Mineração de Dados: Conceitos, Tarefas, Métodos e Ferramentas

Cássio Oliveira Camilo *João Carlos da Silva*

Technical Report - RT-INF_001-09 - Relatório Técnico
August - 2009 - Agosto

Instituto de Informática
Universidade Federal de Goiás
www.inf.ufg.br

Mineração de Dados: Conceitos, Tarefas, Métodos e Ferramentas

Cássio Oliveira Camilo [*] cassio@inf.ufg.br

João Carlos da Silva [†] jcs@inf.ufg.br

Abstract. *This work will be presented the key concepts of Data Mining, main tasks and methods. Besides the traditional methods, some variations and new approaches will be discussed. Finally, a list of the main tools to work with mining.*

Keywords: Data Mining, Tasks, Methods, Tools.

Resumo. *Neste trabalho serão apresentados os conceitos fundamentais da Mineração de Dados, principais tarefas e métodos. Além dos métodos tradicionais, algumas variantes e novas abordagens serão discutidas. Ao final será apresentada uma lista das principais ferramentas para se trabalhar com mineração.*

Palavras-Chave: Mineração de Dados, Tarefas, Métodos, Ferramentas.

1 Introdução

Desde o surgimento dos sistemas computacionais, um dos principais objetivos das organizações tem sido o de armazenar dados. Nas últimas décadas essa tendência ficou ainda mais evidente com a queda nos custos para a aquisição de *hardware*, tornando possível armazenar quantidades cada vez maiores de dados. Novas e mais complexas estruturas de armazenamento foram desenvolvidas, tais como: banco de dados, *Data Warehouses* , Bibliotecas Virtuais, Web e outras [16] [27].

Bramer [6], exemplifica o enorme volume de dados gerado pelas aplicações atuais:

- Os satélites de observação da NASA geram cerca de um *terabyte* de dados por dia;
- O projeto Genoma armazena milhares de *bytes* para cada uma das bilhões de bases genéticas;
- Instituições mantêm repositórios com milhares de transações dos seus clientes;

Com o volume de dados armazenados crescendo diariamente, responder uma questão tornou-se crucial [39]: O que fazer com os dados armazenados? As técnicas tradicionais de exploração de dados não são mais adequadas para tratar a grande maioria dos repositórios. Com

[*] Mestrando em Ciência da Computação - INF/UFG

[†] Orientador - INF/UFG

a finalidade de responder a esta questão, foi proposta, no final da década de 80, a Mineração de Dados, do inglês *Data Mining*.

A Mineração de Dados é uma das tecnologias mais promissoras da atualidade. Um dos fatores deste sucesso é o fato de dezenas, e muitas vezes centenas de milhões de reais serem gastos pelas companhias na coleta dos dados e, no entanto, nenhuma informação útil é identificada [39]. Em seu trabalho, Han [27] refere-se a essa situação como "rico em dados, pobre em informação". Além da iniciativa privada, o setor público e o terceiro setor (ONGt's) também podem se beneficiar com a Mineração de Dados [84].

Witten et al. [88], Olson et al. [58] e Bramer [6] apresentam algumas das áreas nas quais a Mineração de Dados é aplicada de forma satisfatória:

- Retenção de clientes: identificação de perfis para determinados produtos, venda cruzada;
- Bancos: identificar padrões para auxiliar no gerenciamento de relacionamento com o cliente;
- Cartão de Crédito: identificar segmentos de mercado, identificar padrões de rotatividade;
- Cobrança: detecção de fraudes;
- Telemarketing: acesso facilitado aos dados do cliente;
- Eleitoral: identificação de um perfil para possíveis votantes;
- Medicina: indicação de diagnósticos mais precisos;
- Segurança: na detecção de atividades terroristas e criminais [48] [15];
- Auxílio em pesquisas biométricas [38];
- RH: identificação de competências em currículos [9];
- Tomada de Decisão: filtrar as informações relevantes, fornecer indicadores de probabilidade.

Segundo Ponniah [65], o uso da Mineração de Dados permite, por exemplo, que:

- Um supermercado melhore a disposição de seus produtos nas prateleiras, através do padrão de consumo de seus clientes;
- Uma companhia de marketing direcione o envio de mensagens promocionais, obtendo melhores retornos;
- Uma empresa aérea possa diferenciar seus serviços oferecendo um atendimento personalizado;
- Empresas planejem melhor a logística de distribuição dos seus produtos, prevendo picos nas vendas;
- Empresas possam economizar identificando fraudes;
- Agências de viagens possam aumentar o volume de vendas direcionando seus pacotes a clientes com aquele perfil;

Alguns casos de sucesso da Mineração de Dados estão relatados em Ye [91], Han et al. [27], Myatt et al. [54] e Hornick et al. [30].

2 Descoberta de Conhecimento

Segundo Fayyad [20], o modelo tradicional para transformação dos dados em informação (conhecimento), consiste em um processamento manual de todas essas informações por especialistas que, então, produzem relatórios que deverão ser analisados. Na grande maioria das situações, devido ao grande volume de dados, esse processo manual torna-se impraticável. Ainda segundo Fayyad, o *KDD* (*Knowledge Discovery in Databases* ou Descoberta de Conhecimento nas Bases de Dados) é uma tentativa de solucionar o problema causado pela chamada "era da informação": a sobrecarga de dados.

Ainda não é consenso a definição dos termos *KDD* e *Data Mining*. Em Rezende [69], Wang [83] e Han et al. [27] eles são considerados sinônimos. Para Cios et al. [16] e Fayyad [20] o *KDD* refere-se a todo o processo de descoberta de conhecimento, e a Mineração de Dados a uma das atividades do processo. No entanto, todos concordam que o processo de mineração deve ser iterativo, interativo e divido em fases. Na figura 1 podemos ver uma representação do processo de KDD.

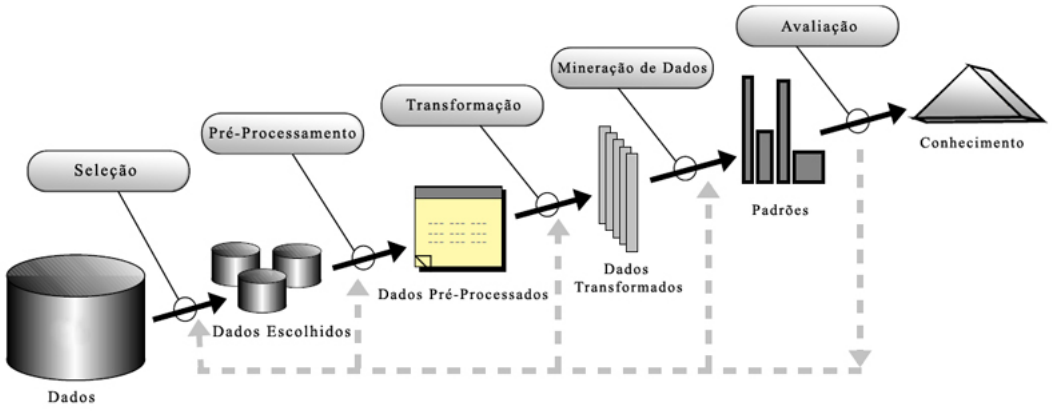

Figura 1: Figura representando o processo de KDD [20]

Uma das definições mais utilizadas para o termo *KDD* é de Fayyad [20], que o define como "um processo não trivial de identificação de novos padrões válidos, úteis e compreensíveis".

Atualmente diversos processos definem e padronizam as fases e atividades da Mineração de Dados. Apesar das particularidades, todos em geral contém a mesma estrutura. Neste trabalho, escolhemos o CRISP-DM (*Cross-Industry Standard Processo of Data Mining*) [14] como modelo, devido à vasta literatura disponível e por atualmente ser considerado o padrão de maior aceitação [39] [28]. Um *ranking* do uso dos principais processos pode ser encontrado em [32].

Como afirma Olson et al. [58], o processo CRISP-DM consiste de seis fases organizadas de maneira cíclica, conforme mostra a figura 2. Além disto, apesar de ser composto por fases, o fluxo não é unidirecional, podendo ir e voltar entre as fases.

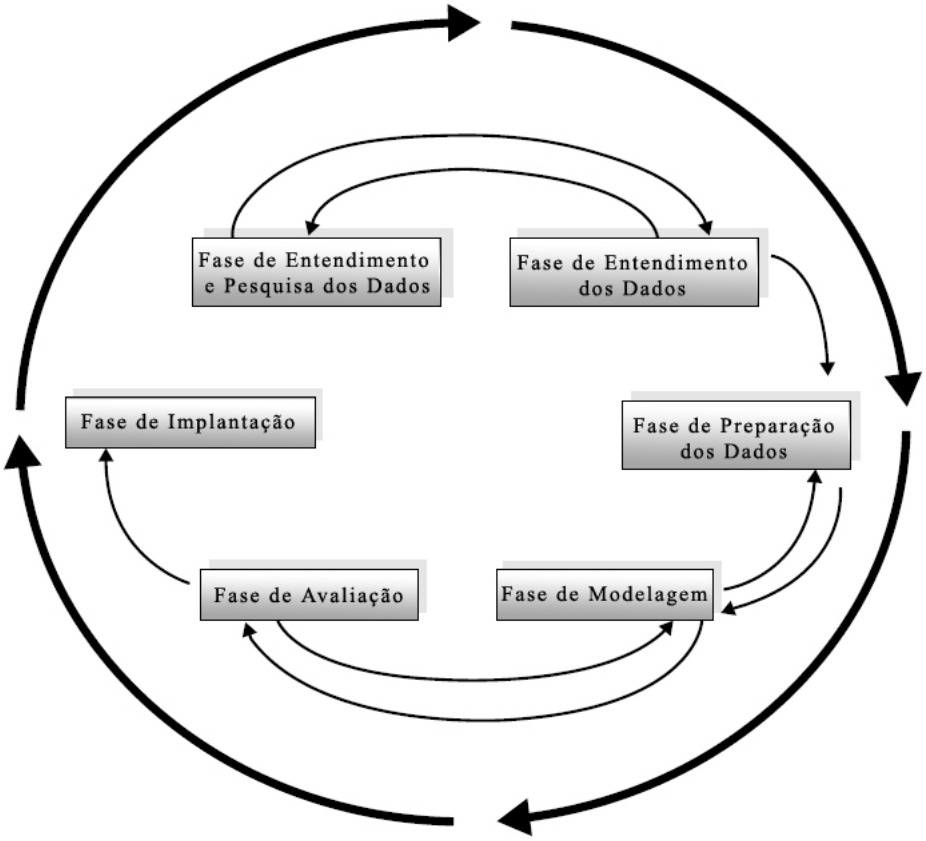

Figura 2: Figura representando o processo CRISP [39]

As fases do processo CRISP-DM são:

1. **Entendimento dos Negócios:** Nessa etapa, o foco é entender qual o objetivo que se deseja atingir com a mineração de dados. O entendimento do negócio irá ajudar nas próximas etapas.

2. **Entendimento dos Dados:** As fontes fornecedoras dos dados podem vir de diversos locais e possuírem diversos formatos. Segundo Olson et al. [58], após definir os objetivos, é necessário conhecer os dados visando:

   - Descrever de forma clara o problema;
   - Identificar os dados relevantes para o problema em questão;
   - Certificar-se de que as variáveis relevantes para o projeto não são interdependentes.

   Normalmente as técnicas de agrupamento e de exploração visual também são utilizadas nesta etapa [58].

3. **Preparação dos Dados:** Devido às diversas origens possíveis, é comum que os dados não estejam preparados para que os métodos de Mineração de Dados sejam aplicados diretamente. Dependendo da qualidade desses dados, algumas ações podem ser necessárias. Este processo de limpeza dos dados geralmente envolve filtrar, combinar e preencher valores vazios.

4. **Modelagem:** É nesta fase que as técnicas (algoritmos) de mineração serão aplicadas. A escolha da(s) técnica(s) depende dos objetivos desejados [48].

5. **Avaliação:** Considerada uma fase crítica do processo de mineração, nesta etapa é necessária a participação de especialistas nos dados, conhecedores do negócio e tomadores de decisão. Diversas ferramentas gráficas são utilizadas para a visualização e análise dos resultados (modelos).

   Testes e validações, visando obter a confiabilidade nos modelos, devem ser executados (*cross validation*, *suplied test set*, *use training set*, *percentage split*) e indicadores para auxiliar a análise dos resultados precisam ser obtidos (matriz de confusão, índice de correção e incorreção de instâncias mineradas, estatística *kappa*, erro médio absoluto, erro relativo médio, precisão, *F-measure*, dentre outros) [27] [88].

6. **Distribuição:** Após executado o modelo com os dados reais e completos é necessário que os envolvidos conheçam os resultados.

Constantemente, novos processos são propostos para se trabalhar com a Mineração de Dados. Aranda et al. [23], propõe um modelo envolvendo o processo RUP e o CRISP-DM. Pechenizkiy et al. [61], propõe um processo baseado no modelo dos Sistemas de Informações.

3 Os Dados

Conhecer o tipo dos dados com o qual se irá trabalhar também é fundamental para a escolha do(s) método(s) mais adequado(s). Pode-se categorizar os dados em dois tipos: quantitativos e qualitativos. Os dados quantitativos são representados por valores numéricos. Eles ainda podem ser discretos e contínuos. Já os dados qualitativos contêm os valores nominais e ordinais (categóricos). Em geral, antes de se aplicar os algoritmos de mineração é necessário explorar, conhecer e preparar os dados.

Nesse sentido, uma das primeiras atividades é obter uma visualização dos dados, de forma que se possa ter uma visão geral, para depois decidir-se quais as técnicas mais indicadas. Diversas são as técnicas utilizadas para a visualização dos dados. Simoff [78], Rezende [69], Myatt [53], Myatt et al. [54], NIST [56] e Canada [10] apresentam diversas abordagens para as visualizações. Keim [33], apresenta um estudo sobre as diversas técnicas de visualização. A figura 3 mostra a evolução dessas técnicas.

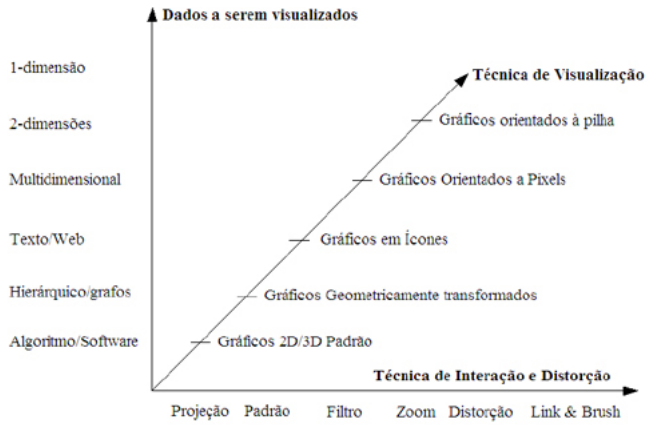

Figura 3: Evolução das técnicas de visualização [33]

Com uma visão inicial dos dados definida, é necessário explorá-los, buscando, além de mais conhecimento sobre os mesmos, encontrarmos valores que possam comprometer sua quali-

dade, tais como: valores em branco ou nulo, valores viciados, variáveis duplicadas, entre outras. À medida em que problemas vão sendo encontrados e o entendimento vai sendo obtido, ocorre a preparação dos dados para que os algoritmos de mineração possam ser aplicados. Segundo Olson et al. [58], o processo de preparação dos dados na maioria dos projetos de mineração, compreende até 50% de todo o processo. Para McCue [48], está etapa pode compreender até 80%.

Han e Kamber [27], descrevem várias técnicas estatísticas de análise de dispersão (*Quartiles*, Variância) e de medida central (média, mediana, moda e faixa de valores) combinadas com gráficos (Histogramas, Frequência, Barra, *BoxPlot*, Dispersão) são usadas para a exploração dos dados. Myatt [53], utiliza a técnica de Análise Exploratória dos Dados (*EDA - Exploratory Data Analisis*) para auxiliar nessa atividade.

O processo de preparação dos dados para a mineração, também chamado de pré-processamento, segundo Han et al. [27], consiste principalmente em:

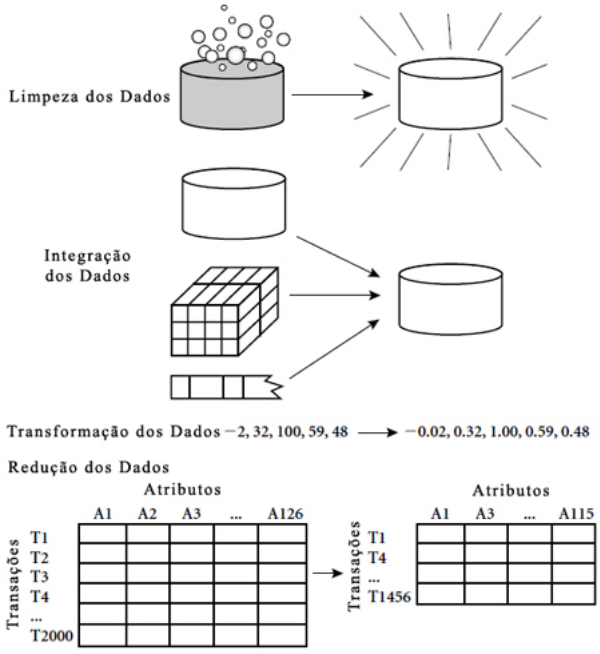

Figura 4: Atividades do pré-processamento [27]

Limpeza dos dados: Frequentemente, os dados são encontrados com diversas inconsistências: registros incompletos, valores errados e dados inconsistentes. A etapa de limpeza dos dados visa eliminar estes problemas de modo que eles não influam no resultado dos algoritmos usados. As técnicas usadas nesta etapa vão desde a remoção do registro com problemas, passando pela atribuição de valores padrões, até a aplicação de técnicas de agrupamento para auxiliar na descoberta dos melhores valores. Devido ao grande esforço exigido nesta etapa, Han et al. [27] propõem o uso de um processo específico para a limpeza dos dados.

Integração dos dados: É comum obter-se os dados a serem minerados de diversas fontes: banco de dados, arquivos textos, planilhas, *data warehouses*, vídeos, imagens, entre outras. Surge então, a necessidade da integração destes dados de forma a termos um repositório único e consistente. Para isto, é necessária uma análise aprofundada dos dados observando redundâncias, dependências entre as variáveis e valores conflitantes (cat-

egorias diferentes para os mesmos valores, chaves divergentes, regras diferentes para os mesmos dados, entre outros).

Transformação dos dados: A etapa de transformação dos dados merece destaque. Alguns algoritmos trabalham apenas com valores numéricos e outros apenas com valores categóricos. Nestes casos, é necessário transformar os valores numéricos em categóricos ou os categóricos em valores numéricos. Não existe um critério único para transformação dos dados e diversas técnicas podem ser usadas de acordo com os objetivos pretendidos. Algumas das técnicas empregadas nesta etapa são: suavização (remove valores errados dos dados), agrupamento (agrupa valores em faixas sumarizadas), generalização (converte valores muito específicos para valores mais genéricos), normalização (colocar as variáveis em uma mesma escala) e a criação de novos atributos (gerados a partir de outros já existentes).

Redução dos dados: O volume de dados usado na mineração costuma ser alto. Em alguns casos, este volume é tão grande que torna o processo de análise dos dados e da própria mineração impraticável. Nestes casos, as técnicas de redução de dados podem ser aplicadas para que a massa de dados original seja convertida em uma massa de dados menor, porém, sem perder a representatividade dos dados originais. Isto permite que os algoritmos de mineração sejam executados com mais eficiência, mantendo a qualidade do resultado. As estratégias adotadas nesta etapa são: criação de estruturas otimizadas para os dados (cubos de dados), seleção de um subconjunto dos atributos, redução da dimensionalidade e discretização. Dentre as diversas técnicas, a *PCA - Principal Components Analysis*, desempenha um papel muito importante na redução da dimensionalidade [77] [79]. Outra técnica muito utilizada é a Discretização Baseada na Entropia [27].

Geralmente, os repositórios usados possuem milhares de registros. Neste contexto, o uso de todos os registros do repositório para a construção do modelo de Mineração de Dados é inviável. Assim, utiliza-se uma amostra (mais representativa possível) que é dividida em três conjuntos:

1. Conjunto de Treinamento (*Training Set*): conjunto de registros usados no qual o modelo é desenvolvido;

2. Conjunto de Testes (*Test Set*): conjunto de registros usados para testar o modelo construído;

3. Conjunto de Validação (*Validation Set*): conjunto de registros usados para validar o modelo construído;

Essa divisão em grupos é necessária para que o modelo não fique dependente de um conjunto de dados específico e, ao ser submetido a outros conjuntos (com valores diferentes dos usados na construção e validação do modelo), apresente resultados insatisfatórios. Este efeito é chamado de efeito *Bias*. A medida que se aumenta a precisão do modelo para um conjunto de dados específico, perde-se a precisão para outros conjuntos.

Apesar da grande maioria dos repositórios conterem um volume alto de registros, em alguns casos o que ocorre é o inverso. Neste caso, algumas estratégias foram desenvolvidas para gerar conjunto de dados a partir dos registros existentes [6] [88] [85].

É importante destacar que, apesar de existir um volume muito grande de dados nas empresas, estes dados raramente são disponibilizados para fins de pesquisas. Assim, muitas vezes,

novos algoritmos são criados de forma teórica em ambientes acadêmicos e, pela falta de dados, não se consegue uma avaliação em um ambiente mais próximo do real. Para auxiliar nas pesquisas, repositórios comuns e públicos com diversas bases de dados foram criados por diversas instituições. Um dos mais conhecidos repositórios, com bases de diferentes negócios, tamanhos e tipos, pode ser encontrado em [64].

4 Mineração de Dados

Por ser considerada multidisciplinar, as definições acerca do termo Mineração de Dados variam com o campo de atuação dos autores. Destacamos neste trabalho três áreas que são consideradas como de maior expressão dentro da Mineração de Dados: Estatística, Aprendizado de Máquina e Banco de Dados. Em Zhou [96], é feita uma análise comparativa sobre as três perspectivas citadas.

- Em Hand et al. [28], a definição é dada de uma perspectiva estatística: "Mineração de Dados é a análise de grandes conjuntos de dados a fim de encontrar relacionamentos inesperados e de resumir os dados de uma forma que eles sejam tanto úteis quanto compreensíveis ao dono dos dados".
- Em Cabena et al. [8], a definição é dada de uma perspectiva de banco de dados: "Mineração de Dados é um campo interdisciplinar que junta técnicas de máquinas de conhecimentos, reconhecimento de padrões, estatísticas, banco de dados e visualização, para conseguir extrair informações de grandes bases de dados".
- Em Fayyad et al. [20], a definição é dada da perspectiva do aprendizado de máquina: "Mineração de Dados é um passo no processo de Descoberta de Conhecimento que consiste na realização da análise dos dados e na aplicação de algoritmos de descoberta que, sob certas limitações computacionais, produzem um conjunto de padrões de certos dados."

Apesar das definições sobre a Mineração de Dados levar a crer que o processo de extração de conhecimento se dá de uma forma totalmente automática, sabe-se hoje que de fato isso não é verdade [39]. Apesar de encontrarmos diversas ferramentas que nos auxiliam na execução dos algoritmos de mineração, os resultados ainda precisam de uma análise humana. Porém, ainda assim, a mineração contribui de forma significativa no processo de descoberta de conhecimento, permitindo aos especialistas concentrarem esforços apenas em partes mais significativa dos dados.

4.1 Tarefas

A Mineração de Dados é comumente classificada pela sua capacidade em realizar determinadas tarefas [39]. As tarefas mais comuns são:

Descrição (*Description*) É a tarefa utilizada para descrever os padrões e tendências revelados pelos dados. A descrição geralmente oferece uma possível interpretação para os resultados obtidos. A tarefa de descrição é muito utilizada em conjunto com as técnicas de análise exploratória de dados, para comprovar a influência de certas variáveis no resultado obtido.

Classificação (*Classification*) Uma das tarefas mais comuns, a Classificação, visa identificar a qual classe um determinado registro pertence. Nesta tarefa, o modelo analisa o conjunto de registros fornecidos, com cada registro já contendo a indicação à qual classe pertence, a fim de 'aprender' como classificar um novo registro (aprendizado supervisionado). Por exemplo, categorizamos cada registro de um conjunto de dados contendo as informações sobre os colaboradores de uma empresa: Perfil Técnico, Perfil Negocial e Perfil Gerencial. O modelo analisa os registros e então é capaz de dizer em qual categoria um novo colaborador se encaixa. A tarefa de classificação pode ser usada por exemplo para:

- Determinar quando uma transação de cartão de crédito pode ser uma fraude;
- Identificar em uma escola, qual a turma mais indicada para um determinado aluno;
- Diagnosticar onde uma determinada doença pode estar presente;
- Identificar quando uma pessoa pode ser uma ameaça para a segurança.

Estimação (*Estimation*) ou Regressão (*Regression*) A estimação é similar à classificação, porém é usada quando o registro é identificado por um valor numérico e não um categórico. Assim, pode-se estimar o valor de uma determinada variável analisando-se os valores das demais. Por exemplo, um conjunto de registros contendo os valores mensais gastos por diversos tipos de consumidores e de acordo com os hábitos de cada um. Após ter analisado os dados, o modelo é capaz de dizer qual será o valor gasto por um novo consumidor. A tarefa de estimação pode ser usada por exemplo para:

- Estimar a quantia a ser gasta por uma família de quatro pessoas durante a volta às aulas;
- Estimar a pressão ideal de um paciente baseando-se na idade, sexo e massa corporal.

Predição (*Prediction*) A tarefa de predição é similar às tarefas de classificação e estimação, porém ela visa descobrir o valor futuro de um determinado atributo. Exemplos:

- Predizer o valor de uma ação três meses adiante;
- Predizer o percentual que será aumentado de tráfego na rede se a velocidade aumentar;
- Predizer o vencedor do campeonato baseando-se na comparação das estatísticas dos times.

Alguns métodos de classificação e regressão podem ser usados para predição, com as devidas considerações.

Agrupamento (*Clustering*) A tarefa de agrupamento visa identificar e aproximar os registros similares. Um agrupamento (ou *cluster*) é uma coleção de registros similares entre si, porém diferentes dos outros registros nos demais agrupamentos. Esta tarefa difere da classificação pois não necessita que os registros sejam previamente categorizados (aprendizado não-supervisionado). Além disso, ela não tem a pretensão de classificar, estimar ou predizer o valor de uma variável, ela apenas identifica os grupos de dados similares, conforme mostra a figura 5. Exemplos:

- Segmentação de mercado para um nicho de produtos;
- Para auditoria, separando comportamentos suspeitos;

- Reduzir para um conjunto de atributos similares registros com centenas de atributos.

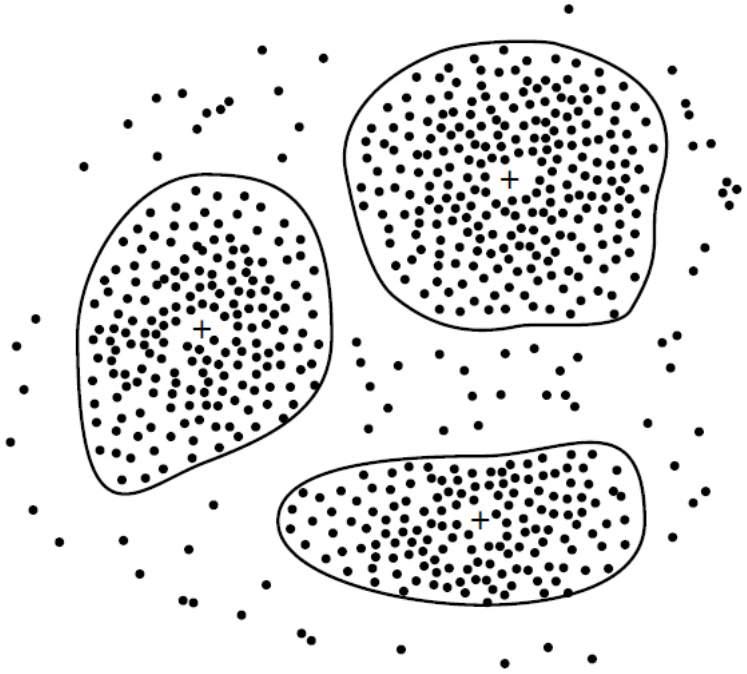

Figura 5: Registros agrupados em três *clusters* [27]

As aplicações das tarefas de agrupamento são as mais variadas possíveis: pesquisa de mercado, reconhecimento de padrões, processamento de imagens, análise de dados, segmentação de mercado, taxonomia de plantas e animais, pesquisas geográficas, classificação de documentos da Web, detecção de comportamentos atípicos (fraudes), entre outras [57]. Geralmente a tarefa de agrupamento é combinada com outras tarefas, além de serem usadas na fase de preparação dos dados.

Associação (*Association*) A tarefa de associação consiste em identificar quais atributos estão relacionados. Apresentam a forma: SE *atributo X* ENTÃO *atributo Y*. É uma das tarefas mais conhecidas devido aos bons resultados obtidos, principalmente nas análises da "Cestas de Compras"(*Market Basket*), onde identificamos quais produtos são levados juntos pelos consumidores. Alguns exemplos:

- Determinar os casos onde um novo medicamento pode apresentar efeitos colaterais;
- Identificar os usuários de planos que respondem bem a oferta de novos serviços.

4.2 Métodos (ou Técnicas)

Tradicionalmente, os métodos de mineração de dados são divididos em aprendizado supervisionado (preditivo) e não-supervisionado (descritivo) [16] [20] [27]. Apesar do limite dessa divisão ser muito tênue (alguns métodos preditivos podem ser descritivos e vice-versa), ela ainda é interessante para fins didáticos [20]. Já existem variações entre os dois tipos de aprendizados. Seliya [73] e Wang [83], são propostas abordagens semi-supervisionadas.

A diferença entre os métodos de aprendizado supervisionados e não-supervisionados reside no fato de que os métodos não-supervisionados não precisam de uma pré-categorização

para os registros, ou seja, não é necessário um atributo alvo. Tais métodos geralmente usam alguma medida de similaridade entre os atributos [48]. As tarefas de agrupamento e associação são consideradas como não-supervisionadas. Já no aprendizado supervisionado, os métodos são providos com um conjunto de dados que possuem uma variável alvo pré-definida e os registros são categorizados em relação a ela. As tarefas mais comuns de aprendizado supervisionado são a classificação (que também pode ser não-supervisionado) e a regressão [48].

Durante o processo de mineração, diversas técnicas devem ser testadas e combinadas a fim de que comparações possam ser feitas e então a melhor técnica (ou combinação de técnicas) seja utilizada [48]. Na figura 6 podemos ver um exemplo de combinação dessas técnicas.

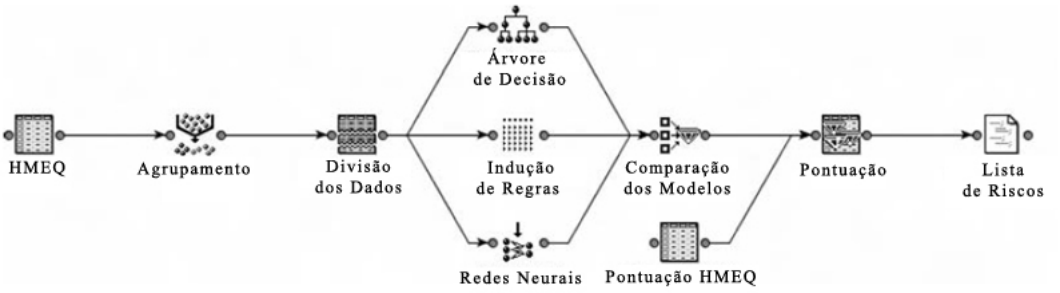

Figura 6: Processo de comparação com algumas técnicas [48]

Os autores classificam os diversos métodos de formas diferentes. Neste trabalho usaremos a classificação adotada por Han et al. [27] para descrever os principais métodos. Nela, os métodos são classificados de acordo com as tarefas que realizam.

- Associações

  É uma das técnicas mais conhecidas de mineração de dados, devido ao problema da Análise da Cesta de Compras. Consiste em identificar o relacionamento dos itens mais frequentes em um determinado conjunto de dados, e permite obter resultados do tipo: SE *compra leite e pão* TAMBÉM *compra manteiga*. Esta construção recebe o nome de Regra de Associação (*Association Rules*). Na figura 7 pode ser visto um exemplo de algumas regras.

Regra 1: SE *idade* = *jovem* AND *estudante* = *não* ENTÃO *compra computadores* = *não*
Regra 2: SE *idade* = *jovem* AND *estudante* = *sim* ENTÃO *compra computadores* = *sim*
Regra 3: SE *idade* = *média* ENTÃO *compra computadores* = *sim*
Regra 4: SE *idade* = *adulto* AND *avaliação de crédito* = *excelente* ENTÃO *compra computadores* = *sim*
Regra 5: SE *idade* = *adulto* AND *avaliuação de crédito* = *ruim* ENTÃO *compra computadores* = *não*

Figura 7: Regra de associação [27]

Mineração de Itens Frequentes (*Frequent Itemset Mining*) Introduzido por Agrawal, Imielinski e Swami [1], essa técnica pode ser visualizada em duas etapas: primeiro, um conjunto de itens frequentes (*Frequent Itemset*) é criado, respeitando um valor mínimo de frequência para os itens. Depois, as regras de associação são geradas pela mineração desse conjunto. Para garantir resultados válidos, os conceitos de suporte e confiança são utilizados em cada regra produzida. A medida de suporte indica o percentual de registros (dentre todo o conjunto de dados) que se encaixam nessa regra. Já a confiança mede o percentual de registros que atendem especificamente

a regra, por exemplo, o percentual de quem compra leite e pão e também compra manteiga.

Para uma regra ser considerada forte, ela deve atender a um certo grau mínimo de suporte e confiança. Um dos mais tradicionais algoritmos de mineração utilizando a estratégia de itens frequentes é o *Apriori* [2]. Diversas variações deste algoritmo, envolvendo o uso de técnicas de *hash*, redução de transações, particionamento e segmentação podem ser encontrados [2]. Mannila [44] apresentada uma variação onde as regras não necessárias são eliminadas. Casanova [11], usa o Algoritmo da Confiança Inversa junto com a Lógica Nebulosa para gerar regras mais precisas. Outros algoritmos também são encontrados: FP-growth e ECLAT (*Equivalence CLASS Transformation*) [93]. Borgelt [4], apresenta uma implementação do FP-growth e faz a comparação dele com outros três algoritmos, dentre eles o *Apriori* e o ECLAT. Em [60], é proposto o método CBMine (*Compressed Binary Mine*) que, segundo os testes, apresentou melhores resultados que os algoritmos tradicionais. Muyeba et all. propõem dois *frameworks* usando lógica nebulosa para a mineração de regras de associação com pesos [52] e para a mineração de itens compostos, chamado CFARM (*Composite Fuzzy ARM*) [51].

Possas et al. [66] propõem uma variação do algoritmo *Apriori* a fim de que um número menor de regras seja gerado. Os resultados apresentaram até 15% de redução. Vasconcelos [81] mostra o uso do *Apriori* para mineração de dados da Web. A abordagem para a mineração de bases em que são gerados muitas regras (colossais), chamada *Pattern-Fusion* é apresentada por Zhu et al. [97].

- Classificações

  As técnicas de classificação podem ser supervisionadas e não-supervisionadas. São usadas para prever valores de variáveis do tipo categóricas. Pode-se, por exemplo, criar um modelo que classifica os clientes de um banco como especiais ou de risco, um laboratório pode usar sua base histórica de voluntários e verificar em quais indivíduos uma nova droga pode ser melhor ministrada. Em ambos os cenários um modelo é criado para classificar a qual categoria um certo registro pertence: especial ou de risco, voluntários A, B ou C.

  Árvores de Decisão (*Decision Trees*) O método de classificação por Árvore de Decisão, funciona como um fluxograma em forma de árvore, onde cada nó (não folha) indica um teste feito sobre um valor (por exemplo, idade > 20). As ligações entre os nós representam os valores possíveis do teste do nó superior, e as folhas indicam a classe (categoria) a qual o registro pertence. Após a árvore de decisão montada, para classificarmos um novo registro, basta seguir o fluxo na árvore (mediante os testes nos nós não-folhas) começando no nó raiz até chegar a uma folha. Pela estrutura que formam, as árvores de decisões podem ser convertidas em Regras de Classificação. O sucesso das árvores de decisão, deve-se ao fato de ser uma técnica extremamente simples, não necessita de parâmetros de configuração e geralmente tem um bom grau de assertividade. Apesar de ser uma técnica extremamente poderosa, é necessário uma análise detalhada dos dados que serão usados para garantir bons resultados. Quinlan [67] apresenta diversas técnicas para reduzir a complexidade das árvores de decisão geradas. Em um artigo recente Yang et al. [90] apresentam um algoritmo para extrair regras acionáveis, ou seja, regras que são realmente úteis

para a tomada de decisões. Um exemplo de árvore de decisão pode ser visto na figura 8.

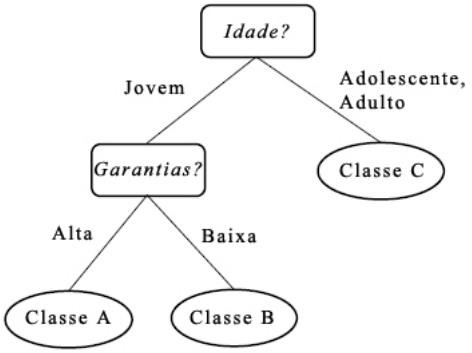

Figura 8: Árvore de Decisão [27]

No final da década de 70, início da década de 80, J. Ross Quinlan desenvolve o ID3 (*Iterative Dichotomiser*), um algoritmo para geração de árvores de decisão. Depois Quinlan desenvolveu o C4.5 (uma versão otimizada do ID3), e que até hoje serve como *benchmark* para novos métodos supervisionados [68]. Foi na mesma época (1984) que um grupo de estatísticos (L. Breiman, J. Friedman, R. Olshen e C. Stone), sem conhecer o trabalho de Quinlan, desenvolveram um algoritmo e publicaram um livro chamado *Classification and Regression Trees (CART)* [7]. Ambos algoritmos são considerados precursores e diversas variações surgiram deles. Eles utilizam a estratégia de dividir-e-conquistar recursiva aplicada de cima para baixo (*top-down*).

Com o argumento de que os algoritmos tradicionais de árvore de decisão precisam carregar todo o conjunto de dados na memória, novos algoritmos capazes de acessar repositórios persistentes foram desenvolvidos: SLIQ [49] e SPRINT [74]. Milagres [50] apresenta uma ferramenta que implementa esses dois algoritmos. Gehrke apresenta um *framework* para auxiliar na execução de algoritmos de classificação e separá-los de questões relativas a escalabilidade [22]. O BOAT (*Bootstrapped Optimistic Algorithm for Tree Construction*) utiliza-se de uma estratégia chamada de "*bootstrapping*"[21]. Chandra apresenta uma otimização do BOAT [12] e uma variação usando lógica nebulosa para o SLIQ [13].

Classificação Bayesiana (*Bayesian Classification*) É uma técnica estatística (probabilidade condicional) baseada no teorema de Thomas Bayes [87]. Segundo o teorema de Bayes, é possível encontrar a probabilidade de um certo evento ocorrer, dada a probabilidade de um outro evento que já ocorreu: Probabilidade(B dado A) = Probabilidade(A e B)/Probabilidade(A). Comparativos mostram que os algoritmos Bayesianos, chamados de *naive Bayes*, obtiveram resultados compatíveis com os métodos de árvore de decisão e redes neurais. Devido a sua simplicidade e o alto poder preditivo, é um dos algoritmos mais utilizados [95]. O algoritmo de *naive Bayes* parte do princípio que não exista relação de dependência entre os atributos. No entanto, nem sempre isto é possível. Nestes casos, uma variação conhecida como *Bayesian Belief Networks*, ou *Bayesian Networks* [55], deve ser utilizada. Em [26], é proposta uma combinação dos algoritmos de *naive Bayes* e Árvore de Decisão para realizar a classificação. Mazlack [47] expõe uma fragilidade na técnica *naive Bayes*.

Classificação Baseada em Regras (*Rule-Based Classification*) A classificação baseada em regras segue a estrutura: *SE* condição *ENTÃO* conclusão (semelhante as regras de associação). Esse tipo de construção geralmente é recuperado de uma árvore de decisão (em estruturas com muitas variáveis, a interpretação dos resultados somente pela árvore de decisão é muito complexa). Uma nova estratégia na obtenção das regras é através de algoritmos de Cobertura Sequencial (*Sequential Covering Algorithm*), diretamente aplicados nos conjuntos de dados. AQ, CN2 e RIPPER são exemplos desses algoritmos. Uma outra forma de obtenção desses regras é através de algoritmos de Regras de Associação.

Redes Neurais (*Neural Networks*) É uma técnica que tem origem na psicologia e na neurobiologia. Consiste basicamente em simular o comportamento dos neurônios. De maneira geral, uma rede neural pode ser vista como um conjunto de unidades de entrada e saída conectados por camadas intermediárias e cada ligação possui um peso associado. Durante o processo de aprendizado, a rede ajusta estes pesos para conseguir classificar corretamente um objeto. É uma técnica que necessita de um longo período de treinamento, ajustes finos dos parâmetros e é de difícil interpretação, não sendo possível identificar de forma clara a relação entre a entrada e a saída. Em contrapartida, as redes neurais conseguem trabalhar de forma que não sofram com valores errados e também podem identificar padrões para os quais nunca foram treinados. Um dos algoritmos mais conhecidos de redes neurais é o *backpropagation* [17], popularizado na década de 80, que realiza o aprendizado pela correção de erros. Na figura 9 podemos ver um exemplo de uma rede neural.

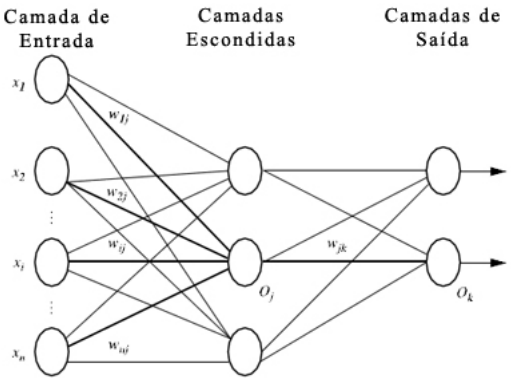

Figura 9: Rede Neural [27]

SVM (*Support Vector Machines*) Apesar de relatos dos anos 60 sobre a técnica de SVM, foi em 1992 que um primeiro artigo foi apresentado por Vladimir Vapnik, Bernhard Boser e Isabelle Guyon [5]. Apesar de ser uma técnica nova, tem chamado muita atenção pelos seus resultados: obtém altos índices de assertividade, permite modelar situações não-lineares complexas gerando modelos de simples interpretação, pode ser usada para relações lineares e não-lineares, entre outros. É utilizado tanto para tarefas de classificação quanto de predição. Atualmente um dos problemas da técnica de SVM é o tempo utilizado no aprendizado. Muitas pesquisas tem se concentrado neste aspecto.

Classificação por Regras de Associação (*Classification by Association Rule*) Recentemente, as técnicas de Regras de Associação estão sendo usadas para a classificação. A ideia geral é buscar por padrões de associações fortes entre

os itens (utilizando-se do conceito de frequência) e as categorias. Basicamente consiste em dois passos: primeiro, os dados de treinamento são analisados para que se obtenha os itens mais frequentes. Em seguida, estes itens são usados para a geração das regras. Alguns estudos demostraram que esta técnica tem apresentado mais assertividade do que algoritmos tradicionais, como o C4.5. Alguns exemplos de algoritmos de classificação são: CBA (*Classification-Based Association*) [42], CMAR (*Classification based on Multiple Association Rules*) [40] e CPAR [92]. [86] mostra uma nova abordagem chamada de CARM (*Classification Association Rule Mining*).

Aprendizado Tardio (*Lazy Learners*) As técnicas de classificação descritas até agora usam um conjunto de dados de treinamento para aprender a classificar um novo registro. Assim, quando são submetidas a um novo registro elas já estão prontas, ou seja, já aprenderam. Existe, no entanto, uma outra categoria de métodos, que somente realizam esse aprendizado quando solicitado para a classificação de um novo registro. Neste caso, o aprendizado é considerado tardio. Apesar de necessitar de um tempo menor de treinamento, esses métodos são muito dispendiosos computacionalmente, pois necessitam de técnicas para armazenar e recuperar os dados de treinamento. Por outro lado, esses métodos permitem um aprendizado incremental.

O algoritmo conhecido como kNN (*k - Nearest Neighbor*), descrito na década de 50, só tornou-se popular na década de 60, com o aumento da capacidade computacional. Basicamente, esse algoritmo armazena os dados de treinamento e quando um novo objeto é submetido para classificação, o algoritmo procura os k registros mais próximos (medida de distância) deste novo registro. O novo registro é classificado na classe mais comum entre todos os k registros mais próximos. No algoritmo chamado de *Case-Based Reasoning* (CBR), ao invés de armazenar os dados de treinamento, ele armazena os casos para a solução dos problemas. Para a classificação de um novo objeto, a base de treinamento é analisada em busca de uma solução. Caso não encontre, o algoritmo sugere a solução mais próxima. Esse algoritmo tem sido bastante utilizado na área de suporte aos usuários, Médica, Engenharia e Direito.

Algoritmo Genético (*Genetic Algorithm*) A ideia dos algoritmos genéticos segue a teoria da evolução. Geralmente, no estágio inicial uma população é definida de forma aleatória. Seguindo a lei do mais forte (evolução), uma nova população é gerada com base na atual, porém, os indivíduos passam por processos de troca genética e mutação. Este processo continua até que populações com indivíduos mais fortes sejam geradas ou que atinga algum critério de parada.

Conjuntos Aproximados (*Rought Set*) É uma técnica que consegue realizar a classificação mesmo com dados impreciso ou errados e é utilizada para valores discretos. A ideia geral destes algoritmos é a de classe de equivalência: eles consideram que os elementos de uma classe são indiscerníveis e trabalham com a ideia de aproximação para a criação das categorias. Por exemplo, uma estrutura (chamada *Rought Set* [25]) é criada para uma classe C. Esta estrutura é cercada por dois outros conjuntos de aproximação (chamados de baixo e alto). O conjunto de baixa aproximação de C contém os registros que certamente são desta classe. O conjunto de alta aproximação contém os registros que não podem ser definidos como não pertencentes à classe C. Um novo registro é classificado mediante a aproximação com um destes conjuntos. Busse [24] faz uma comparação do algoritmo MLEM2 (*Modified Learning from Examples Module, version 2*)) com duas variações. Uma representação dos

conjuntos aproximados pode ser vista na figura 10.

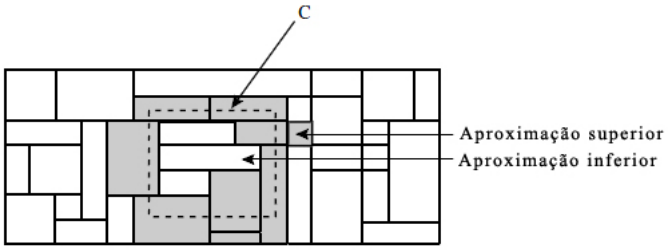

Figura 10: Conjuntos Aproximados [27]

Conjuntos Nebulosos (*Fuzzy Set*) A classificação baseada em regras apresenta um problema relacionado às variáveis contínuas. Elas necessitam de um ponto de corte bem definido, o que às vezes é ruim ou impossível. Por exemplo, SE *salario* > *4.000* ENTÃO *credito* = *ok*. Porém, registros com salário de 3.999 não serão contemplados. Proposta por Lotfi Zadeh em 1965, a ideia dos conjuntos *Fuzzy* é de que, ao invés de realizar um corte direto, essas variáveis sejam discretizadas em categorias e a lógica nebulosa aplicada para definição dos limites destas categorias. Com isso, ao invés de se ter as categorias com limites de corte bem definido, tem-se um certo grau de flexibilidade entre as categorias. Na figura 11 pode-se ver um exemplo de um conjunto nebuloso.

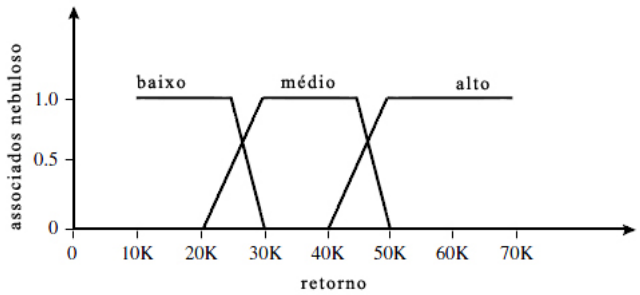

Figura 11: Conjuntos Nebulosos [27]

- Predições Numéricas

  Os métodos de predição visam descobrir um possível valor futuro de uma variável. As predições numéricas visam prever valores para variáveis contínuas. Para a predição de variáveis discretas, as técnicas de classificação já apresentadas podem ser aplicadas. Os métodos mais conhecidos para predição numérica são as regressões, desenvolvidas por Sir Frances Galton (1822 à 1911). Alguns autores tratam as predições numéricas e as regressões como sinônimos, porém, como vimos, alguns métodos de classificação também fazem predições. As técnicas de regressão modelam o relacionamento de variáveis independentes (chamadas preditoras) com uma variável dependente (chamada resposta). As variáveis preditoras são os atributos dos registros, e a resposta é o que se quer predizer.

  Regressão Linear As regressões são chamadas de lineares quando a relação entre as variáveis preditoras e a resposta segue um comportamento linear. Neste caso, é possível criar um modelo no qual o valor de y é uma função linear de x. Exemplo:

y = b + wx. Pode-se utilizar o mesmo princípio para modelos com mais de uma variável preditora. Na figura 12 tem-se um exemplo de uma regressão linear.

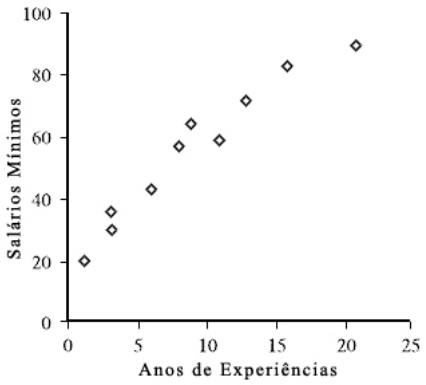

Figura 12: Regressão Linear [27]

Regressão Não-Linear Nos modelos de regressão não-linear, a relação entre as variáveis preditoras e a resposta não segue um comportamento linear. Por exemplo, a relação entre as variáveis pode ser modelada como uma função polinomial. Ainda, para estes casos (Regressão Polinomial), é possível realizar uma conversão pra uma regressão linear. Outros modelos também são encontrados na literatura: *Logistic Regression*, *Poisson Regression* e *Log-Linear Models*.

- Agrupamento

  As técnicas de agrupamento são consideradas como não supervisionadas. Dado um conjunto de registros, são gerados agrupamentos (ou *cluster*), contendo os registros mais semelhantes. Em geral, as medidas de similaridade usadas são as medidas de distâncias tradicionais (Euclidiana, *Manhattan*, etc). Os elementos de um *cluster* são considerados similares aos elementos no mesmo *cluster* e dissimilares aos elementos nos outros *clusters*. Por trabalhar com o conceito de distância (similaridade) entre os registros, geralmente é necessário realizar a transformação dos diferentes tipos de dados (ordinais, categóricos, binários, intervalos) para uma escala comum, exemplo [0.0, 1.0]. Podemos classificar os algoritmos de agrupamento nas seguintes categorias:

  Métodos de Particionamento (*Partitioning Methods*) Dado um conjunto D de dados com n registros e k o número de agrupamentos desejados, os algoritmos de particionamento organizam os objetos em k agrupamentos, tal que $k <= n$. Os algoritmos mais comuns de agrupamento são: *k-Means* e *k-Medoids*.

  k-Means Esse algoritmo usa o conceito da centroide. Dado um conjunto de dados, o algoritmo seleciona de forma aleatória k registros, cada um representando um agrupamento. Para cada registro restante, é calculada a similaridade entre o registro analisado e o centro de cada agrupamento. O objeto é inserido no agrupamento com a menor distância, ou seja, maior similaridade. O centro do *cluster* é recalculado a cada novo elemento inserido. Diferentes variações surgiram: implementando otimizações para escolha do valor do k, novas medidas de dissimilaridade e estratégias para o cálculo do centro do agrupamento. Uma variação bem conhecida do *k-Means* é o *k-Modes*. Nesse caso, ao invés de calcular o centro do agrupamento através da média de distância dos registros, ele usa a moda.

k-Medoids É uma variação do *k-Means*. Neste algoritmo, ao invés de calcular o centro do agrupamento e usá-lo como referência, trabalha-se com o conceito do objeto mais central do agrupamento. As variações mais conhecidas são os algoritmos PAM (*Partitioning Around Medoids*) e CLARA (*Clustering LARge Applications*).

Métodos Hierárquicos (*Hierarchical Methods*) A ideia básica dos métodos hierárquicos é criar o agrupamento por meio da aglomeração ou da divisão dos elementos do conjunto. A forma gerada por estes métodos é um dendrograma (gráfico em formato de árvore, conforme figura 13). Dois tipos básicos de métodos hierárquicos podem ser encontrados: Aglomerativos e Divisivos.

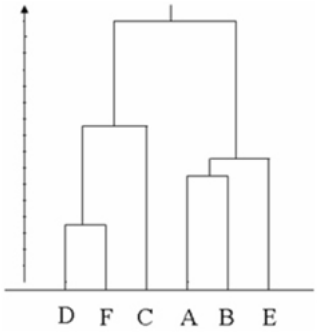

Figura 13: Exemplo de um dendrograma [57]

Aglomerativos Adotam uma estratégia *bottom-up* onde, inicialmente, cada objeto é considerado um agrupamento. A similaridade é calculada entre um agrupamento específico e os outros agrupamentos. Os agrupamentos mais similares vão se unindo e formando novos agrupamentos. O processo continua, até que exista apenas um agrupamento principal. Os algoritmos AGNES (*AGglomerative NESting*) e CURE (*Clustering Using Representatives*) utilizam esta estratégia.

Divisivos Adotam uma estratégia *top-down*, onde inicialmente todos os objetos estão no mesmo agrupamento. Os agrupamentos vão sofrendo divisões, até que cada objeto represente um agrupamento. O algoritmo DIANA (*DIvisive ANAlysis*) utiliza esta estratégia.

Métodos Baseados na Densidade (*Density-Based Methods*) Os métodos de particionamento e hierárquicos geram agrupamentos de formato esféricos (distribuição dos valores dos dados é mais esparsa). No entanto, existem situações em que essa distribuição é mais densa e que tais métodos não apresentam resultados satisfatórios. Os métodos baseado na densidade conseguem melhores resultados. Destacamos os algoritmos: DBSCAN (*A Density-Based Clustering Method Based on Connected Regions with Sufficiently High Density*), OPTICS (*Ordering Points to Identify the Clustering Structure*) e DENCLUE (*DENsity-based CLUstEring*). Em [19], é proposta uma técnica usando a estratégia do *Simulated Annealing*.

Métodos Baseados em Grade (*Grid-Based Methods*) Métodos baseado em grades, utilizam-se da estrutura de grades e dividem os registros nas células desta grade. Apresentam um tempo de processamento bem rápido. Os principais algoritmos são o STING (*STatistical INformation Grid*) e o WaveCluster (*Clustering UsingWavelet Transformation*).

Métodos Baseados em Modelos (*Model-Based Methods*) Os métodos baseados em modelos criam um modelo para cada agrupamento e tentam identificar o melhor modelo para cada objeto. Este método parte da ideia de que os dados são gerados por uma série de probabilidade de distribuições. Os algoritmos EM (*Expectation-Maximization*), uma variação do *k-Means*, COBWEB e CLASSIT implementam o método de agrupamento baseado em modelos.

Apesar de cada método possuir suas peculiaridades e apresentar melhor resultado com um certo tipo de dado, não existe uma classificação única para a escolha e aplicação destes métodos [20].

5 Limitações

Apesar da grande potencialidade oferecida pela Mineração de Dados, alguns fatores devem ser analisados. Wang et all. [85] discutem como alguns desses fatores podem prejudicar as técnicas de mineração:

- As relações entre os atributos precisam ser muito bem definidas, caso contrário os resultados podem ser mal interpretados;
- Permitir que o processo de treinamento execute por muito tempo, até que se consiga obter indícios que possam levar à conclusões factíveis;
- Gerar subsídios para uma conclusão errada tornando-a mais plausível. Porém, uma interpretação falha pode disfarçar as falhas nos dados;
- Usar um grande número de variáveis.

Alguns outros autores mencionam três outros fatores: o alto conhecimento exigido dos usuários, a escolha do repositório e o uso de muitas variáveis. Wang et al. categorizam os problemas encontrados na mineração de dados em quatro grupos [85]: estatísticos, precisão dos dados e padronizações, técnicos (problemas encontrados em diversos métodos, tais como Redes Neurais, Árvores de Decisão, Algoritmos Genéticos e Lógica Nebulosa) e organizacionais. Segundo [72], a Mineração de Dados apesar de revelar padrões e relacionamentos, não os explica. Além disto, alguns relacionamentos e padrões casuais não são capturados.

Outra questão que trás grande impacto na utilização da Mineração de Dados refere-se à privacidade e à legislação. Trabalhar com dados sobre o indivíduo trás implicações que precisam ser consideradas e analisadas [27]. Seifert [72], mostra que o Congresso Americano já aprova leis para gerir o uso da Mineração de Dados e as questões de privacidade ligadas a elas. Zhan et all. [94] apresentam um modelo para se trabalhar a privacidade dos dados.

Em [20], alguns desafios que precisam ser superados são apresentados:

- Técnicas para lidar com base de dados cada vez maiores, chegando a casa dos *Terabytes*;
- Cada vez mais as tabelas possuem mais atributos, aumentando o espaço de busca (alta dimensionalidade);
- Os modelos são construídos usando um conjunto limitado de dados, que podem não conter todos os padrões e com isto, ao serem submetidos a novos dados, se comportam de maneira errônea;

- A velocidade com que os dados mudam faz com que os modelos gerem resultados inválidos;
- O problema da baixa qualidade dos dados;
- Complexidade dos relacionamentos entre os atributos;
- Tornar os padrões descobertos mais legíveis, facilitando o entendimento e a interpretação pelo usuário;
- A baixa interação e a dificuldade de inserção de conhecimento prévio nos modelos;
- Os sistemas cada vez mais dependem de outros sistemas, gerando problemas de integração.

6 Mineração de Estruturas Complexas

A Mineração de Dados foi inicialmente concebida para utilizar-se de repositórios estruturados de dados (Banco de Dados, *Data Warehouse*, Arquivos, etc). Porém, atualmente os dados são representados por diversos formatos: Não estruturado, Espacial e Temporal, Multimídia, *Web*, entre outros. E cada vez mais, existe a necessidade da mineração nestes tipos de dados. Com isto, uma área que vem sendo bastante pesquisada é a Mineração de Dados em estruturas complexas. Em Han et al. [27], algumas dessas estruturas são abordadas:

Mineração de Fluxo de Dados Algumas aplicações trafegam um volume altíssimo de dados, temporalmente ordenados, voláteis e potencialmente infinito. Minerar estas informações após terem sido armazenadas é uma tarefa inviável. Ao invés disso, a mineração ocorre à medida em que os dados são lidos. Kid et all. [34] propõem um *framework* para extração de padrões temporais de fluxos de dados. Koh et all. [36] propõem um algoritmo chamado *appearing-bit-sequence-based incremental mining* para um reconhecimento incremental dos padrões em fluxos de dados.

Mineração de Séries Temporais Bases de Séries Temporais são aquelas que armazenam informações de um certo evento em um intervalo de tempo definido. Por exemplo, bases que armazenam o valor das ações de um mercado, velocidade do vento, medidas da atmosfera. O processo de identificação de padrões em bases desse tipo envolve outras técnicas e análises. Em [29], é apresentado um trabalho para a detecção de fatores de risco na área médica usando a mineração de séries temporais através de algoritmos de agrupamento.

Mineração de Grafos Grafos são muito importantes na modelagem de estruturas complexas, como circuitos, imagens, proteínas, redes biológicas, redes sociais, etc. Variações de algoritmos tradicionais e novos algoritmos tem sido desenvolvido para esse fim [41].

Mineração de Relacionamentos As redes sociais representam o relacionamento (*link*) entre as entidades envolvidas (similar a uma estrutura de grafos). Nas últimas décadas elas tem chamado muita atenção pela riqueza de padrões que podem ser extraídos. Matsuo [46] apresenta uma abordagem para a mineração de redes sociais na internet.

Mineração de Dados Multirelacionais A grande maioria das bases relacionais armazena seus dados de forma normalizada e distribuída. As tabelas que compõem essa base são então relacionadas entre si. No entanto, as técnicas tradicionais de Mineração de Dados

utilizam-se de estruturas mais simples. Devido a isso, as diversas tabelas devem ser agrupadas e simplificadas. Esse processo gera diversos problemas, tais como: variáveis desnecessárias ou duplicadas, complexidade dos dados, tempo de análise e entendimento, etc. A Mineração de Dados Multirelacionais visa criar algoritmos que utilizam as estruturas originais das bases, sem a necessidade de uma conversão.

Mineração de Objetos Diferente das bases relacionais, que armazenam os dados de uma forma estruturada (tabelas), as bases orientadas a objetos, guardam os dados em forma de objetos (formados por um identificador, atributos e métodos).

Mineração de Dados Espaciais Bases espaciais envolvem um conjunto de dados relacionados às questões espaciais, tais como mapas. Possuem informações de topologia e distância organizadas de forma totalmente diferente das bases relacionais. A mineração espacial visa identificar os padrões armazenados nesses dados de uma forma implícita.

Mineração de Dados Multimídia Bases de dados multimídia armazenam dados em formato de áudio, vídeo, imagens, gráficos, texto, etc. Em [89], tem-se um *survey* de reconhecimento de padrões faciais em imagens. Malerba [43] apresenta uma proposta para geração de regras de associação de documentos textuais escaneados.

Mineração de Textos Grande parte dos dados de uma instituição é armazenada de forma semi-estruturada e não-estruturada, através de textos, e-mail, artigos, documentos (atas, memorandos, oficios), etc. A busca de padrões e conhecimento nestes documentos é muito comum. Porém, na maioria das vezes, o resultado obtido é falho: documentos não relacionados, volume muito alto de informações dispensáveis, entre outros. A mineração de textos, visa ajudar neste processo.

Mineração da Internet A mineração da Internet tem sido alvo de recentes pesquisas, pois ela reúne em seu ambiente, quase a totalidade dos tipos de estruturas complexas e simples que existem. Além disso, possui um volume de dados gigantesco. Atende às diversas necessidades e possui os mais diversos conteúdos. A Mineração da Internet (ou *Web Mining*), consiste em minerar as estruturas de ligação, o conteúdo, os padrões de acesso, classificação de documentos, entre outras. Em [75], os conceitos da mineração na internet podem ser analisados. Shimada et all. [76] propõem um método para minerar a opinião das pessoas sobre determinados produtos. Em [45], é proposta uma abordagem para a geração de um mapa de tópicos de páginas da internet.

7 Ferramentas

Diversas ferramentas foram desenvolvidas no intuito de tornar a aplicação da Mineração de Dados uma tarefa menos técnica, e com isto possibilitar que profissionais de outras áreas possam fazer usa dela. Neste sentido, o mercado de ferramentas de mineração de dados tem se tornado bastante atraente.

Clementine Uma das ferramentas líder de mercado, desenvolvida pela SPSS o Clementine suporta o processo CRISP-DM, além de possuir outras facilidades [80].

SAS Enterprise Miner Suite Ferramenta desenvolvida pela empresa SAS. É uma das ferramentas mais conhecidas para mineração. Possui modulos para trabalhar em todas as etapas do processo de mineração [70].

SAS Text Miner Ferramenta da SAS para mineração de textos [71].

WEKA É uma das melhores ferramentas livre. Possui uma série de algoritmos para as tarefas de mineração. Os algoritmos podem ser aplicados diretamente da ferramenta, ou utilizados por programas Java. Fornece as funcionalidades para pré-processamento, classificação, regressão, agrupamento, regras de associação e visualização [82]. Atualmente faz parte da ferramenta de *BI OpenSource Pentaho* [62]. Em [88] a ferramenta é apresentada em detalhes.

Oracle Data Mining (ODM) É uma ferramenta para a Mineração de Dados desenvolvida pela Oracle para o uso em seu banco de dados ORACLE [59].

KXEN Analytic Framework Ferramenta de Mineração de Dados comercial que utiliza conceitos do Professor Vladimir Vapnik como Minimização de Risco Estruturada (Structured Risk Minimization ou SRM) e outros [37].

IBM Intelligent Miner Ferramenta de mineração da IBM para a mineração de dados no banco de dados DB2 [31].

Pimiento Ferramenta livre para mineração de textos [63].

MDR Ferramenta livre em Java para detecção de interações entre atributos utilizando o método da *multifactor dimensionality reduction* (MDR) [18].

LingPipe Ferramenta de mineração livre voltada para análise linguística [3].

KNIME Plataforma de mineração de dados aberta, que implementa o paradigma de *pipelining* de dados [35].

8 Considerações Finais

A Mineração de Dados tornou-se uma ferramenta de apoio com papel fundamental na gestão da informação dentro das organizações. A manipulação dos dados e a análise das informações de maneira tradicional tornou-se inviável devido ao grande volume de dados (coletados diariamente e armazenados em bases históricas). Descobrir padrões implícitos e relacionamentos em repositórios que contém um grande volume de dados de forma manual, deixou de ser uma opção. As técnicas de mineração passaram a estar presentes no dia a dia.

Os dados são considerados hoje como o principal ativo de um projeto de software. Isso se deve, além da redução nos custos de aquisição de *hardware* e *software*, ao desenvolvimento de técnicas capazes de extrair, de forma otimizada, a informação contida, e muitas vezes implícita, nestes dados.

Apesar dos bons resultados obtidos com à aplicação da Mineração de Dados, os desafios ainda são muitos. Diversos problemas relativos ao uso da mineração (tais como a segurança dos dados e a privacidade dos indivíduos), juntamente com o aumento na complexidade das estruturas de armazenamento, criam cenários complexos e desafiadores. Além disso, novas tendências como a Web Semântica, exigem que variações dos algoritmos tradicionais sejam desenvolvidas.

A Mineração de Dados atualmente caminha para uma popularização. As ferramentas, cada vez mais amigáveis e fáceis de serem usadas por usuários que não sejam especialistas em

mineração, desempenham um papel fundamental nesse sentido. Esta popularização é fundamental para o crescimento e a consolidação da Mineração de Dados.

Não resta dúvida de que essa é uma área extremamente promissora e que, apesar dos resultados já obtidos, ainda tem muito para oferecer.

9 Agradecimentos

Ao Prof. Dr. Cedric Luiz de Carvalho, pela avaliação do presente texto e pelas sugestões feitas, as quais muito contribuíram para a melhoria do texto original.

Referências

[1] AGRAWAL, R; IMIELINSKI, T; SWAMI, A. **Mining association rules between sets of items in large databases**. Proc. of the ACM SIGMOD, p. 207–216, 1993.

[2] AGRAWAL, R; SRIKANT, R. **Fast algorithms for mining association rules**. 20th International Conference on Very Large Data Bases, p. 487–499, 1994.

[3] ALIAS-I. **LingPipe**. `http://alias-i.com/lingpipe/`, acessado em Maio de 2009.

[4] BORGELT, C. **An implementation of the FP-growth algorithm**, 2005.

[5] BOSER, B. E; GUYON, I. M; VAPNIK, V. N. **A training algorithm for optimal margin classifiers**. In: PROCEEDINGS OF THE 5TH ANNUAL ACM WORKSHOP ON COMPUTATIONAL LEARNING THEORY, p. 144–152. ACM Press, 1992.

[6] BRAMER, M. **Undergraduate Topics in Computer Science - Principles of Data Mining**. Springer, 2007.

[7] BREIMAN, L; FRIEDMAN, J; OLSHEN, R; STONE, C. **Classification and Regression Trees**. Chapman and Hall/CRC, 1984.

[8] CABENA, P; HADJINIAN, P; STADLER, R; JAAPVERHEES; ZANASI, A. **Discovering Data Mining: From Concept to Implementation**. Prentice Hall, 1998.

[9] CABRAL, L. S; SIEBRA, S. A. **Identificação de competências em currículos usando ontologias: uma abordagem teórica**, 2006.

[10] CANADA, S. **Statistics: Power from Data!** `http://www.statcan.gc.ca/edu/power-pouvoir/toc-tdm/5214718-eng.htm`, acessado em abril de 2009.

[11] CASANOVA, A. A; LABIDI, S. **Algoritmo da Confiança Inversa para Mineração de Dados Baseado em Técnicas de Regras de Associação e Lógica Nebulosa**. XXV Congresso da Sociedade Brasileira de Computação, 2005.

[12] CHANDRA, B; VARGHESE, P. **On improving efficiency of sliq decision tree algorithm**. International Joint Conference on Neural Networks - IJCNN, p. 66–71, 2007.

[13] CHANDRA, B; VARGHESE, P. **Fuzzy sliq decision tree algorithm**. IEEE Transactions on Cybernetics, 38:1294–1301, 2008.

[14] CHAPMAN, P; CLINTON, J; KERBER, R; KHABAZA, T; REINARTZ, T; SHEARER, C; WIRTH, R. **CRISP-DM 1.0**. CRISP-DM consortium, 2000.

[15] Chen, H; Reid, E; Sinai, J; Silke, A; Ganor, B, editors. **Terrorism Informatics - Knowledge Management and Data Mining for Homeland Security**. Springer, 2008.

[16] CIOS, K. J; PEDRYCZ, W; SWINIARSKI, R. W; KURGAN, L. A. **Data Mining - A Knowledge Discovery Approach**. Springer, 2007.

[17] CROCHAT, P; FRANKLIN, D. **An introduction to bayesian networks and their contemporary applications**. `http://ieee.uow.edu.au/~daniel/software/libneural/BPN_tutorial/BPN_English/BPN_English/`, acessado em Maio de 2009.

[18] DARTMOUTH. **MDR**. `http://www.multifactordimensionalityreduction.org/`, acessado em Maio de 2009.

[19] DUCZMAL, L; ASSUNÇÃO, R. **A simulated annealing strategy for cluster detection**.

[20] FAYYAD, U; PIATETSKY-SHAPIRO, G; SMYTH, P. **From Data Mining to Knowledge Discovery in Databases**. American Association for Artificial Intelligence, 1996.

[21] GEHRKE, J; GANTI, V; RAMAKRISHNAN, R; LOH, W.-Y. **BoatŮoptimistic decision tree construction**. Proceedings of the 1999 ACM SIGMOD, p. 169–180, 1999.

[22] GEHRKE, J; RAMAKRISHNAN, R; GANTI, V. **Rainforest Ů a framework for fast decision tree construction of large datasets**. Data Mining and Knowledge Discovery, 4:127–162, 2000.

[23] GONZÁLEZ-ARANDA, P; MENASALVAS, E; RUIZ, S. M. C; SEGOVIA, J. **Towards a methodology for data mining project development: The importance of abstraction**. In: STUDIES IN COMPUTATIONAL INTELLIGENCE, p. 165Ű178. Springer-Verlag, 2008.

[24] GRZYMALA-BUSSE, J. W. **Mlem2 rule induction algorithms: With and without merging intervals**. In: STUDIES IN COMPUTATIONAL INTELLIGENCE, p. 153Ű164. Springer-Verlag, 2008.

[25] GRZYMALA-BUSSE, J. W. **Three approaches to missing attribute values: A rough set perspective**. In: STUDIES IN COMPUTATIONAL INTELLIGENCE, p. 139Ű152. Springer-Verlag, 2008.

[26] HALL, M; FRANK, E. **Combining naive bayes and decision tables**. In 2008 FLAIRS Conference - AAAI, 2008.

[27] HAN, J; KAMBER, M. **Data Mining: Concepts and Techniques**. Elsevier, 2006.

[28] HAND, D; MANNILA, H; SMYTH, P. **Principles of Data Mining**. MIT Press, 2001.

[29] HIRANO, S; TSUMOTO, S. **Detection of risk factors as temporal data mining**. In: PAKDD WORKSHOPS, p. 143–156. Springer-Verlag, 2008.

[30] HORNICK, M. F; MARCADÉ, E; VENKAYALA, S. **Java Data Mining: Strategy, Standard, and Practice A Practical Guide for Architecture, Design, and Implementation**. Elsevier, 2007.

[31] IBM. **Intelligent Miner**. `http://www-01.ibm.com/software/data/iminer/`, acessado em Maio de 2009.

[32] KDNUGGETS.COM. **KDNuggets**. `http://KDNuggets.com`, acessado em Maio de 2009.

[33] KEIM, D. A. **Information visualization and visual data mining**. IEEE Transactions on Visualization and Computer Graphics, p. 1–8, 2002.

[34] KIDA, T; SAITO, T; ARIMURA, H. **Flexible framework for time-series pattern matching over multi-dimension data stream**. In: PAKDD WORKSHOPS, p. 1–12. Springer-Verlag, 2008.

[35] KNIME.COM. **KNIME**. `http://www.knime.org/`, acessado em Maio de 2009.

[36] KOH, J.-L; CHOU, P.-M. **Incrementally mining recently repeating patterns over data streams**. In: PAKDD WORKSHOPS, p. 26–37. Springer-Verlag, 2008.

[37] KXEN. **KXEN**. `http://www.aexis.com/KXEN-Analytic-Framework.htm`, acessado em Maio de 2009.

[38] LADEIRA, M; OLIVEIRA, M. G; ARAÚJO, M. E. C. **Lupa Digital: Agilização da Busca Decadactilar na Identificação Criminal Através de Mineração de Dados**. XXV Congresso da Sociedade Brasileira de Computação, 2005.

[39] LAROSE, D. T. **Discovering Knowledge in Data: An Introduction to Data Mining**. John Wiley and Sons, Inc, 2005.

[40] LI, W; HAN, J; PEI, J. **Cmar: Accurate and efficient classification based on multiple class-association rules**, 2001.

[41] LI, W; NG, W.-K; ONG, K.-L. **Graph-Based Data Mining**, chapter XI, p. 291–307. Idea Group Inc., 2007.

[42] LIU, B; HSU, W; MA, Y. **Integrating classification and association rule mining**. AAAI Workshop of Knowledge Discovery in Databases, 1998.

[43] MALERBA, D; BERARDI, M; CECI, M. **Discovering Spatio-Textual Association Rules in Document Images**, chapter VIII, p. 176–197. IGI, 2008.

[44] MANNILA, H; TOIVONEN, H; VERKAMO, A. I. **Efficient algorithms for discovering association rules**. AAAI Workshop of Knowledge Discovery in Databases, 1994.

[45] MASE, M; YAMADA, S; NITTA, K. **Extracting topic maps from web pages**. In: PAKDD WORKSHOPS, p. 169–180. Springer-Verlag, 2008.

[46] MATSUO, Y; MORI, J; ISHIZUKA, M. **Social Network Mining from the Web**, chapter VII, p. 149–175. IGI, 2008.

[47] MAZLACK, L. J. **Naive rules do not consider underlying causality**. In: STUDIES IN COMPUTATIONAL INTELLIGENCE, p. 213Ű229. Springer-Verlag, 2008.

[48] MCCUE, C. **Data Mining and Predictive Analysis - Intelligence Gathering and Crime Analysis**. Elsevier, 2007.

[49] MEHTA, M; AGRAWAL, R; RISSANEN, J. **Sliq: A fast scalable classifier for data mining**. Procs. of the 5th EDBT, p. 18Ű32, 1996.

[50] MILAGRES, R; SANTOS, L. F; PLASTINO, A. **Midas-uff: Uma ferramenta para mineração de dados**. `http://www.ic.uff.br/~lsantos/publ/sims2004.pdf`, acessado em Maio de 2009, 2004.

[51] MUYEBA, M; KHAN, M. S; COENEN, F. **A framework for mining fuzzy association rules from composite items maybin**. In: PAKDD WORKSHOPS, p. 62–74. Springer-Verlag, 2008.

[52] MUYEBA, M; KHAN, M. S; COENEN, F. **Fuzzy weighted association rule mining with weighted support and confidence framework**. In: PAKDD WORKSHOPS, p. 49–61. Springer-Verlag, 2008.

[53] MYATT, G. J. **Making Sense of Data - A Practical Guide to Exploratory Data Analysis and Data Mining**. John Wiley and Sons, Inc, 2007.

[54] MYATT, G. J; JOHNSON, W. P. **Making Sense of Data II - A Practical Guide to Data Visualization, Advanced Data Mining Methods, and Applications**. John Wiley and Sons, Inc, 2009.

[55] NIEDERMAYER, D. **An introduction to bayesian networks and their contemporary applications**. `http://en.wikipedia.org/wiki/Thomas_Bayes`, acessado em Maio de 2009., 1998.

[56] NIST/SEMATECH. **NIST/SEMATECH e-Handbook of Statistical Methods**. `http://www.itl.nist.gov/div898/handbook/`, acessado em abril de 2009.

[57] OLIVEIRA, R. R; CARVALHO, C. L. **Algoritmos de agrupamento e suas aplicações**. Technical report, Universidade Federal de Goiás, 2008.

[58] OLSON, D. L; DELEN, D. **Advanced Data Mining Techniques**. Springer, 2008.

[59] ORACLE. **Oracle**. `http://www.oracle.com/technology/products/bi/odm/index.html`, acessado em Maio de 2009.

[60] PALANCAR, J; LEÓN, R; PAGOLA, J. M; HECHAVARRÍA, A. **A compressed vertical binary algorithm for mining frequent patterns**. In: STUDIES IN COMPUTATIONAL INTELLIGENCE, p. 197Ű211. Springer-Verlag, 2008.

[61] PECHENIZKIY, M; PUURONEN, S; TSYMBAL, A. **Does relevance matter to data mining research?** In: STUDIES IN COMPUTATIONAL INTELLIGENCE, p. 251Ű275. Springer-Verlag, 2008.

[62] PENTAHO. **Pentaho BI Tools**. `http://www.pentaho.org`, acessado em Maio de 2009.

[63] PIMIENTO. **Pimiento**. `http://erabaki.ehu.es/jjga/pimiento/`, acessado em Maio de 2009.

[64] PM, M; DW, A. **UCI Repository of Machine Learning Databases**. `http://www.ics.uci.edu/`, acessado em abril de 2009.

[65] PONNIAH, P. **Data Warehousing Fundamentals: A Comprehensive Guide for IT Professionals**. John Wiley and Sons, Inc, 2001.

[66] PÔSSAS, B; JR., W. M; CARVALHO, M; RESENDE, R. **Using quantitative information for efficient association rule generation**. ACM SIGMOD Record, 29:19 – 25, 2000.

[67] QUINLAN, J. R. **Simplifying decision trees**. Technical report, Massachusetts Institute of Technology, 1986.

[68] QUINLAN, J. R. **C4.5: Programs for Machine Learning**. Morgan Kaufmann Publishers Inc., 1992.

[69] REZENDE, S. O. **Mineração de Dados**. XXV Congresso da Sociedade Brasileira de Computação, 2005.

[70] SAS. **Enterprise Miner Suite**. `http://www.sas.com/technologies/analytics/datamining/miner/index.html`, acessado em Maio de 2009.

[71] SAS. **SAS Text Miner**. `http://www.sas.com/technologies/analytics/datamining/textminer/index.html`, acessado em Maio de 2009.

[72] SEIFERT, J. W. **Crs report for congress - data mining: An overview**. Technical report, Congressional Research Service, 2004.

[73] SELIYA, N; KHOSHGOFTAAR, T. M. **Software Quality Modeling With Limited Apriori Defect Data**, chapter Chapter 1, p. 1–16. Idea Group Publishing, 2007.

[74] SHAFER, J; AGRAWAL, R; MEHTA, M. **Sprint: A scalable parallel classifier for data mining**. Procs. of the 22nd VLDB, p. 544Ű555, 1996.

[75] SHI, Z; MA, H; HE, Q. **Web Mining: Extracting Knowledge from the World Wide Web**, chapter XIV, p. 197–208. Springer, 2009.

[76] SHIMADA, K; HASHIMOTO, D; ENDO, T. **A graph-based approach for sentiment sentence extraction**. In: PAKDD WORKSHOPS, p. 38–48. Springer-Verlag, 2008.

[77] SHLENS, J. **A Tutorial on Principal Component Analysis**. Salk Insitute for Biological Studies and University of California, 2 edition, December 2005.

[78] Simoff, S. J; Böhlen, M. H; Mazeika, A, editors. **Visual Data Mining - Theory, Techniques and Tools for Visual Analytics**. Springer, 2008.

[79] SMITH, L. I. **A tutorial on Principal Components Analysis**, February 2002.

[80] SPSS. **Clementine**. `http://www.spss.com.br/clementine/index.htm`, acessado em Maio de 2009.

[81] VASCONCELOS, L. M. R; CARVALHO, C. L. **Aplicação de regras de associação para mineração de dados na web**. Technical report, Universidade Federal de Goiás, 2004.

[82] WAIKATO, U. O. **WEKA**. `http://www.cs.waikato.ac.nz/ml/weka/`, acessado em Maio de 2009.

[83] Wang, J, editor. **Encyclopedia of Data Warehousing and Mining**. Idea Group Reference, 2005.

[84] WANG, J; HU, X; ZHU, D. **Data Mining in Public Administration**, chapter XVIII, p. 556–567. IGI, 2008.

[85] WANG, J; HU, X; ZHU, D. **Minimizing the Minus Sides of Mining Data**. In: Taniar, D, editor, DATA MINING AND KNOWLEDGE DISCOVERY TECHNOLOGIES, p. 254–279. IGI Publishing, 2008.

[86] WANG, Y. J; XIN, Q; COENEN, F. **Mining efficiently significant classification association rules**. In: STUDIES IN COMPUTATIONAL INTELLIGENCE, p. 443Ű467. Springer-Verlag, 2008.

[87] WIKIPEDIA. **Thomas bayes**. `http://en.wikipedia.org/wiki/Thomas_Bayes`, acessado em Maio de 2009.

[88] WITTEN, I. H; FRANK, E. **Data Mining - Practical Machine Learning Tools and Techniques**. Elsevier, 2005.

[89] YANG, M.-H; KRIEGMAN, D. J; AHUJA, N. **Detecting faces in images: A survey**. IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, 24(1), January 2002.

[90] YANG, Q; YIN, J; LING, C; PAN, R. **Extracting actionable knowledge from decision trees**. IEEE Transactions on Knowledge and Data Engineering, 19(1):43–56, 2007.

[91] YE, N. **THE HANDBOOK OF DATA MINING**. LAWRENCE ERLBAUM ASSOCIATES, 2003.

[92] YIN, X; HAN, J. **Cpar: Classification based on predictive association rules**, 2001.

[93] ZAKI, M. J. **Scalable algorithms for association mining**. In: IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, volume 12 de **3**, p. 372–390, May 2000.

[94] ZHAN, J; CHANG, L; MATWIN, S. **How to prevent private data from being disclosed to a malicious attacker**. In: STUDIES IN COMPUTATIONAL INTELLIGENCE, p. 517–528. Springer-Verlag, 2008.

[95] ZHANG, H. **The optimality of naive bayes**. In 2004 FLAIRS Conference - AAAI, 2004.

[96] ZHOU, Z.-H. **Three perspectives of data mining**. Artificial Intelligence Journal, p. 139–146, 2003.

[97] ZHU, F; YAN, X; YU, J. H; CHENG, P. H. **Mining colossal frequent patterns by core pattern fusion**. IEEE 23rd International Conference on Data Engineering, 2007. (to appear).