Centro de Engenharias e Ciências Exatas

Avaliação de Métodos para Seleção de Atributos Importantes para Aprendizado de Máquina Supervisionado no Processo de Mineração de Dados

Antonio Rafael Sabino Parmezan
Huei Diana Lee
Newton Spolaôr
Wu Feng Chung

Nº 002

RELATÓRIOS TÉCNICOS DO LABORATÓRIO DE BIOINFORMÁTICA
UNIVERSIDADE ESTADUAL DO OESTE DO PARANÁ

Foz do Iguaçu
Dezembro/2012

Avaliação de Métodos para Seleção de Atributos Importantes para Aprendizado de Máquina Supervisionado no Processo de Mineração de Dados*

Antonio Rafael Sabino Parmezan[1]
Huei Diana Lee[1]
Newton Spolaôr[1,2]
Wu Feng Chung[1]

[1]Laboratório de Bioinformática – Centro de Engenharias e Ciências Exatas
Universidade Estadual do Oeste do Paraná
85867-900 – Foz do Iguaçu, PR, Brasil

[2]Laboratório de Inteligência Computacional – Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo
13560-970 – São Carlos, SP, Brasil
e-mail: {antoniorafaelparmezan, hueidianalee}@gmail.com

Resumo

Tarefas de pré-processamento em Mineração de Dados são de fundamental importância para assegurar a qualidade dos dados fornecidos como entrada para algoritmos de extração de padrões. Nesse contexto, a Seleção de Atributos é realizada, dentre outros motivos, para combinar e identificar os atributos mais significativos em um conjunto de dados. Neste trabalho é apresentado um estudo comparativo entre diferentes tipos de medidas de avaliação de importância de atributos pertencentes às categorias clássica, consistência e precisão. Para tanto, foram investigados alguns dos principais algoritmos de Seleção de Atributos que se baseiam em medidas representativas dessas categorias. Resultados experimentais demonstram que a Seleção de Atributos, por meio da redução da dimensionalidade, pode auxiliar na melhora da qualidade dos dados sob a perspectiva de desempenho preditivo, o que possivelmente contribui para a indução de modelos mais compreensíveis e com menor custo computacional.

Palavras-Chave: Pré-processamento de dados, Medidas de Importância de Atributos, Qualidade de Dados, Aprendizado de Máquina.

Dezembro 2012

*Trabalho desenvolvido com auxílio do Conselho Nacional de Desenvolvimento Científico e Tecnológico e da Universidade Estadual do Oeste do Paraná por meio do Programa Institucional de Bolsas de Iniciação Científica.

Sumário

Lista de Figuras v
Lista de Tabelas vii
Lista de Abreviaturas, Algoritmos e Variáveis viii

1 Introdução **1**

2 Seleção de Atributos **6**

3 Medidas de Importância **11**

4 Avaliação Experimental **14**
4.1 Descrição dos Conjuntos de Dados 14
4.2 Algoritmos Utilizados 17
4.3 Configuração dos Experimentos 22

5 Resultados e Discussão **24**
5.1 Análise dos Atributos Selecionados 24
5.2 Avaliação por Desempenho Preditivo e por Percentagem de Atributos Selecionados 26
5.3 Comparação Estatística dos Modelos Induzidos 31
5.4 Avaliação do Impacto das Abordagens Filtro e *Wrapper* em Relação ao Tempo de Aprendizado 34
5.5 Análise das Categorias de Medidas de Importância de Atributos . 36

6 Considerações Finais **39**

Referências Bibliográficas **46**

A Atributos Selecionados **47**

B Desempenho dos Algoritmos em Relação ao Modelo de Categorização **52**

C Tempo de Aprendizado **57**

Lista de Figuras

1.1 Processo de classificação por meio de um algoritmo de AM . . . 3
1.2 Processo de Mineração de Dados 4

2.1 Exemplo de espaço de estados de atributos considerando $M = 3$ 6
2.2 Seleção de um subconjunto de atributos 8
2.3 Abordagens para avaliação de subconjuntos de atributos 9

3.1 Hierarquia de categorias de medidas de importância 11
3.2 Hierarquia de categorias de medidas de importância considerando a separabilidade de classes/grupos 13

4.1 Configuração dos experimentos 23

5.1 Percentagem de atributos selecionados por cada um dos algoritmos de SA 26
5.2 Modelo de categorização de algoritmos em relação à percentagem de atributos selecionados e à média de erro dos classificadores induzidos 28
5.3 Modelo de categorização de algoritmos aplicado sobre o CD *BreastCancer* considerando o indutor *J48* 29
5.4 Tempo de aprendizado dos MC a partir dos CD Originais e dos subconjuntos de atributos selecionados pelos algoritmos de SA 35

B.1 Modelo de categorização de algoritmos aplicado sobre *Bupa* considerando o indutor *J48* 52
B.2 Modelo de categorização de algoritmos aplicado sobre *Hungarian* considerando o indutor *J48* 53
B.3 Modelo de categorização de algoritmos aplicado sobre *LungCancer* considerando o indutor *J48* 53
B.4 Modelo de categorização de algoritmos aplicado sobre *Pima* considerando o indutor *J48* 54
B.5 Modelo de categorização de algoritmos aplicado sobre *Bupa* considerando o indutor *MLP* 54
B.6 Modelo de categorização de algoritmos aplicado sobre *Hepatitis* considerando o indutor *MLP* 55
B.7 Modelo de categorização de algoritmos aplicado sobre *Hungarian* considerando o indutor *MLP* 55
B.8 Modelo de categorização de algoritmos aplicado sobre *LungCancer* considerando o indutor *MLP* 56

B.9 Modelo de categorização de algoritmos aplicado sobre *Pima* considerando o indutor *MLP* 56

Lista de Tabelas

1.1 Formato padrão do conjunto de exemplos 1

2.1 Combinações de estratégias e direções de busca 7

4.1 Resumo das características dos conjuntos de dados 17
4.2 Características dos algoritmos de SA 21

5.1 Quantidade de atributos selecionados por cada um dos algoritmos de SA e suas respectivas percentagens 25
5.2 Média de erro para cada CD e algoritmos considerados (MC utilizando o algoritmo *J48*) 27
5.3 Média de erro para cada CD e algoritmos considerados (MC utilizando o algoritmo *MLP*) 27
5.4 Categorização dos algoritmos de SA em relação à percentagem de atributos selecionados *versus* erro do MC (classificadores induzidos utilizando o algoritmo *J48*) 30
5.5 Categorização dos algoritmos de SA em relação à percentagem de atributos selecionados *versus* erro do MC (classificadores induzidos utilizando o algoritmo *MLP*) 31
5.6 Média do desempenho preditivo, desvio padrão e percentagem de redução de atributos para cada CD e algoritmos considerados (MC utilizando o indutor *J48*) 32
5.7 Média do desempenho preditivo, desvio padrão e percentagem de redução de atributos para cada CD e algoritmos considerados (MC utilizando o indutor *MLP*) 32
5.8 Comparação entre as medidas de avaliação de importância de atributos 37

A.1 Atributos selecionados — *BreastCancer* 47
A.2 Atributos selecionados — *Bupa* 48
A.3 Atributos selecionados — *Haberman* 48
A.4 Atributos selecionados — *Hepatitis* 48
A.5 Atributos selecionados — *Hungarian* 49
A.6 Atributos selecionados — *LungCancer* 50
A.7 Atributos selecionados — *Pima* 51

C.1 Relação entre a percentagem do número de atributos, o número de exemplos e o tempo de aprendizado dos MC utilizando o indutor *J48* 58

C.2 Relação entre a percentagem do número de atributos, o número de exemplos e o tempo de aprendizado dos MC utilizando o indutor *MLP* 58

Lista de Abreviaturas, Algoritmos e Variáveis

Abreviaturas

AM ..... Aprendizado de Máquina

CBF ..... *Consistency-Based Filter*

CD ..... Conjunto de Dados

CFS ..... *Correlation-based Feature Selection*

CSE ..... *Classifier Subset Evaluator*

d.e.s ..... Diferença estatisticamente significativa

ECM ..... Erro da Classe Majoritária

ESA ..... Erro sem Seleção Atributos

InfoGain ..... *Information-Gain Attribute Ranking*

J48 ..... Algoritmo de indução de árvores de decisão *J48*

MC ..... Modelo Construído

MD ..... Mineração de Dados

MLP ..... *Multilayer Perceptron*

Original ..... CD sem Seleção de Atributos

QL ..... Atributos Qualitativos

QT ..... Atributos Quantitativos

RD ..... Redução da Dimensionalidade

ReliefF ..... Algoritmo de SA *ReliefF*

RS ..... Revisão Sistemática

SA ..... Seleção de Atributos

TA ..... Transformação de Atributos

TM ..... Taxa Média

Algoritmos

CBF 18

CFS 18

CSE 19

InfoGain 19

J48 21

MLP 22

ReliefF 20

Variáveis

A ..... Atributo

C ..... Classe

D ..... CD no formato atributo-valor

M ..... Número de atributos

m ..... Parâmetro do algoritmo *ReliefF*

max_tries ..... Número máximo de iterações

N ..... Número de exemplos

t ..... Limiar de SA

Capítulo 1

Introdução

A difusão de sistemas computacionais nas mais variadas áreas do conhecimento tem contribuído para a geração e o armazenamento de uma quantidade crescente de dados. Esses dados são usualmente organizados em bases de dados computacionais, cuja qualidade afeta diretamente o sucesso da aplicação de processos que visam auxiliar em extração de conhecimento e análise inteligente de dados (Parmezan, 2012; Han et al., 2011; Liu and Motoda, 2008).

Diversos métodos de Aprendizado de Máquina (AM) têm sido propostos para o processamento dessas bases de dados e, consequentemente, para a construção de modelos que permitam a representação de novos conhecimentos adquiridos automaticamente, de modo mais significativo, tanto do ponto de vista de desempenho quanto de compreensibilidade. Entretanto, fatores como a maldição da dimensionalidade[1] podem dificultar a aplicação direta desses métodos (Nogueira, 2009; Alpaydin, 2004; Batista, 2003). Nesse contexto, tarefas de pré-processamento para Redução da Dimensionalidade (RD), isto é, para localizar subconjuntos de menor dimensão, dentro de dados de alta dimensão, podem constituir importante auxílio no processo de descoberta de conhecimento a partir de bases de dados (Liu and Motoda, 1998).

Uma das formas mais simples e utilizadas para a representação de dados é realizada por meio de atributos e seus respectivos valores e é denominada de atributo-valor. Esse formato consiste na descrição de exemplos por atributos, de modo que cada exemplo equivale a uma linha e cada atributo a uma coluna, como ilustrado na Tabela 1.1.

	Atributos				
Exemplos	A_1	A_2	$\ldots$	A_M	Classe (C)
E_1	a_{11}	a_{12}	$\ldots$	a_{1M}	c_1
E_2	a_{21}	a_{22}	$\ldots$	a_{2M}	c_2
E_3	a_{31}	a_{32}	$\ldots$	a_{3M}	c_3
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
E_N	a_{N1}	a_{N2}	$\ldots$	a_{NM}	c_N

Tabela 1.1: Formato padrão do conjunto de exemplos.

[1] A maldição da dimensionalidade faz referência ao aumento exponencial da complexidade de um determinado problema devido ao aumento de sua dimensionalidade. Neste trabalho o termo dimensionalidade refere-se ao número de atributos utilizado para representar um conjunto de exemplos.

Nessa linguagem de descrição de exemplos o tamanho de um Conjunto de Dados (CD) pode ser medido em relação a duas dimensões: número de atributos (M) e número de exemplos (N). Em determinados CD há ainda um atributo especial, denominado classe[2], que representa um conceito a ser aprendido por métodos de AM.

Os atributos podem assumir valores quantitativos e qualitativos e as operações possíveis sobre eles incluem Transformação de Atributos (TA) e Seleção de Atributos (SA). A TA abrange as tarefas de construção, extração e discretização de atributos, enquanto na SA o objetivo é determinar um subconjunto de atributos ou um conjunto ordenado de atributos, ambos, segundo algum critério ou medida de importância (Parmezan, 2012; Liu and Motoda, 2008; Lee et al., 2006).

Como mencionado, uma das preocupações da área de AM consiste em pesquisar e desenvolver métodos computacionais que permitam adquirir, de maneira automática, novos conhecimentos, novas habilidades e novas técnicas de organizar o conhecimento existente (Mitchell, 1997).

Um dos paradigmas de aprendizado comumente utilizados para extrair conhecimento e predizer eventos futuros é a inferência indutiva. Esse paradigma permite obter conclusões realizando inferências a partir de um conjunto de exemplos conhecidos, isto é, fatos observados. A inferência indutiva é um dos recursos mais utilizados pelo cérebro humano para derivar conhecimento novo. Porém, deve ser utilizada com cautela, pois se o número de exemplos for insuficiente ou não forem representativos, as hipóteses obtidas podem ser de pouco valor. A hierarquia do aprendizado indutivo, de acordo com o critério do grau de supervisão presente nos dados, pode ser dividida em (Rezende, 2003): aprendizado não supervisionado, aprendizado semissupervisionado e aprendizado supervisionado.

Aprendizado Não Supervisionado: o objetivo consiste em analisar os exemplos e determinar se subconjuntos desses exemplos podem ser agrupados de acordo com suas características (atributos), tendo em vista que a classe não está definida para nenhum exemplo (Hall, 2000);

Aprendizado Semissupervisionado: apenas uma pequena parte do conjunto de exemplos está rotulado. Logo, a ideia é utilizar essa pequena parcela para auxiliar na determinação das classes de uma quantidade maior de exemplos e permitir que um CD rotulados de dimensão superior ao investigado possa ser construído (Matsubara and Monard, 2005);

Aprendizado Supervisionado: a entrada para um algoritmo de aprendizado supervisionado consiste usualmente de um conjunto de N exemplos (ou casos) de treinamento $\{(\boldsymbol{a}_1, c_1),...,(\boldsymbol{a}_N, c_N)\}$ rotulados com os valores c associados a uma função f desconhecida $c = f(\boldsymbol{a})$, onde os valores $\boldsymbol{a}_i$ são vetores da forma $< a_{i1}, a_{i2},..., a_{iM} >$ cujos componentes são valores discretos ou contínuos relacionados aos M atributos $A = \{A_1,A_2,...,A_M\}$. Ou seja, a_{ij} denota o valor do atributo A_j para o exemplo i (Alpaydin, 2004). Dado esse conjunto de exemplos de treinamento, o algoritmo induz uma hipótese $\boldsymbol{h}$ que deve aproximar a verdadeira função f, tal que dados os

[2]Neste trabalho os termos classe, rótulo e meta são utilizados indistintamente

valores a de um novo exemplo, $\boldsymbol{h}$ prediz o valor c correspondente. Desse modo, cada exemplo é associado a uma classe (rótulo), que pode ser contínua, sendo nesse caso o processo denominado de regressão, ou discreta, denominado de classificação e tratado neste trabalho.

Na Figura 1.1 são esquematizadas as principais fases de um algoritmo de AM para classificação.

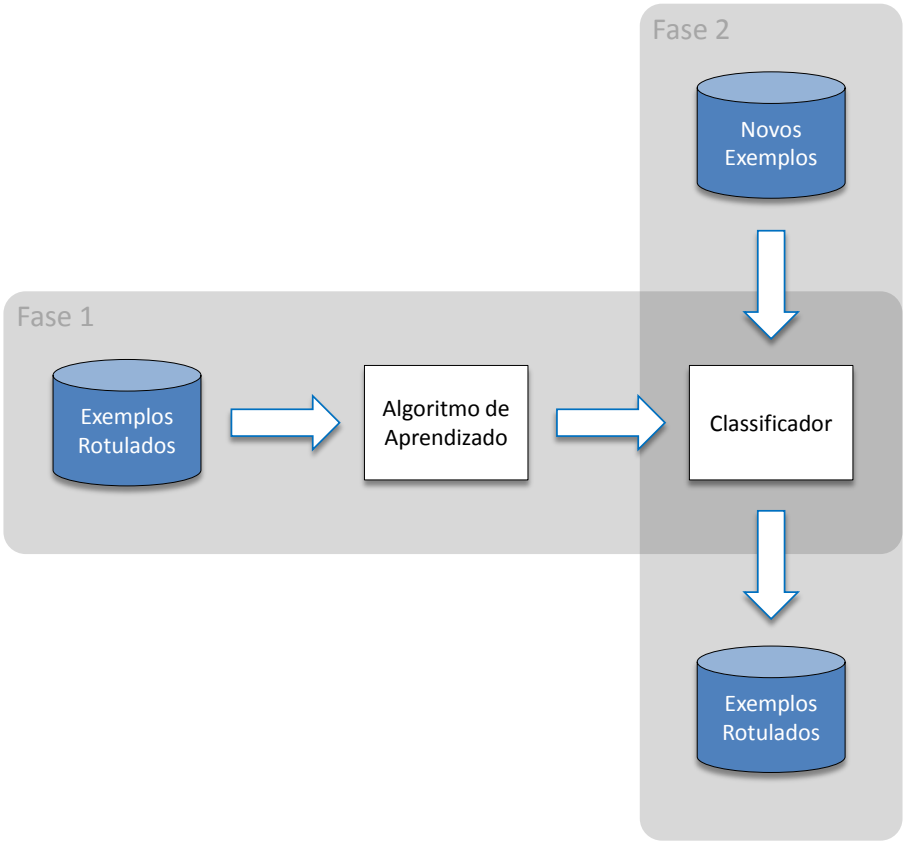


Figura 1.1: Processo de classificação por meio de um algoritmo de AM (Parmezan, 2012).

Na primeira fase os exemplos rotulados (exemplos de treinamento) são fornecidos à um indutor, o qual extrai o conhecimento desses exemplos e gera um classificador representado em uma estrutura interna (modelo). Na segunda fase o classificador gerado é utilizado para rotular novos exemplos (exemplos de teste). Desse modo, o classificador tem habilidade de predizer tanto a classe dos exemplos utilizados em sua construção, quanto a classe de novos exemplos (Pila, 2001).

Um dos métodos que pode auxiliar na avaliação de modelos é o de validação cruzada. Esse método consiste em dividir aleatoriamente o CD em x partes, das quais $x - 1$ são utilizadas para o treinamento e uma serve como base de testes. O processo é repetido x vezes, sendo que cada parte é utilizada uma única vez como conjunto de testes. Como a base de testes é previamente rotulada pode-se estimar a taxa de acerto do modelo induzido comparando-se o resultado obtido com a rotulação disponível na base (Alpaydin, 2004). A validação cruzada é comumente utilizada para obter uma aproximação do erro verdadeiro do Modelo Construído (MC) quanto à capacidade de predição.

Dentre as medidas comumente empregadas para avaliar a qualidade da predição de um modelo de classificação, destacam-se o desempenho preditivo e a acurácia (Han et al., 2011; Lee and Giraud-Carrier, 2008).

O desempenho preditivo com que o classificador prediz a classe dos exemplos está relacionado a sua representação interna utilizando os atributos. Se os atributos não são capazes de representar o conhecimento implícito nos exemplos, o desempenho preditivo do MC é deteriorado, ou seja, haverá uma grande chance da classe de novos exemplos submetidos ao classificador ser predita incorretamente (Parmezan, 2012).

Ainda, o número de atributos utilizados para representar os exemplos é um dos fatores que pode influenciar diretamente no desempenho preditivo do modelo induzido, haja vista que alguns algoritmos de aprendizado não trabalham bem na presença de grande quantidade de atributos (Liu and Motoda, 2008; Lee and Monard, 2003).

Problemas como os descritos podem dificultar a extração de conhecimento a partir de informações contidas em uma base de dados. Desse modo, é necessário que os dados sejam representados e processados de maneira apropriada e o MC seja avaliado e validado (Han et al., 2011; Lee, 2005; Witten and Frank, 2005). Uma das maneiras de se alcançar esse objetivo é por meio da realização do processo de Mineração de Dados (MD), o qual pode ser divido em três fases, como ilustrado na Figura 1.2.

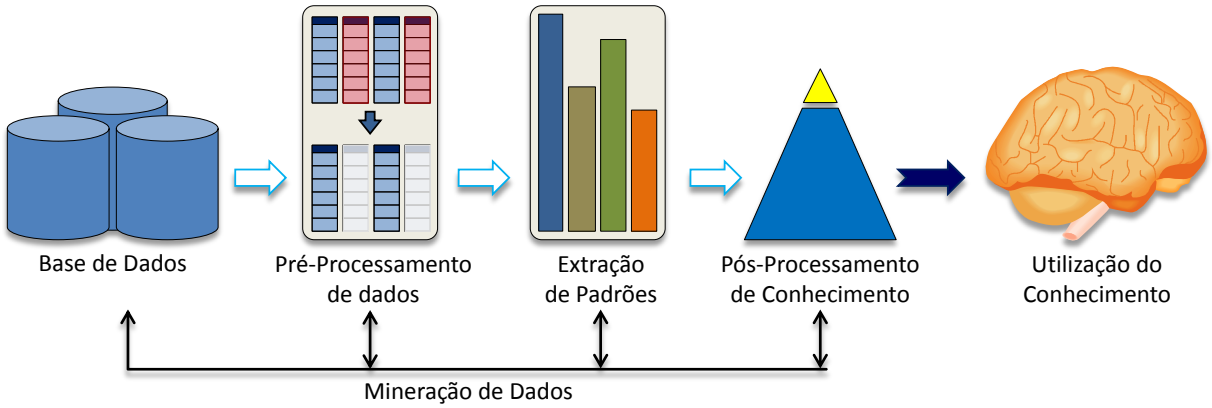


Figura 1.2: Processo de Mineração de Dados (Parmezan, 2012).

A primeira fase, pré-processamento, possui essencialmente dois objetivos: conhecer o domínio da aplicação e a base de dados e prepará-los para a próxima-fase. Dentre as diversas tarefas que podem ser realizadas nessa fase destacam-se: preparação dos dados, limpeza dos dados, transformação de dados e atributos e SA (Lee, 2005; Batista, 2003).

A segunda fase, extração de padrões, tem como objetivo a construção de modelos que possam representar o conhecimento embutido nos dados (Han et al., 2011; Witten and Frank, 2005). Essa fase é apoiada por diversas áreas, entre as quais AM, base de dados, visualização e estatística.

Na última fase, pós-processamento de conhecimento, o objetivo é avaliar, validar e consolidar o conhecimento extraído. A avaliação é realizada, por exemplo, com a interpretação dos resultados por meio de visualização dos padrões extraídos, remoção de padrões irrelevantes ou redundantes e tradu-

ção de padrões úteis para formas compreensíveis para os usuários. Durante essa fase, os resultados devem ser avaliados para garantir que estes sejam estatisticamente significativos e confiáveis. O conhecimento extraído deve ser também validado com relação ao conhecimento prévio do domínio, com o auxílio de especialistas da área, para que possíveis conflitos sejam removidos e o conhecimento seja consolidado (Lee, 2005).

A fase de pré-processamento de dados, considerada como uma das mais custosas por consumir aproximadamente 80% de todo o processo, é de fundamental importância para assegurar que os dados sejam de qualidade (Pyle, 1999). Nesse contexto, tarefas de pré-processamento, tal como a SA, podem auxiliar na melhoria do desempenho de ferramentas de análise de dados, bem como na simplificação da linguagem de descrição de exemplos quando esta possuir mais atributos que os necessários (Lee, 2005).

A tarefa de SA é realizada, dentre outros motivos, para combinar e identificar os atributos mais significativos em um CD. Desse modo, a seleção de um subconjunto de atributos está diretamente ligada à importância desses atributos para o domínio e aos objetivos pretendidos com o processo de extração de conhecimento a partir de bases de dados, assim como à medida ou critério utilizado para avaliar essa importância (Liu and Motoda, 2008).

A realização de pesquisas quanto à avaliação de medidas de importância e métodos para a combinação de atributos é de grande importância para a determinação de abordagens que possam guiar a tarefa de SA. Consequentemente, é possível obter dados reduzidos para a construção de modelos com menor complexidade (por exemplo, realizando induções em menos tempo) ou mais compreensíveis ao usuário final e com qualidade potencialmente superior ou semelhante a modelos obtidos considerando todos os atributos. Adicionalmente, em domínios com um elevado número de atributos, pode-se contribuir com a redução dos efeitos associados à maldição da dimensionalidade (Parmezan, 2012).

Neste trabalho, é apresentado um estudo comparativo entre diferentes tipos de medidas de importância de atributos pertencentes às categorias: clássica (informação, correlação e distância), consistência e precisão. Para tanto, foram investigados alguns dos principais algoritmos de SA que se baseiam em medidas representativas dessas categorias. Resultados experimentais obtidos com diversos CD demonstraram que a SA, por meio da RD, pode auxiliar na melhora da qualidade dos dados sob a perspectiva de desempenho preditivo, o que possivelmente contribui para a construção de modelos mais compreensíveis e com menor custo computacional. Ainda neste trabalho, a média do erro dos MC foi relacionado com a percentagem de redução da quantidade de atributos selecionados por cada algoritmo de SA, através de um modelo proposto em Lee (2005) que avalia essas medidas conjuntamente.

O restante deste trabalho está organizado do seguinte modo: nos Capítulos 2 e 3 são apresentados brevemente conceitos sobre Seleção de Atributos e medidas de importância, respectivamente. No Capítulo 4 são descritos os Conjuntos de Dados e os algoritmos utilizados, bem como a configuração dos experimentos realizados. No Capítulo 5 são exibidos os resultados e discussão dos experimentos. Por fim, considerações finais são apresentadas no Capítulo 6.

Capítulo 2

Seleção de Atributos

A SA pode ser definida como a determinação de um subconjunto ótimo de atributos, segundo algum critério ou medida de importância, que representa a informação importante contida nos dados. Em outras palavras, é um processo que pode escolher um subconjunto de P atributos a partir do conjunto original com M atributos, de modo que $P \leq M$ (Lee, 2005; Liu and Motoda, 1998).

A SA pode ser formulada como um processo de busca, visto que para cada CD com M atributos, existem 2^M subconjuntos de atributos candidatos, em que cada subconjunto é um estado no espaço de busca (Langley, 1994). Na Figura 2.1 é ilustrado o espaço de busca para três atributos.

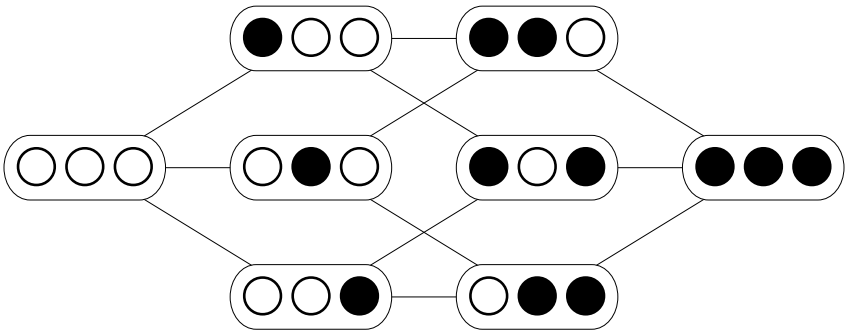


Figura 2.1: Exemplo de espaço de estados de atributos considerando $M = 3$ (Parmezan, 2012).

Observa-se que existe uma ordenação parcial dos estados, pois cada um deles possui um atributo a mais que o estado anterior, sendo o estado inicial à esquerda estabelecido como vazio e o estado final à direita definido como completo.

A busca realizada por um algoritmo de SA pode ser instanciada em relação a basicamente quatro questões: (1) sentido de busca, (2) estratégia de busca, (3) definição do critério de parada e de (4) critérios de avaliação de importância de atributos.

A primeira questão influencia na determinação do ponto de partida (ou pontos de partida) no espaço de busca, isto é, na direção em que a busca será realizada e os operadores que serão utilizados para acessar os estados sucessores. As abordagens para a direção da busca são divididas em (Liu and Motoda, 2008):

- *Forward Selection*: nessa abordagem o estado inicial é estabelecido como vazio (subconjunto vazio de atributos), e os atributos são incluídos um por vez;
- *Backward Elimination*: nessa abordagem o ponto de partida é iniciado com o conjunto de todos os atributos (completo), os quais são removidos sucessivamente;
- *Bidirectional Search*: nessa abordagem duas buscas são processadas simultaneamente, as quais terminam quando ambas atingem o centro do espaço de busca, ou quando uma das buscas encontra os melhores atributos antes de alcançar o centro do espaço de busca;
- *Random Search*: o objetivo dessa abordagem é evitar que a busca fique restrita a ótimos locais. Portanto, não há uma direção específica na qual a busca será realizada, haja vista que o ponto de partida da busca e o modo de adicionar ou remover atributos são decididos aleatoriamente.

A respeito da segunda questão, existem três diferentes táticas para lidar com o problema da estratégia de busca sem recorrer à busca exaustiva: busca completa, busca heurística e busca não-determinística (Liu and Yu, 2002; Schlimmer, 1993). Na primeira estratégia nenhum subconjunto ótimo de atributos é perdido, embora nem todo subconjunto de atributos tenha que ser avaliado. Já a segunda estratégia emprega algum tipo de heurística para conduzir a busca, ou seja, utiliza conhecimento específico do problema para encontrar uma solução de modo mais eficiente em relação à busca não informada[1]. No entanto, há o risco de não encontrar os subconjuntos ótimos. A terceira estratégia, diferentemente das anteriores, procura pelos subconjuntos de atributos aleatoriamente. Desse modo, o subconjunto corrente não aumenta ou diminui, diretamente, a partir de algum subconjunto anterior segundo uma regra determinística.

As possíveis combinações entre estratégias e direções de busca são resumidas na Tabela 2.1 (Liu and Motoda, 1998).

	Estratégias de Busca		
Direções de Busca	Completa	Heurística	Não-determinística
Forward Selection	✔	✔	✗
Backward Elimination	✔	✔	✗
Bidirectional Search	✔	✔	✗
Random Search	✗	✔	✔

Tabela 2.1: Combinações de estratégias e direções de busca (modificado de Liu and Motoda (1998)).

Em relação à terceira questão, a criação de subconjuntos de atributos que deterioram ou não melhoram o desempenho preditivo de classificação pode ser adotada como critério de parada da busca. Outra possibilidade, aplicável a determinados algoritmos de SA, inclui parar a busca quando uma quantidade

[1]A busca não informada, também denominada de busca cega, trabalha somente com as informações necessárias para distinguir o estado alvo de um estado não alvo.

de atributos definida *a priori* é atingida. Uma quantidade de iterações também pode ser estabelecida com o intuito de finalizar esse processo.

Uma das maneiras de realizar SA é por meio da minimização/maximização de múltiplas medidas de importância, como objetivos, para a obtenção de soluções aproximadamente ótimas (Abeel et al., 2010; Spolaôr, 2010; Wang and Huang, 2009; Liu and Motoda, 2008; Lee et al., 2006). Nesse contexto, os algoritmos genéticos multiobjetivo tentam produzir, por aproximações sucessivas, as melhores soluções para um determinado problema. Com isso, é possível evitar a computação de todas as soluções admissíveis e prover auxílio para a otimização de medidas de importância. Embora esse tema não seja tratado neste trabalho, pesquisas demonstram o interesse crescente no uso de meta-heurísticas como os algoritmos genéticos multiobjetivo na SA monorrótulo (Spolaôr et al., 2010).

Associados à quarta questão, os métodos de SA selecionam os atributos pela avaliação individual ou pela avaliação de subconjuntos de atributos.

No caso de avaliação individual, a qual também é denominada de ordenação ou *ranking*, os atributos são avaliados individualmente e ordenados de acordo com algum critério de importância. Desse modo, os atributos que estiverem melhor posicionados podem ser selecionados. Diferentemente, na avaliação de subconjunto ocorre a busca por subconjuntos mínimos de atributos segundo algum critério de importância (Figura 2.2).

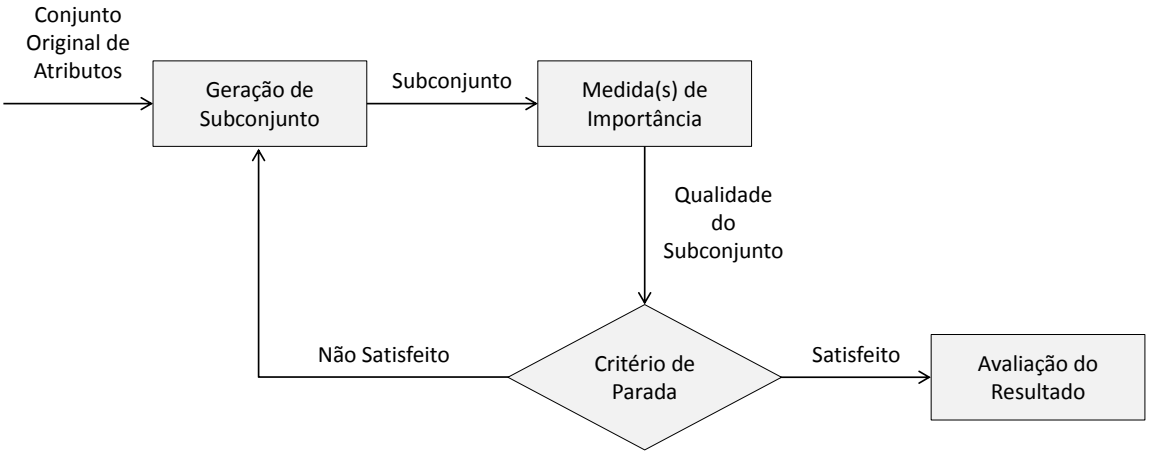


Figura 2.2: Seleção de um subconjunto de atributos (modificado de Liu and Motoda (1998)).

Os métodos de SA por avaliação individual geralmente removem atributos pouco ou descorrelacionados com a classe (atributos irrelevantes), pois espera-se que atributos relevantes com valores correlacionados (atributos redundantes), tenham a mesma importância na discriminação de classes. Contudo, métodos que avaliam subconjuntos de atributos podem remover tanto atributos irrelevantes quanto redundantes (Liu and Motoda, 2008). Outra característica específica da SA baseada em avaliação individual é que um limiar é necessário para definir quais são os atributos a serem selecionados.

Os métodos que realizam a seleção de subconjuntos de atributos podem ser agrupados nas abordagens filtro, *wrapper* e embutida, dependendo de como o algoritmo de SA se relaciona com os algoritmos de aprendizado (extração de

padrões) (Kohavi and John, 1997). Na Figura 2.3 são apresentadas essas três abordagens considerando a participação do algoritmo de aprendizado (Covões, 2010; Pila, 2001).

(a) Abordagem filtro (b) Abordagem *wrapper* (c) Abordagem embutida

Figura 2.3: Abordagens para avaliação de subconjuntos de atributos (modificado de Covões (2010)).

Abordagem Filtro: nessa abordagem o processo de SA ocorre antes da construção de modelos propriamente dita. A ideia é filtrar atributos não importantes segundo algum critério (Baranauskas, 2001; Blum and Langley, 1997; John et al., 1994). Os critérios utilizados refletem características gerais dos CD para selecionar atributos significativos. Sendo assim, métodos aplicados conforme a abordagem filtro são independentes do algoritmo de aprendizado, o qual, simplesmente, recebe como entrada o conjunto de exemplos descrito utilizando somente o subconjunto de atributos importantes identificado pela SA;

Abordagem *Wrapper*: nessa abordagem o processo de SA também ocorre externamente ao algoritmo de construção de modelos, porém utilizando tal algoritmo como uma caixa preta para analisar, a cada iteração, o subconjunto de atributos em questão. Em outras palavras, métodos aplicados conforme a abordagem *wrapper* geram um subconjunto candidato de atributos, executam o algoritmo de construção de modelos considerando os dados representados pelo subconjunto selecionado do conjunto de treinamento, e utilizam o desempenho preditivo resultante do classificador induzido como medida de importância para avaliar o subconjunto de atributos investigado. Esse processo é repetido para cada subconjunto de atributos até que um critério de parada determinado seja satisfeito (Dash and Liu, 1997; Kohavi and John, 1997). Usualmente os subconjuntos de atributos selecionados a partir dessa abordagem tendem a resultar em melhor desempenho no modelo induzido do que os subconjuntos selecionados por meio da abordagem filtro. Isso se deve ao fato de que a indução do modelo por um determinado algoritmo de classificação segue o viés (*bias*) do mesmo algoritmo de aprendizado utilizado durante o processo de SA. Em outras palavras, os atributos selecionados são supostamente ótimos, em termos de desempenho preditivo, para modelos gerados pelo indutor considerado. Entretanto, a abordagem filtro possui, em geral, um menor custo computacional do que a estratégia

wrapper, pois não requer a construção de um novo classificador para cada subconjunto de atributos a ser avaliado;

Abordagem Embutida: nessa abordagem a tarefa de SA é realizada internamente pelo próprio algoritmo de extração de padrões. Portanto, métodos aplicados conforme a abordagem embutida selecionam o subconjunto de atributos no processo de construção do modelo de classificação, durante a fase de treinamento, e geralmente são específicos para um dado algoritmo de aprendizado.

Com relação aos critérios de avaliação de importância de atributos, diversas medidas têm sido propostas na literatura para combinar e identificar os atributos mais significativos em um CD (Parmezan, 2012; Parmezan et al., 2011a,b; Covões, 2010; Parmezan et al., 2010; Spolaôr, 2010; Liu and Motoda, 2008; Lee, 2005; Liu et al., 2004; Yu and Liu, 2004; Dash and Liu, 2003; Guyon and Elisseeff, 2003; Lee and Monard, 2003; Liu et al., 2003; Sousa et al., 2002; Pila, 2001; Dash and Liu, 2000; Traina et al., 2000; Liu and Motoda, 1998). No próximo capítulo são apresentadas algumas das medidas estudadas, bem como definições que auxiliam a determinar em relação a quê um atributo é considerado importante.

Capítulo 3

Medidas de Importância

Existem diferentes definições na literatura para determinar a importância de um atributo. Algumas definições consideram a importância do atributo em relação à classe, enquanto outras trabalham com a SA não supervisionada (Spolaôr et al., 2011b; Nogueira, 2009; He et al., 2005; Mitra et al., 2002; Dy and Brodley, 2000; Talavera, 1999). Essa necessidade de estimativa da importância de atributos é comum, tanto à avaliação individual quanto à avaliação de subconjuntos de atributos. Por meio das medidas de importância é possível, por exemplo, avaliar se os atributos selecionados auxiliam a melhorar o desempenho preditivo do classificador, ou auxiliam a simplificar o modelo construído de modo que ele seja mais compreensível.

As medidas de avaliação de importância de atributos podem ser organizadas em três principais categorias: (1) clássica, a qual inclui medidas de informação, correlação e distância, (2) consistência e (3) precisão (Figura 3.1).

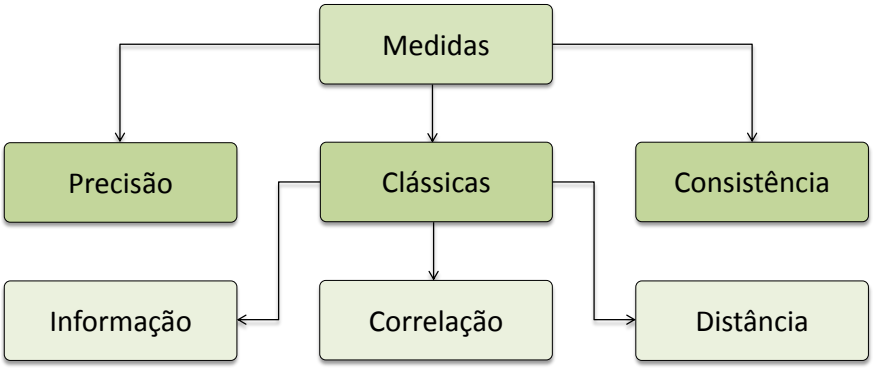


Figura 3.1: Hierarquia de categorias de medidas de importância (Parmezan, 2012).

Quando um atributo A é removido e a medida de importância aplicada nos dados representados pelos atributos restantes é deteriorada, pode-se considerar que A é importante (Liu and Motoda, 1998). Com base nessa definição, a seguir são descritos os principais tipos de medidas de importância que compõem as categorias apresentadas.

Medidas de Informação: permitem determinar o ganho de informação a partir de um atributo, isto é, determinar a diferença entre a incerteza *a priori* e a incerteza *a posteriori* associada ao atributo investigado (Yu and Liu,

2004). Desse modo, um atributo pode ser definido como sendo mais importante que outro quando implica em um maior ganho de informação. Um exemplo usual de medida de informação é a entropia (Mitra et al., 2002);

Medidas de Correlação: também conhecidas como medidas de dependência ou associação. Permitem quantificar a habilidade de predizer o valor de um atributo a partir do valor de outro atributo (Hall, 2000, 1999). Em termos de correlação entre atributos, é possível estimar a redundância de um CD com o auxílio de medidas como a dimensão fractal (Lee et al., 2006), a qual é aplicável a CD rotulados ou não rotulados. Uma medida clássica de dependência é o coeficiente de correlação, o qual pode, por exemplo, ser utilizado em um problema de classificação para encontrar a correlação entre um atributo e a classe;

Medidas de Distância: também denominadas de medidas de separabilidade, divergência ou discriminação. Em SA supervisionada, essas medidas consideram como atributos mais importantes aqueles que possibilitam a separação de exemplos de rótulos distintos e a não separação de exemplos da mesma classe (Kononenko, 1994; Robnik-Sikonja and Kononenko, 2003). Dentre essas medidas, destacam-se às de distância interclasse e intraclasse, as quais estimam respectivamente a separabilidade existente entre as classes na projeção correspondente ao subconjunto de atributos investigado e a densidade dos exemplos dentro de uma classe específica;

Medidas de Consistência: possuem características diferentes em relação às outras medidas, uma vez que são fortemente dependentes do conjunto de treinamento. A utilização dessas medidas permite encontrar um subconjunto mínimo de atributos que satisfaz a proporção de inconsistência aceita, geralmente definida pelo usuário. O conceito de consistência corresponde, em dados rotulados, à não ocorrência de um ou mais pares de exemplos com valores idênticos em cada atributo e rótulos distintos (Arauzo-Azofra et al., 2008; Dash and Liu, 2003). Desse modo, o intuito da análise por consistência é possibilitar a construção de hipóteses lógicas consistentes em um conjunto de treinamento. É importante observar que as medidas de consistência não detectam a ocorrência de atributos redundantes, pois não possibilitam a distinção entre dois atributos igualmente bons;

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 (desempenho preditivo) proporcionar ao modelo gerado será selecionado (Kohavi and John, 1997). Sendo assim, é usual a utilização do mesmo algoritmo tanto para realizar a tarefa de SA quanto para processar o conjunto de exemplos com os atributos selecionados.

Como esquematizado na Figura 3.2, as medidas clássicas e de consistência podem ainda ser agrupadas, visto que tratam da separabilidade de classes no caso de aprendizado supervisionado, ou *clusters* no caso de aprendizado não supervisionado.

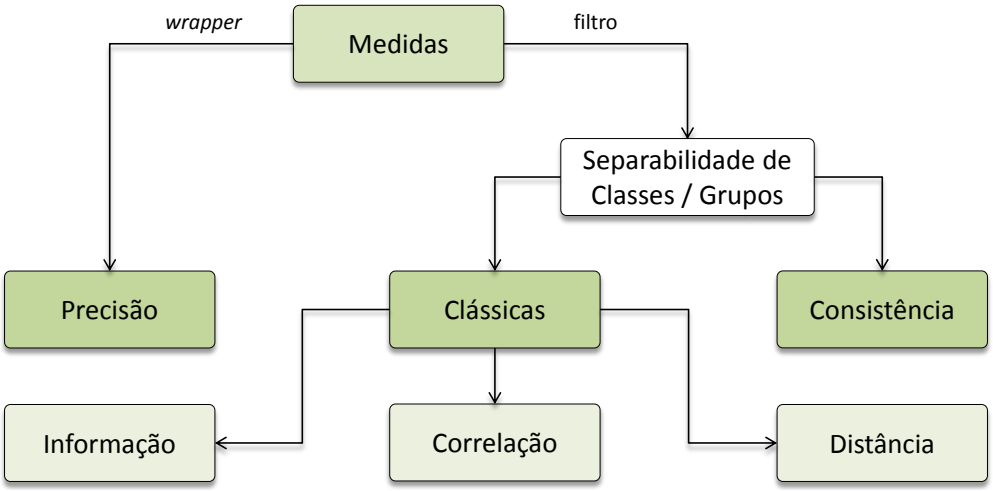


Figura 3.2: Hierarquia de categorias de medidas de importância considerando a separabilidade de classes/grupos (modificado de Liu and Motoda (1998)).

É importante ressaltar que as medidas de precisão são, geralmente, dependentes do algoritmo de aprendizado considerado, pois os subconjuntos de atributos selecionados são importantes em relação ao desempenho preditivo do modelo induzido por um determinado algoritmo. Diferentemente, as medidas de separabilidade de classes/grupos são independentes do algoritmo de aprendizado escolhido para a posterior construção do modelo.

No próximo capítulo descreve-se o procedimento de avaliação experimental, assim como os CD identificados na literatura relacionada e os métodos de SA fundamentados nas medidas de importância apresentadas.

Capítulo 4

Avaliação Experimental

Pode-se pensar que quanto maior o número de atributos em um CD, maior o poder discriminatório do classificador e a facilidade de construir modelos representativos de conhecimento dos dados. Contudo, o mundo real apresenta provas de que nem sempre essa hipótese pode ser aceita como verdadeira. Isso porque, como mencionado, distintos métodos de indução sofrem da maldição da dimensionalidade, a qual pode ser amenizada com o emprego de técnicas para SA importantes.

Porém, ao mesmo tempo em que diversos algoritmos estão disponíveis para a realização da tarefa de SA, cresce a dificuldade em se determinar, *a priori*, qual ou quais desses algoritmos seriam mais apropriados, de acordo com as características do problema (CD) e as características dos próprios algoritmos de SA (Parmezan, 2012). O mesmo ocorre quando se trata de métodos para construção de modelos (Han et al., 2011).

Atrelado à isso está a questão de que medidas considerar para constatar quais algoritmos de aprendizado apresentaram uma performance melhor que outros. Dentre essas medidas, o desempenho preditivo do modelo induzido é o mais frequentemente utilizado. No entanto, em algumas aplicações pode ser de interesse determinar uma solução satisfatória em múltiplos aspectos (otimização multiobjetivo), de modo que várias medidas de avaliação sejam necessárias.

Por esses e outros motivos, avaliações experimentais constituem um importante instrumento da estimativa de que ou quais algoritmos seriam mais apropriados perante um determinado problema (Alpaydin, 2004).

4.1 Descrição dos Conjuntos de Dados

Os CD utilizados para a realização dos experimentos apresentados neste trabalho foram selecionados a partir do resultado de uma Revisão Sistemática (RS) (Kitchenham, 2007) de trabalhos publicados na área de Análise Inteligente de Dados, mais precisamente, descoberta de conhecimento, MD e AM.

A RS consiste basicamente em um método para pesquisa bibliográfica que permite a resolução de questões de pesquisa por meio de procedimentos explícitos para a identificação, seleção e avaliação de publicações (Spolaôr et al., 2012, 2010). Esse processo é realizado com o intuito de explorar trabalhos re-

levantes e avaliar o tema pesquisado de modo justo, rigoroso e replicável. Uma RS pode ainda incluir uma meta-análise, a qual corresponde à uma síntese dos resultados relatados por meio de técnicas estatísticas.

A seguir são descritas algumas das principais questões que deram suporte à RS realizada em meados de Fevereiro de 2011, considerando os motores de busca Portal *ACM*[1], *Scopus*[2] e *CiteSeerX*[3].

- Questão de pesquisa: quais CD biológicos de domínio público, no formato atributo-valor, são frequentemente referenciados na literatura para ilustrar problemas de classificação em Análise Inteligente de Dados?;
- Expressão de busca: (*"biological data" OR "medical data" OR "biological dataset" OR "biological datasets" OR "medical dataset" OR "medical datasets" OR "biological database" OR "biological databases" OR "medical database" OR "medical databases"*) *AND* (*"classifier" OR "classification" OR "inductor" OR "induction" OR "model" OR "models"*) *AND* (*"intelligent data analysis" OR "knowledge discovery in databases" OR "data mining" OR "machine learning" OR "supervised learning" OR "bioinformatics"*);
- Critérios de exclusão utilizados:
  - Publicações com mesmo título indexadas e identificadas simultaneamente por distintos motores de busca;
  - Publicações hospedadas em serviços de acesso restrito não subsidiados pela UNIOESTE;
  - Publicações que são duplicadas em relação aos resultados, com exceção da versão mais completa;
  - Publicações com uma página, pôsteres e apresentações;
  - Publicações que não atendem às questões de pesquisa ou que utilizam CD de expressão gênica;
  - Teses, dissertações, tutoriais e editoriais.
- Critérios de seleção adotados:
  - Trabalhos que avaliam CD de acesso público;
  - Trabalhos que utilizam somente CD no formato atributo-valor para tarefa de classificação.

Houve também preocupação em encontrar CD que ilustrassem problemas diversos, em situações ótimas ou desfavoráveis, isto é, com uma baixa ou alta quantidade de atributos para uma amostra pequena ou grande de dados. Desse modo, critérios de qualidade como o número de atributos, o número de exemplos e a distribuição da classe também foram levados em consideração tanto para a escolha dos CD, quanto para a seleção dos trabalhos que descrevem esses conjuntos.

[1] http://portal.acm.org.
[2] http://www.scopus.com.
[3] http://citeseerx.ist.psu.edu.

É importante ressaltar que adaptações na expressão original de busca foram necessárias, visto que nem todos os motores de busca considerados suportam expressões extensas, bem como apresentam limitações quanto à investigação separada de títulos, resumos e palavras-chave.

Ao todo, foram identificadas 518 publicações, tendo esse número reduzido para 293, após a aplicação dos critérios de exclusão. Posteriormente, essas publicações foram analisadas de acordo com os critérios de seleção e de qualidade adotados. Ao final desse processo, obteve-se 12 trabalhos que auxiliaram na escolha dos CD.

A partir da RS foi possível observar que 7 CD naturais (obtidos a partir de repositórios de dados), do total de 32 identificados, são frequentemente referenciados pela comunidade acadêmica.

Todos os sete conjuntos selecionados estão brevemente descritos a seguir e constituem CD biológicos naturais de acesso público obtidos do repositório de dados UCI (Frank and Asuncion, 2010).

BreastCancer: 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;

Haberman: o problema é predizer se uma paciente morrerá dentro de cinco anos ou sobreviverá por mais tempo, após ser submetida a uma cirurgia de câncer de mama;

Hepatitis: o problema é predizer se um paciente com hepatite é terminal ou não tomando-se como base o histórico e dados clínicos;

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

LungCancer: ilustra o poder de um plano discriminante ótimo em situações desfavoráveis onde há somente uma amostra pequena de dados para uma alta quantidade de atributos, tendo em vista que os dados descrevem três tipos patológicos de câncer de pulmão;

Pima: o problema é predizer se uma paciente, mulher de descendência indígena (etnia Pima) com idade mínima de 21 anos, seria classificada como diabética segundo o critério estabelecido pela Organização Mundial de Saúde, a partir de dados clínicos e laboratoriais.

A Tabela 4.1 apresenta um resumo das características desses 7 CD organizadas do seguinte modo:

- #Exemplos: número de exemplos do conjunto de dados;
- #Atributos (QT + QL): número total de atributos seguido pelo número de atributos Quantitativos (QT) e Qualitativos (QL);

- Classes e %Classe: valores e distribuição das classes, respectivamente;
- Erro da Classe Majoritária (ECM): erro cometido no caso de novos exemplos serem classificados como sendo pertencentes à classe majoritária;
- ?: Existência ou não de valores desconhecidos.

Conjunto de Dados	**#Exemplos**	**#Atributos** (QT + QL)	**Classes**	**%Classe**	**ECM**	**?**
BreastCancer	286	9 (0 + 9)	1 2	70,28% 29,72%	29,72% sobre 1	Sim
Bupa	345	6 (6 + 0)	1 2	42,03% 57,97%	42,03% sobre 2	Não
Haberman	306	3 (2 + 1)	1 2	73,53% 26,47%	26,47% sobre 1	Não
Hepatitis	155	19 (6 + 13)	1 2	20,65% 79,35%	20,65% sobre 2	Sim
Hungarian	294	13 (6 + 7)	0 1	63,95% 36,05%	36,05% sobre 0	Sim
LungCancer	32	56 (0 + 56)	1 2 3	28,13% 40,62% 31,25%	59,38% sobre 2	Sim
Pima	768	8 (8 + 0)	1 2	65,02% 34,98%	34,98% sobre 1	Não

Tabela 4.1: Resumo das características dos conjuntos de dados.

4.2 Algoritmos Utilizados

Os experimentos apresentados neste trabalho foram realizados utilizando cinco algoritmos para a tarefa de SA, os quais são baseados nas principais medidas de importância de atributos dispostas nas categorias clássica, consistência e precisão, e dois algoritmos para indução de modelos.

Todos esses algoritmos, descritos a seguir, são comumente empregados pela comunidade acadêmica e encontram-se implementados no ambiente computacional *Weka*[4] (Witten and Frank, 2005), o qual atualmente é uma das principais ferramentas utilizadas para a tarefa de MD. Além de disponibilizar uma coleção de pacotes com algoritmos de AM, *Weka* é desenvolvido em linguagem de programação *Java* (Deitel and Deitel, 2010), a qual é caracterizada por sua simplicidade e eficiência, além de oferecer acesso livre ao código. Essas propriedades, conjuntamente com a portabilidade e confiabilidade, contribuíram na escolha desse ambiente para a realização deste trabalho.

***CBF*:** o algoritmo *Consistency-Based Filter* (*CBF*) (Dash and Liu, 2003) tem como parâmetros de entrada o conjunto original de dados, o número de atributos do conjunto original, um limiar aceitável de inconsistência e o número

[4] http://www.cs.waikato.ac.nz/ml/weka.

máximo de tentativas para a geração de subconjuntos de atributos, denominado max_tries. A ideia básica desse método, conforme apresentado no Algoritmo 1, é gerar max_tries subconjuntos de atributos e avaliá-los quanto ao seu tamanho e a sua inconsistência em relação à classe. O subconjunto selecionado será aquele que, dentro do número máximo de tentativas, possuir o menor tamanho e a menor inconsistência em relação à classe. Portanto, o *CBF* é considerado um algoritmo probabilístico caracterizado por avaliar subconjuntos de atributos de acordo com sua consistência em relação à um determinado rótulo. Usualmente, a busca favorece subconjuntos pequenos de atributos que apresentam alta consistência com o atributo classe. O algoritmo *CBF* possui complexidade de tempo de $O(max\_tries \cdot N)$. No entanto, se o método de busca utilizado for *forward selection* ou *backward selection*, esse algoritmo apresenta uma complexidade de tempo de $O(N \cdot M^2)$ (Liu and Setiono, 1996).

Algoritmo 1 *Consistency-Based Filter* (*CBF*)

Entrada: D *(CD no formato atributo-valor);* M *(número de atributos);* γ *(limiar aceitável de inconsistência);* max_tries *(número máximo de iterações);*

Saída: S_{best} *(subconjunto de atributos que satisfaz o critério de inconsistência estabelecido);*

1: $C_{best} \leftarrow M$;
2: **for** $i \leftarrow 1$ **to** max_tries **do**
3: $S \leftarrow subConjunto(D)$;
4: $C \leftarrow numAtributos(S)$; */* Número de atributos do subconjunto* S **/*
5: **if** $((C < C_{best})$ **and** $(Inconsistencia(S,D) \leq \gamma))$ **then**
6: $S_{best} \leftarrow S$;
7: $C_{best} \leftarrow C$;
8: **end if**
9: **end for**
10: **return** S_{best}.

***CFS*:** o algoritmo *Correlation-based Feature Selection* (*CFS*) considera a habilidade preditiva individual de cada atributo e o grau de correlação entre esses atributos, incluindo a classe (Hall, 2000). Assim, é desejável que os subconjuntos sejam formados por atributos relevantes e não redundantes. Para tanto, o *CFS* atribui pesos de relevância aos subconjuntos de atributos de acordo com medidas de avaliação de separabilidade como, por exemplo, a medida *Symmetrical Uncertainty* (Press et al., 1992). Ainda nesse método, a busca por subconjuntos de atributos é interrompida quando a adição de atributos não proporciona uma melhoria no valor de avaliação quando comparado ao melhor subconjunto corrente. Em Parmezan (2012) é mostrado, em detalhes, o funcionamento desse algoritmo, o qual possui complexidade de tempo de $O(N \cdot M^2)$ (Hall, 1999).

***CSE*:** o algoritmo *Classifier Subset Evaluator* (*CSE*) (Algoritmo 2), é um método aplicado conforme a abordagem *wrapper* que avalia subconjuntos de atributos a partir do conjunto original de atributos (Kohavi and John, 1997). O *CSE* utiliza internamente um classificador para estimar a importância de um

determinado subconjunto de atributos. Nesse contexto, é importante observar que as medidas de precisão são dependentes do algoritmo de aprendizado considerado, uma vez que os subconjuntos de atributos são importantes em relação ao desempenho preditivo do MC por um determinado algoritmo.

Algoritmo 2 *Classifier Subset Evaluator* (*CSE*)

Entrada: D *(CD no formato atributo-valor);* $AlgCls$ *(algoritmo de AM para classificação);* max_tries *(número máximo de iterações);*

Saída: S_{best} *(subconjunto de atributos que proporcionou ao MC o melhor desempenho preditivo);*

1: $S_{best} \leftarrow conjAtributos(D)$; */* Todos os atributos* $\in D$*, com exceção do atributo classe */*
2: $DP_{best} \leftarrow estimarDP(D,S_{best},AlgCls)$; */* Desempenho preditivo do MC a partir do CD original) */*
3: **for** $i \leftarrow 1$ **to** max_tries **do**
4: $S \leftarrow subConjunto(D)$;
5: $DP \leftarrow estimarDP(D,S,AlgCls)$; */* Desempenho preditivo do MC utilizando o subconjunto* S **/*
6: **if** $(DP > DP_{best})$ **then**
7: $S_{best} \leftarrow S$;
8: $DP_{best} \leftarrow DP$;
9: **end if**
10: **end for**
11: **return** S_{best}.

***InfoGain*:** o método *Information-Gain Attribute Ranking* (*InfoGain*) seleciona os atributos por meio da avaliação individual (Das, 2001). Conforme apresentado no Algoritmo 3, a ideia básica desse método consiste em computar o ganho de informação (Han et al., 2011), baseado na medida de entropia (Mitra et al., 2002), para avaliar a relevância de um atributo com relação à classe.

Algoritmo 3 *Information-Gain Attribute Ranking* (*InfoGain*)

Entrada: D *(CD no formato atributo-valor);*

Saída: L *(lista de atributos ordenados em ordem decrescente de importância segundo a medida ganho de informação);*

1: $A \leftarrow conjAtributos(D)$; */* Todos os atributos* $\in D$*, com exceção do atributo classe */*
2: $M \leftarrow numAtributos(D)$; */* Número de atributos contidos em* A **/*
3: $L \leftarrow novaLista\{\}$;
4: **for** $i \leftarrow 1$ **to** M **do**
5: $insereAtributo(A[i],L,computarInfoGain(A[i],D))$; */* Inserção do atributo* $A[i]$ *na lista* L *em ordem decrescente de acordo com o ganho de informação do respectivo atributo em relação à classe */*
6: **end for**
7: **return** L.

***ReliefF*:** o algoritmo *Relief* trabalha por meio da amostragem aleatória de exemplos do CD e da localização do vizinho mais próximo da mesma classe e do vizinho mais próximo da classe oposta (Kononenko, 1994). Desse modo, os valores 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 m vezes, onde m é o número de vezes que o algoritmo procura por exemplos no CD. *Relief* consegue trabalhar com atributos quantitativos e qualitativos, mas é limitado a problemas de duas classes. Por esse motivo o algoritmo foi estendido em seis variações e a versão denominada *ReliefF* (Algoritmo 4) praticamente substituiu o método original, pois é capaz de lidar com CD incompletos, com ruídos e com múltiplas classes (Spolaôr et al., 2011a).

Algoritmo 4 *ReliefF*

Entrada: D *(CD no formato atributo-valor);* m *(número máximo de iterações);* k *(número de exemplos mais próximos);*

Saída: W *(vetor de dimensão* M*, onde cada posição* $1 \leq j \leq M$ *armazena o peso do atributo correspondente);*

1: **for** $i \leftarrow 1$ **to** M **do** $W[i] \leftarrow 0$ **end for**
2: **for** $i \leftarrow 1$ **to** m **do**
3: $E \leftarrow getExemplo(D)$; */* Seleção de um exemplo* E *em* D*, sem reposição */*
4: $Hit[k] \leftarrow nearHit(k,E,D)$; */* Identificação dos* k *exemplos mais próximos de classe idêntica ao* E **/*
5: **for** $c \neq classe(E)$ **do**
6: $Miss[k] \leftarrow nearMiss(k,E,D(c))$; */* Identificação dos* k *exemplos mais próximos de classe distinta de* E **/*
7: **end for**
8: **for** $j \leftarrow 1$ **to** M **do**
9:
$$W[j] \leftarrow W[j] - \sum_{y \leftarrow 1}^{k} diff(A[j],E,Hit[y]) \;/\; (m \times k) + \sum_{c \neq classe(E)} \left[\frac{P(c)}{1 - P(classe(E))} \sum_{y \leftarrow 1}^{k} diff(A[j],E,Miss[y]) \right] / \; (m \times k);$$
10: **end for**
11: **end for**
12: **return** W.

No *ReliefF*, a influência de ruído nos dados é amenizada por meio da contribuição dos k vizinhos mais próximos da mesma classe do exemplo correntemente considerado e dos k vizinhos mais próximos de cada uma das classes diferentes do exemplo amostrado, ao invés de considerar apenas um único vizinho mais próximo. Esse método procura pelos exemplos mais próximos utilizando a distância de *Manhattan*, e atribui pesos aos atributos de acordo com quão bem eles diferenciam esses exemplos. Esse processo, assim como no *Relief*, também é repetido m vezes. Em geral, m é definido em função do número de exemplos presentes no CD.

No Algoritmo 4 é possível notar o uso de mais de um vizinho mais próximo do exemplo E para cada classe envolvida. Observa-se também a ponderação

da contribuição de vizinhos próximos de classes diferentes de acordo com a probabilidade *a priori* $P(c)$ correspondente à classe c e o tratamento probabilístico de dados incompletos. O cálculo da diferença, $diff$, é semelhante ao do método original *Relief*, porém, considera a probabilidade de dois exemplos possuírem valores diferentes para um determinado atributo. O método *ReliefF* possui complexidade de tempo de $O(m \cdot N \cdot M)$ (Robnik-Sikonja and Kononenko, 2003) e seleciona os atributos por meio da avaliação individual.

Assim como *ReliefF*, o método *InfoGain* também seleciona os atributos por meio da avaliação individual. Desse modo, é preciso estabelecer um limiar para definir o subconjunto de atributos retornado por esses algoritmos.

Usualmente, quando se realiza a avaliação individual, após a ordenação dos atributos segundo uma medida de importância, os t primeiros atributos são escolhidos para compor o subconjunto de atributos importantes selecionado, onde t é definido segundo algum critério, por exemplo, uma percentagem do número original de atributos.

Na Tabela 4.2 é apresentado um resumo das principais características desses algoritmos. As informações dessa tabela estão organizadas do seguinte modo: nas primeiras duas linhas são indicados o modo de avaliação dos atributos (individual ou por subconjuntos de atributos). Nas duas últimas linhas são exibidas as abordagens (filtro ou *wrapper*), na qual esses métodos são utilizados. Nas linhas restantes são apresentadas as categorias de medidas de importância de atributos empregadas por cada um dos algoritmos de SA.

	CBF	***CFS***	***CSE***	***InfoGain***	***ReliefF***
Avaliação Individual				✔	✔
Avaliação de Subconjuntos	✔	✔	✔		
Medida de Consistência	✔				
Medida de Correlação		✔			
Medida de Distância					✔
Medida de Informação				✔	
Medida de Precisão			✔		
Abordagem Filtro	✔	✔		✔	✔
Abordagem *Wrapper*			✔		

Tabela 4.2: Características dos algoritmos de SA.

***J48*:** o *J48* é um algoritmo de indução de árvores de decisão do paradigma simbólico utilizado para realizar inferências indutivas (Han et al., 2011; Witten and Frank, 2005). Esse método utiliza a estratégia de dividir para conquistar,

ou seja, um problema é decomposto em subproblemas mais simples e essa estratégia é empregada de modo recursivo sobre cada subproblema decomposto. O *J48* permite a representação da árvore de decisão como uma disjunção de conjunções, aonde cada ramo da árvore, desde a raiz até a folha, é uma conjunção de condições sobre os atributos e o conjunto de ramos da árvore é disjunto. Desse modo, o algoritmo classifica um dado exemplo partindo da raiz da árvore em direção a algum nó folha que possa indicar a classe desse exemplo. Em outras palavras, cada nó da árvore especifica um teste sobre algum atributo do exemplo e cada ramo do nó um dos possíveis valores do atributo. O *J48* não assume nenhuma distribuição particular para os dados e possibilita que uma decisão complexa (predizer o valor da classe) possa ser decomposta numa sucessão de decisões elementares.

É importante ressaltar que neste trabalho não foi considerada a seleção embutida de atributos fornecida pelo *J48*, na qual 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.

***MLP*:** o *Multilayer Perceptron* (*MLP*) é um algoritmo de aprendizado do paradigma conexionista que envolve unidades altamente conectadas (Haykin, 2009). O nome conexionismo é utilizado para descrever a área de estudo das redes neuronais artificiais, as quais são construções matemáticas inspiradas em conexões neurais do sistema nervoso. O *MLP* é um modelo de rede que apresenta uma ou mais camadas de neurônios entre as camadas de entrada de dados e de saída dos resultados (camadas intermediárias). Desse modo, cada camada tem uma função específica, por exemplo, a camada de saída recebe os estímulos da camada intermediária e gera a resposta final. Essas camadas intermediárias são unidades que não interagem diretamente com o ambiente e funcionam como extratoras de características. Se existirem conexões apropriadas entre as unidades de entrada e um conjunto suficientemente grande de unidades intermediárias, pode-se sempre encontrar a representação que irá produzir o mapeamento correto entre a entrada de dados e a saída dos resultados (classificação). Usualmente, esse tipo de rede neuronal é treinado por meio do algoritmo de retropropagação *Backpropagation* (Haykin, 2009). A aplicação do *MLP* possibilita a manipulação eficiente de grandes volumes de dados, sendo a capacidade de generalização uma das principais características desse método.

4.3 Configuração dos Experimentos

Os experimentos realizados com o suporte do ambiente *Weka* foram organizados em três etapas, as quais são ilustradas na Figura 4.1.

Etapa 1: nessa etapa foi realizada a SA utilizando os algoritmos descritos na Seção 4.2 sobre os sete CD naturais tratados na Seção 4.1. Todos esses algoritmos, à exceção do *CSE*, foram executados com configuração de parâmetros com valores padrão. O método de busca *Best First* com direção *forward* foi empregado para os algoritmos que avaliam subcon-

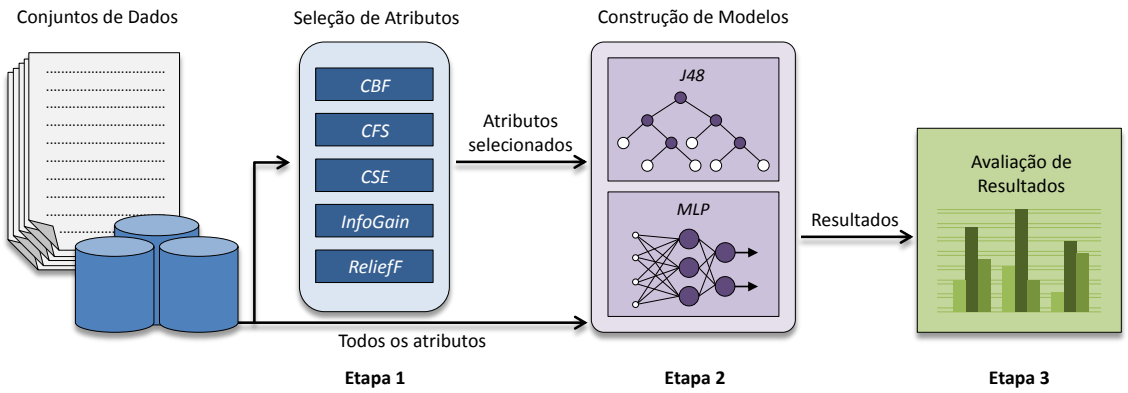


Figura 4.1: Configuração dos experimentos (modificado de Parmezan et al. (2011b)).

juntos de atributos. Para o *CSE* utilizou-se os algoritmos de classificação *J48* e *MLP*. Essas combinações são representadas respectivamente por *CSE*(J) e *CSE*(M) no restante deste trabalho. Para os algoritmos de avaliação individual de atributos, considerou-se um limiar de 30% do total de atributos ordenados em ordem decrescente de importância, na SA. O valor desse limiar foi estabelecido com a finalidade de reduzir, aproximadamente, 70% da dimensionalidade dos CD. É importante ressaltar que esse valor pode variar para mais ou para menos, uma vez que o número original de atributos pertence ao domínio dos inteiros;

Etapa 2: nessa etapa foram construídos modelos com e sem SA (Original) usando os indutores *J48* e *MLP*, totalizando 84 MC [(6 algoritmos de SA $\times$ 7 CD) $\times$ 2 algoritmos de classificação]. Ainda, para o *CSE*, considerou-se o mesmo algoritmo de aprendizado tanto para realizar a tarefa de SA quanto para processar o conjunto de exemplos com os atributos selecionados;

Etapa 3: nessa última etapa, os resultados obtidos com a aplicação dos algoritmos de SA foram comparados com os resultados dos modelos gerados no caso Original, quanto ao desempenho preditivo dos classificadores induzidos estimado por meio de validação cruzada com 10 partições. Os desempenhos preditivos dos MC foram comparados usando o teste estatístico não paramétrico *Kruskal-Wallis* para grupos não pareados, com nível de significância de 5% e seguido do pós-teste de *Dunn*[5].

É importante ressaltar que neste trabalho optou-se pela utilização do *J48* e do *MLP* por se tratarem de algoritmos amplamente discutidos na literatura. Especificamente, foram adotados dois algoritmos de classificação com o intuito de minimizar a influência que um deles poderia ter nos resultados.

[5]Testes estatísticos realizados utilizando *GraphPad InStat* versão 3.05 para *Windows*, http://www.graphpad.com.

Capítulo 5

Resultados e Discussão

Conforme mencionado na Seção 4.3, para cada CD, foi realizada a SA utilizando os algoritmos *CBF*, *CFS*, *CSE*(J), *CSE*(M), *InfoGain* e *ReliefF*, totalizando 42 configurações (6 algoritmos de SA $\times$ 7 CD).

Modelos foram gerados considerando os atributos selecionados por cada algoritmo de SA e também considerando todos os atributos para cada CD sem SA (Original), totalizando 84 MC (42 configurações $\times$ 2 algoritmos de classificação).

O desempenho preditivo dos classificadores induzidos foram estimados por meio de validação cruzada com 10 partições e comparados usando o teste estatístico não paramétrico *Kruskal-Wallis* (nível de significância de 5%) e pós-teste de *Dunn*.

Os resultados obtidos, apresentados a seguir, estão organizados do seguinte modo:

1. atributos selecionados pelos algoritmos utilizados neste trabalho considerando a maneira de avaliação dos atributos (individual ou por subconjuntos de atributos);
2. desempenho dos algoritmos de SA quanto à média de erro do MC e a percentagem de atributos selecionados;
3. desempenho preditivo dos classificadores induzidos e análise da significância estatística dos resultados;
4. impacto dos atributos selecionados pelos métodos aplicados conforme as abordagens filtro e *wrapper* em relação ao tempo de aprendizado;
5. considerações gerais associadas às categorias de avaliação de importância de atributos.

5.1 Análise dos Atributos Selecionados

Além das diversas características próprias do CD, a maneira de avaliação dos atributos (individual ou por subconjuntos de atributos) e a medida utilizada para determinar a importância dos atributos em relação à classe podem influenciar no subconjunto de atributos selecionado pelos algoritmos.

Na Tabela 5.1 é apresentado um resumo da quantidade de atributos selecionados por cada um dos algoritmos de SA. As informações dessa tabela estão organizadas do seguinte modo: na primeira coluna é apresentado o CD ao qual referem-se às informações. Na segunda coluna é indicada a quantidade original de atributos de cada CD. Nas demais colunas, para cada uma das células, na primeira linha são descritos o número de atributos referentes ao subconjunto selecionado por cada um dos algoritmos de SA e na segunda linha é apresentada a respectiva percentagem. As últimas duas linhas da tabela mostram a média de atributos selecionados (Média # A) e a percentagem média (Média % A) por cada algoritmo dentre todos os CD.

Conjunto de Dados	**Original**	***CBF***	***CFS***	***CSE*(J)**	***CSE*(M)**	***InfoGain***	***ReliefF***
BreastCancer	9	8 88,89	5 55,56	5 55,56	8 88,89	3 33,33	3 33,33
Bupa	6	1 16,67	1 16,67	4 66,67	5 83,33	2 33,33	2 33,33
Haberman	3	2 66,67	2 66,67	3 100,00	3 100,00	1 33,33	1 33,33
Hepatitis	19	12 63,16	10 52,63	6 31,58	12 63,16	6 31,58	6 31,58
Hungarian	13	10 76,92	6 46,15	8 61,54	13 100,00	4 30,77	4 30,77
LungCancer	56	4 7,14	8 14,29	2 3,57	4 7,14	17 30,36	17 30,36
Pima	8	8 100,00	4 50,00	6 75,00	8 100,00	2 25,00	2 25,00
Média # A		6	5	5	8	5	5
Média % A		59,92	43,14	56,27	77,50	31,10	31,10

Tabela 5.1: Quantidade de atributos selecionados por cada um dos algoritmos de SA e suas respectivas percentagens.

Nessa tabela, as colunas em tons de cinza diferenciam os algoritmos em relação à três grupos. O primeiro e segundo grupos representam algoritmos que realizam seleção de subconjuntos de atributos. O segundo grupo, especificamente, é o único que abrange métodos aplicados conforme a abordagem *wrapper*. O terceiro grupo contempla algoritmos que realizam avaliação individual dos atributos.

Considerando os percentuais de atributos selecionados por cada método de SA, o algoritmo *CSE*(M) foi o que selecionou os subconjuntos com mais atributos, variando de $63{,}16\%$ até o máximo de $100{,}00\%$ (todos os atributos) para seis do total de sete CD, com exceção de *LungCancer*, para o qual foram selecionados apenas $7{,}14\%$ do total de atributos.

Dentre os algoritmos que selecionam os atributos por avaliação de subconjuntos, o *CFS* foi o que selecionou os menores subconjuntos de atributos, variando de $14{,}29\%$ para o CD *LungCancer* até $66{,}67\%$ para o CD *Haberman*. É interessante observar que o *CFS* foi o único algoritmo, dentre os que avaliam subconjunto, que sempre conseguiu reduzir o número de atributos.

Conforme mencionado, o limiar utilizado para os algoritmos de avaliação individual possibilitou reduzir aproximadamente $70{,}00\%$ da dimensionalidade

dos CD naturais.

Na Figura 5.1 é mostrada graficamente, para cada CD, a percentagem de atributos selecionados por cada um dos algoritmos de SA.

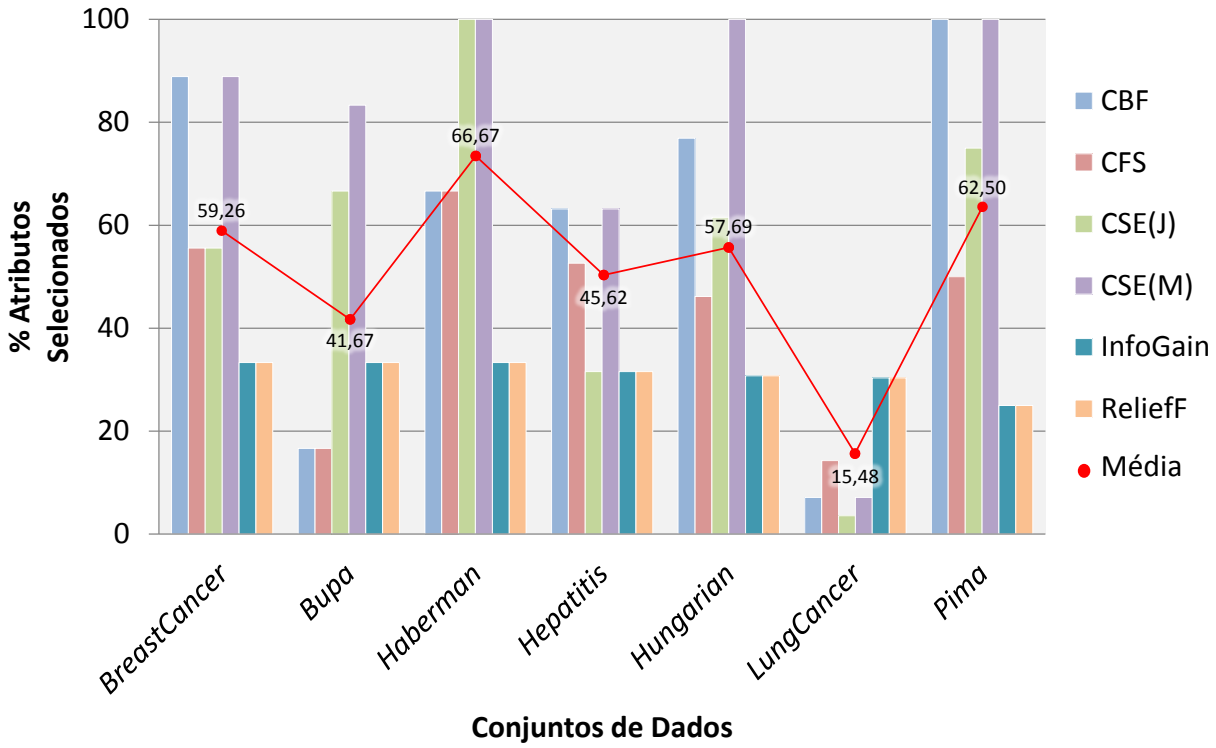


Figura 5.1: Percentagem de atributos selecionados por cada um dos algoritmos de SA.

Considerando somente os atributos selecionados pelos métodos que avaliam subconjuntos de atributos, é interessante notar que *Bupa* e *LungCancer* foram os dois CD, do total de sete, que usualmente obtiveram reduções expressivas do número de atributos quando comparados aos Originais. Ainda, verifica-se que *Pima* e *Haberman* foram os únicos CD nos quais todos os atributos foram considerados importantes em relação à classe por mais de um algoritmo de SA.

As tabelas que exibem os atributos selecionados por cada um dos algoritmos considerados neste trabalho são apresentadas no Apêndice A.

5.2 Avaliação por Desempenho Preditivo e por Percentagem de Atributos Selecionados

O erro médio correspondente aos classificadores induzidos, antes e após a aplicação dos algoritmos de SA, é apresentado nas Tabelas 5.2 (MC utilizando o *J48*) e 5.3 (MC utilizando o *MLP*).

Cada algoritmo de SA foi também avaliado de acordo com a relação entre a quantidade de atributos selecionados e o erro médio dos classificadores induzidos. Essa avaliação foi realizada por meio de um modelo proposto em Lee (2005). Esse modelo categoriza, para cada CD, os algoritmos de SA

	J48					
Conjunto de Dados (ECM)	**Original**	***CBF***	***CFS***	***CSE*(J)**	***InfoGain***	***ReliefF***
BreastCancer (29,72)	25,72	27,49	27,06	25,37	28,14	26,67
Bupa (42,03)	34,16	37,76	37,76	32,48	37,50	31,93
Haberman (26,47)	27,84	27,97	27,97	27,84	28,79	26,47
Hepatitis (20,65)	20,78	20,26	19,73	17,26	16,15	15,78
Hungarian (36,05)	19,78	19,60	21,00	22,33	20,56	21,09
LungCancer (59,38)	20,75	16,42	16,42	15,75	17,42	19,33
Pima (34,98)	25,51	25,51	25,62	24,86	25,35	25,35

Tabela 5.2: Média de erro para cada CD e algoritmos considerados (MC utilizando o algoritmo *J48*).

	MLP					
Conjunto de Dados (ECM)	**Original**	***CBF***	***CFS***	***CSE*(M)**	***InfoGain***	***ReliefF***
BreastCancer (29,72)	33,05	33,34	28,03	33,34	29,86	27,75
Bupa (42,03)	31,27	39,93	39,93	28,71	39,30	31,71
Haberman (26,47)	29,68	27,24	27,24	29,68	26,38	26,99
Hepatitis (20,65)	18,71	21,29	17,50	21,81	16,37	14,67
Hungarian (36,05)	19,68	17,55	18,57	19,68	20,61	21,12
LungCancer (59,38)	31,25	15,25	16,75	15,25	30,42	29,50
Pima (34,98)	25,25	25,25	24,03	25,25	25,50	24,50

Tabela 5.3: Média de erro para cada CD e algoritmos considerados (MC utilizando o algoritmo *MLP*).

quanto ao seu posicionamento em relação às duas medidas mencionadas, dentro de cinco regiões previamente definidas, como ilustrado na Figura 5.2.

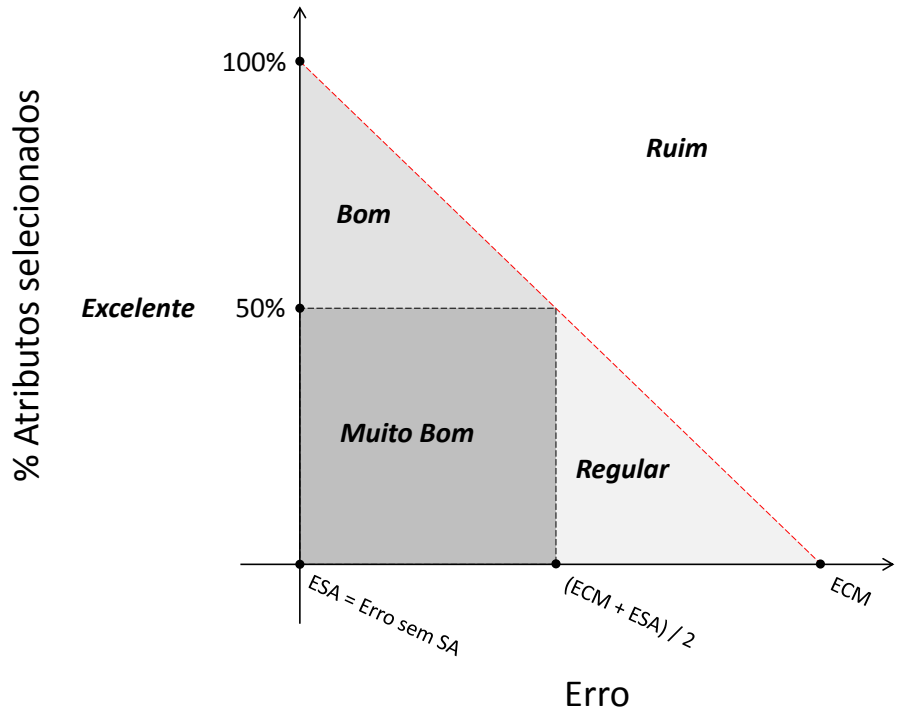


Figura 5.2: Modelo de categorização de algoritmos em relação à percentagem de atributos selecionados e à média de erro dos classificadores induzidos (modificado de Lee (2005)).

Nessa figura, a categoria Muito Bom compreende os modelos que apresentam o melhor compromisso entre a quantidade de atributos selecionados e a média de erro. A região Bom indica quais classificadores apresentam baixas médias de erro com alta quantidade de atributos, enquanto a categoria Regular abrange modelos com altas médias de erro e baixa quantidade de atributos. Os modelos situados na região Excelente apresentam média de erro inferior à média do classificador induzido utilizando o CD Original, enquanto a região restante, localizada acima da reta inclinada que divide o plano em dois semi-planos, representa a categoria Ruim.

A aplicação desse modelo exige a definição de três limiares:

Erro sem SA (ESA): taxa de erro associada ao MC a partir de todos os atributos (CD Original);

ECM: limiar que assume o valor correspondente ao erro da classe majoritária se o mesmo for menor que 50%, ou 50% caso contrário;

Taxa Média (TM): limiar obtido através da média aritmética entre ESA e ECM.

Essa abordagem comparativa foi escolhida para auxiliar na avaliação da performance dos algoritmos utilizados neste trabalho, tendo em vista que na SA é importante não somente avaliar o desempenho preditivo do MC, mas também a quantidade de atributos selecionados por cada um dos métodos

de SA. Consequentemente, o esforço computacional para estimar quanto um modelo pode degradar considerando a redução da quantidade de atributos necessários para sua indução torna-se menos custoso (Alpaydin, 2004).

Na Figura 5.3 é apresentado um exemplo do modelo de avaliação, proposto em Lee (2005) e implementado neste trabalho via linguagem de programação R[1], para o CD *BreastCancer* utilizando o algoritmo *J48* para indução de modelos.

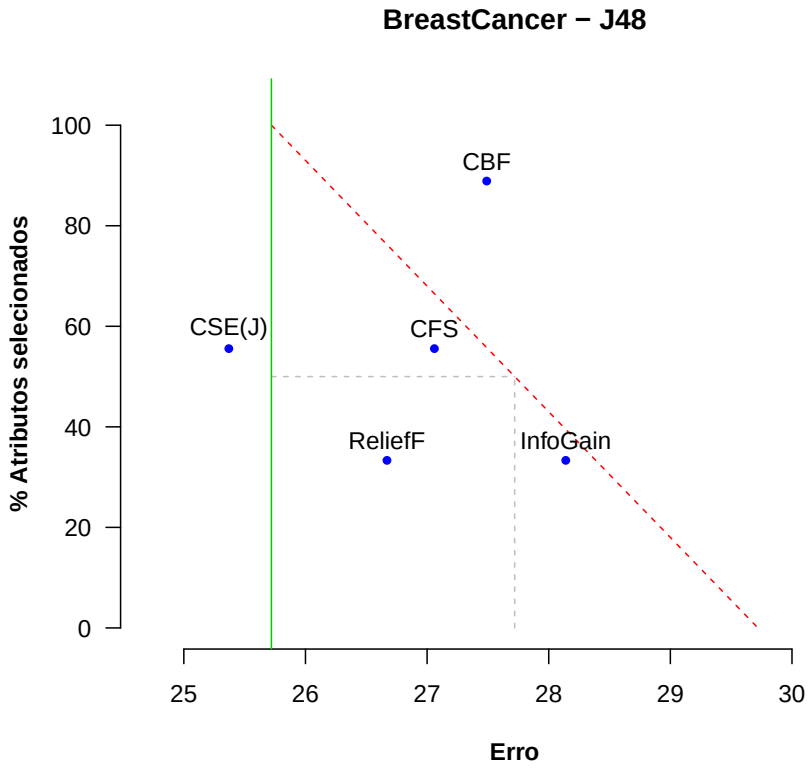


Figura 5.3: Modelo de categorização de algoritmos aplicado sobre o CD *BreastCancer* considerando o indutor *J48*.

De acordo com as informações exibidas na Figura 5.3, o MC a partir do subconjunto de atributos selecionado pelo algoritmo *CSE*(J) foi categorizado como excelente. Diferentemente, o MC utilizando o subconjunto de atributos selecionado pelo algoritmo *CBF* foi categorizado como ruim. Já os modelos induzidos a partir dos subconjuntos de atributos selecionados pelos algoritmos *CFS*, *ReliefF* e *InfoGain* foram categorizados como bom, muito bom e regular, respectivamente.

O MC a partir desse mesmo CD utilizando o algoritmo de indução *MLP* apresentou ESA > ECM. Esse fato impossibilitou a aplicação correta da abordagem comparativa e, portanto, o respectivo modelo de avaliação de desempenho de algoritmos gerado não é exibido neste trabalho.

As figuras resultantes do emprego do modelo de categorização sobre os demais CD naturais que obtiveram ESA < ECM estão localizadas no Apêndice B.

[1] http://www.r-project.org.

Nas Tabelas 5.4 e 5.5 é apresentado um resumo das categorizações dos algoritmos de SA para cada CD quanto ao seu posicionamento dentro das cinco regiões descritas. Nas últimas linhas dessas tabelas é exibida a quantidade de vezes em que o respectivo algoritmo foi classificado como tendo apresentado desempenho Excelente (▲▲▲), Muito Bom (▲▲), Bom (▲), Regular (♦), Ruim (▼), ESA > ECM (↓) e Todos os Atributos Selecionados (—).

	J48				
Conjunto de Dados	***CBF***	***CFS***	***CSE*(J)**	***InfoGain***	***ReliefF***
BreastCancer	▼	▲	▲▲▲	♦	▲▲
Bupa	▲▲	▲▲	▲▲▲	▲▲	▲▲▲
Haberman	↓				
Hepatitis	↓				
Hungarian	▲▲▲	▲▲	▲	▲▲	▲▲
LungCancer	▲▲▲	▲▲▲	▲▲▲	▲▲▲	▲▲▲
Pima	—	▲	▲▲▲	▲▲▲	▲▲▲
Excelente (▲▲▲)	2	1	4	2	3
Muito bom (▲▲)	1	2	0	2	2
Bom (▲)	0	2	1	0	0
Regular (♦)	0	0	0	1	0
Ruim (▼)	1	0	0	0	0
Todos os Atributos Selecionados (—)	1	0	0	0	0

Tabela 5.4: Categorização dos algoritmos de SA em relação à percentagem de atributos selecionados *versus* erro do MC (classificadores induzidos utilizando o algoritmo *J48*).

Considerando cada algoritmo de SA em relação às categorias, o *J48*/*CSE*(J) (MC por meio do indutor J48 a partir do subconjunto de atributos selecionado pelo algoritmo *CSE*(J)) e o *MLP*/*CFS* foram os algoritmos que obtiveram o maior número de categorizações excelentes, cada um deles tendo obtido quatro. Em termos de *J48* e *MLP*, as categorizações muito boas, boas e regulares ocorreram de um modo uniforme entre todos os algoritmos, com exceção do *CFS*. Quanto às categorizações ruins, os MC a partir dos indutores *J48* e *MLP*, obtiveram um (*J48*/*CBF*) e três (*MLP*/*CBF*, *MLP*/*CSE*(M), *MLP*/*InfoGain*) exemplares, respectivamente. Dentre todos os modelos induzidos, o *J48*/*CBF*, *MLP*/*CBF* e *MLP*/*CSE*(M) foram os únicos que não promoveram a redução do número de atributos para um ou mais CD.

Dentre as 25 classificações obtidas a partir dos MC por meio do algoritmo *J48*, doze foram excelentes, sete muito boas, três boas, uma regular, uma ruim e uma selecionou todos os atributos do conjunto Original. É possível observar que dentre essas classificações, 88,00% foram excelentes, muito boas ou boas, 4,00% dos subconjuntos de atributos selecionados foram iguais aos CD Originais, isto é, contendo todos os atributos e apenas 8,00% das classificações foram consideradas regulares ou ruins. Já para as classificações dos MC utilizando o algoritmo *MLP*, quatorze foram excelentes, três muito boas, duas regulares, três ruins e três selecionaram todos os atributos do conjunto Original de atributos. Observa-se que dentre essas 25 classificações, 68,00%

	MLP				
Conjunto de Dados	***CBF***	***CFS***	***CSE*(M)**	***InfoGain***	***ReliefF***
BreastCancer			↓		
Bupa	♦	♦	▲▲▲	▼	▲▲
Haberman			↓		
Hepatitis	▼	▲▲▲	▼	▲▲▲	▲▲▲
Hungarian	▲▲▲	▲▲▲	—	▲▲	▲▲
LungCancer	▲▲▲	▲▲▲	▲▲▲	▲▲▲	▲▲▲
Pima	—	▲▲▲	—	▲▲▲	▲▲▲
Excelente (▲▲▲)	2	4	2	3	3
Muito bom (▲▲)	0	0	0	1	2
Bom (▲)	0	0	0	0	0
Regular (♦)	1	1	0	0	0
Ruim (▼)	1	0	1	1	0
Todos os Atributos Selecionados (—)	1	0	2	0	0

Tabela 5.5: Categorização dos algoritmos de SA em relação à percentagem de atributos selecionados *versus* erro do MC (classificadores induzidos utilizando o algoritmo *MLP*).

foram excelentes ou muito boas, $12{,}00\%$ selecionaram todos os atributos do conjunto Original e $20{,}00\%$ foram consideradas regulares ou ruins.

Do total de 50 categorizações [(5 algoritmos de SA $\times$ 5 CD) $\times$ 2 algoritmos de classificação)], $78{,}00\%$ foram excelentes, muito boas ou boas, $8{,}00\%$ dos subconjuntos de atributos selecionados foram iguais aos CD Originais contento todos os atributos e $14{,}00\%$ das categorizações foram consideradas regulares ou ruins. É importante ressaltar que o *J48* foi o indutor que ofereceu o maior número de MC categorizados como excelentes, muito bons ou bons.

Dentre os sete CD, três (*BreastCancer*, *Heberman* e *Hepatitis*) apresentaram média do erro do MC, a partir do CD sem SA, superior ao ECM.

Portanto, a maior parte dos algoritmos de SA contribuiu para a melhoria, tanto em relação à redução do número de atributos quanto em relação ao desempenho preditivo dos MC utilizando os subconjuntos de atributos selecionados.

5.3 Comparação Estatística dos Modelos Induzidos

Como relatado, para cada CD, os algoritmos foram comparados com o cenário Original quanto ao desempenho preditivo dos MC, estimado por meio de validação cruzada com 10 partições, utilizando o teste estatístico *Kruskal-Wallis* (nível de significância de 5%) e pós-teste de *Dunn*.

Nas Tabelas 5.6 e 5.7 são apresentados, para cada CD sem SA (Original) e cada um dos subconjuntos selecionados pelos algoritmos considerados neste trabalho, a média do desempenho preditivo dos MC utilizando os indutores *J48* e *MLP*, respectivamente, bem como o desvio padrão (entre parênteses) e a

percentagem de redução de atributos.

	J48						
Conjunto de Dados	**Original**	***CBF***	***CFS***	***CSE*(J)**	***InfoGain***	***ReliefF***	**Média SA**
BreastCancer (29,72)	74,28(6,05) 0,00%	72,51(6,32) 11,11%	72,94(5,47) 44,44%	74,63(5,57) 44,44%	**71,86(5,65)** 66,67%	73,33(6,27) 66,67%	73,05(5,92) 46,67%
Bupa (42,03)	65,84(7,40) 0,00%	62,24(8,67) 83,33%	62,24(8,67) 83,33%	67,52(7,92) 33,33%	**62,50(6,78)** 66,67%	68,07(6,41) 66,67%	64,52(8,17) 66,67%
Haberman (26,47)	72,16(4,70) 0,00%	72,03(3,98) 33,33%	72,03(3,98) 33,33%	72,16(4,70) 0,00%	71,21(3,98) 66,67%	73,53(0,95) 66,67%	72,19(3,82) 40,00%
Hepatitis (20,65)	79,22(9,57) 0,00%	79,74(8,96) 36,84%	80,27(9,04) 47,37%	82,74(7,96) 68,42%	**83,85(7,22)** 68,42%	**84,22(8,19)** 68,42%	82,16(8,47) 57,89%
Hungarian (36,05)	80,22(7,95) 0,00%	80,40(7,97) 23,08%	79,00(7,30) 53,85%	77,67(7,08) 38,46%	79,44(7,25) 69,23%	78,91(5,75) 69,23%	79,08(7,13) 50,77%
LungCancer (59,38)	79,25(21,50) 0,00%	83,58(17,10) 92,86%	83,58(17,10) 85,71%	84,25(16,41) 96,43%	82,58(19,54) 69,64%	80,67(20,20) 69,64%	82,93(18,10) 82,86%
Pima (34,98)	74,49(5,27) 0,00%	74,49(5,27) 0,00%	74,38(5,04) 50,00%	75,14(5,11) 25,00%	74,65(5,02) 75,00%	74,65(5,02) 75,00%	74,66(5,08) 45,00%
Média	75,07(11,39) 0,00%	75,00(11,30) 40,08%	74,92(11,10) 56,86%	76,30(10,18) 43,73%	75,16(11,57) 68,90%	76,20(10,59) 68,90%	

Tabela 5.6: Média do desempenho preditivo, desvio padrão e percentagem de redução de atributos para cada CD e algoritmos considerados (MC utilizando o indutor *J48*).

	MLP						
Conjunto de Dados	**Original**	***CBF***	***CFS***	***CSE*(M)**	***InfoGain***	***ReliefF***	**Média SA**
BreastCancer (29,72)	66,95(8,51) 0,00%	66,66(8,92) 11,11%	**71,97(7,34)** 44,44%	66,66(8,92) 11,11%	70,14(6,87) 66,67%	**72,25(6,78)** 66,67%	69,53(8,17) 40,00%
Bupa (42,03)	68,73(7,38) 0,00%	**60,07(7,32)** 83,33%	**60,07(7,32)** 83,33%	71,29(7,24) 16,67%	**60,70(7,74)** 66,67%	68,29(6,68) 66,67%	64,08(8,66) 63,33%
Haberman (26,47)	70,32(6,70) 0,00%	72,76(5,65) 33,33%	72,76(5,65) 33,33%	70,32(6,70) 0,00%	**73,62(5,90)** 66,67%	**73,01(2,16)** 66,67%	72,49(5,54) 40,00%
Hepatitis (20,65)	81,29(9,17) 0,00%	78,71(8,95) 36,84%	82,50(9,76) 47,37%	78,19(9,77) 36,84%	83,63(7,92) 68,42%	**85,33(9,00)** 68,42%	81,67(9,49) 51,58%
Hungarian (36,05)	80,32(6,36) 0,00%	82,45(6,58) 23,08%	81,43(7,06) 53,85%	80,32(6,36) 0,00%	79,39(6,62) 69,23%	78,88(6,90) 69,23%	80,49(6,81) 43,08%
LungCancer (59,38)	68,75(22,83) 0,00%	**84,75(19,03)** 92,86%	**83,25(20,74)** 85,71%	**84,75(19,03)** 92,86%	69,58(22,11) 69,64%	70,50(22,14) 69,64%	78,57(21,73) 82,14%
Pima (34,98)	74,75(4,90) 0,00%	74,75(4,90) 0,00%	75,97(4,47) 50,00%	74,75(4,90) 0,00%	75,50(4,70) 75,00%	75,50(4,70) 75,00%	75,29(4,74) 40,00%
Média	73,02(12,19) 0,00%	74,31(12,69) 40,08%	**75,42(12,71)** 56,86%	75,18(11,54) 22,50%	73,22(12,45) 68,90%	74,82(11,52) 68,90%	

Tabela 5.7: Média do desempenho preditivo, desvio padrão e percentagem de redução de atributos para cada CD e algoritmos considerados (MC utilizando o indutor *MLP*).

As informações dessas tabelas estão organizadas do seguinte modo: na primeira coluna é apresentado o CD ao qual referem-se às informações seguido, entre parênteses, pelo ECM. Nas demais colunas, para cada uma das células, na primeira linha são descritos a média do desempenho preditivo do classificador induzido e o desvio padrão entre parênteses, enquanto que na segunda linha é exibida a percentagem de redução do número de atributos. Na última coluna é apresentada a média geral do desempenho preditivo, o desvio padrão e a média da percentagem de redução de atributos para cada CD. Na última linha é mostrada a média do desempenho preditivo e o desvio padrão referente ao comportamento geral dos algoritmos de SA.

As médias dos desempenhos preditivos conjuntamente com os desvios padrão que apresentaram diferença estatisticamente significativa (d.e.s) em relação ao cenário Original estão diferenciados em três cores, conforme o modo de comparação dos resultados.

A aplicação dos testes estatísticos foi organizada em duas etapas. Na Etapa 1 cada uma das células das Tabelas 5.6 e 5.7 foi comparada com sua respectiva célula Original utilizando o teste estatístico *Kruskal-Wallis*, o qual, dado um conjunto de comparações, permite verificar a ocorrência de diferenças significativas. Quando constatado esse comportamento, empregou-se o pós-teste de *Dunn* para identificar quais comparações apresentaram tal diferença. Na Etapa 2 cada uma das colunas referentes aos algoritmos de SA foi comparado com a coluna Original, similarmente à etapa anterior, a fim de analisar o comportamento geral dos métodos considerados.

Nesse contexto, estão destacados nas cores azul (Etapa 1) e verde (Etapa 2) os resultados associados aos algoritmos de SA que proporcionaram aprimoramento da performance dos classificadores induzidos. Já na cor vermelho (Etapa 1) estão indicados os resultados que proporcionaram degradação do desempenho preditivo dos modelos gerados. Ainda, é importante ressaltar que durante a execução da Etapa 2 não houveram deteriorações com diferenças significativas.

Observa-se na Tabela 5.6 que para o CD *Hepatitis*, os métodos de avaliação individual *InfoGain* e *ReliefF* apresentaram melhora no desempenho preditivo dos MC, com d.e.s. Diferentemente, para *BreastCancer* e *Bupa*, o algoritmo *InfoGain* proporcionou piora do desempenho preditivo dos classificadores induzidos. Adicionalmente, os modelos gerados a partir dos CD Originais *Haberman* e *Hepatitis* apresentaram média do erro superior ao ECM.

Como pode ser visibilizado na Tabela 5.7, foi possível identificar d.e.s quando comparados individualmente, para cada CD, os algoritmos de SA em relação ao Original. Para *BreastCancer*, os algoritmos *CFS* e *ReliefF* apresentaram melhora na performance dos MC a partir dos subconjuntos de atributos selecionados. Diferentemente, para *Bupa* os algoritmos *CBF*, *CFS* e *InfoGain* apresentaram degradação da performance dos classificadores induzidos. Esse fato pode ter sido causado pela expressiva redução do número de atributos. Já para o CD *Haberman*, *InfoGain* e *ReliefF* aprimoraram o desempenho dos MC, igualmente, para *Hepatitis*, em relação a *ReliefF*. *LungCancer* foi o único CD no qual todos os algoritmos promoveram expressivas reduções de atributos e apresentaram melhora do desempenho preditivo, com d.e.s, para três do total de cinco MC a partir dos subconjuntos de atributos selecionados. Outra característica específica desse CD é o maior número de atributos.

É importante ressaltar que os MC utilizando os CD Originais *BreastCancer* e *Haberman* apresentaram média do erro superior ao ECM. Como descrito, na última linha da Tabela 5.7 é apresentada a média referente ao desempenho geral de cada um dos algoritmos de SA sobre os sete CD. Assim sendo, foi possível identificar d.e.s entre o *CFS* em relação ao Original. Esse resultado indica que, para os CD utilizados e considerando o indutor *MLP*, o *CFS*, baseado em correlação, mostrou-se adequado para a tarefa de SA.

5.4 Avaliação do Impacto das Abordagens Filtro e *Wrapper* em Relação ao Tempo de Aprendizado

Como mencionado, na abordagem filtro o processo de SA independe do método de aprendizado considerado, enquanto que na abordagem *wrapper* esse método é utilizado como uma caixa preta para analisar a cada iteração o subconjunto de atributos investigado. Essa diferença torna ambas as abordagens bastante distintas, tanto em relação aos atributos selecionados pelos algoritmos de SA, quanto em relação à eficiência final do aprendizado.

Neste trabalho o termo tempo de aprendizado refere-se ao tempo necessário para a construção de um modelo a partir do CD Original ou usando um dos subconjuntos de atributos selecionados pelos algoritmos de SA.

Em geral, os métodos aplicados conforme a abordagem filtro são pouco custosos computacionalmente. Além disso, por serem baseados apenas em propriedades intrínsecas aos próprios dados de entrada, esses métodos podem ser acoplados a qualquer processo de aprendizado.

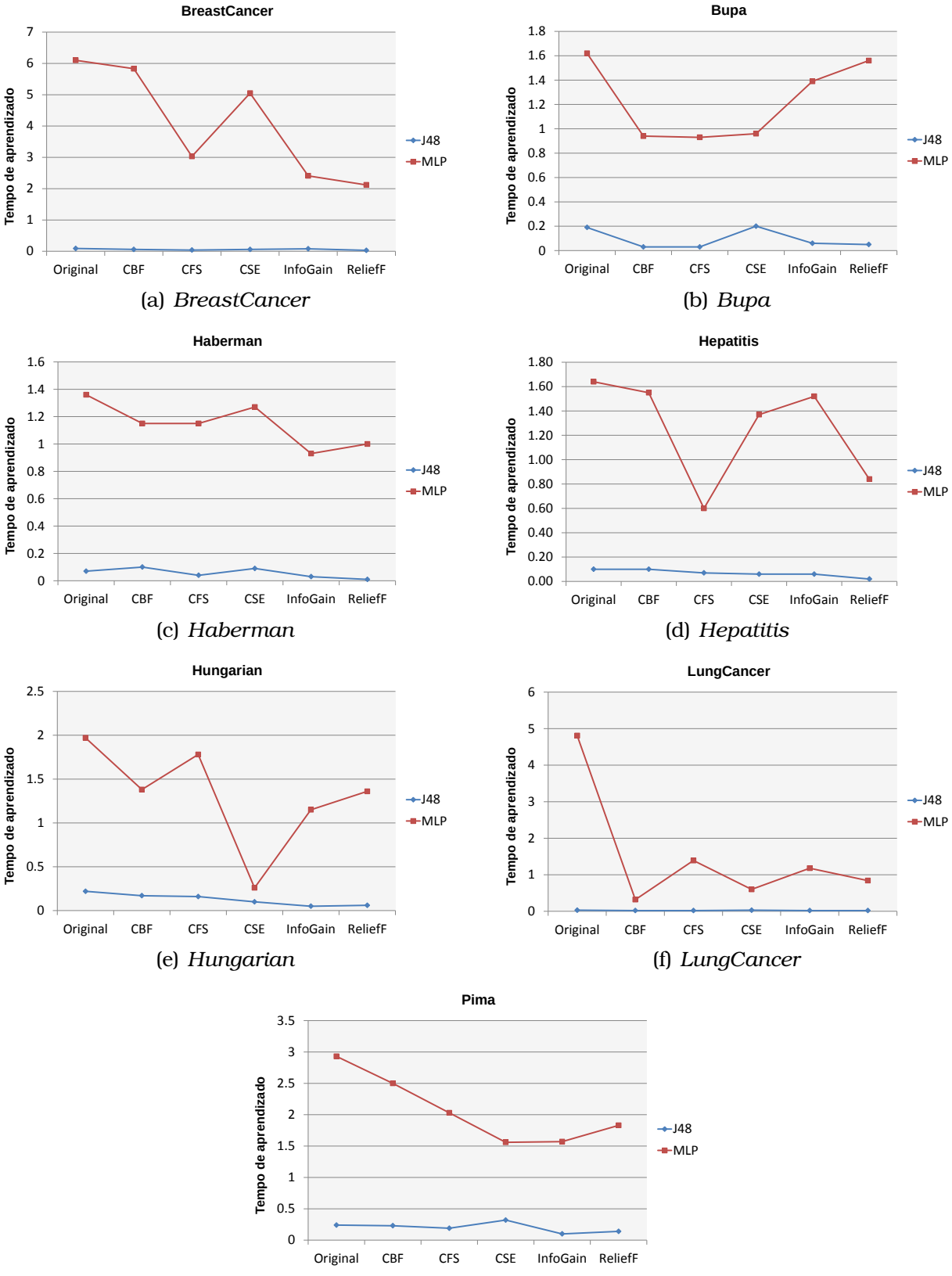
Diferentemente, a abordagem *wrapper* é comumente mais vantajosa que a filtro sob a perspectiva de otimização do desempenho preditivo de classificação, tendo em vista que os subconjuntos de atributos são usualmente avaliados utilizando o mesmo critério usado para estimar a performance do MC. Entretanto, considerar o próprio algoritmo de aprendizado para avaliar os subconjuntos de atributos requer um considerável poder computacional. Por isso, métodos aplicados conforme a abordagem *wrapper* deveriam ser evitados especialmente quando o algoritmo de extração de padrões for demasiadamente complexo ou, então, quando a dimensionalidade dos CD for significativamente alta. Esta segunda restrição se opõe a um dos objetivos da SA, que é o de redução do volume de dados.

Grande parte da pesquisa relacionada à SA tem como objetivo a melhoria de um método de classificação quanto ao seu desempenho preditivo. No entanto, em MD, existem outros fatores importantes a serem considerados como, por exemplo, o tempo de aprendizado.

A velocidade do aprendizado é relevante em casos onde o ambiente de extração de padrões modifica-se constantemente. Embora a redução do número de atributos possa implicar em um aumento da velocidade do aprendizado, esta nem sempre é otimizada, haja vista que um tempo adicional relativo à SA deve ser levado em consideração. Por outro lado, entretanto, em qualquer situação, quanto maior o número de atributos que descrevem os exemplos de treinamento, maior é o tempo de aprendizado.

Na Figura 5.4 é apresentado graficamente, para cada CD, os tempos de aprendizado, em segundos, obtidos por meio da aplicação dos indutores *J48* e *MLP*[2]. Nessa figura, a sigla *CSE* refere-se tanto ao método *CSE*(J), quando usado o algoritmo de classificação *J48*, quanto ao método *CSE*(M), quando empregado o algoritmo de aprendizado *MLP*.

[2]Contagem do tempo realizada em computador *Core Two Duo* 2.20 Ghz com 4 Gb de memória e sistema operacional *Windows* 7.



(a) *BreastCancer* (b) *Bupa*

(c) *Haberman* (d) *Hepatitis*

(e) *Hungarian* (f) *LungCancer*

(g) *Pima*

Figura 5.4: Tempo de aprendizado dos MC a partir dos CD Originais e dos subconjuntos de atributos selecionados pelos algoritmos de SA.

No Apêndice C são apresentadas detalhadamente as informações contidas nos gráficos da Figura 5.4, de modo a complementar essa seção.

Considerando apenas os modelos gerados pelo *J48*, o subconjunto de atributos selecionado do CD *Pima* pelo algoritmo aplicado conforme a abordagem *wrapper*, proporcionou ao classificador induzido o maior tempo de aprendizado (0,32 *segundos*), ultrapassando até mesmo o tempo do seu respectivo Original. Isso pode ter sido causado pelo fato de que o *CSE*(J) conseguiu reduzir apenas 25,00% da dimensionalidade desse CD, sendo que *Pima* apresenta o maior número de exemplos. O subconjunto identificado do CD *Haberman* pelo método *ReliefF*, aplicado conforme a abordagem filtro, forneceu o menor tempo de aprendizado dentre todos os MC (0,01 *segundos*). É interessante observar que *Haberman* possui 306 exemplos e o limiar utilizado para o algoritmo *ReliefF* possibilitou a redução de aproximadamente 70,00% da dimensionalidade desse CD.

Com relação aos modelos induzidos pelo *MLP*, o subconjunto de atributos selecionado do CD *BreastCancer* pelo algoritmo *CBF* proporcionou ao MC o maior tempo de aprendizado (5,83 *segundos*). Já para *LungCancer* esse mesmo algoritmo selecionou o subconjunto que forneceu ao classificador induzido o menor tempo de aprendizado (0,32 *segundos*).

Observa-se que o indutor *MLP* apresentou-se mais custoso computacionalmente que o algoritmo de classificação *J48*. Um dos fatores que pode influenciar no custo do *MLP* é a retropropagação, a qual é utilizada para otimizar os pesos durante o treinamento.

Como foi possível constatar na maioria dos casos, os subconjuntos de atributos identificados pelos algoritmos de SA colaboraram direta ou indiretamente para a redução do tempo de aprendizado dos MC.

5.5 Análise das Categorias de Medidas de Importância de Atributos

Pode-se afirmar que um método de SA possui duas características principais: (1) uma medida de importância que avalia cada subconjunto candidato de atributos e (2) um processo de busca pelo melhor subconjunto de atributos, usualmente aquele que fornece o melhor resultado utilizando a medida de importância.

No caso dos métodos aplicados conforme a abordagem filtro, a propriedade fundamental a ser investigada é a maneira como os atributos são avaliados. Já nos métodos aplicados conforme a abordagem *wrapper* é necessária uma atenção especial aos métodos de busca, uma vez que são empregados algoritmos de AM para a avaliação dos subconjuntos de atributos.

Nesta seção são apresentadas algumas considerações relevantes relacionadas às categorias de medidas de importância de atributos consideradas. Além disso, são exibidas algumas das propriedades que possivelmente influenciam na escolha de um método de SA para um determinado experimento.

Na Tabela 5.8, as diferentes medidas de importância de atributos estudadas neste trabalho são comparadas quanto a três quesitos:

Medidas de Importância	Generalidade	Complexidade Temporal	Desempenho Preditivo
Medidas de Consistência	Sim	Média	—
Medidas de Correlação	Sim	Baixa	—
Medidas de Distância	Sim	Baixa	—
Medidas de Informação	Sim	Baixa	—
Medidas de Precisão	Não	Alta	Muito Alto

Tabela 5.8: Comparação entre as medidas de avaliação de importância de atributos (Dash and Liu, 1997).

- Generalidade: se o subconjunto de atributos selecionado pode ser utilizado em conjunto com distintos métodos de AM;
- Complexidade Temporal: tempo necessário para a seleção do subconjunto de atributos;
- Desempenho Preditivo: quão precisa é a classificação utilizando o subconjunto de atributos selecionado.

As informações dessa tabela estão organizadas do seguinte modo: na primeira coluna são apresentados os diferentes tipos de medidas de importância de atributos. Na segunda e terceira coluna são exibidas às classificações dessas medidas quanto ao quesito generalidade e complexidade temporal, respectivamente. Na última coluna as medidas de importância são classificadas de acordo com o quesito desempenho preditivo. Nessa coluna, o símbolo (—) indica que nada se pode concluir sobre o desempenho preditivo do MC a partir do subconjunto de atributos selecionado utilizando a medida de importância correspondente.

As informações presentes nessa tabela permitem aferir que a medida de precisão, apesar de garantir um alto desempenho preditivo, deve ser evitada quando há limitação de tempo. É mostrado também que o processo perde a generalidade quando a medida de importância usada for baseada no desempenho preditivo do classificador induzido. Contudo, estudos evidenciam que, em algumas situações, atributos selecionados utilizando o desempenho preditivo de um determinado algoritmo de AM para classificação podem ser utilizados com sucesso por outros classificadores (Kohavi and John, 1997).

Neste trabalho as vantagens da redução do conjunto de atributos foram apresentadas, analisadas experimentalmente e discutidas. Entretanto, é importante notar que a redução do conjunto de atributos pode não ser interessante em algumas situações, como em casos onde há poucos atributos ou, então, quando se sabe sobre a relevância dos dados.

Outra questão a ser considerada é que os algoritmos de SA existentes não são de uso geral. Comumente cada um desses métodos apresenta algumas

restrições com relação aos tipos de dados aos quais podem ser aplicados. A seguir são descritas três das principais características associadas aos dados que devem ser levadas em consideração para a escolha de um método de SA.

- Tipo dos dados: é necessário observar os tipos de valores que os atributos podem assumir (quantitativo ou qualitativo). Alguns métodos podem não ser capazes de tratar todos esses tipos de dados. Além disso, considerando as classes, a maioria dos métodos de aprendizado trabalham apenas com problemas de classificação em que cada exemplo possui uma única classe associada (monorrótulo). Porém, estudos demostram a proposição de diferentes algoritmos de aprendizado para lidar com múltiplas classes (multirrótulo) (Tsoumakas and Katakis, 2007);

- Dimensão dos dados: devem ser considerados tanto a capacidade de o algoritmo trabalhar com um conjunto de treinamento muito pequeno quanto com um grande volume de dados (Liu and Motoda, 2008);

- Presença de ruído: alguns métodos podem não ser capazes de manipular dados com ruído, conflitantes ou, então, dados desconhecidos. Neste último caso, muitas técnicas tem sido aplicadas, sendo algumas delas bastante simples como a substituição dos valores desconhecidos pela média ou mediana do atributo (Batista, 2003).

Por fim, é importante ressaltar que a SA já foi explorada com outras intenções além da redução do volume de dados como, por exemplo, a aplicação de técnicas não supervisionadas de SA em mineração de textos (Nogueira, 2009) e a utilização de atributos selecionados como atributos de saída em aprendizado multitarefa (Caruana and Sa, 2003; Sweden et al., 1998).

Capítulo 6

Considerações Finais

Neste trabalho foi apresentado um estudo comparativo entre as categorias de importância de atributos: clássica (informação, correlação e distância), consistência e precisão. Para tanto, foram investigados alguns dos principais algoritmos de SA que se baseiam em medidas representativas dessas categorias. Resultados experimentais obtidos com diversos CD demonstraram que a SA, por meio da redução da dimensionalidade, pode auxiliar na melhora da qualidade dos dados sob a perspectiva de desempenho preditivo, o que contribui para a construção de modelos de indução mais compreensíveis e com um menor custo computacional.

No decorrer deste trabalho, verificou-se que a pesquisa em SA tem permitido a proposição e o desenvolvimento de diversos algoritmos com o intuito de selecionar atributos (características) importantes, com respeito a um critério de avaliação específico.

Do ponto de vista prático, a escolha pela utilização de um determinado algoritmo de SA (ou um conjunto de algoritmos) deve ser conduzida de acordo com o conhecimento do domínio, em geral detido pelos especialistas do domínio, e o conhecimento de detalhes técnicos sobre os algoritmos de SA, usualmente detido por especialistas da área computacional (Parmezan, 2012).

No entanto, ao mesmo tempo em que diversos algoritmos estão disponíveis para a realização da tarefa de SA, cresce a dificuldade em se determinar, *a priori*, qual ou quais desses algoritmos seriam mais apropriados, de acordo com as características do problema (CD) e as características dos algoritmos de SA, uma vez que nenhum algoritmo pode ser considerado o melhor independentemente do problema abordado (Kalousis et al., 2004).

Uma das maneiras para auxiliar no problema da seleção/escolha de algoritmos é aplicar Meta-Aprendizado. Essa técnica baseia-se no acúmulo de experiência por meio do desempenho de múltiplas aplicações de um determinado método, diferentemente das estratégias tradicionais, as quais empregam conhecimento especialista ou processos delongados de avaliação empírica (Souza, 2010; Brazdil et al., 2009).

Dentre as aplicações mais usuais de Meta-Aprendizado, inclui-se o problema de gerar regras capazes de relacionar o desempenho de algoritmos de construção de modelos com as diversas características dos CD, isto é, construir meta-modelos para a recomendação efetiva de métodos de extração de padrões, baseada em meta-dados (meta-variáveis) extraídas dos CD e dos al-

goritmos de indução de modelos (Vilalta and Drissi, 2002; Domingos, 1997).

Como diferentes algoritmos de aprendizado são baseados em distintos conjuntos de suposições sobre os dados (*bias* ou viés de indução), é importante que haja flexibilidade. Em outras palavras, o algoritmo de aprendizado deve gerar bons modelos se o *bias* dos dados coincidir com o *bias* do algoritmo, além da relevância dos dados. Desse modo, a utilização de meta-dados, como propriedades do problema, propriedades dos algoritmos, baseadas, por exemplo, em medidas de precisão, ou ainda padrões previamente construídos, podem auxiliar a selecionar e/ou combinar diferentes métodos (Parmezan, 2012).

Os conceitos de Meta-Aprendizado foram empregados originalmente para selecionar algoritmos para tarefas de classificação. Nos últimos anos, esses conceitos têm sido explorados na seleção de algoritmos em outros domínios de aplicação, tais como previsão de séries temporais, sistemas de planejamento, otimização e bioinformática (Souza, 2010; Liu and Yu, 2005; Kalousis et al., 2004; Das, 2001).

A construção de meta-modelos especificamente para a recomendação de algoritmos de SA para um determinado experimento ainda é um assunto pouco explorado, embora constitua um importante problema relacionado ao processo de MD, mais precisamente em relação à fase de pré-processamento (Parmezan et al., 2012a,b)

Nesse contexto, como trabalhos futuros, pretende-se relacionar o comportamento dos algoritmos estudados neste trabalho com propriedades morfológicas extraídas dos CD, visando, por meio de métodos para Meta-Aprendizado, a recomendação de algoritmos mais apropriados para SA.

Como consequência de uma escolha apropriada de algoritmo(s) de SA, é possível obter auxílio na geração de CD de qualidade para a construção de modelos mais significativos em termos de desempenho e compreensibilidade.

Referências Bibliográficas

Abeel, T., Helleputte, T., de Peer, Y. V., Dupont, P., and Saeys, Y. (2010). Robust biomarker identification for cancer diagnosis with ensemble feature selection methods. *Bioinformatics*, 26(3):392–398.

Alpaydin, E. (2004). *Introduction to machine learning*. The MIT Press, Cambridge, United States of America.

Arauzo-Azofra, A., Benitez, J. M., and Castro, J. L. (2008). Consistency measures for feature selection. *Journal of Intelligent Information Systems*, 30(3):273–292.

Baranauskas, J. A. (2001). Extração automática de conhecimento por múltiplos indutores. Tese de Doutorado, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo.

Batista, G. E. A. P. A. (2003). Pré-processamento de dados em aprendizado de máquina supervisionado. Tese de Doutorado, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo.

Blum, A. L. and Langley, P. (1997). Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97:245–271.

Brazdil, P. B., Giraud-Carrier, C., Soares, C., and Vilalta, R. (2009). *Metalearning: applications to data mining*. Springer-Verlag, Berlin.

Caruana, R. and Sa, V. R. D. (2003). Benefitting from the variables that variable selection discards. *Journal of Machine Learning Research*, 3:1245–1264.

Covões, T. F. (2010). Seleção de atributos via agrupamento. Dissertação de Mestrado, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo.

Das, S. (2001). Filters, wrappers and a boosting based hybrid for feature selection. In *International Conference on Machine Learning*, pages 74–81. Williams College.

Dash, M. and Liu, H. (1997). Feature selection for classification. *Intelligent Data Analysis*, 1(3):131–156.

Dash, M. and Liu, H. (2000). Feature selection for clustering. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 110–121.

Dash, M. and Liu, H. (2003). Consistency-based search in feature selection. *Artificial Intelligence*, 151(1-2):155–176.

Deitel, P. J. and Deitel, H. M. (2010). *Java como programar*. Prentice Hall Press, São Paulo, Brasil, 8 edition.

Domingos, P. (1997). Why does bagging work? A bayesian account and its implications. In *International Conference on Knowledge Discovery and Data Mining*, pages 155–158.

Dy, J. G. and Brodley, C. E. (2000). Feature subset selection and order identification for unsupervised learning. In *International Conference on Machine Learning*, pages 247–254.

Frank, A. and Asuncion, A. (2010). UCI machine learning repository. http://archive.ics.uci.edu/ml.

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning*, 3:1157–1182.

Hall, M. (1999). *Correlation-based feature subset selection for machine learning*. PhD thesis, Department of Computer Science, University of Waikato.

Hall, M. A. (2000). Correlation-based feature selection for discrete and numeric class machine learning. In *International Conference on Machine Learning*, pages 359–366.

Han, J., Kamber, M., and Pei, J. (2011). *Data mining: concepts and techniques*. Morgan Kaufmann, California, United States of America, 3 edition.

Haykin, S. S. (2009). *Neural networks and learning machines*. Prentice Hall, Saddle River, New Jersey, 3 edition.

He, X., Cai, D., and Niyogi, P. (2005). Laplacian score for feature selection. In *Advances in Neural Information Processing Systems*, pages 507–514.

John, G., Kohavi, R., and Pfleger, K. (1994). Irrelevant features and the subset selection problem. In *International Conference on Machine Learning*, pages 167–173.

Kalousis, A., Gama, J., and Hilario, M. (2004). On data and algorithms: understanding inductive performance. *Machine Learning*, 54:275–312.

Kitchenham, B. A. (2007). Guidelines for performing systematic literature reviews in software engineering. Technical report, Evidence-based Software Engineering, United Kingdom.

Kohavi, R. and John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324.

Kononenko, I. (1994). Estimating attributes: analysis and extention of Relief. In *European Conference on Machine Learning*, pages 171–182, Amsterdam. Springer-Verlag.

Langley, P. (1994). Selection of relevant features in machine learning. pages 140–144. AAAI Fall Symposium on Relevance.

Lee, H. D. (2005). Seleção de atributos importantes para a extração de conhecimento de bases de dados. Tese de Doutorado, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo.

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.

Lee, H. D., Monard, M. C., and Wu, F. C. (2006). A fractal dimension based filter algorithm to select features for supervised learning. *Lecture Notes in Computer Science*, 4140:278–288.

Lee, J. W. and Giraud-Carrier, C. (2008). Predicting algorithm accuracy with a small set of effective meta-features. In *International Conference on Machine Learning and Applications*, pages 808–812.

Liu, H. and Motoda, H. (1998). *Feature selection for knowledge discovery and data mining*. Kluwer Academic Publishers, Massachusetts, United States of America.

Liu, H. and Motoda, H. (2008). *Computational methods of feature selection*. Chapman & Hall/CRC data mining and knowledge discovery, Minnesota, United States of America.

Liu, H., Motoda, H., and Yu, L. (2004). A selective sampling approach to active feature selection. *Artificial Intelligence*, 159:49–74.

Liu, H. and Setiono, R. (1996). A probabilistic approach to feature selection - A filter solution. In *International Conference on Machine Learning*, pages 319–327.

Liu, H. and Yu, L. (2002). Feature selection for data mining. Department of Computer Science and Engineering, Arizona State University.

Liu, H. and Yu, L. (2005). Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on Knowledge and Data Engineering*, 17:491–502.

Liu, H., Yu, L., Dash, M., and Motoda, H. (2003). Active feature selection using classes. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 474–485.

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.

Mitchell, T. M. (1997). *Machine learning*. McGraw-Hill, United States of America.

Mitra, P., Murthy, C. A., and Pal, S. K. (2002). Unsupervised feature selection using feature similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):301–312.

Nogueira, B. M. (2009). Avaliação de métodos não-supervisionados de seleção de atributos para mineração de textos. Dissertação de Mestrado, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo.

Parmezan, A. R. S. (2012). Proposta de um módulo de monitoramento de qualidade de dados com assistência inteligente: um estudo de caso para o sistema médico de auxílio à cirurgia coloproctológica. Monografia de Graduação, Centro de Engenharias e Ciências Exatas, Universidade Estadual do Oeste do Paraná.

Parmezan, A. R. S., Lee, H. D., Ferrero, C. A., Zalewski, W., Maletzke, A. G., and Wu, F. C. (2010). Estudo comparativo entre métodos de seleção de atributos baseados em medidas de precisão e correlação aplicados a bases de dados. *XVIII Simpósio Internacional de Iniciação Científica da Universidade de São Paulo*, pages 1–1.

Parmezan, A. R. S., Lee, H. D., and Wu, F. C. (2011a). Redução da dimensionalidade em bases de dados naturais através de métodos de filtro para seleção de atributos importantes. *XIX Simpósio Internacional de Iniciação Científica da Universidade de São Paulo*, pages 1–1.

Parmezan, A. R. S., Lee, H. D., and Wu, F. C. (2012a). Estudo preliminar da construção de um modelo de recomendação de algoritmos de seleção de atributos utilizando meta-aprendizado. *XX Simpósio Internacional de Iniciação Científica da Universidade de São Paulo*, pages 1–1.

Parmezan, A. R. S., Wu, F. C., and Lee, H. D. (2011b). Estudo de medidas de importância e algoritmos para seleção de atributos para mineração de dados. *XX Encontro Anual de Iniciação Científica*, pages 1–4.

Parmezan, A. R. S., Wu, F. C., and Lee, H. D. (2012b). Meta-aprendizado no auxílio à seleção de atributos: um estudo para medidas de correlação e consistência. *XXI Encontro Anual de Iniciação Científica*, pages 1–4.

Pila, A. D. (2001). Seleção de atributos relevantes para aprendizado de máquina utilizando a abordagem de rough sets. Dissertação de Mestrado, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo.

Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. (1992). *Numerical recipes in C: the art of scientific computing, Second Edition*. Cambridge University Press, 2 edition.

Pyle, D. (1999). *Data preparation for data mining*. Morgan Kaufmann, California, United States of America.

Rezende, S. O. (2003). *Sistemas inteligentes: fundamentos e aplicações*. Editora Manole, São Paulo, Brasil.

Robnik-Sikonja, M. and Kononenko, I. (2003). Theoretical and empirical analysis of ReliefF and RReliefF. *Machine Learning*, 53(1-2):23–69.

Schlimmer, J. C. (1993). Efficiently inducing determinations: a complete and systematic search algorithm that uses optimal pruning. In *International Conference on Machine Learning*, pages 284–290.

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 on Fractals and Self-similarity in Data Mining: Issues and Approaches*, pages 26–30.

Souza, B. F. (2010). Meta-aprendizagem aplicada à classificação de dados de expressão gênica. Tese de Doutorado, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo.

Spolaôr, N. (2010). Aplicação de algoritmos genéticos multiobjetivo ao problema de seleção de atributos. Dissertação de Mestrado, Universidade Federal do ABC.

Spolaôr, N., Cherman, E. A., and Monard, M. C. (2011a). Uso do ReliefF para seleção de atributos em dados multirrótulo. In *XXXVII Conferencia Latinoamericana de Informática*, pages 960–975.

Spolaôr, N., Lorena, A. C., and Lee, H. D. (2010). Uma revisão sistemática sobre aplicações de metaheurísticas multiobjetivo ao problema de seleção de atributos. Relatório Técnico, Universidade Federal do ABC.

Spolaôr, N., Lorena, A. C., and Lee, H. D. (2011b). Algoritmos genéticos multiobjetivo para a seleção de stributos. *VIII Encontro Nacional de Inteligência Artificial*, pages 938–949.

Spolaôr, N., Monard, M. C., and Lee, H. D. (2012). A systematic review to identify feature selection publications in multi-labeled data. Technical Report 374, Institute of Mathematics and Computer Science, University of São Paulo.

Sweden, S., Caruana, R., and Sa, V. R. D. (1998). Using feature selection to find inputs that work better as extra outputs. In *International Conference on Artificial Neural Networks*, pages 1–6.

Talavera, L. (1999). Feature selection as a preprocessing step for hierarchical clustering. In *International Conference on Machine Learning*, pages 389–397.

Traina, C., Traina, A. J. M., Wu, L., and Faloutsos, C. (2000). Fast feature selection using fractal dimension. In *Brazilian Data Base Symposium*, pages 158–171.

Tsoumakas, G. and Katakis, I. (2007). Multi-label classification: an overview. *International Journal of Data Warehousing and Mining*, 2007:1–13.

Vilalta, R. and Drissi, Y. (2002). A perspective view and survey of meta-learning. *Artificial Intelligence Review*, 18:77–95.

Wang, C.-M. and Huang, Y.-F. (2009). Evolutionary-based feature selection approaches with new criteria for data mining: a case study of credit approval data. *Expert Systems with Applications*, 36(3-2):5900–5908.

Witten, I. H. and Frank, E. (2005). *Data mining: practical machine learning tools and techniques*. Morgan Kaufmann, California, United States of America.

Yu, L. and Liu, H. (2004). Efficient feature selection via analysis of relevance and redundancy. *Journal of Machine Learning Research*, 5:1205–1224.

Apêndice A

Atributos Selecionados

Neste apêndice localizam-se as tabelas referentes aos atributos selecionados pelos métodos de SA, complementando a Seção 5.1.

As informações dessas tabelas estão organizadas do seguinte modo: na primeira coluna são exibidos os tipos dos atributos, enquanto que na segunda coluna são apresentados cada atributo que compõem o respectivo CD. Nas demais colunas são indicados os atributos selecionados (•) pelos algoritmos utilizados neste trabalho. Por fim, na última linha são mostradas as respectivas quantidades de atributos selecionados (# Atributos) por cada algoritmo de SA.

BreastCancer

Tipo	**Atributos**	***CBF***	***CFS***	***CSE*(J)**	***CSE*(M)**	***InfoGain***	***ReliefF***
Nominal	*age*	•			•	6	4
Nominal	*menopause*	•			•	9	•3
Nominal	*tumor_size*	•	•		•	•3	6
Nominal	*inv_nodes*	•	•	•	•	•2	9
Nominal	*node_caps*		•	•		4	8
Nominal	*deg_malig*	•	•	•	•	•1	•1
Nominal	*breast*	•		•	•	8	5
Nominal	*breast_quad*	•			•	7	•2
Nominal	*irradiat*	•	•	•	•	5	7
# Atributos	9	8	5	5	8	3	3

Tabela A.1: Atributos selecionados — *BreastCancer*.

Bupa

Tipo	Atributos	*CBF*	*CFS*	*CSE*(J)	*CSE*(M)	*InfoGain*	*ReliefF*
Numérico	*mcv*				•	3	5
Numérico	*alkphos*					•2	6
Numérico	*sgpt*			•	•	5	•1
Numérico	*sgot*			•	•	6	4
Numérico	*gammagt*	•	•	•	•	•1	•2
Numérico	*drinks*			•	•	4	3
# Atributos	6	1	1	4	5	2	2

Tabela A.2: Atributos selecionados — *Bupa*.

Haberman

Tipo	Atributos	*CBF*	*CFS*	*CSE*(J)	*CSE*(M)	*InfoGain*	*ReliefF*
Numérico	*attribute_01*			•	•	3	3
Nominal	*attribute_02*	•	•	•	•	2	•1
Numérico	*attribute_03*	•	•	•	•	•1	2
# Atributos	3	2	2	3	3	1	1

Tabela A.3: Atributos selecionados — *Haberman*.

Hepatitis

Tipo	Atributos	*CBF*	*CFS*	*CSE*(J)	*CSE*(M)	*InfoGain*	*ReliefF*
Numérico	*age*	•	•		•	10	16
Nominal	*sex*		•	•		11	17
Nominal	*steroid*	•		•	•	14	7
Nominal	*antivirals*				•	13	12
Nominal	*fatigue*					•6	•4
Nominal	*malaise*	•	•		•	7	•2
Nominal	*anorexia*			•	•	15	18
Nominal	*liver_big*	•			•	16	13
Nominal	*liver_firm*	•				17	9
Nominal	*spleen_palpable*	•		•		12	•6
Nominal	*spiders*	•	•		•	•4	•1
Nominal	*ascites*	•	•	•	•	•3	•5
Nominal	*varices*		•		•	8	8
Numérico	*bilirubin*	•	•		•	•2	15
Numérico	*alk_phosphate*					19	11
Numérico	*sgot*	•				18	19
Numérico	*albumin*	•	•	•	•	•1	10
Numérico	*protime*	•	•		•	9	14
Nominal	*histology*		•			•5	•3
# Atributos	19	12	10	6	12	6	6

Tabela A.4: Atributos selecionados — *Hepatitis*.

Hungarian

Tipo	**Atributos**	***CBF***	***CFS***	***CSE*(J)**	***CSE*(M)**	***InfoGain***	***ReliefF***
Numérico	*age*			•	•	10	7
Nominal	*sex*	•	•	•	•	5	•2
Nominal	*cp*	•	•	•	•	•1	•1
Numérico	*trestbps*				•	13	8
Numérico	*chol*	•		•	•	11	11
Nominal	*fbs*	•	•		•	7	9
Nominal	*restecg*	•			•	8	•4
Numérico	*thalach*	•		•	•	•4	10
Nominal	*exang*	•	•	•	•	•2	5
Numérico	*oldpeak*	•	•	•	•	•3	•3
Nominal	*slope*	•	•	•	•	6	6
Numérico	*ca*				•	12	13
Nominal	*thal*	•			•	9	12
# Atributos	13	10	6	8	13	4	4

Tabela A.5: Atributos selecionados — *Hungarian*.

LungCancer

Tipo	**Atributos**	***CBF***	***CFS***	***CSE*(J)**	***CSE*(M)**	***InfoGain***	***ReliefF***
Nominal	*attribute01*	•	•		•	•2	•11
Nominal	*attribute02*		•			29	32
Nominal	*attribute03*	•			•	20	35
Nominal	*attribute04*					56	55
Nominal	*attribute05*					37	41
Nominal	*attribute06*					54	52
Nominal	*attribute07*					36	39
Nominal	*attribute08*					27	26
Nominal	*attribute09*	•	•	•	•	•1	•1
Nominal	*attribute10*					38	53
Nominal	*attribute11*					42	47
Nominal	*attribute12*					28	44
Nominal	*attribute13*					18	42
Nominal	*attribute14*		•			•3	•7
Nominal	*attribute15*					•17	•16
Nominal	*attribute16*					40	24
Nominal	*attribute17*					45	49
Nominal	*attribute18*					47	21
Nominal	*attribute19*					48	36
Nominal	*attribute20*					•8	•3
Nominal	*attribute21*					•5	•10
Nominal	*attribute22*					32	•17
Nominal	*attribute23*					55	45
Nominal	*attribute24*					31	•8
Nominal	*attribute25*					22	46
Nominal	*attribute26*					53	40
Nominal	*attribute27*					39	43
Nominal	*attribute28*					52	54
Nominal	*attribute29*					51	56
Nominal	*attribute30*					46	50
Nominal	*attribute31*					44	34
Nominal	*attribute32*					34	20
Nominal	*attribute33*					33	19
Nominal	*attribute34*					21	22
Nominal	*attribute35*					50	51
Nominal	*attribute36*					43	27
Nominal	*attribute37*					24	38
Nominal	*attribute38*					•13	•6
Nominal	*attribute39*					•15	•12
Nominal	*attribute40*					35	31
Nominal	*attribute41*					•16	•4
Nominal	*attribute42*	•	•	•	•	•6	•2
Nominal	*attribute43*					23	•9
Nominal	*attribute44*					•14	•5
Nominal	*attribute45*		•			•4	37
Nominal	*attribute46*					•7	33
Nominal	*attribute47*					•10	•13
Nominal	*attribute48*		•			•12	•14
Nominal	*attribute49*					•11	•15
Nominal	*attribute50*					•9	23
Nominal	*attribute51*					41	29
Nominal	*attribute52*					49	30
Nominal	*attribute53*					25	25
Nominal	*attribute54*					26	28
Nominal	*attribute55*		•			30	18
Nominal	*attribute56*					19	48
# Atributos	56	4	8	2	4	17	17

Tabela A.6: Atributos selecionados — *LungCancer*.

Pima

Tipo	**Atributos**	***CBF***	***CFS***	***CSE*(J)**	***CSE*(M)**	***InfoGain***	***ReliefF***
Numérico	*preg*	•		•	•	6	4
Numérico	*plas*	•	•	•	•	•1	•1
Numérico	*pres*	•		•	•	8	7
Numérico	*skin*	•			•	5	3
Numérico	*insu*	•			•	4	8
Numérico	*mass*	•	•	•	•	•2	•2
Numérico	*pedi*	•	•	•	•	7	6
Numérico	*age*	•	•	•	•	3	5
# Atributos	8	8	4	6	8	2	2

Tabela A.7: Atributos selecionados — *Pima*.

Apêndice B

Desempenho dos Algoritmos em Relação ao Modelo de Categorização

De modo a complementar os resultados introduzidos na Seção 5.2 e considerando os indutores *J48* e *MLP*, neste apêndice são apresentados, para os demais CD que obtiveram ESA < ECM, as figuras resultantes do emprego do modelo de categorização de algoritmos em relação ao desempenho preditivo e a percentagem de atributos selecionados.

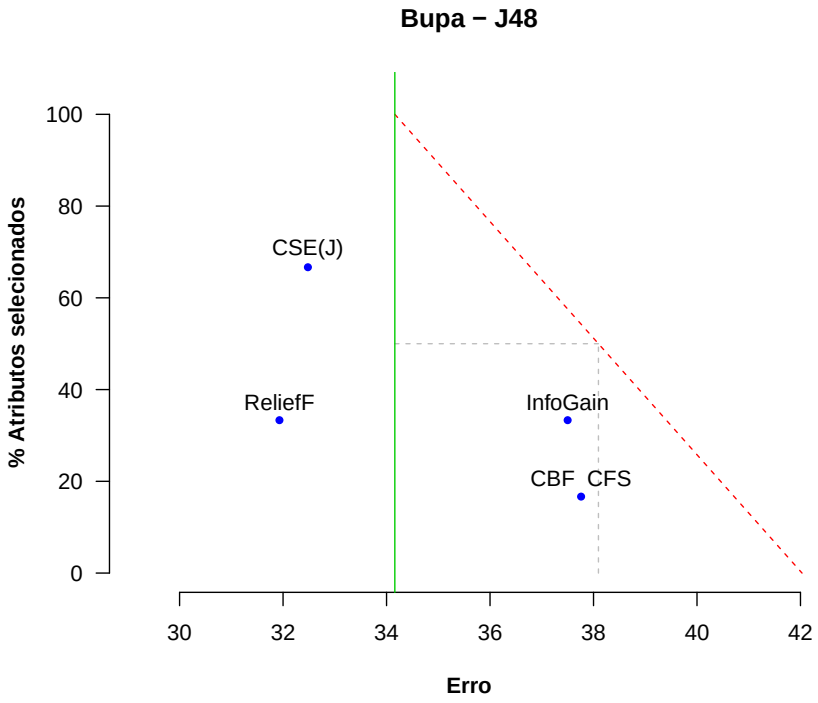


Figura B.1: Modelo de categorização de algoritmos aplicado sobre *Bupa* considerando o indutor *J48*.

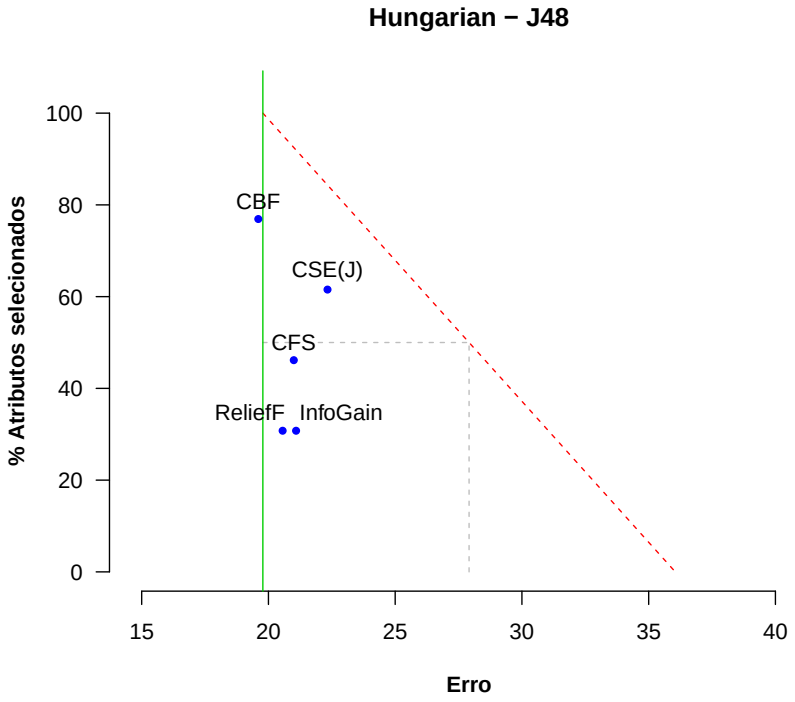


Figura B.2: Modelo de categorização de algoritmos aplicado sobre *Hungarian* considerando o indutor *J48*.

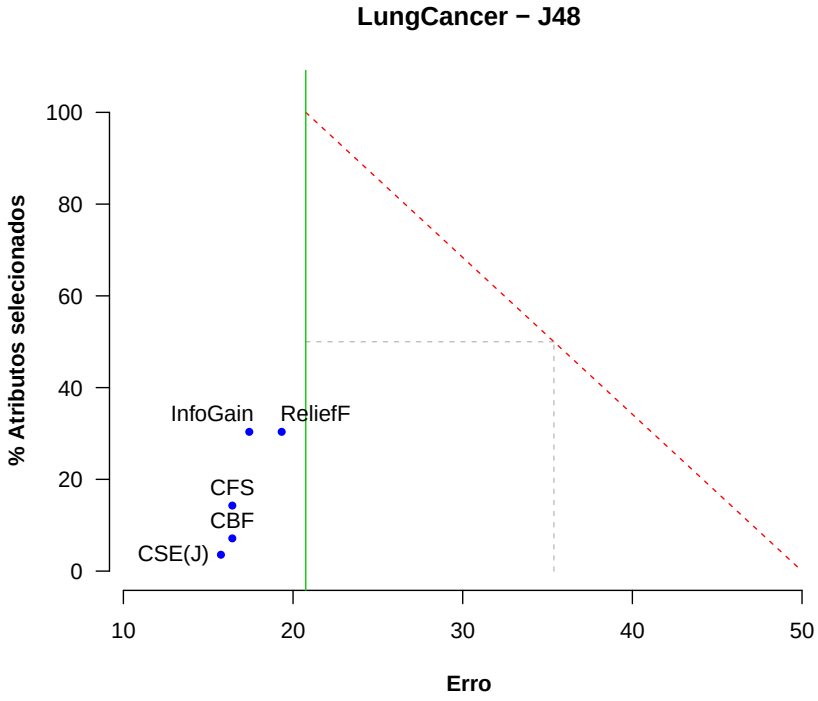


Figura B.3: Modelo de categorização de algoritmos aplicado sobre *LungCancer* considerando o indutor *J48*.

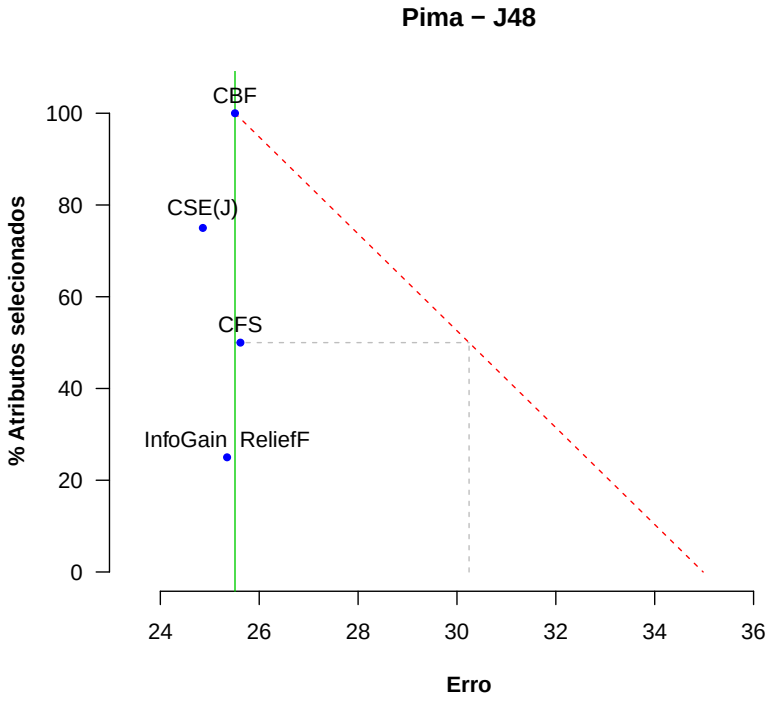


Figura B.4: Modelo de categorização de algoritmos aplicado sobre *Pima* considerando o indutor *J48*.

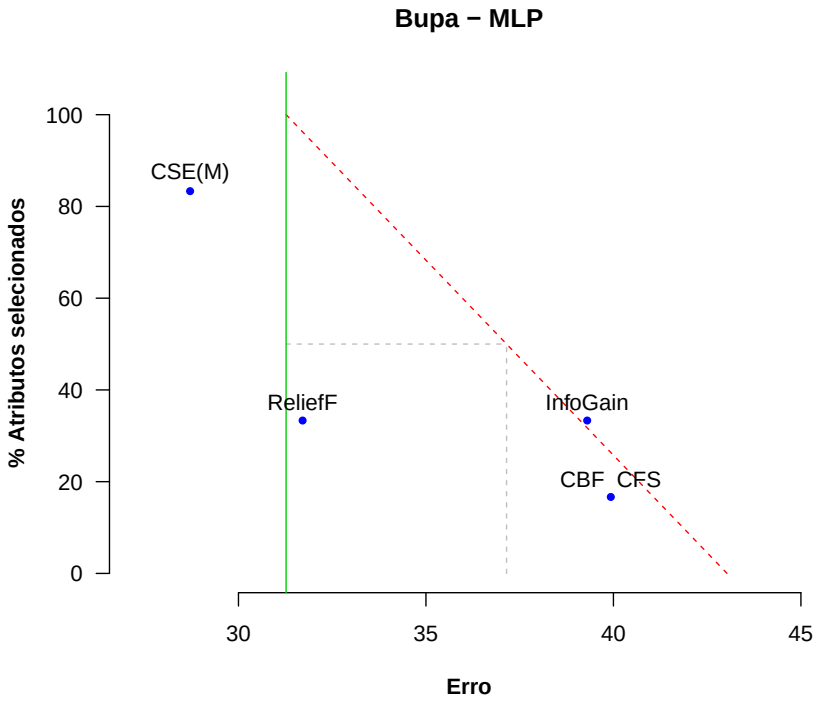


Figura B.5: Modelo de categorização de algoritmos aplicado sobre *Bupa* considerando o indutor *MLP*.

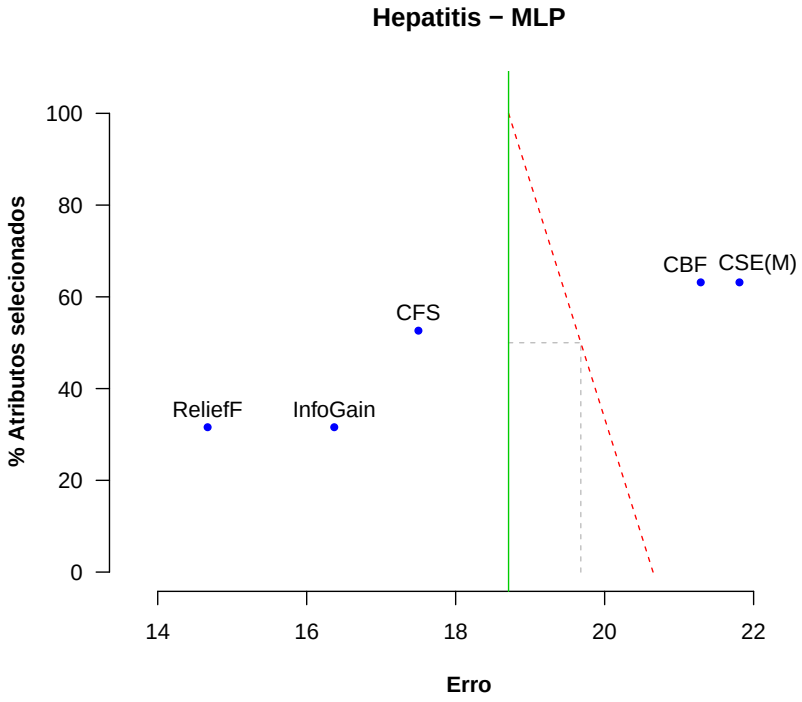


Figura B.6: Modelo de categorização de algoritmos aplicado sobre *Hepatitis* considerando o indutor *MLP*.

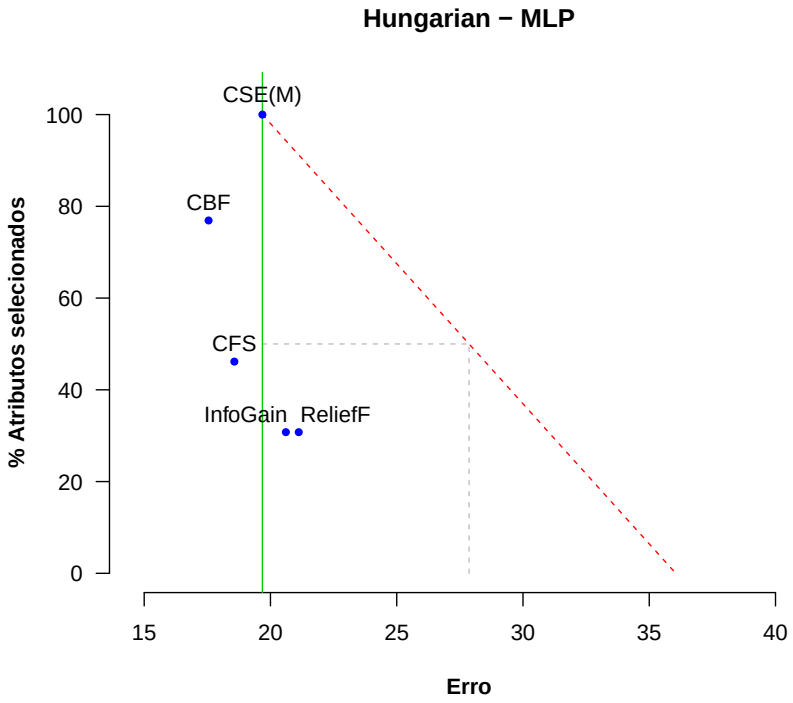


Figura B.7: Modelo de categorização de algoritmos aplicado sobre *Hungarian* considerando o indutor *MLP*.

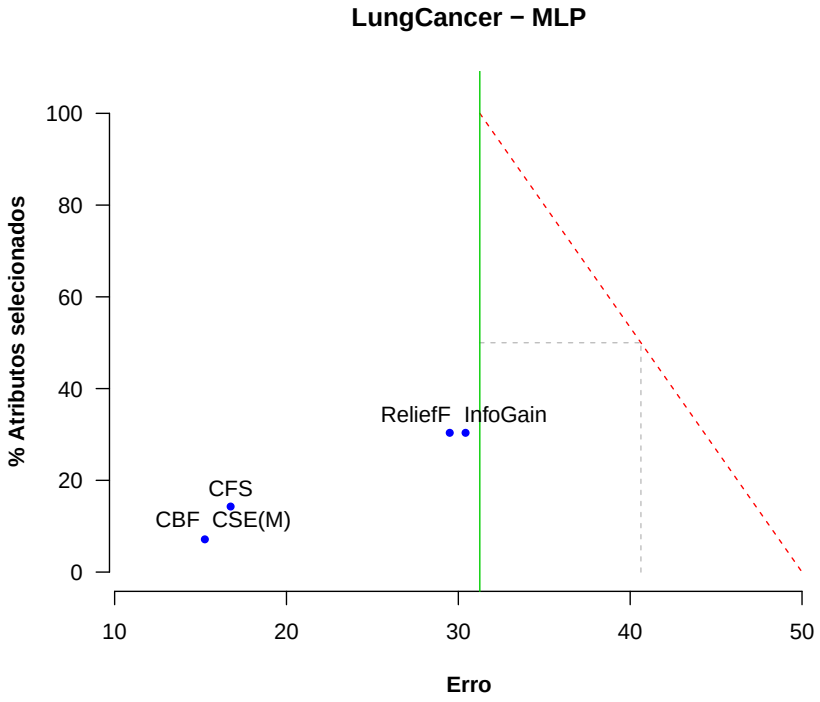


Figura B.8: Modelo de categorização de algoritmos aplicado sobre *LungCancer* considerando o indutor *MLP*.

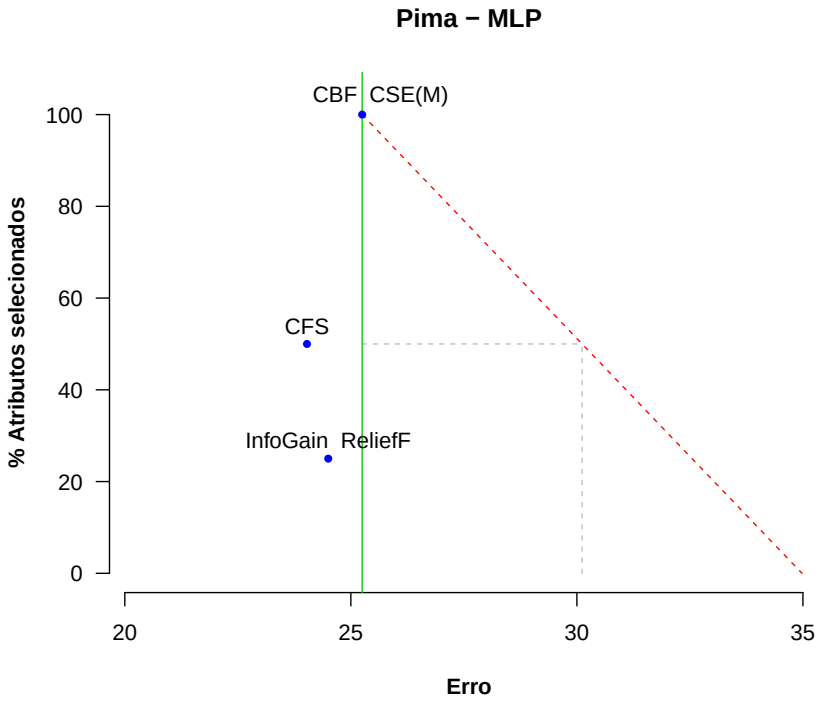


Figura B.9: Modelo de categorização de algoritmos aplicado sobre *Pima* considerando o indutor *MLP*.

Apêndice C

Tempo de Aprendizado

Neste apêndice são apresentadas as tabelas referentes aos tempos (em segundos) necessários para a construção dos modelos, por meio dos algoritmos de classificação *J48* e *MLP*, utilizando os CD Originais e os subconjuntos de atributos selecionados. Nessas tabelas os tempos de aprendizado são relacionados com a percentagem do número de atributos e o número de exemplos presentes nos conjuntos e subconjuntos de dados investigados.

As informações dessas tabelas estão organizadas do seguinte modo: na primeira coluna é apresentado o CD ao qual referem-se às informações seguido, entre parênteses, pelo seu respectivo número de exemplos (N). Na segunda coluna, para cada CD, na primeira linha é indicada a percentagem original de atributos e, na segunda linha é mostrado o tempo de aprendizado do MC. Nas demais colunas, para cada uma das células, na primeira linha são exibidas as percentagens de atributos selecionados pelos métodos de SA, enquanto que na segunda linha é apresentado o respectivo tempo necessário para a indução do modelo. Por fim, na última linha dessas tabelas é exibida a percentagem média de atributos selecionados e a média do tempo de aprendizado dos MC a partir dos subconjuntos de dados identificados.

	J48					
Conjunto de Dados (*N*)	**Original**	***CBF***	***CFS***	***CSE*(J)**	***InfoGain***	***ReliefF***
BreastCancer	100,00	88,89	55,56	55,56	33,33	33,33
(286)	0,09	0,06	0,04	0,06	0,08	0,03
Bupa	100,00	16,67	16,67	66,67	33,33	33,33
(345)	0,19	0,03	0,03	0,20	0,06	0,05
Haberman	100,00	66,67	66,67	100,00	33,33	33,33
(306)	0,07	0,10	0,04	0,09	0,03	0,01
Hepatitis	100,00	63,16	52,63	31,58	31,58	31,58
(155)	0,10	0,10	0,07	0,06	0,06	0,02
Hungarian	100,00	76,92	46,15	61,54	30,77	30,77
(294)	0,22	0,17	0,16	0,10	0,05	0,06
LungCancer	100,00	7,14	14,29	3,57	30,36	30,36
(32)	0,03	0,02	0,02	0,03	0,02	0,02
Pima	100,00	100,00	50,00	75,00	25,00	25,00
(768)	0,24	0,23	0,19	0,32	0,10	0,14
Média		59,92	43,14	56,27	31,10	31,10
		0,10	0,08	0,12	0,06	0,05

Tabela C.1: Relação entre a percentagem do número de atributos, o número de exemplos e o tempo de aprendizado dos MC utilizando o indutor *J48*.

	MLP					
Conjunto de Dados (*N*)	**Original**	***CBF***	***CFS***	***CSE*(M)**	***InfoGain***	***ReliefF***
BreastCancer	100,00	88,89	55,56	88,89	33,33	33,33
(286)	6,10	5,83	3,03	5,05	2,41	2,12
Bupa	100,00	16,67	16,67	83,33	33,33	33,33
(345)	1,62	0,94	0,93	0,96	1,39	1,56
Haberman	100,00	66,67	66,67	100,00	33,33	33,33
(306)	1,36	1,15	1,15	1,27	0,93	1,00
Hepatitis	100,00	63,16	52,63	63,16	31,58	31,58
(155)	1,64	1,55	0,60	1,37	1,52	0,84
Hungarian	100,00	76,92	46,15	100,00	30,77	30,77
(294)	1,97	1,38	1,78	0,26	1,15	1,36
LungCancer	100,00	7,14	14,29	7,14	30,36	30,36
(32)	4,81	0,32	1,39	0,60	1,18	0,84
Pima	100,00	100,00	50,00	100,00	25,00	25,00
(768)	2,93	2,50	2,03	1,56	1,57	1,83
Média		59,92	43,14	75,50	31,10	31,10
		1,95	1,56	1,58	1,45	1,36

Tabela C.2: Relação entre a percentagem do número de atributos, o número de exemplos e o tempo de aprendizado dos MC utilizando o indutor *MLP*.