Pós-processamento de Regras de Associação

Edson Augusto Melanda

Draft impresso em 31 de janeiro de 2005

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito: //2004

Assinatura:_____

Pós-processamento de Regras de Associação

Edson Augusto Melanda

Orientadora: *Prof^ª Dr^a Solange Oliveira Rezende*

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos necessários para obtenção do título de Doutor em Ciências de Computação e Matemática Computacional.

USP - São Carlos
outubro/2004

*“É difícil dizer o que é impossível,
a fantasia de ontem é a esperança de hoje
e a realidade de amanhã...”*

Robert Goddard

Este documento foi preparado utilizando-se o formatador de textos L^AT_EX. Sua bibliografia é gerada automaticamente pelo B_IB_TE_X, utilizando o estilo Chicago.

*À minha Fernanda,
pela existência e carinho,
sempre.*

À Solange Oliveira Rezende.

Agradecimentos

Aos meus pais, Edson e Diva, e meus irmãos Carlos e Vanessa pelo amor, carinho e respeito sempre demonstrados e pela compreensão nos momentos de ausência.

Aos meus sogros Célia e Milton pelo incentivo constante

À “professora” Solange que foi muito mais que uma orientadora, uma verdadeira amiga. Muito obrigado por seu apoio, incentivo e paciência. Você é um exemplo de pessoa que devemos nos espelhar.

À Jaqueline amiga e companheira em todos os momentos.

Às amigas Claudinha, Mariza e Patricia pelos momentos de alegria e descontração.

Às três mosqueteiras Camila, Marina, Roberta pela ajuda imprescindível na correção da tese, especialmente no “multirão final”.

Aos amigos, Daniel, Marcos Paula, Marcos Aurélio, Valmir, Gustavo, Cláudia Milaré, Veronica, Huei, Katti, Eduardo, Ana Carolina, Walter, Rodrigo, Jean, Gedson, Ana Carolina, entre vários outros amigos que contribuíram na minha formação e também pelos bons momentos que passamos juntos.

Aos amigos Alex, Gilson, Thiago e Paulo pela contribuição nos experimentos e implementações.

Aos amigos Ronaldo e Edson Takashi por ajudar nas minhas dúvidas sobre L^AT_EX e Linux.

À professora Maria Carolina, juntamente com a professora Solange torna o LABIC um ótimo ambiente, propício ao trabalho.

A todos do LABIC pelos bons momentos e por fazer do laboratório um ambiente agradável, propício para a realização de qualquer trabalho.

A todos os funcionários do ICMC pela atenção, respeito, dedicação e educação.

À Universidade de São Paulo pela oportunidade.

À Universidade Federal de São Carlos e ao seu Departamento de Engenharia Civil, pelo afastamento parcial concedido para a realização deste trabalho.

Aos colegas do Núcleo de Geoprocessamento (NGeo) pela compreensão e grande apoio dado durante o período de realização do doutorado, em especial ao amigo Sérgio.

Resumo

A demanda por métodos de análise e descoberta de conhecimento em grandes bases de dados tem fortalecido a pesquisa em Mineração de Dados. Dentre as tarefas associadas a essa área, tem-se Regras de Associação. Vários algoritmos foram propostos para tratamento de Regras de Associação, que geralmente têm como resultado um elevado número de regras, tornando o Pós-processamento do conhecimento uma etapa bastante complexa e desafiadora. Existem medidas para auxiliar essa etapa de avaliação de regras, porém existem lacunas referentes a inexistência de um método intuitivo para priorizar e selecionar regras. Além disso, não é possível encontrar metodologias específicas para seleção de regras considerando mais de uma medida simultaneamente. Esta tese tem como objetivo a proposição, desenvolvimento e implementação de uma metodologia para o Pós-processamento de Regras de Associação. Na metodologia proposta, pequenos grupos de regras identificados como potencialmente interessantes são apresentados ao usuário especialista para avaliação. Para tanto, foram analisados métodos e técnicas utilizadas em Pós-processamento de conhecimento, medidas objetivas para avaliação de Regras de Associação e algoritmos que geram regras. Dessa perspectiva foram realizados experimentos para identificar o potencial das medidas a serem empregadas como filtros de Regras de Associação. Uma avaliação gráfica apoiou o estudo das medidas e a especificação da metodologia proposta. Aspecto inovador da metodologia proposta é a utilização do método de Pareto e a combinação de medidas para selecionar as Regras de Associação. Por fim foi implementado um ambiente para avaliação de Regras de Associação, denominado *ARInE*, viabilizando o uso da metodologia proposta.

Abstract

The large demand of methods for knowledge discovery and analysis in large databases has continuously increased the research in data mining area. Among the tasks associated to this area, one can find Association Rules. Several algorithms have been proposed for treating Association Rules. However, these algorithms give as results a huge amount of rules, making the knowledge post-processing phase very complex and challenging.

There are several measures that can be used in this evaluation phase, but there are also some limitations regarding to the absence of an intuitive method to rank and select rules. Moreover, it is not possible to find specific methodologies for selecting rules, considering more than one measure simultaneously.

This thesis has as objective the proposal, development and implementation of a post-processing methodology for Association Rules. In the proposed methodology, small groups of rules, which have been identified as potentially interesting, are presented to the expert for evaluation. In this sense, methods and techniques for knowledge post-processing, objective measures for rules evaluation, and Association Rules algorithms have been analyzed. From this point of view, several experiments have been realized for identifying the potential of such measures to be used to filter Association Rules. The study of measures and the specification of the proposed methodology have been supported by a graphical evaluation. The novel aspect of the proposed methodology consists on using the Pareto's method and combining measures for selecting Association Rules. Finally, an environment for evaluating Association Rules, named as *ARInE*, has been implemented according to the proposed methodology.

Sumário

Sumário	x
Lista de Figuras	xiii
Lista de Tabelas	xvi
Lista de Algoritmos	xvii
1 Introdução	1
1.1 Contextualização	2
1.2 Objetivos	5
1.3 Hipótese	5
1.4 Metodologia	6
1.5 Principais Contribuições	7
1.6 Organização do Trabalho	8
2 Mineração de Dados	11
2.1 Considerações Iniciais	11
2.2 O Processo de Mineração de Dados	13
2.2.1 Identificação do Problema	14
2.2.2 Pré-processamento	15
2.2.3 Extração de Padrões	19
2.2.4 Pós-processamento	22
2.2.5 Utilização do Conhecimento	24
2.3 Considerações Finais	24
3 A Tarefa Regras de Associação	25
3.1 Considerações Iniciais	25
3.2 Conceitos e Definições	26
3.2.1 Base de Dados para Mineração de Dados por Regras de Associação	26
3.2.2 Formalização da Tarefa de Regras de Associação	27
3.2.3 Cálculo das Medidas Suporte e Confiança	28

3.2.4	Suporte e Confiança – uma Análise Crítica	29
3.2.5	Geração de <i>Itemsets</i> Frequentes	31
3.2.6	Exemplo de Extração de Regras de Associação	33
3.3	Algoritmos para Regras de Associação	35
3.3.1	AIS e SetM	35
3.3.2	O Algoritmo Apriori	36
3.3.3	AprioriTid	41
3.3.4	Algoritmo Simples para Gerar Regras de Associação	42
3.4	Aspectos Complementares Sobre Regras de Associação	44
3.4.1	Regras de Associação com Taxonomia	44
3.4.2	Regras de Associação com Restrições Sobre Atributos	46
3.4.3	Regras de Associação em Atributos Categóricos e Numéricos	47
3.4.4	Regras de Associação em Padrões Sequenciais	48
3.4.5	Regras de Associação Periódicas	49
3.4.6	Regras de Associação Ponderadas	50
3.4.7	Regras de Associação Negativas	50
3.4.8	Regras de Associação de Proporções	50
3.5	Algumas Ferramentas para Geração de Regras de Associação	51
3.6	Considerações Finais	51
4	Avaliação de Regras de Associação: Técnicas e Ambientes	55
4.1	Motivação	55
4.2	Princípios de Interessabilidade de Regras	56
4.3	Técnicas para Avaliação de Regras	58
4.3.1	Avaliação por Consulta	58
4.3.2	Medidas de Avaliação de Conhecimento	58
4.3.3	Poda de Regras	59
4.3.4	Sumarização e Generalização	59
4.3.5	Agrupamento de Regras	60
4.4	Medidas para Avaliação de Regras de Associação	60
4.4.1	Tabela de Contingência e Frequências	60
4.4.2	Medidas Objetivas Derivadas da Tabela de Contingência	62
4.4.3	Exemplo de Cálculo das Medidas Objetivas	67
4.4.4	Medidas Subjetivas	72
4.5	Técnicas de Apoio à Análise de Medidas	74
4.5.1	Análise de Pareto	74
4.5.2	Análise Fatorial	76
4.6	Trabalhos Relacionados com Avaliação de Regras	79

4.6.1	Requisitos de um Ambiente para Exploração de Regras	79
4.6.2	Aplicativo AIAS	83
4.6.3	Ambiente DISCOVER	86
4.6.4	Ambiente RULEE	90
4.6.5	Módulo Computacional RULEE_GAR	95
4.7	Considerações Finais	97
5	Medidas Objetivas como Filtros	99
5.1	Considerações Iniciais	99
5.2	Hipótese de Comportamento Gráfico	100
5.3	Análise do Comportamento Gráfico das Medidas	102
5.3.1	Metodologia utilizada para a realização dos experimentos	102
5.3.2	Aplicativos e Algoritmos Utilizados	103
5.3.3	Bases de Dados Utilizadas	106
5.3.4	Extração de Regras	107
5.3.5	Análise dos Resultados	108
5.3.6	Análises Complementares Realizadas para Validar a Análise Manual	126
5.4	Síntese da Análise do Comportamento Gráfico	128
5.5	Restritividade dos Filtros	133
5.6	Considerações Finais	137
6	Uma Metodologia para Pós-processamento de Regras de Associação	139
6.1	Considerações Iniciais	139
6.2	Proposição da Metodologia	140
6.2.1	Definição do <i>focus-ruleset</i>	141
6.2.2	Análise e Seleção das Medidas Objetivas	142
6.2.3	Aplicação dos Filtros	143
6.2.4	Avaliação do Usuário Especialista	145
6.3	Representação Padronizada de Regras de Associação	145
6.3.1	Gramáticas e Sintaxes para Representar Regras de Associação . . .	147
6.3.2	Cálculo da Tabela de Contingência	152
6.3.3	Conversão de Formatos	155
6.3.4	Considerações Sobre a Sintaxe Padrão	164
6.4	Implementação da Metodologia Proposta – o Ambiente <i>ARInE</i>	165
6.4.1	Relação <i>ARInE</i> – RULEE	165
6.4.2	Descrição do Processo de Análise de Regras de Associação usando o <i>ARInE</i>	169
6.5	Considerações Finais	178

7	Conclusões	179
7.1	Principais Contribuições	180
7.2	Trabalhos Futuros	182
	Referências	199

Lista de Figuras

2.1	Etapas do processo de Mineração de Dados.	13
2.2	Tarefas de Mineração de Dados.	19
3.1	Espaço de busca para o conjunto de itens $A = \{a, b, c, d, e\}$	32
3.2	Separação de <i>itemsets</i> freqüentes e não-freqüentes no espaço de busca de $A = \{a, b, c, d, e\}$	33
3.3	<i>Itemsets</i> candidatos armazenados em uma árvore- <i>hash</i>	39
3.4	Exemplo de taxonomia	45
4.1	Medidas objetivas e subjetivas	61
4.2	Formatos característicos da distribuição de Pareto	75
4.3	Tela do aplicativo AIAS	85
4.4	Funcionalidades do Ambiente DISCOVER	86
4.5	Arquitetura geral do Ambiente RULEE	93
4.6	Processo para generalização de Regras de Associação no RULEE_GAR	96
5.1	Gráfico que apresenta relação entre medidas para um conjunto genérico de regras	100
5.2	Formatos característicos de funções das medidas objetivas	101
5.3	Metodologia utilizada nos experimentos	103
5.4	Medida Confiança	109
5.5	Medida Convicção	111
5.6	Medida Confiança Negativa	112
5.7	Medida Novidade	113
5.8	Medida Confiança Relativa	115
5.9	Medida Confiança Negativa Relativa	116
5.10	Medida Sensibilidade Relativa	118
5.11	Medida Especificidade Relativa	119
5.12	Medida Satisfação	120

5.13	Medida Sensibilidade	121
5.14	Medida Especificidade	123
5.15	Medida Suporte	124
5.16	Medida J_1	125
5.17	Descrição dos atributos considerados para execução do algoritmo de classificação	128
5.18	Árvore de decisão simplificada para os gráficos das medidas	129
5.19	Frequência dos formatos gráficos	130
5.20	Distribuição dos tipos de gráficos por base de dados	130
5.21	Distribuição dos tipos de gráficos por medida	131
6.1	Processo de identificação de regras interessantes	141
6.2	Gráficos de dispersão de valores de medidas para Regras de Associação	143
6.3	Gráficos de distribuição de frequência de valores de medidas para Regras de Associação	144
6.4	Gráficos para análise de Pareto	145
6.5	Metodologia dos formatos padrão para comunicação entre as funcionalidades do Ambiente DISCOVER	147
6.6	Produções da gramática para Regras de Associação	149
6.7	Regra de Associação gerada pelo algoritmo do MineSetTM	149
6.8	Regra de Associação da Figura 6.7 na sintaxe padrão	150
6.9	Produções da gramática para Regras de Associação no Formato Compacto	151
6.10	Regra de Associação da Figura 6.7 na sintaxe padrão compacta	152
6.11	Conversão das saídas dos algoritmos de Regras de Associação	155
6.12	Algumas regras geradas pelo <i>Apriori</i> utilizando o conjunto de dados <i>Weather Nominal</i>	158
6.13	Regras da Figura 6.12 convertidas para o formato padrão compacto de Regras de Associação	158
6.14	Algumas regras geradas pelo <i>WEKA</i> utilizando o conjunto de dados <i>Weather Nominal</i>	160
6.15	Regras da Figura 6.14 convertidas para o formato padrão compacto de Regras de Associação	161
6.16	Algumas regras geradas pelo MineSetTM utilizando o conjunto de dados <i>Weather Nominal</i>	162
6.17	Regras da Figura 6.16 convertidas para o formato padrão compacto de Regras de Associação	162
6.18	Algumas regras geradas pelo <i>Magnum Opus</i> utilizando o conjunto de dados <i>Weather Nominal</i>	163

6.19	Regras da Figura 6.18 convertidas para o formato padrão compacto de Regras de Associação	164
6.20	Relação entre <i>ARInE</i> e RULEE	166
6.21	Diagrama físico da base de dados utilizada pelos ambientes <i>ARInE</i> e RULEE	168
6.22	Página inicial do Ambiente <i>ARInE</i>	169
6.23	Seleção de projeto e base de regras	170
6.24	Projeto e base de regras selecionados	171
6.25	Construção de consulta para seleção de regras	172
6.26	Resultado da seleção de regras	173
6.27	Seleção de medidas – tela inicial	174
6.28	Seleção de medidas – após análise fatorial	174
6.29	Exemplo de resultado empregando análise de dispersão	175
6.30	Exemplo de resultado empregando análise de discretização	176
6.31	Exemplo de resultado empregando análise de Pareto	177
6.32	Exemplo de visualização de regras para um intervalo de análise	178

Lista de Tabelas

2.1	Conjunto de exemplos no formato atributo-valor	16
3.1	Tabela booleana de itens-transações	26
3.2	Tabela de itens-transações	27
3.3	Exemplo de base de dados binária	31
3.4	Regras de associação geradas a partir da base de dados da Tabela 3.3 . . .	31
3.5	Relação de itens comprados por transação do Exemplo 1	34
3.6	<i>Itemsets</i> freqüentes do Exemplo 1, página 33	34
3.7	Regras geradas no Exemplo 1	35
3.8	Conjunto <i>Resposta</i> contendo os <i>itemsets</i> freqüentes	41
3.9	Ferramentas para Regras de Associação	52
4.1	Problemática de Regras de Associação	56
4.2	Tabela de contingência	61
4.3	Freqüências obtidas a partir da tabela de contingência	62
4.4	Amostra da base de dados QVU	68
4.5	Tabela de contingência para a Regra 1	69
4.6	Freqüências obtidas a partir tabela de contingência para a Regra 1	69
4.7	Tabela de contingência para a Regra 2	70
4.8	Freqüências obtidas a partir tabela de contingência para a Regra 2	70
4.9	Síntese do exemplo de cálculo das medidas objetivas	71
4.10	Exemplo do uso de análise fatorial para identificar uma estrutura em um grupo de variáveis	78
5.1	Características das bases de dados artificiais	107
5.2	Parâmetros utilizados nos algoritmos para Regras de Associação	108
5.3	Número de regras geradas com a aplicação dos algoritmos de Regras de Associação	108
5.4	Distribuição do número de gráficos segundo os clusters	127

5.5	Resumo da categorização dos gráficos realizada na Seção 5.3.5	132
5.6	Resumo da categorização dos gráficos utilizando a técnica de cluster	133
5.7	Número de regras por percentual de elevação no valor mínimo da medida .	134
5.8	Número de regras selecionadas para 5% de elevação	135
5.9	Número de regras selecionadas, combinando as quatro medidas	135
5.10	Regras selecionadas para avaliação do especialista do domínio	136
6.1	Base de dados <i>Weather Nominal</i>	156

Lista de Algoritmos

1	Apriori.	37
2	Função <i>apriori-gen</i>	38
3	Algoritmo AprioriTid	42
4	Gera Regras de Associação.	43

Introdução

A popularização da Internet, maior e principal rede de computadores do planeta e os avanços tecnológicos em Banco de Dados, sistemas de comunicação, mecanismos de coleta e armazenamento de dados, têm proporcionado às organizações a capacidade de armazenar informações detalhadas sobre cada transação efetuada, gerando grandes bases de dados que, em alguns casos, chegam a dezenas de gigabytes ou até dezenas de terabytes de informações coletadas por ano (Piatetsky-Shapiro 1999; Padmanabhan & Tuzhilin 1999; Barroso et al. 2003).

Paralelamente a esta evolução tecnológica, as organizações passaram a reconhecer o valor das informações e do conhecimento contidos em suas bases de dados. A crescente demanda por informações precisas, detalhadas e atualizadas, necessárias ao gerenciamento das organizações, levou à busca por meios mais rápidos e eficientes de recuperação da informação e do conhecimento, incentivando a realização de pesquisas voltadas para o processo de transformação de dados em conhecimento.

Frente às demandas apresentadas, métodos predominantemente manuais utilizados para a transformação de dados e informações em conhecimento, tornam o processo de extração de conhecimento dispendioso (em termos financeiros e de tempo), subjetivo e, geralmente, inviável quando aplicado a grandes bases de dados. Nesse sentido, buscando automatizar o processo de análise, várias pesquisas têm sido desenvolvidas na área de

Mineração de Dados (Fayyad et al. 1996b; Mannila 1996; Piatetsky-Shapiro 1999; Netz et al. 2000).

1.1 Contextualização

O processo de extração de conhecimento de grandes bases de dados não pode ser entendido como um sistema de análise automática, mas sim, como um processo iterativo e iterativo, centrado na interação entre engenheiros de conhecimento, especialistas do domínio e usuários. Como descreve Mannila (1996), não se pode esperar a obtenção de conhecimento útil simplesmente submetendo um conjunto de dados a uma “caixa preta”.

Cabe destacar que sendo a Mineração de Dados um processo iterativo, suas etapas – Compreensão do Domínio, Pré-processamento de Dados, Extração de Padrões, Pós-processamento do Conhecimento e Utilização do Conhecimento – são interconectadas de tal modo que alterações, mesmo sutis, em uma determinada etapa podem afetar significativamente quaisquer das etapas seguintes ou ainda determinar o reinício de todo o processo (Fayyad et al. 1996; Rezende et al. 2003).

As tarefas de Mineração de Dados utilizadas para identificação de padrões podem ser agrupadas nas categorias: preditivas e descritivas. As tarefas preditivas constroem uma hipótese, a partir da generalização de exemplos com classes previamente definidas, na tentativa de prever um tipo de comportamento para novos casos (atributo meta). Os principais tipos de problemas de predição são a Classificação e a Regressão. As tarefas descritivas consistem na identificação de comportamentos intrínsecos do conjunto de dados. Das tarefas descritivas destacam-se *Clustering*, Regras de Associação e Sumarização.

Dentre as tarefas de Mineração de Dados, Regras de Associação é uma das técnicas que tem despertado grande interesse (Baesens et al. 2000). Na área acadêmica, pesquisas vêm sendo desenvolvidas e organizações têm utilizado seus resultados em aplicações relacionadas a comércio, contratos de seguro, seguridade social, geoprocessamento, serviços bancários e outras áreas (Liu et al. 2000; Clementini et al. 2000; Semenova et al. 2001; Aggelis 2004). Exemplo do emprego de Regras de Associação na área comercial pode ser observado em páginas na Internet, que apresentam sugestões do tipo “... quem comprou o produto que você procura **também** comprou estes outros produtos ...”.

Em termos gerais, a tarefa Regras de Associação permite identificar padrões intra-transações em uma base de dados. Especificamente, uma Regra de Associação caracteriza

o quanto a presença de um conjunto de atributos nos registros de uma base de dados implica na presença de algum outro conjunto distinto de atributos nos mesmos registros (Agrawal & Srikant 1994). Para identificação de Regras de Associação diversos algoritmos foram propostos, dentre os quais pode-se citar: *AIS* (Agrawal et al. 1993), *Apriori* e *AprioriTid* (Agrawal & Srikant 1994), *Opus* (Webb 1995), *Closet* (Pei et al. 2000) e *FP-Growth* (Han et al. 2000; Wang et al. 2002).

As regras geradas pela tarefa Regras de Associação não possuem como característica o atributo meta, podendo qualquer atributo estar presente na regra. Outro aspecto a ser considerado é que as Regras de Associação são empregadas de maneira individual e não de maneira conjunta como no caso das Regras de Classificação – cada regra expressa distintas regularidades existentes nos dados e, geralmente, retratam situações diferentes umas das outras (Weiss & Indurkha 1998). Essa condição determina que a avaliação e interpretação do conhecimento extraído por meio do processo de Mineração de Dados seja tratada de modo diferenciado.

No cenário presente com tendências como marketing individualizado, gerenciamento de relações com clientes (CRM), conteúdo de Web personalizado, dentre outras, vem ocorrendo a disseminação, intensificação e popularização do uso de técnicas de Extração de Conhecimento na resolução de problemas concretos. Essa demanda tem aumentado a exigência de confiabilidade no conhecimento extraído e reforça a necessidade de maior ênfase na análise e avaliação do conhecimento adquirido.

Considerando-se que os usuários de Sistemas Inteligentes normalmente têm conhecimento prévio sobre o domínio para o qual a aplicação está sendo desenvolvida, o Pós-processamento irá contribuir para ratificar/retificar o conhecimento existente.

Segundo Bruha & Famili (2000), a realização de pesquisas em Pós-processamento do conhecimento (principalmente em tópicos como avaliação da qualidade, grau de interesse, compreensibilidade e visualização do conhecimento extraído) tem obtido grande destaque e vem ao encontro das necessidades atuais à medida que busca desenvolver meios para identificar o conhecimento interessante e incorporar o conhecimento prévio do domínio.

Apesar das vantagens da utilização do processo de Mineração de Dados, sua aplicação pode levar, em alguns casos, à descoberta de conhecimento muito óbvio ou irrelevante que pouco acrescenta ao conhecimento já existente sobre o domínio (Padmanabhan & Tuzhilin 1999).

No caso particular de Regras de Associação, a possibilidade de descoberta de todas

as associações existentes nas transações de uma base de dados é o aspecto de maior relevância da tarefa. Entretanto, esta característica determina a geração, usualmente, de um número surpreendente de regras, dificultando sobremaneira a interpretação do conjunto de regras pelo usuário. Para superar esta problemática, técnicas de Pós-processamento do conhecimento devem ser aplicadas.

Em Regras de Associação, a maioria das abordagens para a avaliação do conhecimento extraído consiste na utilização de uma ou mais métricas para ordenar regras (ou realizar algum tipo de poda) e, assim, apresentá-las para a avaliação do usuário final. Hilderman & Hamilton (2000) propõem metodologia para seleção de regras em duas etapas. Na primeira etapa a medida qui-quadrado é utilizada para estabelecer o grau de significância para as regras, que são ordenadas por medidas objetivas na segunda etapa. Bruzzese & Davino (2002) propõem um *framework* que combina técnicas estatísticas e gráficas para a poda de conjuntos de regras. Jorge et al. (2002) propõem metodologia e ferramenta para navegação e visualização de regras baseada em um conjunto de operadores que permitem focar características particulares de um conjunto de regras considerando uma variedade de aspectos. Blanchard et al. (2003) utilizam recursos de realidade virtual e medidas objetivas para Pós-processamento visual de Regras de Associação. As regras são representadas visualmente como árvores com diferentes valores de tamanho, cor e distância do observador associados às medidas objetivas.

Lacunas podem ser identificadas nos trabalhos de Pós-processamento de Regras de Associação. A primeira delas refere-se à inexistência de um método intuitivo para priorizar e selecionar regras estabelecendo intervalos de valores de medidas para sua análise. Outra lacuna está na ausência de metodologias específicas para seleção de Regras de Associação considerando mais de uma medida simultaneamente.

Trabalhos vêm sendo desenvolvidos no Laboratório de Inteligência Computacional (LABIC)¹ do Instituto de Ciências Matemáticas e de Computação (ICMC) da USP São Carlos, abordando temas relacionados ao processo de Mineração de Dados e Textos. Está em desenvolvimento o ambiente computacional DISCOVER para a extração de conhecimento e o ambiente RULEE para exploração de regras.

O objetivo no desenvolvimento do Ambiente DISCOVER é disponibilizar, aos pesquisadores, um ambiente integrado de suporte às etapas do processo de aquisição de conhecimento com funcionalidades para o Aprendizado de Máquina e a Mineração de Dados e Textos. A compatibilidade entre os diferentes módulos do ambiente é baseada em sintaxes padrões utilizadas para representar dados e conhecimento.

¹<http://labic.icmc.usp.br>.

O Ambiente RULEE oferece mecanismos para exploração de regras de Classificação, Regressão e Associação, utilizando medidas objetivas e subjetivas. Por tratar-se de ambiente genérico, não possui os mecanismos específicos e o desempenho necessário para auxiliar nos processos de filtragem e seleção de Regras de Associação.

Considerando que na área de Mineração de Dados grande parte das pesquisas realizadas focalizam as etapas de Pré-processamento e Extração de Padrões (especialmente à tarefa Classificação), é importante desenvolver trabalhos centrados na etapa de Pós-processamento de conhecimento. No Ambiente DISCOVER, a situação é semelhante. Grande parte dos trabalhos desenvolvidos estão associados a estas etapas, existindo a necessidade de trabalhos que abordem outras questões relacionadas a estas áreas.

1.2 *Objetivos*

No contexto apresentado, este trabalho teve como objetivo investigar o processo de Mineração de Dados utilizando a tarefa Regras de Associação, com enfoque em aspectos relacionados ao Pós-processamento do conhecimento adquirido. Dessa perspectiva, busca-se solucionar a problemática da análise do grande número de Regras de Associação geradas, uma vez que, como destacam Silberschatz & Tuzhilin (1995) e Pugliesi (2004), é contraproducente o fornecimento de grande quantidade de padrões ao usuário.

O objetivo principal desta tese é propor uma metodologia para o Pós-processamento de Regras de Associação. Nessa metodologia, pequenos grupos de regras potencialmente interessantes são apresentados ao usuário para sua avaliação. A definição destes grupos de regras deve ser baseada no uso de medidas objetivas.

1.3 *Hipótese*

A hipótese adotada neste trabalho é: em um conjunto de Regras de Associação existem poucas regras que apresentam valores elevados em diversas medidas objetivas, a maioria das regras apresenta valores elevados em apenas uma medida. Essa condição permite obter um conjunto de regras reduzido para ser analisado por um especialista.

1.4 Metodologia

O problema foi abordado sistematicamente, de maneira iterativa, de acordo com o processo de Mineração de Dados, consistindo especialmente nas fases de Pré-processamento, Extração de Padrões e Pós-processamento.

Para tanto, foi necessário analisar métodos e técnicas utilizadas em Pós-processamento de conhecimento e propor uma metodologia para explorar Regras de Associação, por meio de medidas objetivas.

O ponto inicial da pesquisa consistiu no levantamento dos algoritmos que geram Regras de Associação. A intenção foi analisar o conteúdo do arquivo de saída dos algoritmos para identificar informações fornecidas com as regras e a forma pela qual o algoritmo representa a regra, no sentido de definir um formato padronizado de representação. Paralelamente, procedeu-se a identificação das medidas objetivas empregadas na avaliação de regras visando conhecer as características das medidas e definir como poderiam ser calculadas de maneira padronizada. Todas as medidas consideradas podem ser determinadas a partir da tabela de contingência.

À medida que estes estudos foram concluídos, foi possível especificar uma sintaxe padrão para representar as Regras de Associação. Nessa sintaxe, além dos elementos constituintes da regra, é representada a tabela de contingência associada à regra. A intenção de estabelecer uma sintaxe padrão é facilitar a manipulação dos conjuntos de regras nas etapas posteriores de execução do projeto e evitar a necessidade de re-implementação de procedimentos de acesso aos conjuntos de regras e para cálculo das medidas identificadas. Também foi necessário definir procedimento para calcular a tabela de contingência a partir dos resultados dos algoritmos que geram Regras de Associação.

Especificada a sintaxe padrão para Regras de Associação e os procedimentos para cálculo da tabela de contingência, foi implementada biblioteca em *PERL* para converter arquivos de regras para o formato especificado na sintaxe padrão. Atualmente está implementada a conversão a partir do algoritmo *Apriori* (versão desenvolvida por Borgelt (2003)) e dos aplicativos *MineSetTM*, *MagnumOpus* e *WEKA*. Vale ressaltar que a inclusão de novos algoritmos e aplicativos na biblioteca de conversão é tarefa de fácil realização.

À medida que se sucederam as etapas do trabalho e considerando os objetivos estabelecidos, foi verificada a necessidade de analisar o potencial das medidas identificadas inicialmente, para emprego como filtros para regras. Assim, foi realizado conjunto de

experimentos com esse objetivo. Nesses experimentos foi avaliado o potencial de utilização de cada medida utilizando-se como referência o princípio de Pareto. Esse princípio estabelece, em linhas gerais, que grande parte das conseqüências de um fenômeno provêm de um pequeno número de causas. No presente contexto, a medida é o fenômeno, os valores da medida são as conseqüências e as regras são as causas. Desse modo, para que uma medida seja adequada como filtro, um número reduzido de regras deve possuir valores elevados para a medida. O resultado das análises realizadas nos experimentos foi sintetizado e, com o apoio de técnicas de *Clustering* e Classificação, análises complementares foram realizadas no intuito de identificar padrões gerais de comportamento para as medidas. Como resultado geral destes experimentos, constatou-se que algumas medidas têm comportamento semelhante e, principalmente, que o desempenho das medidas varia em relação às bases de dados e aos parâmetros empregados nos algoritmos de extração de regras. Essa conclusão demanda a criação de um método para agrupar medidas semelhantes. Possível abordagem para agrupar as medidas é o uso da técnica estatística de análise fatorial, que permite agrupar atributos para um conjunto de exemplos – no caso, as medidas são os atributos e os exemplos são os valores das medidas para cada regra.

A partir dos resultados e conclusões obtidos, foi possível especificar uma metodologia de Pós-processamento de Regras de Associação, fundamentada na hipótese inicial da tese, e que atende aos objetivos aqui estabelecidos e também às necessidades identificadas durante a realização dos experimentos. A metodologia proposta é disponibilizada com a implementação de um ambiente para avaliação de Regras de Associação – *ARInE* (*Environment for Association Rules Interactive Exploration*), implementado utilizando a linguagem de programação *PERL* e interface Web, sendo a análise de regras realizada com o uso de um navegador.

1.5 Principais Contribuições

Uma das contribuições desta tese é a proposta e implementação de método para calcular os valores da tabela de contingência a partir de valores de medidas presentes nos resultados de cada algoritmo de extração de regras, o que agiliza sobremaneira o cálculo destes valores.

Além disso, foram catalogadas e disponibilizadas diversas medidas objetivas, bem como gerada uma caracterização dessas medidas em relação à capacidade para filtrar regras e à manutenção desta capacidade para diferentes bases de dados e parâmetros para algoritmos de Mineração de Dados.

Destaca-se ainda, como principal contribuição desta tese, a metodologia de Pós-processamento proposta para avaliação exploratória e interativa de conhecimento representado por Regras de Associação. Essa metodologia permite ao usuário “navegar” em um conjunto de regras utilizando técnicas baseadas em medidas objetivas para obter pequenos grupos de regras interessantes. Tais contribuições foram disponibilizadas no Ambiente *ARInE*, proposto, desenvolvido e implementado durante este trabalho.

1.6 Organização do Trabalho

Esta tese está dividida em 7 capítulos. Neste capítulo foi contextualizada a problemática abordada na tese, além de apresentados os objetivos, a metodologia utilizada e as principais contribuições deste trabalho.

No Capítulo 2 é descrito o processo de Mineração de Dados com o detalhamento das principais etapas do referido processo.

No Capítulo 3 é apresentada a tarefa de Regras de Associação incluindo as definições, formalizações e os principais algoritmos. Também são descritas variantes da tarefa e aplicativos e implementações para extração de regras.

A questão da avaliação do conhecimento representado por Regras de Associação é abordada no Capítulo 4. São apresentadas as principais técnicas utilizadas nas referências consultadas. Entre elas destacam-se as técnicas que empregam medidas na avaliação de regras, abordagem adotada neste trabalho. Nesse sentido, são descritas diversas medidas (objetivas e subjetivas) utilizadas para avaliar regras e duas técnicas de apoio à análise com medidas objetivas. Adicionalmente, são descritos alguns trabalhos relacionados ao Pós-processamento de Regras de Associação.

No Capítulo 5 são apresentados os experimentos realizados para verificação da potencialidade do uso de medidas objetivas como “filtros” para Regras de Associação. Foram realizados experimentos preliminares para verificação do comportamento gráfico de algumas das medidas descritas no Capítulo 4, a partir dos quais foi elaborada a hipótese de comportamento para as medidas. Foi realizado conjunto de experimentos utilizando 12 bases de dados e 13 medidas objetivas, a fim de verificar a hipótese elaborada e analisar a estabilidade do comportamento das medidas em relação a diferentes bases de dados e diferentes parâmetros para extração de regras. Os resultados obtidos foram analisados e interpretados utilizando-se três técnicas distintas. O capítulo inclui a síntese dessa inves-

tigação e, adicionalmente, estudo experimental sobre a capacidade restritiva de algumas das medidas avaliadas com filtros para regras.

No Capítulo 6 é proposta uma metodologia de Pós-processamento de Regras de Associação desenvolvida a partir da hipótese e dos objetivos desta tese, assim como a partir dos resultados experimentais obtidos no Capítulo 5. Essa metodologia utiliza medidas objetivas para avaliar a qualidade das regras, a análise de Pareto como suporte ao usuário na definição de valores de “corte” das medidas e, todo o processo de análise é auxiliado por gráficos das referidas medidas. É ainda apresentado o Ambiente *ARInE* que implementa a metodologia proposta, com a apresentação de suas funcionalidades e sua implementação do Ambiente, bem como descrição do processo de análise de Regras de Associação no *ARInE*. Além disso, são apresentadas as sintaxes padrão definidas e utilizadas no *ARInE* que permitem sua integração com os ambientes DISCOVER e RULEE e destes com diferentes algoritmos para extração de Regras de Associação.

As conclusões e principais contribuições desta tese, assim como as propostas de trabalhos futuros, são apresentados no Capítulo 7. Por fim, são apresentadas as bibliografias referenciadas nesta tese.

Mineração de Dados

2.1 *Considerações Iniciais*

A constante evolução da computação, relacionada especialmente às tecnologias de coleta, armazenamento e transmissão de dados, ao aumento da velocidade de comunicação e à redução dos custos associados a estas tecnologias, tem proporcionado às organizações a capacidade de armazenar e processar suas operações de maneira detalhada. Isso pode gerar, em pouco tempo, *gigabytes*, *terabytes* ou até *pentabytes* de dados, o que excede em muito a capacidade humana de fazer análises e identificar informações proveitosas.

Tradicionalmente a transformação de dados em informações úteis – em conhecimento – tem ocorrido por meio de análises e interpretações realizadas a partir de procedimentos fundamentalmente manuais, apesar de muitas vezes auxiliados por aplicativos, tais como planilhas eletrônicas e gerenciadores de Banco de Dados. Em geral os dados são analisados por um ou mais especialistas familiarizados com a natureza dos dados, que comparam gráficos e relatórios, cruzam informações, utilizando-se do conhecimento prévio que possuem sobre o domínio da aplicação. Assim, os especialistas atuam como uma espécie de interface entre os dados e os possíveis usuários das informações (Monard et al. 1998). Esse

procedimento usualmente é lento, subjetivo e custoso, tornando-se impraticável quando se trabalha com grande volume de dados.

Diante das dificuldades para analisar grandes volumes de dados e do reconhecimento por parte das organizações do potencial das vantagens competitivas que podem ser obtidos a partir de suas bases de dados, estudos têm sido direcionados ao desenvolvimento de tecnologias de extração automática de conhecimento de bases de dados. Esse campo de pesquisa é chamado Extração de Conhecimento de Base de Dados, geralmente referenciado na literatura *Knowledge Discovery in Databases* (KDD) ou Mineração de Dados (MD). Alguns autores consideram os termos KDD e MD referentes a processos distintos (Fayyad et al. 1996c). Neste trabalho, os termos KDD e MD serão empregados indistintamente como referência ao processo de extração de conhecimento a partir de dados. Essa abordagem é adotada em (Rezende et al. 2003) e justifica-se pelo agrupamento de algumas etapas e pela dimensão das bases de dados usualmente utilizadas.

A definição do processo KDD aceita por diversos pesquisadores em Mineração de Dados foi elaborada por Fayyad et al. (1996a): “Extração de Conhecimento de bases de dados é o processo de identificação de padrões válidos, novos, potencialmente úteis e compreensíveis embutidos nos dados”. Para melhor compreensão do conteúdo dessa definição, deve-se considerar individualmente cada componente:

Dados – Conjunto de fatos ou casos em um repositório de dados. Por exemplo: dados correspondem aos valores dos campos de um registro de vendas em uma base de dados qualquer;

Padrões – Denota alguma abstração de um subconjunto dos dados em alguma linguagem descritiva de conceitos;

Processo – A Extração de Conhecimento de bases de dados envolve diferentes etapas – a preparação dos dados, a busca por padrões, a avaliação do conhecimento;

Válidos – Os padrões descobertos devem satisfazer funções ou limiares que garantam que os exemplos cobertos e os casos relacionados ao padrão encontrado sejam aceitáveis;

Novos – Padrões identificados devem fornecer novas informações sobre os dados. O grau de novidade é um indicador da originalidade desses padrões;

Úteis – Os padrões descobertos devem ser incorporados e utilizados;

Compreensíveis – Um dos objetivos da realização do processo de Mineração de Dados é identificar padrões descritos em alguma linguagem que possa ser compreendida pelos usuários permitindo uma análise mais aprofundada da base de dados;

Conhecimento – O conhecimento é dependente do domínio da aplicação e está relacionado com a capacidade de inovação de um conceito, sua utilidade e inovação para o usuário.

2.2 O Processo de Mineração de Dados

Diferentes abordagens para a divisão do processo de Mineração de Dados em etapas são encontradas na literatura. Fayyad et al. (1996b) propõem a divisão do processo em nove etapas. Weiss & Indurkha (1998) dividem o processo em quatro etapas. Neste trabalho é considerada a divisão do processo proposta por Rezende et al. (2003), que agrupa o processo em um ciclo com três grandes etapas: Pré-processamento, Extração de Padrões e Pós-processamento. Adicionalmente, os autores propõem a inclusão de uma fase anterior ao processo de Mineração de Dados, que se refere à identificação do problema a ser abordado, e uma fase posterior ao processo, que se refere à utilização do conhecimento obtido. Na Figura 2.1 essas etapas são relacionadas esquematicamente.

Figura 2.1: Etapas do processo de Mineração de Dados. Fonte: (Rezende et al. 2003).

O processo de Mineração de Dados é baseado na interação entre as diferentes categorias de usuários e o sucesso de sua aplicação depende, em grande parte, da qualidade dessa interação. Os usuários do processo podem ser agrupados em classes, a saber:

Especialista do domínio – Usuário que possui conhecimento amplo do domínio da aplicação e oferece suporte ao analista na execução do processo.

Analista – Usuário especialista no processo de Extração de Conhecimento e responsável pela sua execução; conhece com profundidade cada etapa constituinte do processo.

Usuário final – Representa o grupo de usuários que utiliza o conhecimento extraído no processo de Mineração de Dados como instrumento de apoio nos processos de tomada de decisão. Não devem possuir necessariamente conhecimento aprofundado do domínio da aplicação.

Nesta classificação cabe mencionar as situações em que um indivíduo é agente em mais de uma classe, como no caso do especialista do domínio que também é usuário final e/ou auxilia e executa funções do analista. A descoberta de conhecimento útil pelo analista é pouco provável sem o suporte do especialista que pode identificar o conhecimento considerado interessante em um domínio particular.

2.2.1 *Identificação do Problema*

O estudo do domínio da aplicação e a definição de objetivos e metas a serem alcançados no processo de Mineração de Dados são identificados nessa fase. Também é realizada a identificação e seleção dos conjuntos de dados a serem utilizados durante o processo.

A participação de especialistas do domínio da aplicação fornecendo conhecimento prévio sobre o domínio e dando suporte aos analistas na tarefa de identificação de padrões é aspecto determinante no sucesso do processo de Extração de Conhecimento. É imprescindível adquirir conhecimentos iniciais sobre o domínio da aplicação, antes de se proceder a execução das tarefas de Mineração de Dados (Fayyad et al. 1996c). Nesta fase devem ser definidas as metas do processo e identificados os critérios de desempenho, o tipo de conhecimento a ser extraído (tipo caixa-preta ou simbólico) e a relação entre simplicidade e precisão do conhecimento extraído.

O conhecimento sobre o domínio subsidia tomadas de decisão nas etapas subsequentes do processo de Mineração de Dados: Pré-processamento, Extração de Padrões e Pós-

processamento. Na etapa de Pré-processamento, este conhecimento contribui para a escolha do conjunto de dados mais adequado à extração de padrões; para a identificação dos valores válidos para os atributos; para a definição de critérios de preferência entre os possíveis atributos; para o estabelecimento de restrições de relacionamentos e para a geração de novos atributos. Na etapa de Extração de Padrões, o conhecimento sobre o domínio contribui para a seleção de um critério de preferência entre os modelos gerados e ainda no ajuste dos parâmetros do processo de mineração. Também pode auxiliar a produção de um conhecimento inicial a ser utilizado como entrada do algoritmo de extração visando aumentar a precisão ou melhorar a compreensibilidade do modelo final. Na etapa de Pós-processamento, esse conhecimento pode contribuir para a avaliação dos padrões extraídos, determinando se os mesmos são interessantes para o usuário (Hilderman & Hamilton 2001; Liu & Hsu 1996; Piatetsky-Shapiro & Matheus 1994).

Entender o domínio dos dados é naturalmente um pré-requisito para se extrair conhecimento útil – o usuário final do sistema deve ter algum grau de entendimento sobre a área de aplicação para que quaisquer informações significantes sejam obtidas. Por outro lado, se existem especialistas com profundo conhecimento sobre o domínio, obter novas informações com o uso de ferramentas semi-automáticas pode ser pouco provável, como no caso de domínios estáveis, no qual os seres humanos tiveram tempo para adquirir o conhecimento especializado em detalhes (exemplo: áreas de comércio em que os produtos e clientes permanecem os mesmos por um longo período).

2.2.2 Pré-processamento

Usualmente os dados selecionados para o processo de Mineração de Dados não estão em formato adequado para a Extração de Conhecimento. Problemas gerados no processo de coleta de dados, tais como erros de digitação ou de leitura de dados por sensores, podem originar dados incorretos ou inconsistentes que necessitam de tratamento. Limitações na capacidade de memória ou no tempo de processamento podem impossibilitar a aplicação direta dos algoritmos de Extração de Padrões a todo o conjunto de dados. Nesse sentido, torna-se necessária a aplicação de métodos para tratamento, limpeza, redução do volume de dados, antes de se iniciar a etapa de Extração de Padrões. É importante destacar que os objetivos do processo de extração definidos na fase de Identificação do Problema, devem sempre nortear a execução das operações de Pré-processamento, de tal maneira que o conjunto de dados resultante apresente as características necessárias para que os objetivos estabelecidos sejam atingidos (Batista 2003).

Na etapa de Pré-processamento podem ser executadas diversas operações no conjunto de dados, a saber:

Extração e integração – As fontes de dados disponíveis para Mineração de Dados podem ser encontradas nos mais diversos formatos – arquivos texto, arquivos no formato de planilhas, Banco de Dados, *Data Warehouse*, dentre outros. Nesse sentido, é necessária a obtenção desses dados e sua unificação, constituindo-se assim uma única fonte de dados no formato atributo-valor, como ilustrado na Tabela 2.1.

A Tabela 2.1 representa um conjunto de dados com n exemplos de m atributos. Uma linha t_i representa o i -ésima transação¹ ($i = 1, 2, \dots, n$) e uma coluna a_j representa o j -ésimo atributo ($j = 1, 2, \dots, m$). Um valor $v_{i,j}$ corresponde ao valor do atributo a_j do exemplo t_i . A coluna C representa um atributo especial, denominado classe ou atributo meta. Em Mineração de Dados preditivo, o valor desse atributo é o valor a ser predito pelo modelo encontrado para novos exemplos. Quando a função de Mineração de Dados é descritiva, como no caso de Regras de Associação ou *clustering*, o atributo meta não é definido.

Tabela 2.1: *Conjunto de exemplos no formato atributo-valor*

	a_1	a_2	...	a_m	C
t_1	$v_{1,1}$	$v_{1,2}$	...	$v_{1,m}$	c_1
t_2	$v_{2,1}$	$v_{2,2}$	...	$v_{2,m}$	c_2
t_3	$v_{3,1}$	$v_{3,2}$	...	$v_{3,m}$	c_3
...	...	...	...	...	...
t_n	$v_{n,1}$	$v_{n,2}$	...	$v_{n,m}$	c_n

Transformação – Após a extração e unificação os dados devem ser adequados para utilização em algoritmos de Extração de Padrões. Algumas transformações podem ser aplicadas aos dados, tais como: resumo – quando dados de transações são agrupados para constituir resumos diários, semanais, mensais; transformação de tipo – quando um atributo do tipo data é transformado em outro tipo para que o algoritmo de Extração de Padrões possa utilizá-lo de maneira mais adequada; normalização de atributos contínuos – quando os valores são colocados em intervalos definidos (por exemplo, entre 0 e 1).

Limpeza – A qualidade dos dados é um fator de relevância no processo de Extração de Conhecimento considerando-se que os resultados gerados usualmente são utilizados

¹Optou-se pelo uso do termo transação, ao invés de exemplo, pois no caso das Regras de Associação o conjunto de exemplos é composto por um conjunto de transações.

em um processo de tomada de decisão. É possível que a qualidade dos dados esteja comprometida em função de erros (de digitação ou na leitura dos dados por sensores, por exemplo) gerados no processo de coleta. Nesse sentido, pode ser necessária a aplicação de técnicas de limpeza no conjunto de dados a fim de garantir a qualidade dos mesmos.

A limpeza pode ser realizada utilizando o conhecimento do domínio através da identificação de registros com valor inválido de algum atributo, de granularidade incorreta ou de exemplos errôneos, por exemplo. A decisão das estratégias de tratamento de atributos incompletos, de remoção de ruído e de tratamento de conjunto de exemplos não balanceados são maneiras de efetuar limpeza dos dados independentemente do domínio da aplicação (Batista et al. 2000).

Redução do volume de dados – Em alguns casos, devido a restrições no espaço de memória ou no tempo de processamento, pode ser necessária a aplicação de métodos para redução do volume de dados, para viabilizar a utilização de alguns algoritmos de Extração de Padrões.

Weiss & Indurkha (1998) sugerem três abordagens distintas na redução do volume de dados, a saber:

- **Redução do número de exemplos:** deve ser feita mantendo-se as características do conjunto de dados original, isto é, através da geração de amostras representativas do conjunto de dados (Glymour et al. 1997). A abordagem mais utilizada para redução do número de exemplos é a amostragem aleatória, considerando-se que tende a produzir amostras representativas (Weiss & Indurkha 1998).

Cabe destacar que se a amostra não for representativa, ou se a quantidade de exemplos for insuficiente para caracterizar os padrões embutidos nos dados, os modelos encontrados podem não ser representativos da realidade, não tendo portanto qualquer valor. Outro aspecto a ser considerado é que a utilização de uma quantidade relativamente pequena de exemplos pode dar origem ao fenômeno *overfitting* – o modelo gerado é representativo do conjunto de treinamento, mas não é adequado para a representação de novos exemplos (Fayyad et al. 1996b).

- **Redução do número de atributos:** pode ser uma maneira de redução do espaço de busca pela solução. Ocorre por meio da seleção de um subconjunto dos atributos existentes, de tal modo que não haja impacto na qualidade da solução final. A aplicação desse procedimento deve, preferencialmente, contar com o apoio do especialista do domínio, uma vez que a remoção de um atributo

potencialmente útil para o modelo pode diminuir consideravelmente a qualidade do conhecimento extraído. A seleção de atributos extras deve ser considerada porque não são conhecidos, a priori, os atributos importantes para atingir os objetivos estabelecidos para o processo de Mineração de Dados.

Uma maneira alternativa de redução do número de atributos é a indução construtiva, na qual um novo atributo é criado a partir do valor de outros, descartando-se os atributos originais. A indução construtiva visa aumentar a qualidade do conhecimento extraído. Vale dizer que, em alguns casos, pode ocorrer aumento no número de atributos com o uso da indução construtiva (Lee 2000).

- **Redução do número de valores de um atributo:** os processos de **discretização** ou de **suavização** de valores de um atributo são geralmente aplicados nessa tarefa.

No processo de **discretização** de um atributo ocorre a substituição de um atributo contínuo (inteiro ou real) por um atributo discreto, por meio do agrupamento de seus valores. Em termos gerais, um algoritmo de discretização tem como entrada os valores de um atributo contínuo e produz como saída uma pequena lista de intervalos ordenados. Cada intervalo é representado na forma $[V_{inferior} : V_{superior}]$, sendo $V_{inferior}$ e $V_{superior}$, respectivamente, os limites inferior e superior do intervalo. Os métodos de discretização podem ser classificados em supervisionados ou não supervisionados, locais ou globais e parametrizados ou não parametrizados (Félix 1998).

No processo de **suavização** de valores, o número de valores distintos de um atributo é reduzido, sem que haja discretização do atributo. Durante este processo os valores são agrupados, de maneira que cada grupo de valores é substituído por um valor numérico representativo, podendo ser a média, a mediana ou mesmo valores de borda de cada grupo (Weiss & Indurkha 1998).

As operações de Pré-processamento descritas devem ser realizadas de maneira criteriosa e com o devido cuidado, para que as informações presentes nos dados originais continuem presentes nas amostras geradas e os modelos finais sejam representativos da realidade expressa nos dados originais.

2.2.3 Extração de Padrões

A etapa de Extração de Padrões é direcionada ao cumprimento dos objetivos definidos na fase de Identificação do Problema. Nessa etapa é realizada a escolha, a configuração e a execução de um ou mais algoritmos para extração de conhecimento.

Por tratar-se de processo iterativo, pode ser necessário que essa etapa seja executada repetidas vezes para ajuste do conjunto de parâmetros visando a obtenção de resultados mais adequados aos objetivos pré estabelecidos. Ajustes podem ser necessários, por exemplo, para a melhoria do grau de precisão ou do nível de compreensibilidade do conhecimento extraído.

A etapa Extração de Padrões inclui as atividades escolha da tarefa de Mineração de Dados e escolha do algoritmo a ser empregado, além da extração dos padrões propriamente dita (Fayyad et al. 1996b):

Escolha da Tarefa – A escolha da tarefa é realizada em função dos objetivos desejáveis para a solução a ser encontrada. As funções possíveis na Extração de Padrões podem ser agrupadas em atividades **preditivas** e **descritivas**, conforme ilustrado na Figura 2.2.

Figura 2.2: Tarefas de Mineração de Dados. Fonte: (Rezende et al. 2003).

Atividades relacionadas à predição, ou **Mineração de Dados preditiva**, consistem

na generalização de exemplos ou experiências passadas com respostas conhecidas em um modelo capaz de identificar a classe (atributo meta) de um novo exemplo. Os dois principais tipos de problemas de predição são Classificação e Regressão. Problemas de Classificação consistem na predição de um valor categórico – por exemplo, predizer se o cliente é do tipo bom ou mau pagador. Nos problemas de regressão, o atributo a ser predito consiste num valor contínuo, por exemplo, predizer o lucro ou a perda em um empréstimo financeiro (Weiss & Indurkha 1998).

Atividades relacionadas à descrição, ou **Mineração de Dados descritiva**, consistem na identificação de comportamentos intrínsecos do conjunto de dados, sendo que esses dados não possuem um atributo-meta especificado. Algumas das tarefas descritivas são *Clustering*, Regras de Associação e Sumarização. A tarefa Regras de Associação, foco deste trabalho, é descrita detalhadamente no Capítulo 3.

Escolha do Algoritmo – Uma vez definida a tarefa de Mineração de Dados a ser empregada, existe uma variedade de algoritmos para sua execução. A definição do algoritmo de extração e a configuração de seus parâmetros são atividades incluídas nesta etapa. A escolha do algoritmo é subordinada à linguagem de representação dos padrões a serem encontrados. Pode-se utilizar algoritmos indutores de árvores de decisão ou regra de produção, por exemplo, se o objetivo é realizar uma classificação. Dos tipos mais freqüentes de representação de padrões, destacam-se (Mitchell 1998): Árvores de Decisão, regras de produção, modelos lineares, modelos não lineares (Redes Neurais Artificiais), modelos baseados em exemplos (KNN – *K-Nearest Neighbor*, Raciocínio Baseado em Casos) e modelos de dependência probabilística (Redes Bayesianas).

A maioria dos algoritmos utilizados em Mineração de Dados podem ser vistos como composição de princípios e técnicas básicas (Fayyad et al. 1996b). A especificação de um algoritmo para Mineração de Dados para uma determinada tarefa inclui a definição dos componentes de algoritmos (Hand et al. 2001):

- a **tarefa** de Mineração de Dados que o algoritmo deve realizar – Classificação, regressão ou Regras de Associação, por exemplo. Diferentes tipos de algoritmos são necessários para diferentes tipos de tarefas.
- a **estrutura** (funcional) do padrão que se está ajustando aos dados (por exemplo, um modelo de regressão linear). A estrutura define os limites do que pode ser aproximado ou aprendido. Dentro desses limites, os dados levam a um padrão particular. Em Regras de Associação, a estrutura é definida por meio das regras obtidas;
- a **função** selecionada para determinar a qualidade dos modelos filtrados a

partir dos dados observados (por exemplo, erro quadrático). Essa função é crítica para o aprendizado. Em Regras de Associação, as principais funções utilizadas são o suporte e a confiança;

- o **método de busca** utilizado para busca dos padrões sobre os parâmetros e estruturas. Em Regras de Associação, um dos métodos mais utilizados é a busca em largura com poda;
- a **técnica de gerenciamento de dados** utilizada para armazenar, indexar e recuperar os dados durante o processo de extração de dados.

Kohavi et al. (1997) demonstram experimentalmente que não existe um único algoritmo adequado para todas as tarefas de Mineração de Dados. Sendo assim, para realizar a tarefa desejada podem ser utilizados uma variedade de algoritmos diferentes. Isto leva à obtenção de diferentes modelos que, na etapa de Pós-processamento, serão tratados para fornecer o conjunto de padrões mais adequado para o usuário final. Especificamente no caso de Extração de Padrões por Regras de Associação, Zeng et al. (2001) mostram experimentalmente a independência dos padrões extraídos em relação ao algoritmo de Regras de Associação utilizado.

Extração de Padrões – A etapa de Extração de Padrões propriamente dita no processo de Mineração de Dados consiste na aplicação dos algoritmos selecionados para realizar a extração dos padrões embutidos nos dados.

Cabe destacar que dependendo da função adotada (predição ou descrição) podem ser necessárias diversas execuções dos algoritmos de Extração de Padrões. Por exemplo, com o objetivo de se obter uma avaliação mais precisa da taxa de erro de um classificador, tem-se utilizado métodos de re-amostragem (*resampling*). No método *k-fold cross-validation* – o conjunto de exemplos é sub-dividido em k partições mutuamente exclusivas e, em cada iteração do método, $k-1$ partições são fornecidas ao algoritmo de aprendizado. A partição restante é utilizada para calcular a taxa de erro da iteração. A média das taxas de erro de cada uma das k iterações é usada como a taxa de erro final, estimada pelo método. Outro exemplo é o uso de técnicas de combinação de preditores que têm sido pesquisadas com o objetivo de construir um preditor mais preciso pela combinação de vários outros. O resultado dessa combinação é chamado de *ensemble*. A utilização de *ensembles* tem obtido melhores resultados que a utilização de um único preditor (Dietterich 2000; Breiman 2000).

No caso das Regras de Associação o algoritmo pode ser executado uma única vez sem que haja comprometimento da qualidade do conhecimento extraído, pois os diversos algoritmos de associação devem apresentar a mesma solução (Zeng et al. 2001). Adicionalmente, considerando-se que os parâmetros de entrada para avaliação

dos resultados parciais (durante a geração das regras) são geralmente o **suporte** e a **confiança**, alterações nesse conjunto de valores significa apenas incluir ou excluir novas regras do conjunto de regras extraídas.

A disponibilização do conjunto de padrões extraídos nesta etapa para o usuário ou a sua incorporação a um Sistema Inteligente é realizada tendo-se concluída a análise e/ou processamento dos padrões na etapa de Pós-processamento.

2.2.4 Pós-processamento

O Pós-processamento é uma etapa importante do processo de Mineração de Dados, na qual os padrões extraídos podem ser simplificados, avaliados, visualizados ou simplesmente documentados para o usuário final. Considerando-se que o conhecimento extraído possa ser empregado na resolução de problemas do mundo real, seja por meio de um Sistema Inteligente ou por um ser humano fornecendo apoio em um processo de tomada de decisão, devem ser verificadas, nesta etapa, questões do tipo: se o conhecimento extraído representa o conhecimento do especialista; a maneira que o conhecimento extraído difere do conhecimento do especialista; em que parte o conhecimento do especialista está correto.

A resposta a essas questões geralmente não é trivial, considerando-se que os algoritmos de extração de padrões podem gerar uma quantidade elevada de padrões, muitos dos quais podem não ser importantes, relevantes ou interessantes para o usuário. Sabe-se também que não é produtivo fornecer uma grande quantidade de padrões ao usuário que, normalmente, procura uma pequena lista de padrões interessantes. Sendo assim, é de grande importância o desenvolvimento de técnicas de suporte para que seja possível fornecer aos usuários exclusivamente os padrões mais interessantes (Silberschatz & Tuzhilin 1995).

Esta etapa inclui conjunto de métodos e procedimentos que podem ser agrupados nas seguintes categorias (Bruha & Famili 2000):

Filtragem do Conhecimento – Pode ser realizada por meio de mecanismos de pós-poda, para o caso de Árvores de Decisão, ou de truncagem, no caso de regras de decisão. É aplicável principalmente nos casos em que os algoritmos de indução produzem árvores de decisão com muitas folhas ou regras de decisão muito específicas, cobrindo poucos exemplos (*overfitting*). O emprego de restrição de atributos ou ordenação de regras por meio métricas (medidas de avaliação) é uma maneira alternativa de filtragem do conhecimento, especialmente útil para Regras de Associação.

Interpretação e Explicação – É usualmente aplicada quando o conhecimento obtido é utilizado por um usuário final ou num Sistema Inteligente. O conhecimento pode ser documentado, visualizado ou modificado de modo a torná-lo compreensível ao usuário. O conhecimento extraído pode ser comparado ao conhecimento preexistente para a verificação de conflitos ou de conformidade e pode ser sumarizado e/ou combinado com o conhecimento prévio do domínio.

Avaliação – Pode ser realizada por meio dos critérios precisão, compreensibilidade, complexidade computacional, grau de interesse, dentre outros. No Capítulo 4 são apresentadas medidas e critérios que podem ser empregados nesta avaliação.

Integração do Conhecimento – Os sistemas tradicionais de suporte à decisão são dependentes de uma única técnica, estratégia e modelo. Os novos e sofisticados sistemas possibilitam combinar ou refinar os resultados de vários modelos de tal modo que se pode obter maior precisão e melhor desempenho do sistema na sua totalidade. Uma das técnicas que pode ser empregada, como mencionado anteriormente, é o uso da combinação de classificadores (*ensembles*) (Breiman 2000).

No desenvolvimento de um ambiente para avaliação de conhecimento, Freitas (1998a) sugere que sejam considerados, inicialmente, aspectos objetivos de avaliação do conhecimento para a seleção de regras potencialmente interessantes ao usuário e, num segundo momento, aspectos subjetivos como um filtro final para identificação das regras de fato interessantes. Dentro dessa abordagem, um exemplo de ferramenta de Pós-processamento é o aplicativo **AIAS** desenvolvido por Liu et al. (2000), descrito na Seção 4.6.2, página 83.

A análise do conhecimento extraído poderá determinar se o processo de extração de padrões deve ser ou não repetido. Caso o conhecimento extraído não seja de interesse para o usuário ou não esteja de acordo como objetivos estabelecidos na fase de Identificação do Problema, pode ser necessária a realização de etapas específicas do processo ou de todo o processo, ajustando-se os parâmetros utilizados ou realizando-se melhorias na seleção de dados. A investigação do conhecimento extraído também pode determinar a iteração do processo de Extração de Conhecimento, uma vez que podem ser especificados novos objetivos a serem atingidos.

2.2.5 Utilização do Conhecimento

O objetivo maior do processo de Extração de Conhecimento é o uso do conhecimento obtido. Assim, após ser avaliado e validado na etapa de Pós-processamento, o conhecimento extraído é consolidado na fase de Utilização do Conhecimento.

Após a sua consolidação, o conhecimento pode ser utilizado para resolver eventuais conflitos entre o conhecimento pré-existente (fornecido pelo especialista) e o conhecimento obtido com o processo de Mineração de Dados (Fayyad et al. 1996b), incorporado a um Sistema Inteligente, ou ainda utilizado diretamente pelo usuário final para apoio a algum processo de tomada de decisão.

2.3 Considerações Finais

O processo de Mineração de Dados tem como objetivo automatizar a tarefa de extrair conhecimento útil a partir de grandes volumes de dados. Não se trata de um sistema de análise automática, mas de um processo de natureza iterativa e interativa, baseado na interação entre especialistas do domínio, usuários e responsáveis pela execução do processo. Nesse sentido, como descreve Mannila (1996), não é possível presumir que a mera submissão de um conjunto de dados a uma “caixa preta” proporcione a extração de algum conhecimento útil.

Neste capítulo, o processo de Mineração de Dados e as etapas que o constituem foram descritos em linhas gerais. A predição e a descrição são os dois principais tipos de atividade do processo de Mineração de Dados. Na atividade preditiva, os exemplos fornecidos possuem uma classe associada, e o objetivo é prever o valor da classe de novos exemplos não rotulados. Na atividade descritiva, o objetivo é identificar comportamentos intrínsecos existentes no conjunto de dados.

No próximo capítulo é abordada a tarefa de Regras de Associação, uma tarefa descritiva que pode ser empregada no processo de Mineração de Dados e que é foco deste trabalho. A tarefa é descrita em linhas gerais com a apresentação dos principais conceitos, notações e algoritmos. Também são apresentados aspectos complementares e uma síntese das ferramentas disponíveis para geração de Regras de Associação.

A Tarefa Regras de Associação

3.1 Considerações Iniciais

Na classificação usualmente empregada em Mineração de Dados, a tarefa Regras de Associação é classificada como atividade de **Mineração de Dados descritiva**, como apresentado na Figura 2.2, página 19.

Apresentada inicialmente por Agrawal et al. (1993), a tarefa Regras de Associação tem tido destaque nos últimos anos, seja em aplicações práticas ou na área acadêmica (Piatetsky-Shapiro 1999; Baesens et al. 2000; Srikant 2001). Dada sua aplicabilidade a problemas de negócio, e elevada compreensibilidade da tarefa mesmo por aqueles não especialistas em Mineração de Dados (Hipp et al. 2002).

A tarefa de Regras de Associação foi desenvolvida inicialmente a partir da análise dos itens presentes em compras de supermercados, com o objetivo de identificar relações do tipo “*um cliente que compra os produtos $X_1, X_2, \dots, X_n$ também irá comprar os produtos $Y_1, Y_2, \dots, Y_m$ com probabilidade $c\%$* ”. As Regras de Associação não estão restritas a análises de dependência no contexto de aplicações de varejo, podendo ser aplicadas a uma ampla gama de problemas relacionados a contratos de seguros, saúde, bio-informática, geoprocessamento (Liu et al. 2000; Clementini et al. 2000; Semenova et al. 2001).

Em termos gerais, uma Regra de Associação determina o quanto a presença de um conjunto de atributos $\mathbf{s}$ nos registros de uma base de dados implica na presença de algum outro conjunto distinto de atributos $\mathbf{u}$ nos mesmos registros (Agrawal & Srikant 1994). Conforme a definição original de Agrawal et al. (1993), uma Regra de Associação é representada como uma implicação na forma $LHS \Rightarrow RHS$, em que LHS e RHS são, respectivamente, o lado esquerdo (*Left Hand Side*) e o lado direito (*Right Hand Side*) da regra, definidos por conjuntos disjuntos de atributos.

3.2 Conceitos e Definições

Nesta Seção são apresentadas notações e definições baseadas nos trabalhos de Srikant & Agrawal (1997) e Adamo (2001), necessárias à compreensão do processo de Mineração de Dados por Regras de Associação, conforme sua definição inicial. Cabe lembrar que outras abordagens, como as apresentadas na Seção 3.4, modificam algumas definições de maneira a considerar aspectos não abrangidos pela definição inicial de Regras de Associação.

3.2.1 Base de Dados para Mineração de Dados por Regras de Associação

Diferentemente da base de dados apresentada na Tabela 2.1, página 16, a base utilizada pelos algoritmos de Regras de Associação é uma tabela booleana de itens-transações (Tabela 3.1).

Tabela 3.1: Tabela booleana de itens-transações

	a_1	a_2	$\cdots$	a_m
t_1	1	1	$\cdots$	1
t_2	0	1	$\cdots$	1
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
t_n	1	0	$\cdots$	1

A base de dados utilizada pelos algoritmos de Regras de Associação pode também ser uma tabela de itens-transações, como apresentado na Tabela 3.2. Essa tabela é uma simplificação da tabela booleana de itens-transações, na qual itens com valor 1 são mantidos e itens com valor 0 são removidos das transações da tabela de itens-transações.

Tabela 3.2: *Tabela de itens-transações*

t_1	a_1	a_2	$\cdots$	a_m
t_2		a_2	$\cdots$	a_m
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
t_n	a_1		$\cdots$	a_m

Haverá necessidade de Pré-processamento dos dados quando a base de dados estiver em formato distinto dos apresentados.

3.2.2 Formalização da Tarefa de Regras de Associação

As Regras de Associação são definidas por Agrawal & Srikant (1994):

Seja D uma base de dados composta por um conjunto de itens $A = \{a_1, \dots, a_m\}$ ordenados lexicograficamente e por um conjunto de transações $T = \{t_1, \dots, t_n\}$, na qual cada transação $t_i \in T$ é composta por um conjunto de itens tal que $t_i \subseteq A$.

A Regra de Associação é uma implicação na forma $LHS \Rightarrow RHS$, em que $LHS \subset A$, $RHS \subset A$ e $LHS \cap RHS = \emptyset$. A regra $LHS \Rightarrow RHS$ ocorre no conjunto de transações T com **confiança** $conf$ se em $conf\%$ das transações de T em que ocorre LHS ocorre também RHS . A regra $LHS \Rightarrow RHS$ tem **suporte** sup se em $sup\%$ das transações em T ocorre $LHS \cup RHS$.

Vale dizer que nem todas as regras descobertas a partir de um conjunto de transações são interessantes ou úteis. Para prevenir esta ocorrência restrições devem ser estabelecidas. As mais comumente empregadas são **suporte mínimo (sup-min)** e **confiança mínima (conf-min)**. Neste caso, são consideradas exclusivamente regras com valores de **suporte** e **confiança** superiores aos valores mínimos definidos. O emprego destas restrições leva à redução do número de regras geradas e também do esforço (tempo) de processamento. As medidas **suporte** e **confiança** são formalmente definidas na Seção 3.2.3.

Anteriormente à descrição dos procedimentos para geração de Regras de Associação, é necessário definir os termos *itemset* e *k-itemset*. Um *itemset* ¹ é definido como um

¹Em Adamo (2001) é empregado o termo CAS - *Canonical Attribute Sequences* (Sequência Canônica de Atributos) ao invés do termo *itemset*.

conjunto de itens ordenados lexicograficamente. Já um k -*itemset* é um conjunto de k itens ordenados lexicograficamente. Um k -*itemset* freqüente é definido como um conjunto de k itens ordenados lexicograficamente que possui valor de **suporte** maior que o **suporte** mínimo (**sup-min**) definido pelo usuário.

Nas etapas I e II relacionadas, são definidos os procedimentos para geração de Regras de Associação:

- I) Encontrar os k -*itemsets* com suporte maior ou igual que **suporte** mínimo especificado pelo usuário (**sup-min**). Os *itemsets* com *suporte* igual ou superior a **sup-min** são definidos *itemsets* freqüentes, os demais conjuntos são denominados *itemsets* não-freqüentes. Na Seção 3.2.5 é apresentado detalhadamente o procedimento para geração dos *itemsets*. Na Seção 3.3.2 é apresentado algoritmo para identificação do conjunto de k -*itemsets* freqüentes.
- II) Utilizar os k -*itemsets* freqüentes, com $k \geq 2$, para gerar as Regras de Associação. Para cada *itemset* freqüente $l \subseteq A$, encontrar os subconjuntos $\tilde{a}$ de itens de l , não vazios e diferentes de l . Para cada subconjunto $\tilde{a} \subseteq l$, gerar uma regra na forma $\tilde{a} \Rightarrow (l - \tilde{a})$ se a razão de $sup(l)$ por $sup(\tilde{a})$ é maior ou igual à **confiança** mínima especificada pelo usuário (**conf-min**). Com um conjunto de *itemsets* freqüentes $\{abcd\}$ e um subconjunto de *itemsets* freqüentes $\{ab\}$, por exemplo, pode-se gerar uma regra do tipo $ab \Rightarrow cd$, desde que $conf(ab \Rightarrow cd) \geq \mathbf{conf-min}$, na qual, $conf(ab \Rightarrow cd) = sup(abcd)/sup(ab)$. Na Seção 3.3.4, é apresentado um algoritmo para gerar regras a partir de *itemsets* freqüentes.

3.2.3 Cálculo das Medidas Suporte e Confiança

As medidas **suporte** e **confiança** são as mais empregadas em Regras de Associação, tanto na etapa de Pós-processamento do conhecimento adquirido, quanto na etapa de seleção dos subconjuntos de atributos durante o processo de geração de regras. Para facilitar a compreensão das medidas, estas são definidas nas Equações 3.1 a 3.3:

suporte – quantifica a incidência de um *itemset* X ou de uma da regra no conjunto de dados, ou seja, indica a freqüência com que X ou $LHS \cup RHS$ ocorrem no conjunto de dados. O **suporte** para um *itemset* X pode ser definido pela equação:

$$sup(X) = \frac{n(X)}{N} \times 100, \quad (3.1)$$

sendo $n(X)$ o número de transações nas quais X ocorre e N o número total de transações (exemplos) consideradas.

O **suporte** de uma regra $LHS \Rightarrow RHS$ é representado na equação relacionada:

$$sup(LHS \Rightarrow RHS) = sup(LHS \cup RHS) = \frac{n(LHS \cup RHS)}{N} \times 100, \quad (3.2)$$

sendo $n(LHS \cup RHS)$ o número de transações nas quais LHS e RHS ocorrem juntos e N o número total de transações (exemplos) consideradas.

Assim, o **suporte** é equivalente à frequência $f(LHS, RHS)$, ou seja, determina a frequência com que LHS e RHS ocorrem simultaneamente no conjunto de dados, como definido na Seção 4.4.1, página 60.

confiança – indica a frequência com que LHS e RHS ocorrem juntos em relação ao número total de transações em que LHS ocorre. Assim, a **confiança** de uma regra $LHS \Rightarrow RHS$ pode ser representada por:

$$conf(LHS \Rightarrow RHS) = \frac{sup(LHS \cup RHS)}{sup(LHS)} = \frac{n(LHS \cup RHS)}{n(LHS)} \times 100, \quad (3.3)$$

sendo $n(LHS)$ o número de transações nas quais LHS ocorre. A **confiança** é equivalente à frequência condicional $f(RHS|LHS)$, ou seja, representa a frequência de ocorrência de RHS , dado que LHS ocorre.

3.2.4 Suporte e Confiança – uma Análise Crítica

Tradicionalmente no estudo de Regras de Associação tem-se utilizado como medidas básicas para a avaliação das regras o **suporte** e a **confiança**. Contudo, o uso de apenas estas duas medidas não oferece recursos discriminatórios suficientes. A sua utilização tem sido objeto de diversos estudos críticos. Adamo (2001) apresenta um conjunto de observações a partir da síntese desses estudos.

Observação 1 – Considerando as Tabelas 3.3 e 3.4, pode-se observar que para valores de **sup-min** = 25% e **conf-min** = 50% as regras $a \Rightarrow b$ e $a \Rightarrow c$ são válidas, possuindo a última os maiores valores de **suporte** e **confiança**. Esta condição pode ser considerada um paradoxo dado que $\frac{f(b|a)}{f(b)} = 2$ e $\frac{f(c|a)}{f(c)} = 0.875$, significando que o atributo b é positivamente correlacionado com a , enquanto c é negativamente correlacionado com

a ; ou seja, a probabilidade de b ocorrer aumenta quando o atributo a é considerado (de 0,25 para 0,50). Por outro lado, a probabilidade de c ocorrer diminui quando a é considerado (de 0,875 para 0,750).

Da Tabela 3.3 pode-se observar que a predominância (em termos de **suporte** e **confiança**) da regra $a \Rightarrow c$ sobre $a \Rightarrow b$ deve-se à ocorrência freqüente do atributo c . Apenas com o uso do **suporte** e da **confiança** não é possível identificar a ocorrência desta situação, que pode conduzir à geração de regras espúrias, como neste caso.

Observação 2 – Considerando ainda os mesmos valores de **sup-min** e **conf-min**, para comparar as regras $a \Rightarrow b$ e $a \Rightarrow \neg b$. Na Tabela 3.4 observa-se que os valores de **suporte** e **confiança** são os mesmos e que $\neg b$ é negativamente correlacionado com a , o que torna a regra $a \Rightarrow \neg b$ espúria. Novamente os valores das medidas foram influenciados pela ocorrência freqüente de um atributo. Adicionalmente, não há meios de prevenir a geração de ambas as regras: ou a regra $a \Rightarrow b$ não é gerada, ou é gerada conjuntamente com a regra contraditória $a \Rightarrow \neg b$, o que não é aceitável.

Observação 3 – Considerando uma base de dados que possua vários atributos com elevado valor de **suporte**, como o caso do atributo **sexo** em uma base de dados censitários, ou atributos provenientes de processo de discretização ou ainda atributos negados (atributo $\neg b$, na Tabela 3.3). Considerando-se também que o conjunto de medidas **suporte/confiança** favorece a geração de regras cujo conseqüente ocorra com grande freqüência, pode-se esperar a geração de muitas regras com elevado valor de **suporte**. No caso particular de regras envolvendo a negação de atributos, é possível a geração de muitas regras espúrias.

Observação 4 – Os atributos a e d não são correlacionados, já que $\frac{f(d|a)}{f(d)} = 1$ (Tabela 3.4), significando que não existe relação estatística significativa entre os atributos. Contudo, a regra $a \Rightarrow d$ tem os mesmos valores de **suporte** e **confiança** que a regra $a \Rightarrow b$.

As observações anteriores confirmam a necessidade de considerar outras medidas para a avaliação de regras, especialmente no caso das Regras de Associação. Contudo, apesar das deficiências apresentadas, o uso do **suporte** é importante por representar a significância estatística da regra (Tan & Kumar 2000).

Tabela 3.3: Exemplo de base de dados binária. Fonte: (Adamo 2001)

	Atributos				
	a	b	c	$\neg b$	d
1	1	1	0	0	1
2	1	1	1	0	1
3	1	0	1	1	0
4	1	0	1	1	0
5	0	0	1	1	1
6	0	0	1	1	1
7	0	0	1	1	0
8	0	0	1	1	0

Tabela 3.4: Regras de associação geradas a partir da base de dados da Tabela 3.3. Fonte: (Adamo 2001)

$LHS \Rightarrow RHS$	$f(LHS)$	$f(RHS)$	$f(RHS LHS)$	$\frac{f(RHS LHS)}{f(RHS)}$	$sup(LHS \Rightarrow RHS)$	$conf(LHS \Rightarrow RHS)$
$a \Rightarrow b$	0,50	0,25	0,50	$2 > 1$	0,25	0,50
$a \Rightarrow c$	0,50	0,875	0,75	$0,857 < 1$	0,375	0,75
$a \Rightarrow \neg b$	0,50	0,75	0,50	$0,666 < 1$	0,25	0,50
$a \Rightarrow d$	0,50	0,50	0,50	$1 = 1$	0,25	0,50

3.2.5 Geração de Itemsets Frequentes

Considerando-se uma base de dados D no formato de uma tabela booleana itens-transações ou de uma tabela itens-transações é possível gerar todos os k -itemsets frequentes e, a partir destes, produzir as Regras de Associação. Exemplo do processo de geração de itemsets frequentes é relacionado a seguir.

No início do processo de mineração de Regras de Associação cada itemset $X \subseteq A$ (A é um conjunto de itens distintos) é considerado potencialmente frequente. Ou seja, o espaço inicial de busca de itemsets frequentes é composto de todos os subconjuntos de A com exceção do conjunto vazio. Assim, mesmo os conjuntos com poucos itens tendem a possuir um espaço de busca grande. O espaço de busca para um conjunto de itens $A = \{a, b, c, d, e\}$ é ilustrado na Figura 3.1, na página seguinte.

Para que não haja necessidade de se percorrer todo o espaço de busca à procura dos itemsets de fato frequentes, algoritmos modernos para minerar Regras de Associação utilizam um método que gera e testa itemsets candidatos. Esses algoritmos geram conjuntos de itemsets potencialmente frequentes chamados conjuntos de itemsets candidatos. Utilizam-se da propriedade de linha de fronteira (*downward closure*) do suporte de um itemset (Agrawal & Srikant 1994) – que determina que todo subconjunto de um itemset

Figura 3.1: Espaço de busca para o conjunto de itens $A = \{a, b, c, d, e\}$. Fonte: (Hipp et al. 2002)

freqüente deve ser freqüente (para remover os *itemsets* com pelo menos um subconjunto de itens não-freqüente). Então, calcula-se o valor de **suporte** para cada *itemset* candidato (não removido) utilizando a base de dados D , removendo-se, na seqüência, os *itemsets* candidatos com **suporte** inferior ao **suporte** mínimo definido pelo usuário. O algoritmo inicia uma nova iteração utilizando os *itemsets* freqüentes gerados na última iteração e tem encerrada sua execução quando não houver nenhum *itemset* potencialmente freqüente podendo ser considerado um *itemset* candidato.

Na Figura 3.2, a linha em negrito (representa o **suporte** mínimo definido pelo usuário) separa os *itemsets* freqüentes (parte superior da linha) dos *itemsets* não-freqüentes (parte inferior da linha). A existência dessa linha é garantida pela propriedade da linha de fronteira do **suporte** de um *itemset*. Assim, ao invés de se percorrer todo o espaço de busca à procura de *itemsets* freqüentes, percorre-se apenas o espaço de busca acima da linha em negrito, que representa os *itemsets* freqüentes.

Os *itemsets* freqüentes para gerar Regras de Associação podem ser obtidos utilizando-se diferentes algoritmos, como: **AIS** (Agrawal et al. 1993), **SETM** (Houtsma & Swami 1995) APUD Adamo (2001), **Closet** (Pei et al. 2000), **FP-Growth** (Wang et al. 2002; Han et al. 2000), **Direct Hashing and Pruning (DHP)** (Park et al. 1997) APUD Adamo (2001), **Charm** (Zaki & Hsiao 2002), **Opus** (Webb 1995), **Dynamic Set Cou-**

Figura 3.2: Separação de itemsets frequentes e não-frequentes no espaço de busca $A = \{a, b, c, d, e\}$. Fonte: (Hipp et al. 2002)

ting (DIC) (Brin et al. 1997) APUD Adamo (2001), **Apriori** e **AprioriTid** (Agrawal & Srikant 1994). Ainda que diferentes, esses algoritmos (teoricamente) devem gerar sempre um mesmo resultado para um mesmo valor de **suporte** e **confiança** mínimos e para um mesmo conjunto de dados (Zeng et al. 2001). Por ser considerado, historicamente, um dos mais importantes algoritmos para gerar *itemsets* frequentes, o algoritmo **Apriori** é descrito na Seção 3.3.2.

3.2.6 Exemplo de Extração de Regras de Associação

Para uma maior compreensão dos procedimentos para identificação das Regra de Associação é apresentado o Exemplo 1, que utiliza o roteiro descrito nas duas etapas da Seção 3.2.2.

Exemplo 1 Seja D uma base de dados que contém um conjunto de itens $A = \{\text{bermuda, calça, camiseta, sandália, tênis}\}$ e um conjunto de transações $T = \{1, 2, 3, 4\}$, no qual a relação de itens comprados por cada transação t_i é apresentada na Tabela 3.5, na página seguinte.

Tabela 3.5: Relação de itens comprados por transação

Transações	itens comprados
1	calça, camiseta, tênis
2	camiseta, tênis
3	bermuda, tênis
4	calça, sandália

Considerando o valor de **sup-min** = 50% (2 transações) e **conf-min** = 50%, é possível obter as Regras de Associação contidas na Tabela 3.7 seguindo os passos:

- I) Encontrar todos os k -itemsets contidos na Tabela 3.5 e que possuam suporte maior ou igual a **sup-min** (itemsets freqüentes). Na Tabela 3.6 são apresentados os k -itemsets freqüentes.

Tabela 3.6: Itemsets freqüentes

Itemsets Freqüentes	Suporte
{tênis}	75%
{calça}	50%
{camiseta}	50%
{camiseta, tênis}	50%

- II) Com os k -itemsets freqüentes, com $k \geq 2$, gerar todas as Regras de Associação contidas na Tabela 3.7, na próxima página, da seguinte maneira:

regra 1: $tênis \Rightarrow camiseta$,

- $suporte = suporte(\{tênis, camiseta\}) = 50\%$, que é igual a **sup-min**.
- $confiança = \frac{suporte(\{tênis, camiseta\})}{suporte(\{tênis\})} = \frac{50}{75} = 66,66\%$, que é maior do que **conf-min**.

regra 2: $camiseta \Rightarrow tênis$,

- $suporte = suporte(\{camiseta, tênis\}) = 50\%$, que é igual a **sup-min**.
- $confiança = \frac{suporte(\{camiseta, tênis\})}{suporte(\{camiseta\})} = \frac{50}{50} = 100\%$, que é maior do que **conf-min**.

Com base nos *itemsets* obtidos no passo I, no passo II são geradas as Regras de Associação. É importante destacar que a complexidade de um sistema de mineração de Regras de

Tabela 3.7: Regras geradas

Regra	suporte	confiança
tênis $\Rightarrow$ camiseta	50%	66,66%
camiseta $\Rightarrow$ tênis	50%	100%

Associação é dependente do algoritmo utilizado para gerar os *itemsets* freqüentes (Zhang & Zhang 2002).

Na próxima seção são apresentados os primeiros algoritmos para Regras de Associação. O **Apriori** extrai todos os *itemsets* freqüentes que estão contidos em uma base de dados e o algoritmo simples proposto por Agrawal & Srikant (1994), utiliza os *itemsets* extraídos para gerar Regras de Associação. São apresentados, adicionalmente, outros exemplos de algoritmos de Regras de Associação.

3.3 Algoritmos para Regras de Associação

Nesta seção são apresentados alguns dos algoritmos propostos para extração de Regras de Associação. Na seleção dos algoritmos apresentados foi considerada sua importância pela freqüência com que são apresentados e utilizados pela comunidade.

3.3.1 AIS e SetM

Os algoritmos **AIS** (Agrawal et al. 1993) e **SetM** Agrawal et al. (1993) foram os primeiros desenvolvidos para aplicação em Regras de Associação (Agrawal et al. 1996; Adamo 2001).

No **AIS**, os *itemsets* candidatos são gerados e contados *on-the-fly*, conforme a base de dados é percorrida. Após a leitura de uma transação, é determinado quais dos *itemsets* previstos para serem freqüentes no passo anterior ($k-1$) estão presentes na transação corrente. Novos *itemsets* candidatos são gerados pela extensão destes *itemsets* freqüentes por outros atributos presentes nesta transação. Um *itemset* freqüente l_i é estendido apenas por aqueles atributos que são considerados freqüentes e ocorrem em ordem lexicográfica após qualquer outro item de l_i . Os candidatos gerados a partir de uma transação ou são adicionados ao conjunto C_k de *itemsets* candidatos mantido pelo passo atual (k), ou os

contadores das entradas correspondentes que foram criados em uma etapa anterior são incrementados. Dessa maneira, o processo iterativo continua até a geração de todos os *itemsets* freqüentes (Agrawal & Srikant 1994).

O algoritmo **SetM** foi motivado pelo desejo de usar SQL (*Structured Query Language*) para computar os *itemsets* freqüentes. Como **AIS**, o algoritmo **SetM** também gera os *itemsets* candidatos *on-the-fly*, baseando-se nas transações lidas da base de dados. Dessa maneira, o **SetM** gera cada *itemset* candidato como o **AIS** gera. Contudo, para usar a operação **join** padrão do SQL na geração das candidatas, **SetM** separa a etapa de geração das candidatas da contagem dos *itemsets*. Uma cópia dos *itemsets* candidatos é salva junto com um identificador da transação geradora em uma estrutura seqüencial. No final da etapa, o valor do **suporte** é determinado por meio da ordenação e agregação desta estrutura seqüencial (Agrawal & Srikant 1994). Este algoritmo não é objeto de detalhamento porque, em termos gerais, apresenta grande semelhança com o algoritmo **AIS** (Agrawal & Srikant 1994; Adamo 2001).

Os algoritmos **AIS** e **SetM** possuem pouco interesse prático na descoberta de Regras de Associação, à medida que possuem um desempenho operacional muito inferior aos algoritmos **Apriori** e **AprioriTid**. Esta condição foi mostrada nos experimentos realizados por Agrawal & Srikant (1994), nos quais as diferenças de desempenho (em termos de tempo) variam, em geral, de 2 a mais de 10 vezes quando reduz-se o valor do **suporte** mínimo.

3.3.2 O Algoritmo **Apriori**

Os algoritmos **Apriori** e **AprioriTid** diferem fundamentalmente dos algoritmos **AIS** e **SetM** na maneira pela qual cada *itemset* candidato é gerado e contado. Tanto no **Apriori** como no **AprioriTid**, os *itemsets* candidatos a serem contados em uma dada iteração são gerados a partir dos *itemsets* freqüentes gerados na iteração anterior – sem que se considerem as transações na base de dados. Aspecto fundamental nesta consideração é que qualquer subconjunto de um *itemset* freqüente deve também ser freqüente. Dessa maneira, pode-se gerar o conjunto de *k-itemsets* através da combinação dos $(k - 1)$ -*itemsets* e da posterior eliminação dos *itemsets* não freqüentes. Este procedimento resulta na geração de um número menor de *itemsets* candidatos, reduzindo significativamente o esforço necessário para identificação dos *itemsets* freqüentes (Agrawal & Srikant 1994).

O algoritmo **Apriori**, desenvolvido por Agrawal & Srikant (1994), é apresentado no Al-

goritmo 1. O algoritmo é utilizado para encontrar todos os k -itemsets freqüentes contidos em uma base de dados. Esse algoritmo gera um conjunto de k -itemsets candidatos e então percorre a base de dados para determinar se os mesmos são freqüentes, identificando desse modo todos os k -itemsets freqüentes. Utiliza-se a notação L_k para representar o conjunto de k -itemsets freqüentes e C_k para representar o conjunto de k -itemsets candidatos.

Algoritmo 1 Apriori (Agrawal & Srikant 1994).

Require: Uma base de dados D composta por um conjunto de itens $A = \{a_1, \dots, a_m\}$ ordenados lexicograficamente e por um conjunto de transações $T = \{t_1, \dots, t_n\}$, na qual cada transação $t_i \in T$ é composta por um conjunto de itens ordenados lexicograficamente tal que $t_i \subseteq A$.

Ensure: Resposta.

```

1:  $L_1 := \{1\text{-itemsets freqüente}\};$ 
2: for ( $k := 2; L_{k-1} \neq \emptyset; k++$ ) do
3:    $C_k := \text{apriori-gen}(L_{k-1});$  //Gera novos conjuntos candidatos
4:   for all (transações  $t \in T$ ) do
5:      $C_t := \text{subset}(C_k, t);$  //Conjuntos candidatos contidos em  $t$ 
6:     for all candidatos  $c \in C_t$  do
7:        $c.\text{count}++;$ 
8:     end for
9:   end for
10:   $L_k := \{c \in C_k \mid c.\text{count} \geq \text{sup-min}\};$ 
11: end for
12: Resposta  $:= \bigcup_k L_k$ 

```

Inicialmente o algoritmo conta a ocorrência de itens, determinando os 1-itemsets freqüentes que são armazenados em L_1 . O passo seguinte (passo k) é dividido em duas etapas. Na primeira (linha 3 do Algoritmo 1) o conjunto de itemsets freqüentes L_{k-1} obtido no passo $(k-1)$ é utilizado para gerar o conjunto de k -itemsets candidatos C_k usando a função **apriori-gen** (descrita no Algoritmo 2). A seguir (linhas 4 a 9 do Algoritmo 1), a base de dados é percorrida para determinar o valor do **suporte** dos k -itemsets candidatos em C_k . Finalmente, são identificados os k -itemsets freqüentes em cada passo (linha 10). A solução final é dada pela união dos conjuntos L_k de k -itemsets freqüentes (Agrawal & Srikant 1994). Essa solução é utilizada como entrada para algum algoritmo que gera Regras de Associação, tal como o Algoritmo 4, página 43.

A seguir são apresentadas as funções **apriori-gen** e **subset** que fazem parte do algoritmo **Apriori**.

Função *apriori-gen*

A função *apriori-gen* é apresentada no Algoritmo 2. Essa função usa como argumento L_{k-1} e retorna um superconjunto do conjunto de todos os $(k-1)$ -*itemsets*. A função executa inicialmente um *join* dos elementos dos *itemsets* em L_{k-1} com o último elemento de outros *itemsets*, diferentes do primeiro, em L_{k-1} (linhas 1 a 4 do Algoritmo 2). Em seguida são podados os k -*itemsets* que possuem algum subconjunto de tamanho $(k-1)$ não pertencente a L_{k-1} (linhas 5 a 11 do Algoritmo 2). Essa poda é orientada conforme a propriedade *downward closure* de suporte de um *itemset*, descrita na Seção 3.2.5.

Algoritmo 2 Função *apriori-gen* (Agrawal & Srikant 1994).

Require: Um conjunto de *itemsets* freqüentes L_{k-1} .

Ensure: Um conjunto de *itemsets* candidatos C_k .

```

1: insert into  $C_k$ 
2: select  $p.item_1, p.item_2, \dots, p.item_{k-1}, q.item_{k-1}$  from  $L_{k-1}p, L_{k-1}q$ 
3: where  $p.item_1 = q.item_1, \dots, p.item_{k-2} = q.item_{k-2},$ 
4:        $p.item_{k-1} < q.item_{k-1};$ 
   // A seguir, é realizada a etapa de poda, onde todos os itemsets  $c \in C_k$  são removidos se
   // algum  $(k-1)$ -subconjunto de  $c$  não pertencer a  $L_{k-1}$ .
5: for all itemset  $c \in C_k$  do
6:   for all  $(k-1)$ -subconjuntos  $s$  de  $c$  do
7:     if  $(s \notin L_{k-1})$  then
8:       delete  $c$  de  $C_k$ ;
9:     end if
10:  end for
11: end for

```

No Exemplo 2 é mostrado como a função *apriori-gen* pode ser utilizada para gerar os *itemsets* candidatos.

Exemplo 2 Suponha um conjunto qualquer de 3-*itemsets* freqüentes $L_3 = \{\{bermuda, calça, camiseta\}, \{bermuda, calça, sandália\}, \{bermuda, camiseta, tênis\}, \{bermuda, camiseta, sandália\}, \{calça, camiseta, sandália\}\}$. Aplicando a função *apriori-gen* no conjunto de 3-*itemsets* freqüentes, tem-se:

- Para as linhas 1 a 4 do Algoritmo 2 é realizado o *join* dos itens de L_3 e gerado um conjunto de 4-*itemsets* candidatos $C_4 = \{\{bermuda, calça, camiseta, sandália\}, \{bermuda, camiseta, sandália, tênis\}\}$.
- Então, nas linhas de 5 a 11 será removido de C_4 o conjunto $\{bermuda, camiseta, sandália, tênis\}$, uma vez que, o subconjunto $\{bermuda, sandália, tênis\}$ desse conjunto de itens não está contido em L_3 .

Dessa maneira, a função **apriori-gen** retorna o conjunto de 4-itemsets candidatos $C_4 = \{\{\text{bermuda}, \text{calça}, \text{camiseta}, \text{sandália}\}\}$.

Função subset

A função **subset** retorna os k -itemsets candidatos que estão contidos em uma dada transação t_i . Para isso os itemsets candidatos são armazenados em uma árvore-hash. Cada nó da árvore pode conter uma lista de itemsets ou uma tabela hash (nó folha ou nó intermediário, respectivamente). Partindo do nó raiz, a função encontra todos os itemsets candidatos presentes na transação t_i . Se um nó folha é atingido e o itemset encontrado está contido na transação t_i , uma referência é adicionada ao conjunto de resposta. Se um nó intermediário é atingido a partir de um item a_j , cada item é pesquisado (hash) após a_j em t_i . Isso é possível porque os itens estão em ordem lexicográfica. No nó raiz, todos os itens a_j em t_i são pesquisados. No Exemplo 3 é ilustrado como a função **subset** é utilizada.

Exemplo 3 Dados um conjunto qualquer de 2-itemsets candidatos $C_2 = \{\{\text{bermuda}, \text{calça}\}, \{\text{bermuda}, \text{camiseta}\}, \{\text{camiseta}, \text{meia}\}, \{\text{sandália}, \text{tênis}\}\}$ armazenado em uma árvore-hash, conforme apresentado na Figura 3.3 (neste exemplo a árvore possui apenas um nível) e uma transação $t_i = \{\text{camiseta}, \text{meia}, \text{tênis}\}$, a função **subset** irá encontrar todos os itemsets contidos na transação verificando apenas as células correspondentes aos valores contidos na transação $t_i = \{\text{camiseta}, \text{meia}, \text{tênis}\}$, o que resultará no 2-itemset $\{\text{camiseta}, \text{meia}\}$.

Figura 3.3: Itemsets candidatos armazenados em uma árvore-hash.

No Exemplo 4, na página seguinte é apresentado o funcionamento geral do algoritmo **Apriori** aplicado a um pequeno conjunto de transações.

Exemplo 4 Seja D uma base de dados que contém um conjunto de itens $A = \{\text{bermuda, calça, camiseta, sandália, tênis}\}$ e um conjunto de transações $T = \{1, 2, 3, 4\}$, no qual a relação de itens comprados por cada transação t_i é apresentada na Tabela 3.5, página 34 do Exemplo 1. O valor do suporte mínimo (**sup-min**) é igual a 50%.

Ao aplicar o **Apriori** na base de dados D da Tabela 3.5, o algoritmo irá percorrer a base de dados gerando um conjunto C_1 de 1-itemsets candidatos e em seguida produzirá um conjunto L_1 de 1-itemsets freqüentes (linha 1 do Algoritmo 1), como apresentado no esquema abaixo relacionado.

Percorre a Base D ⇒	C_1		L_1																				
	<table><tr><th><i>itemset</i></th><th><i>Suporte</i></th></tr><tr><td>{calça}</td><td>50%</td></tr><tr><td>{camiseta}</td><td>50%</td></tr><tr><td>{tênis}</td><td>75%</td></tr><tr><td>{bermuda}</td><td>25%</td></tr><tr><td>{sandália}</td><td>25%</td></tr></table>	<i>itemset</i>	<i>Suporte</i>	{calça}	50%	{camiseta}	50%	{tênis}	75%	{bermuda}	25%	{sandália}	25%		<table><tr><th><i>itemset</i></th><th><i>Suporte</i></th></tr><tr><td>{calça}</td><td>50%</td></tr><tr><td>{camiseta}</td><td>50%</td></tr><tr><td>{tênis}</td><td>75%</td></tr></table>	<i>itemset</i>	<i>Suporte</i>	{calça}	50%	{camiseta}	50%	{tênis}	75%
	<i>itemset</i>	<i>Suporte</i>																					
	{calça}	50%																					
	{camiseta}	50%																					
	{tênis}	75%																					
{bermuda}	25%																						
{sandália}	25%																						
<i>itemset</i>	<i>Suporte</i>																						
{calça}	50%																						
{camiseta}	50%																						
{tênis}	75%																						

Na seqüência é executada a função **apriori-gen** (linha 3 do Algoritmo 1) gerando um conjunto C_2 de 2-itemsets candidatos. O próximo passo, realizado nas linhas 4 a 11 do Algoritmo 1, é verificar quais dos 2-itemsets candidatos em C_2 são freqüentes, gerando um conjunto L_2 de 2-itemsets freqüentes, como é apresentado a seguir.

C_2		C_2		L_2																
<table> <tr><th><i>itemset</i></th></tr> <tr><td>{calça,camiseta}</td></tr> <tr><td>{calça,tênis}</td></tr> <tr><td>{camiseta,tênis}</td></tr> </table>	<i>itemset</i>	{calça,camiseta}	{calça,tênis}	{camiseta,tênis}	<p><i>Percorre a Base D</i></p> <p>$\Rightarrow$</p>	<table> <tr><th><i>itemset</i></th><th><i>Suporte</i></th></tr> <tr><td>{calça,camiseta}</td><td>25%</td></tr> <tr><td>{calça,tênis}</td><td>25%</td></tr> <tr><td>{camiseta,tênis}</td><td>50%</td></tr> </table>	<i>itemset</i>	<i>Suporte</i>	{calça,camiseta}	25%	{calça,tênis}	25%	{camiseta,tênis}	50%		<table> <tr><th><i>itemset</i></th><th><i>Suporte</i></th></tr> <tr><td>{camiseta,tênis}</td><td>50%</td></tr> </table>	<i>itemset</i>	<i>Suporte</i>	{camiseta,tênis}	50%
<i>itemset</i>																				
{calça,camiseta}																				
{calça,tênis}																				
{camiseta,tênis}																				
<i>itemset</i>	<i>Suporte</i>																			
{calça,camiseta}	25%																			
{calça,tênis}	25%																			
{camiseta,tênis}	50%																			
<i>itemset</i>	<i>Suporte</i>																			
{camiseta,tênis}	50%																			

Vários passos são realizados pelo algoritmo **Apriori** até que não seja mais possível gerar um conjunto de itemsets freqüentes.

Ao término da execução do algoritmo é gerado o conjunto *Resposta* (Tabela 3.8) pela união de todos os conjuntos L_k que contêm os k -itemsets freqüentes (linha 12 do Algoritmo 1).

Tabela 3.8: *Conjunto Resposta contendo os itemsets freqüentes*

<i>itemsets</i> Freqüentes	Suporte
$\{\text{tênis}\}$	75%
$\{\text{calça}\}$	50%
$\{\text{camiseta}\}$	50%
$\{\text{camiseta, tênis}\}$	50%

O conjunto *Resposta* é utilizado como entrada para algum algoritmo que gera Regras de Associação, como o algoritmo apresentado na Seção 3.3.4.

3.3.3 AprioriTid

O algoritmo **AprioriTid**, desenvolvido por (Agrawal & Srikant 1994) é descrito no Algoritmo 3, e assim como o **Apriori**, utiliza a função *apriori-gen* para determinação dos *itemsets* candidatos. O **AprioriTid** difere basicamente do **Apriori** por não utilizar a base de dados para a determinação do **suporte** dos *itemsets* candidatos após a primeira iteração. Para tanto, estabelece uma codificação baseada nos *itemsets* candidatos do passo anterior. Esta codificação é armazenada no conjunto $\overline{C}_k$ no formato $\langle Tid, \{X_k\} \rangle$, sendo cada X_k um potencial k -*itemset* presente em uma transação com identificador *Tid*. Quando $k = 1$, $\overline{C}_k$ corresponde à base de dados D , quando $k > 1$, $\overline{C}_k$ é gerado pelo algoritmo (linha 10 do Algoritmo 3). Cada membro de $\overline{C}_k$ correspondente à transação t_i é dado por $\langle t_i.Tid, \{c \in C_k | c \text{ está contido em } t_i\} \rangle$. Se uma transação não possui nenhum *itemset* candidato então $\overline{C}_k$ não possui uma entrada para esta transação. Dessa maneira, o número de elementos em $\overline{C}_k$ pode ser menor que o número de transações na base de dados D , principalmente para valores elevados de k , o que reduz consideravelmente a necessidade de leitura de dados (Agrawal & Srikant 1994).

Algoritmo 3 Algoritmo **AprioriTid** (Agrawal & Srikant 1994).

Require: Uma base de dados D composta por um conjunto de itens $A = \{a_1, \dots, a_m\}$ ordenados lexicograficamente e por um conjunto de transações $T = \{t_1, \dots, t_n\}$, no qual cada transação $t_i \in T$ é composta por um conjunto de itens tal que $t_i \subseteq A$.

Ensure: Resposta.

```

1:  $L_1 = \{1\text{-itemset freqüente}\}$ 
2:  $\overline{C}_1 = D$ 
3: for ( $k = 2; L_{k-1} \neq \emptyset; k++$ ) do
4:    $C_k = \text{apriori-gen}(L_{k-1})$ ; //Gera novos conjuntos candidatos
5:    $\overline{C}_k = \emptyset$ 
6:   for all entradas  $t \in \overline{C}_{k-1}$  do
7:      $C_t = \{c \in C_k | (c - c[k]) \in t.\text{conjunto de itemset} \wedge (c - c[k-1]) \in t.\text{conjunto de itemset}\}$ 
       //determina os itemsets candidatos em  $C_k$  contidos na transação com o identificador  $t.Tid$ 
8:     for all candidato  $c \in C_t$  do
9:        $c.\text{count}++$ 
10:    end for
11:    if ( $C_t \neq \emptyset$ ) then
12:       $\overline{C}_k += \langle t.Tid, C_t \rangle$ 
13:    end if
14:  end for
15:   $L_k = \{c \in C_k | c.\text{count} \geq \text{sup-min}\}$ 
16: end for
17: Resposta =  $\bigcup_k L_k$ ;
```

3.3.4 Algoritmo Simples para Gerar Regras de Associação

Entre os algoritmos que geram Regras de Associação para uma base de dados a partir dos *itemsets* freqüentes, é apresentado, no Algoritmo 4, na página oposta, o algoritmo proposto por Agrawal & Srikant (1994), um dos mais simples para realização dessa tarefa.

O algoritmo é executado para os k -*itemsets* freqüentes, com $k \geq 2$. Inicialmente são gerados os subconjuntos não vazios de um *itemset* freqüente. Em seguida são geradas regras do tipo $LHS \Rightarrow RHS$ utilizando os subconjuntos definidos, que satisfazem a condição: **confiança** da regra maior ou igual a **confiança** mínima especificada pelo usuário (**conf-min**).

O procedimento para gerar uma Regra de Associação a partir de um conjunto de *itemsets* freqüentes é apresentado no Exemplo 5 (será considerado um dos conjuntos de *itemsets* freqüentes contidos na Tabela 3.8 do Exemplo 4).

Exemplo 5 Seja $\text{resposta}_4 = \{\text{camiseta}, \text{tênis}\}$, o 2-*itemset* freqüente contido no conjunto Resposta apresentado na Tabela 3.8 e a **confiança** mínima (**conf-min**) igual a 50%,

Algoritmo 4 Gera Regras de Associação (Agrawal & Srikant 1994).

Require: Um conjunto *Resposta* contendo todos os *k-itemsets* freqüentes, com $k \geq 2$.

Ensure: Um conjunto de Regras de Associação.

```

1: for all (k-itemset freqüente  $l_k, k \geq 2$ ) do
2:   Call genrules ( $l_k, l_k$ );
3: end for //O procedimento genrules gera todas as regras válidas  $\tilde{a} \Rightarrow (l_k - \tilde{a})$ , para todo
    $\tilde{a} \subset a_m$ 
4: procedure genrules ( $l_k$ : k-itemset freqüente,  $a_m$ : m-itemset freqüente)
5:    $A := \{(m-1) - \text{itemsets } a_{m-1} \mid a_{m-1} \subset a_m\}$ ;
6:   for all ( $a_{m-1} \in A$ ) do
7:      $conf := sup(l_k) / sup(a_{m-1})$ ;
8:     if ( $conf \geq \text{conf-min}$ ) then
9:       Output: regra  $a_{m-1} \Rightarrow (l_k - a_{m-1})$ , com confiança =  $conf$  e suporte =  $sup(l_k)$ ;
10:    if ( $m-1 > 1$ ) then
11:      Call genrules ( $l_k, a_{m-1}$ ); //Gera regras com subconjuntos de  $a_{m-1}$  como antecedente
12:    end if
13:  end if
14: end for

```

no procedimento *genrules* são gerados todos os subconjuntos $\tilde{a}_j$ a partir de $resposta_4$ (linha 5 do Algoritmo 4):

Subconjunto $\tilde{a}_1 = \{\text{tênis}\}$ e subconjunto $\tilde{a}_2 = \{\text{camiseta}\}$.

Como mostrado a seguir, o passo seguinte na execução do algoritmo consiste no cálculo da *confiança* desses subconjuntos e na geração de uma Regra de Associação para cada subconjunto com *confiança* maior ou igual a **conf-min** (linhas 6 a 9 do Algoritmo 4).

regra gerada pelo subconjunto $\tilde{a}_1 : \text{tênis} \Rightarrow \text{camiseta}$,

- $suporte = suporte(\{\text{tênis}, \text{camiseta}\}) = 50\%$.
- $confiança = \frac{suporte(\{\text{tênis}, \text{camiseta}\})}{suporte(\{\text{tênis}\})} = \frac{50}{75} = 66,66\%$, que é maior que **conf-min**.

regra gerada pelo subconjunto $\tilde{a}_2 : \text{camiseta} \Rightarrow \text{tênis}$,

- $suporte = suporte(\{\text{camiseta}, \text{tênis}\}) = 50\%$.
- $confiança = \frac{suporte(\{\text{camiseta}, \text{tênis}\})}{suporte(\{\text{camiseta}\})} = \frac{50}{50} = 100\%$, que é maior que **conf-min**.

Em seguida é verificado que $m-1 = 1$ (m possui valor 2), e a execução do algoritmo é concluída.

3.4 Aspectos Complementares Sobre Regras de Associação

Até o presente momento foram considerados nas descrições e exemplos apresentados atributos do tipo binário, seguindo a abordagem adotada por Agrawal et al. (1993). Entretanto, cabe dizer que as bases de dados de interesse podem ser compostas por outros tipos de atributos (categóricos e numéricos). Nesta seção serão apresentadas questões complementares relacionadas aos atributos: o uso de taxonomia de atributos, de restrições de atributos e de atributos categóricos e numéricos. Também serão apresentados estudos relacionados ao uso de Regras de Associação na identificação de padrões sequenciais, de ocorrências periódicas, de associações negativas e uso de pesos em atributos.

3.4.1 Regras de Associação com Taxonomia

As taxonomias refletem uma visão coletiva ou individual, sempre arbitrária, de como os atributos podem ser hierarquicamente classificados. Eventualmente, múltiplas taxonomias podem estar presentes simultaneamente refletindo a existência de diferentes pontos de vista ou a possibilidade de classificações distintas para o mesmo conjunto de atributos. Na Figura 3.4 é apresentado um exemplo do uso de taxonomia para alguns itens de vestuário: “camisetas e bermudas são roupas leves; calças e roupas leves são roupas; tênis e sandálias são calçados”.

As principais motivações apresentadas por Srikant (2001) para o emprego de taxonomias em Regras de Associação são apresentadas a seguir:

- regras simples (aquelas cujos elementos são compostos exclusivamente por itens terminais na taxonomia – atributos) – podem não ter **suporte** suficiente para ser incluídas na solução, mas podem produzir conhecimento interessante quando agrupadas segundo uma dada taxonomia;
- generalização de regras muito especializadas. Como no caso anterior, mesmo considerando-se regras simples com elevados níveis de **suporte** e **confiança**, estas ainda podem ser agrupadas em regras gerais, melhorando-se a compreensibilidade por parte do usuário (também com maiores valores de **suporte** e **confiança**);
- identificação de regras interessantes com uso de informações da taxonomia. Liu et al. (2000) utilizam esta tarefa no aplicativo **AIAS** apresentado na Seção 4.6.2.

Figura 3.4: Exemplo de taxonomia

Do ponto de vista da lógica proposicional, a introdução de taxonomias é uma maneira de se complementar a capacidade descritiva das Regras de Associação com disjunções. Por exemplo, “roupas leves $\Rightarrow$ calçados” pode ser lido como (camisetas ou bermudas $\Rightarrow$ tênis ou sandália). Pode-se observar que esta é uma maneira restritiva de introdução de disjunções, uma vez que os argumentos de disjunção não são automaticamente descobertos, como o são os argumentos de conjunção². As disjunções devem ser introduzidas nas regras, explícita e arbitrariamente, por meio de taxonomias especificadas pelo usuário (Adamo 2001).

As regras com taxonomias são denominadas **Regras de Associação generalizadas** e buscam identificar associação entre os itens que estejam no mesmo nível de taxonomia ou em níveis diferentes (Srikant & Agrawal 1995). Um exemplo de Regra de Associação generalizada é “camiseta $\Rightarrow$ calçado”.

Uma maneira direta, mas pouco eficiente, de se obter esse tipo de regra seria a geração de um atributo novo para cada nível hierárquico (geralmente os níveis da hierarquia excluem o nível de fundo, isto é, os 1-*itemset*). Algoritmos que incorporam o uso de informação hierárquica no processo de extração de padrões foram propostos por Srikant & Agrawal (1995) e por Han & Fu (1995). Uma abordagem orientada a objetos é apresentada por Scott & Liu (1996). Thomas & Sarawagi (1998) propõem o emprego de consultas do

²As conjunções estão presentes no antecedente e no conseqüente de uma Regra de Associação de maneira implícita, conforme a definição desta.

tipo SQL para identificar Regras de Associação em múltiplos níveis. Finalmente, **Regras de Associação flexíveis em múltiplos níveis** são apresentadas por Shen & Shen (1998).

3.4.2 Regras de Associação com Restrições Sobre Atributos

Como já mencionado, os usuários geralmente não estão interessados em todo o conjunto de regras, mas sim num conjunto particular que contenha alguns atributos. Por exemplo, regras que contenham os atributos A e B como conseqüentes ou regras que não contenham o atributo C como antecedente. O uso de restrição sobre atributos é uma maneira alternativa para reduzir o escopo do conhecimento extraído.

A aplicação de restrições sobre atributos na etapa de Pós-processamento é trivial, podendo ser realizada por meio da aplicação de filtros. Porém pode-se obter vantagens significativas em se tratando da velocidade de processamento considerando-se estas restrições durante o processo de geração de regras.

Em seu trabalho Srikant et al. (1997) definem as **Regras de Associação com restrições** e propõem os algoritmos – **Reorder**, **MultipleJoin** e **Direct** –, derivados do algoritmo **Apriori**, que consideram a inclusão de restrição sobre atributos. São adotadas funções do tipo booleana para indicar a presença ou ausência dos atributos de interesse. Também são adotadas restrições sobre a taxonomia dos atributos, podendo-se considerar ou desconsiderar regras com presença ou ausência de ancestrais ou descendentes de determinados atributos. Dentre os resultados apresentados pelos autores, destaca-se a melhoria de 5 a 20 vezes no tempo de processamento em função do nível de seletividade (restrições) adotado para os atributos. Os resultados foram obtidos a partir da comparação com o desempenho do algoritmo **Apriori**.

Um exemplo do uso de restrições que podem ser empregadas é “ $(camisetas \wedge calçados) \vee (descendentes(roupas\ leves) \wedge \neg ancestrais(sandálias))$ ” que expressa a restrição nas regras contêm *camisetas* e *calçados* ou os descendentes de *roupas leves* e não contêm os ancestrais de *sandálias*. Ao invés de descobrir todas as regras e depois realizar a poda utilizando as restrições desejadas, os autores incorporam o uso de restrições no algoritmo de geração das Regras de Associação.

A abordagem denominada **Consultas de associações com restrição**, tratada por Ng et al. (1998) e Lakshmanan et al. (1999), permite lidar com restrições mais complexas no processo de geração de Regras de Associação. Como exemplo, a expressão :

$(S_1, S_2) | S_1 \cdot \text{Tipo} \subseteq \text{Lanches} \wedge S_2 \cdot \text{Tipo} \subseteq \text{cervejas} \wedge \max(S_1, \text{Preço}) \leq \min(S_2, \text{Preço})$ que encontra os *lanches* mais baratos que as *cervejas*. Outra abordagem é proposta por Bayardo et al. (1999), que impõe, além das restrições relacionadas ao **suporte** mínimo e à **confiança** mínima, uma nova restrição para garantir que toda regra possua “vantagem preditiva” em relação às suas simplificações.

3.4.3 Regras de Associação em Atributos Categóricos e Numéricos

Como apresentado na Seção 3.2, é considerado que a base de dados sobre a qual são aplicados os algoritmos de Regras de Associação está representada numa tabela booleana, onde a presença de um atributo em uma transação é representada por “1” e sua ausência por “0”. Contudo, a maioria das bases de dados possui atributos de maior complexidade: atributos categóricos, podendo assumir vários valores, e atributos numéricos. Para considerar esses tipos de atributos foi introduzido por Srikant & Agrawal (1996a) o conceito **Regras de Associação quantitativas**.

Um exemplo de Regra de Associação quantitativa é $\langle \text{Idade} : 30 \dots 40 \rangle \wedge \langle \text{Casado} : \text{Sim} \rangle \Rightarrow \langle \text{NUMCARROS} : 2 \rangle (40\%; 90\%)$ que indica que em 40% das transações totais, 90% das pessoas cuja idade está entre 30 e 40 anos e são casadas têm dois carros. O algoritmo proposto pelos autores busca dividir os valores de atributos quantitativos e categóricos em intervalos, de tal modo a maximizar a associação entre os itens da regra.

Segundo Srikant & Agrawal (1996a), uma maneira de considerar a complexidade desses tipos de atributos é realizar o mapeamento dos atributos para a forma binária. No caso dos atributos discretos, pode-se fazer o simples mapeamento criando-se um novo atributo para cada valor do atributo em questão. No caso dos atributos contínuos, uma possível solução é discretizar os atributos e então realizar o mapeamento para a tabela binária.

Os autores apresentam as dificuldades relacionadas a esta abordagem considerando-se o **suporte** mínimo e a **confiança** mínima:

- no caso do **suporte** mínimo (**sup-min**), se o número de intervalos para um atributo numérico (ou valores, para atributos categóricos) é elevado, o valor de **suporte** para cada intervalo definido pode ser baixo. Sendo assim, sem o uso de intervalos amplos, algumas regras envolvendo este atributo podem não ser encontradas por não possuírem **suporte** suficiente;
- no caso da **confiança** mínima (**conf-min**), há alguma perda de informação quando

particionam-se valores em intervalos. Algumas regras podem ter **confiança** mínima apenas quando um item no antecedente consiste em um único valor (ou num pequeno intervalo). A perda de informação é ampliada à medida que se aumenta o tamanho do intervalo.

Os problemas apresentados conduzem a uma situação dúbia – se os intervalos são amplos a regra pode não ter **confiança** suficiente; se os intervalos são pequenos a regra pode não apresentar **suporte** suficiente. Srikant & Agrawal (1996a) contornam esse problema considerando todos os intervalos contíguos aos intervalos adotados (para o caso do **sup-min**) e aumentando o número de intervalos considerados para o caso da **conf-min**.

Abordagem alternativa para essa problemática é apresentada por Fukuda et al. (1996), que introduziu o conceito de **Regras de Associação otimizadas**, no qual a discretização dos valores dos atributos numéricos é realizada de maneira a maximizar os valores de **suporte** e **confiança** das regras. Idéia semelhante foi investigada por Rastogi & Shim (1998), considerando, além dos atributos numéricos, atributos do tipo categórico. Wang et al. (1998) propõem a fusão de intervalos baseada no grau de interesse para combinar diferentes intervalos em um único intervalo, de maneira a maximizar o grau de interesse de uma regra.

Em estudo relacionado a atributos numéricos, Kuok et al. (1998) propõem **Regras de Associação do tipo fuzzy**. Uma Regra de Associação *fuzzy* é da forma “Se X for A , então Y é B ” com X , Y conjuntos de atributos e A , B conjuntos *fuzzy* que descrevem X e Y respectivamente. É assumido que conjuntos *fuzzy* para cada atributo são fornecidos como entrada. Fu et al. (1998) empregam esquema de agrupamento (*clustering*) para extrair os conjuntos *fuzzy*.

3.4.4 Regras de Associação em Padrões Sequenciais

Regras de Associação visam a descoberta de co-ocorrências em um dado intervalo de tempo. Com o armazenamento de dados por um longo período e o desenvolvimento de Bancos de Dados Temporais, a descoberta de padrões sequenciais se tornou assunto importante. Exemplo de padrão sequencial: *clientes tipicamente alugam “Guerra nas Estrelas”, “O império contra ataca” e “O Retorno do Jedi”* (Agrawal & Srikant 1995). Os itens em um padrão sequencial precisam ser consecutivos e estar numa ordem determinada. Três algoritmos foram propostos por Agrawal & Srikant (1995) para extrair padrões sequenciais a partir de uma base de dados de transações. Srikant & Agrawal (1996b) ampliam essa

abordagem, permitindo o uso de janelas correções e de informações hierárquicas sobre os atributos.

Mannila et al. (1995) e Mannila & Toivonen (1996) definem os termos *episódios freqüentes* – uma coleção de eventos em um certo padrão –, e *episódios generalizados* – episódios que satisfazem determinadas condições – em uma sequência de dados. Valery et al. (1998) e Das et al. (1998) também propõem algoritmos para a descoberta de episódios freqüentes.

3.4.5 Regras de Associação Periódicas

Em uma sequência de dados, Regras de Associação podem identificar propriedades periódicas. Algumas regras podem ter valor de **suporte** superior ao **suporte** mínimo para um pequeno período de tempo, mesmo não tendo **suporte** tão elevado quando computado no Banco de Dados completo.

Özden et al. (1998) introduzem o conceito de **Regras de Associação cíclicas**, que são regras com **confiança** e **suporte** acima dos valores especificados em intervalos de tempo regulares. Esse tipo de regra, por exemplo, pode conter a afirmação “*Pessoas compram jornais junto com leite todos os domingos*”. Ao invés de identificar conjuntos de regras a cada instante de tempo e então produzir as regras periódicas a partir destes conjuntos, os algoritmos propostos por Özden et al. (1998) incorporam heurísticas para otimizar o processo. Estes algoritmos podem ser aplicados apenas em casos nos quais as regras são repetidas a todo instante de tempo t .

Ramaswamy et al. (1998) estendem o estudo de **Regras de Associação periódicas** para encontrar **Regras de Associação em calendários**. O termo calendário pode ser entendido como um período de tempo recorrente (semana, mês etc). Nessa abordagem, busca-se identificar regras que sigam padrões em um calendário especificado pelo usuário. Além disso, os algoritmos foram propostos para extrair regras considerando quaisquer calendários. Estes estudos estão baseados no conceito de **periodicidade completa**, i.e., a regra deve ser válida em todos os instantes de tempo definidos por um dado padrão (calendário). Han & andYiwen Yin (1999) propõem flexibilizar essa restrição buscando identificar *padrões periódicos parciais*, um tipo de periodicidade menos restritiva.

3.4.6 Regras de Associação Ponderadas

Nos algoritmos até agora apresentados, os atributos presentes na base de dados são considerados como tendo o mesmo grau de importância. Entretanto, em determinadas situações, alguns desses atributos podem possuir importância diferenciada, como no caso de fatores condicionantes de uma doença.

Cai et al. (1998) introduzem o conceito de **Regras de Associação ponderadas**, que considera atributos com diferentes graus de importância na geração de regras a partir da atribuição de pesos. Os pesos geralmente estão relacionados às promoções de determinados produtos, ao preço destes ou a outra qualquer característica diferenciadora. Considerando que a abordagem tradicional de identificação dos *itemsets* (*bottom-up*) não é válida neste caso, os autores propõem um novo algoritmo e definem o **suporte ponderado** de um *itemset* e de uma Regra de Associação.

3.4.7 Regras de Associação Negativas

Em **Regras de Associação negativas** considera-se que se um determinado item ocorre em uma transação outro não deverá ocorrer (Savasere et al. 1998). Exemplo desse tipo de regra é “a maioria das pessoas que compra comida congelada não compra legumes”.

Uma solução possível para essa abordagem é definir os valores de **suporte** mínimo e **confiança** mínima tão baixos quanto possível. Esta solução entretanto gera muitas Regras de Associação negativas desinteressantes. Uma abordagem mais eficiente é proposta por Savasere et al. (1995), na qual são extraídas regras para as quais se espera um grau alto de associação positiva e o **suporte** é significativamente menor que o esperado.

3.4.8 Regras de Associação de Proporções

Em **Regras de Associação de proporções**, apresentada por Korn et al. (1998), o objetivo é identificar correlações existentes entre quantidades ou preços de itens diferentes. Exemplo desse tipo de regra “*clientes gastam tipicamente 1:2:5 reais em pão: leite: manteiga*”.

No algoritmo proposto por Korn et al. (1998) é definido método para identificar a qualidade das regras denominado de “erro de aproximação”. As **Regras de Associação**

de proporções podem ser empregadas, por exemplo, para calcular valores ausentes, ainda que vários valores estejam ausentes simultaneamente.

3.5 Algumas Ferramentas para Geração de Regras de Associação

As ferramentas para geração de Regras de Associação, em geral, tratam-se de aplicativos que auxiliam, com maior ou menor grau de facilidade para o usuário, a realização do processo de Mineração de Dados como um todo. Geralmente dispõem de facilidades que permitem a unificação, transformação, limpeza e redução dos dados; a realização de análises estatísticas; a escolha entre diferentes funções de Extração de Conhecimento; a visualização e avaliação do conhecimento extraído.

A análise destas ferramentas objetiva, principalmente, a coleta de subsídios para estabelecer um formato padrão de representação de regras, atributos e dados; a identificação dos parâmetros a serem armazenados para avaliação das regras na etapa de Pós-processamento e a definição das maneiras de apresentação dos padrões extraídos.

A Tabela 3.9 sintetiza algumas das características das ferramentas de Mineração de Dados que possuem capacidade para gerar Regras de Associação. A lista de ferramentas analisadas foi obtida no site www.kdnuggets.com e complementada com os aplicativos Oracle e Microsoft Data Analyzer.

3.6 Considerações Finais

Neste capítulo foi realizada descrição pormenorizada da tarefa para Extração de Conhecimento denominada Regras de Associação. Foram apresentadas sua definição formal e a notação empregada, descritos os algoritmos utilizados e analisados aspectos complementares sobre atributos a serem considerados. O capítulo inclui, adicionalmente, algumas ferramentas para Mineração de Dados que disponibilizam esta tarefa.

Em termos gerais, as Regras de Associação permitem identificar associações entre atributos ou conjuntos de atributos de uma base de dados. Apesar de sua utilidade em aplicações práticas, os algoritmos desta tarefa tendem a produzir grande quantidade de regras, muitas das quais sem qualquer interesse para o usuário (Liu et al. 2000). Analisando manualmente um grande volume de regras, dificilmente serão identificadas

Tabela 3.9: Ferramentas para Regras de Associação

Nome	Algoritmo	Sistema Operacional	Fabricante – site	Tipo de Aplicativo*
comerciais				
Azmy SuperQuery	n/d	Windows95/NT	AZMY Thinkware Inc. – www.azmy.com	pacote
Clementine Suite			clementine – SPSS www.spss.com	pacote
IREX			www.giwebb.com	
Nuggets			www.data-mine.com	pacote
PolyAnalyst	n/d	Microsoft Windows NT 4.0 com Service Pack 4 ou superior OU Windows95/98/2000 mais o Microsoft Internet Explorer 5 ou superior	Megaputer Intelligence – www.megaputer.com	pacote
Magnum Opus	Opus	Windows98/Me/2000/XP OU Unix (Solaris 2.5) OU Unix (Irix) OU Unix (Linux)	Rule Quest – www.rulequest.com	específico
CBA	Apriori	Windows95/98/NT 4.0 OU Linux	National University of Singapore – www.comp.nus.edu.sg/liub	pacote
WizRule	n/d	Windows95/98/ML/NT/2000	WizSoft Inc. – www.wizsoft.com	específico
WizWhy	n/d	Windows95/98/ML/NT/2000	WizSoft Inc. – www.wizsoft.com	específico
XAffinity	n/d	Windows95/98/NT/2000	Exclusive Ore Inc – www.exclusiveore.com www.xore.com	específico
XpertRule Miner	Apriori	Windows95/98/NT/2000/XP	Attar Software Ltd. – www.attar.com	pacote
TMiner	TBAR (otimização Apriori)	Windows95/98/NT/2000 OU Linux	Grupo IdBIS	pacote
DataMite	n/d	Windows 3.1 ou superior	Dr Philip Vasey através do LPA Prolog e ProData	pacote
Microsoft Data Analyzer	n/d	Windows98/Me/NT4.0 com Service Pack 6 (SP6) or superior/Windows2000/XP	Microsoft Corp. – www.microsoft.com	pacote
Oracle	n/d	Windows OU Solaris OU HP-UX OU IBM AIX OU Compaq Tru64 OU Linux	Oracle Corp. – www.oracle.com	pacote
MineSet TM	n/d	WindowsNT/2000 OU Unix	Silicon Graphics Inc. – www.sgi.com	pacote
Intelligent Miner	n/a	n/a	IBM Corp. – www.ibm.com	n/a
gratuitos				
Apriori			fuzzy.cs.uni-magdeburg.de/borgelt	
Apriori, FP-growth and Eclat			implementations by Bart Goethals www.cs.helsinki.fi/u/goethals	
ARtool			www.cs.umb.edu laurARtool	
FIMI			fimi.cs.helsinki.fi	
WEKA	Apriori	plataforma JAVA	University of Waikato – www.cs.waikato.ac.nz	pacote
obs:	n/a – não analisado n/d – não disponível Indica se o aplicativo é específico para Regras de Associação, ou se é um pacote que disponibiliza outras técnicas de Mineração de Dados.			

aquelas realmente interessantes. A solução dessa questão passa necessariamente pela etapa

de Pós-processamento do conhecimento extraído, especialmente, por meio do emprego de medidas que permitem analisar e avaliar as regras sob variados aspectos.

No capítulo a seguir é apresentado um elenco de medidas que podem ser utilizadas para avaliação de regras, são mostrados princípios de interessabilidade de regras e o aplicativo **AIAS** (que considera aspectos subjetivos na análise do interesse de Regras de Associação). O capítulo inclui um experimento para verificação do potencial de uso das medidas objetivas descritas como auxiliares na etapa de Pós-processamento do conhecimento.

Avaliação de Regras de Associação: Técnicas e Ambientes

4.1 Motivação

A partir da proposição da tarefa Regras de Associação por Agrawal et al. (1993), a maioria dos esforços em pesquisas sobre a mesma tem sido direcionada ao aprimoramento do desempenho dos algoritmos de extração de regras, em especial no sentido de propor soluções mais eficientes para o problema da identificação e contagem dos *itemsets*. Em paralelo, diversos aplicativos foram desenvolvidos incorporando os avanços propiciados por essas pesquisas. Existe outra problemática associada às Regras de Associação que está relacionada à quantidade de regras geradas, mesmo quando a tarefa é aplicada a bases de dados relativamente pequenas. Na Tabela 4.1, são apresentados os resultados obtidos por meio da aplicação de algoritmos de Regras de Associação a algumas bases de dados. Pode-se notar a grande quantidade de regras geradas quando se compara às quantidades usuais obtidas por outras tarefas de Mineração de Dados (classificação e regressão, por exemplo). Esta quantidade geralmente excede a dimensão da base de dados minerada e, principalmente, vai além da capacidade de interpretação do ser humano. Adicionalmente, o especialista do domínio ou o usuário final do sistema está interessado em uma pequena fração das regras, aquelas que apresentam algum conhecimento útil,

interessante ou inovador (Imieliński & Virmani 1998). Assim, é necessário desenvolver metodologias e ferramentas eficientes que apoiem o usuário na identificação de regras de seu interesse durante a etapa de Pós-processamento do conhecimento.

Tabela 4.1: *Problemática de Regras de Associação*

Base	Nº de atributos	Nº de exemplos	Nº de regras	suporte (%)	confiança (%)	tempo (s)
Talassemia Alfa	10	339	8.827	2	50	–
Qualidade de vida urbana	19	120	24.713	2	50	–
Art A	10	18.539	2.147	6	25	0,2
Art B	1.000	1.000	18.150	6	25	2,6
Art E	100	25.000	596.626	6	25	340,3
nt60tl150mi1pa20	995	60.000	1.459.070	6	25	1.896,2

Desse modo, são apresentados neste capítulo princípios de interessabilidade de regras, para se definir as principais características que regras interessantes devem possuir. São apresentadas, adicionalmente, algumas técnicas de Pós-processamento de conhecimento propostas por diversos pesquisadores. Na sequência são descritas diferentes medidas (objetivas e subjetivas) para a avaliação de regras e também duas técnicas utilizadas no apoio à análise de regras utilizando medidas objetivas. No final do capítulo são descritos, de forma sucinta, o aplicativo *AIAS*, os ambientes *DISCOVER* e *RULEE* e o módulo computacional *RULEE_GAR*, instrumentos que podem ser empregados na avaliação de conhecimento.

4.2 Princípios de Interessabilidade de Regras

Medidas do grau de interesse de uma regra são baseadas em conceitos tais como os de utilidade e inesperabilidade. A possibilidade do usuário em obter proveito da sua utilização é um indicador da interessabilidade da mesma. A inesperabilidade indica que as regras que representam um conhecimento “inesperado” para o usuário podem ser consideradas interessantes, uma vez que irão ampliar ou contestar os conhecimentos que o usuário possui sobre o domínio (Silberschatz & Tuzhilin 1995).

Vale dizer que a inesperabilidade e a utilidade são conceitos não excludentes entre si, podendo ser combinados nas formas apresentadas (Silberschatz & Tuzhilin 1996):

- regras inesperadas e utilizáveis;

- regras inesperadas e não utilizáveis;
- regras esperadas e utilizáveis.

As regras pertencentes às duas primeiras categorias são identificadas pela aplicação de medidas de inesperabilidade, enquanto as regras pertencentes à primeira ou terceira categoria são identificadas com o uso de medidas de utilidade.

Piatetsky-Shapiro (1991) propõe princípios intuitivos que uma medida deve satisfazer para avaliar o grau de interesse (IR) de uma regra. Segundo o autor, esse tipo de medida deve associar valores elevados a regras “fortes” e valores baixos a regras “fracas”. Uma “regra forte” pode ser definida como uma descrição de regularidade, com alta confiança para uma grande quantidade de exemplos, uma “regra fraca” representa uma regularidade com alta confiança, para um número reduzido de exemplos.

Dada a regra $LHS \Rightarrow RHS$, Piatetsky-Shapiro (1991) define os princípios:

- 1) $IR = 0$ se $n(LHS \mid RHS) = \frac{n(LHS) \cdot n(RHS)}{N}$. Não há interesse pela regra se o antecedente e o conseqüente são estatisticamente independentes;
- 2) $IR \nearrow$ (aumenta uniformemente) com $n(LHS \mid RHS)$ enquanto outros parâmetros permanecem constantes;
- 3) $IR \searrow$ (diminui uniformemente) com $n(LHS)$ (ou $n(RHS)$) enquanto outros parâmetros permanecem constantes.

Implicações interessantes são conduzidas por estes princípios. Do primeiro princípio tem-se: $IR > 0$ se $n(LHS \mid RHS) > \frac{n(LHS) \cdot n(RHS)}{N}$, indicando que LHS está correlacionado positivamente com RHS . De maneira análoga, a condição $IR < 0$ se $n(LHS \mid RHS) < \frac{n(LHS) \cdot n(RHS)}{N}$, indica que LHS está negativamente correlacionado com RHS . Daí tem-se que IR é similar ao coeficiente de correlação entre LHS e RHS . Dos princípios 2 e 3, tem-se que IR possui um máximo local quando $n(LHS \mid RHS) = n(LHS)$ ou $n(LHS) = n(RHS)$ e um máximo global quando $n(LHS \mid RHS) = n(LHS) = n(RHS)$, dado que a relação $n(LHS \mid RHS) \leq n(LHS) \leq n(RHS)$ é sempre válida.

A aplicação destes princípios pode ser estendida a outras medidas. Por exemplo, no caso das medidas apresentadas nas Seções 4.4.2 e 4.4.4, IR aumenta com a **confiança** e IR diminui com a **confiança esperada**. Ou ainda IR aumenta com a complexidade da regra e IR diminui com o tamanho do conjunto de exemplos.

4.3 Técnicas para Avaliação de Regras

Diversos trabalhos avaliam o conhecimento obtido pelo processo de Mineração de Dados. Nessa seção são descritos alguns desses trabalhos, categorizados e agrupados conforme a predominância da técnica empregada.

4.3.1 Avaliação por Consulta

Essa abordagem permite que o usuário explore o conjunto de regras por meio do uso de uma linguagem de consulta (tipicamente inspirada no SQL - *Structured Query Language*).

Usando esse tipo de facilidade é possível obter subconjuntos de regras contendo determinados elementos e/ou satisfazendo alguma restrição definida pelo usuário, que envolvam, por exemplo, o suporte e a confiança (Jorge 2004).

Trabalhos que utilizam essa abordagem são: Paula (2003), Hipp et al. (2002), Goethals et al. (1998) e Imieliński & Virmani (1998).

4.3.2 Medidas de Avaliação de Conhecimento

Uma variedade de medidas para avaliação de conhecimento tem sido pesquisada com a finalidade de fornecer subsídios ao usuário no entendimento e utilização do conhecimento adquirido. Essas medidas podem ser categorizadas quanto à forma de avaliação e quanto ao objeto de avaliação.

Quanto ao modo de avaliação, as medidas podem ser objetivas ou subjetivas. As medidas objetivas dependem exclusivamente da estrutura dos padrões (regras) e dos dados utilizados no processo de extração de conhecimento. As medidas subjetivas dependem fundamentalmente dos usuários que irão interpretar o conhecimento (Silberschatz & Tuzhilin 1996).

Com relação ao objeto de avaliação, as medidas permitem avaliar o desempenho ou a qualidade de uma regra. O desempenho de uma regra está associado à fidelidade com que representa os dados. Para a avaliação da qualidade dos padrões gerados são utilizados os conceitos de compreensibilidade e grau de interesse. A facilidade de um ser humano inter-

pretar um dado conjunto de regras está relacionada à compreensibilidade deste conjunto, podendo ser estimada, por exemplo, pelo número de regras presentes no conjunto e pelo número de condições em cada regra.

O grau de interesse, por sua vez, é uma avaliação de natureza qualitativa realizada a partir de estimativas da quantidade de conhecimento interessante (inovador, inesperado) presente nas regras. As medidas do grau de interesse são representadas, por exemplo, pelo grau de utilidade das regras para o usuário final do processo de Extração de Conhecimento (Horst 1999b).

Alguns trabalhos que fazem uso das medidas de interesse são: Hussain et al. (2000), Hussain et al. (2000), Liu et al. (2000), Tan & Kumar (2000), Hilderman & Hamilton (1999), Lavrac et al. (1999), Dong & Li (1998), Freitas (1998b), Silberschatz & Tuzhilin (1996) e Silberschatz & Tuzhilin (1995).

4.3.3 Poda de Regras

Uma das abordagens utilizadas para reduzir o número de regras geradas é a poda (*pruning*), que tem o objetivo de eliminar (excluir) regras redundantes ou que não são interessantes para o usuário (Jorge et al. 2002; Baesens et al. 2000).

Alguns trabalhos que fazem uso dessa abordagem são: Jaroszewicz & Simovici (2002), Liu et al. (1999) e Toivonen et al. (1995).

4.3.4 Sumarização e Generalização

A técnica de sumarização objetiva resumir o conhecimento extraído por meio da criação de regras gerais ou conceitos abstratos que sejam facilmente entendidos pelo usuário. As regras gerais retratam conceitos amplos do conhecimento descoberto, enquanto as regras específicas podem ser exploradas posteriormente, em uma análise pormenorizada (Baesens et al. 2000).

A técnica de generalização utiliza taxonomias para transformar regras específicas em conceitos gerais, produzindo conjuntos de regras mais compactos e geralmente mais compreensíveis aos usuários (Srikant & Agrawal 1997).

Alguns trabalhos que fazem uso dessas abordagens são: Domingues (2004), Jorge (2004), Nawarecki et al. (2003), Liu et al. (2000), Liu et al. (2000) e Liu et al. (1999).

4.3.5 Agrupamento de Regras

O objetivo da técnica de agrupamento (*clustering*) é fornecer ao usuário grupos de regras que correspondam a diferentes temas do domínio em questão (Jorge 2004). Essa abordagem procura identificar agrupamentos que representam características específicas de determinados grupos de regras contidas no conjunto original. Com base nesse agrupamento, técnicas de sumarização/generalização podem ser utilizadas para resumir e/ou compactar as regras de cada agrupamento encontrado. Cada um desses agrupamentos pode representar particularidades da base de dados. Alguns trabalhos que fazem uso dessa abordagem são: Jorge (2004) e Toivonen et al. (1995).

4.4 Medidas para Avaliação de Regras de Associação

Na etapa de Pós-processamento do conhecimento, uma das abordagens possíveis é o emprego de medidas para avaliação de regras. Nesta seção são apresentadas algumas destas medidas, definidas a partir dos trabalhos de Rathjens (1996), Wang et al. (1998), Lavrac et al. (1999), Liu et al. (2000) e Adamo (2001).

A organização do trabalho, neste tópico, prevê a divisão das medidas nas categorias – medidas de caráter objetivo e medidas subjetivas – como mostra a Figura 4.1. Adicionalmente é realizado um exemplo de cálculo para as medidas objetivas.

Visando padronizar e sintetizar a representação das medidas objetivas, é feita sua descrição com base nas frequências associadas a cada termo da regra. As frequências são obtidas a partir da tabela de contingência descrita na sequência.

4.4.1 Tabela de Contingência e Frequências

A tabela de contingência para uma regra $LHS \Rightarrow RHS$ representa sua cobertura. Na Tabela 4.2, LHS denota o antecedente da regra, e $\overline{LHS}$ o complemento de LHS ; de forma análoga RHS denota o conseqüente e $\overline{RHS}$ seu complemento. $n(LHS \text{ } RHS)$ denota o

Figura 4.1: Medidas objetivas e subjetivas

número de transações nas quais LHS e RHS são verdadeiros; $n(\overline{LHS} \ RHS)$ denota o número de transações nas quais LHS é falso e RHS é verdadeiro; os demais elementos são representados de modo análogo. N denota o número total de transações.

Tabela 4.2: Tabela de contingência

	RHS	$\overline{RHS}$	
LHS	$n(LHS \ RHS)$	$n(LHS \ \overline{RHS})$	$n(LHS)$
$\overline{LHS}$	$n(\overline{LHS} \ RHS)$	$n(\overline{LHS} \ \overline{RHS})$	$n(\overline{LHS})$
	$n(RHS)$	$n(\overline{RHS})$	N

Em termos genéricos a frequência relativa associada ao conjunto X pode ser calculada como $n(X)/N$ e denotada por $f(X)$. Assim definido, a frequência relativa pode ser utilizada como estimativa de probabilidade. A notação $f(X|Y)$ segue a definição usual de probabilidade, dada por:

$$f(X|Y) = \frac{f(XY)}{f(Y)} \quad (4.1)$$

sendo X e Y subconjuntos das N transações.

Com base na tabela de contingência são calculadas as frequências relativas referentes a cada elemento da tabela. As frequências a serem utilizadas nas medidas de desempenho são apresentadas na Tabela 4.3.

Tabela 4.3: Freqüências obtidas a partir da tabela de contingência

$f(LHS) = \frac{n(LHS)}{N}$	$f(LHS \text{ } RHS) = \frac{n(LHS \text{ } RHS)}{N}$
$f(RHS) = \frac{n(RHS)}{N}$	$f(\overline{LHS} \text{ } \overline{RHS}) = \frac{n(\overline{LHS} \text{ } \overline{RHS})}{N}$
$f(\overline{LHS}) = \frac{n(\overline{LHS})}{N}$	$f(LHS \text{ } \overline{RHS}) = \frac{n(LHS \text{ } \overline{RHS})}{N}$
$f(\overline{RHS}) = \frac{n(\overline{RHS})}{N}$	$f(\overline{LHS} \text{ } RHS) = \frac{n(\overline{LHS} \text{ } RHS)}{N}$

4.4.2 Medidas Objetivas Derivadas da Tabela de Contingência

Medidas objetivas são aquelas que dependem, para sua determinação, exclusivamente da estrutura dos padrões (regras) extraídos e dos dados utilizados no seu processo de extração.

Utilizadas pelo aplicativo **MineSetTM**, em conjunto com as medidas **suporte** e **confiança**, a **confiança esperada** e o **Lift** são definidos (Rathjens 1996):

Confiança esperada: quantifica a incidência de *RHS* no conjunto de dados, independente da ocorrência de *LHS*. A diferença entre **confiança** e **confiança esperada** indica o ganho de poder descritivo ao considerar *LHS*. É representada por:

$$ConfEsp = f(RHS) \quad (4.2)$$

Lift: é a razão, em termos percentuais, entre **confiança** e **confiança esperada**. Quanto maior o **Lift**, maior o grau de inesperabilidade da regra. É representado por:

$$Lift = \frac{f(RHS|LHS)}{f(RHS)} \quad (4.3)$$

Em estudo sobre Regras de Associação para atributos numéricos, Wang et al. (1998) apresentam as medidas J , J_1 e J_2 , que, de maneira geral, expressam a relação entre a generalidade e a capacidade discriminatória de uma regra.

Medida J : o primeiro termo da Equação 4.4, $f(LHS)$, representa a generalidade da regra. O termo entre colchetes avalia o “poder discriminatório” de LHS em relação a RHS , isto é, o ganho de discriminação obtido pela consideração de LHS , quando comparado à regra $true \Rightarrow RHS$. A medida é dada por:

$$J = f(LHS) \cdot \left[f(RHS | LHS) \cdot \log_2 \frac{f(RHS|LHS)}{f(RHS)} + \right. \\ \left. + [1 - f(RHS | LHS)] \cdot \log_2 \frac{1-f(RHS|LHS)}{1-f(RHS)} \right] \quad (4.4)$$

Medida J_1 : derivada da medida J , quando considera-se a capacidade de discriminação nos casos em que $f(RHS | LHS)$ é superior a $f(RHS)$. A medida J_1 é definida por:

$$J_1 = f(LHS) \cdot f(RHS | LHS) \cdot \log_2 \frac{f(RHS | LHS)}{f(RHS)} \quad (4.5)$$

Medida J_2 : a medida J_2 representa o “excedente” (*surplus*) de associação da regra $LHS \Rightarrow RHS$ em relação ao nível de associação especificado pela **confiança mínima** (**conf-min**). A medida J_2 é definida por:

$$J_2 = f(LHS) \cdot [f(RHS | LHS) - \mathbf{conf-min}] \quad (4.6)$$

Lavrac et al. (1999) apresentam um conjunto de dezesseis medidas, dentre as quais **suporte** e **confiança** (já definidas anteriormente). As demais medidas são apresentadas nas Equações 4.7 à 4.20.

Confiança negativa: representa a confiança na não ocorrência de associações, ou seja, dado que o antecedente é falso, o conseqüente também é falso.

$$NegRel = f(\overline{RHS} | \overline{LHS}) = \frac{f(\overline{LHS} \overline{RHS})}{f(\overline{LHS})} \quad (4.7)$$

Sensibilidade: é definida como a frequência condicional de que LHS é verdadeiro dado que RHS é verdadeiro. Pode ser interpretada também como sendo a **confiança** da regra $RHS \Rightarrow LHS$.

$$Sens = f(LHS | RHS) = \frac{f(LHS \text{ } RHS)}{f(RHS)} \quad (4.8)$$

Especificidade: é dada pela frequência condicional de que LHS seja falso dado que RHS é falso.

$$Spec = f(\overline{RHS} | \overline{LHS}) = \frac{f(\overline{LHS} \text{ } \overline{RHS})}{f(\overline{RHS})} \quad (4.9)$$

Cobertura: representa a fração das instâncias cobertas pelo antecedente da regra (LHS), podendo ser considerada como medida de generalidade da regra.

$$Cov = f(LHS) \quad (4.10)$$

Novidade: objetiva identificar o quão inovadora, interessante ou não-usual é uma dada regra. É definida pela comparação entre o valor observado da ocorrência de LHS e RHS e o valor esperado de ocorrência se LHS e RHS fossem independentes.

$$Nov = f(LHS \text{ } RHS) - f(RHS) \cdot f(LHS) \quad (4.11)$$

Cabe destacar que o primeiro termo da medida, $f(LHS \text{ } RHS)$, é o **suporte** da regra, indicando que valores elevados de **suporte** apenas são interessantes quando valores de $f(LHS)$ e/ou $f(RHS)$ são relativamente pequenos. Compara o **suporte** da regra ao **suporte** de cada termo, no caso de independência estatística.

De maneira semelhante à medida **suporte**, a medida **novidade**, tal como definida, é simétrica em relação aos lados da regra, embora essa característica, intuitivamente, não seja uma condição sempre verdadeira. A medida **satisfação**, definida na sequência, se utilizada conjuntamente com a medida **novidade** permite diferenciar as regras na referida simetria (Lavrac et al. 1999).

Satisfação: representa a queda relativa do erro entre $true \Rightarrow RHS$ e $LHS \Rightarrow RHS$.

$$Sat = \frac{f(\overline{RHS}) - f(\overline{RHS} | LHS)}{f(\overline{RHS})} \quad (4.12)$$

Confiança relativa¹: representa o ganho de confiança em relação à regra

¹Optou-se pelo termo **Confiança Relativa** ao invés daquele originalmente utilizado por Lavrac et al.

$\text{true} \Rightarrow RHS$ pela inclusão de LHS .

$$\begin{aligned} RConf &= f(RHS|LHS) - f(RHS) \\ &= \frac{f(LHS \text{ } RHS)}{f(LHS)} - f(RHS) \end{aligned} \quad (4.13)$$

De maneira similar à definição da **confiança relativa**, são definidas as medidas **confiança negativa relativa**, **sensibilidade relativa** e **especificidade relativa**.

Confiança negativa relativa: representa, de modo semelhante à **confiança relativa**, a utilidade de considerar LHS como antecedente de RHS .

$$\begin{aligned} RNegRel &= f(\overline{RHS}|\overline{LHS}) - f(\overline{RHS}) \\ &= \frac{f(\overline{LHS} \text{ } \overline{RHS})}{f(\overline{LHS})} - f(\overline{RHS}) \end{aligned} \quad (4.14)$$

Sensibilidade relativa: representa o ganho de sensibilidade em relação a $LHS \Rightarrow \text{true}$.

$$\begin{aligned} RSens &= f(LHS|RHS) - f(LHS) \\ &= \frac{f(LHS \text{ } RHS)}{f(RHS)} - f(LHS) \end{aligned} \quad (4.15)$$

Especificidade relativa: de forma análoga à **sensibilidade relativa**, representa o ganho de especificidade em relação a $LHS \Rightarrow \text{true}$.

$$\begin{aligned} RSpec &= f(\overline{RHS}|\overline{LHS}) - f(\overline{LHS}) \\ &= \frac{f(\overline{LHS} \text{ } \overline{RHS})}{f(\overline{RHS})} - f(\overline{LHS}) \end{aligned} \quad (4.16)$$

Considerando que as medidas relativas (Equações 4.13 a 4.16) fornecem mais informações sobre a utilidade de uma regra que as medidas absolutas (Equações 4.7 a 4.12), é comum obter-se valores elevados de **confiança relativa** para regras altamente específicas. Para contornar esse inconveniente, as medidas relativas são ponderadas, definindo, desse modo, o novo conjunto de medidas apresentado na sequência (Lavrac et al. 1999).

Confiança relativa ponderada: definida pela ponderação da **confiança relativa** por $f(LHS)$, a medida avalia de maneira balanceada a generalidade e a **confiança relativa**

(1999) – **Precisão Relativa** – uma vez que a definição do termo **precisão** adotada pelos autores é coincidente com a definição de **confiança**, quando trabalha-se com Regras de Associação.

de uma regra.

$$\begin{aligned} WRConf &= f(LHS) \cdot [f(RHS|LHS) - f(RHS)] \\ &= f(LHS) \cdot \left[\frac{f(LHS \text{ } RHS)}{f(LHS)} - f(RHS) \right] \end{aligned} \quad (4.17)$$

Confiança negativa relativa ponderada: de maneira análoga à medida anterior, avalia balanceadamente a generalidade e a **confiança negativa relativa** de uma regra.

$$\begin{aligned} WRNegRel &= f(\overline{LHS}) \cdot [f(\overline{RHS}|\overline{LHS}) - f(\overline{RHS})] \\ &= f(\overline{LHS}) \cdot \left[\frac{f(\overline{LHS} \text{ } \overline{RHS})}{f(\overline{LHS})} - f(\overline{RHS}) \right] \end{aligned} \quad (4.18)$$

Sensibilidade relativa ponderada: definida pela ponderação da sensibilidade relativa por $f(RHS)$, estabelece um balanceamento entre a **confiança negativa relativa** e a generalidade de uma regra.

$$\begin{aligned} WRSens &= f(RHS) \cdot [f(LHS|RHS) - f(LHS)] \\ &= f(RHS) \cdot \left[\frac{f(LHS \text{ } RHS)}{f(RHS)} - f(LHS) \right] \quad \text{bigs} \end{aligned} \quad (4.19)$$

Especificidade relativa ponderada: de maneira similar à medida anterior, estabelece um balanceamento entre a **especificidade relativa** e a generalidade de uma regra.

$$\begin{aligned} WRSpec &= f(\overline{RHS}) \cdot [f(\overline{RHS}|\overline{LHS}) - f(\overline{LHS})] \\ &= f(\overline{RHS}) \cdot \left[\frac{f(\overline{LHS} \text{ } \overline{RHS})}{f(\overline{RHS})} - f(\overline{LHS}) \right] \end{aligned} \quad (4.20)$$

Vale dizer que as medidas relativas ponderadas (Equações 4.17 a 4.20) são equivalentes entre si e numericamente iguais à medida **novidade** (Equação 4.11) (Lavrac et al. 1999).

Adamo (2001) sugere o uso da **convicção** como medida complementar ao **suporte** e à **confiança**, para utilização durante o processo de geração de regras.

Convicção: de modo similar à medida **confiança**, a **convicção** representa o “poder” associativo entre o antecedente e o conseqüente de uma regra.

A **convicção**, entretanto, apresenta maior capacidade distintiva e a capacidade de diferenciar a regra $LHS \Rightarrow RHS$ da regra $RHS \Rightarrow LHS$. A medida é representada por:

$$Conv = \frac{1 - f(RHS)}{1 - \frac{f(LHS \text{ } RHS)}{f(RHS)}} \quad (4.21)$$

Com a finalidade de ilustrar a aplicação das medidas de avaliação descritas, é apresentado na seção seguinte exemplo de cálculo dessas medidas.

4.4.3 Exemplo de Cálculo das Medidas Objetivas

Na realização do exemplo foi considerada amostra de cerca de 25% da base de dados QVU (descrita na Seção 5.3.3) que contém 5 atributos e 30 exemplos (apresentada na Tabela 4.4).

Com o auxílio do aplicativo *MineSetTM*² foram geradas as Regras de Associação, considerando-se os valores *default* (**sup-min** = 2%, **conf-min** = 50%) e no máximo três atributos por regra.

Das 1081 regras geradas pelo aplicativo, duas foram selecionadas para exemplificar o cálculo das medidas. Para a Regra 1, a tabela de contingência é apresentada na Tabela 4.5. As frequências utilizadas no cálculo das medidas são apresentadas na Tabela 4.6. Para a Regra 2, a tabela de contingência e as frequências utilizadas no cálculo das medidas são apresentadas, respectivamente, nas Tabelas 4.7 e 4.8.

Regra 1: ESGOTO_P (... 92,8372] and QV (0,42407 ...] → PerOcpPr (... 51,945]

Interpretação – A regra significa que, para um percentual de até 92,8372% de domicílios atendidos por rede de esgotamento sanitário e com índice de qualidade de vida superior a 0,42407, o percentual de domicílios ocupados pelo proprietário é inferior a 51,945%.

Regra 2: ESGOTP_P (97,68871 ... 98,05199] → RENDCHEF (152,17 ... 182,25]

Interpretação – A Regra 2 significa que para o percentual de domicílios atendidos pela rede de esgotamento sanitário entre 97,68871% e 98,05199%, a renda dos chefes de família está entre 152,17 e 182,25.

Da análise da Tabela 4.9 nota-se que, de modo geral, os valores das medidas para a Regra 1 são ligeiramente melhores em relação aos valores para a Regra 2. No caso dos

²Optou-se pela utilização deste aplicativo para a geração das Regras de Associação, pelos recursos que oferece (discretização automática de atributos, por exemplo) e pela sua disponibilidade no Laboratório de Inteligência Computacional (LABIC), onde foram executados os experimentos.

Tabela 4.4: Amostra da base de dados QVU

ACESSIBI	RENDCHIEF	PerOcpPr	ESGOTO_P	QV
0,789	93,709	78,45	97,17314488	0,11777
0,891	384,23	53,55	98,93617021	-1,2293
0,826	126,643	77,73	93,11740891	0,50799
0,871	96,779	74,09	97,5708502	-0,32863
0,873	214,555	52,92	94,50171821	-0,83267
0,813	141,812	66,84	99,23469388	-0,77391
0,747	222,116	73,11	97,65013055	-0,39171
0,872	125,145	69,17	96,61654135	-0,29879
0,567	294,715	65,99	95,34883721	-0,04481
0,807	87,151	50,80	91,63987138	0,73012
0,901	358,373	50,97	97,08737864	-0,58398
0,839	249,401	75,60	94,8	-0,03436
0,736	117,313	46,69	89,75903614	0,94363
0,860	138,913	57,93	92,5566343	-0,24133
0,819	182,882	74,62	98,18731118	-0,1029
0,798	243,806	78,21	100	-0,33591
0,814	355,959	68,50	94,09448819	0,33306
0,828	153,942	59,82	97,91666667	-0,09933
0,802	159,127	64,29	97,72727273	-0,17741
0,845	121,325	71,53	98,61111111	-0,36991
0,855	288,019	68,09	98,93617021	-0,54511
0,798	82,664	75,55	97,44525547	0,20686
0,709	132,032	44,62	71,20253165	1,01364
0,852	112,659	67,25	98,68995633	-0,0892
0,830	181,623	66,99	99,27184466	-0,66318
0,776	132,616	79,87	98,67986799	0,34015
0,646	150,401	77,65	95,26515152	0,09698
0,758	117,645	60,74	95,55555556	-0,07704
0,816	217,136	72,54	95,07772021	-0,49715
0,852	124,015	64,71	97,23183391	-0,13719

Descrição dos atributos

ACESSIBI	Nível de acessibilidade às atividades urbanas.
RENDCHIEF	Renda dos chefes de família.
PerOcpPr	Percentual de domicílios ocupados pelos proprietários.
ESGOTO_P	Percentual de domicílios atendidos pela rede de coleta de esgoto sanitário.
QV	Nível de qualidade de vida urbana.

valores de Lift e confiança esperada a Regra 2 apresenta valores melhores, indicando um ganho de informação pela consideração do *LHS*. Analisando-se intuitivamente as regras, esta semelhança de representatividade de informações pode ser confirmada, uma vez que ambas as regras, em linhas gerais, relacionam nível de atendimento por infra-estrutura urbana (ESGOTO_P) e nível de renda (RENDCHIEF e PerOcpPr) de forma positiva, isto é, para baixos valores de oferta de infra-estrutura, o nível de renda tende a ser baixo, com a situação inversa também sendo verdadeira.

Tabela 4.5: Tabela de contingência para a Regra 1

	RHS	$\overline{RHS}$	
LHS	3	0	3
$\overline{LHS}$	1	26	27
	4	26	30

LHS	ESGOTO_P (... 92,8372] and QV (0,42407 ...]
RHS	PerOcpPr (... 51,945]

Tabela 4.6: Frequências obtidas a partir tabela de contingência para a Regra 1

$f(LHS) = \frac{n(LHS)}{N} = 0,1$	$f(LHSRHS) = \frac{n(LHSRHS)}{N} = 0,1$
$f(RHS) = \frac{n(RHS)}{N} = 0,13$	$f(\overline{LHS}RHS) = \frac{n(\overline{LHS}RHS)}{N} = 0,87$
$f(\overline{LHS}) = \frac{n(\overline{LHS})}{N} = 0,90$	$f(LHS\overline{RHS}) = \frac{n(LHS\overline{RHS})}{N} = 0$
$f(\overline{RHS}) = \frac{n(\overline{RHS})}{N} = 0,87$	$f(\overline{LHS}\overline{RHS}) = \frac{n(\overline{LHS}\overline{RHS})}{N} = 0,03$

Os valores das medidas confiança relativa ponderada, confiança negativa relativa ponderada, sensibilidade relativa ponderada e especificidade relativa ponderada são idênticos para cada regra individualmente. Esta situação ocorre sempre, atingindo o objetivo estabelecido por Lavrac et al. (1999) em definir uma medida unificada para avaliação de regras. Nesse sentido, no experimento descrito nesta seção, foi considerada apenas a medida confiança relativa ponderada.

Tabela 4.7: Tabela de contingência para a Regra 2

	RHS	$\overline{RHS}$	
LHS	2	0	2
$\overline{LHS}$	1	27	28
	3	27	30

LHS	ESGOTP_P (97,68871 ... 98,05199]
RHS	RENDCHF (152,17 ... 182,25]

Tabela 4.8: Freqüências obtidas a partir tabela de contingência para a Regra 2

$f(LHS) = \frac{n(LHS)}{N} = 0,067$	$f(LHSRHS) = \frac{n(LHSRHS)}{N} = 0,067$
$f(RHS) = \frac{n(RHS)}{N} = 0,1$	$f(\overline{LHS}RHS) = \frac{n(\overline{LHS}RHS)}{N} = 0,9$
$f(\overline{LHS}) = \frac{n(\overline{LHS})}{N} = 0,933$	$f(LHS\overline{RHS}) = \frac{n(LHS\overline{RHS})}{N} = 0$
$f(\overline{RHS}) = \frac{n(\overline{RHS})}{N} = 0,9$	$f(\overline{LHS}\overline{RHS}) = \frac{n(\overline{LHS}\overline{RHS})}{N} = 0,033$

Tabela 4.9: Síntese do exemplo de cálculo das medidas objetivas

Medida	Expressão	Regra 1	Regra 2
Suporte	$Sup = f(LHS RHS)$	10%	6,7%
Confiança	$Conf = f(RHS LHS)$	100%	100%
Confiança Esperada	$ConfEsp = f(RHS)$	13,3%	10%
Lift	$Lift = \frac{f(RHS LHS)}{f(RHS)}$	7,5%	10%
Medida J	$J = f(LHS) \cdot \left[f(RHS LHS) \log_2 \left[\frac{f(RHS LHS)}{f(RHS)} \right] + [1 - f(RHS LHS)] \cdot \log_2 \frac{1 - f(RHS LHS)}{1 - f(RHS)} \right]$	indefinido	indefinido
Medida J ₁	$J_1 = f(LHS) \cdot f(RHS LHS) \cdot \log_2 \frac{f(RHS LHS)}{f(RHS)}$	0,294	0,223
Medida J ₂	$J_2 = f(LHS) \cdot [f(RHS LHS) - \text{conf-min}]$	0,05	0,033
Confiança negativa	$NegRel = f(\overline{RHS} \overline{LHS}) = \frac{f(\overline{LHS} \overline{RHS})}{f(\overline{LHS})}$	0,967	0,965
Sensibilidade	$Sens = f(LHS RHS) = \frac{f(LHS \text{ } RHS)}{f(RHS)}$	0,77	0,67
Especificidade	$Spec = f(\overline{RHS} \overline{LHS}) = \frac{f(\overline{LHS} \overline{RHS})}{f(\overline{RHS})}$	1	1
Cobertura	$Cov = f(LHS)$	0,1	0,067
Novidade	$Nov = f(LHS \text{ } RHS) - f(RHS) \cdot f(LHS)$	0,087	0,06
Satisfação	$Sat = \frac{f(\overline{RHS}) - f(\overline{RHS} LHS)}{f(\overline{RHS})}$	1	1
Confiança relativa	$RConf = f(RHS LHS) - f(RHS) \\ = \frac{f(LHS \text{ } RHS)}{f(LHS)} - f(RHS)$	0,9	0,9
Confiança negativa relativa	$RNegRel = f(\overline{RHS} \overline{LHS}) - f(\overline{RHS}) \\ = \frac{f(\overline{LHS} \overline{RHS})}{f(\overline{LHS})} - f(\overline{RHS})$	0,097	0,065
Sensibilidade relativa	$RSens = f(LHS RHS) - f(LHS) \\ = \frac{f(LHS \text{ } RHS)}{f(RHS)} - f(LHS)$	0,67	0,603
Especificidade relativa	$RSpec = f(\overline{RHS} \overline{LHS}) - f(\overline{LHS}) \\ = \frac{f(\overline{LHS} \overline{RHS})}{f(\overline{RHS})} - f(\overline{LHS})$	0,1	0,067
Confiança relativa ponderada	$WRConf = f(LHS)(f(RHS LHS) - f(RHS)) \\ = f(LHS)\left(\frac{f(LHS \text{ } RHS)}{f(LHS)} - f(RHS)\right)$	0,087	0,06
Confiança negativa relativa ponderada	$WRNegRel = f(\overline{LHS})(f(\overline{RHS} \overline{LHS}) - f(\overline{RHS})) \\ = f(\overline{LHS})\left(\frac{f(\overline{LHS} \overline{RHS})}{f(\overline{LHS})} - f(\overline{RHS})\right)$	0,087	0,06
Sensibilidade relativa ponderada	$WRsens = f(RHS)(f(LHS RHS) - f(LHS)) \\ = f(RHS)\left(\frac{f(LHS \text{ } RHS)}{f(RHS)} - f(LHS)\right)$	0,087	0,06
Especificidade relativa ponderada	$WRSpec = f(\overline{RHS})(f(\overline{RHS} \overline{LHS}) - f(\overline{LHS})) \\ = f(\overline{RHS})\left(\frac{f(\overline{LHS} \overline{RHS})}{f(\overline{RHS})} - f(\overline{LHS})\right)$	0,087	0,06
Convicção	$Conv = \frac{1 - f(RHS)}{1 - \frac{f(LHS \text{ } RHS)}{f(LHS)}}$	3,77	2,72

4.4.4 Medidas Subjetivas

Uma maneira de incluir a subjetividade no processo de análise de regras é considerar o conhecimento prévio do domínio. Assim, Liu et al. (2000) propõem as seguintes medidas:

Identificação de conformidade: permite identificar e classificar regras em conformidade com um *conhecimento impreciso* ou uma *impressão geral*³ fornecidos pelo usuário do domínio. A conformidade é avaliada nos dois lados da regra pela expressão:

$$confm_{i,j} = L_{i,j} \cdot R_{i,j} \quad (4.22)$$

Conseqüente inesperado: permite avaliar se o conseqüente (*RHS*) da regra é inesperado. É representado pela expressão:

$$unExpConseq_{i,j} = \begin{cases} 0 & \text{se } L_{i,j} - R_{i,j} \leq 0, \\ L_{i,j} - R_{i,j} & \text{se } L_{i,j} - R_{i,j} > 0. \end{cases} \quad (4.23)$$

Antecedente inesperado: permite avaliar se o antecedente (*LHS*) da regra é inesperado. É representado na expressão:

$$unExpCond_{i,j} = \begin{cases} 0 & \text{se } R_{i,j} - L_{i,j} \leq 0, \\ R_{i,j} - L_{i,j} & \text{se } R_{i,j} - L_{i,j} > 0. \end{cases} \quad (4.24)$$

Antecedente e conseqüente inesperados: permite avaliar se o antecedente e o conseqüente da regra são inesperados. É expresso:

$$bsUnExp_{i,j} = 1 - \max[confm_{i,j}, unExpConseq_{i,j}, unExpCond_{i,j}] \quad (4.25)$$

na qual:

$L_{i,j}$ é um fator que mede o quanto do *LHS* da regra está em conformidade com o *LHS* do conhecimento fornecido pelo especialista. Idem para $R_{i,j}$ em relação a *RHS*. Os valores destes fatores variam de 0 (para nenhuma conformidade) a 1 (para conformidade completa).

A determinação para impressões gerais de $L_{i,j}$ e $R_{i,j}$ é dada por:

$$\frac{LM_{i,j}}{LN_i} > \frac{RM_{i,j}}{RN_i}$$

³Conhecimento impreciso e impressões gerais são definidos na Seção 4.6.2

$$\text{se} \quad L_{i,j} = \min\left(\frac{LM_{i,j}}{LN_i}, \frac{SM_{i,j}}{SN_i}\right) \quad (4.26)$$

$$\text{então} \quad R_{i,j} = \frac{RM_{i,j}}{RN_i} \quad (4.27)$$

$$\text{senão} \quad R_{i,j} = \min\left(\frac{RM_{i,j}}{RN_i}, \frac{SM_{i,j}}{SN_i}\right) \quad (4.28)$$

$$L_{i,j} = \frac{LM_{i,j}}{LN_i} \quad (4.29)$$

sendo que:

- LN_i e RN_i representam, respectivamente, o número total de itens no antecedente e no conseqüente da regra descoberta;
- $LM_{i,j}$ e $RM_{i,j}$ representam, respectivamente, o número total de itens no antecedente e no conseqüente da regra descoberta que “casam” com o conjunto de itens especificado pelo usuário;
- $SM_{i,j}$ se refere ao número de itens do conjunto especificado pelo usuário que “casam” com itens do conjunto de regras descobertas;
- SN_i se refere ao número de itens do conjunto especificado pelo usuário. Caso $SN_i = 0$, então a razão $\frac{SM_{i,j}}{SN_i} = 1$.

A determinação para conhecimento impreciso de $L_{i,j}$ e $R_{i,j}$ é dada por:

$$L_{i,j} = \min\left(\frac{LM_{i,j}}{LN_i}, \frac{LSM_{i,j}}{LSN_i}\right) \quad (4.30)$$

$$R_{i,j} = \min\left(\frac{RM_{i,j}}{RN_i}, \frac{RVM_{i,j}}{RVN_i}\right) \quad (4.31)$$

sendo que:

- LSN_i e RVN_i representam, respectivamente, o total de elementos no antecedente e no conseqüente do conhecimento impreciso;
- $LSM_{i,j}$ e $RVM_{i,j}$ representam, cada qual, o número de elementos no antecedente e no conseqüente da regra que “casam” com o conhecimento impreciso.

Nesta seção foram apresentadas medidas objetivas e subjetivas que avaliam o conhecimento extraído, sob diversos aspectos, permitindo identificar regras interessantes para o usuário.

4.5 Técnicas de Apoio à Análise de Medidas

Nesta seção são apresentadas a Análise de Pareto – utilizada para priorização de análises e de controle em diversas áreas (controle de estoque, gerenciamento de qualidade, controle de perdas) – e a análise fatorial, técnica estatística de análise multivariada empregada para redução de variáveis. Essas técnicas fundamentaram a metodologia para avaliação de regras proposta no Capítulo 6.

4.5.1 Análise de Pareto

Em 1896 o economista Vilfredo Pareto, em seu livro *Cours d'économie politique*, afirmou que em todos os países e em todas as épocas, a distribuição de renda e bens segue o padrão logarítmico regular expresso na Equação 4.32. Desde sua proposição, essa relação tem se mostrado consideravelmente estável (válida) em estudos empíricos (Fonseca & Ussher 2004).

$$\log(N) = \log(A) + m \cdot \log(x) \quad (4.32)$$

sendo N o número de pessoas que recebem renda acima de x e A e m valores constantes a serem definidos.

Dos trabalhos de Vilfredo Pareto e Max Lorenz (que representou a distribuição de bens por meio de uma curva cumulativa) e também a partir de estudos empíricos, Joseph M. Juran propôs o Princípio de Pareto (também conhecido como regra 80-20 ou lei dos poucos vitais e muitos triviais), generalizando para diversos domínios do conhecimento as descobertas de Pareto e Lorens (Juran 1974).

Tratando de fenômenos naturais o princípio de Pareto expressa que, em várias situações, a maior parte das conseqüências do fenômeno (80%) é determinado por um pequeno número de causas (20%).

Adicionalmente, para os 20% de fatores causa, o princípio também se aplica, ou seja, nesse pequeno universo, 80% das consequências também derivam de 20% das causas e assim por diante. Esse princípio é caso especial de um fenômeno mais amplo: a distribuição de Pareto (Wikipedia 2004).

A distribuição de Pareto (também conhecida como distribuição de Bradford) é uma distribuição estatística do tipo potência. As distribuições desse tipo significam que a probabilidade de se atingir um certo tamanho x é proporcional a x^{-k} , com $k \geq 1$.

Os fenômenos que seguem a distribuição de Pareto, possuem a distribuição de probabilidades caracterizada pela Equação 4.33. Seu comportamento gráfico é representado na Figura 4.2.

$$P(X > x) = \left(\frac{x}{x_{\min}}\right)^{-k} \quad (4.33)$$

com X sendo a variável aleatória representativa do fenômeno, x um número qualquer maior que $x_{\min}$ (menor valor que X pode assumir) e k um parâmetro positivo.

Figura 4.2: Formatos característicos da distribuição de Pareto. Fonte: (Evans et al. 1993)

Diferentemente das distribuições gaussianas, as distribuições do tipo potência não possuem uma escala “típica” e são usualmente referenciadas “livres de escala”. Por exemplo, a distribuição do tamanho de cidades (tipo potência), inclui poucas metrópoles com tamanho muito superior ao tamanho médio das cidades. Por outro lado a distribuição gaussiana – que rege, por exemplo, a distribuição de altura entre seres humanos – não permite que uma pessoa tenha altura muito superior à altura média (Adamic & Huberman 2002).

Dos fenômenos que seguem esse tipo de distribuição, podem ser citados:

- a distribuição de bens individuais antes da criação da classe média;
- a distribuição de bens individuais após a criação da classe média;
- o consumo de água e a geração de esgoto;
- a distribuição de pessoas em cidades;
- os tamanhos de cidades;
- os defeitos em produtos;
- o tamanho de arquivos que trafegam pela internet e usam protocolo TCP;
- a distribuição do número de páginas em sites na *web*.

Baseando-se no Princípio de Pareto pode ser realizada a Análise de Pareto (também conhecida como Método ABC), na qual os casos representativos do fenômeno em estudo são colocados em ordem crescente e é realizada a soma cumulativa de cada contribuição, gerando uma curva. Dessa curva é possível dividir os casos em três grupos (A, B e C). No primeiro estão presentes poucos casos (20%) mas que representam grande parte do fenômeno (80%). No segundo grupo existe uma quantidade maior de casos que representam uma contribuição mediana para o fenômeno. Finalmente no terceiro grupo, uma grande quantidade de casos é pouco representativa do fenômeno.

A partir do Princípio de Pareto foi elaborada a hipótese para filtragem de regras adotada neste trabalho e descrita no Capítulo 5.

4.5.2 Análise Fatorial

Nesta seção é apresentada, em linhas gerais, a técnica de análise fatorial, uma abordagem estatística multivariada, que apóia a identificação de relações estruturais gerais no conjunto de dados. Detalhes sobre a aplicação dessa técnica e interpretação dos resultados são apresentados por Hair et al. (1998). Gorsuch (1974) mostra, detalhadamente, o conjunto de operações estatísticas e matemáticas relacionadas à técnica.

A principal vantagem do uso de técnicas de análise multivariada é a possibilidade de se considerar múltiplas variáveis na busca do entendimento de relações em algum fenômeno,

o que não é possível com a utilização de métodos uni-variados e bi-variados. O aumento do número de variáveis implica também em aumentar a possibilidade de que as variáveis não sejam todas não-correlacionadas e representantes de conceitos distintos. Por outro lado, podem ser identificados grupos de variáveis inter-relacionadas que representam um conceito mais geral contido nos dados.

A análise fatorial é utilizada para analisar estas inter-relações e explicar as variáveis em termos de suas características em comum. O objetivo geral desse tipo de análise é determinar uma maneira de condensar a informação representada pelas variáveis originais utilizando um conjunto menor de variáveis com mínimo de perda de informação.

A análise fatorial difere das técnicas de dependência (regressão múltipla, análise discriminante, análise multivariada de variância, ou correlação canônica), nas quais uma ou mais variáveis são consideradas explicitamente dependentes e as demais variáveis são consideradas independentes.

Na análise fatorial, os fatores são formados para maximizar a descrição de todo o conjunto de variáveis e não para prever uma variável (ou variáveis) dependente. Considerando uma analogia com as técnicas de dependência, cada variável observada (original) é uma variável dependente que é função de algum conjunto de fatores implícito e latente, que por sua vez, são constituídos por todas as outras variáveis. Desse modo, cada variável é predita por todas as outras. Em outra perspectiva, cada fator pode ser visto como uma variável dependente que é função de todo o conjunto de variáveis observadas. Cabe diferenciar a análise fatorial de outro método estatístico multivariado, a clusterização. Enquanto a análise fatorial permite criar grupos de variáveis a partir de casos existentes em um conjunto de dados, a clusterização permite criar grupos de casos.

Da análise fatorial, primeiramente são identificados aspectos implícitos representativos das relações estruturais (denominadas fatores) e depois é determinado o grau com que cada variável é descrita por cada fator. Uma vez determinados os fatores e a descrição de cada variável os objetivos do uso da análise fatorial (sumarização e redução de dados) podem ser atingidos. Na sumarização de dados, a análise fatorial permite identificar aspectos implícitos – mas não facilmente observados – que, quando interpretados e entendidos, descrevem os dados utilizando um número menor de conceitos em comparação com o número de variáveis originais. A redução de dados pode ser alcançada por meio da identificação das variáveis mais representativas ou por meio da criação de um conjunto novo de variáveis para substituir parcial ou completamente o conjunto de variáveis originais.

Um exemplo de aplicação da análise fatorial é apresentado na Tabela 4.10 que re-

apresenta a matriz de correlação para nove indicadores relacionados com elementos de avaliação em uma loja, dentre os quais: número de atendentes, preço e oferta de produtos. Uma pergunta que poderia ser realizada é: “*Há necessidade de medir e analisar esses indicadores separadamente ou é possível definir grupos mais abrangentes de indicadores?*”. Analisando a Tabela 4.10(a), apesar de ser observada a ocorrência de alta correlação entre algumas variáveis, visualmente é difícil identificar algum padrão de agrupamento. A partir dos resultados da análise fatorial é possível agrupar as variáveis conforme apresentado na Tabela 4.10(b), viabilizando a identificação de alguns padrões. Primeiro, quatro variáveis relacionadas às experiências dos compradores dentro da loja foram agrupadas. Depois, três variáveis relacionadas à organização e à disponibilidade de produtos e por fim, duas variáveis relacionadas a qualidade e preço dos produtos foram agrupadas (Hair et al. 1998).

Tabela 4.10: Exemplo do uso de análise fatorial para identificar uma estrutura em um grupo de variáveis

(a) Matriz de correlação original

	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉
V ₁ Preço	1.000								
V ₂ Atendentes	0.427	1.000							
V ₃ Retorno do cliente	0.302	0.771	1.000						
V ₄ Disp. dos produtos	0.470	0.497	0.427	1.000					
V ₅ Qualidade dos produtos	0.765	0.406	0.307	0.472	1.000				
V ₆ Prof. das prateleiras	0.281	0.445	0.423	0.713	0.325	1.000			
V ₇ Larg. das prateleiras	0.354	0.490	0.471	0.719	0.378	0.724	1.000		
V ₈ Serviços na loja	0.242	0.719	0.733	0.428	0.240	0.311	0.435	1.000	
V ₉ Atmosfera na loja	0.372	0.737	0.774	0.479	0.326	0.429	0.466	0.710	1.000

(b) Matriz de correlação das variáveis depois de agrupadas com análise fatorial

	V ₃	V ₈	V ₉	V ₂	V ₆	V ₇	V ₄	V ₁	V ₅
V ₃ Retorno do cliente	1.000								
V ₈ Serviços na loja	0.733	1.000							
V ₉ Atmosfera na loja	0.774	0.710	1.000						
V ₂ Atendentes	0.771	0.719	0.737	1.000					
V ₆ Prof. das prateleiras	0.423	0.311	0.429	0.445	1.000				
V ₇ Larg. das prateleiras	0.471	0.435	0.466	0.490	0.724	1.000			
V ₄ Disp. dos produtos	0.427	0.428	0.479	0.497	0.713	0.719	1.000		
V ₁ Preço	0.302	0.242	0.372	0.427	0.281	0.354	0.470	1.000	
V ₅ Qualidade dos produtos	0.307	0.240	0.326	0.406	0.325	0.378	0.472	0.765	1.000

No contexto deste trabalho, a análise fatorial é utilizada para agrupar medidas objetivas empregadas na avaliação de regras permitindo verificações adicionais na etapa de Pós-processamento de Regras de Associação como apresenta os Capítulos 5 e 6.

4.6 Trabalhos Relacionados com Avaliação de Regras

Nesta seção são apresentados alguns ambientes e aplicativos que apóiam o processo de avaliação do conhecimento. Inicialmente são apresentados alguns requisitos desejáveis em um ambiente de avaliação de regras. Em seguida é descrito brevemente o aplicativo *AIAS*, que por meio do conhecimento prévio do domínio de aplicação e do uso de medidas subjetivas para avaliação de regras, busca identificar as regras mais interessantes a partir de uma coleção de Regras de Associação. Na sequência são descritos os ambientes *DISCOVER* e *RULEE* e o módulo computacional *RULEE_GAR*, desenvolvidos por pesquisadores do *LABIC* para apoiar as etapas do processo de Extração de Conhecimento. Os ambientes são bastante relacionados com o ambiente *ARInE* que é proposto neste trabalho e descrito no Capítulo 6.

4.6.1 Requisitos de um Ambiente para Exploração de Regras

O desenvolvimento de um Ambiente para Exploração de Regras, que auxilie os especialistas do domínio e usuários finais na compreensão e identificação do conhecimento interessante, é de grande relevância para o sucesso do processo de Extração de Conhecimento. Além disso, ambiente desse tipo pode possibilitar que novas pesquisas relacionadas aos métodos para avaliação de regras e ao processo de Mineração de Dados como um todo, possam ser realizadas. O desenvolvimento de um Ambiente para Exploração de Regras vem suprir essas deficiências por disponibilizar o conhecimento e facilitar o acesso aos usuários, além de agregar métodos de avaliação das regras em diferentes aspectos como precisão, compreensibilidade e interessabilidade.

Aspecto importante do processo de Extração de Conhecimento é a dificuldade de análise das regras extraídas já que se encontram em arquivos texto. Pela utilização de ferramentas comerciais mais completas, algumas funcionalidades adicionais podem estar disponíveis, como as ferramentas *3D* para visualização dos modelos. Entretanto, existem poucos métodos implementados para auxiliar a análise, identificação e utilização do conhecimento interessante.

A implementação de um Ambiente para Exploração de Regras implica no desenvolvimento de uma arquitetura que facilite o cálculo e armazenamento do valor das medidas de avaliação das regras, e a incorporação de novos métodos para cálculo de novas medidas. É interessante que cada conjunto de regras disponibilizado apresente um conjunto de medidas tanto objetivas quanto subjetivas, de apoio à análise do usuário. Vemos também que com o acúmulo desses dados sobre a avaliação fica disponível uma nova base de dados valiosa para os pesquisadores do processo de Extração de Conhecimento. A partir desses dados, análises podem ser feitas sobre as medidas ou métodos de avaliação visando, dentre outras possibilidades, identificar medidas mais efetivas para avaliação e estabelecer relacionamentos entre as mesmas. Somando-se a esses dados o *feedback* fornecido pelos usuários durante a consulta às regras disponibilizadas, algumas análises bastante interessantes podem ser feitas. Dentre essas análises utilizando o *feedback* do usuário, destaca-se a possibilidade da análise das funcionalidades mais utilizadas com o objetivo de encontrar medidas de avaliação mais efetivas e verificar a existência de possíveis relações entre medidas objetivas e subjetivas.

Para auxiliar o usuário na compreensão e identificação do conhecimento interessante, algumas características desejáveis para um Ambiente para Exploração de Regras são:

Interatividade e iteratividade – Como citado anteriormente, muitos pesquisadores do processo de Extração de Conhecimento de Bases de Dados têm concentrado suas pesquisas na fase de descoberta do conhecimento. Entretanto, a inferência de padrões é somente parte do problema em se obter conhecimento útil a partir de dados. Além disso, o processo de Extração de Conhecimento de Bases de Dados é descrito de modo a enfatizar a importância da interação com o usuário e da iteração entre suas fases, sendo que a principal iteração do processo é a repetição das fases de Pré-processamento e a extração de padrões após a análise do conhecimento na etapa de Pós-processamento. A qualidade do conhecimento extraído pode ser aprimorada em cada iteração ao reutilizar informações coletadas nas iterações anteriores (Klemettinen et al. 1994; Klemettinen et al. 1996; Klemettinen et al. 1997).

No entanto, a interatividade e iteratividade do processo de Extração de Conhecimento também podem ser realizadas na fase de disponibilização do conhecimento. A iteratividade dessa fase pode ser feita por meio da visualização do conjunto de regras, permitindo ao usuário escolher a cada momento um tipo de filtro ou medida para ordenação das regras e assim identificar as mais interessantes. Caso não seja obtido sucesso, outros filtros podem ser escolhidos. Nesse sentido, uma condição importante para um Ambiente para Exploração de Regras é permitir que o usuá-

rio possa explorar iterativamente e interativamente o conhecimento, especificando a cada instante o foco de interesse desejado.

Medidas de avaliação – Outro requisito importante para um Ambiente para Exploração de Regras é a disponibilização ao usuário de diversos métodos para poda e ordenação dos padrões. Além disso, para seleção de regras, tanto as medidas objetivas quanto as subjetivas devem estar disponíveis. As medidas objetivas podem ser consideradas como filtro inicial para seleção de regras, enquanto que medidas subjetivas podem ser utilizadas para focar o interesse do usuário em um conjunto de regras realmente interessante (Freitas 1998a; Freitas 1998b).

Na disponibilização, as medidas de avaliação podem ser utilizadas para:

Poda do conjunto de regras: o usuário pode escolher alguma medida de avaliação para selecionar um subconjunto das regras apresentadas. Para isso, define um limite de valor para a medida, apresentando somente as regras nas quais o valor da medida é superior ou inferior ao limite estipulado, ou utiliza uma medida que selecione somente regras interessantes segundo critério definido.

Ordenação: o usuário pode escolher medida para ordenar o conjunto de regras apresentado.

Visualização de modelos ou regras – Como em Mineração de Dados busca-se por padrões de comportamento contidos nos dados que não são previamente conhecidos pelos usuários, a compreensão dos modelos encontrados pode ser difícil. Dessa maneira, a compreensão dos modelos é um aspecto que precisa ser bastante pesquisado. Uma das maneiras utilizadas para auxiliar os usuários na compreensão e interpretação do conhecimento é a utilização de ambientes em que os modelos são visualizados pelo usuário de maneira adequada.

Diversas ferramentas de Mineração de Dados possuem módulos de visualização de modelos e de dados bastante completos e úteis. Entretanto, nessas ferramentas a visualização é realizada somente nas estações em que a ferramenta está instalada. Um aspecto adicional de um Ambiente para Exploração de Regras é a possibilidade de visualização do conhecimento por meio de uma interface baseada na Web. Algumas vantagens obtidas pela visualização do conhecimento pela Web são a independência de plataforma, facilidade de uso e, especialmente, a acessibilidade, que permite a visualização do conhecimento pelos usuários do processo, independente de sua localização física (Grinstein & Ward 2002; Ma et al. 2000).

Durante a visualização de um modelo, é importante que o usuário possa interagir com este. Isso significa permitir que o usuário possa selecionar diferentes valores

de atributos e verificar os resultados produzidos. Esta interação possibilita a compreensão pelo usuário de situações de causa-efeito, ou seja, o usuário pode entender quanto cada atributo do conjunto de dados afeta o modelo final (Thearling et al. 2002).

A visualização interativa dos modelos deve permitir que usuários com pouco conhecimento em Aprendizado de Máquina e Estatística sejam capazes de compreender fatores que influenciam as soluções e que também possam visualizar o modelo completo para compreenderem como os exemplos são tratados (Becker et al. 2002).

Integração – Outro modo de auxiliar o usuário na compreensão do conhecimento é fornecer-lhe um contexto, como quando se mostra a relação existente entre o conhecimento extraído e o conjunto de dados a partir do qual foi gerado. Isto é chamado integração (Thearling et al. 2002). Um tipo de integração é a possibilidade de se selecionar parte de um modelo e obter acesso aos dados originais a partir dos quais essa parte do modelo foi gerada.

Ambientes colaborativos – Outro requisito importante em um Ambiente para Exploração de Regras é que este seja um ambiente do tipo colaborativo, permitindo a interação e integração entre os diversos usuários do ambiente (Foster & Gee 2002). No caso de um Ambiente para Exploração de Regras, o primeiro passo para fornecer um suporte eficiente a múltiplos usuários é armazenar as seções dos usuários em um sistema de *logs*.

O sistema de *logs* deve armazenar os trabalhos dos usuários, permitindo a repetição e análise por esses ou outros usuários dos trabalhos realizados (Foster & Gee 2002). Isto permite ganho geral de produtividade entre os usuários do ambiente, além de possibilitar que as informações contidas nos *logs* sejam utilizadas pelos analistas do processo de Extração de Conhecimento para auxiliar na obtenção de melhores resultados em uma próxima iteração do processo.

A utilização da Web como interface para disponibilização de regras facilita também a implementação de ambiente para trabalho cooperativo dos usuários do processo de Extração de Conhecimento. Dessa maneira, os diversos usuários do processo podem analisar e compreender o conhecimento individualmente e fornecer e compartilhar *feedback* para o ambiente por meio de comentários, idéias, dicas ou sugestões (Ma et al. 2000). Além disso, com esse tipo de interface, os diversos usuários finais e especialistas do domínio podem analisar os conjuntos de regras em suas estações de trabalho sem a necessidade de instalar ferramentas específicas de Mineração de Dados.

A partir das informações contidas nos *logs* de utilização dos usuários e de seu *feedback*, análises podem ser feitas, dentre as quais:

- os especialistas do processo de Extração de Conhecimento podem verificar o que é de interesse do usuário, fornecendo novos subsídios para apoiar uma próxima iteração de todo o processo e trazer resultados mais interessantes para o usuário final;
- as funcionalidades mais utilizadas podem ser analisadas com o objetivo de se encontrar medidas de avaliação mais efetivas;
- a existência de alguma relação entre medidas objetivas e subjetivas pode ser verificada a partir das consultas realizadas pelos usuários.

4.6.2 Aplicativo AIAS

O aplicativo *Association Interestingness Analysis System* (AIAS), desenvolvido por Liu et al. (2000), permite identificar regras interessantes, em especial regras inesperadas, a partir de um conjunto de Regras de Associação. Para tanto, o AIAS emprega o conhecimento existente do domínio para ordenar as regras segundo diferentes critérios de interessabilidade e diversos tipos de inesperabilidade.

Para análise das regras é iniciado o procedimento a partir da definição de um arquivo de análise de interessabilidade contendo:

- arquivo de regras (em formato proprietário);
- especificação da taxonomia do conhecimento;
- definição do conhecimento impreciso;
- definição de impressões gerais.

Para a elaboração dos três últimos itens apresentados é necessário conhecimento do especialista. Os itens são detalhados na seqüência:

Taxonomia do conhecimento – O conceito de taxonomia empregado é similar ao apresentado na Seção 3.4.1. Aqui são especificadas categorias e níveis de abstração para os atributos. Adicionalmente, pode-se criar uma hierarquia de classes para os mesmos.

Conhecimento impreciso (CI) – O conhecimento impreciso possibilita informar um conhecimento que o especialista supõe ser verdadeiro. É expresso na forma:

$$CI(< LHS \rightarrow RHS >) \text{ (suporte, confiança)}$$

Impressões gerais (IG) – As impressões gerais possibilitam informar uma relação que o especialista acredita existir entre os atributos especificados. IG é expresso na forma:

$$IG(< a_1, a_2, \dots, a_k >) \text{ (suporte, confiança)}$$

Durante o processo de avaliação, o aplicativo realiza quatro tipos distintos de análises para um dado conhecimento impreciso ou uma impressão geral (definidos na Seção 4.4.4), a saber: **Identificação de Conformidade**, **Conseqüente Inesperado**, **Antecedente Inesperado**, **Antecedente e Conseqüente Inesperados**.

Após a análise das regras utilizando-se as medidas propostas, a ferramenta possibilita a visualização de diferentes aspectos das regras potencialmente interessantes (Liu et al. 2000):

Tipos de interesse – as regras são categorizadas conforme o tipo de interesse, propiciando ao usuário a análise do conhecimento sob diferentes aspectos de interessabilidade.

Graus de interesse – as regras são separadas pelo grau de interesse (regras com maior grau de conformidade ou as mais inesperadas, por exemplo).

Atributos interessantes – a apresentação dos atributos interessantes pode ser considerada mais produtiva que a apresentação da regra como um todo.

Na Figura 4.3 é ilustrada a combinação de aspectos visuais com o uso de medidas. No círculo central é informado o tipo de conhecimento prévio considerado (Conhecimento Impreciso ou Impressões Gerais). Nos quadrantes ao redor do círculo são apresentados os tipos de análise: **Identificação de Conformidade** no quadrante inferior; **Antecedente Inesperado** no quadrante esquerdo; **Conseqüente Inesperado** no quadrante direito e **Antecedente e Conseqüente Inesperado** no quadrante superior.

Cada quadrante é subdividido em faixas de valores das respectivas medidas subjetivas. O número de regras que satisfaz cada faixa de valor é indicado sobre a linha que liga o

Figura 4.3: Tela do aplicativo AIAS

círculo central aos retângulos laterais. Adicionalmente são apresentados os atributos que estão ou não em conformidade com o tipo de conhecimento em análise.

Selecionando-se os elementos na tela, informações detalhadas são apresentadas. Por exemplo, clicando-se sobre o elemento correspondente à **conformidade** na faixa de 0.5 são apresentados o conjunto de regras que satisfazem esta condição e os valores associados de conformidade, suporte e confiança.

O aplicativo possibilita a identificação de subconjuntos de regras com características específicas, permite maior facilidade na exploração interativa do conhecimento e, dentre outras facilidades, proporciona uma visão global do conhecimento extraído. A presença do especialista do domínio no processo de análise é fundamental, tanto no fornecimento do conhecimento prévio (taxonomia, conhecimento impreciso e impressões gerais) como na análise dos resultados apresentados pelo aplicativo.

4.6.3 Ambiente Discover

O Ambiente DISCOVER foi proposto com o objetivo de fornecer um ambiente integrado para apoiar as etapas do processo de Extração de Conhecimento de Dados e Textos (Baranauskas & Batista 2000). Nesse ambiente são utilizados algoritmos de Aprendizado de Máquina implementados pela comunidade científica, bem como módulos com finalidades específicas desenvolvidos pelos pesquisadores do LABIC. Entre as funcionalidades desses módulos estão: pré-processamento de dados, pré-processamento de textos e pós-processamento de conhecimento. As principais funcionalidades do Ambiente DISCOVER são ilustradas na Figura 4.4, a seguir.

Figura 4.4: Funcionalidades do Ambiente DISCOVER

O Ambiente DISCOVER é capaz de acessar os dados na forma atributo-valor ou no formato texto, por meio de métodos para Pré-processamento de dados e de textos, desenvolvidos por pesquisadores do LABIC. Caso a quantidade de dados exceda a capacidade dos algoritmos de extração de padrões, amostras representativas devem ser geradas para serem fornecidas aos algoritmos. Os resultados dos algoritmos de extração de padrões ou modelos podem ser avaliados por meio de diversas medidas, entre elas precisão, compreensibilidade e grau de interesse.

No início, o projeto DISCOVER consistia apenas em um repositório de *scripts*⁴ a serem implementados na linguagem *PERL* (Till 1996) – *Practical Extraction and Report Language* – e posteriormente reutilizados de modo a facilitar a execução de experimentos. A opção por essa linguagem foi determinada em função das vantagens e fatores relacionados:

- permite desenvolver códigos de maneira rápida;
- fornece um mecanismo poderoso para trabalhar com expressões regulares, muito utilizadas nessas implementações;
- permite a manipulação de grande quantidade de dados de maneira eficiente e é bastante flexível com relação aos tipos de dados;
- facilita a escrita de protótipos e *scripts*, bem como funções genéricas e pacotes;
- o CPAN⁵ fornece um excelente repositório de código reutilizável dessa linguagem;
- outras.

O projeto DISCOVER, a priori, consistia em um conjunto de ferramentas independentes que poderiam ser desenvolvidas conforme as necessidades dos usuários, e reunidas em um repositório compartilhado de tal forma que os usuários pudessem combinar ferramentas para executar experimentos de maior complexidade.

Posteriormente, foi proposto criar um ambiente integrado, no qual os *scripts* seriam substituídos por bibliotecas de classes empacotadas como componentes. A partir daí, os programas têm sido desenvolvidos como componentes. O Ambiente DISCOVER possibilita a realização de experimentos por meio da composição desses componentes.

No entanto, para tornar o DISCOVER um ambiente integrado, sua implementação tornou-se mais complexa. Surgiu a necessidade de maior comunicação entre os participantes e de interação entre os diferentes componentes. Com relação à arquitetura do ambiente, em Prati (2003) é proposto *framework* para a integração dos componentes do Ambiente DISCOVER baseado em *software patterns*, no qual os componentes são integrados por meio de linguagem baseada em XML. Além disso, a interface gráfica responsável pela composição dos componentes deveria ser implementada de acordo com a arquitetura proposta para o ambiente. Essa interface é apresentada em Geromini (2002).

⁴Scripts são pequenos programas que realizam tarefas atômicas.

⁵<http://www.cpan.org/>.

Por meio de interfaces bem definidas entre os módulos e procedimentos para conversão de dados e de conhecimento para as sintaxes padrão (principal forma de comunicação entre as funcionalidades) são utilizados no DISCOVER algoritmos já desenvolvidos, testados e consolidados pela comunidade científica, evitando dessa forma sua reimplementação e garantindo maior confiabilidade dos experimentos realizados. Batista (2001) define a sintaxe padrão para representação de dados como uma extensão do formato utilizado como entrada do algoritmo C4.5. No Ambiente DISCOVER, a extensão foi realizada para adequação do formato dos arquivos de dados.

A biblioteca, de nome *Dataset* foi implementada para manipulação do conjunto de dados na sintaxe padrão. Esta biblioteca apresenta métodos para conversão de dados na sintaxe padrão para os formatos de entrada dos algoritmos C4.5, C4.5rules, C5.0, Cubist, CN2, Newid, RT, RETIS, CART, WEKA, SNNS e SVMTorch (Batista 2001; Dosualdo & Rezende 2003a).

No Ambiente DISCOVER foi desenvolvida sintaxe padrão para representação de Regras de Classificação, bem como *scripts* para conversão de regras ou árvores de decisão extraídas por algoritmos de Aprendizado de Máquina para esse formato, além de *scripts* para cálculo da tabela de contingência para cada regra (Prati et al. 2001b; Prati et al. 2001a).

Há *scripts* implementados que convertem a saída dos algoritmos ID3, C4.5, C4.5rules, C5.0/See5, CN2, OC1, *Ripper*, T2 e MC4 para o formato padrão de Regras de Classificação (Prati et al. 2001b). Para o cálculo dos valores da tabela de contingência e geração dos arquivos no formato padrão estendido foram desenvolvidos *scripts* por Prati et al. (2001a).

Foi desenvolvida também sintaxe padrão para representação de Regras de Regressão adequada para englobar o tipo de representação da hipótese induzida por vários algoritmos de regressão (Pugliesi et al. 2003; Dosualdo & Rezende 2003a). Além da definição da sintaxe para Regras de Regressão, foram desenvolvidos *scripts* para conversão dos conjuntos de exemplos da sintaxe padrão do DISCOVER para a sintaxe dos algoritmos M5, RT, Cubist, CART e RETIS e *scripts* para conversão da saída desses algoritmos para a sintaxe padrão de regressão.

De modo geral, o projeto DISCOVER pode ser entendido como um conjunto de métodos aplicados sobre os dados ou sobre o conhecimento extraído a partir dos dados. Dessa maneira, é importante que o projeto DISCOVER ofereça uma base sólida para manipulação de dados e conhecimento. Essa base é composta por sintaxes padrões utilizadas para

representação de dados e conhecimento, e ainda por bibliotecas que oferecem um conjunto de funcionalidades básicas.

Dentre as diversas ferramentas e sistemas implementados para o projeto DISCOVER podem ser citados:

- o sistema **XRULER** (Baranauskas 2001), no qual um conjunto de exemplos é dividido em amostras para treinamento e teste. As amostras de treinamento são submetidas a diferentes indutores e os classificadores obtidos são convertidos para o formato padrão de regras de decisão. Daí, a partir do subconjunto de regras que cobrem os exemplos obtém-se o classificador simbólico final;
- a biblioteca de classes DISCOVER *Object Library* – *DOL*, que implementa a manipulação de dados na sintaxe padrão (Batista & Monard 2003; Batista 2003). Essa biblioteca apresenta métodos para conversão de dados na sintaxe padrão para os formatos de entrada de diversos sistemas de Aprendizado de Máquina. Além disso, a biblioteca fornece vários métodos para pré-processar os dados, entre os quais podem ser citados: cálculo de estatísticas descritivas básicas, embaralhamento, amostragem aleatória e complementar;
- uma biblioteca que transforma os Classificadores Simbólicos (induzidos pelos algoritmos *ID3*, *C4.5*, *C4.5rules*, *C5.0/See5*, *CN2*, *OC1*, *Ripper*, *T2* e *MC4*) para o formato padrão de regras de decisão do DISCOVER (Prati et al. 2001b);
- uma biblioteca que transforma os Regressores Simbólicos (induzidos pelos algoritmos *M5*, *RT*, *Cubist*, *CART* e *RETIS*) para o formato padrão de Regras de Regressão do DISCOVER (Pugliesi et al. 2003; Dosualdo 2003);
- o Ambiente Computacional **SNIFFER** (Batista & Monard 2003; Batista 2003), que gerencia avaliações e comparações experimentais de algoritmos de Aprendizado de Máquina. Entre suas principais funcionalidades podem ser citadas: gerenciamento de sintaxes dos sistemas de aprendizado, aplicação de métodos de *resampling* para estimar a taxa de erro verdadeira, cálculo e comparação de medidas de desempenho;
- a ferramenta **PreTexT** (Matsubara et al. 2003; Martins 2003; Matsubara 2004) realiza o Pré-processamento de uma coleção de documentos utilizando a abordagem *bag-of-words*. Entre as principais funcionalidades da ferramenta, podem ser citadas: transformação de palavras em *stems*; criação da tabela atributo-valor no formato padrão do DISCOVER; redução da dimensionalidade do conjunto de atributos; utilização de indução construtiva na criação de novos atributos na tabela atributo-valor;

- o Ambiente Computacional **DiPER** (Dosualdo 2003; Dosualdo & Rezende 2003b) auxilia na avaliação de um modelo de regressão simbólico. Entre as suas principais funcionalidades, podem ser citadas: conversão dos regressores gerados por diversos algoritmos de regressão para o formato padrão de regras de regressão do DISCOVER; tratamento semântico do conjunto de regras no formato padrão; cálculo de medidas de precisão e compreensibilidade para as regras de regressão; construção de uma tabela de contingência para regras de regressão; realização de teste de hipóteses, comparando regressores induzidos por diferentes algoritmos de regressão sobre um mesmo conjunto de dados.

Além dos trabalhos já implementados descritos aqui, cabe ressaltar que diversos outros trabalhos desenvolvidos ou em desenvolvimento pelos pesquisadores do LABIC com funcionalidades para processamento de dados e textos, Aprendizado de Máquina e processamento de padrões serão integrados ao projeto DISCOVER, entre os quais podem ser citados: Horst (1999a), Nagai (2000), Lee (2000), Imamura (2001), Bernardini (2002), Gonçalves (2002), Geromini (2002), Prati (2003), Bernardini (2003), Martins (2003), Lee (2003), Pugliesi (2004), Matsubara (2004), Ferro (2004).

A partir dos diversos trabalhos e parcerias realizadas pelo grupo de pesquisadores do LABIC, surgiu a necessidade de se desenvolver um ambiente para o Pós-processamento de Conhecimento, que auxilie os usuários do processo de Mineração de Dados na compreensão e identificação de conhecimento interessante, de maneira colaborativa, além de possibilitar o acesso remoto a esse conhecimento. Assim, para atender essas necessidades, foi projetado e implementado o Ambiente RULEE descrito na próxima seção.

4.6.4 Ambiente RuleE

Um Ambiente para Exploração de Regras que viabilize tanto a análise quanto a disponibilização de regras é essencial para a realização do Pós-processamento do conhecimento obtido no processo de Mineração de Dados. Com o intuito de mostrar a viabilidade desse tipo de ambiente, foi desenvolvido por Paula (2003) o Ambiente RULEE (*Rule Exploration Environment*).

O RULEE atende alguns dos requisitos apresentados na Seção 4.6.1 e fornece as funcionalidades necessárias para mostrar a viabilidade de um ambiente de Pós-processamento como o desejado.

O Ambiente RULEE suporta análise de Regras de Classificação, Regressão e Associação. O ambiente tem como entrada arquivos com conjuntos de regras representados utilizando, para cada uma das tarefas de Mineração de Dados mencionadas, a sintaxe padrão definida no Projeto DISCOVER – descrito na Seção 4.6.3, página 86. Por utilizar esses formatos de regras e pela utilização dos *scripts* de conversão dos diversos tipos de regras para a sintaxe padrão, também implementados no DISCOVER, podem ser inseridas no ambiente as saídas de uma grande variedade de algoritmos para Extração de Conhecimento de Dados.

Após a execução dos algoritmos, seus resultados são convertidos para a sintaxe padrão do tipo de problema tratado (classificação, regressão ou associação). A partir das regras expressas na sintaxe padrão e de um conjunto de dados para teste, devem ser utilizados os *scripts* para cálculo dos valores da tabela de contingência elaborados, respectivamente, por Prati et al. (2001a), Dosualdo & Rezende (2003a) e ?. Os resultados desses *scripts* são utilizados como entrada no RULEE, não sendo portanto necessário o cálculo dessas informações internamente pelo ambiente.

O RULEE disponibiliza um conjunto de medidas para avaliação de regras para classificação, regressão e associação, permitindo que diversos tipos de análise sejam realizadas. A partir de um conjunto de regras inserido no ambiente, os valores das diversas medidas são calculados e ficam disponíveis para os usuários. O ambiente provê fácil acesso às regras e aos valores dessas medidas e permite que as medidas sejam utilizadas para a ordenação e seleção de conjuntos de regras.

O Ambiente RULEE apresenta arquitetura para facilitar o armazenamento e a disponibilização dos valores das medidas desenvolvidas permitindo a fácil utilização do conjunto de medidas existentes. O cálculo do valor de novas medidas deve ser realizado com procedimentos desenvolvidos pelos próprios pesquisadores.

O RULEE disponibiliza dois modos diferentes para utilização. O primeiro modo apresenta uma biblioteca de classes que permite acesso às diversas funções implementadas, como inserção, recuperação de regras e de valores de medidas, bem como acesso aos dados disponíveis na base de dados do ambiente. A utilização da biblioteca é realizada por meio de programas escritos na linguagem *PERL*, sendo voltada para usuários analistas do processo de Mineração de Dados que necessitam recuperar informações em seus *scripts* para Pós-processamento ou mesmo para definir alguma função desejada que não esteja implementada no ambiente.

O segundo modo de interação utiliza a interface Web. Essa interface possui a função

principal de facilitar o acesso às informações armazenadas no ambiente e apresenta diversas funções para acesso às regras e aos valores de medidas, além de facilitar a inserção de novos conjuntos de dados. A interface é voltada para usuários não especialistas ou usuários que não necessitam da flexibilidade da utilização da biblioteca de classes. O objetivo principal no uso dessa interface é facilitar o acesso e a apresentação das regras descobertas a partir de uma interface mais intuitiva, na qual os usuários finais ou especialistas do domínio possam acessá-la sem a necessidade da leitura de arquivos texto, facilitando a interação destes usuários com o conhecimento extraído.

Como mencionado anteriormente, o Ambiente RULEE foi implementado buscando atender as especificações de um Ambiente para Exploração de Regras genérico, apresentadas na Seção 4.6.1. Sendo um primeiro trabalho desenvolvido, foi implementada a arquitetura básica do ambiente possibilitando, assim, que características complementares pudessem ser implementadas em trabalhos futuros. Dessa maneira, as principais características ou requisitos implementados no Ambiente RULEE são:

- disponibilização das medidas objetivas para avaliação de regras descritas no *framework* apresentado em Lavrac et al. (1999) e a medida *RI*;
- implementação de consultas para recuperação dos conjuntos de regras e dos respectivos valores de medidas;
- implementação de uma interface simples baseada na Web;
- armazenamento de valores inseridos por usuários como indicativos da interessabilidade, compreensibilidade, utilidade e novidade das regras, além de comentários anexos às regras, conjuntos de regras e consultas.

Considerando que o Ambiente *ARInE* (descrito no Capítulo 6) proposto nesta tese, utiliza-se de componentes do Ambiente RULEE são apresentados, na sequência, alguns detalhes da implementação desse ambiente.

A implementação do Ambiente RULEE é baseada em três elementos:

Biblioteca de classes – Na qual são implementadas as funções disponíveis no ambiente.

Base de dados – Na qual são armazenados o conhecimento no formato de regras, as medidas de avaliação de regras, os dados referentes aos usuários e os dados utilizados na extração e avaliação de conhecimento.

Interface – Baseada na Web visando facilitar o acesso aos conjuntos de regras.

A arquitetura geral do Ambiente RULEE é apresentada na Figura 4.5. No contexto do RULEE, o processo de análise e disponibilização de regras está baseado na idéia de se utilizar um Repositório de Regras e Medidas, entidade principal do ambiente. No Repositório são armazenados os conjuntos de regras, os valores das medidas e as consultas realizadas pelos usuários sobre o repositório de regras e medidas.

Figura 4.5: Arquitetura geral do Ambiente RULEE. Fonte: Paula (2003)

Na Figura 4.5, a disponibilização das regras deve ser iniciada pela inserção no Repositório de Regras dos conjuntos de regras representados na sintaxe padrão. Esta atividade é realizada a partir dos métodos existentes no Módulo de Gerenciamento. Quando disponíveis no Repositório, as regras podem ser acessadas pela interface Web ou pela biblioteca de classes, podendo ser analisadas pelos usuários e utilizadas em algum processo de tomada de decisão. Os elementos da figura que estão no retângulo pontilhado correspondem ao Ambiente RULEE.

A descrição detalhada dos elementos representados na Figura 4.5 é relacionada na sequência:

Módulos de processamento – São os programas escritos para processamento de dados, regras e informações dos usuários. Seguindo o padrão de desenvolvimento do Ambiente Discover, esses programas foram implementados na linguagem *PERL*. Os módulos de processamento encontrados no ambiente são:

- **Módulo de gerenciamento:** responsável pelo gerenciamento das regras disponibilizadas no ambiente. Este módulo recebe requisições da Interface do ambiente, busca as regras correspondentes no Repositório de Regras, ativa os módulos de Pós-processamento ou recupera algum conjunto de dados no Repositório de Dados para então retornar o resultado da requisição para a Interface. Este módulo permite obter ou calcular os valores de medidas para as regras e está implementado como uma biblioteca de classes.
- **Módulos de pós-processamento:** são responsáveis pelo cálculo das medidas de avaliação das regras disponibilizadas. Esses módulos contêm procedimentos para cálculo das medidas e apresentam métodos distintos para tratamento de Regras de Classificação, Associação e Regressão. É importante destacar que no Ambiente RULEE são utilizados os módulos de Pós-processamento implementados no Projeto DISCOVER— por exemplo, o módulo computacional RULEE_GAR proposto por Domingues (2004) e o ambiente *ARInE* proposto neste trabalho.

Interface – Por meio da Interface o usuário pode acessar todas as informações disponibilizadas no Ambiente RULEE. A Interface é baseada na Web a fim de aproveitar algumas das suas vantagens, como a independência de plataforma e, especialmente, a facilidade de utilização pelos usuários do processo, independente de sua localização física.

Repositórios – Correspondem à base de dados desenvolvida utilizando o gerenciador MySql, ou aos arquivos nos quais são armazenadas as informações disponibilizadas. No ambiente são encontrados três tipos de repositórios:

- **Repositório de Informações de Usuário:** conjunto de tabelas MySql nas quais são armazenadas as informações relacionadas aos usuários do ambiente. Algumas informações armazenadas são: cadastro de usuários, comentários e votações em regras.
- **Repositório de Regras e Medidas:** conjunto de tabelas para armazenamento das regras e dos valores das medidas a serem disponibilizadas no ambiente. Um aspecto importante foi a definição do modelo deste repositório, para facilitar o armazenamento e a recuperação das regras, o cadastro e inserção de novas medidas para avaliação e o cálculo e recuperação dos valores das medidas para as regras.
- **Repositório de Dados:** arquivos nos quais são armazenados os dados referentes aos conjuntos de regras disponibilizados. Estes dados podem ser utilizados

para auxiliar o usuário na compreensão das regras, assim como podem ser utilizados por algum módulo de Pós-processamento. Para facilitar a utilização dos métodos de Pós-processamento implementados no Ambiente Discover, estes arquivos de dados devem seguir a sintaxe padrão para representação de dados proposta em Batista (2001). Estes arquivos são armazenados em um subdiretório do ambiente, sendo referenciados em sua base de dados.

Conjuntos de Regras – Representam as regras extraídas e padronizadas para suas respectivas sintaxes padrão. Como citado anteriormente, o Ambiente RULEE trata regras expressas no formato padrão, permitindo que o resultado de uma variedade de algoritmos possa ser manipulado pela utilização das bibliotecas de conversão.

4.6.5 Módulo Computacional RULEE_GAR

Nesta seção é apresentado o módulo computacional RULEE_GAR, que objetiva fornecer funcionalidades para generalizar Regras de Associação e também para analisar as regras generalizadas. A generalização é realizada utilizando o algoritmo *IGART* (*Interactive Generalization of Association Rules using Taxonomy* – Generalização Interativa de Regras de Associação usando Taxonomia) descrito em Domingues (2004). Para prover as funcionalidades de análise de Regras de Associação generalizadas, o módulo utiliza a base de dados e a biblioteca de classes desenvolvidas para o Ambiente de Exploração de Regras RULEE (Paula 2003). O módulo computacional RULEE_GAR foi desenvolvido como Módulo de Pós-processamento do Ambiente RULEE.

O processo de generalização de Regras de Associação de acordo com o algoritmo *IGART* é apresentado na Figura 4.6.

Figura 4.6: Processo para generalização de Regras de Associação no RULEE_GAR.
 Fonte: Domingues (2004)

O processo generaliza apenas um dos lados (*LHS* ou *RHS*) das Regras de Associação. Inicialmente as regras (representadas na sintaxe padrão definida por Melanda & Rezende (2003b)) são agrupadas em subconjuntos que apresentam antecedente ou conseqüente semelhante. Se o processo for utilizado para generalizar o lado esquerdo das regras (*LHS*), os subconjuntos são gerados utilizando conseqüentes (*RHS*) semelhantes. Se o processo for utilizado para generalizar o lado direito das regras (*RHS*), os subconjuntos são gerados utilizando antecedentes (*LHS*) semelhantes. A Figura 4.6 ilustra o processo de generalização do lado esquerdo das regras, por conseqüência, os subconjuntos são agrupados utilizando semelhanças no lado direito das regras. Em seguida são utilizadas as taxonomias para generalizar cada subconjunto. Ao final da generalização, os itens de cada regra são ordenados lexicograficamente e a regra é armazenada em um conjunto de Regras de Associação generalizadas no repositório de regras do RULEE

Concluído esse processo é calculada a tabela de contingência para cada Regra de Associação generalizada, uma vez que a sintaxe padrão proposta por Melanda & Rezende (2003b) inclui a regra acompanhada por sua tabela de contingência. O processo de generalização é finalizado com o cálculo da tabela de contingência. Os trabalhos relacionados com o RULEE_GAR são: Domingues et al. (2003), Domingues & Rezende (2004) e Domingues (2004).

4.7 Considerações Finais

Neste capítulo foi apresentada a variedade de medidas que podem ser empregadas para avaliação do conhecimento extraído por Regras de Associação. Foram apresentados princípios de interessabilidade de regras, que estabelecem diretrizes para identificar se uma regra é interessante ou não. Foram descritas medidas (objetivas e subjetivas) para avaliação de regras e apresentadas as técnicas de Análise de Pareto e análise fatorial, que constituem a base para a metodologia de análise de conhecimento proposta neste trabalho. Também foi apresentado, resumidamente, o aplicativo **AIAS** que incorpora considerações de caráter subjetivo no processo de análise e avaliação do conhecimento extraído. Foi realizada descrição do projeto DISCOVER e do ambiente RULEE, desenvolvidos no LABIC e com os quais o ambiente *ARInE* (descrito no Capítulo 6) está integrado.

Medidas Objetivas como Filtros

5.1 *Considerações Iniciais*

A abordagem utilizada neste trabalho prevê a utilização de medidas objetivas não apenas para ordenar, mas para filtrar regras. Nesse caso a principal dificuldade é estabelecer um valor “corte” para cada medida.

Visando estabelecer um método para definir o valor de “corte”, foram realizados experimentos nos quais se calculou para um conjunto de regras, o valor de algumas das medidas objetivas apresentadas na Seção 4.4.2, página 62. Então foi elaborado um gráfico com os valores destas medidas para cada regra, ordenado respectivamente pelas medidas *confiança* e *suporte*.

O gráfico obtido é apresentado na Figura 5.1. Observa-se que algumas medidas, como *confiança negativa*, *confiança relativa*, *sensibilidade*, *cobertura*, apresentam comportamento semelhante e outras possuem valores alternados em relação às demais *confiança esperada* e *confiança relativa*. Algumas medidas (*confiança* e *satisfação*, por exemplo) apresentam valor máximo em cerca de 50% das regras, outras apresentam valores reduzidos na grande maioria das regras e valores elevados em pequeno número de regras (*suporte* e *lift*, dentre outras).

Figura 5.1: Gráfico que apresenta relação entre medidas para um conjunto genérico de regras.

Da análise realizada estabeleceu-se a hipótese: medidas com valores semelhantes em muitas regras e valores diferenciados em poucas regras podem seguir o Princípio de Pareto (Seção 4.5.1, página 74). Deste princípio é definido método para estabelecer o ponto de “corte” de uma medida. As medidas que não satisfazem essa condição podem ser empregadas como “pré-filtros” ou como medidas complementares.

Nesta seção é realizada análise detalhada do comportamento gráfico de medidas descritas na Seção 4.4.2, página 62, e é verificado seu desempenho como filtro para seleção de regras. Inicialmente é definida hipótese para o comportamento gráfico esperado. São realizados experimentos para analisar a variabilidade do comportamento gráfico em relação a diferentes bases de dados e parâmetros do processo de Mineração de Dados. Finalmente é verificado o desempenho das medidas como filtros para regras, utilizadas de forma isolada ou combinadas a outras medidas.

5.2 Hipótese de Comportamento Gráfico

A partir do gráfico apresentado na Figura 5.1 e com o auxílio de uma planilha eletrônica, foram isolados os gráficos de cada medida e ordenados os valores das medidas de maneira decrescente. A partir dos gráficos resultantes foi possível identificar quatro for-

matos característicos de curvas: **decrécimo inicial acentuado**, **decaimento linear**, **decaimento após patamar** e **sigmoidal**. Esses formatos característicos são ilustrados na Figura 5.2.

Figura 5.2: *Formatos característicos de funções das medidas objetivas*

O formato de curva com **decrécimo inicial acentuado**, Figura 5.2(a), permite identificar que para um pequeno número de regras ocorre uma grande variação no intervalo de valores da medida. Os demais formatos característicos não permitem essa identificação. O formato **sigmoidal**, Figura 5.2(d) possui viabilidade de uso como filtro quando considerado em suas regiões extremas.

Vale lembrar que medidas com comportamento gráfico do tipo **decrécimo inicial acentuado** seguem o Princípio de Pareto, e dessa maneira têm maior capacidade restritiva quando utilizadas como filtros para conjuntos de regras.

5.3 Análise do Comportamento Gráfico das Medidas

Nesta seção são descritos os experimentos realizados para verificação do comportamento gráfico das medidas objetivas descritas na Seção 4.4, página 60. A análise é realizada considerando-se a intensidade restritiva da medida para uso como filtro e, também, a manutenção associada a seu comportamento gráfico para diferentes bases de dados e parâmetros dos algoritmos de Mineração de Dados.

Destaca-se que o objetivo principal destes experimentos é identificar o padrão geral de comportamento gráfico para cada medida objetiva analisada, considerando-se conjuntos de regras com características diversas. Outro aspecto a ser destacado é que a classificação dos gráficos, em conformidade com os quatro formatos característicos, tem forte componente subjetivo, uma vez que os gráficos são produzidos a partir de dados experimentais e não de uma expressão matemática.

5.3.1 Metodologia utilizada para a realização dos experimentos

A metodologia utilizada para realização dos experimentos seguiu as etapas do processo de Mineração de Dados descritas no Capítulo 2 e é ilustrada na Figura 5.3. Foram realizados experimentos com dados artificiais e reais.

Na etapa de obtenção dos dados, como apresentado na Figura 5.3, foram selecionadas 10 bases de dados (8 artificiais e 2 reais). Utilizou-se o aplicativo *Gen* para gerar as bases de dados artificiais (ART A, ART B, ..., ART H). As bases de dados reais BMS-POS e QVU são descritas na Seção 5.3.3. O uso de bases de dados artificiais é justificado pela possibilidade de garantir a existência diversidade de características entre as bases de dados.

Na etapa de Pré-processamento de dados foi utilizado exclusivamente o *script* de conversão (Tid2Set) para disponibilizar os dados no formato do aplicativo *Apriori*.

Na etapa de Extração de Padrões foram utilizados os aplicativos *Apriori* e *MineSetTM* para geração das Regras de Associação para cada base de dados. No Pós-processamento de dados os conjuntos de regras gerados foram convertidos para a sintaxe padrão de Regras de Associação, descrita na Seção 6.3). Para cada arquivo de regras no formato padrão foram calculados os valores das medidas utilizando-se o *script PERL Tc2Medidas*, implementado por Melanda & Rezende (2004b).

Figura 5.3: Metodologia utilizada nos experimentos

Utilizando software específico para criação de gráficos de funções, são criados os gráficos para cada medida objetiva e cada base de dados.

Uma descrição dos aplicativos, das bases de dados e os resultados serão apresentados nas próximas seções.

5.3.2 Aplicativos e Algoritmos Utilizados

Para a realização desses experimentos foram utilizados o aplicativo *Gen*, na geração dos dados artificiais, e os aplicativos *Apriori* e *MineSetTM* para gerar as regras.

Aplicativo Gen

O aplicativo *Gen* produz bases artificiais de dados sintéticos, de acordo com parâmetros definidos pelo usuário. O aplicativo está disponível no endereço <http://www.almaden.ibm.com> e tem sido amplamente utilizado em testes de algoritmos de Regras de Associação.

As funcionalidades do aplicativo *Gen* são agrupadas nas categorias abaixo relacionadas:

Tipos de bases: bases de transações, bases de transações com taxonomias e bases de padrões sequenciais.

Formato do arquivo de saída: formato binário ou ASCII. O formato binário consiste em <Cliente-ID, Transação-ID, Número-de-Itens, Lista-de-Itens.>, sendo cada qual um inteiro de 4 bytes. O formato ASCII consiste em <Cliente-ID, Transação-ID, Item> cada qual com até 10 bytes.

Parâmetros de geração das bases: o aplicativo permite definir características com as quais a base será gerada. Para as bases de transações estão disponíveis as opções: número de transações; número médio de itens por transação; número de itens distintos; número de padrões; tamanho médio máximo dos padrões; correlação entre os padrões; confiança média de uma regra; nome dos arquivos de saída (sem extensão); semente utilizada na geração de números aleatórios.

Além do arquivo de dados, é gerado automaticamente um arquivo de descrição (com extensão *.pat*) contendo a descrição dos dados, uma lista de itens com alta probabilidade de estarem presentes em um *itemset* e uma lista com *itemsets* com grande probabilidade de estar em uma transação.

Apriori

O aplicativo *Apriori* é uma implementação de Borgelt (2003), do tradicional algoritmo de Regras de Associação, de mesmo nome desenvolvido por Agrawal et al. (1993). O aplicativo foi desenvolvido na filosofia GNU, sendo utilizado nos experimentos a versão 4.08. Como características principais desta implementação do Apriori, podem ser destacadas:

Suporte: utiliza definição diferente da original, contudo é possível a utilização da definição original de **Suporte**, adicionando-se a opção **-o** na linha de comando;

Métricas: permite utilizar métricas complementares na geração de regras. As métricas são ativadas pela opção **-e** da linha de comando, seguida pelo número da métrica. Adicionalmente é necessário especificar o valor mínimo da métrica, dado pela opção **-d** seguido do referido valor. As métricas disponíveis são:

1. **Diferença absoluta de confiança à priori:** diferença entre a confiança da regra $TRUE \Rightarrow X$ e a confiança da regra $Y \Rightarrow X$;
2. **Diferença do quociente de confiança em relação à 1:** diferença entre o valor 1 e a razão entre a confiança da regra $TRUE \Rightarrow X$ e a confiança da regra $Y \Rightarrow X$;
3. **Diferença absoluta do ganho de valor em relação à 1:** semelhante à métrica anterior quando a confiança da regra $Y \Rightarrow X$ é menor que a da regra $TRUE \Rightarrow X$.
4. **Diferença de informação à priori:** representa o ganho de informação.
5. **Medida χ^2 normalizada:** bastante conhecida na estatística, é frequentemente empregada para mensurar a diferença entre uma distribuição supostamente independente entre duas variáveis discretas e a distribuição conjunta real, para determinar a intensidade de interdependência das variáveis entre si.

Tipos de saída: permite gerar além das regras apenas os *itemsets* e *Hyperedges* (conjuntos de itens que são fortemente preditivos uns em relação aos outros).

Restrição de atributos: é possível especificar, por meio de um arquivo, uma lista dos itens que estarão presentes nas regras, seja no antecedente, no conseqüente ou em ambos.

Tipos de atributos: trata somente atributos discretos;

Tid2Set: utilitário para conversão de formatos de dados. Transforma o formato <identificador da transação, item> por linha, em uma transação por linha no formato item1, item2, ..., itemN.

MineSetTM

O MineSetTM da Silicon Graphics é um aplicativo que disponibiliza várias ferramentas para Mineração de Dados, dentre as quais uma específica para Regras de Associação. As principais características deste aplicativo, referentes a Regras de Associação, são:

Métricas: além das métricas Suporte e Confiança, gera também as métricas Confiança Esperada e Lift.

Geração de regras: permite controlar o número de atributos por regra: sem limites de atributos, um atributo no antecedente e outro no conseqüente (*default*), outro número de atributos especificado pelo usuário.

Formato de saída: salva as regras geradas em diversos formatos, incluindo *MineSetTM* binário e ASCII.

Visualização de regras: possui módulo específico para visualização de regras, no qual é possível aplicar uma série de filtros sobre os valores das métricas de avaliação e sobre os atributos presentes. Para as regras geradas considerando-se apenas um atributo no antecedente e outro no conseqüente, existe uma ferramenta de visualização gráfica.

Atributos contínuos: discretiza automaticamente atributos contínuos.

5.3.3 Bases de Dados Utilizadas

Neste experimento foram utilizadas dez bases de dados sobre domínios variados e com características distintas. Dessas bases, duas incluem dados reais sobre Qualidade de Vida Urbana (QVU) e comércio eletrônico (BMS-POS) e oito são bases de dados geradas artificialmente.

A base sobre Qualidade de Vida Urbana é composta pelo conjunto de variáveis de qualidade de vida consideradas na formulação do Indicador da Qualidade de Vida Urbana (IQVU) e pelo próprio índice. Os dados são referentes aos setores urbanos da cidade de São Carlos, SP, Brasil. Após realizadas as atividade de Pré-processamento, a base passou a ter um total de 120 exemplos, 19 atributos contínuos e sem valores ausentes.

A base BMS-POS ¹ descreve o conjunto de transações em um site de comércio eletrônico num período de dois meses. Foi utilizada no KDD-cup 2000 (Kohavi et al. 2000). A base pré-processada, inclui 1.657 atributos e 515.597 exemplos. Os atributos estão relacionados a três categorias de informação: *clickstreams*, *order information* e *registration form*.

As bases de dados artificiais foram geradas com o auxílio do aplicativo Gen descrito anteriormente. Para identificar possíveis variações no comportamento das medidas, as bases artificiais (denominadas de ART) foram geradas com as mais variadas características com relação aos números de atributos, de exemplos, de atributos por exemplo e de atributos por *itemset*. Os atributos utilizados são do tipo discreto, dadas as características específicas do problema de Regras de Associação. As características dessas bases são

¹Agradecimento à Blue Martini Software pela contribuição dos dados BMS-POS.

descritas na Tabela 5.1, lembrando-se que para as características não especificadas foram utilizados valores default do aplicativo Gen.

Tabela 5.1: Características das bases de dados artificiais

Nome da base	Nº de atributos	Nº de exemplos	Nº médio de atributos por exemplo	Nº médio de atributos por <i>itemset</i>
ART A	1000	1000	25	4
ART B	10	25000	5	14
ART C	10	25000	9	14
ART D	100	25000	80	8
ART E	1000	1000	100	20
ART F	19	25000	25	15
ART G	30	25000	25	15
ART H	30	2500	25	15

5.3.4 Extração de Regras

Com o auxílio do aplicativo *MineSetTM*, foram geradas Regras de Associação, considerando-se os valores *default* (suporte mínimo de 2%, confiança mínima de 50%) e no máximo três atributos por regra, para a base QVU.

A base BMS-POS e as bases artificiais foram mineradas com o auxílio do aplicativo *Apriori*. Foram utilizadas as definições originais de **Suporte** e **Confiança**, variando-se os valores mínimos dessas medidas. Para os demais parâmetros foram utilizados valores *default*. Para verificar a influência de outro parâmetro, a base ART D foi minerada variando-se o número máximo de atributos por regra. Foram considerados os limites de 2, 3 e 4 atributos (o valor *default* para esse parâmetro é 5). Uma síntese dos parâmetros utilizados para a geração de regras é apresentada na Tabela 5.2.

Os aplicativos *MineSetTM* e *Apriori* foram utilizados em bases distintas pela maior facilidade no formato de entrada de dados. Cabe lembrar que por tratar-se de uma tarefa descritiva de Mineração de Dados, o conjunto de regras gerado é independente dos algoritmos e dos aplicativos empregados.

Tabela 5.2: *Parâmetros utilizados nos algoritmos para Regras de Associação*

Nome da base	Suporte Mínimo (%)	Confiança Mínima (%)	Nº de atributos por regra
ART A	6	25	5
ART B	6	25	5
ART C	6	25	5
ART D ₂	6	25	2
ART D ₃	6	25	3
ART D ₄	6	25	4
ART E	6	25	5
ART F	6	25	5
ART G	6	25	5
ART H	6	25	5
QVU	2	50	3
BMS-POS	2	25	5

5.3.5 Análise dos Resultados

O número de regras obtidos a partir dos algoritmos de extração de regras utilizando os parâmetros apresentados na Tabela 5.2, é apresentado na Tabela 5.3.

Tabela 5.3: *Número de regras geradas com a aplicação dos algoritmos de Regras de Associação*

Nome da base de dados	Nº de regras geradas	Nome da base de dados	Nº de regras geradas
ART A	44	ART E	20.000
ART B	2.000	ART F	80.000
ART C	2.500	ART G	420.000
ART D ₂	9.000	ART H	430.000
ART D ₃	440.000	QVU	25.000
ART D ₄	600.000	BMS-POS	65.843

Para cada regra gerada foram calculados os valores das medidas descritas na Seção 4.4, página 60. A seguir, para cada base de dados, cada medida foi ordenada individualmente em sentido decrescente e os valores plotados em gráficos, com os valores das medidas nas ordenadas e o número sequencial das regras nas abscissas. Desse procedimento foi gerado um total de 156 gráficos, cada qual para cada medida e cada execução do algoritmo para Mineração de Dados em um conjunto de dados (13 medidas e 12 execuções dos algoritmos para Mineração de Dados). Alguns desses gráficos serão apresentados e comentados na próxima seção e uma síntese dos resultados obtidos será apresentada na Seção 5.4.

Em Melanda & Rezende (2004b) é realizada a descrição pormenorizada do conjunto de gráficos.

Observando-se as Figuras de 5.4 a 5.16 é possível identificar classes de comportamento gráfico para as funções – decréscimo após um patamar constante, decréscimo linear, decréscimo inicial acentuado. Como já mencionado, para uso como filtro são mais adequadas as medidas que a função apresenta decréscimo inicial acentuado.

Analisando os gráficos da medida **confiança** (Figura 5.4), pode-se observar que, de modo geral, a função da medida decai mais acentuadamente quando da ocorrência de um patamar relativamente constante (Figuras 5.4(b) a 5.4(e)). Essa condição não é verificada unicamente nas Figuras 5.4(a) e 5.4(f), que apresentam decaimento próximo do linear. Na Figura 5.4(e), observa-se a ocorrência de um patamar inicial constante com valor máximo da medida até cerca de 50% do número total de regras e decréscimo mais acentuado após este valor (ou seja, 50% das regras possuem o valor máximo da medida nesse conjunto de dados). Medidas que têm funções com tal comportamento, em geral são pouco úteis para filtrar regras, contudo, podem ser empregadas em outros tipos de análises ou como um “pré-filtro”. Pode-se observar que a forma dos gráficos das Figuras 5.4(a) e 5.4(e) distingue-se consideravelmente das demais. A instabilidade na forma gráfica da medida dificulta sua utilização em um sistema automatizado e portanto requer a supervisão do usuário. No que se refere à variação de parâmetros para uma mesma base de dados, a medida não apresentou variações, como pode-se observar nas Figuras 5.4(c) e 5.4(d).

Figura 5.4: *Medida Confiança*

Nos gráficos da medida **convicção** (Figura 5.5), é possível identificar um padrão com decréscimo inicial acentuado dos valores da medida, Figuras 5.5(a), 5.5(b), 5.5(e) e 5.5(f).

Figura 5.4: *Medida Confiança (continuação).*

A Figura 5.5(e) mostra que, para as 2500 primeiras regras (cerca de 10% do total de regras), o valor da medida **convicção** varia em cerca de 70% (de 1,00 para aproximadamente de 0,30) e, para as demais regras, o valor de **convicção** varia de 0,30 para 0,10. Essa condição possibilita a identificação de um número reduzido de regras e portanto a medida é considerada adequada para utilização como filtro.

Com relação à estabilidade na forma das medidas, pode-se observar que o conjunto de bases apresenta, em termos gerais, comportamento semelhante. Contudo, as bases ART D₂ e ART D₃, se diferenciam das demais. ART D₂ (Figura 5.5(c)) apresenta decréscimo após patamar relativamente constante e ART D₃ (Figura 5.5(d)) apresenta decaimento linear. Considerando a relativa estabilidade da medida, pode-se utilizá-la como filtro, desde que haja monitoramento do usuário. Quanto à variabilidade da medida em relação aos parâmetros de Mineração de Dados, na Tabela 5.2, para uma mesma base de dados

Figura 5.5: *Medida Convicção*

pode-se identificar certa discrepância no comportamento gráfico da medida (Figuras 5.5(c) e 5.5(d)).

Na Figura 5.6 são apresentados os gráficos da medida **confiança negativa**. Pode-se visualizar uma variedade de formas, que, no caso das curvas nas Figuras 5.6(a) e 5.6(e), podem ser considerados como apresentado decaimento do tipo linear. O comportamento da medida para as bases de dados representadas nas Figuras 5.6(c) e 5.6(d) mostra decaimento inicial acentuado. Comportamento peculiar pode ser observado nas Figuras 5.6(b), 5.6(e) e 5.6(f), que apresentam curva com pequeno decaimento inicial acentuado, com eventuais inflexões, seguido por decaimento suave.

Figura 5.6: *Medida Confiança Negativa*

Na Figura 5.6(e) pode-se observar decréscimo no valor da **confiança negativa** com variação de 1,00 para 0,70 (39%) para as primeiras 11.200 regras (44,8%). Na sequência ocorre variação de 76,1% para 55,2% de regras, indicando decaimento aproximadamente linear.

Considerando a variedade de formas para as diferentes bases de dados, e que a capacidade restritiva que se aproxima do tipo da linear, é aconselhável que a medida **confiança**

(e) QVU

(f) BMS-POS

Figura 5.6: Medida Confiança Negativa (continuação).

negativa seja empregada como um filtro complementar e desde que haja supervisão do usuário.

Os gráficos plotados para análise da medida novidade estão relacionados na Figura 5.7. As curvas representadas nas Figuras 5.7(c), 5.7(d), 5.7(e) e 5.7(f) apresentam em conjunto decaimento inicial acentuado – comportamento desejado para utilização da medida como filtro – que tem em 5.7(e) e 5.7(f) sua forma mais acentuada. A curva da Figura 5.7(a) apresenta decréscimo acentuado a partir de um patamar relativamente constante. A variação inicial ocorre apenas para as duas primeiras regras.

(a) ART A

(b) ART C

Figura 5.7: Medida Novidade

Figura 5.7: Medida Novidade (continuação).

Na Figura 5.7(b), o gráfico apresenta decréscimo inicial acentuado e subsequente decaimento linear, aqui descrito como do tipo sigmoidal.

Também nesse caso, considerando-se a variabilidade no comportamento gráfico da medida para as diferentes bases de dados utilizadas, é aconselhável que a medida **novidade** seja empregada com a supervisão do usuário.

Analisando-se os gráficos da medida **confiança relativa** (Figura 5.8), em termos gerais, há três tipos de comportamento gráficos distintos. As Figuras 5.8(a) e 5.8(e) apresentam curva com decaimento próximo do tipo linear. Esse comportamento qualifica a aplicação da medida como um filtro complementar.

O gráfico da Figura 5.8(f) tem formato do tipo sigmoidal. Funções dessa forma são interessantes para realização de análises preliminares considerando a variação que ocorre

Figura 5.8: *Medida Confiança Relativa*

nas extremidades da curva. Contudo, a identificação das regiões de análise deve ser supervisionada pelo usuário.

Os gráficos das Figuras 5.8(b), 5.8(c) e 5.8(d) apresentam em conjunto decaimento inicial acentuado, podendo ser empregados como filtro.

Da análise da Figura 5.8(e) pode-se observar que, com a elevação do valor limite inferior de 0 para 0,25 (variação de 25%) ocorre a redução do número de regras para aproximadamente 16.000 (há 36% de redução). Esse comportamento indica baixa capacidade restritiva da medida para a base QVU, quando comparada com o grau de restrição da medida para a base ART D₂. Nessa base, para o mesmo percentual de alteração no limite inferior (25%), ocorre redução de 77% no número de regras.

Como nos casos anteriores, considerando-se a variabilidade de formas obtidas para as diferentes bases de dados utilizadas, é aconselhável que o emprego da medida **confiança relativa** seja acompanhado por um usuário.

Nos gráficos plotados para medida **confiança negativa relativa** (Figura 5.9), de modo geral, a função apresenta decaimento inicial acentuado. Essa condição não se verifica unicamente no caso da curva da Figura 5.9(a), que apresenta formato próximo do tipo sigmoidal.

Figura 5.9: Medida Confiança Negativa Relativa

No gráfico da Figura 5.9(e) para as primeiras 1250 regras ocorre variação no valor da medida **confiança negativa relativa** de 0,78 para 0,04 (94,9%) e variação de 0,04 para 0,000 (5,1%) para as demais regras, indicando a potencialidade restritiva da medida, para uso como filtro de regras.

Com relação à estabilidade no comportamento gráfico, apenas a curva da Figura 5.9(a) apresenta características ligeiramente distintas (possivelmente em função da quantidade reduzida de regras – apenas 44). Desse modo a medida **confiança negativa relativa** pode ser adequada para uso como filtro sem a supervisão do usuário nas bases de dados estudadas.

Figura 5.9: Medida Confiança Negativa Relativa (continuação).

Na Figura 5.10 são apresentados os gráficos da medida **sensibilidade relativa**. Na Figura 5.10(a) ocorre decréscimo aproximadamente linear nos valores da medida. Nas demais figuras (da Figura 5.10(b) à Figura 5.10(f)) ocorre decaimento inicial acentuado nos valores da medida.

A análise da Figura 5.10(e) permite identificar o decaimento nos valores da medida da ordem de 75% (de 0,98 para 0,25) para as primeiras 2.500 regras, e de cerca de 25% para os 90% restantes das regras, indicando a capacidade restritiva do filtro.

Com relação ao formato gráfico e à sua estabilidade, a medida **sensibilidade relativa** pode ser considerada adequada para uso como um filtro para regras sem a supervisão de usuários.

Figura 5.10: *Medida Sensibilidade Relativa*

Nos gráficos da medida especificidade relativa o decréscimo inicial acentuado é identificado como padrão. Na Figura 5.11(b) apesar do formato sigmoidal da curva, há uma queda inicial bastante acentuada, que indica a adequação da medida para utilização como filtro.

(a) ART A

(b) ART C

(c) ART D₂ (d) ART D₃

(e) QVU

(f) BMS-POS

Figura 5.11: Medida Especificidade Relativa

Da análise da Figura 5.11(e) verifica-se que para as primeiras 1.250 regras (5,0% do total), a medida especificidade relativa varia de 0,77 para 0,05 (variação de 93,5%) e de 0,05 para 0,00 para os outros 96% das regras, refletindo a capacidade de utilização como filtro da medida.

O comportamento gráfico dessa medida pode ser tido como estável considerando-se que apenas para a base de dados ART A (Figura 5.11(a)) ocorre variação no formato (mais uma vez, essa variação pode estar associada ao pequeno número de regras existentes). Nesse sentido a medida pode ser considerada adequada para uso como filtro sem supervisão do usuário.

Observando-se os gráficos da medida **satisfação** (Figura 5.12) identifica-se que para as Figuras 5.12(b), 5.12(c) e 5.12(d) ocorre decaimento inicial acentuado da curva. A Figura 5.12(a) mostra decaimento aproximadamente linear e a Figura 5.12(e) decaimento nos valores após um patamar relativamente constante. Na Figura 5.12(f) o comportamento da função se aproxima do tipo sigmoidal, com decréscimo inicial mais acentuado.

Figura 5.12: *Medida Satisfação*

A análise comparativa das Figuras 5.12(c) e 5.12(e) mostra a grande variação da capacidade de restrição da medida como filtro. Para a Figura 5.12(c) o valor da **satisfação** decresce de 80% (1,0 para 0,2), para as primeiras 2.500 regras, enquanto que na

Figura 5.12: *Medida Satisfação (continuação).*

Figura 5.12(e) o valor permanece máximo e constante para as primeiras 12.500 regras (cerca de 50% do total de regras). A medida **satisfação**, considerando-se a grande variabilidade apresentada pode ser empregada como pré-filtro ou filtro principal, sob a supervisão do usuário.

A análise dos gráficos da medida **sensibilidade** (Figura 5.13) mostra variabilidade significativa no comportamento gráfico da função. Nas Figuras 5.13(a), 5.13(b), 5.13(e) e 5.13(f) ocorre a situação de decréscimo inicial acentuado. Para a base de dados representada na Figura 5.13(c) o formato da função mostra decaimento após patamar constante. Na Figura 5.13(d) o formato da função é aproximadamente linear.

Figura 5.13: *Medida Sensibilidade*

Figura 5.13: Medida Sensibilidade (continuação).

No gráfico da Figura 5.13(e), é observado que para o intervalo de 1,00 a 0,26 do valor da medida **sensibilidade** (variação de cerca de 74%), aproximadamente 2.500 regras estão contidas no intervalo. Enquanto que para os demais 90% das regras, os valores da medida variam de 0,26 a 0,00, o que representa cerca de 26% do intervalo de valores.

Considerando o comportamento gráfico obtido para as diferentes bases de dados, a medida **satisfação** pode ser empregada como um filtro secundário ou um pré-filtro conforme orientação do usuário.

Os gráficos da medida **especificidade** são apresentados na Figura 5.14. De modo geral a curva apresenta decréscimo acentuado após patamar relativamente constante. Esse comportamento ocorre nos gráficos das Figuras 5.14(a), 5.14(b), 5.14(e) e 5.14(f). Nas Figuras 5.14(c) e 5.14(d) o formato da função é aproximadamente linear.

Figura 5.14: *Medida Especificidade*

A avaliação dos gráficos apresentados nas Figuras 5.14(c) e 5.14(e) indica a grande variação da capacidade de restrição da medida como filtro. Para a Figura 5.14(c) o valor da medida varia de 77% para 50% das regras, enquanto na Figura 5.14(e) o valor da especificidade permanece máximo e constante para cerca de 50% das regras.

Considerando a predominância do formato com decaimento acentuado após patamar, a inexistência do formato com decréscimo inicial acentuado e a ocorrência de apenas dois gráficos com decaimento próximo do modo linear, a medida **especificidade** pode ser empregada como um “pré-filtro” automatizado, sem a supervisão de um usuário.

Nos gráficos da medida **suporte** é possível identificar o padrão com decaimento inicial acentuado (Figuras 5.15(a), 5.15(b), 5.15(e) e 5.15(f)). A oscilação que ocorre na curva da Figura 5.15(b) é irrelevante para a classificação realizada, uma vez que se está buscando um comportamento geral para a medida. Nas Figuras 5.15(c) e 5.15(d) o decaimento da função é próximo do modo linear, condição que determina uma redução menor no número de regras.

Figura 5.15: *Medida Suporte*

A análise da Figura 5.15(e), mostra que para as primeiras 1.250 regras (valor de cerca de 5% do total de regras), o valor da medida suporte varia de 0,76 para 0,06 – variação de 92%. Para as demais 23.750 regras o valor do suporte varia de 0,06 para 0,02.

Figura 5.15: Medida Suporte (continuação).

A partir dos resultados obtidos, a medida **suporte** pode ser considerada para uso como filtro principal sem supervisão do usuário.

Na Figura 5.16 encontram-se os gráficos da medida J_1 para as diferentes bases de dados. Para as bases representadas nas Figuras 5.16(a) e 5.16(b) o comportamento da função se aproxima do modelo sigmoidal, com a curva da Figura 5.16(a) tendendo para o modo linear. As curvas das Figuras 5.16(c), 5.16(d), 5.16(e) e 5.16(f) apresentam decaimento inicial acentuado, formato que é predominante para esta medida.

Figura 5.16: Medida J_1

Ao observar detalhadamente a Figura 5.16(e), identifica-se variação de 69% (de 0,3 para 0,09) para as primeiras 1.250 regras, o que demonstra a capacidade de restrição que

(c) ART D₂ (d) ART D₃

(e) QVU

(f) BMS-POS

Figura 5.16: Medida J_1 (continuação).

a medida J_1 oferece e a adequação desta para uso como filtro principal ou secundário, com a supervisão do usuário.

5.3.6 Análises Complementares Realizadas para Validar a Análise Manual

Nesta seção são realizadas duas análises complementares dos gráficos apresentados na Seção 5.3.5. A partir dessas análises buscou-se identificar de modo automatizado o formato dos gráficos ou reconhecer algum padrão característico de comportamento, para validar as análises realizadas na seção anterior. Objetivo secundário dessas análises é verificar a possibilidade de automatização do processo de categorização das medidas.

Utilizando Técnicas de Cluster

Uma abordagem alternativa adotada na análise do comportamento gráfico das funções utiliza a técnica de *clustering* sobre as imagens dos gráficos. Esse procedimento foi adotado para validar a análise realizada de forma manual na Seção 5.3.5. Foram utilizados os conceitos de similaridade de imagens e de cluster. Optou-se por analisar os gráficos por meio de similaridade de imagens, ao invés da análise por semelhança de equações, uma vez que se deseja obter uma similaridade genérica, relacionada principalmente ao formato de decaimento da curva.

Os arquivos de imagens dos gráficos foram convertidos para arquivos de imagens no formato PBM ASCII. A partir destes arquivos, foram calculados os descritores: área sob a curva (AC), comprimento (CC) e momento da curva (MC). A seguir, esses dados foram processados utilizando o algoritmo *k-means* (MacQueen 1967), variando-se o número de clusters de 3 a 5. Na Tabela 5.4 é apresentada a síntese com os resultados das execuções do algoritmo que descreve o número de clusters especificados, os atributos considerados (AC, CC e MC) e o número de gráficos em cada cluster.

Para cada execução do algoritmo de *clustering* os arquivos de imagem foram agrupados e verificados visualmente com relação à similaridade das curvas. A divisão do conjunto de imagens em três clusters e considerando as variáveis AC e MC foi a solução que apresentou melhores resultados, conforme avaliação visual realizada (segunda linha da Tabela 5.4).

Tabela 5.4: Distribuição do número de gráficos segundo os clusters

Nº de clusters	Atributos considerados	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
3	AC, CC, MC	96	48	25		
3	AC, MC	92	52	25		
4	AC, CC, MC	54	16	90	9	
4	AC, MC	20	90	54	5	
5	AC, CC, MC	47	82	5	20	15
5	AC, MC	47	16	81	19	6

Utilizando Técnicas de Classificação

Para identificar a existência de alguma relação entre a forma gráfica da medida, as características da base de dados, os parâmetros do processo de Mineração de Dados e os descritores gráficos, foi realizada verificação da possibilidade de classificação automática dos gráficos utilizando o algoritmo *C4.5* (Quinlan 1993). Os dados utilizados para execução

do algoritmo de classificação foram obtidos a partir das Tabelas de 5.1 a 5.4 e da análise dos gráficos das medidas realizada na Seção 5.3.5 (na Tabela 5.5, próxima seção). Os atributos considerados são apresentados na Figura 5.17, que representa o arquivo *.names* utilizado na execução do algoritmo de classificação.

```
L,P,I,S.
medida: especificidade,confianca_relativa_negativa,novidade,
        especificidade_relativa,satisfacao,conviccao,
        sensibilidade_relativa,confianca_negativa,confianca_relativa,
        sensibilidade,confianca,J1,suporte.
base: D2,BMS-POS,D4,B,I,A,E,H,C,D3,J,QVU.
atributos: continuous.
exemplos: continuous.
atributos_por_exemplo: continuous.
suporte: continuous.
confianca: continuous.
atributos_por_regra: continuous.
regras: continuous.
Area: continuous.
Perimetro: continuous.
momento: continuous.
sum: continuous.
```

Figura 5.17: Descrição dos atributos considerados para execução do algoritmo de classificação

A árvore de decisão resultante do processo de classificação é apresentada na Figura 5.18. Observa-se que o principal atributo discriminador para o formato gráfico é o **momento da curva**: para valores de momento acima de $1.92e+7$ a curva é do tipo P (decréscimo após patamar constante) e para valores desse atributo inferiores a $9.83e+6$ a curva é do tipo I (decréscimo inicial acentuado). Para valores de momento intermediários a esses limites é necessário considerar outros atributos, em especial a medida. Nesse sentido, pode ser possível estabelecer um método automatizado para identificar a forma do gráfico a partir de descritores de imagens e características do processo de Mineração de Dados.

5.4 Síntese da Análise do Comportamento Gráfico

Esta seção resume as análises dos gráficos das medidas objetivas realizadas (Seção 5.3.5 - Figuras 5.19, 5.20 e 5.21, Tabela 5.5; Seção 5.3.6 - Tabela 5.6). Para cada medida e cada base de dados é indicada a classificação atribuída na interpretação dos gráficos e também a classe mais freqüente (moda) para cada medida.

Aspecto que pode ser verificado neste experimento é a freqüência de ocorrência para

```

C4.5 [release 8] decision tree generator
-----
Read 156 cases (13 attributes) from .data

Simplified Decision Tree:

momento > 1.92e+07 : P (29.2/12.5)
momento <= 1.92e+07 :
|
|   momento <= 9.83e+06 : I (75.5/14.3)
|   momento > 9.83e+06 :
|   |
|   |   medida = especificidade: P (1.3/1.0)
|   |   medida = confianca_relativa_negativa: I (5.0/2.3)
|   |   medida = novidade: S (6.0/2.3)
|   |   medida = especificidade_relativa: S (3.0/1.1)
|   |   medida = satisfacao: S (4.0/2.2)
|   |   medida = conviccao: I (1.0/0.8)
|   |   medida = sensibilidade_relativa: L (2.0/1.8)
|   |   medida = confianca_negativa: S (5.0/2.3)
|   |   medida = confianca: L (1.0/0.8)
|   |   medida = J1: S (7.0/3.4)
|   |   medida = suporte: I (3.0/2.1)
|   |   medida = confianca_relativa:
|   |   |   atributos_por_regra <= 4 : I (2.0/1.0)
|   |   |   atributos_por_regra > 4 :
|   |   |   |   numero_exemplos <= 2500 : L (2.0/1.8)
|   |   |   |   numero_exemplos > 2500 :
|   |   |   |   |   momento <= 1.44e+07 : I (2.0/1.0)
|   |   |   |   |   momento > 1.44e+07 : S (2.0/1.0)
|   |   medida = sensibilidade:
|   |   |   sum <= 367 : I (2.0/1.0)
|   |   |   sum > 367 : L (3.0/1.1)
|
|
Tree saved

Evaluation on training data (156 items):
      Before Pruning      After Pruning
-----
Size      Errors      Size      Errors      Estimate
60      14( 9.0%)      26      31(19.9%)      (34.4%)  <<

```

Figura 5.18: *Árvore de decisão simplificada para os gráficos das medidas*

cada formato gráfico (Figura 5.19). Os formatos com decréscimo linear e decréscimo após patamar são os menos frequentes com ocorrência em 12,8% e 14,1%, respectivamente. Os formatos com decréscimo inicial acentuado e do tipo sigmoidal, que têm maior restritividade como filtro, representam em conjunto a maioria das ocorrências, 73,1% dos casos (55,2% e 17,9%, respectivamente). Nesse sentido, pode-se considerar que as medidas objetivas representam um bom recurso para utilização como filtros. Cabe lembrar que o uso dessas medidas deve ser acompanhado pelo usuário, porque as medidas avaliam aspectos diferentes de qualidade da regra e principalmente devido à variabilidade existente nos formatos gráficos.

A Figura 5.20 mostra a distribuição do tipo de gráfico por base de dados. De modo geral, há predominância do tipo Decréscimo Inicial, com os demais formatos distribuídos

Figura 5.19: *Frequência dos formatos gráficos*

entre as diferentes bases. Há variabilidade na distribuição de formatos para a base ART D, para a qual foram alternados os parâmetros utilizados no *Apriori*.

Figura 5.20: *Distribuição dos tipos de gráficos por base de dados*

A Figura 5.21 mostra a distribuição do tipo de gráfico por medida avaliada. Mais uma

vez verifica-se a predominância do tipo Decréscimo Inicial. A variabilidade do tipo de gráfico em relação às medidas neste caso é menor, com várias medidas apresentando apenas dois tipos de comportamento gráfico diferentes (confiança relativa negativa, especificidade, suporte, entre outras).

Figura 5.21: Distribuição dos tipos de gráficos por medida

Analisando as Figuras 5.20 e 5.21, observa-se que os diferentes tipos de formatos gráficos estão distribuídos com maior uniformidade entre as bases de dados que em relação às medidas. Esse é um resultado desejável, considerando que o ideal seria um comportamento gráfico característico da medida e independente da base de dados.

A partir das análises realizadas na Seção 5.3.5, para o conjunto de bases de dados, foi elaborada a Tabela 5.5. Nessa tabela é apresentado o formato gráfico das medidas analisadas para cada base de dados considerada e o valor da moda associada a cada medida. As cores associadas aos diferentes formatos são utilizadas para facilitar a comparação dos dados contidos nesta tabela com aqueles da Tabela 5.6.

Na Tabela 5.5, as medidas **confiança** e **especificidade** apresentaram, em geral, curva com decréscimo após patamar relativamente constante. A medida **confiança negativa** apresentou, em geral, curvas com decréscimo sigmoidal. Para a medida J_1 as curvas com decréscimo inicial acentuado e decréscimo sigmoidal ocorreram com igual frequência. Para as demais medidas o comportamento gráfico predominante foi aquele com decréscimo inicial acentuado. Dessa análise pode-se inferir que a maioria das medidas são adequadas para uso como filtro.

Na Tabela 5.6 é apresentada, para a divisão em três clusters, a categoria em que cada gráfico de medida foi classificado e a moda associada a cada medida. O cluster de número

Tabela 5.5: *Resumo da categorização dos gráficos realizada na Seção 5.3.5*

	Bases de dados												Moda
	Artificiais										Reais		
Medida	A	B	C	D ₍₂₎	D ₍₃₎	D ₍₄₎	E	F	G	H	QVU	BMS-POS	
confiança	L	P	P	P	P	P	I	P	L	P	P	P	P
convicção	I	I	I	I	L	L	P	I	I	I	I	I	I
confiança negativa	L	S	S	S	I	I	I	S	S	S	S	S	L
novidade	P	I	S	S	I	I	I	S	S	S	S	I	I
confiança relativa	L	I	S	I	I	I	I	I	S	S	I	I	L
confiança relativa negativa	S	I	I	I	I	I	I	I	I	I	I	I	I
sensibilidade relativa	L	I	I	I	I	I	I	I	I	I	I	I	I
especificidade relativa	S	I	S	I	I	I	I	S	I	I	I	I	I
satisfação	L	I	S	I	I	I	I	L	S	S	I	I	P
sensibilidade	I	I	I	L	L	L	P	L	I	I	I	I	I
especificidade	P	P	P	L	L	L	P	P	P	P	P	P	P
suporte	I	I	I	I	L	L	L	I	I	I	I	I	I
J ₁	S	I	S	S	I	I	I	S	S	S	I	I	I/S

Legenda	
I	Decréscimo Inicial Acentuado
L	Decréscimo Linear
P	Decréscimo após Patamar
S	Decréscimo Sigmoidal

1 contém o grupo de gráficos com funções que apresentam decréscimo inicial acentuado (formato mais interessante para uso como filtro). Os clusters de números 2 e 3, contém os grupos de gráficos com funções que apresentam decréscimo linear e decréscimo após patamar constante, respectivamente. Foi incluída a classificação constante da Tabela 5.5 por meio das cores de fundo de cada célula, para maior facilidade na análise comparativa das tabelas.

Da análise dos dados contidos na Tabela 5.6, pode-se observar que as medidas **confiança negativa relativa** e **sensibilidade relativa** são aquelas que apresentam a menor variabilidade no formato gráfico – em apenas 1 das 12 bases de dados analisadas a forma foi diferente. O resultado do agrupamento realizado para a medida foi semelhante ao da análise do tipo manual. As medidas **especificidade**, **especificidade relativa** e **suporte** também apresentaram menor variação no formato gráfico. Esse aspecto indica que tais medidas poderiam ser empregadas como filtros em um sistema pouco dependente da supervisão do usuário. Para as demais medidas seria necessário supervisão do usuário, como o caso da medida **confiança**, que apresenta grande variabilidade nas diferentes bases de dados (1 gráfico com Decréscimo Inicial Acentuado, 2 gráficos com Decréscimo Linear e 9 gráficos com Decréscimo após Patamar).

Comparando a coluna moda das Tabelas 5.5 e 5.6 é possível constatar que, em grande

Tabela 5.6: *Resumo da categorização dos gráficos utilizando a técnica de cluster*

	Bases de dados												Moda
	Artificiais										Reais		
Medida	A	B	C	D ₍₂₎	D ₍₃₎	D ₍₄₎	E	F	G	H	QVU	BMS-POS	
confiança	2	3	3	3	3	3	1	3	3	3	3	2	3
convicção	1	1	2	1	2	2	1	1	1	1	1	1	1
confiança negativa	2	1	2	1	1	1	2	2	2	2	2	3	2
novidade	2	1	2	1	1	1	1	2	2	2	1	1	1
confiança relativa	2	1	1	1	1	1	1	2	2	1	2	2	1
confiança relativa negativa	2	1	1	1	1	1	1	1	1	1	1	1	1
sensibilidade relativa	2	1	1	1	1	1	1	1	1	1	1	1	1
especificidade relativa	2	1	2	1	1	1	1	1	2	1	1	1	1
satisfação	2	1	2	1	3	2	1	2	2	2	3	2	2
sensibilidade	1	2	2	3	2	2	1	1	1	2	1	1	1
especificidade	3	3	3	2	2	3	3	3	3	3	3	3	3
suporte	1	1	1	2	2	2	1	1	1	1	1	1	1
J ₁	2	1	2	1	1	1	1	1	2	2	1	1	1

Legenda	
1	Decréscimo Inicial Acentuado
2	Decréscimo Linear
3	Decréscimo após Patamar
	Decréscimo Sigmoidal

parte das medidas, o valor da moda foi coincidente. Apenas no caso das medidas **confiança negativa** e **satisfação** o resultado foi divergente. A medida **confiança negativa**, foi inicialmente classificada como tendo o formato sigmoidal. A classificação por cluster, identificou a medida como do tipo linear (deve-se observar que o formato sigmoidal é bastante parecido com o linear, diferenciando-se apenas nas extremidades da curva). Os gráficos com comportamento sigmoidal ficaram restritos aos clusters 1 e 2, com predominância no cluster 2. Esse comportamento pode ser explicado pela semelhança entre os formatos linear e sigmoidal. Os gráficos com o formato de decaimento inicial foram predominantes nas duas análises realizadas, por cluster e manual.

5.5 Restritividade dos Filtros

Nesta seção é avaliado o uso de medidas objetivas como filtros individuais e como filtros compostos, sendo considerada fundamentalmente a redução do número de regras. Para esta avaliação a base de dados QVU foi selecionada pela possibilidade de interpretação das regras geradas.

As medidas **confiança**, **J₁**, **especificidade relativa** e **suporte** foram selecionadas para ilus-

trar medidas de alta e baixa capacidade restritiva. A análise da capacidade restritiva de cada medida é realizada aumentando-se o valor mínimo de cada medida.

Na Tabela 5.7 são apresentados os resultados do uso de cada medida como filtro individual. O valor mínimo de cada medida foi aumentado em 5%, 10%, 30%, 50% e 80% (em relação ao seu intervalo de valores) e foram selecionadas as regras com valor da medida superior ao novo valor mínimo. Assim, para cada percentual de incremento e para cada medida são apresentados os números de regras que satisfazem essa condição e o percentual dessas regras em relação ao total.

Em termos gerais, as medidas **suporte** e **especificidade relativa** são as mais restritivas. A elevação de apenas 5% no valor mínimo de cada medida conduz a uma redução de aproximadamente 90% no número de regras. Para uma elevação de 10% no valor mínimo da medida, o número de regras atinge cerca de 99% de redução. Assim, essas medidas podem ser consideradas filtros altamente restritivos. Em termos do princípio de Pareto, tem-se que para 90% das conseqüências (5% de elevação) ocorrem 10% das causas, 99% das conseqüências (10% de elevação) são produzidos por apenas 1% das causas.

Tabela 5.7: Número de regras por percentual de elevação no valor mínimo da medida

Medida	Percentual de elevação no valor mínimo da medida				
	5%	10%	30%	50%	80%
	Número de regras selecionadas (percentual de redução no número de regras)				
Confiança	21.169 (14,34)	21.095 (14,64)	19.526 (20,99)	15.467 (37,41)	12.360 (49,99)
Suporte	2.137 (91,35)	303 (98,77)	131 (99,47)	74 (99,70)	2 (99,99)
Especificidade Relativa	2.888 (88,31)	234 (99,05)	57 (99,77)	11 (99,96)	2 (99,99)
J ₁	16.139 (34,69)	11.143 (54,91)	1.414 (94,28)	296 (98,80)	12 (99,95)

A medida J₁ pode ser considerada um filtro medianamente restritivo, porque apenas com elevação acima de 50% no valor mínimo da medida, ocorre uma redução expressiva no número de regras. Para a medida **confiança**, o valor máximo da medida ocorre em mais de 50% das regras, portanto essa medida não é adequada para uso como um filtro isolado.

O uso combinado de medidas incrementa a capacidade restritiva dos filtros. Na Tabela 5.8 são apresentados os resultados para 5% de incremento no valor mínimo de cada medida – o percentual menos restritivo – representando cerca de 95% do fenômeno (medida). Cabe lembrar que os valores presentes na diagonal da tabela são referentes ao uso da medida de maneira individual. Os demais valores correspondem à combinação das medidas duas-a-duas. Dentre as medidas analisadas, o **suporte** apresenta a redução mais expressiva no número de regras, quando combinado com as demais medidas e a **confiança**

apresenta a redução menos expressiva. O menor número de regras ocorre com a combinação das medidas **especificidade relativa** e **suporte** – 1.585 regras selecionadas das 24.713 regras, representando redução de aproximadamente 93% no número total de regras.

Tabela 5.8: *Número de regras selecionadas para 5% de elevação*

Medida	Confiança	Suporte	Especificidade Relativa	J ₁
Confiança	21.169	2.129	2.751	13.016
Suporte	2.129	2.137	1.585	1.732
Especificidade Relativa	2.751	1.585	2.888	2.843
J ₁	13.016	1.732	2.843	16.139

Além da combinação das medidas duas-a-duas, o uso das quatro medidas em conjunto como um filtro único também foi explorado. Os resultados são descritos na Tabela 5.9. Ocorre redução expressiva no número de regras em todas as situações e, a partir de 10% de elevação do valor mínimo de cada medida (90% de representatividade do fenômeno), o número de regras é reduzido o suficiente para ser apresentado a um especialista do domínio, para análise individualizada das regras. Para percentuais de elevação superiores a 30% a combinação de medidas não é eficiente, uma vez que não ocorre redução significativa no número de regras.

Tabela 5.9: *Número de regras selecionadas, combinando as quatro medidas*

Percentual de redução	5%	10%	30%	50%	80%
Número de regras	1.355	168	52	11	2

Na Tabela 5.10 são apresentadas as 11 (onze) regras selecionadas aplicando o percentual de 50% de incremento e combinando quatro medidas (ver Tabela 5.9). Estas regras são as mesmas 11 regras selecionadas quando aplicando-se o percentual de 50% de aumento para os valores da medida **especificidade relativa** (Tabela 5.7).

Para avaliar a qualidade do processo de filtragem, o conjunto de regras foi apresentado a um especialista. As regras de número 1 a 4, 9 e 10 podem ser consideradas “conhecimento óbvio” por estarem representando os atributos altamente correlacionados: **dpi_p**, **porpdimp** e **pdp**. As regras de número 5 a 8 são consideradas interessantes pelo especialista, principalmente as regras 7 e 8, que contradizem as expectativas: para valores reduzidos de **dpi_p** e **porpdimp**, são esperados altos valores de **Pess_Dom**.

Tabela 5.10: Regras selecionadas para avaliação do especialista do domínio. Fonte: Melanda & Rezende (2004a)

Regra	LHS $\rightarrow$ RHS	Conf.	Sup.	Espec. Rel.	J ₁
1	dpi_p in (... 12.0%] $\rightarrow$ porpdimp in (... 3.0%]	1.00	0.76	0.76	0.29
2	porpdimp in (... 3.0%] $\rightarrow$ dpi_p in (... 12.0%]	1.00	0.76	0.76	0.29
3	dpi_p in (... 12.0%] $\rightarrow$ porpdp in (99.8% ...]	0.94	0.60	0.46	0.17
4	porpdimp in (... 3.0%] $\rightarrow$ porpdp in (99.8% ...]	0.93	0.60	0.46	0.17
5	dpi_p in (... 12.0%] $\rightarrow$ SInstSan in (... 0.5%] and porpdimp in (... 3.0%]	1.0	0.57	0.57	0.22
6	porpdimp in (... 3.0%] $\rightarrow$ SInstSan in (... 0.5%] and dpi_p in (... 12.0%]	1.0	0.57	0.57	0.22
7	dpi_p in (... 12.0%] $\rightarrow$ Pess_Dom in (2.5 ... 3.5] and porpdimp in (... 3.0%]	1.00	0.51	0.51	0.20
8	porpdimp in (... 3.0%] $\rightarrow$ Pess_Dom in (2.5 ... 3.5] and dpi_p in (... 12.0%]	1.00	0.51	0.51	0.20
9	dpi_p in (... 12.0%] $\rightarrow$ porpdimp in (... 3.0%] and porpdp in (99.8% ...]	1.00	0.60	0.60	0.23
10	porpdimp in (... 3.0%] $\rightarrow$ dpi_p in (... 12.0%] and porpdp in (99.8% ...]	1.00	0.60	0.60	0.23
11	dpi_p in (... 12.0%] and porpdimp in (... 3.0%] $\rightarrow$ porpdp in (99.8% ...]	0.93	0.60	0.46	0.17

Legenda

<p>dpi_p – Percentual de domicílios permanentes improvisados</p> <p>porpdimp – Percentual de domicílios improvisados</p> <p>porpdp – Percentual de domicílios permanentes</p>	<p>SInstSan – Percentual de domicílios sem instalações sanitárias</p> <p>Pess_Dom – Número médio de pessoas por domicílio</p>
--	---

5.6 Considerações Finais

O emprego de medidas objetivas de avaliação de regras na etapa de Pós-processamento permite o estabelecimento de filtros para seleção de regras, uma vez que estas medidas fornecem uma indicação da força de associação (hipotética) entre o *LHS* e o *RHS* (Melanda & Rezende 2004a).

Nesse sentido, o experimento realizado permite constatar que algumas medidas são mais restritivas que outras e portanto mais adequadas à constituição de filtros. Outras medidas apresentam menor capacidade de restrição e podem ser empregadas como “pré-filtros” ou como medidas complementares. Também é observado que algumas medidas, aquelas com formato do tipo sigmoidal, são interessantes porque produzem pequenos conjuntos de regras em seus valores extremos.

Com relação à estabilidade do formato gráfico, e a conseqüente estabilidade da capacidade restritiva, ocorreu significativa variação apesar de, na maioria dos casos, o comportamento gráfico ter sido do tipo com decaimento inicial acentuado.

O experimento comparando a restritividade adotada para os filtros e a qualidade das regras geradas, avaliadas por usuário especialista, permite concluir que o uso de medidas isoladas e/ou combinadas proporciona resultados interessantes. Com relação à restritividade dos filtros combinados é constatado que níveis elevados de restritividade apresentam resultados semelhantes aos dos filtros isolados, não sendo portanto uma opção adequada nos estudos realizados.

Sendo assim, a supervisão do usuário é requerida durante um processo de análise utilizando tais medidas. Os resultados obtidos subsidiam a elaboração de um sistema que apóia o usuário na identificação de regras potencialmente interessantes. Essa identificação pode ser realizada utilizando uma ou mais medidas para filtrar regras. O controle de utilização das medidas deve ser realizado interativamente, pelo usuário, de acordo com a intensidade restritiva de cada medida. Nesse sentido é proposto o ambiente computacional *ARInE*, que é apresentado no capítulo a seguir.

Uma Metodologia para Pós-processamento de Regras de Associação

6.1 Considerações Iniciais

O emprego de métricas na avaliação de conhecimento extraído com a tarefa de Regras de Associação é um procedimento bastante usual. A maioria das abordagens utiliza uma ou mais métricas para ordenar as regras e então apresentá-las para avaliação do usuário final.

A partir das demandas para análise de Regras de Associação identificadas durante os experimentos apresentados no Capítulo 5 e dos resultados dos referidos experimentos, foi proposta uma metodologia para o pós-processamento de Regras de Associação fundamentada em análises gráficas de medidas objetivas. O diferencial dessa metodologia é o uso do Método de Pareto para definir um valor de “corte” nos valores das medidas objetivas durante a seleção de regras.

A metodologia proposta é viabilizada com o auxílio do ambiente computacional *ARInE-Environment for Association Rules Interactive Exploration* – e definiu o fluxo de análise de regras disponibilizado no ambiente, bem como suas principais funcionalidades.

Neste capítulo inicialmente é apresentada a metodologia proposta para Pós-processamento de Regras de Associação. A seguir, é apresentada definição de representação padronizada de Regras de Associação, visando incorporar a metodologia proposta no Projeto DISCOVER. Também é descrito neste capítulo o Ambiente *ARInE* que implementa a metodologia de Pós-processamento de Regras de Associação proposta nesta tese.

6.2 Proposição da Metodologia

A metodologia para Pós-processamento de Regras de Associação proposta nesta tese foi elaborada a partir dos objetivos iniciais da tese, de necessidades identificadas durante a realização dos experimentos apresentados no Capítulo 5 e dos resultados obtidos nesses experimentos.

Em linhas gerais, na metodologia proposta busca-se identificar pequenos grupos de regras para serem avaliadas por um especialista. A identificação destes grupos de regras inicia-se a partir de um conjunto de regras resultante da etapa de Extração de Conhecimento. Por meio de consultas simples pode-se selecionar regras com atributos ou itens que se deseja analisar constituindo, assim, um novo conjunto de regras (**focus-ruleset**). A partir da definição do **focus-ruleset** é iniciado o processo de análise das regras utilizando-se medidas objetivas. Para cada medida é possível analisar graficamente a distribuição de seus valores para o universo de regras do **focus-ruleset** utilizando os gráficos de dispersão e de barras e o gráfico de Pareto. Os gráficos de dispersão e de barras podem ser utilizados para a analisar a distribuição de valores em termos gerais e, também, para identificar algum comportamento diferenciado bastante evidente. Com o gráfico de Pareto é possível identificar o percentual do valor acumulado da medida, para o qual existe relação de poucas regras e elevado valor acumulado. Assim, define-se o conjunto de regras potencialmente interessantes para uma medida. Esse processo pode ser repetido para outras medidas, criando-se novos conjuntos potencialmente interessantes. É realizada a intersecção destes conjuntos e as regras resultantes são analisadas e interpretadas por especialista que irá identificar aquelas de seu interesse, ou seja, o conjunto de regras interessantes. Cabe dizer que o especialista também pode analisar os conjuntos de regras potencialmente interessantes individualmente.

O fluxo de análise adotado na metodologia proposta é apresentado na Figura 6.1, consistindo das seguintes etapas:

1. escolha do conjunto de regras. Este conjunto contém as regras geradas por um algoritmo para gerar Regras de Associação;
2. definição do **focus-ruleset** por meio consultas simples ao conjunto de regras;
3. análise de similaridade entre medidas, utilizando análise fatorial ou outros métodos de análise de similaridade (etapa opcional);
4. seleção das medidas a serem utilizadas como filtros. É determinada pelos interesses do especialista e subsidiada pela análise de similaridade entre medidas;
5. aplicação dos filtros apoiado por análises gráficas;
6. apresentação das regras selecionadas ao especialista para avaliação final;

Figura 6.1: Processo de identificação de regras interessantes

6.2.1 Definição do **focus-ruleset**

Na primeira etapa do processo de identificação das regras interessantes, o usuário, a partir de um conjunto de Regras de Associação define seu conjunto de regras de foco utilizando métodos simples de consulta à base de regras. Nas consultas podem ser especificados os termos e definidos se ocorrem no antecedente, no conseqüente ou em qualquer

parte da regra. Os termos podem ser nomes ou valores de atributos, ou ainda sua combinação. O resultado da consulta é apresentado ao usuário, que pode optar por realizar uma nova consulta, salvar ou excluir todas as regras resultantes, ou ainda realizar estas duas operações individualmente para cada regra. As regras salvas serão analisadas em etapas posteriores. As regras excluídas têm duas funções: as consultas subseqüentes não retornam regras que já foram excluídas; o usuário posteriormente pode analisar as regras excluídas. Esta etapa de definição do conjunto de regras foco é opcional, contudo é bastante útil quando o usuário está interessado em descobrir regras relacionadas a apenas alguns atributos.

6.2.2 *Análise e Seleção das Medidas Objetivas*

Definido o **focus-ruleset** é necessário selecionar as medidas objetivas que serão utilizadas para avaliar as regras. Para isso é proposto o uso da técnica de análise fatorial (Seção 4.5.2) para apoiar ao usuário na escolha das medidas.

A análise fatorial irá agrupar medidas correlacionadas. Isso permite adotar duas linhas de análise:

medidas correlacionadas: a análise usando medidas correlacionadas permite identificar regras que “fogem” do padrão de correlação apresentado pelas medidas. Por apresentarem um comportamento diferenciado estas regras podem ser interessantes, devendo então serem verificadas por um especialista.

medidas não correlacionadas: a análise usando medidas não correlacionadas, de maneira semelhante, permite identificar que não seguem um padrão de comportamento esperado – por exemplo valores semelhantes em medidas não correlacionadas;

Cabe lembrar que o processo proposto é iterativo e interativo, podendo o usuário selecionar uma medida, utilizá-la como filtro, avaliar regras e escolher outra medida e repetir o processo. Na próxima seção é descrito como as medidas objetivas selecionadas podem ser utilizadas como filtros.

6.2.3 Aplicação dos Filtros

Definido o conjunto de regras a ser analisado e selecionadas as medidas objetivas, a próxima fase consiste na aplicação de filtros utilizando estas medidas. Cada filtro aplicado gera um subconjunto de regras que pode ser interpretado pelo especialista ou ser combinado com o resultado dos demais filtros gerando um outro conjunto de regras potencialmente interessantes. Esses conjuntos são apresentados a um especialista para validação, identificando assim regras interessantes. Na Figura 6.1 pode-se observar que o processo de análise proposto é iterativo, com diversas etapas necessitando de intervenção do usuário e iterativo com alguns ciclos, se necessário.

A aplicação de filtros representa o conceito central da metodologia proposta, que é identificar pequenos conjuntos de regras de maneira que o usuário/especialista possa interpretá-los e identificar as regras que considere interessantes. Estes pequenos conjuntos podem ser, por exemplo, identificados visualmente observando-se as áreas em destaque no gráfico de dispersão da Figura 6.2(a). Contudo, nem sempre a identificação destes conjuntos é trivial, como no caso apresentado na Figura 6.2(b) na qual os pontos estão dispersos. Nesse caso é necessário o uso de outros métodos de análise. Para tanto, são propostas outras duas maneiras de análise também baseadas no estudo de gráficos: a distribuição de frequência e a análise de Pareto.

Figura 6.2: Gráficos de dispersão de valores de medidas para Regras de Associação

Na análise da distribuição de frequência, os valores de uma dada medida são discretizados e é verificado o número de regras para cada intervalo. Na Figura 6.3 são exemplificados os resultados obtidos para esse tipo de análise. Esse procedimento visa analisar conjunta e comparativamente regras com características semelhantes (valor da medida).

Figura 6.3: Gráficos de distribuição de frequência de valores de medidas para Regras de Associação

A análise de Pareto é baseada no Princípio de Pareto (Seção 4.5.1, página 74), que afirma “para a maioria dos fenômenos naturais, 80% das conseqüências são originadas por apenas 20% das causas”. No âmbito da avaliação de regras de associação, o Princípio de Pareto pode ser assim interpretado: “apenas 20% das regras são responsáveis por 80% do valor acumulado de uma medida”. Desse modo é possível priorizar a análise de regras considerando-se apenas as regras que mais contribuem para uma dada medida. Na análise de Pareto, é gerado o gráfico da soma cumulativa dos valores da medida para cada regra considerada. Então o gráfico (Figura 6.4) é apresentado ao usuário que o interpreta e define o percentual limite de corte para seleção das regras. Como referência, é destacado o percentual de 80% no gráfico.

No caso apresentado na Figura 6.4(a) a relação 80/20 não é válida, mas pode ser obtida a relação 60/30 que indica que para 60% de representatividade da medida são selecionadas apenas 30% das regras. Já na Figura 6.4(b) é obtida a relação 80/50 (para 80% de representatividade da medida, são selecionados 50% das regras). A relação dessa proporção indica que a medida pode ser mais útil como um pré-filtro para combinação com outra medida.

Independente do modo de seleção utilizado (dispersão, análise de distribuição ou análise de Pareto), cabe lembrar que é possível reduzir ainda mais o número de regras a ser apresentado ao usuário especialista, por meio da união ou da intersecção dos conjuntos

Figura 6.4: Gráficos para análise de Pareto

de regras de maior contribuição para cada medida, de maneira semelhante à apresentada nas Tabelas 5.8 e 5.9, página 135.

6.2.4 Avaliação do Usuário Especialista

Para facilitar e flexibilizar as análises propostas, os conjuntos de regras são inicialmente armazenados no banco de dados do Ambiente RULEE. Para integração ao RULEE e conseqüentemente ao Projeto DISCOVER, foi necessário definir uma representação padronizada para as Regras de Associação (Melanda & Rezende 2003b). Detalhes da relação *ARInE* – RULEE são apresentados na Seção 6.4.

6.3 Representação Padronizada de Regras de Associação

Em Regras de Associação, apesar da semelhança na forma de representação das hipóteses, cada algoritmo salva regras geradas com sintaxes diferentes. Os algoritmos além das regras, fornecem as métricas **suporte** e **confiança**, geralmente acompanhadas de métricas complementares – *lift*, *confiança esperada*, *leverage*, entre outras – variando de algoritmo para algoritmo.

Devido às diferentes sintaxes de representação das regras geradas pelos algoritmos, qualquer medida ou estatística que necessite ser calculada sobre as regras geradas por diferentes algoritmos deveria ser implementada várias vezes. Esse fato também dificulta a interpretação das regras pelo usuário, que deverá lidar com símbolos e formatos diferentes para cada conjunto de regras e para cada algoritmo utilizado.

Para propiciar ao usuário um conjunto de informações padronizadas sobre cada regra e evitar retrabalho nas implementações das métricas de avaliação, foi definida uma sintaxe padrão para representação de Regras de Associação. Com base nessa sintaxe, são padronizadas as saídas dos algoritmos de associação e convertida para um formato único no qual as regras não são ordenadas.

Considerando que a maioria das medidas objetivas para avaliação de regras encontradas na literatura podem ser derivadas da tabela de contingência, o formato padrão para Regras de Associação contém as frequências relativas $f(LHS \ RHS)$, $f(LHS \ \overline{RHS})$, $f(\overline{LHS} \ \overline{RHS})$ e $f(\overline{LHS} \ RHS)$, além do número total de exemplos considerados N . Os valores absolutos das medidas podem ser obtidos pela multiplicação das frequências relativas por N . As frequências marginais $f(LHS)$, $f(RHS)$, $f(\overline{LHS})$ e $f(\overline{RHS})$ também podem ser obtidas pela soma das linhas e colunas da tabela de contingência (Tabela 4.2, página 61). Assim, para cada regra são acrescentados os valores de frequência relativa e o número de exemplos, do seguinte modo:

$$[f(LHS \ RHS), f(LHS \ \overline{RHS}), f(\overline{LHS} \ \overline{RHS}), f(\overline{LHS} \ RHS), N]$$

Esse conjunto de informações é calculado a partir das métricas fornecidas pelos algoritmos de Regras de Associação, conforme apresentado na Seção 6.3.2.

Sendo assim, uma Regra de Associação na sintaxe padrão apresenta a formatação mostrada na Expressão 6.1.

itens do antecedente, itens do consequente,

$$[f(LHS \ RHS), f(LHS \ \overline{RHS}), f(\overline{LHS} \ \overline{RHS}), f(\overline{LHS} \ RHS), n] \quad (6.1)$$

Cabe observar que a metodologia desenvolvida para análise de Regras de Associação faz parte da proposta do DISCOVER. Desse modo, foi desenvolvida uma biblioteca com *scripts* para conversão dos dados na sintaxe padrão de exemplos para o formato de en-

Figura 6.5: Metodologia dos formatos padrão para comunicação entre as funcionalidades do Ambiente DISCOVER

trada dos algoritmos de extração de Regras de Associação *Apriori* e dos softwares *WEKA*, *MineSetTM* e *MagnumOpus*, e também, os *scripts* para conversão da saída desses algoritmos para a sintaxe padrão de Regras de Associação. Essa abordagem segue o padrão estabelecido para comunicação entre as funcionalidades do Ambiente DISCOVER, como apresentado na Figura 6.5.

6.3.1 Gramáticas e Sintaxes para Representar Regras de Associação

Para a definição de uma sintaxe padrão para Regras de Associação, é especificada uma gramática a ser utilizada na conversão das regras geradas pelos algoritmos de associação para essa sintaxe. A definição dessa gramática levou em consideração a padronização definida pelo grupo do LABIC que trabalha com Extração de Conhecimento de Dados. Foi considerada também, a necessidade de otimização no processamento de arquivos de regras e a compreensibilidade das regras na sintaxe padrão para os usuários.

Nesse contexto, foi proposta a sintaxe padrão para Regras de Associação, que possui boa compreensibilidade e legibilidade permitindo fácil leitura e interpretação das regras pelo usuário. Contudo, considerando o elevado número de regras e o tamanho dos arquivos gerados foi necessário desenvolver uma sintaxe padrão compacta permitindo o processamento mais ágil e proporcionando arquivos de regras com tamanho consideravelmente

inferior (de 30% a 40% menor, conforme alguns experimentos). Porém a compreensibilidade desta sintaxe é significativamente menor quando comparada à sintaxe padrão.

Sintaxe Padrão para Regra de Associação

A gramática que define a sintaxe padrão para Regras de Associação, é definida como uma quádrupla $G = (V_n, V_t, P, S)$:

S : <regra associação>

V_n : { <regra associação>, <nro regra>, <regra_a>, <extensão>, <antecedente>, <conseqüente>, <fator>, <termo>, <valor numérico>, <valor categórico>, <comparação>, <operador relacional>, <identificador>, <sinal>, <dígitos>, <dígito>, <letra> }

V_t : { IF, THEN, AND, <=, >=, =, <, >, <>, [,], +, -, ., ,, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z, _ }

P : é descrito usando a notação EBNF e representado na Figura 6.6.

$[\alpha] = \alpha | \lambda$

$\{\alpha\} =$ repetição da cadeia α zero ou mais vezes

$\alpha | \beta = \alpha$ ou β devem ser escolhidos

Não terminais aparecem entre < e >, e terminais aparecem em negrito>.

As regras geradas por um algoritmo são convertidas para a sintaxe padrão de Regras de Associação utilizando as produções apresentadas na Figura 6.6. Na seqüência são apresentados alguns exemplos dessa conversão.

Exemplo de regra gerada pelo **MineSetTM** (apresentado na Figura 6.7), convertido para a sintaxe padrão de Regras de Associação pode ser observado na Figura 6.8. Deve-se lembrar que, no formato original, as regras são armazenadas em arquivo-texto, com uma regra por linha. Na sintaxe padrão as regras ainda são armazenadas em arquivo-texto só que não mais uma regra por linha.

```

<regra associação> ::= <nro regra> , <regra_a> , <extensão>
<nro regra> ::= [ R <dígitos> ]
<regra_a> ::= IF <antecedente> THEN <consequente>
<extensão> ::= [<valor numérico> , <valor numérico> , <valor numérico> ,
                <valor numérico> , <dígitos> ]
<antecedente> ::= TRUE | <fator>
<consequente> ::= <fator>
<fator> ::= <termo> | <termo> AND <fator>
<termo> ::= <identificador> | <comparação>
<comparação> ::= <identificador> <operador relacional> <valor numérico>
                | <identificador> <operador relacional> <valor categórico>
<operador relacional> ::= <= | >= | = | < | > | <>
<valor categórico> ::= <identificador>
<valor numérico> ::= [ <sinal> ] <dígitos> [ . <dígitos> ]
                | [ <sinal> ] <dígitos> [ . <dígitos> ] e <sinal> <dígitos>
<sinal> ::= + | -
<dígitos> ::= <dígito> { <dígito> }
<dígito> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
<identificador> ::= <letra> | <dígito> { <letra> | <dígito> }
<letra> ::= a | b | c | d | e | f | g | h | i | j | k | l | m | n | o
          | p | q | r | s | t | u | v | w | x | y | z | A | B | C | D
          | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S
          | T | U | V | W | X | Y | Z | _

```

Figura 6.6: *Produções da gramática para Regras de Associação*

```

2 2 7.1429 100 42.86 2.3333 outlook=overcast and temperature=cool
                                humidity=normal and play=yes

```

Figura 6.7: *Regra de Associação gerada pelo algoritmo do MineSetTM*

```

[R0547] IF outlook=overcast
      AND temperature=cool
      THEN humidity=normal
      AND play=yes
      [0.071429,0.000000,0.571400,0.357171,14]

```

Figura 6.8: Regra de Associação da Figura 6.7 na sintaxe padrão

Observe-se também que algumas informações fornecidas pela regra original (por exemplo: 2 2 7.1429 100) não são armazenadas na sintaxe padrão, uma vez que essas medidas normalmente são distintas para cada algoritmo de associação. As informações necessárias à avaliação de regras – tabela de contingência – são calculadas, como descrito na Seção 6.3.2, e armazenadas de modo padronizado e independente do algoritmo que as produziu.

Sintaxe Padrão Compacta para Regras de Associação

A gramática que define a sintaxe padrão compacta para Regras de Associação, é definida como uma quádrupla $G = (V_n, V_t, P, S)$:

S : <regra associação>

V_n : { <regra associação>, <nro regra>, <regra_a>, <extensão>, <antecedente>, <consequente>, <fator>, <termo>, <valor numérico>, <valor categórico>, <comparação>, <operador relacional>, <identificador>, <sinal>, <dígitos>, <dígito>, <letra> }

V_t : { &, <=, >=, =, <, >, <>, [,], +, -, ., ,, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z, - }

P : é descrito usando a notação EBNF e representado na Figura 6.9.

As regras geradas são convertidas para a sintaxe padrão compacta de Regras de As-

```

<regra associação> ::= <nro regra> , <regra_a> , <extensão>
<nro regra> ::= [ R <dígitos> ]
<regra_a> ::= <antecedente> , <consequente>
<extensão> ::= [ <valor numérico> , <valor numérico> , <valor numérico> ,
                  <valor numérico> , <dígitos> ]
<antecedente> ::= TRUE | <fator>
<consequente> ::= <fator>
<fator> ::= <termo> | <termo> & <fator>
<termo> ::= <identificador> | <comparação>
<comparação> ::= <identificador> <operador relacional> <valor numérico>
                  | <identificador> <operador relacional> <valor categórico>
<operador relacional> ::= <= | >= | = | < | > | <>
<valor categórico> ::= <identificador>
<valor numérico> ::= [ <sinal> ] <dígitos> [ . <dígitos> ]
                  | [ <sinal> ] <dígitos> [ . <dígitos> ] e <sinal> <dígitos>
<sinal> ::= + | -
<dígitos> ::= <dígito> { <dígito> }
<dígito> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
<identificador> ::= <letra> | <dígito> { <letra> | <dígito> }
<letra> ::= a | b | c | d | e | f | g | h | i | j | k | l | m | n | o
           | p | q | r | s | t | u | v | w | x | y | z | A | B | C | D
           | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S
           | T | U | V | W | X | Y | Z | _

```

Figura 6.9: Produções da gramática para Regras de Associação no Formato Compacto

sociação utilizando as produções apresentadas na Figura 6.9. A regra apresentada na Figura 6.7, convertida para a sintaxe padrão compacta pode ser vista na Figura 6.10. Cabe destacar que todas as regras convertidas também são armazenadas em arquivo-texto com uma regra por linha. Assim como na sintaxe padrão, também são desconsideradas algumas informações presentes no arquivo de regras original.

```
[R0547],outlook=overcast & temperature=cool,humidity=normal &  
play=yes, [0.071429,0.000000,0.571400,0.357171,14]
```

Figura 6.10: *Regra de Associação da Figura 6.7 na sintaxe padrão compacta*

6.3.2 Cálculo da Tabela de Contingência

Diferentemente de outras técnicas de Mineração de Dados, a técnica Regras de Associação gera um número elevado de regras (da ordem de milhares ou centenas de milhares de regras), com cada regra representando um conhecimento único e independente de outras regras.

O cálculo da tabela de contingência para tal quantidade de regras demanda tempo considerável, já que é necessário avaliar, para cada regra, todos os exemplos existentes na base de dados, ou seja, ocorre uma passagem pela base de dados para cada regra. Assim, para um conjunto de um milhão de regras seria necessário um milhão de passagens pela base, o que evidentemente tornaria o processo muito lento.

Para contornar essa dificuldade, a alternativa adotada é calcular a tabela de contingência a partir de métricas fornecidas pelos algoritmos, calculadas durante o processo de geração das regras.

Suporte e confiança são duas métricas fornecidas pelos algoritmos de Regras de Associação, contudo são insuficientes para elaboração da tabela de contingência o que determina o emprego de outra métrica. A *confiança esperada*, o *lift* (razão entre *confiança* e *confiança esperada*) ou a *diferença de confiança esperada* (diferença entre *confiança* e *confiança esperada*), são métricas usualmente fornecidas pelos algoritmos em conjunto com *suporte* e *confiança*. Considerando que a *confiança esperada* pode ser obtida de maneira simples a partir das outras duas medidas mencionadas, esta foi adotada para ilustrar os cálculos da tabela de contingência.

Na sequência é apresentado como a tabela de contingência é elaborada a partir do *suporte*, *confiança* e *confiança esperada*. Estas medidas são definidas, respectivamente, pelas Equações 3.2, 3.3 (página 29 do Capítulo 3) e 4.2 (página 62 do Capítulo 4), e para

facilitar o entendimento são apresentadas novamente como se segue:

$$suporte(LHS \Rightarrow RHS) = suporte(LHS \cup RHS) = \frac{n(LHS \cup RHS)}{N} \times 100 \quad (6.2)$$

$$confiana(LHS \Rightarrow RHS) = \frac{suporte(LHS \cup RHS)}{Sup(LHS)} = \frac{n(LHS \cup RHS)}{n(LHS)} \times 100 \quad (6.3)$$

$$ConfEsp = f(RHS) = \frac{n(RHS)}{N} \times 100 \quad (6.4)$$

As Equações 6.5 e 6.6 que representam, respectivamente, o número de transações (exemplos) nas quais LHS ocorre e, o número de transações nas quais RHS não ocorre. Estes valores auxiliarão na elaboração da tabela de contingência para cada regra de associação.

Das Equações 6.2 e ?? é determinado $n(LHS)$:

$$\begin{aligned} n(LHS) &= p(LHS) \cdot N \\ &= N \cdot \frac{p(LHS \mid RHS)}{p(RHS \mid LHS)} \\ &= N \cdot \frac{suporte}{Confiana} \end{aligned} \quad (6.5)$$

Da Equação 6.4 é determinado $n(\overline{RHS})$:

$$\begin{aligned} n(\overline{RHS}) &= N - n(RHS) \\ &= N - ConfEsp \cdot N \\ &= N \cdot (1 - ConfEsp) \end{aligned} \quad (6.6)$$

A partir das Equações 6.2 a 6.6 são definidos os valores da tabela de contingência, expressos pelas Equações 6.7 a 6.10:

1. $n(LHS \mid RHS)$: número de verdadeiros positivos, pode ser obtido a partir da Equação

ção 6.2.

$$f(LHS \mid RHS) = \frac{suporte}{100} \cdot N \quad (6.7)$$

2. $n(LHS \mid \overline{RHS})$: número de falsos negativos, pode ser obtido a partir das Equações 6.5 e 6.7.

$$\begin{aligned} n(LHS \mid \overline{RHS}) &= n(LHS) - n(LHS \mid RHS) \\ &= N \cdot \frac{suporte}{confiana} - suporte \cdot N \\ &= N \cdot suporte \cdot \left(\frac{1}{Confiana} - 1 \right) \end{aligned} \quad (6.8)$$

3. $n(\overline{LHS} \mid RHS)$: número de falsos positivos, pode ser obtido a partir das Equações 6.4 e 6.7.

$$\begin{aligned} n(\overline{LHS} \mid RHS) &= n(RHS) - n(LHS \mid RHS) \\ &= ConfEsp \cdot N - suporte \cdot N \\ &= N \cdot (ConfEsp - suporte) \end{aligned} \quad (6.9)$$

4. $f(\overline{LHS} \mid \overline{RHS})$: frequência de verdadeiros negativos, pode ser obtido a partir das Equações 6.6 e 6.8.

$$\begin{aligned} f(\overline{LHS} \mid \overline{RHS}) &= n(\overline{RHS}) - n(LHS \mid \overline{RHS}) \\ &= N \cdot (1 - ConfEsp) - N \cdot suporte \cdot \left(\frac{1}{Confiana} - 1 \right) \\ &= N \cdot \left(1 - ConfEsp - suporte \cdot \left(\frac{1}{Confiana} - 1 \right) \right) \end{aligned} \quad (6.10)$$

Assim, utilizando-se apenas as medidas **suporte**, **confiança** e **confiança esperada**, é calculada a tabela de contingência para cada regra com as expressões apresentadas nas Equações 6.7, 6.8, 6.9 e 6.10. Cabe lembrar que esse recurso é usado para eliminar a necessidade de percorrer a base de dados para construção da tabela de contingência, otimizando o processo de cálculo.

6.3.3 Conversão de Formatos

Como já mencionado, as regras geradas por algoritmos diferentes quase sempre são expressas em formatos específicos e no Projeto DISCOVER a comunicação entre as atividades desenvolvidas em cada etapa do processo é realizada utilizando sintaxes padrão. O processo de conversão das regras para a sintaxe padrão de Regras de Associação é realizado pela classe **Rass_Conv** que contém os métodos que convertem a saída de cada algoritmo de associação para a sintaxe padrão ou para a sintaxe padrão compacta. A classe também permite a conversão entre as duas sintaxes. Na Figura 6.11 é apresentado o esquema funcional do módulo de conversão de sintaxes.

Figura 6.11: Conversão das saídas dos algoritmos de Regras de Associação

No restante da seção são descritos os sistemas e algoritmos de associação para os quais foram implementados os conversores, bem como exemplos de Regras de Associação no formato original de cada sistema e depois de convertidas para o formato padrão de Regras de Associação. Os aplicativos *Apriori* e *WEKA* são *freeware*. Para o *MineSetTM* existe uma versão licenciada para o LABIC e o aplicativo *Magnum Opus* foi utilizado em sua versão demonstrativa. Para facilitar a compreensão das regras geradas, os algoritmos foram executados utilizando a base *Weather Nominal* obtida do conjunto de exemplos do aplicativo *WEKA* e apresentada na Tabela 6.1.

Tabela 6.1: Base de dados Weather Nominal

outlook	temperature	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Apriori

O aplicativo *Apriori*, implementado por Christian Borgelt, é descrito na Seção 5.3.2. Para obter todas as informações necessárias para o cálculo da tabela de contingência e para que o módulo conversor atue e converta as regras para o formato padrão, o aplicativo deve ser executado com as opções: `-o -e1 -d0 -v -a -p%.9f%`. A opção `-o` indica que devem ser utilizadas as definições originais de **suporte** e **confiança**. A opção `-e1 -d0 -v` indica que as regras, além de satisfazer a condição de **suporte** e **confiança** mínimos, devem satisfazer a condição de “**diferença de confiança esperada**” superior ao valor especificado na opção `-d` que, para não afetar os resultados, deve possuir o valor 0 (zero) e a opção `-v` para apresentar o valor da medida no arquivo de saída. A opção `-a` inclui o valor absoluto do **suporte** (número de transações) e a opção `-p%.9f%` formata os valores de todas as medidas com 9 casas decimais. Cabe lembrar que a **diferença de confiança esperada** representa a diferença entre a **confiança esperada** e a **confiança** para uma regra.

Na Figura 6.12 é apresentado exemplo do arquivo de saída gerado pelo aplicativo *Apriori* e, na Figura 6.13, as regras desse exemplo na sintaxe padrão compacta. O formato geral da regra para o *Apriori* é: “*RHS* $\leftarrow$ *LHS* (suporte/número de exemplos cobertos pelo suporte, confiança, diferença de confiança esperada)”. Considerando a regra *outlook=sunny* $\leftarrow$ *temperature=cool* (7.142857143%/1, 25.000000000%, 10.714285714%), a tabela de contingência é calculada assim:

1. N:

$$\begin{aligned}
 \text{suporte}(LHS \text{ } RHS) &= 7.142857143 \\
 &= \frac{1}{N} \cdot 100 \\
 N &= \frac{1}{0.07142857143} \\
 &= 14
 \end{aligned}$$

2. $f(LHS \text{ } RHS)$:

$$\begin{aligned}
 f(LHS \text{ } RHS) &= \frac{\text{suporte}}{100} \\
 &= 0.07142857143
 \end{aligned}$$

3. $f(LHS \text{ } \overline{RHS})$:

$$\begin{aligned}
 f(LHS \text{ } \overline{RHS}) &= f(LHS) - f(LHS \text{ } RHS) \\
 &= \frac{\text{suporte}}{\text{confiana}} - \frac{\text{suporte}}{100} \\
 &= \frac{7.142857143}{25.000000000} - 0.07142857143 \\
 &= 0.21428571429
 \end{aligned}$$

4. $f(\overline{LHS} \text{ } RHS)$:

$$\begin{aligned}
 f(\overline{LHS} \text{ } RHS) &= f(RHS) - f(LHS \text{ } RHS) \\
 &= \frac{\text{ConfEsp}}{100} - \frac{\text{suporte}}{100} \\
 &= 0.357142857 - 0.07142857143 \\
 &= 0.285714285
 \end{aligned}$$

5. $f(\overline{LHS} \text{ } \overline{RHS})$:

$$\begin{aligned}
 f(\overline{LHS} \text{ } \overline{RHS}) &= f(\overline{RHS}) - f(LHS \text{ } \overline{RHS}) \\
 &= (1 - \frac{\text{ConfEsp}}{100}) - (\frac{\text{suporte}}{\text{confiana}} - \frac{\text{suporte}}{100}) \\
 &= 0.64285714285 - 0.21428571429 \\
 &= 0.42857142856
 \end{aligned}$$

```

play=no <- (35.714285714%/5, 35.714285714%, 0.000000000%)
temperature=mild <- (42.857142857%/6, 42.857142857%, 0.000000000%)
windy=FALSE <- (57.142857143%/8, 57.142857143%, 0.000000000%)
play=yes <- (64.285714286%/9, 64.285714286%, 0.000000000%)
outlook=sunny <- temperature=cool (7.142857143%/1, 25.000000000%, 10.714285714%)
temperature=cool <- outlook=sunny (7.142857143%/1, 20.000000000%, 8.571428571%)
play=no <- temperature=cool (7.142857143%/1, 25.000000000%, 10.714285714%)
temperature=cool <- play=no (7.142857143%/1, 20.000000000%, 8.571428571%)
outlook=overcast <- temperature=hot humidity=normal (7.142857143%/1, 100.000000000%, 71.428571429%)
temperature=hot <- outlook=overcast humidity=normal (7.142857143%/1, 50.000000000%, 21.428571429%)
windy=FALSE <- temperature=hot outlook=overcast (14.285714286%/2, 100.000000000%, 42.857142857%)
humidity=normal <- outlook=overcast play=yes (14.285714286%/2, 50.000000000%, 0.000000000%)
play=yes <- temperature=hot outlook=overcast humidity=high
                                     (7.142857143%/1, 100.000000000%, 35.714285714%)
humidity=high <- temperature=hot outlook=overcast play=yes (7.142857143%/1, 50.000000000%, 0.000000000%)
play=yes <- outlook=rainy temperature=mild humidity=normal windy=FALSE
                                     (7.142857143%/1, 100.000000000%, 35.714285714%)
windy=FALSE <- outlook=rainy temperature=mild humidity=normal play=yes
                                     (7.142857143%/1, 100.000000000%, 42.857142857%)
humidity=normal <- outlook=rainy temperature=mild windy=FALSE play=yes
                                     (7.142857143%/1, 50.000000000%, 0.000000000%)

```

Figura 6.12: Algumas regras geradas pelo Apriori utilizando o conjunto de dados Weather Nominal

```

[R0001],TRUE,play=no,[0.357143,0.642857,0.000000,0.000000,14]
[R0002],TRUE,temperature=mild,[0.428571,0.571429,0.000000,0.000000,14]
[R0003],TRUE,windy=FALSE,[0.571429,0.428571,0.000000,0.000000,14]
[R0004],TRUE,play=yes,[0.642857,0.357143,0.000000,0.000000,14]
[R0005],temperature=cool,outlook=sunny,[0.071428571,0.214285714,0.285714286,0.428571429,14]
[R0006],outlook=sunny,temperature=cool,[0.071428571,0.285714286,0.214285714,0.428571429,14]
[R0007],temperature=cool,play=no,[0.071428571,0.214285714,0.285714286,0.428571429,14]
[R0008],play=no,temperature=cool,[0.071428571,0.285714286,0.214285714,0.428571429,14]
[R0009],temperature=hot & humidity=normal,outlook=overcast,[0.071429,0.000000,0.714286,0.214286,14]
[R0010],outlook=overcast & humidity=normal,temperature=hot,[0.071429,0.071429,0.642857,0.214286,14]
[R0011],temperature=hot & outlook=overcast,windy=FALSE,[0.142857,0.000000,0.428571,0.428571,14]
[R0012],outlook=overcast & play=yes,humidity=normal,[0.142857,0.142857,0.357143,0.357143,14]
[R0013],temperature=hot & outlook=overcast & humidity=high,play=yes,
                                     [0.071429,0.000000,0.357143,0.571429,14]
[R0014],temperature=hot & outlook=overcast & play=yes,humidity=high,
                                     [0.071429,0.071429,0.428571,0.428571,14]
[R0015],outlook=rainy & temperature=mild & humidity=normal & windy=FALSE,play=yes,
                                     [0.071429,0.000000,0.357143,0.571429,14]
[R0016],outlook=rainy & temperature=mild & humidity=normal & play=yes,windy=FALSE,
                                     [0.071429,0.000000,0.428571,0.500000,14]
[R0017],outlook=rainy & temperature=mild & windy=FALSE & play=yes,humidity=normal,
                                     [0.071429,0.071429,0.428571,0.428571,14]

```

Figura 6.13: Regras da Figura 6.12 convertidas para o formato padrão compacto de Regras de Associação

WEKA

O ambiente *WEKA* (*Waikato Environment for Knowledge Analysis*) foi desenvolvido na Universidade de Waikato¹ na Nova Zelândia dentro da filosofia GNU de código fonte aberto. Possui ferramentas para pré-processamento de dados, classificação, regressão, *clustering*, Regras de Associação e visualização.

Os diversos algoritmos de Aprendizado de Máquina, implementados na linguagem Java, são representados cada qual por uma classe, o que permite que as classes sejam organizadas em pacotes, que são simplesmente diretórios contendo uma coleção de classes relacionadas. Isso é bastante útil porque certos algoritmos compartilham muitas funcionalidades, e desse modo, várias classes em um pacote podem ser utilizadas por mais de um algoritmo (Witten & Frank 1999). O pacote **weka.associations** contém a implementação do algoritmo Apriori proposto por Agrawal et al. (1993).

Para obter todas as informações necessárias para o cálculo da tabela de contingência e para que o módulo conversor atue e converta as regras para o formato padrão, é necessário que o módulo de Regras de Associação do *WEKA* seja executado com a opção *metricType* diferente de *confidence* (confiança), assim serão calculadas métricas adicionais que permitirão a elaboração da tabela de contingência sem a necessidade de consultas à base de dados.

Na Figura 6.14 é apresentado exemplo do arquivo de saída gerado pelo aplicativo *WEKA* e, na Figura 6.15, as regras desse exemplo na sintaxe padrão compacta. O formato geral da regra para o *WEKA* é: “número da regra *LHS* $n(LHS) ==> RHS$ $n(RHS)$ confiança lift leaverage convicção”. A tabela de contingência é calculada a partir dos valores de $n(LHS)$, $n(RHS)$ e lift. Considerando a regra: **1. humidity=normal play=no 1 ==> outlook=rainy temperature=cool windy=TRUE 1 conf:(1) < lift:(14) > lev:(0.07) [0] conv:(182)** e os valores $n(LHS)=1$, $n(RHS)=1$ e lift=14, tem-se a tabela de contingência [0.071429, 0.000000, 0.928571, 0.000000, 14].

Diferentemente do que ocorre com o algoritmo Apriori, na conversão do formato original do *WEKA* para o formato padrão de Regras de Associação ocorre a perda de detalhes do processo de geração de regras (parte superior da Figura 6.14).

¹www.waikato.ac.nz

```

=== Run information ===

Scheme:      weka.associations.Apriori -N 100000 -T 1 -C 0.1 -D 0.02 -U 1.0 -M 0.02 -S -1.0
Relation:    weather.symbolic
Instances:    14
Attributes:   5
              outlook
              temperature
              humidity
              windy
              play
=== Associator model (full training set) ===

Apriori
=====

Minimum support: 0.02
Minimum metric <lift>: 0.1
Number of cycles performed: 48

Generated sets of large itemsets:

Size of set of large itemsets L(1): 12
Size of set of large itemsets L(2): 54
Size of set of large itemsets L(3): 96
Size of set of large itemsets L(4): 64
Size of set of large itemsets L(5): 14

Best rules found:

  1. humidity=normal play=no 1 ==> outlook=rainy temperature=cool windy=TRUE 1
    conf:(1) < lift:(14)> lev:(0.07) [0] conv:(182)
  2. temperature=cool play=no 1 ==> outlook=rainy humidity=normal windy=TRUE 1
    conf:(1) < lift:(14)> lev:(0.07) [0] conv:(182)
  3. outlook=rainy humidity=normal windy=TRUE 1 ==> temperature=cool play=no 1
    conf:(1) < lift:(14)> lev:(0.07) [0] conv:(182)
420. humidity=normal play=yes 6 ==> outlook=rainy temperature=cool windy=FALSE 1
    conf:(0.17) < lift:(2.33)> lev:(0.04) [0] conv:(182)
421. temperature=cool play=yes 3 ==> outlook=rainy humidity=normal windy=FALSE 1
    conf:(0.33) < lift:(2.33)> lev:(0.04) [0] conv:(168)
422. outlook=rainy play=yes 3 ==> temperature=cool humidity=normal windy=FALSE 1
    conf:(0.33) < lift:(2.33)> lev:(0.04) [0] conv:(168)
622. windy=TRUE 6 ==> outlook=sunny temperature=mild humidity=normal 1
    conf:(0.17) < lift:(2.33)> lev:(0.04) [0] conv:(182)
623. outlook=sunny temperature=mild humidity=normal 1 ==> windy=TRUE 1
    conf:(1) < lift:(2.33)> lev:(0.04) [0] conv:(112)
624. outlook=sunny humidity=normal 2 ==> temperature=mild windy=TRUE 1
    conf:(0.5) < lift:(2.33)> lev:(0.04) [0] conv:(154)
1122. outlook=overcast temperature=mild humidity=high 1 ==> play=yes 1
    conf:(1) < lift:(1.56)> lev:(0.03) [0] conv:(70)
1123. play=yes 9 ==> outlook=overcast temperature=hot humidity=normal 1
    conf:(0.11) < lift:(1.56)> lev:(0.03) [0] conv:(182)
1124. outlook=overcast temperature=hot humidity=normal 1 ==> play=yes 1
    conf:(1) < lift:(1.56)> lev:(0.03) [0] conv:(70)
1997. windy=FALSE play=yes 6 ==> outlook=sunny 1
    conf:(0.17) < lift:(0.47)> lev:(-0.08) [-1] conv:(126)
1998. outlook=sunny 5 ==> windy=FALSE play=yes 1
    conf:(0.2) < lift:(0.47)> lev:(-0.08) [-1] conv:(112)
2000. humidity=normal 7 ==> play=no 1
    conf:(0.14) < lift:(0.4)> lev:(-0.11) [-1] conv:(126)

```

Figura 6.14: Algumas regras geradas pelo WEKA utilizando o conjunto de dados Weather Nominal

```

[R0001],humidity=normal & play=no,outlook=rainy & temperature=cool & windy=TRUE,
[0.071429,0.000000,0.928571,0.000000,14]
[R0002],temperature=cool & play=no,outlook=rainy & humidity=normal & windy=TRUE,
[0.071429,0.000000,0.928571,0.000000,14]
[R0003],outlook=rainy & humidity=normal & windy=TRUE,temperature=cool & play=no,
[0.071429,0.000000,0.928571,0.000000,14]
[R0420],humidity=normal & play=yes,outlook=rainy & temperature=cool & windy=FALSE,
[0.071429,0.357143,0.571326,0.000102,14]
[R0421],temperature=cool & play=yes,outlook=rainy & humidity=normal & windy=FALSE,
[0.071429,0.142857,0.714081,0.071633,14]
[R0422],outlook=rainy & play=yes,temperature=cool & humidity=normal & windy=FALSE,
[0.071429,0.142857,0.714081,0.071633,14]
[R0622],windy=TRUE,outlook=sunny & temperature=mild & humidity=normal,
[0.071429,0.357143,0.571326,0.000102,14]
[R0623],outlook=sunny & temperature=mild & humidity=normal,windy=TRUE,
[0.071429,0.000000,0.570815,0.357756,14]
[R0624],outlook=sunny & humidity=normal,temperature=mild & windy=TRUE,
[0.071429,0.071429,0.713979,0.143164,14]
[R1122],outlook=overcast & temperature=mild & humidity=high,play=yes,
[0.071429,0.000000,0.358974,0.569597,14]
[R1123],play=yes,outlook=overcast & temperature=hot & humidity=normal,
[0.071429,0.571429,0.357346,-0.000204,14]
[R1124],outlook=overcast & temperature=hot & humidity=normal,play=yes,
[0.071429,0.000000,0.358974,0.569597,14]
[R1997],windy=FALSE & play=yes,outlook=sunny,[0.071429,0.357143,0.288247,0.283181,14]
[R1998],outlook=sunny,windy=FALSE & play=yes,[0.071429,0.285714,0.288754,0.354103,14]
[R2000],humidity=normal,play=no,[0.071429,0.428571,0.214286,0.285714,14]

```

Figura 6.15: Regras da Figura 6.14 convertidas para o formato padrão compacto de Regras de Associação

MineSetTM

O aplicativo MineSetTM da SGI, é descrito na Seção 5.3.2, página 105. Na Figura 6.16 é apresentado exemplo do arquivo de saída, no formato ASCII, gerado por este aplicativo. Na Figura 6.17 são apresentadas as regras desse exemplo na sintaxe padrão compacta.

O formato geral da regra para o MineSetTM é: “número de atributos no antecedente, número de atributos no conseqüente, suporte, confiança, confiança esperada, lift *LHS RHS*”. A tabela de contingência é calculada a partir dos valores de suporte, confiança, confiança esperada. Considerando a regra: **1 1 21.4286 60 35.71 1.68 outlook=sunny play=no** e os valores $n(LHS)=1$, $n(RHS)=1$ e lift=14, tem-se a tabela de contingência **[0.214286,0.142857,0.500043,0.142814,14]**.

```

1 1 21.4286 60 35.71 1.68 outlook=sunny play=no
1 1 28.5714 100 64.29 1.5556 outlook=overcast play=yes
2 1 14.2857 66.67 64.29 1.037 outlook=rainy;and;temperature=mild play=yes
2 1 14.2857 50 35.71 1.4 temperature=cool;and;humidity=normal outlook=rainy
1 2 14.2857 50 21.43 2.3333 temperature=cool outlook=rainy;and;humidity=normal
1 2 21.4286 60 42.86 1.4 outlook=rainy windy=FALSE;and;play=yes
2 2 7.1429 50 21.43 2.3333 temperature=mild;and;humidity=normal outlook=rainy;and;windy=FALSE
3 1 7.1429 100 64.29 1.5556 outlook=rainy;and;temperature=mild;and;humidity=normal play=yes
2 2 7.1429 50 21.43 2.3333 temperature=mild;and;humidity=normal outlook=rainy;and;play=yes
2 2 7.1429 50 14.29 3.5 outlook=rainy;and;humidity=high temperature=mild;and;play=no
3 1 7.1429 50 35.71 1.4 outlook=rainy;and;temperature=cool;and;humidity=normal play=no
1 3 14.2857 50 21.43 2.3333 temperature=hot outlook=sunny;and;humidity=high;and;play=no
1 3 14.2857 50 14.29 3.5 outlook=overcast temperature=hot;and;windy=FALSE;and;play=yes
4 1 7.1429 100 64.29 1.5556 outlook=rainy;and;temperature=mild;and;humidity=normal;and;
windy=FALSE play=yes

```

Figura 6.16: Algumas regras geradas pelo MineSetTM utilizando o conjunto de dados Weather Nominal

```

[R0001]outlook=sunny,play=no,[0.214286,0.142857,0.500043,0.142814,14]
[R0002]outlook=overcast,play=yes,[0.285714,0.000000,0.357100,0.357186,14]
[R0003]outlook=rainy & temperature=mild,play=yes,[0.142857,0.071418,0.285682,0.500043,14]
[R0004]temperature=cool & humidity=normal,outlook=rainy,[0.142857,0.142857,0.500043,0.214243,14]
[R0005]temperature=cool,outlook=rainy & humidity=normal,[0.142857,0.142857,0.642843,0.071443,14]
[R0006]outlook=rainy,windy=FALSE & play=yes,[0.214286,0.142857,0.428543,0.214314,14]
[R0007]temperature=mild & humidity=normal,outlook=rainy & windy=FALSE,
[0.071429,0.071429,0.714271,0.142871,14]
[R0008]outlook=rainy & temperature=mild & humidity=normal,play=yes,
[0.071429,0.000000,0.357100,0.571471,14]
[R0009]temperature=mild & humidity=normal,outlook=rainy & play=yes,
[0.071429,0.071429,0.714271,0.142871,14]
[R0010]outlook=rainy & humidity=high,temperature=mild & play=no,[0.071429,0.071429,0.785671,0.071471,14]
[R0011]outlook=rainy & temperature=cool & humidity=normal,play=no,
[0.071429,0.071429,0.571471,0.285671,14]
[R0012]temperature=hot,outlook=sunny & humidity=high & play=no,[0.142857,0.142857,0.642843,0.071443,14]
[R0013]outlook=overcast,temperature=hot & windy=FALSE & play=yes,
[0.142857,0.142857,0.714243,0.000043,14]
[R0014]outlook=rainy & temperature=mild & humidity=normal & windy=FALSE,play=yes,
[0.071429,0.000000,0.357100,0.571471,14]

```

Figura 6.17: Regras da Figura 6.16 convertidas para o formato padrão compacto de Regras de Associação

Magnum Opus

O aplicativo *Magnum Opus*, comercializado pela Rule Quest², é a implementação do algoritmo Opus proposto por Webb (1995). Esse aplicativo apresenta como principais características:

- além do suporte, considera as métricas *coverage*, *lift*, *leverage* e *strength*;

²www.rulequest.com

- define o número máximo de regras a serem geradas;
- permite definir o número máximo de atributos em *LHS*, mas gera regras com apenas um item em *RHS*;
- possui filtros para regras consideradas improdutivas, triviais e insignificantes;
- possui recursos, apesar de limitados, para tratar atributos contínuos.

Na Figura 6.18 é apresentado exemplo do arquivo de saída gerado pelo aplicativo *Magnum Opus* e, na Figura 6.19 são apresentadas as regras desse exemplo na sintaxe padrão compacta.

O formato geral da regra para o *Magnum Opus*: “*LHS, RHS, cobertura, (#cobertura), suporte, (#suporte), strength, lift, leverage, (#leverage)*” (a representação *(#nome_medida)* indica o número de exemplos cobertos pela medida). A tabela de contingência é calculada a partir dos valores de *cobertura*, *suporte* e *lift*. Considerando a regra: *humidity=normal, play=yes, 0.500, 7, 0.429, 6, 0.857, 1.33, 0.1071, 1* e os valores *cobertura=0.500*, *suporte=1.33* e *lift=14*, tem-se a tabela de contingência *[0.429000, 0.071583, 0.284056, 0.215361, 14]*.

```
LHS,RHS,Coverage,(Coverage),Support,(Support),Strength,Lift,Leverage,(Leverage)
humidity=normal,play=yes,0.500,7,0.429,6,0.857,1.33,0.1071,1
humidity=normal,temperature=cool,0.500,7,0.286,4,0.571,2.00,0.1429,2
humidity=high,play=no,0.500,7,0.286,4,0.571,1.60,0.1071,1
humidity=normal & windy=FALSE,play=yes,0.286,4,0.286,4,1.000,1.56,0.1020,1
humidity=normal & play=yes,temperature=cool,0.429,6,0.214,3,0.500,1.75,0.0918,1
outlook=sunny & humidity=high,play=no,0.214,3,0.214,3,1.000,2.80,0.1378,1
outlook=rainy & windy=FALSE & play=yes,humidity=normal,0.214,3,0.143,2,0.667,1.33,0.0357,0
temperature=cool & humidity=normal & windy=FALSE,play=yes,0.143,2,0.143,2,1.000,1.56,0.0510,0
outlook=sunny & temperature=hot & humidity=high,play=no,0.143,2,0.143,2,1.000,2.80,0.0918,1
temperature=cool & humidity=normal & windy=FALSE & play=yes,outlook=rainy,
0.143,2,0.071,1,0.500,1.40,0.0204,0
outlook=overcast & temperature=cool & humidity=normal & windy=TRUE,play=yes,
0.071,1,0.071,1,1.000,1.56,0.0255,0
outlook=rainy & temperature=cool & humidity=normal & windy=TRUE,play=no,
0.071,1,0.071,1,1.000,2.80,0.0459,0
outlook=rainy & temperature=cool & windy=TRUE,play=no,0.071,1,0.071,1,1.000,2.80,0.0459,0
outlook=rainy & temperature=cool & humidity=normal & windy=FALSE,play=yes,
0.071,1,0.071,1,1.000,1.56,0.0255,0
outlook=rainy & temperature=cool & play=yes,windy=FALSE,0.071,1,0.071,1,1.000,1.75,0.0306,0
```

Figura 6.18: Algumas regras geradas pelo *Magnum Opus* utilizando o conjunto de dados *Weather Nominal*

```

[R0001],humidity=normal,play=yes,[0.429000,0.071583,0.284056,0.215361,14]
[R0002],humidity=normal,temperature=cool,[0.286000,0.214876,0.499624,-0.000500,14]
[R0003],humidity=high,play=no,[0.286000,0.214876,0.428249,0.070875,14]
[R0004],humidity=normal & windy=FALSE,play=yes,[0.286000,0.000000,0.358974,0.355026,14]
[R0005],humidity=normal & play=yes,temperature=cool,[0.214000,0.214000,0.500286,0.071714,14]
[R0006],outlook=sunny & humidity=high,play=no,[0.214000,0.000000,0.642857,0.143143,14]
[R0007],outlook=rainy & windy=FALSE & play=yes,humidity=normal,[0.143000,0.071393,0.427103,0.358504,14]
[R0008],temperature=cool & humidity=normal & windy=FALSE,play=yes,[0.143000,0.000000,0.358974,0.498026,14]
[R0009],outlook=sunny & temperature=hot & humidity=high,play=no,[0.143000,0.000000,0.642857,0.214143,14]
[R0010],temperature=cool & humidity=normal & windy=FALSE & play=yes,outlook=rainy,
[0.071000,0.071000,0.571857,0.286143,14]
[R0011],outlook=overcast & temperature=cool & humidity=normal & windy=TRUE,play=yes,
[0.071000,0.000000,0.358974,0.570026,14]
[R0012],outlook=rainy & temperature=cool & humidity=normal & windy=TRUE,play=no,
[0.071000,0.000000,0.642857,0.286143,14]
[R0013],outlook=rainy & temperature=cool & windy=TRUE,play=no,[0.071000,0.000000,0.642857,0.286143,14]
[R0014],outlook=rainy & temperature=cool & humidity=normal & windy=FALSE,play=yes,
[0.071000,0.000000,0.358974,0.570026,14]
[R0015],outlook=rainy & temperature=cool & play=yes,windy=FALSE,[0.071000,0.000000,0.428571,0.500429,14]

```

Figura 6.19: Regras da Figura 6.18 convertidas para o formato padrão compacto de Regras de Associação

6.3.4 Considerações Sobre a Sintaxe Padrão

Uma sintaxe padrão para representar Regras de Associação é importante pois permite que as regras geradas pelos diferentes algoritmos, em diferentes formatos, sejam analisadas e exploradas utilizando um conjunto padrão de métricas, visto que conversores podem ser implementados para colocar as regras em um formato único. Nesse sentido, a definição de uma sintaxe padrão para representar Regras de Associação, e a implementação de uma classe em *PERL* para converter as saídas dos algoritmos de associação *Apriori*, *MineSetTM*, *WEKA* e *Magnum Opus* para essa sintaxe, viabilizam a implementação de diversas medidas de pós-processamento uma única vez e a utilização destas com regras geradas pelos diferentes algoritmos.

Considerando as restrições quanto ao tamanho dos arquivos de regras geradas, os requisitos de compreensibilidade do conhecimento pelo usuário e a eficiência no processamento dos arquivos de regras foi necessário adotar duas sintaxes: uma padrão – com elevada compreensibilidade; outra compacta – com arquivos cerca de 30% menores. A sintaxe padrão é utilizada para apresentação direta a um usuário (por meio de arquivos no formato texto) e a sintaxe padrão compacta é utilizada para a inserção dos conjuntos de regras nos ambientes *RULEE* e *ARInE*.

Por fim, a partir da definição dessas sintaxes foi possível incrementar a potencialidade do Projeto DISCOVER para trabalhar problemas de associação.

6.4 Implementação da Metodologia Proposta – o Ambiente *ARInE*

A implementação do Ambiente *ARInE* (*Association Rules Interactive Explorer*) objetiva, primeiramente, viabilizar a metodologia de análise de Regras de Associação proposta na Seção 6.2 por meio de métodos automatizados e iterativos. Foi projetado e implementado para disponibilizar um ambiente interativo com o objetivo de apoiar a avaliação de Regras de Associação. A criação de um ambiente específico para avaliação de Regras de Associação é motivada pelas características diferenciadoras entre esse tipo de regra e outros tipos de regras, a saber: complexidade da regra, atividade de Mineração de Dados e, principalmente volume de conhecimento obtido. Adicionalmente, o ambiente provê o Projeto DISCOVER de uma ferramenta computacional específica para análise de Regras de Associação.

Nesse sentido, foram verificados inicialmente os requisitos gerais para um Ambiente para Exploração de Regras que nortearam o projeto do Ambiente *ARInE*. Considerando que o Ambiente *ARInE* está inserido no Projeto DISCOVER (descrito na Seção 4.6.3, página 86), utiliza algumas funcionalidades do Ambiente RULEE (descrito na Seção 4.6.4, página 90) e, também, as peculiaridades das Regras de Associação, foi necessário definir uma representação padronizada para as Regras de Associação. Desse modo, seguindo os requisitos, as demandas e padrões definidos na Seção 4.6.1, página 79 foi implementado o Ambiente *ARInE* com funcionalidades para análise interativa, iterativa e gráfica de Regras de Associação, fornecendo assim o suporte necessário à execução do Pós-processamento de Regras de Associação segundo a metodologia proposta nesta tese.

Inicialmente o ambiente *ARInE* foi concebido como módulo computacional de Pós-processamento do Ambiente RULEE, apresentado na Seção 4.6.4. Entretanto, por limitações de desempenho de vários métodos disponibilizados pelo RULEE, optou-se por desenvolver um ambiente complementar com características e otimizações especificamente elaboradas para tratar Regras de Associação.

6.4.1 Relação *ARInE* – RULEE

Na Figura 6.20 é ilustrada a relação *ARInE* – RULEE, com a região sombreada indicando os elementos do RULEE utilizados pelo *ARInE*. Do Módulo de Gerenciamento, são utilizados exclusivamente os procedimentos para controle de seção e o direito de acesso aos

repositórios pelos usuários. Do repositório de regras e medidas são utilizados os métodos para recuperação de regras e para cálculo e recuperação dos valores de medidas objetivas.

Figura 6.20: Relação entre *ARInE* e *RULEE*

Para prover o Ambiente *RULEE* das funcionalidades necessárias para tratar Regras de Associação e para viabilizar o funcionamento do Ambiente *ARInE* foram realizadas otimizações em alguns módulos *PERL* e alterações na estrutura do banco de dados utilizado. As principais alterações nos métodos *PERL* foram realizadas nas classes *Ruleset*, que provê métodos de manipulação dos conjuntos de regras, e *Selection* para recuperação de regras e medidas armazenadas no repositório:

InsertAssociationRules – este método é utilizado para inserir novos conjuntos de Regras de Associação no repositório de regras, a partir de um arquivo de regras na sintaxe padrão compacta de Regras de Associação. O método insere as regras, os valores da tabela de contingência e calcula os valores das medidas objetivas definidas como “pré-calculadas” (na inclusão de uma medida no ambiente deve ser especificado se a medida será pré-calculada ou não). Neste método foram realizadas alterações que otimizam o uso de memória, permitindo a inserção de grandes conjuntos de regras. Com relação ao tempo de processamento também foram obtidas melhorias especialmente para conjuntos de regras de tamanho (em bytes) superior à quantidade de memória disponível no computador que está instalado o sistema;

Remove – este método é utilizado para excluir um conjunto regras. Originalmente, de-

vido a limitações do gerenciador de banco de dados, eram selecionadas todas as regras do conjunto a ser excluído. Então para cada regra era executado o método *Rule_Remove* que excluía os valores de medidas associados à respectiva regra e então excluía a regra. Com esse procedimento, era executado uma instrução SELECT que retornava todas as regras (que eram mantidas na memória durante o processo de exclusão) e duas instruções DELETE para cada regra (uma apaga os valores das medidas e outra apaga a regra), ou seja, para a exclusão de um conjunto de regras com um milhão e meio de regras seria necessária a execução de mais de três milhões de instruções SQL. Na alteração implementada, é executado uma única instrução DELETE que exclui todas as regras e os valores das medidas. Esta alteração viabilizou a exclusão de conjuntos de regras com mais de 20.000 regras.

WhereCondition – este método constrói uma cláusula WHERE a partir de uma expressão simples como a apresentada na Figura 6.25. As modificações realizadas permitiram incluir a avaliação de expressões que contenham valores para atributos. Anteriormente, não era possível utilizar a expressão `atributo=valor_do_atributo`

A principal alteração efetuada no banco de dados utilizado pelo Ambiente RULEE foi a inclusão de quatro novas tabelas ARINE, ARINE_EXCLUDED_RULES, ARINE_INTERSECTION_RULES e ARINE_SAVED_SESSION, como pode ser observado na Figura 6.21 (tabelas destacadas).

O ambiente *ARInE* utiliza do ambiente RULEE a estrutura para acesso aos repositórios de regras e de medidas.

Figura 6.21: Diagrama físico da base de dados utilizada pelos ambientes ARInE e RULEE

6.4.2 Descrição do Processo de Análise de Regras de Associação usando o *ARInE*

A interface do Ambiente *ARInE* está disponível no endereço 143.107.231.137/arine/index.html e é apresentada na Figura 6.22. Cabe lembrar que para completo acesso às funcionalidades do ambiente é necessário que o navegador esteja com a opção *Java Script* ativada.

Figura 6.22: Página inicial do Ambiente *ARInE*

Inicialmente, tem-se acesso à descrição geral do Ambiente *ARInE*, do processo de análise disponibilizado e das medidas objetivas possíveis de serem empregadas na análise, por meio dos *links* introdução, processo de análise e descrição das medidas, respectivamente. O acesso à essas funcionalidades é realizado sem a necessidade de *login* no sistema, para as demais funcionalidades do sistema é requisitada a autenticação do usuário, com login e senha.

Após a autenticação do usuário, por meio do *link* projeto e ruleset, devem ser escolhidos

o projeto e o conjunto de regras a serem analisados, como ilustrado na Figura 6.23. Existe a opção de recuperar uma seção de análise previamente realizada e salva, isso fará com que seja recuperado o conjunto de regras de foco do usuário – **focus-ruleset** - apresentado na metodologia descrita na Figura 6.1, página 141. Nesta página, também é possível visualizar as descrições de cada base de regra para o projeto atual, desde que tenham sido fornecidas pelo usuário que as inseriu no sistema RULEE.

Figura 6.23: Seleção de projeto e base de regras

Escolhidos o projeto e o conjunto de regras ou recuperada uma seção de análise, é informado na parte superior da página o projeto e o conjunto de regras “ativo” para análise e a descrição da base de regra aparece em destaque, na parte inferior da página, conforme mostrado na Figura 6.24.

A próxima etapa é a definição do **focus-ruleset**, ilustrada nas Figuras 6.25 e 6.26, que consiste de: uma ou mais consultas à base de regras, excluir regras não desejadas e salvar as regras que serão analisadas (**focus-ruleset**). Caso tenha sido recuperada uma seção de análise é possível ir diretamente para análise com as medidas objetivas (*link medidas*) ou, opcionalmente, visualizar e remover as regras do **focus-ruleset** (*link regras selecionadas*) e adicionar novas regras ao **focus-ruleset**, utilizando os mesmos procedimentos utilizados para a sua definição inicial.

A seleção de regras (acessada por meio do *link regras* da seção **Seleção**) é realizada na página apresentada na Figura 6.25, sendo possível escolher as características desejáveis

Figura 6.24: Projeto e base de regras selecionados

para as regras. Essas características estão relacionadas à presença de atributos, valores de atributos ou itens na regra. Essa presença pode ser considerada em todo o corpo da regra, apenas no antecedente ou somente no conseqüente. O exemplo apresentado na Figura 6.25 seleciona regras que possuam os atributos *ddpp550* e *ddpp240* no antecedente e, no conseqüente, valores *bom* ou *ótimo* para o atributo *mgt*.

O resultado da consulta é então apresentado, conforme mostrado na Figura 6.26, sendo possível salvar ou excluir as regras do processo de análise. Isso pode ser realizado para todas as regras de uma vez, ou de maneira individualizada, selecionando-se as caixas de verificação ao lado de cada regra. Ao salvar uma regra, esta é incluída no **focus-ruleset**, desde que não tenha sido excluída antes.

O processo de consultar e salvar/excluir é repetido até que o conjunto de regras para análise esteja definido e com pelo menos uma regra salva no **focus-ruleset**. Caso deseje-se analisar todas as regras de um conjunto de regras, deve-se utilizar a opção **selecionar todas**, na página apresentada na Figura 6.25, e depois **salvar todas**, na página representada na Figura 6.26. Essas restrições são adotadas por questões de otimização necessárias à etapa de avaliação gráfica.

Após definido o **focus-ruleset** pode-se selecionar as medidas para a análise utilizando os métodos disponibilizados. Essa operação é acessada por meio do *link* **medidas**, que dá

Location Edit View Go Bookmarks Tools Settings Window Help

Location: /arine2/index.html?session_id=37&project_code=ART&ruleset_id=7&username=edson&Load_Session_Name=Neiorea&page=load_session

ARInE Projeto Ativo RuleSet Ativo

ART 7

Especifique as características das regras

Monte sua consulta

Atributo ou Item mgt=bom mgt=otimo

Presente ☒ Na regra ☐ No antecedente (LHS) ☐ No conseqüente (RHS)

Contém Contém todos Contém qualquer AND OR ()

Especifique as restrições para consultar as regras

RULE CONTAINS ALL (ddpp550 ddpp240) AND RULE CONTAINS ANY (mgt=bom mgt=otimo)

Selecionar todas Executar consulta Limpar

Figura 6.25: Construção de consulta para seleção de regras

acesso à página apresentada na Figura 6.27. As seguintes funcionalidades estão disponíveis nesta opção:

- selecionar as medidas para análise, utilizando as caixas de verificação;
- obter descrição detalhada de cada medida, acessando o *link* associado ao nome da medida;
- realizar análises de similaridade entre as medidas, considerando os dados de valores das medidas para o **focus-ruleset** utilizando análise fatorial. O resultado obtido utilizando a análise fatorial é apresentado na Figura 6.28;
- definir os intervalos/percentuais para cada tipo de análise. Essa definição pode ser feita especificando-se os intervalos individualmente, como o exemplo na coluna **Todos** – no canto inferior direito da Figura 6.27 – que irá dividir os valores das medidas nos intervalos percentuais de 10, 25, 45, 80, 90. Um valor único de variação também pode ser informado, como o exemplo apresentado na coluna **Pareto** (canto inferior esquerdo da mesma figura), que irá produzir intervalos de 10 em 10%;
- analisar o conjunto de regras utilizando análise de Pareto, dispersão e frequência, conforme descritas na Seção 6.2.

Figura 6.26: Resultado da seleção de regras

Cabe lembrar que esta página é gerada dinamicamente, o que significa que se uma nova medida for inserida no repositório de medidas, a mesma será disponibilizada automaticamente.

Na Figura 6.28 observa-se o resultado da análise fatorial, com as medidas agrupadas conforme os fatores obtidos (cor de fundo), neste caso 4 fatores. Para as medidas em que não foi possível estabelecer o fator predominante, a cor de fundo permaneceu a mesma. Com esse agrupamento são criadas duas situações de análise: medidas semelhantes (pertencentes a um mesmo grupo) e medidas distintas (pertencentes a grupos distintos).

Para as medidas semelhantes, que são altamente correlacionadas, pode-se buscar por regras que não sigam este padrão de correlação. Uma vez que a regra não segue um padrão, pode ser que represente algum conhecimento inovador. De maneira semelhante esse tipo de análise pode ser feito para as medidas distintas.

Na sequência são apresentadas as telas (Figuras 6.29 a 6.31) com exemplos de resultados do uso de cada tipo de análise: dispersão, frequência e Pareto. A estrutura de apresentação das informações é semelhante para todas as análises. Na parte superior da página é apresentada uma tabela contendo, na primeira coluna, os intervalos em que os valores das medidas foram divididos e, nas demais colunas, para cada medida, o número e o percentual de regras para cada intervalo. Ao número e percentual de regras está associado um *link* para a apresentação das respectivas regras.

Figura 6.27: Seleção de medidas – tela inicial

Figura 6.28: Seleção de medidas – após análise fatorial

Na Figura 6.29 é apresentado um exemplo de resultado obtido com análise de dispersão. Como já mencionado, inicialmente é apresentada uma tabela com o número de regras para cada intervalo e medida. A seguir, para cada medida são apresentadas as informações:

1. valores limites da medida: mínimo e máximo teórico e no conjunto de regras ana-

lisados. Cabe lembrar que caso na definição da medida não tenha sido especificado seus limites teóricos, são utilizados os limites presentes no conjunto de regras;

2. histograma com o número de regras para cada intervalo especificado;
3. gráfico de dispersão, considerando, como limites, os valores da medida presentes no conjunto de regras;
4. gráfico de dispersão, considerando, como limites, os valores teóricos da medida.

Figura 6.29: Exemplo de resultado empregando análise de dispersão

Na Figura 6.30 é apresentado um exemplo de resultado para a análise de discretização. Como já mencionado, inicialmente é apresentada uma tabela com o número de regras para cada intervalo e medida. A tabela apresentada mostra adicionalmente uma coluna com o número (e percentual) de regras, resultado da combinação das medidas. A seguir, para cada medida são apresentados os histogramas, que permitem identificar rapidamente os intervalos com mais ou com menos regras.

Na Figura 6.31 é apresentado um exemplo de resultado para a análise de Pareto. Da mesma maneira que para o caso da análise de discretização, é apresentada uma tabela com o número de regras para cada intervalo e medida e, o resultado da combinação das

Figura 6.30: Exemplo de resultado empregando análise de discretização

medidas. Neste caso, os intervalos são definidos com a redução do percentual acumulado do valor da medida e o número de regras também é acumulado. A seguir, são apresentados os gráficos da análise Pareto para cada medida, que permitem identificar a região na qual a contribuição das regras é mais intensa (porção da curva com inclinação mais acentuada). No gráfico também é indicada, como referência, a linha correspondente ao percentual de 80% relacionado ao valor teórico do Princípio de Pareto. Ao lado do gráfico é disponibilizado um recurso adicional para exploração interativa de regras, no qual são especificados os limites percentuais superiores e inferiores do intervalo que se deseja observar as regras.

O recurso para visualização das regras para os três tipos de análises (Pareto, dispersão e frequência) é semelhante. Por meio de *links* associados a cada número de regras tem-se acesso à página para visualização de regras e avaliação final pelo especialista, como apresentado na Figura 6.32. Caso existam regras interessantes, estas podem ser salvas no conjunto de regras interessantes. As regras interessantes podem ser acessadas pelo *link* regras interessantes localizada no menu lateral.

Figura 6.31: Exemplo de resultado empregando análise de Pareto

Na seção **Utilitários** do menu lateral estão presentes as seguintes funcionalidades:

seção de análise: permite salvar ou recuperar uma seção de análise. A seção de análise é constituída pelo **focus-ruleset** e pelo conjunto de regras interessantes;

conjuntos de regras: permite inserir e excluir conjuntos de regras. Estas funcionalidades são executadas pelo Ambiente RULEE;

RULEE_GAR: módulo computacional, associado ao Ambiente RULEE, para generalização de Regras de Associação com o uso de taxonomias, descrito brevemente na Seção 4.6.5, página 95. Cabe lembrar que, a estrutura do repositório de regras utilizada, permite ao Ambiente *ARInE* tratar Regras de Associação generalizadas de modo idêntico às não-generalizadas.

Por fim, o *logout* do ambiente é realizado por meio do *link* **Finalizar seção**. Com este procedimento é encerrada a seção atual e são apagados arquivos e tabelas temporárias.

Location: [o_regras-348.chave_primaria=Added_Value%20Support0%2020%2040%2060%2080%20100&intervalo=2&page=view_rules_intersecao](#)

ARInE Projeto Ativo RuleSet Ativo
ART 7

Regras do conjunto 7 com filtro combinado para as medidas: Added_Value, Support

ID REGRA	DESCRICAO DA REGRA
3855	IF ddpp120=[-0.294-inf] & mgt=bom THEN ddpp365=[1.1-12.05]
3856	IF ddpp365=[1.1-12.05] THEN ddpp120=[-0.294-inf] & mgt=bom
3923	IF ddpp120=[-0.294-inf] & mgt=bom THEN ddpp240=[0.17-6.35]
3924	IF ddpp240=[0.17-6.35] THEN ddpp120=[-0.294-inf] & mgt=bom
3947	IF ddpp120=[-0.294-inf] & mgt=bom THEN ddpp550=[1.24-13.4]
3948	IF ddpp550=[1.24-13.4] THEN ddpp120=[-0.294-inf] & mgt=bom
3969	IF mgt=bom THEN ddpp550=[1.24-13.4]
3970	IF ddpp550=[1.24-13.4] THEN mgt=bom
3987	IF mgt=bom THEN ddpp120=[-0.294-inf] & ddpp240=[0.17-6.35]
3988	IF ddpp120=[-0.294-inf] & ddpp240=[0.17-6.35] THEN mgt=bom
3989	IF mgt=bom THEN ddpp240=[0.17-6.35]
3990	IF ddpp240=[0.17-6.35] THEN mgt=bom

Page loaded.

Figura 6.32: Exemplo de visualização de regras para um intervalo de análise

6.5 Considerações Finais

Neste capítulo foi apresentado o Ambiente *ARInE*, um Ambiente para Exploração de Regras desenvolvido especificamente para manipular Regras de Associação com o apoio de medidas objetivas. Além de implementar otimizações na exploração e visualização de regras, é provido pelo ambiente uma estrutura que viabiliza a metodologia de análise e avaliação para Regras de Associação proposta. A metodologia proposta para analisar regras, permite identificar pequenos grupos de regras, os quais são apresentados a um especialista para validação. A análise de Pareto é a principal técnica adotada para a identificação destes pequenos grupos. No sentido de garantir a independência do Ambiente *ARInE* dos algoritmos para extração de Regras de Associação e, principalmente, a integração com os demais projetos desenvolvidos no LABIC (DISCOVER e RULEE) foi desenvolvida e apresentada uma representação padronizada para Regras de Associação. Na seção final do capítulo foi descrita a implementação do *ARInE*, detalhando sua relação com o ambiente RULEE e o uso da metodologia por meio de sua interface Web.

Conclusões

A etapa de Pós-processamento do conhecimento tem despertado interesse como objeto de pesquisa em anos recentes. Esse interesse é motivado principalmente pela intensificação do uso do conhecimento extraído em aplicações práticas e pelo grande volume desse conhecimento que, em geral, inviabiliza sua análise manual. No caso da extração de padrões por Regras de Associação, a questão do grande volume de conhecimento gerado é uma problemática ainda mais expressiva, considerando que esta tarefa é conhecida por gerar um elevado número de regras.

Com o avanço da tecnologia de Mineração de Dados e sua utilização para solucionar problemas reais em processos de tomada de decisão, pode-se notar a importância da etapa de Pós-processamento do conhecimento, que tem papel fundamental no sucesso do processo como um todo. A utilização efetiva do conhecimento descoberto em um processo de apoio à tomada de decisão depende de fatores como a possibilidade do usuário compreender o conhecimento e conseguir identificar, por exemplo, o conjunto de regras mais interessante sob seu ponto de vista. Assim, a avaliação dos conjuntos de regras quanto a sua qualidade, compreensibilidade, utilidade e grau de interesse é de grande relevância para auxiliar o especialista na utilização do conhecimento descoberto.

Nesse sentido, o desenvolvimento de uma metodologia para identificar conhecimen-

to interessante por meio do pós-processamento de padrões extraídos adquire significado especial, principalmente no caso do uso de Regras de Associação.

Na etapa de Pós-processamento, o emprego de medidas objetivas para a avaliação de regras é tarefa usual, considerando que estas medidas fornecem uma indicação da força de associação (hipotética) entre os lados da regra. Contudo, de maneira geral, a maioria dos trabalhos limita-se a ordenar regras por uma ou mais medidas e apresentá-las ao especialista para avaliação.

No contexto apresentado, este trabalho investigou a tarefa Regras de Associação no processo de Mineração de Dados com enfoque nos aspectos relacionados ao Pós-processamento do conhecimento adquirido. Nessa perspectiva, buscou solucionar a problemática da análise do grande número de Regras de Associação.

7.1 Principais Contribuições

O objetivo principal desta tese é desenvolver e implementar uma metodologia para o Pós-processamento de Regras de Associação que permita a um especialista “navegar” por conjuntos de regras, concentrando-se em determinados grupos de regras. Nessa metodologia, pequenos grupos de regras potencialmente interessantes, devem ser apresentados ao usuário para avaliação. A definição destes grupos de regras interessantes é baseada no uso de medidas objetivas. Para auxiliar o especialista no uso da metodologia, foi proposto e desenvolvido um ambiente interativo, que viabiliza a exploração de regras seguindo a metodologia definida.

No decorrer do trabalho foi constatado não ser possível definir uma medida que apresente bom desempenho na análise de regras para todas as bases de dados e parâmetros utilizados nos algoritmos de extração de padrões. Assim, na metodologia proposta neste trabalho, são fornecidos métodos de apoio para seleção da(s) medida(s) mais adequada(s) para cada caso. Esses métodos podem ser categorizados em: métodos para análise de similaridade entre medidas e métodos para avaliação do desempenho da medida como filtro para seleção de regras.

Métodos para análise de similaridade entre medidas são necessários considerando a variedade de medidas disponíveis para avaliação de regras. Seria inviável verificar o desempenho de todas as medidas para cada base de regras analisada. Assim, foi proposto o uso da técnica de *análise fatorial* (Seção 4.5.2) para agrupar medidas altamente cor-

relacionadas. Com o uso dessa técnica é possível analisar apenas o desempenho de uma medida de cada grupo em vez de se analisar todas as medidas.

Para análise do desempenho da medida como filtro, é utilizada a **análise de Pareto** (Seção 4.5.1) que permite avaliar com o auxílio de gráficos, o potencial restritivo da medida e estabelecer um valor de “corte” para a seleção de regras usando esta medida. Métodos complementares para análise de medidas, também apoiados em gráficos, são providos de: **gráfico de dispersão**, para conhecimento geral da dispersão dos valores e **gráfico de barras**, para verificação da concentração de regras nas diferentes faixas de valores da medida.

Assim, destacam-se as principais contribuições desta tese:

- i) proposta de uma metodologia para avaliação exploratória e interativa de conhecimento representado por Regras de Associação. Essa metodologia permite ao usuário “navegar” num conjunto de regras utilizando técnicas baseadas em medidas objetivas para obter pequenos grupos de regras de interesse;
- ii) caracterização das métricas analisadas, quanto a capacidade para filtrar regras e quanto a manutenção desta capacidade perante diferentes bases de dados e parâmetros utilizados nos algoritmos de extração de padrões;
- iii) definição de uma sintaxe padrão para Regras de Associação incorporada aos ambientes DISCOVER e RULEE;
- iv) disponibilização de análises de regras generalizadas, utilizando interface com o módulo computacional RULEE_GAR. Os conjuntos de regras generalizadas também podem ser analisados por meio da metodologia proposta;
- v) disponibilização de uma biblioteca que transforma as Regras de Associação (geradas pelo algoritmo *Apriori* e pelos aplicativos *MineSetTM*, *MagnumOpus* e *WEKA*) para o formato padrão de Regras de Associação do DISCOVER;
- vi) otimização dos métodos de acesso ao repositório de regras do Ambiente RULEE, associados aos três tipos de regras suportadas pelo ambiente (classificação, regressão e associação);
- vii) viabilização de métodos otimizados para Pós-processamento de Regras de Associação no Ambiente DISCOVER;
- viii) proposição, desenvolvimento e implementação do Ambiente *ARInE*, no qual são disponibilizadas as contribuições descritas anteriormente.

Além das contribuições gerais apresentadas, destacam-se as contribuições específicas desta tese para a área de Mineração de Dados:

- I) Uso da análise de Pareto para definir os intervalos de valores de medidas utilizados na seleção de regras. Esse tipo de análise permite identificar as regras de maior contribuição para a medida. A identificação é realizada definindo-se um percentual limite de contribuição e esse percentual limite é identificado a partir dos gráficos gerados.
- II) Combinação de medidas para selecionar regras. A partir da análise de Pareto, são selecionados conjuntos de regras para cada medida. A intersecção destes conjuntos produz um conjunto de cardinalidade igual ou menor que os conjuntos iniciais, reduzindo assim a quantidade de regras. Considerando que as regras dos conjuntos iniciais são as mais representativas para uma dada medida, o conjunto resultante contém as regras mais representativas para as medidas que foram combinadas.

A tese aqui concluída sobre Pós-processamento de Regras de Associação é trabalho pioneiro no LABIC e inicia novo foco de pesquisa, que empresta contribuição aos trabalhos de doutorado, por Carvalho (2004), e de mestrado, desenvolvido por Domingues (2004) e em desenvolvimento por Pivato (2003), Sinoara (2004), Martins (2004).

7.2 *Trabalhos Futuros*

Além dos trabalhos em desenvolvimento relacionados e aqui motivados, são apresentadas algumas propostas de trabalhos futuros que visam complementar, incrementar e dar seqüência a esta tese:

- realizar validação da metodologia proposta utilizando bases de dados reais e especialistas do domínio;
- verificar se as medidas com Decréscimo Após Patamar podem ser empregadas como filtro utilizando concepção inversa da adotada para medidas com decréscimo inicial acentuado, ou seja, analisar as regras com os menores valores das medidas;
- analisar detalhadamente a classificação automática de medidas por meio de descritores de forma (momento e área da curva, por exemplo), no sentido de reduzir o grau de dependência da supervisão do usuário;

- realizar estudos empíricos para analisar a semelhança de medidas utilizando análise fatorial, com o objetivo de aumentar o grau de automatização das análises;
- implementar sistema de *log* detalhado no Ambiente *ARInE* para estabelecer relações entre características das bases de regras analisadas e os procedimentos adotados pelo usuário durante a análise;
- otimizar procedimentos ainda não alterados no Ambiente RULEE e nos repositórios de regras e medidas para tratar de maneira eficiente os conjuntos de Regras de Associação;
- estudar o significado semântico da combinação de medidas;
- implementar outros métodos de agrupamento de medidas;
- verificar empiricamente o resultado da aplicação da análise de Pareto sucessivamente ao mesmo conjunto de regras;
- implementar a análise das *regras excluídas* pelo usuário durante o processo de avaliação;
- estender as funcionalidades do módulo conversor de formatos para incorporar outros algoritmos implementados pela comunidade científica;
- incluir funcionalidades no Ambiente *ARInE* de maneira que seja possível realizar o processo de Mineração de Dados para Regras de Associação como um todo, em uma única interface.

Adicionalmente, pesquisas comparando a restritividade adotada para os filtros e a qualidade das regras geradas (avaliadas por usuários especialistas) e a avaliação da estabilidade das medidas para outras bases de dados, são etapas a serem desenvolvidas para complementar o estudo realizado.

Referências

- Adamic, L. A., & B. A. Huberman (2002). Zipf's law and the internet. *Glottometrics* 3, 143–150. Disponível em: <http://www.hpl.hp.com/research/idl/papers/ranking/adamicglottometrics.pdf> (último acesso: 26/08/2004), RAM-Verlag. 75
- Adamo, J.-M. (2001). *Data Mining for Association Rules and Sequential Patterns*. New York, NY: SV. 26, 27, 29, 31, 32, 33, 35, 36, 45, 60, 66
- Aggelis, V. (2004). Association rules model of e-banking services. In A. Zanasi, N. F. F. Ebecken, & C. A. Brebbia (Eds.), *Data Mining V*, pp. 46–55. WIT Press. 2
- Agrawal, R., T. Imielinski, & A. N. Swami (1993). Mining association rules between sets of items in large databases. In P. Buneman & S. Jajodia (Eds.), *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pp. 207–216. 3, 25, 26, 32, 35, 44, 55, 104, 159
- Agrawal, R., H. Mannila, R. Srikant, H. Toivonen, & I. Verkamo (1996). *Fast Discovery and Data Mining*, pp. 307–328. Menlo Park, CA: AAAI. 35
- Agrawal, R., & R. Srikant (1994). Fast algorithms for mining association rules. In J. B. Bocca, M. Jarke, & C. Zaniolo (Eds.), *Proceedings of the 20th International Conference on Very Large Data Bases, VLDB'94*, pp. 487–499. Morgan Kaufmann. 3, 26, 27, 31, 33, 35, 36, 37, 38, 41, 42, 43
- Agrawal, R., & R. Srikant (1995). Mining sequential patterns. In P. S. Yu & A. S. P. Chen (Eds.), *Eleventh International Conference on Data Engineering*, Taipei, Taiwan, pp. 3–14. IEEE Computer Society Press. 48
- Baesens, B., S. Viaene, & J. Vanthienen (2000). Post-processing of association rules. In *Proceedings of the special workshop on post-processing, The Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'2000)*, Boston, MA, USA, pp. 2–8. 2, 25, 59

- Baranauskas, J. A. (2001). *Extração Automática de Conhecimento por Múltiplos Indutores*. Ph. D. thesis, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. 89
- Baranauskas, J. A., & G. E. A. P. A. Batista (2000). O projeto Discover: Idéias iniciais (comunicação pessoal). 86
- Barroso, L. A., J. Dean, & U. Hölzle (2003). Web search for a planet: The google cluster architecture. *IEEE Micro* 23(2), 22–28. 1
- Batista, G. E. A. P. A. (2001). Sintaxe padrão do arquivo de exemplos do projeto DISCOVER. <http://www.icmc.usp.br/~gbatista/Discover/SintaxePadraoFinal.html> (último acesso: 21/01/2003). 88, 95
- Batista, G. E. A. P. A. (2003). *Pré-processamento de Dados em Aprendizado de Máquina Supervisionado*. Ph. D. thesis, Instituto de Ciências Matemáticas e de Computação – USP – São Carlos. 15, 89
- Batista, G. E. A. P. A., A. C. P. L. Carvalho, & M. C. Monard (2000). Applying one-sided selection to unbalanced datasets. In *Proceedings of the Mexican Congress on Artificial Intelligence MICAI, Lecture Notes in Artificial Intelligence*, pp. 315–325. 17
- Batista, G. E. A. P. A., & M. C. Monard (2003). Descrição da arquitetura e do projeto do ambiente computacional DISCOVERlearning environment – dle. Technical Report 187, Instituto de Ciências Matemáticas e de Computação – USP – São Carlos. Disponível em: ftp://ftp.icmc.usp.br/pub/BIBLIOTECA/rel_tec/RT_187.pdf (último acesso: 28/01/2004). 89
- Bayardo, R. J., R. Agrawal, & D. Gunopulos (1999). Constraint-based rule mining in large, dense databases. In *Proceedings of the 15th International Conference on Data Engineering, 23-26 March 1999, Sydney, Australia*, pp. 188–197. IEEE Computer Society. 47
- Becker, B., R. Kohavi, & D. Sommerfield (2002). Visualizing the simple bayesian classifier. In U. Fayyad, G. G. Grinstein, & A. Wierse (Eds.), *Information Visualization in Data Mining and Knowledge Discovery*, pp. 237–249. Morgan Kaufmann Publishers. 82
- Bernardini, F. C. (2002). Combinação de classificadores para melhorar o poder preditivo e descritivo de ensembles. Master’s thesis, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. 90
- Bernardini, F. C. (2003). Combinação de classificadores simbólicos utilizando medidas de regras de conhecimento. Qualificação de Doutorado, ICMC-USP. 90

- Blanchard, J., F. Guillet, & H. Briand (2003). Exploratory visualization for association rule rummaging. In *Proceedings of the 4th International Workshop on Multimedia Data Mining MDM'03 in conjunction with KDD'03*, pp. 107–114. 4
- Borgelt, C. (2003). Efficient implementations of Apriori and Eclat. In *Workshop Frequent Item Set Mining Implementations (FIMI 2003, Melbourne, FL, USA)*. Disponível em: http://fuzzy.cs.uni-magdeburg.de/~borgelt/papers/fimi_03.pdf (último acesso: 22/10/2004). 6, 104
- Breiman, L. (2000). Some infinity theory for predictor ensemble. Technical Report 577, University of California. 21, 23
- Brin, S., R. Motwani, J. D. Ullman, & S. Tsur (1997). Dynamic itemset counting and implication rules for market basket data. In J. Peckham (Ed.), *Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data*, pp. 255–264. 33
- Bruha, I., & A. F. Famili (2000). Postprocessing in machine learning and data mining. In *The Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 110–114. 3, 22
- Bruzzese, D., & C. Davino (2002). Visual post-analysis of association rules. In S. J. Simoff, M. Noirhomme-Fraiture, & M. H. Böhlen (Eds.), *2nd International Workshop on Visual Data Mining, 6th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD-02)*, Helsinki, Finland, pp. 55–66. 4
- Cai, C. H., A. W.-C. Fu, C. H. Cheng, & W. W. Kwong (1998). Mining association rules with weighted items. In *International Database Engineering and Application Symposium*, pp. 68–77. 50
- Carvalho, V. O. (2004). Generalização no processo de mineração de regras de associação. Monografia de Qualificação de Doutorado, Instituto de Ciências Matemáticas e de Computação – USP – São Carlos. 182
- Clementini, E., P. D. Felice, & K. Koperski (2000). Mining multiple-level spatial association rules for objects with a broad boundary. *Data & Knowledge Engineering* 34, 251–270. 2, 25
- Das, G., K.-I. Lin, H. Mannila, G. Renganathan, & P. Smyth (1998). Rule discovery from time series. In *Knowledge Discovery and Data Mining*, pp. 16–22. 49
- Dietterich, T. G. (2000). Ensemble methods in machine learning. *Multiple Classifier Systems 1857*, 1–15. 21
- Domingues, M. A. (2004). Generalização de regras de associação. Master's thesis, Instituto de Ciências Matemáticas e de Computação – USP – São Carlos. 60, 94, 95, 96, 97, 182

- Domingues, M. A., & S. O. Rezende (2004). Descrição de um algoritmo para generalização de regras de associação. Technical Report 228, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. Disponível em: ftp://ftp.icmc.usp.br/pub/BIBLIOTECA/rel_tec/RT_228.zip (último acesso: 22/10/2004). 97
- Domingues, M. A., S. O. Rezende, & M. F. Paula (2003). Implementação de taxonomias para regras de associação em um ambiente de pós-processamento. In *Anais do IV Workshop de Inteligência Artificial (ATAI 2003) das Jornadas Chilenas de Computacion*, pp. 1–6. 97
- Dong, G., & J. Li (1998). Interestingness of discovered association rules in terms of neighborhood-based unexpectedness. In *Proceedings of the 2nd Pacific-Asia Conference Knowledge Discovery and Data Mining, PAKDD'98*, Volume 1394 of *LNAI*, pp. 72–86. 59
- Dosualdo, D. G. (2003). Investigação de regressão no processo de mineração de dados. Master's thesis, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. 89, 90
- Dosualdo, D. G., & S. O. Rezende (2003a). Análise da precisão de métodos de regressão. Technical Report 197, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. Disponível em: ftp://ftp.icmc.usp.br/pub/BIBLIOTECA/rel_tec/RT_197.zip (último acesso: 27/01/2004). 88, 91
- Dosualdo, D. G., & S. O. Rezende (2003b). Descrição do ambiente computacional diper. Technical Report 204, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. Disponível em: ftp://ftp.icmc.usp.br/pub/BIBLIOTECA/rel_tec/RT_204.zip (último acesso: 27/01/2004). 90
- Evans, M., N. Hastings, & B. Peacock (1993). *Statistical distributions* (second ed.). New York: J. Wiley. 75
- Fayyad, U. M., D. Haussler, & Z. Stolorz (1996). KDD for science data analysis: Issues and examples. See Simoudis, Han, & Fayyad (1996), pp. 50–56. 2
- Fayyad, U. M., G. Piatetsky-Shapiro, & P. Smyth (1996a). From data mining to knowledge discovery in databases. *AI Magazine* 17(3), 37–54. 12
- Fayyad, U. M., G. Piatetsky-Shapiro, & P. Smyth (1996b). The KDD process for extracting useful knowledge from volumes of data. *CACM* 39(11), 27–34. 2, 13, 17, 19, 20, 24
- Fayyad, U. M., G. Piatetsky-Shapiro, & P. Smyth (1996c). Knowledge discovery and data mining: Towards a unifying framework. In *Proceedings of the 2nd International*

- Conference on Knowledge Discovery and Data Mining KDD'96*, pp. 82–88. AAAI Press. 12, 14
- Ferreira, A. B. H. (1999). *Novo Dicionário Aurélio - Século XXI*. Rio de Janeiro - RJ: Editora Nova Fronteira.
- Ferro, M. (2004). Aquisição de conhecimento de conjuntos de exemplos no formato atributo valor utilizando aprendizado de máquina relacional. Dissertação de Mestrado, ICMC-USP. 90
- Félix, L. C. M. (1998). Data mining no processo de extração de conhecimento de bases de dados. Master's thesis, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. 18
- Fonseca, G. L., & L. Ussher (2004). The history of economic thought website. <http://cepa.newschool.edu/het/profiles/pareto.htm> (último acesso: 26/08/2004). 74
- Foster, M., & A. G. Gee (2002). The data visualization environment. In U. Fayyad, G. G. Grinstein, & A. Wierse (Eds.), *Information Visualization in Data Mining and Knowledge Discovery*, pp. 87–93. Morgan Kaufmann Publishers. 82
- Freitas, A. A. (1998a). A multi-criteria approach for the evaluation of rule interestingness. In *Proceedings of the International Conference on Data Mining*, Rio de Janeiro, RJ, pp. 7–20. WIT Press. 23, 81
- Freitas, A. A. (1998b). On objective measures of rule surprisingness. In J. M. Zytkow & M. Quafafou (Eds.), *Principles of Data Mining and Knowledge Discovery*, Volume 1510 of *LNAI*, pp. 1–9. 59, 81
- Fu, A., M. Wong, S. Sze, W. Wong, W. Wong, & W. Yu (1998). Finding fuzzy sets for the mining of fuzzy association rules for numerical attributes. In *Proceedings of the First International Symposium on Intelligent Data Engineering and Learning (IDEAL'98)*, pp. 263–268. 48
- Fukuda, T., Y. Morimoto, S. Morishita, & T. Tokuyama (1996). Mining optimized association rules for numeric attributes. In *Proceedings of the fifteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, pp. 182–191. 48
- Geromini, M. R. (2002). Projeto e desenvolvimento da interface gráfica do sistema DISCOVER. Monografia do Exame de Qualificação de Mestrado, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. 87, 90
- Glymour, C., D. Madigan, D. Pregibon, & P. Smyth (1997). Statistical themes and lessons for data mining. *Data Mining and Knowledge Discovery* 1(1), 11–28. 17

- Goethals, B., J. V. Bussche, & K. Vanhoof (1998). Decision support queries for the interpretation of data mining results. Manuscript. 58
- Gonçalves, L. S. M. (2002). Categorização em text mining. Master's thesis, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. 90
- Gorsuch, R. L. (1974). *Factor analysis*. Philadelphia: Saunders. 76
- Grinstein, G. G., & M. O. Ward (2002). Introduction to data visualization. In U. Fayyad, G. G. Grinstein, & A. Wierse (Eds.), *Information Visualization in Data Mining and Knowledge Discovery*, pp. 21–45. Morgan Kaufmann Publishers. 81
- Hair, J. F., R. E. Anderson, R. L. Tatham, & W. C. Black (1998). *Multivariate Data Analysis*. Upper Saddle River, New jersey: Prentice Hall. 76, 78
- Han, J., & G. D. andYiwen Yin (1999). Efficient mining of partial periodic patterns in time series database. In *Proceedings of the 15th International Conference on Data Engineering, 23-26 March 1999, Sydney, Australia*, pp. 106–115. IEEE Computer Society. 49
- Han, J., & Y. Fu (1995). Discovery of multiple-level association rules from large databases. In U. Dayal, P. M. D. Gray, & S. Nishio (Eds.), *VLDB'95, Proceedings of 21th International Conference on Very Large Data Bases, September 11-15, 1995, Zurich, Switzerland*, pp. 420–431. Morgan Kaufmann Publishers Inc. 45
- Han, J., J. Pei, & Y. Yin (2000). Mining frequent patterns without candidate generation. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 1–12. ACM Press. 3, 32
- Hand, D., H. Mannila, & P. Smyth (2001). *Principles of Data Mining*. The Mit Press. 20
- Hilderman, R. J., & H. J. Hamilton (1999). Knowledge discovery and interestingness measures: a survey. Technical Report Report CS 99–04, Department of Computer Science, University of Regina. 59
- Hilderman, R. J., & H. J. Hamilton (2000). Applying objective interestingness measures in data mining systems. In D. Zighed & J. Komorowski (Eds.), *Proceedings of the 4th European Symposium on Principles of Data Mining and Knowledge Discovery*, Lyon, France, pp. 432–439. SV. LNAI. 4
- Hilderman, R. J., & H. J. Hamilton (2001). Evaluation of interestingness measures for ranking discovered knowledge. In *Proceedings of the 5th Pacific-Asia Conference on Knowledge Discovery and Data Mining PAKDD'01*, pp. 247–259. 15
- Hipp, J., U. Güntzer, & G. Nakhaeizadeh (2002). Data mining of association rules

- and the process of knowledge discovery in databases. In *Advances in Data Mining*, Volume 2394 of *LNAI*, pp. 15–36. 25, 32, 33
- Hipp, J., C. Mangold, U. Guntzer, & G. Nakhaeizadeh (2002). Efficient rule retrieval and postponed restrict operations for association rule mining. In M.-S. Cheng, P. S. Yu, & B. Liu (Eds.), *6th Pacific-Asia Conference on Knowledge Discovery and Data Mining PAKDD'02*, Volume 2336 of *LNCS*, pp. 52–65. 58
- Horst, P. S. (1999a). Avaliação do conhecimento adquirido por algoritmos de aprendizado de máquina utilizando exemplos. Master's thesis, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. 90
- Horst, P. S. (1999b). Avaliação do conhecimento adquirido por algoritmos de aprendizado de máquina utilizando exemplos. Master's thesis, ICMC-USP, São Carlos - SP. Dissertação de Mestrado, ICMC-USP. 59
- Houtsma, M., & A. Swami (1995). Set-oriented mining for association rules in relational databases. In P. S. Yu & A. L. P. Chen (Eds.), *Proceedings of the 11th International Conference on Data Engineering*, pp. 25–33. 32
- Hussain, F., H. Liu, & H. Lu (2000). Relative measure for mining interesting rules. In *4th European Conference on Principles and Practice of Knowledge Discovery in Databases PKDD'00 – Workshop in Knowledge Management: Theory and Applications*. Disponível em: http://eric.univ-lyon2.fr/~pkdd2000/Download/WS5_09.pdf (último acesso: 22/10/2004). 59
- Hussain, F., H. Liu, E. Suzuki, & H. Lu (2000). Exception rule mining with a relative interestingness measure. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining PAKDD'00*, pp. 86–97. 59
- Imamura, C. Y. (2001). Pré-processamento para extração de conhecimento de bases textuais. Master's thesis, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. 90
- Imieliński, T., & A. Virmani (1998). Association rules...and what's next? – towards second generation data mining systems. In W. Litwin, T. Morzy, & G. Vossen (Eds.), *Advances in Databases and Information Systems, Proceedings of the 2nd East European Symposium ADBIS'98*, Volume 1475 of *LNCS*, pp. 6–25. 56, 58
- Jaroszewicz, S., & D. A. Simovici (2002). Pruning redundant associations rules using maximum entropy principle. In M.-S. Chen, P. S. Yu, & B. Liu (Eds.), *Advances in Knowledge Discovery and Data Mining, 6th Pacific-Asia Conference, PAKDD'02*, Volume 2336 of *LNAI*, pp. 135–147. 59
- Jorge, A. (2004). Hierarchical clustering for thematic browsing and summarization of large sets of association rules. In M. W. Berry, U. Dayal, C. Kamath, &

- D. Skillicorn (Eds.), *Proceedings of the 4th SIAM International Conference on Data Mining SIAM'04*. Disponível em: <http://www.niaad.liacc.up.pt/~amjorge/Projectos/content-mgm/documents.html> (último acesso: 22/10/2004). 58, 60
- Jorge, A., J. Poças, & P. Azevedo (2002). A post-processing environment for browsing large sets of association rules. In S. J. Simoff, M. Noirhomme-Fraiture, & M. H. Böhlen (Eds.), *2nd International Workshop on Visual Data Mining, 6th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD-02)*, Helsinki, Finland, pp. 43–54. 4, 59
- Juran, J. (1974). The non-pareto principle; mea culpa. www.juran.com/pdf/SP7518.doc (último acesso: 26/08/2004). 74
- Klemettinen, M., H. Mannila, P. Ronkainen, H. Toivonen, & A. I. Verkamo (1994). Finding interesting rules from large sets of discovered association rules. In *Proceedings of the Third International Conference on Information and Knowledge Management*, pp. 401–407. 80
- Klemettinen, M., H. Mannila, & H. Toivonen (1996). Interactive exploration of discovered knowledge: A methodology for interaction, and usability studies. Technical Report C-1996-3, University of Helsinki, Finland. 80
- Klemettinen, M., H. Mannila, & H. Toivonen (1997). A data mining methodology and its application to semi-automatic knowledge acquisition. In *DEXA Workshop*, pp. 670–677. 80
- Kohavi, R., C. Brodley, B. Frasca, L. Mason, & Z. Zheng (2000). KDD-Cup 2000 organizers' report: Peeling the onion. *SIGKDD Explorations* 2(2), 86–98. Disponível em: <http://www.ecn.purdue.edu/KDDCUP> (último acesso: 22/10/2004). 106
- Kohavi, R., D. Sommerfield, & J. Dougherty (1997). Data mining using MLC⁺⁺: A machine learning library in C⁺⁺. *International Journal on Artificial Intelligence Tools* 6(4), 537–566. 21
- Korn, F., A. Labrinidis, Y. Kotidis, & C. Faloutsos (1998). Ratio rules: A new paradigm for fast, quantifiable data mining. In A. Gupta, O. Shmueli, & J. Widom (Eds.), *VLDB'98, Proceedings of 24rd International Conference on Very Large Data Bases, August 24-27, 1998, New York City, New York, USA*, pp. 582–593. Morgan Kaufmann. 50
- Kuok, C. M., A. Fu, & M. H. Wong (1998). Mining fuzzy association rules in databases. *SIGMOD Rec.* 27(1), 41–46. 48
- Lakshmanan, L. V. S., R. Ng, J. Han, & A. Pang (1999). Optimization of constrained frequent set queries with 2-variable constraints. In *Proceedings of the 1999 ACM SIGMOD international conference on Management of data*, pp. 157–168. 46

- Lavrac, N., P. Flach, & B. Zupan (1999). Rule evaluation measures: A unifying view. In S. Džeroski & P. Flach (Eds.), *Ninth International Workshop on Inductive Logic Programming (ILP'99)*, Volume 1634 of *Lecture Notes in Artificial Intelligence*, pp. 174–185. Springer-Verlag. Disponível em: <http://link.springer.de/link/service/series/0558/papers/1634/16340174.pdf> (último acesso: 27/01/2004). 59, 60, 63, 64, 65, 66, 69, 92
- Lee, H. D. (2000). Seleção e construção de features relevantes para o aprendizado de máquina. Master's thesis, Instituto de Ciências Matemáticas e de Computação – USP – São Carlos. 18, 90
- Lee, H. D. (2003). Seleção e transformação de atributos para a extração de conhecimento de bases de dados. Qualificação de Doutorado, ICMC-USP. 90
- Liu, B., & W. Hsu (1996). Post-analysis of learned rules. *AAAI 1*, 828–834. 15
- Liu, B., W. Hsu, S. Chen, & Y. Ma (2000). Analyzing the subjective interestingness of association rules. *IEEE Intelligent Systems & their Applications 15*(5), 47–55. 2, 23, 25, 44, 51, 59, 60, 72, 83, 84
- Liu, B., W. Hsu, & Y. Ma (1999). Pruning and summarizing the discovered associations. In *Proceedings of the 5th ACM SIGKDD international conference on Knowledge discovery and data mining KDD'99*, pp. 125–134. 59, 60
- Liu, B., M. Hu, & W. Hsu (2000). Multi-level organization and summarization of the discovered rules. In *Knowledge Discovery and Data Mining SIGKDD'00*, pp. 208–217. 60
- Ma, Y., B. Liu, & C. K. Wong (2000). Web for data mining: Organizing and interpreting the discovered rules using the web. *SIGKDD Explorations 2*(1), 16–23. 81, 82
- MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical statistics and probability 1*, 281–297. 127
- Mannila, H. (1996). Data mining: Machine learning, statistics, and databases. In *Statistical and Scientific Database Management*, pp. 2–9. 2, 24
- Mannila, H., & H. Toivonen (1996). Discovering generalized episodes using minimal occurrences. See Simoudis, Han, & Fayyad (1996), pp. 146–151. 49
- Mannila, H., H. Toivonen, & A. I. Verkamo (1995). Discovering frequent episodes in sequences. In U. M. Fayyad & R. Uthurusamy (Eds.), *First International Conference on Knowledge Discovery and Data Mining (KDD'95)*, Montreal, Canada, pp. 210 – 215. AAAI Press. 49

- Martins, C. A. (2003). *Uma abordagem para pré-processamento de dados textuais em algoritmos de aprendizado*. Ph. D. thesis, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. 89, 90
- Martins, C. D. (2004). Geração semi-automática de taxonomias para generalização de regras de associação. Monografia de Qualificação de Mestrado, Instituto de Ciências Matemáticas e de Computação – USP – São Carlos. 182
- Matsubara, E. T. (2004). O algoritmo de aprendizado semi-supervisionado CO-TRAINING e sua aplicação na rotulação de documentos. Dissertação de Mestrado, ICMC-USP. <http://www.icmc.usp.br/~edson/artigos/dissertacao.zip>. 89, 90
- Matsubara, E. T., C. A. Martins, & M. C. Monard (2003). PreText: uma ferramenta para pré-processamentos de textos utilizando a abordagem bag-of-words. Technical Report 209, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. Disponível em: http://www.icmc.usp.br/~biblio/BIBLIOTECA/rel_tec/RT_209.zip (último acesso: 27/01/2004). 89
- Melanda, E. A., & S. O. Rezende (2002). Medidas objetivas para análise e interpretação de regras de associação. In *16th Brazilian Symposium on Artificial Intelligence SBIA'02 – I Workshop de Teses e Dissertações em Inteligência Artificial WTDIA'02*. Printed on CD. 10p.
- Melanda, E. A., & S. O. Rezende (2003a). Pós-processamento de regras de associação com o uso de medidas objetivas. In *IV Workshop on Advances & Trends in AI for Problem Solving ATAI'03, In Jornadas Chilenas de Computacion*, pp. 1–6.
- Melanda, E. A., & S. O. Rezende (2003b). Sintaxe padrão para representar regras de associação. Technical Report 206, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. Disponível em: http://www.icmc.usp.br/~biblio/BIBLIOTECA/rel_tec/rt_206.pdf (último acesso: 27/01/2004). 97, 145
- Melanda, E. A., & S. O. Rezende (2003c). Uso combinado de medidas objetivas como filtro de regras de associação. In *Proceedings of the Fourth Congress of Logic Applied to Technology Conference Proceedings LAPTEC'03*, Volume 2, pp. 170–178.
- Melanda, E. A., & S. O. Rezende (2004a). Combining quality measures to identify interesting association rules. In *IX Ibero-American Conference on Artificial Intelligence, LANAI 3315.*, Puebla. Springer-Verlag. 136, 137
- Melanda, E. A., & S. O. Rezende (2004b). Investigação do comportamento gráfico de medidas para regras de associação. Relatório técnico, ICMC-USP, São Carlos. No prelo. 102, 109
- Mitchell, T. M. (1998). *Machine Learning*. MGH. 20

- Monard, M. C., C. W. Caulkins, J. A. Baranauskas, R. B. T. Oliveira, & S. O. Rezende (1998). Data preparation, reduction and prediction in the context of data mining. Technical Report 81, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. Disponível em: ftp://ftp.icmc.usp.br/pub/BIBLIOTECA/re1_tec/RT_81.pdf (último acesso: 22/10/2004). 11
- Nagai, W. A. (2000). Avaliação do conhecimento extraído de problemas de regressão. Master's thesis, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. 90
- Nawarecki, E., M. A. Valenta, & A. Zygmunt (2003). Mining association rules. In *Artificial Intelligence Methods AI-METH'03*, pp. 203–206. 60
- Netz, A., S. Chaudhuri, J. Bernhardt, & U. M. Fayyad (2000). Integration of data mining with database technology. In A. E. Abbadi, M. L. Brodie, S. Chakravarthy, U. Dayal, N. Kamel, G. Schlageter, & K.-Y. Whang (Eds.), *VLDB 2000, Proceedings of 26th International Conference on Very Large Data Bases, September 10-14, 2000, Cairo, Egypt*, pp. 719–722. Morgan Kaufmann. 2
- Ng, R. T., L. V. S. Lakshmanan, J. Han, & A. Pang (1998). Exploratory mining and pruning optimizations of constrained associations rules. *SIGMOD Rec.* 27(2), 13–24. 46
- Padmanabhan, B., & A. Tuzhilin (1999). Unexpectedness as a measure of interestingness in knowkdge discovery. *Decision Support Systems* 27(3), 303–318. 1, 3
- Park, J. S., M.-S. Chen, & P. S. Yu (1997). Using a hash-based method with transaction trimming for mining association rules. *IEEE Transactions on Knowledge and Data Engineering* 9(5), 813–825. 32
- Paula, M. F. (2003). Ambiente para exploração de regras. Master's thesis, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. 58, 90, 93, 95
- Pei, J., J. Han, & R. Mao (2000). CLOSET: An efficient algorithm for mining frequent closed itemsets. In D. Gunopulos & R. Rastogi (Eds.), *Proceedings of ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, pp. 21–30. 3, 32
- Piatetsky-Shapiro, G. (1991). *Discovery, Analysis, and Presentation of Strong Rules*, pp. 229–248. MIT. 57
- Piatetsky-Shapiro, G. (1999). The data-mining industry coming of ages. *IEEE Intelligent Systems* 14(6), 32–34. 1, 2, 25
- Piatetsky-Shapiro, G., & C. J. Matheus (1994). The interestingness of deviations. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining (KDD-94)*, pp. 23–36. 15

- Pivato, M. A. (2003). Mineração de regras de associação espaciais no ambiente discover. Monografia de Qualificação de Mestrado, Instituto de Ciências Matemáticas e de Computação – USP – São Carlos. 182
- Prati, R. C. (2003). O framework de integração do sistema DISCOVER. Master's thesis, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. 87, 90
- Prati, R. C., J. A. Baranauskas, & M. C. Monard (2001a). Extração de informações padronizadas para a avaliação de regras induzidas por algoritmos de aprendizado de máquina simbólico. Technical Report 145, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. Disponível em: ftp://ftp.icmc.usp.br/pub/BIBLIOTECA/rel_tec/RT_145.ps.zip (último acesso: 22/10/2004). 88, 91
- Prati, R. C., J. A. Baranauskas, & M. C. Monard (2001b). Uma proposta de unificação da linguagem de representação de conceitos de algoritmos de aprendizado de máquina simbólicos. Technical Report 137, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. Disponível em: ftp://ftp.icmc.usp.br/pub/BIBLIOTECA/rel_tec/RT_137.ps.zip (último acesso: 28/01/2004). 88, 89
- Pugliesi, J. B. (2004). *Pós-processamento de Regras de Regressão*. Ph. D. thesis, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. 5, 90
- Pugliesi, J. B., D. G. Dosualdo, & S. O. Rezende (2003). Sintaxe padrão para representar regras de regressão no DISCOVER. Technical Report 193, Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, SP, Brasil. Disponível em: ftp://ftp.icmc.usp.br/pub/BIBLIOTECA/rel_tec/RT_193.zip (último acesso: 27/01/2004). 88, 89
- Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann. 127
- Ramaswamy, S., S. Mahajan, & A. Silberschatz (1998). On the discovery of interesting patterns in association rules. In *Proceedings of the 24rd International Conference on Very Large Data Bases*, pp. 368–379. Morgan Kaufmann Publishers Inc. 49
- Rastogi, R., & K. Shim (1998). Mining rules with categorical and numeric attributes. In *Proceedings of the Fourteenth International Conference on Data Engineering, February 23-27, 1998, Orlando, Florida, USA*, pp. 503–512. IEEE Computer Society. 48
- Rathjens, D. (1996). *MineSet User's Guide*. Silicon Graphics, Inc. 60, 62
- Rezende, S. O. (2003). *Sistemas Inteligentes: Fundamentos e Aplicações* (1 ed.). Barueri, SP: Manole. 197

- Rezende, S. O., J. B. Pugliesi, E. A. Melanda, & M. F. Paula (2003). *Mineração de Dados* (1 ed.), Chapter 12, pp. 307–335. In Rezende (2003). 2, 12, 13, 19
- Savasere, A., E. Omiecinski, & S. B. Navathe (1995). An efficient algorithm for mining association rules in large databases. In *Proceedings of the 21th International Conference on Very Large Data Bases*, pp. 432–444. Morgan Kaufmann Publishers Inc. 50
- Savasere, A., E. Omiecinski, & S. B. Navathe (1998). Mining for strong negative associations in a large database of customer transactions. In *Proceedings of the Fourteenth International Conference on Data Engineering*, pp. 494–502. IEEE Computer Society. 50
- Scott, & L. Liu (1996). An object-oriented approach to multilevel association rule mining. In K. Barker & M. T. Özsu (Eds.), *Proceedings of 5th Intl. Conf. on Information and Knowledge Management (CIKM'96)*, New York, pp. 65–72. ACM. 45
- Semenova, T., M. Hegland, W. Graco, & G. Williams (2001). Effectiveness of mining association rules for identifying trends in large health databases. In F. J. Kurfess & M. Hilario (Eds.), *Workshop on Integrating Data Mining and Knowledge Management, The 2001 IEEE International Conference on Data Mining - ICDM'01*, San Jose, California, USA. Disponível em: cui.unige.ch/~hilario/icdm-01/DM-KM-Final/Semenova.pdf (último acesso: 22/10/2004). 2, 25
- Shen, L., & H. Shen (1998). Mining flexible multiple-level association rules in all concept hierarchies (extended abstract). In *Proceedings of the 9th International Conference on Database and Expert Systems Applications*, pp. 786–795. Springer-Verlag. 46
- Silberschatz, A., & A. Tuzhilin (1995). On subjective measures of interestingness in knowledge discovery. *Proceedings of the 1st International Conference on Knowledge Discovery and Data Mining KDD'95* 1, 275–281. 5, 22, 56, 59
- Silberschatz, A., & A. Tuzhilin (1996). What makes patterns interesting in knowledge discovery systems. *IEEE Transactions on Knowledge and Data Engineering* 8(6), 970–974. 56, 58, 59
- Simoudis, E., J. Han, & U. M. Fayyad (Eds.) (1996). *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*, Menlo Park, CA. AAAI: AAAI. 188, 193
- Sinoara, R. A. (2004). Avaliação subjetiva de regras de associação no ambiente rulee. Monografia de Qualificação de Mestrado, Instituto de Ciências Matemáticas e de Computação – USP – São Carlos. 182
- Srikant, R. (2001). Association rules: Past, present and future. ICCS 2001 Internatio-

- nal Workshop on Concept Lattice-based theory, methods and tools for Knowledge Discovery in Databases. Invited Talk. 25, 44
- Srikant, R., & R. Agrawal (1995). Mining generalized association rules. In *Proceedings of the 21th International Conference on Very Large Data Bases*, pp. 407–419. Morgan Kaufmann Publishers Inc. 45
- Srikant, R., & R. Agrawal (1996a). Mining quantitative association rules in large relational tables. In *Proceedings of the 1996 ACM SIGMOD international conference on Management of data*, pp. 1–12. 47, 48
- Srikant, R., & R. Agrawal (1996b). Mining sequential patterns: Generalizations and performance improvements. In P. M. G. Apers, M. Bouzeghoub, & G. Gardarin (Eds.), *Proc. 5th Int. Conf. Extending Database Technology, EDBT*, Volume 1057, pp. 3–17. Springer-Verlag. 48
- Srikant, R., & R. Agrawal (1997). Mining generalized association rules. *Future Generation Computer Systems* 13(2/3), 161–180. 26, 59
- Srikant, R., Q. Vu, & R. Agrawal (1997). Mining association rules with item constraint. In *Proceedings of the Third International Conference of Knowledge Discovery and Data Mining*, Newport Beach, California, pp. 67–73. 46
- Tan, P.-N., & V. Kumar (2000). Interestingness measures for association patterns: A perspective. In *The Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining KDD'00*. 30, 59
- Thearling, K., B. Becker, D. DeCoste, W. D. Mawby, M. Pilote, & D. Sommerfield (2002). Visualizing data mining models. In U. Fayyad, G. G. Grinstein, & A. Wierse (Eds.), *Information Visualization in Data Mining and Knowledge Discovery*, pp. 205–222. Morgan Kaufmann Publishers. 82
- Thomas, S., & S. Sarawagi (1998). Mining generalized association rules and sequential patterns using sql queries. In *Poster at the 4th International Conference on Knowledge Discovery and Data Mining (KDD-98)*, New York, USA. 45
- Till, D. (1996). *Teach Yourself Perl 5 in 21 Days*. Indianapolis, USA: Sams Publishing. 87
- Toivonen, H., M. Klemettinen, P. Ronkainen, K. Hatonen, & H. Mannila (1995). Pruning and grouping discovered association rules. Workshop Notes of the ECML'95 Workshop on Statistics, Machine Learning, and Knowledge Discovery in Databases, 47–52, MLnet. 59, 60
- Valery, D. Wijesekera, & J. Srivastava (1998). Pattern directed mining of sequence data. In R. Agrawal & P. Stolorz (Eds.), *Proceedings of the 4th Intl. Conf. on Knowledge*

- Discovery and Data Mining (KDD'98)*, New York, New York, pp. 51–57. ACM Press. 49
- Wang, K., L. Tang, J. Han, & J. Liu (2002). Top down fp-growth for association rule mining. In *Proceedings of the 6th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, pp. 334–340. Springer-Verlag. 3, 32
- Wang, K., S. H. W. Tay, & B. Liu (1998). Interestingness-based interval merger for numeric association rules. In *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD'98)*, New York City, New York, USA, pp. 121–128. 48, 60, 63
- Webb, G. I. (1995). OPUS: An efficient admissible algorithm for unordered search. *Journal of Artificial Intelligence Research* 3, 431–465. 3, 32, 162
- Weiss, S. M., & N. Indurkha (1998). *Predictive Data Mining: A Practical Guide*. San Francisco, CA: MK. 3, 13, 17, 18, 20
- Wikipedia (2004). Pareto principle. http://en.wikipedia.org/wiki/Pareto_principle (último acesso: 22/10/2004). Última visita em 26/08/2004. 75
- Witten, I. H., & E. Frank (1999). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. São Francisco, Califórnia: Morgan Kaufmann. Disponível em: <http://www.cs.waikato.ac.nz/~ml/weka/book.html> (último acesso: 22/10/2004). 159
- Zaki, M., & C. Hsiao (2002). Charm: An efficient algorithm for closed itemset mining. In R. Grossman, J. Han, V. Kumar, H. Mannila, & R. Motwani (Eds.), *2nd SIAM International Conference on Data Mining*. 32
- Özden, B., S. Ramaswamy, & A. Silberschatz (1998). Cyclic association rules. In *Proceedings of the Fourteenth International Conference on Data Engineering*, pp. 412–421. IEEE Computer Society. 49
- Zeng, Z., R. Kohavi, & L. Manson (2001). Real world performance of association rule algorithms. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, California, pp. 401–406. 21, 33
- Zhang, C., & S. Zhang (2002). *Association rule mining: models and algorithms*, Volume 2307 of *LNAI*. Springer. 35