SCC0173- Mineração de Dados

Pre-processamento de dados

Docente: Solange Rezende
PAE: Brucce

Tópicos

- Introdução
- Amostragem
- Qualidade de dados
  - Limpeza de dados
- Transformação de dados
- Seleção de atributos

Pré-processamento

- Prepara os dados para seu uso por algoritmos de AM
- Procura ajudar melhorar desempenho do algoritmo
  - Custo
    - Tempo
    - Memória
  - Qualidade da previsão
    - Acurácia preditiva

Exemplo

- Primeiro passo:
  - Eliminar atributos irrelevantes

Nome	Febre	Enjoo	Batimentos	Dor	Salário	Diagnóstico
João	sim	sim	baixo	sim	1000	doente
Pedro	não	não	baixo	não	1100	saudável
Maria	sim	sim	alto	não	600	saudável
José	sim	não	baixo	sim	2000	doente
Ana	sim	não	alto	sim	1800	saudável
Leila	não	não	alto	sim	900	doente

Amostragem de dados

- Seleção de exemplos
- Base de dados grande
  - Algoritmo de AM não precisa usar todo conjunto de dados
  - Eficiência X acurácia
- Amostra
  - Pode levar à mesma acurácia com um esforço computacional menor
  - Deve ser representativa

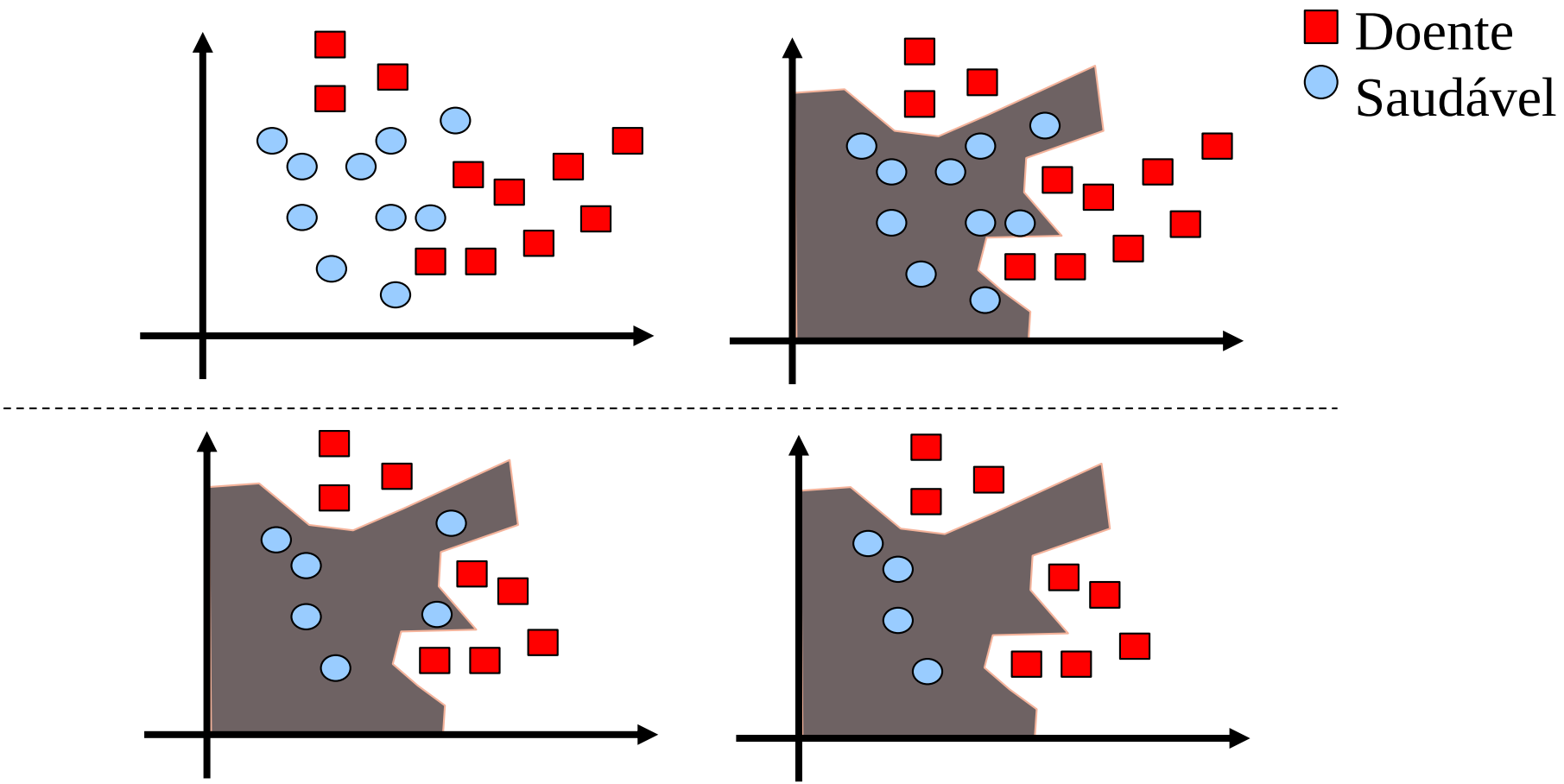
Amostra representativa

- Mantém propriedades de interesse do conjunto de dados original
  - Ex.: $média_{amostra} = normal_{pop\text{-}original}$
- Fornece uma estimativa da informação contida na população original
- Tem efeito semelhante ao uso de toda a população como não é possível garantir que isso ocorra
  - Técnicas de amostragem aumentam chances

Tipos simples de amostragem

- Amostragem aleatória simples
- Amostragem estratificada
- Amostragem progressiva

Exemplos de amostragem



Amostragem progressiva

Começa com pequenas amostras
Enquanto acurácia do modelo preditivo aumentar

Progressivamente aumenta tamanho da amostra

- Confirmar com outras amostras de tamanho semelhante à escolhida
- Boa estimativa de um tamanho adequado

Qualidade de dados

- Em geral, **dados não foram gerados para uso em AM**
  - Produzidos para outros propósitos
  - Frequentemente apresentam problemas
- Algoritmos de AM precisam geralmente de dados “limpos”
  - Entra lixo, sai lixo
  - Problemas nos dados precisam ser detectados e corrigidos
    - **Limpeza de dados**

Qualidade de dados

- Fontes dos problemas
  - Processos de medições e de coleta de dados
- Erros podem ter causa
  - Sistemática
    - Mais fácil de detectar e corrigir
  - “Aleatória”

Possíveis causas de erros

- Falha humana
- Falha no processo ou dispositivo de coleta de dados
- Limitações do dispositivo de coleta
- Má fé
- Mudança de conceito

Possíveis consequências de erros

- Valores de atributos ou de exemplos inteiros podem ser perdidos
- Obtenção de exemplos
  - Espúrios ou duplicados
    - Ex.: diferentes registros para mesma pessoa que morou em endereços diferentes
  - **Inconsistentes**
    - Ex.: engenheiro de 3 anos de idade

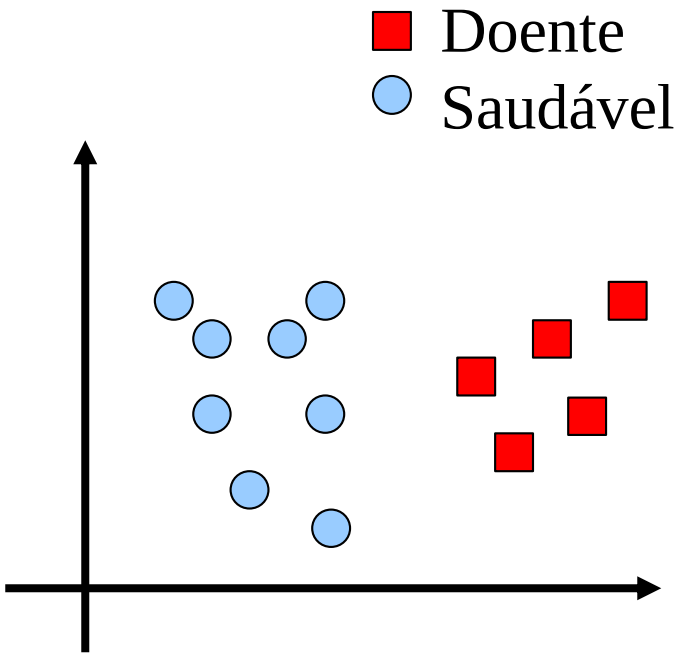
Limpeza

- **Correção de erros** detectados nos dados **deve lidar com**:
  - Atributos com ruídos
  - *Outliers*
  - Atributos com valores ausentes
  - Atributos e exemplos com valores inconsistentes
  - Atributos e exemplos redundantes

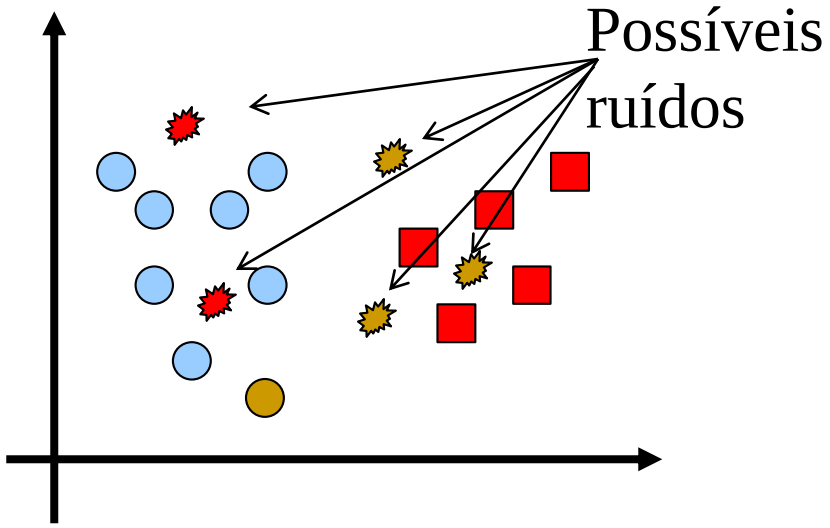
Ruídos

- **Podem levar a um super-ajuste do modelo** obtido
- Não é possível ter certeza de um valor ter ruído
  - Tem-se apenas um indício
    - A menos que valor seja inconsistente
  - *Outliers* podem sugerir a presença de ruído
- Nos atributos de entrada ou no atributo alvo
  - Consequências diferentes

Exemplo



Dados sem ruído

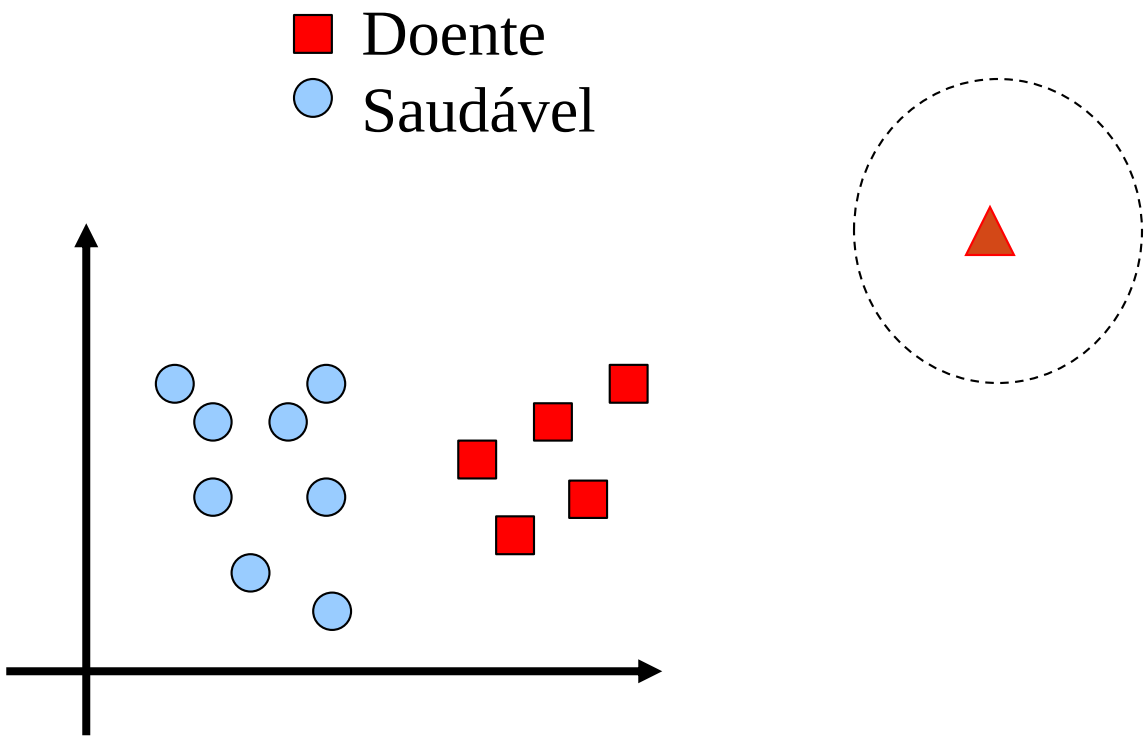


Dados com possíveis ruídos

Outliers

- Várias definições
  - Exemplos ou valores anômalos
    - Exemplos que **têm características diferentes da grande maioria dos demais** exemplos
      - **Valor(es)** de um atributo que **destoa(m) dos valores típicos** para o atributo
- Ao contrário de ruídos, ***outliers*** **podem ser exemplos ou valores legítimos**
  - Em várias aplicações, objetivo é encontrar *outliers*

Outliers



Valores ausentes

- Não é raro um exemplo não ter valores para um ou mais atributos
- Possíveis causas:

– Atributo não foi considerado quando os primeiros dados foram coletados
– Desconhecimento do valor do atributo por ocasião do preenchimento
– Distração, mal entendido ou declinamento na hora do preenchimento
– Não necessidade ou obrigação de apresentar um valor para atributo(s) de algumas instâncias
– Inexistência de valor para o atributo em algumas instâncias
– Problema com dispositivo / processo de coleta

Exemplo

- Valores ausentes

Nome	Febre	Enjoo	Batimentos	Dor	Salário	Diagnóstico
João	sim	sim	baixo	sim	1000	doente
	não	não	baixo	não	1100	saudável
Maria	sim	sim		não	600	saudável
José	sim	não	baixo	sim		doente
Sérgio	não	não	baixo	não	1100	saudável
Ana	sim	não	alto	sim	1800	saudável
Leila		não	alto		900	doente
Marta	sim	não	baixo	sim	2000	doente

Valores ausentes

- Alternativas
  - **Ignorar** valores ausentes
    - Utilizar apenas os valores que estão presentes
      - Ex.: Menos atributos no cálculo da distância entre exemplos
    - Modificar algoritmo para lidar com valores ausentes
  - **Descartar exemplos** com atributos sem valores
  - **Preenchimento de atributos** sem valores

Valores Ausentes
Descarte de exemplos

- Geralmente empregado quando
  - Um dos atributos ausentes é o atributo classe
  - Exemplo tem muitos valores ausentes
- Não é indicado quando:
  - Ocorre com poucos atributos do exemplo
  - Há risco de descartar dados importantes

Valores Ausentes

Descarte de exemplos

Preenchimento de valor

- Alternativa mais utilizada:
  - Utilizar método ou heurística para sugerir valores automaticamente
    - Diferentes abordagens podem ser seguidas

Valores Ausentes
Descarte de exemplos
Preenchimento de valor

Abordagens para preenchimento

- Criação de um novo valor
  - Dados categóricos nominais (sem ordem)
- **Média** (mediana, moda) **de todos os valores do atributo**
  - Para série de valores, entre valores anterior e posterior
  - Moda = valor ou intervalo mais frequente

Valores Ausentes

Descarte de exemplos

Preenchimento de valor

Abordagens para preenchimento

- **Média** (mediana, moda) dos valores dos atributos
  - Dos **exemplos mais próximos** e/ou da mesma classe
- **Valor induzido por algum estimador**
  - Valor presente em exemplos semelhantes
- **Criação de um novo atributo**
  - Marcando exemplos em que um atributo tinha valor ausente

Valores ausentes

- Observações
  - Em alguns casos, a ausência de valor é uma informação importante sobre o exemplo
  - Existem situações em que o valor precisa estar ausente
    - Ex.: Atributo número do apartamento para quem mora em uma casa
    - Ao invés de ausente, é um valor inexistente
- Difícil tratar de forma automática

Exercício

- Tratar dos valores ausentes da tabela abaixo

Nome	Profissão	Nível	Peso	Altura	Salário	Situação
João	Encanador	Médio	70	180	3000	adimplente
Lia		Superior	200	174	7000	inadimplente
Maria	Advogado	Médio		180	600	adimplente
José	Médico	Superior	100		2000	inadimplente
Sérgio	Bancário		82	178	5000	inadimplente
Ana	Professor	Fundam.	77	188	1800	adimplente
Luísa	Médico	Superior	100	36	2000	inadimplente
José	Médico	Médio	340		800	

Valores inconsistentes

- Dados podem conter valores inconsistentes
  - **Atributos preditivos**
    - Ex. Código postal invalido para uma cidade
      - Erro / engano
      - Proposital (fraude)
  - **Atributo alvo**
    - Podem levar a exemplos conflitantes (ambiguidade)
      - Ex.: valores iguais para atributos preditivos e diferentes para atributo alvo

Valores inconsistentes

- Algumas inconsistências são de fácil detecção
  - Violação de relações conhecidas entre atributos
    - Ex.: Valor de atributo A é sempre menor que valor de atributo B
  - Valor inválido para o atributo
    - Ex.: altura com valor negativo
  - Em outros casos, informações adicionais precisam ser consideradas

Exemplo

- Exemplos inconsistentes

Nome	Febre	Enjoo	Batimentos	Dor	Salário	Diagnóstico
João	sim	sim	baixo	sim	1000	doente
Pedro	não	não	baixo	não	1100	saudável
Maria	sim	sim	alto	não	600	saudável
José	sim	não	baixo	sim	2000	doente
Sérgio	não	não	baixo	não	1100	doente
Ana	sim	não	alto	sim	1800	saudável
Leila	não	não	alto	sim	900	doente
Marta	sim	não	alto	sim	3000	doente

Exemplos redundantes

- Não trazem informação nova
- Exemplos (quase) duplicados
  - Ex.: Pessoas em diferentes BDs com mesmo nome, mas endereço com pequenas diferenças
- Deduplicação
  - Detectar e eliminar (ou combinar) duplicações
  - Cuidado para não eliminar ou combinar exemplos que representam dados diferentes

Exemplo

Nome	Febre	Enjoo	Batimentos	Dor	Salário	Diagnóstico
João	sim	sim	baixo	sim	1000	doente
Segio	não	não	baixo	não	1100	saudável
Maria	sim	sim	alto	não	600	saudável
José	sim	não	baixo	sim	2000	doente
Sérgio	não	não	baixo	não	1100	saudável
Ana	sim	não	alto	sim	1800	saudável
Leila	não	não	alto	sim	900	doente
Marta	sim	não	baixo	sim	2000	doente

Exercício

- Definir problemas existentes na tabela abaixo:

Nome	Profissão	Nível	Peso	Altura	Salário	Situação
João	Encanador		70	180	3000	adimplente
Lia	Médico	Superior	200	174	7000	inadimplente
Maria	Advogado	Médio	90	180	600	adimplente
José	Médico	Superior	200	174	7000	inadimplente
Sérgio	Bancário	Superior	82	178	5000	inadimplente
Ana	Professor	Fundam.	77	188	1800	adimplente
Luísa	Médico	Superior	100	-6	2000	inadimplente

Dados desbalanceados

- **Quando número de exemplos varia para as diferentes classes**
  - Natural em alguns domínios
  - 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)**
- Uma alternativa: balanceamento artificial

Transformação de dados

- **Mudam o tipo de um atributo**
  - Conversão de valores simbólicos para numéricos
  - Binarização
  - Conversão de valores numéricos para simbólicos
  - Normalização de valores numéricos
  - Tradução de atributos

Conversão de valores simbólicos

- Algumas técnicas trabalham apenas com valores numéricos
  - Valores simbólicos precisam ser convertidos para numéricos
- Conversão depende de:
  - **Ordenação dos valores**
    - Se existe (ordinal) ou não
  - **Número de valores**
    - Se igual a 2 (binários) ou maior que 2

Conversão ordinal para numérico

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

Binarização

- Valores consecutivos diferem em 1 bit
- Codificar cada valor por um vetor binário
  - Código cinza:
    - 000, 010, 011, 001, 101, 111, 110, 100
  - Código termômetro:
    - 001, 011, 111

Código cinza

- Existem vários códigos cinza
  - Não é único
- Um código cinza para 3 bits:
  - 000, 010, 011, 001, 101, 111, 110, 100
- 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

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 nominal para binário

- **Codificações**
  - **1-de-n**
    - Codificação canônica
    - Fácil calcular moda = posição com maior número de valores 1
    - Valores nominais podem gerar vetores longos
  - **m-de-n**
    - Dos n valores, m são iguais a 1
    - Vários códigos

Conversão numérico para ordinal

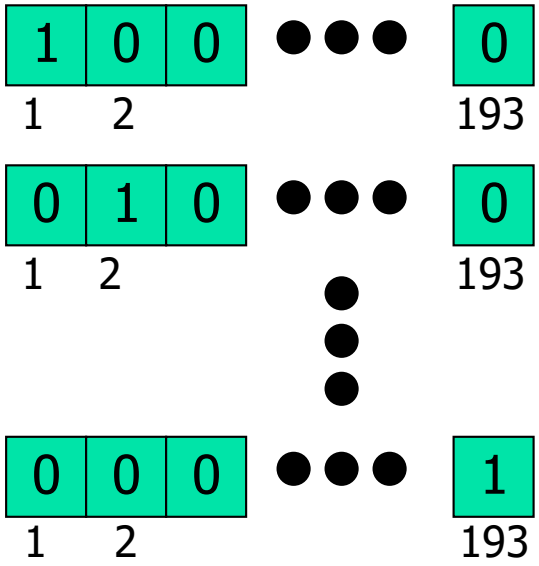
- **Discretização de valores**
  - Transformar valores numéricos em intervalos (ou categorias)
- **Sub-tarefas**
  - Definição do **número de categorias**
    - Geralmente feito pelo usuário
  - Definição de **como mapear valores** dos atributos contínuos **para essas categorias**
    - Por frequência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

Pseudo códigos

- 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 4 (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

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

Funções simples

- Uma **função** matemática simples é **aplicada a cada valor do atributo**
  - Possíveis transformações para atributo x:
    - x^k, log(x), e^x, $\sqrt{x}$, 1/x, sqrt(x), seno(x) e |x|
  - Função $\log_{10}$
    - Usada para comprimir atributos com um grande intervalo de possíveis valores

Normalização

- Faz com que conjunto de valores de um atributo tenha uma dada propriedade
- Alternativas
  - Pela amplitude
    - Re-escala
    - Padronização
  - Pela distribuição

Re-escala

- Para re-escalar 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

- Re-escalar os valores 12, 5, 4, 10, 20, 3 para:
  - O intervalo [-1, +1]
  - O intervalo [-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 re-escala 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 re-escalar
- Em algumas aplicações
  - Atributos mais importantes podem ter limites maiores

Normalização pela distribuição

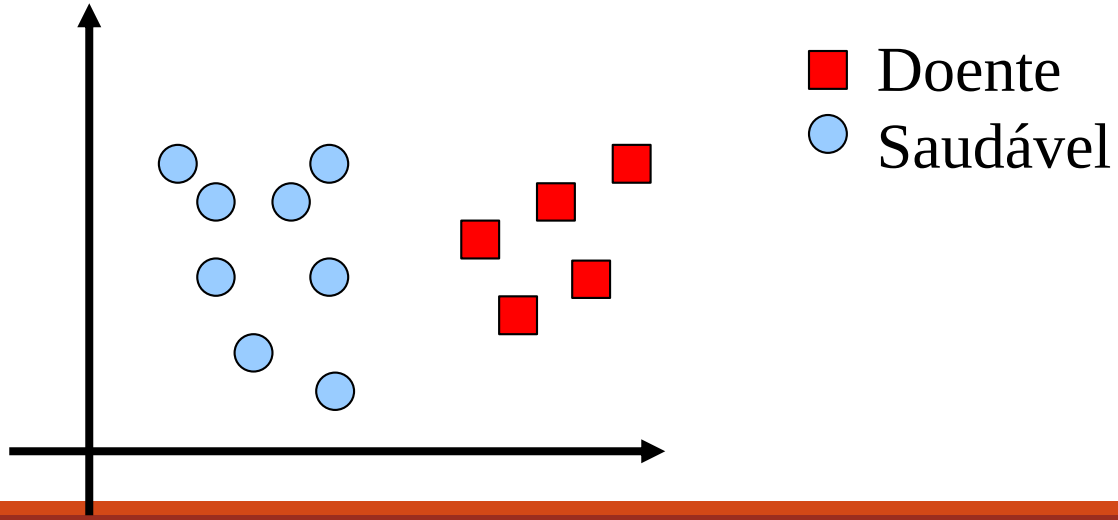
- Muda distribuição de valores de um atributo
  - Ex.: função *log*, valor absoluto
    - Normalização para valor absoluto
      - Supor que apenas a magnitude do valor de um atributo é importante
        - Converter valor de todos os atributos para o valor positivo correspondente
          - -4, 5 e –2 se tornam 4, 5 e 2

Tradução

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

Distribuição de dados

- Supor que dados são representados por pontos em um espaço d n-dimensões
  - Valores dos atributos são os valores das coordenadas



Distribuição de dados

- Número de possíveis exemplos cresce exponencialmente com o aumento de atributos
  - Espaço formado por 1 atributo com 10 possíveis valores: 10 possíveis exemplos
  - Espaço formado por 5 atributos com 10 possíveis valores: 10^5 possíveis exemplos
  - Problemas com poucos exemplos e muitos atributos:
    - Dados se tornam muito esparsos

Dados esparsos

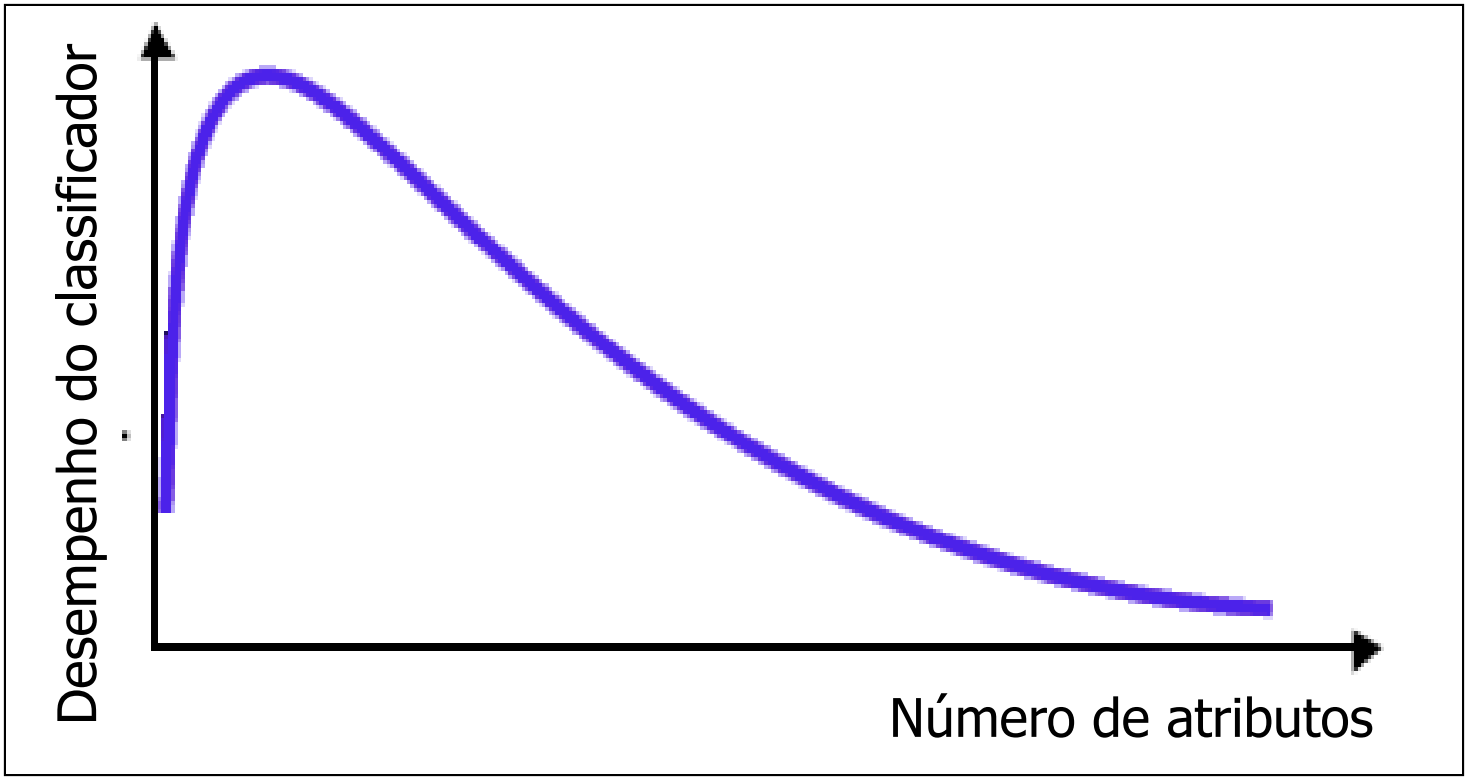
- **Ausência de exemplos em várias das regiões do espaço de exemplos**
- Distâncias entre exemplos convergem para um mesmo valor
  - Exemplos tendem a ser equidistante
  - Prejudica o desempenho de algoritmos de AM baseados em distância

Maldição da dimensionalidade

- Número de exemplos necessários para manter desempenho de um modelo
  - Cresce exponencialmente com o número de atributos
- Na prática, difícil o número de exemplos disponíveis
- Alternativa: redução de dimensionalidade

laboratório de análises Fleury

Maldição da dimensionalidade



Redução de dimensionalidade

- Alguns conjuntos de dados podem ter um número muito grande de atributos
  - Tabela com frequência de cada palavra que aparece em um texto
  - Tabela com dados de expressão gênica
- Reduzir dimensão
  - Agregação de atributos
    - Criar novos atributos que são uma combinação dos atributos originais
  - Seleção de atributos

Seleção de atributos

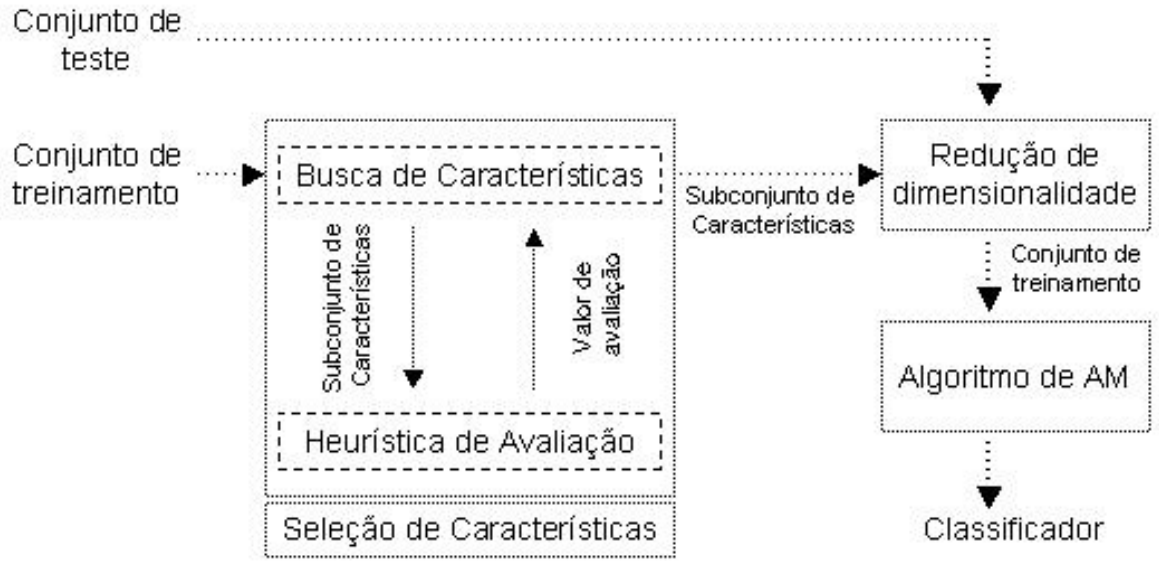
- Permite
  - Identificar atributos importantes
  - Melhorar desempenho de algoritmo de para indução de modelos
  - Minimizar os efeitos de ruídos
  - Reduzir custo de coleta de dados

Seleção de atributos

- **Abordagens**
  - Embutida
    - Seleção é feita pelo algoritmo de Aprendizado de Máquina (AM)
  - Filtro
  - Wrapper
- **Heurísticas**
  - **Ordenação**
  - **Subconjunto**

Filtros

- **Seleção de atributos independe do algoritmo de AM utilizado**
  - Ex.: verifica co-relação entre atributos



Vantagens

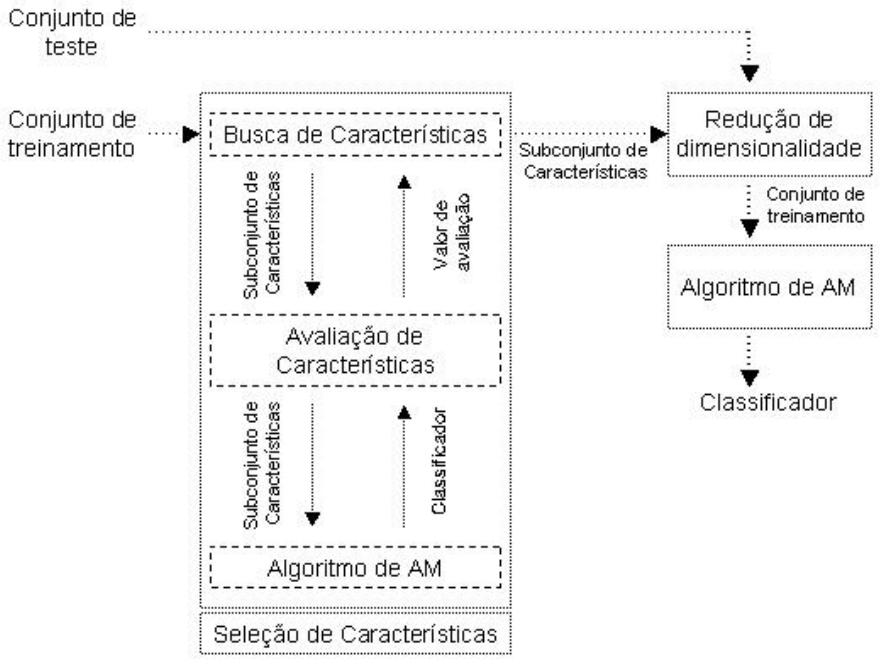
- **Não depende do algoritmo de AM**
  - Os atributos selecionados podem ser utilizados por diferentes algoritmos de AM
- **Baixo custo computacional**
  - Podem ser muito rápidos
- **Conseguem lidar** de forma eficiente com uma **grande quantidade de dados**

Desvantagens

- Ignora interação com o algoritmo de AM
  - Não considerar o viés do algoritmo AM usado pode levar a modelos pouco eficientes para os dados

Wrappers

- Utilizam o **algoritmo de AM para selecionar** atributos
  - Ex. Atributos que levam a menos erros de classificação para um algoritmo de AM



Vantagens

- Melhor conjunto de atributos para um dado algoritmo de AM
- Pode selecionar também melhor número de atributos
- Geralmente melhora desempenho obtido pelo algoritmo de AM

Desvantagens

- Risco de *overfitting*
- Desempenho depende do algoritmo de AM
- Precisa ser repetido quando um novo algoritmo de AM for utilizado
- Custo computacional elevado
  - Por causa do grande número de execuções do algoritmo de AM
    - Nem sempre, existem estratégias eficientes

Ordenação X Seleção

Seleção

Atributos originais

1	2	3	4	5	6	7	8	9	10

Atributos ordenados

4	7	2	6	9	1	10	5	8	3

Atributos Selecionados

4	7	**2**	**6**	9	**1**	10	**5**	8	3

\+ -

Ordenação

Atributos originais

1	2	3	4	5	6	7	8	9	10

Atributos ordenados

4	7	2	6	9	1	10	5	8	3

Atributos Selecionados

4	7	**2**	6	9	**1**	10	**5**	8	**3**

\+ -

Exercício

- Ordenar os atributos mais importantes para o diagnóstico de pacientes

Febre	Enjôo	Batim.	Dor	Diagnóstico
1	0	1	1	0
0	1	0	1	1
1	1	1	0	1
1	0	0	1	0
1	0	1	1	1
0	0	1	0	0

Exercício

Febre	Enjôo	Batim.	Dor	Diagnóstico
1	0	1	1	0
0	1	0	1	1
1	1	1	0	1
1	0	0	1	0
1	0	1	1	1
0	0	1	0	0

Escores:

Febre: 3/6

Enjôo: 5/6

Batimentos: 4/6

Dores: 2/6

Ranking:

1- Enjôo

2- Batimentos

3- Febre

4 - Dores

Exercício

- Selecionar o subconjunto de 2 atributos mais importante para o diagnóstico de pacientes

Febre	Enjôo	Batim.	Dor	Diagnóstico
1	0	1	1	0
0	1	0	1	1
1	1	1	0	1
1	0	0	1	0
1	0	1	1	1
0	0	1	0	0

Exercício

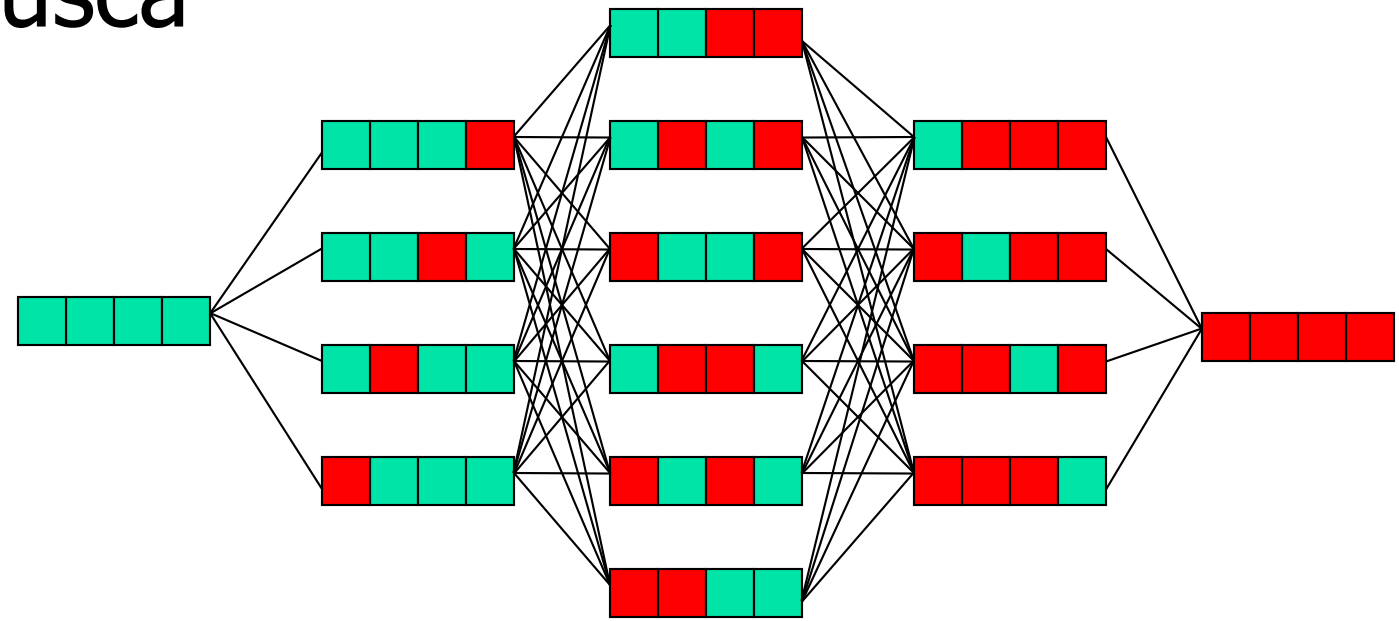
Febre	Enjôo	Batim.	Dor	Diagnóstico
1	0	1	1	0
0	1	0	1	1
1	1	1	0	1
1	0	0	1	0
1	0	1	1	1
0	0	1	0	0

Febre ⊗ Batimentos

└ coincide

Seleção de subconjunto

Busca



Espaço de busca com quatro atributos (dimensões)

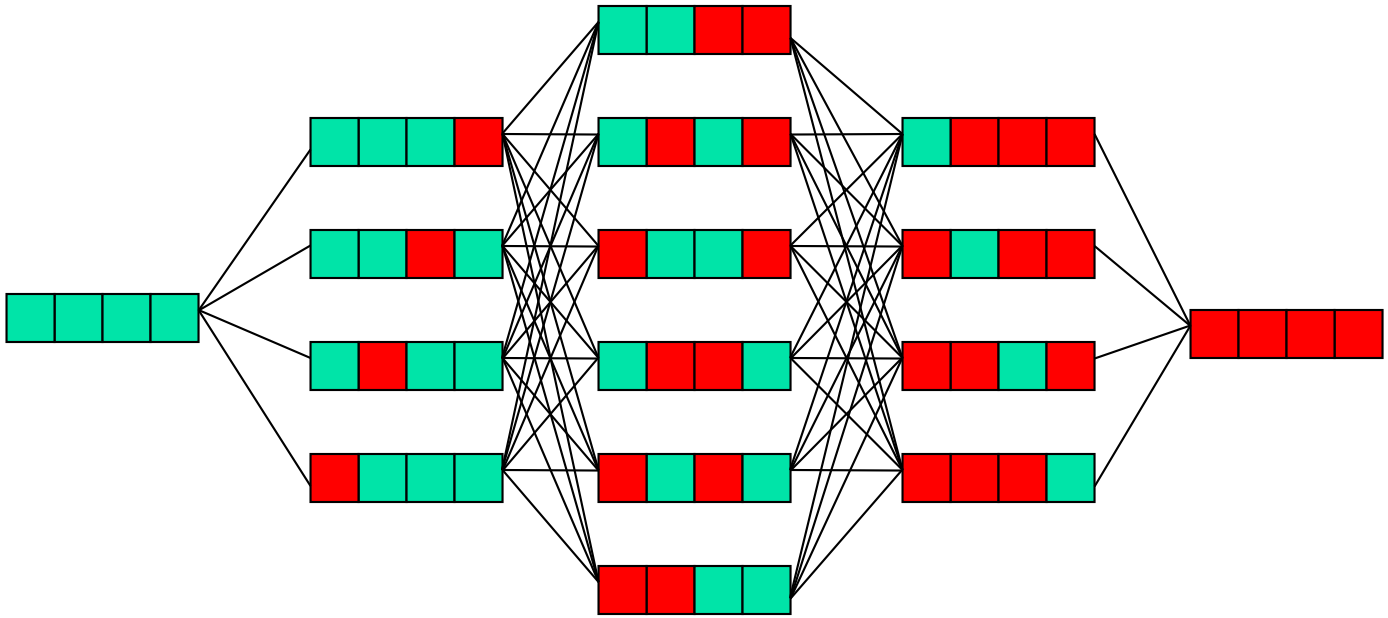
Seleção de subconjunto

- Quatro aspectos precisam ser tratados:
  - Ponto de início da busca e da geração de subconjuntos
  - Estratégia de busca
  - Estratégia de avaliação
  - Critério de parada

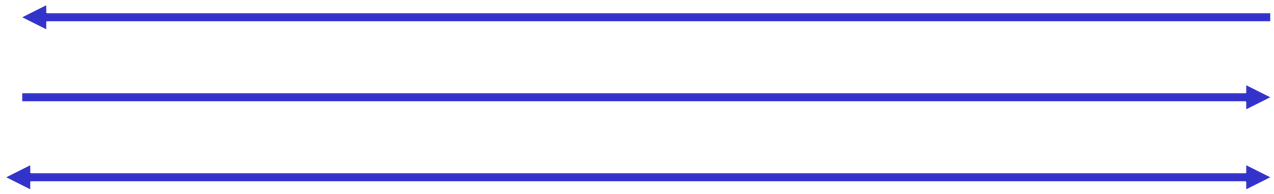
Geração de subconjuntos

- Existem quatro alternativas
  - Geração para trás (*backward generation*)
    - Começa com todos os atributos e remove um por vez
  - Geração para frente (*forward generation*)
    - Começa sem nenhum atributo e inclui um atributo por vez
  - Geração bidirecional (*biderectional generation*)
    - Busca pode variar direção e atributos podem ser adicionados e removidos
  - Geração estocástica (*random generation*)
    - Ponto de partida da busca e atributos a serem removidos ou adicionados são decididos de forma estocástica

Geração de subconjuntos



Backward
Feedforward
Bidirecional e
Estocástico



Estratégia de busca

- Define o algoritmo usado para realizar a busca
  - Busca completa (exponencial ou exaustiva)
    - Avalia todos os possíveis subconjuntos
  - Busca heurística (sequencial)
    - Utiliza regras e métodos para conduzir a busca
    - Não garante que uma solução ótima seja encontrada
  - Busca não-determinística
    - Relacionado com a geração estocástica
    - Geralmente utiliza metaheurística
      - Regra moldada para o problema investigado
    - Não garante que uma solução ótima seja encontrada

Considerações finais

- Pré-processamento
- Amostragem
- Limpeza de dados
- Transformação de dados
- Maldição da dimensionalidade
- Redução do número de atributos

Exercício

- Escolher 3 conjuntos de dados da UCI e, para cada conjunto
  - Aplicar uma técnica de amostragem dos dados
  - Aplicar técnicas para limpeza de dados
  - Criar uma variação com todos os atributos numéricos
  - Criar uma variação com todos os atributos simbólicos
  - Selecionar atributos usando uma técnica baseada em filtro e uma baseada em wrapper

Perguntas