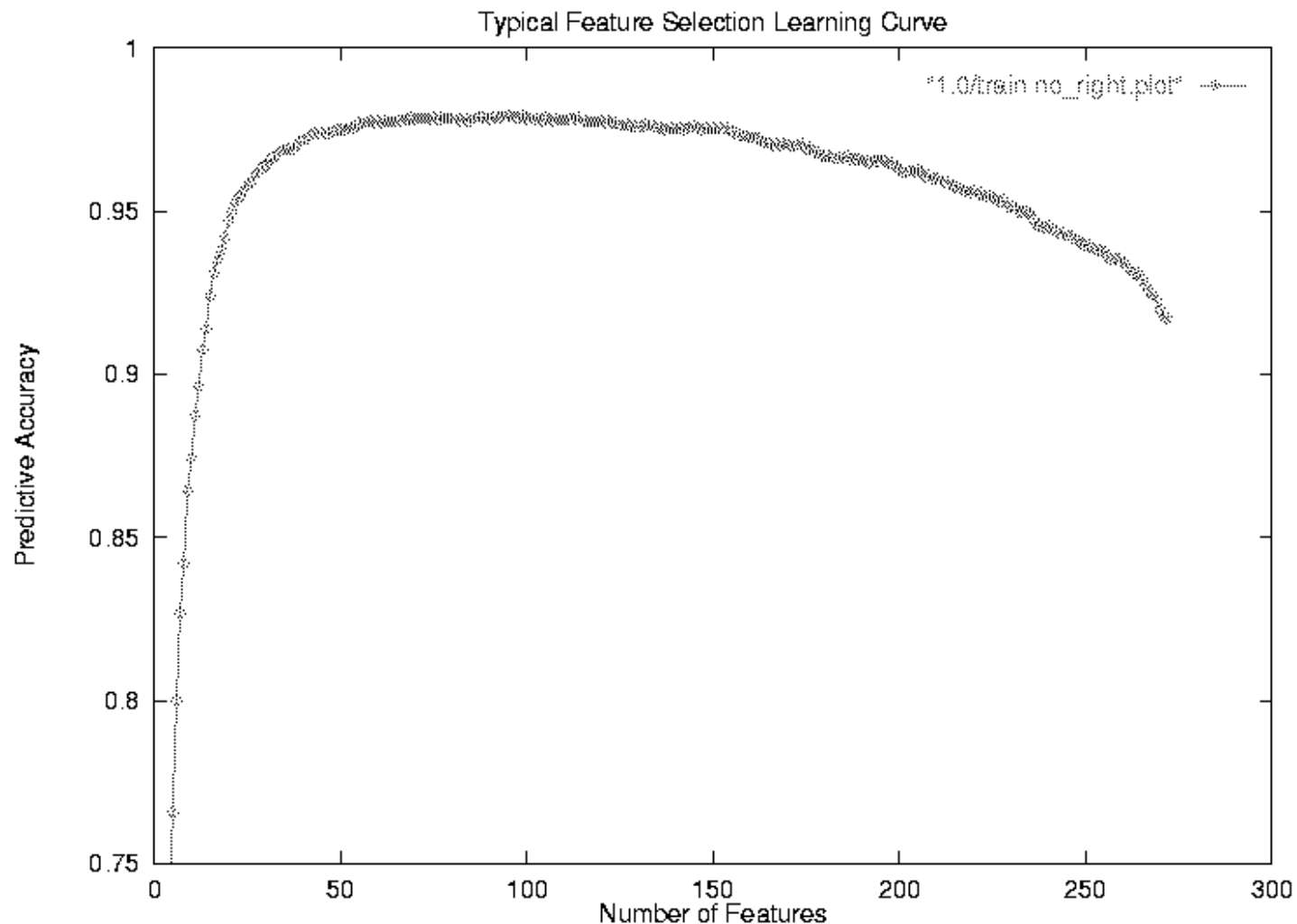
[Física] .Exatidão de uma medição ou de um instrumento de medição.

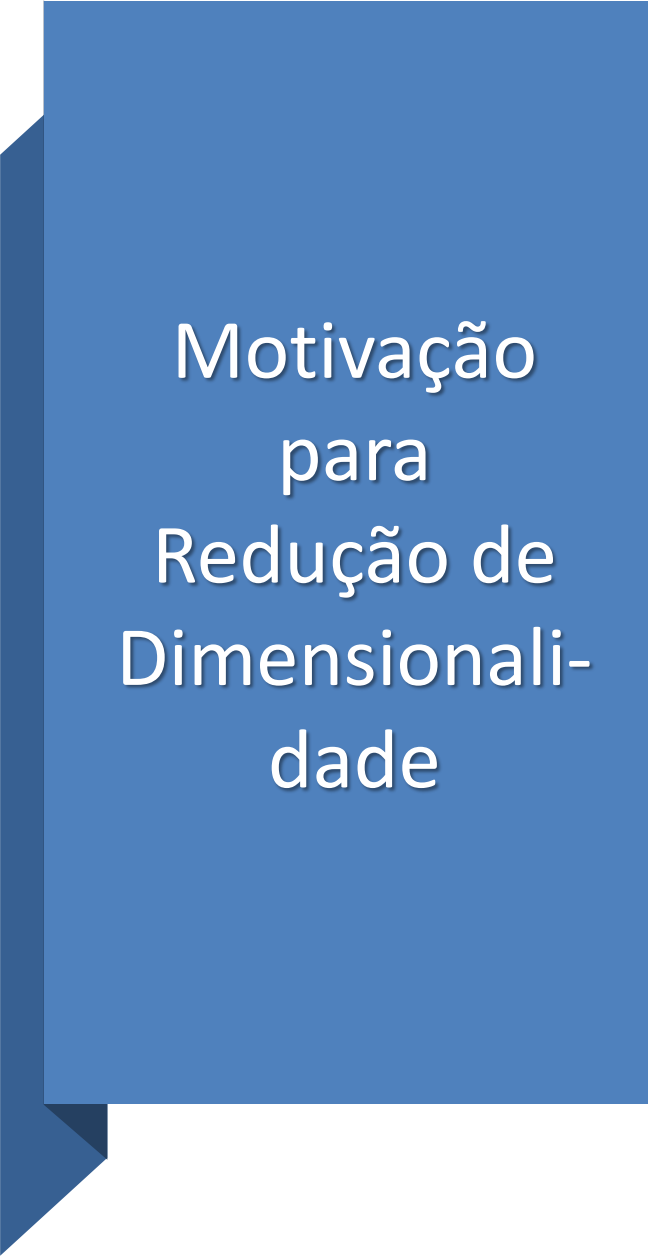
# Precisão x Acurácia



<https://arkadlab.com/acuracia-precisao-e-exatidao/>

# Motivação para Redução de Dimensionalidade





## Motivação para Redução de Dimensionali- dade

- possibilidade de melhora da **precisão/acurácia** dos classificadores
- melhora da **compreensibilidade**, por exemplo dos dados e das regras
- possibilidade da **diminuição dos custos de coleta** dos dados
- possibilidade da **redução dos custos de processamento** de grandes quantidades de dados

# Redução de Dimensionalidade

Alternativa simples:

- Remover campos com nenhuma ou pouca variabilidade
- Examinar o número de valores distintos no campo
  - *Rule of thumb: remover um campo no qual quase todos os valores são os mesmos, por exemplo null, ou que estão abaixo de um limite mínimo*
  - *limite mínimo pode ser 0,5% ou em geral menos de 5% do número de elementos da menor classe*

# Redução de Dimensionalidade



Ranking de  
Atributos

Seleção de  
Subconjunto  
de Atributos



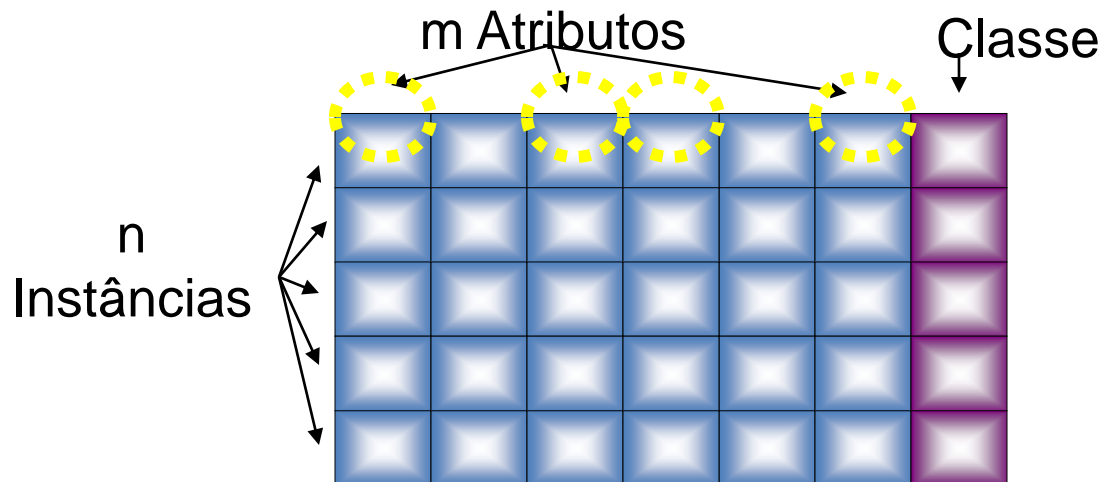
# Seleção de Atributos

## **Ranking de Atributos:**

- Os atributos são avaliados e ordenados segundo algum critério de importância, por exemplo os enumerados em estratégias de avaliação
- Os  $m$  mais relevantes são selecionados para compor o subconjunto final

# Seleção de Atributos

**Seleção de um Subconjunto de Atributos - SSA:**  
tarefa de encontrar o melhor subconjunto de atributos de acordo com alguma medida de importância (Attribute Subset Selection - FSS)



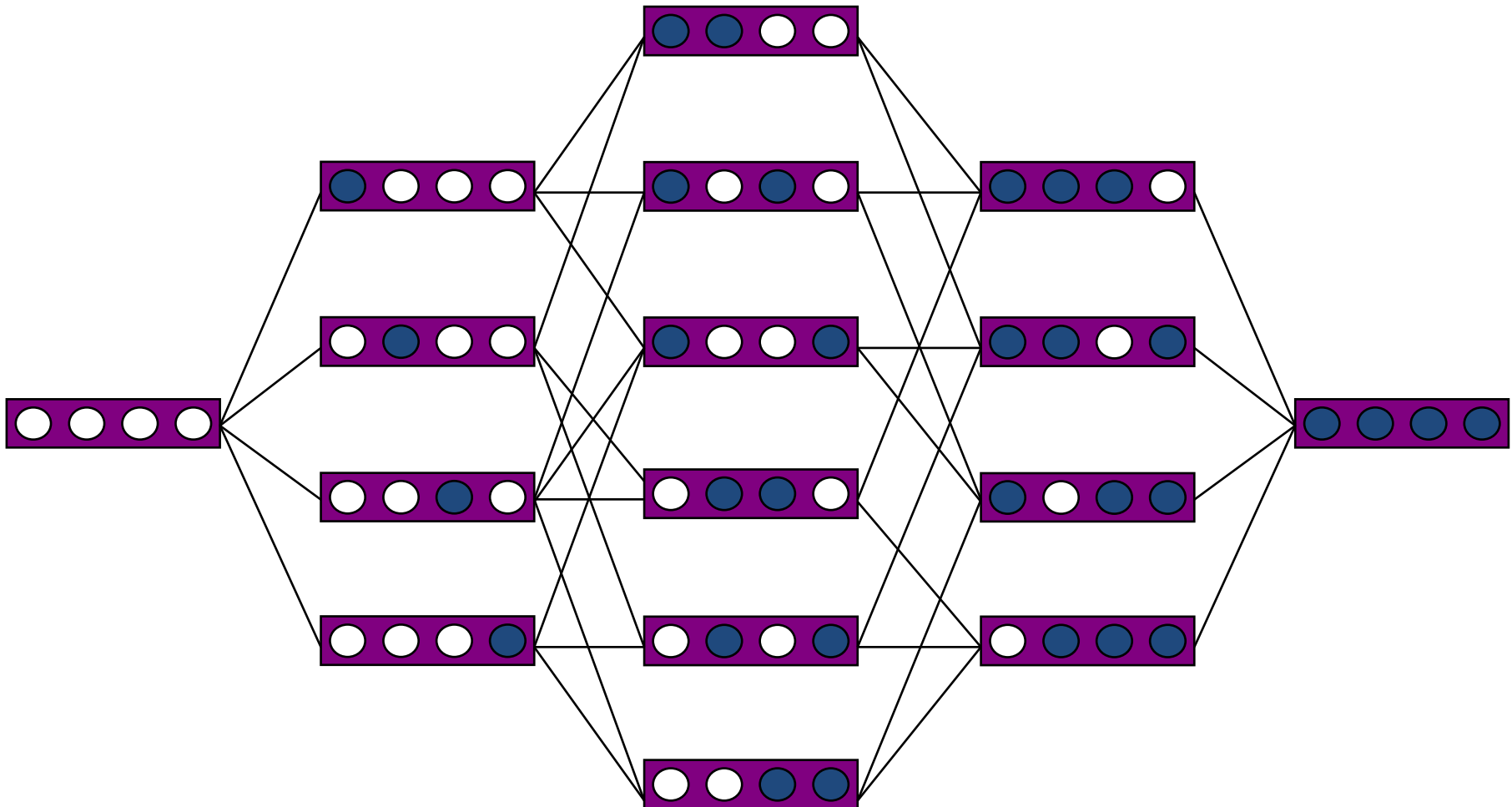
# Seleção de *Atributos*

SSA

Problema de Seleção de Atributos pode ser visto como um problema de **busca**, no qual cada estado do espaço de busca especifica um subconjunto de possíveis *atributos*

# Seleção de *Atributos*

SSA - Espaço de Busca com Quatro Atributos



# Seleção de *Atributos*

SSA - Espaço de Busca

A estrutura do espaço de busca sugere que cada método de seleção de atributos posicione-se em relação a 4 questões, as quais determinam:

- ◆ o **ponto de partida** para a busca no espaço
- ◆ a **organização** da busca
- ◆ a **estratégia** empregada na avaliação de subconjuntos alternativos
- ◆ o **critério de parada** da busca

# Seleção de *Atributos*

SSA - Ponto de Partida

Ponto de partida determina a direção da busca, definindo se a seleção será:

- Forward
- Backward
- Outward

# Seleção de *Atributos*

SSA - Organização da Busca

- Busca exaustiva impraticável
- Abordagens mais realísticas:
  - método *Greedy*
  - método *Stepwise* de seleção ou eliminação
  - método *Best-first*

# Seleção de *Atributos*

SSA - Organização da Busca

- Busca exaustiva impraticável
- Abordagens mais realistas
  - método *Greedy*
  - método *Stepwise* de seleção ou eliminação
  - método *Best-first*

Em cada ponto da busca, considera modificações locais para o subconjunto corrente de atributos, seleciona uma e então interage, nunca reconsiderando a escolha realizada



# Seleção de *Atributos*

SSA - Organização da Busca

- Busca exaustiva impraticável

- Abordagens mais realistas

- método *Greedy*
- método *Stepwise* de seleção
- método *Best-first*

Considera tanto a adição quanto a remoção de *atributos* em cada ponto de decisão, podendo retornar a uma decisão anterior

# Seleção de *Atributos*

SSA - Organização da Busca

- Busca exaustiva impraticável

- Abordagens mais realistas

- método *Greedy*
- método *Stepwise* de seleção
- método *Best-first*

Busca em “largura” utilizando função de avaliação, isto é, escolhe para a expansão o melhor candidato de acordo com sua previsão

# Seleção de *Atributos*

SSA - Organização da Busca

- Busca exaustiva impraticável

Estima o custo para se chegar ao objetivo

- Abordagens mais realistas

- método *Greedy*
- método *Stepwise* de seleção
- método *Best-first*

Busca em largura utilizando função de avaliação, isto é, escolhe para a expansão o melhor candidato de acordo com sua previsão.

# Seleção de *Atributos*

SSA - Estratégia da Avaliação

- Estratégia que será utilizada para avaliar os subconjuntos alternativos de atributos
- Alguns critérios:
  - habilidade do atributo discriminar entre as classes
  - teoria da informação
  - precisão do conjunto de treinamento ou conjunto separado de avaliação

# Seleção de *Atributos*

SSA - Critério de Parada

Alguns critérios de parada são:

- parar de adicionar ou remover atributos quando nenhuma das alternativas melhora a precisão da classificação
- continuar gerando os candidatos enquanto a precisão não se degradar
- continuar gerando os candidatos até que o outro extremo do espaço de busca seja alcançado e selecionar o melhor desses subconjuntos

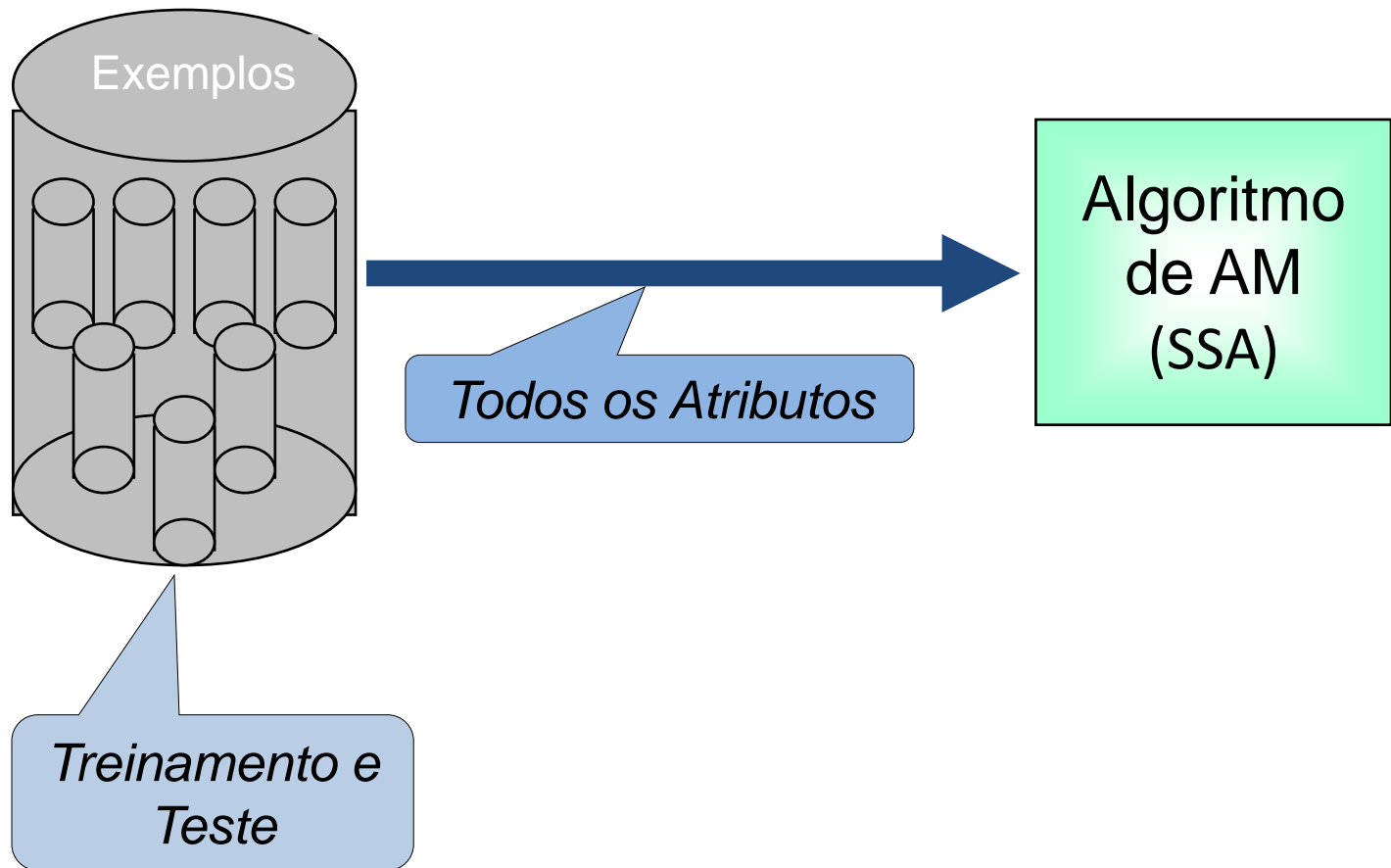
# Seleção de *Atributos*

SSA – Interação com Algoritmo de Aprendizado

- Um aspecto importante é como a estratégia de seleção de um subconjunto de atributos interage com o algoritmo básico de indução
- Pode-se subdividir em três abordagens:
  - *Embedded*
  - *Filtro*
  - *Wrapper*

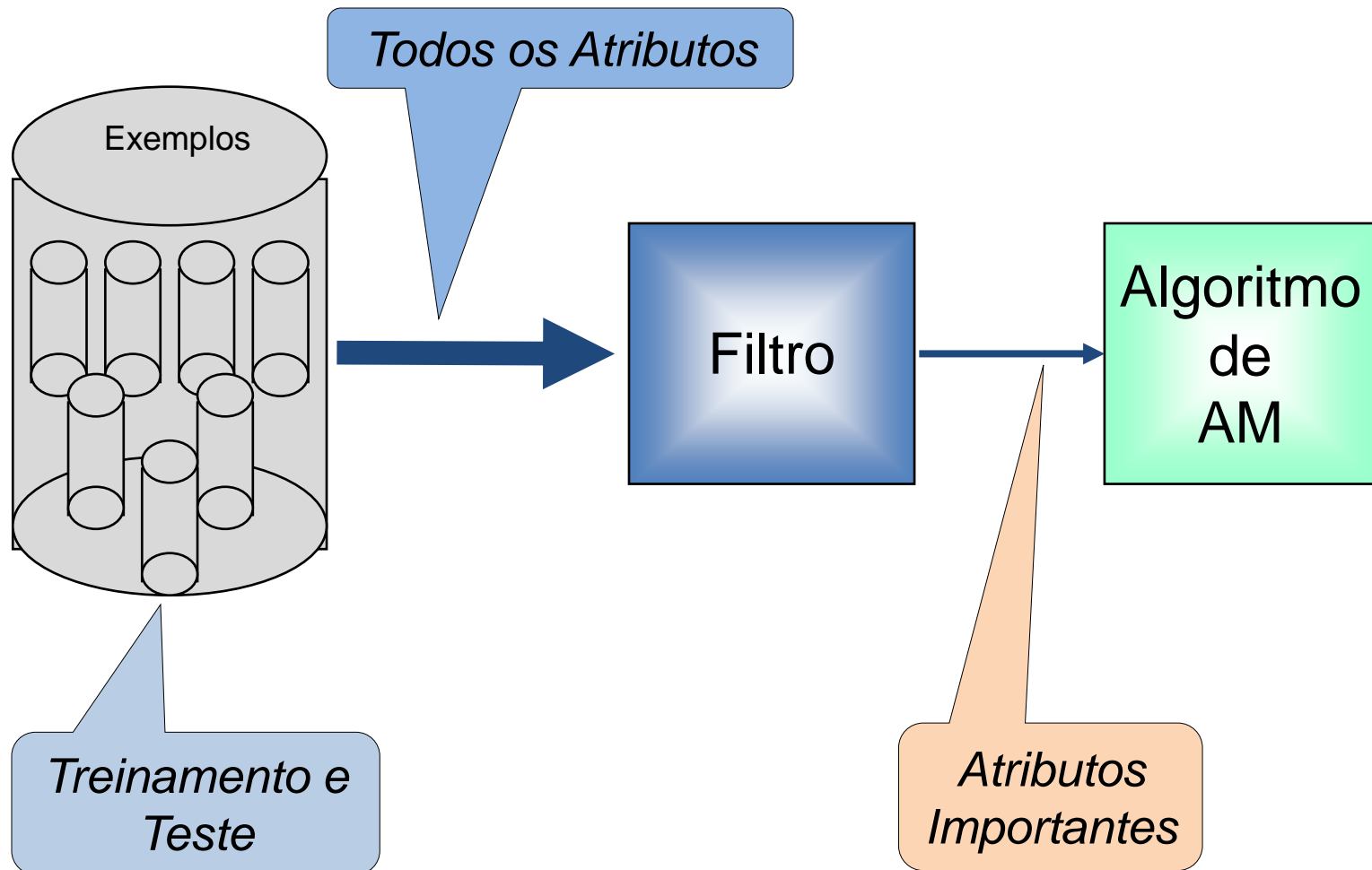
# Abordagem *Embedded*

SSA – Interação com Algoritmo de Aprendizado



# Abordagem *Filtro*

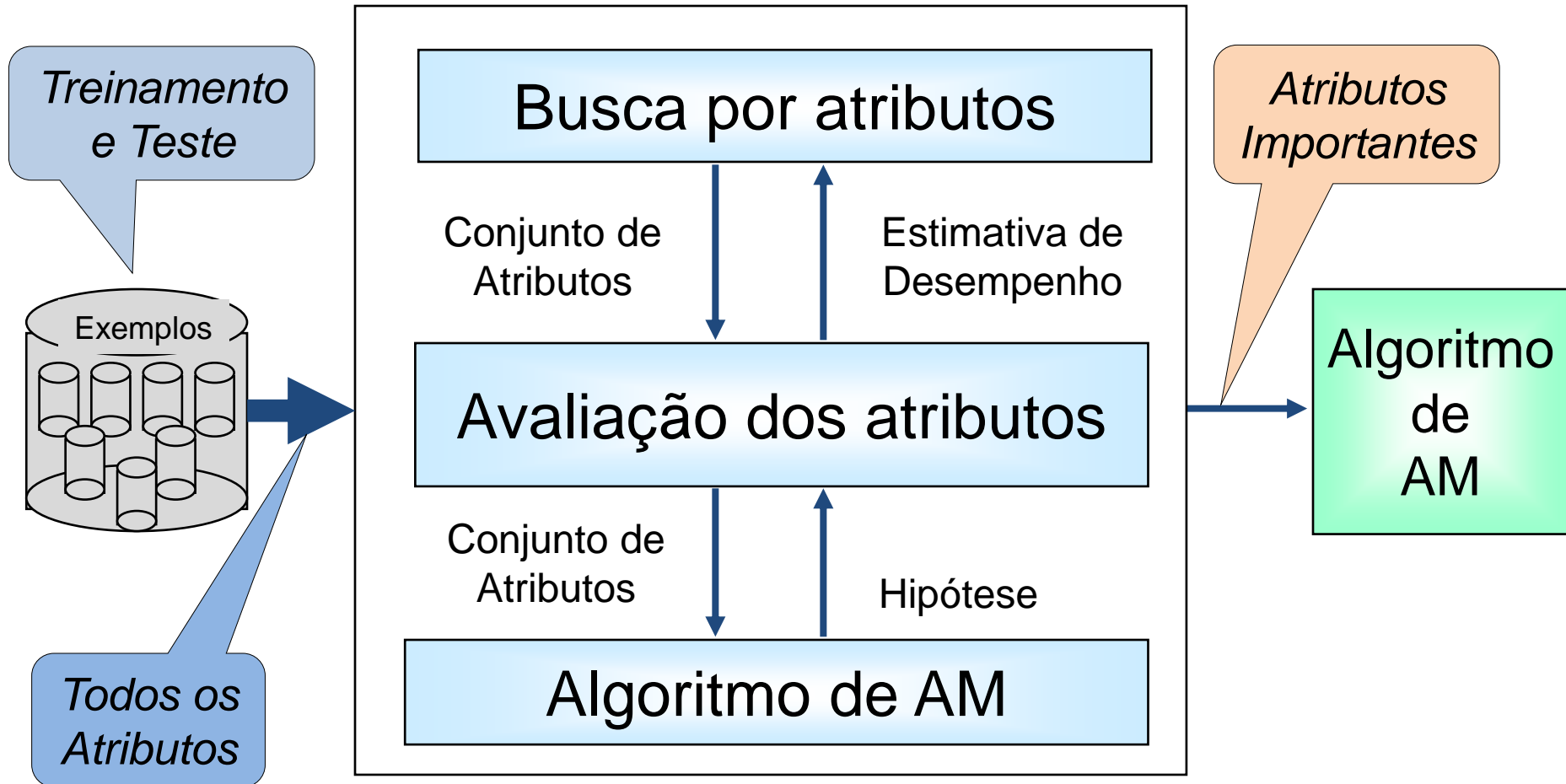
SSA – Interação com Algoritmo de Aprendizado





# Abordagem *Wrapper*

SSA – Interação com Algoritmo de Aprendizado



# Transformação de Atributos

Extração de Atributos

Construção de Atributos

# Transformação de Atributos

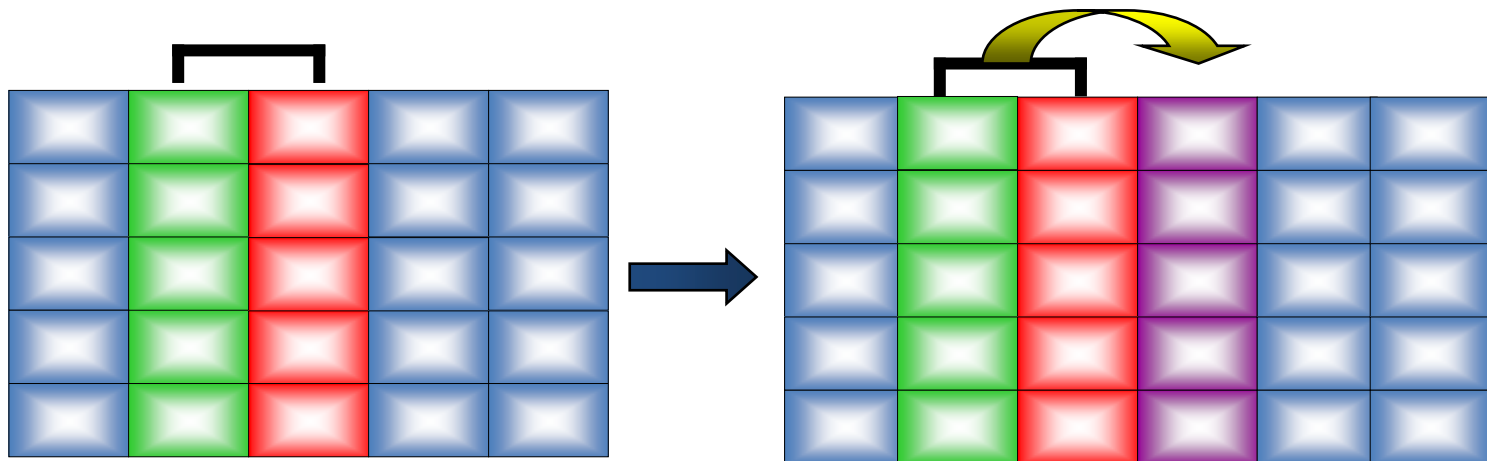
Além da seleção de atributos, é possível construir novos atributos por meio da combinação dos atributos originais para:

- criar atributos mais significativos ou
- “reduzir” temporariamente a dimensionalidade por meio de técnicas de transformação de atributos

Também denominado de Aprendizado Construtivo ou Indução Construtiva

# Transformação de Atributos

Consiste na aplicação de operadores construtivos a atributos já existentes, resultando na definição de uma ou mais novas atributos.



# Transformação de Atributos

## Exemplo intuitivo

Dado um conjunto de dados com atributos “massa” e “altura” de uma pessoa, é possível combinar esses atributos para construir o atributo Índice de Massa Corporal (IMC)

$$\text{IMC} = \frac{\text{massa}}{(\text{altura} \times \text{altura})}$$

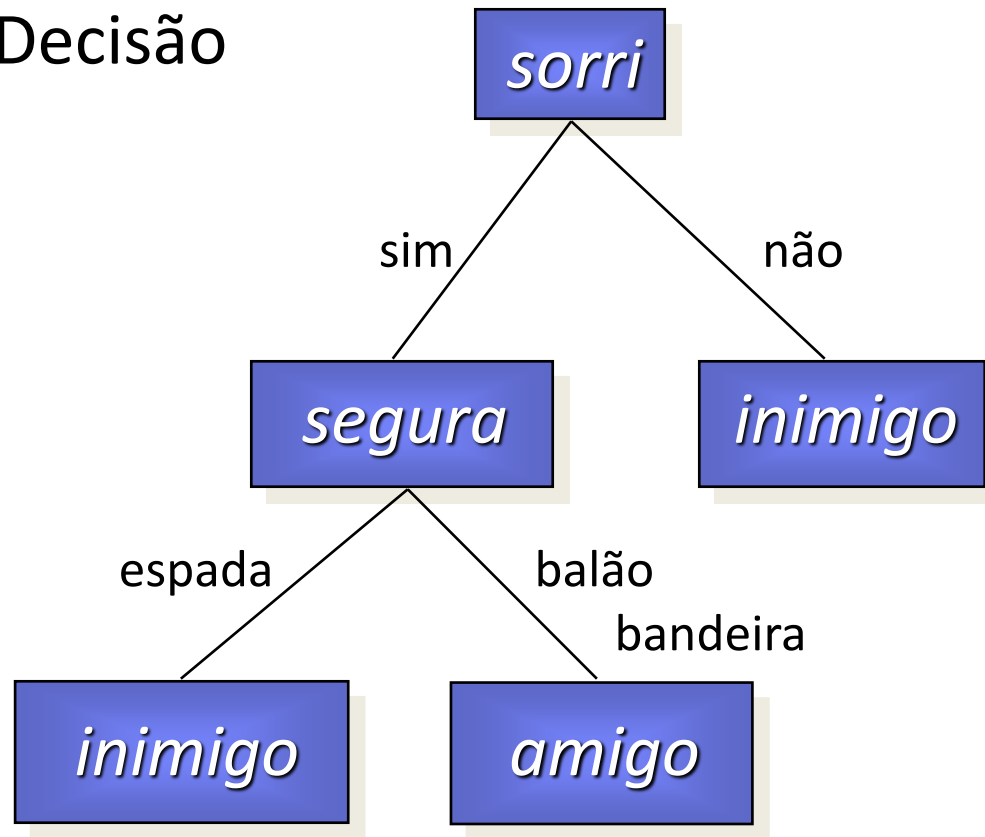
# Transformação de Atributos

## Exemplo de Robôs Amigos e Inimigos

Atributo-valor					classe
<i>sorri</i>	<i>segura</i>	<i>tem-gravata</i>	<i>cabeça</i>	<i>corpo</i>	
sim	balão	sim	quadrada	quadrado	amigo
não	espada	sim	quadrada	triangular	inimigo
sim	bandeira	sim	redonda	redondo	amigo
sim	espada	sim	triangular	redondo	inimigo
sim	balão	não	triangular	triangular	amigo
não	bandeira	não	redonda	quadrado	inimigo

# Transformação de Atributos

## Árvore de Decisão



# Transformação de Atributos

## Regras de Decisão

*Se* sorri = sim e  
segura = espada  
*então* inimigo.

*Se* sorri = sim e  
segura = balão ou  
bandeira  
*então* amigo.

*Se* sorri = não  
*então* inimigo.

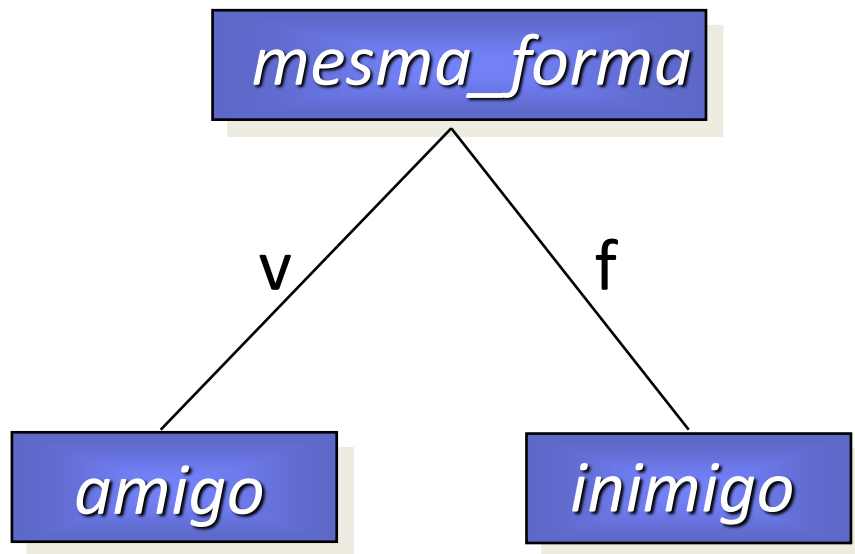


# Transformação de Atributos

Atributo-valor						classe
<i>sorri</i>	<i>segura</i>	<i>tem-gravata</i>	<i>cabeça</i>	<i>corpo</i>	<i>mesma_forma</i>	
sim	balão	sim	quadrada	quadrado	v	amigo
não	espada	sim	quadrada	triangular	f	inimigo
sim	bandeira	sim	redonda	redondo	v	amigo
sim	espada	não	triangular	redondo	f	inimigo
sim	balão	não	triangular	triangular	v	amigo
não	bandeira	não	redonda	quadrado	f	inimigo

# Transformação de Atributos

## Árvore e Regras de Decisão



Se *mesma\_forma* = v  
*então* amigo.

Se *mesma\_forma* = f  
*então* inimigo.

# Transformação de Atributos

*Principal Component Analysis (PCA)*

Análise de Componentes Principais

- Dado um conjunto  $D$  com  $n$  instâncias e  $p$  atributos  $(x_1, x_2, \dots, x_p)$ , uma transformação linear para um novo conjunto de atributos  $z_1, z_2, \dots, z_p$  pode ser calculada como:

$$z_1 = a_{11} x_1 + a_{21} x_2 + \dots + a_{p1} x_p$$

$$z_2 = a_{12} x_1 + a_{22} x_2 + \dots + a_{p2} x_p$$

...

$$z_p = a_{1p} x_1 + a_{2p} x_2 + \dots + a_{pp} x_p$$

- Componentes Principais (PCs) são tipos específicos de combinações lineares que são escolhidas de tal modo que  $z_p$  (PCs) tenham as seguintes características:

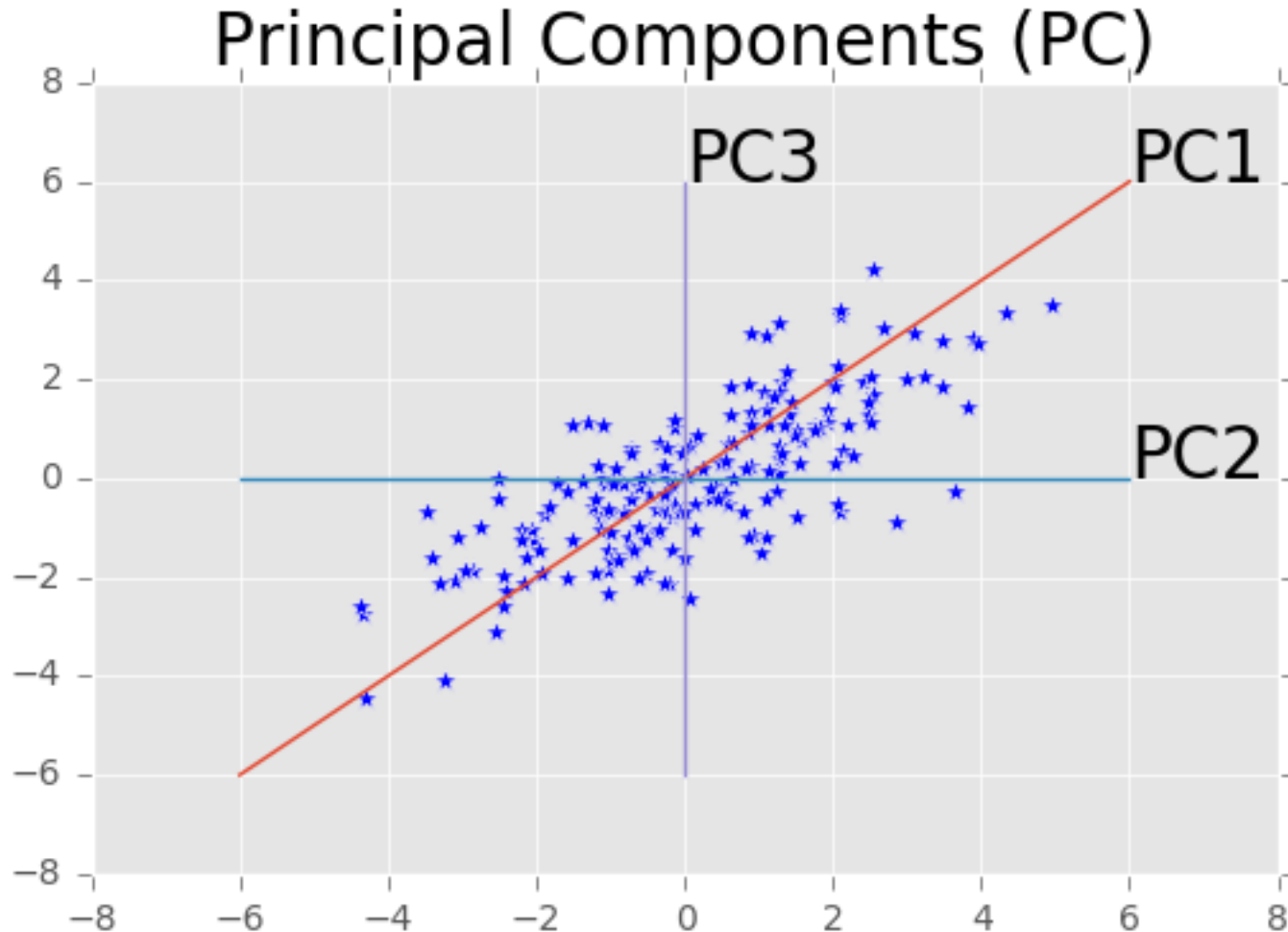
# Análise de Componentes Principais

## Características

- As  $z$  componentes principais (PC) são não-correlacionadas (independentes)
- As PCs são ordenadas de acordo com quantidade da variância dos dados originais que elas contêm (ordem decrescente)
  - A primeira PC “explica” (contém) a maior porcentagem da variabilidade do conjunto de dados original
  - A segunda PC define a próxima maior parte, e assim por diante
  - Em geral, apenas algumas das primeiras PCs são responsáveis pela maior parte da variabilidade do conjunto de dados
  - O restante das PCs tem uma contribuição insignificante

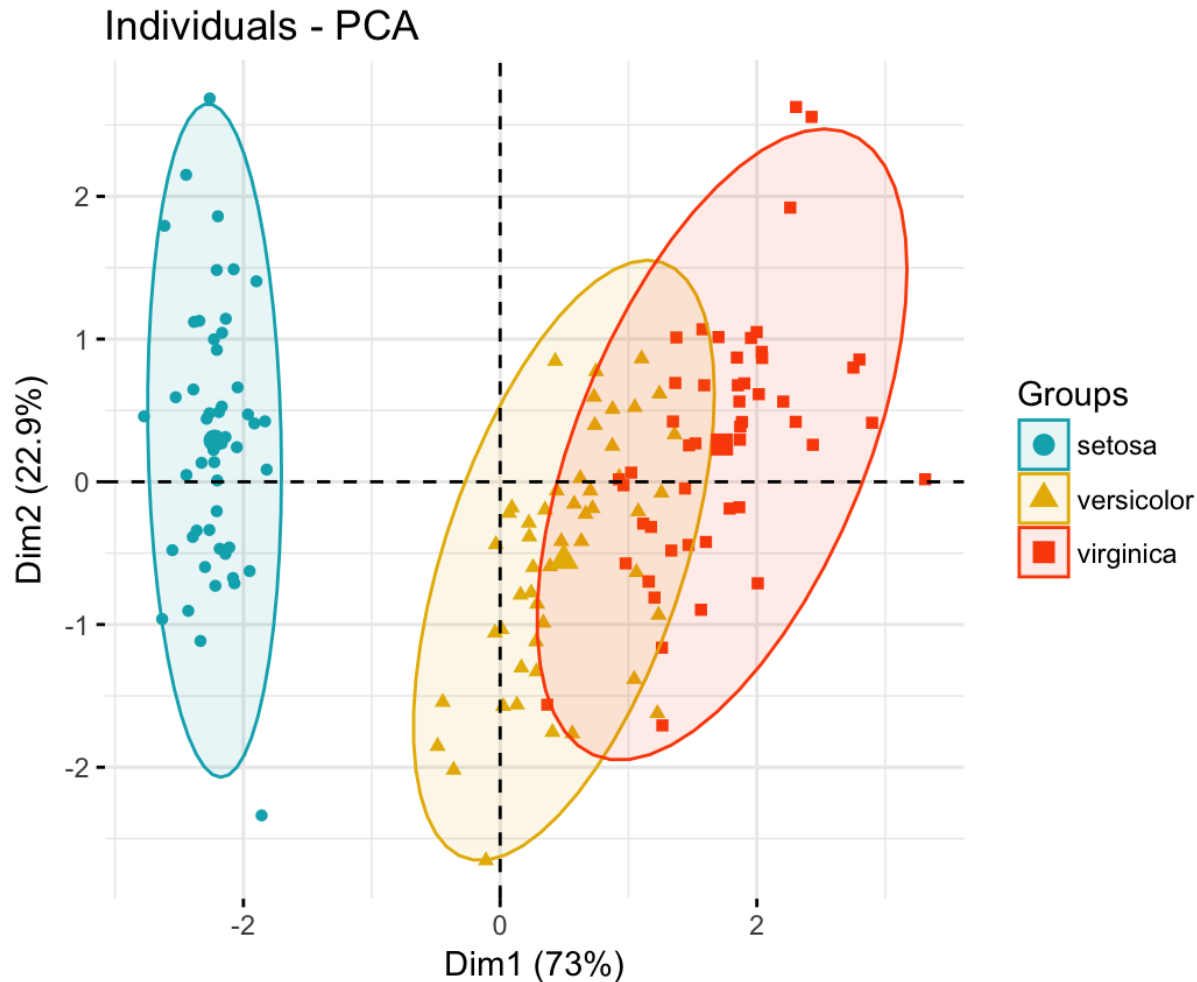
# Análise de Componentes Principais

Exemplo



# Análise de Componentes Principais

## Exemplo



# Análise de Componentes Principais

## Limitações

- Assume apenas relações lineares entre os atributos
- A interpretação dos resultados (por exemplo, classificador gerado) em termos dos atributos originais pode ficar mais difícil

# Detecção de Duplicados



# Detecção de Duplicados

- Algoritmos para determinar se dois ou mais registros são representações da mesma entidade
- Normalmente é preciso comparar cada registro com todos os outros (produto cartesiano)
- **Alternativa:** Método da Vizinhança Ordenada
  - Ordenação de uma Chave construída a partir dos atributos
  - Registros duplicados ficarão próximos entre si
  - Compara-se os registros no interior de uma janela deslizando de tamanho fixo

# Outliers

# Outliers e Erros (Ruídos)

- **Outliers** são valores de um atributo supostamente fora do intervalo deste atributo
- Frequentemente dados reais podem apresentar **casos raros** ou **erros (ruídos)** nas características ou classes
- O ruído pode fazer com que generalizações válidas não sejam encontradas

# Efeito do Ruído

## Exemplo

Imagine que há muitos exemplos positivos como #1 e #2, mas somente um exemplo negativo como o #5 que na verdade resultou de um erro de classificação

Exemplo	Tamanho	Cor	Forma	Classe
1	pequeno	vermelho	círculo	positivo
2	grande	vermelho	círculo	positivo
3	pequeno	vermelho	triângulo	negativo
4	grande	azul	círculo	negativo
5	médio	vermelho	círculo	negativo

# Detecção de Outliers e Erros (ruídos)

## Técnicas gráficas

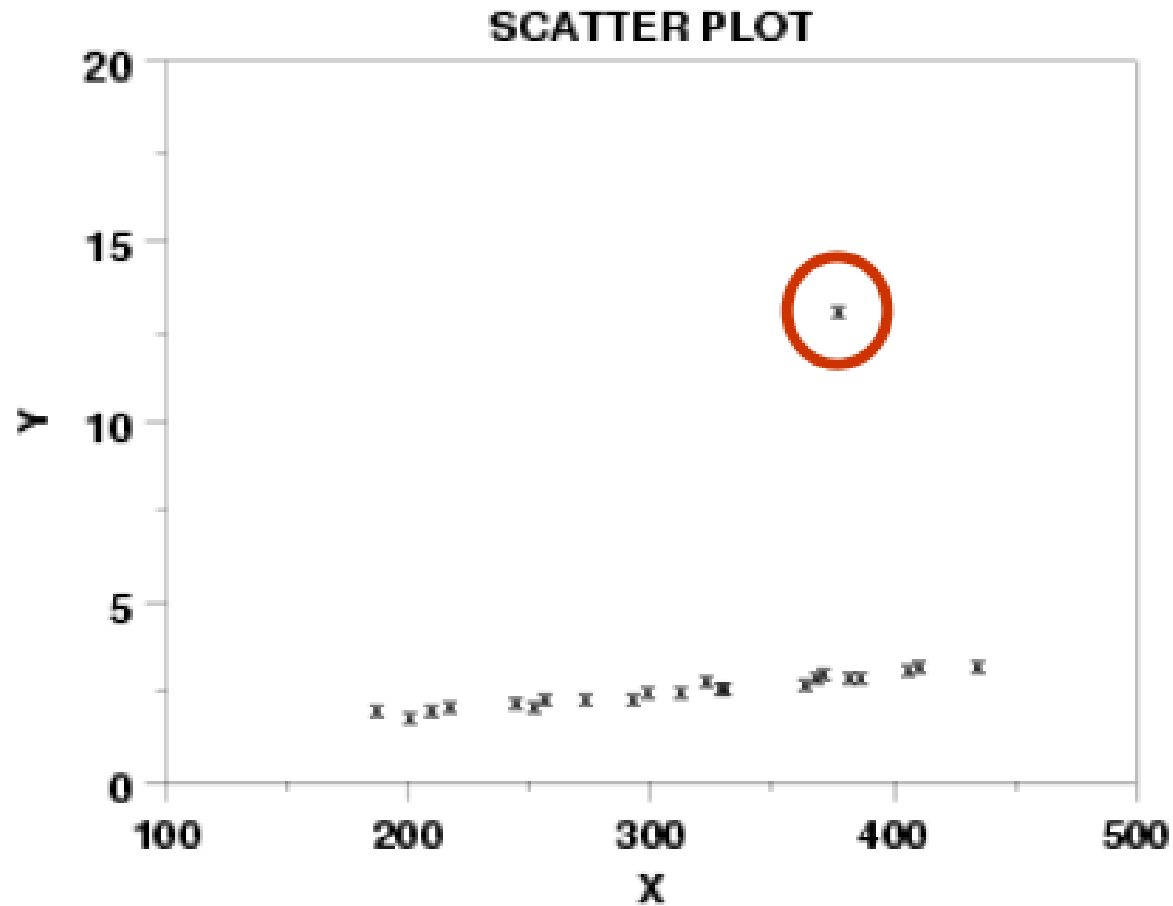


**Scatter  
plots**



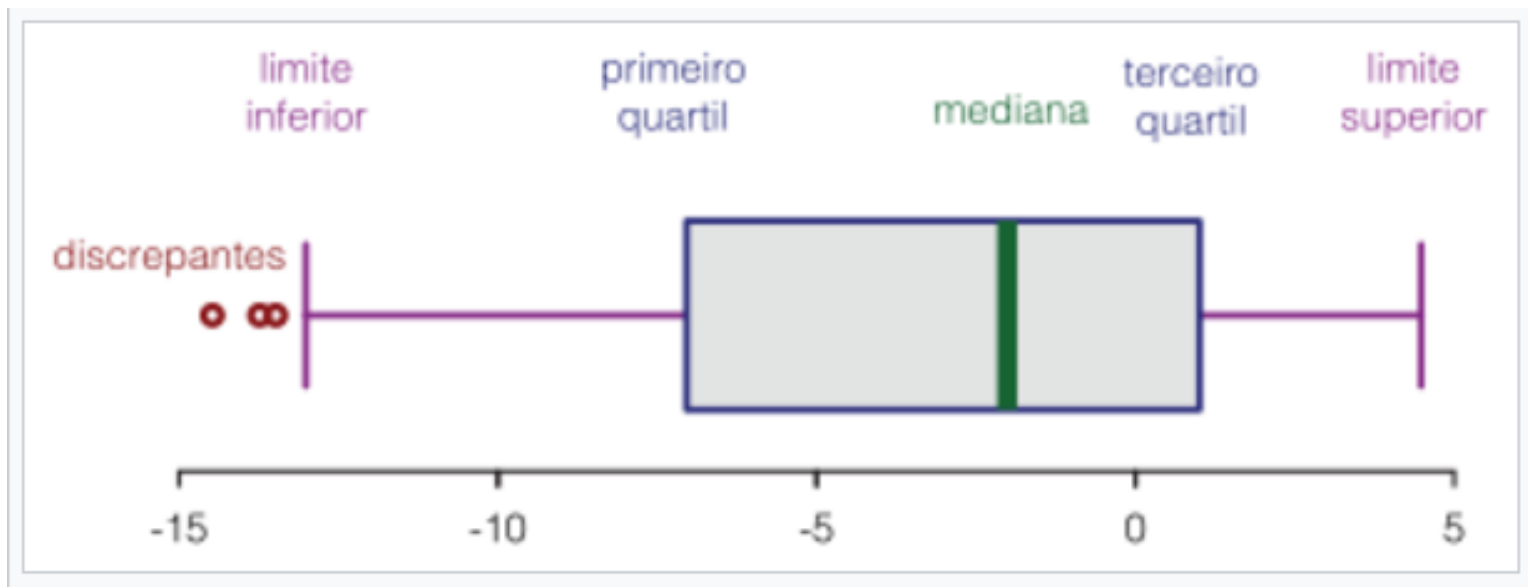
**Box  
plots**

# Scatter Plot



# Box Plot

Indicam localização e variação



# Box Plot

Exemplo:

30, 171, 184, 201, 212, 250, 265, 270, 272, 289, 305, 306  
322 322 336 346 351 370 390 404 409 411 306, 322,  
322, 336, 346, 351, 370, 390, 404, 409, 411, 436, 437,  
439, 441, 444, 448, 451, 453, 470, 480, 482, 487, 494,  
495, 499, 503, 514, 521, 522, 527, 548, 550, 559, 560,  
570, 572, 574, 578, 585, 592, 592, 607, 616, 618, 621,  
629, 637, 638, 640, 656, 668, 707, 709, 719, 737, 739,  
752, 758, 766, 792, 792, 794, 802, 818, 830, 832, 843,  
858, 860, 869, 918, 925, 953, 991, 1000, 1005, 1068,  
1441



# Box Plot

- Média: 576,08
- Mediana (Percentil 50%): 559,50
- Mínimo: 30
- Máximo: 1441
- Percentil 25% (Q1): 436
- Percentil 75% (Q2): 739
- $IQ = 739 - 436 = 303$
- **$L1 = Q1 - 1,5 \times IQ = -18,5$**
- **$L2 = Q1 - 3,0 \times IQ = -473$**
- $U1 = Q2 + 1,5 \times IQ = 1193,50$
- $U2 = Q2 + 3,0 \times IQ = 1648,00$

Valor < L1 ou Valor > U1: outlier leve  
Valor < L2 ou Valor > U2: outlier grave

# Box plot

- Indicam localização e variação
- Exemplo:

30, 171, 184, 201, 212, 250, 265, 270, 272, 289, 305, 306  
322 322 336 346 351 370 390 404 409 411 306, 322,  
322, 336, 346, 351, 370, 390, 404, 409, 411, **436**, 437,  
439, 441, 444, 448, 451, 453, 470, 480, 482, 487, 494,  
495, 499, 503, 514, 521, 522, 527, 548, 550, **559**, 560,  
570, 572, 574, 578, 585, 592, 592, 607, 616, 618, 621,  
629, 637, 638, 640, 656, 668, 707, 709, 719, 737, **739**,  
752, 758, 766, 792, 792, 794, 802, 818, 830, 832, 843,  
858, 860, 869, 918, 925, 953, 991, 1000, 1005, 1068,  
**1441**

1441 > U1: outlier leve

# Box plot

Média: 576,08

Outlier: 1441

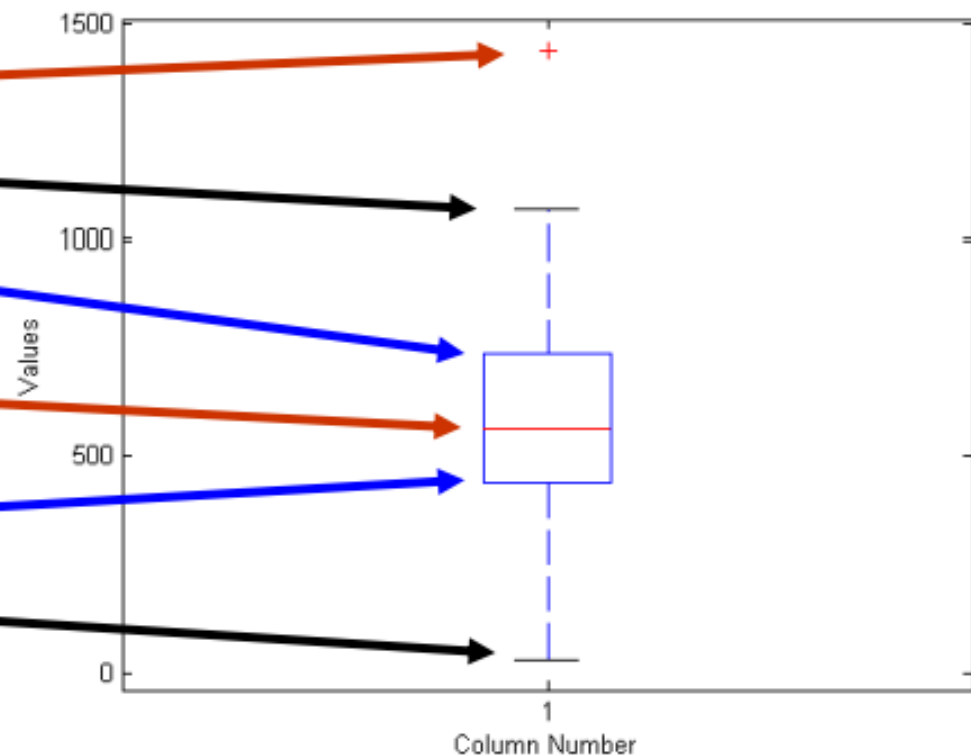
Novo Máximo: 1068

Percentil 75%: 739

Mediana: 559,50

Percentil 25%: 436

Mínimo: 30



## Técnicas de Correção de Outliers

- Ignorar o registro
- Corrigir o valor manualmente
- Usar uma constante global
- Usar o valor médio do atributo na base
- Usar o valor médio entre os exemplos mais próximos
- Usar o valor médio do atributo na classe
- Usar o valor mais provável

Alguns slides foram baseados em apresentações de:

- Profa. Huei Diana Lee
- Profa. Maria Carolina Monard
- Prof. Ronaldo Cristiano Prati.
- Prof. Walter Nagai
- Prof. E. Keogh
- Prof. Nitin Patel
- Prof. José Augusto Baranauskas
- Prof. Gustavo E.A.P.A. Batista
- Prof. Patrick H. Winston
- Profa. Ana Carolina Lorena
- Prof. André C. P. L. F. Carvalho
- Prof. Ricardo Campello
- Profa. Solange O. Rezende
- Prof. Marcilio C. P. Souto
- Prof. Carlos Soares
- Prof. Paulo Horst
- Profa. Aurora Trinidad Ramirez Pozo