Seleção de atributos importantes para a
extração de conhecimento de
bases de dados

Huei Diana Lee

Seleção de atributos importantes para a extração de conhecimento de bases de dados

Huei Diana Lee

Orientador: *Prof^a. Dr^a. Maria Carolina Monard*

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

USP – São Carlos
Dezembro/2005

Este documento foi preparado com o formatador de textos L^AT_EX. O sistema de citações de referências bibliográficas utiliza o padrão *Chicago* do sistema bib_TE_X. Alguns termos utilizados neste trabalho não foram traduzidos da língua inglesa para a portuguesa por serem amplamente aceitos e difundidos na comunidade acadêmica de aprendizado de máquina. Todos os endereços de Internet utilizados nas referências bibliográficas tiveram seu último acesso realizado no mês de setembro de 2005.

Resumo

O desenvolvimento da tecnologia e a propagação de sistemas computacionais nos mais variados domínios do conhecimento têm contribuído para a geração e o armazenamento de uma quantidade constantemente crescente de dados, em uma velocidade maior da que somos capazes de processar. De um modo geral, a principal razão para o armazenamento dessa enorme quantidade de dados é a utilização deles em benefício da humanidade. Diversas áreas têm se dedicado à pesquisa e a proposta de métodos e processos para tratar esses dados. Um desses processos é a Descoberta de Conhecimento em Bases de Dados, a qual tem como objetivo extrair conhecimento a partir das informações contidas nesses dados. Para alcançar esse objetivo, usualmente são construídos modelos (hipóteses), os quais podem ser gerados com o apoio de diferentes áreas tal como a de Aprendizado de Máquina.

A Seleção de Atributos desempenha uma tarefa essencial dentro desse processo, pois representa um problema de fundamental importância em aprendizado de máquina, sendo frequentemente realizada como uma etapa de pré-processamento. Seu objetivo é selecionar os atributos mais importantes, pois atributos não relevantes e/ou redundantes podem reduzir a precisão e a compreensibilidade das hipóteses induzidas por algoritmos de aprendizado supervisionado. Vários algoritmos para a seleção de atributos relevantes têm sido propostos na literatura. Entretanto, trabalhos recentes têm mostrado que também deve-se levar em conta a redundância para selecionar os atributos importantes, pois os atributos redundantes também afetam a qualidade das hipóteses induzidas. Para selecionar alguns e descartar outros, é preciso determinar a importância dos atributos segundo algum critério. Entre os vários critérios de importância de atributos propostos, alguns estão baseados em medidas de distância, consistência ou informação, enquanto outros são fundamentados em medidas de dependência.

Outra questão essencial são as avaliações experimentais, as quais representam um importante instrumento de estimativa de performance de algoritmos de seleção de atributos, visto que não existe análise matemática que permita prever qual algoritmo de seleção de atributos será melhor que outro. Essas comparações entre performance de algoritmos são geralmente realizadas por meio da análise do erro do modelo construído a partir dos subconjuntos de atributos selecionados por esses algoritmos. Contudo, somente a consideração desse parâmetro não é suficiente; outras questões devem ser consideradas, tal como a percentagem de redução da quantidade de atributos desses subconjuntos de atributos selecionados.

Neste trabalho é proposto um algoritmo que separa as análises de relevância e de redundância de atributos e introduz a utilização da Dimensão Fractal para tratar atributos redundantes em aprendizado supervisionado. É também proposto um modelo de avaliação de performance de algoritmos de seleção de atributos baseado no erro da hipótese construída e na percentagem de redução da quantidade de atributos selecionados. Resultados experimentais utilizando vários conjuntos de dados e diversos algoritmos consolidados na literatura, que selecionam atributos importantes, mostram que nossa proposta é competitiva com esses algoritmos. Outra questão importante relacionada à extração de conhecimento a partir de bases de dados é o formato no qual os dados estão representados. Usualmente, é necessário que os exemplos estejam descritos no formato atributo-valor. Neste trabalho também propomos uma metodologia para dar suporte, por meio de um processo semi-automático, à construção de conjuntos de dados nesse formato, originados de informações de pacientes contidas em laudos médicos que estão descritos em linguagem natural. Esse processo foi aplicado com sucesso a um caso real.

Abstract

Progress in computer systems and devices applied to a different number of fields, have made it possible to collect and store an increasing amount of data. Moreover, this technological advance enables the storage of a huge amount of data which is difficult to process unless new approaches are used. The main reason to maintain all these data is to use it in a general way for the benefit of humanity. Many areas are engaged in the research and proposal of methods and processes to deal with this growing data. One such process is Knowledge Discovery from Databases, which aims at finding valuable and interesting knowledge which may be hidden inside the data. In order to extract knowledge from data, models (hypothesis) are usually developed supported by many fields such as Machine Learning.

Feature Selection plays an important role in this process since it represents a central problem in machine learning and is frequently applied as a data pre-processing step. Its objective is to choose a subset from the original features that describes a data set, according to some importance criterion, by removing irrelevant and/or redundant features, as they may decrease data quality and reduce comprehensibility of hypotheses induced by supervised learning algorithms. Most of the state-of-art feature selection algorithms mainly focus on finding relevant features. However, it has been shown that relevance alone is not sufficient to select important features. Different approaches have been proposed to select features, among them the filter approach. The idea of this approach is to remove features before the model's induction takes place, based on general characteristics from the data set. For the purpose of selecting features and discarding others, it is necessary to measure the features' goodness, and many importance measures have been proposed. Some of them are based on distance measures, consistency of data and information content, while others are founded on dependence measures.

As there is no mathematical analysis capable of predicting whether a feature selection algorithm will produce better feature subsets than others, it is important to empirically evaluate the performance of these algorithms. Comparisons among algorithms' performance is usually carried out through the model's error analysis. Nevertheless, this sole parameter is not complete enough, and other issues, such as percentage of the feature's subset reduction should also be taken into account.

In this work we propose a filter that decouples features' relevance and redundancy analysis, and introduces the use of Fractal Dimension to deal with redundant features. We also propose a performance evaluation model based on the constructed hypothesis' error and the percentage of reduction obtained from the selected feature subset. Experimental results obtained using well known feature selection algorithms on several data sets show that our proposal is competitive with them. Another important issue related to knowledge extraction from data is the format the data is represented. Usually, it is necessary to describe examples in the so-called attribute-value format. This work also proposes a methodology to support, through a semi-automatic process, the construction of a database in the attribute-value format from patient information contained in medical findings which are described in natural language. This process was successfully applied to a real case.

*Aos meus pais,
Lee e Polly, Wu e Man Li*

À Maria Carolina Monard

*Ao meu marido,
Paulo.*

Agradecimentos Especiais

Meados de 1997... um amigo da época da graduação foi muito gentil em me acompanhar em minha primeira visita ao local onde eu teria a oportunidade de conhecer pessoas e realizar sonhos que marcariam minha vida.

Uma das principais pessoas nessa jornada é a professora Maria Carolina Monard. Ainda me lembro como se fosse há alguns dias atrás. Ela era, na época, orientadora dele de mestrado. Nunca imaginei como a professora Carolina seria importante em minha vida como um todo. Naquela oportunidade, também conheci alguns outros professores que seriam importantes nessa jornada.

No ano seguinte, iniciei o mestrado e tive a oportunidade de ter como minha orientadora a professora Carolina. Com o passar do tempo, percebi que aquela havia sido uma oportunidade única e que poucos haveriam de ter a sorte de ter como mentora uma pessoa como ela. Ela me guiou pelos caminhos de iniciação ao mundo da pesquisa... Os alunos da professora Carolina têm o privilégio de contarem com alguém que única e exclusivamente quer o seu bem e fará tudo que está ao seu alcance para que eles tenham as melhores oportunidades para o crescimento profissional e intelectual. Muitas foram as ocasiões em que desfrutamos dessa preocupação que a professora Carolina, juntamente com a professora Solange Rezende e o professor André Carvalho, têm para com a formação de seus alunos. Em muitas dessas ocasiões, eles abriram mão de seu conforto pessoal para que os alunos tivessem, por exemplo, a oportunidade de participar de congressos. Nunca poderemos agradecer todas as oportunidades que eles nos proporcionaram. Não poderia deixar de mencionar também a postura ética da professora Carolina, no trabalho e na vida. O exemplo por ela dado é inspirador.

A professora Carolina juntamente com os professores Ricardo Góes e João José Fagundes, orientadores de mestrado e doutoramento de meu marido, Wu, estiveram presentes e colaboraram em muitos momentos importantes, como na inauguração do LABI. Têm sido grandes incentivadores para todos os projetos profissionais e de vida que temos desenvolvido. Eles têm nos mostrado que o importante nessa jornada não é apenas o caminho, mas a paisagem que desfrutamos ao longo dessa caminhada que apenas se inicia...

Carolina, nunca terei palavras suficientes para demonstrar toda minha admiração por você e gratidão por todos esses anos em que me orientou, no mais amplo sentido da palavra, no mestrado e no doutoramento. Sempre a terei como meu exemplo de educadora a ser seguido. Se conseguirmos alcançar com nossos alunos apenas uma pequena parte do que você nos proporciona, teremos conseguido contribuir de alguma maneira para a formação deles.

A palavra *amigo* é definida no dicionário como:

s. m., o que quer bem; adj., favorável; aliado; afeiçoado; que tem amizade.

Sinto-me imensamente afortunada por tê-la, não somente como minha mentora, mas como minha amiga.

Agradecimentos

Não poderia deixar de agradecer ao meu estimado amigo que me acompanhou naquela primeira visita ao ICMC em 1997. Gustavo Batista me apresentou a sua orientadora, professora Carolina, e talvez na época, também não tivesse idéia de como ela faria diferença em nossas vidas. Obrigada pela amizade e apoio nos diversos momentos que compartilhamos.

Gostaria de agradecer Richardson Floriani Voltolini por todo apoio e incentivo que me proporcionou. É um privilégio trabalhar com você e acima de tudo tê-lo como meu amigo.

Gostaria também de agradecer Ronaldo Prati e Edson Matsubara, colegas notáveis e amigos estimados. Muito obrigada pelo apoio e incentivo. Ao Ronaldo, agradeço as diversas oportunidades que tivemos para discutir questões de meu trabalho.

Aos amigos e companheiros de todos os momentos de alegria e muito trabalho, Patrícia Rufino, Ana Carolina Lorena, Katti Faceli e Eduardo Spinosa, agradeço o incentivo e a amizade que tornaram essa jornada muito mais alegre e agradável.

Gostaria de agradecer também à professora Solange Rezende pelo apoio e incentivo durante todos esses anos.

Ao José Augusto Baranauskas, que apesar de não termos tido a oportunidade de trabalhar juntos durante o doutoramento, aprendi muito com ele durante meu mestrado.

Aos demais colegas do LABIC, com os quais compartilhei todos esses anos.

Gostaria também de agradecer Elaine Parros, sempre muito gentil e disposta a responder minhas dúvidas e com a qual tive o prazer de discutir e trocar muitas idéias sobre meu trabalho.

Agradecimentos também ao Humberto Razente por me auxiliar inicialmente com os esclarecimentos sobre o MDE.

Aos professores Ricardo Góes e João José Fagundes, que embora não tenham sido meus orientadores diretos no mestrado e no doutoramento, têm representado papéis fundamentais em minha formação e têm nos brindado com sua sabedoria e amizade.

À Ana Cristina de Moraes e ao Willian Adalberto Silva, amigos queridos, obrigada pelo incentivo e apoio.

Agradecimentos a Beth, Laura, Ana (Seção de Pós-graduação) e Marília (Seção de Eventos), por sempre serem prestativas e eficientes no tratamento dos assuntos da pós. Agradecimentos também pela estimada amizade.

Agradecimentos também a Maria Lima, Rose Zambon, Rose Casali e Sandra Soligon (Biblioteca Prof. Achille Bassi) por sempre serem prestativas nas diversas oportunidades.

Ao professor Dorival Leão pelos diversos esclarecimentos sobre estatística.

Ao amigo e colega Renato Machado pelo apoio e incentivo.

Ao amigo e colega Juan Carlos Sotuyo, homem de grandes qualidades: coragem e visão para o futuro, obrigada pelo incentivo e amizade.

Ao amigo e colega Jorge Habib Hanna El Khouri, pilar da razão, obrigada pelo apoio e amizade.

Ao amigo e colega, Antonio Marcos Hachisuca, pelo incentivo e amizade nestes anos todos.

Aos “meninos” do LABI: Daniel Honorato, Andrés Ferrero, Maksoel Niz, Joylan Maciel, Eduardo Burin e André Maletzke, pelo apoio e incentivo. Sinto muito por não estar mais presente.

Aos amigos Letícia Peres e Fabiano Silva pelo incentivo e amizade.

Ao amigo e colega Sérgio Dalmás, pela amizade e apoio.

Aos amigos e colegas Annete Faesarella e Antonio Ruiz Nogueira pelo incentivo e amizade.

À Neide pelo apoio de sempre.

À Unioeste e demais colegas, agradecimentos pelo apoio.

Ao ITAI — Instituto de Tecnologia em Automação e Informática e ao PTI – Parque Tecnológico de Itaipu que me apoiaram em diversas oportunidades.

À D.Teresinha, pelo enorme carinho que sempre teve comigo.

À Lurdes Pereira por me auxiliar a cuidar de casa nos longos períodos em que estive fora.

Ao Sandro Esteves pela parceria e pelos trabalhos que desenvolvemos em conjunto.

Ao Paulo Feng Chung Wu pelo apoio no desenvolvimento dos diversos trabalhos e projetos.

E finalmente, aos meus pais queridos, Lee e Polly, Wu Fu Tai e Man Li, pela confiança e apoio dados desde meus primeiros anos de vida. O incentivo de vocês foi fundamental para que pudéssemos construir nossas bases e alcançar nossos objetivos. Vocês são nossos exemplos de vida.

Aos meus irmãos Linda, Jen e mais recentemente Chiang, pelo incentivo e apoio nesses anos todos.

Ao meu querido marido e amigo, Paulo, pelo apoio e amor incondicionais, sem o quais não poderia ter chegado a essa etapa da jornada. Meu companheiro querido de todos os momentos agradeço pelo incentivo e suporte e com quem tenho a sorte e o prazer de compartilhar essa caminhada.

Sumário

Resumo	v
Abstract	vii
Dedicatória	ix
Agradecimentos Especiais	xi
Agradecimentos	xiii
Sumário	xv
Lista de Figuras	xix
Lista de Tabelas	xxi
Lista de Abreviaturas, Algoritmos e Variáveis	xxv
1 Introdução	1
1.1 Objetivos	5
1.2 Principais Contribuições	6
1.3 Organização do Trabalho	8
2 Terminologia	11
2.1 Considerações Iniciais	11
2.2 Definição da Notação Utilizada neste Trabalho	11
2.3 Definição das Operações Realizadas sobre os Atributos	13
2.4 Considerações Finais	15
3 Seleção de Atributos	17

3.1	Considerações Iniciais	17
3.2	O Problema da Seleção de Atributos	17
3.3	Seleção de um Subconjunto de Atributos como um Problema de Busca . . .	19
3.4	Abordagens para a Seleção de Atributos	24
3.4.1	Abordagem <i>Embedded</i>	24
3.4.2	Abordagem Filtro	25
3.4.3	Abordagem <i>Wrapper</i>	26
3.5	Considerações Finais	27
4	Importância de Atributos	29
4.1	Considerações Iniciais	29
4.2	Medidas de Avaliação de Atributos	29
4.2.1	Importância em Relação à Consistência	32
4.2.2	Importância em Relação à Dependência	33
4.2.3	Importância em Relação à Informação	40
4.2.4	Importância em Relação à Distância	41
4.2.5	Importância em Relação à Precisão	42
4.3	Dimensões da Seleção de Atributos	44
4.4	Considerações Finais	45
5	O Algoritmo FDimBF — <i>Fractal Dimension-Based Filter</i>	47
5.1	Considerações Iniciais	47
5.2	Fractais	47
5.3	Dimensão Fractal de um Conjunto de Dados	48
5.4	Descrição do Algoritmo <i>Fractal Dimension-Based Filter</i>	50
5.5	Considerações Finais	55
6	Avaliação Experimental	57
6.1	Considerações Iniciais	57
6.2	Descrição dos Conjuntos de Dados	58
6.3	Algoritmos Utilizados	60
6.4	Configuração dos Experimentos	64
6.5	Resultados e Discussão	67

6.5.1	Dimensão Fractal e Comportamento Fractal dos Conjuntos de Dados	67
6.5.2	Subconjuntos de Atributos Seleccionados	71
6.5.3	Formatos Aproximados de Distribuição dos Valores dos Atributos em Relação aos Atributos Seleccionados pelo Algoritmo FDimBF . .	75
6.5.4	Modelo de Performance dos Algoritmos em Relação à Precisão e a Quantidade de Atributos Seleccionados	76
6.5.5	Análise da Significância Estatística dos Resultados	81
6.5.6	Características dos Conjuntos de Dados Associadas à Utilização da Dimensão Fractal como uma Medida Adequada para a Seleção de Atributos	84
6.5.6.1	Características Gerais dos Conjuntos de Dados e Adequa- ção do Uso dos Algoritmos FDimBF	86
6.5.6.2	Padrões Encontrados na Aplicação dos Algoritmos FDimBF para os Conjuntos de Dados	88
6.6	Considerações Finais	90
7	Estudo de Caso	91
7.1	Considerações Iniciais	91
7.2	Etapas Realizadas para o Desenvolvimento do Estudo de Caso	92
7.3	Análise Seminal e Processamento de Sêmen Diagnóstico — Etapa 1	92
7.4	Coleta de Dados — Etapa 2	95
7.5	Metodologia para Construção de Bases de Dados a Partir de Laudos Médicos	96
7.5.1	Primeira Fase	97
7.5.2	Segunda Fase	100
7.6	Aplicação da Metodologia aos Laudos de Análise Seminal Completa — Etapa 3	101
7.7	Limpeza e Preparação dos Dados — Etapa 4	103
7.8	Seleção de Atributos — Etapa 5 — e Construção dos Modelos — Etapa 6 .	107
7.9	Avaliação dos Modelos: Resultados e Discussão — Etapa 7	108
7.9.1	Dimensão Fractal e Comportamento Fractal do Conjunto de Dados	108
7.9.2	Subconjuntos de Atributos Seleccionados	109
7.9.3	Formatos Aproximados de Distribuição dos Valores dos Atributos em Relação aos Atributos Seleccionados pelo Algoritmo FDimBF . .	114

7.9.4	Performance dos Algoritmos em Relação à Precisão e a Quantidade de Atributos Seleccionados	115
7.9.5	Avaliação dos Modelos Construídos	116
7.9.5.1	Medidas de Avaliação	116
7.9.5.2	Avaliação dos Modelos Utilizando Medidas Objetivas de Regras	119
7.9.5.3	Avaliação dos Modelos por Especialistas do Domínio . . .	120
7.10	Considerações Finais	125
8	Conclusão	127
8.1	Limitações	131
8.2	Trabalhos futuros	132
A	Distribuição dos Valores dos Atributos	135
B	Avaliação das Regras por meio de Medidas Objetivas e Subjetiva	139
	Referências Bibliográficas	145

Lista de Figuras

1.1	Fases do processo de descoberta de conhecimento	2
2.1	Hierarquia de tipos de atributos	12
2.2	Hierarquia de operações envolvendo atributos	14
3.1	Exemplo de espaço de estados de atributos	20
3.2	Abordagem <i>Embedded</i>	25
3.3	Abordagem <i>Filtro</i>	25
3.4	Abordagem <i>Wrapper</i>	26
4.1	Hierarquia de tipos de medidas de avaliação de atributos	31
4.2	As três principais dimensões da seleção de atributos: estratégia de busca, medidas de avaliação e direção da busca	45
5.1	Triângulo de Sierpinsky	48
5.2	Construção do Triângulo de Sierpinsky	49
5.3	Modelo para seleção de atributos (Yu and Liu, 2004)	50
5.4	Visão geral do Algoritmo FDimBF	53
5.5	Seleção de atributos utilizando o algoritmo FDimBF para conjuntos de dados contendo atributos numéricos e nominais	55
5.6	Localização do algoritmo FDimBF dentro do espaço de características dos métodos de seleção de atributos	56
6.1	Configuração dos experimentos	65
6.2	Gráfico gerado utilizando o método <i>Box Count Plot</i> — Hungarian	68
6.3	Gráfico gerado utilizando o método <i>Box Count Plot</i> — Waveform	69
6.4	Número de atributos selecionados e a respectiva percentagem <i>versus</i> o al- goritmo de SA	74

6.5	Tipos de formatos aproximados das distribuições dos valores dos atributos	75
6.6	Relação entre percentagem de atributos selecionados, média do erro e erro padrão dos modelos construídos: (a) Modelo geral e (b) Conjunto de dados Pima	78
7.1	Exemplo de laudo de análise seminal completa	95
7.2	Metodologia proposta	97
7.3	Local e característica	99
7.4	Estrutura base do dicionário	100
7.5	Construção da base de dados	100
7.6	Gráfico gerado utilizando o método <i>Box Count Plot</i> para o conjunto de dados de processamento de sêmen utilizando atributos selecionados por FDimBF(1)	109
7.7	Gráfico gerado utilizando o método <i>Box Count Plot</i> para o conjunto de dados de processamento de sêmen utilizando atributos selecionados por FDimBF(2)	110
7.8	Representação gráfica: relação entre percentagem de atributos selecionados, média do erro e erro padrão dos modelos construídos; A: $\mathcal{C}4.5$, B: ReliefF, C: CFS, D: FCBF, E: CBF, G: FDimBF(2) e T: sem SA.	115
A.1	Distribuições dos valores dos atributos — A	135
A.2	Distribuições dos valores dos atributos — B	136
A.3	Distribuições dos valores dos atributos — C	137

Lista de Tabelas

2.1	Formato padrão do conjunto de exemplos	12
3.1	Combinações de estratégias e direções de busca.	22
3.2	Amostra de exemplos para ilustrar o paradigma da busca heurística para a SA	24
4.1	Exemplos para ilustrar as definições de importância	34
4.2	Exemplos gerados pela função de paridade	34
4.3	Atributos importantes e não importantes	35
4.4	Exemplos para ilustrar a redundância de atributos	37
4.5	Exemplos do conceito meta $Y = X_1 \oplus X_2$	39
4.6	Atributos importantes e não importantes segundo as definições de importância	39
4.7	Exemplos para ilustrar que um atributo importante não é necessariamente ótimo	43
6.1	Resumo dos conjuntos de dados	60
6.2	Características dos algoritmos de SA	64
6.3	Informações associadas à dimensão fractal dos conjuntos de dados	70
6.4	Resultado da análise dos gráficos de comportamento dos conjuntos de dados quanto à característica fractal	71
6.5	Resumo da quantidade de atributos selecionados por cada um dos algoritmos e suas respectivas percentagens	73
6.6	Formatos da distribuição aproximada dos valores dos atributos	76
6.7	Média de erro e erro padrão para cada conjunto de dados e cada algoritmo considerados	77
6.8	Algoritmos presentes nos gráficos	80

6.9	Classificação dos algoritmos em relação a percentagem de atributos selecionados <i>versus</i> erro do modelo construído	80
6.10	Siglas para os conjuntos de dados	82
6.11	Comparação entre o número original de atributos e o número de atributos selecionados pelos algoritmos de SA. Comparação entre médias de erros dos modelos construídos (em negrito resultados estatisticamente significativos)	82
6.12	Comparação entre os números de atributos selecionados pelos algoritmos de SA. Comparação entre as médias de erros dos modelos construídos (em negrito resultados estatisticamente significativos)	83
6.13	Resumo do número de vezes em que cada algoritmo seleciona um subconjunto menor de atributos	84
6.14	Descrição dos atributos da meta-base	85
6.15	Resumo das meta-bases	86
7.1	Exemplo de padronização	98
7.2	Exemplo de padronização - Análise Seminal Completa	102
7.3	Atributos identificados para compor a base de dados	103
7.4	Resumo do conjunto de dados Processamento de Sêmen	105
7.5	Atributos do conjunto de dados utilizado nos experimentos	107
7.6	Informações associadas à dimensão fractal do conjunto de dados de processamento de sêmen	109
7.7	Atributos selecionados por cada um dos algoritmos para o conjunto de dados de processamento de sêmen diagnóstico	111
7.8	Número de vezes que cada atributo apareceu no modelo construído	113
7.9	Número de atributos para cada tipo de formato aproximado de distribuição	114
7.10	Número de atributos para cada tipo de formato aproximado de distribuição por subconjunto de atributos	114
7.11	Média do erro e erro padrão para o conjunto de dados processamento de sêmen — ECM = 40,05%	115
7.12	Matriz de contingência para uma regra	117
7.13	Matriz de contingência com frequências relativas para uma regra	118
7.14	Avaliação dos modelos utilizando medidas objetivas	119
7.15	Avaliação das regras dos modelos pelos especialistas do domínio	122
B.1	Medidas objetivas e subjetiva — Conjunto original de atributos	140

B.2	Medidas objetivas e subjetiva — Subconjunto de atributos selecionado por $\mathcal{C}4.5$	140
B.3	Medidas objetivas e subjetiva — Subconjunto de atributos selecionado por ReliefF	141
B.4	Medidas objetivas e subjetiva — Subconjunto de atributos selecionado por CFS	141
B.5	Medidas objetivas e subjetiva — Subconjunto de atributos selecionado por FCBF	142
B.6	Medidas objetivas e subjetiva — Subconjunto de atributos selecionado por CBF	142
B.7	Medidas objetivas e subjetiva — Subconjunto de atributos selecionado por FDimBF(2)	143

Lista de Abreviaturas, Algoritmos e Variáveis

Abreviaturas	
AM	Aprendizado de Máquina
AS	Análise Seminal
ASC	Análise Seminal Completa
DF	Dimensão Fractal
D_2	Dimensão Fractal de Correlação D_2
DLE	<i>Discover Learning Environment</i>
DOL	<i>Discover Object Library</i>
DSX	<i>Discover Dataset Syntax</i>
ECM	Erro da Classe Majoritária
FDR	<i>Fractal Dimension Reduction</i>
FIV	Fertilização <i>In Vitro</i>
HOC	Hiper-estimulação Ovariana Controlada
IA	Inteligência Artificial
IAE	Índice de Avaliação do Especialista
ICSI	Injeção Intracitoplasmática do Espermatozoide no Óvulo
KDD	<i>Knowledge Discovery on Databases</i>
LiBOC	<i>Linear Box-Occupancy Counter</i>
MD	Mineração de Dados
MDE	<i>Measure Distance Exponent</i>
MDL	<i>Minimum Description Length</i>
OMS	Organização Mundial da Saúde
pD	Dimensão Fractal Parcial
RA	Reprodução Assistida
RBD	Registro na Base de Dados
SA	Seleção de Atributos
SSA	Seleção de um Subconjunto de Atributos
SU	<i>Symmetrical Uncertainty</i>
TDIDT	<i>Top Down Induction of Decision Trees</i>

Algoritmos	
C4.5	Algoritmo para indução de árvores de decisão C4.5
CBF	<i>Consistency-based Filter</i>
CFS	<i>Correlation-based Feature Selection</i>
FCBF	<i>Fast Correlation-based Filter</i>
FDimBF	<i>Fractal Dimension-Based Filter</i>
FDimBF(1)	<i>Fractal Dimension-Based Filter - medida de informação</i>
FDimBF(2)	<i>Fractal Dimension-Based Filter - medida de distância</i>

ReliefF	Algoritmo para SA ReliefF
Variáveis	
M	Número de atributos
N	Número de exemplos
m	Parâmetro do algoritmo ReliefF
5.1 Algoritmo <i>Fractal Dimension-Based Filter</i> — FDimBF	51
5.2 Algoritmo <i>AtributosNa-</i> <i>oRedundantes</i>	52

Capítulo 1

Introdução

O desenvolvimento e a utilização de tecnologias para a aquisição e o armazenamento de dados, nas mais diversas áreas do conhecimento, têm permitido o acúmulo de dados em uma velocidade maior que a capacidade humana possui para processá-los. De um modo geral, a principal razão para o armazenamento dessa enorme quantidade de dados é a utilização deles em benefício da humanidade. Diversos métodos têm sido propostos para o processamento desses dados com o objetivo de extrair conhecimento da informação contida nessas bases de dados.

A forma mais simples de representação de dados é realizada por meio de atributos. O tamanho do conjunto de dados pode ser medido em duas dimensões: o número de atributos (M) e o número de exemplos (N). Ambos M e N podem ser muito grandes, dificultando a aplicação direta de métodos para a construção de modelos que representam o conhecimento embutido nessas bases de dados. Para que esses dados brutos possam tornar-se úteis, é necessário que eles sejam representados de maneira apropriada, processados e o modelo construído, avaliado e validado. Uma das maneiras de se alcançar esse objetivo é por meio da realização do processo de descoberta de conhecimento de bases de dados (*Knowledge Discovery on Databases — KDD*) (Fayyad et al., 1996b). Esse processo pode ser, basicamente, dividido em três fases, como ilustrado na Figura 1.1:

1. pré-processamento de dados;
2. mineração de dados e
3. pós-processamento de conhecimento.

A primeira fase, pré-processamento, possui, essencialmente, dois objetivos: conhecer o domínio da aplicação e os dados e prepará-los para a próxima fase. Entre as diversas tarefas realizadas nessa fase pode-se citar: preparação de dados, limpeza de dados, transformação de dados e atributos e Seleção de Atributos — SA. Nessa fase é importante, ainda, obter um conhecimento preliminar dos dados, o qual pode ser alcançado utilizando métodos de visualização de dados e/ou métodos simples de estatística, como medidas de

Figura 1.1: Fases do processo de descoberta de conhecimento (Baranauskas, 2001)

médias e desvios-padrão (Michalski et al., 1998). Esse entendimento prévio dos dados pode auxiliar na seleção de algoritmos mais apropriados para a próxima fase, a mineração de dados (Rezende et al., 2003; Witten and Frank, 2000). Desse modo, a fase de pré-processamento de dados é de fundamental importância para assegurar que os dados sejam de boa qualidade e apropriados para realizar, na prática, a mineração de dados (Bastista, 2003; Hand et al., 2001). Essa fase é tida como uma das tarefas mais trabalhosas e demoradas de KDD. É considerado que aproximadamente 80% do tempo despendido no processo de KDD seja utilizado para pré-processar os dados (Pyle, 1999).

A segunda fase, Mineração de Dados — MD — tem como objetivo principal a construção de modelos que possam representar o conhecimento embutido nos dados (Han and Kamber, 2000; Weiss and Indurkha, 1998). Essa fase pode ser apoiada por diversas áreas, entre as quais Aprendizado de Máquina — AM —, Base de Dados, Visualização e Estatística. Assim, resultados expressos em formatos diversos podem ser produzidos dependendo dos métodos escolhidos para a realização da MD.

Na última fase, pós-processamento, o objetivo é avaliar, validar e consolidar o conhecimento extraído. A avaliação é realizada com a interpretação dos resultados por meio, por exemplo, de visualização dos padrões extraídos, remoção de padrões irrelevantes ou redundantes e tradução de padrões úteis para formas compreensíveis para os usuários. Durante essa fase, os resultados devem ainda ser avaliados para garantir que eles são estatisticamente significativos e confiáveis. O conhecimento extraído deve ser também validado com relação ao conhecimento prévio do domínio para que possíveis conflitos sejam removidos. Finalmente, o conhecimento extraído é consolidado incorporando-o a sistemas de apoio à tomada de decisão, ou por meio da documentação desse conhecimento para o usuário.

Todas essas três fases são importantes para que o processo de KDD seja realizado com sucesso. No entanto, inicialmente, os esforços de pesquisadores dessa área estavam concentrados, principalmente, na tarefa ou processo de construção do modelo utilizando algoritmos de aprendizado supervisionado dos mais diversos paradigmas, nos quais os dados são rotulados com os valores de um atributo que especifica a classe a qual pertencem.

Uma das maiores preocupações estava concentrada na precisão dos modelos construídos por esses algoritmos. Mais recentemente, a comunidade de extração de conhecimento tem-se voltado com maior atenção para as fases de pós e pré-processamento. Como mencionado, tarefas de pré-processamento incluem quaisquer operações sobre os dados antes do processo de construção do modelo propriamente dito. Sendo assim, um dos mais importantes exemplos de tarefas realizadas durante essa fase inclui a seleção de atributos, o qual representa o principal objeto de estudo deste trabalho.

Pode-se considerar a seleção de atributos sob dois aspectos:

Qual o resultado da seleção de atributos?

Por que realizar a seleção de atributos?

Combinando-se ambas as considerações, a SA pode ser definida como um processo de escolha de um subconjunto ótimo de atributos, que representa a informação importante contida nos dados, segundo algum critério (Liu and Motoda, 1998). Esse subconjunto de tamanho P pode ser de tamanho igual ou menor ao conjunto de atributos original de tamanho M . A seleção de atributos permite, por exemplo, a ordenação de atributos de acordo com algum critério de importância, a redução da dimensionalidade do espaço de busca de atributos e a remoção de dados contendo ruídos, entre outros. A seleção de atributos pode ser particularmente importante, por exemplo, em casos nos quais a medição de certos atributos é custosa, pois pode permitir que um subconjunto representativo e menor que o original seja selecionado. Como resultado da realização de SA, é possível melhorar a qualidade dos dados e os modelos construídos durante o processo de mineração de dados podem ser mais compreensíveis. Além disso, como consequência da melhoria da qualidade dos dados, é possível que a performance de algoritmos de aprendizado seja aperfeiçoada, ou não deteriorada de modo significativo, e o processo de mineração de dados seja mais rápido.

Os diversos modelos de SA propostos na literatura podem ser categorizados nos modelos *wrapper* e filtro, dependendo de como o algoritmo de SA se relaciona com os algoritmos de aprendizado supervisionado, os quais utilizarão, posteriormente, os atributos selecionados na construção do modelo (Liu and Motoda, 1998). Além dos atributos irrelevantes, tem sido observado que atributos *redundantes* também afetam a precisão dos classificadores induzidos e, portanto, deveriam ser eliminados (Koller and Sahami, 1996; Hall, 2000). Considera-se que dois atributos são redundantes entre si, parcial ou completamente, quando seus valores estão correlacionados.

De modo geral, os métodos de seleção de atributos escolhem os atributos pela *avaliação individual* ou pela *avaliação de subconjuntos* de atributos. No caso de avaliação individual, freqüentemente, os atributos são ordenados considerando a sua importância na discriminação das classes, *i.e.*, tratam a relevância dos atributos. Esses métodos somente removem atributos irrelevantes pois espera-se que atributos redundantes tenham a

mesma importância na discriminação das classes. Contudo, métodos que avaliam subconjuntos de atributos buscando por subconjuntos mínimos podem remover tanto atributos irrelevantes quanto redundantes. Assim, a maioria dos métodos existentes para a SA que tratam tanto relevância quanto redundância de atributos, o fazem de maneira implícita por meio da avaliação de subconjuntos de atributos.

Desse modo, independentemente da abordagem escolhida para a realização de seleção de atributos, quer por avaliação individual ou por avaliação de subconjuntos, a estimativa de importância de atributos é comum à ambas. Várias medidas foram propostas na literatura para definir a importância de atributos. Alguns métodos da abordagem filtro consideram medidas de consistência para determinar que atributos são importantes, *i.e.*, procuram selecionar atributos que mantenham a consistência das classes — “todas” as combinações de valores de um subconjunto de atributos são associadas com uma única classe (Dash and Liu, 2003; Liu and Setiono, 1996; Almuallim and Diettrich, 1991). Outros métodos procuram eliminar atributos cuja informação é incorporada por outros atributos (Hall, 2000; Koller and Sahami, 1996). Há métodos, ainda, cujo objetivo é ordenar os atributos considerando medidas de distância (Robnik-Sikonja and Kononenko, 2003; Kira and Rendell, 1992).

Entre as diversas abordagens existentes com a finalidade de determinar a importância de atributos, foi proposto, recentemente, um algoritmo de seleção de atributos não redundantes baseado no cálculo da dimensão fractal para problemas não supervisionados (Traina et al., 2000). A teoria dos fractais tem sido aplicada a uma série de tarefas, algumas das quais associadas ao processo de descoberta de conhecimento a partir de bases de dados, como a detecção de *clusters* e a redução de dimensionalidade de dados não supervisionados. Outras áreas nas quais a teoria dos fractais tem sido utilizada incluem a procura por correlação nos dados para determinar a relação entre duas ou mais bases de dados multi-relacionais e o desenvolvimento de estruturas de indexação para dados complexos de alta dimensionalidade (Traina et al., 2005). Neste trabalho, investigamos o uso da dimensão fractal para encontrar atributos redundantes para problemas supervisionados. Destacamos que não é de nosso conhecimento que a teoria dos fractais tenha sido utilizada para seleção de atributos dentro do contexto de aprendizado de máquina supervisionado, como proposto neste trabalho.

De um modo geral, pode-se considerar a seleção de atributos como uma tarefa que pode auxiliar a simplificar a linguagem de descrição de exemplos quando ela possui mais atributos que os necessários. Com isso, surge naturalmente a seguinte questão:

Quantos e quais atributos são suficientes/necessários para descrever o problema do domínio?

Não há uma resposta única para essa questão. A escolha de um conjunto de atributos suficientes/necessários para descrever o problema está diretamente ligada à importância

desses atributos para o domínio e dos objetivos pretendidos com o processo de extração de conhecimento de bases de dados, bem como ao critério utilizado para avaliar essa importância.

Como mencionado, a maioria dos métodos existentes para a seleção de atributos que tratam tanto relevância quanto redundância de atributos, realizam avaliação de subconjuntos de atributos. Ainda que esses métodos geralmente apresentem melhores resultados que os métodos que não lidam com a redundância de atributos, o seu elevado custo computacional pode torná-los ineficientes para conjuntos de dados com alta dimensionalidade. Recentemente foi proposto o uso da abordagem filtro considerando o modelo de tratamento da relevância e da redundância de atributos como dois procedimentos separados (Yu and Liu, 2004). A vantagem desse modelo sobre o modelo anterior é que, por meio da separação das análises de relevância e de redundância, existe a possibilidade de diminuir o custo computacional na busca por um subconjunto que aproxima o subconjunto ótimo. Esse modelo é objeto de estudo deste trabalho.

1.1 Objetivos

Tendo em vista os problemas anteriormente descritos, os objetivos deste trabalho são:

1. Pesquisar e comparar métodos da abordagem filtro para a seleção de atributos para aprendizado de máquina supervisionado, considerando a aplicação desses algoritmos de aprendizado no processo de extração de conhecimento de bases de dados e
2. Propor um método para seleção de atributos, utilizando medidas de distância e informação para análise de relevância e medida de dependência, mais especificamente a dimensão fractal, para a análise de redundância

sob a seguinte hipótese:

1. A realização das análises de relevância e redundância, separadamente, auxilia a seleção de atributos relevantes e não redundantes e
2. A dimensão fractal constitui uma medida apropriada para remoção de redundância em conjuntos de dados para algoritmos de aprendizado de máquina supervisionado

baseada nas seguintes assertivas:

1. A seleção de atributos importantes auxilia na simplificação da linguagem de descrição de exemplos permitindo, entre outros, a redução de dimensionalidade dos dados sem ônus significativo de performance¹;

¹O ônus significativo ou não de performance, que pode não ser apenas a precisão do modelo construído após a SA, pode ser medido de diversos modos e depende dos objetivos pretendidos com o processo de extração de conhecimento.

2. O conjunto de atributos importantes deve conter não apenas atributos relevantes, mas também atributos não redundantes e
3. A dimensão fractal provê a quantidade de atributos não redundantes de um conjunto de dados que apresente característica de um fractal estatisticamente auto-similar.

1.2 Principais Contribuições

As principais contribuições desta tese podem ser organizadas do seguinte modo:

- Proposta de seleção de atributos para aprendizado supervisionado por meio da realização das análises de relevância e redundância, separadamente, utilizando o algoritmo FDimBF, desenvolvido neste trabalho, o qual considera a análise de redundância usando a dimensão fractal;
- Proposta de um modelo de avaliação de performance de algoritmos de seleção de atributos, o qual considera ambos o erro do modelo construído com os atributos selecionados e a proporção da redução no número de atributos selecionados pelos algoritmos de seleção de atributos em relação ao conjunto original de atributos do conjunto de dados em questão e
- Desenvolvimento de uma metodologia para auxiliar no mapeamento de laudos médicos que consistem de dados semi-estruturados para bases de dados estruturadas, as quais permitem gerar formatos apropriados para a aplicação de algoritmos de aprendizado de máquina.

É importante ressaltar que, em aprendizado supervisionado, muitos dos algoritmos propostos para seleção de atributos escolhem apenas atributos relevantes, ou seja, importantes em relação à classe, não tratando o problema de atributos redundantes. Ainda, as diversas medidas que permitem detectar redundância nos dados podem encontrar relações de redundância de diferentes tipos, desde correlações lineares e, menos freqüentemente, correlações não-lineares. O uso da dimensão fractal para a análise de redundância em problemas de aprendizado supervisionado, proposto neste trabalho, permite que relações de redundância não-lineares, e até mesmo fracionárias, entre atributos de um conjunto de dados, possam ser encontradas. Isso torna-se particularmente importante pois a maior parte dos conjuntos de dados reais não cumprem a suposição de uniformidade nos dados e de independência entre os atributos, pois esses dados não são bem comportados e exibem uma dimensão intrínseca (fractal), usualmente mais baixa que a dimensão na qual o conjunto de dados está imerso. Essa diferença ocorre devido ao fato da dimensão fractal considerar a existência de correlação entre atributos, enquanto a dimensão imersa pode não considerar. Como mencionado, o conceito de dimensão fractal tem sido utilizado, entre outras aplicações, para selecionar atributos em domínios não supervisionados, na área

de reconhecimento de padrões, entretanto, não é de nosso conhecimento sua utilização na área de mineração de dados quando algoritmos de aprendizado de máquina supervisionado são usados para a indução de modelos simbólicos utilizando simultaneamente atributos contínuos e discretos. Resultados experimentais utilizando vários algoritmos consolidados de SA e o algoritmo aqui proposto, mostram que esse algoritmo é apropriado para ser utilizado em conjunto com algoritmos de aprendizado supervisionado.

Com relação à segunda contribuição, consideramos que a avaliação do resultado obtido por algoritmos de seleção de atributos não pode ser limitada à medição da performance em termos do erro cometido pelo modelo construído a partir do subconjunto de atributos selecionados mas é também importante considerar o nível de redução dos atributos utilizados para construir esse modelo. Neste trabalho é proposto um modelo de avaliação de performance de algoritmos de SA, no qual a performance desses algoritmos, traduzida pelos modelos construídos a partir dos subconjuntos de atributos por eles selecionados, é avaliada ponderando-se o erro do modelo construído a partir desses atributos selecionados e a redução na quantidade de atributos selecionados em relação ao conjunto original de atributos. O modelo proposto mostrou ser uma alternativa para a avaliação da performance de algoritmos de SA, que considera tanto o erro dos modelos construídos com um subconjunto dos atributos originais quanto a redução da quantidade de atributos promovida pelos algoritmos de SA considerados.

Uma outra questão bastante importante e limitante da análise de dados contidos em documentos semi-estruturados, em parte, descritos em linguagem natural, como os laudos médicos, é a disponibilidade desses dados em formatos adequados para a aplicação de métodos associados ao processo de extração de conhecimento de bases de dados. Usualmente, algoritmos de aprendizado necessitam que os conjuntos de dados sejam representados no formato atributo-valor. Quando os dados não estão armazenados em bases de dados estruturadas, como é o caso dos laudos médicos, a abordagem habitual é o mapeamento manual das informações contidas nesses laudos para conjunto de dados estruturados durante a fase de pré-processamento de dados. Essa tarefa, além de ser bastante custosa, pode sofrer influência de algum grau de subjetividade na transcrição dos dados contidos nos laudos para o formato atributo-valor. A terceira principal contribuição deste trabalho é a proposta de uma metodologia que permite auxiliar o mapeamento desse tipo de documentos, com a participação de especialistas do domínio, em conjuntos de dados estruturados apropriados para a extração de conhecimento. A metodologia proposta neste trabalho, a qual foi implementada e utilizada em um caso real, mostrou ser uma ferramenta muito útil para a semi-automatização do mapeamento de laudos médicos para conjuntos de dados estruturados.

1.3 Organização do Trabalho

Este trabalho está organizado do seguinte modo:

Capítulo 2: Terminologia

Neste capítulo são apresentadas considerações sobre a notação empregada neste trabalho, bem como algumas observações sobre a terminologia encontrada na literatura tanto em relação ao termo atributo e seus sinônimos quanto em relação às operações que envolvem os atributos.

Capítulo 3: Seleção de Atributos

Neste capítulo são discutidos diversos aspectos relacionados à seleção de atributos e as diferentes perspectivas sob as quais a seleção de atributos pode ser considerada. O problema da seleção de um subconjunto de atributos é focado como um problema de busca e as diversas abordagens para a seleção de atributos são apresentadas.

Capítulo 4: Importância de Atributos

Neste capítulo são apresentadas algumas das medidas utilizadas para avaliar a importância de atributos, bem como diversas definições que usam essas medidas para definir em relação a que um atributo é considerado importante.

Capítulo 5: Algoritmo Proposto

Neste capítulo é apresentado o algoritmo FDimBF proposto neste trabalho para a seleção de atributos, o qual considera as análises de relevância e redundância em duas etapas separadas. São apresentados conceitos sobre fractais e dimensão fractal, utilizados para realizar a remoção de atributos redundantes de conjuntos de dados durante a seleção de atributos.

Capítulo 6: Avaliação Experimental

Neste capítulo é apresentada uma série de experimentos, os quais são realizados utilizando algoritmos e conjuntos de dados amplamente considerados na literatura da área de seleção de atributos. São apresentados resultados que avaliam diversos aspectos relacionados à dimensão fractal dos conjuntos de dados, características dos subconjuntos de atributos selecionados e performance dos modelos construídos a partir desses subconjuntos. Também é apresentada a análise de uma meta-base obtida nesses experimentos, contendo informações sobre as relações entre as características dos conjuntos de dados e a aplicação do algoritmo proposto.

Capítulo 7: Estudo de Caso

Neste capítulo é realizado um estudo de caso utilizando um conjunto de dados reais da área médica, os quais encontram-se representados por meio de laudos médicos em um formato semi-estruturado. Ainda neste capítulo é descrita a metodologia proposta, e a implementação dessa metodologia, com a finalidade de mapear laudos médicos para conjuntos de dados no formato adequado para a aplicação de algoritmos de aprendizado. Além da avaliação experimental semelhante à realizada no Capítulo 6, é também realizada uma avaliação dos modelos construídos com o auxílio

de especialistas do domínio médico, com o objetivo de verificar se o conhecimento gerado apresenta coerência com o conhecimento prévio e se pode ser considerado interessante e/ou novo.

Capítulo 8: Conclusão

Neste capítulo são apresentadas as conclusões deste trabalho, as principais contribuições, as limitações e os trabalhos futuros.

Apêndice A: Distribuição dos Valores dos Atributos

Neste Apêndice são mostradas as distribuições aproximadas dos valores dos atributos do conjunto de dados reais Processamento de Sêmen, utilizado no estudo de caso.

Apêndice B: Avaliação das Regras usando Medidas Objetivas e Subjetiva

Neste Apêndice são mostrados os resultados das medidas de avaliação objetiva e subjetiva de regras de decisão, utilizadas neste trabalho, para cada uma das regras que compõem os modelos construídos considerando os subconjuntos derivados a partir do conjunto de dados Processamento de Sêmen, utilizado no estudo de caso.

Capítulo 2

Terminologia

2.1 Considerações Iniciais

Neste capítulo são feitas algumas considerações sobre a notação empregada neste trabalho, bem como algumas observações sobre a terminologia encontrada na literatura tanto em relação ao termo atributo e seus sinônimos quanto em relação às operações que envolvem os atributos.

2.2 Definição da Notação Utilizada neste Trabalho

Diversas linguagens podem ser utilizadas para descrever conjuntos de dados, entre as quais linguagens baseadas em atributos e valores desses atributos, comumente denominadas de atributo-valor.

Esse tipo de linguagem é utilizada em diversas áreas, tais como aprendizado de máquina, estatística e base de dados. Além da sua simplicidade, a linguagem atributo-valor tem-se mostrado suficientemente poderosa para variadas aplicações em descoberta de conhecimento e mineração de dados.

Um sistema de aprendizado de máquina recebe, usualmente, como entrada um conjunto de dados, denominado de conjunto de exemplos ou casos, representados por um conjunto de atributos, também designados de características, propriedades e medições. Alguns autores fazem distinção entre os termos atributo e variável, definindo o último como sendo o dado bruto de entrada enquanto o termo atributo é utilizado para “variáveis” construídas a partir das variáveis originais de entrada (Guyon and Elisseeff, 2003). Neste trabalho os termos atributo, variável e características serão usados indistintamente.

Os atributos podem ter valores discretos ou numéricos ou serem do tipo complexo ou composto. Atributos discretos são variáveis que podem assumir apenas um número limitado de valores, podendo ser divididos em ordenados (os valores possuem ordem) e

nominais (os valores não possuem ordem). Ao contrário dos atributos discretos, os atributos numéricos possuem valores no domínio dos números reais, podendo assim assumir um número infinito de valores. Os atributos do tipo complexo podem representar tipos não usuais de dados como imagens, audio e DNA. Já os atributos do tipo composto são representados por atributos formados por uma combinação de atributos dos tipos anteriormente citados. Na Figura 2.1 é mostrada a hierarquia dos tipos de atributos. Neste trabalho serão tratados atributos discretos e numéricos.

Figura 2.1: Hierarquia de tipos de atributos

O formato atributo-valor, comumente usado para representar exemplos, é apresentado na Tabela 2.1. Em aprendizado supervisionado, cada exemplo é associado a uma classe (rótulo), que pode ser discreta, sendo nesse caso denominada de classificação, tratada neste trabalho, ou numérica, denominada de regressão. No caso de aprendizado não supervisionado, não há a informação sobre a classe associada a cada exemplo.

	Atributos				
Exemplos	X_1	X_2	$\dots$	X_M	Classe (Y)
E_1	x_{11}	x_{12}	$\dots$	x_{1M}	y_1
E_2	x_{21}	x_{22}	$\dots$	x_{2M}	y_2
E_3	x_{31}	x_{32}	$\dots$	x_{3M}	y_3
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
E_N	x_{N1}	x_{N2}	$\dots$	x_{NM}	y_N

Tabela 2.1: Formato padrão do conjunto de exemplos

Em um problema típico de aprendizado supervisionado, o algoritmo de aprendizado recebe como entrada um conjunto de exemplos S de treinamento utilizado para a construção do modelo e um conjunto de exemplos de teste T utilizado para avaliar o modelo construído. Esses conjuntos de dados possuem as seguintes características:

1. cada exemplo dos conjuntos de treinamento e teste é descrito por M atributos;
2. o conjunto de treinamento possui N exemplos E_i , $i = 1, \dots, N$, no qual $E_i = \{(\vec{x}_1, y_1), \dots, (\vec{x}_N, y_N)\}$ e $\vec{x}_i = \langle x_{i1}, x_{i2}, \dots, x_{iM} \rangle$;

3. um valor x_{ij} do atributo X_j , pertencente ao exemplo E_i , pode ser discreto ou numérico;
4. o domínio de cada atributo é denotado por D_i ;
5. um exemplo (caso ou registro) é um ponto no espaço de exemplos $D_1 \times D_2 \times \dots \times D_M$;
6. a classe y é tal que $y = f(\vec{x})$, sendo f a verdadeira função (desconhecida) e
7. os valores y_i referem-se ao valor do atributo classe Y , tipicamente pertencentes a um conjunto discreto de classes C_v , $v = 1, \dots, N_{Cl}$ da forma $\{C_1, \dots, C_{N_{Cl}}\}$ quando se trata de classificação e ao conjunto dos números reais no caso da regressão.

Dado o conjunto de treinamento S ao algoritmo de aprendizado supervisionado, um modelo $\mathbf{h}$ consiste da *hipótese* construída sobre a função f . Dados novos exemplos $\vec{x}$, o modelo $\mathbf{h}$ prediz o valor y correspondente.

Embora o algoritmo de aprendizado visibilize apenas a amostra fixa S , geralmente é útil considerar duas informações adicionais que auxiliam na definição da importância de atributos:

1. a distribuição de probabilidade $\mathcal{D}$ sobre o espaço de exemplos e
2. a função de predição f que mapeia os exemplos para as classes.

Sendo assim, é possível modelar a amostra S como sendo gerada pela seleção repetida de exemplos a partir de $\mathcal{D}$, rotulando esses exemplos de acordo com a função de predição f . Pode-se notar que a função de predição f pode tanto ser determinística quanto probabilística, nesse último caso, para um determinado exemplo $\vec{x}$, $f(\vec{x})$ seria a distribuição de probabilidade sobre as classes ao invés de uma única classe.

2.3 Definição das Operações Realizadas sobre os Atributos

O termo seleção de atributos já foi utilizado para denominar, de um modo mais geral, as operações tanto de Seleção de um Subconjunto de Atributos — SSA — quanto de construção de atributos (Guyon and Elisseeff, 2003; Lee, 2000). Porém, muitos autores têm considerado a SA e a SSA como sendo a mesma tarefa (Liu and Yu, 2002; Molina et al., 2002). Em outras palavras, é um processo que escolhe um subconjunto de P atributos, ótimo de acordo com um determinado critério, a partir do conjunto original com M atributos de maneira que $P \leq M$ (Blum and Langley, 1997; Dash and Liu, 1997). Porém, a SA pode ser considerada uma tarefa que engloba duas outras sub-tarefas, a SSA e o *Ranking* de atributos (Liu and Motoda, 1998).

Quanto ao termo transformação, pode-se considerar essa tarefa sob dois aspectos, a transformação de dados e a transformação de atributos. No primeiro caso, as transformações se referem tanto a dimensão dos exemplos quanto dos atributos, pois operações tais como resumo dos dados, transformação de tipos dos dados e normalização dos dados são realizadas (Baranauskas, 2001; Fayyad et al., 1996b). No segundo caso, a transformação se refere a operações ligadas aos atributos. Alguns autores classificam como tarefas de transformação de atributos, os processos de extração de atributos e construção de atributos (Motoda and Liu, 2002; Liu and Yu, 2002). Há ainda os autores que consideram as tarefas de construção de atributos e extração de atributos as mesmas (Guyon and Elisseeff, 2003). Existem alguns autores que também incluem nessa categoria a tarefa de discretização de atributos (Liu and Motoda, 1998).

Uma hierarquia de algumas operações possíveis sobre atributos é mostrada na Figura 2.2, baseada em Liu and Motoda (1998). Neste trabalho será considerada essa hierarquia e, como mencionado, será abordado o tema de seleção de atributos. Observando

Figura 2.2: Hierarquia de operações envolvendo atributos

a Figura 2.2 é possível notar que a SA, considerada do ponto de vista da saída resultante, pode fornecer o conjunto de atributos ordenados (*ranking*) ou um subconjunto ótimo de atributos segundo uma determinada medida de importância. Ordenar os atributos de acordo com uma medida de importância, pode fazer com que dois atributos, por exemplo altamente correlacionados, sejam considerados muito importantes. Se em seguida os primeiros Q atributos dessa lista são escolhidos como sendo os mais relevantes, esse subconjunto pode conter atributos redundantes. Entretanto, isso pode não acontecer se a abordagem de seleção de um subconjunto de atributos for escolhida pois, dependendo da medida de importância considerada, o subconjunto ótimo de atributos pode não conter atributos redundantes.

Quanto à transformação de atributos, segundo essa hierarquia, três objetivos podem ser alcançados:

1. a construção de novos atributos, a qual permite que atributos que, possivelmente, podem representar melhor o conceito embutido nos dados sejam compostos. Com isso, o espaço de atributos é aumentado;
2. a extração de atributos, a qual permite que o conjunto original de atributos possa ser mapeado em um conjunto menor de atributos. Desse modo, técnicas de visualização podem ser aplicadas e uma análise preliminar dos dados realizada e
3. a discretização dos atributos, a qual por meio do mapeamento dos valores numéricos em intervalos de valores, permite nomear cada intervalo com um valor discreto.

É interessante notar que as operações de transformação e seleção sobre os atributos não são exclusivas. Ou seja, é possível, por exemplo, após a construção de novos atributos — Figura 2.2 na página oposta, realizar uma SA para selecionar um subconjunto ótimo a partir desse novo conjunto de atributos segundo alguma medida de importância.

Neste trabalho consideramos a tarefa de seleção de atributos como aquela que engloba tanto a seleção de um subconjunto de atributos quanto a que considera um *ranking* dos atributos. Neste último caso, pode ocorrer também uma pseudo SSA, já que é possível considerar um limiar mínimo para a medida de importância utilizada para ordenar os atributos.

2.4 Considerações Finais

Neste capítulo foram definidas a notação, bem como a linguagem atributo-valor de descrição de exemplos utilizada neste trabalho. Ainda, como existem algumas variações quanto a definição de termos, tais como a seleção e a transformação de atributos, foi apresentada neste capítulo uma hierarquia das operações realizadas sobre os atributos, a qual será considerada neste trabalho.

Capítulo 3

Seleção de Atributos

3.1 Considerações Iniciais

A seleção de atributos é um tema de pesquisa e desenvolvimento constante desde os anos 70 nas áreas de reconhecimento de padrões, aprendizado de máquina e mineração de dados (Liu and Yu, 2002), representando um papel importante na fase de pré-processamento no processo de KDD (Kohavi and John, 1997; Fayyad et al., 1996a). A seleção de atributos permite, por exemplo, a ordenação de atributos segundo algum critério de importância, a redução da dimensionalidade do espaço de busca de atributos e a remoção de dados contendo ruídos entre outros. A SA pode ser particularmente importante, por exemplo, em casos nos quais a medição de certos atributos é custosa, pois pode permitir que um subconjunto, representativo e menor que o original, seja selecionado. Como resultado da realização de SA, a qualidade dos dados pode ser melhorada e os modelos construídos durante o processo de mineração de dados podem ser mais compreensíveis. Além disso, como consequência da melhoria da qualidade dos dados, é possível que a performance de algoritmos de aprendizado seja aperfeiçoada ou não deteriorada de forma significativa e o processo de mineração de dados seja mais rápido. Neste capítulo são discutidas a seleção de atributos como um problema de busca e as diversas abordagens para a SA.

3.2 O Problema da Seleção de Atributos

Do ponto de vista conceitual, a tarefa de aprendizado de conceitos pode ser dividida em duas sub-tarefas:

1. decidir que atributos utilizar na descrição do conceito e
2. decidir como combinar esses atributos.

Sob essa visão, a seleção de atributos importantes e a eliminação dos não importantes constitui um dos principais problemas a serem tratados em AM.

Do ponto de vista prático, é desejável que algoritmos de aprendizado trabalhem bem em domínios que contenham muitos atributos não importantes. Em outras palavras, um dos objetivos é que o número de exemplos de treinamento necessários para se atingir uma dada precisão do modelo construído por um algoritmo de aprendizado (complexidade da amostra) cresça lentamente em relação ao número de atributos não importantes. Por exemplo, na tarefa de classificação de texto, não é incomum a descrição de exemplos utilizando mais de 10^4 atributos para representar um texto, sabendo-se que apenas uma pequena fração desses atributos é crucial para o aprendizado do conceito (Matsubara and Monard, 2005; Martins, 2003; Sebastiani, 2002). Nos últimos anos, uma quantidade crescente de pesquisas em AM, tanto teóricas quanto práticas, tem-se voltado para o desenvolvimento de algoritmos que trabalhem bem na presença de muitos atributos, em geral, irrelevantes¹. Entretanto, tem sido observado que somente o critério de relevância não é suficiente para a seleção de atributos importantes. Trabalhos recentes têm mostrado que também deve-se levar em conta o critério de redundância para selecionar os atributos importantes, pois atributos redundantes afetam a qualidade das hipóteses induzidas. (Hall, 2000; Koller and Sahami, 1996). Assim, intuitivamente, seria desejável que um algoritmo de aprendizado utilizasse apenas os “atributos importantes” para o aprendizado do conceito (Lee, 2000; Blum and Langley, 1997; Kohavi and John, 1997).

A meta da SA pode ser formalizada do seguinte modo (Yu and Liu, 2004): seja $X' \subset X$ um subconjunto de atributos de X , e $f'(\mathbf{x}')$ os valores associados aos vetores correspondentes a X' . O objetivo da SA consiste em selecionar o subconjunto mínimo de atributos X' tal que $\mathbf{P}(C|y = f'(\mathbf{x}')) \approx \mathbf{P}(C|y = f(\mathbf{x}))$, onde $\mathbf{P}(C|y = f'(\mathbf{x}'))$ e $\mathbf{P}(C|y = f(\mathbf{x}))$ são as distribuições de probabilidades das N_{C_i} possíveis classes dados os valores dos atributos de X' e X respectivamente. Esse subconjunto mínimo X' é denominado subconjunto *ótimo* de atributos.

O problema de seleção de atributos pode ser examinado a partir de diferentes perspectivas. As quatro principais são:

1. Como procurar pelos “melhores” atributos?
2. O que deve ser considerado para determinar os melhores atributos e qual o critério para avaliação desses atributos?
3. Como os subconjuntos de atributos devem ser gerados para a seleção, adicionando ou excluindo um atributo do subconjunto existente ou mudando o subconjunto (geração sequencial ou paralela)?
4. Como os requisitos das aplicações consideradas influenciam a SA?

¹Neste trabalho, o termo *relevância* será considerado relacionado à importância de um atributo em relação à classe. O termo *importância* será utilizado para denominar a importância de um atributo em relação a alguma medida, não necessariamente relacionada à classe.

Em relação à primeira questão, o problema da SA pode ser tratado, como mencionado anteriormente, de duas maneiras: como uma ordenação de atributos segundo alguma medida de importância (*ranking*) ou como a busca por um subconjunto ótimo de atributos. Em geral, quando a primeira abordagem é escolhida, após a ordenação dos atributos, os Q primeiros são escolhidos para formar o subconjunto de atributos importantes selecionado. Quando a segunda abordagem é a escolhida, usualmente, os subconjuntos de atributos são avaliados, de acordo com uma determinada medida de importância. Esse assunto será tratado com mais detalhes na próxima seção, na qual também é tratada a terceira questão: como os subconjuntos para seleção devem ser gerados? A questão da determinação de quais são os melhores atributos ou que critério utilizar para a avaliação deles está relacionada à determinação da importância dos atributos tratada no Capítulo 4. Finalmente, a última questão, a influência da aplicação sobre a SA, depende dos requisitos impostos por essa aplicação, por exemplo, em termos de tempo e resultados desejados, entre outros.

3.3 Seleção de um Subconjunto de Atributos como um Problema de Busca

A busca é um tópico chave no estudo da seleção de subconjuntos de atributos (Langley, 1994). Desse modo, ela constitui um paradigma conveniente para a representação das diversas abordagens para a tarefa de SSA (principalmente para aquelas que realizam seleção explícita), na qual cada estado no espaço de busca é composto por um subconjunto de possíveis atributos. Qualquer método de seleção de subconjuntos atributos pode ser caracterizado por sua instanciação em relação a, basicamente, quatro questões, as quais determinam a natureza do processo de busca. São elas:

1. o(s) ponto(s) de partida ou a direção em que a busca será realizada;
2. a estratégia da busca;
3. o critério para avaliação dos subconjuntos gerados e
4. o critério de parada.

A primeira questão que deve ser tratada é a determinação do ponto de partida (ou pontos de partida) no espaço de busca. Essa determinação por sua vez influencia a direção em que a busca será realizada e os operadores que serão utilizados para a geração dos estados sucessores. Na Figura 3.1 é ilustrado o espaço de busca para quatro atributos.

Pode-se observar que existe uma ordenação parcial dos estados, pois cada um deles possui um atributo a mais que o estado anterior, sendo o estado inicial (mais a esquerda) estabelecido como vazio (subconjunto vazio de atributos). Essa abordagem é geralmente

Figura 3.1: Exemplo de espaço de estados de atributos (Langley, 1996)

conhecida como seleção *forward*. Já a abordagem que inicia o ponto de partida com o conjunto de todos os atributos e sucessivamente removendo-os, é denominado de eliminação *backward*. Podem também ser empregadas variações das abordagens mencionadas anteriormente, tais como a geração ou seleção bidirecional e a seleção randômica (Liu and Motoda, 1998).

Se não se tem nenhuma informação sobre P , *i.e.*, o número de atributos do subconjunto ótimo de atributos, há uma grande chance de, na maioria das vezes, ele estar localizado na região central do espaço de busca. Desse modo, uma seleção bidirecional pode ser vantajosa. A seleção bidirecional inicia-se em ambas direções, em outras palavras, duas buscas são processadas simultaneamente, as quais terminam em dois casos:

- quando uma das buscas encontra os melhores P atributos antes de alcançar o centro do espaço de busca ou
- quando ambas atingem o centro do espaço de busca.

Essa maneira de geração de subconjuntos de atributos possui vantagens de ambas as seleções *forward* e *backward*. É interessante notar que os subconjuntos de atributos encontrados pelas duas seleções podem não ser os mesmos devido a seqüência de adição/exclusão de atributos.

Outro tipo de geração de subconjuntos de atributos, o qual complementa as outras abordagens, é a seleção ou geração randômica. Nessa abordagem não há uma direção específica na qual a busca será realizada. O objetivo dessa abordagem é evitar que a busca fique presa em um mínimo local por meio da não fixação de como os subconjuntos são gerados.

A segunda questão a ser tratada no processo de busca está relacionada à estratégia da busca. A cada ponto nesse processo de busca, modificações locais ao conjunto de atributos são consideradas, uma dessas é selecionada e uma nova iteração é realizada. Claramente, uma busca exaustiva em todo o espaço de estados é impraticável, já que para um número M de atributos há 2^M possíveis estados. Existem três diferentes estratégias para lidar com esse problema (Liu and Yu, 2002):

Busca completa: surge da resposta para a questão “É necessário recorrer à busca exaustiva para que um subconjunto ótimo seja gerado?”. Em alguns casos, a busca ser completa, ou seja, nenhum subconjunto ótimo de atributos é perdido, não significa que a busca tenha que ser exaustiva (todo subconjunto de atributos tenha que ser avaliado). Embora não suficiente, pois a complexidade do espaço de busca é de $O(2^M)$, se o critério de avaliação possui determinadas propriedades, tais como a propriedade de ser monotônico, um subconjunto ótimo de atributos pode ser encontrado sem que os 2^M subconjuntos tenham que ser avaliados. Entretanto, se a propriedade da monotonicidade não for satisfeita, somente a busca exaustiva garante que uma solução ótima possa ser encontrada (Schlimmer, 1993);

Busca heurística: emprega algum tipo de heurística para conduzir a busca. Evita que a busca seja exaustiva, porém, ao mesmo tempo, corre-se o risco de não encontrar os subconjuntos ótimos. Essa estratégia é muito mais rápida e a complexidade do espaço de busca pode ser reduzida à $O(M^2)$ ou menor, utilizando, por exemplo, o algoritmo guloso *Best-First*. Esse algoritmo expande, em cada nível o nó corrente, avalia os filhos desse nó e repete o processo a partir do melhor desses filhos até que não seja possível expandir o nó corrente. A complexidade da busca de *Best-First* é de $O(b \cdot M)$ onde b é o máximo número de filhos que um nó pode ter, ou seja $O(M^2)$ (Liu and Yu, 2002) e

Busca não-determinística: diferentemente das duas estratégias anteriores, procura pelos subconjuntos de atributos aleatoriamente, ou seja, o subconjunto corrente não aumenta ou diminui, diretamente, a partir de algum subconjunto anterior segundo uma regra determinística. Embora a complexidade do espaço de busca seja ainda de $O(2^M)$, essa estratégia busca, usualmente, por um número menor que 2^M subconjuntos por meio da fixação de um número máximo possível de iterações.

As possíveis combinações entre as estratégias e direções de busca são resumidas na Tabela 3.1. É interessante notar que para a busca não-determinística, apenas a seleção randômica é considerada possível. Embora seja ainda possível controlar o crescimento ou diminuição do subconjunto de atributos nesse modo de seleção, nenhuma relação sequencial pode ser determinada entre dois subconjuntos gerados consecutivamente.

A terceira questão a ser tratada no processo de busca considera o critério utilizado na avaliação dos subconjuntos alternativos de atributos. Uma métrica usualmente empregada

	Estratégias de Busca		
Direções de Busca	Completa	Heurística	Não-determinística
Seleção <i>Forward</i>	sim	sim	não
Eliminação <i>Backward</i>	sim	sim	não
Seleção Bidirecional	sim	sim	não
Seleção Randômica	não	sim	sim

Tabela 3.1: Combinações de estratégias e direções de busca (Liu and Motoda, 1998).

envolve a habilidade de um atributo discriminar entre as classes que ocorrem num conjunto de dados de treinamento. Diversos algoritmos de aprendizado incorporam um critério baseado na Teoria da Informação, enquanto outros medem diretamente a precisão do conjunto de treinamento ou de um conjunto separado de avaliação. É importante então determinar “o que é um bom atributo?”. Como mencionado, esse assunto será tratado com mais detalhes no Capítulo 4.

Outro aspecto importante é como medir quão bom um determinado atributo é, segundo um critério de avaliação. Em outras palavras, como ele interage com o algoritmo de aprendizado. Essa interação pode ser subdividida em, basicamente, três abordagens para SA (Kohavi and John, 1997):

1. *embedded*, a qual é embutida no próprio algoritmo de aprendizado;
2. filtro, a qual é utilizada para filtrar atributos durante um passo de pré-processamento sem considerar o algoritmo de aprendizado que utilizará esse subconjunto de atributos e
3. *wrapper*, a qual emprega o próprio algoritmo de aprendizado como uma “caixa preta”.

As abordagens filtro e *wrapper*, provêm diferentes ênfases à redução de dimensionalidade ou melhora da precisão. Recentemente, abordagens híbridas têm sido propostas para combinar as vantagens de ambos filtros e *wrappers*. Essas abordagens são discutidas brevemente na Seção 3.4 na página 24.

De acordo com a disponibilidade de informação sobre a classe associada a cada exemplo, pode-se ainda ter a seleção de atributos supervisionada (Liu and Motoda, 2002; Dash and Liu, 1997; Weiss and Kulikowski, 1991), ou a seleção não supervisionada (Dash et al., 2002; Dash and Liu, 2000; Dy and Brodley, 2000; Dash and Liu, 1999; Talavera, 1999), caso contrário.

Finalmente, a quarta decisão a ser tomada refere-se ao critério de parada da busca. Alguns possíveis critérios são:

- parar de remover ou adicionar atributos quando nenhuma das alternativas melhora a precisão da estimativa para a classificação;
- continuar revisando o subconjunto de atributos enquanto a precisão não se degrada;

- continuar gerando subconjuntos candidatos até que o outro extremo do espaço de busca seja alcançado e escolher o melhor desses subconjuntos;
- parar quando o subconjunto de atributos selecionado separar perfeitamente todas as classes (assumindo que não há ruído nos dados) e
- ordenar os atributos segundo alguma pontuação de importância e utilizar um parâmetro de sistema para determinar o ponto de parada, por exemplo, o número de atributos desejado para o subconjunto. Essa alternativa é mais robusta que a anterior.

Para exemplificar o paradigma da busca heurística utilizada para representar a seleção de um subconjunto de atributos, considere os conceitos que podem ser expressados como uma disjunção de atributos e uma estratégia gulosa (*greedy*):

- inicie com a disjunção de zero atributos;
- entre os atributos não presentes em nenhum exemplo negativo (atributo dito “seguro”) escolha aquele que incluído na hipótese corrente proporcione o maior acréscimo no número de exemplos positivos corretamente classificados e
- repita até que não existam mais atributos “seguros” que poderiam aumentar o número de exemplos positivos classificados e então pare.

Assim, para as quatro questões, em relação às quais os métodos de SA podem ser instanciados, tem-se para esse exemplo:

- ponto de partida: ponto mais a esquerda da Figura 3.1 na página 20 (seleção *forward*);
- estratégia da busca: move-se incrementalmente apenas a direita;
- critério de avaliação: avalia os subconjuntos baseados na performance do algoritmo de aprendizado sobre o conjunto de dados com uma penalidade muito alta para a classificação errada de exemplos negativos e
- critério de parada: quando não é mais possível melhorar a performance.

Para mostrar como os atributos são selecionados pelo algoritmo de SA heurística do exemplo, considere o conjunto de exemplos apresentado na Tabela 3.2.

A seleção é iniciada com uma disjunção de zero atributos. O primeiro atributo a ser selecionado é X_1 , pois observando apenas os exemplos positivos, ele é o atributo seguro que proporciona maior acréscimo no número de exemplos positivos corretamente classificados. Todos os outros atributos classificam apenas um exemplo positivo, enquanto

	Atributos					
Exemplos	X_1	X_2	X_3	X_4	X_5	Classe
E_1	1	0	0	0	0	+
E_2	1	1	0	0	0	+
E_3	0	0	1	1	0	+
E_4	0	0	0	0	1	+
E_5	0	0	0	0	0	-

Tabela 3.2: Amostra de exemplos para ilustrar o paradigma da busca heurística para a SA

X_1 classifica corretamente dois exemplos positivos. Depois, provavelmente o próximo atributo a ser selecionado será X_3 , o qual classifica corretamente mais um exemplo positivo, diferente daqueles já classificados com a escolha de X_1 . X_2 não seria selecionado pois o mesmo exemplo classificado por ele já foi classificado com a escolha de X_1 . O próximo atributo escolhido provavelmente é X_5 , pois X_4 , assim como X_2 , também classifica um exemplo positivo já classificado por outro atributo selecionado anteriormente. Finalmente, a seleção termina pois não há mais atributos seguros, além de todos os exemplos positivos já terem sido classificados.

3.4 Abordagens para a Seleção de Atributos

Como mencionado anteriormente, as principais abordagens para a seleção de subconjuntos de atributos podem ser agrupadas em: *embedded*, filtro e *wrapper* (Kohavi and John, 1997). Uma outra abordagem denominada híbrida constitui uma combinação, quer seja de diversas maneiras de interação entre o critério de avaliação e o algoritmo básico de indução, quer seja de diferentes critérios de avaliação (Bins and Draper, 2001; Das, 2001; Dash and Liu, 1998). Um desses algoritmos híbridos utiliza *boosting*² e incorpora algumas características da abordagem *wrapper* em um filtro (Das, 2001). Um outro exemplo de abordagem híbrida emprega uma combinação das buscas probabilística e completa (Dash and Liu, 1998). As três primeiras abordagens, *embedded*, filtro e *wrapper*, são descritas a seguir. Essas abordagens realizam a SA de forma explícita.

3.4.1 Abordagem *Embedded*

Nessa abordagem a tarefa da seleção dos atributos é realizada internamente pelo próprio algoritmo de aprendizado. Em outras palavras, a seleção de um subconjunto atributos está embutida no algoritmo de AM como ilustrada na Figura 3.2.

²*Boosting* tem como finalidade melhorar o desempenho de um sistema de aprendizado (Schapire, 1990). A principal idéia é construir diversos modelos ao invés de apenas um. A classe para um novo caso é predita por meio da votação dos diversos modelos construídos e a nova classe determinada pelo valor mais votado. A indução desses modelos é realizada de modo seqüencial, *i.e.*, um modelo é construído e os exemplos erroneamente classificados por ele são considerados com maior peso para a construção do próximo modelo. Esse processo continua, em geral, por um número pré-determinado de iterações.

Figura 3.2: Abordagem *Embedded*

A maioria dos algoritmos de aprendizado pertencentes à classe de métodos *eager* possuem uma abordagem *embedded* para a seleção de atributos. Esses métodos substituem, gulosamente, os exemplos de treinamento pelo conceito induzido, geralmente na forma de conjunto de regras, árvore de decisão ou rede neural (Monard and Baranauskas, 2003). Posteriormente, somente o conceito induzido é utilizado para classificar novos exemplos. São exemplos desses métodos os algoritmos de aprendizado simbólico CN2 (Clark and Niblett, 1989), C4.5 (Quinlan, 1993) e ID3 (Quinlan, 1983).

3.4.2 Abordagem Filtro

Essa abordagem de seleção de atributos, a qual é utilizada neste trabalho, introduz um processo separado, o qual ocorre antes da aplicação do algoritmo de aprendizado propriamente dito — Figura 3.3. A idéia é filtrar atributos irrelevantes, segundo algum critério, tais como os descritos no Capítulo 4, antes do aprendizado ocorrer (John et al., 1994). Esse passo de pré-processamento considera características gerais do conjunto de dados para selecionar alguns atributos e excluir outros. Sendo assim, métodos de filtros são independentes do algoritmo de aprendizado que, simplesmente, receberá como entrada o conjunto de exemplos descrito utilizando somente o subconjunto de atributos importantes fornecido pelo filtro.

Figura 3.3: Abordagem *Filtro* (Baranauskas, 2001)

Um dos esquemas mais simples de filtragem é a avaliação de cada atributo indivi-

dualmente, baseada na sua correlação com o conceito meta, escolhendo o subconjunto de P atributos que fornecem o melhor valor dessa correlação (Blum and Langley, 1997).

3.4.3 Abordagem *Wrapper*

Essa abordagem também ocorre externamente ao algoritmo básico de aprendizado, porém utilizando tal algoritmo como uma caixa preta para analisar, a cada iteração, o subconjunto de atributos em questão — Figura 3.4. Em outras palavras, métodos *wrapper* geram um subconjunto candidato de atributos, executam o algoritmo de aprendizado considerando apenas esse subconjunto de atributos selecionado do conjunto de treinamento, e utilizam a precisão resultante do classificador induzido para avaliar o subconjunto de atributos em questão. Esse processo é repetido para cada subconjunto de atributos até que o critério de parada determinado seja satisfeito.

Figura 3.4: Abordagem *Wrapper*

Um argumento utilizado com muita frequência para apoiar a utilização da abordagem *wrapper* é que o mesmo algoritmo de aprendizado que vai usar o subconjunto de atributos selecionado deve prover uma estimativa melhor de precisão que um outro algoritmo, o qual pode possuir um *bias*³ de aprendizado totalmente diferente (Kohavi and John, 1997). Porém, a maior desvantagem dos métodos *wrapper* é o custo computacional, o qual resulta da execução do algoritmo de aprendizado para avaliar cada subconjunto de atributos a ser considerado (Pila, 2001; Lee et al., 1999; Baranauskas et al., 1999; Baranauskas and Monard, 1998; Kohavi and John, 1997).

³O *bias* de aprendizado é definido como qualquer preferência de uma hipótese sobre outra, além da simples consistência com os exemplos.

3.5 Considerações Finais

Neste capítulo foi introduzido o problema da seleção de atributos. A seleção de subconjuntos de atributos foi descrita como um problema de busca e foram apresentadas as principais abordagens utilizadas na seleção de atributos, *i.e.*, *embedded*, filtro e *wrapper*. Diversas medidas têm sido propostas para a avaliação da importância de atributos, bem como o uso dessas medidas para determinar o que significa um atributo ser considerado importante. No próximo capítulo são apresentadas várias medidas para a avaliação de atributos, assim como definições de importância de atributos encontradas na literatura.

Capítulo 4

Importância de Atributos

4.1 Considerações Iniciais

Como mencionado anteriormente, o objetivo da seleção de atributos é, a partir do conjunto original de M atributos, selecionar um subconjunto ótimo de P atributos, tal que $P \leq M$. Para selecionar esse subconjunto ótimo de atributos é necessário definir o que significa um atributo ser bom, ou seja, responder à pergunta:

Importante em relação a que ?

Neste capítulo são apresentadas algumas das medidas utilizadas para avaliar a importância de atributos, bem como diversas definições que usam essas medidas para definir em relação a que um atributo é considerado importante.

4.2 Medidas de Avaliação de Atributos

A necessidade de estimativa da importância de atributos é comum, tanto à avaliação individual quanto à avaliação de subconjuntos de atributos, qualquer que seja a estratégia de busca. A questão da avaliação é complexa e multidimensional (Liu and Motoda, 1998). Por exemplo, a avaliação pode ser considerada em termos de:

1. se os atributos selecionados auxiliam a melhorar a precisão do classificador ou
2. se os atributos selecionados auxiliam a simplificar o modelo construído de modo que ele seja mais compreensível.

Assim, a importância de um atributo pode ser definida, de uma maneira geral, como:

Definição 4.2.1 *Importância de um Atributo* (Liu and Motoda, 1998): Um atributo é dito importante se quando removido a medida de importância considerada em relação aos atributos restantes é deteriorada.

Algumas das medidas de importância de atributos, as quais podem ser utilizadas para avaliar os atributos ou determinar em relação a que são importantes, são apresentadas a seguir.

Medidas de Informação - Determinam o ganho de informação a partir de um atributo. O ganho de informação de um atributo X_i é definido como sendo a diferença entre a incerteza *a priori* e a incerteza *a posteriori* considerando-se X_i . Assim, um atributo X_i é preferido ao atributo X_j se o ganho de informação a partir do atributo X_i é maior que a partir do atributo X_j . Um exemplo de medida de informação é a entropia;

Medidas de Distância - Também denominadas de medidas de separabilidade, divergência ou discriminação. Para um problema de duas classes, um atributo X_i é preferido ao atributo X_j se X_i provê uma diferença maior que X_j entre as probabilidades condicionais das duas classes. Um exemplo desse tipo de medida é a distância Euclidiana;

Medidas de Dependência - Também conhecidas como medidas de correlação ou associação. Qualificam a habilidade de prever o valor de uma variável (atributo) a partir do valor de outra. Em outras palavras, quantificam o quão fortemente duas variáveis estão associadas ou correlacionadas uma com a outra. Uma medida clássica de dependência é o coeficiente de correlação, o qual pode, por exemplo, ser usado para encontrar a correlação entre um atributo e a classe, em um problema de classificação. Se a correlação de um atributo X_i com a classe C é maior que a correlação do atributo X_j com C , então X_i pode ser considerado mais importante que X_j . Uma possível variação é determinar a dependência de um atributo em relação aos outros; esse valor indica o grau de redundância do atributo;

Medidas de Consistência - Possuem características diferentes das outras medidas, pois são fortemente dependentes do conjunto de treinamento e preferem hipóteses consistentes que possam ser definidas a partir do menor número possível de atributos. Assim, essas medidas encontram o subconjunto mínimo de atributos que satisfaz a proporção de inconsistência aceita, geralmente definida pelo usuário. Porém, um problema associado às medidas de consistência é que elas não conseguem distinguir entre dois atributos igualmente bons e, conseqüentemente, não conseguem detectar atributos redundantes. A inconsistência é definida como dois exemplos possuindo os mesmos valores de atributos, mas classes diferentes e

Medidas de Precisão - Referem-se a tarefas de predição. Dados um determinado algoritmo de aprendizado e os diversos subconjuntos de atributos, o que maior precisão proporcionar ao modelo gerado será selecionado. Desse modo, é natural a utilização do mesmo algoritmo que irá processar o conjunto de exemplos com os atributos selecionados para realizar a tarefa de seleção de atributos.

Os três primeiros tipos de medidas estão intimamente relacionados, sendo portanto possível agrupar as medidas apresentadas em três principais categorias:

1. medidas clássicas (informação, distância e dependência);
2. medidas de consistência e
3. medidas de precisão.

As medidas clássicas e de consistência podem ainda ser agrupadas já que tratam da separabilidade de classes no caso de aprendizado supervisionado, ou *clusters* no caso de aprendizado não supervisionado, como ilustrado na Figura 4.1, baseada em Liu and Motoda (1998).

Figura 4.1: Hierarquia de tipos de medidas de avaliação de atributos (Liu and Motoda, 1998)

É importante notar que as medidas de precisão são dependentes do algoritmo de aprendizado considerado, pois os subconjuntos de atributos serão importantes em relação à precisão do modelo (classificador) induzido por um determinado algoritmo. Já as medidas de separabilidade de classes/grupos são independentes do algoritmo de aprendizado escolhido para a posterior construção do modelo.

Existem diferentes definições na literatura para determinar a importância de um atributo. A maioria dessas definições considera a importância do atributo em relação à classe. Neste trabalho o termo *relevância* será associado à importância especificamente em relação à classe. Como destacado anteriormente, é importante tratar também a redundância de atributos. O seguinte exemplo, citado frequentemente na literatura, ilustra esse conceito:

considerando o conjunto $X = \{X_1, X_2, X_3, X_4, X_5\}$ de atributos e $y = f(X_1, X_2)$ uma função booleana, há somente oito possíveis exemplos tal que $X_2 = \overline{X}_3$ e $X_4 = \overline{X}_5$. Assim, para determinar o conceito meta tem-se: X_1 é indispensável; X_2 ou X_3 , mas não ambos, podem ser ignorados já que $y = f(X_1, \overline{X}_3)$; X_4 e X_5 podem ser ignorados. Nesse caso, existem dois subconjuntos X' ótimos, $\{X_1, X_2\}$ e $\{X_1, X_3\}$, e a meta da SA é encontrar pelo menos um desses subconjuntos. Entretanto, o número de subconjuntos de atributos cresce exponencialmente com o número de atributos em X e encontrar o subconjunto ótimo de atributos pode ser NP (Kohavi and John, 1997).

Nas seções seguintes são apresentadas algumas das definições sobre importância de atributos propostas por diversos autores, as quais foram unificadas para a terminologia utilizada neste trabalho, e uma classificação da medida a qual pertencem, segundo a hierarquia apresentada na Figura 4.1. Nessas definições é considerado $x_i \in \{x_{1i}, x_{2i}, \dots, x_{Ni}\}$. Notar que x_i pode ser qualquer um dos valores assumidos por um determinado atributo no conjunto de dados, enquanto $\vec{x}_i$ representa o vetor de valores de todos atributos de um exemplo i em particular, *i.e.*, $\vec{x}_i = \langle x_{i1}, x_{i2}, \dots, x_{iM} \rangle$ — Seção 2.2 na página 11.

4.2.1 Importância em Relação à Consistência

Uma definição bastante natural está ligada à idéia de importância em relação a um objetivo. Em uma tarefa de classificação o objetivo poderia ser considerado o conceito meta, como mostra a seguinte definição:

Definição 4.2.2 (*Importância em relação ao Conceito Meta – Medida de Consistência*) (Almuallim and Diettrich, 1991) Um atributo X_i é importante para o conceito meta f se X_i aparece em toda fórmula booleana que representa f e não importante caso contrário.

Essa definição de importância é muito simples e considera que tanto os atributos quanto a classe são booleanos, além da não existência de ruído nos dados. Por exemplo, considere que um determinado algoritmo de aprendizado $\mathcal{I}$ gerou um classificador composto pelo seguinte conjunto de regras:

If $X_1=1$ **and** $X_2=0$ **then** classe=1
If $X_1=1$ **and** $X_3=0$ **then** classe=1
If $X_1=0$ **and** $X_2=1$ **then** classe=0

Segundo essa definição, o atributo X_1 seria importante, pois ele aparece em todas as regras que descrevem o conceito, enquanto X_2 e X_3 seriam não importantes.

Uma outra definição que emprega a medida de consistência como critério de avaliação de atributos é a seguinte:

Definição 4.2.3 (*Importância em relação ao Conceito Meta — Medida de Consistência*) (Dash and Liu, 2003; Liu and Setiono, 1996) Um subconjunto de atributos importantes é definido por meio da **taxa de inconsistência** definida como:

1. um exemplo é considerado **inconsistente** se existirem pelo menos dois exemplos exatamente iguais exceto pelo valor da classe;
2. a **contagem de inconsistência** para um exemplo é dada pelo número de vezes que esse exemplo aparece nos dados subtraído o maior número entre as diferentes classes e
3. a **taxa de inconsistência** de um subconjunto de atributos é a soma de todas as contagens de inconsistência de todos os exemplos do subconjunto nos dados dividido pelo número N de exemplos.

Por exemplo, se para um determinado subconjunto de atributos, um exemplo E_i inconsistente aparece N_{E_i} vezes dos quais N_{C_1} pertencem à classe C_1 , N_{C_2} pertencem à classe C_2 e N_{C_3} pertencem à classe C_3 , sendo $N_{E_i} = N_{C_1} + N_{C_2} + N_{C_3}$. Se N_{C_3} é o maior valor entre os três, a contagem de inconsistência é dada $(N_{E_i} - N_{C_3})$. Desse modo, dados um subconjunto de atributos e um limiar mínimo de taxa de inconsistência, definida pelo usuário, caso a taxa de inconsistência desse subconjunto seja menor que o limiar, ele será dito consistente. Em geral, essa medida é combinada com alguma outra, por exemplo o tamanho do subconjunto de atributos em questão.

4.2.2 Importância em Relação à Dependência

Uma outra definição muito simples, que classifica atributos importantes como aqueles que variam sistematicamente com a classe associada, é apresentada a seguir.

Definição 4.2.4 (*Importância Probabilística — Medida de Dependência*) (Gennari et al., 1989) Um atributo X_i é importante sss¹ existe algum x_i e y para os quais $P(X_i = x_i) > 0$ tal que²

$$P(Y = y | X_i = x_i) \neq P(Y = y)$$

Segundo essa definição, X_i é importante, se para algum de seus valores, a estimativa para a classe Y é modificada ou, em outras palavras, se Y é condicionalmente dependente de X_i .

Por exemplo, considerando o atributo X_1 na Tabela 4.1 e o valor que ele assume para o exemplo E_1 tem-se:

$$P(Y = + | X_1 = \clubsuit) \neq P(Y = +)$$

¹se e somente se.

²Em geral, as definições dadas aqui aplicam-se a atributos discretos, porém podem ser estendidas para atributos numéricos substituindo-se $P(X = x)$ por $P(X \leq x)$.

pois $P(Y = +|X_1 = \clubsuit) = 1$ enquanto que $P(Y = +) = \frac{3}{4}$, ou seja, X_1 é importante. Pode ser observado que a classe é também condicionalmente dependente dos atributos X_2 e X_3 , que também serão considerados importantes segundo essa definição.

Exemplos	X_1	X_2	X_3	Classe
E_1	$\clubsuit$	$\heartsuit$	$\diamondsuit$	+
E_2	$\clubsuit$	$\clubsuit$	$\diamondsuit$	+
E_3	$\heartsuit$	$\heartsuit$	$\clubsuit$	+
E_4	$\heartsuit$	$\heartsuit$	$\heartsuit$	-

Tabela 4.1: Exemplos para ilustrar as definições de importância

Pode-se notar que essa definição considera apenas o atributo em questão e a classe para o cálculo da importância. Porém, essa definição falha na captura da importância de atributos para problemas com características como as da Função de Paridade³, na qual todos os exemplos são equiprováveis (Weisstein, 2005).

Considere a Função de Paridade definida na Tabela 4.2, na qual $P(Y = 1) = \frac{1}{2}$. Calculando as probabilidades para os atributos X_1 , X_2 , X_3 e a classe Y tem-se:

$$P(Y = 1|X_1 = 1) = P(Y = 0|X_1 = 1) = P(Y = 1|X_1 = 0) = P(Y = 0|X_1 = 0) = \frac{1}{2}$$

$$P(Y = 1|X_2 = 1) = P(Y = 0|X_2 = 1) = P(Y = 1|X_2 = 0) = P(Y = 0|X_2 = 0) = \frac{1}{2}$$

$$P(Y = 1|X_3 = 1) = P(Y = 0|X_3 = 1) = P(Y = 1|X_3 = 0) = P(Y = 0|X_3 = 0) = \frac{1}{2}$$

Desse modo, para o caso da Função de Paridade

$$P(Y = y|X_i = x_i) = P(Y = y)$$

e portanto nenhum dos três atributos é considerado importante segundo essa definição.

Exemplos	X_1	X_2	X_3	Classe
E_1	0	0	0	0
E_2	0	0	1	1
E_3	0	1	0	1
E_4	0	1	1	0
E_5	1	0	0	1
E_6	1	0	1	0
E_7	1	1	0	0
E_8	1	1	1	1

Tabela 4.2: Exemplos gerados pela função de paridade

Assim, para que possa ser levada em consideração a influência de todos os atributos e o caso em que todos os exemplos são equiprováveis, a Definição 4.2.4 pode ser redefinida da seguinte maneira.

³A Função de Paridade é definida como o total de entradas com valor $1 \bmod 2$, i.e., retorna 1 se o número de entradas iguais a 1 for ímpar e 0 caso contrário.

Seja $S_i = \{ X_1, \dots, X_{i-1}, X_{i+1}, \dots, X_M \}$ o subconjunto de todos os atributos exceto X_i , e s_i uma atribuição de valores a todos os atributos em S_i .

Definição 4.2.5 (*Importância Probabilística — Medida de Dependência*) (John et al., 1994) Um atributo X_i é importante sss existe algum x_i , y e s_i para $P(X_i = x_i) > 0$ tal que

$$P(Y = y, S_i = s_i | X_i = x_i) \neq P(Y = y, S_i = s_i)$$

Na definição seguinte, X_i é importante se a probabilidade da classe, dados todos os atributos, pode mudar quando o conhecimento sobre o valor de X_i é eliminado.

Definição 4.2.6 (*Importância Probabilística — Medida de Dependência*) (John et al., 1994) Um atributo X_i é importante sss existe algum x_i , y e s_i para $P(X_i = x_i, S_i = s_i) > 0$ tal que

$$P(Y = y | X_i = x_i, S_i = s_i) \neq P(Y = y | S_i = s_i)$$

Para os exemplos na Tabela 4.1, os atributos considerados importantes e não importantes pelas Definições 4.2.5 e 4.2.6 não são os mesmos, como mostra a Tabela 4.3.

Definição	X_1	X_2	X_3
Definição 4.2.5 na página 35	Importante	Importante	Importante
Definição 4.2.6 na página 35	Não Importante	Não Importante	Importante

Tabela 4.3: Atributos importantes e não importantes

Para tanto, foram calculadas as probabilidades dos atributos X_1 , X_2 e X_3 para cada um dos possíveis valores assumidos pelos mesmos. Um atributo será então considerado importante, segundo a Definição 4.2.5, se pelo menos um dos valores assumidos por X_i e S_i verificam que

$$P(Y = y, S_i = s_i | X_i = x_i) \neq P(Y = y, S_i = s_i).$$

Por exemplo, para $X_1 = \clubsuit$ segundo a Definição 4.2.5:

$$P(Y = +, S_1 = \{\heartsuit, \diamondsuit\} | X_1 = \clubsuit) \neq P(Y = +, S_1 = \{\heartsuit, \diamondsuit\})$$

Dessa maneira, X_1 é um atributo importante segundo essa definição, pois $P(Y = +, S_1 = \{\heartsuit, \diamondsuit\} | X_1 = \clubsuit) = \frac{1}{2}$ e $P(Y = +, S_1 = \{\heartsuit, \diamondsuit\}) = \frac{1}{4}$. Fazendo o mesmo cálculo para os atributos X_2 e X_3 , pode-se verificar que os mesmos também serão considerados importantes segundo essa definição.

Do mesmo modo, para determinar quais atributos são importantes segundo a Definição 4.2.6, é necessário encontrar apenas um X_i e S_i para os quais

$$P(Y = y|X_i = x_i, S_i = s_i) \neq P(Y = y|S_i = s_i)$$

Assim, X_1 é considerada não importante segundo essa definição pois para nenhum dos valores assumidos por X_1 e S_1 a definição é satisfeita, pois:

$$P(Y = +|S_1 = \{\heartsuit, \diamondsuit\}, X_1 = \clubsuit) = P(Y = +|S_1 = \{\heartsuit, \diamondsuit\}) = 1$$

$$P(Y = +|S_1 = \{\heartsuit, \clubsuit\}, X_1 = \heartsuit) = P(Y = +|S_1 = \{\heartsuit, \clubsuit\}) = 1$$

Igualmente, para os valores assumidos por X_2 , tem-se:

$$P(Y = +|S_2 = \{\clubsuit, \diamondsuit\}, X_2 = \heartsuit) = P(Y = +|S_2 = \{\clubsuit, \diamondsuit\}) = 1$$

$$P(Y = +|S_2 = \{\clubsuit, \diamondsuit\}, X_2 = \clubsuit) = P(Y = +|S_2 = \{\clubsuit, \diamondsuit\}) = 1$$

sendo portanto X_2 também não importante segundo essa definição. Porém, realizando o cálculo para o atributo X_3 pode-se verificar que:

$$P(Y = +|S_3 = \{\heartsuit, \heartsuit\}, X_3 = \clubsuit) \neq P(Y = +|S_3 = \{\heartsuit, \heartsuit\})$$

pois $P(Y = +|S_1 = \{\heartsuit, \heartsuit\}|X_3 = \clubsuit) = 1$ e $P(Y = +|S_3 = \{\heartsuit, \heartsuit\}) = \frac{1}{2}$. Sendo assim, esse atributo é considerado importante segundo essa definição.

Blum and Langley (1997) apresentam uma definição de importância de atributos equivalente à Definição 4.2.6 dada por John et al. (1994).

Definição 4.2.7 (*Importância em relação ao Conceito Meta – Medida de Dependência*) (Blum and Langley, 1997) Um atributo X_i é importante para o conceito meta sss existe um par de exemplos E_i e E_j , $i \neq j$, no espaço de exemplos tal que E_i e E_j diferem somente na atribuição de valores ao atributo X_i e $f(\vec{x}_i) \neq f(\vec{x}_j)$.

Em outras palavras, o atributo X_i é importante se existe algum exemplo no espaço de exemplos para o qual a modificação do valor de X_i afeta a classificação dada pelo conceito meta. Os exemplos E_3 e E_4 na Tabela 4.1 na página 34 ilustram essa definição, com E_3 e E_4 diferindo somente no valor do atributo X_3 e $f(\vec{x}_3) \neq f(\vec{x}_4)$. Assim, segundo essa definição o atributo X_3 seria importante, bem como segundo a Definição 4.2.6.

Observe que a Definição 4.2.7 — Importância em relação ao Conceito Meta — é muito mais restrita que a Definição 4.2.4 na página 33 — Importância Probabilística — pois a primeira exige que todos os valores dos atributos (exceto o valor do atributo em questão e a classe) de dois exemplos quaisquer sejam iguais. Por outro lado, a Definição 4.2.4 é mais abrangente, pois atribui a importância a um dado atributo comparando a probabilidade de uma dada classe ocorrer sob duas situações: considerando a influência daquele atributo ou não.

Note também que a Definição 4.2.6, e a Definição equivalente 4.2.7 na página oposta, têm a desvantagem de que o algoritmo de aprendizado, dado acesso a apenas a amostra S , não é necessariamente capaz de determinar quando algum atributo X_i é importante ou não. Um outro agravante para essa questão é se a representação dos atributos é redundante, pode não ser possível encontrar dois exemplos que diferem apenas pelo valor de um atributo, já que pelo menos um desses exemplos teria probabilidade zero. Por exemplo, considere a Tabela 4.4, na qual os domínios dos valores dos atributos X_1 , X_2 e X_3 são, respectivamente, $D_1 = \{\bullet, \diamond\}$, $D_2 = \{\triangleleft, \natural\}$ e $D_3 = \{\star, \oslash\}$.

Exemplos	X_1	X_2	X_3	Classe
E_1	$\bullet$	$\natural$	$\star$	$+$
E_2	$\diamond$	$\triangleleft$	$\star$	$-$
E_3	$\bullet$	$\natural$	$\oslash$	$+$
E_4	$\diamond$	$\triangleleft$	$\oslash$	$-$

Tabela 4.4: Exemplos para ilustrar a redundância de atributos

Pode-se observar que não é possível encontrar nesse conjunto de exemplos um único par E_i e E_j tal que seja possível determinar um atributo importante, justamente por causa da redundância dos atributos X_1 e X_2 , que verificam

$$X_1 = \bullet \text{ sss } X_2 = \natural \text{ e } X_1 = \diamond \text{ sss } X_2 = \triangleleft$$

Para tentar contornar algumas das desvantagens apresentadas por essa definição, John et al. (1994) colocam a necessidade de distinguir entre dois graus de importância: *forte* e *fraca*, definidas em termos do classificador de Bayes⁴ — o classificador ótimo para um dado problema.

Assim, um atributo X_i é dito *fortemente importante* se a simples remoção desse atributo resultar na deterioração da performance do classificador de Bayes (John et al., 1994). Já um atributo X_i é dito *fracamente importante* se não for fortemente importante e existir um subconjunto de atributos, Z , tal que a performance do classificador de Bayes sobre Z é pior que a performance sobre $Z \cup \{X_i\}$. Ambos os graus de importância são definidos a seguir; observe que a definição de forte importância é equivalente à Definição 4.2.6 na página 35.

Definição 4.2.8 (*Forte Importância — Medida de Dependência*) (John et al., 1994) Um atributo X_i é *fortemente importante* sss existe algum x_i , y e s_i para $P(X_i = x_i, S_i = s_i) > 0$ tal que

$$P(Y = y | X_i = x_i, S_i = s_i) \neq P(Y = y | S_i = s_i)$$

Definição 4.2.9 (*Fraca Importância — Medida de Dependência*) (John et al., 1994) Um atributo X_i é *fracamente importante* sss não é fortemente importante e existe algum

⁴Um classificador de Bayes é uma regra que prevê a classe mais provável para um dado exemplo, baseado na distribuição, assumida como sendo conhecida, do conjunto de dados considerado.

subconjunto S'_i de S_i para o qual existe algum x_i , y e s'_i para $P(X_i = x_i, S'_i = s'_i) > 0$ tal que

$$P(Y = y | X_i = x_i, S'_i = s'_i) \neq P(Y = y | S'_i = s'_i)$$

Em outras palavras, forte importância implica que o atributo é indispensável no sentido que esse atributo não pode ser removido sem a perda de precisão da classificação pelo classificador de Bayes. Por outro lado, fraca importância implica que o atributo pode algumas vezes contribuir para a precisão da predição. Considera-se que um atributo é importante se ele é fortemente ou fracamente importante, caso contrário é dito não importante segundo essa medida de importância.

Para exemplificar essas definições, considere o mesmo conjunto de exemplos apresentados na Tabela 4.1 na página 34. Os atributos X_1 e X_2 são considerados como não sendo de forte importância segundo a Definição 4.2.8 (equivalente à Definição 4.2.6 na página 35). Porém, aplicando-se a definição de fraca importância para o atributo X_1 tem-se que:

$$P(Y = + | S'_1 = \{\heartsuit\}, X_1 = \clubsuit) \neq P(Y = + | S'_1 = \{\heartsuit\})$$

concluindo-se que X_1 é fracamente importante, pois $P(Y = + | S'_1 = \{\heartsuit\}, X_1 = \clubsuit) = 1$ e $P(Y = + | S'_1 = \{\heartsuit\}) = \frac{2}{3}$. No entanto, o mesmo não pode ser comprovado para o atributo X_2 . Portanto, segundo as definições anteriores, os atributos X_1 e X_3 seriam considerados importantes, enquanto que X_2 seria considerado como um atributo não importante.

Blum and Langley (1997), também apresentam definições de forte e fraca importância.

Definição 4.2.10 (*Forte Importância em relação à Amostra/Distribuição — Medida de Dependência*) (Blum and Langley, 1997) Um atributo X_i é fortemente importante para uma amostra S se existe um par de exemplos E_v e E_q , para $v \neq q$, tal que E_v e E_q diferem somente na atribuição de valores ao atributo X_i e possuem diferentes classes (ou possuem diferentes distribuições das classes se esses aparecem múltiplas vezes em S). Similarmente, o atributo X_i é fortemente importante para a meta f e a distribuição $\mathcal{D}$ se existem exemplos E_v e E_q , com probabilidade não nula sobre $\mathcal{D}$, que diferem somente na atribuição de valores ao atributo X_i e satisfazem $f(\vec{x}_v) \neq f(\vec{x}_q)$.

Definição 4.2.11 (*Fraca Importância em relação à Amostra/Distribuição — Medida de Dependência*) (Blum and Langley, 1997) Um atributo X_i é fracamente importante para uma amostra S ou para a meta f e a distribuição $\mathcal{D}$ se for possível remover um subconjunto de atributos tal que o atributo X_i torna-se fortemente importante.

Essas noções de importância são úteis sob o ponto de vista de um algoritmo que deve decidir quais atributos manter e quais ignorar. É importante manter os atributos

fortemente importantes, pois, em geral, removê-los significa introduzir ambigüidade na amostra. Já atributos fracamente importantes podem ou não ser importantes dependendo de quais outros atributos são ignorados.

Para mostrar como as diversas definições de importância podem classificar diferentemente, e muitas vezes de modo inesperado, os mesmos atributos, a seguir é apresentado um exemplo que utiliza o conceito XOR — Tabela 4.5.

Sejam os atributos X_1, X_2, X_3, X_4 e X_5 booleanos tal que X_4 e X_5 são negações de X_2 e X_3 respectivamente, isto é, $X_4 = \overline{X_2}$ e $X_5 = \overline{X_3}$. O espaço de exemplos é composto por apenas oito casos. Assume-se que todos sejam equiprováveis. O conceito meta f é definido por:

$$Y = X_1 \oplus X_2 \text{ (}\oplus \text{ denota XOR)}$$

Exemplos	X_1	X_2	X_3	X_4	X_5	Classe (Y)
E_1	1	1	1	0	0	0
E_2	1	1	0	0	1	0
E_3	1	0	1	1	0	1
E_4	1	0	0	1	1	1
E_5	0	1	1	0	0	1
E_6	0	1	0	0	1	1
E_7	0	0	1	1	0	0
E_8	0	0	0	1	1	0

Tabela 4.5: Exemplos do conceito meta $Y = X_1 \oplus X_2$

Note que o conceito meta possui uma expressão booleana equivalente, $Y = X_1 \oplus \overline{X_4}$. Os atributos considerados importantes e não importantes para esse exemplo, segundo cada uma das definições apresentadas até agora, são mostrados na Tabela 4.6.

Definição	Atributos Importantes	Atributos Não Importantes
Definição 4.2.2 na página 32	X_1	X_2, X_3, X_4, X_5
Definição 4.2.4 na página 33	Nenhum	Todos
Definição 4.2.5 na página 35	Todos	Nenhum
Definição 4.2.6 na página 35	X_1	X_2, X_3, X_4, X_5

Tabela 4.6: Atributos importantes e não importantes segundo as definições de importância

Segundo a Definição 4.2.2, apenas o atributo X_1 é considerado importante, pois $Y = X_1 \oplus X_2$ ou $Y = X_1 \oplus \overline{X_4}$. Já de acordo com a Definição 4.2.4, todos os atributos são considerados não importantes pois todos os exemplos são equiprováveis, isto é, para cada valor assumido por Y e X_i , existem dois exemplos que combinam com esses valores. Segundo a Definição 4.2.5, todos os atributos são considerados importantes pois $P(Y = y, S_i = s_i | X_i = x_i) \neq P(Y = y, S_i = s_i)$ para todos os X_i e S_i , isto é, cada valor de X_i restringe o espaço de exemplos à metade e, sendo cada exemplo único, tem-se que $P(Y = y, S_i = s_i | X_i = x_i) = \frac{1}{4}$. Já $P(Y = y, S_i = s_i) = \frac{1}{8}$, pois como anteriormente dito não existem exemplos iguais para a mesma classe. Finalmente, segundo a Definição 4.2.6,

X_3 e X_5 são claramente não importantes, já que conhecer ou não seus valores não influencia a decisão por um valor da classe ou por outro. O mesmo ocorre com X_2 e X_4 , pois os valores desses atributos não acrescentam nenhuma informação a S_2 e S_4 , respectivamente.

Uma outra definição de importância, a qual permite detectar a redundância de atributos, usa o conceito de Dimensão Fractal — DF — apresentada em maiores detalhes no próximo capítulo. Algumas das aplicações que utilizam a teoria de fractais incluem a determinação de estrutura de indexação de alta dimensionalidade e a detecção de agrupamentos. Porém, a teoria dos Fractais é ainda pouco utilizada no problema de seleção de atributos para algoritmos de aprendizado supervisionado, a qual será pesquisada neste trabalho. Deve ser observado que a DF considera o atributo classe em igualdade de condições aos outros atributos.

Definição 4.2.12 (*Importância em relação à Dimensão Fractal — Medida de Dependência*) (Traina et al., 2000) Dada a Dimensão Fractal⁵, calculada utilizando-se todos os atributos do conjunto de dados, um atributo é dito importante se a sua exclusão causar uma alteração significativa⁶ no valor da Dimensão Fractal.

Utilizando o método *Box Count Plot* para determinar a dimensão fractal e o mesmo conjunto de exemplos apresentado na Tabela 4.5, todos os atributos foram considerados importantes segundo a definição que utiliza como medida de importância a dimensão fractal.

4.2.3 Importância em Relação à Informação

As definições de importância de atributos apresentadas até agora são baseadas em medidas de consistência e dependência — Figura 4.1 na página 31. Uma outra medida de separabilidade de classes/grupos é a medida de informação. A seguir é apresentada uma definição de importância que usa essa medida de avaliação para a determinação da importância de atributos.

Definição 4.2.13 (*Importância em relação à Entropia — Medida de Informação*) (Bell and Wang, 2000) Dados três conjuntos de atributos A , B e C com uma probabilidade conjunta p , seja $I(A; B|C)$ a informação mútua entre A e B dado C e $H(A|B)$ a entropia de Shannon de A dado B . Se $H(B|C) \neq 0$ então a importância de atributos de A para B dado C , denotado por $r_p(A; B|C)$, é definido como:

$$r_p(A; B|C) = \frac{I(A; B|C)}{H(B|C)} = \frac{H(B|C) - H(B|A, C)}{H(B|C)}$$

Se $H(B|C) = 0$ então $r_p(A; B|C) = 0$.

⁵O conceito da dimensão fractal será abordado na Seção 5.2 na página 47.

⁶Dependente da variação que o usuário determinar como significativa.

Essa definição afirma que a importância de A para B dado C é indicada pela redução relativa da incerteza de B quando A e C são conhecidos. Com essa noção pode-se expressar um grau de importância colocando-se que A é importante para B dado C com um grau $r_p(A; B|C)$. Esse é denominado o *caso condicional* no sentido de que a importância entre A e B está condicionada por C e $r_p(A; B|C)$ é então denominada *importância condicional*. Quando C é desconsiderado, a importância entre dois conjuntos de atributos não é condicionada a outro conjunto, desse modo tem-se um *caso incondicional* e $r_p(A; B)$ é denominada de *importância incondicional* e pode ser redefinida como:

$$r_p(A; B) = \frac{I(A; B)}{H(B)} = \frac{H(A) - H(A|B)}{H(B)}$$

Se A é discreto com distribuição de probabilidade $p(A)$, a entropia de Shannon $H(A)$ é definida como:

$$H(A) = - \sum_A p(A) \log_2(p(A))$$

A entropia condicional, a qual quantifica a incerteza restante sobre A dado B , é definida como:

$$H(A|B) = \sum_{AB} p(A, B) \log_2(p(A|B))$$

A informação mútua $I(A; B)$ entre A e B é definida como $I(A; B) = H(B) - H(B|A)$, a qual fornece uma medida da quantidade de informação que um atributo contém sobre o outro.

Definição 4.2.14 (*Importância em relação à Symmetrical Uncertainty — Medida de Informação*) (Press et al., 1992) Considerando a entropia de Shannon $H(A)$, $H(B)$ e $H(B|A)$ e a informação mútua de $I(A; B) = H(B) - H(B|A)$, a importância de A para B é definida como:

$$SU(A; B) = 2 \left[\frac{I(A; B)}{H(A) + H(B)} \right]$$

Essa medida ameniza a preferência da medida de informação mútua por atributos que contenham mais valores e restringe o valor da medida em $[0, 1]$. O valor 1 indica que o valor de um atributo prediz completamente o valor do outro. Por outro lado, o valor 0 indica que os dois atributos são independentes. Além disso, os dois atributos são tratados simetricamente (Yu and Liu, 2004).

4.2.4 Importância em Relação à Distância

Uma definição de importância de atributos empregando a medida de distância — Figura 4.1 na página 31 — é apresentada a seguir.

Definição 4.2.15 (*Importância em relação à Distância — Medida de Distância*) (Robnik-Sikonja and Kononenko, 2003; Kira and Rendell, 1992) Dados dois atributos X_i e X_j , a importância de cada atributo é definida como:

$$W[X_i = x_i | Y = y] = P(X_i = x_i | Y = \neg y) - P(X_i = x_i | Y = y)$$

X_i será mais importante que X_j se $W[X_i | Y = y] > W[X_j | Y = y]$.

Com essa definição de importância de atributos, o objetivo é que bons atributos separem exemplos com diferentes classes e não separem exemplos que apresentam as mesmas classes.

4.2.5 Importância em Relação à Precisão

As definições de importância apresentadas até então são independentes do algoritmo de aprendizado que irá utilizar o subconjunto de atributos selecionado para a descrição dos exemplos de treinamento — Figura 4.1 na página 31. É importante observar que nesse caso não existe garantia de que um atributo, ainda que importante, será útil (ou inútil) para um determinado algoritmo de aprendizado.

Uma definição, bastante natural, visibiliza o problema da seleção de um subconjunto de atributos reduzindo-o ao problema de encontrar um subconjunto ótimo para um determinado algoritmo de aprendizado. Nessa definição é considerada a medida de precisão, a qual é dependente do algoritmo de aprendizado, suas heurísticas e seu *bias*.

Definição 4.2.16 (*Importância em relação à Precisão — Medida de Precisão*) (Kohavi and John, 1997) Dados um algoritmo de aprendizado $\mathcal{I}$ e uma amostra de dados S com atributos $X_1, X_2, \dots, X_M$ e uma distribuição $\mathcal{D}$ sobre o espaço de exemplos classificados, um subconjunto ótimo de atributos, X_{opt} , é um subconjunto de atributos tal que a precisão do classificador induzido $\mathbf{h} = \mathcal{I}(\mathcal{D})$ é máxima.

Deve-se notar que um subconjunto ótimo de atributos não precisa ser único, pois é possível alcançar a mesma precisão utilizando diferentes subconjuntos de atributos. Um exemplo disso seria a substituição de um atributo por outro que é perfeitamente correlacionado à ele.

Um dos problemas de utilizar, na prática, essa definição é que a distribuição $\mathcal{D}$ dos exemplos não é conhecida. Assim, é necessário estimar a precisão do algoritmo de aprendizado utilizando a amostra de exemplos. A definição a seguir contempla esse aspecto.

Definição 4.2.17 (*Utilidade Incremental — Medida de Precisão*) (Caruana and Freitag, 1994) Dada uma amostra de dados S , um algoritmo de aprendizado $\mathcal{I}$ e um subconjunto de atributos F , no qual $\{X_i\} \cap F = \emptyset$, um atributo X_i é incrementalmente útil para $\mathcal{I}$

em relação a F se a precisão da hipótese produzida por $\mathcal{I}$ considerando o conjunto de atributos $\{X_i\} \cup F$ é melhor que a precisão alcançada utilizando-se apenas o subconjunto de atributos F .

Essa definição é especialmente natural para os algoritmos de seleção de atributos que realizam a busca no espaço de subconjuntos de atributos, adicionando e removendo atributos desses subconjuntos, tais como as abordagens *embedded* e *wrapper* descritas na Seção 3.4 na página 24.

É importante ressaltar que, na prática, classificadores gerados por algoritmos de aprendizado podem beneficiar-se da omissão de atributos, inclusive dos atributos de importância forte — Definições 4.2.8 na página 37 e 4.2.10 na página 38. Assim, é interessante observar que a importância de um atributo não implica que ele esteja no subconjunto ótimo de atributos quando a medida de precisão é considerada.

A fim de exemplificar, considere o conjunto de exemplos descrito por três atributos, X_1 , X_2 e X_3 , o qual tem como universo de possíveis exemplos $\{0,1\}^3$ — Tabela 4.7. Seja o conceito meta $f(\vec{x}) = (X_1 \wedge X_2) \vee X_3$. Sob qualquer uma das definições de importância anteriormente apresentadas, todos os três atributos são considerados importantes.

Exemplos	X_1	X_2	X_3	Classe
E_1	1	1	1	1
E_2	1	1	0	1
E_3	1	0	1	1
E_4	1	0	0	0
E_5	0	1	1	1
E_6	0	1	0	0
E_7	0	0	1	1
E_8	0	0	0	0

Tabela 4.7: Exemplos para ilustrar que um atributo importante não é necessariamente ótimo

Nesse caso, porém, o único subconjunto ótimo de atributos, segundo a definição de importância em relação a precisão, é $\{X_3\}$, pois utilizando X_3 para prever a classe tem-se uma precisão de $\frac{7}{8}$, isto é, a máxima precisão para todos os possíveis subconjuntos de atributos. Empregando-se qualquer outro atributo ou qualquer outro subconjunto de atributos a precisão será no máximo de $\frac{5}{8}$. Desse modo, um atributo ser considerado importante não implica que ele estará necessariamente no subconjunto ótimo de atributos.

Sob um ponto de vista diferente das definições anteriormente apresentadas, em muitos casos, ao invés de determinar exatamente quais atributos são importantes, o que se quer é empregar a importância como uma *medida de complexidade*. Isso significa que quer-se utilizar a importância para indicar “o quão complicada” uma função é, ou seja, deseja-se que o algoritmo de aprendizado tenha uma boa performance quando o valor dessa medida for baixo. A definição apresentada a seguir considera essa medida.

Definição 4.2.18 (*Importância como uma Medida de Complexidade — Medida de Precisão*) (Blum and Langley, 1997) Dada uma amostra de dados S e um conceito meta f , definir $r(S, f)$ como o menor número de atributos importantes para f de acordo com a Definição 4.2.7 na página 36, tal que o erro sobre S seja o mínimo possível para o algoritmo de aprendizado.

Em outras palavras, o que se quer é o menor número de atributos necessários para se alcançar uma performance ótima sobre S por meio do conceito representado por f . A razão pela qual especifica-se f é a possibilidade da existência de um atributo, tal como o CPF de uma pessoa, o qual é altamente importante sob o ponto de vista do conteúdo da informação, porém, esse atributo pode ser inútil dependendo do tipo de modelo que está sendo procurado.

Uma outra questão importante está relacionada a “o que medir: um atributo ou vários atributos?” durante o processo de avaliação para a adição ou exclusão desses atributos no subconjunto de atributos importantes. Essa questão está relacionada aos diferentes *biases* e usos dos atributos por parte dos algoritmos; se univariado ou multivariado. É portanto importante notar que a escolha do algoritmo de seleção de atributos depende da necessidade da realização de SA.

4.3 Dimensões da Seleção de Atributos

Como mencionado anteriormente, a busca, considerando-se direção e estratégia, representa um papel importante no problema de seleção de atributos. Uma outra questão também importante para a seleção de atributos está relacionada ao critério de avaliação dos atributos. A determinação do critério de parada está, em geral, associada à uma combinação particular entre a busca e o critério de avaliação. Desse modo, é possível definir uma estrutura tridimensional, ilustrada na Figura 4.2, que representa o espaço de características de métodos de SA de acordo com a direção e a estratégia da busca e o critério de avaliação apresentados neste trabalho.

Existem 27 possíveis combinações de métodos de seleção de atributos considerando-se todas as possibilidades nesse espaço tridimensional. As abordagens híbridas, mencionadas anteriormente, são aquelas que necessitam de mais de um ponto em uma das coordenadas para serem caracterizadas (Molina et al., 2002).

Analisando os métodos de SA do ponto de vista da saída fornecida, eles podem ser agrupados em duas categorias. Em uma das categorias os atributos são ordenados segundo algum critério de avaliação, *i.e.*, o método fornece como saída um *Ranking* dos atributos. A outra categoria está relacionada à escolha de um conjunto mínimo de atributos que satisfazem um critério de avaliação.

Figura 4.2: As três principais dimensões da seleção de atributos: estratégia de busca, medidas de avaliação e direção da busca (Liu and Motoda, 1998)

4.4 Considerações Finais

Neste capítulo foram apresentadas diversas definições de medidas utilizadas na literatura para determinar a importância de atributos. Não importa a escolha da abordagem para seleção de atributos: avaliação individual ou avaliação de subconjuntos, a necessidade de estimativa da importância de atributos é uma questão comum a ser tratada.

Uma grande parte dos algoritmos que realizam SA, o fazem considerando a importância dos atributos em relação à classe apenas, muitas vezes não tratando o problema de atributos redundantes. No próximo capítulo é apresentado o algoritmo proposto neste trabalho, o qual seleciona atributos realizando as análises de relevância e redundância de atributos separadamente utilizando a dimensão fractal para remover atributos redundantes.

Capítulo 5

O Algoritmo FDimBF — *Fractal Dimension-Based Filter*

5.1 Considerações Iniciais

Grande parte dos algoritmos existentes para seleção de atributos, os quais tratam tanto relevância quanto redundância de atributos, o fazem por meio da avaliação de subconjuntos de atributos. Embora esses métodos apresentem, usualmente, melhores resultados que os que não consideram o problema de atributos redundantes, o alto custo computacional pode torná-los ineficientes para conjuntos de dados com alta dimensionalidade. Recentemente, foi proposto o uso da abordagem filtro considerando o modelo de tratamento da relevância e da redundância de atributos como dois procedimentos separados. Esse modelo apresenta a vantagem, sobre a abordagem habitual, de que separando-se a determinação de que atributos são relevantes da procura por atributos não redundantes, o custo computacional da busca por um subconjunto que aproxima o subconjunto ótimo pode ser diminuído.

Neste capítulo é proposto um algoritmo, baseado no modelo de seleção de atributos que separa a análise de relevância e de redundância em duas etapas separadas, que considera a dimensão fractal do conjunto de dados para remover os atributos redundantes. Ainda que o conceito de DF seja freqüentemente utilizado na detecção de agrupamentos de dados e na indexação de estruturas de alta dimensionalidade, não é de nosso conhecimento que esse conceito tenha sido utilizado na tarefa de seleção de atributos para algoritmos de aprendizado de máquina supervisionados, como proposto neste trabalho.

5.2 Fractais

Fractais são definidos pela propriedade de auto-similaridade, ou seja, apresentam, parcial ou integralmente, as mesmas características para diferentes variações na escala em que

estão sendo analisados. Assim, partes do fractal, o qual pode ser uma estrutura, um objeto ou um conjunto de dados, são similares, exata ou estatisticamente, ao fractal como um todo. Fractais possuem, em geral, características incomuns, por exemplo, o conhecido Triângulo de Sierpinsky — Figura 5.1. Ele não pode ser considerado um objeto Euclidiano unidimensional, pois possui perímetro infinito, nem tão pouco um objeto Euclidiano bidimensional já que possui área nula. Dessa maneira, pode-se considerar uma dimensão fracionária, denominada de Dimensão Fractal (Mandelbrot, 1985).

Figura 5.1: Triângulo de Sierpinsky

Fractais estatisticamente auto-similares podem ser encontrados na natureza, por exemplo formações de nuvens, folhas e flores, topografias e cadeias de montanhas, entre outros. Muitos dos conjuntos de dados reais comportam-se como fractais. Desse modo, torna-se natural a idéia de aplicar conceitos da teoria dos fractais para a análise desses conjuntos de dados (Traina et al., 2005; Mandelbrot, 1985).

5.3 Dimensão Fractal de um Conjunto de Dados

A utilização do conceito de Dimensão Fractal está associada à existência de redundância nos conjuntos de dados e da possibilidade desses conjuntos serem bem aproximados em dimensões menores. A idéia principal é empregar a DF do conjunto de dados, a qual é relativamente não afetada por atributos redundantes, para determinar a quantidade e quais são os atributos não redundantes segundo o critério de DF (Sousa et al., 2002).

Pode-se definir, desse modo, os conceitos de *dimensão imersa* e *dimensão intrínseca*. O primeiro conceito corresponde à dimensão do espaço de endereçamento, ou seja, o número de atributos do conjunto de dados. Porém, o conjunto de dados pode estar representando um objeto que possui uma dimensão menor que a do espaço em que está imerso. Assim, a dimensão intrínseca é a dimensão espacial do objeto representado pelo conjunto de dados. Conceitualmente, se um conjunto de dados possui todas as suas variáveis (atributos) independentes umas das outras, então sua dimensão intrínseca será igual a sua dimensão imersa. Porém, toda vez que existir uma correlação entre duas ou mais variáveis, a dimensão intrínseca do conjunto de dados é reduzida de acordo. Usualmente, correlações entre os atributos ou a própria existência dessas correlações não é conhecida. Por meio da dimensão intrínseca do conjunto de dados é possível decidir quantos atri-

butos são necessários para caracterizá-lo. Diferentes tipos de correlação podem reduzir a dimensão intrínseca em diferentes proporções, até mesmo em proporções fracionárias. Desse modo, pode-se utilizar o conceito de Dimensão Fractal como sendo a dimensão intrínseca do conjunto de dados (Traina et al., 2000).

Existem diversas medidas para a DF. Para fractais exatamente auto-similares, *i.e.*, que podem ser caracterizados por meio de regras de construção bem definidas, a Dimensão Fractal é dada pela Equação 5.1:

$$D = \frac{\log(R)}{\log(\frac{1}{\epsilon})} \quad (5.1)$$

onde R representa a quantidade de réplicas e $\frac{1}{\epsilon}$ em que escala as réplicas são geradas a cada iteração.

Para o exemplo do triângulo de Sierpinsky mencionado na Seção 5.2, a DF seria $D = \log(3)/\log(2) = 1,58496$, pois são geradas três réplicas em escala $1:\frac{1}{2}$ a cada iteração — Figura 5.2.

Figura 5.2: Construção do Triângulo de Sierpinsky

Para fractais estatisticamente auto-similares, como conjuntos de dados reais, uma das maneiras de definir a DF é dada pela Dimensão Fractal de Correlação D_2 , que pode ser calculada pelo método *Box-Count Plot* (Faloutsos and Kamel, 1994). Nesse método, a idéia consiste, primeiramente, na construção de um reticulado sobre o conjunto de dados de células de lado r . Então, conta-se o número de pontos dentro da i -ésima célula de tamanho r , denominado $C_{r,i}$. A Dimensão Fractal de Correlação D_2 é definida pela Equação 5.2:

$$D_2 = \frac{\partial \log(S_2(r))}{\partial \log(r)}, r \in [r_{min}, r_{max}] \quad (5.2)$$

onde

$$S_2(r) = \sum_i C_{r,i}^2 \quad (5.3)$$

Em teoria, fractais exatamente auto-similares são infinitos. Na prática, conjuntos de dados reais, os quais possuem um número finito de pontos, são considerados fractais estatisticamente auto-similares para um determinado intervalo de escala $r \in [r_{min}, r_{max}]$ se obedecem uma regra de construção bem definida nesse intervalo. Desse modo, a dimensão

intrínseca de um determinado conjunto de dados pode ser medida como o coeficiente angular da reta que melhor se ajusta ao trecho linear do gráfico em escala logarítmica de $S_2(r)$ por r (Traina et al., 2000). Neste trabalho, o termo Dimensão Fractal de Correlação será simplesmente denominado de Dimensão Fractal.

5.4 Descrição do Algoritmo *Fractal Dimension-Based Filter*

O algoritmo proposto neste trabalho para a seleção de atributos (Lee et al., 2005c,d,e; Lee and Monard, 2003), denominado de *Fractal Dimension-Based Filter* — FDimBF —, pertence à abordagem filtro e segue o modelo proposto por Yu and Liu (2004), ilustrado na Figura 5.3. Nesse modelo a seleção de atributos é realizada em duas etapas:

- primeiramente é executada a *análise de relevância* para determinar o subconjunto de atributos relevantes em relação à classe, removendo os atributos irrelevantes e
- na segunda etapa, por meio da *análise de redundância*, são determinados e removidos os atributos numéricos redundantes a partir do subconjunto que contém apenas os atributos relevantes, produzindo o subconjunto final de atributos selecionados.

Figura 5.3: Modelo para seleção de atributos (Yu and Liu, 2004)

O algoritmo de Yu and Liu (2004), *Fast Correlation-Based Filter* — FCBF —, utiliza a medida *Symmetrical Uncertainty* (Press et al., 1992) como a medida de correlação para aproximar tanto a análise de relevância quanto a análise de redundância. O FCBF apresenta a vantagem, sobre as abordagens tradicionais para avaliação de subconjuntos de atributos, de que por meio da separação das tarefas de análise de relevância e de redundância, o alto custo da busca por subconjuntos de atributos pode ser amenizado.

FDimBF é descrito pelo Algoritmo 5.1. Nesse algoritmo a análise de relevância pode ser realizada utilizando qualquer medida, denominada de medida de importância MI , que permita medir a importância de cada atributo em relação ao atributo meta, *i.e.*, à classe. Essa etapa é realizada no algoritmo no trecho identificado nas linhas 3 a 7. É importante observar que no algoritmo FDimBF propomos, nessa etapa, a realização de avaliação individual dos atributos em relação à classe. Desse modo, é possível manter um procedimento de menor complexidade computacional nessa etapa, no qual um maior

número de atributos é analisado. Na segunda etapa, linhas 9 a 12, são selecionados apenas os atributos não redundantes a partir dos atributos escolhidos como relevantes na primeira etapa. Atributos redundantes, considerando a dimensão fractal, podem ser definidos como aqueles que quando excluídos do conjunto de dados não causam uma modificação significativa no valor da DF recalculada, denominada de Dimensão Fractal Parcial pD .

Algoritmo 5.1 Algoritmo *Fractal Dimension-Based Filter* — FDimBF

Require: $E = \{E_1, E_2, \dots, E_N\}$, um conjunto de dados composto por N exemplos descritos por M atributos $X = \{X_1, X_2, \dots, X_M\}$ e rotulados com os respectivos valores $y_i, i = 1 \dots N, y_i \in \{C_1, C_2, \dots, C_{N_{Cl}}\}$ do atributo classe Y

Ensure: $X_{otimo} \subseteq X$, subconjunto “ótimo” de atributos relevantes e não redundantes

- 1: // *Análise de relevância utilizando a medida de importância MI*
- 2: $X' = \emptyset$
- 3: **for all** $X_i \in X$ **do**
- 4: **if** X_i é relevante em relação à Y usando a medida de importância MI **then**
- 5: $X' = X' \cup \{X_i\}$;
- 6: **end if**
- 7: **end for**
- 8: // $X' \subseteq X$, tal que X' contém os atributos relevantes do conjunto de exemplos E
- 9: L = conjunto dos M exemplos E descritos apenas pelos atributos relevantes em X' segundo a medida de importância MI , i.e. sem o atributo classe Y ;
- 10: // *Calcular a dimensão fractal D do conjunto L e encontrar o conjunto de atributos não redundantes X_{otimo}*
- 11: $D = \text{DimensaoFractal}(L)$;
- 12: $X_{otimo} = \text{AtributosNaoRedundantes}(L, X', D)$;
- 13: **Return** X_{otimo} .

A busca por atributos não redundantes é realizada de modo *backward* da seguinte maneira: inicialmente é calculado o valor da dimensão fractal, D , a partir do conjunto de dados contendo os atributos relevantes selecionados na etapa de análise de relevância — função *DimensaoFractal* na linha 11 do Algoritmo 5.1. Após determinar a DF do conjunto de atributos relevantes, é necessário determinar quais são os $\lceil D \rceil$ atributos não redundantes. Isso é realizado pela função *AtributosNaoRedundantes* na linha 12 do Algoritmo 5.1, a qual é descrita pelo Algoritmo 5.2.

A função *AtributosNaoRedundantes* calcula o valor de pD , ignorando um atributo por vez. Em outras palavras, a pD é calculada tomando-se em consideração todos os atributos exceto o j -ésimo atributo sob observação. Para cada j -ésimo atributo não considerado, é calculada a pD dos outros atributos com o objetivo de encontrar, em cada iteração, o atributo que permite a diferença mínima entre D e pD . Esse valor indica a contribuição desse atributo para caracterizar o conjunto de dados. Assim, o atributo com mínimo valor $D - pD$, é inserido na cabeça de uma lista ordenada *ListaOrdenadaAtributos*, é retirado do conjunto de atributos e o processo é repetido até não haver mais atributos para analisar — linhas 2 até 12 do Algoritmo 5.2. Ao final, são selecionados os primeiros

Algoritmo 5.2 Algoritmo *AtributosNaoRedundantes*

Require: Conjunto de dados L descritos pelos atributos em X' cuja dimensão fractal é D

Ensure: $X_{otimo} \subseteq X'$, subconjunto de atributos não redundantes segundo a dimensão fractal

```

1: ListaOrdenadaAtributos = [ ] ;
2: while  $X' \neq \emptyset$  do
3:    $AttribDiff = \emptyset$ ;
4:   for all  $X_j \in X'$  do
5:     Computar a dimensão parcial  $pD_{X_j}$  do atributo  $X_j$ , i.e. considerando todos os
       atributos em  $X'$  exceto o atributo  $X_j$ ;
6:      $AttribDiff = \{(X_j, D - pD_{X_j})\} \cup AttribDiff$ ;
7:   end for
8:   Selecionar  $X_a$  do conjunto de elementos  $(X_i, D - pD_{X_i})$  de  $AttribDiff$  tal que
        $D - pD_{X_i}$  é mínimo;
9:   // Inserir  $X_a$  na cabeça da lista ListaOrdenadaAtributos
10:   $ListaOrdenadaAtributos = [X_a | ListaOrdenadaAtributos]$ ;
11:   $X' = X' - \{X_a\}$ ;
12: end while
13:  $X_{otimo}$  é o conjunto que contém os  $\lceil D \rceil$  primeiros elementos (atributos) da lista
    ListaOrdenadaAtributos;
14: Return  $X_{otimo}$ .
```

$\lceil D \rceil$ elementos (atributos) da lista *ListaOrdenadaAtributos*, os quais apresentam a maior contribuição para a caracterização do conjunto de dados com dimensão fractal D — linha 13 do Algoritmo 5.2.

Como pode ser observado, a complexidade do Algoritmo 5.1 está relacionada à complexidade da análise de relevância dos atributos utilizando uma determinada medida de importância MI , bem como à complexidade da seleção de atributos não redundantes.

Para realizar a análise de relevância na primeira etapa do algoritmo FDimBF, entre as diversas medidas de importância de atributos existentes, propomos o uso de duas medidas:

1. medida baseada em informação, algoritmo FDimBF(1) e
2. medida baseada em distância, algoritmo FDimBF(2).

Neste trabalho, a análise de relevância em relação ao atributo classe é realizada por meio da utilização dos algoritmos $\mathcal{C}4.5$ (Quinlan, 1993) e ReliefF (Robnik-Sikonja and Kononenko, 2003), os quais implementam internamente o uso de medidas de informação e de distância para avaliar atributos — Figura 5.4. Especificamente, para a análise de relevância utilizando a medida de informação, utilizamos a árvore construída pelo algoritmo $\mathcal{C}4.5$, o qual considera a medida de informação *razão de ganho de informação* para determinar quais atributos são mais importantes para o particionamento dos dados. Desse modo, os atributos são classificados de acordo com o número de vezes que aparecem nas

regras construídas a partir da árvore de decisão induzida. Para um conjunto de N exemplos descritos por M atributos, a complexidade de $\mathcal{C}4.5$ é $O(N \cdot \log N \cdot M)$ — Seção 6.3 na página 60.

Para medir a relevância dos atributos em relação à classe usando uma medida de distância, utilizamos o algoritmo ReliefF para ordenar os atributos. Esse algoritmo procura pelos exemplos mais próximos da mesma classe e de classes diferentes, utilizando a distância de *Manhattan*, e atribui pesos aos atributos de acordo com quão bem eles diferenciam esses exemplos. Esse processo é repetido m vezes. Em geral, m é definido em função do número de exemplos presentes no conjunto de dados. A complexidade de ReliefF é $O(m \cdot N \cdot M)$ — Seção 6.3 na página 60.

Figura 5.4: Visão geral do Algoritmo FDimBF

Como mencionado anteriormente, para tratar a análise de redundância, neste trabalho propomos a utilização da dimensão fractal. A principal idéia é descartar atributos que não afetam a dimensão fractal do conjunto de dados, considerando que essa medida é, relativamente, pouco influenciada por atributos redundantes (Traina et al., 2005; Sousa et al., 2002). Para esse propósito, foi utilizado neste trabalho o algoritmo *Fractal Dimension Reduction* — FDR (Traina et al., 2000), o qual está disponível na ferramenta *Measure Distance Exponent* — MDE (Traina et al., 2003). Esse algoritmo encontra o atributo que menos afeta a DF quando ele não é considerado para representar os dados. Considerando o pior caso, no qual não há atributos irrelevantes e os M atributos devem ser processados na segunda etapa, a complexidade de FRD é $O(N \cdot M^2)$ (Traina et al., 2000). Nesse caso, o FDR é executado M vezes pela função *AtributosNaoRedundantes* na linha 12 no algoritmo FDimBF — Algoritmo 5.1 na página 51 — descrita pelo Algoritmo 5.2, com o objetivo de ranquear todos os M atributos para selecionar os $\lceil D \rceil$ atributos não redundantes que descrevem os exemplos. Assim, a complexidade de *AtributosNaoRedundantes* é $O(N \cdot M^3)$.

Especificamente, para calcular a dimensão fractal para cada subconjunto de atributos avaliados, o FDR utiliza o algoritmo *Linear Box-Occupancy Counter* — LiBOC — (versão 04-21-2005) (Traina et al., 2005) — linha 11 no Algoritmo 5.1 na página 51.

Esse algoritmo, cuja complexidade é $O(M)$, é baseado em uma estrutura de reticulado multi-nível, a qual permite que o conjunto de dados seja lido apenas uma vez para o processamento de $S_2(r) = \sum_i C_{r,i}^2$. Portanto, a complexidade do algoritmo FDimBF, independentemente da medida utilizada para realizar a análise de relevância, é $O(N.M^3)$.

Ressalta-se que muitos dos algoritmos de SA tratam, internamente, apenas atributos nominais. Assim, se o conjunto de dados contém atributos numéricos, eles são discretizados pelo algoritmo antes de efetivamente realizar a SA. Esse é o caso de alguns dos algoritmos, como FCBF (*Fast Correlation-Based Filter*) (Yu and Liu, 2004) e CFS (*Correlation-based Feature Selection*) (Hall, 2000), utilizados neste trabalho para realizar a avaliação experimental. Por outro lado, o algoritmo por nós proposto trata efetivamente atributos numéricos durante a análise de redundância, *i.e.*, sem a necessidade que eles sejam discretizados. Assim, atributos nominais são tratados pelo algoritmo FDimBF somente durante a análise de relevância — Figura 5.3 — pois a DF, utilizada para tratar a redundância de atributos, exige que os mesmos sejam numéricos.

Deve ser observado que a discretização constitui uma função sobrejetora, *i.e.*, diversos valores originais dos atributos são mapeados para um mesmo valor discretizado e não é possível reverter o processo, ou seja, a partir do valor discretizado não é possível retornar ao valor original do atributo. Embora a discretização constitua um processo importante de transformação de atributos, quando aplicado ao contexto de seleção de atributos como ocorre nos algoritmos considerados neste trabalho, é importante observar que nesses algoritmos, os atributos numéricos são discretizados internamente pelo algoritmo e, então, a seleção dos atributos é realizada. Desse modo, o resultado da SA é dado em função da análise dos atributos discretizados, porém, o usuário não tem conhecimento dos parâmetros utilizados para a realização desse processo interno de discretização. Após, a construção dos modelos a partir dos atributos selecionados por esses algoritmos é realizada usando os atributos originais nominais e numéricos (*i.e.* não discretizados) selecionados por esses algoritmos.

Como mencionado, o algoritmo FDimBF trata efetivamente atributos numéricos durante a análise de redundância não sendo necessária a discretização desses atributos. Porém, para atributos nominais não é possível calcular a dimensão fractal. Desse modo, para o caso em que os conjuntos de dados apresentem ambos atributos numéricos e nominais, é proposta uma modificação na metodologia utilizada pelo algoritmo FDimBF para a SA, como mostra a Figura 5.5

As duas etapas, nas quais a SA é realizada, ocorrem do seguinte modo:

1. análise de relevância: realizada considerando o conjunto original de atributos (Num1: numéricos e Nom1: nominais), utilizando as medidas de informação e distância e
2. análise de redundância: realizada somente para os atributos numéricos selecionados como relevantes pela etapa de análise de relevância (Num2), utilizando a dimensão

Figura 5.5: Seleção de atributos utilizando o algoritmo FDimBF para conjuntos de dados contendo atributos numéricos e nominais

fractal.

Ao final dessas duas etapas de seleção, os atributos considerados como importantes pelo algoritmo FDimBF serão os atributos selecionados como relevantes e não redundantes (Nom2: nominais relevantes e Num3: numéricos relevantes e não redundantes). É importante notar que, desse modo, o algoritmo FDimBF, quando utilizado para conjuntos de dados contendo atributos numéricos e nominais, seleciona atributos nominais relevantes e atributos numéricos relevantes e não redundantes.

Na Seção 4.3 na página 44 foi apresentada uma estrutura tridimensional, na qual os métodos de seleção de atributos podem ser situados. Essa estrutura é definida pela direção e estratégia da busca e pelo critério de avaliação de importância dos atributos. O algoritmo proposto neste trabalho pode ser localizado dentro desse espaço de características de métodos de SA como é mostrado na Figura 5.6, na qual as características de FDimBF são ressaltadas.

5.5 Considerações Finais

Neste capítulo foi apresentado o algoritmo, FDimBF, proposto neste trabalho para a seleção de atributos utilizando a dimensão fractal como medida de importância para a remoção de atributos redundantes. Esse algoritmo escolhe os atributos importantes em duas etapas: primeiramente seleciona os atributos relevantes, *i.e.*, importantes em relação à classe; posteriormente, remove os atributos redundantes segundo a dimensão fractal do conjunto de dados. Ao final do processo de seleção de atributos, somente atributos relevantes e atributos numéricos não redundantes farão parte do subconjunto de atributos selecionados. É importante ressaltar que a separação das análises de relevância e redundância pode auxiliar na diminuição do custo computacional pela busca de subconjuntos de atributos importantes.

Figura 5.6: Localização do algoritmo FDimBF dentro do espaço de características dos métodos de seleção de atributos

No próximo capítulo é apresentada uma avaliação experimental do algoritmo proposto juntamente com diversos algoritmos freqüentemente citados na literatura para a realização de seleção de atributos.

Capítulo 6

Avaliação Experimental

6.1 Considerações Iniciais

Embora as características gerais de um conjunto de dados possam prover uma idéia de que subconjunto de algoritmos poderiam produzir melhores resultados, não há análise matemática capaz de determinar que um algoritmo de aprendizado construirá melhores modelos que outros algoritmos (Dietterich, 1989; Langley, 1988). O mesmo ocorre quando se trata de algoritmos de seleção de atributos. Desse modo, avaliações experimentais constituem um importante instrumento da estimativa de que algoritmo(s) seria(m) mais apropriado(s) para a tarefa de seleção de atributos.

Associada à questão da avaliação de modelos está a questão de que parâmetros considerar para determinar quais algoritmos apresentaram uma performance melhor que outros. Usualmente, avalia-se separadamente medidas derivadas da experimentação desses algoritmos. Em geral, a medida mais considerada é o erro associado ao modelo construído usando os atributos selecionados pelos algoritmos de SA. Porém, examinar somente o erro do modelo gerado não é suficiente, é necessário combinar outros parâmetros como a percentagem de redução da quantidade de atributos obtida com a seleção de atributos.

Neste capítulo é apresentada uma avaliação experimental do algoritmo FDimBF, juntamente com outros algoritmos de SA freqüentemente citados na literatura utilizando conjuntos de dados naturais, os quais são amplamente considerados para a realização de avaliações empíricas da performance de algoritmos de SA. É também proposto um modelo para a avaliação da performance de algoritmos de seleção de atributos, o qual combina o erro do modelo construído a partir do subconjunto de atributos selecionados e a percentagem de redução da quantidade de atributos desse subconjunto em relação ao conjunto original.

6.2 Descrição dos Conjuntos de Dados

Os conjuntos de dados utilizados para a realização dos experimentos apresentados a seguir, foram selecionados a partir de uma minuciosa pesquisa bibliográfica de trabalhos publicados na área de seleção de atributos, os quais são freqüentemente referenciados pela comunidade. Nesses trabalhos são utilizados conjuntos de dados reais, naturais e artificiais, sendo:

- reais: extraídos diretamente de bases de dados, por exemplo, de empresas ou hospitais;
- naturais: obtidos de repositório de dados como o repositório da UCI (Newman et al., 1998) e
- artificiais: gerados computacionalmente a partir da função verdadeira $f(\mathbf{x})$ a ser aprendida — Seção 2.2 na página 11.

A partir dessa pesquisa bibliográfica, foram selecionados 21 trabalhos que utilizam um total de 99 conjuntos de dados diferentes. Esses conjuntos de dados foram ordenados considerando o número de trabalhos nos quais foram utilizados. Após, foram considerados para seleção posterior somente os conjuntos de dados referenciados em pelo menos dois trabalhos. No final desse processo foram selecionados 11 conjuntos de dados supervisionados pouco desbalanceados com atributos numéricos. Esses conjuntos de dados pouco desbalanceados foram selecionados com o objetivo de não introduzir interferências associadas ao uso de um ou outro método para tratar esse problema (Batista et al., 2004).

Todos os 11 conjuntos de dados selecionados, brevemente descritos a seguir, constituem conjuntos de dados naturais obtidos do Repositório de Dados UCI (Newman et al., 1998).

Breast Cancer: o problema é predizer se uma amostra de tecido de mama obtida de uma paciente é maligna ou benigna baseada em dados histológicos;

Bupa: o problema é predizer se um paciente, do sexo masculino, possui ou não disfunção hepática tomando-se como base diversos exames sanguíneos e a quantidade de álcool consumida;

German: nesse conjunto de dados, parte do projeto europeu StatLog (Michie et al., 1994), o problema é classificar pessoas, descritas por atributos como propósito do empréstimo e histórico de crédito, como sendo boas ou más pagadoras, isto é, apresentando risco de crédito bom ou ruim. Esse conjunto de dados é disponibilizado em dois formatos: um contendo somente atributos simbólicos e outro contendo todos os atributos numéricos.

Hungarian: o problema consiste em prever se um paciente possui ou não doença cardíaca baseado em dados laboratoriais, clínicos e de eletrocardiograma.

Ionosphere: esse conjunto de dados descreve dados sobre radares. Bons resultados desses radares são considerados se mostram evidência de algum tipo de estrutura na ionosfera, caso contrário os resultados são considerados ruins.

Pima: o problema é prever se uma paciente, mulher de descendência indígena Pima com idade mínima de 21 anos, seria classificada como diabética, segundo o critério estabelecido pela Organização Mundial de Saúde, fornecidos dados clínicos e laboratoriais;

Satimage: esse conjunto de dados, também parte do projeto StatLog, consiste de valores multi-espectrais de pixels de vizinhança 3×3 em uma imagem de satélite e a classificação associada ao pixel central de cada vizinhança. O objetivo é prever essa classificação, dados os valores multi-espectrais.

Segment: esse conjunto de dados apresenta dados sobre segmentação de imagens. Os exemplos, regiões de tamanho 3×3 , foram gerados a partir de imagens de anúncios publicitários. Essas imagens foram segmentadas manualmente para criar a classificação para cada pixel.

Sonar: o objetivo é classificar sinais de sonares refletidos de um cilindro de metal ou de um cilindro aproximado de rocha.

Vehicle: o objetivo é classificar tipos de veículos, usando um conjunto de atributos extraídos a partir de suas silhuetas. O veículo pode ser visto de diversos ângulos. Esse conjunto de dados também faz parte do projeto StatLog.

Waveform: esse conjunto de dados está relacionado com a classificação de tipos de ondas.

A Tabela 6.1 mostra um resumo¹ das características desses 11 conjuntos de dados organizado do seguinte modo:

- # Exemplos: número de exemplos do conjunto de dados;
- # Atributos (num.,nom.): número total de atributos juntamente com o número de atributos numéricos (num.) e nominais (nom.);
- Classes e Classe %: valores e distribuição das classes;

¹Esses valores foram calculados utilizando recursos disponíveis na linguagem R (R Development Core Team, 2005) (<http://www.r-project.org/>) e na ferramenta *Data Characterization Tool* — DCT (Köpt, 2002) (<http://www.metal-kdd.org/>).

- Erro da CM: erro cometido no caso de novos exemplos serem classificados como sendo pertencentes à classe majoritária — CM e
- ?: existência ou não de valores desconhecidos.

Conjunto de Dados	# Exemplos	# Atributos (num.,nom.)	Classes	Classe %	Erro da CM	?
Breast Cancer	699	9 (9,0)	2	65,52%	34,48%	Sim
Bupa	345	6 (6,0)	4	34,48%	sobre 2	
			1	42,03%	42,03%	Não
			2	57,97%	sobre 2	
German	1000	24 (24,0)	1	70,00%	30,00%	Não
			2	30,00%	sobre 1	
Hungarian	294	13 (13,0)	0	63,95%	36,05%	Sim
			1	36,05%	sobre 0	
Ionosphere	351	34 (34,0)	0	64,10%	35,90%	Não
			1	35,90%	sobre 0	
Pima	769	8 (8,0)	0	65,02%	34,98%	Não
			1	34,98%	sobre 0	
Satimage	4435	36 (36,0)	1	24,20%	75,80%	Não
			2	10,80%	sobre 1	
			3	21,70%		
			4	09,40%		
			5	10,60%		
			7	23,40%		
Segment	2310	19 (19,0)	1	14,30%	85,70%	Não
			2	14,30%	sobre	
			3	14,30%	qualquer	
			4	14,30%	atributo	
			5	14,30%		
			6	14,30%		
			7	14,30%		
Sonar	208	60 (60,0)	0	46,60%	46,60%	Não
			1	53,40%	sobre 1	
Vehicle	846	18 (18,0)	1	25,10%	74,20%	Não
			2	25,70%	sobre 3	
			3	25,80%		
			4	23,50%		
Waveform	5000	21 (21,0)	0	33,10%	66,10%	Não
			1	32,90%	sobre 2	
			2	33,90%		

Tabela 6.1: Resumo dos conjuntos de dados

6.3 Algoritmos Utilizados

Os experimentos apresentados neste trabalho foram realizados utilizando quatro algoritmos freqüentemente utilizados na abordagem filtro para a seleção de atributos, descritos a seguir, além do algoritmo FDimBF proposto neste trabalho. É descrito também o algoritmo *C4.5* (Quinlan, 1993), o qual é um algoritmo para indução de árvores de decisão que realiza seleção embutida de atributos ao construir essas árvores de decisão.

ReliefF: O algoritmo Relief (Kira and Rendell, 1992) trabalha por meio da amostragem aleatória de exemplos do conjunto de dados e da localização do vizinho mais próximo da mesma classe e do vizinho mais próximo da classe oposta. Os valores dos atributos dos vizinhos mais próximos são comparados aos da classe amostrada e utilizados para atualizar os pesos de relevância de cada atributo em relação à classe. Esse processo é repetido um número m de vezes. A idéia do Relief é que atributos importantes devem diferenciar exemplos de classes diferentes e possuir valores similares para exemplos da mesma classe. A proposta original do algoritmo Relief, a qual permitia trabalhar com duas classes, foi posteriormente estendida no algoritmo ReliefF para lidar com ruído e conjuntos de dados contendo múltiplas classes (Kononenko, 1994). No ReliefF, a influência de ruído nos dados é amenizada por meio da distribuição da contribuição dos k vizinhos mais próximos da mesma classe do exemplo correntemente considerado e de k vizinhos mais próximos de cada uma das classes diferentes do exemplo amostrado, ao invés de considerar apenas um único vizinho mais próximo.

É interessante notar que quanto maior o valor de m , *i.e.*, o número de exemplos amostrados a partir do conjunto de dados, mais confiáveis são as estimativas fornecidas pelo algoritmo ReliefF, embora aumentar m signifique aumentar o tempo necessário para a execução desse algoritmo. ReliefF apresenta uma complexidade de tempo de $O(m \cdot N \cdot M)$, onde N é a quantidade de exemplos do conjunto de dados, M é o número de atributos desse conjunto de dados e m , como mencionado anteriormente, o número de vezes que o algoritmo procura por exemplos no conjunto de dados para calcular os pesos para os atributos (Robnik-Sikonja and Kononenko, 2003).

FCBF: O algoritmo FCBF (*Fast Correlation-Based Filter*) (Yu and Liu, 2004) realiza a seleção de atributos em duas etapas: primeiramente, os atributos são analisados para determinar o subconjunto de atributos relevantes em relação à classe, removendo os atributos irrelevantes; na segunda etapa, por meio da análise de redundância, são determinados e removidos os atributos redundantes a partir do subconjunto que contém apenas os atributos relevantes, produzindo o subconjunto final de atributos selecionados. Nesse algoritmo é utilizada a medida *Symmetrical Uncertainty* — SU (Press et al., 1992) como a medida de correlação para aproximar tanto a análise de relevância quanto a análise de redundância.

Assim, na primeira etapa, a medida SU entre cada atributo e a classe é calculada para todos os atributos, os quais são classificados de acordo com sua relevância em relação à classe. Apenas os atributos que possuem um valor SU maior que um limiar mínimo, que determina quão relevantes os atributos devem ser para serem considerados, são analisados na próxima etapa. Na segunda etapa, os atributos são avaliados na ordem em que foram classificados na etapa anterior, de acordo com a redundância de uns em relação aos outros, produzindo um subconjunto final

contendo apenas os atributos relevantes e não redundantes. É importante notar que no algoritmo FCBF os atributos numéricos são discretizados utilizando o algoritmo para discretização de atributos *Minimum Description Length* — MDL — proposto por Fayyad and Irani (1993).

O FCBF apresenta a vantagem, sobre as abordagens tradicionais para avaliação de subconjuntos de atributos, de que por meio da separação das tarefas de análise de relevância e de redundância, ele evita o alto custo da busca por subconjuntos. Esse algoritmo apresenta uma complexidade de tempo de $O(M^2)$ (Yu and Liu, 2004).

CBF: O algoritmo CBF (*Consistency-Based Filter* — CBF) (Liu and Setiono, 1996) é um algoritmo probabilístico que avalia os subconjuntos de atributos de acordo com sua consistência em relação à classe. Algoritmos que consideram essa medida procuram por combinações de atributos cujos valores particionem os dados em subconjuntos com alguma classe majoritária. Usualmente, a busca favorece subconjuntos pequenos de atributos que apresentam alta consistência com o atributo classe. Na proposta original desse algoritmo, apenas atributos discretos podem ser considerados para análise e os subconjuntos de atributos são determinados utilizando o algoritmo de Las Vegas (Brassard and Bratley, 1997), o qual realiza escolhas probabilísticas para auxiliá-lo na procura mais rápida por soluções corretas. A idéia desse algoritmo é, por um número máximo de tentativas *max_tries*, gerar subconjuntos de atributos e avaliá-los quanto ao seu tamanho e a sua inconsistência em relação à classe. Ao final, o subconjunto de atributos selecionado será aquele que, dentro do número máximo de tentativas, possuir o menor tamanho e a menor inconsistência. Esse algoritmo apresenta uma complexidade de tempo de $O(max_tries \cdot N)$ (Liu and Setiono, 1996). Porém, se a busca por subconjuntos de atributos for realizada utilizando *forward selection* ou *backward selection*, esse algoritmo apresenta uma complexidade de tempo de $O(N \cdot M^2)$.

CFS: O algoritmo CFS (*Correlation-based Feature Selection*) (Hall, 2000) classifica os subconjuntos de atributos de acordo com medidas de avaliação de separabilidade. Uma das medidas empregadas é a *Symmetrical Uncertainty* (Press et al., 1992). Esse algoritmo é composto, basicamente, por duas etapas: (1) avaliação da relação entre os atributos e da correlação entre atributos e classe e (2) busca por subconjuntos de atributos e avaliação desses subconjuntos. Desse modo, o CFS considera a habilidade preditiva individual de cada atributo e o grau de correlação entre esses atributos, incluindo a classe. Para a seleção de atributos utilizando o algoritmo CFS e a medida SU, é necessário que os atributos sejam discretizados.

Como esse algoritmo avalia subconjuntos de atributos, é necessário definir como a busca será conduzida e o critério de parada para a busca por esses subconjuntos. A implementação desse algoritmo permite realizar a busca de três maneiras: *forward selection*, *backward selection* e *best first*. Quanto ao critério de parada da busca,

nessa implementação foi estipulado que a busca termina após a geração de cinco subconjuntos de atributos que não mostrem melhores resultados quando comparados ao melhor subconjunto corrente. O algoritmo CFS apresenta uma complexidade de tempo de $O(N \cdot M^2)$ (Hall, 1999). Na implementação atual do algoritmo CFS utilizada neste trabalho, os atributos numéricos também são discretizados por CFS utilizando o algoritmo para discretização de atributos *Minimum Description Length* — MDL — proposto por Fayyad and Irani (1993).

C4.5: O algoritmo C4.5 (Quinlan, 1993) é um dos sucessores do algoritmo ID3 (Quinlan, 1986), o qual pertence a uma classe mais genérica de algoritmos de aprendizado de máquina denominado *Top Down Induction of Decision Trees* — TDIDT. Um nó em uma árvore de decisão representa um teste de um atributo em particular. De um modo simplificado, a construção de uma árvore de decisão procede do seguinte modo: usando o conjunto de treinamento, um atributo é escolhido para particioná-lo de acordo com o valor desse atributo. Sucessivamente, para cada subconjunto de dados outros atributos são selecionados, segundo algum critério, para particioná-lo. Esse processo continua enquanto cada subconjunto contém exemplos pertencentes a classes diferentes. Uma vez que um subconjunto uniforme, *i.e.*, todos ou quase todos os exemplos naquele subconjunto pertencem à mesma classe, um nó folha é criado e rotulado com o nome da respectiva classe. Diversas extensões foram adicionadas ao C4.5, tais como tratamento de atributos numéricos, valores faltantes ou desconhecidos e o uso do critério de razão de ganho ao invés do critério de ganho, usado na versão original do ID3 para selecionar os atributos que particionam os subconjuntos de exemplos. O propósito original do C4.5 não é a seleção de atributos, porém, como o algoritmo realiza seleção embutida de atributos ao construir a árvore de decisão, os atributos utilizados como nós de decisão podem ser interpretados como sendo relevantes em relação à classe e ordenados de acordo com o número de vezes que aparecem nas regras geradas a partir da árvore. O C4.5 apresenta complexidade de tempo de $O(N \cdot \log N \cdot M)$ (Witten and Frank, 2000).

FDimBF: O algoritmo FDimBF (*Fractal Dimension-Based Filter*) (Lee et al., 2005c,d,e; Lee and Monard, 2003), realiza, assim como o algoritmo FCBF, a seleção de atributos em duas etapas. Na primeira etapa, os atributos relevantes em relação à classe são selecionados. Na segunda etapa, somente os atributos não redundantes são selecionados a partir do subconjunto de atributos escolhidos na etapa anterior. O subconjunto final de atributos selecionados será composto por atributos relevantes em relação à classe e não redundantes entre si — Seção 5.4 na página 50.

Para a análise de relevância, neste trabalho são propostas duas versões desse algoritmo. FDimBF(1) considera uma medida de informação para a seleção de atributos relevantes em relação à classe. Essa medida é implementada por meio da utilização do algoritmo C4.5. Os atributos que pertencem às regras, construídas a partir

da árvore de decisão, são considerados como relevantes e classificados em ordem de relevância de acordo com o número de vezes que aparecem nessas regras. Já FDimBF(2) considera uma medida de distância para selecionar atributos relevantes em relação à classe, a qual é implementada por meio da utilização do algoritmo ReliefF, durante a primeira etapa. Ambos FDimBF(1) e FDimBF(2) consideram a medida de dimensão fractal para a seleção de atributos não redundantes na segunda etapa do processo de seleção de atributos — Seção 5.3 na página 48. O algoritmo FDimBF apresenta uma complexidade de tempo de $O(N \cdot M^3)$.

Apesar de FDimBF realizar busca por subconjuntos durante a segunda etapa na seleção de atributos, há dois fatores que contribuem para que sua complexidade seja comparável a diversos algoritmos encontrados na literatura:

1. durante a primeira etapa, menos custosa, é analisado um número maior de atributos; desse modo, durante a segunda etapa, mais custosa, há em geral uma diminuição do número de atributos a serem analisados e
2. durante a segunda etapa, é utilizado um algoritmo de baixa complexidade para o cálculo da DF e a seleção de atributos não redundantes.

É importante notar que as duas versões do algoritmo proposto neste trabalho tratam diretamente atributos numéricos, sem a necessidade que eles sejam discretizados, como ocorre com os outros algoritmos utilizados neste trabalho.

A Tabela 6.2 resume as principais características desses algoritmos e do algoritmo FDimBF proposto neste trabalho.

	C4.5	ReliefF	CFS	FCBF	CBF	FDimBF(1)	FDimBF(2)
Avaliação Individual	✓	✓		✓		✓	✓
Avaliação de Subconjuntos			✓	✓	✓	✓	✓
Medida de Informação	✓		✓	✓		✓	
Medida de Distância		✓					✓
Medida de Dependência						✓	✓
Medida de Consistência					✓		

Tabela 6.2: Características dos algoritmos de SA

6.4 Configuração dos Experimentos

Os experimentos realizados foram organizados em quatro etapas, as quais são ilustradas na Figura 6.1.

Figura 6.1: Configuração dos experimentos

Etapa 1: nessa etapa foram realizadas a limpeza e a preparação dos dados. A tarefa de limpeza dos dados consistiu na remoção de valores desconhecidos da seguinte maneira: para valores desconhecidos concentrados em alguns poucos exemplos, esses exemplos foram removidos, enquanto que para valores desconhecidos concentrados em um atributo, a coluna correspondente foi removida do conjunto de dados. A principal razão para a remoção de valores desconhecidos do conjunto de dados é que alguns dos algoritmos utilizados nesses experimentos tratam valores faltantes de modo especial (Batista and Monard, 2003a), enquanto outros algoritmos não tratam esse tipo de informação. Assim, com o intuito de não introduzir interferências associadas ao uso de um ou outro método para tratar esse problema, foi decidida a remoção de valores desconhecidos do conjunto de dados. Ao final dessa etapa, os dados foram transformados para a sintaxe requerida por cada um dos algoritmos e ferramentas utilizados neste trabalho.

Etapa 2: nessa etapa foi realizada a seleção de atributos utilizando os algoritmos descritos na Seção 6.3 na página 60 e o algoritmo por nós proposto — Seção 5.4 na página 50. Todos esses algoritmos, à exceção do algoritmo proposto neste trabalho, estão implementados na ferramenta Weka (Witten and Frank, 2000) e foram executados considerando seus parâmetros configurados com os valores padrão e a busca *forward* para os algoritmos CFS, FCBF e CBF. Deve ser observado que os algoritmos marcados com * na Figura 6.1 são aqueles que tratam tanto o problema da relevância de atributos, em relação ao atributo classe, quanto o problema da redundância de atributos.

Etapa 3: nessa etapa foram induzidos os modelos (classificadores) usando todos os atributos remanescentes da Etapa 1 e apenas os atributos selecionados na etapa anterior. Esses modelos foram construídos utilizando o algoritmo C4.5 (Quinlan, 1993), executado usando parâmetros com valores padrão. Esse algoritmo foi escolhido por

permitir a construção de modelos simbólicos a partir dos dados, os quais são importantes quando o objetivo é a extração de conhecimento de bases de dados.

Etapa 4: nessa última etapa, os resultados foram avaliados por meio da estimativa da média do erro de cada um dos modelos construídos usando validação cruzada com 10 partições (*10 fold cross-validation*). Esse modo de avaliação foi escolhido pois, para conjuntos de dados naturais ou reais, o conhecimento prévio sobre que atributos são importantes, em geral, não está disponível. Desse modo, a precisão preditiva é comumente utilizada como uma medida indireta para avaliar a qualidade dos atributos selecionados.

Dos 11 conjuntos de dados considerados neste trabalho, somente dois foram submetidos à limpeza de dados: Breast Cancer e Hungarian. O primeiro conjunto de dados possuía originalmente 699 exemplos e nove atributos. Nesse conjunto de dados os valores faltantes estavam concentrados em alguns poucos exemplos, assim, após a realização dessa tarefa, passou a ser representado por 683 exemplos e o mesmo número de atributos. Já o conjunto de dados Hungarian, o qual continha 294 exemplos descritos por 13 atributos, possuía valores faltantes concentrados tanto em exemplos quanto em atributos. Desse modo, após a limpeza de dados, o novo conjunto de dados Hungarian passou a ser descrito por 261 exemplos e 10 atributos.

Para auxiliar na tarefa de construção dos modelos e avaliação desses modelos por meio de validação cruzada com 10 partições, foi utilizado o ambiente para gerenciamento de experimentos **SNIFFER**, o qual faz parte do projeto **DISCOVER** (Prati, 2003; Batista, 2003; Batista and Monard, 2003b, 2005, 2002). Esse projeto constitui um ambiente computacional, em desenvolvimento no Laboratório de Inteligência Computacional — LABIC — que tem como objetivo auxiliar os usuários nas três fases do processo de descoberta de conhecimento — Figura 1.1 na página 2.

O projeto **DISCOVER** oferece vantagens em relação a outros sistemas com objetivos semelhantes, pois permite a visão unificada que os formatos baseados em padrões proporcionam ao pesquisador (desenvolvedor) de novos componentes. Os padrões de representação foram definidos por área, tendo sido proposta por Prati et al. (2001a) uma sintaxe padrão para representação de conhecimento de diversos indutores simbólicos denominada $\mathcal{PBM}$ (Prati et al., 2002, 2001b). Para a representação de dados foi proposta uma sintaxe padrão (Batista, 2001), denominada *Discover Dataset Syntax* — DSX —, a qual permite a utilização da biblioteca de classes *Discover Object Library* — DOL — (Batista and Monard, 2005), para, entre outras funcionalidades, converter os arquivos de dados para as sintaxes utilizadas por diversos sistemas de aprendizado simbólico, tais como $\mathcal{C4.5}$, $\mathcal{C4.5rules}$ e $\mathcal{CN2}$.

6.5 Resultados e Discussão

Para cada conjunto de dados, foi realizada a seleção de atributos usando as duas versões do algoritmo proposto neste trabalho, *i.e.*, FDimBF(1) e FDimBF(2), e os algoritmos C4.5, ReliefF, CFS, CBF e FCBF, totalizando 77 experimentos. Como mencionado anteriormente, foram gerados modelos considerando os atributos selecionados pelos algoritmos citados e também considerando os conjuntos de dados descritos pelos conjuntos originais de atributos (sem SA), totalizando 88 modelos construídos. Os erros dos classificadores foram estimados por meio de validação cruzada com 10 partições e comparados usando o teste estatístico não paramétrico Kruskal-Wallis para grupos não pareados, com nível de significância de 95%, seguido do pós-teste de Dunn² (Motulsky, 1995).

Os resultados, os quais estão descritos detalhadamente em Lee et al. (2005b), apresentados a seguir, estão organizados da seguinte maneira:

1. dimensão fractal e comportamento dos conjuntos de dados quanto à característica fractal;
2. subconjuntos de atributos selecionados pelos algoritmos considerados neste trabalho, bem como a redução do número de atributos;
3. distribuição aproximada para cada um dos atributos de cada conjunto de dados;
4. modelo de performance dos algoritmos em relação à precisão e a quantidade de atributos selecionados;
5. análise da significância estatística dos resultados e
6. características dos conjuntos de dados associadas à utilização da DF como uma medida adequada.

6.5.1 Dimensão Fractal e Comportamento Fractal dos Conjuntos de Dados

Como mencionado anteriormente, a medida de Dimensão Fractal é utilizada neste trabalho para a determinação de quantos atributos são não redundantes a partir do subconjunto de atributos relevantes de um conjunto de dados. De uma maneira simplificada, a classificação de quais atributos são importantes, segundo a DF, é realizada por meio da determinação dos atributos que, quando retirados do conjunto de dados, causam uma mudança significativa no valor da DF recalculada. Desse modo, os atributos são classificados de acordo com sua importância para o cálculo da Dimensão Fractal.

²Testes estatísticos realizados utilizando GraphPad InStat versão 3.06 para Windows, GraphPad Software, <http://www.graphpad.com>.

Para a análise de resultados associados à DF, é importante observar os seguintes três aspectos:

1. formato da curva de comportamento do conjunto de dados;
2. número de pontos utilizados para construir essa curva e
3. ajuste da reta, que determina a DF, em relação à curva.

Dois exemplos, construídos com o auxílio da ferramenta MDE, são ilustrados nas Figuras 6.2 e 6.3, correspondentes aos conjuntos de dados Hungarian e Waveform, respectivamente. Nessas figuras é possível observar: (a) curva de comportamento do conjunto de dados, que representa o gráfico em escala logarítmica da soma do número de pontos existentes em uma célula de lado r pelo tamanho da célula r , e (b) reta que aproxima o cálculo da Dimensão Fractal desse conjunto de dados.

Figura 6.2: Gráfico gerado utilizando o método *Box Count Plot* — Hungarian

A curva (a) da Figura 6.2 para o conjunto de dados Hungarian ilustra o comportamento de um conjunto de dados que apresenta característica de um fractal estatisticamente auto-similar, pois é possível identificar um trecho “bastante” linear na curva de comportamento desse conjunto de dados. Ao fazer essa identificação, é importante também observar o número de pontos utilizados para construir essa curva, sendo que o número mínimo considerado razoável é de três pontos (Sousa, 2005). Em relação ao ajuste da reta ao trecho, aproximadamente linear da curva, é importante também observar se esse ajuste foi realizado sobre um número aceitável de pontos. No exemplo do conjunto de dados Hungarian,

Figura 6.3: Gráfico gerado utilizando o método *Box Count Plot* — Waveform

a curva foi construída tomando em consideração sete pontos e a reta foi ajustada sobre cinco pontos. Por outro lado, a curva (a) da Figura 6.3 para o conjunto de dados Waveform representa um caso em que o conjunto de dados parece não exibir característica fractal, pois apresenta um trecho em que praticamente não há variação na quantidade de pontos dentro de células de diferentes tamanhos, apresentando somente uma pequena variação no trecho final da curva. Esse comportamento do conjunto de dados também é refletido no ajuste da reta da DF o qual, para o conjunto de dados Waveform, foi realizado sobre apenas dois pontos.

A Tabela 6.3 apresenta as informações associadas ao cálculo da DF para cada um dos 11 conjuntos de dados utilizados nos experimentos, organizadas do seguinte modo:

- Algoritmo: indica a abordagem do algoritmo proposto, *i.e.*, se a seleção de atributos relevantes em relação à classe foi realizada previamente aplicando a medida de ganho de informação — FDimBF(1) — ou a medida de distância — FDimBF(2);
- # Atrib. Orig.: número de atributos após a remoção de valores desconhecidos do conjunto de dados³;
- # Ex.: número de exemplos após a remoção de valores desconhecidos do conjunto de dados⁴;

³Apenas o conjunto de dados Hungarian apresentou valores faltantes concentrados em atributos.

⁴Ambos os conjuntos de dados Breast Cancer e Hungarian apresentaram valores faltantes concentrados em exemplos.

- # Atrib. Relev.: número de atributos relevantes selecionados a partir da aplicação das medidas de ganho de informação — FDimBF(1) — ou de distância — FDimBF(2);
- DF: dimensão fractal do conjunto de dados, considerando somente os atributos relevantes em relação à classe;
- # Atrib. Selec.: número de atributos selecionados utilizando a DF como medida de redundância;
- # Pontos (Curva): número de pontos utilizados para a construção da curva de comportamento do conjunto de dados e
- # Pontos (Reta): número de pontos utilizados para ajustar a reta sobre a curva de comportamento do conjunto de dados.

Conjunto de Dados	Algoritmo	# Atrib. Orig.	# Ex.	# Atrib. Relev.	DF	# Atrib. Selec.	# Pontos (Curva)	# Pontos (Reta)
Breast Cancer	FDimBF(1)	9	683	7	2,20	3	4	3
	FDimBF(2)			9	2,30	3	4	3
Bupa	FDimBF(1)	6	345	6	3,79	4	6	4
	FDimBF(2)			5	3,42	4	7	4
German	FDimBF(1)	24	1000	24	11,46	12	6	2
	FDimBF(2)			24	11,39	12	6	2
Hungarian	FDimBF(1)	10	261	9	2,35	3	7	5
	FDimBF(2)			10	3,60	4	5	4
Ionosphere	FDimBF(1)	34	351	15	2,79	3	5	5
	FDimBF(2)			33	3,23	4	4	3
Pima	FDimBF(1)	8	769	7	2,75	3	5	3
	FDimBF(2)			8	3,14	4	5	3
Satimage	FDimBF(1)	36	4435	36	5,09	6	4	3
	FDimBF(2)			36	5,09	6	4	3
Segment	FDimBF(1)	19	2310	16	3,07	4	8	4
	FDimBF(2)			18	3,07	4	8	4
Sonar	FDimBF(1)	60	208	15	4,95	5	3	3
	FDimBF(2)			60	9,54	10	2	2
Vehicle	FDimBF(1)	18	846	18	5,83	6	6	4
	FDimBF(2)			18	5,83	6	6	4
Waveform	FDimBF(1)	21	5000	21	2,02	3	3	2
	FDimBF(2)			21	2,02	3	3	2

Tabela 6.3: Informações associadas à dimensão fractal dos conjuntos de dados

Dos 11 conjuntos de dados considerados neste trabalho, apenas para Sonar, quando utilizado com FDimBF(2), *i.e.*, ReliefF, não foi possível construir uma curva de comportamento do conjunto de dados com um mínimo de três pontos. Quanto ao número de pontos usados para o ajuste da reta para o cálculo da DF, em dois casos, conjuntos de dados German e Waveform, a reta foi ajustada com menos de três pontos para FDimBF(1) e em três casos, conjuntos de dados German, Sonar e Waveform, para FDimBF(2).

O resultado da análise dos gráficos de comportamento dos conjuntos de dados quanto a sua característica fractal é apresentado na Tabela 6.4, onde:

- # Pontos (Curva, Reta): mostra, respectivamente, o número de pontos utilizado pelo MDE para interpolar a curva e a reta de ajuste correspondente e
- Caract. Fractal: classifica a característica fractal do conjunto de dados como Muito Bom, Bom, Mediano e Ruim considerando o formato da curva de comportamento do conjunto de dados e o número de pontos usado para construir essa curva.

Algoritmo	Conjunto de Dados	# Pontos (Curva, Reta)	Caract. Fractal	Conjunto de Dados	# Pontos (Curva, Reta)	Caract. Fractal
FDimBF(1) FDimBF(2)	Breast Cancer	(4, 3) (4, 3)	Bom Bom	Satimage	(4, 3) (4, 3)	Mediano Mediano
FDimBF(1) FDimBF(2)	Bupa	(6, 4) (7, 4)	Muito Bom Muito Bom	Segment	(8, 4) (8, 4)	Bom Bom
FDimBF(1) FDimBF(2)	German	(6, 2) (6, 2)	Ruim Ruim	Sonar	(3, 3) (2, 2)	Muito Bom Ruim
FDimBF(1) FDimBF(2)	Hungarian	(7, 5) (5, 4)	Muito Bom Muito Bom	Vehicle	(6, 4) (6, 4)	Bom Bom
FDimBF(1) FDimBF(2)	Ionosphere	(5, 5) (4, 3)	Muito Bom Muito Bom	Waveform	(3, 2) (3, 2)	Ruim Ruim
FDimBF(1) FDimBF(2)	Pima	(5, 3) (5, 3)	Bom Bom			

Tabela 6.4: Resultado da análise dos gráficos de comportamento dos conjuntos de dados quanto à característica fractal

Uma análise dos gráficos de comportamento dos conjuntos de dados mostrou que em relação à característica fractal houve quatro e três Muito Bom, cinco e cinco Bom, um e um Mediano e dois e três Ruim para FDimBF(1) e FDimBF(2), respectivamente.

6.5.2 Subconjuntos de Atributos Selecionados

Dois principais fatores, além das características próprias do conjunto de dados, podem influenciar no subconjunto de atributos selecionado por algoritmos de seleção de atributos e estão relacionados com:

- a avaliação dos atributos, *i.e.*, se os atributos são avaliados individualmente ou considerando um subconjunto de atributos e
- a medida utilizada para determinar a importância dos atributos.

Como mencionado anteriormente, neste trabalho foram considerados quatro algoritmos frequentemente citados na literatura para a seleção de atributos e o algoritmo $\mathcal{C}4.5$, além do algoritmo FDimBF proposto — Tabela 6.2 na página 64. Um desses algoritmos — ReliefF — realiza a seleção utilizando o critério de avaliação individual de atributos e os outros dois — CFS e CBF — o critério de avaliação de subconjuntos de atributos. O algoritmo FCBF realiza a avaliação individual de atributos na primeira etapa de análise de relevância, enquanto na segunda etapa, análise de redundância, avalia subconjuntos de atributos. O algoritmo $\mathcal{C}4.5$, embora seu propósito original não seja seleção de atributos, realiza SA embutida ao construir a árvore de decisão. Desse modo, os atributos utilizados como nós de decisão podem ser interpretados como sendo relevantes em relação à classe e ordenados de acordo com o número de vezes que aparecem nas regras geradas a partir da árvore. O algoritmo FDimBF realiza, assim como FCBF, a primeira parte da seleção de atributos por meio de avaliação individual de atributos e a segunda parte por meio de avaliação de subconjuntos de atributos. Em relação à medida utilizada para determinar a importância dos atributos, esses algoritmos usam medidas de distância (ReliefF e FDimBF(2)), dependência (FDimBF(1) e FDimBF(2)), informação ($\mathcal{C}4.5$, CFS, FCBF e FDimBF(1)) e consistência (CFS).

A Tabela 6.5 apresenta um resumo da quantidade de atributos selecionados por cada um dos algoritmos e suas respectivas percentagens. Também é apresentada essa informação para o algoritmo $\mathcal{C}4.5$, o qual é utilizado na etapa de seleção de atributos relevantes em relação à classe no algoritmo FDimBF(1). A organização das informações dessa tabela é descrita a seguir.

Na primeira coluna é apresentado o conjunto de dados ao qual referem-se as informações. Na segunda coluna é indicada a quantidade original, *i.e.*, depois da remoção de valores desconhecidos, de atributos de cada conjunto de dados. Para cada um deles, na primeira linha são descritos o número de atributos referentes ao subconjunto selecionado por cada um dos algoritmos e na segunda linha é apresentada a respectiva percentagem. As últimas duas linhas mostram a média de atributos selecionados por cada algoritmo (Média # Atrib.) e a respectiva percentagem (Média % Atrib.).

Considerando somente o tamanho dos subconjuntos de atributos selecionados por cada algoritmo, o algoritmo ReliefF foi o que selecionou os maiores subconjuntos de atributos, variando de um mínimo de 83,33% do total de atributos para o conjunto de dados Bupa até o máximo de 100,00% (todos os atributos) para oito do total de 11 conjuntos de dados. O algoritmo CFS selecionou o menor número de atributos, em relação ao número de atributos selecionados pelos outros algoritmos, em um conjunto de dados: German (8,33%). Para outros três conjuntos de dados, o algoritmo CFS selecionou, juntamente

com outros algoritmos, o menor número de atributos: Bupa (16,67%) em conjunto com FCBF e CBF e Hungarian (30,00%) e Pima (37,50%) em conjunto com FDimBF(1). O algoritmo FDimBF, nas duas versões, foi o que mais freqüentemente selecionou os menores subconjuntos de atributos: cinco vezes FDimBF(1) em conjunto com FDimBF(2), duas vezes FDimBF(1) sozinho para os conjuntos de dados Ionosphere (8,82%) e Sonar (8,33%) e duas vezes FDimBF(1) em conjunto com o algoritmo CFS, como mencionado anteriormente.

	Sem SA	C4.5	ReliefF	CFS	FCBF	CBF	FDimBF(1)	FDimBF(2)
Breast Cancer	9	7 77,78	9 100,00	9 100,00	9 100,00	7 77,78	3 33,33	3 33,33
Bupa	6	6 100,00	5 83,33	1 16,67	1 16,67	1 16,67	4 66,67	4 83,33
German	24	24 100,00	24 100,00	2 8,33	15 62,50	15 62,50	12 50,00	12 50,00
Hungarian	10	9 90,00	10 100,00	3 30,00	6 60,00	5 50,00	3 30,00	4 40,00
Ionosphere	34	15 44,12	33 97,06	14 41,18	33 97,06	7 20,59	3 8,82	4 11,76
Pima	8	7 87,50	8 100,00	3 37,50	8 100,00	8 100,00	3 37,50	4 50,00
Satimage	36	36 100,00	36 100,00	23 63,89	36 100,00	12 33,33	6 16,67	6 16,67
Segment	19	16 84,21	18 94,74	5 26,32	18 94,74	9 47,37	4 21,05	4 21,05
Sonar	60	15 25,00	60 100,00	19 31,67	21 35,00	14 23,33	5 8,33	10 16,67
Vehicle	18	18 100,00	18 100,00	11 61,11	18 100,00	18 100,00	6 33,33	6 33,33
Waveform	21	21 100,00	21 100,00	15 71,43	19 90,48	12 57,14	3 14,29	3 14,29
Média # Atrib.		16	22	10	17	10	5	5
Média % Atrib.		82,60	97,74	44,37	77,86	53,52	29,09	32,16

Tabela 6.5: Resumo da quantidade de atributos selecionados por cada um dos algoritmos e suas respectivas percentagens

As Figuras 6.4a a 6.4k mostram graficamente, para cada conjunto de dados, o número de atributos selecionados e a respectiva percentagem *versus* o algoritmo de SA. Entre parênteses é apresentada a média da percentagem de atributos selecionados pelos algoritmos de SA para cada conjunto de dados.

É interessante observar que dentre os 11 conjuntos de dados, em cinco deles, Breast Cancer, Satimage, Segment, Vehicle e Waveform, as abordagens FDimBF(1) e FDimBF(2) selecionaram o mesmo subconjunto final de atributos. Para dois conjuntos de dados, Ionosphere e Sonar, as duas abordagens do algoritmo FDimBF selecionaram subconjuntos de atributos totalmente diferentes. No restante dos conjuntos de dados, os subconjuntos de atributos selecionados incluem alguns dos atributos escolhidos por FDimBF(1) e FDimBF(2) em comum.

Figura 6.4: Número de atributos selecionados e a respectiva percentagem *versus* o algoritmo de SA

6.5.3 Formatos Aproximados de Distribuição dos Valores dos Atributos em Relação aos Atributos Selecionados pelo Algoritmo FDimBF

Com o intuito de verificar se a distribuição dos valores dos atributos originais exerce alguma influência sobre os subconjuntos de atributos selecionados por FDimBF(1) e (2), esses três conjuntos de atributos foram analisados do ponto de vista do formato aproximado da distribuição da maioria dos atributos presentes neles e classificados de acordo com seis tipos, T_1 , T_2 , T_3 , T_4 , T_5 e T_6 , como é mostrado na Figura 6.5. Além disso, uma outra relação de interesse é saber se os algoritmos FDimBF selecionam, preferencialmente, atributos cujos valores obedecem a algum tipo de distribuição.

Figura 6.5: Tipos de formatos aproximados das distribuições dos valores dos atributos

A Tabela 6.6 apresenta, para cada conjunto de dados e abordagem de FDimBF, em que tipo de formato aproximado de distribuição os atributos podem ser classificados. Dos 11 conjuntos de dados considerados neste trabalho, sete deles possuem a maioria dos atributos com formato aproximado de distribuição do tipo T_3 , três do tipo T_1 e um do tipo T_2 .

Em relação aos atributos selecionados pelos algoritmos FDimBF, é interessante notar que do total de 11 conjuntos de dados considerados, em 10 deles, a maioria dos atributos apresentaram formatos aproximados de distribuição dos valores semelhantes para

Conjunto de Dados	Todos os Atributos	Atributos Seleccionados por (FDimBF(1), FDimBF(2))
Breast Cancer	T_1	(T_1, T_1)
Bupa	T_3	(T_3, T_3)
German	T_1	(T_4, T_4)
Hungarian	T_3	(T_5, T_3)
Ionosphere	T_2	(T_3, T_3)
Pima	T_3	(T_3, T_3)
Satimage	T_3	(T_3, T_3)
Segment	T_1	(T_5, T_5)
Sonar	T_3	(T_3, T_3)
Vehicle	T_3	(T_5, T_5)
Waveform	T_3	(T_3, T_3)

Tabela 6.6: Formatos da distribuição aproximada dos valores dos atributos

FDimBF(1) e FDimBF(2). Desses 10 conjuntos de dados, em cinco deles isso se deve aos subconjuntos de atributos seleccionados pelas duas abordagens serem os mesmos, como mencionado anteriormente. Nos outros seis conjuntos, quatro deles, Bupa, German, Hungarian e Pima, apresentam intersecção entre os subconjuntos e dois, Ionosphere e Sonar, apresentam subconjuntos de atributos totalmente diferentes.

Essa classificação dos subconjuntos de atributos quanto ao formato da distribuição dos valores da maioria de seus atributos mostra que mais de 50% dos subconjuntos apresenta uma distribuição do tipo T_3 de seus valores.

6.5.4 Modelo de Performance dos Algoritmos em Relação à Precisão e a Quantidade de Atributos Seleccionados

Os resultados dos experimentos foram também avaliados quanto à relação entre a quantidade de atributos seleccionados e o erro dos modelos construídos — Tabela 6.7.

Uma das medidas mais freqüentemente utilizadas para avaliar algoritmos de AM é a precisão do modelo induzido. Entretanto, no caso de redução de atributos é importante não somente avaliar a precisão, mas também encontrar métodos que permitam combinar a precisão do modelo induzido e a redução da quantidade de atributos. Dessa maneira, a seguinte questão deve ser levada em conta:

Como combinar ambos critérios tal que um deles não domine o outro?

Em outras palavras:

Quanto o modelo pode degradar considerando a redução da quantidade de atributos necessários para sua indução?

Ambas medidas, precisão e redução da quantidade de atributos estão relacionadas, e há inúmeras maneiras de definir essa relação. Neste trabalho propomos o seguinte

	Sem SA	C4.5	ReliefF	CFS
BreastCancer	5,27 $\pm$ 1,03	4,83 $\pm$ 0,54	5,27 $\pm$ 1,03	5,27 $\pm$ 1,03
Bupa	29,57 $\pm$ 2,38	29,57 $\pm$ 2,38	33,63 $\pm$ 3,11	36,77 $\pm$ 2,72
German	26,60 $\pm$ 1,37	26,60 $\pm$ 1,37	26,60 $\pm$ 1,37	28,00 $\pm$ 0,68
Hungarian	23,40 $\pm$ 2,05	21,87 $\pm$ 2,01	23,40 $\pm$ 2,05	21,48 $\pm$ 2,89
Ionosphere	9,97 $\pm$ 1,96	11,40 $\pm$ 0,85	10,55 $\pm$ 2,00	10,27 $\pm$ 0,98
Pima	24,32 $\pm$ 1,28	25,10 $\pm$ 1,50	24,32 $\pm$ 1,28	25,35 $\pm$ 1,14
Satimage	14,05 $\pm$ 0,43	14,05 $\pm$ 0,43	14,05 $\pm$ 0,43	13,66 $\pm$ 0,49
Segment	3,03 $\pm$ 0,35	3,46 $\pm$ 0,54	3,29 $\pm$ 0,30	3,59 $\pm$ 0,25
Sonar	24,05 $\pm$ 3,70	24,95 $\pm$ 2,97	24,05 $\pm$ 3,70	23,95 $\pm$ 2,64
Vehicle	26,95 $\pm$ 1,16	26,95 $\pm$ 1,16	26,95 $\pm$ 1,16	31,68 $\pm$ 1,50
Waveform	23,80 $\pm$ 0,51	23,80 $\pm$ 0,51	23,80 $\pm$ 0,51	22,38 $\pm$ 0,41
	FCBF	CBF	FDimBF(1)	FDimBF(2)
BreastCancer	5,27 $\pm$ 1,03	4,98 $\pm$ 0,62	4,40 $\pm$ 0,54	4,40 $\pm$ 0,54
Bupa	36,77 $\pm$ 2,72	36,77 $\pm$ 2,72	42,01 $\pm$ 1,37	33,03 $\pm$ 2,17
German	26,40 $\pm$ 1,90	26,40 $\pm$ 1,90	25,50 $\pm$ 1,49	26,30 $\pm$ 0,79
Hungarian	23,38 $\pm$ 2,78	23,40 $\pm$ 1,97	24,53 $\pm$ 2,45	22,21 $\pm$ 2,18
Ionosphere	10,55 $\pm$ 2,00	11,40 $\pm$ 1,86	19,38 $\pm$ 2,48	19,36 $\pm$ 1,72
Pima	24,32 $\pm$ 1,28	24,32 $\pm$ 1,28	25,50 $\pm$ 1,49	34,89 $\pm$ 3,74
Satimage	14,05 $\pm$ 0,43	13,55 $\pm$ 0,53	16,80 $\pm$ 0,56	16,80 $\pm$ 0,56
Segment	3,29 $\pm$ 0,30	3,51 $\pm$ 0,30	6,15 $\pm$ 0,35	6,15 $\pm$ 0,35
Sonar	26,38 $\pm$ 1,71	25,52 $\pm$ 4,27	38,02 $\pm$ 2,37	34,55 $\pm$ 3,42
Vehicle	26,95 $\pm$ 1,16	26,95 $\pm$ 1,16	33,92 $\pm$ 1,00	33,92 $\pm$ 1,00
Waveform	23,24 $\pm$ 0,69	24,86 $\pm$ 0,88	35,16 $\pm$ 0,78	35,16 $\pm$ 0,78

Tabela 6.7: Média de erro e erro padrão para cada conjunto de dados e cada algoritmo considerados

modelo a fim de avaliar essas medidas conjuntamente. A relação entre erro e redução da quantidade de atributos é representada graficamente com o objetivo de auxiliar na avaliação da performance dos algoritmos considerando ambas as medidas, como mostrado na Figura 6.6a, sendo:

- Eixo X: representa a média do erro, obtido usando validação cruzada com 10 partições e
- Eixo Y: representa a percentagem de atributos selecionados em relação ao total de atributos (apresentado entre parênteses).

Nesse gráfico, para cada conjunto de dados, os algoritmos de SA são classificados quanto ao seu posicionamento em relação à percentagem de atributos selecionados e a média do erro e o erro padrão do modelo construído considerando os atributos selecionados por esses algoritmos, dentro de cinco regiões definidas a seguir. Primeiramente, duas grandes áreas são delimitadas pela reta que liga o ponto 100% (número total de atributos do conjunto de dados) no eixo X ao ponto ECM no eixo Y, sendo ECM igual ao Erro da Classe Majoritária caso seja menor que 50%, ou igual a 50% caso contrário. Essa reta é

Figura 6.6: Relação entre percentagem de atributos selecionados, média do erro e erro padrão dos modelos construídos: (a) Modelo geral e (b) Conjunto de dados Pima

definida pela Equação 6.1:

$$\begin{cases} y = \frac{100 (ECM - x)}{ECM - ESA} \\ ECM = \text{if } ECM \geq 50 \text{ then } 50 \text{ else } ECM \end{cases} \quad (6.1)$$

Nesse modelo de avaliação, considerou-se que essa reta representa uma proporção mínima entre o que se espera em termos da relação entre a percentagem de atributos selecionados e a média do erro do modelo construído considerando os atributos selecionados. Essa reta divide o plano em dois semi-planos. Qualquer modelo construído com os atributos selecionados por um algoritmo de SA que esteja localizado na região acima dessa reta pode ser considerado de performance Ruim (▼) — Equação 6.2.

$$Ruim(\blacktriangledown) \left\{ y \geq \frac{100 (ECM - x)}{ECM - ESA} \right. \quad (6.2)$$

Abaixo dessa reta e delimitadas pelos eixos X e Y , outras três regiões foram definidas:

- Muito Bom (▲▲): retângulo que delimita a região que corresponde a 50% ou menos de atributos selecionados e até 50% da diferença entre ECM e o erro do modelo construído considerando todos os atributos — ESA —, *i.e.*, sem a realização de

seleção de atributos — Equação 6.3;

$$Muito\ Bom(\blacktriangle\blacktriangle) \left\{ \begin{array}{l} 0 \leq y < 50 \\ ESA < x < \frac{ECM - ESA}{2} \end{array} \right. \quad (6.3)$$

- Bom ($\blacktriangle$): região acima da região Muito Bom — Equação 6.4 e

$$Bom(\blacktriangle) \left\{ \begin{array}{l} y < \frac{100(ECM - x)}{ECM - ESA} \\ 50 \leq y < 100 \\ ESA < x < \frac{ECM - ESA}{2} \end{array} \right. \quad (6.4)$$

- Regular ($\diamond$): região ao lado direito da região Muito Bom — Equação 6.5.

$$Regular(\diamond) \left\{ \begin{array}{l} y < \frac{100(ECM - x)}{ECM - ESA} \\ 0 \leq y < 50 \\ \frac{ECM - ESA}{2} \leq x < ECM \end{array} \right. \quad (6.5)$$

Uma quinta região, denominada Excelente ($\blacktriangle\blacktriangle\blacktriangle$), foi definida como sendo a área à esquerda do eixo X — Equação 6.6.

$$Excelente(\blacktriangle\blacktriangle\blacktriangle) \left\{ \begin{array}{l} 0 < y < 100 \\ x \leq ESA \end{array} \right. \quad (6.6)$$

Assim, qualquer algoritmo que permita a seleção de subconjuntos de atributos que melhorem a precisão do modelo construído é considerado de performance excelente.

Na Figura 6.6b é apresentado um exemplo do modelo de avaliação por nós proposto para o conjunto de dados Pima. Nessa figura é possível identificar a média do erro e o erro padrão considerando o conjunto original de atributos, denominado T, e as posições dos algoritmos, os quais são representados pelas siglas na Tabela 6.8, dentro das cinco regiões descritas anteriormente. Para esse conjunto de dados, o modelo construído utilizando o subconjunto de atributos selecionado por $\mathcal{C}4.5$ foi considerado bom. Já os algoritmos ReliefF, FCBF e CBF selecionaram todos os atributos do conjunto de dados como sendo importantes. O modelo construído utilizando o subconjunto de atributos selecionado por FDimBF(2) foi considerado ruim, pois encontra-se na região acima da reta definida pelos

pontos 100% de atributos selecionados e ECM. Já para a seleção de atributos utilizando os algoritmos CFS e FDimBF(1), os modelos construídos foram considerados muito bons.

Sigla	Algoritmo
A	C4.5
B	ReliefF
C	CFS
D	FCBF
E	CBF
F	FDimBF(1)
G	FDimBF(2)

Tabela 6.8: Algoritmos presentes nos gráficos

A Tabela 6.9 mostra um resumo da classificação dos algoritmos de SA para cada conjunto de dados quanto ao posicionamento dentro das regiões definidas — Figura 6.6a. Nos casos em que o conjunto de atributos selecionados foi igual ao conjunto original de atributos do conjunto de dados, o algoritmo foi classificado como Todos os Atributos Selecionados (—). Para cada conjunto de dados é ainda apresentada, na última coluna — CRes —, uma classificação do resultado da aplicação dos algoritmos de SA indicada por $\uparrow$ (número de classificações Excelente, Muito Bom e Bom maior ou igual a cinco), $\downarrow$ (maioria das classificações Regular e Ruim) e $\sim$ (número de classificações Todos os Atributos Selecionados representa em torno de 50% dos casos). Nas últimas linhas dessa tabela é mostrado um resumo da quantidade de vezes em que o respectivo algoritmo foi classificado como tendo apresentado desempenho Excelente, Muito Bom, Bom, Regular, Ruim e Todos os Atributos Selecionados.

Algoritmo	C4.5	ReliefF	CFS	FCBF	CBF	FDimBF(1)	FDimBF(2)	CRes
Breast Cancer	▲▲▲	—	—	—	▲▲▲	▲▲▲	▲▲▲	$\sim$
Bupa	—	▼	◇	◇	◇	▼	▲	$\downarrow$
German	—	—	▲▲	▲▲▲	▲▲▲	▲▲▲	▲▲▲	$\uparrow$
Hungarian	▲▲▲	—	▲▲▲	▲▲▲	▲▲▲	▲▲	▲▲▲	$\uparrow$
Ionosphere	▲▲	▲	▲▲	▲	▲▲	▲▲	▲▲	$\uparrow$
Pima	▲	—	▲▲	—	—	▲▲	▼	$\sim$
Satimage	—	—	▲▲▲	—	▲▲▲	▲▲	▲▲	$\sim$
Segment	▲	▲	▲▲	▲	▲▲	▲▲	▲▲	$\uparrow$
Sonar	▲▲	—	▲▲▲	▲▲	▲▲	◇	▲▲	$\uparrow$
Vehicle	—	—	▲	—	—	▲▲	▲▲	$\sim$
Waveform	—	—	▲▲▲	▲▲▲	▲	▲▲	▲▲	$\uparrow$
Excelente (▲▲▲)	2	0	4	3	4	2	3	
Muito (▲▲)	2	0	4	1	3	7	6	
Bom (▲)	2	2	1	2	1	0	1	
Regular (◇)	0	0	1	1	1	1	0	
Ruim (▼)	0	1	0	0	0	1	1	
Todos os Atributos Selecionados (—)	5	8	1	4	2	0	0	

Tabela 6.9: Classificação dos algoritmos em relação a percentagem de atributos selecionados *versus* erro do modelo construído

Os algoritmos de SA contribuíram para a redução do número de atributos selecio-

nados em relação ao conjunto original de atributos em seis, identificados por $\uparrow$, dos 11 conjuntos de dados considerados neste trabalho, *i.e.*, houve cinco ou mais casos classificados como Excelente, Muito Bom ou Bom. Para quatro conjuntos de dados, identificados por $\sim$, a aplicação dos algoritmos de SA não promoveu a redução dos subconjuntos de atributos selecionados em 50% dos casos, embora para todos eles, os outros 50% dos casos tenham sido classificados como Excelente, Muito Bom ou Bom. Apenas em um caso, identificado por $\downarrow$, cinco dos modelos construídos utilizando os subconjuntos selecionados pelos algoritmos de SA foram classificados como Regular e Ruim.

Considerando cada algoritmo de SA em relação aos tipos de classificação, de acordo com o modelo proposto, os algoritmos CFS e CBF foram os que obtiveram o maior número de classificações excelentes, cada um deles tendo obtido quatro. Quanto às classificações muito boas, FDimBF(1) e FDimBF(2) obtiveram sete e seis, respectivamente. Classificações boas e regulares ocorreram de um modo uniforme entre todos os algoritmos. O algoritmo ReliefF juntamente com as duas versões de FDimBF foram os únicos a apresentarem classificações ruins. Ressalta-se que as duas versões do algoritmo FDimBF foram os algoritmos que obtiveram o maior número, nove, de classificações excelente e muito bom, seguidas por CFS e CBF, cada um com oito e sete classificações desses tipos, respectivamente. É interessante observar que o algoritmo ReliefF foi o que apresentou maior número, oito, de seleções de subconjuntos iguais aos conjuntos originais de atributos (não houve redução do número de atributos selecionados) e que os algoritmos FDimBF(1) e FDimBF(2) foram os únicos a promover redução do número de atributos selecionados para todos os conjuntos de dados.

Do total de 77 classificações (11 conjuntos de dados $\times$ sete algoritmos de SA), 18 foram excelentes, 23 muito boas, nove boas, quatro regulares, três ruins e 20 selecionaram todos os atributos do conjunto original de atributos. É possível observar que 64,94% das classificações foram excelentes, muito boas ou boas, 25,97% dos subconjuntos de atributos selecionados foram iguais aos conjuntos originais de atributos e apenas 9,09% foram regulares ou ruins, tendo portanto a maioria dos algoritmos de SA contribuído, utilizando os subconjuntos de atributos selecionados, para a melhoria, quer em relação à redução do número de atributos quer em relação à precisão dos modelos construídos no modelo de classificação proposto — Figura 6.6a na página 78.

6.5.5 Análise da Significância Estatística dos Resultados

Como mencionado anteriormente, para cada conjunto de dados, os algoritmos foram comparados entre si quanto à média do erro do modelo induzido estimado por meio de validação cruzada com 10 partições utilizando o teste não paramétrico Kruskal-Wallis para grupos não pareados, seguido do pós-teste de Dunn, e nível de significância de 95%. Esses resultados foram comparados também em relação ao número de vezes que cada algoritmo, para um determinado conjunto de dados, seleciona menos atributos com uma média de

erro sem diferença estatística.

A Tabela 6.11 apresenta, para cada conjunto de dados sem SA (Orig.) e cada um dos subconjuntos selecionados pelos algoritmos considerados neste trabalho — $\mathcal{C}4.5$, ReliefF, CFS, FBCF, CBF e as duas versões de FDimBF representadas por DF(1) e DF(2) nessa tabela — o número original de atributos e o número de atributos selecionados por cada um desses algoritmos. As comparações entre as médias dos erros dos modelos construídos que apresentaram diferença estatisticamente significativa estão em negrito. As siglas utilizadas para representar os conjuntos de dados são apresentadas na Tabela 6.10.

Sigla	Conjunto de Dados
BCa	Breast Cancer
Bup	Bupa
Ger	German
Hun	Hungarian
Ion	Ionosphere
Pim	Pima
Sat	Satimage
Seg	Segment
Son	Sonar
Veh	Vehicle
Wav	Waveform

Tabela 6.10: Siglas para os conjuntos de dados

	BCa	Bup	Ger	Hun	Ion	Pim	Sat	Seg	Son	Veh	Wav
Orig.- $\mathcal{C}4.5$	9-7	6-6	24-24	10-9	34-15	8-7	36-36	19-16	60-15	18-18	21-21
Orig.-ReliefF	9-9	6-5	24-24	10-10	34-33	8-8	36-36	19-18	60-60	18-18	21-21
Orig.-CFS	9-9	6-1	24-2	10-3	34-14	8-3	36-23	19-5	60-19	18-11	21-15
Orig.-FCBF	9-9	6-1	24-15	10-6	34-33	8-8	36-36	19-18	60-21	18-18	21-19
Orig.-CBF	9-7	6-1	24-15	10-5	34-7	8-8	36-12	19-9	60-14	18-18	21-12
Orig.-DF(1)	9-3	6-4	24-12	10-3	34-3	8-3	36-6	19-4	60-5	18-6	21-3
Orig.-DF(2)	9-3	6-4	24-12	10-4	34-4	8-4	36-6	19-4	60-10	18-6	21-3

Tabela 6.11: Comparação entre o número original de atributos e o número de atributos selecionados pelos algoritmos de SA. Comparação entre médias de erros dos modelos construídos (em negrito resultados estatisticamente significativos)

O algoritmo ReliefF foi o que proporcionou menor número de reduções, apenas três, do número de atributos nos 11 conjuntos de dados. Os algoritmos que apresentaram maior número de vezes em que houve redução no subconjunto de atributos selecionados foram as duas versões de FDimBF, as quais sempre reduziram os subconjuntos de atributos selecionados. Por outro lado, esses dois algoritmos apresentaram degradação da performance do modelo de classificação induzido, com diferença estatisticamente significativa, em nove do total de 22 comparações entre as médias dos erros dos modelos induzidos considerando os subconjuntos de atributos selecionados pelos algoritmos FDimBF e o conjunto original de atributos, *i.e.*, sem SA.

Uma análise análoga à apresentada anteriormente foi realizada comparando-se todos os algoritmos entre si. Na Tabela 6.12 são apresentados, para cada conjunto de dados (si-

glia e número original de atributos) e cada comparação, o número de atributos selecionados pelos algoritmos identificados na primeira coluna. Assim como na Tabela 6.11, nessa tabela as comparações entre as médias de erros que apresentaram diferença estatisticamente significativa estão apresentadas em negrito.

Na Tabela 6.13 são apresentados, para cada conjunto de dados, o número de vezes em que cada algoritmo selecionou um subconjunto menor de atributos considerando as comparações em que os algoritmos apresentaram estatisticamente performances similares em relação ao erro (Ganho). Por exemplo, para o conjunto de dados Breast Cancer (BCa), o algoritmo *C4.5* comparado a todos os outros algoritmos, selecionou três vezes (*C4.5-ReliefF*, *C4.5-CFS* e *C4.5-FCBF*) subconjuntos menores de atributos — segunda coluna e terceira linha da Tabela 6.12. Na penúltima coluna é apresentado o total de Ganhos para cada algoritmo (Ganhos). Na última coluna dessa tabela, é informado o número total de conjuntos de dados para os quais cada algoritmo foi o vencedor (Ganhos por Conjunto de Dados). Casos nos quais foram selecionados o mesmo número de atributos por dois algoritmos não foram computados.

	BCa 9	Bup 6	Ger 24	Hun 10	Ion 34	Pim 8	Sat 36	Seg 19	Son 60	Veh 18	Wav 21
<i>C4.5-ReliefF</i>	7-9	6-5	24-24	9-10	15-33	7-8	36-36	16-18	15-60	18-18	21-21
<i>C4.5-CFS</i>	7-9	6-1	24-2	9-3	15-14	7-3	36-23	16-5	15-19	18-11	21-15
<i>C4.5-FCBF</i>	7-9	6-1	24-15	9-6	15-33	7-8	36-36	16-18	15-21	18-18	21-19
<i>C4.5-CBF</i>	7-7	6-1	24-15	9-5	15-7	7-8	36-12	16-9	15-14	18-18	21-12
<i>C4.5-DF(1)</i>	7-3	6-4	24-12	9-3	15-3	7-3	36-6	16-4	15-5	18-6	21-3
<i>C4.5-DF(2)</i>	7-3	6-4	24-12	9-4	15-4	7-4	36-6	16-4	15-10	18-6	21-3
<i>ReliefF-CFS</i>	9-9	5-1	24-2	10-3	33-14	8-3	36-23	18-5	60-19	18-11	21-15
<i>ReliefF-FCBF</i>	9-9	5-1	24-15	10-6	33-33	8-8	36-36	18-18	60-21	18-18	21-19
<i>ReliefF-CBF</i>	9-7	5-1	24-15	10-5	33-7	8-8	36-12	18-9	60-14	18-18	21-12
<i>ReliefF-DF(1)</i>	9-3	5-4	24-12	10-3	33-3	8-3	36-6	18-4	60-5	18-6	21-3
<i>ReliefF-DF(2)</i>	9-3	5-4	24-12	10-4	33-4	8-4	36-6	18-4	60-10	18-6	21-3
<i>CFS-FCBF</i>	9-9	1-1	2-15	3-6	14-33	3-8	23-36	5-18	19-21	11-18	15-19
<i>CFS-CBF</i>	9-7	1-1	2-15	3-5	14-7	3-8	23-12	5-9	19-14	11-18	15-12
<i>CFS-DF(1)</i>	9-3	1-4	2-12	3-3	14-3	3-3	23-6	5-4	19-5	11-6	15-3
<i>CFS-DF(2)</i>	9-3	1-4	2-12	3-4	14-4	3-4	23-6	5-4	19-10	11-6	15-3
<i>FCBF-CBF</i>	9-7	1-1	15-15	6-5	33-7	8-8	36-12	18-9	21-14	18-18	19-12
<i>FCBF-DF(1)</i>	9-3	1-4	15-12	6-3	33-3	8-3	36-6	18-4	21-5	18-6	19-3
<i>FCBF-DF(2)</i>	9-3	1-4	15-12	6-4	33-4	8-4	36-6	18-4	21-10	18-6	19-3
<i>CBF-DF(1)</i>	7-3	1-4	15-12	5-3	7-3	8-3	12-6	9-4	14-5	18-6	12-3
<i>CBF-DF(2)</i>	7-3	1-4	15-12	5-4	7-4	8-4	12-6	9-4	14-10	18-6	12-3
<i>DF(1)-DF(2)</i>	3-3	4-4	12-12	3-4	3-4	3-4	6-6	4-4	5-10	6-6	3-3

Tabela 6.12: Comparação entre os números de atributos selecionados pelos algoritmos de SA. Comparação entre as médias de erros dos modelos construídos (em negrito resultados estatisticamente significativos)

Do total de 11 conjuntos de dados, os algoritmos FDimBF não apresentaram boa performance para dois deles, Segment e Waveform. Para Segment, todos os subconjuntos de atributos selecionados pelas duas versões de FDimBF apresentaram erros estatisticamente maiores que os erros apresentados pelos modelos construídos utilizando os atributos selecionados pelos outros algoritmos de SA. Para Waveform, as comparações entre as duas

	BCa	Bup	Ger	Hun	Ion	Pim	Sat	Seg	Son	Veh	Wav	Ganhos	Ganhos por Conjunto de Dados
ReliefF	0	1	0	0	0	0	0	0	0	0	0	1	0
C4.5	3	0	0	1	2	3	0	2	3	0	0	14	0
CFS	0	4	6	5	3	5	3	4	2	4	3	39	6
FCBF	0	4	2	2	0	0	0	0	1	0	2	11	1
CBF	3	4	2	2	4	0	4	3	4	0	4	30	3
DF(1)	5	2	4	5	6	5	2	0	4	3	1	37	4
DF(2)	5	2	4	4	3	4	2	0	5	3	1	33	2

Tabela 6.13: Resumo do número de vezes em que cada algoritmo seleciona um subconjunto menor de atributos

versões de FDimBF e os algoritmos ReliefF, C4.5, CFS e FCBF resultaram em diferenças estatisticamente significativas com erros maiores para FDimBF. Os algoritmos CBF e FDimBF(1) e (2) apresentaram erros estatisticamente similares, tendo porém as duas versões de FDimBF selecionado apenas um quarto do total de atributos selecionados por CBF. Em outros quatro conjuntos de dados houve diferença estatisticamente significativa entre os erros de FDimBF (maiores) e os outros algoritmos (menores): Ionosphere, Satimage, Sonar e Vehicle.

Embora em 13,85% das comparações os algoritmos FDimBF tenham apresentado performances piores quanto ao erro dos modelos construídos com os subconjuntos de atributos selecionados, quando comparados aos outros algoritmos de SA, considerando um panorama geral do número de ganhos, as duas versões de FDimBF juntamente com o algoritmo CFS, apresentaram os maiores números de ganhos do número de vezes em que selecionaram menos atributos com performance estatisticamente similar. Do ponto de vista de ganhos por conjunto de dados, a mesma classificação geral foi seguida, tendo CFS vencido em seis do total de 11 conjuntos de dados e FDimBF(1) vencido em quatro casos.

6.5.6 Características dos Conjuntos de Dados Associadas à Utilização da Dimensão Fractal como uma Medida Adequada para a Seleção de Atributos

Algoritmos de aprendizado podem ser utilizados para extrair conhecimento dos dados obtidos nos diversos experimentos realizados. O nosso objetivo é relacionar o comportamento do algoritmo FDimDF com as diversas características dos 11 conjuntos de dados utilizados nos experimentos. O processo de gerar um conjunto de regras capazes de relacionar esses dois conceitos é denominado de *meta-aprendizado*, e os dados utilizados de *meta-dados*. Nesta seção é apresentado o meta-aprendizado realizado para analisar a relação entre as características dos conjuntos de dados e a adequabilidade do uso dos algoritmos FDimDF para a seleção de atributos.

Os meta-dados são compostos por 132 exemplos⁵ descritos por oito atributos — Tabela 6.14 —, os quais não possuem valores desconhecidos nem exemplos conflitantes ou duplicados. Nessa tabela são apresentados a identificação e o tipo (Tipo), o nome e uma descrição para cada atributo.

Atributo (Tipo)	Nome	Descrição
a1 (nominal)	abordagem1	conjunto original de atributos e subconjuntos de atributos selecionados por cada um dos sete algoritmos de SA considerados neste trabalho a serem comparados com abordagem2
a2 (nominal)	abordagem2	algoritmos de SA considerados neste trabalho
a3 (nominal)	comparacao	comparação entre número de atributos selecionados pela abordagem1 e pela abordagem2
a4 (nominal)	diferenca	se há diferença significativa entre as médias dos erros das abordagens 1 e 2 medidos por validação cruzada com 10 partições
a5 (nominal)	fractal	característica fractal do conjunto de dados
a6 (numérico)	pontos	número de pontos utilizados para construir a curva de comportamento do conjunto de dados
a7 (numérico)	proporcao	proporção do número de exemplos em relação ao número de atributos
a8 (nominal)	formato-orig	formato da distribuição da maioria dos atributos originais do conjunto de dados sem SA

Tabela 6.14: Descrição dos atributos da meta-base

Essa análise foi realizada sob dois aspectos:

1. características gerais dos conjuntos de dados e adequação da utilização da dimensão fractal como medida para a remoção de atributos redundantes e
2. padrões encontrados na aplicação dos algoritmos FDimBF para os conjuntos de dados considerados neste trabalho.

Desse modo, os meta-dados foram organizados em duas meta-bases, Meta1 e Meta2, para as quais é apresentado o resumo das características na Tabela 6.15.

É importante notar que para essas duas meta-bases Meta1 (características gerais dos conjuntos de dados descritas pelos atributos a5, a6, a7 e a8) e Meta2 (características gerais dos conjuntos de dados associadas às características da aplicação dos algoritmos FDimBF descritas por todos os oito atributos apresentados na Tabela 6.14), foi considerada como classe o desempenho das versões FDimBF(1) e FDimBF(2) em relação à classificação no gráfico de percentagem de atributos selecionados por média do erro do modelo construído utilizando validação cruzada com 10 partições — Tabela 6.9 na página 80.

Para cada uma das meta-bases, foi utilizada a ferramenta *See5* (Rulequest-Research, 1999), uma versão posterior do algoritmo *C4.5*, executado usando parâmetros com valores padrão, para a indução de regras de decisão. A seguir são apresentadas essas duas análises.

⁵Onze conjuntos de dados $\times$ 12 comparações: sem SA, *C4.5*, ReliefF, CFS, FCBF e CBF comparados a FDimBF(1) e FDimBF(2).

Conjunto de Dados	# Exemplos	Atributos Utilizados	# Atributos (num.,nom.)	Classes	Classe %	Erro da CM
Meta1	132	a5, a6, a7 e a8	4 (2,2)	excelente muito bom bom regular ruim	22,73% 59,09% 4,55% 4,55% 9,09%	40,91% sobre muito bom
Meta2	132	todos	8 (2,6)	excelente muito bom bom regular ruim	22,73% 59,09% 4,55% 4,55% 9,09%	40,91% sobre muito bom

Tabela 6.15: Resumo das meta-bases

6.5.6.1 Características Gerais dos Conjuntos de Dados e Adequação do Uso dos Algoritmos FDimBF

A primeira meta-base Meta1 contém quatro atributos, a5, a6, a7 e a8 — Tabela 6.15 — os quais apresentam, como mencionado, informações sobre características gerais dos conjuntos de dados:

- a5 (**fractal**): característica fractal do conjunto de dados — Tabela 6.3 na página 70;
- a6 (**pontos**): número de pontos utilizados para construir a curva de comportamento do conjunto de dados — Tabela 6.3 na página 70;
- a7 (**proporcao**): proporção de exemplos por atributos e
- a8 (**formato-orig**): formato da distribuição da maioria dos atributos originais do conjunto de dados sem SA — Tabela 6.6 na página 76.

Como mencionado anteriormente, foi considerado como classe o desempenho de cada um dos casos em relação à classificação no gráfico de percentagem de atributos selecionados por média do erro do modelo construído utilizando validação cruzada com 10 partições.

O objetivo dessa primeira análise foi verificar se existe alguma relação entre o desempenho dos algoritmos FDimBF e as características gerais dos conjuntos de dados considerados. Em outras palavras, o intuito foi encontrar características dos conjuntos de dados que pudessem prover uma idéia se a utilização da dimensão fractal como medida para a remoção de atributos redundantes era adequada e, conseqüentemente, se os algoritmos FDimBF poderiam ser apropriados para a seleção de atributos relevantes e não redundantes para conjuntos de dados com certas características.

O modelo induzido utilizando a meta-base Meta1 consiste de oito regras e o erro aparente desse modelo é de 4,5%, o qual encontra-se concentrado na classe Muito Bom.

A estimativa do erro desse modelo e o erro padrão utilizando 10 partições com validação cruzada são de 6,1% e 1,5%, respectivamente.

Todos os quatro atributos da meta-base Meta1 participam do modelo induzido. Considerando como critério de importância o número de vezes em que os atributos aparecem nas regras induzidas, os atributos mais importantes desse modelo, foram a proporção de exemplos por atributo (em seis das oito regras) e o formato da distribuição da maioria dos atributos do conjunto original de dados (em cinco das nove regras).

Em relação à proporção de exemplos por atributo na representação de conjuntos de dados, não há um consenso sobre que proporção seria adequada, porém, uma regra geral é que quanto maior essa proporção melhor deve ser essa representação. A determinação dessa quantidade depende de diversos fatores, dentre eles os métodos que serão usados para explorar e construir modelos a partir desses dados e a própria complexidade do domínio ao qual esses dados pertencem. Dentre as proporções recomendadas na literatura, há a descrição de que uma proporção mínima aceitável seria de cinco exemplos para cada atributo. Porém, há autores que defendam que uma proporção mais aceitável seria de 10 exemplos, enquanto outros propõem que essa proporção deva ser de 20 exemplos por atributo (Hair et al., 1998).

As regras induzidas com o conjunto de meta-dados Meta1 mostraram diversos padrões consistentes com o conhecimento prévio, tal qual a regra:

```
SE proporcao > 96,13
  ENTÃO classe = Muito Bom [36; 0,974]
```

Os números entre colchetes indicam que essa regra cobre 36 exemplos do total de 78 exemplos dessa classe do conjunto de meta-dados Meta1 com grau de confiança⁶ de 0,974. Essa regra indica que havendo uma alta proporção entre o número de exemplos por atributo, o conjunto de dados apresentaria uma performance muito boa utilizando o algoritmo FDimBF.

Outra regra que apresentou conhecimento consistente tem como base a proporção de exemplos por atributo e o número de pontos considerados para construir a curva de comportamento do conjunto de dados, mostrada a seguir:

```
SE pontos <= 6
  E proporcao > 10,32
  E proporcao <= 41,67
  ENTÃO classe = Excelente [18; 0,950]
```

⁶Calculado usando a razão de Laplace: $\frac{E_{corr} - E_{incorr} + 1}{E_{corr} + 2}$ na qual E_{corr} e E_{incorr} correspondem ao número de exemplos classificados corretamente e ao número de exemplos classificados incorretamente pela regra, respectivamente (Rulequest-Research, 1999).

Uma análise dos exemplos cobertos por essa regra, revelou que todos apresentavam entre cinco e seis pontos considerados para construir a curva de comportamento do conjunto de dados e proporção de exemplos por atributo variando de 26,1 a 41,67. Assim, conjuntos de dados apresentando razoável número de pontos e proporção de exemplos por atributo, teriam boa probabilidade de apresentarem excelente performance utilizando o algoritmo FDimBF.

Outra regra interessante, apresentada a seguir, indica que conjuntos de dados apresentando formato da distribuição Tipo 2 (T_2) para a maioria dos atributos do conjunto original de dados permitiriam a seleção de subconjuntos de atributos, por meio do algoritmo FDimBF, que gerariam muito bons modelos.

SE formato-orig = Tipo 2

ENTÃO classe = Muito Bom [12; 0,929]

É interessante ressaltar que os exemplos classificados como Excelente apresentam formato da distribuição da maioria dos atributos do conjunto original de dados dos tipos 1 (T_1) e 3 (T_3) e os exemplos classificados como Muito Bom foram todos classificados como dos tipos 1, 2 e 3.

6.5.6.2 Padrões Encontrados na Aplicação dos Algoritmos FDimBF para os Conjuntos de Dados

Como citado anteriormente, a segunda meta-base Meta2 — Tabela 6.15 na página 86 — contém informações sobre características gerais dos conjuntos de dados e características da aplicação dos algoritmos FDimBF. Essas informações são descritas por todos os oito atributos apresentados na Tabela 6.14 na página 85 e os exemplos classificados do mesmo modo como foram rotulados em Meta1.

O modelo induzido com essa meta-base consiste de 12 regras. O erro aparente desse modelo é de 0,0% e a estimativa do erro e o erro padrão utilizando 10 partições com validação cruzada são ambos de 0,0%. Isso mostra que para o conjunto de metadados Meta2 foi encontrada uma hipótese que cobre perfeitamente o(s) conjunto(s) de treinamento, *i.e.* a hipótese (modelo) induzida é completa e consistente.

Dos oito atributos da meta-base Meta2, somente os atributos a1 (**abordagem1**) e a3 (**comparacao**) não participam do modelo induzido. Nesse modelo, os atributos considerados mais importantes, segundo o critério de importância de número de vezes em que os atributos aparecem nas regras induzidas, foram o número de pontos utilizados para construir a curva de comportamento do conjunto de dados (nove vezes), juntamente com a proporção de exemplos por atributo (oito vezes) e o formato da distribuição da maioria dos atributos do conjunto original de dados (sete vezes). A determinação desses atributos como sendo os mais importantes está em sintonia com o conhecimento prévio.

Dentre essas 12 regras algumas apresentaram conhecimento similar ao apresentado pelas regras induzidas usando Meta1. Entre elas, a de que conjuntos de dados apresentando formato da distribuição tipo 2 (T_2) para a maioria dos atributos do conjunto original de dados permitiriam a seleção de subconjuntos de atributos, por meio do algoritmo FDimBF, que gerariam muito bons modelos.

Outra regra interessante, a qual também considera o formato da distribuição da maioria dos atributos do conjunto original de dados, é a seguinte:

```
SE diferenca = nao
  E formato-orig = Tipo 1
  ENTÃO classe = Excelente [24; 0,962]
```

Nessa regra, a qual cobre 24 exemplos, o algoritmo FDimBF apresentando performance estatisticamente similar aos outros algoritmos de seleção de atributos considerados, e o conjunto de dados apresentando formato da distribuição tipo 1 (T_1), permitiriam a seleção de subconjuntos de atributos para a construção de modelos com performances excelentes.

Um outra regra interessante é a seguinte:

```
SE abordagem2 = FDimBF(1)
  E pontos <= 5
  E proporcao > 26,1
  E formato = Tipo 3
  ENTÃO classe = Muito Bom [18; 0,950]
```

pois cobre bem muitos exemplos. Entretanto, o formato de distribuição do tipo 3 (T_3) também participa de outra regra que cobre poucos exemplos, utiliza FDimBF(2) e cuja classe é Ruim.

Adicionalmente, foram encontrados outros padrões que descrevem relações entre a aplicação dos algoritmos, o número de pontos da curva de comportamento, a característica fractal e a distribuição dos valores da maioria dos atributos originais do conjunto de dados. Uma das regras que descreve esses padrões indica que, mesmo que a característica fractal do conjunto de dados seja mediana, se o número de pontos não for pequeno, o desempenho será também muito bom (cobertura de 24 exemplos com grau de confiança de 0,962).

```
SE fractal = regular
  E pontos > 5
  ENTÃO classe = Muito Bom [24; 0,962]
```

6.6 Considerações Finais

Neste capítulo foram apresentados os resultados da avaliação experimental de diversos algoritmos de seleção de atributos, comumente citados na literatura, conjuntamente com o algoritmo FDimBF proposto neste trabalho. Para a realização desses experimentos, foram utilizados diversos conjuntos de dados naturais, selecionados a partir de uma minuciosa pesquisa bibliográfica de trabalhos publicados na área de seleção de atributos, os quais são freqüentemente referenciados pela comunidade. Foi também apresentado um modelo para a avaliação da performance dos algoritmos de SA, o qual considera tanto o erro do modelo gerado a partir dos subconjuntos de atributos selecionados por esses algoritmos quanto a percentagem de redução da quantidade de atributos desses subconjuntos.

Os resultados foram discutidos para cada uma das questões relacionadas à seleção de atributos e à medida de dimensão fractal, considerada neste trabalho para a análise de redundância de atributos. Os conjuntos de dados utilizados na avaliação experimental foram analisados quanto à dimensão fractal e à exibição de comportamento fractal. Os resultados obtidos foram também discutidos em relação aos subconjuntos de atributos selecionados e às características dos atributos desses conjuntos de dados e dos subconjuntos de atributos selecionados. O modelo de avaliação de performance foi aplicado a cada um dos algoritmos de SA considerados. Ao final, foi construída uma meta-base contendo informações sobre os resultados da aplicação do algoritmo FDimBF aos conjuntos de dados. Foram então construídos modelos sobre essa meta-base com a finalidade de explorar possíveis características gerais dos conjuntos de dados associadas à performance de FDimBF.

Os resultados obtidos nessa avaliação experimental mostram que o algoritmo FDimBF é comparável a outros algoritmos de SA, selecionando os menores subconjuntos de atributos importantes com performances similares a algoritmos como o CFS (*Correlation-Based Feature Selection*). Assim, concluímos que a dimensão fractal pode ser também considerada uma boa candidata para realizar seleção de atributos na área de aprendizado de máquina, na qual não é de nosso conhecimento que ela tenha sido utilizada.

Capítulo 7

Estudo de Caso

7.1 Considerações Iniciais

No capítulo anterior foi apresentada uma avaliação experimental do algoritmo FDimBF e do modelo de avaliação de performance propostos neste trabalho, utilizando conjuntos de dados naturais obtidos a partir do repositório de dados da UCI (Newman et al., 1998).

Neste capítulo é apresentado um estudo de caso usando um conjunto de dados reais. Os algoritmos de SA usados neste trabalho, incluindo o algoritmo FDimBF, são avaliados segundo os mesmos critérios apresentados no capítulo anterior. Neste estudo de caso, utilizando um conjunto de dados reais da área médica, contou-se com a participação de especialistas do domínio que atuam nos projetos de Computação Aplicada à Medicina (Wu et al., 2006, 2005a,b, 2004; Voltolini et al., 2003; Verza Junior et al., 2003; Machado et al., 2002) e Análise Inteligente de Dados (Honorato et al., 2005b; Monard and Lee, 2003; Ferro et al., 2002; Esteves et al., 2001). Esses projetos estão sendo desenvolvidos em uma parceria entre o Laboratório de Bioinformática — LABI — Universidade Estadual do Oeste do Paraná, UNIOESTE; o Laboratório de Inteligência Computacional — LABIC — Universidade de São Paulo, USP/São Carlos; o Serviço de Coloproctologia da Faculdade de Ciências Médicas — FCM — Universidade Estadual de Campinas, Unicamp e o Centro de Referência em Infertilidade Masculina — Androfert.

Ressalta-se que os dados originais desse estudo de caso, os quais consistem de laudos semi-estruturados de processamento de sêmen, foram mapeados para o formato atributo-valor por meio de uma metodologia proposta neste trabalho. Essa metodologia foi implementada em um sistema computacional, que auxilia o especialista na construção de bases de dados estruturadas a partir de laudos médicos semi-estruturados.

7.2 Etapas Realizadas para o Desenvolvimento do Estudo de Caso

Neste trabalho, a organização do estudo de caso inclui a utilização da metodologia para mapeamento de laudos médicos proposta com a finalidade de auxiliar na construção da base de dados estruturada. A partir dessa base de dados, foi possível extrair o conjunto de dados no formato atributo-valor. Desse modo, esse estudo de caso compreende as seguintes sete etapas:

1. Aquisição de conhecimento do domínio e interação com especialistas do domínio;
2. Coleta de dados (laudos semi-estruturados);
3. Aplicação da metodologia para mapeamento dos laudos no formato atributo-valor;
4. Limpeza e preparação dos dados;
5. Seleção de atributos;
6. Construção dos modelos e
7. Avaliação dos modelos.

A seguir são descritas as tarefas em cada uma dessas etapas, incluindo a apresentação da metodologia proposta e implementada neste trabalho para o mapeamento de laudos médicos.

7.3 Análise Seminal e Processamento de Sêmen Diagnóstico — Etapa 1

Entre 10,00% a 20,00% dos casais em fase reprodutiva apresentam problemas relacionados à fertilidade. Um casal é considerado infértil caso não ocorra a gravidez após um período aproximado de um ano de relacionamento sexual ativo, sem uso de qualquer método anticoncepcional. É estimado que de 4,00% a 17,00% desses casais procurem assistência médica. Aproximadamente 55,00% dos casos de infertilidade conjugal estão relacionados a fatores masculinos, sendo esses fatores os responsáveis exclusivos por cerca de 35,00% dos casos de infertilidade (Bendhack and Damião, 1999). Diversos são os fatores que interferem na fertilidade masculina, tal qual o uso de fumo, álcool e drogas, contato excessivo com defensivos agrícolas, trauma testicular, anomalias congênitas, como criptorquidia¹,

¹Testículo retido; ausência completa ou incompleta da descida dos testículos intra-abdominais para o saco escrotal.

cirurgias vesicais², retroperitoneais³ e pélvicas (Medeiros, 1993).

Para que ocorra a fecundação do óvulo, um dos fatores fundamentais é a existência de um número suficiente de espermatozóides, e que os mesmos tenham movimentação efetiva no interior do aparelho reprodutor feminino a fim de que esses espermatozóides possam atingir e penetrar no interior do óvulo, originando o processo de fecundação. Além disso, é fundamental que exista um número adequado de espermatozóides com tamanho e formato normais. Espermatozóides que apresentam estrutura anormal, por exemplo, duas cabeças, cauda enrolada, cabeça muito grande ou muito pequena, não serão capazes de alcançar o óvulo e fecundá-lo. Qualquer processo que afeta negativamente a quantidade, a estrutura e o funcionamento dos espermatozóides pode causar infertilidade masculina (Esteves, 2005).

A *análise seminal* é um dos primeiros exames a serem realizados para a avaliação da fertilidade masculina. Esse exame proporciona os melhores dados disponíveis para a detecção da normalidade aparente do ejaculado (Medeiros, 1993). Inicialmente é importante avaliar se o volume do esperma, o pH, a viscosidade, a cor, o odor e a liquefação do sêmen estão dentro dos parâmetros considerados normais. Após, diversos testes são realizados, como vitalidade e morfologia dos espermatozóides, presença de leucócitos no sêmen e contagem do número e motilidade de espermatozóides.

Em casos nos quais é indicada a utilização de técnicas de Reprodução Assistida — RA — pela impossibilidade ou insucesso no tratamento das causas da infertilidade, é necessária a realização de processamento de sêmen. Esse exame é denominado *processamento de sêmen terapêutico* quando o propósito é selecionar ou recuperar a maior quantidade possível de espermatozóides de boa qualidade para serem utilizados na reprodução assistida. Por outro lado, quando esse exame é realizado na fase de investigação, *i.e.*, para quantificar a qualidade do sêmen, o exame é conhecido como *processamento de sêmen diagnóstico*. O número de espermatozóides recuperados pelo processamento de sêmen é um dos critérios considerados para a determinação de que técnica de reprodução assistida, de menor ou de maior complexidade, é a mais apropriada. Essa variação de complexidade no tratamento caracteriza-se por meio de três técnicas para a reprodução assistida:

Inseminação Intra Uterina — IUI: nessa técnica, os espermatozóides são depositados diretamente no interior do útero (menor complexidade);

Fertilização *In Vitro* — FIV: nessa técnica, a aproximação dos gametas, masculino e feminino, é realizada em laboratório, de modo que a fertilização ocorra espontaneamente em um ambiente que seja similar ao meio tubário (complexidade mediana) e

Injeção Intracitoplasmática do Espermatozóide no Óvulo — ICSI: nessa técnica,

²Cirurgias de bexiga.

³Região que fica atrás do peritônio.

o espermatozóide selecionado é inserido diretamente no interior do óvulo, com o auxílio de micromanipuladores que operam micropipetas (maior complexidade).

O processamento de sêmen constitui um processo bastante custoso. A realização desse processo pode elevar o custo do exame em aproximadamente 70% do valor de uma análise seminal. Essa elevação de custo se deve principalmente a três fatores: necessidade de equipamentos especiais, mão de obra qualificada e tempo gasto para a realização do exame.

Assim, um dos interesses deste estudo de caso é tentar prever a quantidade de espermatozóides recuperados pelo processamento de sêmen diagnóstico antes mesmo da realização dessa análise, a partir de exames menos custosos como a análise seminal. Desse modo, dependendo da qualidade da predição do modelo gerado, os especialistas poderiam decidir por uma técnica sem a necessidade da realização do processamento de sêmen diagnóstico, baseando-se em informações da análise seminal. Um outro interesse desse estudo de caso é a avaliação do conhecimento adquirido, conjuntamente com os especialistas do domínio, com a finalidade de verificar se esse conhecimento, ou parte dele, pode ser considerado interessante e/ou novo.

Deve ser observado que algumas informações são obtidas tanto pelo exame de análise seminal (baixo custo) quanto pelo exame de processamento de sêmen (alto custo), tal como a classificação de motilidade, a qual categoriza os espermatozóides em:

1. Grau A: motilidade rápida, linear e progressiva;
2. Grau B: motilidade linear lenta ou movimentos não-lineares;
3. Grau C: motilidade não-progressiva e
4. Grau D: imóveis.

Essas informações, determinadas em ambos os exames, poderiam ser utilizadas para classificar cada caso. Desse modo, os graus de motilidade juntamente com o número de espermatozóides recuperados a partir do processamento de sêmen poderiam ser usados para atribuir as classes aos exemplos contendo informações apenas de análises seminais. É importante notar que esses laudos contêm, para um mesmo paciente, tanto informações sobre análise seminal quanto sobre processamento de sêmen, denominada neste trabalho de Análise Seminal Completa — ASC.

Na próxima seção é descrita a coleta de dados para a realização desse estudo de caso.

7.4 Coleta de Dados — Etapa 2

Os dados coletados para a realização deste estudo de caso consistem de laudos médicos que contêm informações relacionadas à análise seminal e ao processamento de sêmen. Foram coletados 717 laudos realizados no período de 31/03/1997 a 01/04/2005 em um centro de reprodução assistida localizado em Campinas, SP. Na Figura 7.1 é apresentado um exemplo desses laudos, os quais se encontravam em formato digital no padrão Word para Windows®.

LABORATÓRIO DE ANDROLOGIA (resultado de exame – 021/02)	
data: 23.01.02	nome do paciente: XX
idade: 38 anos	
exame solicitado:	Análise seminal completa
requisitado por:	Dr. YY
diagnóstico clínico:	Avaliação inicial
local da coleta:	(X) sala de coleta do laboratório () em casa
método de coleta:	(X) masturbação () outros
ejaculado:	(X) total () parcial

AVALIAÇÃO MACROSCÓPICA:	
hora da coleta: 11:30 h	processamento: 35 minutos após a coleta
tempo de abstinência: 02 dias	volume: 3,4 mL (normal: > 2 mL)
cor: (X) branco-opalescente () amarelo-opalescente () translúcido	
odor: característico	pH: 8,0 (normal ≥ 7,2)
viscosidade: () normal (X) aumentada	liquefação: (X) completa () incompleta (grumos)

AVALIAÇÃO MICROSCÓPICA:	
concentração (milhões/mL): 93,5 X 10 ⁶ espermatozoides	(normal ≥ 20,0 X 10 ⁶ /mL)
concentração total (milhões): 317,9 X 10 ⁶ espermatozoide	(normal ≥ 40,0 X 10 ⁶)
motilidade: 77% móveis	(normal ≥ 50% móveis)
classificação da motilidade: grau A: 05%; grau B: 66% grau C: 06%; grau D: 23%	(normal ≥ 50% graus A+B, ≥ 25% grau A; critério da O.M.S.)
vitalidade: 79% vivos	(normal ≥ 75% vivos)
detecção de leucócitos no sêmen (teste de peroxidase): () negativo (X) positivo	
- nº de leucócitos polimorfonucleares: 0,09 X 10 ⁶ /mL	(normal < 1,0 X 10 ⁶ /mL)
- nº de células redondas peroxidase-negativas: 2,16 X 10 ⁶ /mL	

MORFOLOGIA ESPERMÁTICA:	
Morfologia Estrita de Kruger: 09% formas normais	(normal > 14% formas normais)
OBS: Formas anormais predominantemente amorfas	

TESTE HIPO-OSMÓTICO: 82 % inchados	
(normal > 60% esperm. inchados)	

DETERMINAÇÃO DE FRUTOSE NO PLASMA SEMINAL:	
() presente () ausente (X) não solicitada	

PROCESSAMENTO DE SÊMEN:	
(X) diagnóstico () terapêutico	
() "swim-up" (X) gradiente descontinuo () mini gradiente descontinuo	
preparo do sêmen: () BWW () Enhance S-Plus (X) Percoll () HTF	
motilidade: grau A: 29%; grau B: 68%; grau C: 03%	
nº de espermatozoides recuperados/mL: 35,9 X 10 ⁶ /mL	
nº total de espermatozoides recuperados: 35,9 X 10 ⁶	

CONCLUSÕES:	
1. Teratozoospermia	
2. Ausência de sinais indicativos de infecção no fluido seminal.	
3. Integridade funcional espermática preservada.	
4. Número de espermatozoides móveis recuperados após processamento de sêmen pela técnica do gradiente descontinuo coloidal: 35,9 X 10 ⁶ .	

REFERÊNCIAS:	
1. Mortimer D. Semen analysis. In: Practical Laboratory Andrology. Oxford University Press, New York, 1994.	
2. World Health Organization (WHO): Laboratory Manual for the Examination of Human Semen and Sperm-Cervical Mucus Interaction. 4th. ed. Cambridge. The Press Syndicate of the University of Cambridge, 1999.	

(a)

(b)

Figura 7.1: Exemplo de laudo de análise seminal completa

Na próxima seção é descrita em maiores detalhes a metodologia proposta e implementada neste trabalho, a qual foi aplicada na etapa 3 — Seção 7.6 na página 101 — deste estudo de caso.

7.5 Metodologia para Construção de Bases de Dados a Partir de Laudos Médicos

Como mencionado anteriormente, a etapa de pré-processamento é, freqüentemente, a etapa mais custosa, consumindo em torno de 80% do tempo usado para realizar o processo de extração de conhecimento. Ela tem como objetivo realizar tarefas tais como preparação, redução e transformação dos dados (Pyle, 1999). Ainda na etapa de pré-processamento, é necessário que os dados estejam representados no formato apropriado para a etapa de mineração de dados, *i.e.*, de construção dos modelos, sendo um dos formatos mais comumente utilizados o formato atributo-valor — Tabela 2.1 na página 12.

Na área médica, é freqüente a apresentação de resultados de exames em laudos semi-estruturados descritos em linguagem natural. Para a construção de conjuntos de dados no formato atributo-valor é necessário que a informação contida nesses laudos seja interpretada e transformada para o formato atributo-valor. Esse processo, além de ser custoso, está sujeito à interpretação subjetiva de quem o está realizando (Ferro, 2004; Lee, 2000). Desse modo, processos para auxiliar na semi-automatização dessa tarefa poderiam prover ganho em tempo, além de proporcionar a padronização no tratamento das informações contidas em laudos médicos.

Neste trabalho é proposta uma metodologia para dar suporte à construção de bases de dados estruturadas a partir de laudos médicos semi-estruturados descritos em linguagem natural (Honorato et al., 2005a, 2004b). Nessa metodologia, inicialmente é construído um dicionário, com o auxílio de especialistas do domínio, a partir da identificação de padrões que ocorrem nos laudos. Esse dicionário é então utilizado para mapear os laudos médicos, por meio de casamento de padrões, para conjuntos de dados no formato atributo-valor. A metodologia proposta e implementada neste trabalho⁴ é composta por duas fases ilustradas na Figura 7.2.

A primeira fase caracteriza-se pela construção de um dicionário do domínio do conhecimento considerado, o qual é empregado para o processamento de laudos desse mesmo domínio durante a próxima fase. Na primeira fase o auxílio do especialista é de fundamental importância para o sucesso da construção do dicionário. Na segunda fase, como mencionado, o dicionário é utilizado para a transformação de laudos médicos desse domínio, por meio de casamento de padrões, para a construção da base de dados no formato apropriado para a mineração de dados. Deve ser observado que após o dicionário ser construído nessa primeira fase com o conjunto de dados disponíveis, esse dicionário pode ser armazenado e utilizado posteriormente para mapear automaticamente novos laudos, *i.e.*, sem a necessidade de construí-lo novamente. Ambas as fases são descritas a seguir.

⁴A implementação foi realizada na linguagem Perl (Schwartz et al., 1997) utilizando o paradigma de orientação a objetos.

Figura 7.2: Metodologia proposta (Honorato et al., 2005a)

7.5.1 Primeira Fase

A construção do dicionário é realizada por meio de quatro etapas iterativas e interativas:

1. identificação de frases únicas;
2. construção de arquivo de padronização;
3. remoção de *stopwords* e aplicação de *stemming* e
4. construção da base de conhecimento do dicionário.

O objetivo das três primeiras etapas, descritas a seguir, é auxiliar no processo de identificação dos padrões contidos nos laudos para que esses padrões possam ser mapeados para o dicionário.

Identificação de frases únicas: consiste na identificação de frases únicas existentes na coleção de laudos utilizada para a construção da base de dados. Supõe-se que as informações presentes nos laudos estejam mapeadas por meio de frases, nos quais cada frase refere-se a um diagnóstico, um prognóstico ou uma observação do médico sobre o exame realizado. O formato dos laudos médicos pode variar de acordo

com a área de especialidade, porém, muitos desses documentos possuem formatos similares. As frases contidas em cada laudo são coletadas em um único documento e organizadas em ordem alfabética. Esse processo permite reunir frases repetidas, uma vez que a mesma frase está presente, freqüentemente, em diversos laudos. Essas frases repetidas são removidas e apenas um exemplar de cada frase é mantido. Ao final dessa etapa, obtém-se como resultado um primeiro conjunto de frases únicas — CFU1 — relacionado à coleção de laudos.

Construção de arquivo de padronização: a freqüente utilização de sinônimos na descrição de informações semelhantes presentes nos laudos médicos ou a presença de frases que expressam informações de uma maneira diferente da que será utilizada pelo dicionário, faz com que a padronização das informações contidas nos laudos seja necessária. Após a obtenção de CFU1 é possível identificar parte das informações que poderão ser padronizadas. A construção do arquivo de padronização é realizada à medida que informações que podem ser padronizadas são identificadas e continua até o final da primeira etapa da metodologia proposta. A aplicação da padronização permitirá que as informações contidas nos laudos estejam mapeadas em um formato padrão para ser utilizado pelo dicionário e pelo processo de preenchimento da base de dados na segunda fase da metodologia proposta. Na etapa de construção do arquivo de padronização, o auxílio de especialistas do domínio é de fundamental importância. Na Tabela 7.1 são apresentados dois exemplos de padronização.

Antes da Padronização	Depois da Padronização
coloração esbranquiçada	<i>anormal</i>
calibre e distensibilidade normais	<i>calibre normal</i> <i>distensibilidade normal</i>

Tabela 7.1: Exemplo de padronização

Na segunda linha é apresentada uma palavra composta sem padronização e a respectiva palavra padronizada pelo especialista. Na terceira linha é apresentada uma frase, a qual depois de padronizada pelo especialista transforma-se em outras duas frases.

Remoção de *stopwords* e aplicação de *stemming*: essa etapa tem como objetivo auxiliar, por meio da remoção de palavras que não são de interesse para a aplicação e da remoção de redundância, no processo de identificação dos padrões utilizados pelos especialistas nos laudos. Para tanto, é realizada a remoção de *stopwords* sobre o CFU1, gerando CFU2. *Stopwords* são palavras consideradas não relevantes para a análise do texto, tais como conjunções, artigos e preposições. A lista dessas palavras é construída por meio de interações com especialistas da área do domínio, nas quais, além das palavras usuais citadas, outras palavras do domínio podem ser acrescentadas na lista de *stopwords* sem que alterem, após a aplicação do algoritmo, o sentido do texto que está sendo analisado. Além da remoção de *stopwords*, é também

aplicado o processo de *stemming* que permite auxiliar na remoção de redundância de CFU2. Esse processo consiste na identificação das diferentes inflexões referentes à mesma palavra e sua substituição por um radical comum (Sebastiani, 2002). Deve ser observado que o processo de *stemming* é também analisado pelo especialista, pois não há garantias de que palavras com *stemmings* iguais tenham o mesmo significado.

Assim, as variações morfológicas das palavras são removidas, sinalizando as frases redundantes e, desse modo, possibilitando a redução da dimensão de CFU2. Tanto o CFU3, construído a partir de CFU2 após a aplicação de *stemming*, quanto o CFU2, são utilizados em dois momentos: para ajudar o especialista durante a análise das frases únicas na identificação de padrões e para auxiliar na decisão de como as informações serão organizadas na construção do dicionário.

Definição da estrutura do dicionário: como mencionado anteriormente, o dicionário é utilizado para auxiliar a construção da base de dados, isto é, no preenchimento dos valores dos atributos da base de dados utilizando as informações contidas nos laudos. Desse modo, antes de iniciar a construção do dicionário, é necessário o especialista definir quais atributos vão compor a base de dados. Após identificados os atributos, é criada a estrutura da base de dados que receberá informações a partir do processamento dos laudos utilizando o dicionário.

Assim, a construção do dicionário é realizada, conjuntamente com o especialista do domínio, com base nas informações existentes em CFU2 e CFU3 e no arquivo de padronização. Laudos médicos de diversas especialidades possuem informações organizadas na forma de estrutura anatômica e característica associada a essa estrutura. Desse modo, na metodologia proposta, a estrutura base do dicionário é composta por locais e características. Como pode ser observado em um exemplo dentro do contexto médico — Figura 7.3 —, no momento da confecção do laudo pelo especialista, mapeia-se primeiramente o local (terço distal) que está sendo observado no exame de endoscopia digestiva alta e, na sequência, a característica macroscópica desse local (com erosões). Esse procedimento é repetido até que todas as informações relacionadas ao exame tenham sido preenchidas no laudo.

Figura 7.3: Local e característica (Honorato et al., 2005a)

Na Figura 7.4 é ilustrada a estrutura base do dicionário. Conforme apresentado na figura, a lista de locais armazena o nome de um determinado local e cada local

possui uma lista de uma ou mais características associadas. A lista de características armazena, além do nome da característica, o número correspondente à posição do atributo no Registro na Base de Dados — RBD — e o valor que deverá ser armazenado no atributo correspondente do RBD.

Figura 7.4: Estrutura base do dicionário (Honorato et al., 2005a)

7.5.2 Segunda Fase

O objetivo dessa fase é processar a coleção de laudos com base nas informações mapeadas na estrutura do dicionário (locais e características) para preencher o valor dos atributos presentes na estrutura do RBD — Figura 7.5.

Figura 7.5: Construção da base de dados

Cada laudo corresponde a um exemplo na base de dados no formato atributo-valor. O processo de armazenamento recebe como entrada um laudo, no qual previamente foi aplicado o processo de padronização, e uma frase é extraída. A execução do processo de armazenamento é realizada por meio de ciclos de interações de pesquisa entre a estrutura do dicionário e a frase extraída do laudo. Primeiramente, é verificada a existência do primeiro local da lista de locais do dicionário na frase extraída. Se estiver presente, cada uma das características associadas a esse local é pesquisada na frase em análise e as informações associadas às características encontradas são armazenadas no RBD, por

meio da verificação na estrutura do dicionário da posição do atributo no qual deverá ser armazenado. O mesmo procedimento é novamente realizado para todos os locais e suas respectivas características, até que todos os locais descritos no dicionário tenham sido pesquisados na frase corrente. Esse processo é repetido até que todas as frases do laudo tenham sido completamente processadas. Ao final, o RBD, preenchido com as informações desse laudo, é inserido na base de dados e uma nova iteração é iniciada com o processamento do próximo laudo.

Esta metodologia foi aplicada na etapa 3 da estudo de caso, descrita a seguir.

7.6 Aplicação da Metodologia aos Laudos de Análise Seminal Completa — Etapa 3

Para o mapeamento dessa coleção de laudos de análise seminal completa, foi aplicada uma simplificação da metodologia anteriormente proposta, já que os laudos apresentavam informações mais estruturadas quando comparadas a outros tipos de laudos, por exemplo laudos de endoscopia digestiva alta (Honorato et al., 2005a).

Como mencionado, a construção da base de dados utilizando o sistema computacional que implementa a metodologia proposta, foi realizada a partir de uma coleção de 717 laudos médicos, na qual não consta a identificação dos pacientes. Os laudos armazenam informações semi-estruturadas descritas em linguagem natural relacionadas à análise seminal completa, conforme ilustra o laudo apresentado na Figura 7.1 na página 95.

Primeiramente, essa coleção de laudos foi mapeada para frases, as quais nesse tipo específico de laudo são representadas pela informação delimitada por “:”. Esse processo resultou em 25970 frases, que após a identificação de frases únicas existentes resultou no CFU1 contendo apenas um exemplar de cada frase, totalizando 475 frases, *i.e.*, uma redução de 98,17% no número de frases. Em seguida, foi iniciada a construção do arquivo de padronização, utilizando como base as informações contidas no CFU1. Os especialistas do domínio atuaram intensamente nessa etapa auxiliando na padronização das palavras, tais como mapeamento de sinônimos em uma palavra chave, bem como o mapeamento de algumas frases do CFU1 que não encontravam-se em um formato adequado para serem utilizadas na construção do dicionário. Na Tabela 7.2 é ilustrado um exemplo de aplicação do algoritmo de padronização, realizado neste trabalho, baseado nas informações preenchidas com o subsídio dos especialistas no arquivo de padronização.

No exemplo, são apresentados alguns possíveis valores, identificados como características, para a informação diagnóstico clínico, identificada como um local, antes e após a aplicação da padronização. As frases transformadas pelo processo de padronização estão em grafia itálica.

O mapeamento das informações identificadas em CFU1 e o auxílio dos especialistas

Antes da Padronização	Depois da Padronização
avaliacao - falha fiv	<i>avaliacao pre-trat</i>
avaliacao - oligozoospermia	<i>avaliacao pre-trat</i>
avaliacao de leucocitospermia	<i>avaliacao pre-trat</i>
avaliacao imunologica	<i>avaliacao pre-trat</i>
avaliacao - varicocele sub-clinica	<i>avaliacao pre-cir</i>
avaliacao inicial (varicocele)	<i>avaliacao pre-cir</i>

Tabela 7.2: Exemplo de padronização - Análise Seminal Completa

do domínio, resultaram em 38 atributos — Tabela 7.3. Com o dicionário estruturado, iniciou-se a segunda fase — Seção 7.5.2 na página 100 — da metodologia proposta: a construção da base de dados, no formato atributo-valor, por meio do mapeamento das informações contidas nos laudos para os registros da base dados, cada laudo correspondendo a um registro nessa base de dados.

Id. do Atributo	Nome do Atributo	Descrição do Atributo
#0	id-laudo	Identificação do laudo
#1	data	Data de realização do exame
#2	idade	Idade do paciente (anos)
#3	diagnostico-clinico	Avaliação inicial dada pelo médico
#4	ejaculado	Ejaculação total ou parcial 1: total; 2: parcial
#5	local-coleta	Local de coleta da amostra de sêmen 1: sala de coleta do laboratório; 2: em casa
#6	metodo-coleta	Método utilizado para a coleta 1: masturbação; 2: outros
#7	requisitado-por	Identificação do médico que solicitou o exame
#8	cor	Cor do sêmen coletado 1: branco-opalescente (normal); 2: amarelo-opalescente; 3: translúcido
#9	hora-coleta	Hora de coleta da amostra de sêmen
#10	processamento	Processamento do sêmen realizado após esta quantidade de minutos depois da coleta
#11	odor	Odor do sêmen coletado 1: característico; 2: forte; 3: urina
#12	pH	pH do sêmen coletado
#13	tempo-abstinencia	Tempo de abstinência (dias)
#14	volume	Volume de sêmen coletado (mililitros — mL)
#15	viscosidade	Viscosidade do sêmen coletado 1: normal; 2: aumentada
#16	liquefacao	Liquefação do sêmen coletado 1: completa; 2: incompleta
#17	concentracao	Concentração de espermatozoides

continuação na próxima página

continuação da página anterior

Id. do Atributo	Nome do Atributo	Descrição do Atributo
		por mL coletado
#18	concentracao-total	Concentração total de espermatozóides
#19	motilidade-grau-a	Classificação da motilidade Grau A
#20	motilidade-grau-b	Classificação da motilidade Grau B
#21	motilidade-grau-c	Classificação da motilidade Grau C
#22	motilidade-grau-d	Classificação da motilidade Grau D
#23	motilidade	% de espermatozóides móveis (A+B+C)
#24	vitalidade	% de espermatozóides vivos
#25	det-leu-sem	Detecção de leucócitos no sêmen 1: negativo; 2: positivo
#26	nro-leu-pol	Número de leucócitos polimorfonucleares
#27	nro-cel-red	Número de células redondas peroxidase-negativas
#28	morfologia-Kruger	% de espermatozóides com morfologia normal segundo técnica estrita de Kruger
#29	tes-hip-osm	Teste Hipo-osmótico (% de inchados)
#30	pro-sem-1	Tipo de processamento de sêmen 1: diagnóstico; 2: terapêutico
#31	pro-sem-2	Técnica utilizada 1: swim-up; 2: gradiente descontínuo; 3: mini gradiente descontínuo
#32	lav-sem	Preparo do sêmen 1: BWW; 2: Enhance S-Plus; 3: Percoll; 4: HTF
#33	pro-sem-mot-grau-a	Classificação da motilidade segundo processamento de sêmen Grau A
#34	pro-sem-mot-grau-b	Classificação da motilidade segundo processamento de sêmen Grau B
#35	pro-sem-mot-grau-c	Classificação da motilidade segundo processamento de sêmen Grau C
#36	nro-esp-mov-rec	Número de espermatozóides móveis recuperados/mL
#37	nro-tot-esp-mov-rec	Número total de espermatozóides móveis recuperados

Tabela 7.3: Atributos identificados para compor a base de dados

7.7 Limpeza e Preparação dos Dados — Etapa 4

Nesta etapa, tendo em vista o objetivo da análise a ser realizada neste estudo de caso, foram acrescentados os seguintes dois atributos à base de dados descrita na Tabela — 7.3:

- **motilidade-progressiva:** composta pela percentagem de espermatozóides de motilidade rápida, linear e progressiva (Grau A) e linear lenta e progressiva (Grau B) na análise seminal e
- **classe:** definida por três valores, baseados nas técnicas que são utilizadas no tra-

tamento para a reprodução assistida, associados à percentagem de espermatozóides classificados como de motilidade Grau A e Grau B no processamento de sêmen juntamente com a quantidade de espermatozóides recuperados:

- classe 1: $x < 1 \times 10^6$ — ICSI;
- classe 2: $1 \times 10^6 \leq x < 5 \times 10^6$ — FIV e
- classe 3: $x \geq 5 \times 10^6$ — IUI.

para x — Equação 7.1 — representando milhões de espermatozóides por mililitro (mL).

$$x = \frac{(A + B)E}{100} \quad (7.1)$$

sendo $A = \text{pro-sem-mot-grau-a}$, $B = \text{pro-sem-mot-grau-b}$ e $E = \text{nro-esp-mov-rec}$ — Tabela 7.3.

Um terceiro atributo foi modificado, a partir do atributo nominal #9 (**hora-coleta**), para o atributo **periodo-coleta** cujos valores foram atribuídos do seguinte modo:

- manhã: coletas realizadas antes das 12:00 horas e
- tarde: coletas realizadas depois das 12:00 horas.

Os atributos #0 (**id-laudo**), #1 (**data**) e #7 (**requisitado-por**) foram removidos, pois não apresentam relação direta com o domínio do problema. Já os atributos #30 a #37, foram removidos do conjunto de dados a ser considerado nos experimentos, pois estão relacionados ao processamento de sêmen, utilizados somente na construção da classe, os quais não devem ser considerados para a construção dos modelos que pretendem usar informações de análise seminal para prever o resultado do processamento de sêmen.

O tratamento de valores desconhecidos⁵ foi realizado seguindo a mesma metodologia utilizada na avaliação experimental apresentada no Capítulo 6: para valores desconhecidos concentrados em alguns poucos exemplos, esses exemplos foram removidos, enquanto que para valores desconhecidos concentrados em um atributo, a coluna correspondente foi removida do conjunto de dados. Os atributos que foram removidos em virtude de valores desconhecidos foram: #4 (**ejaculado**) e #16 (**liquefacao**) e #29 (**tes-hip-osm**). A principal razão para a remoção de valores desconhecidos do conjunto de dados, como mencionado no Capítulo 6, é que alguns dos algoritmos utilizados nos experimentos tratam valores faltantes de modo especial (Batista and Monard, 2003a), enquanto outros algoritmos não tratam esse tipo de informação. Assim, com o intuito de não introduzir interferências associadas ao uso de um ou outro método para tratar esse problema, foi decidida a remoção de valores desconhecidos do conjunto de dados.

⁵Atributos com valores desconhecidos ou faltantes foram considerados neste trabalho como aqueles cujo valor não foi informado nos laudos por algum motivo.

Outros dois atributos foram removidos por possuírem todos os exemplos contendo os mesmos valores: #6 (**metodo-coleta**, único valor ocorrido: 1) e #11 (**odor**, único valor ocorrido: 1). É importante ressaltar que apenas os casos de processamento de sêmen diagnóstico foram selecionados para compor o conjunto de dados final. No restante deste trabalho, os termos processamento de sêmen diagnóstico e processamento de sêmen serão utilizados indistintamente.

Desse modo, do total de 38 atributos originalmente registrados por meio da metodologia para mapear laudos médicos em uma base de dados estruturada, foi construído o conjunto de dados — Tabela 7.4 — a ser utilizado nos experimentos do estudo de caso. Esse conjunto é composto por 23 atributos acrescido do atributo classe contendo 407 exemplos do total de 717 iniciais e um exemplo duplicado. É importante notar que alguns exemplos foram removidos por apresentarem apenas dados referentes à análise seminal, impossibilitando a cálculo do valor do atributo classe.

Conjunto de Dados	# Exemplos	# Atributos (num.,nom.)	Classes	Classe %	Erro da CM
Proc	407	23 (17,6)	1	20,88%	40,05% sobre 3
			2	19,17%	
			3	59,95%	

Tabela 7.4: Resumo do conjunto de dados Processamento de Sêmen

Na Tabela 7.5 são apresentados os atributos considerados nesse conjunto de dados para a realização do estudo de caso. Para cada atributo é mostrado:

- Id.: identificação do atributo;
- Id. Orig.: identificação original do atributo — Tabela 7.3;
- Nome do Atributo e Descrição do Atributo: nome e descrição do atributo e
- #Valores Distintos: quantidade de valores distintos do atributo e o tipo⁶.

Os atributos acrescentados ou modificados são indicados por **novo**.

Id.	Id. Orig.	Nome do Atributo	Descrição do Atributo	#Valores Distintos		
				possíveis	atuais	tipo
#0	#2	idade	Idade do paciente (anos)	—	33	numérico
#1	#3	diagnostico-clinico	Avaliação inicial dada pelo médico	7	7	nominal
#2	#5	local-coleta	Local de coleta da amostra de sêmen 1: sala de coleta			

continuação na próxima página

⁶Esses valores foram calculados utilizando recursos disponíveis na linguagem R (R Development Core Team, 2005) (<http://www.r-project.org/>) e na ferramenta *Data Characterization Tool* — DCT (Köpt, 2002) (<http://www.metal-kdd.org/>).

continuação da página anterior

Id.	Id. Orig.	Nome do Atributo	Descrição do Atributo	#Valores Distintos		
				possíveis	atuais	tipo
			do laboratório; 2: em casa	2	2	nominal
#3	#8	cor	Cor do sêmen coletado 1: branco-opalescente (normal); 2: amarelo-opalescente; 3: translúcido	3	3	nominal
novo #4	#9	periodo-coleta	Período de coleta do sêmen: manhã ou tarde	2	2	nominal
#5	#10	processamento	Processamento do sêmen realizado após esta quantidade de minutos depois da coleta	—	22	numérico
#6	#12	pH	pH do sêmen coletado	—	6	numérico
#7	#13	tempo-abstinencia	Tempo de abstinência (dias)	—	12	numérico
#8	#14	volume	Volume de sêmen coletado (mL)	—	69	numérico
#9	#15	viscosidade	Viscosidade do sêmen coletado: 1: normal; 2: aumentada	2	2	nominal
#10	#17	concentracao	Concentração de espermatozoides por mL coletado	—	337	numérico
#11	#18	concentracao-total	Concentração total de espermatozoides	—	396	numérico
#12	#19	motilidade-grau-a	Classificação da motilidade Grau A	—	24	numérico
#13	#20	motilidade-grau-b	Classificação da motilidade Grau B	—	70	numérico
#14	#21	motilidade-grau-c	Classificação da motilidade Grau C	—	41	numérico
#15	#22	motilidade-grau-d	Classificação da motilidade Grau D	—	74	numérico
#16	#23	motilidade	% de espermatozoides móveis (A+B+C)	—	74	numérico
novo #17		motilidade-progressiva	% de espermatozoides móveis (A+B)	—	74	numérico
#18	#24	vitalidade	% de espermatozoides vivos	—	64	numérico
#19	#25	det-leu-sem	Deteção de leucócitos no sêmen: 1: negativo;			

continuação na próxima página

continuação da página anterior

Id.	Id. Orig.	Nome do Atributo	Descrição do Atributo	#Valores Distintos		
				possíveis	atuais	tipo
			2: positivo	2	2	nominal
#20	#26	nro-leu-pol	Número de leucócitos polimorfonucleares	—	106	numérico
#21	#27	nro-cel-red	Número de células redondas peroxidase-negativas	—	176	numérico
#22	#28	morfologia-Kruger	% de espermatozóides com morfologia normal segundo técnica estrita de Kruger	—	23	numérico
novo#23		classe	Classe 1: $x < 1 \times 10^6$ 2: $1 \times 10^6 \leq x < 5 \times 10^6$ 3: $x \geq 5 \times 10^6$ sendo x = milhões de espermatozóides por mL	3	3	nominal

Tabela 7.5: Atributos do conjunto de dados utilizado nos experimentos

Ao final dessa etapa, os dados foram transformados para a sintaxe requerida por cada um dos algoritmos e ferramentas utilizados neste trabalho.

7.8 Seleção de Atributos — Etapa 5 — e Construção dos Modelos — Etapa 6

Os experimentos realizados neste estudo de caso foram desenvolvidos utilizando a metodologia aplicada na avaliação experimental apresentada no Capítulo 6.

Para os algoritmos C4.5, ReliefF, CFS, CBF e FCBF, a seleção de atributos foi realizada oferecendo a esses algoritmos o conjunto de dados completo contendo ambos atributos numéricos e nominais. Já para os algoritmos FDimBF, a seleção de atributos foi realizada segundo o modelo apresentado na Figura 5.5 na página 55 para conjuntos de dados que apresentam tanto atributos numéricos quanto atributos nominais.

Em relação à estimativa de performance, foram gerados modelos considerando os atributos selecionados pelos algoritmos citados e também considerando o conjunto de dados descrito pelo conjunto original de atributos (sem SA), totalizando oito modelos. Esses modelos foram construídos usando o algoritmo *See5*⁷. Os erros dos classificadores (mode-

⁷Versão comercial do algoritmo de aprendizado supervisionado C4.5 (Quinlan, 1993), que induz árvores e regras de decisão (Rulequest-Research, 1999).

los) foram estimados por meio de validação cruzada com 10 partições e comparados usando o teste estatístico não paramétrico Kruskal-Wallis para grupos não pareados, com nível de significância de 95%.⁸ (Motulsky, 1995). Essa comparação mostrou que não há diferença estatisticamente significativa ($p\text{-valor} = 0,8986$) entre os erros dos modelos construídos usando os diferentes subconjuntos de atributos selecionados por cada um dos algoritmos de seleção de atributos, portanto não havendo a necessidade de realizar nenhum pós-teste. É importante ressaltar que o algoritmo *See5* foi escolhido para ser usado na construção dos modelos nesse estudo de caso, pois no processo de extração de conhecimento de bases de dados, como mencionado, o objetivo é dar suporte à descoberta do conhecimento contido nos dados. Desse modo, é de fundamental importância a utilização de métodos que permitam a construção de modelos que possam ser facilmente interpretados, por exemplo, por especialistas do domínio, como é o caso de modelos construídos utilizando algoritmos de aprendizado de máquina simbólico supervisionado entre os quais o algoritmo *See5*.

7.9 Avaliação dos Modelos: Resultados e Discussão — Etapa 7

Os resultados obtidos com a realização dos experimentos neste estudo de caso, apresentados nas próximas seções, foram organizados de modo similar à organização realizada no Capítulo 6.

7.9.1 Dimensão Fractal e Comportamento Fractal do Conjunto de Dados

Como mencionado, para a análise de resultados associados à DF, é importante observar:

1. formato da curva de comportamento do conjunto de dados;
2. número de pontos utilizados para construir essa curva e
3. ajuste da reta que determina a DF em relação à curva.

Nas Figuras 7.6 e 7.7 são mostradas: (a) curva de comportamento do conjunto de dados, que representa o gráfico em escala logarítmica da soma dos pontos existentes em uma célula de lado r pelo tamanho da célula r , e (b) reta que aproxima o cálculo da dimensão fractal desse conjunto de dados. As duas curvas apresentam característica fractal estatisticamente auto-similar, pois há um trecho, relativamente linear na curva, que mostra a existência de uma variação proporcional da medida que os tamanhos das

⁸Testes estatísticos realizados utilizando GraphPad InStat versão 3.06 para Windows, GraphPad Software, <http://www.graphpad.com>.

células variam. Em relação ao número de pontos usados na construção dessa curva de comportamento, em ambos os casos, foram utilizados quatro pontos. Porém, quanto à dimensão fractal, o subconjunto de atributos selecionado por FDimBF(1) não apresentou o número mínimo de três pontos para o ajuste da reta no cálculo da DF. Sendo assim, esse subconjunto não foi considerado neste trabalho para as análises posteriores. Já o algoritmo FDimBF(2) apresentou um ajuste da reta para o cálculo da DF utilizando três pontos e uma DF de 6,06, sendo portanto sete o número de atributos não redundantes.

Figura 7.6: Gráfico gerado utilizando o método *Box Count Plot* para o conjunto de dados de processamento de sêmen utilizando atributos selecionados por FDimBF(1)

Na Tabela 7.6 são apresentadas informações sobre as características fractais e seleção de atributos associadas ao subconjunto de atributos selecionado por FDimBF(2).

# Atrib. Originais	# Exemplos	# Atrib. Relevantes Numéricos	DF	# Atrib. Selecionados	# Pontos (Curva)	# Pontos (Reta)
23	407	15	6,06	7	4	3

Tabela 7.6: Informações associadas à dimensão fractal do conjunto de dados de processamento de sêmen

7.9.2 Subconjuntos de Atributos Selecionados

Os algoritmos considerados neste trabalho realizam, basicamente, seleção de atributos de dois modos: avaliação individual de atributos — ReliefF e $\mathcal{C}4.5$ — e avaliação de subconjuntos de atributos — CFS e CBF. O algoritmo $\mathcal{C}4.5$, utilizado para SA, realiza SA

Figura 7.7: Gráfico gerado utilizando o método *Box Count Plot* para o conjunto de dados de processamento de sêmen utilizando atributos selecionados por FDimBF(2)

embutida, pois os atributos usados como nós de decisão na árvore construída podem ser considerados relevantes em relação à classe e a ordem de relevância pode ser dada pelo número de vezes que aparecem na árvore. Pode-se considerar que o algoritmo FDimBF, assim como o algoritmo FCBF, realiza SA utilizando uma combinação dessas duas abordagens, pois durante a primeira etapa, a análise de relevância, há uma ordenação dos atributos segundo sua relevância em relação à classe e, durante a segunda etapa, há a busca por subconjuntos de atributos não redundantes a partir dos atributos considerados relevantes.

Na Tabela 7.7 são apresentados os resultados para o conjunto de dados de processamento de sêmen e a SA utilizando todos os algoritmos considerados neste trabalho. Essas informações estão organizadas da seguinte maneira: nome do atributo, informações sem a realização de seleção de atributos (Orig.) e atributos selecionados por cada um dos seis algoritmos — C4.5, ReliefF, CFS, FCBF, CBF e FDimBF(2). Nas últimas linhas, são apresentados o número de atributos selecionados (# Atrib. Selec.) e a respectiva percentagem (% Atrib. Selec.) para cada um dos algoritmos de SA. Para esses algoritmos também são mostrados o erro aparente, a média do erro da validação cruzada com 10 partições e o erro padrão correspondente.

Os resultados dos experimentos mostram que os algoritmos de SA selecionaram quantidades de atributos variando desde 26,09% para o algoritmo CFS até 91,30% do total de atributos para o algoritmo ReliefF. Esses resultados seguem a tendência apresentada

Id.	Nome do Atributo	Orig.	C4.5	ReliefF	CFS	FCBF	CBF	FDimBF(2)
#0	idade	*	*	*				*
#1	diagnostico-clinico	*	*	*		*	*	*
#2	local-coleta	*	*	*		*		*
#3	cor	*	*	*	*	*	*	*
#4	periodo-coleta	*	*	*		*	*	*
#5	processamento	*	*					
#6	pH	*						
#7	tempo-abstinencia	*	*	*				*
#8	volume	*		*				*
#9	viscosidade	*	*	*		*	*	*
#10	concentracao	*	*	*	*	*	*	
#11	concentracao-total	*	*	*	*	*	*	
#12	motilidade-grau-a	*	*	*		*	*	
#13	motilidade-grau-b	*	*	*		*	*	
#14	motilidade-grau-c	*	*	*		*		
#15	motilidade-grau-d	*	*	*		*	*	*
#16	motilidade	*	*	*		*	*	
#17	motilidade-progressiva	*	*	*	*	*	*	*
#18	vitalidade	*		*	*	*	*	
#19	det-leu-sem	*	*	*		*	*	*
#20	nro-leu-pol	*	*	*				
#21	nro-cel-red	*	*	*		*	*	*
#22	morfologia-Kruger	*	*	*	*	*	*	
#23	classe							
	# Atrib. Selec.	23	20	21	6	17	15	12
	% Atrib. Selec.	100,00	86,96	91,30	26,09	73,91	65,22	52,17
	Erro Aparente	10,30	10,10	9,10	15,20	12,50	11,80	16,70
	Média do Erro	28,70	27,80	29,00	24,80	26,00	26,80	29,20
	Erro Padrão	2,20	1,90	1,70	1,40	2,20	2,20	1,60

Tabela 7.7: Atributos selecionados por cada um dos algoritmos para o conjunto de dados de processamento de sêmen diagnóstico

na avaliação experimental utilizando conjuntos de dados naturais descritos no Capítulo 6. Os algoritmos CFS e FDimBF selecionaram os menores subconjuntos de atributos, com a diferença de que na avaliação experimental utilizando dados naturais o algoritmo FDimBF selecionou o primeiro e o segundo menores subconjuntos de atributos, e o algoritmo CFS o terceiro menor subconjunto. Já neste estudo de caso, essa ordem foi invertida, tendo CFS selecionado o menor subconjunto enquanto FDimBF selecionou o segundo menor subconjunto de atributos. Em relação aos outros algoritmos, as colocações e percentagens permaneceram similares às encontradas na avaliação experimental usando os conjuntos de dados naturais.

Quanto à frequência com que os atributos são selecionados pelos algoritmos de SA, pode-se observar que há dois atributos que foram escolhidos como atributos importantes por todos os algoritmos considerados: #3 (cor) e #17 (motilidade-progressiva). Os seguintes outros nove atributos foram selecionados por cinco dos seis algoritmos:

- #1 (diagnostico-clinico);
- #4 (periodo-coleta);

- #9 (viscosidade);
- #10 (concentracao);
- #11 (concentracao-total);
- #15 (motilidade-grau-d);
- #19 (det-leu-sem);
- #21 (nro-cel-red) e
- #22 (morfologia-Kruger).

Uma outra questão interessante está relacionada a atributos que nunca foram selecionados pelos algoritmos de SA. Neste estudo de caso, apenas o atributo #6 (pH) não foi considerado importante por nenhum dos algoritmos de SA considerados.

Os resultados desses experimentos também foram analisados em relação à quantidade de vezes em que os atributos apareceram nos modelos construídos usando os subconjuntos de atributos selecionados por cada um dos algoritmos de SA — Tabela 7.8. Nessa tabela, para cada algoritmo de SA, são apresentados o número de vezes que cada atributo participa em cada um dos modelos e o total de vezes que ele participa em todos os modelos construídos (Total). Nas três últimas linhas dessa tabela, são mostrados o número de atributos utilizados nos modelos (#Atrib. Modelos), o número de atributos selecionados por cada algoritmo de SA (#Atrib.Selec.⁹) e a percentagem de utilização desses atributos em relação ao total selecionado pelo respectivo algoritmo (%Uso Atrib.).

É possível observar que os três atributos mais freqüentes nos modelos construídos foram #3 (cor), #11 (concentracao-total) e #17 (motilidade-progressiva). Desses três atributos, dois deles, #3 (cor) e #17 (motilidade-progressiva), foram também os atributos selecionados por todos os algoritmos de SA — Tabela 7.7. Uma outra questão interessante está relacionada a atributos que não foram escolhidos pelos algoritmos de SA, ou caso tenham sido escolhidos, não foram utilizados nos modelos construídos. Neste estudo de caso, essa situação ocorreu somente para o atributo #8 (volume), o qual foi escolhido pelos algoritmos ReliefF e FDimBF(2), porém não aparece nos dois modelos correspondentes.

Ainda em relação à quantidade de atributos selecionados pelos seis algoritmos de SA e efetivamente utilizados nos modelos construídos, podem ser identificados três grupos. Os três algoritmos que utilizaram uma quantidade maior dos atributos previamente selecionados durante a SA foram C4.5, CFS e CBF, para os quais foram usados nos modelos construídos 100,00% dos atributos por eles selecionados. O modelo construído a partir dos atributos escolhidos pelo algoritmo ReliefF usou uma quantidade um pouco menor de

⁹Também mostrado na Tabela 7.7.

atributos em comparação com os três primeiros algoritmos, 85,71%, similar à quantidade usada pelo modelo construído usando todos os atributos, 86,96%, *i.e.*, sem SA. Já nos modelos construídos usando os atributos selecionados por FCBF e FDimBF(2), foram utilizados 70,59% e 75,00% do total de atributos selecionados por esses dois algoritmos, respectivamente.

Id.	Nome do Atributo	Orig.	C4.5	ReliefF	CFS	FCBF	CBF	DF(2)	Total
#0	idade	5	6	9	0	0	0	2	22
#1	diagnostico -clinico	3	3	5	0	0	2	8	21
#2	local-coleta	2	2	2	0	0	0	0	6
#3	cor	2	4	4	6	5	2	10	33
#4	periodo-coleta	1	1	4	0	0	1	2	9
#5	tempo -processamento	3	5	0	0	0	0	0	8
#6	pH	0	0	0	0	0	0	0	0
#7	tempo -abstinencia	1	1	3	0	0	0	5	10
#8	volume	0	0	0	0	0	0	0	0
#9	viscosidade	1	1	2	0	2	2	0	8
#10	concentracao	2	1	3	1	2	2	0	11
#11	concentracao -total	16	17	20	11	9	14	0	87
#12	motilidade -grau-a	1	2	6	0	0	1	0	10
#13	motilidade -grau-b	4	4	5	0	2	2	0	17
#14	motilidade -grau-c	5	5	12	0	5	0	0	27
#15	motilidade -grau-d	1	2	3	0	0	1	6	13
#16	motilidade	7	7	9	0	3	5	0	31
#17	motilidade -progressiva	6	5	8	8	4	4	15	50
#18	vitalidade	0	0	0	4	1	2	0	7
#19	det-leu-sem	1	1	0	0	2	3	2	9
#20	nro-leu-pol	1	1	2	0	0	0	0	4
#21	nro-cel-red	3	2	4	0	3	1	6	19
#22	morfologia -Kruger	3	5	5	3	4	4	0	24
#23	classe								
	#Atrib. Modelo	20	20	18	6	12	15	9	
	#Atrib. Selec.	23	20	21	6	17	15	12	
	%Uso Atrib.	86,96	100,00	85,71	100,00	70,59	100,00	75,00	

Tabela 7.8: Número de vezes que cada atributo apareceu no modelo construído

7.9.3 Formatos Aproximados de Distribuição dos Valores dos Atributos em Relação aos Atributos Selecionados pelo Algoritmo FDimBF

A informação em relação aos formatos aproximados de distribuição dos valores dos atributos — Figura 6.5 na página 75 — do conjunto de dados considerado neste estudo de caso, é apresentada na Tabela 7.9. As distribuições dos valores dos atributos são mostradas no Apêndice A.

Tipo de Formato	# Atrib.
T_1	10
T_2	3
T_3	8
T_4	0
T_5	0
T_6	2

Tabela 7.9: Número de atributos para cada tipo de formato aproximado de distribuição

Do total de 23 atributos do conjunto de dados, a maior ocorrência de formatos aproximados foi do tipo T_1 , com 10 atributos e do tipo T_3 com oito atributos. Não houve ocorrência de atributos dos tipos T_4 e T_5 .

Na Tabela 7.10 é apresentada a quantidade de atributos de cada tipo de formato aproximado de distribuição que ocorreu nos atributos selecionados por cada um dos algoritmos de SA e no conjunto de dados original.

Tipo do Formato	Orig.	C4.5	ReliefF	CFS	FCBF	CBF	FDimBF(2)
Tipo 1	10	11	11	3	10	8	6
Tipo 2	3	2	2	0	2	2	1
Tipo 3	8	7	8	3	5	4	5
Tipo 4	0	0	0	0	0	0	0
Tipo 5	0	0	0	0	0	0	0
Tipo 6	2	0	0	0	0	0	0

Tabela 7.10: Número de atributos para cada tipo de formato aproximado de distribuição por subconjunto de atributos

É interessante observar que todos os algoritmos selecionaram atributos com formatos aproximados seguindo a tendência mostrada pelo conjunto original de dados: maioria de atributos dos tipos T_1 e T_3 . Os algoritmos CFS e FDimBF(2) apresentaram proporções mais equilibradas na escolha de atributos desses dois tipos, enquanto FCBF e CBF, embora tenham seguido a tendência geral, escolheram uma proporção maior de atributos do tipo T_1 quando comparados aos outros algoritmos. Essa tendência mostra resultados similares, para o algoritmo FDimBF, aos alcançados na avaliação experimental realizada utilizando conjuntos de dados naturais, na qual 10 do total de 11 conjuntos de dados apresentaram maioria dos atributos dos tipos T_1 e T_3 — Tabela 6.6 na página 76.

7.9.4 Performance dos Algoritmos em Relação à Precisão e a Quantidade de Atributos Seleccionados

O modelo de avaliação que considera a relação entre a precisão do modelo — Tabela 7.11 — e a redução da quantidade de atributos, introduzido na Seção 6.5.4 na página 76, foi também utilizado para avaliar os resultados deste estudo de caso — Figura 7.8.

	Orig.	C4.5	ReliefF	CFS	FCBF	CBF	FDimBF(2)
Média do Erro	28,70	27,80	29,00	24,80	26,00	26,80	29,20
Erro Padrão	2,20	1,90	1,70	1,40	2,20	2,20	1,60

Tabela 7.11: Média do erro e erro padrão para o conjunto de dados processamento de sêmen — ECM = 40,05%

Figura 7.8: Representação gráfica: relação entre percentagem de atributos seleccionados, média do erro e erro padrão dos modelos construídos; A: C4.5, B: ReliefF, C: CFS, D: FCBF, E: CBF, G: FDimBF(2) e T: sem SA.

Nesse gráfico é possível observar que todos os algoritmos, à exceção de ReliefF e FDimBF(2), foram classificados como de performance excelente. Os algoritmos ReliefF e FDimBF(2) foram classificados, segundo esse modelo de avaliação, como sendo de performance boa, tendo apresentado um erro médio um pouco acima do erro cometido utilizando todos os atributos, embora sem diferença estatisticamente significativa. Dentre os algoritmos de SA considerados, CFS foi o algoritmo que selecionou o menor número de atributos, 26,09%. Em seguida, o algoritmo FDimBF foi o que selecionou menos atributos, com 52,17% de atributos escolhidos do total de 23 atributos do conjunto original de dados.

É interessante ressaltar que os resultados obtidos também mostram a mesma tendência dos resultados apresentados no Capítulo 6, na qual, as duas versões do algoritmo FDimBF, juntamente com o algoritmo CFS, apresentaram as boa performance segundo esse modelo.

7.9.5 Avaliação dos Modelos Construídos

Diversas medidas têm sido propostas com o objetivo de auxiliar o usuário a entender e a utilizar o conhecimento construído por sistemas de aprendizado de máquina que induzem regras de decisão (Lavrač et al., 1999). Algumas dessas medidas, denominadas de medidas objetivas, baseadas na matriz de contingência com frequências relativas, foram aplicadas aos modelos construídos utilizando os atributos selecionados pelos algoritmos de seleção de atributos considerados neste estudo de caso. Além disso, esses modelos foram avaliados por especialistas do domínio para realizar a comparação do conhecimento prévio em relação ao conhecimento embutido nas regras.

Nas próximas seções são apresentadas as medidas de avaliação objetivas e as avaliações realizadas utilizando essas medidas, bem como a avaliação realizada pelos especialistas do domínio.

7.9.5.1 Medidas de Avaliação

Em problemas de classificação, usualmente utiliza-se a matriz de confusão, a qual referencia o classificador que representa o modelo ou hipótese induzida como um todo, como base para calcular medidas de avaliação, tais como precisão (ou erro). Quando são considerados classificadores simbólicos, é possível calcular medidas para avaliar as regras de decisão que constituem o modelo, de modo individual, utilizando a matriz de contingência de cada regra (Mitchell, 1997).

As regras de decisão são geralmente representadas como:

$$R : \underbrace{if < complexo >}_{Body\ ou\ B} then \underbrace{< classe = C_i >}_{Head\ ou\ H}$$

ou resumidamente $B \rightarrow H$, sendo C_i um dos possíveis valores para a classe e $< complexo >$ uma disjunção de conjunções de testes para os atributos da forma:

$$X_i\ op\ Valor$$

na qual X_i é o nome do atributo, op é um operador pertencente ao conjunto $\{=, \neq, <, \leq, >, \geq\}$ e $Valor$ é um valor válido para o atributo X_i .

A *cobertura* de uma regra $R : B \rightarrow H$ é definida como segue: exemplos que satisfazem B (o corpo da regra) compõem o conjunto de cobertura de R ; em outras palavras,

esses exemplos são cobertos por R . Uma *regra de classificação* assume a forma **if** B **then** classe = C_i , onde $C_i \in \{C_1, \dots, C_{N_{C_i}}\}$, ou seja, a cabeça H de uma regra de classificação é classe = C_i , com $C_i \in \{C_1, \dots, C_{N_{C_i}}\}$,

Dados uma regra R e um exemplo $E_i = (\vec{x}_i, y_i)$, sendo y_i sua respectiva classe, a regra pode ser aplicada ao exemplo e o resultado previsto pela cabeça H da regra comparado à classe verdadeira y_i do exemplo. Essa comparação resulta em quatro possíveis situações:

1. o exemplo é coberto corretamente pela regra, *i.e.*, B e H são ambos verdade;
2. o exemplo é incorretamente coberto pela regra, *i.e.*, B é verdade porém H falso;
3. o exemplo não é coberto pela regra, no entanto a classe prevista pela cabeça H da regra é a mesma classe do exemplo, *i.e.*, B é falso mas H é verdadeiro e
4. o exemplo não é coberto pela regra e a classe prevista pela cabeça H da regra não é a mesma classe do exemplo, *i.e.*, ambos B e H são falsos.

A matriz de contingência para uma determinada regra é construída aplicando a essa regra um conjunto de exemplos de teste T contendo n exemplos — Tabela 7.12. A matriz de contingência pode também ser representada em termos de frequências relativas — Tabela 7.13 — na qual os valores são divididos por n . Desse modo, a frequência relativa $\frac{\epsilon}{n}$, associada a um evento ϵ , pode ser utilizada como uma estimativa de probabilidade para o evento ϵ , denotada por $P(\epsilon)$.

	H	$\bar{H}$	
B	hb	$\bar{h}b$	b
$\bar{B}$	$h\bar{b}$	$\bar{h}\bar{b}$	$\bar{b}$
	h	$\bar{h}$	n

hb = número de exemplos para os quais H é verdade e B é verdade

$\bar{h}b$ = número de exemplos para os quais H é falso e B é verdade

$h\bar{b}$ = número de exemplos para os quais H é verdade e B é falso

$\bar{h}\bar{b}$ = número de exemplos para os quais H é falso e B é falso

b = número de exemplos para os quais B é verdade

$\bar{b}$ = número de exemplos para os quais B é falso

h = número de exemplos para os quais H é verdade

$\bar{h}$ = número de exemplos para os quais H é falso

n = número total de exemplos

Tabela 7.12: Matriz de contingência para uma regra

Diversas medidas objetivas para a avaliação de regras têm sido propostas na literatura. As medidas mais relevantes foram unificadas sob uma mesma visão por Lavrač et al. (1999). Nesse *framework*, as definições são dadas em termos de frequências relativas derivadas da tabela de contingência da regra. As medidas de avaliação de regras utilizadas neste trabalho são:

	H	$\bar{H}$	
B	f_{hb}	$f_{\bar{h}b}$	f_b
$\bar{B}$	$f_{h\bar{b}}$	$f_{\bar{h}\bar{b}}$	$f_{\bar{b}}$
	f_h	$f_{\bar{h}}$	1

Tabela 7.13: Matriz de contingência com frequências relativas para uma regra

Definição 7.9.1 Precisão (Accuracy): também denominada de consistência ou confiabilidade, é associada à quão específica uma regra é para o problema. Ela pode ser definida como a probabilidade condicional de H ser verdade dado que B é verdade — Equação 7.2. Quanto maior o valor da precisão, mais precisamente a classe em questão será coberta por essa regra.

$$Acc(B \rightarrow H) = P(H|B) = \frac{P(HB)}{P(B)} = \frac{f_{hb}}{f_b} \quad (7.2)$$

Definição 7.9.2 Sensitividade: também denominada de completeza ou recall, é associada ao número relativo de exemplos da classe prevista em H cobertos pela regra. Pode ser definida como a probabilidade condicional de B ser verdade dado que H é verdade — Equação 7.3. Quanto maior o valor dessa medida, mais exemplos serão cobertos por essa regra.

$$Sens(B \rightarrow H) = P(B|H) = \frac{P(HB)}{P(H)} = \frac{f_{hb}}{f_h} \quad (7.3)$$

Definição 7.9.3 Suporte: também denominado de frequência, é associado ao número relativo de exemplos cobertos corretamente pela regra. Pode ser definido como a probabilidade de H e B serem verdade — Equação 7.4. Quanto maior o valor dessa medida, maior será o número de exemplos da classe em questão cobertos corretamente pela regra.

$$Sup(B \rightarrow H) = P(HB) = f_{hb} \quad (7.4)$$

Definição 7.9.4 Novidade: é associada à indicação da existência de algo interessante ou fora do comum na regra. Ela pode ser definida como se a probabilidade de H e B ocorrerem juntos não puder ser inferida pelas probabilidades de H e B isoladamente, i.e., H e B não são estatisticamente independentes. Essa medida é obtida comparando-se o valor esperado de $P(HB)$ com os valores de $P(H)$ e $P(B)$ — Equação 7.5. Desse modo, quanto mais o valor esperado diferir do observado maior será a probabilidade de existir uma correlação verdadeira e inesperada entre H e B .

$$Nov(B \rightarrow H) = P(HB) - P(H)P(B) = f_{hb} - f_h \cdot f_b \quad (7.5)$$

As medidas de precisão, sensitividade e suporte variam de 0 a 1. Já a medida de novidade varia entre $-0,25 \leq Nov(B \rightarrow H) \leq 0,25$. Quanto mais próximo de 0,25, mais intensa é associação entre H e B , ao passo que quanto menor o valor, i.e., mais próximo de -0,25, mais intensa é a associação entre $\bar{H}$ e B .

Além dessas medidas de avaliação de regras, o número de condições da regra também foi considerada como uma medida associada à complexidade sintática da respectiva regra.

7.9.5.2 Avaliação dos Modelos Utilizando Medidas Objetivas de Regras

Para cada um dos modelos construídos utilizando os subconjuntos de atributos selecionados pelos seis algoritmos de SA — $\mathcal{C}4.5$, ReliefF, CFS, FCBF, CBF e FDimBF(2) — além do modelo construído considerando o conjunto original de atributos, foram calculadas as quatro medidas objetivas de avaliação de regras anteriormente descritas. O cálculo dessas medidas foi realizado por meio de um conjunto de ferramentas disponibilizadas no projeto DISCOVER (Prati et al., 2001a).

Na Tabela 7.14 é apresentado um resumo dos resultados dessas medidas para o conjunto de regras que constituem cada um dos modelos construídos. Na primeira coluna é indicado o algoritmo utilizado para selecionar os subconjuntos de atributos, juntamente com o conjunto original de atributos, e nas quatro colunas seguintes são apresentadas as médias das medidas de precisão (Acc), sensibilidade (Sens), suporte (Sup) e novidade (Nov)¹⁰. Nas duas últimas colunas são mostrados o número médio de condições das regras para cada modelo e o número de regras desse modelo, *i.e.*, a sua complexidade sintática. Os valores individuais dessas medidas para cada regra dos modelos construídos são apresentados no Apêndice B.

Conjunto de Atributos	Média das Medidas de Avaliação de Regra				Complexidade Sintática	
	Acc	Sens	Sup	Nov	#Condições	#Regras
Orig.	$0,88 \pm 0,05$	$0,29 \pm 0,08$	$0,10 \pm 0,04$	$0,05 \pm 0,01$	$4,25 \pm 0,67$	16
$\mathcal{C}4.5$	$0,94 \pm 0,02$	$0,27 \pm 0,08$	$0,10 \pm 0,04$	$0,05 \pm 0,01$	$4,41 \pm 0,69$	17
ReliefF	$0,89 \pm 0,06$	$0,19 \pm 0,06$	$0,07 \pm 0,02$	$0,03 \pm 0,01$	$4,24 \pm 0,65$	25
CFS	$0,84 \pm 0,06$	$0,38 \pm 0,08$	$0,15 \pm 0,04$	$0,06 \pm 0,01$	$2,75 \pm 0,38$	12
FCBF	$0,84 \pm 0,06$	$0,33 \pm 0,10$	$0,10 \pm 0,05$	$0,05 \pm 0,01$	$3,23 \pm 0,45$	13
CBF	$0,78 \pm 0,06$	$0,25 \pm 0,09$	$0,08 \pm 0,04$	$0,04 \pm 0,01$	$3,07 \pm 0,49$	15
FDimBF(2)	$0,80 \pm 0,06$	$0,25 \pm 0,09$	$0,09 \pm 0,04$	$0,03 \pm 0,01$	$3,29 \pm 0,60$	17

Tabela 7.14: Avaliação dos modelos utilizando medidas objetivas

Pode ser observado que, à exceção do modelo construído utilizando os atributos selecionados pelo algoritmo CBF, todos os outros seis algoritmos, incluindo o modelo constituído utilizando todos os atributos, apresentaram modelos com média da precisão das regras individuais acima de 80,00%. Especificamente, o modelo construído utilizando os atributos escolhidos por $\mathcal{C}4.5$ foi o que apresentou maior número de regras com precisão maior que 80,00%, 15 do total de 17 regras. Quanto à medida de sensibilidade, todos os modelos apresentaram regras com valores acima de 0,20, *i.e.*, que cobrem mais de 20,00% de exemplos da classe prevista. Em relação ao suporte do conjunto das regras que

¹⁰Ainda que as medidas objetivas de avaliação de regras sejam importantes para avaliar cada regra individualmente, a média e o erro padrão dessas medidas para o conjunto de regras que constituem cada classificador (modelo) gerado, permite ter uma visão geral da qualidade dessas regras.

compõem os modelos considerados, apenas os subconjuntos selecionados por $\mathcal{C}4.5$, CFS e FCBF apresentaram médias superiores à 10,00%, embora os outros tenham apresentado médias variando de 6,00% a 9,00%, o que mostra que as regras que constituem os modelos cobriram corretamente, em média, no mínimo 6,00% dos exemplos das classes correspondentes, até 15,00% para CFS. Para a medida de novidade, as médias apresentadas variam de 0,03 a 0,06, indicando que há uma associação, mas pequena, entre as condições das regras e as classes consideradas¹¹.

Os modelos foram analisados também em relação ao número de condições que as regras apresentam. Essa medida pode fornecer um indicativo da complexidade sintática das regras, ou seja, quanto maior o número de condições maior a complexidade sintática da regra. Nos modelos em análise, as médias dos números de condições do conjunto de regras que constituem cada modelo foram relativamente uniformes. Três modelos — conjunto original de atributos, $\mathcal{C}4.5$ e ReliefF — apresentaram média do número de condições acima de quatro. Outros três algoritmos — FCBF, CBF e FDimBF(2) — apresentaram média entre 3,00 a 3,30. O algoritmo CFS apresentou a menor complexidade sintática do conjunto de regras, uma média de 2,75 condições por regra. Esse algoritmo também foi o que apresentou o modelo composto pelo menor número de regras, enquanto o algoritmo ReliefF foi o que apresentou o modelo com o maior número de regras, 25.

7.9.5.3 Avaliação dos Modelos por Especialistas do Domínio

Os modelos construídos a partir dos subconjuntos de atributos selecionados pelos algoritmos de SA, juntamente com o modelo considerando o conjunto original de atributos, bem como as regras que constituem esses modelos, foram também avaliados por especialistas do domínio.

Para a realização dessa avaliação foi utilizado um Índice de Avaliação do Especialista — IAE — que classifica cada regra nas seguintes cinco categorias:

- Ruim (0);
- Mediano Baixo (1);
- Mediano (2);
- Bom (3) e
- Muito Bom (4).

Para situar cada regra dentro dessa classificação, inicialmente, os especialistas avaliaram se os atributos e os valores dos atributos que compunham cada regra estavam de

¹¹Deve ser observado que a medida de novidade é uma medida objetiva. Em outras palavras, o especialista pode considerar que uma regra com baixo valor dessa medida contém conhecimento novo e, reciprocamente, o especialista pode considerar que uma regra com valores próximos a -0,25 ou a 0,25 não apresenta conhecimento novo.

acordo com as respectivas classes dadas por essas regras, independentemente das medidas objetivas dessas regras. Por exemplo, considere a seguinte regra:

```
SE concentracao_total > 162,75
  E motilidade > 48
  ENTÃO classe = 3 [152/3; 0,974]
```

Nessa regra¹², é considerado que se a concentração total de espermatozóides na amostra recolhida é maior que 162,75 milhões e a motilidade, composta pelas motilidades de grau A, B e C, é maior que 48,00%, o caso será classificado na classe 3, isto é, com possibilidade de se utilizar técnicas de reprodução assistida de menor complexidade como a IUI. Essa regra foi classificada como de boa qualidade, já que é coerente com o conhecimento do domínio, pois pelos parâmetros seminais da Organização Mundial da Saúde — OMS —, para o atributo concentração total de espermatozóides considera-se como normal o valor de 40 milhões de espermatozóides, e para o atributo motilidade de grau A adicionado ao B considera-se como normal um valor maior que 50,00%, *i.e.*, presença de 50,00% ou mais de espermatozóides de motilidade progressiva.

Na Tabela 7.15 é apresentado um resumo da avaliação dos especialistas¹³ contendo:

- número de regras e respectiva percentagem de cada categoria do IAE para cada modelo contruído utilizando os subconjuntos de atributos selecionados pelos algoritmos de SA, incluindo o conjunto original, sem SA, de atributos;
- número total de regras e média do IAE das regras para cada modelo, calculada do seguinte modo:

$$\text{Média IAE} = \sum_{i=0}^4 \frac{\text{IAE}_i \cdot n_{R_{\text{IAE}_i}}}{n_R}$$

sendo IAE_i = índice i de avaliação dos especialistas, $n_{R_{\text{IAE}_i}}$ = número de regras que foram classificadas como do índice i de avaliação dos especialistas e n_R = número total de regras do modelo e

- número total de regras para cada categoria do IAE na última coluna.

Dos sete modelos considerados para a avaliação, apenas o modelo construído utilizando os atributos selecionados por FCBF apresentou mais de 30,00% das regras classificadas como ruins (38,46%). Quanto à quantidade de regras medianas baixas, os modelos que mais apresentaram esse tipo de regras foram $\mathcal{C}4.5$ e $\text{FDimBF}(2)$, ambos com 29,41%. Em contraste, FCBF não apresentou nenhuma regra dessa categoria, porém apresentou

¹²Como mencionado anteriormente, os números entre colchetes indicam quantos exemplos a regra cobre (corretos/errados) e o grau de confiança dessa regra.

¹³Os valores individuais do IAE para cada regra dos modelos são apresentados no Apêndice B.

IAE	Orig.	C4.5	ReliefF	CFS	FCBF	CBF	FDimBF(2)	Total
0	3	3	2	3	5	2	4	22
	18,75	17,65	8,00	25,00	38,46	13,34	23,53	19,13
1	4	5	2	1	0	3	5	20
	25,00	29,41	8,00	8,33	0,00	20,00	29,41	17,39
2	2	2	5	2	5	5	4	25
	12,50	11,76	20,00	16,67	38,46	33,33	23,53	21,74
3	7	6	12	5	3	5	4	42
	43,75	35,29	48,00	41,67	23,08	33,33	23,53	36,52
4	0	1	4	1	0	0	0	6
	0,00	5,88	16,00	8,33	0,00	0,00	0,00	5,22
# Regras	16	17	25	12	13	15	17	115
Média IAE	1,81	1,82	2,56	2,00	1,46	1,87	1,47	1,91

Tabela 7.15: Avaliação das regras dos modelos pelos especialistas do domínio

a maior proporção de regras de qualidade mediana, segundo a avaliação dos especialistas. Logo após, está o algoritmo CBF, o qual apresentou 33,33% do total das regras que compõem o modelo classificadas como de qualidade mediana. O modelo que mais apresentou, proporcionalmente, o maior número de regras de boa qualidade foi o que utilizou os atributos selecionados com ReliefF (48,00%). Os outros modelos apresentaram proporções que variam desde 23,08% (FCBF) a 43,75% (sem SA) de regras de boa qualidade. Quanto às regras classificadas como de muito boa qualidade pelos especialistas, apenas os algoritmos C4.5, ReliefF e CFS apresentaram regras dessa categoria.

Em geral, o algoritmo ReliefF foi o que apresentou maior proporção de regras consideradas boas e muito boas. Esse algoritmo foi o que apresentou também a maior média do IAE (2,56), porém, é importante notar que esse modelo é constituído por 25 regras, o que resultou em um modelo 36,00% maior que o modelo construído utilizando todos os atributos originais do conjunto de dados de processamento de sêmen. Em comparação, o algoritmo CFS, apresentou a segunda maior média do IAE (2,00) e o menor modelo contendo apenas 12 regras, *i.e.*, menos da metade do número de regras que ReliefF. É importante considerar também que ReliefF apresentou uma das maiores médias de número de condições das regras enquanto CFS apresentou o menor número de condições, portanto, apresentando regras menos complexas — Tabela 7.14. Deve ser observado que modelos compostos por muitas regras e que usam muitos atributos indicam que foi construído um modelo com alta complexidade sintática.

Uma outra avaliação realizada foi a análise conjunta dos resultados das medidas objetivas de regras com a medida subjetiva do índice de avaliação dos especialistas. Para tanto, foram consideradas apenas as regras que apresentaram precisão acima de 80,00% e sensibilidade maior que 20,00%. Isso significa que, nessa avaliação, foram consideradas apenas as regras que classificaram corretamente pelo menos 80,00% dos exemplos por ela cobertos e que classificaram corretamente um mínimo de 20,00% dos exemplos da classe tratada pela regra.

Para o modelo construído com o conjunto original de atributos, seis regras atingiram

ambos os requisitos descritos acima. Dessas seis regras, três foram classificadas como regras de boa qualidade segundo a avaliação dos especialistas, duas de qualidade mediana baixa e uma como sendo ruim. As regras classificadas como boas apresentaram, de um modo geral, conhecimento coerente com o domínio. As regras de qualidade mediana baixa, foram assim classificadas pois apresentaram atributos com valores de teste que poderiam gerar dúvida, do ponto de vista dos especialistas, em relação à classe prevista pela regra. A única regra classificada como ruim dentro dos requisitos acima citados, foi a seguinte:

```
SE motilidade_grau_b <= 24
  E motilidade <= 48
  ENTÃO classe = 1 [36/3; 0,895]
```

pois foi considerado que os limiares usados para esses dois atributos não são suficientemente baixos para assegurar que os exemplos classificados por essa regra estariam na classe 1.

Os modelos construídos a partir dos atributos selecionados por *C4.5* e *ReliefF* apresentaram mais de 80,00% das regras similares às obtidas no modelo construído utilizando todos os atributos. As regras desses modelos que satisfazem os requisitos estabelecidos para essa análise, a qual combina precisão e sensibilidade, foram as mesmas avaliadas no modelo construído utilizando todos os atributos. Isso deve-se ao fato desses dois algoritmos terem selecionado 20 e 21 do total de 23 atributos do conjunto original de dados, respectivamente. Um exemplo de uma regra classificada como de boa qualidade é:

```
SE concentracao_total > 162,75
  E motilidade > 48
  ENTÃO classe = 3 [152/3; 0,974]
```

pois se a concentração total de espermatozóides (em milhões) é bastante alta (valor normal segundo a OMS maior ou igual a 40 milhões) e a motilidade está próxima ao limiar normal (parâmetro considerado normal pela OMS para a motilidade progressiva igual a 50), o exemplo será classificado como da classe 3, *i.e.*, haverá uma quantidade bastante alta de espermatozóides, o que permitirá a aplicação de uma técnica de reprodução assistida de menor complexidade, como a IUI.

Para o modelo construído utilizando o subconjunto de atributos selecionado por *CFS*, também houve seis regras dentro das especificações de precisão e sensibilidade consideradas para escolher as regras a serem avaliadas quanto à comparação entre os resultados das medidas objetivas e subjetiva. É interessante notar que essas seis regras apresentaram valores de sensibilidade bastante altos, apenas uma delas com 27,00% e as demais com valores superiores a 45,00%. Isso mostra que essas regras cobriram corretamente um grande número de exemplos das classes respectivas. Quanto ao IAE, duas regras foram classificadas segundo esse índice como regras ruins. As quatro regras restantes foram consideradas como pertencentes uma a cada categoria do IAE — mediana, mediana

baixa, boa e muito boa. É interessante notar que a regra considerada muito boa pelos especialistas é composta apenas por uma condição:

```
SE concentracao_total > 162,75
  ENTÃO classe = 3 [155/5; 0,962]
```

Esse conhecimento foi considerado interessante, pois parece estabelecer um limiar bem acima do considerado normal e sem considerar outros atributos para a determinação da classe 3. Outra questão interessante é de que essa regra obteve um valor de 0,14 para a medida de novidade, o que mostra uma sintonia da medida objetiva de novidade com a opinião dos especialistas. Essa regra também apresentou altos valores para $Acc = 0,97$, $Sens = 0,61$ e $Sup = 0,37$. Esse modelo, construído utilizando o subconjunto de atributos selecionado por CFS, não apresentou regras iguais às encontradas no modelo construído utilizando todos os atributos originais do conjunto de dados.

Cinco regras foram selecionadas, seguindo os limiares de precisão e sensibilidade estabelecidos, no modelo construído utilizando os atributos selecionados por FCBF. Dessas cinco regras, uma foi classificada pelos especialistas como ruim, três foram consideradas de qualidade mediana e uma foi classificada como uma boa regra. Por exemplo, a seguinte regra, considerada de boa qualidade:

```
SE cor = translucido
  E concentracao <= 4,6
  E morfologia_Kruger < 6
  ENTÃO classe = 1 [33/2; 0,914]
```

classifica exemplos que apresentam concentração por mL menor ou igual a 4,6, muito abaixo do valor considerado normal segundo a OMS que é igual a 20 milhões de espermatozóides por mL, e menos de 6,00% de espermatozóides de morfologia normal (valor de referência segundo a OMS de 14%), como sendo da classe 1. Isso significa que casos com essas características devem ser tratados por meio de métodos mais complexos de reprodução assistida, como ICSI.

No modelo construído com os atributos selecionados pelo algoritmo CBF, apenas duas regras apresentaram precisão e sensibilidade maiores que 80,00% e 20,00%, respectivamente. Uma dessas regras foi considerada de qualidade mediana baixa e a outra foi classificada como de boa qualidade pelos especialistas. Ambas as regras são compostas por duas condições, tendo o atributo *motilidade* em comum.

No modelo construído utilizando os atributos escolhidos por FDimBF(2), também apenas duas regras apresentaram os limiares de precisão e sensibilidade estabelecidos. Essas duas regras foram classificadas pelos especialistas como de qualidade mediana e mediana baixa. Ambas apresentaram três condições sendo uma regra da classe 1 e a

outra da classe 3. Ainda que essas duas regras, listadas a seguir, tenham sido consideradas de não muito boa qualidade, de acordo com os especialistas elas apresentaram alguns aspectos interessantes, como os valores dos atributos `motilidade-grau-d` e `motilidade-progressiva`:

```
SE motilidade_grau_d > 36
  E motilidade_progressiva <= 34
  E nro_celulas <= 3.04
  ENTÃO classe = 1 [61/8; 0,857]

SE cor = branco_opalescente
  E motilidade_grau_d <= 49
  E motilidade_progressiva > 34
  ENTÃO classe = 3 [219/28; 0,869]
```

Essas regras foram consideradas de pouca qualidade, pois os limiares utilizados pelos atributos para realizar a decisão não foram suficientemente discriminativos para evitar, do ponto de vista dos especialistas, dúvidas quanto a qual classe um novo exemplo classificado por essas regras seria atribuído. Ainda assim, segundo os especialistas, é possível identificar questões interessantes nessas duas regras, nas quais os valores considerados dos atributos `motilidade-grau-d` e `motilidade-progressiva` criam uma separação coerente para as classes 1 e 3.

Deve ser observado que na avaliação inicial dos especialistas, *i.e.*, sem restrições dos limiares para os valores das medidas de avaliação de regras — Tabela 7.15 na página 122 — os especialistas indicaram um total de 11 regras interessantes¹⁴. Porém, considerando também os limiares estipulados para as medidas de precisão ($\text{Acc} \geq 0,80$) e sensibilidade ($\text{Sens} \geq 0,20$), apenas uma regra alcançou os requisitos estabelecidos para as medidas objetivas e foi considerada ao mesmo tempo interessante pelos especialistas. Essa regra, apresentada durante a apreciação conjunta de medidas objetivas e subjetiva para a avaliação de regras, pertence ao modelo construído a partir do subconjunto de atributos selecionados pelo algoritmo CFS.

7.10 Considerações Finais

Neste capítulo foi apresentado um estudo de caso no qual foi utilizado um conjunto de dados reais da área médica. Uma particularidade desses dados é a sua representação original na forma de laudos médicos semi-estruturados. Na área médica há uma grande parcela de dados, a partir dos quais potencialmente poderia ser extraído conhecimento, os

¹⁴Somente o modelo construído a partir dos atributos selecionados pelo algoritmo CBF não apresentou regras desse tipo.

quais estão na forma de laudos descritos em parte em linguagem natural. Para que esses dados possam ser analisados por meio de processos como a extração de conhecimento de bases de dados, é necessário que eles estejam representados em formatos apropriados, tal como o formato atributo-valor.

Para a realização desse estudo de caso, os laudos de processamento de sêmen foram mapeados para um conjunto de dados no formato atributo-valor por meio de uma metodologia proposta e implementada neste trabalho, a qual também foi apresentada neste capítulo. Essa metodologia, com o auxílio de especialistas, permite que seja primeiramente criado um dicionário, o qual é posteriormente utilizado para transcrever as informações contidas nos laudos para um conjunto de dados no formato atributo-valor. Uma vez construído o dicionário, novos exemplos podem ser automaticamente mapeados para esse formato.

Foi também realizada uma avaliação experimental semelhante à realizada no Capítulo 6, com o auxílio de especialistas do domínio médico, para verificar se o conhecimento gerado apresenta coerência com o conhecimento prévio e se pode ser considerado interessante e/ou novo. Os resultados obtidos mostram que é importante a avaliação experimental dos algoritmos de seleção de atributos, como mencionado no capítulo anterior, pois cada algoritmo pode mostrar-se mais apropriado dependendo do conjunto de dados considerado. A análise dos resultados indicou algumas questões interessantes, como uma possível relação entre os atributos `motilidade-grau-a` e `motilidade-grau-d`, as quais poderão ser investigadas no futuro.

No próximo capítulo são apresentadas as principais contribuições deste trabalho, bem como as limitações encontradas e os trabalhos futuros.

Capítulo 8

Conclusão

A seleção de atributos constitui uma tarefa de fundamental importância dentro do processo de extração de conhecimento a partir de bases de dados, o qual é conduzido com o apoio de diversas áreas, entre as quais a área de aprendizado de máquina. A seleção de atributos é usualmente realizada como uma etapa de pré-processamento com o objetivo de selecionar os atributos mais importantes para a posterior construção de modelos que permitam explicitar o conhecimento embutido nos dados. Quando o objetivo é a descoberta de conhecimento a partir de dados, a existência de atributos irrelevantes e/ou redundantes pode comprometer a compreensibilidade das hipóteses (modelos) induzidas por algoritmos de aprendizado supervisionado, além de poder reduzir a precisão dos modelos construídos.

Por várias décadas, a seleção de atributos tem sido foco constante de pesquisa em diversas áreas, entre as quais aprendizado de máquina e mineração de dados. Diversos algoritmos foram propostos na literatura para tratar o problema de seleção de atributos, porém, a maioria desses algoritmos concentram seus esforços na procura por atributos relevantes. Todavia, trabalhos recentes têm mostrado que o tratamento de atributos redundantes é também importante, pois, juntamente com os atributos irrelevantes, podem afetar a qualidade dos dados e, conseqüentemente, a qualidade das hipóteses induzidas por algoritmos de aprendizado. Desse modo, é necessário definir critérios de importância para a seleção de atributos relevantes e não redundantes.

Neste trabalho foi realizada uma minuciosa pesquisa bibliográfica com o objetivo de coletar e unificar diversas definições de importância de atributos encontradas na literatura. Essas definições foram organizadas de acordo com o tipo de medida utilizada para determinar a importância de um atributo. Entre os vários critérios de importância de atributos propostos na literatura, alguns estão baseados em medidas de distância, consistência ou informação, enquanto outros são fundamentados em medidas de dependência — Capítulo 4.

Neste trabalho foi também proposto um algoritmo, baseado no modelo que desassocia em duas etapas a procura por atributos importantes, *i.e.*, as análises de relevância

e redundância são realizadas em duas etapas distintas — Capítulo 5. Esse modelo foi proposto recentemente (Yu and Liu, 2004) e permite que o alto custo computacional necessário para a busca por subconjuntos de atributos, a qual constitui a abordagem usual para a seleção de atributos relevantes e não redundantes, seja diminuída. No algoritmo FDimBF proposto neste trabalho, é introduzida a utilização da dimensão fractal como critério para análise de redundância dos atributos (Lee et al., 2005d; Lee and Monard, 2003). Ressaltamos que não é de nosso conhecimento a utilização da dimensão fractal na área de mineração de dados quando algoritmos de aprendizado de máquina supervisionado são usados para a indução de hipóteses. Para isso, foram implementadas duas versões do algoritmo FDimBF utilizando medidas de informação e distância para a análise de relevância, e a dimensão fractal para a análise de redundância. Nossa proposta foi analisada experimentalmente utilizando diversos conjuntos de dados, freqüentemente utilizados na literatura, para avaliação empírica do algoritmo FDimBF e comparada a diversos algoritmos consolidados na área de seleção de atributos (Lee et al., 2005b,c,e). Os resultados mostram que o algoritmo FDimBF proposto é competitivo com os outros algoritmos de seleção de atributos considerados — $\mathcal{C}4.5$ (Quinlan, 1993), ReliefF (Kononenko, 1994), CFS (Hall, 2000), FCBF (Yu and Liu, 2004) e CBF (Liu and Setiono, 1996). Em geral, as duas abordagens de FDimBF apresentaram performance similar tanto em relação ao número de atributos selecionados quanto em relação à precisão dos modelos construídos a partir dos subconjuntos de atributos escolhidos. Essas duas versões do algoritmo FDimBF, juntamente com o algoritmo CFS (Hall, 2000), foram os que, em uma visão global, apresentaram os melhores resultados experimentais para os conjuntos de dados considerados.

A avaliação de performance de algoritmos de seleção de atributos constitui um tema essencial, pois ainda não existe uma análise formal capaz de determinar, previamente, a superioridade de um algoritmo de seleção de atributos em relação a outro (Dietterich, 1989; Langley, 1988). Esse problema constitui uma questão complexa e é influenciada por diversos fatores, tais como o domínio da aplicação e os algoritmos que serão utilizados, posteriormente, para construir os modelos. Usualmente, os algoritmos de seleção de atributos são avaliados quanto à precisão das hipóteses induzidas pelos algoritmos de aprendizado considerando os subconjuntos de atributos por eles selecionados. Todavia, para um determinado problema em questão, somente a avaliação de um parâmetro não é capaz de fornecer subsídios suficientes para se determinar qual algoritmo apresenta melhor desempenho quando comparado a outros. Assim, é necessário incluir outros parâmetros nessa avaliação. A redução da quantidade de atributos, proporcionada pelo processo de seleção de atributos, é uma questão diretamente ligada ao propósito fundamental dessa tarefa. Desse modo, essa redução também constitui um parâmetro importante para a avaliação da performance de algoritmos de seleção de atributos.

Neste trabalho é proposto um modelo de avaliação que combina a precisão da hipótese induzida usando o subconjunto de atributos selecionado pelo algoritmo de seleção

de atributos com a redução obtida nesse subconjunto em relação ao conjunto original de atributos — Seção 6.5.4 na página 76. Esse modelo foi também utilizado para avaliar o desempenho do algoritmo FDimBF e dos outros algoritmos de seleção de atributos considerados na avaliação experimental realizada neste trabalho (Lee et al., 2005b,c,e). Nesse modelo, o desempenho combinado dos algoritmos é classificado dentro de cinco categorias, permitindo uma análise da performance do algoritmo em relação ao modelo construído usando todos os atributos do conjunto original de atributos e em relação ao erro da classe majoritária. A aplicação desse modelo proporciona a obtenção de uma visão geral do desempenho dos algoritmos em diversos conjuntos de dados considerando não apenas a precisão dos modelos, mas também, a redução da quantidade de atributos realizada pelos algoritmos de seleção de atributos.

Uma outra contribuição deste trabalho é a proposta e a implementação de uma metodologia para auxiliar no mapeamento de laudos médicos em conjuntos de dados no formato atributo-valor (Honorato et al., 2005a, 2004b) — Seção 7.5 na página 96. Frequentemente, laudos médicos apresentam informações organizadas de forma semi-estruturada, e parte dessas informações estão descritas em linguagem natural. O mapeamento das informações contidas nesses documentos é, usualmente, realizado de modo manual por meio da interpretação de cada laudo e a transcrição das informações para os atributos do conjunto de dados que os representam. Esse procedimento é bastante custoso e pode ficar sujeito a variações devidas à subjetividade dos que estão realizando o processo de transcrição. Na metodologia proposta, na qual é necessária uma intensa participação de especialistas do domínio, é construído um dicionário que permite o mapeamento das informações contidas nos laudos para o conjunto de dados. Após a construção desse dicionário, novos laudos podem ser automaticamente transcritos para um conjunto de dados no formato atributo-valor com resultados altamente satisfatórios. Essa metodologia foi também por nós aplicada com sucesso a outros domínios médicos (Honorato et al., 2005c, 2004a,c). Neste trabalho é apresentada a aplicação dessa metodologia em um caso real, no qual laudos contendo informações de processamento de sêmen foram mapeados para um conjunto de dados analisado no estudo de caso (Lee et al., 2005a).

O estudo de caso é apresentado no último capítulo deste trabalho, conjugando todas as contribuições desta tese — Capítulo 7. Esse estudo de caso foi desenvolvido como parte dos projetos de Computação Aplicada à Medicina (Wu et al., 2006, 2005a,b, 2004; Voltolini et al., 2003; Verza Junior et al., 2003; Machado et al., 2002) e de Análise Inteligente de Dados (Honorato et al., 2005b; Monard and Lee, 2003; Ferro et al., 2002; Esteves et al., 2001), desenvolvidos em uma parceria entre o Laboratório de Bioinformática da Universidade Estadual do Oeste do Paraná, o Laboratório de Inteligência Computacional da Universidade de São Paulo em São Carlos, o Serviço de Coloproctologia da Faculdade de Ciências Médicas da Universidade Estadual de Campinas e o Centro de Referência em Infertilidade Masculina. O algoritmo FDimBF proposto, juntamente com os outros algoritmos de seleção de atributos utilizados durante a avaliação experimental sobre con-

juntos de dados naturais, foram aplicados nesse estudo de caso. Esse conjunto de dados reais, como mencionado anteriormente, foi construído a partir de laudos médicos, os quais foram mapeados para o formato atributo-valor por meio da metodologia proposta neste trabalho. Os resultados desse estudo de caso, organizado do mesmo modo que a avaliação experimental com conjuntos de dados naturais, foram também avaliados usando medidas objetivas de avaliação de regras. A utilização dessas medidas tem como objetivo permitir que as regras que compõem as hipóteses induzidas pelos algoritmos de aprendizado usando os subconjuntos de atributos selecionados por FDimBF e pelos outros algoritmos de seleção de atributos, sejam avaliadas objetivamente por meio de conceitos como precisão, sensibilidade, suporte e novidade. Além disso, é também de fundamental importância a avaliação dessas hipóteses por especialistas do domínio.

As medidas objetivas provêm um importante instrumento para avaliação de regras que compõem as hipóteses induzidas e podem ser utilizadas para, por exemplo, realizar uma pré-seleção das regras a serem avaliadas por especialistas em casos nos quais o número de regras é elevado (Bernardini et al., 2003). Neste trabalho, com relação à avaliação dos especialistas, foi considerada uma medida, denominada de índice de avaliação do especialista, como uma medida subjetiva na qual os especialistas classificam as regras de acordo com seu conhecimento prévio do domínio. Nesse estudo de caso, primeiramente foram avaliadas as medidas objetivas e a medida subjetiva em separado e, posteriormente, em conjunto. Os resultados dessa avaliação mostraram que de fato a análise de especialistas do domínio é essencial quando o objetivo é a extração de conhecimento a partir de dados. Ainda que as medidas objetivas de regras possam fornecer um indicativo, para determinar se uma regra constitui conhecimento novo e/ou interessante, é necessário o confronto direto com o conhecimento preliminar dos especialistas do domínio. Nesse estudo de caso, foram identificadas várias regras consideradas interessantes pelos especialistas do domínio, porém, apenas uma pequena parcela dessas regras também foi apontada pelas medidas objetivas como apresentando confiança suficiente para serem consideradas. Ainda assim, diversas questões ligadas à área do domínio foram identificadas e constituem objetos de trabalhos futuros.

Quanto à performance dos algoritmos, o que em geral apresentou melhores resultados foi o algoritmo CFS, embora não tenha ocorrido diferença estatisticamente significativa com 95% de significância, quanto à precisão das hipóteses induzidas usando os subconjuntos de atributos selecionados por todos os algoritmos de seleção de atributos considerados. Em relação ao número de atributos selecionados, CFS foi o que escolheu o menor número de atributos, seguido pelo algoritmo FDimBF. Em relação às avaliações utilizando medidas objetivas e a medida subjetiva índice de avaliação do especialista, o modelo construído utilizando os atributos selecionados por CFS foi o que apresentou as melhores regras.

Desse modo, considerando os resultados provenientes das avaliações experimentais usando os conjuntos de dados naturais e o conjunto de dados reais, podemos concluir

que o algoritmo FDimBF, o qual realiza as análises de relevância e redundância separadamente, pode ser uma boa alternativa para a seleção de atributos importantes considerando a dimensão fractal como critério para selecionar atributos não redundantes na área de aprendizado de máquina, na qual não é de nosso conhecimento que ela tenha sido utilizada.

8.1 Limitações

Como mencionado, alguns dos algoritmos de seleção de atributos tratam, indiretamente, atributos numéricos, pois lidam apenas com atributos nominais, discretizando previamente qualquer atributo numérico, *i.e.*, antes de realizar a SA. Esse é o caso de dois — CFS e FCBF — dos cinco algoritmos de seleção de atributos considerados neste trabalho. Em outras palavras, esses algoritmos tratam, internamente, apenas de atributos nominais. Embora a discretização constitua um processo importante de transformação de atributos, quando aplicado ao contexto de seleção de atributos, o subconjunto de atributos selecionados é fornecido de acordo com os atributos numéricos discretizados. Diferentes algoritmos de discretização de atributos realizam essa transformação nos dados de modos distintos. No entanto, o usuário, mesmo conhecendo o algoritmo interno para discretização utilizado pelos algoritmos de SA, não possui informação sobre os parâmetros usados para a realização desse processo de discretização. Ainda assim, o modelo somente pode ser construído, posteriormente, usando os atributos nominais e os atributos numéricos sem discretização.

Contrastando essa questão, o algoritmo FDimBF proposto neste trabalho, trata efetivamente os atributos numéricos na fase de análise de redundância, *i.e.*, sem que haja a necessidade de discretizá-los. Desse modo, os atributos utilizados para a posterior construção do modelo são os mesmos considerados pelo algoritmo para a seleção de atributos, em outras palavras, esses atributos não sofrem um processo de transformação. Todavia, o algoritmo FDimBF apresenta a restrição de não ser capaz de realizar a análise de redundância para atributos nominais, já que a medida de dimensão fractal pode ser calculada somente para atributos numéricos. Assim, o algoritmo FDimBF seleciona atributos nominais relevantes e atributos numéricos relevantes e não redundantes.

Não obstante essa restrição da análise de redundância não ser realizada para atributos nominais, os resultados obtidos durante a avaliação experimental conduzida neste trabalho mostram que o algoritmo FDimBF é competitivo com outros algoritmos consolidados e frequentemente citados na literatura, como os algoritmos ReliefF, CFS e FCBF. Embora os modelos construídos utilizando os subconjuntos de atributos selecionados pelos algoritmos de SA considerados neste trabalho tenham apresentado precisões estatisticamente similares, em termos da redução do número de atributos selecionados, o algoritmo FDimBF, juntamente com o algoritmo CFS, obtiveram os melhores resultados.

Especificamente em relação ao algoritmo FDimBF, existem outras duas questões

que devem ser consideradas. A primeira refere-se ao método usado para o cálculo da dimensão fractal. Como mencionado, conjuntos de dados reais podem exibir características de fractais estatisticamente auto-similares, os quais não apresentam regras de construção explícitas, sendo necessário aplicar métodos como o *Box Count Plot* (Faloutsos and Kamel, 1994). Esses métodos provêem, teoricamente, uma boa aproximação da dimensão fractal do conjunto de dados (Schroeder, 1991). Existem vários algoritmos propostos para implementar esse método. Neste trabalho utilizamos o algoritmo LiBOC, o qual possui complexidade linear em relação ao número de exemplos. A segunda questão está relacionada ao algoritmo usado, internamente, para o ajuste da reta que determina a dimensão fractal do conjunto de dados em análise. Diferentes algoritmos para realizar o ajuste da reta ao trecho linear da curva podem resultar em diferentes aproximações de valores da dimensão fractal, assim como diferentes algoritmos que implementam o cálculo aproximado da dimensão fractal por meio de métodos como o *Box Count Plot*, podem originar variações nos valores da dimensão fractal encontrada. Desse modo, os resultados apresentados neste trabalho aplicam-se à utilização da dimensão fractal dos conjuntos de dados calculada, especificamente, pelos algoritmos citados. É importante ressaltar que o algoritmo LiBOC, foi escolhido neste trabalho por apresentar performance superior aos melhores algoritmos citados na literatura que possuem o mesmo objetivo (Traina et al., 2000).

Teoricamente, o algoritmo FDimBF deve ser utilizado somente para conjuntos de dados que apresentem característica de um fractal estatisticamente auto-similar. Portanto, é importante a intervenção do usuário para observar:

1. o formato da curva de comportamento do conjunto de dados;
2. o número de pontos utilizados para construir essa curva e
3. o ajuste da reta, que determina a dimensão fractal, em relação à curva

os quais representam aspectos importantes associados à dimensão fractal.

8.2 Trabalhos futuros

Durante o desenvolvimento deste trabalho foram identificadas diversas questões interessantes para serem investigadas em trabalhos futuros.

Uma dessas questões está relacionada ao modelo de avaliação de performance de algoritmos de seleção de atributos, para o qual é importante a utilização de mecanismos de avaliação que considerem mais de um parâmetro para a realização dessa análise. Uma outra questão está associada ao uso de meta-aprendizado para a determinação de qual algoritmo poderia produzir melhores resultados para um determinado conjunto de dados. No trabalho de Brazdil et al. (2003), essas duas questões são tratadas com a proposição

de modelos multivariados que auxiliam na avaliação de performance de algoritmos e com a aplicação de meta-aprendizado para determinar que algoritmo, de acordo com as características do conjunto original de dados, poderia ser mais apropriado para a tarefa por ele realizado. Neste trabalho propusemos um modelo de avaliação de performance de algoritmos de SA, o qual combina a precisão do modelo construído a partir do subconjunto de atributos selecionado pelo algoritmo de SA com a redução no tamanho desse subconjunto, proporcionando uma medida categórica dessa performance. Pretende-se investigar outros modelos que combinam ambos parâmetros para a avaliação de performance de algoritmos de seleção de atributos, mas que forneçam um índice numérico para essa medida. Com relação a meta-aprendizado para associar as características do conjunto original de dados ao comportamento do algoritmo FDimBF, pretende-se incrementar a meta-base utilizada neste trabalho com os resultados obtidos em futuros experimentos utilizando outras bases de dados, a fim de verificar, entre outros, se a estimativa do erro gerado com os atributos considerados no conjunto de meta-dados Meta2, continua sendo de 0%.

Outra questão refere-se ao algoritmo LiBOC. Está sendo incorporada a esse algoritmo a possibilidade de um ajuste da reta à curva de comportamento do conjunto de dados com intervenção do usuário (Sousa, 2005). Essa nova funcionalidade permitiria ao usuário realizar o ajuste por meio de ferramentas de visualização, que em tarefas de exploração dos dados, bastante comuns no processo de extração de conhecimento de bases de dados, seriam muito úteis. Situação como a ocorrida no estudo de caso apresentado neste trabalho com o subconjunto de dados selecionado pelo algoritmo FDimBF(1), poderia ser melhor explorada com um ajuste controlado pelo usuário. Desse modo, após a incorporação dessa facilidade ao algoritmo LiBOC, um dos trabalhos futuros consiste em aplicar o ajuste com intervenção do usuário para o estudo de caso realizado neste trabalho.

Uma outra questão a ser pesquisada é a influência da discretização realizada por alguns dos algoritmos de seleção de atributos considerados neste trabalho. Como mencionado, essa transformação dos dados é realizada internamente por esses algoritmos, sem que o usuário conheça exatamente os parâmetros usados para realizá-la. Se ao usuário não é possível discretizar os atributos numéricos como foram discretizados pelo algoritmo durante o processo de seleção de atributos, o modelo será construído a partir de um subconjunto de atributos que não é exatamente o que foi considerado para o processo de seleção de atributos. Pretende-se pesquisar essa questão, avaliando experimentalmente os algoritmos de seleção de atributos com essas características, fornecendo a esses algoritmos os atributos previamente discretizados e analisar o impacto dessa transformação ser realizada antes da SA ou internamente à SA.

Em relação ao estudo de caso desenvolvido neste trabalho, foram identificadas diversas questões interessantes, do ponto de vista do domínio da aplicação, para possíveis trabalhos futuros. Entre essas questões de interesse, pode-se citar um indicativo de alguma relação entre os valores de motilidade de grau A e motilidade de grau D no conjunto de dados considerado neste trabalho. Ainda em relação ao estudo de caso, seria interessante

investigar se a discretização, guiada por especialistas do domínio, de alguns dos atributos numéricos, poderia auxiliar em uma melhor definição das bordas de separação dos exemplos no conjunto de dados, pois os especialistas consideraram que alguns dos limiares utilizados nas regras induzidas não são suficientemente discriminantes para a separação de exemplos de classes distintas.

Um conceito bastante utilizado na área médica e que tem sido aplicado cada vez mais na área computacional, está relacionado às curvas *Receiver Operating Characteristic* — ROC (Fawcett, 2003). As curvas ROC constituem uma ferramenta útil para organizar classificadores e visibilizar sua performance. Pretende-se investigar a utilização da área sob uma curva ROC (*Area under a ROC Curve* — AUC) como uma possível medida para compor um modelo de avaliação de performance de algoritmos de seleção de atributos. Ainda com relação ao tema das curvas ROC, foi proposto recentemente, por Prati and Flach (2005), o algoritmo ROCCER, o qual tem como objetivo selecionar regras considerando a curva ROC. A partir de um conjunto redundante de regras, o algoritmo ROCCER busca por um conjunto menor de regras, de tal maneira que esse conjunto reduzido maximize a AUC. Resultados reportados nesse trabalho indicam que o ROCCER tem um desempenho compatível aos algoritmos C4.5 e CN2 em termos da AUC, mas com um número de regras selecionadas consideravelmente menor que o número de regras induzidas por C4.5 e CN2. Assim, um outro trabalho futuro inclui avaliar os algoritmos de seleção de atributos considerados neste trabalho, utilizando o algoritmo ROCCER para gerar modelos considerando os atributos selecionados.

Apêndice A

Distribuição dos Valores dos Atributos

Neste Apêndice são mostradas as distribuições aproximadas dos valores dos atributos do conjunto de dados reais Processamento de Sêmen, utilizado no estudo de caso.

Figura A.1: Distribuições dos valores dos atributos — A

(a) pH

(b) tempo-abstinencia

(c) volume

(d) viscosidade

(e) concentracao

(f) concentracao-total

(g) motilidade-grau-a

(h) motilidade-grau-b

(i) motilidade-grau-c

Figura A.2: Distribuições dos valores dos atributos — B

(a) motilidade-grau-d

(b) motilidade

(c) motilidade-progressiva

(d) vitalidade

(e) det-leo-sem

(f) nro-leo-pol

(g) nro-cel-red

(h) morfologia-Kruger

(i) classe

Figura A.3: Distribuições dos valores dos atributos — C

Apêndice B

Avaliação das Regras por meio de Medidas Objetivas e Subjetiva

Neste Apêndice são mostrados os resultados das medidas de avaliação objetiva e subjetiva para cada uma das regras que compõem os modelos construídos considerando os subconjuntos derivados a partir do conjunto de dados Processamento de Sêmen, utilizado no estudo de caso.

Para cada modelo construído usando o conjunto original de atributos e os subconjuntos de atributos selecionados pelos algoritmos de SA — $\mathcal{C}4.5$, ReliefF, CFS, FCBF, CBF e FDimBF(2) — são apresentadas as seguintes informações:

- Regra: identificação da regra;
- Acc, Sens, Sup e Nov: valores das medidas objetivas de precisão, sensibilidade, suporte e novidade para cada uma das regras;
- # Condições: número de condições que compõem cada regra;
- IAE: valor do índice de avaliação do especialista para cada regra;
- Classe: classe correspondente de cada regra;
- # Medidas Acima: número de medidas que alcançaram os limiares ($\text{Acc} \geq 0,80$; $\text{Sens} \geq 0,20$; $\text{Sup} \geq 0,05$ e $\text{Nov} \geq 0,05$) estabelecidos para as medidas objetivas para cada regra e
- # Regras Acima e % Regras Acima: nas últimas duas linhas, número e respectiva percentagem de regras que alcançaram os limiares estabelecidos para as medidas objetivas para cada medida.

Regra	Acc	Sens	Sup	Nov	# Condições	IAE	Classe	# Medidas Acima
R1	0,95	0,41	0,09	0,07	3	1	1	4
R2	0,93	0,59	0,12	0,10	3	1	1	4
R3	0,92	0,39	0,08	0,06	2	0	1	4
R4	0,78	0,51	0,11	0,08	4	3	1	3
R5	1,00	0,10	0,02	0,02	7	3	2	1
R6	0,94	0,21	0,04	0,03	6	3	2	2
R7	1,00	0,09	0,02	0,01	6	1	2	1
R8	1,00	0,08	0,01	0,01	9	2	2	1
R9	1,00	0,06	0,01	0,01	4	2	2	1
R10	1,00	0,05	0,01	0,01	2	3	2	1
R11	0,78	0,09	0,02	0,01	5	3	2	0
R12	0,67	0,10	0,02	0,01	5	0	2	0
R13	0,49	0,27	0,05	0,03	4	0	2	2
R14	0,98	0,61	0,37	0,14	2	3	3	4
R15	0,90	0,30	0,18	0,06	5	3	3	4
R16	0,75	0,85	0,51	0,10	1	1	3	3
Média	0,88	0,29	0,10	0,05	4,25	1,81		
SE	0,05	0,08	0,04	0,01	0,67	0,39		
# Regras Acima	11	9	5	7				
% Regras Acima	68,75	56,25	31,25	43,75				

Tabela B.1: Medidas objetivas e subjetiva — Conjunto original de atributos

Regra	Acc	Sens	Sup	Nov	# Condições	IAE	Classe	# Medidas Acima
R1	0,95	0,41	0,09	0,07	3	1	1	4
R2	0,93	0,59	0,12	0,10	3	1	1	4
R3	0,92	0,39	0,08	0,06	2	0	1	4
R4	0,78	0,51	0,11	0,08	4	3	1	3
R5	1,00	0,10	0,02	0,02	7	3	2	1
R6	0,94	0,21	0,04	0,03	6	3	2	2
R7	1,00	0,09	0,02	0,01	6	1	2	1
R8	1,00	0,08	0,01	0,01	9	2	2	1
R9	1,00	0,06	0,01	0,01	4	2	2	1
R10	1,00	0,06	0,01	0,01	6	0	2	1
R11	1,00	0,05	0,01	0,01	2	3	2	1
R12	0,88	0,09	0,02	0,01	6	1	2	1
R13	0,98	0,61	0,37	0,14	2	3	3	4
R14	1,00	0,12	0,07	0,03	6	4	3	2
R15	1,00	0,10	0,06	0,02	3	0	3	2
R16	0,90	0,30	0,18	0,06	5	3	3	4
R17	0,75	0,85	0,51	0,10	1	1	3	3
Média	0,94	0,27	0,10	0,05	4,41	1,82		
SE	0,02	0,08	0,04	0,01	0,69	0,41		
# Regras Acima	15	8	9	7				
% Regras Acima	88,24	47,06	52,94	41,18				

Tabela B.2: Medidas objetivas e subjetiva — Subconjunto de atributos selecionado por C4.5

Regra	Acc	Sens	Sup	Nov	# Condições	IAE	Classe	# Medidas Acima
R1	0,95	0,41	0,09	0,07	3	1	1	4
R2	0,94	0,36	0,08	0,06	3	4	1	4
R3	0,93	0,59	0,12	0,10	3	1	1	4
R4	0,25	0,59	0,12	0,02	1	0	1	2
R5	0,94	0,21	0,04	0,03	6	3	2	2
R6	1,00	0,09	0,02	0,01	7	2	2	1
R7	1,00	0,09	0,02	0,01	6	3	2	1
R8	1,00	0,06	0,01	0,01	4	3	2	1
R9	1,00	0,05	0,01	0,01	6	3	2	1
R10	1,00	0,05	0,01	0,01	2	2	2	1
R11	0,77	0,13	0,02	0,02	10	2	2	0
R12	0,78	0,09	0,02	0,01	7	2	2	0
R13	0,78	0,09	0,02	0,01	5	3	2	0
R14	0,60	0,19	0,04	0,03	5	2	2	0
R15	0,50	0,03	0,00	0,00	1	0	2	0
R16	0,98	0,61	0,37	0,14	2	4	3	4
R17	1,00	0,10	0,06	0,02	3	3	3	2
R18	1,00	0,08	0,05	0,02	4	4	3	1
R19	1,00	0,08	0,05	0,02	4	4	3	1
R20	1,00	0,07	0,04	0,02	5	3	3	1
R21	1,00	0,07	0,04	0,02	3	3	3	1
R22	0,97	0,14	0,09	0,03	4	3	3	2
R23	0,94	0,18	0,11	0,04	3	3	3	2
R24	0,92	0,10	0,06	0,02	4	3	3	2
R25	0,90	0,30	0,18	0,06	5	3	3	4
Média	0,89	0,19	0,07	0,03	4,24	2,56		
SE	0,06	0,06	0,02	0,01	0,65	0,35		
# Regras Acima	19	7	10	5				
% Regras Acima	76,00	28,00	40,00	20,00				

Tabela B.3: Medidas objetivas e subjetiva — Subconjunto de atributos selecionado por ReliefF

Regra	Acc	Sens	Sup	Nov	# Condições	IAE	Classe	# Medidas Acima
R1	0,90	0,45	0,09	0,07	3	2	1	4
R2	0,85	0,72	0,15	0,11	2	1	1	4
R3	0,82	0,27	0,06	0,04	4	0	1	3
R4	1,00	0,05	0,01	0,01	4	3	2	1
R5	0,77	0,22	0,04	0,03	4	3	2	1
R6	0,80	0,05	0,01	0,01	4	2	2	1
R7	0,29	0,69	0,13	0,05	1	0	2	2
R8	0,97	0,61	0,37	0,14	1	4	3	4
R9	0,96	0,20	0,12	0,04	4	3	3	2
R10	0,96	0,11	0,07	0,03	2	3	3	2
R11	0,89	0,66	0,40	0,13	2	3	3	4
R12	0,88	0,54	0,32	0,10	2	0	3	4
Média	0,84	0,38	0,15	0,06	2,75	2,00		
SE	0,06	0,08	0,04	0,01	0,38	0,45		
# Regras Acima	9	8	9	5				
% Regras Acima	75,00	66,67	75,00	41,67				

Tabela B.4: Medidas objetivas e subjetiva — Subconjunto de atributos selecionado por CFS

Regra	Acc	Sens	Sup	Nov	# Condições	IAE	Classe	# Medidas Acima
R1	1,00	0,15	0,03	0,03	4	2	1	1
R2	0,95	0,41	0,09	0,07	3	2	1	4
R3	0,94	0,36	0,08	0,06	3	3	1	4
R4	0,93	0,59	0,12	0,10	3	2	1	4
R5	0,92	0,39	0,08	0,06	2	2	1	4
R6	1,00	0,05	0,01	0,01	4	0	1	1
R7	0,78	0,09	0,02	0,01	5	3	2	0
R8	0,70	0,09	0,02	0,01	3	2	2	0
R9	0,67	0,10	0,02	0,01	6	0	2	0
R10	0,29	0,95	0,18	0,06	1	0	2	3
R11	1,00	0,07	0,04	0,02	3	3	3	1
R12	0,93	0,11	0,06	0,02	4	0	3	2
R13	0,82	0,95	0,57	0,15	1	0	3	4
Média	0,84	0,33	0,10	0,05	3,23	1,46		
SE	0,06	0,10	0,05	0,01	0,45	0,40		
# Regras Acima	9	6	7	6				
% Regras Acima	69,23	46,15	53,85	46,15				

Tabela B.5: Medidas objetivas e subjetiva — Subconjunto de atributos selecionado por FCBF

Regra	Acc	Sens	Sup	Nov	# Condições	IAE	Classe	# Medidas Acima
R1	0,94	0,20	0,04	0,03	4	2	1	2
R2	0,92	0,39	0,08	0,06	2	1	1	4
R3	1,00	0,07	0,01	0,01	3	2	1	1
R4	0,58	0,85	0,18	0,11	1	0	1	3
R5	1,00	0,06	0,01	0,01	5	1	2	1
R6	0,91	0,13	0,02	0,02	7	3	2	1
R7	1,00	0,04	0,01	0,01	2	1	2	1
R8	0,80	0,10	0,02	0,01	4	2	2	1
R9	0,75	0,08	0,01	0,01	4	2	2	0
R10	0,68	0,22	0,04	0,03	3	3	2	1
R11	0,67	0,23	0,04	0,03	3	3	2	1
R12	0,56	0,37	0,07	0,05	2	2	2	2
R13	0,56	0,06	0,01	0,01	2	0	2	0
R14	0,50	0,09	0,02	0,01	2	3	2	0
R15	0,85	0,93	0,56	0,16	2	3	3	4
Média	0,78	0,25	0,08	0,04	3,07	1,87		
SE	0,06	0,09	0,04	0,01	0,49	0,34		
# Regras Acima	7	6	4	3				
% Regras Acima	46,67	40,00	26,67	20,00				

Tabela B.6: Medidas objetivas e subjetiva — Subconjunto de atributos selecionado por CBF

Regra	Acc	Sens	Sup	Nov	# Condições	IAE	Classe	# Medidas Acima
R1	0,87	0,62	0,13	0,10	3	1	1	4
R2	0,42	0,75	0,16	0,08	1	0	1	3
R3	1,00	0,05	0,01	0,01	4	3	2	1
R4	1,00	0,04	0,01	0,01	2	1	2	1
R5	0,83	0,13	0,02	0,02	6	2	2	1
R6	1,00	0,03	0,00	0,00	3	1	2	1
R7	0,80	0,05	0,01	0,01	3	3	2	1
R8	0,80	0,05	0,01	0,01	5	3	2	1
R9	1,00	0,01	0,00	0,00	1	0	2	1
R10	0,70	0,09	0,02	0,01	8	3	2	0
R11	0,68	0,17	0,03	0,02	4	1	2	0
R12	0,30	0,59	0,11	0,04	1	0	2	2
R13	0,93	0,11	0,07	0,02	4	2	3	2
R14	0,87	0,78	0,47	0,15	3	2	3	4
R15	0,86	0,10	0,06	0,02	3	2	3	2
R16	0,84	0,18	0,11	0,03	4	1	3	2
R17	0,75	0,49	0,29	0,06	1	0	3	3
Média	0,80	0,25	0,09	0,03	3,29	1,47		
SE	0,06	0,09	0,04	0,01	0,60	0,36		
# Regras Acima	10	5	8	4				
% Regras Acima	58,82	29,41	47,06	23,53				

Tabela B.7: Medidas objetivas e subjetiva — Subconjunto de atributos selecionado por FDimBF(2)

Referências Bibliográficas

- Almuallim, H. and Diettrich, T. G. (1991). Learning boolean concepts in the presence of many irrelevant features. *Artificial Intelligence*, 69:279–306. Citado nas páginas 4 e 32.
- Baranauskas, J. A. (2001). Extração Automática de Conhecimento por Múltiplos Indutores. Tese de Doutorado, ICMC-USP, <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-08102001-112806>. Citado nas páginas 2, 14, e 25.
- Baranauskas, J. A. and Monard, M. C. (1998). Experimental feature selection using the wrapper approach. In *Proc. of the International Conference on Data Mining*, pages 161–170, Rio de Janeiro, RJ. Citado na página 26.
- Baranauskas, J. A., Monard, M. C., and Horst, P. S. (1999). Evaluation of feature selection by wrapping around the *CN2* inducer. In *Encontro Nacional de Inteligência Artificial (ENIA/SBC)*, pages 315–326, Rio de Janeiro, RJ. <http://www.fmrp.usp.br/~augusto/>. Citado na página 26.
- Batista, G. E., Prati, R. C., and Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning data. *SIGKDD Explorations: Special issue on Learning from Imbalanced Datasets*, 6(1):20–29. <http://www.acm.org/sigs/sigkdd/explorations/>. Citado na página 58.
- Batista, G. E. A. P. A. (2001). Sintaxe padrão do arquivo de exemplos do projeto DISCOVER. <http://www.icmc.sc.usp.br/~gbatista/SintaxePadraoFinal.htm>. Citado na página 66.
- Batista, G. E. A. P. A. (2003). Pré-processamento de Dados em Aprendizado de Máquina Supervisionado. Tese de Doutorado, ICMC-USP, <http://www.icmc.usp.br/~gbatista/pdfs/TeseDoutorado.pdf>. Citado nas páginas 2 e 66.
- Batista, G. E. A. P. A. and Monard, M. C. (2002). A study of k-nearest neighbour as an imputation method. In Abraham, A., del Solar, J. R., and Köppen, M., editors, *Soft Computing Systems: Design, Management and Applications*, pages 251–260, Santiago, Chile. IOS Press. <http://www.icmc.usp.br/~gbatista>. Citado na página 66.
- Batista, G. E. A. P. A. and Monard, M. C. (2003a). An analysis of four missing data treatment methods for supervised learning. *Applied Artificial Intelligence*, 17(5):519–533. Citado nas páginas 65 e 104.
- Batista, G. E. A. P. A. and Monard, M. C. (2003b). Descrição da arquitetura e do projeto do ambiente computacional DISCOVER LEARNING ENVIRONMENT — DLEF. Technical Report 187, ICMC-USP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/RT_187.pdf. Citado na página 66.

- Batista, G. E. A. P. A. and Monard, M. C. (2005). The DISCOVER OBJECT LIBRARY — DOL user's manual. Technical report, ICMC-USP. (em preparação). Citado na página 66.
- Bell, D. A. and Wang, H. (2000). A formalism for relevance and its application in feature subset selection. *Mach. Learn.*, 41:175–195. Citado na página 40.
- Bendhack, D. A. and Damião, R., editors (1999). *Guia Prático de Urologia*. BG Editora e Produções Culturais, rio de Janeiro, 1 edition. http://www.sbu-mg.org.br/Guia_pratico.htm. Citado na página 92.
- Bernardini, F. C., Monard, M. C., Lee, H. D., and Esteves, S. C. (2003). Um algoritmo para selecionar regras de conhecimento utilizando medidas de avaliação de regras. In *Anais do Workshop on Advances and Trends in AI for Problem Solving, Jornadas Chilenas de Computación, Sociedad Chilena de Ciencias de la Computación*, pages 7–13, Chillan, Chile. http://www.icmc.usp.br/~fbernard/atai2003_bernardini.zip. Citado na página 130.
- Bins, J. and Draper, B. (2001). Feature selection from huge feature sets. In *International Conference on Computer Vision*, volume 2, pages 159–165, Vancouver, Canada. <http://citeseer.nj.nec.com/452418.html>. Citado na página 24.
- Blum, A. L. and Langley, P. (1997). Selection of relevant features and examples in machine learning. *Artificial Intelligence*, pages 245–271. Citado nas páginas 13, 18, 26, 36, 38, e 44.
- Brassard, G. and Bratley, P. (1997). *Fundamentals of Algorithms*. Prentice Hall, New Jersey. Citado na página 62.
- Brazdil, P. B., Soares, C., and Costa, J. P. D. (2003). Ranking learning algorithms: Using IBL and meta-learning on accuracy and time results. *Mach. Learn.*, 50(3):251–277. Citado na página 132.
- Caruana, R. A. and Freitag, D. (1994). How useful is relevance ? *Working Notes of the AAAI Fall Symposium on Relevance*, pages 25–29. Citado na página 42.
- Clark, P. and Niblett, T. (1989). The CN2 induction algorithm. *Mach. Learn.*, 3(4):261–283. Citado na página 25.
- Das, S. (2001). Filters, wrappers and a boosting based hybrid for feature selection. In *8th Int. Conf. on Machine Learning*, pages 74–81, Williams College. Citado na página 24.
- Dash, M., Choi, K., Scheuermann, P., and Liu, H. (2002). Feature selection for clustering – a filter solution. In *International Conference on Data Mining*, pages 115–122, Maebashi City, Japan. Citado na página 22.
- Dash, M. and Liu, H. (1997). Feature selection for classification. *Intelligent Data Analysis*, 1(3):131–156. [http://dx.doi.org/10.1016/S1088-467X\(97\)00008-5](http://dx.doi.org/10.1016/S1088-467X(97)00008-5). Citado nas páginas 13 e 22.
- Dash, M. and Liu, H. (1998). Hybrid search of feature subsets. In *Pacific Rim International Conference on Artificial Intelligence*, pages 238–249. Citado na página 24.

- Dash, M. and Liu, H. (1999). Handling large unsupervised data via dimensionality reduction. In *ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*. <http://www.informatik.uni-trier.de/~ley/db/indices/a-tree/d/Dash:Manoranjan.html>. Citado na página 22.
- Dash, M. and Liu, H. (2000). Feature selection for clustering. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 110–121. <http://citeseer.nj.nec.com/320394.html>. Citado na página 22.
- Dash, M. and Liu, H. (2003). Consistency-based search in feature selection. *Artif. Intell.*, 151(1-2):155–176. Citado nas páginas 4 e 33.
- Dietterich, T. G. (1989). Limitations on inductive learning. In *Proceedings of the sixth international workshop on Machine learning*, pages 124–128, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc. Citado nas páginas 57 e 128.
- Dy, J. G. and Brodley, C. E. (2000). Feature subset selection and order identification for unsupervised learning. In *Proc. 17th International Conf. on Machine Learning*, pages 247–254. Morgan Kaufmann, San Francisco, CA. <http://citeseer.nj.nec.com/dy00feature.html>. Citado na página 22.
- Esteves, S. C. (2005). Infertilidade Masculina. <http://www.androfert.com.br/masculina.asp>. Citado na página 93.
- Esteves, S. C., Lee, H. D., and Monard, M. C. (2001). Inteligência artificial aplicada à andrologia: Um estudo de caso do processamento de sêmen diagnóstico. In *Anais do XXVII Congresso Brasileiro de Urologia*, pages 1–1, Fortaleza, CE. Citado nas páginas 91 e 129.
- Faloutsos, C. and Kamel, I. (1994). Beyond uniformity and independence: Analysis of r-trees using the concept of fractal dimension. In *Proc. of the 13th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pages 4–13, Minneapolis, MN. Citado nas páginas 49 e 132.
- Fawcett, T. (2003). ROC graphs: Notes and practical considerations for data mining researchers. Technical Report HPL-2003-4, HP Labs. http://www.hpl.hp.com/personal/Tom_Fawcett/papers/HPL-2003-4.pdf. Citado na página 134.
- Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996a). From data mining to knowledge discovery in databases. *AI Magazine*, 17:37–54. Citado na página 17.
- Fayyad, U. M. and Irani, K. B. (1993). Multi-interval discretisation of continuous-valued attributes. In *Proc. of the Thirteenth International Joint Conference on Artificial Intelligence*, pages 1022–1027. Morgan Kaufmann. Citado nas páginas 62 e 63.
- Fayyad, U. M., Piatetsky-Shapiro, G., and Smyth, P. (1996b). *From Data Mining to Knowledge Discovery: An Overview*, pages 1–30. American Association for Artificial Intelligence, Menlo Park, CA. Citado nas páginas 1 e 14.
- Ferro, M. (2004). Aquisição de conhecimento de conjuntos de exemplos no formato atributo-valor utilizando aprendizado de máquina relacional. Dissertação de Mestrado, ICMC-USP, <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-16112004-095938/>. Citado na página 96.

- Ferro, M., Lee, H. D., and Esteves, S. C. (2002). Intelligent data analysis: A case study of the diagnostic sperm processing. In *Proc. of International Conference on Computer Science, Software Engineering, Information Technology, e-Business and Applications*, pages 116–120, Foz do Iguaçu, PR. Citado nas páginas 91 e 129.
- Gennari, J. H., Langley, P., and Fisher, D. (1989). Models of incremental concept formation. *Artificial Intelligence*, 40:11–61. Citado na página 33.
- Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning*, 3:1157–1182. Citado nas páginas 11, 13, e 14.
- Hair, P. E., Anderson, R., Tatham, R., and Black, W. (1998). *Multivariate Data Analysis*. Prentice Hall, New Jersey. Citado na página 87.
- Hall, M. (1999). *Correlation-based Feature Subset Selection for Machine Learning*. PhD thesis, Department of Computer Science. <http://www.cs.waikato.ac.nz/~mhall/thesis.pdf>. Citado na página 63.
- Hall, M. A. (2000). Correlation-based feature selection for discrete and numeric class machine learning. In *Proc. of the 17th Int. Conf. on Machine Learning*, pages 359–366, San Francisco, CA. Morgan Kaufmann. Citado nas páginas 3, 4, 18, 54, 62, e 128.
- Han, J. and Kamber, M. (2000). *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA. Citado na página 2.
- Hand, D. J., Smyth, P., and Mannila, H. (2001). *Principles of Data Mining*. MIT Press, Cambridge, MA. Citado na página 2.
- Honorato, D. D. F., Lee, H. D., Monard, M. C., Wu, F. C., Machado, R. B., Neto, A. P., and Ferrero, C. A. (2005a). Uma metodologia para auxiliar no processo de construção de bases de dados. In *Anais do V Encontro Nacional de Inteligência, XXV Congresso da Sociedade Brasileira de Computação*, pages 593–601, Porto Alegre, RS. http://www.unisinos.br/_diversos/congresso/sbc2005/_dados/anais/pdf/arq0239.pdf. Citado nas páginas 96, 97, 99, 100, 101, e 129.
- Honorato, D. D. F., Lee, H. D., Wu, F. C., Machado, R. B., Neto, A. P., Fagundes, J. J., and Góes, J. R. N. (2005b). H.pylori-mindsys: Um protótipo de sistema baseado em conhecimento para auxílio na predição da bactéria helicobacter pylori em doenças pépticas. In *Anais do V Workshop de Informática Médica*, pages 1–1, Porto Alegre, RS. Citado nas páginas 91 e 129.
- Honorato, D. F., Lee, H. D., Ferrero, C. A., Wu, F. C., Monard, M. C., Neto, A. P., and Machado, R. B. (2004a). Construção de uma base de dados para a extração de conhecimento aplicada a doenças pépticas por meio de indexação automática. In *Anais do XII Simpósio Internacional de Iniciação Científica da USP*, pages 1–1, São Paulo, SP. Citado na página 129.
- Honorato, D. F., Lee, H. D., Machado, R. B., Wu, F. C., Neto, A. P., and Ferrero, C. A. (2004b). Utilização da indexação automática para auxílio à construção de uma base de dados para a extração de conhecimento aplicada à doenças péptica. In *Anais do I WorkComp Sul*, pages 1–9, Palhoça, SC. Citado nas páginas 96 e 129.

- Honorato, D. F., Lee, H. D., Wu, F. C., Machado, R. B., Monard, M. C., and Ferrero, C. A. (2005c). Construção de uma base de dados estruturada a partir de laudos médicos de endoscopia digestiva alta. In *Anais da III Jornada Científica da Unioeste*, pages 1–6, Marechal Cândido Rondon, PR. Citado na página 129.
- Honorato, D. F., Lee, H. D., Wu, F. C., Machado, R. B., and Neto, A. P. (2004c). Indexação automática para extração de conhecimento de base de dados médica relacionada a doenças pépticas. In *Anais do XIII Encontro Anual de Iniciação Científica*, pages 2–2, Londrina, PR. Citado na página 129.
- John, G., Kohavi, R., and Pfleger, K. (1994). Irrelevant features and the subset selection problem. In Kaufmann, M., editor, *Proc. of the 11th Int. Conf. on Machine Learning*, pages 167–173, San Francisco, CA. Citado nas páginas 25, 35, 36, e 37.
- Kira, K. and Rendell, L. (1992). A practical approach to feature selection. In *Proc. of the 9th Int. Conf. on Machine Learning*, pages 249–256, Aberdeen, Scotland. Morgan Kaufmann. Citado nas páginas 4, 42, e 61.
- Kohavi, R. and John, G. H. (1997). Wrappers for feature subset selection. *Artif. Intell.*, 97(1-2):273–324. Citado nas páginas 17, 18, 22, 24, 26, 32, e 42.
- Koller, D. and Sahami, M. (1996). Toward optimal feature selection. In *Proc. of the 13th Int. Conf. on Machine Learning*, pages 284–292, Bari, Italy. Citado nas páginas 3, 4, e 18.
- Kononenko, I. (1994). Estimating attributes: Analysis and extension of Relief. In *Proc. of the 1994 European Conference on Machine Learning*, pages 171–182, Amsterdam. Springer-Verlag. Citado nas páginas 61 e 128.
- Köpt, C. (2002). The data characterization tool DCT. Technical report, DaimlerChrysler AG, Center of Research and Technology. http://www.liacc.up.pt/ML/METAL/Consortium/doc/DC_Tool.ps. Citado nas páginas 59 e 105.
- Langley, P. (1988). Machine learning as an experimental science. *Mach. Learn.*, 3(1):5–8. Citado nas páginas 57 e 128.
- Langley, P. (1994). Selection of relevant features in machine learning. pages 140–144. *Proc. of the AAAI Fall Symposium on Relevance*. AAAI Press. Citado na página 19.
- Langley, P. (1996). *Elements of Machine Learning*. Morgan Kaufmann Publishers, Inc, San Francisco, CA. Citado na página 20.
- Lavrač, N., Flach, P., and Zupan, B. (1999). Rule evaluation measures: a unifying view. In *Proc. of the Ninth International Workshop on Inductive Logic Programming*, volume 1634, pages 74–185. Springer-Verlag. *Lectures Note in Artificial Intelligence*. Citado nas páginas 116 e 117.
- Lee, H. D. (2000). Seleção e construção de features relevantes para o aprendizado de máquina. Dissertação de Mestrado, ICMC-USP, <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-15032002-113112>. Citado nas páginas 13, 18, e 96.
- Lee, H. D. and Monard, M. C. (2003). Seleção de atributos para algoritmos de aprendizado de máquina supervisionado utilizando como filtro a dimensão fractal. *Revista de La Sociedad Chilena de Ciencia de La Computación*, 4(1):1–8. Citado nas páginas 50, 63, e 128.

- Lee, H. D., Monard, M. C., and Baranauskas, J. A. (1999). Empirical comparison of wrapper and filter approaches for feature subset selection. Technical Report 94, ICMC - USP, São Carlos, SP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_94.ps.zip. Citado na página 26.
- Lee, H. D., Monard, M. C., Honorato, D. F., and Wu, F. C. (2005a). A methodology to construct structured data bases from medical findings. (a ser submetido ao periódico *Computer Methods and Programs in Biomedicine*). Citado na página 129.
- Lee, H. D., Monard, M. C., Voltolini, R. F., and Wu, F. C. (2005b). Avaliação experimental e comparação de algoritmos de seleção de atributos importantes com o algoritmo FDimBF baseado na dimensão fractal. Technical Report 264, ICMC-USP. http://www.icmc.usp.br/~biblio/download/RT_264.pdf. Citado nas páginas 67, 128, e 129.
- Lee, H. D., Monard, M. C., Voltolini, R. F., and Wu, F. C. (2005c). Proposta de um algoritmo de seleção de atributos importantes para aprendizado supervisionado utilizando a dimensão fractal para tratamento de redundância: Avaliação experimental. In *Proc. of the Sixth Workshop on Artificial Intelligence, Jornadas Chilenas de Computación*, volume 1, pages 1–10, Valdivia, Chile. Citado nas páginas 50, 63, 128, e 129.
- Lee, H. D., Monard, M. C., and Wu, F. C. (2005d). Feature subset selection for supervised learning using fractal dimension. In *Frontiers in Artificial Intelligence and Applications*, volume 132, pages 135–142, Himeji, Japan. IOS Press. Citado nas páginas 50, 63, e 128.
- Lee, H. D., Monard, M. C., and Wu, F. C. (2005e). Seleção de atributos relevantes e não redundantes usando a dimensão fractal do conjunto de dados. In *Anais do V Encontro Nacional de Inteligência Artificial, XXV Congresso da Sociedade Brasileira de Computação*, pages 444–453, Porto Alegre, RS. http://www.unisinos.br/_diversos/congresso/sbc2005/_dados/anais/pdf/arq0122.pdf. Citado nas páginas 50, 63, 128, e 129.
- Liu, H. and Motoda, H. (1998). *Feature Selection for Knowledge Discovery and Data Mining*. Kluwer Academic Publishers, Massachusetts. Citado nas páginas 3, 13, 14, 20, 22, 29, 31, e 45.
- Liu, H. and Motoda, H. (2002). Feature selection with selective sampling. In *9th Int. Conf. on Machine Learning*, pages 395–402, Sydney, Australia. Citado na página 22.
- Liu, H. and Setiono, R. (1996). A probabilistic approach to feature selection – a filter solution. In *Proc. of the 13th Int. Conf. on Machine Learning*, pages 319–327, Bari, Italy. Citado nas páginas 4, 33, 62, e 128.
- Liu, H. and Yu, L. (2002). Feature selection for data mining. http://www.public.asu.edu/~huanliu/feature_selection.html. Citado nas páginas 13, 14, 17, e 21.
- Machado, R. B., Lee, H. D., Wu, F. C., Fagundes, J. J., Góes, J. R. N., Coy, C. S. R., Voltolini, R. F., and Metz, J. (2002). A computational system to evaluate biomechanical colonic test: Sabi - biomechanical data acquisition and analysis system. In *Proc. of International Conference on Computer Science, Software Engineering, Information Technology, e-Business and Applications*, pages 352–356, Foz do Iguaçu, PR. Citado nas páginas 91 e 129.

- Mandelbrot, B. B. (1985). *The Fractal Geometry of Nature: Updated and Augmented*. W. H. Freeman and Company, New York. Citado na página 48.
- Martins, C. A. (2003). Uma Abordagem para Pré-processamento de Dados Textuais em Algoritmos de Aprendizado. Tese de Doutorado, ICMC-USP, <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-08032004-164855/>. Citado na página 18.
- Matsubara, E. T. and Monard, M. C. (2005). Utilizando algoritmos de aprendizado semi-supervisionado multivisão como rotuladores de texto. In *Anais do Workshop em Tecnologia da Informação, XXV Congresso da Sociedade Brasileira de Computação*, pages 2108–2117, Porto Alegre, RS. http://www.unisinos.br/_diversos/congresso/sbc2005/_dados/anaais/pdf/arq0107.pdf. Citado na página 18.
- Medeiros, A. S. (1993). *Semiologia Urológica*. Medesi, Rio de Janeiro, RJ. Citado na página 93.
- Michalski, R. S., Bratko, I., and Kubat, M., editors (1998). *Machine Learning and Data Mining: Methods and Applications*. John Wiley and Sons. West Sussex, England. Citado na página 2.
- Michie, D., Spiegelhalter, D. J., Taylor, C. C., and Campbell, J., editors (1994). *Machine Learning, Neural and Statistical Classification*. Ellis Horwood, New Jersey. Citado na página 58.
- Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill. Citado na página 116.
- Molina, L. C., Belanche, L., and Nebot, A. (2002). Feature selection algorithms: A survey and experimental evaluation. In *International Conference on Data Mining*, pages 306–313, Maebashi City, Japan. Citado nas páginas 13 e 44.
- Monard, M. C. and Baranauskas, J. A. (2003). *Indução de Regras e Árvores de Decisão*, chapter 5, pages 115–140. Volume 1 of Rezende (2003), 1 edition. Citado na página 25.
- Monard, M. C. and Lee, H. D. (2003). *Processamento de Sêmen Diagnóstico*, pages 461–463. Volume 1 of Rezende (2003), 1 edition. Parte II, Aplicação V. Citado nas páginas 91 e 129.
- Motoda, H. and Liu, H. (2002). Feature selection, extraction and construction. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. <http://www.public.asu.edu/~huanliu/pakdd02wk.ps>. Citado na página 14.
- Motulsky, H. (1995). *Intuitive Biostatistics*. Oxford University Press, New York. Citado nas páginas 67 e 108.
- Newman, D., Hettich, S., Blake, C., and Merz, C. (1998). UCI repository of machine learning databases. <http://www.ics.uci.edu/~mlearn/MLRepository.html>. Citado nas páginas 58 e 91.
- Pila, A. D. (2001). Seleção de Atributos Relevantes para Aprendizado de Máquina utilizando a Abordagem de Rough Sets. Dissertação de Mestrado, ICMC-USP, http://www.teses.usp.br/teses/disponiveis/55/55134/tde-13022002-153921/publico/dissertacao_AD.pdf. Citado na página 26.
- Prati, R. C. (2003). O *framework* de integração do sistema DISCOVER. Dissertação de Mestrado, ICMC-USP, <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-20082003-152116/publico/root.pdf>. Citado na página 66.

- Prati, R. C., Baranauskas, J. A., and Monard, M. C. (2001a). Extração de informações padronizadas para a avaliação de regras induzidas por algoritmos de aprendizado de máquina simbólico. Technical Report 145, ICMC-USP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/RT_145.ps.zip. Citado nas páginas 66 e 119.
- Prati, R. C., Baranauskas, J. A., and Monard, M. C. (2001b). Uma proposta de unificação da linguagem de representação de conceitos de algoritmos de aprendizado de máquina simbólicos. Technical Report 137, ICMC-USP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/RT_137.ps.zip. Citado na página 66.
- Prati, R. C., Baranauskas, J. A., and Monard, M. C. (2002). Padronização da sintaxe e informações sobre regras induzidas a partir de algoritmos de aprendizado de máquina simbólico. *Revista Eletrônica de Iniciação Científica*, 2(3). <http://www.sbc.org.br/reic/edicoes/2002e3>. Citado na página 66.
- Prati, R. C. and Flach, P. A. (2005). ROCCER: an algorithm for rule learning based on roc analysis. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI'05)*, pages 823–828. IJCAI. Citado na página 134.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1992). *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, New York. Citado nas páginas 41, 50, 61, e 62.
- Pyle, D. (1999). *Data Preparation for Data Mining*. Morgan Kaufmann, California. Citado nas páginas 2 e 96.
- Quinlan, J. R. (1983). *Machine Learning: An Artificial Intelligence Approach*. Morgan Kaufmann. California. Citado na página 25.
- Quinlan, J. R. (1986). Induction of decision trees. *Mach. Learn.*, 1(1):81–106. Citado na página 63.
- Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann. California. Citado nas páginas 25, 52, 60, 63, 65, 107, e 128.
- R Development Core Team (2005). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. <http://www.R-project.org>. Citado nas páginas 59 e 105.
- Rezende, S. O. (2003). *Sistemas Inteligentes: Fundamentos e Aplicações*. Editora Manole, Barueri, SP, Brasil. Citado nas páginas 151 e 152.
- Rezende, S. O., Pugliesi, J. B., Melanda, E. A., and Paula, M. F. (2003). *Mineração de Dados*, chapter 12, pages 307–336. Volume 1 of Rezende (2003), 1 edition. Citado na página 2.
- Robnik-Sikonja, M. and Kononenko, I. (2003). Theoretical and empirical analysis of ReliefF and RReliefF. *Mach. Learn.*, 53(1-2):23–69. Citado nas páginas 4, 42, 52, e 61.
- Rulequest-Research (1999). Data mining tools See5 and C5.0. <http://www.rulequest.com/see5-info.html>. Citado nas páginas 85, 87, e 107.
- Schapire, R. E. (1990). The strength of weak learnability. 5(2):197–227. Citado na página 24.

- Schlimmer, J. C. (1993). Efficiently inducing determinations: A complete and systematic search algorithm that uses optimal pruning. In *Int. Conf. on Machine Learning*, pages 284–290. Citado na página 21.
- Schroeder, M. (1991). *Fractals, Chaos, Power Laws: Minutes from an Infinite Paradise*. W. H. Freeman and Company, New York. Citado na página 132.
- Schwartz, R., Christiansen, T., and Pyle, L. W. (1997). *Learning Perl*. California, 2 edition. Citado na página 96.
- Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47. Citado nas páginas 18 e 99.
- Sousa, E. P. M. (2005). Extensões do algoritmo LiBOC. Comunicação pessoal. Citado nas páginas 68 e 133.
- Sousa, E. P. M., Traina, C., Traina, A. J. M., and Faloutsos, C. (2002). How to use fractal dimension to find correlations between attributes. In *Workshop Notes of KDD 2002 Workshop on Fractals and Self-similarity in Data Mining: Issues and Approaches*, pages 26–30, Edmonton, Canada. Citado nas páginas 48 e 53.
- Talavera, L. (1999). Feature selection as a preprocessing step for hierarchical clustering. In *Proc. 16th International Conf. on Machine Learning*, pages 389–397. Morgan Kaufmann, San Francisco, CA. <http://citeseer.nj.nec.com/talavera99feature.html>. Citado na página 22.
- Traina, C., Sousa, E. P. M., and Traina, A. J. M. (2005). *Using Fractals in Data Mining*, volume 1 of 1, chapter 24, pages 599–630. Wiley-IEEE Press, New Jersey, NJ, 1 edition. Citado nas páginas 4, 48, e 53.
- Traina, C., Traina, A. J. M., and Faloutsos, C. (2003). MDE – measure distance exponent manual. (Internal Document). Citado na página 53.
- Traina, C., Traina, A. J. M., Wu, L., and Faloutsos, C. (2000). Fast feature selection using fractal dimension. In *Proc. of the 15th Brazilian Data Base Symposium*, pages 158–171, João Pessoa, Brasil. Citado nas páginas 4, 40, 49, 50, 53, e 132.
- Verza Junior, S., Barbosa, M. F., Lee, H. D., Wu, F. C., Monard, M. C., and Esteves, S. C. (2003). Comparação entre gradientes coloidais de diferentes volumes para recuperação de espermatozóides. *Int Braz J Urol*, 29:120–120. Citado nas páginas 91 e 129.
- Voltolini, R. F., Metz, J., Machado, R. B., Lee, H. D., Wu, F. C., Fagundes, J. J., and Góes, J. R. N. (2003). Sabi 2.0: Um sistema para a realização de testes biomecânicos em material viscoelástico não linear. In *Proc. of the Fourth Congress of Logic Applied to Technology*, pages 93–99, Marília, SP. Citado nas páginas 91 e 129.
- Weiss, S. M. and Indurkha, N. (1998). *Predictive Data Mining: A Practical Guide*. Morgan Kaufmann, San Francisco, CA. Citado na página 2.
- Weiss, S. M. and Kulikowski, C. A. (1991). *Computer Systems that Learn*. Morgan Kaufmann Publishers, Inc. Citado na página 22.
- Weisstein, E. W. (2005). Parity. MathWorld – A Wolfram Web Resource <http://mathworld.wolfram.com/Parity.html>. Citado na página 34.

- Witten, I. H. and Frank, E. (2000). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, California. Citado nas páginas 2, 63, e 65.
- Wu, F. C., Lee, H. D., Machado, R. B., Dalmás, S., Coy, C. S., Góes, J. R. N., and Fagundes, J. J. (2004). Energia total de ruptura: Um teste biomecânico para avaliação de material biológico com propriedade viscoelástica não-linear. *Acta Cir Bras*, 19(6):609–616. Citado nas páginas 91 e 129.
- Wu, F. C., Lee, H. D., Niz, M. A. K., Ayrizono, M. L. S., Coy, C. S. R., Góes, J. R. N., and Fagundes, J. J. (2006). Estudo comparativo da resistência de rupturade cólon descendente por meio de ensaio uniaxial força de ruptura à tração e energia total de ruptura: Trabalho experimental em ratos. *Acta Cir Bras*, 21(2). (in print). Citado nas páginas 91 e 129.
- Wu, F. C., Lee, H. D., Rocha, A. A., Ayrizono, M. L. S., Coy, C. R. S., Góes, J. R. N., and Fagundes, J. J. (2005a). Estudo da ação de aderências sobre anastomoses cólicas por meio de ensaios biomecânicos uniaxiais e biaxiais: Trabalho experimental em ratos. In *Anais do LIV Congresso Brasileiro de Coloproctologia*, Goiânia, GO. Sociedade Brasileira de Coloproctologia. (in print). Citado nas páginas 91 e 129.
- Wu, F. C., Lee, H. D., Rocha, A. A., Ayrizono, M. L. S., Coy, C. R. S., Góes, J. R. N., and Fagundes, J. J. (2005b). Estudo dos efeitos de diferentes concentrações de oxigênio e da hiperoxigenação hiperbárica sobre anastomose cólica comprometida ou não pela isquemia: Trabalho experimental em ratos. In *Anais do LIV Congresso Brasileiro de Coloproctologia*, Goiânia, GO. Sociedade Brasileira de Coloproctologia. (in print). Citado nas páginas 91 e 129.
- Yu, L. and Liu, H. (2004). Efficient feature selection via analysis of relevance and redundancy. *Journal of Machine Learning Research*, 5:1205–1224. Citado nas páginas xix, 5, 18, 41, 50, 54, 61, 62, e 128.