

Aprendizado de Máquina: Pré-processamento de dados

Ana Carolina Lorena e Filipe Verri

ITA

Março de 2021

Outline

- 1 Introdução
- 2 Integração de Dados
- 3 Eliminação manual de atributos
- 4 Amostragem de dados
- 5 Dados desbalanceados
- 6 Limpeza de Dados
- 7 Transformação de Dados
- 8 Redução de dimensionalidade
- 9 Referências

Pré-processamento de dados

- Desempenho de técnicas de AM é afetado pela qualidade dos dados
 - Conjuntos de dados podem ter diferentes características, dimensões ou formatos
 - Atributos numéricos vs simbólicos
 - Limpos vs com ruídos e imperfeições (valores incorretos, inconsistentes, duplicados ou ausentes)
 - Atributos independentes vs relacionados
 - Poucos vs muitos objetos e/ou atributos

Pré-processamento:

Minimizar/eliminar problemas nos dados; tornar dados mais adequados para uso por um determinado algoritmo de AM

Pré-processamento

Benefícios:

- **Facilitar o uso** posterior de técnicas de AM
 - Ou tornar mais adequado para a técnica
 - Ex. algumas trabalham somente com entradas numéricas
- Obtenção de **modelos mais fiéis** à distribuição dos dados
 - Melhorar qualidade dos dados
- **Redução de complexidade computacional**
 - Tempo e custo
- Tornar **mais fáceis e rápidos** ajustes de parâmetros
- Facilitar a **interpretação** dos padrões extraídos

Pré-processamento

Grupos de tarefas de pré-processamento:

- Integração de dados
- Eliminação manual de atributos
- Amostragem de dados
- Redução de dimensionalidade
- Balanceamento de dados
- Limpeza de dados
- Transformação de dados

Observação:

Não existe ordem fixa para aplicação das diferentes técnicas de pré-processamento

Pré-processamento

Integração de dados

Dados podem estar em mais de um banco de dados

Diferentes conjuntos de dados integrados: pode levar a inconsistências e redundâncias

Eliminação manual de atributos

Alguns atributos não possuem relação com o problema solucionado (ex RG em diagnóstico)

Amostragem de dados e redução de dimensionalidade

Algoritmos de AM podem ter dificuldades quando precisam lidar com uma grande quantidade de dados (objetos, atributos ou ambos)

Ex. redundância e inconsistência

Pré-processamento

Balanceamento de dados

Conjunto de dados desbalanceado: proporção de exemplos em algumas classes pode ser muito maior do que em outras

Maioria dos algoritmos de AM tem dificuldade neste cenário

Limpeza de dados

Presença de ruídos, dados incompletos e inconsistentes pode afetar desempenho dos algoritmos

Alguns são incapazes de lidar com dados incompletos

Transformação de dados

Vários algoritmos de AM têm dificuldades em usar os dados em seu formato original

Ex. transformação de valores simbólicos para numéricos

Outline

- 1 Introdução
- 2 Integração de Dados
- 3 Eliminação manual de atributos
- 4 Amostragem de dados
- 5 Dados desbalanceados
- 6 Limpeza de Dados
- 7 Transformação de Dados
- 8 Redução de dimensionalidade
- 9 Referências

Integração de dados

- Dados podem vir de diferentes fontes
 - Requer integração de diferentes conjuntos de dados
 - Cada um pode ter atributos diferentes para os mesmos objetos

Identificação de entidade

Identificar os objetos em comum

Normalmente por busca por atributos comuns nos conjuntos, de preferência que tenham valor único para cada objeto

Ex. identificação de paciente

Integração de dados

Dificuldades:

- Atributos correspondentes com nomes diferentes
- Dados podem ter sido atualizados em momentos diferentes

Metadados:

Comum usar **metadados** para minimizar esses problemas

Metadados: dados sobre os dados, que descrevem suas principais características

Outline

- 1 Introdução
- 2 Integração de Dados
- 3 Eliminação manual de atributos**
- 4 Amostragem de dados
- 5 Dados desbalanceados
- 6 Limpeza de Dados
- 7 Transformação de Dados
- 8 Redução de dimensionalidade
- 9 Referências

Eliminação manual de atributos

- Há atributos que claramente não contribuem para o aprendizado
 - Ex. conjunto de dados hospital

Id.	Nome	Idade	Sexo	Peso	Manchas	Temp. #	Int. Est.	Diagnóstico
4201	João	28	M	79	Grandes	38,0	2	SP Doente
3217	Maria	18	F	67	Pequenas	39,5	4	MG Doente
4039	Luiz	49	M	92	Grandes	38,0	2	RS Saudável
1920	José	18	M	43	Grandes	38,5	20	MG Doente
4340	Cláudia	21	F	52	Médias	37,6	1	PE Saudável
2301	Ana	22	F	72	Pequenas	38,0	3	RJ Doente
1322	Marta	19	F	87	Grandes	39,0	6	AM Doente
3027	Paulo	34	M	67	Médias	38,4	2	GO Saudável

Atributos Id. e Nome:

Não contribuem para estimar se um paciente tem doença ou não

Eliminação manual de atributos

- Normalmente, o conjunto de atributos é definido de acordo com a experiência de especialista
 - Ex. conjunto de dados hospital

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Est.	Diagnóstico
28	M	79	Grandes	38,0	2	SP	Doente
18	F	67	Pequenas	39,5	4	MG	Doente
49	M	92	Grandes	38,0	2	RS	Saudável
18	M	43	Grandes	38,5	20	MG	Doente
21	F	52	Médias	37,6	1	PE	Saudável
22	F	72	Pequenas	38,0	3	RJ	Doente
19	F	87	Grandes	39,0	6	AM	Doente
34	M	67	Médias	38,4	2	GO	Saudável

Estado:

Médico pode decidir que atributo associado ao estado de origem do paciente também não é relevante para seu diagnóstico clínico, dependendo da doença

Eliminação manual de atributos

- Ex. conjunto de dados hospital
 - Após eliminação manual dos atributos

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
28	M	79	Grandes	38,0	2	Doente
18	F	67	Pequenas	39,5	4	Doente
49	M	92	Grandes	38,0	2	Saudável
18	M	43	Grandes	38,5	20	Doente
21	F	52	Médias	37,6	1	Saudável
22	F	72	Pequenas	38,0	3	Doente
19	F	87	Grandes	39,0	6	Doente
34	M	67	Médias	38,4	2	Saudável

Eliminação manual de atributos

- Outro atributo irrelevante facilmente detectado:
 - Atributo que possui o mesmo valor para todos objetos
 - Não traz informação para ajudar a distinguí-los
- Há ainda atributos irrelevantes de identificação não tão clara
 - Técnicas de [seleção de atributos](#) podem ajudar a identificar

Outline

- 1 Introdução
- 2 Integração de Dados
- 3 Eliminação manual de atributos
- 4 Amostragem de dados**
- 5 Dados desbalanceados
- 6 Limpeza de Dados
- 7 Transformação de Dados
- 8 Redução de dimensionalidade
- 9 Referências

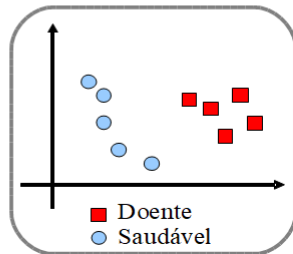
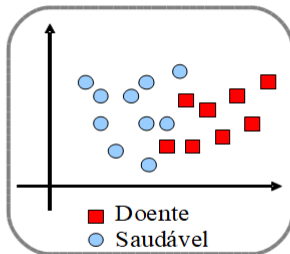
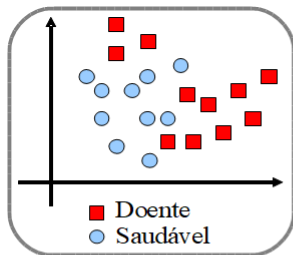
Amostragem de dados

- Algoritmos de AM podem ter dificuldades em lidar com um número grande de objetos
 - Saturação de memória
 - Aumento do tempo computacional para ajustar os parâmetros do modelo
- Contudo, quanto mais dados, maior tende a ser a acurácia do modelo

Atenção:

Procurar balanço entre eficiência computacional e acurácia do modelo

Amostragem de dados



Amostragem de dados

- Amostra dos dados
 - Pode levar ao mesmo desempenho do conjunto completo, a menor custo computacional
 - Deve ser representativa

Amostra representativa:

- Aproximadamente as mesmas propriedades do conjunto de dados original
- Fornecer uma estimativa da informação contida na população original
- Uso deve ter efeito semelhante ao de toda a população
- Permitir conclusão do todo a partir de uma parte

Amostragem de dados

Técnicas de amostragem:

Amostragem aleatória simples

- Variações: **com** e **sem reposição** de exemplos (semelhantes quando tamanho da amostra é bem menor que o do conjunto original)

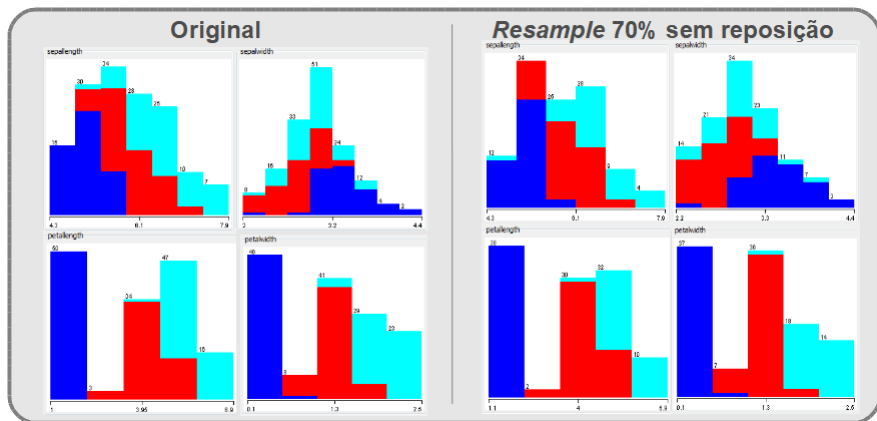
Amostragem estratificada

- Quando classes têm propriedades diferentes (ex. números de objetos diferentes)
- Variações: manter o **mesmo** número de objetos para cada classe ou manter o número **proporcional** ao original

Amostragem progressiva

- Começa com amostra pequena e vai aumentando enquanto acurácia preditiva continuar a melhorar

Amostragem de dados



Amostragem de dados

Especialista também pode auxiliar a decidir subconjunto de objetos a serem usados

- Ex. somente pacientes do sexo feminino são acometidos por uma doença de interesse

Outline

- 1 Introdução
- 2 Integração de Dados
- 3 Eliminação manual de atributos
- 4 Amostragem de dados
- 5 Dados desbalanceados**
- 6 Limpeza de Dados
- 7 Transformação de Dados
- 8 Redução de dimensionalidade
- 9 Referências

Dados desbalanceados

- Tópicos de **classificação** de dados
 - Número de objetos varia para as diferentes classes
 - Típico da aplicação
 - Ex. 80% dos pacientes que vão a um hospital estão doentes
 - Ou problema na geração/coleta dos dados

Classe majoritária

Contém a maior parte dos exemplos

Classe minoritária

Tem o menor número de exemplos no conjunto

Dados Desbalanceados

- Acurácia preditiva de classificador deve ser maior que a obtida atribuindo um novo objeto à classe majoritária
 - Vários algoritmos de AM têm o desempenho prejudicado para dados muito desbalanceados
 - Tendem a favorecer a classificação na classe majoritária

Dados desbalanceados

- Alternativas para lidar com dados desbalanceados:
 - Obter novos dados para a classe minoritária (na maioria dos casos não é possível...)
 - Balancear artificialmente o conjunto de dados, redefinindo o tamanho do conjunto de dados
 - Usar diferentes custos de classificação por classe
 - Induzir um modelo para uma classe

Dados desbalanceados

- Técnicas de rebalanceamento:

Redefinir tamanho do conjunto de dados

- Acréscimo/eliminação de exemplos na classe minoritária/majoritária
- **Acréscimo:** risco de objetos que não representam situações reais e *overfitting*
- **Eliminação:** risco de perda de objetos importantes e *underfitting*

Usar custos de classificação diferentes para as classes

- **Dificuldades:** definição dos custos, incorporar custos em alguns algoritmos de AM
- Pode apresentar baixo desempenho quando muitos objetos da classe majoritária são semelhantes

Dados desbalanceados

- Técnicas de rebalanceamento:

Induzir modelo para uma única classe

- Técnicas de classificação para uma classe, treinadas usando somente exemplos de uma classe
- Aprendem classe(s) separadamente

Outros usos de *one-class classification*:

- Detecção de novidades (ex. falhas em máquinas)
- Detecção de *outliers*
- Comparação de conjuntos de dados (evitar retreinar classificadores para dados semelhantes)

Outline

- 1 Introdução
- 2 Integração de Dados
- 3 Eliminação manual de atributos
- 4 Amostragem de dados
- 5 Dados desbalanceados
- 6 Limpeza de Dados**
- 7 Transformação de Dados
- 8 Redução de dimensionalidade
- 9 Referências

Limpeza de dados

- Qualidade dos dados:
 - Em geral, dados não foram produzidos para uso em AM
 - Exemplos de problemas:
 - **Ruídos:** erros ou valores diferentes do esperado
 - **Inconsistências:** não combinam/contradizem valores de outros atributos no mesmo objeto
 - **Redundâncias:** objetos/atributos com mesmos valores
 - **Dados incompletos:** ausência de valores de atributos

Principal dificuldade:

Detecção de dados ruidosos

Limpeza de dados

- Exemplos de causas de erros:
 - Falha humana
 - Falha no processo de coleta de dados
 - Limitações do dispositivo de medição
 - Má fé
 - Valor de atributo muda com o tempo

Atenção:

Alguns erros são sistemáticos e mais fáceis de detectar e corrigir

Limpeza de dados

- Consequências:
 - Valores ou objetos inteiros podem ser perdidos
 - Objetos espúrios ou duplicados podem ser obtidos
 - Ex. diferentes registros para mesma pessoa que morou em endereços diferentes
 - Inconsistências
 - Ex.: pessoa com 2 m pesando 10 Kg

Limpeza de dados

- Algumas técnicas de AM conseguem lidar com algumas imperfeições nos dados
 - Outras não conseguem ou apresentam dificuldades
- Porém de forma geral, qualidade das análises pode ser deteriorada

Importância:

Todas as técnicas se beneficiam de melhora na qualidade dos dados, que pode ser obtida por meio de etapa de limpeza

Dados incompletos

- Ausência de valores para alguns atributos de alguns objetos
 - Ex. conjunto de dados hospital

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
--	M	79	--	38,0	--	Doente
18	F	67	Pequenas	39,5	4	Doente
49	M	92	Grandes	38,0	2	Saudável
18	--	43	Grandes	38,5	20	Doente
21	F	52	Médias	37,6	1	Saudável
22	F	72	Pequenas	38,0	3	Doente
--	F	87	Grandes	39,0	6	Doente
34	M	67	Médias	38,4	2	Saudável

Dados incompletos

- Possíveis causas:
 - Atributo não era importante quando primeiros dados foram coletados (ex. e-mail na década de 90)
 - Desconhecimento do valor do atributo (ex. não saber tipo sanguíneo de paciente em seu cadastro)
 - Falta de necessidade/obrigação de apresentar valor (ex. salário em hospital)
 - Inexistência de valor para o atributo (ex. número de partos para pacientes do sexo masculino)
 - Problema com equipamento para coleta, transmissão e armazenamento de dados

Dados incompletos

- Algumas técnicas de AM são incapazes de lidar com valores ausentes
 - Geram erro de execução
- Alternativas para lidar com valores ausentes:
 - Eliminar os objetos com valores ausentes
 - Definir e preencher manualmente os valores ausentes
 - Utilizar método/heurística para definir valores automaticamente
 - Empregar algoritmos de AM que lidam internamente com valores ausentes

Dados incompletos

- Técnicas:

Eliminar objetos

- Mais empregada quando classe está ausente
- Não indicada quando número de atributos com valores ausentes varia muito entre os objetos ou quando muitos objetos têm valores ausentes

Definir/preencher manualmente

- Não é factível para muitos valores ausentes

Usar heurística

- Alternativa mais usada

Dados incompletos

- Técnicas para definição automática de valores:

Criar valor “desconhecido”

- Comum a todos ou diferente para cada atributo

Utilizar média/moda/ mediana dos valores conhecidos

- Usando todos os objetos ou somente aqueles da mesma classe
- **Variação:** usar valor mais frequente entre k vizinhos mais próximos

Usar indutor para estimar o valor

- Valor a ser definido passa a ser o atributo alvo
- Usa informação dos outros atributos para inferir o ausente

Dados incompletos

- Usando média/moda
 - Ex. conjunto de dados hospital

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
27	M	79	Grandes	38,0	4	Doente
18	F	67	Pequenas	39,5	4	Doente
49	M	92	Grandes	38,0	2	Saudável
18	F	43	Grandes	38,5	20	Doente
21	F	52	Médias	37,6	1	Saudável
22	F	72	Pequenas	38,0	3	Doente
27	F	87	Grandes	39,0	6	Doente
34	M	67	Médias	38,4	2	Saudável

Pode gerar inconsistências

Ex. paciente de 2 anos com 60 kg

Dados inconsistentes

- Possuem valores conflitantes em seus atributos
 - Nos atributos de entrada
 - Ex. 3 anos de idade e 120 kg
 - Entradas iguais e saída diferente
 - Ex. conjunto de dados hospital

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
28	M	79	Grandes	38,0	2	Doente
18	F	67	Pequenas	39,5	4	Doente
49	M	92	Grandes	38,0	2	Saudável
18	M	43	Grandes	38,5	20	Doente
21	F	52	Médias	37,6	1	Saudável
22	F	72	Pequenas	38,0	3	Doente
19	F	87	Grandes	39,0	6	Doente
22	F	72	Pequenas	38,0	3	Saudável

Dados inconsistentes

- Possíveis causas:
 - **Erro**/engano
 - Presença de ruídos nos dados
 - Proposital (**fraude**)
 - **Problemas** na integração dos dados
 - Ex. conjuntos de dados com escalas diferentes para uma mesma medida

Dados 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 verificadas

Dados redundantes

- Valores que não trazem informação nova
 - **Objetos** redundantes
 - Muito semelhante(s) a outro(s) no conjunto de dados (
 - Ex.: Pessoas em diferentes BDs com mesmo endereço e pequenas diferenças nos nomes)
 - **Atributos** redundantes
 - Valor pode ser deduzido a partir do valor de um ou mais atributos
- Possíveis causas: **problemas** na coleta, entrada, armazenamento, integração ou transmissão ou desconhecimento dos dados

Dados redundantes

- Ex. conjunto de dados hospital

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
28	M	79	Grandes	38,0	2	Doente
18	F	67	Pequenas	39,5	4	Doente
49	M	92	Grandes	38,0	2	Saudável
18	F	67	Pequenas	39,5	4	Doente
21	F	52	Médias	37,6	1	Saudável
22	F	72	Pequenas	38,0	3	Doente
19	F	87	Grandes	39,0	6	Doente
34	M	67	Médias	38,4	2	Saudável

Duplicação

Dados redundantes

- Objetos redundantes participam mais de uma vez do ajuste do modelo
 - Pode assim ser considerado um perfil mais importante que o dos outros
 - Pode também aumentar custo computacional
- Passos para eliminar objetos redundantes:
 - Identificar as redundâncias
 - Eliminar as redundâncias
 - Remoção ou combinação dos valores

Dados redundantes

- **Atributo redundante:** valor pode ser estimado a partir de pelo menos um dos demais atributos
 - Atributos com a mesma informação preditiva
 - Ex. atributos idade e data de nascimento
 - Ex. atributos quantidade de vendas, valor por venda e venda total
 - Atributo redundante pode supervalorizar um dado aspecto dos dados
 - Pode também tornar mais lento o processo de indução

Atributos redundantes

São geralmente eliminados por técnicas de **seleção de atributos**

Dados redundantes

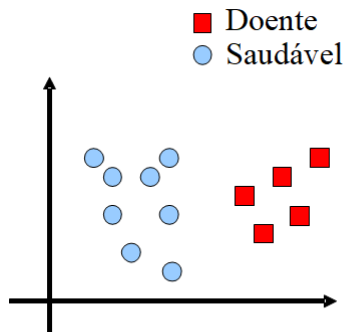
- Redundância de atributo está relacionada à sua correlação com um ou mais dos demais atributos
 - Dois atributos estão correlacionados quando têm perfil de variação semelhante para diferentes objetos
 - Ex. conjunto de dados hospital

Idade	Sexo	Peso	Manchas	Temp.	# Int.	# Vis.	Diagnóstico
28	M	79	Grandes	38,0	2	2	Doente
18	F	67	Pequenas	39,5	4	4	Doente
49	M	92	Grandes	38,0	2	2	Saudável
18	M	43	Grandes	38,5	20	20	Doente
21	F	52	Médias	37,6	1	1	Saudável
22	F	72	Pequenas	38,0	3	3	Doente
19	F	87	Grandes	39,0	6	6	Doente
34	M	67	Médias	38,4	2	2	Saudável

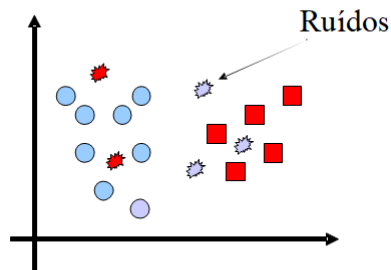
Ruídos

- Objetos que aparentemente não pertencem à distribuição que gerou os dados
- Várias causas possíveis
- Podem levar a superajuste do modelo
 - Algoritmo pode se ater às especificidades dos ruídos
- Mas eliminação pode levar à perda de informação importante
 - Algumas regiões do espaço de atributos podem não ser consideradas

Ruídos



Dados sem ruído



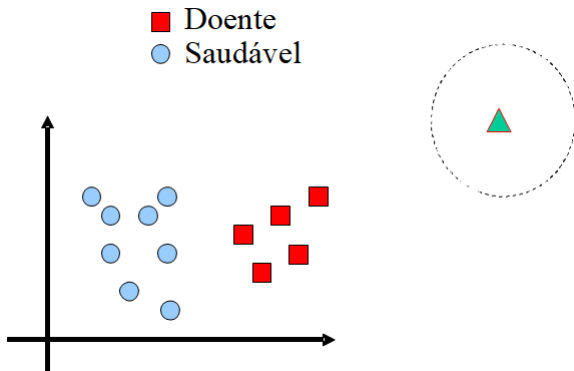
Dados com ruído

Outliers

- Valores que estão além dos limites aceitáveis ou são muito diferentes dos demais (exceções)
 - Podem ser valores legítimos
 - Ex. conjunto de dados hospital

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
28	M	79	Grandes	38,0	2	Doente
18	F	300	Pequenas	39,5	4	Doente
49	M	92	Grandes	38,0	2	Saudável
18	M	43	Pequenas	38,5	20	Doente
21	F	52	Médias	37,6	1	Saudável
22	F	72	Pequenas	38,0	3	Doente
19	F	87	Grandes	39,0	6	Doente
34	M	67	Médias	38,4	2	Saudável

Outliers



Ruídos

- Algumas técnicas de pré-processamento:
 - Técnicas baseadas em distribuição
 - Técnicas de encestamento
 - Técnicas baseadas em agrupamento dos dados
 - Técnicas baseadas em distância
 - Técnicas baseadas em regressão ou classificação

Ruídos

- Técnicas:

Baseadas em distribuição:

- Ruídos identificados como observações que diferem de uma distribuição usada na modelagem dos dados
- **Problema:** distribuição dos dados normalmente não é conhecida a priori

Encestamento:

- Suavizam valor de atributo
- 1º: ordena valores de atributo;
- 2º: divide em cestas (faixas), cada uma com o mesmo número de valores
- 3º: substitui valores em uma mesma cesta, por ex., por média/moda

Ruídos

- Técnicas:

Agrupamento:

- Agrupa objetos/atributos de acordo com semelhança
- Atributos/objetos que não formam grupo são ruídos ou *outliers*
- Objetos colocados em um grupo que pertence a outra classe também são considerados ruídos

Baseadas em distâncias:

- Presença de ruído em atributo frequentemente faz com que ele se distancie dos demais objetos de sua classe
- Verificar a que classe pertencem os vizinhos mais próximos de x
- Se são de classe diferente, x pode ser ruído ou *borderline* (próximo à fronteira de separação das classes, podem ser inseguros)

Ruídos

- Técnicas:

Baseadas em regressão/classificação:

- Usam função de regressão ou classificação para, dado um valor com ruído, estimar seu valor verdadeiro (regressão para atributo contínuo e classificação para simbólico)

Outline

- 1 Introdução
- 2 Integração de Dados
- 3 Eliminação manual de atributos
- 4 Amostragem de dados
- 5 Dados desbalanceados
- 6 Limpeza de Dados
- 7 Transformação de Dados**
- 8 Redução de dimensionalidade
- 9 Referências

Transformação de dados

- Algumas técnicas de AM são limitadas à manipulação de valores de determinado tipo
 - Apenas numéricos ou simbólicos
- Algumas técnicas de AM têm desempenho influenciado pela variação dos valores numéricos

Conversão simbólico-numérico

- Atributo simbólico com dois valores
 - Um dígito binário é suficiente
 - Ex. presença/ausência = 1/0
 - Se ordinal, 0 indica o menor valor e 1 o maior valor
- Atributo simbólico com mais valores
 - Conversão depende se o atributo é **nominal** ou **ordinal**

Conversão simbólico-numérico

- Atributo **nominal** com mais valores
 - Inexistência de relação de ordem deve ser mantida
 - Diferença entre quaisquer dois valores numéricos deve ser a mesma
 - **Codificação canônica**: uso de c bits para c valores
 - Cada posição na sequência binária corresponde a um valor possível do atributo nominal
 - Cada sequência possui apenas um bit com valor 1
 - Distância de Hamming entre quaisquer dois valores é 2

Conversão simbólico-numérico

- Ex. conjunto de dados hospital
 - Conversão de atributo Sexo para numérico

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
28	0	79	Grandes	38,0	2	Doente
18	1	67	Pequenas	39,5	4	Doente
49	0	92	Grandes	38,0	2	Saudável
18	0	43	Grandes	38,5	20	Doente
21	1	52	Médias	37,6	1	Saudável
22	1	72	Pequenas	38,0	3	Doente
19	1	87	Grandes	39,0	6	Doente
34	0	67	Médias	38,4	2	Saudável

M = 0
F = 1

Conversão simbólico-numérico

- Atributo **nominal** com mais que dois valores
 - Ex. codificação **canônica** (**1-para-c** ou **topológica**)

Atributo	Código 1-para-c
Azul	100000
Amarelo	010000
Verde	001000
Preto	000100
Marrom	000010
Branco	000001

Atenção:

Dependendo do número de valores nominais, pode gerar cadeias muito grandes de bits. Ex.: 193 nomes de países

Conversão simbólico-numérico


- Atributo **nominal** com mais que dois valores
 - Representação por **conjunto de pseudoatributos** do tipo binário, inteiro ou real
 - Ex. 193 países

Pseudoatributo	# Valores
Continente	7
PIB	1
População	1
Temp_méd_anual	1
Área	1

Conversão simbólico-numérico

- Atributo **ordinal** com mais que dois valores
 - Relação de ordem deve ser preservada
 - Ordenar valores ordinais e codificar cada um de acordo com sua posição na ordem com inteiro ou real

Atributo	Valor inteiro
Primeiro	0
Segundo	1
Terceiro	2
Quarto	3
Quinto	4
Sexto	5



Distância entre valores
varia de acordo com
proximidade entre eles

Conversão simbólico-numérico

- Ex. conjunto de dados hospital
 - Conversão de atributo ordinal Manchas

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
28	M	79	3	38,0	2	Doente
18	F	67	1	39,5	4	Doente
49	M	92	3	38,0	2	Saudável
18	M	43	3	38,5	20	Doente
21	F	52	2	37,6	1	Saudável
22	F	72	1	38,0	3	Doente
19	F	87	3	39,0	6	Doente
34	M	67	2	38,4	2	Saudável

Grandes = 3
 Médias = 2
 Pequenas = 1

Conversão simbólico-numérico

- Atributo **ordinal** com mais que dois valores
 - Se for necessário usar valores binários, pode ser utilizado:
 - **Código cinza**: correção de erros em comunicação
 - **Código termômetro**: aumento de valores se assemelha a aumento de temperatura em termômetro

Atributo	Código cinza	Código termômetro
Primeiro	000	00000
Segundo	001	00001
Terceiro	011	00011
Quarto	010	00111
Quinto	110	01111
Sexto	100	11111

Conversão numérico-simbólico

- Atributo discreto e binário: conversão é trivial
 - Associa um nome a cada valor
 - Também se são sequências binárias sem relação de ordem
- Demais casos: **discretização**
 - Transforma valores numéricos em intervalos (categorias)
 - Existem vários métodos diferentes para discretização
 - **Paramétricos**: usuário pode influenciar definição dos intervalos
 - **Não paramétricos**: usam apenas informações presentes nos valores dos atributos

Conversão numérico-simbólico

- Métodos de discretização podem ser:
 - **Supervisionados:** usa informação da classe
 - Melhor resultado, não leva a mistura de classes
 - Ex. escolher pontos de corte que maximizam pureza dos intervalos (entropia)
 - **Não supervisionados**
- Método de discretização deve definir:
 - Como mapear valores quantitativos para qualitativos
 - Tamanho dos intervalos
 - Quantidade de valores nos intervalos

Conversão numérico-simbólico

- Algumas estratégias:

Larguras iguais

- Dividir valores em subintervalos com mesma largura
- **Problema:** desempenho afetado pela presença de *outliers*

Uso de um algoritmo de agrupamento

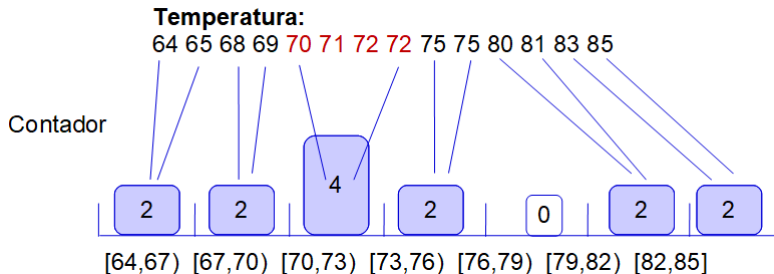
Frequências iguais

- Mesmo número de objetos em cada intervalo
- **Problema:** pode gerar intervalos de tamanhos muito diferentes

Inspeção visual

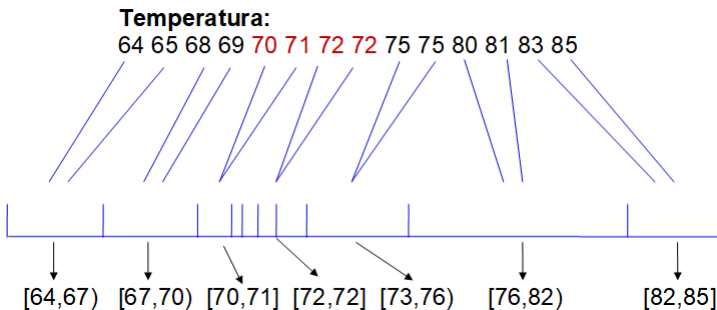
Conversão numérico-simbólico

- Ex: discretização com larguras iguais



Conversão numérico-simbólico

- Ex: discretização com frequências iguais



Transformação de atributos numéricos

- Algumas vezes é necessário transformar o valor de um atributo numérico em outro valor numérico
 - Quando o intervalo de valores são muito diferentes, levando a grande variação
 - Quando vários atributos estão em escalas diferentes
 - Para evitar que um atributo predomine sobre outro
- Porém, em alguns casos pode ser importante preservar a variação

Transformação de atributos numéricos

- Transformação é aplicada aos valores de um dado atributo de todos os objetos
- Uma transformação muito usada: **normalização**
 - Faz com que conjunto de valores de um atributo tenha uma determinada propriedade
 - Quando escalas de valores de atributos distintos são muito diferentes
 - Evita que um atributo predomine sobre o outro

Normalização

- Deve ser aplicada a cada atributo individualmente
 - Duas formas:

Por amplitude

- Por **reescala**: define nova escala (máximo e mínimo) de valores para atributos
- Por **padronização**: define um valor central e de espalhamento comuns para todos os atributos

Por distribuição

- Muda a escala de valores
- Ex. Ordena valores dos atributos e substitui cada valor por sua posição no ranking (valores 1, 5, 9, e 3 viram 1, 3, 4 e 2)
- Se valores originais forem distintos, resultado é distribuição uniforme

Normalização por reescala

- Reescalar: adicionar/subtrair/multiplicar/dividir por uma constante
- Normalização min-max
 - São definidos inicialmente mínimo e máximo para os novos valores
 - Depois, para cada atributo aplica:

$$v_{novo} = min + \frac{v_{atual} - menor}{maior - menor}(max - min) \quad (1)$$

Normalização por reescala

- Ex. conjunto de dados hospital
 - Normalização de Idade entre 0 (min) e 1 (max)

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
28	M	79	Grandes	38,0	2	Doente
18	F	67	Pequenas	39,5	4	Doente
49	M	92	Grandes	38,0	2	Saudável
18	M	43	Grandes	38,5	20	Doente
21	F	52	Médias	37,6	1	Saudável
22	F	72	Pequenas	38,0	3	Doente
19	F	87	Grandes	39,0	6	Doente
34	M	67	Médias	38,4	2	Saudável

Maior = 49
 Menor = 18

Normalização por reescala

- Ex. conjunto de dados hospital
 - Normalização de Idade entre 0 (min) e 1 (max)

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
28	M	79	Grandes	38,0	2	Doente
18	F	67	Pequenas	39,5	4	Doente
49	M	92	Grandes	38,0	2	Saudável
18	M	43	Grandes	38,5	20	Doente
21	F	52	Médias	37,6	1	Saudável
22	F	72	Pequenas	38,0	3	Doente
19	F	87	Grandes	39,0	6	Doente
34	M	67	Médias	38,4	2	Saudável

$$v_{\text{novo}} = \frac{v_{\text{atual}} - 18}{49 - 18}$$

Normalização por reescala

- Ex. conjunto de dados hospital
 - Normalização de Idade entre 0 (min) e 1 (max)

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
0,32	M	79	Grandes	38,0	2	Doente
0	F	67	Pequenas	39,5	4	Doente
1	M	92	Grandes	38,0	2	Saudável
0	M	43	Grandes	38,5	20	Doente
0,1	F	52	Médias	37,6	1	Saudável
0,13	F	72	Pequenas	38,0	3	Doente
0,03	F	87	Grandes	39,0	6	Doente
0,52	M	67	Médias	38,4	2	Saudável

Normalização por reescala

- Ex. conjunto de dados hospital
 - Normalização de #Int. entre 0 (min) e 1 (max)

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
0,32	M	79	Grandes	38,0	2	Doente
0	F	67	Pequenas	39,5	4	Doente
1	M	92	Grandes	38,0	2	Saudável
0	M	43	Grandes	38,5	20	Doente
0,1	F	52	Médias	37,6	1	Saudável
0,13	F	72	Pequenas	38,0	3	Doente
0,03	F	87	Grandes	39,0	6	Doente
0,52	M	67	Médias	38,4	2	Saudável

Maior = 20
 Menor = 1

Normalização por reescala

- Ex. conjunto de dados hospital
 - Normalização de #Int. entre 0 (min) e 1 (max)

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
0,32	M	79	Grandes	38,0	2	Doente
0	F	67	Pequenas	39,5	4	Doente
1	M	92	Grandes	38,0	2	Saudável
0	M	43	Grandes	38,5	20	Doente
0,1	F	52	Médias	37,6	1	Saudável
0,13	F	72	Pequenas	38,0	3	Doente
0,03	F	87	Grandes	39,0	6	Doente
0,52	M	67	Médias	38,4	2	Saudável

$$v_{\text{novo}} = \frac{v_{\text{atual}} - 1}{20 - 1}$$

Normalização por reescala

- Ex. conjunto de dados hospital
 - Normalização de #Int. entre 0 (min) e 1 (max)

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
0,32	M	79	Grandes	38,0	0,05	Doente
0	F	67	Pequenas	39,5	0,16	Doente
1	M	92	Grandes	38,0	0,05	Saudável
0	M	43	Grandes	38,5	1	Doente
0,1	F	52	Médias	37,6	0	Saudável
0,13	F	72	Pequenas	38,0	0,11	Doente
0,03	F	87	Grandes	39,0	0,26	Doente
0,52	M	67	Médias	38,4	0,05	Saudável

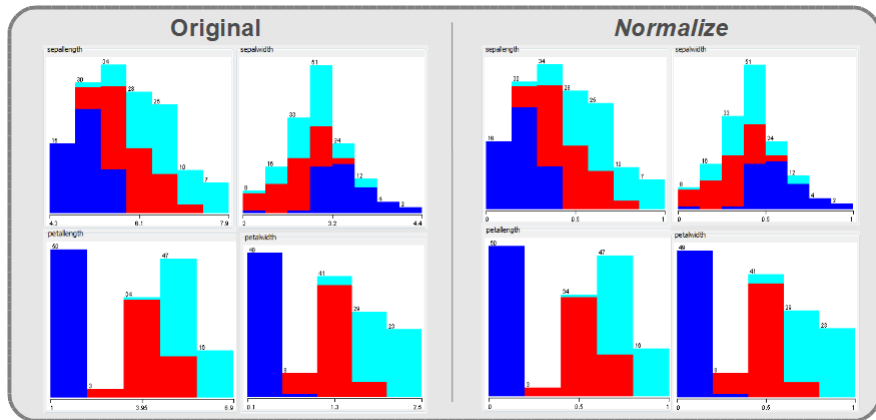
Normalização por reescala

- Ex. conjunto de dados hospital
 - Observe o *efeito de outlier*

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
0,32	M	79	Grandes	38,0	0,05	Doente
0	F	67	Pequenas	39,5	0,16	Doente
1	M	92	Grandes	38,0	0,05	Saudável
0	M	43	Grandes	38,5	1	Doente
0,1	F	52	Médias	37,6	0	Saudável
0,13	F	72	Pequenas	38,0	0,11	Doente
0,03	F	87	Grandes	39,0	0,26	Doente
0,52	M	67	Médias	38,4	0,05	Saudável

Normalização por reescala

- Ex. conjunto de dados iris



Normalização por padronização

- Para padronizar valores de atributos basta:
 - Adicionar/subtrair por uma medida de localização
 - Multiplicar/dividir por uma medida de escala
- Lida melhor com outliers
- Ex. atributos com média 0 e variância 1:

$$v_{novo} = \frac{v_{atual} - \bar{\mathbf{x}}^i}{sd(\bar{\mathbf{x}}^i)} \quad (2)$$

Importante:

Diferentes atributos podem ter limites superiores e inferiores diferentes, mas terão os mesmos valores para as medidas de escala e espalhamento

Normalização por padronização

- Padronização de Idade com média 0 e variância 1

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
28	M	79	Grandes	38,0	2	Doente
18	F	67	Pequenas	39,5	4	Doente
49	M	92	Grandes	38,0	2	Saudável
18	M	43	Grandes	38,5	20	Doente
21	F	52	Médias	37,6	1	Saudável
22	F	72	Pequenas	38,0	3	Doente
19	F	87	Grandes	39,0	6	Doente
34	M	67	Médias	38,4	2	Saudável

Média = 21,5
Desv_pad = 10,79

Normalização por padronização

- Padronização de Idade com média 0 e variância 1

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
28	M	79	Grandes	38,0	2	Doente
18	F	67	Pequenas	39,5	4	Doente
49	M	92	Grandes	38,0	2	Saudável
18	M	43	Grandes	38,5	20	Doente
21	F	52	Médias	37,6	1	Saudável
22	F	72	Pequenas	38,0	3	Doente
19	F	87	Grandes	39,0	6	Doente
34	M	67	Médias	38,4	2	Saudável

$$v_{\text{novo}} = \frac{v_{\text{atual}} - 21,5}{10,79}$$

Normalização por padronização

- Padronização de Idade com média 0 e variância 1

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
0,6	M	79	Grandes	38,0	2	Doente
-0,32	F	67	Pequenas	39,5	4	Doente
2,55	M	92	Grandes	38,0	2	Saudável
-0,32	M	43	Grandes	38,5	20	Doente
-0,05	F	52	Médias	37,6	1	Saudável
0,05	F	72	Pequenas	38,0	3	Doente
-0,23	F	87	Grandes	39,0	6	Doente
1,16	M	67	Médias	38,4	2	Saudável

Média = 0
Desv_pad = 1

Normalização por padronização

- Padronização de $\#Int.$ com média 0 e variância 1

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
0,6	M	79	Grandes	38,0	2	Doente
-0,32	F	67	Pequenas	39,5	4	Doente
2,55	M	92	Grandes	38,0	2	Saudável
-0,32	M	43	Grandes	38,5	20	Doente
-0,05	F	52	Médias	37,6	1	Saudável
0,05	F	72	Pequenas	38,0	3	Doente
-0,23	F	87	Grandes	39,0	6	Doente
1,16	M	67	Médias	38,4	2	Saudável

Média = 2,5
desv_pad = 6,26

Normalização por padronização

- Padronização de $\#Int.$ com média 0 e variância 1

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
0,6	M	79	Grandes	38,0	2	Doente
-0,32	F	67	Pequenas	39,5	4	Doente
2,55	M	92	Grandes	38,0	2	Saudável
-0,32	M	43	Grandes	38,5	20	Doente
-0,05	F	52	Médias	37,6	1	Saudável
0,05	F	72	Pequenas	38,0	3	Doente
-0,23	F	87	Grandes	39,0	6	Doente
1,16	M	67	Médias	38,4	2	Saudável

$$v_{\text{novo}} = \frac{v_{\text{atual}} - 2,5}{6,26}$$

Normalização por padronização

- Padronização de $\#Int.$ com média 0 e variância 1

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
0,6	M	79	Grandes	38,0	-0,08	Doente
-0,32	F	67	Pequenas	39,5	0,24	Doente
2,55	M	92	Grandes	38,0	-0,08	Saudável
-0,32	M	43	Grandes	38,5	2,8	Doente
-0,05	F	52	Médias	37,6	-0,24	Saudável
0,05	F	72	Pequenas	38,0	0,08	Doente
-0,23	F	87	Grandes	39,0	0,56	Doente
1,16	M	67	Médias	38,4	-0,08	Saudável

Média = 0
Desv_pad = 1

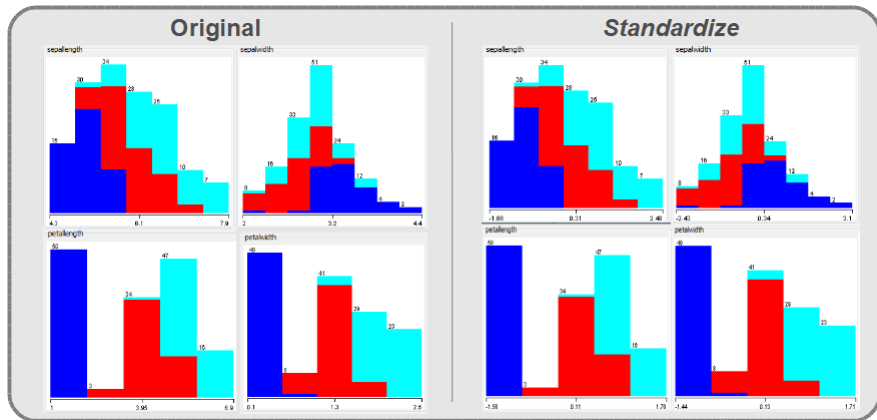
Normalização por padronização

- Observe o *efeito de outlier*

Idade	Sexo	Peso	Manchas	Temp.	# Int.	Diagnóstico
0,6	M	79	Grandes	38,0	-0,08	Doente
-0,32	F	67	Pequenas	39,5	0,24	Doente
2,55	M	92	Grandes	38,0	-0,08	Saudável
-0,32	M	43	Grandes	38,5	2,8	Doente
-0,05	F	52	Médias	37,6	-0,24	Saudável
0,05	F	72	Pequenas	38,0	0,08	Doente
-0,23	F	87	Grandes	39,0	0,56	Doente
1,16	M	67	Médias	38,4	-0,08	Saudável

Normalização por padronização

- Ex. conjunto de dados iris



Transformação de atributos numéricos

- Outro tipo de transformação: **tradução**
 - Valor é traduzido por um mais facilmente manipulável
 - Ex. converter data de nascimento para idade
 - Ex. converter temperatura de F para C
 - Ex. localização por GPS para código postal

Transformação de atributos numéricos

- Outro tipo de transformação: aplicação de função simples
 - Aplicação a cada valor do atributo
 - Ex. log, exp, raiz, seno, $1/x$, abs
 - Ex. apenas magnitude dos valores é importante converter para valor absoluto
 - Funções raiz, log e $1/x$: aproximam uma distribuição Gaussiana
 - Função log: comprimir dados com grande intervalo de valores

Outline

- 1 Introdução
- 2 Integração de Dados
- 3 Eliminação manual de atributos
- 4 Amostragem de dados
- 5 Dados desbalanceados
- 6 Limpeza de Dados
- 7 Transformação de Dados
- 8 Redução de dimensionalidade**
- 9 Referências

Redução de dimensionalidade

- Muitos problemas possuem número elevado de atributos
 - Ex. dados de expressão gênica
 - Ex. imagens, se cada pixel for considerado um atributo

Problema:

Maldição da dimensionalidade

Maldição da dimensionalidade

- Supor dados representados por pontos em um hipervolume
 - Valores de atributos dão as coordenadas
- Hipervolume cresce exponencialmente com a adição de novos atributos
 - 1 atributo com 10 possíveis valores: 10 possíveis objetos
 - 5 atributos com 10 possíveis valores: 10^5 possíveis objetos
 - Problemas com poucos exemplos e muitos atributos: dados se tornam muito esparsos

Problema:

Sem exemplos em várias das regiões do espaço de objetos; instâncias parecem equidistantes (dificultando encontrar padrões)

Maldição da dimensionalidade

- Número de exemplos necessários para manter desempenho cresce exponencialmente com o número de atributos
 - Na prática, o número de exemplos de treinamento é fixo
 - Pode haver necessidade de [redução de dimensionalidade](#)

Redução de dimensionalidade

Vantagens:

- Alguns algoritmos de AM que têm dificuldades em lidar com número elevado de atributos
- Melhorar desempenho do modelo induzido pela identificação e eliminação de ruídos nos atributos
- Reduzir custo computacional do modelo
- Resultados mais compreensíveis

Redução de dimensionalidade

- Técnicas podem ser divididas em duas abordagens:

Agregação

- Combinação dos atributos originais por funções lineares ou não lineares
- Ex. PCA (Principal Component Analysis), que elimina redundâncias por correlação
- Levam à perda dos valores originais

Seleção de atributos

- Identificar os atributos mais importantes
- Manter os **relevantes**
- Remover os **redundantes** e **inconsistentes**
- Diferentes critérios podem ser usados para medir importância

PCA

- Conjunto de d atributos $(\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^d)$
 - Transformação linear para um novo conjunto de d atributos
 - Componentes principais (PCs) são tipos específicos de combinações lineares

$$\mathbf{z}^1 = a_{11} \mathbf{x}^1 + a_{21} \mathbf{x}^2 + \dots + a_{d1} \mathbf{x}^d$$

$$\mathbf{z}^2 = a_{12} \mathbf{x}^1 + a_{22} \mathbf{x}^2 + \dots + a_{d2} \mathbf{x}^d$$

...

$$\mathbf{z}^d = a_{1d} \mathbf{x}^1 + a_{2d} \mathbf{x}^2 + \dots + a_{dd} \mathbf{x}^d$$

PCA

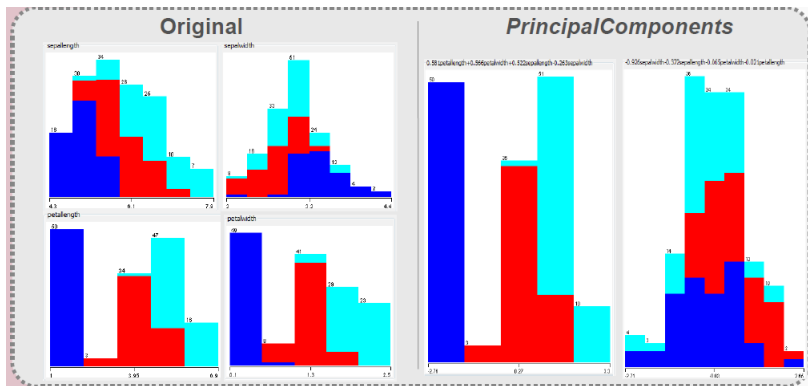
- Propriedades das componentes principais:
 - As d componentes principais são não correlacionadas (independentes)
 - Toma-se os autovalores e autovetores da matriz de covariância dos dados
 - As PCs são ordenadas de acordo com a quantidade de variância dos dados originais que elas contêm (maiores auto-valores)
 - Primeira componente “explica” (contém) a maior variabilidade do conjunto de dados
 - Segunda componente define próxima parte, e assim por diante

Componentes principais

- Em geral apenas algumas das primeiras PCs são responsáveis pela maior parte da variabilidade nos dados
- O restante das PCs tem contribuição insignificante e pode ser eliminada

PCA

- Ex. conjunto de dados iris



Seleção de Atributos

- **Objetivo:** encontrar subconjunto ótimo de atributos de acordo com algum critério
- Permite:
 - Identificar atributos **importantes**
 - Melhorar o **desempenho** de várias técnicas
 - **Reduzir** necessidade de memória e tempo
 - **Eliminar** atributos irrelevantes e reduzir ruídos
 - **Lidar** com a maldição da dimensionalidade
 - **Simplificar** o modelo gerado, tornando mais fácil sua compreensão
 - **Facilitar** a visualização dos dados
 - **Reduzir** o custo de coleta dos dados

Seleção de Atributos

- Na prática, é difícil identificar atributos passíveis de eliminação
 - Redundantes
 - Irrelevantes

Algumas razões:

- Número grande de exemplos
- Número grande de atributos
- Relações complexas entre atributos

Necessidade de técnicas automáticas para seleção de atributos

Seleção de Atributos

- Técnicas podem ser classificadas de diferentes formas:
 - Quanto à maneira de avaliar os atributos selecionados e interação com o algoritmo de AM
 - Interação com o algoritmo de AM
 - Quanto a considerar cada atributo individualmente ou em subconjuntos
 - Quanto à medida de importância usada
 - Quanto ao uso ou não da classe
 - Em conjuntos de dados rotulados

Seleção de Atributos

- Quanto à avaliação dos atributos: técnicas podem estar integradas a um algoritmo de indução ou serem independentes do algoritmo

Embutida

- Seleção é embutida ou integrada no próprio algoritmo de AM

Filtro

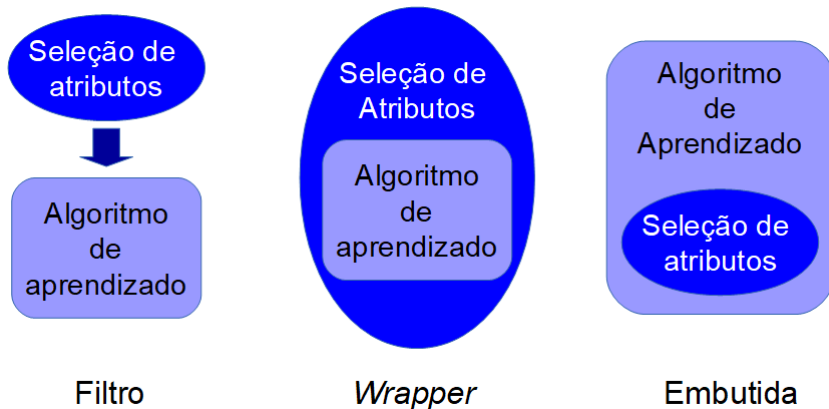
- Filtra atributos, sem levar em consideração o(s) algoritmo(s) de AM que os utilizará(ão)
- Verificam características dos dados

Wrapper

- Usa algoritmo de AM como “caixa-preta” para a seleção, que avalia os subconjuntos

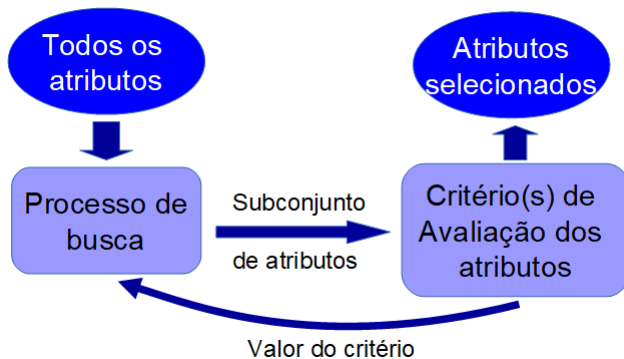
Seleção de Atributos

- Abordagens de avaliação dos atributos:



SA Filtro

- Seleção de atributos independente do algoritmo de AM (ex. verificando correlação entre os atributos)



SA filtro

Vantagens:

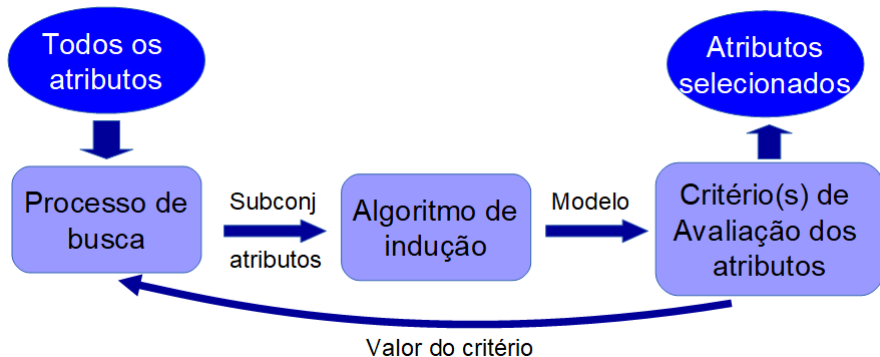
- Geralmente mais rápida, com heurísticas para avaliar os subconjuntos que são pouco custosas
- Atributos selecionados podem ser usados por diferentes algoritmos de AM
- Conseguem lidar com uma grande quantidade de dados

Desvantagem:

- Independência do algoritmo de AM
 - Não considerar o viés do algoritmo de AM pode levar a modelos pouco efetivos para a técnica em particular

SA wrapper

- Usa algoritmo de AM para avaliar os atributos, ex. desempenho preditivo usando o subconjunto



SA wrapper

Vantagens:

- Melhor subconjunto para cada algoritmo de AM, podendo selecionar menos atributos também
- Geralmente leva a modelos com melhor desempenho preditivo/descritivo

Desvantagem:

- Risco de overfitting
- Subconjunto depende do algoritmo de AM
 - Para cada novo algoritmo, deve ser repetido
- Custo computacional elevado

Seleção de Atributos

- Quanto à seleção ser individual ou coletiva:

Ordenação (análise univariada)

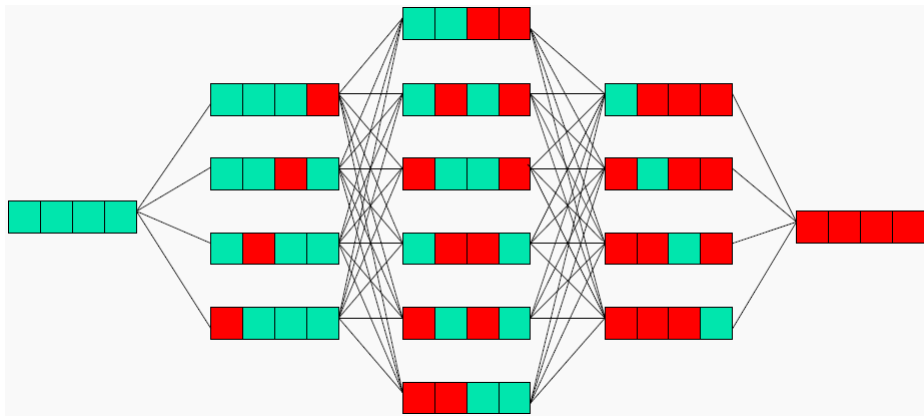
- Seleção é individual
- Atributos são ordenados de acordo com relevância segundo algum critério e atributos no topo são selecionados (ranking)
- Tende a selecionar atributos correlacionados
- Necessidade de definir limiar de seleção

Seleção de subconjunto (análise multivariada)

- Seleciona subconjunto dos atributos originais
- Verifica como atributos atuam de forma coletiva, em conjunto
- É computacionalmente mais cara
- Pode ser formulada como um problema de busca

Seleção de Subconjunto de Atributos

- Busca: cada ponto no espaço de busca pode ser visto como um possível subconjunto de atributos



Seleção de Subconjunto de Atributos

- Deve-se definir na busca:
 - Ponto(s) de partida ou **direção** da busca
 - **Estratégia** de busca
 - **Critério usado na avaliação** dos subconjuntos
 - Critério de **parada**

Seleção de Subconjunto de Atributos

Pontos de partida/direção da busca:

Para trás (backward)

- Começa com todos atributos
- Remove um por vez

Para frente (forward)

- Começa sem nenhum atributo
- Inclui um por vez

Bidirecional

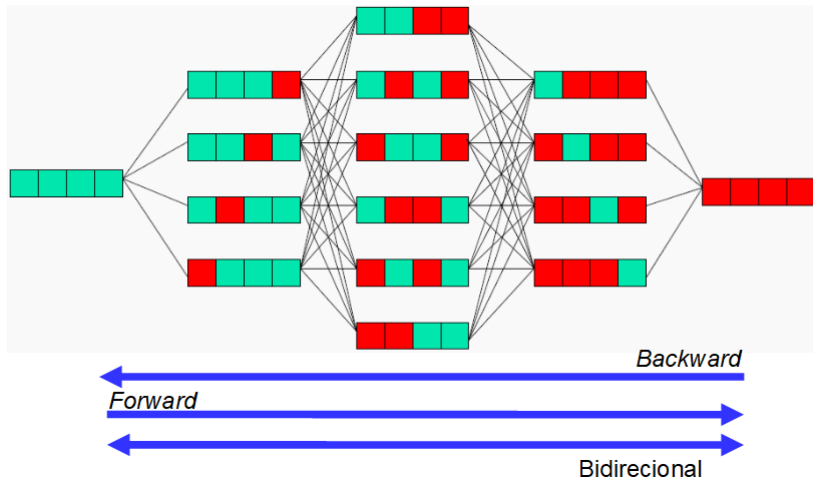
- Pode começar em qualquer ponto
- Atributos podem ser adicionados ou removidos

Estocástica (random)

- Ponto de partida da busca de atributos a serem adiciona-dos/removidos é decidido estocasticamente

Seleção de subconjunto de atributos

Pontos de partida/direção da busca:



Seleção de Subconjunto de Atributos

Estratégias de busca possíveis:

Busca completa

- Avalia todos os possíveis subconjuntos
- Pode não necessitar visitar todos exaustivamente

Busca heurística

- Utiliza regras e métodos heurísticos na busca
- Não garante encontrar a solução ótima

Busca não determinística

- Relacionada com a geração estocástica
- Boa solução pode ser encontrada antes do final da busca
- Não garante encontrar o ótimo

Seleção de subconjunto de atributos

Relações entre sentidos e estratégias de busca:

Sentido	Estratégia		
	Completa	Heurística	Não determinística
Forward	Sim	Sim	Não
Backward	Sim	Sim	Não
Estocástico	Não	Sim	Sim

Seleção de subconjunto de atributos

CrITÉRIOS de **parada** possíveis:

- Quando todos os subconjuntos forem testados (exaustiva)
- Quando um número máximo de alternativas é testado
- Quando atinge um número de atributos desejado
- Enquanto adição/remoção de atributos não deteriora desempenho do modelo de AM

Seleção de atributos

Tipos de critérios para **avaliar** a importância dos atributos:

Consistência

- Indicam se subconjunto permite construir projeção consistente dos dados
- Possibilitar a construção de hipóteses lógicas consistentes em um conjunto de dados

Dependência

- Mensuram capacidade de prever o valor de um atributo a partir do valor de outro atributo

Distância

- Mantêm fato de que exemplos próximos têm alguma relação

Seleção de atributos

Tipos de critérios para **avaliar** a importância dos atributos:

Informação

- Atributos mais importantes levam a maior ganho de informação

Precisão

- Atributos mais importantes levam a melhor desempenho preditivo de um modelo
- Geralmente associada à abordagem wrapper

SA: exemplo

Conjunto de dados iris

```
Search Method:
  Attribute ranking.
Attribute Evaluator (supervised, Class (nominal): 5 class):
  Information Gain Ranking Filter

Ranked attributes:
1.418  3  petallength
1.378  4  petalwidth
0.698  1  sepallength
0.376  2  sepalwidth

Selected attributes: 3,4,1,2 : 4
```

SA: exemplo

Conjunto de dados iris

Search Method:

Greedy Stepwise (forwards).

Start set: no attributes

Merit of best subset found: 0.973

Attribute Subset Evaluator (supervised, Class (nominal): 5 class):
Consistency Subset Evaluator

Selected attributes: 3,4 : 2
petallength
petalwidth

SA: exemplo

Conjunto de dados iris

Search Method:

Greedy Stepwise (forwards).

Start set: no attributes

Merit of best subset found: 0.033

Attribute Subset Evaluator (supervised, Class (nominal): 5 class):

Wrapper Subset Evaluator

Learning scheme: weka.classifiers.functions.SMO

Accuracy estimation: classification error

Number of folds for accuracy estimation: 5

Selected attributes: 1,2,4 : 3

sepalength

sepalwidth

petalwidth

Outline

- 1 Introdução
- 2 Integração de Dados
- 3 Eliminação manual de atributos
- 4 Amostragem de dados
- 5 Dados desbalanceados
- 6 Limpeza de Dados
- 7 Transformação de Dados
- 8 Redução de dimensionalidade
- 9 Referências

Referências

- Capítulo 3 do livro Inteligência Artificial: uma abordagem de Aprendizado de Máquina, 2011, LTC
- Spolaôr, N. (2010) Aplicação de Algoritmos Genéticos Multiobjetivo ao problema de seleção de atributos. Dissertação de Mestrado em Engenharia de Informação, Universidade Federal do ABC.
- T. F. Covões (2010) Seleção de atributos via agrupamento. Dissertação de Mestrado em Ciência da Computação e Matemática Computacional, Universidade de São Paulo, São Carlos.
- Alguns slides foram baseados em apresentações de:
 - Profa Dra Ana Carolina Lorena, ITA
 - Prof Dr André C. P. L. F. Carvalho, ICMC-USP
 - Prof Dr Edson Pimentel, UFABC
 - Softwares utilizados:
 - RStudio
 - Weka