Proposta de um Algoritmo de Seleção de Atributos Importantes para Aprendizado Supervisionado Utilizando a Dimensão Fractal para Tratamento de Redundância: Avaliação Experimental *

Huei Diana Lee[1 2] **, Maria Carolina Monard**[2] **,**
Richardson Floriani Voltolini[2] **, Feng Chung Wu**[1 3]

[1]Laboratório de Bioinformática (LABI) – Universidade Estadual do Oeste do Paraná
Caixa Postal 961, 85870-650 – Foz do Iguaçu, PR, Brasil

[2]Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo
Caixa Postal 668, 13560-970 – São Carlos, SP, Brasil

[3]Instituto de Tecnologia em Automação e Informática (ITAI) – Foz do Iguaçu, PR, Brasil

huei@unioeste.br, mcmonard@icmc.usp.br, rvf@icmc.usp.br, wufc@unioeste.br

Resumo. *Em aprendizado de máquina, a tarefa de pré-processamento do conjunto de dados inclui selecionar os atributos mais importantes para realizar o aprendizado. A seleção de atributos é de fundamental importância pois, no caso de aprendizado supervisionado, atributos não relevantes ou redundantes podem reduzir a precisão e a compreensibilidade das hipóteses induzidas por esses algoritmos. Vários algoritmos para a seleção de atributos relevantes têm sido propostos na literatura. Entretanto, tem sido observado que somente o critério de relevância não é suficiente para a seleção de atributos importantes. Trabalhos recentes têm mostrado que também deve-se levar em conta o critério de redundância para selecionar os atributos importantes, pois atributos redundantes afetam a qualidade das hipóteses induzidas. Vários modelos têm sido propostos para tratar tanto relevância quanto redundância de atributos, porém, alguns desses modelos apresentam um custo computacional muito alto. Um modelo mais recente sugere realizar o tratamento de relevância e redundância como dois processos separados. A vantagem desse modelo é que, por meio dessa separação, é possível diminuir o custo computacional da busca pelo subconjunto que aproxima o subconjunto ótimo de atributos. Neste trabalho é proposto um algoritmo que separa as análises de relevância e de redundância. Nesse algoritmo encontram-se implementadas duas medidas para realizar a análise de relevância, uma medida baseada em ganho de informação e outra baseada em distância. Quanto à redundância, é proposto o uso da Dimensão Fractal do subconjunto de atributos relevantes selecionados na etapa anterior. Resultados experimentais utilizando vários conjuntos de dados e diversos algoritmos que selecionam atributos importantes, mostram que a Dimensão Fractal é um critério apropriado para filtrar atributos redundantes no aprendizado supervisionado.*

Palavras-Chaves: *Seleção de Atributos; Dimensão Fractal; Aprendizado de Máquina.*

1. Introdução

Em aprendizado supervisionado, a indução de um classificador $\mathbf{h}$ é influenciado pelos valores dos atributos do conjunto de exemplos. Teoricamente, o uso de um maior número de atributos para descrever os exemplos deveria fornecer um maior poder de discriminação para aproximar a verdadeira função $y = f(\mathbf{x})$ desconhecida, onde $\mathbf{x}$ representa um exemplo do conjunto de dados, composto por N exemplos, na forma $\{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_N, y_N)\}$ rotulados com

*Trabalho desenvolvido com o apoio do Instituto de Tecnologia em Automação e Informática – ITAI, do Parque Tecnológico de Itaipu – PTI e da FAPESP Processo nº 04/04885-8.

os valores y, sendo $\mathbf{x}_i$ vetores da forma $(x_{i1}, x_{i2}, ..., x_{iM})$ cujos componentes são valores discretos ou contínuos relacionados aos *atributos* $A = \{A_1, A_2, ...A_M\}$. No caso dos valores y pertencerem a um conjunto discreto de N_{C_l} classes, *i.e.* $y \in \{C_1, ..., C_{N_{C_l}}\}$, a tarefa de aprendizado é chamada de *classificação*, tratada neste trabalho. Porém, isso pode não ocorrer na presença de atributos irrelevantes e/ou redundantes, os quais, freqüentemente, confundem o algoritmo de aprendizado. A área de Seleção de Atributos — SA —, explorada não somente em estatística mas também em Aprendizado de Máquina — AM — e Mineração de Dados — MD (Liu and Motoda, 1998), tem como objetivo determinar, segundo algum critério, que atributos são importantes. Os resultados obtidos, tanto teórica quanto experimentalmente, mostram que a SA melhora a predição de classificadores e reduz a complexidade do modelo $\mathbf{h}$.

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

Os diversos modelos de SA propostos na literatura podem ser categorizados nos modelos *wrapper* e filtro (Liu and Motoda, 1998). Além dos atributos irrelevantes, tem sido observado que atributos *redundantes* também afetam a precisão dos classificadores induzidos e, portanto, deveriam ser eliminados (Koller and Sahami, 1996; Hall, 2000). Considera-se que dois atributos são redundantes entre si quando seus valores estão correlacionados, parcial ou completamente. O seguinte exemplo, citado freqüentemente na literatura, ilustra esse conceito: considerando o conjunto $A = \{A_1, A_2, A_3, A_4, A_5\}$ de atributos e $y = f(A1, A2)$ uma função booleana, há somente oito possíveis exemplos tal que $A_2 = \overline{A}_3$ e $A_4 = \overline{A}_5$. Assim, para determinar o conceito meta tem-se: A_1 é indispensável; A_2 ou A_3, mas não ambos, podem ser ignorados já que $y = f(A1, \overline{A}_3)$; A_4 e A_5 podem ser ignorados. Nesse caso, existem dois subconjuntos A' ótimos, $\{A_1, A_2\}$ e $\{A_1, A_3\}$, e a meta da SA é encontrar pelo menos um desses subconjuntos. Entretanto, o número de subconjuntos de atributos cresce exponencialmente com o número de atributos em A e encontrar o subconjunto ótimo de atributos pode ser NP (Kohavi and John, 1997).

Em geral, os métodos de SA selecionam os atributos pela *avaliação individual* ou pela *avaliação de subconjuntos* de atributos. No caso de avaliação individual, os atributos são ordenados considerando a sua importância na discriminação das classes. Esses métodos somente removem atributos irrelevantes pois espera-se que atributos redundantes tenham a mesma importância na discriminação das classes. Contudo, métodos que avaliam subconjuntos de atributos procurando por subconjuntos mínimos podem remover tanto atributos irrelevantes quanto redundantes. Assim, a maioria dos métodos existentes para a SA que tratam tanto relevância quanto redundância de atributos, o fazem de maneira implícita por meio da avaliação de subconjuntos de atributos. Ainda que esses métodos geralmente apresentem melhores resultados que os métodos que não lidam com a redundância de atributos, seu alto custo computacional pode torná-los ineficientes para conjuntos de dados com alta dimensionalidade. Recentemente foi proposto o uso da abordagem filtro considerando o modelo de tratamento da relevância e da redundância de atributos como dois procedimentos separados (Yu and Liu, 2004). A vantagem desse modelo sobre o modelo anterior é que, por meio da separação da análise de relevância e de redundância, é possível diminuir o custo computacional da busca por um subconjunto que aproxima o subconjunto ótimo .

Neste trabalho propomos um algoritmo para a SA, baseado no modelo proposto por Yu and Liu (2004), utilizando a Dimensão Fractal —DF — como procedimento para tratar a redundância de atributos (Lee and Monard, 2003). Ainda que o conceito de DF seja freqüentemente utilizado na detecção de agrupamento de dados e na indexação de estruturas de alta dimensionalidade, não é de nosso conhecimento que a DF tenha sido utilizada para realizar SA para algoritmos de aprendizado de máquina supervisionados, como proposto neste trabalho. Resultados experimentais obtidos com diversos conjuntos de dados utilizando diferentes filtros para a SA e o algoritmo por nós proposto, que usa a DF para tratar redundância, são apresentados. Esses resultados mostram que a DF é um critério apropriado para tratar redundância de atributos.

Este trabalho está organizado do seguinte modo: nas Seções 2 e 3 são apresentados brevemente conceitos sobre fractais e dimensão fractal. Na Seção 4 é descrito o algoritmo proposto neste trabalho. Na Seção 5 são descritos os conjuntos de dados utilizados. A configuração dos experimentos é descrita na Seção 6. Resultados e discussão dos experimentos realizados são apresentados na Seção 7 e considerações finais são apresentadas na Seção 8.

2. Fractais

Fractais são definidos pela propriedade de auto-similaridade, ou seja, apresentam, parcial ou integralmente, as mesmas características para diferentes variações na escala em que estão sendo analisados. Assim, partes do fractal, o qual pode ser uma estrutura, um objeto ou um conjunto de dados, são similares, exata ou estatisticamente, ao fractal como um todo. Fractais possuem, em geral, características incomuns, por exemplo, o conhecido Triângulo de Sierpinsky — Figura 1. Ele não pode ser considerado um objeto Euclidiano unidimensional, pois possui perímetro infinito, nem tão pouco um objeto Euclidiano bidimensional já que possui área nula. Dessa maneira, pode-se considerar uma dimensão fracionária, denominada de dimensão fractal (Mandelbrot, 1985). Muitos dos conjuntos de dados reais comportam-se como fractais. Desse modo, torna-se natural a idéia de aplicar conceitos da teoria dos fractais para a análise desses conjuntos de dados (Faloutsos and Kamel, 1994).

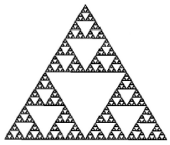

Figura 1: Triângulo de Sierpinsky

3. Dimensão Fractal de um Conjunto de Dados

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

Pode-se definir, desse modo, as idéias de *dimensão imersa* e *dimensão intrínseca*. A primeira idéia corresponde à dimensão do espaço de endereçamento, ou seja, o número de atributos do conjunto de dados. Porém, esse conjunto de dados pode estar representando um objeto que possui uma dimensão menor que a do espaço em que está imerso. Assim, a dimensão intrínseca é a dimensão espacial do objeto representado pelo conjunto de dados. Conceitualmente, se um conjunto de dados possui todas as suas variáveis (atributos) independentes umas das outras, então sua dimensão intrínseca será igual a sua dimensão imersa. Porém, toda vez que existir uma correlação entre duas ou mais variáveis, a dimensão intrínseca do conjunto de dados é reduzida de acordo. Usualmente, correlações entre os atributos ou a própria existência dessas correlações não é conhecida. Por meio da dimensão intrínseca do conjunto de dados é possível decidir quantos atributos são necessários para caracterizá-lo. Diferentes tipos de correlação podem reduzir a dimensão intrínseca em diferentes proporções, até mesmo em proporções fracionárias. Desse modo, pode-se utilizar o conceito de dimensão fractal como sendo a dimensão intrínseca do conjunto de dados (Traina et al., 2000).

Existem diversas medidas para a DF. Para fractais exatamente auto-similares, *i.e.* que podem ser caracterizados por meio de regras de construção bem definidas, a dimensão fractal é dada por: $D = log(R)/log(\frac{1}{e})$ onde R representa a quantidade de réplicas e $\frac{1}{e}$ em que escala as réplicas são geradas a cada iteração. Para o exemplo do triângulo de Sierpinsky, a DF seria $D = log(3)/log(2) = 1,58496$, pois são geradas três réplicas em escala 1:$\frac{1}{2}$ a cada iteração — Figura 2.

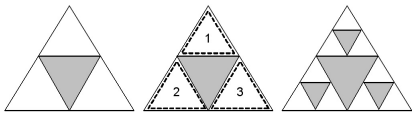

Figura 2: Construção do Triângulo de Sierpinsky

Para fractais estatisticamente auto-similares, como conjuntos de dados reais, uma das maneiras para a definição da DF é representada pela Dimensão Fractal de Correlação D_2, que pode ser calculada pelo método *Box Count Plot* (Faloutsos and Kamel, 1994). A idéia consiste, primeiramente, na construção de um reticulado sobre o conjunto de dados de células de lado r. Então, conta-se o número de pontos dentro da i-ésima célula de tamanho r, denominado $C_{r,i}$. A Dimensão Fractal de Correlação D_2 é definida por: $D_2 = \frac{\partial log(\sum_i {C_{r,i}}^2)}{\partial log(r)}, r \in [r_{min}, r_{max}]$.

Em teoria, fractais exatamente auto-similares são infinitos. Na prática, conjuntos de dados reais, os quais possuem um número finito de pontos, são considerados fractais estatisticamente auto-similares para um determinado intervalo de escalas $r \in [r_{min}, r_{max}]$ se obedecem um regra de construção bem definida nesse intervalo. Desse modo, a dimensão intrínseca de um determinado conjunto de dados pode ser medida como o coeficiente angular da reta que

melhor se ajusta ao trecho linear do gráfico em escala logarítmica de $\sum_i {C_{r,i}}^2$ por r (Traina et al., 2000). Neste trabalho, o termo Dimensão Fractal de Correlação será simplesmente denominado de dimensão fractal.

4. Algoritmo Proposto

O algoritmo para a seleção de atributos proposto neste trabalho, denominado *Fractal Dimension-Based Filter* — FDimBF —, pertence à abordagem filtro e segue o modelo proposto por Yu and Liu (2004), ilustrado na Figura 3, o qual realiza a seleção de atributos em duas etapas: primeiramente é efetuada a *análise de relevância* para determinar o subconjunto de atributos relevantes em relação à classe, removendo os atributos irrelevantes; na segunda etapa, por meio da *análise de redundância*, são determinados e removidos os atributos redundantes a partir do subconjunto que contém apenas os atributos relevantes, produzindo o subconjunto final de atributos selecionados.

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

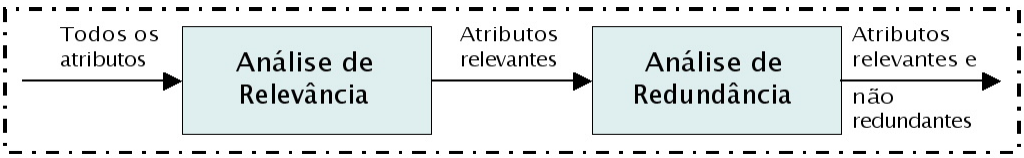

Figura 3: Modelo para seleção de atributos (Yu and Liu, 2004)

No algoritmo FDimBF, propomos o uso da Dimensão Fractal como medida para tratar a redundância de atributos. Para realizar a análise de relevância, propomos o uso de duas medidas: uma baseada em ganho de informação, algoritmo FDimBF(1), e outra baseada em distância, algoritmo FDimBF(2). Especificamente, para realizar a análise de relevância em relação ao atributo classe usando a medida de ganho de informação, utilizamos o algoritmo $\mathcal{C}4.5$ (Quinlan, 1993) e os atributos são classificados de acordo com o número de vezes que aparecem nas regras induzidas. Para medir a relevância dos atributos em relação à classe usando uma medida de distância, utilizamos o algoritmo ReliefF (Robnik-Sikonja and Kononenko, 2003) para ordenar os atributos. Esse algoritmo procura pelos exemplos mais próximos da mesma classe e de classes diferentes, e atribui pesos aos atributos de acordo com quão bem eles diferenciam esses exemplos. Esse processo é repetido m vezes. Em geral, m é definido em função do número de exemplos presentes no conjunto de dados.

Como mencionado anteriormente, para tratar a análise de redundância, neste trabalho propomos a utilização da dimensão fractal. Para medir a DF dos conjuntos de dados, foi utilizada a ferramenta *Measure Distance Exponent* — MDE (Traina et al., 2003). Atributos redundantes, considerando a dimensão fractal, podem ser definidos como aqueles que quando excluídos do conjunto de dados não causam uma modificação no valor da DF recalculada. Essa é a idéia do MDE, a qual consiste na medição do valor da DF, D, a partir do conjunto de dados original e do valor da Dimensão Fractal Parcial, pD, ignorando um atributo por vez. Em outras palavras, a pD é calculada tomando-se em consideração todos os atributos exceto o i-ésimo atributo sob observação. O processo continua selecionando o atributo que permite a diferença mínima entre D e pD. Se a diferença é menor que um limiar mínimo, o qual determina quão preciso o conjunto de dados, descrito por apenas os atributos selecionados, precisa ser para preservar as características do conjunto de dados original, esse atributo pode ser considerado como de contribuição pequena para a caracterização do conjunto de dados original. Esse processo continua, considerando o restante dos atributos e fazendo com que $D = pD$ aplicando o procedimento descrito, até que não existam mais atributos a serem removidos. Ao final do processo, os atributos estarão inversamente ordenados de acordo com sua contribuição, em termos de redundância, para a medição da DF do conjunto de dados (Sousa et al., 2002).

É interessante ressaltar que muitos dos algoritmos de SA tratam, internamente, apenas atributos nominais. Assim, se o conjunto de dados contém atributos numéricos, eles são discretizados pelo algoritmo antes de efetivamente realizar a SA. Esse é o caso dos algoritmos utilizados neste trabalho. Por outro lado, o algoritmo por nós proposto trata efetivamente atributos numéricos, *i.e.* sem a necessidade que eles sejam discretizados, mas atributos nominais são tratados somente durante a análise de relevância — Figura 3 — pois a DF, utilizada para tratar a redundância de atributos, exige que os mesmos sejam numéricos.

5. Descrição dos Conjuntos de Dados

Os conjuntos de dados utilizados para a realização dos experimentos, foram selecionados a partir de uma extensa pesquisa bibliográfica de trabalhos publicados na área de SA, os quais são freqüentemente referenciados pela comunidade. Nesses trabalhos são utilizados três tipos de conjuntos de dados: (1) reais, os quais são extraídos diretamente de bases de dados, por exemplo, de empresas ou hospitais; (2) naturais, obtidos de repositório de dados e (3) artificiais, os quais são gerados computacionalmente a partir da função verdadeira $f(\mathbf{x})$ a ser aprendida. A partir dessa pesquisa bibliográfica, foram selecionados 21 trabalhos que utilizam um total de 99 conjuntos de dados diferentes. Esses conjuntos de dados foram ordenados considerando o número de trabalhos nos quais foram utilizados. Após, foram considerados para seleção posterior somente os conjuntos de dados referenciados em pelo menos dois trabalhos. No final desse processo foram selecionados 11 conjuntos de dados com atributos numéricos e pouco desbalanceados, com o objetivo de não introduzir interferências associadas ao problema de desbalanceamento de classes (Batista et al., 2004).

Todos os 11 conjuntos de dados selecionados constituem conjuntos de dados naturais obtidos do Repositório da UCI (Blake et al., 1998). A Tabela 1 mostra um resumo das características desses conjuntos de dados organizado do seguinte modo: # Exemplos — número de exemplos do conjunto de dados; # Atributos (num.,nom.) — número total de atributos juntamente com o número de atributos numéricos (num.) e nominais (nom.); Erro da CM — erro cometido no caso de novos exemplos serem classificados como sendo pertencentes à classe majoritária e ? que indica a existência ou não de valores desconhecidos.

Conjunto de Dados	# Exemplos (num.,nom.)	# Atributos	Erro da CM	?
Breast Cancer	699	9 (9,0)	34,48% sobre 2	Sim
Bupa	345	6 (6,0)	42,03% sobre 2	Não
German	1000	24 (24,0)	30,00% sobre 1	Não
Hungarian	294	13 (13,0)	36,05% sobre 0	Sim
Ionosphere	351	34 (34,0)	35,90% sobre 0	Não
Pima	769	8 (8,0)	34,98% sobre 0	Não
Satimage	4435	36 (36,0)	75,80% sobre 1	Não
Segment	2310	19 (19,0)	85,70% sobre qualquer atributo	Não
Sonar	208	60 (60,0)	46,60% sobre 1	Não
Vehicle	846	18 (18,0)	74,20% sobre 3	Não
Waveform	5000	21 (21,0)	66,10% sobre 2	Não

Tabela 1: Resumo dos conjuntos de dados

6. Configuração dos Experimentos

Os experimentos realizados foram organizados em quatro etapas, as quais são ilustradas na Figura 4.

Etapa 1: nessa etapa foram realizadas a limpeza e a preparação dos dados. A tarefa de limpeza dos dados consistiu na remoção de valores desconhecidos da seguinte maneira: para valores desconhecidos concentrados em alguns poucos exemplos, esses exemplos foram removidos, enquanto que para valores desconhecidos concentrados em um atributo, a coluna correspondente foi removida do conjunto de dados. A principal razão para a remoção de valores desconhecidos

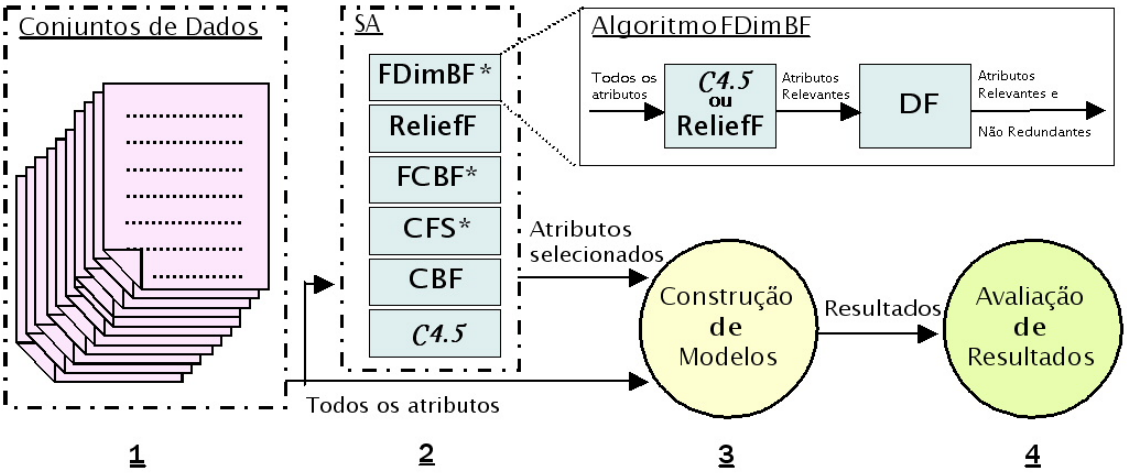

Figura 4: Configuração dos experimentos

do conjunto de dados é que alguns dos algoritmos utilizados nesses experimentos tratam valores faltantes de modo especial (Batista and Monard, 2003a), enquanto outros algoritmos não tratam esse tipo de informação. Assim, com o intuito de não introduzir interferências associadas ao uso de um ou outro método para tratar esse problema, foi decidida a remoção de valores desconhecidos do conjunto de dados. Ao final dessa etapa, os dados foram transformados para a sintaxe requerida por cada um dos algoritmos e ferramentas utilizados neste trabalho.

Etapa 2: nessa etapa foi realizada a seleção de atributos utilizando quatro algoritmos para SA, o algoritmo $\mathcal{C}4.5$ e o algoritmo FDimBF por nós proposto. Três desses algoritmos realizam a seleção de atributos por meio da avaliação individual de atributos — ReliefF (Kira and Rendell, 1992), FCBF (*Fast Correlation-Based Filter*) (Yu and Liu, 2004) e $\mathcal{C}4.5$ (Quinlan, 1993) — e outros dois selecionam atributos avaliando subconjuntos de atributos — CFS (*Correlation-Based Feature Selection*) (Hall, 2000) e CBF (*Consistency-Based Filter* — CBF) (Liu and Setiono, 1996). O algoritmo ReliefF busca por exemplos vizinhos de diferentes classes e atribui pesos aos atributos de acordo com quão bem eles diferenciam esses exemplos como mencionado anteriormente. O algoritmo FCBF realiza a seleção de atributos em duas etapas: primeiramente realiza a análise de relevância para determinar o subconjunto de atributos relevantes em relação à classe, removendo os atributos irrelevantes; na segunda etapa, por meio da análise de redundância, determina e remove os atributos redundantes a partir do subconjunto que contém apenas os atributos relevantes, produzindo o subconjunto final de atributos selecionados. Como citado, o algoritmo $\mathcal{C}4.5$ foi utilizado para induzir regras de decisão. Os atributos selecionados são aqueles que participam do modelo induzido, sendo que a ordem de importância desses atributos é dada pelo número de vezes que aparecem no conjunto de regras induzidas. O algoritmo CFS também é composto, basicamente, por duas etapas: (1) avaliação da correlação entre os atributos e da correlação entre atributos e classe e (2) busca por subconjuntos de atributos e avaliação desses subconjuntos. Já o algoritmo CBF avalia os subconjuntos de atributos de acordo com sua consistência em relação à classe buscando por combinações de atributos cujos valores particionem os dados em subconjuntos com alguma classe majoritária. Todos esses algoritmos, executados considerando seus parâmetros configurados com os valores padrões, a exceção do algoritmo FDimBF proposto neste trabalho e do algoritmo $\mathcal{C}4.5$, estão implementados na ferramenta Weka (Witten and Frank, 2000). Deve ser observado que os algoritmos marcados com $*$ na Figura 4 são aqueles que tratam tanto o problema da relevância de atributos, em relação ao atributo classe, quanto o problema da redundância de atributos.

Etapa 3: nessa etapa foram induzidos os modelos (classificadores) usando todos os atributos remanescentes da Etapa 1 e apenas os atributos selecionados na etapa anterior. Esses modelos foram construídos utilizando o algoritmo $\mathcal{C}4.5$ (Quinlan, 1993).

Etapa 4: nessa última etapa, os resultados foram avaliados por meio da estimativa da média do erro de cada um dos

modelos construídos usando validação cruzada com 10 partições (*10 fold cross-validation*)[1]. Esse modo de avaliação foi escolhido pois, para conjuntos de dados naturais ou reais, o conhecimento prévio sobre que atributos são importantes, em geral, não está disponível. Desse modo, a precisão preditiva é comumente utilizada como uma medida indireta para avaliar a qualidade dos atributos selecionados.

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

7. Resultados e Discussão

Para cada conjunto de dados, foi realizada a seleção de atributos usando as duas versões do algoritmo proposto neste trabalho, *i.e.* FDimBF(1) e FDimBF(2), e os algoritmos $\mathcal{C}4.5$, ReliefF, CFS, CBF e FCBF, totalizando 77 experimentos[2]. Como mencionado anteriormente, foram gerados modelos considerando os atributos selecionados pelos algoritmos citados e também considerando os conjuntos de dados descritos pelos conjuntos originais de atributos (sem SA), totalizando 88 modelos construídos. Esses resultados foram avaliados quanto à relação entre a quantidade de atributos selecionados e o erro dos modelos construídos. Essa relação é representada graficamente com o objetivo de auxiliar na avaliação da performance dos algoritmos considerando ambas as medidas, como mostrado na Figura 5a, onde o eixo X representa a percentagem de atributos selecionados em relação ao total de atributos (apresentado entre parênteses) e o eixo Y representa a média do erro, obtido usando validação cruzada com 10 partições.

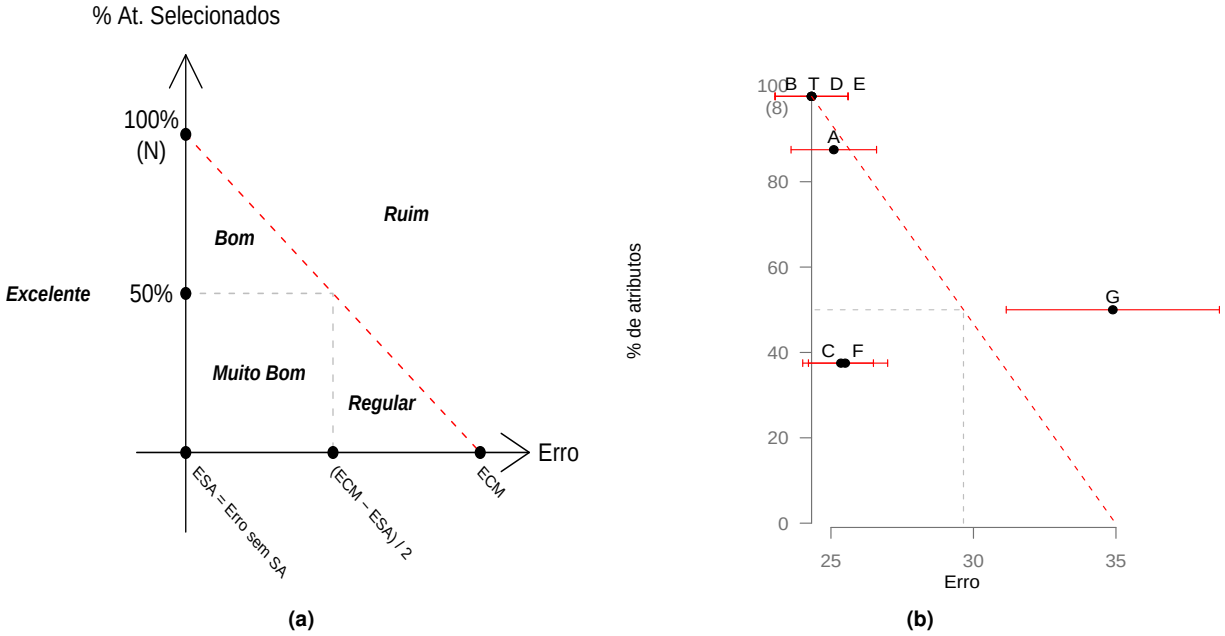

Figura 5: Relação entre percentagem de atributos selecionados, média do erro e erro padrão dos modelos construídos: (a) Modelo geral e (b) Conjunto de dados Pima

[1]Para auxiliar na tarefa de construção dos modelos e avaliação desses modelos por meio de validação cruzada com 10 partições, foi utilizado o ambiente para gerenciamento de experimentos SNIFFER, que faz parte do projeto DISCOVER (Batista and Monard, 2003b).

[2]Os resultados numéricos de todos esses experimentos encontram-se em http://www.foz.unioeste.br/labi/documentos/RT1.pdf.

Nesse gráfico, para cada conjunto de dados, os algoritmos de SA são classificados quanto ao seu posicionamento em relação à percentagem de atributos selecionados e a média do erro e o erro padrão do modelo construído considerando os atributos selecionados por esses algoritmos, dentro de cinco regiões definidas. Primeiramente, duas grandes áreas são delimitadas pela reta que liga o ponto 100% (número total de atributos do conjunto de dados) no eixo X ao ponto ECM no eixo Y, sendo ECM igual ao Erro da Classe Majoritária caso seja menor que 50%, ou igual a 50% caso contrário. Nesse modelo de avaliação, considerou-se que essa reta representa uma proporção mínima entre o que se espera em termos da relação entre percentagem de atributos selecionados e média do erro do modelo construído considerando os atributos selecionados. Assim, qualquer modelo construído com os atributos selecionados por um algoritmo de SA que esteja localizado na região acima dessa reta pode ser considerado de performance Ruim (▼). Abaixo dessa reta e delimitadas pelos eixos X e Y, outras três regiões foram definidas:

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

Uma quarta região, denominada Excelente (▲▲▲), foi definida como sendo a área à esquerda do eixo X. Assim, qualquer algoritmo que permita a seleção de subconjuntos de atributos que melhorem a precisão do modelo construído é considerado de performance excelente. Nos casos em que o conjunto de atributos selecionados foi igual ao conjunto original de atributos do conjunto de dados, o algoritmo foi classificado como Todos os Atributos Selecionados (—).

Na Figura 5b é apresentado um exemplo desse gráfico para o conjunto de dados Pima. Nessa figura é possível identificar a média do erro e o erro padrão considerando o conjunto original de atributos, denominado T, e as posições dos algoritmos — A: $\mathcal{C}4.5$, B: ReliefF, C: CFS, D: FCBF, E: CBF, F: FDimBF(1) e G: FDimBF(2) — dentro das cinco regiões descritas anteriormente. Para esse conjunto de dados, o modelo construído utilizando o subconjunto de atributos selecionado por $\mathcal{C}4.5$ foi considerado bom. Já os algoritmos ReliefF, FCBF e CBF selecionaram todos os atributos do conjunto de dados como sendo importantes. O modelo construído utilizando o subconjunto de atributos selecionado por FDimBF(2) foi considerado ruim, pois encontra-se na região acima da reta definida pelos pontos 100% de atributos selecionados e ECM. Já para a seleção de atributos utilizando os algoritmos CFS e FDimBF(1), os modelos construídos foram considerados muito bons.

A Tabela 2 mostra um resumo da classificação dos algoritmos de SA para cada conjunto de dados quanto ao posicionamento dentro das regiões definidas — Figura 5a. Para cada conjunto de dados é ainda apresentada, na última coluna — CRes —, uma classificação do resultado da aplicação dos algoritmos de SA indicada por $\uparrow$ (número de classificações Excelente, Muito Bom e Bom maior ou igual a cinco), $\downarrow$ (número de classificações Todos os Atributos Selecionados representa em torno de 50% dos casos) e $\sim$ (maioria das classificações Regular e Ruim). Nas últimas linhas dessa tabela é mostrado um resumo da quantidade de vezes em que o respectivo algoritmo foi classificado como tendo apresentado desempenho Excelente, Muito Bom, Bom, Regular, Ruim e Todos os Atributos Selecionados.

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

Considerando cada algoritmo de SA em relação aos tipos de classificação, os algoritmos CFS e CBF foram os que obtiveram o maior número de classificações excelentes, cada um deles tendo obtido quatro. Quanto às classificações muito boas, FDimBF(1) e FDimBF(2) obtiveram sete e seis, respectivamente. Classificações boas e regulares ocorreram de um modo uniforme entre todos os algoritmos. O algoritmo ReliefF juntamente com as duas versões de FDimBF foram os únicos a apresentar classificações ruins. Ressalta-se que as duas versões do algoritmo FDimBF foram os algoritmos que obtiveram o maior número, nove, de classificações excelente e muito bom, seguidas

Algoritmo	$\mathcal{C}4.5$	ReliefF	CFS	FCBF	CBF	FDimBF(1)	FDimBF(2)	CRes
Breast Cancer	▲▲▲	—	—	—	▲▲▲	▲▲▲	▲▲▲	$\sim$
Bupa	—	▼	◇	◇	◇	▼	▲	$\downarrow$
German	—	—	▲▲	▲	▲▲▲	▲▲▲	▲▲▲	$\uparrow$
Hungarian	▲▲▲	—	▲▲▲	▲▲▲	▲▲▲	▲▲	▲▲▲	$\uparrow$
Ionosphere	▲▲	▲	▲▲	▲	▲▲	▲▲	▲▲	$\uparrow$
Pima	▲	—	▲▲	—	—	▲▲	▼	$\sim$
Satimage	—	—	▲▲▲	—	▲▲▲	▲▲	▲▲	$\sim$
Segment	▲	▲	▲▲	▲	▲▲	▲▲	▲▲	$\uparrow$
Sonar	▲▲	—	▲▲▲	▲▲	▲▲	◇	▲▲	$\uparrow$
Vehicle	—	—	▲	—	—	▲▲	▲▲	$\sim$
Waveform	—	—	▲▲▲	▲▲▲	▲	▲▲	▲▲	$\uparrow$
Excelente (▲▲▲)	2	0	4	2	4	2	3	
Muito (▲▲)	2	0	4	1	3	7	6	
Bom (▲)	2	2	1	3	1	0	1	
Regular (◇)	0	0	1	1	1	1	0	
Ruim (▼)	0	1	0	0	0	1	1	
Todos os Atributos Selecionados (—)	5	8	1	4	2	0	0	

Tabela 2: Classificação dos algoritmos em relação a percentagem de atributos selecionados $\times$ erro do modelo construído

por CFS e CBF, cada um com oito e sete classificações desses tipos, respectivamente. É interessante observar que o algoritmo ReliefF foi o que apresentou maior número, oito, de seleções de subconjuntos iguais aos conjuntos originais de atributos (não houve redução do número de atributos selecionados) e que os algoritmos FDimBF(1) e FDimBF(2) foram os únicos a promover redução do número de atributos selecionados para todos os conjuntos de dados.

Do total de 77 classificações (11 conjuntos de dados $\times$ sete algoritmos de SA), 17 foram excelentes, 23 muito boas, 10 boas, quatro regulares, três ruins e 20 selecionaram todos os atributos do conjunto original de atributos. É possível observar que $64,94\%$ das classificações foram excelentes, muito boas ou boas, $25,97\%$ dos subconjuntos de atributos selecionados foram iguais aos conjuntos originais de atributos e apenas $9,09\%$ foram regulares ou ruins, tendo portanto a maioria dos algoritmos de SA contribuído, utilizando os subconjuntos de atributos selecionados, para a melhoria, quer em relação à redução do número de atributos quer em relação à precisão dos modelos construídos no modelo de classificação proposto — Figura 5a.

8. Considerações Finais

Neste trabalho foi apresentada a proposta de um algoritmo, FDimBF, para a seleção de atributos importantes, bem como uma série de experimentos, nos quais a abordagem proposta é comparada a alguns algoritmos freqüentemente citados na literatura. FDimBF realiza a SA em duas etapas: seleção de atributos relevantes em relação à classe e remoção de atributos redundantes. A primeira etapa é realizada utilizando tanto uma medida baseada em ganho de informação — FDimBF(1) — quanto uma medida baseada em distância — FDimBF(2). A segunda etapa é realizada utilizando como medida de correlação a dimensão fractal do conjunto de dados. Ao final desse processo, o algoritmo terá selecionado um subconjunto de atributos relevantes e não redundantes. Deve ser observado que a maioria dos algoritmos de SA não trata esses dois problemas, pois considera como atributos importantes somente aqueles que são relevantes em relação à classe.

Os resultados obtidos utilizando 11 conjuntos de dados e outros algoritmos de SA mostram que o algoritmo proposto é comparável a outros algoritmos de SA, selecionando os menores subconjuntos de atributos importantes com performances similares a algoritmos como o CFS (*Correlation-Based Feature Selection*). Assim, consideramos que a DF pode ser também considerada uma boa candidata para realizar seleção de atributos na área de aprendizado de máquina, na qual não é de nosso conhecimento que ela tenha sido utilizada.

Referências

Batista, G. E., R. C. Prati, and M. C. Monard (2004). A study of the behavior of several methods for balancing machine learning data. *SIGKDD Explorations: Special issue on Learning from Imbalanced Datasets 6*(1), 20–29.

Batista, G. E. A. P. A. and M. C. Monard (2003a). An Analysis of Four Missing Data Treatment Methods for Supervised Learning. *Applied Artificial Intelligence 17*(5), 519–533.

Batista, G. E. A. P. A. and M. C. Monard (2003b). Descrição da Arquitetura e do Projeto do Ambiente Computacional DISCOVER LEARNING ENVIRONMENT — DLE. Technical Report 187, ICMC-USP. `ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/RT_187.pdf`.

Blake, C., E. Keogh, and C. Merz (1998). UCI repository of machine learning databases. `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

Faloutsos, C. and I. Kamel (1994). Beyond uniformity and independence: Analysis of r-trees using the concept of fractal dimension. In *Proc.of the 13th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, Minneapolis, MN, pp. 4–13.

Hall, M. A. (2000). Correlation-based feature selection for discrete and numeric class machine learning. In *Proc. of the 17th Int. Conf. on Machine Learning*, San Francisco, CA, pp. 359–366. Morgan Kaufmann.

Kira, K. and L. Rendell (1992). A pratical approach to feature selection. In *Proc. of the 9th Int. Conf. on Machine Learning*, Aberdeen, Scotland, pp. 249–256. Morgan Kaufmann.

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

Koller, D. and M. Sahami (1996). Toward optimal feature selection. In *Proc. of the 13th Int. Conf. on Machine Learning*, Bari, Italy, pp. 284–292.

Lee, H. D. and M. C. Monard (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.

Liu, H. and H. Motoda (1998). *Feature Selection for Knowledge and Data Mining*. Massachusetts: Kluwer Academic Publishers.

Liu, H. and R. Setiono (1996). A probabilistic approach to feature selection – a filter solution. In *Proc.of the 13th Int. Conf. on Machine Learning*, Bari, Italy, pp. 319–327.

Mandelbrot, B. B. (1985). *The Fractal Geometry of Nature: Updated and Augmented*. New York: W. H. Freeman and Company.

Press, W. H., S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery (1992). *Numerical Recipes in C: The Art of Scientific Computing*. New York: Cambridge University Press.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann. California.

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

Sousa, E. P. M., C. Traina, A. J. M. Traina, and C. Faloutsos (2002). How to use fractal dimension to find correlations between attributes. In *Workshop Notes of KDD 2002 Workshop on Fractals and Self-similarity in Data Mining: Issues and Approaches*, Edmonton, Canada, pp. 26–30.

Traina, C., A. J. M. Traina, and C. Faloutsos (2003). MDE – measure distance exponent manual. (Internal Document).

Traina, C., A. J. M. Traina, L. Wu, and C. Faloutsos (2000). Fast feature selection using fractal dimension. In *Proc. of the 15th Brazilian Data Base Symposium*, João Pessoa, Brasil, pp. 158–171.

Witten, I. H. and E. Frank (2000). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. California: Morgan Kaufmann.

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