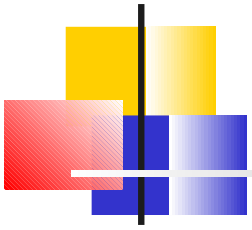
Ciência de Dados

Revisado: Roseli Romero

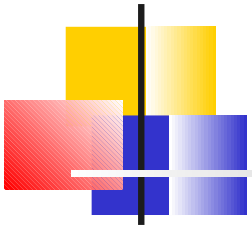
Pré-processamento de dados – Parte II

Prof. Dr. André C. P. L. F. de Carvalho
Dr. Isvani Frias-Blanco
ICMC-USP



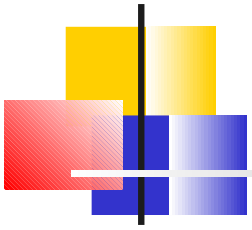
Dados desbalanceados

- Número de objetos varia para as diferentes classes
  - Natural ao domínio
  - Problema com geração / coleta de dados
- Várias técnicas de AM não conseguem lidar com esse problema
  - Tendência a classificar na(s) classe(s) majoritária(s)



Dados desbalanceados

- Alternativas
  - Alteração do conjunto de dados
    - Balanceamento artificial
  - Utilizar diferentes custos de classificação para as diferentes classes
  - Induzir um modelo para uma das classes
  - Alteração do projeto de algoritmos para lidar com desbalanceamento

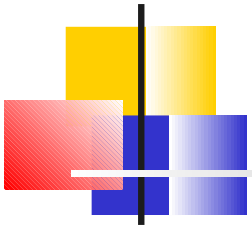


Balanceamento artificial

- Redefinir o tamanho do conjunto de dados
  - Acrescentar objetos (sobreamostragem)
    - Replicar objetos da classe minoritária

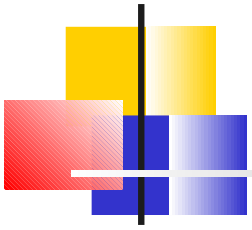
    não adiciona informação
  - Eliminar objetos (subamostragem)
    - Ignorar objetos da classe majoritária

    Para Remover informação
  - Abordagem híbrida



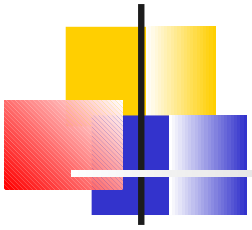
Dados desbalanceados

- Alguns problemas em algoritmos de AM só aparecem quando os dados estão desbalanceados
- Atenção!!!
  - Pode ser que uma distribuição igualitária das classes não seja boa
    - Mesmo se a população apresentar essa distribuição



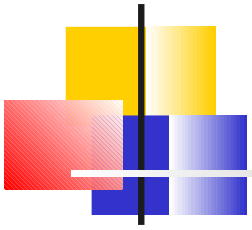
Transformação de dados

- Mudam o tipo de um atributo
- Conversão de valores entre tipos
  - Qualitativos para quantitativos
    - Binarização
  - Quantitativos para qualitativos
- Normalização de valores numéricos
- Tradução de atributos



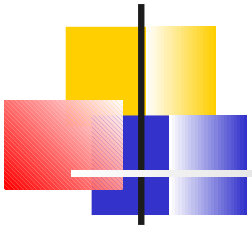
Qualitativos para quantitativos

- Algumas técnicas trabalham apenas com valores numéricos
- Conversão depende de:
  - Existência de ordenação dos valores
    - Se existe (ordinal), manter
    - Se não existe (nominal), não inserir
  - Número de valores
    - Se igual a 2 (binários) ou maior que 2



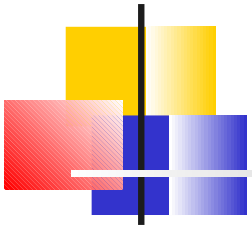
Conversão de valor ordinal

- Codificar para valor inteiro positivo
  - Ex. Pequeno: 1, médio: 2 e grande: 3
- Algumas técnicas trabalham apenas com valores quantitativos binários
  - Binarização



Binarização de ordinal

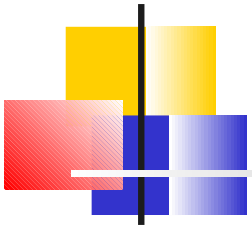
- Transformação no sistema numérico binário correspondente?
  - Perde ordenação
- Valores consecutivos devem diferir em 1 bit
- Codificar cada valor por um vetor binário que mantém ordenação
  - Código cinza: 000, 001, 010, 011, ...
  - Código termômetro: 001, 011, 111



Código cinza

- Existem vários códigos cinza
  - Não é único
- Um código cinza para 3 bits:
  - 000, 001, 011, 010, ...
- Um código cinza para 2 bits:
  - 00, 01, 11, 10

Dígito	Binário	Código cinza
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

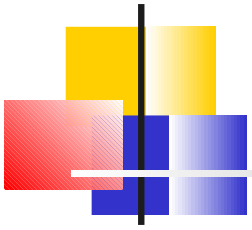


Algoritmo código cinza

1 Começa com todos os bits iguais a zero

2 Para cada novo número

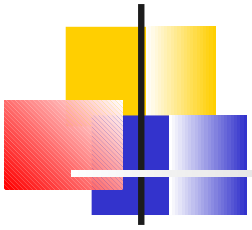
Mudar o valor do bit mais a direita que gera uma nova sequência de bits



Código termômetro

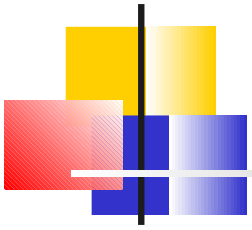
- Utiliza mais bits que código cinza
  - Tamanho cresce linearmente com número de valores

Dígito	Binário	Código termômetro
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0111
4	0100	1111



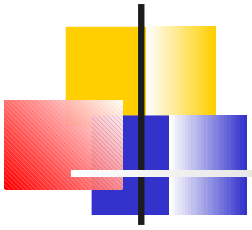
Conversão de valor nominal

- Transforma para valor quantitativo
  - Não deve inserir relação de ordem
- Codificação binária nominal sem relação de ordem
- Codificações
  - 1-de-n (n = número de valores)
  - m-de-n



Conversão de valor nominal

- Codificação 1-de-n
  - Codificação canônica
  - Fácil calcular moda = posição com maior número de valores 1
  - Quantidade de valores pode gerar vetores longos
- Codificação m-de-n
  - Dos n valores, m são iguais a 1 e os demais 0
  - Vários códigos

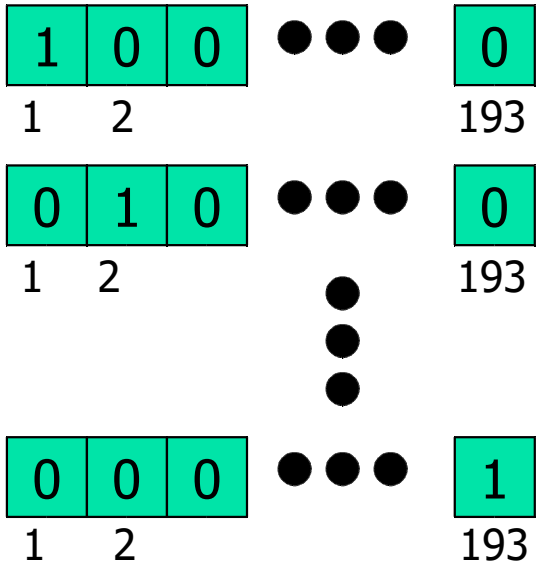


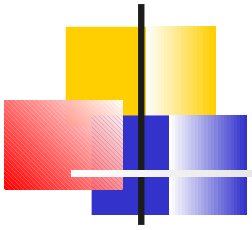
Conversão de valor nominal

- Número de valores de um atributo pode ser muito grande
- Pseudo atributos
  - Cria valores novos, artificiais
- Ex.: Atributo é nome de país
  - Existem 193 países (192 representados na ONU + Vaticano)
  - Alternativa de codificação:
    - Transformar valores nominais em valores numéricos utilizando a codificação 1-de-n

Alternativa 1

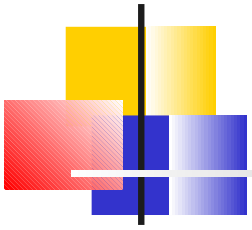
- Transformar valores nominais em valores binários utilizando a codificação 1-de-n
  - Maldição da dimensionalidade
  - Grande parte dos elementos possui valor 0
    - Valores esparsos





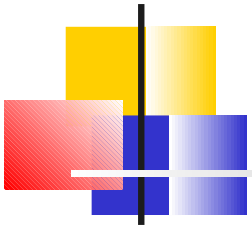
Alternativa 2

- Transformar 193 atributos em 10 pseudo-atributos
  - Continente: 7 valores binários
  - IDH: 1 valor real
  - População: 1 valor inteiro
  - Área: 1 valor inteiro



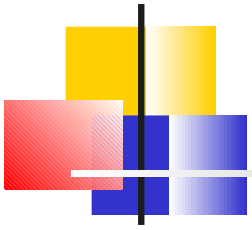
Exercício

- Transformar valores do atributo nome de automóvel em pseudo-atributos
  - Ex.: Uno, fox, amarok, corsa, zafira, corolla, TR4, gol, palio, doblo, clio, kangoo, omega



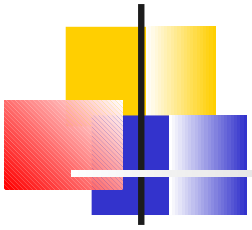
Quantitativos para qualitativos

- Discretização de valores
  - Transformar valores numéricos em intervalos (ou categorias)
- Subtarefas
  - Definição do número de categorias
    - Geralmente feito pelo usuário
  - Definição de como mapear valores dos atributos numéricos para essas categorias
    - Por frequência ou largura dos intervalos
    - Geralmente feito por um algoritmo



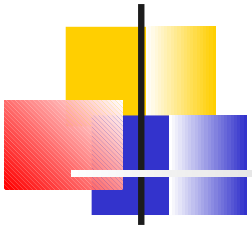
Transformação de atributos

- Muda valor numérico de um atributo para outro valor numérico
  - Limites de valores para atributos distintos podem ser muito diferentes
    - Evitar que um atributo predomine sobre outro
      - A menos que isso seja importante
  - Valores podem estar concentrados em uma determinada faixa ou região
  - Possível necessidade de binarização



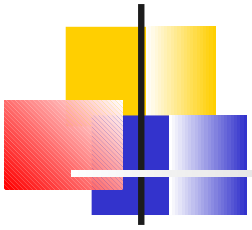
Transformação de atributos

- Aplicada aos valores de um atributo específico para todos os exemplos
- Variações
  - Funções simples
  - Normalização
  - Padronização



Funções simples

- Uma função matemática simples é aplicada a cada valor do atributo
  - Muda distribuição de valores de um atributo
  - Possíveis transformações para um atributo *x* de um conjunto de dados:
    - x^k, *log(x)*, e^x, $\sqrt{x}$, *1/x*, *sqrt(x)*, *seno(x)* e *|x|*

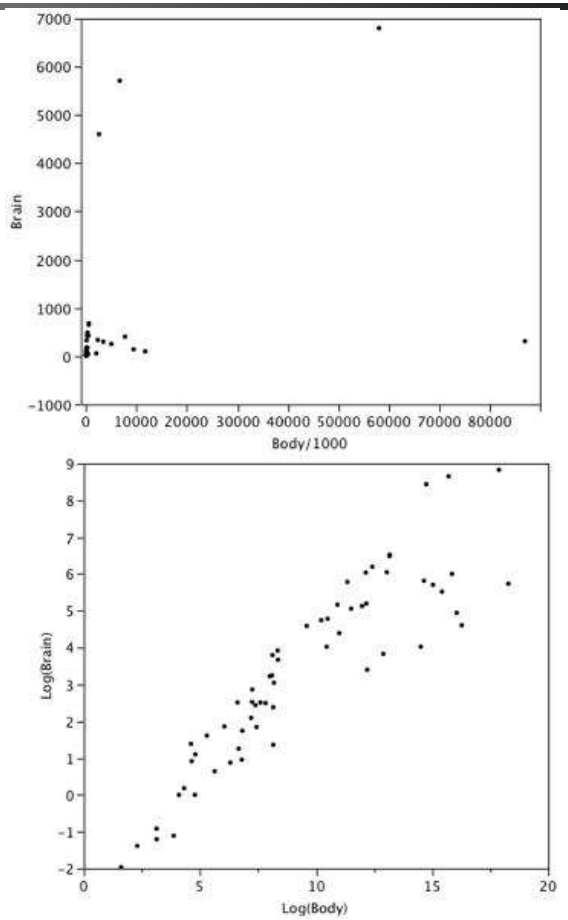


Funções simples

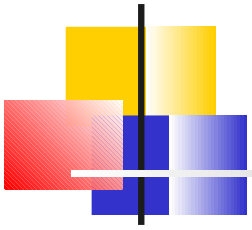
- Valor absoluto
  - Em algumas aplicações, apenas magnitude do valor de um atributo é importante
  - Converte valor de todos os atributos para o valor positivo correspondente
    - Ex.: -4, 5 e –2 se tornam 4, 5 e 2

Funções simples

- Utilizando função $\log_{10}$
  - Comprime valores de atributos com um grande intervalo de possíveis valores
  - Ex.: relação, para alguns animais, entre:
    - Peso do cérebro e
    - Peso do corpo



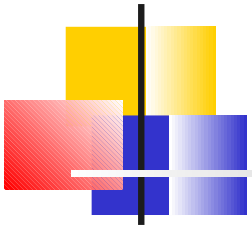
http://onlinestatbook.com/2/transformations/log.html



Normalização

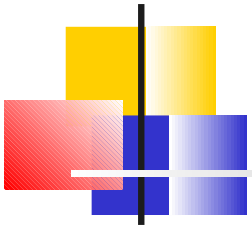
- Para normalizar os valores de um atributo:
  1. Adicionar ou subtrair uma constante
  2. Multiplicar ou dividir por uma constante
- Utilizado para mudar intervalo de valores dos dados
  - Permite converter todos os valores de um atributo para o intervalo [0, 1]

$$x' = \frac{(x - \min_x)}{(\max_x - \min_x)}$$



Exercício

- Normalizar os valores 12, 5, 4, 10, 20, 3 para os intervalos:
  - [-1, +1]
  - [-7, 12]



Padronização

- Para padronizar os valores de um atributo:
  1. Adicionar ou subtrair uma medida de localização
  2. Multiplicar ou dividir por uma medida de espalhamento
- Se os valores têm uma distribuição Gaussiana
  - Subtrair a média
  - Dividir pelo desvio padrão
  - Produz valores com distribuição normal (0,1)

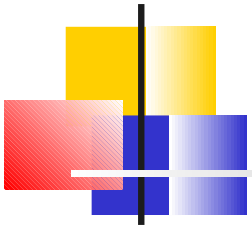
$$x' = \frac{(x - \bar{x})}{\sigma}$$

Exercício

- Converter os seguintes valores numéricos utilizando normalização e padronização

Valores	Re-escala	Padronização
3		
9		
5		
11		
5		
7		

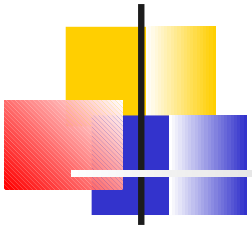
$$\text{var}(x) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$$



Exercício

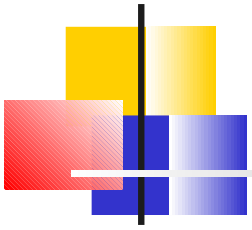
- Converter os dados abaixo para valores numéricos no intervalo [0, 1]

Febre	Enjoo	Batimentos	Vacina	Diagnóstico
baixa	sim	baixo	A	doente
média	não	normal	C	saudável
alta	sim	alto	B	saudável
alta	não	baixo	A	doente
baixa	não	alto	D	saudável
média	não	sem	C	doente



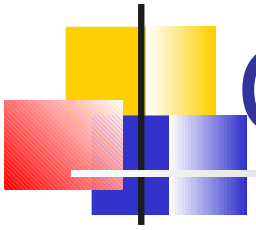
Conversão de valores numéricos

- É preferível padronizar a normalizar
- Em algumas aplicações
  - Atributos mais importantes podem ser deixados com limites maiores



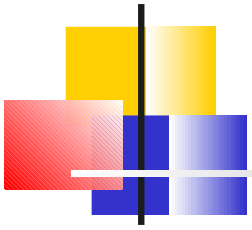
Tradução

- Ocorre devido a limitações no formato utilizado para armazenar o atributo
  - Algumas técnicas podem ter dificuldades com o formato original
  - Exemplos
    - Conversão de hora para valor inteiro
    - Conversão de data para valor inteiro
    - Conversão de nome de rua para código postal

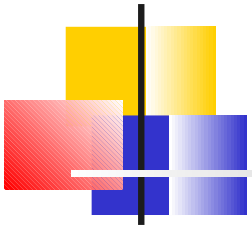


Considerações finais

- Qualidade de dados
  - Fontes de problemas
- Pré-processamento
- Limpeza de dados
- Desbalanceamento
- Transformação de dados



Perguntas



Exercício

- Que alternativa para lidar com dados desbalanceados você sugere para os casos abaixo?
  - 2 classes, com 5 e 90 exemplos cada
  - 2 classes com 3000 e 200 exemplos cada
  - 3 classes com 14, 100 e 600 exemplos cada