RODRIGO CLEMENTE THOM DE SOUZA

UMA METODOLOGIA PARA CLASSIFICAÇÃO DE DADOS NOMINAIS BASEADA NO PROCESSO *KDD*: ÊNFASE AOS ALGORITMOS CULTURAIS, ESTIMAÇÃO DE DISTRIBUIÇÃO E ANÁLISE DE CORRESPONDÊNCIA MÚLTIPLA

Tese apresentada como requisito parcial à obtenção do grau de Doutor em Ciências no Programa de Pós-Graduação em Métodos Numéricos em Engenharia, Setor de Tecnologia e Setor de Ciências Exatas, da Universidade Federal do Paraná.

Orientadora: Profa. Dra. Maria Teresinha Arns Steiner
Co-orientador: Prof. Dr. Leandro dos Santos Coelho

CURITIBA
2013

S729m

Souza, Rodrigo Clemente Thom de

Uma metodologia para classificação de dados nominais baseada no processo KDD : ênfase aos algoritmos culturais, estimação de distribuição e análise de correspondência múltipla / Rodrigo Clemente Thom de Souza. – Curitiba, 2013.

159f. : il. [algumas color.] ; 30 cm.

Tese (doutorado) - Universidade Federal do Paraná, Setor de Ciências Exatas, Programa de Pós-graduação em Métodos Numéricos em Engenharia, 2013.

Orientador: Maria Teresinha Arns Steiner -- Co-orientador: Leandro dos Santos Coelho.

Bibliografia: p. 135-150.

1.Reconhecimento de Padrões. 2.Inteligência Artificial. 3. Algoritimos. I. Universidade Federal do Paraná. II. Steiner, Maria Teresinha Arns. III. Coelho, Leandro dos Santos. IV. Título.

CDD: 006.31

TERMO DE APROVAÇÃO

RODRIGO CLEMENTE THOM DE SOUZA

UMA METDOLOGIA PARA A CLASSIFICAÇÃO DE DADOS NOMINAIS BASEADA NO PROCESSO KDD: ÊNFASE AOS ALGORITMOS CULTURAIS, ESTIMAÇÃO DE DISTRIBUIÇÃO E ANÁLISE DE CORRESPONDÊNCIA MÚLTIPLA

Tese aprovada como requisito parcial para obtenção do grau de Doutor no Programa de Pós-Graduação em Métodos Numéricos em Engenharia, Setores de Tecnologia e de Ciências Exatas, Universidade Federal do Paraná, pela seguinte banca examinadora:

Profa. Dra. Maria Teresinha Arns Steiner
Programa de Pós-Graduação em Métodos Numéricos em Engenharia, UFPR

Prof. Dr. Leandro dos Santos Coelho
Programa de Pós-Graduação em Engenharia de Produção e Sistemas, PUCPR

Prof. Dr. Júlio Cesar Nievola
Programa de Pós-Graduação em Informática Aplicada, PUCPR

Prof. Dr. Anderson Roges Teixeira Góes
Departamento de Expressão Gráfica, UFPR

AGRADECIMENTOS

Agradeço infinitamente a Deus pela luz colocada em nossos caminhos a cada dia!

À Virgem Maria por sempre interceder por nós, sobretudo nos momentos de maior aflição!

Ao imenso coração de meus pais! Obrigado Senhor Deus por ter escolhido este lar de amor como berço para o meu nascimento!

À minha doce esposa que está incondicionalmente ao meu lado, sempre! Obrigado Senhor Deus por ter escolhido como minha companheira justamente a pessoa mais maravilhosa que já conheci!

À minha filha que já é, sem dúvida, o ser vivo que mais amo neste mundo!

Ao meu irmão, pelo exemplo que ele sempre foi e sempre será para mim!

Aos pais e à irmã de minha esposa, que são minha segunda família! Em especial à Beth, a quem atribuo um papel muito importante no meu doutorado.

À minha orientadora e ao meu co-orientador pelos incontáveis ensinamentos, pela confiança e pelo empenho para que, juntos, alcançássemos o sucesso!

Aos demais membros da banca e a todos os professores e amigos que contribuíram cada um à sua maneira com minha formação!

Lutarei sempre para retribuir a todos vocês!

Muitas vezes as pessoas são egocêntricas, ilógicas e insensatas.
Perdoe-as assim mesmo!

Se você é gentil, podem acusá-lo de interesseiro.
Seja gentil assim mesmo!

Se você é um vencedor terá alguns falsos amigos e alguns inimigos verdadeiros.
Vença assim mesmo!

Se você é bondoso e franco poderão enganá-lo.
Seja bondoso e franco assim mesmo!

O que você levou anos para construir, alguém pode destruir de uma hora para a outra.
Construa assim mesmo!

Se você tem paz e é feliz, poderão sentir inveja.
Seja feliz assim mesmo!

O bem que você faz hoje, poderão esquecê-lo amanhã.
Faça o bem assim mesmo!

Dê ao mundo o melhor de você, mas isso pode nunca ser o bastante.
Dê o melhor de você assim mesmo!

Veja você que, no final das contas é entre você e Deus.
Nunca foi entre você e os outros!

Madre Teresa de Calcutá

RESUMO

A classificação de padrões é um problema de aprendizado supervisionado do campo da ciência conhecido como Reconhecimento de Padrões (RP), através do qual se deseja discriminar instâncias de dados em diferentes classes. A solução para este problema é obtida por meio de algoritmos (classificadores) que buscam por padrões de relacionamento entre classes em casos conhecidos (treinamento), usando tais relações para classificar casos desconhecidos (teste). O desempenho em termos de acurácia preditiva dos algoritmos que se propõem a realizar tal tarefa depende muito da qualidade e dos tipos de dados contidos nas bases. Visando melhorar a qualidade dos dados e dar tratamento adequado aos tipos de dados utilizados, o presente trabalho faz uso do processo de Descoberta de Conhecimento em Bases de Dados (*Knowledge Discovery in Databases*; *KDD*), no qual a classificação é uma das tarefas da etapa conhecida como Mineração de Dados (*Data Mining*; *DM*). As etapas aqui aplicadas antes da classificação são a seleção de atributos *wrapper* e um processo de transformação de atributos baseado em Análise Geométrica de Dados (*Geometric Data Analysis*; *GDA*). Para a seleção de atributos é proposta uma nova técnica baseada em Algoritmo de Estimação de Distribuição (*Estimation of Distribution Algorithm*; *EDA*) e em Algoritmos Culturais (AC) batizada de *Belief-Based Incremental Learning* (*BBIL*). Para a transformação de atributos é aqui proposta a utilização de uma alternativa à clássica Análise de Componentes Principais (*Principal Component Analysis*; *PCA*) para lidar especificamente com dados nominais: a Análise de Correspondência Múltipla (*Multiple Correspondence Analysis*; *MCA*). Na etapa de *DM*, de fato, faz-se a aplicação de dois tradicionais classificadores da área de RP, Naïve Bayes e Função Discriminante Linear de Fisher (*Linear Discriminant Analysis*; *LDA*). Apoiado em argumentos teóricos e em testes empíricos realizados com nove diferentes conjuntos de dados nominais, o presente trabalho objetiva avaliar a capacidade do *MCA* e do *BBIL* em melhorar o desempenho de classificadores em termos de acurácia preditiva média. Com o objetivo de se beneficiar simultaneamente das vantagens de ambos os tratamentos de dados são avaliadas duas combinações entre estas técnicas. A primeira trata-se da transformação *GDA* sobre os atributos previamente selecionados e, a segunda, a seleção de *factor scores* do *MCA* utilizando o *BBIL* (metodologia proposta). Os resultados dos experimentos confirmam a melhoria no desempenho de classificação proporcionada pelos tratamentos realizados e atestam a superioridade da metodologia proposta na maioria das situações analisadas.

Palavras-chave: classificação de padrões, dados nominais, seleção de atributos, algoritmo de estimação de distribuição, algoritmos culturais, análise geométrica de dados, análise de correspondência múltipla.

ABSTRACT

Pattern classification is a supervised learning problem in the field of science known as Pattern Recognition, through which to discriminate data instances in different classes. The solution to this problem is obtained through algorithms (classifiers) that search for patterns of relationships between classes in known cases (training), using such relationships to classify unknown cases (test). The performance in terms of predictive accuracy of the algorithms they propose to undertake such a task depends on the quality and types of data from databases. To improve the quality of data and provide proper treatment to the types of data used, the present work makes use of the process of Knowledge Discovery in Databases (KDD), in which classification is one of the tasks of step known as Data Mining (DM). The steps here applied before classification are feature selection wrapper and a process of transformation of attributes based on Geometric Data Analysis (GDA). For feature selection is proposed a new technique based on Estimation of Distribution Algorithm (EDA) and Cultural Algorithms named Belief-Based Incremental Learning (BBIL). For the transformation of attributes is proposed here the use of an alternative to the classical Principal Component Analysis (PCA) to deal specifically with nominal data: the Multiple Correspondence Analysis (MCA). In the stage of DM, in fact, it is the application of two traditional classifiers area Pattern Recognition, Naïve Bayes (NB) and Fisher Linear Discriminant Function (Linear Discriminant Analysis, LDA). Supported by theoretical arguments and empirical tests conducted with nine different nominal datasets, this study aims to evaluate the ability of the MCA and BBIL in improving the performance of classifiers in terms of predictive accuracy. In order to benefit simultaneously from the advantages of both data treatments are evaluated two combinations of these techniques. The first one refers to the transformation GDA on the attributes previously selected, and the second one, the selection of the MCA factor scores using the BBIL (proposed methodology). The experimental results confirm the improvement in classification performance provided by the treatments performed and attest to the superiority of the proposed methodology in most analysed situations.

Keywords: pattern classification, nominal data, attribute selection, estimation of distribution algorithm, cultural algorithms, geometric data analysis, multiple correspondence analysis.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

ABNT - Associação Brasileira de Normas Técnicas

AC - Algoritmos Culturais

ACO - Ant Colony Optimization

AE - Algoritmos Evolucionários

AG - Algoritmos Genéticos

ANEEL - Agência Nacional de Energia Elétrica

ANOVA - Analysis of Variance

BBIL - Belief-Based Incremental Learning

BDE - Binary Differential Evolution

BOA - Bayesian Optimization Algorithm

CA - Correspondence Analysis

C&R - Classification and Regression

CGA - Compact Genetic Algorithm

CLP - Controladores Lógicos Programáveis

DAG - Directed Acyclic Graph

DFA - Discriminant Function Analysis

DARPA - Defense Advanced Research Projects Agency

DNA - DeoxyriboNucleic Acid

DPSO - Discrete Particle Swarm Optimization

ED - Evolução Diferencial

EDA - Estimation of Distribution Algorithms

EE - Estratégias Evolutivas

FCM - Fuzzy C-Means

GDA - Geometric Data Analysis

GSVD - Generalized Singular Value Decomposition

IC - Inteligência Computacional

IEC - International Electrotechnical Commission

IEEE - Institute of Electrical and Electronics Engineers

KDD - Knowledge Discovery in Databases

LDA - Linear Discriminant Analysis

MAP - Maximum a Posteriori

MATLAB - Matrix Laboratory

MCA - Multiple Correspondence Analysis

MDA - Multiple Discriminant Analysis

NBR - Norma Brasileira

NP - Non-deterministic Polynomial-time

NB - Naïve-Bayes

PBIL - Population-Based Incremental Learning

PC - Principal Component

PG - Programação Genética

PCA - Principal Component Analysis

PLSR - Partial Least Squares Regression

PPGMNE - Programa de Pós-Graduação em Métodos Numéricos em Engenharia

PRODIST - Procedimento de Distribuição

PSO - Particle Swarm Optimization

QEE - Qualidade de Energia Elétrica

RB - Redes Bayesianas

RP - Reconhecimento de Padrões

RBF - Radial Basis Function

RL - Regressão Logística

RNA - Redes Neurais Artificiais

SA - Simulated Annealing

SEP - Sistemas Elétricos de Potência

SIA - Sistemas Imunológicos Artificiais

SN - Sistemas Nebulosos

SSGA - Steady State Genetic Algorithm

SVM - Support Vector Machine

UCI - University of California Irvine

UFPR - Universidade Federal do Paraná

UMDA - Univariate Marginal Distribution Algorithm

VTCD - Variação de Tensão de Curta Duração

LISTA DE ABREVIATURAS

A - Ampère
GB - GigaByte
GHz - GigaHertz
Hz - Hertz
kV - quilovolt
ms - milissegundo
pu - por unidade

SUMÁRIO

1 INTRODUÇÃO

O *KDD*, sigla para *Knowledge Discovery in Databases*, em português, Descoberta de Conhecimento em Bases de Dados, estabelece-se como a área de pesquisa que lida com a descoberta de conhecimento em conjuntos de dados gerados a partir de processos experimentais e observacionais. O *KDD* engloba uma série de etapas desde a seleção de dados até a interpretação do conhecimento descoberto a partir dos mesmos. Uma etapa do *KDD*, a mineração de dados, trata especificamente da extração de padrões geralmente "ocultos" nos dados.

Extrair padrões de um conjunto de dados significa encontrar um modelo que se ajuste e descreva adequadamente estes dados. Deseja-se ainda que o modelo obtido seja válido com algum grau de certeza sobre novos dados e que o conhecimento contido nestes padrões seja útil e compreensível. Uma das tarefas (objetivos) de extração de padrões desempenhada pela mineração de dados é a classificação de padrões.

A classificação realiza a extração de padrões por meio de algoritmos de aprendizagem que buscam uma função que mapeie (classifique) instâncias de dados em uma dentre duas ou mais classes definidas a priori. A aplicação direta de algoritmos de classificação de padrões sem análise e tratamento prévios dos dados, conhecida na literatura como "dragagem de dados", trata-se de uma atividade arriscada, levando inúmeras vezes à extração de padrões inválidos, incompreensíveis ou ausentes de significado (Fayyad *et al.*, 1996).

Por esta razão, o tratamento dos dados antes de serem submetidos à classificação de padrões vem gradativamente ganhando importância na literatura. Isto é notado em trabalhos como os de Steiner (1995), Schneider (2005), Katragadda (2008), Smith e Martinez (2011) e Tsai *et al.* (2013). Trabalhando com um mesmo objetivo final, mas com propósitos diferentes, duas etapas do *KDD* anteriores a Mineração de Dados, denominadas de seleção e transformação de dados, são o tema do presente trabalho.

A seleção de atributos realiza a seleção de um subconjunto de atributos relevantes para a construção de modelos robustos de classificação. Ela também auxilia na obtenção de uma melhor compreensão sobre os dados por

apresentar os atributos mais importantes para a classificação e, em algumas situações, a maneira como eles se relacionam uns com os outros.

Segundo Bianchi *et al*. (2009), uma meta-heurística é um algoritmo projetado para resolver problemas complexos de otimização. A seleção de atributos guiada (e visando) à melhoria do desempenho de classificadores é chamada de *wrapper*. Este trabalho propõe a hibridização de duas meta-heurísticas de Algoritmos Evolucionários (AE), o *Estimation of Distribution Algorithms* (*EDA*; em português, Algoritmos de Estimação de Distribuição) e os Algoritmos Culturais (AC) para a implementação de um *wrapper* para a seleção de atributos.

Para a etapa de transformação de dados, o presente trabalho aplica uma abordagem denominada *Geometric Data Analysis* (*GDA*; em português, Análise Geométrica de Dados). Apesar do nome, o *GDA* não usa apenas conceitos de geometria, mas também de Álgebra Linear e de estatística. Sua componente geométrica refere-se ao uso do conceito de “nuvem de pontos” em um espaço geométrico Euclidiano. Da álgebra linear, a abordagem utiliza conceitos de autovetores e, da estatística, conceitos de variância e covariância (Le Roux e Rouanet, 2010).

A técnica de *GDA* apresentada no presente trabalho, denominada *Multiple Correspondence Analysis* (*MCA*; em português, Análise de Correspondência Múltipla), visa melhorar o desempenho dos classificadores quando aplicados a conjuntos de dados nominais.

1.1 PROBLEMA E HIPÓTESE DE PESQUISA

A maioria da literatura sobre o processo *KDD* possui enfoque na etapa de mineração de dados (Nettleton, 2013; Chen e Sue, 2013; Strohmeier e Piazza, 2013). Alguns chegam a afirmar inclusive ser esta a etapa mais importante de todo o processo. Todavia, as outras etapas parecem ser tão (ou mais) importantes que a mineração de dados para o sucesso de uma aplicação de *KDD* na prática (Fayyad *et al.*, 1996).

Os tratamentos específicos sobre os dados, tais como limpeza e transformação de dados podem melhorar o desempenho de algoritmos de

mineração de dados, em especial de classificadores. O presente trabalho pretende elucidar esta questão visando mostrar que a aplicação de tratamentos prévios aos dados pode trazer benefícios ao resultado do *KDD* como um todo. Para alcançar a resposta a este problema, são analisados os resultados da classificação de padrões aplicada a nove diferentes conjuntos de dados com dois dos principais algoritmos de classificação da literatura: o *Linear Discriminant Analysis* (*LDA*, em português, Análise Discrminante Linear (Zollanvari *et al.*, 2013)) e o *Naïve-Bayes* (*NB* (Catal *et al.*, 2011)). Dos nove conjuntos de dados, três deles referem-se a estudos de casos reais, um da área elétrica e outro da área médica; os demais seis conjuntos são bases de dados *benchmark* da literatura.

Um dos tratamentos de dados aqui abordado é a seleção de atributos guiada (e visando) à melhoria do desempenho de classificadores, o que na literatura é chamado de *wrapper*. Este trabalho propõe a hibridização de duas meta-heurísticas de Algoritmos Evolucionários (AE), o *Estimation of Distribution Algorithm* (*EDA*) e os Algoritmos Culturais (AC) para a implementação de um *wrapper* de seleção de atributos. Para a seleção de atributos, além de ser analisada a melhoria de desempenho de classificadores em termos de acurácia preditiva média, também são avaliados outros critérios (fração de atributos selecionados, *overfitting* e custo computacional). Além disso, analisa-se o desempenho do *MCA*, um método ainda pouco explorado na área de *KDD*, como técnica de transformação específica para dados nominais.

O presente trabalho avalia a combinação entre estes dois diferentes tipos de tratamentos de dados (seleção de atributos e transformação geométrica), primeiramente, aplicando o *GDA* sobre atributos previamente selecionados e, posteriormente, aplicando algoritmos de seleção de atributos para a seleção de *factor scores* do *MCA*, através de uma metodologia inédita, descrita na seção 4.1 e avaliada na seção 5.5.

1.2 JUSTIFICATIVA, DELIMITAÇÃO E CONTRIBUIÇÕES DA TESE

O desempenho em termos de acurácia preditiva de algoritmos de classificação de padrões depende da qualidade e dos tipos de dados contidos

nas bases utilizadas. Segundo Agresti (2002) existem, basicamente, dois[1] tipos de dados nessas bases: contínuos e categóricos.

Uma base de dados contínua é aquela em que todos os elementos de cada atributo nela contidos são números contínuos. Uma base de dados categórica é aquela em que todos os atributos são nominais (não existe uma ordem específica para as categorias) ou ordinais (se representam valores numéricos ou intervalos). Uma base de dados composta tanto por atributos contínuos quanto por atributos categóricos é chamada de base de dados mista.

Para facilitar operações matemático-computacionais em bases de dados nominais, foco do presente trabalho, é comum representar os valores (categorias) de seus atributos por números. Mesmo assim, o simples fato dos valores não estarem ordenados dificulta a aplicação de algoritmos de classificação que consideram distâncias Euclidianas entre os dados. Como os algoritmos de classificação em geral realizam operações numéricas sobre números reais, deve ser evitada a aplicação direta dos mesmos sobre dados nominais.

Apesar de existirem algoritmos específicos para manipulação de dados nominais, como a Regressão Logística (Yule, 1900 *apud* Agresti, 2002), uma alternativa que se apresenta, além da adoção destes algoritmos específicos, é aplicar previamente algum tipo de transformação geométrica sobre os dados para que estes possam ser operados por algoritmos mais "gerais".

Uma abordagem pouco conhecida de Estatística Multivariada que lida com transformações geométricas sobre dados é o *GDA*. O *GDA* representa os conjuntos de dados como objetos geométricos em espaços Euclidianos multidimensionais construídos a partir de tabelas de dados com base em estruturas matemáticas de Álgebra Linear. Apesar da pouca atenção dada ao *GDA* pela comunidade científica de Reconhecimento de Padrões (RP), seu método mais clássico (Le Roux e Rouanet, 2005), o *Principal Component Analysis* (*PCA*, em português, Análise de Componentes Principais), é bastante difundido (e utilizado) no meio acadêmico de RP.

---

[1] Outros tipos de dados são ou podem ser modificados para se tornarem dados contínuos ou categóricos.

O *PCA* é considerado uma das bases fundamentais da maioria dos métodos modernos para tratamento de dados multivariados visando a classificação de padrões. Entretanto, embora seja eficiente quando aplicado a dados contínuos, a matemática baseada em distâncias adotada pelo *PCA* não apresenta a mesma adequação para dados nominais.

Assim, o presente trabalho propõe a aplicação de outro método de *GDA*, mais conhecido na comunidade científica de Análise Estatística de Dados, denominado *MCA*. Desde a criação do *MCA* por Benzécri (1973), poucos trabalhos relacionaram esta técnica à classificação de padrões e ao *KDD*. Uma primeira contribuição do presente trabalho é apresentar o *MCA*, não como ferramenta de análise estatística, mas como método de transformação linear sobre dados nominais antes da aplicação de algoritmos de classificação de padrões.

Com exceção de alguns trabalhos de autores franceses (Saporta e Niang, 2006; Bougeard *et al.*, 2011), país de origem de Benzécri, poucos pesquisadores tem dado a devida atenção ao *MCA* como abordagem de transformação de dados no contexto do *KDD*. Assim, o presente tabalho pretende “lançar luz” sobre a aplicação da transformação de dados por *MCA* para melhorar o desempenho de classificadores e estimular o interesse no tema, uma vez que, sob o ponto de vista do autor, a técnica não tem recebido a merecida atenção para este fim. O trabalho compara o desempenho dos classificadores sem nenhum tipo de tratamento prévio (dados “brutos”), com a transformação por *PCA* e por *MCA*.

O presente trabalho propõe também uma segunda contribuição, um novo algoritmo de seleção de atributos. A seleção de atributos é uma das etapas do pré-processamento de dados, vital para melhorar a acurácia de algoritmos de classificação convencionais. Geralmente utilizada para amenizar o problema de conjuntos de dados com muitos atributos, esta forma de limpeza de dados pode ser também útil quando aplicada a conjuntos de dados menores, como os utilizados no presente trabalho.

Uma das mais conhecidas, simples e eficazes estratégias para seleção de atributos é o *wrapper*. Esta estratégia utiliza-se do próprio algoritmo de classificação de padrões para avaliar cada subconjunto de atributos gerado por

um algoritmo de seleção de atributos. Para cada subconjunto, constrói-se uma nova base de dados para classificação e, o resultado desta nova classificação é usado como função de avaliação (*fitness*) do novo subconjunto de atributos. Assim, o *wrapper* de seleção de atributos pode ser entendido como um algoritmo de otimização.

Uma vez que o *wrapper* é um problema *Non-deterministic Polynomial-time hard* (*NP-hard*), diversas abordagens meta-heurísticas têm sido propostas nos últimos anos para solucioná-lo, com especial destaque para os AE e outros algoritmos baseados em populações[2]. O algoritmo proposto para seleção de atributos no presente trabalho é uma hibridização de duas técnicas de AE: *EDA* e AC.

Diferentemente dos AE clássicos tal como, por exemplo, os Algoritmos Genéticos (AG), o *EDA* não necessita de determinadas operações, tais como cruzamento e mutação (e seus respectivos parâmetros). Sob este aspecto, o *EDA* pode ser considerado mais simples do que os AE clássicos. No *EDA*, os indivíduos são amostras e, a cada iteração do algoritmo (geração), as amostras são ajustadas a um modelo probabilístico de distribuição para determinação das melhores novas amostras.

Os AC, por sua vez, formam um complemento à metáfora adotada pelos AE com base na evolução cultural das populações humanas. AC se baseiam em teorias de sociologia e arqueologia para modelar a evolução cultural das sociedades humanas.

Tanto o *EDA* quanto os AC são técnicas ainda pouco exploradas com a finalidade de seleção de atributos. Assim, o presente trabalho apresenta de forma inédita a aplicação do *EDA* e dos AC de forma conjunta para este fim. O *EDA* em que o trabalho se baseia é o *Population-Based Incremental Learning* (*PBIL*, traduzido livremente para o português como Aprendizado Incremental Baseado em Populações). A variante do *PBIL* desenvolvida, hibridizada com AC e batizada de *Belief-Based Incremental Learning* (*BBIL*, em português,

[2] População neste contexto significa o conjunto de soluções candidatas de cada iteração (geração) de um algoritmo baseado em populações.

Aprendizado Incremental Baseado em Crenças), é comparada com o algoritmo *EDA-PBIL* original e com a mais popular abordagem de AE[3], os AG.

Apoiado em argumentos teóricos e em testes empíricos realizados com nove diferentes bases de dados nominais, o presente trabalho analisa o potencial da transformação geométrica por *MCA* e do seletor de atributos por *BBIL* na melhoria do desempenho de classificadores. O trabalho não propõe o desenvolvimento de novos classificadores, mas sim a melhoria nos resultados de dois bem conhecidos algoritmos da área de RP: *LDA* e *NB*. Além disso, ele também não visa a comparação entre estes dois classificadores.

Os resultados das simulações são tabulados para comparação dos desempenhos dos classificadores sem e com submissão a transformação geométrica, a seleção de atributos e a ambos. Esta combinação entre ambos apresenta-se como uma terceira contribuição do presente trabalho, no sentido de oferecer, simultaneamente, maior qualidade aos dados por meio da seleção de atributos (conjuntamente) mais relevantes proporcionada pelo *BBIL* e um tratamento mais adequado a dados nominais resultantes do *MCA*. As simulações realizadas utilizam a acurácia preditiva (ou acurácia de classificação) como critério de desempenho e o *n-fold cross-validation* como método de validação e estimação de erros dos classificadores.

1.3 OBJETIVOS

O objetivo geral do presente trabalho é apresentar uma metodologia embasada no processo *KDD* para melhoria do desempenho na classificação de dados nominais utilizando *EDA*, AC e *MCA*. Tal metodologia é aplicada a três bases de dados reais, duas da área elétrica e outra da área médica e, também, a seis problemas teste (*benchmarks*) de RP.

Os objetivos específicos são:

---

[3] Apanhados sobre AE incluindo as técnicas abordadas no presente trabalho podem ser obtidos em Boussaïd *et al.* (2013), Das *et al.* (2011) e Hauschild e Pelikan (2011).

- Apresentar os conceitos teóricos e resultados empíricos das técnicas abordadas no presente trabalho envolvendo a transformação geométrica por *MCA* e a seleção de atributos pelo *EDA* de inspiração cultural*;*
- Contribuir com a elucidação da questão acerca de que a aplicação de tratamentos prévios aos dados pode trazer benefícios significativos ao resultado do *KDD* como um todo;
- Apresentar uma nova forma de aplicação sobre dados nominais de algoritmos de classificação projetados para operar exclusivamente no $\Re^n$;
- Difundir o *MCA* junto à comunidade científica de RP como método de transformação de dados para o processo *KDD* ressaltando sua adequação para dados nominais;
- Realizar uma análise comparativa da acurácia de classificadores com transformação prévia por *MCA*, com transformação por *PCA* e sem transformação prévia;
- Propor um novo algoritmo híbrido de AE usando *EDA* e AC;
- Aplicar de forma inédita o novo algoritmo proposto na forma de *wrapper* para seleção de atributos visando a classificação de padrões, medindo acurácia preditiva, *overfitting*, fração de atributos selecionados e custo computacional;
- Avaliar a aplicação combinada da seleção de atributos e da transformação geométrica, primeiramente, aplicando o *GDA* sobre atributos previamente selecionados e, posteriormente, aplicando algoritmos de seleção de atributos para a seleção de *factor scores* do *MCA*, através de uma metodologia inédita;
- Validar a aplicação das técnicas propostas em problemas *benchmarks* com dados conhecidos da comunidade científica; e
- Aplicar as técnicas propostas em estudos de caso reais de diferentes áreas (a elétrica e a médica).

1.4 ESTRUTURA DO TRABALHO

Este primeiro capítulo apresentou a introdução ao tema e à problematização da tese, abordando sua relevância, bem como os objetivos e a estrutura do trabalho.

O segundo capítulo apresenta uma revisão da literatura sobre a aplicação de tratamentos de *KDD* para melhoria do desempenho de classificadores, com trabalhos sobre transformações geométricas em dados nominais e seleção de atributos baseada em populações.

O terceiro capítulo trata da fundamentação teórica a respeito do *KDD* e suas etapas estudadas no presente trabalho em três seções. A primeira seção trata das técnicas de seleção de atributos, a segunda, das técnicas *GDA* e a terceira da técnicas de classificação de padrões.

No quarto capítulo são detalhados a metodologia proposta, as bases de dados dos problemas analisados referentes a dois estudos de caso (um da área elétrica e outro da área médica) e problemas *benchmarks* de RP, bem como os recursos computacionais utilizados para os experimentos.

No quinto capítulo são mostradas as configurações dos algoritmos utilizados, os resultados obtidos em cada experimento, além de análises comparativas entre as técnicas abordadas.

E no sexto e último capítulo são apresentadas as conclusões e limitações do trabalho, bem como sugestões para continuidade desta pesquisa.

2 TRABALHOS RELACIONADOS

Steiner (1995) mostrou a importância da aplicação de tratamentos estatísticos prévios nos dados antes da aplicação de algoritmos de classificação de padrões. No trabalho de Steiner (1995), estes tratamentos visavam, dentre outros objetivos, verificar a importância de se realizar o tratamento dos dados preliminarmente à aplicação de técnicas de RP.

Diversas outras formas de tratamento preliminarmente à classificação são sugeridas na literatura (Berthold *et al.*, 2010). O presente trabalho aborda duas dessas formas: (i) transformações geométricas e (ii) seleção de atributos.

O presente capítulo apresenta uma revisão literária do estado da arte sobre estes tratamentos do *KDD* para melhoria do desempenho de classificadores. Mais especificamente, a seção 2.1 apresenta trabalhos recentes relacionados à aplicação de transformações geométricas sobre dados nominais e a seção 2.2 à seleção de atributos baseada em populações.

2.1 TRANSFORMAÇÃO GEOMÉTRICA SOBRE DADOS NOMINAIS

Uma alternativa à aplicação direta de classificadores específicos para dados nominais, tais como Regressão Logística Multinomial[4], *Cobweb* (Fisher, 1987) e *Barycentric Discriminant Analysis* (Abdi e Williams, 2010a), é a aplicação prévia de transformações sobre os dados para posterior classificação com algoritmos para dados contínuos.

Formas comuns de transformação para dados nominais incluem codificações baseadas em frequência (Uyar *et al.*, 2009), projeção randômica (Ahmad, 2009), *feature bundling* (Kusiak, 2001) e *feature content modification* (Maimon e Rokach, 2005). Menos comuns, no entanto, são trabalhos que o fazem por meio de transformações geométricas com base em técnicas de *GDA*, também chamadas por alguns autores da área de RP de *feature extraction* (Li e Davis, 2011).

[4] Generalização da Regressão Logística para problemas não dicotômicos.

Um trabalho que explora esta alternativa é o de Katragadda (2008). Em seu trabalho, o autor aplica a transformação Gifi (Gifi, 1989) e o *optimal scaling* (bastante semelhante ao *MCA*) sobre atributos nominais de uma base mista, com o objetivo de "normalizar" a base e, consequentemente, melhorar o desempenho de classificadores. Originalmente a transformação Gifi era aplicável somente a dados contínuos, contudo, uma adaptação proposta por Michailidis e de Leeuw (1996) permitiu sua aplicação a dados categóricos. Katragadda (2008) utiliza esta versão modificada.

Como sua base era composta por tipos mistos (atributos contínuos e categóricos), o autor aplicou a transformação Gifi somente aos atributos categóricos, deixando os contínuos intactos. Os dados transformados para o "espaço Gifi" foram então submetidos à transformação por *optimal scaling*, no qual cada variável é decomposta em uma matriz de indicadores e um vetor de pesos.

Após as transformações, os dados resultantes equivaliam aos dados no espaço original, mas com os valores categóricos substituídos por seus pesos e indicadores correspondentes. Os "novos atributos" juntamente com os atributos contínuos que haviam permanecido intactos formaram então uma nova base de dados. Com esta nova base, técnicas multivariadas clássicas foram aplicadas, tais como Regressão Logística (RL) e Análise Discriminante, obtendo-se bons resultados.

Em um trabalho mais recente, Bougeard *et al.* (2011) exploram o *Categorical Multiblock Redundancy Analysis* como alternativa ao *Multiblock Partial Least Squares* em um problema de classificação de dados nominais de epidemiologia veterinária. O foco do trabalho é na exploração da capacidade "multiblocos" (manipulação de múltiplos blocos de dados coletados a partir de um mesmo conjunto de amostras) das técnicas, e não propriamente na classificação de padrões.

O método adotado por este grupo de pesquisadores transforma os atributos nominais em variáveis latentes projetando os atributos originais sobre o subespaço gerado por essas variáveis. O objetivo do trabalho é classificar granjas de suínos em três grupos: fria, temperada e temperada com gases, utilizando-se de 19 atributos de predição nominais relacionados aos sistemas

de aquecimento e ventilação das granjas. Os autores concluem que os resultados obtidos pelo *Categorical Multiblock Redundancy Analysis* mostraram-se relevantes como técnica de transformação para dados nominais.

Saporta e Niang (2006) apresentam a metodologia *Disqual*, um algoritmo baseado em *MCA* para transformação geométrica de dados visando à classificação. A metodologia é dividida em duas partes: (i) aplicação do *MCA* sobre os atributos de predição e (ii) aplicação da *LDA* usando as coordenadas dos *factor scores* (explicadas na seção 3.2.2) obtidas pelo *MCA* como novos atributos de predição.

Os autores compararam o algoritmo com a RL para preditores categóricos, com o *Partial Least Squares Regression* (*PLSR*) e com a *Barycentric Discrimination Analysis* em um estudo de caso de avaliação de clientes de uma seguradora de veículos da Bélgica. Os resultados do algoritmo foram satisfatórios, equivalendo-se ou mostrando-se superior aos métodos com os quais foi comparado.

O presente trabalho também acopla o *MCA* a classificadores, contudo operando na forma de *wrapper* de seleção de atributos. A revisão literária sobre seleção de atributos baseada em populações, abordada no presente trabalho, é apresentada a seguir, na seção 2.2.

2.2 SELEÇÃO DE ATRIBUTOS BASEADA EM POPULAÇÕES

Diversos trabalhos têm sido propostos nos últimos anos para a seleção de atributos baseada em populações. Pedrycz e Ahmad (2012) utilizam abordagens baseadas em populações (AG e *PSO*) para seleção de características em um problema de *fuzzy clustering*. Este processo é conduzido por um critério chamado pelos autores de “critério de retenção da estrutura”. Sutilmente diferente de um *wrapper* de seleção de atributos, o objetivo deste algoritmo é selecionar as características da informação mantendo o máximo possível da estrutura presente nos dados originais.

A informação é fragmentada pelo algoritmo *Fuzzy C-Means* (*FCM*) e a retenção da estrutura de cada atributo é determinada pela sua capacidade de representação dos padrões originais. Dada a natureza combinatória da seleção

de atributos, o processo de otimização subjacente é realizado através dos algoritmos baseados em populações. Os resultados do método proposto mostram que o conteúdo estrutural da maioria dos conjuntos de dados pode ser preservado, mesmo para uma redução significativa do número de atributos.

He *et al.* (2009) propõem um novo algoritmo *wrapper* para seleção de atributos denominado *discrete Binary Differential Evolution* (*BDE*). O algoritmo proposto baseia-se em uma abordagem de AE conhecida como Evolução Diferencial (ED). A ED padrão (Storn e Price, 1997; Thom de Souza *et al*., 2009) gera novos indivíduos pela adição da diferença vetorial ponderada entre dois indivíduos aleatórios da população a um terceiro indivíduo e gera novos indivíduos pela adição da diferença (daí o nome diferencial) vetorial ponderada entre dois indivíduos aleatórios da população a um terceiro indivíduo.

O algoritmo proposto por He *et al.* (2009) codifica as soluções na forma binária dependente de probabilidades. Assim, o método pode ser usado para lidar com problemas de otimização discreta, como é o caso da seleção de atributos. Os classificadores utilizados foram o *Support Vector Machine* (*SVM*), a árvore de decisão *Classification and Regression Tree* (*C&R Tree*) e Redes Neurais Artificiais *Radial Basis Function* (RNA-*RBF*). O método apresentou considerável melhora na taxa de classificações corretas para todos os classificadores com todas as seis bases de dados nos quais foi aplicado.

Huang *et al.* (2012) utilizam *Ant Colony Optimization* (*ACO*) como método de seleção de atributos para melhoria da acurácia de classificação de sinais de eletromiografia. Desenvolvido por Dorigo e Stützle (2004) para solucionar problemas de otimização combinatória, o *ACO* tem como inspiração o comportamento das formigas na busca por alimento.

Em geral, o *ACO* apresenta duas etapas: (i) soluções candidatas são construídas usando um modelo de feromônio, que é uma distribuição de probabilidade parametrizada sobre o espaço de soluções e (ii) as soluções candidatas são usadas para modificar os valores do feromônio de tal modo que estes novos valores influenciem as amostras futuras com soluções de melhor qualidade. Embora neste trabalho, o enfoque maior tenha sido na redução de dimensionalidade, os resultados da acurácia de classificação melhoraram consideravelmente.

Diversos trabalhos recentes têm mostrado a equivalência e, em alguns casos, a superioridade do *EDA* em relação ao outras técnicas baseadas em populações para seleção de atributos. Saeys *et al.* (2006) utilizaram o *EDA* para ranquear, por ordem de relevância, os atributos para o processo de classificação, abordagem conhecida como *feature ranking wrapper*. Abegaz *et al.* (2011) comparam o *EDA* com o *steady state Genetic Algorithm* (*SSGA*) como *wrappers* de seleção de atributos em dados biométricos. Neste trabalho, a acurácia do *EDA* mostrou-se superior à do *SSGA* em todos os conjuntos de dados analisados.

González *et al.* (2009) usaram um tipo específico de *EDA*, conhecido como *Univariate Marginal Distribution Algorithm* (*UMDA*), como *wrapper* para melhoria de desempenho de um classificador baseado em RL em dados de *microarray* de *DeoxyriboNucleic Acid* (*DNA*). Segundo os autores, a seleção de atributos é utilizada, pois a Regressão Logística não apresenta bons resultados em situações em que o número de variáveis é maior que o número de instâncias (situação frequente em dados de *microarray*). Os resultados empíricos também confirmam a superioridade do *EDA* utilizado em relação aos algoritmos com os quais foi comparado.

Hong *et al.* (2008) combinam o *PBIL* e o *Clustering Ensembles* para a resolução de um problema de aprendizagem não supervisionada (agrupamento). A abordagem proposta obtém uma solução de agrupamento inicial pelo *Clustering Ensembles* e aplica o *PBIL* para encontrar o subconjunto de atributos que melhor ajusta a solução de agrupamento obtida.

Assim, enquanto o *Clustering Ensembles* aproveita as melhores soluções de agrupamento combinando-as em uma única solução “consensual”, o *PBIL* parte desta solução para buscar o subconjunto de atributos ótimo para o problema. O algoritmo de agrupamento treinado no novo subconjunto (subconjunto ótimo) deve encontrar a solução de agrupamento mais similar àquela obtida pelo *Clustering Ensembles* com os dados originais (todos os atributos).

A abordagem proposta por Hong *et al.* (2008), útil especialmente para problemas de aprendizagem sem critério de consenso, foi aplicada em quinze conjuntos de dados reais com estas características. O resultado da abordagem

foi superior aos obtidos através dos métodos *scatter separability* e *DB-index* para geração da solução inicial com e sem o seletor de atributos baseado em *PBIL*.

Uma extensão do *PBIL* proposta por Shapiro (2002) tenta cobrir de forma mais uniforme as regiões de um espaço de busca e, de forma mais detalhada, os pontos nas regiões observadas. Ele faz isso, modificando a dinâmica do *PBIL* para que o algoritmo tenha condições de "rever" algumas decisões mal tomadas.

A extensão proposta não é aplicada ao problema de seleção de atributos, mas sim a dois outros problemas bem caracterizados: o *needle-in-the-haystack*, em português, "agulha no palheiro" (problema de busca em um espaço onde todas as soluções candidatas são igualmente boas, exceto uma, que é o ótimo global) e o *one-max* (problema monótono contínuo com uma solução global ótima).

Shapiro (2002) alega que seu algoritmo é mais independente da taxa de aprendizagem que o *PBIL* padrão, e por isso, permite dar "passos" maiores. Embora esta extensão amenize um pouco o problema da dependência dos passos iniciais do *PBIL*, ao permitir que uma região já vasculhada seja visitada novamente, e tenha se mostrado boa para encontrar o ótimo global para o problema *needle-in-the-haystack*, para o outro problema (o *one-max*), a técnica apresentou um custo computacional alto e sem qualquer garantia de encontrar o ótimo.

Esta técnica apesar de eficiente para problemas "*flat*" como o *needle-in-the-haystack*, não se mostrou tão adequada em problemas "bem comportados" como o *one-max*. Em resumo, esta extensão do *PBIL* pode consumir muito tempo computacional "perambulando" por regiões já visitadas, devido à indecisão sobre que direção tomar.

Keramati *et al.* (2011) propõem a aplicação de AC para seleção de atributos na forma de *wrapper* para melhoria do desempenho de um classificador *NB*. O AC complementa um outro algoritmo (geralmente também baseado em populações) ao adicionar um componente cultural ao algoritmo ao qual complementa. Em seu trabalho, Keramati *et al.* (2011) comparam um AG clássico com sua versão combinada a um AC.

O componente utilizado foi o conhecimento situacional obtido através do processo de busca e o AG cultural mostrou desempenho superior quando comparado ao AG clássico em um conjunto de dados *benchmark*. Os autores atribuem a superioridade do AG cultural à sua rápida convergência. Esta rápida convergência permitiu ao AG, além de sua habitual capacidade de "*exploration*" (busca em "amplitude") de boas regiões no espaço de soluções, uma melhor habilidade de "*exploitation*" (busca em "profundidade") para escolha das melhores soluções dentro de uma boa região. Assim, espera-se que o uso do AC combinado ao *PBIL*, conforme proposto no presente trabalho, também traga ganhos ao *PBIL* tanto em termos de *exploration* quanto de *exploitation* na tarefa de seleção de atributos.

Reunanen (2012) descreve "armadilhas" relacionadas à comparação entre algoritmos de seleção de atributos. O autor critica a confiança indiscriminada em afirmações presentes na literatura, mesmo aquelas bem aceitas e disseminadas. Ele alerta que, mesmo para pesquisadores experientes, é surpreendentemente fácil obter resultados inválidos, se a metodologia utilizada não for rigorosa o suficiente.

Reunanen (2012) confirma suas proposições por meio de testes empíricos comparando dois algoritmos de seleção de atributos, o *Sequential Backward Selection* (*SBS*) e o *Sequential Backward Floating Selection* (*SBFS*). Ambos são aplicados a um problema de seleção de atributos da área elétrica relacionado à injeção de corrente alternada. O objetivo do problema é a minimização do número de eletrodos necessários para inferência da distribuição de condutividade em um volume alvo.

Basicamente, todas as afirmações criticadas pelo autor são relacionadas à ignorância sobre o *overfitting*. Algumas destas principais "falácias" sobre o *overfitting* apontadas pelo autor incluem: buscas mais exaustivas são capazes de encontrar subconjuntos de atributos melhores que técnicas mais simples; podas no subconjunto de atributos, frequentemente, melhoram a acurácia de classificação; o tamanho do subconjunto de atributos ótimo é, normalmente, muito menor do que o número total de atributos candidatos.

3 *KNOWLEDGE DISCOVERY IN DATABASES* (*KDD*)

O processo *KDD* pode ser definido como um conjunto de atividades contínuas que compartilham o conhecimento descoberto a partir de bases de dados (Steiner *et al.*, 2007). Segundo Fayyad *et al.* (1996), o *KDD* é composto por cinco etapas:

1. Seleção dos dados: etapa de preparação e seleção dos dados utilizados;
2. Pré-processamento dos dados: etapa de remoção ou atenuação de possíveis ruídos presentes nos dados selecionados;
3. Transformação dos dados: etapa em que são aplicados tratamentos e transformações sobre os dados para melhor adequá-los à extração de padrões;
4. Mineração de dados (*data mining*): busca e extração de padrões nos dados por meio de algoritmos;
5. Interpretação e avaliação dos resultados: análise da relevância e refinamento do conhecimento descoberto para o domínio em questão.

A Figura 3.1 apresenta as etapas do processo de *KDD*. O enfoque deste trabalho concentra-se na etapa (2) de pré-processamento por seleção de atributos e na etapa (3) de transformação por análise geométrica de dados, visando a melhoria de desempenho da etapa (4) de mineração de dados por meio de algoritmos de classificação de padrões.

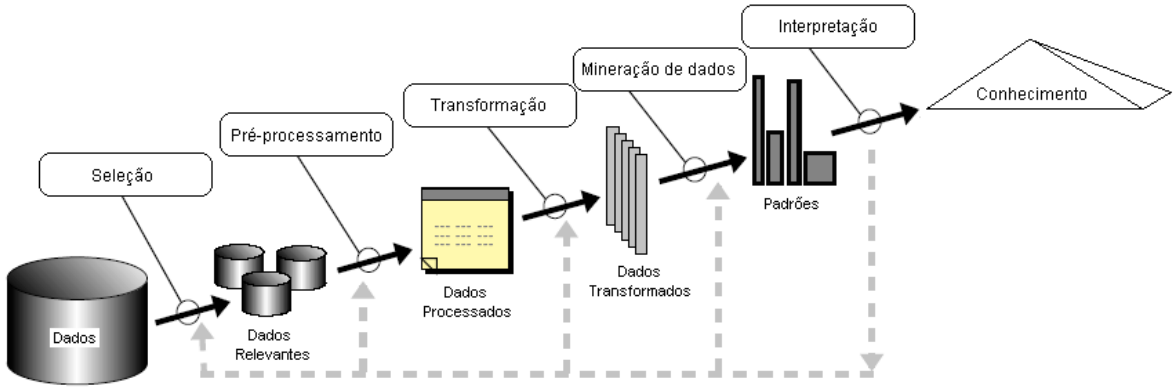

Figura 3.1 – Etapas do processo de *KDD* (Fayyad *et al.*, 1996).

Portanto, o subprocesso do *KDD* operacionalizado no presente trabalho é apresentado na Figura 3.2. Este subprocesso recebe como entradas os

dados relevantes previamente selecionados na etapa (1) e retorna como saída os padrões a serem posteriormente interpretados na etapa (5) pelos tomadores de decisão, isto é, por especialistas nas áreas de negócio de cada conjunto de dados.

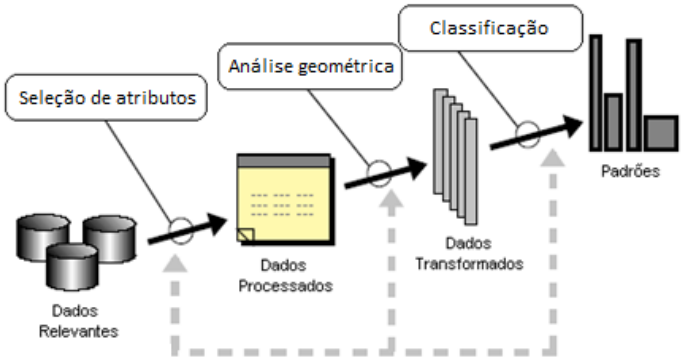

Figura 3.2 – Etapas do *KDD* operacionalizadas no presente trabalho.

Os conceitos teóricos relacionados às três etapas apresentadas na Figura 3.2, seleção de atributos, análise geométrica de dados e classificação de padrões, bem como os algoritmos utilizados para cada uma destas etapas no presente trabalho, são apresentados respectivamente nas seções 3.1, 3.2 e 3.3.

3.1 SELEÇÃO DE ATRIBUTOS

À primeira vista, quanto maior a quantidade de dados, melhor deverá ser o desempenho dos algoritmos de mineração de dados. Esta afirmação é parcialmente verdadeira, pois a qualidade dos algoritmos não só depende da quantidade, mas também da qualidade dos dados (informações relevantes). Por exemplo, a adição de novas colunas com valores aleatórios em um conjunto de dados poderia aumentá-lo de tamanho, mas não aumentaria a quantidade de informação relevante.

A presença de atributos irrelevantes ou de atributos correlacionados pode gerar algum grau de degradação na acurácia de classificadores (Kohavi e John, 1997). Por esta razão, limitar a quantidade de atributos somente aos que mostrarem-se potencialmente úteis ao processo de extração de padrões, torna-

se uma tarefa importante antes da aplicação dos algoritmos de classificação (Berthold *et al.*, 2010).

O processo de identificação e eliminação dos atributos irrelevantes ou prejudiciais à extração de padrões é denominado seleção de atributos. A seleção de atributos é uma das formas de “limpeza” executadas na fase de pré-processamento de dados e apresenta-se como uma tarefa de grande importância a ser realizada antes da aplicação de algoritmos de extração de padrões.

Existem ainda outras razões para a aplicação de algoritmos de seleção de atributos, como a eficiência computacional, por exemplo. O consequente ganho computacional devido à redução de dimensionalidade provida pela seleção de atributos torna-a especialmente útil a bases de dados com muitas variáveis. Mesmo assim, a aplicação da seleção de atributos a conjuntos de dados com poucas variáveis também pode trazer consideráveis resultados, sobretudo em termos de redução do viés presente nos dados (Berthold *et al.*, 2010).

O objetivo da seleção de atributos é reduzir o número de atributos e, também, melhorá-los qualitativamente em um problema de classificação, de modo a simplificar o seu entendimento e facilitar a sua solução. Idealmente, deseja-se selecionar um subconjunto ótimo de atributos a partir do conjunto completo de atributos disponíveis. Um subconjunto de atributos é dito ótimo quando o uso de qualquer outro não resultaria em um melhor desempenho final (Kohavi e John, 1997). O subconjunto ótimo não é necessariamente único, sendo possível existirem subconjuntos diferentes que permitam obter desempenhos equivalentes.

A ideia é obter atributos significativos e em um número justificável. Isto não somente reduz o custo computacional, como também pode ajudar a prevenir o classificador de *overfitting* dos dados (Csirik e Bunke, 2011) e a tornar o problema mais fácil de ser manipulado.

Existem basicamente dois critérios para se mensurar a qualidade de um atributo:

a) Relevância: mede o ganho proporcionado por um determinado atributo para a predição das classes. John *et al*. (1994) distinguem três tipos de

atributos: os fortemente relevantes; os fracamente relevantes; e os irrelevantes. É importante salientar que a relevância de um atributo não implica em sua presença no subconjunto ótimo (embora seja o caso mais comum). Como exemplo de atributos relevantes que podem não pertencer ao subconjunto ótimo podemos citar atributos fracamente relevantes (que podem estar sendo substituídos por outros redundantes que contenham a mesma informação) e atributos muito ruidosos que, mesmo relevantes, podem dificultar de alguma forma a aprendizagem.

b) Redundância: dois atributos são redundantes entre si, em menor ou maior grau, se eles são parcial ou completamente correlacionados. Embora Yu e Liu (2004) afirmem que um classificador possa ser “distraído” por atributos altamente correlacionados, Guyon e Elisseeff (2003) ponderam que o uso de alguns poucos atributos redundantes pode ajudar a reduzir ruídos e melhorar a classificação em algumas situações.

Diferentes modelos de classificação apresentam diferentes reações aos graus de relevância e redundância dos atributos apresentados, e até mesmo ao número de atributos utilizados. Por isso, um subconjunto de atributos que seja ótimo para um determinado modelo, não necessariamente o será para outro. Em algumas situações práticas, pode ser preferível, por exemplo, um subconjunto que funcione bem para diferentes classificadores, mesmo que não seja o subconjunto ótimo para nenhum deles.

A literatura divide os algoritmos de seleção de atributos em duas abordagens principais: filtro e *wrapper* (Yu e Liu, 2004). Alguns autores acrescentam ainda uma terceira abordagem, conhecida como *embedded*, na qual a seleção de atributos é realizada como parte integrante do classificador (o próprio classificador é capaz de decidir quais são os atributos relevantes para representar o conhecimento extraído). Um exemplo de *embedded* é o classificador *C*4.5 (Quinlan, 1993).

Uma breve definição das abordagens filtro e *wrapper* é apresentada a seguir:

3.1.1 FILTRO

O termo filtro vem do inglês, *filter*, devido à forma com que é aplicado, em uma etapa anterior ao treinamento do classificador, ao qual não é em momento algum apresentado o conjunto completo de atributos (John *et al*., 1994; Kohavi e John, 1997). A ideia é filtrar os atributos segundo algum critério, frequentemente de natureza estatística, eliminando os atributos irrelevantes (John *et al*., 1994). Esta filtragem considera características gerais do conjunto de dados para selecionar alguns atributos e excluir outros. Sendo assim, o filtro é independente do algoritmo de classificação que, simplesmente, receberá como entrada a saída fornecida pelo filtro.

Os atributos são agrupados conforme propriedades que presumem relevância, como ortogonalidade e conteúdo de informação. A aplicação do filtro pode ser um processo relativamente rápido. De maneira geral, os filtros operam unicamente com base nas propriedades estatísticas intrínsecas dos dados na tentativa de determinar a relevância de cada atributo. Embora não haja qualquer forma de realimentação entre o algoritmo de seleção de atributos e o algoritmo de classificação, é possível escolher ou projetar filtros adequados às características conhecidas de um classificador específico (Guyon e Elisseeff, 2003).

A literatura cita o baixo custo computacional e a natureza “genérica” dos resultados produzidos pelo filtro, como uma de suas vantagens (Kohavi e John, 1997). No entanto, a natureza genérica (o fato dos resultados não estarem vinculados a um classificador específico) possui aspectos positivos e negativos. Resultados mais genéricos podem ser utilizados satisfatoriamente em uma gama maior de problemas e diferentes algoritmos de classificação, mas a ausência de uma “sintonia” específica pode ter como consequência a falta de garantia de bons resultados finais para um classificador específico. Em algumas situações práticas o classificador é definido com antecedência, tornando preferíveis atributos com maior aderência ao classificador pré-determinado.

Exemplos de filtros incluem testes estatísticos (como o *chi*-quadrado e o *t-student*), medidas de distância (como a distância Euclidiana) e medidas

relacionadas a ganho de informação. A Figura 3.3 apresenta o fluxo de funcionamento do filtro. Nesta figura, os dados brutos com todos os atributos disponíveis começam sendo submetidos ao processo de seleção de atributos. O algoritmo de seleção “filtra” uma parte destes atributos com base em algum critério pré-definido e, ao final deste processo, o subconjunto de atributos resultante (atributos selecionados) é adotdo pelo algoritmo de classificação.

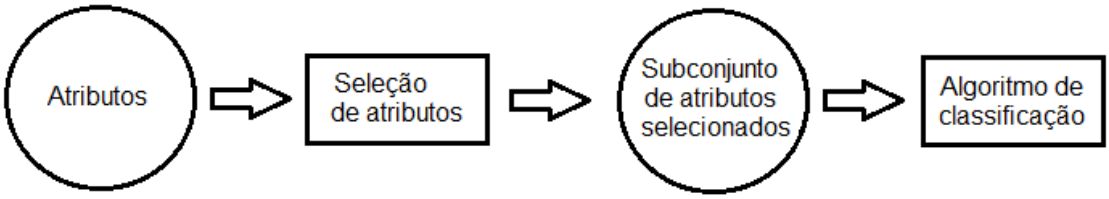

Figura 3.3 – Representação do filtro.

3.1.2 *WRAPPER*

O termo “*wrapper*” deriva do inglês “embrulho”, se referindo ao fato do algoritmo de seleção de atributos estar “empacotado” em torno de um algoritmo de classificação, em outras palavras, o algoritmo de seleção de atributos se utiliza de um classificador para avaliar o desempenho dos subconjuntos de atributos por ele gerados. O objetivo primário do *wrapper* é obter um subconjunto de atributos que seja o mais indicado para este classificador. Com isso, o *wrapper* não possui garantias de bom desempenho para outros tipos de classificadores.

Enquanto a abordagem filtro visa encontrar características relevantes nos atributos e opera independentemente do algoritmo de classificação, na abordagem *wrapper* a avaliação da relevância e da redundância de atributos é feita de maneira implícita por meio da avaliação de subconjuntos de atributos.

Este processo ocorre iterativamente em duas etapas: (i) para cada subconjunto analisado, constrói-se um novo modelo de classificação e (ii) o modelo é avaliado, de tal modo que o resultado desta avaliação é usado como função de avaliação do subconjunto de atributos. Este processo permite que o *wrapper* busque não apenas atributos relevantes, mas um subconjunto de

atributos que seja o mais indicado para determinada aplicação. A Figura 3.4 ilustra o funcionamento do *wrapper*.

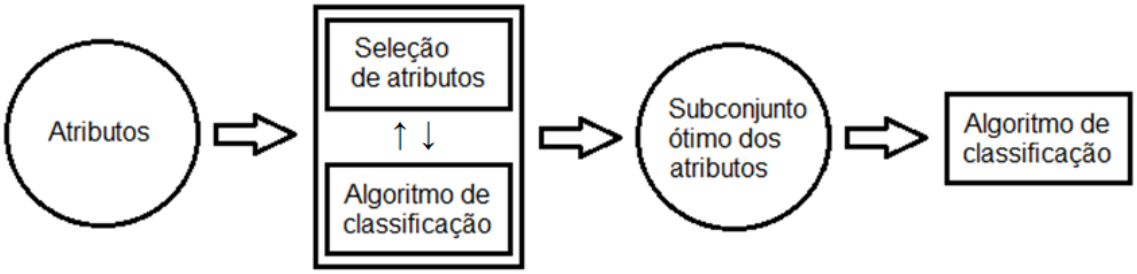

Figura 3.4 – Representação do *wrapper*.

De maneira geral, o *wrapper* produz melhores resultados que o filtro devido à sua capacidade em considerar dependências entre atributos e à maior sinergia resultante da interação entre os algoritmos de seleção de atributos e de classificação (Hall e Holmes, 2003; Saeys *et al*., 2006). Espera-se que a utilização de um classificador com o subconjunto de atributos selecionado por ele próprio (com a mesma polarização de aprendizado) forneça uma estimativa de melhor acurácia (Kohavi e John, 1997; Baranauskas, 1999).

Apesar das vantagens, o *wrapper* possui dois principais inconvenientes: apresenta risco mais elevado de *overfitting* e maior custo computacional (Saeys *et al*., 2006; Kohavi e John, 1997; Baranauskas e Monard, 1998). Sugestões para atenuar o problema do custo computacional incluem a utilização de classificadores mais “leves” (Saeys *et al*., 2006); a adoção de métodos de estimação de erro simplificados, como o *holdout*; e, no caso da necessidade de estimação de erro por validação cruzada, usar um número reduzido de *folds* (5 ao invés de 10, por exemplo). Embora a literatura não apresente muitas formas de se diminuir o impacto do *overfitting* para o *wrapper*, uma recomendação (nem sempre possível) é usar um número maior de amostras.

Apesar destes inconvenientes, o uso do *wrapper* justifica-se devido ao seu objetivo primário: obter um subconjunto de atributos que seja o mais indicado para um classificador pré-determinado. Evidentemente, a viabilidade de sua implementação prática está condicionada à existência de recursos computacionais suficientes à disposição. Além disso, as desvantagens servem de estímulo à pesquisa de novas técnicas de *wrapper* de modo a continuar

obtendo bons resultados, mas com um custo computacional razoável e menor impacto de *overfitting*.

Outro ponto importante no que se refere à estratégia do *wrapper* é o método de busca utilizado para seleção de atributos. O presente trabalho agrupa estes métodos em quatro categorias:

a) Métodos exaustivos: testam todas as possibilidades de soluções na chamada "força bruta". Métodos exaustivos não são recomendados pelo simples fato de que a seleção de atributos é um problema *NP-hard* (Amaldi e Kann, 1998). Cada estado no espaço de busca de seleção de atributos corresponde a certo conjunto de atributos de um total de 2^n diferentes combinações de atributos possíveis, onde n é o total de atributos do conjunto de dados. Realizar uma busca exaustiva em um espaço de soluções como este pode ser computacionalmente intratável. Assim, buscas exaustivas devem ser aplicadas somente a conjuntos de dados pequenos (com poucos atributos e poucas instâncias).

b) Métodos aleatórios: possuem a vantagem de serem menos afetados por mínimos locais. Em contrapartida, têm baixa capacidade de controlar o processo de busca de modo a conseguir convergir adequadamente. Um de seus principais representantes é o algoritmo Las Vegas (Liu e Setiono, 1996).

c) Métodos construtivos: geralmente se utilizam de algoritmos de busca *greedy* apresentando risco de convergência para mínimos locais e, raramente, encontrando a melhor solução global (Kohavi e John, 1997; Vafaie e De Jong, 1993). Os métodos mais famosos são o *Forward Selection*, que começa com um subconjunto vazio e vai incluindo "bons" atributos a cada passo e, seu método "inverso", o *Backward Elimination*, que começa com um subconjunto formado por todos os atributos e vai eliminando atributos "ruins".

Os algoritmos terminam sua execução quando um subconjunto de determinado tamanho é encontrado ou quando um mínimo é alcançado. A vantagem do *Backward Elimination* é que determinar atributos irrelevantes é, normalmente, mais fácil. Por outro lado, os subconjuntos resultantes são usualmente maiores que aqueles obtidos pelo *Forward Selection*. Ambos os métodos convertem o problema 2^n em n^2 ao incluir, a cada passo, o atributo de maior (ou excluir o de menor) ganho de acurácia de classificação. Contudo,

ambas as abordagens apresentam uma propriedade indesejável: uma vez que um atributo é incluído ou removido, através do *Forward Selection* e *Backward Elimination*, respectivamente, não existe volta.

Para contornar alguns problemas e melhorar os resultados destes métodos, diversas técnicas têm sido propostas na literatura como: buscas bi-direcionais (Caruana e Freitag, 1994; Pudil *et al*., 1994), que permitem adição e remoção de atributos em uma mesma execução; buscas sequenciais (Hall e Holmes, 2003; Reunanen, 2012), que convertem o problema 2^n em $2n$ ao ranquear os atributos antes da aplicação do *wrapper* (o algoritmo constrói n subconjuntos de atributos, sendo que o primeiro é formado somente pelo melhor do ranking, o segundo pelos dois melhores e, assim por diante); entre outras.

d) Meta-heurísticas[5]: como o espaço de subconjuntos possíveis cresce exponencialmente com o número de atributos, estes métodos de busca são, em geral, os mais indicados para guiar a busca a um subconjunto ótimo. O motivo é que as meta-heurísticas de otimização têm maior capacidade de evitar mínimos locais que os métodos construtivos, sem tanta dificuldade de convergência quanto os métodos mais "puramente" aleatórios. Devido ao seu mecanismo "aleatório-orientado", diversas abordagens meta-heurísticas têm sido propostas nos últimos anos para seleção de atributos, com especial destaque para os algoritmos de Inteligência Computacional (IC).

A IC dispõe de uma série de algoritimos inspirados em fenômenos naturais e sociais para solução computacional de problemas, tais como as RNAs, Sistemas Nebulosos (SNs), *Simulated Annealing* (*SA*) e algoritmos baseados em populações (Castro, 2007).

Os algoritmos baseados em populações avaliam soluções candidatas factíveis dentro do espaço de busca. As avaliações são realizadas iterativamente, sendo que a cada iteração são avaliadas várias soluções. Em outras palavras, o processo de resolução do problema corresponde a uma sequência de ações (passos) que levam a um desempenho desejado, ou melhoram o desempenho relativo de soluções candidatas. Exemplos de

[5] Um *survey* recente sobre meta-heurísticas pode ser obtido em Lopes *et al.* (2013).

algoritmos baseados em populações incluem a Inteligência Coletiva, os Sistemas Imunológicos Artificias (SIA) e os AE (Boussaïd *et al*., 2013).

Os AE realizam a avaliação das populações com base em operadores genéticos (seleção, cruzamento, mutação, *etc*.) inspirados na Teoria da Evolução Natural de Charles Darwin. Fazem parte deste grupo de algoritmos, técnicas de AG, Programação Genética (PG), Estratégias Evolutivas (EE), ED, AC, *EDA*, entre outras.

A Figura 3.5 apresenta uma taxonomia das técnicas mencionadas de IC[6], destacando em contorno vermelho os algoritmos utilizados para seleção de atributos no presente trabalho.

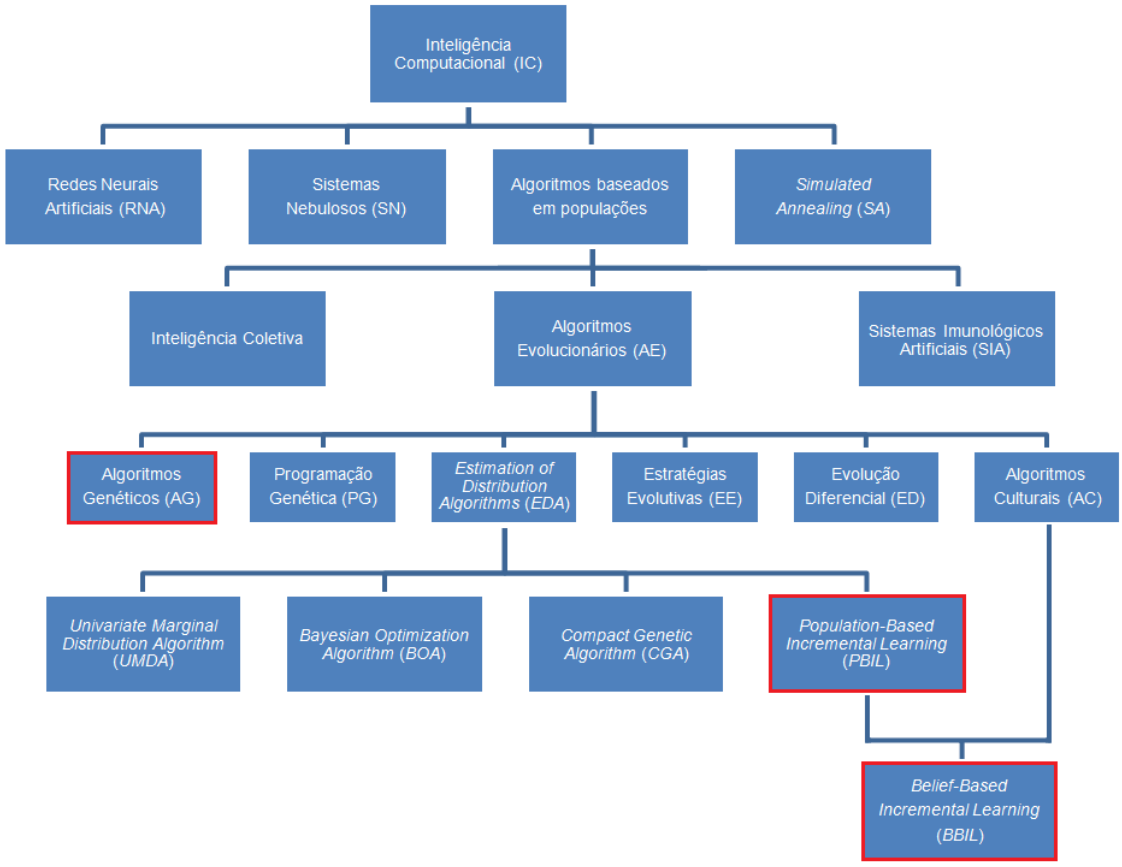

Figura 3.5 – Taxonomia das técnicas de IC destacando os algoritmos utilizados para seleção de atributos no presente trabalho.

Uma das técnicas de AE mais exploradas na literatura para este fim, geralmente apresentando resultados satisfatórios, são os AG (Tan *et al*., 2006;

[6] A IC é uma subárea da Computação Natural (Castro, 2007).

Martin-Bautista e Vila, 1999; Shi *et al*., 2011). Mesmo assim, existem inúmeras situações nas quais algoritmos mais simples e rápidos obtêm resultados superiores ao AG para alguns conjuntos de dados (Jain e Zongker, 1997). Nos AG (Holland, 1992), operadores genéticos como cruzamento e mutação são aplicados iterativamente a uma população de subconjuntos de atributos existente para gerar novos subconjuntos. Os melhores subconjuntos são selecionados de acordo com a acurácia de classificação a cada iteração.

Porém, quanto maior o número de subconjuntos de atributos avaliados, maior é o esforço computacional e a propensão ao *overfitting* (Csirik e Bunke, 2011). Como as buscas nos AG são intensivas, a probabilidade de se selecionar um subconjunto de atributos que cause *overfitting* é grande (Loughrey e Cunningham, 2004).

3.1.3 *ESTIMATION OF DISTRIBUTION ALGORITHMS* (*EDA*)

Diferentemente dos AG, outra classe de AE, denominada *EDA*, não necessita de determinadas operações, como cruzamento e mutação (e seus respectivos parâmetros). Sob este aspecto, os *EDA* podem ser considerados mais simples do que os AE clássicos, como os AG. Nos *EDA*, os indivíduos são amostras. A cada geração, as amostras são ajustadas a um modelo probabilístico de distribuição para determinação das melhores novas amostras.

Os *EDA* aliam a maior capacidade de métodos mais aleatórios em evitar mínimos locais (ao avaliar soluções descritas por distribuições de probabilidade) e a habilidade dos AE em geral de controlar evolutivamente a convergência do processo de busca. A Figura 3.6 ilustra o funcionamento de um *EDA* na forma de um fluxograma.

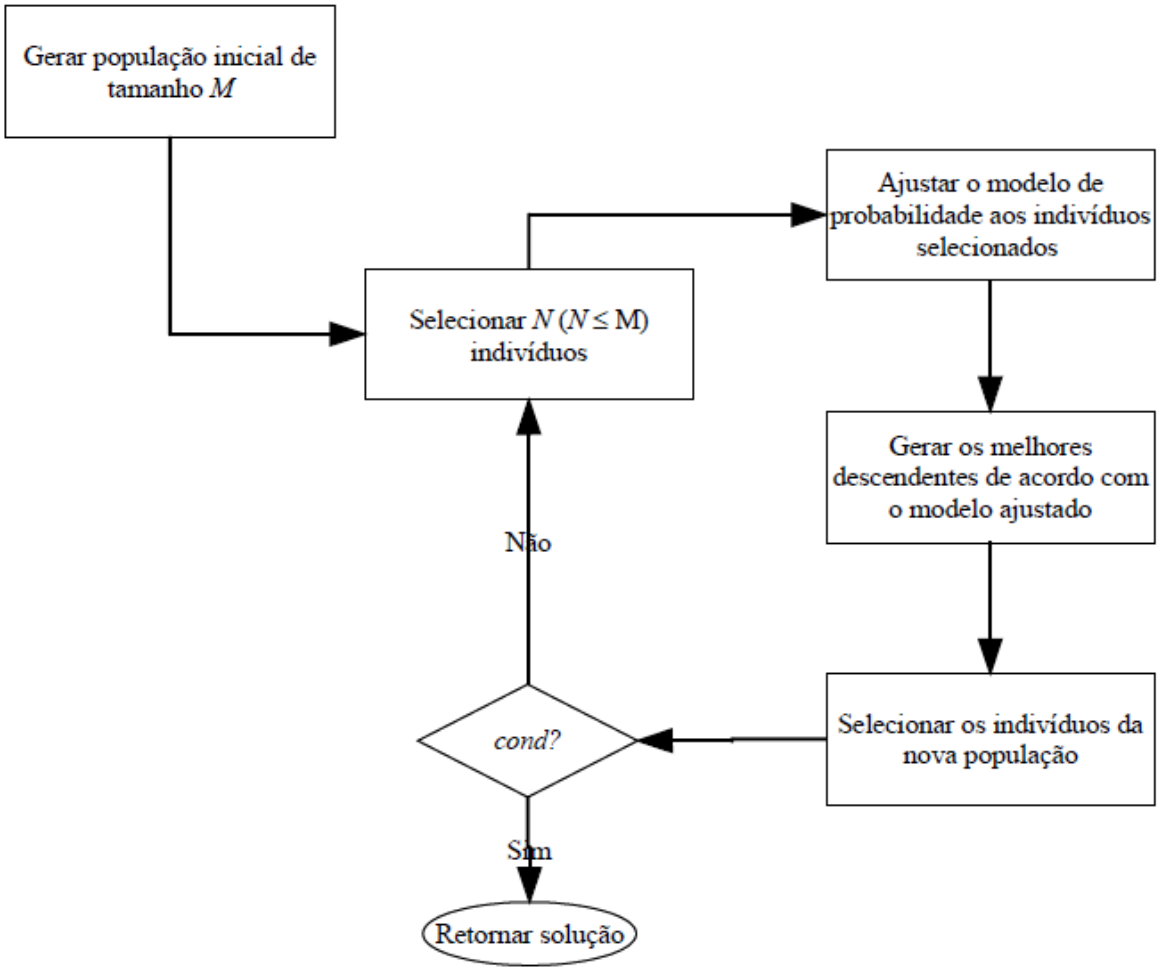

Figura 3.6 – Fluxograma de funcionamento de um *EDA*.

Após a geração de uma população inicial de tamanho M, selecionam-se os $N \leq M$ melhores indivíduos (amostras). Para cada geração, o algoritmo ajusta o modelo de probabilidade à população, gera os melhores descendentes de acordo com o modelo (sem cruzamento ou mutação) e seleciona os $N \leq M$ indivíduos da nova população. O algoritmo repete este processo até que um critério de parada pré-determinado seja atingido, em geral um *fitness* mínimo ou um número máximo de iterações.

O algoritmo proposto neste trabalho baseia-se no *PBIL*, um tipo específico de *EDA*, é apresentado a seguir na seção 3.1.3.1.

3.1.3.1 *POPULATION-BASED INCREMENTAL LEARNING* (*PBIL*)

Introduzido por Baluja (1994), o *PBIL* representa a população de indivíduos por um vetor de probabilidades $p_{l-1}(x)$, obtido pela expressão (3.1):

$$p_{l-1}(x) = \left(p_{l-1}(x_1), \dots, p_{l-1}(x_i), \dots, p_{l-1}(x_n)\right) \tag{3.1}$$

onde $p_{l-1}(x_i)$ refere-se a probabilidade de se obter o valor "1" no *i*-ésimo componente de D_{l-1}, a população de indivíduos na geração imediatamente anterior à l-ésima geração.

A cada geração, por simulação do vetor de probabilidades $p_{l-1}(x)$, é obtido um conjunto de M indivíduos considerados como a nova população. Cada um desses M indivíduos são avaliados e os $N \leq M$ melhores são selecionados. Indivíduos selecionados são denotados conforme a expressão (3.2):

$$x_{1:M}^{l-1}, \dots, x_{i:M}^{l-1}, \dots, x_{N:M}^{l-1}. \tag{3.2}$$

Estes indivíduos selecionados são usados para atualizar o vetor de probabilidades a partir do qual a próxima população de indivíduos será gerada, conforme a expressão (3.3):

$$p_l(x) = (1-\alpha)p_{l-1}(x) + \alpha \frac{1}{N} \sum_{k=1}^{N} x_{k:M}^{l-1}, \tag{3.3}$$

onde $\alpha \in (0,1]$ é um parâmetro do algoritmo denominado taxa de aprendizagem. Este processo de adaptação do vetor de probabilidades continua até sua convergência. A expressão (3.3) é usada quando o problema possui variáveis binárias, como é o caso do problema de seleção de atributos, no qual o valor "0" para o atributo (variável binária) implica em sua retirada e "1" em sua permanência.

Com o objetivo de estimar a distribuição de probabilidade conjunta da qual a próxima população de indivíduos é gerada, a maior parte das abordagens *EDA* considera somente o subconjunto de indivíduos selecionados e descarta o subconjunto de indivíduos não selecionados. Em contraste a estas

outras abordagens, o *PBIL* se baseia no vetor de probabilidades $p_{l-1}(x)$ para o cálculo do vetor da geração seguinte, $p_l(x)$.

O *PBIL* tem sido aplicado com sucesso nos últimos anos a vários problemas, tais como sincronização de sistemas caóticos (Coelho e Grebogi, 2010), identificação de raízes em problemas de restrições geométricas (Arinyo *et al.* 2011) e problema da mochila multidimensional (Wang *et al.*, 2012).

3.1.3.2 *PBIL* APLICADO À SELEÇÃO DE ATRIBUTOS

A aplicação do *PBIL* para a seleção de atributos é também bastante simples. Cada atributo é representado por uma variável binária (gene) dentro de um indivíduo, conforme a expressão (3.3). A população inicial é gerada aleatoriamente e é inicializado um vetor de probabilidades com 50% de chance de cada atributo ser selecionado. A geração aleatória da população é feita da seguinte forma: para cada gene dentro de um indivíduo, são gerados números aleatórios pertencentes ao intervalo [0, 1] obedecendo a probabilidade para cada atributo contida no vetor de probabilidades. Se o número gerado for menor que a probabilidade daquele atributo ser escolhido o atributo é selecionado, caso contrário, não.

Após a geração de toda a população, avalia-se o *fitness* (o desempenho do algoritmo de extração de padrões para o caso de *wrappers*) de cada indivíduo, selecionando o(s) mais apto(s). O vetor de probabilidades é atualizado pelo(s) indivíduo(s) selecionado(s) de acordo com a taxa de aprendizagem α previamente parametrizada e, uma nova geração é criada com base no vetor de probabilidades recém-atualizado. À medida que as gerações vão passando a probabilidade de cada atributo vai sendo influenciada por novos bons indivíduos gerados obedecendo a distribuição de probabilidade do último vetor atualizado. O processo termina quando um critério de parada previamente determinado é satisfeito.

3.1.4 ALGORITMOS CULTURAIS (AC)

Os AC foram propostos por Reynolds (1994) como um complemento à metáfora adotada pelos AE com base na evolução cultural. Os AC se baseiam em teorias de sociologia e arqueologia para modelar a evolução cultural e consistem de dois níveis, a micro e a macro-evolução.

A micro-evolução (nível populacional) é, em geral, formada de um algoritmo de busca aleatória baseada em população, como um AG, por exemplo. O processo evolutivo no nível populacional baseia-se, principalmente, em micro-interações entre os indivíduos. Tais interações são responsáveis pela diversificação das soluções candidatas. A macro-evolução (nível cultural) extrai do nível populacional experiências que podem ser armazenadas e usadas para impactar o processo de busca. Neste contexto, um AC pode incorporar conhecimento obtido durante o processo evolucionário usado em um AE para tornar a busca mais eficiente. A Figura 3.7 ilustra o funcionamento básico de um AC.

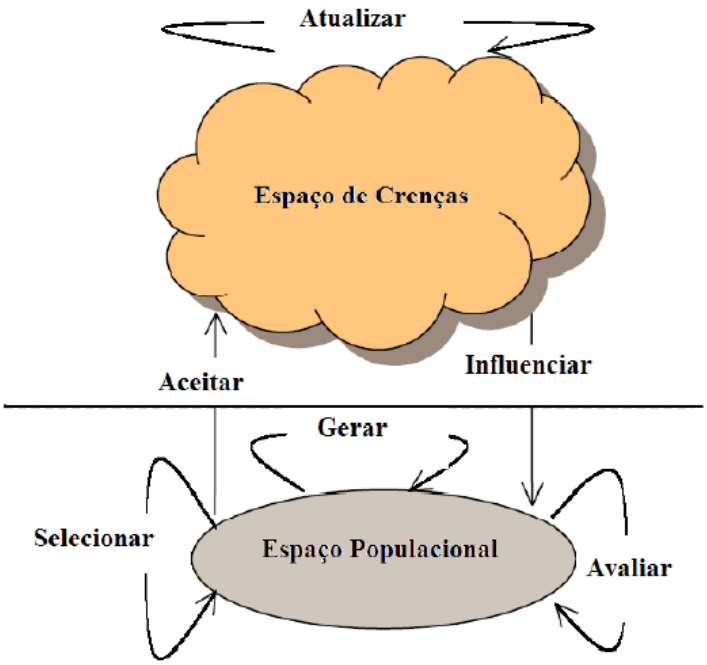

Figura 3.7 – Funcionamento básico de um AC (Reynolds, 1994).

O espaço populacional é o conjunto de possíveis soluções modelado pelo processo de micro-evolução. O espaço de crenças é o repositório de informações da macro-evolução no qual os indivíduos podem armazenar suas

experiências para outros indivíduos aprendê-las indiretamente. Um conjunto de protocolos de comunicação controla as interações entre os dois espaços. Estes protocolos determinam quais membros da população são aceitáveis para ajustar o conhecimento contido no espaço de crenças e, quanto este conteúdo deveria impactar as populações nas diferentes gerações.

Cinco diferentes tipos de conhecimento são tipicamente usados para representar o conhecimento no espaço de crenças. Esses tipos são: conhecimento situacional, conhecimento histórico, conhecimento normativo, conhecimento topográfico e conhecimento do domínio. Reynolds e Ali (2008) sugerem que qualquer conhecimento cultural pode ser expresso como alguma combinação desses cinco tipos de conhecimento.

O conhecimento situacional armazena os melhores indivíduos da geração anterior no espaço de crenças a ser usado depois para influenciar a próxima geração. Segundo Iacoban *et al.* (2003), ele contém um conjunto de indivíduos da população que serve como exemplo para o resto da população. A quantidade de exemplos pode variar de implementação para implementação, mas costuma ser pequena. O uso do conhecimento situacional faz o algoritmo convergir mais rapidamente (Keramati *et al.*, 2011).

O conhecimento histórico (ou temporal) guarda as últimas k mudanças significativamente importantes na direção da busca como forma de evitar mínimos locais. Esse conhecimento foi motivado pela necessidade de desenvolver aprendizado em ambientes dinâmicos.

O conhecimento normativo representa um conjunto de intervalos que caracterizam os valores assumidos pelas variáveis que compõem as melhores soluções. Esses intervalos servem, geralmente, para guiar os ajustes (mutações) que ocorrem nos indivíduos.

O conhecimento topográfico armazena as diferentes regiões promissoras dentro do espaço de busca e faz com que novos indivíduos as explorem. Este conhecimento foi proposto com o intuito de extrair padrões de comportamento do espaço de busca.

O conhecimento do domínio, como o próprio nome pressupõe, é específico de cada aplicação. Ele representa conhecimento sobre o domínio (conceitos, regras e princípios) do problema para guiar a busca. Esse é o tipo

de conhecimento menos utilizado, pois é o mais difícil de ser extraído e representado.

Os AC têm sido usados com sucesso em aplicações de problemas de otimização global (Sun *et al.*, 2012) e irrestrita (Alami *et al.*, 2007), problemas elétricos (Coelho *et al.*, 2009; Khodabakhshian e Hemmati, 2013), problemas de escalonamento (Soza *et al*., 2011; Reynolds e Ali, 2008), problemas de qualidade (Ma e Zhang, 2013), entre outros.

3.1.5 *BELIEF-BASED INCREMENTAL LEARNING* (*BBIL*)

A literatura apresenta alguns trabalhos comparando algoritmos de *EDA* e de AG para seleção de atributos. Inza e Larrañaga (2001) apresentaram experimentos com o *EDA-EBNA* (*Estimation of Bayesian Network Algorithm*) e relataram que o método encontrou subconjuntos que resultaram em acurácias similares ao AG, mas com a vantagem de necessitar menos gerações para convergir. Cantú-Paz (2002) compara três algoritmos de *EDA* (*Compact Genetic Algorithms - CGA*, *Extended CGA* e *Bayesian Optimization Algorithm - BOA*) com o AG e conclui que o *EDA* tem menor dificuldade em lidar com o problema do *linkage*, isto é, em agrupar adequadamente os subconjuntos de genes dependentes entre si. O *linkage* é um importante obstáculo à aplicação dos AG em problemas com relacionamentos desconhecidos entre variáveis, como é o caso da seleção de atributos (Cantú-Paz, 2002).

A principal diferença entre o *EDA* e o AG está na forma como “evoluem” suas populações. Conforme já mostrado, ao invés de utilizar operadores de cruzamento e mutação, o *EDA* utiliza um modelo probabilístico dos indivíduos que sobreviveram à seleção para gerar novos indivíduos. A geração de indivíduos com base neste modelo probabilístico parece ser a razão pela qual o *EDA* preserva de forma mais consistente os relacionamentos existentes entre os genes. Espera-se que o tratamento adequado às dependências entre atributos faça com que a população do algoritmo proposto baseado em *EDA-PBIL* evolua de forma a considerar melhor as possíveis situações (combinações entre valores de atributos) do mundo real.

Por outro lado, a introdução de um componente cultural baseado no conhecimento situacional dos AC, implementado para controlar o tamanho do passo de cada iteração na busca da melhor solução do *PBIL*, deve fazer com que o custo computacional aumente em relação ao *PBIL* padrão ao se utilizar passos menores em várias iterações. Espera-se que a incorporação deste componente cultural ao *PBIL* dê maior robustez ao seu processo de convergência, sem aumentar demasiadamente o custo computacional ao ser comparado ao *PBIL* padrão.

O acréscimo de um componente cultural ao *PBIL* traz a vantagem de armazenar os melhores indivíduos da geração anterior no espaço de crenças a ser usado depois para influenciar a próxima geração, pois “servem de exemplo” para o resto da população. O componente cultural acoplado ao *PBIL* funciona como uma taxa de aprendizagem adaptativa que busca ajustar o tamanho do passo do algoritmo a cada iteração. A taxa de rejeição diminui o passo, caso o *PBIL* sofra uma queda de desempenho e, aumenta o passo, caso contrário.

Assim, o presente trabalho propõe o acoplamento de um componente cultural de AC a um seletor de atributos *wrapper* baseado em *PBIL* visando a melhoria do desempenho da tarefa de classificação em comparação com *wrappers* baseados em AG e ao *PBIL* padrão. Como o espaço populacional de AC do algoritmo proposto é formado pelo *PBIL* e seu espaço de crenças armazena o conhecimento situacional com base na crença (em inglês, *belief*) dos outros indivíduos, o algoritmo proposto foi batizado de *Belief-Based Incremental Learning* (*BBIL*).

No *BBIL*, a melhor solução provisória (indivíduo mais apto) é armazenada e fica influenciando os novos indivíduos com uma probabilidade extra (parametrizável) no vetor de probabilidades. Metaforicamente, o conhecimento situacional (sabedoria) do indivíduo mais apto da história está sendo transmitida para as próximas gerações.

Sempre que o *fitness* da população piora, ela aumenta a rejeição ao conhecimento situacional do mais sábio (melhor solução obtida até o momento). Em outras palavras, a influência do indivíduo “mais sábio” vai diminuindo a cada geração até o surgimento de um novo mais sábio. Quando um novo indivíduo mais sábio é identificado, a nova geração aceita esta nova

sabedoria e dá-se início a um novo ciclo de passagem da sabedoria para as novas gerações.

Essencialmente, o *BBIL* difere do *PBIL* em dois pontos: (1) a inclusão de um novo parâmetro $\beta \in (0,1]$, chamado taxa de rejeição, representando a rejeição da população à sabedoria do indivíduo mais apto e, (2) uma nova expressão (3.4) para atualização da taxa de aprendizagem (inicial) α a cada geração sem um novo indivíduo "mais sábio", tal que

$$\alpha_l = \alpha_{l-1} - \alpha_{l-1} * \beta \tag{3.4}$$

Configurando-se adequadamente o parâmetro de rejeição, vislumbra-se uma convergência mais robusta para o *BBIL* do que para o *PBIL* padrão. A nova variante do *PBIL* proposta também pode ser aplicada para outros problemas de otimização, em especial, a qualquer problema em que se aplique o *PBIL*.

Para uma explicação didática do *BBIL* aplicado à seleção de atributos a seção 3.1.5.1 apresenta um exemplo simples de seu funcionamento.

3.1.5.1 EXEMPLO DO *BBIL* PARA SELEÇÃO DE ATRIBUTOS

Seja x um conjunto de dados com $p = 4$ atributos, $p_1(x) = [0{,}5\ 0{,}5\ 0{,}5\ 0{,}5]$ o vetor de probabilidades inicial e os parâmetros $\alpha = 0{,}1$ e $\beta = 0{,}2$. O vetor de probabilidades $p_1(x)$ indica a probabilidade de se obter o valor "1" em cada um dos $p = 4$ atributos da população. A população inicial D_1, gerada aleatoriamente de acordo com o vetor de probabilidades $p_1(x)$, é composta pelos seguintes indivíduos em (3.5):

$$D_1 = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} \tag{3.5}$$

O *fitness* de cada indivíduo é apresentado em (3.6):

$$fitness(D_1) = \begin{bmatrix} 0{,}52 \\ 0{,}70 \\ 0{,}85 \\ 0{,}34 \end{bmatrix} \tag{3.6}$$

Portanto, o melhor indivíduo da população na primeira geração é o D_1^3, isto é, o terceiro indivíduo (linha) da matriz D_1. Após a seleção do melhor indivíduo, o próximo passo é atualizar o vetor de probabilidades com base no indivíduo selecionado aplicando-se a função apresentada em (3.3). Com $\alpha = 0{,}1$, o novo vetor de probabilidades será $p_2(x) = [0{,}45\ 0{,}45\ 0{,}55\ 0{,}55]$.

A nova população D_2, gerada com base no novo vetor de probabilidades $p_2(x)$, é apresentada em (3.7):

$$D_2 = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{bmatrix} \tag{3.7}$$

e seu respectivo *fitness* em (3.8):

$$fitness(D_2) = \begin{bmatrix} 0{,}65 \\ 0{,}71 \\ 0{,}59 \\ 0{,}91 \end{bmatrix}. \tag{3.8}$$

O melhor indivíduo da segunda geração é o D_2^4. Como foi encontrado um indivíduo melhor que a melhor solução anterior, continua-se atualizando o vetor de probabilidades usando somente a taxa de aprendizagem $\alpha = 0{,}1$, sem considerar a taxa de rejeição β. O novo vetor de probabilidades será $p_3(x) = [0{,}495\ 0{,}405\ 0{,}605\ 0{,}605]$.

A nova população D_3, gerada com base em $p_3(x)$, é apresentada em (3.9):

$$D_3 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \tag{3.9}$$

e seu respectivo *fitness* em (3.10):

$$fitness(D_3) = \begin{bmatrix} 0{,}89 \\ 0{,}70 \\ 0{,}77 \\ 0{,}85 \end{bmatrix}. \tag{3.10}$$

O melhor indivíduo da terceira geração é o D_3^1. Como desta vez não foi encontrado um indivíduo com maior *fitness* que a melhor solução até o momento, atualiza-se a taxa de aprendizagem α antes do vetor de probabilidades. Como a nova população não melhorou com o último vetor de probabilidades, o algoritmo mantém o último vetor de probabilidades, porém seus indivíduos diminuem sua aceitação sobre ele.

O parâmetro $\alpha = 0{,}1$ é atualizado com base na taxa de rejeição $\beta = 0{,}2$. Aplicando a equação (3.4), a nova taxa de aprendizagem passa a ser $\alpha = 0{,}08$ e o vetor de probabilidades $p_4(x) = [0{,}5346\ 0{,}3726\ 0{,}6534\ 0{,}6534]$. Observando o vetor de probabilidades, nota-se uma tendência para a escolha de todos os atributos, com exceção do segundo.

Esta tendência se deve ao fato de que, quanto mais cada componente do vetor $p_l(x)$ se aproxima de “1”, maior a probabilidade de seleção de seu respectivo atributo e, quanto mais se aproxima de “0”, maior a probabilidade de rejeitá-lo. Deste modo, a cada geração, a probabilidade de cada componente de $p_l(x)$ vai convergindo para “0” ou “1”. O processo termina quando um critério de parada é satisfeito.

Uma vez que o vetor de probabilidades é usado para gerar a população de cada geração e, este vetor é atualizado pela taxa de aprendizagem, esta taxa afeta a escolha das porções do espaço de soluções que serão exploradas. Assim, a configuração da taxa de aprendizagem tem um impacto direto no conflito entre *exploration* e *exploitation*. Neste contexto, *exploration* é a habilidade do algoritmo em realizar buscas em amplitude no espaço de busca, enquanto *exploitation* refere-se à habilidade do algoritmo no uso da informação obtida sobre o espaço para aprofundar-se em buscas futuras (Baluja, 1994).

Espera-se que o auto ajuste da taxa de aprendizagem durante o treinamento proporcionado pelo *BBIL* possa também aumentar a eficiência do algoritmo em termos de *exploration* e *exploitation*. Aparentemente, uma alternativa ainda mais simples e intuitiva que o *BBIL* seria simplesmente diminuir a taxa de aprendizagem do *PBIL* padrão, reduzindo o tamanho de seu passo. De fato, isto tornaria o *PBIL* mais “minucioso” durante a varredura de uma forma bastante simples. Porém, o simples aumento da quantidade de passos pode trazer como consequência, por exemplo, uma elevação indesejada do número de avaliações de *fitness*.

O raciocínio é simples: Taxas de aprendizagem menores geralmente desaceleram a convergência do algoritmo. Esta desaceleração introduz maior diversidade na população e, com uma busca mais diversificada, aumentam-se as chances de se encontrar boas soluções de uma forma mais “ampla” sobretudo nas primeiras gerações. Em contrapartida, à medida que se aumenta a capacidade de *exploration*, a convergência vai se tornando cada vez mais lenta e diminui-se a capacidade de *exploitation* das informações obtidas nas buscas anteriores.

Em contrapartida, taxas de aprendizagem maiores aceleram a convergência do algoritmo e, esta aceleração concentra a população em uma porção mais delimitada do espaço de soluções. Ao se conhecer melhor esta pequena porção, aumentam-se as chances de se “aprofundar” em boas soluções dentro da mesma. À medida que se aumenta a capacidade de *exploitation*, a convergência vai se tornando cada vez mais rápida, o que pode levar a uma convergência prematura (principalmente, no caso de uma taxa de aprendizagem fixa) e diminui-se a capacidade de se explorar grandes porções do espaço de soluções.

3.2 *GEOMETRIC DATA ANALYSIS* (*GDA*)

Segundo Hair *et al.* (2009), Estatística Multivariada é o ramo da Estatística que estuda técnicas que permitem a análise simultânea de duas ou mais variáveis (em um sentido amplo, cobrindo variáveis numéricas e variáveis nominais).

A *GDA*[7] pode ser definida como a subárea da Estatística Multivariada que representa conjuntos de dados como nuvens de pontos e baseia a interpretação dos dados nestas nuvens (Le Roux e Rouanet, 2005). As nuvens de pontos tratam-se de objetos geométricos em espaços Euclidianos multidimensionais construídos a partir de tabelas de dados com base em estruturas matemáticas de Álgebra Linear.

Detalhes sobre a geometria das nuvens de pontos (ponto médio da nuvem, distância entre pontos, variância da nuvem, subnuvens e eixos de uma nuvem) podem ser obtidos em Le Roux e Rouanet (2010).

A *GDA* pode ser dividida em três categorias: *PCA*, *Correspondence Analysis* (*CA*) e *MCA*. As seções 3.2.1, 3.2.2 e 3.2.3 explicam conceitualmente e através de exemplos o uso de cada uma destas três categorias, respectivamente.

3.2.1 ANÁLISE DE COMPONENTES PRINCIPAIS

Embora as origens de técnicas estatísticas sejam frequentemente difíceis de serem traçadas, é geralmente aceito que as primeiras descrições da técnica hoje conhecida como Análise de Componentes Principais (em inglês, *Principal Component Analysis* ou, simplesmente, *PCA*) tenham sido dadas por Pearson (1901) *apud* Jolliffe (2002) e Hotelling (1933) *apud* Jolliffe (2002).

O principal propósito do *PCA* é reduzir a dimensionalidade de um conjunto de dados formado por variáveis interrelacionadas, conservando ao máximo a variação presente nos dados. Isto é alcançado através de uma transformação para um novo conjunto de variáveis não correlacionadas, denominadas “componentes principais” (em inglês, *principal component*, abreviado como *PC*), ordenadas das que retêm a maior parte para as que retêm a menor parte da variação presente em todas as variáveis originais. O cálculo dos *PC* se resume à resolução do problema de autovalores e autovetores de uma matriz simétrica definida positiva (Jolliffe, 2002).

---

[7] O termo *Geometric Data Analysis* foi sugerido pela primeira vez em 1996 por Patrick Suppes (*Stanford University*)

Usualmente, seleciona-se um subconjunto dos *PCs* (pela ordem) de modo a melhor representar o conjunto de dados original, porém com menos dimensões do que o mesmo. A técnica é particularmente útil se o número de *PCs* selecionados for consideravelmente menor que o número de variáveis originais (alta redução de dimensionalidade).

Embora a aplicação do *PCA* como técnica de redução de dimensionalidade seja provavelmente sua aplicação mais prevalente, ele possui muitas outras utilidades e formas de aplicação. O presente trabalho apresenta o *PCA* como ferramenta de *GDA* para tratamento de dados com a finalidade de melhorar (e viabilizar) a classificação de padrões. O *PCA* neste caso funciona mais como uma ferramenta da etapa de transformação do *KDD* do que como uma técnica de redução de dimensionalidade por si só (embora o problema, na maioria das vezes, tenha suas dimensões reduzidas ao final do processo).

O *PCA*, bem como o *MCA* (a outra técnica de *GDA* abordada neste trabalho, e detalhada na seção 3.2.3), evitam problemas associados a multilcolinearidade (variáveis altamente correlacionadas umas com as outras) entre os atributos de predição. Os *PCs* resultantes da aplicação do método, por não serem correlacionados entre si, passam a ser usados como preditores no lugar dos atributos originais. Além disso, espera-se que o uso destes novos atributos aumente o desempenho dos classificadores.

3.2.1.1 DEFINIÇÃO DE *PRINCIPAL COMPONENT* (*PC*)

Suponha que x seja um vetor de p variáveis aleatórias, e que as variâncias das p variáveis aleatórias e a estrutura das covariâncias ou correlações entre as p variáveis sejam significativas.

Ao invés de observar as p variâncias e todas as $\frac{1}{2}p(p-1)$ correlações ou covariâncias, o *PCA* observa as “novas” variáveis que preservam a maior parte da informação fornecida pelas variâncias e correlações ou covariâncias. Apesar do *PCA* não ignorar as covariâncias e correlações, ele dá prioridade às variâncias.

O primeiro passo é observar a função linear $z_1 = \alpha_1^T x$ dos elementos de x que tenham variância máxima, conforme a equação (3.11):

$$z_1 = \alpha_1^T x = \alpha_{11}x_1 + \alpha_{12}x_2 + \cdots + \alpha_{1p}x_p = \sum_{j=1}^{p} \alpha_{1j}x_j \qquad (3.11)$$

onde α_1 é um vetor de p constantes: $\alpha_{11}, \alpha_{12}, \ldots, \alpha_{1p}$.

Em seguida, observe a função linear $z_2 = \alpha_2^T x$, não correlacionada com $z_1 = \alpha_1^T x$, que tenha variância máxima, e assim por diante, até que no *k*-ésimo estágio a função linear $z_k = \alpha_k^T x$ seja encontrada com a variância máxima e não correlacionada com $z_1 = \alpha_1^T x, z_2 = \alpha_2^T x, \ldots, z_{k-1} = \alpha_{k-1}^T x$. Desta forma, a k-ésima "nova" variável, $z_k = \alpha_k^T x$, é definida como o *k*-ésimo *PC*.

É possível encontrar até p *PCs*, mas se espera que a maior parte da variação em x possa ser explicada por m *PCs*, onde $m \leq p$. A Figura 3.8 ilustra a redução de complexidade alcançada pela transformação das variáveis originais em *PCs* por meio de um caso simples (onde $p = 2$), mas possível de ser visualizado num plano bidimensional.

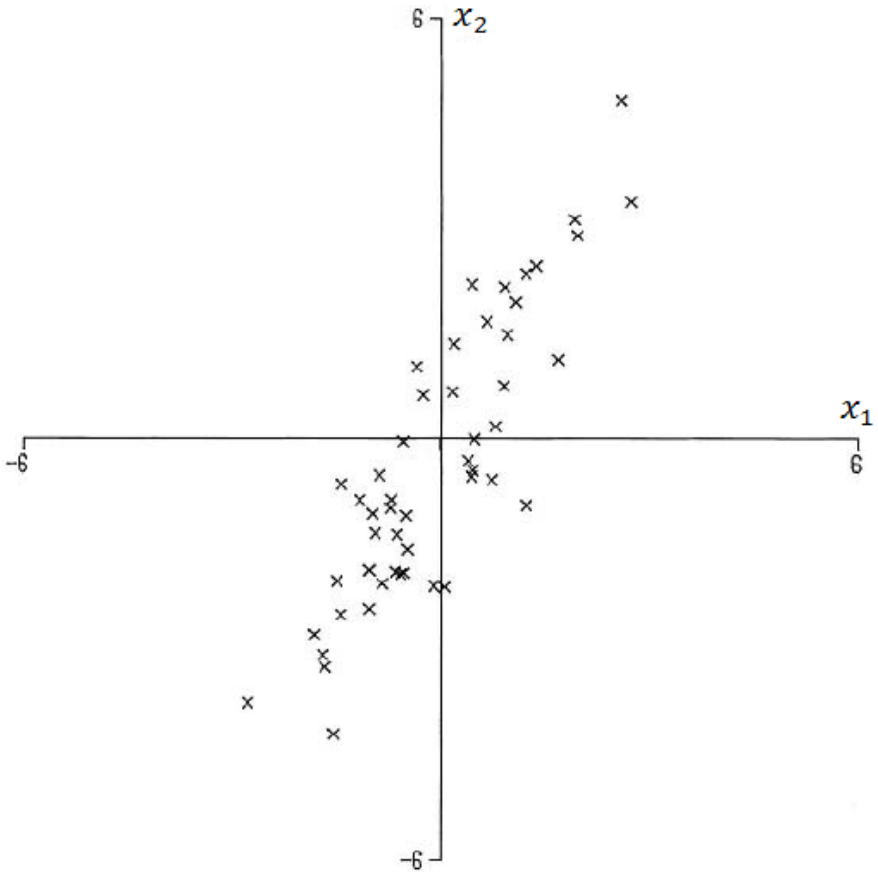

Figura 3.8 – Dados x_1 e x_2 , variáveis correlacionadas.

A Figura 3.8 mostra os dados correlacionados das variáveis x_1 e x_2. Há uma considerável variação para as duas variáveis, embora uma variação um pouco maior na direção de x_2 do que de x_1.

Após a transformação das variáveis originais x_1 e x_2 nos *PCs* z_1 e z_2, obtém-se o gráfico da Figura 3.9. Nota-se neste gráfico uma grande variação na direção de z_1, bem maior que a pequena variação na direção de z_2. Além disso, observa-se que os *PCs* são não correlacionados, pois as "novas" variáveis são ortogonais entre si.

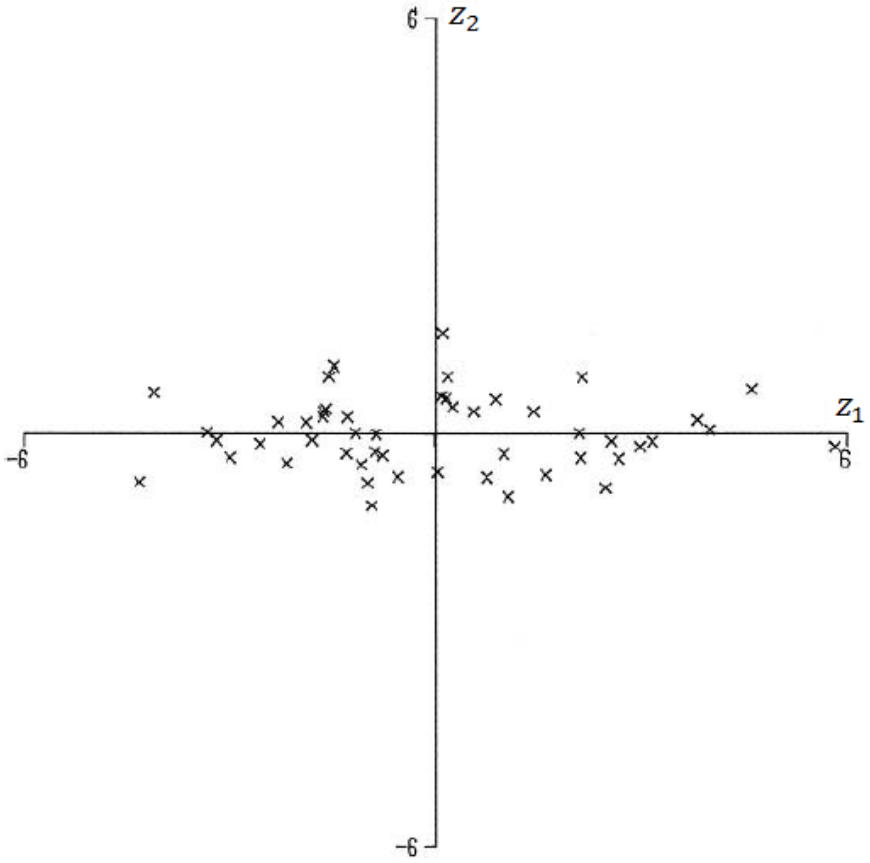

Figura 3.9 – *PCs* z_1 e z_2 , variáveis correlacionadas.

Genericamente, se um conjunto de p variáveis tiverem correlações substanciais entre si, então os primeiros *PCs* representam a maior parte da variação das variáveis originais. Por outro lado, os últimos *PCs* identificam direções com pouca variação, isto é, relações quase constantes entre as variáveis originais.

3.2.1.2 DETERMINAÇÃO DOS *PCs*

Seja C a matriz de covariância conhecida do vetor das variáveis aleatórias x, onde seus elementos $c_{i,j}$ são as covariâncias entre o *i*-ésimo e o *j*-ésimo elementos de x para $i \neq j$ e a variância do *j*-ésimo elemento de x para $i = j$.

Assim, para $k = 1,2, \ldots, p$, o *k*-ésimo *PC* é dado por $z_k = \alpha_k^T x$ onde α_k é um autovetor da matriz de covariância C correspondente ao *k*-ésimo maior autovalor λ_k. Além disso, se α_k for definido como um vetor unitário $(\alpha_k^T . \alpha_k = 1)$, então var($z_k$) = λ_k, ou seja, a variância de z_k será o próprio autovalor λ_k.

3.2.1.3 SELEÇÃO DOS *PCs* (GRÁFICO "*SCREE*")

Desenvolvido por Cattell (1966) *apud* Ledesma e Mora (2007), o gráfico "*Scree*" é uma das formas de se determinar um subconjunto de *PCs* que represente adequadamente a variação total dos dados originais. A técnica consiste em observar um gráfico das variâncias de todos os *PCs* obtidos e decidir qual o valor de m para o qual as encostas das linhas que conectam os pontos são "íngremes" à esquerda de m, e "não íngremes" à direita de m.

Conforme ilustrado na Figura 3.10, este valor de m, que define um "cotovelo" no gráfico, é então tomado como sendo o número de *PCs* a serem mantidos. Seu nome (*scree*, em português "talude") deriva da semelhança da forma típica deste gráfico com o acúmulo de cascalho ao pé da encosta de uma montanha. Embora antiga e subjetiva, esta técnica é ainda bastante utilizada por ser empírica, intuitivamente plausível e funcionar bem na prática. Por estes motivos, o presente trabalho utiliza o gráfico *scree* para escolha do subconjunto de *PCs* obtidos.

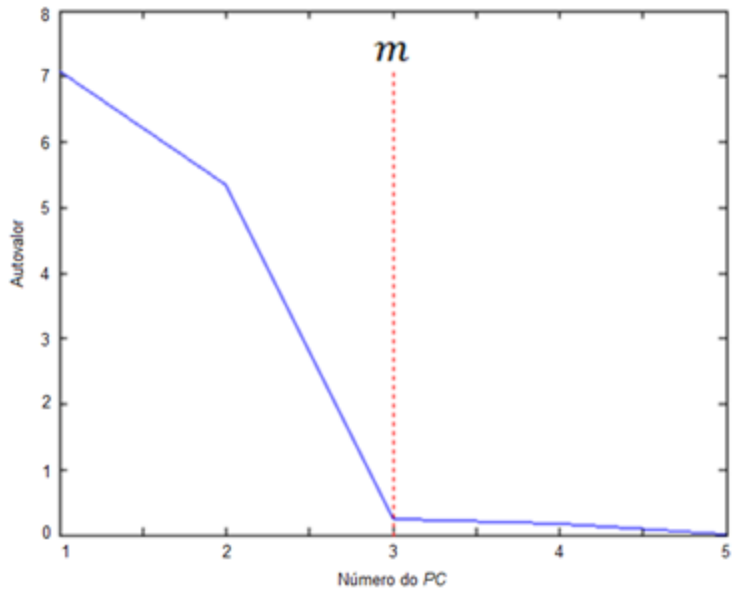

Figura 3.10 – Exemplo de gráfico *scree* com $k = 5$ e $m = 3$.

3.2.2 *CORRESPONDENCE ANALYSIS* (*CA*)

O *PCA* é notoriamente uma das mais tradicionais e eficientes técnicas para transformação de dados do processo *KDD* (Jolliffe, 2002). Por este motivo, não é raro que analistas de dados acabem por utilizá-lo de forma indiscriminada. Embora uma das técnicas mais eficientes, o *PCA* pode não ser adequado e não apresentar resultados satisfatórios em toda e qualquer situação.

Uma alternativa ao *PCA* para lidar com determinados tipos de dados é o *CA*, técnica atribuída ao matemático francês Jean-Paul Benzécri, originalmente desenvolvida para analisar tabelas de contingência (Benzécri, 1973).

O objetivo do *CA* é transformar uma tabela de dados em dois conjuntos de *factor scores*: um para as linhas e um para as colunas. Os *factor scores* visam obter a melhor representação da estrutura de semelhança das linhas e colunas da tabela. Além disso, os *factor scores* podem ser representados como mapas, que exibem a informação essencial da tabela original. Nestes mapas, linhas e colunas são exibidas como pontos cujas coordenadas são os *factor scores* e onde as dimensões são chamadas fatores. Curiosamente, os *factor scores* das linhas e colunas têm a mesma variância e, consequentemente,

ambas podem ser convenientemente representadas em um único mapa (Benzécri, 1992).

Como técnica, o *CA* foi descoberto (e redescoberto) muitas vezes e, por este motivo, variações do mesmo podem ser encontradas com vários nomes diferentes, tais como "*dual-scaling*", "*optimal scaling*" e "*reciprocal averaging*". As múltiplas identidades do *CA* são uma consequência de seu grande número de aplicações, podendo servir de solução para uma série de problemas aparentemente diferentes (Abdi e Williams, 2010b).

3.2.3 *MULTIPLE CORRESPONDENCE ANALYSIS* (*MCA*)

O *CA* opera sobre uma tabela de contingência uma análise de duas variáveis, uma para as linhas e outra para as colunas. O *MCA* é a extensão do *CA* que permite a análise dos padrões de relacionamento entre mais de duas variáveis.

Desde a concepção do *CA* por Benzécri (1973) e de sua extensão, o *MCA*, poucos trabalhos relacionaram estas técnicas ao *KDD*. Com exceção de alguns trabalhos recentes de autores franceses (Saporta e Niang, 2006; Bougeard *et al.*, 2011), país de origem de Benzécri, poucos pesquisadores têm dado a devida atenção ao *MCA* como abordagem de transformação de dados no contexto do *KDD*.

Assim, o presente trabalho também visa "lançar luz" sobre a aplicação da transformação de dados por *MCA* e estimular o interesse no tema, uma vez que, sob o ponto de vista do autor, a técnica não tem recebido a merecida atenção neste contexto. Para ilustrar o funcionamento do *MCA* de maneira didática, o presente trabalho o faz através de um exemplo numérico (Abdi e Williams, 2010b).

3.2.3.1 EXEMPLO DE APLICAÇÃO DO *MCA*

A Tabela 3.1 apresenta a frequência com que alguns escritores franceses utilizam três sinais de pontuação em suas obras: o ponto final (“.”), a

vírgula (“,”) e todos os outros pontos (ponto de interrogação, ponto de exclamação, dois pontos e ponto-e-vírgula).

Tabela 3.1 – Número de vezes que cada escritor usa cada sinal.

Escritor	Ponto final	Vírgula	Outros
Rousseau	7.836	13.112	6.026
Chateaubriand	53.655	102.383	42.413
Hugo	115.615	184.541	59.226
Zola	161.926	340.479	62.754
Proust	38.177	105.101	12.670
Giraudoux	46.371	58.367	14.299

A Tabela 3.1 é apresentada na forma matricial em (3.12):

$$X = \begin{bmatrix} 7.836 & 13.112 & 6.026 \\ 53.655 & 102.383 & 42.413 \\ 115.615 & 184.541 & 59.226 \\ 161.926 & 340.479 & 62.754 \\ 38.177 & 105.101 & 12.670 \\ 46.371 & 58.367 & 14.299 \end{bmatrix}. \tag{3.12}$$

A matriz é denotada por X e possui $i = 6$ linhas e $j = 3$ colunas. De modo análogo à Tabela 3.1, na matriz X, as linhas representam os autores e as colunas os tipos de sinais de pontuação. Na interseção entre a linha i e a coluna j, encontra-se o número de vezes que o *i*-ésimo autor usou o *j*-ésimo sinal de pontuação em uma de suas obras.

3.2.3.2 A PRIMEIRA (MÁ) IDEIA: TRANSFORMAÇÃO POR *PCA*

Uma transformação sobre estes dados poderia revelar uma nuvem de pontos (mapa) com as semelhanças e diferenças no estilo de pontuação entre os autores. Neste mapa, os autores seriam representados por pontos e as distâncias entre os pontos expressariam o grau de similaridade de estilo entre os autores. Assim, dois pontos próximos um ao outro indicariam que os autores

correspondentes usam a pontuação de modo similar e dois pontos distantes entre si, o oposto.

A primeira ideia para obter este mapa seria, ingenuamente, aplicar o tradicional *PCA* sobre X. O resultado desta aplicação é apresentado na Figura 3.11.

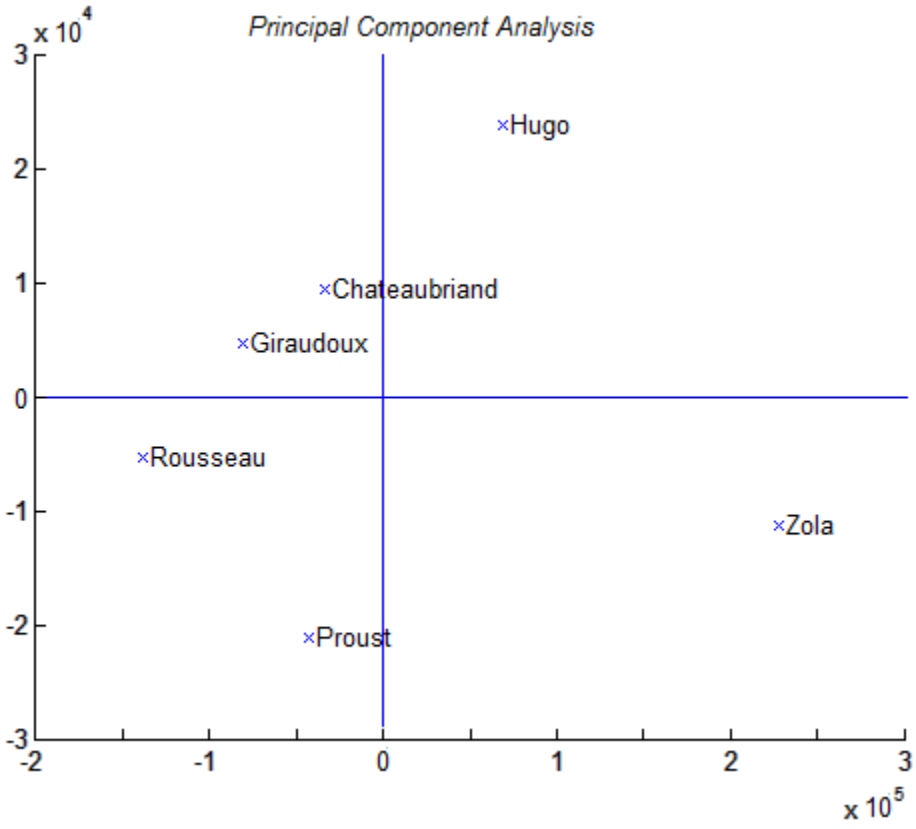

Figura 3.11 – *PCA* da matriz X.

Suponha agora que, sem o conhecimento da maioria dos estudiosos de literatura francesa, o escritor Zola escrevesse um pequeno romance sob o pseudônimo de Aloz. Neste romance, ele manteve sua maneira usual de usar os sinais de pontuação, mas por ser um romance pequeno, ele obviamente produziu um número menor de sinais de pontuação do que em suas outras obras. A matriz (3.13) registra o número de ocorrências de cada sinal de pontuação para Aloz:

$$[2.699 \quad 5.675 \quad 1.046]. \tag{3.13}$$

Para facilitar a visualização e a comparação, o vetor de Zola é reproduzido em (3.14):

$$[161.926 \quad 340.479 \quad 62.754]. \tag{3.14}$$

Então Aloz e Zola têm o mesmo estilo de pontuação, mas diferem somente na sua prolixidade. Uma boa análise deveria revelar similaridade de estilo, contudo como mostra a Figura 3.12, o *PCA* falha ao apresentar o grau de similaridade entre Aloz e Zola.

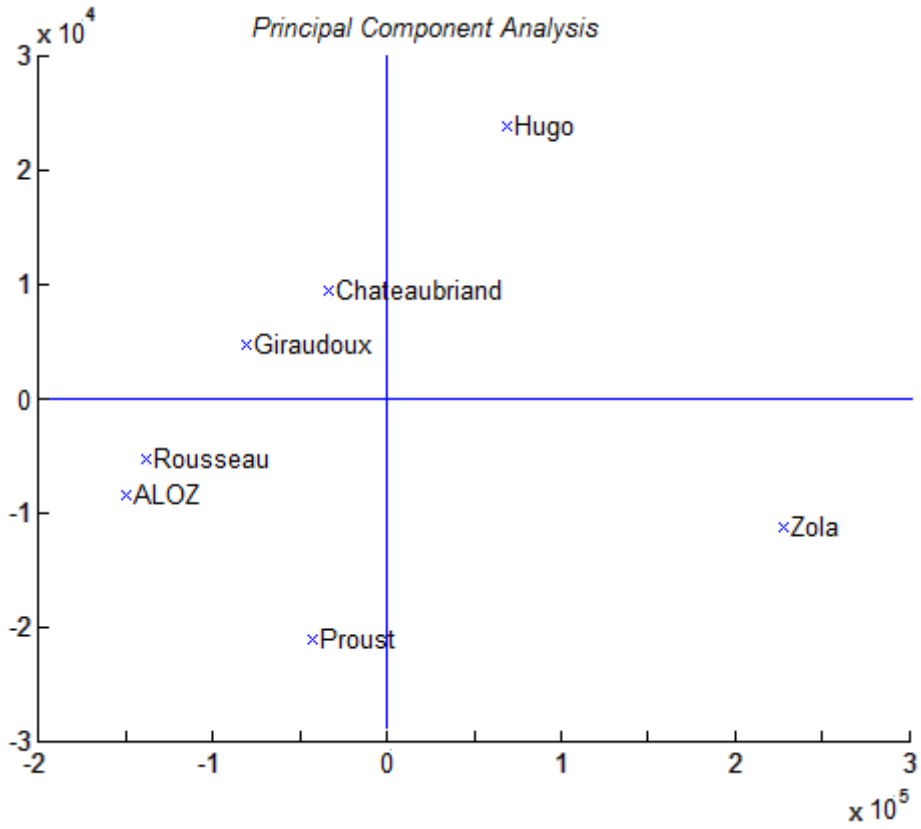

Figura 3.12 – *PCA* da matriz X após a inclusão do "autor" Aloz.

Nesta figura 3.12, Aloz é projetado mais distante de Zola do que de qualquer outro autor. Este exemplo ilustra como a transformação por *PCA* pode nem sempre ser a melhor alternativa. O *PCA* pode ter sido mais "sensível", por exemplo, ao número de sinais de pontuação do que na maneira como foram utilizados em si.

Ao invés de simplesmente levar em consideração o número absoluto de uso dos sinais de pontuação por cada autor, o algoritmo deveria analisar o "estilo" de cada um, expresso por frequências relativas desses usos. Isto sugere que a matriz de dados original deva ser transformada de modo que o resultado descreva a proporção do uso dos sinais de pontuação, e não o número de vezes que cada sinal é utilizado.

3.2.3.3 TRANSFORMAÇÃO POR *MCA*

A matriz resultante da transformação por *MCA*, chamada matriz de perfil das linhas e, denotada por R é obtida dividindo-se cada linha pela sua soma, conforme apresentado em (3.15):

$$R = diag\left\{X \begin{matrix} 1 \\ j \times 1 \end{matrix}\right\}^{-1} X = \begin{bmatrix} 0{,}2905 & 0{,}4861 & 0{,}2234 \\ 0{,}2704 & 0{,}5159 & 0{,}2137 \\ 0{,}3217 & 0{,}5135 & 0{,}1648 \\ 0{,}2865 & 0{,}6024 & 0{,}1110 \\ 0{,}2448 & 0{,}6739 & 0{,}0812 \\ 0{,}3896 & 0{,}4903 & 0{,}1201 \end{bmatrix}, \quad (3.15)$$

onde o operador $diag$ transforma um vetor em uma matriz diagonal com os elementos do vetor sobre a diagonal, e $\begin{matrix} 1 \\ j \times 1 \end{matrix}$ é um vetor de tamanho j formado apenas por números "1".

O "escritor médio" seria alguém que usa cada sinal de pontuação de acordo com a proporção na amostra. O perfil deste escritor médio seria o baricentro (também chamado de centroide, centro de massa, ou centro de gravidade) da matriz. Aqui, o baricentro de R é um vetor com $j = 3$ elementos, denotado por c, e obtido pela expressão (3.16):

$$c^T = \left(\begin{matrix} 1 \\ 1 \times i \end{matrix} \times X \times \begin{matrix} 1 \\ j \times 1 \end{matrix}\right)^{-1} \times \begin{matrix} 1 \\ 1 \times i \end{matrix} X = [0{,}2973 \quad 0{,}5642 \quad 0{,}1385]. \quad (3.16)$$

onde $\left({}_{1\times i}^{1}\times X\times {}_{j\times 1}^{1}\right)^{-1}$ é a matriz inversa do total de X e ${}_{1\times i}^{1}X$ é o total das colunas de X.

Se todos os autores pontuassem da mesma maneira, todos pontuariam como o escritor médio. Assim, para se estudar as diferenças entre autores, é necessário analisar a matriz de desvios do escritor médio. Esta matriz de desvios é denotada como Y e é obtida pela expressão (3.17):

$$Y = R - \left({}_{i\times 1}^{1}\times c^{T}\right) = \begin{bmatrix} -0{,}0068 & -0{,}0781 & +0{,}0849 \\ -0{,}0269 & -0{,}0483 & +0{,}0752 \\ +0{,}0244 & -0{,}0507 & +0{,}0263 \\ -0{,}0107 & +0{,}0507 & -0{,}0275 \\ -0{,}0525 & +0{,}1097 & -0{,}0573 \\ +0{,}0923 & -0{,}0739 & -0{,}0184 \end{bmatrix}. \quad (3.17)$$

No *MCA* atribui-se uma massa a cada linha e um peso a cada coluna. A massa de cada linha reflete sua importância na amostra. Em outras palavras, a massa de cada linha é a proporção desta linha no total da tabela. As massas das linhas são armazenadas em um vetor denotado por m, obtido pela expressão (3.18):

$$m = \left({}_{1\times i}^{1}\times X\times {}_{j\times 1}^{1}\right)^{-1} \times {}_{j\times 1}^{1}X = \quad (3.18)$$

$$m = [0{,}0189 \quad 0{,}1393 \quad 0{,}2522 \quad 0{,}3966 \quad 0{,}1094 \quad 0{,}0835]^{T}.$$

A partir do vetor m é obtida a matriz de massas M como apresentado na expressão (3.19):

$$M = diag\{m\}. \quad (3.19)$$

O peso de cada coluna reflete sua importância para a discriminação entre os autores. Assim, o peso de uma coluna reflete a informação que esta coluna fornece para a identificação de uma determinada linha. A ideia é que colunas usadas frequentemente não forneçam muita informação e, colunas que são usadas raramente forneçam muita informação. Uma forma de se medir a frequência de uso de uma coluna é contar a proporção de vezes que ela é usada, que é igual ao valor de sua componente coluna no baricentro. Além

disso, o peso de uma coluna é computado como a inversa da componente coluna do baricentro. O vetor peso das colunas, denotado por w, pode ser obtido conforme (3.20):

$$w = [w_j] = [c_j^{-1}] = \begin{bmatrix} \frac{1}{0,2973} \\ \frac{1}{0,5642} \\ \frac{1}{0,1385} \end{bmatrix} = \begin{bmatrix} 3,3641 \\ 1,7724 \\ 7,2190 \end{bmatrix}. \quad (3.20)$$

De modo análogo ao vetor de massas, a partir do vetor w é definida a matriz de pesos W como apresentado na expressão (3.21):

$$W = diag\{w\}. \quad (3.21).$$

A matriz de desvios Y é decomposta através da expressão (3.22) denominada *Generalized Singular Value Decomposition* (*GSVD*) respeitando as restrições impostas pelas matrizes M (massas das linhas) e W (pesos das colunas):

$$Y = P\Delta Q^T \text{ com } P^T M P = Q^T W Q = I. \quad (3.22)$$

A expressão (3.23) apresenta os valores obtidos decompondo-se a matriz Y do exemplo através da decomposição *GSVD*:

$$Y = P \times \Delta \times Q^T = \quad (3.23)$$

$$Y = \begin{bmatrix} +1,7962 & +0,9919 \\ +1,4198 & +1,4340 \\ +0,7739 & -0,3978 \\ -0,6878 & +0,0223 \\ -1,6801 & +0,8450 \\ +0,3561 & -2,6275 \end{bmatrix} \times \begin{bmatrix} 0,1335 & 0 \\ 0 & 0,0747 \end{bmatrix} \times \begin{bmatrix} +0,1090 & -0,4114 & +0,3024 \\ -0,4439 & +0,2769 & +0,1670 \end{bmatrix}.$$

As linhas da matriz X são agora representadas pelos seus *factor scores* que são projeções das observações sobre os vetores singulares. A linha de *factor scores* é armazenada em uma matriz F de dimensões $i \times l$ (no exemplo, $i = 3$ e $l = 2$), onde l é o número de valores sigulares diferentes de zero. A matriz F é obtida por (3.24):

$$F = P\Delta = \begin{bmatrix} +0{,}2398 & +0{,}0741 \\ +0{,}1895 & +0{,}1071 \\ +0{,}1033 & -0{,}0297 \\ -0{,}0918 & +0{,}0017 \\ -0{,}2243 & +0{,}0631 \\ +0{,}0475 & -0{,}1963 \end{bmatrix}. \quad (3.24)$$

A variância dos *factor scores* para uma dada dimensão é igual ao valor singular quadrático desta dimensão (a variância das observações é calculada levando-se em consideração suas massas). Ou de modo equivalente, diz-se que a variância dos *factor scores* é igual ao autovalor desta dimensão (o autovalor é o quadrado do valor singular). Isto pode ser verificado na expressão (3.25):

$$F^T M F = \Delta^2 = \begin{bmatrix} 0{,}1335^2 & 0 \\ 0 & 0{,}0747^2 \end{bmatrix} = \begin{bmatrix} 0{,}0178 & 0 \\ 0 & 0{,}0056 \end{bmatrix}. \quad (3.25)$$

A Figura 3.13 mostra os resultados da aplicação do *MCA* na forma de uma nuvem de *factor scores*, onde cada ponto representa uma linha da matriz X (no exemplo, um autor). Nesta figura, Zola e seu pseudônimo Aloz aparecem sobrepostos, indicando uma combinação perfeita entre ambos.

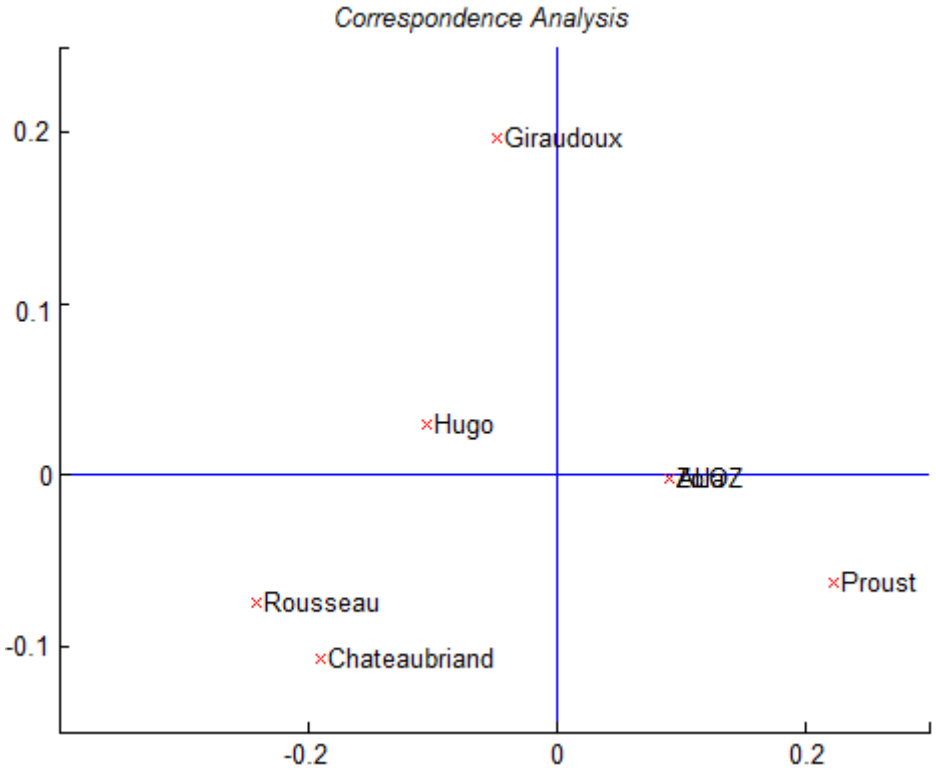

Figura 3.13 – *MCA* da matriz X.

3.2.4 TRANSFORMAÇÃO POR *MCA* SOBRE DADOS NOMINAIS

Conjuntos de dados nominais caracterizam-se por possuir valores mutuamente exclusivos para cada atributo (variável). O *MCA* utiliza estes valores, mas codificados através de uma forma particular de representação: a forma binária. Na forma binária cada elemento só pode assumir valor “0” ou valor “1”. A transformação de valores nominais binários consiste em se decompor cada uma das variáveis do conjunto de dados em múltiplos “níveis”, codificando cada um destes níveis como uma diferente variável binária.

Para exemplificar o processo, seja a variável nominal “gênero” dividida em dois níveis: “masculino” e “feminino”. Os padrões para cada um destes níveis poderiam então ser codificados, por exemplo, com “[0 1]” para “masculino” e “[1 0]” para “feminino”. Ao término do processo de “binarização”, todas as variáveis nominais originais terão sido decompostas em colunas binárias (uma para cada diferente valor existente para aquela variável) com uma e somente uma coluna assumindo o valor “1” por variável.

Dados codificados desta forma possuem maior propensão em retratar de forma mais fidedigna as diferenças entre os valores de cada atributo, ao considerar todos os valores equidistantes entre si.

A transformação de dados nominais binarizados por *MCA* sugere ganho de desempenho de classificação em relação à classificação “direta” dos dados brutos ou até mesmo quando comparado ao *PCA*. Presume-se isto devido à aparente maior habilidade do *MCA* em considerar as variações “ocultas” nos dados.

Apoiado nestes argumentos e em testes empíricos realizados com nove diferentes bases de dados nominais e dois dos mais tradicionais e difundidos classificadores da área de RP, o presente trabalho enaltece a adequação do *MCA* com dados nominais e o indica como técnica de transformação de dados visando a melhoria de desempenho de classificadores. Os classificadores utilizados são apresentados na seção 3.3.

3.3 CLASSIFICAÇÃO DE PADRÕES

O *KDD* é um processo interativo e iterativo, ou seja, o gestor observa o resultado, forma um novo conjunto de questões para refinar a busca em um dado aspecto das descobertas, realimenta o sistema com novos parâmetros para uma nova busca, e assim sucessivamente. Ao final do processo, um relatório das descobertas é gerado, que passa então a ser interpretado pelo gestor e o conhecimento é "descoberto" (Steiner *et al.*, 2007).

Segundo Witten e Frank (2005), mineração de dados é a fase do *KDD* em que ocorre a extração de relacionamentos e padrões implícitos, previamente desconhecidos e potencialmente úteis a partir de bases de dados. Esses relacionamentos e padrões representam conhecimento valioso sobre os dados e, consequentemente, sobre o domínio do mundo real que eles representam. Para Freitas (2002), o conhecimento a ser descoberto deve ser correto, compreensível e útil, e a estratégia de descoberta deve ser, por sua vez, eficiente, genérica e flexível.

Na fase da mineração de dados escolhe-se um algoritmo propício para cada aplicação (Witten e Frank, 2005). O termo propício refere-se, principalmente, ao tipo de "tarefa" que o algoritmo deve realizar (objetivo do processo). Usualmente, cada tarefa extrai um tipo de conhecimento diferente da base de dados e, portanto, utiliza algoritmos diferentes. Os tipos de tarefas realizadas pelos algoritmos de mineração de dados podem ser: classificação, agrupamento, associação e predição numérica[8]. De modo geral, o que estes algoritmos fazem é procurar por padrões presentes nos dados.

Uma das tarefas comumente resolvidas com técnicas de mineração de dados é a classificação. Ela pode ser definida como um processo de aprendizagem de máquina que visa compreender a estrutura subjacente de semelhanças entre as instâncias de uma mesma classe (instâncias já classificadas), visando determinar as classes de instâncias em que ainda se desconhecem as classes. Para Witten e Frank (2005) a classificação resume-

[8] Embora alguns autores considerem a predição numérica um tipo de classificação, o presente trabalho categoriza-a como uma tarefa da mineração de dados, conforme Witten e Frank (2005), referencial teórico adotado no presente trabalho.

se ao uso (por uma máquina) de instâncias já classificadas para se aprender um modo de classificar instâncias ainda não classificadas.

Neste contexto, considere uma base de dados relacional, onde cada coluna representa um domínio, chamado atributo, que possui um número finito de valores. Cada linha é chamada de registro, que é um conjunto de valores de atributos. Essa base de dados é chamada de base de treinamento se houver um atributo especial chamado classe. Do contrário, é chamada de base de teste. Os registros de ambas as bases são, em geral, mutuamente exclusivos.

A literatura apresenta uma grande variedade de abordagens de classificação cada uma das quais com diversos algoritmos (técnicas), tais como Árvores de Decisão, Regras de Decisão, *Kernel Machines*, RNA, Análise Multivariada e RB.

O desempenho de cada algoritmo depende de diversos fatores, tais como o modelo de representação do conhecimento adotado, a estrutura dos dados (tipos de variáveis, número de classes, entre outros), o comportamento dos dados (não linearidade, frequência de registros incompletos, entre outros) e características do próprio algoritmo (número de parâmetros, custo computacional, robustez, entre outros).

O presente trabalho analisa o desempenho de duas técnicas de classificação bastante conhecidas na literatura. A primeira delas, pertencente à abordagem de Análise Multivariada (apresentada na subseção 3.3.1) é a *Discriminant Function Analysis* (*DFA*). A segunda técnica trata-se de uma RB (apresentada na subseção 3.3.2) conhecida como *NB*.

3.3.1 ANÁLISE MULTIVARIADA

As técnicas de análise estatística multivariada são projetadas para a identificação de padrões em conjuntos de dados multivariados. As principais técnicas desta abordagem são as análises discriminantes (*DFA* e suas derivações) e as regressões (*PLSR*, RL e suas derivações). Interessantes análises comparativas entre estas técnicas podem ser encontradas em Kano *et al.* (2002) e Tiplica *et al.* (2001).

3.3.1.1 *DISCRIMINANT FUNCTION ANALYSIS* (*DFA*)

Os termos “discriminar” e “classificar” foram introduzidos na área de Estatística por Fisher no primeiro tratamento moderno dos problemas de separação de conjuntos. Desde o trabalho de Fisher, em 1936, numerosos trabalhos têm sido desenvolvidos aplicando técnicas de análise discriminante para a tarefa de classificação de padrões (Chamroukhi *et al.*, 2013; Steiner *et al.*, 2006).

A *DFA* foi inicialmente desenvolvida com o propósito de classificar instâncias de dados em um dentre dois conjuntos claramente definidos denominados classes. Alguns anos mais tarde, a *DFA* foi generalizada para problemas de classificação com qualquer número de classes, recebendo o nome de *Multiple Discriminant Analysis* (*MDA*).

Definição Formal

Dadas duas amostras A e B de observações multivariadas $x \in \Re^n$, deseja-se transformar estas observações multivariadas em observações univariadas Y’s, de tal modo que estejam separadas tanto quanto possível. O método cria os Y’s como combinações lineares dos x’s, ou seja, $Y = c^T x$, em que $c \in \Re^n$.

A melhor combinação é obtida da razão entre “o quadrado da distância entre as médias dos conjuntos A e B (x_A e x_B)” e “a variância de Y”. Neste contexto, a *DFA* é dada por (3.26).

$$Y = c^T x = (x_A - x_B)^T \sum^{-1} x \tag{3.26}$$

onde x é o vetor das variáveis aleatórias correspondentes às características amostrais observadas.

Assim, para a classificação de um novo padrão $x_0 \in \Re^n$, se $x_0 \in A$, então $y_0 = c^T x_0 \geq \frac{1}{2}(x_A - x_B)^T \sum^{-1}(x_A + x_B)$, ou se $x_0 \in B$, então $y_0 = c^T x_0 < \frac{1}{2}(x_A - x_B)^T \sum^{-1}(x_A + x_B)$.

Representação Geométrica

Supondo ainda a existência das duas amostras A e B e duas medidas, x_1 e x_2, para cada membro dos dois conjuntos de amostras (classes), a Figura 3.14 representa, através de um diagrama de dispersão, a associação da variável x_1 com a variável x_2 para cada membro dos dois conjuntos.

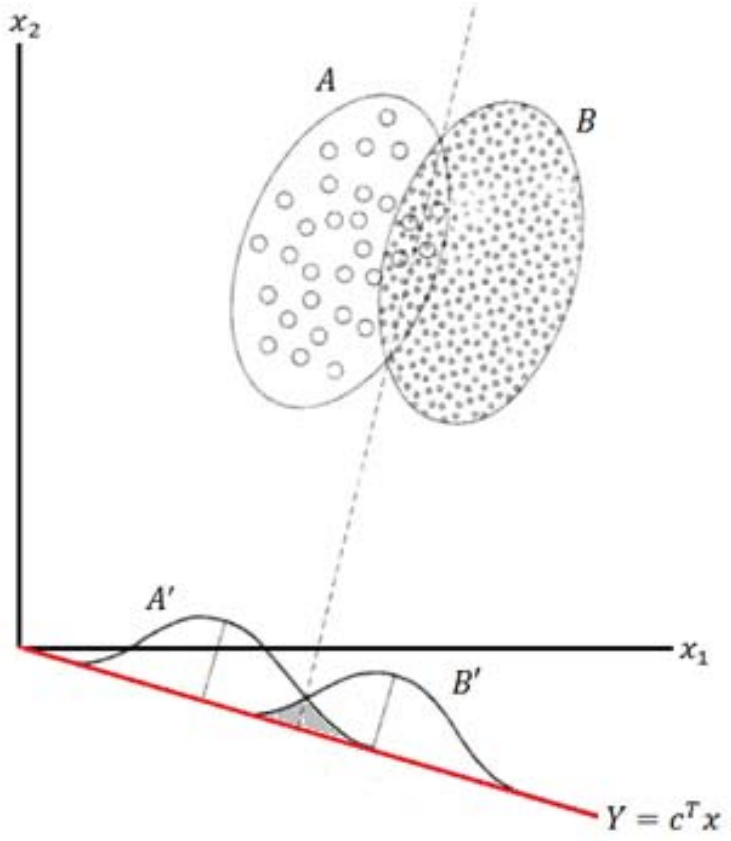

Figura 3.14 – Representação gráfica de um *DFA* (Hair *et al.*, 2009).

Os pontos (círculos) grandes representam as medidas das variáveis para os membros do conjunto A e os pequenos correspondem ao grupo B. As elipses desenhadas em torno dos pontos pequenos e grandes envolveriam alguma proporção pré-especificada dos pontos, geralmente 95% ou mais em cada conjunto. Traçando-se uma reta pelos dois pontos nos quais as elipses se interceptam e projetando-se a reta sobre um novo eixo Y, obtém-se a menor sobreposição entre as distribuições univariadas A' e B' (representada pela área sombreada). Nenhuma outra combinação linear obteria uma sobreposição menor através das elipses formadas pelos diagramas de dispersão.

O eixo Y (reta vermelha do gráfico da Figura 3.14) expressa os perfis das variáveis x_1 e x_2 como números únicos, chamados *discriminant scores*. Encontrando uma combinação linear das variáveis originais x_1 e x_2, consegue-se projetar os resultados como uma função discriminante. Se todos os pontos pequenos e grandes são projetados sobre o novo eixo Y, o resultado condensa a informação sobre diferenças de conjuntos (mostradas no gráfico x_1x_2) em um conjunto de pontos (*discriminant scores*) sobre um único eixo, mostrado pelas distribuições A' e B'.

Para um dado problema de *DFA*, quando uma combinação linear das variáveis independentes é determinada, obtém-se um *discriminant score* para cada amostra. Os *discriminant scores* são computados de acordo com uma regra estatística que visa maximizar a variância entre os grupos e minimizar a variância dentro deles. Se a variância entre os grupos é grande em relação à variância dentro dos grupos, diz-se que a função discriminante separa bem os grupos.

O algoritmo implementado no presente trabalho é o *LDA*, versão linear da *DFA*.

3.3.2 REDES BAYESIANAS (RB)

Classificações baseadas em estimativas de probabilidades são, em geral, mais interessantes do que as que não fazem uso de probabilidades, pois permitem que os resultados sejam ranqueados com base nas estimativas.

Embora possam ser obtidas probabilidades a partir de Árvores de Decisão calculando-se as frequências relativas de cada classe em um nó folha e, a partir de um conjunto de Regras de Decisão, através do exame das instâncias cobertas por uma regra particular, estas técnicas não fazem uso da estimativa de probabilidades para a aprendizagem da classificação em si (Witten e Frank, 2005).

Uma alternativa bem fundamentada matematicamente de representar distribuições de probabilidade de forma concisa e facilmente compreensiva são as RB[9].

A história das RB tem como ponto de partida um relatório técnico apresentado em 1976 no *DARPA* (*Defense Advanced Research Projects Agency*). Este relatório tinha o objetivo de contornar os principais inconvenientes das Árvores de Decisão, como a fragmentação do conjunto de treino e a replicação de sub-árvores.

Inspirado neste trabalho, Pearl (1986; 1988) propõe as RB para representação e análise de modelos envolvendo incertezas. Também conhecidas como Redes Casuais (ou Grafos de Dependência Probabilística), as RB podem ser consideradas como uma representação gráfica e informativa da distribuição de probabilidade conjunta das variáveis que envolvem o domínio de um problema (Jensen e Nielsen, 2001).

A partir da década de 1990, as RB difundiram-se e foram utilizadas para inúmeras aplicações. Diferem de outras técnicas de representação de conhecimento e análise probabilística, pois a incerteza é manipulada de uma forma matematicamente rigorosa, mantendo, de certo modo, uma notação simples e, em geral, provendo ganhos de eficiência.

As RB fornecem uma representação compacta e intuitiva das relações entre as variáveis (atributos) do conjunto de dados por meio de um grafo direcionado. Os nós deste grafo representam as variáveis aleatórias do problema e cada arco conectando dois nós indica a dependência probabilística direta entre duas variáveis.

O modelo gráfico resultante (ou rede) proporciona duas importantes utilidades:

(i) Visualização das relações probabilísticas: o modelo gráfico fornece informações diretas e acuradas a respeito das interações entre as variáveis de interesse; e

---

[9] As RB são assim denominadas por combinarem a representação de problemas em forma de rede (na verdade, um grafo) e o uso de estatística Bayesiana. O termo Bayesiano se deve ao matemático Thomas Bayes, a quem foi atribuída (postumamente) em 1763, na academia de ciências *Royal Society of London*, a autoria do teorema que leva o seu nome.

(ii) Inferência: Por ser intrinsicamente um modelo de inferência, as RB também podem ser usadas como algoritmos de classificação, tais como *NB* (Langley, 1992), *Tree-Augmented Bayesian Network* (Friedman, 1997), *K-dependence Bayesian classifier* (Sahami, 1996) e *Condensed Semi Naïve-Bayesian Network* (Perez *et al*., 2006), entre outros.

Teorema de Bayes

Seja Ω o conjunto de resultados possíveis, onde por "resultado possível" entende-se qualquer resultado elementar e indivisível de um experimento, seja também $\mathbb{A}$ a classe dos eventos aleatórios e, supondo que a todo $B \in \mathbb{A}$ seja associado um número real $P(B)$, chamado probabilidade de B, se a sequência (finita ou enumerável) de eventos aleatórios $A_1, A_2, \ldots$ formar uma partição de Ω, como em (3.27).

$$P(B) = \sum_i P(A_i)P(B|A_i), \ \forall B \in \mathbb{A} \tag{3.27}$$

A expressão algébrica (3.27) é chamada de Teorema da Probabilidade Total (ou Absoluta). Usando esse teorema, pode-se calcular a probabilidade de A_i dada a ocorrência de B, como em (3.28).

$$P(A_i|B) = \frac{P(A_i \cap B)}{P(B)} \Rightarrow P(A_i|B) = \frac{P(A_i)P(B|A_i)}{\sum_j P(A_j)P(B|A_j)} \tag{3.28}$$

A equação (3.28) é conhecida como Teorema de Bayes. Esta fórmula é útil quando se conhecem as probabilidades dos A_i e a probabilidade condicional de B dado A_i, mas não se conhece diretamente a probabilidade de B.

Na classificação de padrões deseja-se determinar a melhor hipótese para um espaço H, dado o conjunto de dados de treinamento D. Uma forma de se determinar a melhor hipótese é escolher a hipótese mais provável, dado D e mais algum conhecimento inicial sobre as probabilidades *a priori* das várias hipóteses em H.

O teorema de Bayes é a base das RB e provê uma maneira de calcular a probabilidade *a posteriori* $P(h|D)$, conforme a equação (3.29).

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}, \tag{3.29}$$

onde $P(h)$ é a probabilidade *a priori* da hipótese h, $P(D)$ é a probabilidade *a priori* do conjunto de dados de treinamento e $P(D|h)$ é a probabilidade do dado observado dada a hipótese h. Intuitivamente, $P(h|D)$ aumenta com $P(h)$ e com $P(D|h)$ e diminui à medida que $P(D)$ aumenta.

Em muitos cenários, considera-se um conjunto de hipóteses candidatas H com o objetivo de se encontrar a hipótese mais provável $h \in H$, dado o conjunto D de dados observados (ou pelo menos a hipótese mais provável, se houver várias). A hipótese mais provável é chamada de hipótese *Maximum a Posteriori* (*MAP*) h_{MAP} representada pela equação (3.30).

$$h_{MAP} \equiv \underset{h \in H}{argmax}\, P(D|h)P(h) \tag{3.30}$$

Pode-se notar que no passo final o termo $P(D)$ foi eliminado. Isto se deve ao fato de que este termo é uma constante independente de h.

Definição Formal

Conforme explicado anteriormente, as RB são representadas por grafos direcionados. Um grafo direcionado pode ser definido como um par ordenado $G(V, E)$, onde V é um conjunto de elementos denominados nós (ou vértices) e E é um conjunto de elementos denominados arestas (ou arcos).

Para maior simplicidade na implementação computacional das RB, o grafo é usualmente definido pelo seu conjunto de nós V e um mapa $A: V \times V \mapsto \{0, 1\}$ que indica o conjunto de arestas do grafo. Se existe uma aresta ligando um nó v_1 a outro v_2, então $A(v_1, v_2) = 1$ e se não existe uma aresta com essa propriedade, então $A(v_1, v_2) = 0$. Na representação gráfica de um grafo cada nó

é representado por um círculo e a existência de uma aresta conectando v_1 a v_2 é representada por uma seta ligando estes dois nós.

Nas RB, o grafo deve ainda ser acíclico, isto é, para qualquer $v \epsilon V$, não existe nenhum caminho ligando v a v. Uma consequência de um grafo ser acíclico é a de que sempre há pelo menos um nó ao qual nenhum outro se conecta, denominado *nó raiz*. Também é consequência de um grafo ser acíclico a existência de pelo menos um nó que não se conecta a qualquer outro, denominado *nó folha*.

Uma RB é, portanto, um grafo acíclico direcionado (tradicionalmente abreviado como *DAG*, do inglês, *Directed Acyclic Graph*) acrescido de uma família de funções de probabilidade. A cada nó do grafo acíclico está associada uma variável aleatória, entendendo-se por variável aleatória o própro nó. A essa variável aleatória está associada a sua probabilidade condicional dados todos os valores possíveis dos nós que se conectam a ela. Essa caracterização gera uma única probabilidade conjunta de todos os nós, no entanto pode existir mais de uma RB para representar uma determinada probabilidade conjunta dos nós.

Uma definição mais formal para RB, baseada em Jensen e Nielsen (2001) consiste em:

Seja $\boldsymbol{X} = \{X_1, X_2, \dots, X_n\}$ um conjunto de variáveis aleatórias multivariadas, cujos componentes X_i são também variáveis aleatórias, e um conjunto de arestas direcionadas entre as variáveis. Uma letra minúscula correspondente x_i denota uma atribuição de estado ou valor para a variável aleatória X_i. Cada variável tem um conjunto finito de estados mutuamente exclusivos e as variáveis juntamente com os arcos direcionados formam um *DAG*. Um grafo é acíclico se não existe nenhum caminho dirigido $X_1 \rightarrow X_2 \rightarrow \cdots \rightarrow X_n$, tal que $X_1 = X_n$.

$Parents(X_i)$ representa o conjunto de variáveis que possuem um arco direcionado apontando para X_i, também chamados de pais de X_i. Para cada variável A com pais $B_1, B_2, \dots B_n$ é atrelada uma tabela de probabilidade condicional $P(A|B_1, B_2, \dots B_n)$. Salienta-se, no entanto, que se um determinado nó X_i não possui pais, a tabela é reduzida a uma tabela de probabilidade incondicional $P(X_i)$, denominada por probabilidade *a priori*. Generalizando,

considere uma RB contendo n nós, X_1 até X_n, tomados nesta ordem. Para um valor particular de $\boldsymbol{X} = \{X_1, X_2, \ldots, X_n\}$, a distribuição de probabilidade conjunta é representada por:

$$p(\boldsymbol{X}) = p(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) \tag{3.31}$$

ou, de modo mais compacto, $p(x_1, x_2, \ldots, x_n)$. A regra da cadeia da teoria da probabilidade permite a fatoração de probabilidades conjuntas, assim:

$$\begin{aligned} p(\boldsymbol{X}) &= p(x_1)p(x_2|x_1) \ldots p(x_n|x_1, x_2, \ldots, x_{n-1}) \\ &= \prod_i p\,(x_n|x_1, x_2, \ldots, x_{n-1}). \end{aligned} \tag{3.32}$$

Como a estrutura da RB implica que o valor de um nó particular é condicional somente sobre os valores de seus nós pais, a equação (3.32) pode ser reduzida à equação (3.33):

$$p(\boldsymbol{X}) = \prod_i p\,(X_i|Parents(X_i)). \tag{3.33}$$

O Apêndice A do presente trabalho mostra um exemplo acadêmico de aplicação de RB.

3.3.2.1 *NAÏVE-BAYES* (*NB*)

O classificador *NB* usa uma abordagem probabilística para atribuir uma classe a cada instância do conjunto de dados, assumindo que todos os atributos são condicionalmente independentes das outras classes (Mitchell, 1997). O *NB* foi utilizado por Dombal *et al.* (1972) e introduzido para a tarefa de classificação por Duda e Hart (1973). Contudo, estudos relacionados a esta técnica podem ser rastreados ao menos até Minsky (1963).

Michie *et al.* (1994) fornecem um detalhado estudo comparando o *NB* com outros algoritmos de aprendizado, incluindo RNA e Árvores de Decisão. Esses pesquisadores mostram que na maioria das vezes o classificador *NB* é competitivo, e que em muitos casos, ele supera os outros métodos. Domingos

e Pazzani (1997) forneceram resultados teóricos que mostram que, apesar de sua simplicidade, o *NB* pode classificar melhor do que a maioria dos classificadores.

Em um classificador *NB*, cada atributo de predição tem o atributo classe como seu único filho. Isto significa que a estrutura é fixa e classificar equivale, portanto, a estimar parâmetros. O classificador *NB* é aplicado em tarefas de aprendizagem nas quais cada instância x é descrita por uma conjunção de valores de atributos, onde uma função $f(x)$ pode assumir qualquer valor de um conjunto finito V. Após a apresentação de um conjunto de instâncias de treino, descrito pela tupla $< a_1, a_2, \dots, a_n >$, o classificador prediz (classifica) o valor de uma nova instância.

Na classificação da nova instância, o método determina seu valor mais provável, v_{MAP}, dado os valores $< a_1, a_2, \dots, a_n >$ que descrevem a instância,

$$v_{MAP} = \underset{v_j \in V}{argmax}\ P(v_j | a_1, a_2, \dots, a_n). \tag{3.34}$$

A equação (3.34) pode ser reescrita aplicando o Teorema de Bayes:

$$\begin{aligned} v_{MAP} &= \underset{v_j \in V}{argmax}\ \frac{P(a_1, a_2, \dots, a_n | v_j) P(v_j)}{P(a_1, a_2, \dots, a_n)} \\ &= \underset{v_j \in V}{argmax}\ P(a_1, a_2, \dots, a_n | v_j) P(v_j). \end{aligned} \tag{3.35}$$

Agora podem ser calculados os dois termos da equação resultante (3.35), baseado nos dados de treino. Para calcular a probabilidade $P(v_j)$, basta contar a frequência em que cada valor de v_j ocorre nos dados de treino.

Contudo, o cálculo da probabilidade dos termos de $P(a_1, a_2, \dots, a_n | v_j)$ não é possível a menos que se tenha um conjunto de dados de treino bastante grande. O problema é que o número de termos é igual ao número de possíveis exemplos multiplicado pelo número dos possíveis valores procurados. Então, é necessário verificar muitas vezes cada exemplo em todo o conjunto de dados para se obter estimativas confiáveis.

O classificador *NB* é baseado na simples suposição de que os valores dos atributos são condicionalmente independentes dado o valor desejado. Em outras palavras, a suposição significa que, dado o valor desejado da instância, a probabilidade de ser observada a conjução $a_1, a_2, \ldots, a_n$ é o produtório das probabilidades para os atributos individuais:

$$P(a_1, a_2, \ldots, a_n | v_j) = \prod_i P(a_i | v_j). \quad (3.36)$$

Substituindo o produtório (3.36) na equação (3.35), obtem-se o classificador *NB* (3.37)

$$v_{NB} = \underset{v_j \in V}{argmax}\, P(v_j) \prod_i P(a_i | v_j), \quad (3.37)$$

onde v_{NB} denota o valor desejado informado pelo classificador *NB*.

Em um classificador *NB* o número de termos $P(a_i|v_j)$ distintos que devem ser estimados a partir do conjunto de dados de treino é o produto do número de valores distintos dos atributos de predição pelo número de valores distintos das classes, um número muito menor do que se fossem estimados os termos $P(a_1, a_2, \ldots, a_n | v_j)$ como contemplado anteriormente.

O método de aprendizado *NB* envolve um passo no qual vários termos são estimados, baseado em suas frequências a partir dos dados do conjunto de treino. O conjunto destas estimativas corresponde à hipótese aprendida. Esta hipótese é então usada para classificar cada nova instância aplicando a regra na equação (3.37).

Para Mitchell (1997), uma diferença interessante entre o método *NB* e outros métodos de aprendizado é que não existe nenhuma procura explícita pelo espaço das possíveis hipóteses (neste caso, o conjunto dos possíveis valores que podem ser obtidos a partir dos vários termos $P(v_j)$ e $P(a_i|v_j)$). A hipótese é formada simplesmente contando-se a frequência de várias combinações de dados dentro dos exemplos de treino.

4 MATERIAIS E MÉTODOS

O presente capítulo apresenta a metodologia proposta detalhada na forma de fluxograma na Figura 4.1; as bases de dados dos estudos de caso e *benchmarks* analisados; uma breve introdução teórica contemplando a justificativa para a solução de cada estudo de caso, além de detalhes técnicos a respeito dos recursos computacionais de *hardware* e *software* utilizados.

4.1 METODOLOGIA PROPOSTA

A metodologia proposta recebe como entrada um conjunto formado exclusivamente por dados nominais e apresenta como saída, a acurácia preditiva do melhor modelo de classificação encontrado.

Para simplificar a visualização da metodologia proposta e de como as técnicas utilizadas se relacionam entre si, o fluxograma da Figura 4.1 é dividido em três partes. A primeira parte, destacada em verde, representa a transformação geométrica por *MCA*; a segunda, em azul, mostra o *wrapper* de seleção de atributos baseado em *PBIL* sendo que, em seu interior é destacada a terceira parte, em vermelho, que trata da aplicação do AC. O fluxograma completo é formado por 10 passos, sendo cada um destes detalhados a seguir. A primeira parte é composta pelos três primeiros passos; a segunda pelos passos quatro a sete e dez; e a terceira pelos passos oito e nove.

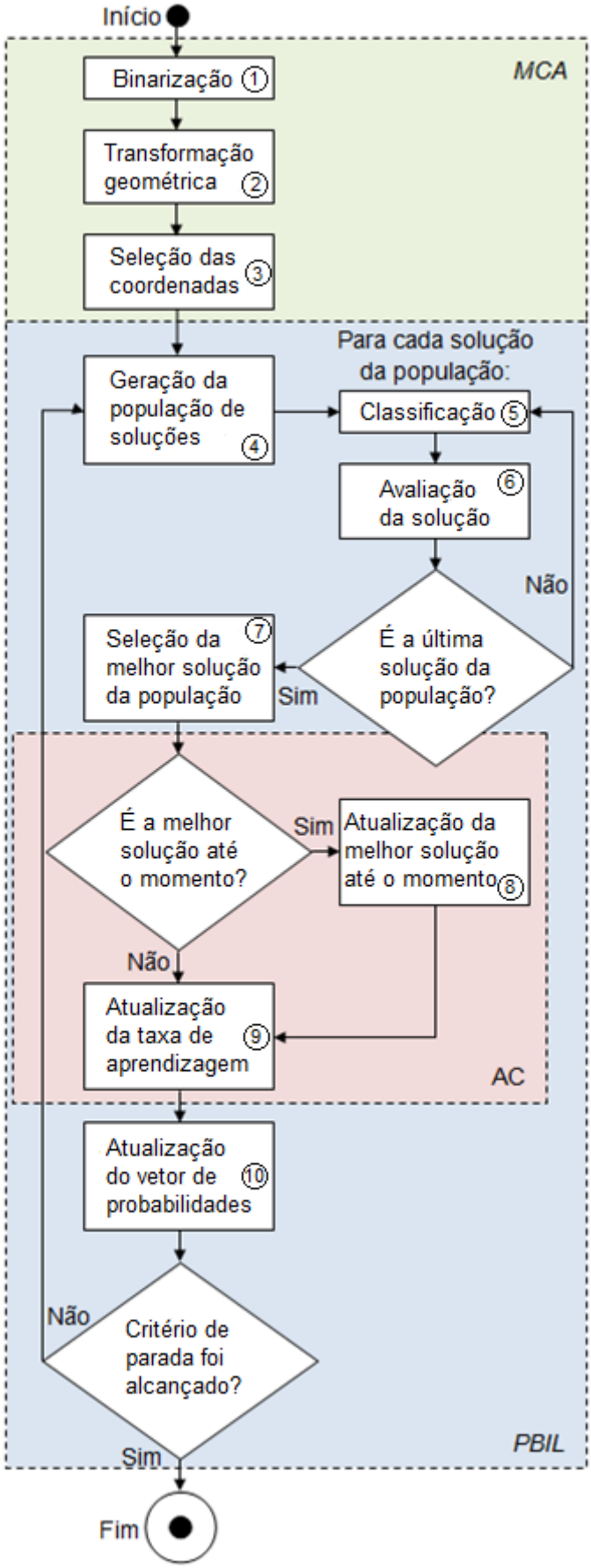

Figura 4.1 – Fluxograma da metodologia proposta.

No primeiro passo, os valores de cada atributo de predição pertencente aos dados nominais brutos são decompostos em níveis, cada um dos quais codificados como uma diferente variável binária, de tal modo que cada elemento só possa assumir o valor “0” ou “1”. Ao final deste primeiro passo, todos os atributos originais terão sido decompostos em atributos binários com uma e somente uma variável (coluna) assumindo o valor “1” por variável, obedecendo a determinação descrita na seção 3.2.4.

O segundo passo trata da aplicação da transformação geométrica sobre os novos atributos de predição binários. O presente trabalho compara os resultados dos classificadores com e sem a aplicação da transformação geométrica por *MCA*, além da aplicação da transformação por *PCA*. Ao final do segundo passo, obtêm-se as coordenadas dos dados transformados. No terceiro passo é realizada a seleção das coordenadas que melhor representem a variação total presente nos dados transformados por meio do gráfico *scree*.

O quarto passo consiste na geração da população de soluções candidatas para alimentação do *wrapper* de seleção de atributos. O presente trabalho compara o algoritmo de seleção de atributos proposto *BBIL* com outras duas técnicas de AE baseadas em populações: o *PBIL* padrão, algoritmo de *EDA* do qual o *BBIL* deriva, e o AG, base da maioria dos AE.

O quinto passo trata da aplicação de métodos de classificação de padrões utilizando os atributos selecionados por cada solução. O presente trabalho aplica dois diferentes classificadores, como já comentado, o *NB* e o *LDA*. A estimação dos erros para avaliação da acurácia preditiva de cada solução é realizada no sexto passo, onde o conjunto de dados é dividido em conjunto para treinamento e conjunto para teste utilizando-se a validação cruzada.

O sétimo passo compreende a seleção da solução com a melhor acurácia. No caso do *BBIL*, se a solução selecionada possui acurácia inferior à melhor solução local obtida até o momento, atualiza-se a taxa de aprendizagem aplicando-se a taxa de rejeição. Caso contrário, aplica-se o oitavo passo: a solução selecionada assume como a nova melhor solução atual e a taxa de aprendizagem volta ao seu patamar máximo.

No nono passo ocorre a atualização do vetor de probabilidades com base na taxa de aprendizagem e na melhor solução atual. Finalmente, no décimo passo, avalia-se se a condição de parada foi satisfeita, caso tenha sido satisfeita, a solução ótima foi encontrada, caso contrário, repete-se todo o processo a partir do quarto passo, gerando-se uma nova população de soluções. Em suma, a metodologia proposta combina a aplicação do *MCA* com o *wrapper* de seleção de atributos baseado em *BBIL*.

O critério adotado para avaliação da classificação foi a acurácia média obtida para todas as iterações sobre os dados de validação. Também conhecida como taxa de classificação correta, a acurácia corresponde à razão entre o número de instâncias classificadas corretamente e o número total de instâncias disponíveis em um conjunto de validação. A geração dos conjuntos de validação e estimação de erros utilizado no presente trabalho é a validação cruzada.

Na validação cruzada, cada instância do conjunto de dados completo é distribuída aleatoriamente em n partições[10] (aproximadamente) iguais estratificadas, cada uma das quais devendo ser utilizada somente uma vez como dados de validação, enquanto as demais são usadas como dados de treinamento. O procedimento de treinamento e validação/teste é repetido n vezes, de modo que, no final, cada instância tenha sido utilizada apenas uma vez como amostra de validação para todo o ensaio.

A acurácia é, então, calculada sobre cada um dos n conjuntos de validação. Ao final do processo é calculada a média das n acurácias a fim de se obter a acurácia média. O presente trabalho utiliza $n = 10$ como parâmetro do número de partições da validação cruzada, uma configuração conhecida como 10-*fold cross-validation*, número padrão de partições adotado na literatura de RP (Witten e Frank, 2005).

As bases de dados utilizadas para análise da metodologia proposta são apresentadas e descritas em detalhes nas seções 4.2, 4.3 e 4.4.

[10] O número de partições n trata-se de um inteiro parametrizado pelo usuário.

4.2 ESTUDO DE CASO 1: APLICAÇÃO NA ÁREA ELÉTRICA

Esta pesquisa teve início a partir de um projeto aprovado na Agência Nacional de Energia Elétrica (ANEEL), firmado entre a Universidade Federal do Paraná (UFPR), através do Programa de Pós-Graduação em Métodos Numéricos (PPGMNE), e uma concessionária de energia elétrica brasileira, com o objetivo principal de classificar a Qualidade de Energia Elétrica (QEE) em relação a afundamentos de tensão ocorridos na rede de distribuição de tal concessionária. Posteriormente os algoritmos desenvolvidos foram aplicados às demais bases de dados.

4.2.1 PROBLEMA 1: DETECÇÃO DE AFUNDAMENTOS

Para o presente estudo de caso foram obtidos dados provenientes de um sistema de registro de falhas relacionadas a interrupções de uma concessionária brasileira. Esta concessionária possui também dados de variações de tensão provenientes de instrumentos de medição conectados em uma barra de distribuição de 13.8 kV situada na mesma subestação onde foram coletados os registros de interrupções (Riella *et al.*, 2008). As propriedades físicas medidas por estes sensores são então associadas aos registros de interrupções para a geração do conjunto de dados utilizados no processo de classificação de afundamentos de tensão.

A associação entre os registros das interrupções e dos afundamentos de tensão “sentidos” na subestação foi baseada na data e hora de cada ocorrência no decorrer do período de 01/02/2008 a 31/05/2008 para uma única subestação que possui 12 alimentadores (Góes, 2012).

O conjunto de dados gerado pela associação entre estes registros possui 176 instâncias (ocorrências de falha), sendo que cada uma destas instâncias ficou composta por 5 atributos: *Área_Elétrica*, *Alimentador*, *Componente*, *Clima* e *Afundamento*.

Os atributos *Área_Elétrica*, *Alimentador* e *Componente* determinam o local onde ocorreu o evento, informando se a área é de baixa ou de alta tensão; o *Alimentador* na subestação indica o nome do mesmo; em

Componente afetado poderemos ter condutor, isolador, poste, transformador, dentre outros; o atributo *Clima* indica se no momento do evento o tempo era normal, chuvoso ou chuvoso com vento. O atributo *Afundamento* corresponde à classe do problema e indica se a rede detectou algum afundamento de tensão na mesma subestação no horário aproximado da ocorrência da respectiva interrupção. Das 176 instâncias do conjunto de dados de interrupções, 64 geram afundamentos de tensão. A Tabela 4.1 apresenta alguns registros do conjunto de dados.

Tabela 4.1 – Exemplos de instâncias do problema 1.

Área_Elétrica	***Alimentador***	***Componente***	***Clima***	***Afundamento***
Alta tensão	Oswaldo Cruz	Jumper	Chuvoso	sim
Alta tensão	Oswaldo Cruz	Jumper	Normal	sim
Alta tensão	São Miguel	Condutor	Normal	sim
Alta tensão	Santa Rita	Condutor	Chuvoso	sim
Alta tensão	Tucano	Condutor	Chuvoso com vento	sim
Baixa tensão	São Miguel	Condutor	Normal	não
Baixa tensão	Santa Rita	Condutor	Normal	sim

4.2.2 PROBLEMA 2: DIAGNÓSTICO DE CAUSAS

Uma das principais dificuldades encontradas pelas concessionárias de energia é a determinação da causa-raiz por detrás das ocorrências de afundamentos de tensão. A descoberta eficiente destas causas deve permitir maior eficácia em medidas corretivas e preventivas de reparo e planejamento do sistema elétrico, tais como ajustes de sistemas de proteção e dimensionamento de equipamentos.

Das 64 instâncias de afundamentos de tensão da base de dados utilizada na seção 4.2.3, somente 40 apresentam uma causa identificada pelos técnicos eletricistas de campo. As causas das outras 24 instâncias são simplesmente relatadas como “não identificadas”. Estas instâncias não identificadas podem ser usadas pelo algoritmo, posteriormente, como dados de

teste para que sejam atribuídas automaticamente suas causas pelo próprio classificador conforme seu aprendizado.

Portanto, esta base de dados ficou composta, por 40 instâncias, cada uma das quais compostas por 7 atributos: *Área_Elétrica*, *Alimentador*, *Componente*, *Clima*, *Magnitude_De_Tensão_Remanescente*, *Duração_Do_Afundamento* e *Causa*.

Os valores para *Magnitude_De_Tensão_Remanescente* são numéricos e dados na forma percentual com precisão de uma casa decimal e, por este motivo, foram separados em 4 intervalos: de 10% a 29,9%, de 30% a 49,9%, de 50% a 69,9% e de 70% a 90%. Não havia instâncias com magnitudes inferiores a 10% nem superiores a 90%, pois variações destas ordens não são consideradas no presente trabalho como afundamentos de tensão, de acordo com a definição do *IEEE* (1995).

Os valores para *Duração_Do_Afundamento* são numéricos e dados em milissegundos, por este motivo foram separados em 3 intervalos: de 0ms a 499ms, de 500ms a 999ms e de 1.000ms a 60.000ms (1 minuto). Não havia instâncias com durações superiores a 1 minuto, por motivos análogos.

O atributo *Causa*, como o próprio nome diz, indica a causa relatada pelo técnico como tendo sido a provável causa da falta (abalroamento, descarga atmosférica, árvores, animais, vandalismo, objetos estranhos na rede, corrosão e falha humana). Cada uma dessas causas irá representar, neste Problema 2 aqui abordado, uma classe para o problema. A Tabela 4.2 apresenta alguns registros do conjunto de dados do problema de diagnóstico de causas de afundamentos de tensão.

Tabela 4.2 – Exemplos de instâncias do problema 2.

Área_Elétrica	*Alimentador*	*Componente*	*Clima*	*Magnitude*	*Duração*	*Causa*
Alta tensão	Oswaldo Cruz	Jumper	Chuvoso	70% a 90%	0ms a 499ms	Corrosão
Alta tensão	Oswaldo Cruz	Jumper	Normal	30% a 49,9%	500ms a 999ms	Corrosão
Alta tensão	São Miguel	Condutor	Normal	50% a 69,9%	0ms a 499ms	Descarga atmosférica
Baixa tensão	São Miguel	Condutor	Normal	50% a 69,9%	500ms a 999ms	Descarga atmosférica
Baixa tensão	Santa Rita	Condutor	Normal	50% a 69,9%	500ms a 999ms	Vandalismo

4.3 ESTUDO DE CASO 2: APLICAÇÃO NA ÁREA MÉDICA

Nos últimos anos, diversos trabalhos têm sido apresentados na literatura sobre classificação de padrões aplicados à área médica (Steiner *et al.*, 2006, Escalante *et al.*, 2012, Tay *et al.*, 2013, Chen *et al.*, 2013). O presente trabalho aplica o algoritmo proposto a um problema médico aparentemente ainda inexplorado no contexto de *KDD*, o diagnóstico de rinoconjuntivite alérgica em pacientes pediátricos asmáticos.

O objetivo do presente estudo de caso é diagnosticar a rinoconjuntivite alérgica por meio de classificação de padrões utilizando dados clínicos e exames laboratoriais obtidos a partir do trabalho de Westphal *et al.* (2009). Os dados utilizados são apresentados na seção 4.3.2.

4.3.1 PROBLEMA 3: DIAGNÓSTICO DE RINOCONJUNTIVITE

O presente estudo de caso dispõe de dados clínicos (anamnese e exames físicos) e exames laboratoriais (testes cutâneos alérgicos) provenientes de prontuários médicos do Serviço de Alergia e Pneumologia Pediátrica do Hospital de Clínicas da UFPR (Universidade Federal do Paraná). Os dados clínicos utilizados foram obtidos a partir de um formulário padrão que inclui questões específicas sobre doenças alérgicas, enquanto os resultados de

exames laboratoriais referem-se a testes cutâneos alérgicos (*prick tests*) realizados com extratos de dois tipos de ácaros (*Dermatophagoides pteronyssinus* e *Blomia tropicalis*), um tipo de barata (*Blatella germanica*), um tipo de pólen de gramínea (*Lolium perenne*) e epitélio de cão e de gato.

O conjunto de dados utilizado ficou composto por 212 padrões (pacientes asmáticos), com idades entre 0 e 18 anos, selecionados entre Janeiro de 2001 e Janeiro de 2006. Cada uma das 212 instâncias da base de dados é composta por 16 atributos: *Gênero, Pai/Mãe Atópico, D. pteronyssinus, B. tropicalis, B. germanica, L. perenne, cão, gato, Asma, Internações por Asma, Rinite, Dermatite Atópica, Prurigo, Angioedema, Urticária* e *Rinoconjuntivite Alérgica*.

Os atributos *D. pteronyssinus, B. tropicalis, B. germanica, L. perenne, cão* e *gato* referem-se aos resultados dos testes cutâneos alérgicos. Os atributos *Asma, Internações por Asma, Rinite, Dermatite Atópica, Prurigo, Angioedema* e *Urticária* determinam a relação da rinoconjuntivite alérgica com outras doenças alérgicas. Os atributos *Gênero* e *Pai ou Mãe Atópico* tratam da predisposição genética. O atributo *Rinoconjuntivite Alérgica* corresponde à classe do problema e indica se o paciente apresenta sintomas desta doença (ou não). Dos 212 pacientes do conjunto de dados, 114 apresentam rinoconjuntivite alérgica. A Tabela 4.3 apresenta algumas instâncias do conjunto de dados.

Tabela 4.3 – Exemplos de instâncias do problema 3.

Rino-conjuntivite Alérgica	Predisposição genética		Teste cutâneo alérgico (*prick test*)[11]						Outras doenças alérgicas						
	Gênero	Pai/Mãe Atópico	*D. pteronys.*	*B. tropicalis*	*B. germanica*	*L. perenne*	Cão	Gato	Asma[12]	Internação por Asma	Rinite	Dermatite Atópica	Prurigo	Angioedema	Urticária
sim	masculino	sim	positivo	positivo	negativo	negativo	positivo	negativo	leve	não	não	não	não	não	não
não	masculino	sim	positivo	negativo	negativo	negativo	negativo	negativo	leve	não	sim	sim	sim	sim	sim
sim	masculino	sim	negativo	positivo	negativo	negativo	negativo	negativo	grave	não	não	não	não	não	não
não	feminino	não	positivo	positivo	negativo	negativo	negativo	positivo	moderada	não	não	não	não	não	não
não	masculino	não	positivo	negativo	positivo	positivo	positivo	positivo	moderada	não	não	não	não	não	não
não	masculino	não	negativo	negativo	negativo	negativo	negativo	negativo	grave	sim	sim	sim	sim	sim	sim
sim	feminino	não	positivo	positivo	positivo	positivo	positivo	positivo	moderada	não	não	não	não	não	não

[11] Extratos alergênicos do laboratório farmacêutico *International Pharmaceutical Immunology* (*ASAC Pharma Brazil*®).

[12] As categorias do atributo *Asma* foram obtidas conforme o *Global Initiative for Asthma* (*GINA*, 2006).

4.4 *BENCHMARKS* DE CLASSIFICAÇÃO DE DADOS NOMINAIS

Com o intuito de melhor analisar a metodologia proposta, o presente trabalho fez uso de seis problemas teste (*benchmarks*) disponíveis na literatura de classificação de padrões: *SPECT Heart, Soybean Large, Kr-vs-Kp, Promoter, Splice Junction* e *Audiology Standardized*. As bases escolhidas são de domínio público, compostas exclusivamente por dados nominais e podem ser acessadas no repositório do *UCI Machine Learning Repository* (Bache e Lichman, 2013).

4.4.1 PROBLEMA 4: *SPECT HEART*

O termo *SPECT*, sigla em inglês para *Single-Photon Emission Computed Tomography*, refere-se a uma imagem de tomografia computadorizada utilizada para avaliação de doenças cardiológicas relacionadas à perfusão miocárdica.

O diagnóstico é obtido comparando-se imagens do coração do paciente em estado de descanso e sob cansaço máximo. A interpretação de tais imagens, geralmente realizada de forma visual, pode levar a erros e inconsistências no diagnóstico. Assim, a classificação de padrões de imagens cardíacas de *SPECT* apresenta-se como uma interessante ferramenta de auxílio na tomada de decisão dos médicos cardiologistas (Sacha *et al.*, 2002).

A base de dados *SPECT* é composta por 267 instâncias (pacientes). Cada paciente é classificado em uma dentre duas classes: “normal” ou “anormal”. A base de dados é composta por 23 atributos, todos binários.

4.4.2 PROBLEMA 5: *SOYBEAN LARGE*

O *Soybean* refere-se a um repositório de doenças da soja disponibilizado em duas versões, o *Soybean Small*, com 47 instâncias de dados e o *Soybean Large*, utilizado no presente trabalho. O Soybean Large possui 307 instâncias, 34 atributos de predição e 19 classes indicando o causador das doenças da soja: *diaporthe-stem-canker, charcoal-rot,*

rhizoctonia-root-rot, phytophthora-rot, brown-stem-rot, powdery-mildew, downy-mildew, brown-spot, bacterial-blight, bacterial-pustule, purple-seed-stain, anthracnose, phyllosticta-leaf-spot, alternarialeaf-spot, frog-eye-leaf-spot, diaporthe-pod-&-stem-blight, cyst-nematode, 2-4-d-injury e *herbicide-injury*.

A origem desta base de dados é atribuída aos pesquisadores Michalski e Chilausky em um artigo publicado em 1980 entitulado "*Learning by Being Told and Learning from Examples: An Experimental Comparison of the Two Methods of Knowledge Acquisition in the Context of Developing an Expert System for Soybean Disease Diagnosis*".

4.4.3 PROBLEMA 6: *KR-VS-KP*

O problema *King-Rook vs. King-Pawn*, abreviado como *Kr-vs-Kp*, visa identificar se as peças brancas de um jogo de xadrez podem vencer uma partida dados determinados movimentos de peças pretas. A base de dados foi codificada pelo Professor Rob Holte do *Turing Institute*, em Glasgow (Escócia). A base é formada por 3.196 instâncias, 35 atributos nominais, indicando possíveis sequências de movimentos das peças e, 2 classes, *white_can_win* e *white_cannot_win*.

4.4.4 PROBLEMA 7: *PROMOTER*

O problema *Promoter* visa classificar regiões promotoras de bactéricas *Escherichia coli* com base em sequências de genes. A base de dados utilizada no presente trabalho é versão original disponibilizada pelos Professores C. Harley e R. Reynolds da *McMaster University*, em Ontario (Canadá), e pelo Professor T. Record da *University of Wisconsin*, em Wisconsin (EUA). A base é composta por 106 instâncias, 57 atributos nominais contendo o nucleotídeo de cada gene e, 2 classes, indicando se a região é ou não promotora.

4.4.5 PROBLEMA 8: *SPLICE JUNCTION*

Assim como a base *Promoter,* a base de dados *Splice Junction* também trata de sequências de genes. A base de dados utilizada no presente trabalho é versão original disponibilizada pelos Professores G. Towell, M. Noordewier e J. Shavlikda da *State University of New Jersey* (EUA).

De maneira geral, os *splice junctions* são os pontos em uma sequência de genes em que o *DNA* "supérfluo" é removido durante o processo de criação de proteínas em organismos superiores. A base de dados utilizada é composta por 3.190 instâncias (sequências de genes), 60 atributos de predição nominais e 3 classes, visando reconhecer as partes da sequência de genes removidas após a criação de proteínas.

4.4.6 PROBLEMA 9: *AUDIOLOGY STANDARDIZED*

O problema *Audiology Standardized* trata do diagnóstico de doenças audiológicas com base em sintomas e resultados de exames físicos. A base de dados utilizada no presente trabalho é uma versão padronizada a partir da base original denominada *Audiology*, disponibilizada pelo Professor Jergen do *Baylor College of Medicine*, em Houston, Texas. Foram removidos os dois atributos e as oito instâncias contendo *missing data* (dados ausentes), além de seis classes que continham somente uma instância cada.

Assim, a base utilizada no presente trabalho é composta por 186 instâncias, 68 atributos nominais e 18 classes, a saber:
cochlear_unknown, mixed_cochlear_age_otitis_media, mixed_poss_noise_om, cochlear_age, normal_ear, cochlear_poss_noise, cochlear_age_and_noise, mixed_cochlear_unk_ser_om, conductive_discontinuity, retrocochlear_unknown, conductive_fixation, cochlear_noise_and_heredity, mixed_cochlear_unk_fixation, otitis_media, possible_menieres, possible_brainstem_disorder, mixed_cochlear_age_s_om e *mixed_cochlear_unk_discontinuity*.

A base de dados padronizada tem sido usada em diversos trabalhos (Hutter e Zaffalon, 2004; Athitsos e Sclaroff, 2004). Detalhes da padronização podem ser obtidos em: http://archive.ics.uci.edu.

4.5 RECURSOS COMPUTACIONAIS

O caráter computacional dos métodos numéricos para análise de dados e avaliação de técnicas de RP exige a adoção de uma plataforma que permita implementações flexíveis de programas computacionais. O presente trabalho utiliza o *software MATLAB* como plataforma de programação. Isto se deve, principalmente, a três fatores: a profunda disseminação desta plataforma no meio acadêmico, sua sintaxe simples e sua capacidade em lidar com algoritmos de computação numérica por meio de matrizes.

Em termos de *hardware*, todas as simulações foram realizadas usando um computador *Sony Vaio*® com processador *Intel Core*® i5-2410M 2.30 GHz, com 4 GB de memória *RAM* e sistema operacional *Microsoft Windows*® 7 *Home Premium* 64 bits *Service Pack* 1.

5 ANÁLISE DOS RESULTADOS

Os algoritmos de classificação utilizados no presente trabalho, *NB* e *LDA*, foram inicialmente configurados para a execução de 100 iterações[13] e o critério principal adotado para avaliação dos mesmos foi a acurácia média obtida para todas as iterações sobre os dados de validação. A geração dos conjuntos de validação e estimação de erros foi feita por validação cruzada. Uma descrição detalhada deste método pode ser obtida em Witten e Frank (2005).

A Tabela 5.1 apresenta os resultados da classificação em termos de acurácia preditiva média por *NB* e *LDA* sobre as bases de dados dos problemas descritos no capítulo 4 (dados brutos). As melhores acurácias de cada problema é marcada em negrito.

Tabela 5.1 – Resultados das classificações dos dados brutos.

Problema	*NB*				*LDA*			
	Média (%)	Desvio Padrão (%)	Mediana (%)	Tempo (s)	Média (%)	Desvio Padrão (%)	Mediana (%)	Tempo (s)
Problema 1: Detec. afund.	73,2596	0,3720	73,2741	3,45	**73,3859**	0,3968	73,3341	2,19
Problema 2: Diag. causas	84,2002	1,2961	84,2181	1,99	**89,9855**	0,9907	90,0681	1,89
Problema 3: Rinoconjuntivite	64,5446	0,7056	64,4596	2,33	**65,3929**	1,0655	65,6427	3,20
Problema 4: *SPECT*	77,2067	0,4413	77,1767	2,95	**80,8926**	0,5555	80,8926	3,83
Problema 5: *Soybean*	**89,0327**	0,3120	89,0141	6,73	88,4793	0,4627	88,3516	8,23
Problema 6: *Kr-vs-Kp*	86,8401	0,2502	86,8734	347,89	**92,3526**	2,3317	92,3580	361,05
Problema 7: *Promoter*	91,6206	0,4450	91,6603	2,48	**98,8874**	0,3412	98,9398	3,45
Problema 8: *Splice*	82,5188	0,1012	82,5305	913,85	**83,4158**	0,0939	83,4132	850,19
Problema 9: *Audiology*	**75,5837**	1,1709	75,5679	5,83	75,5180	1,3024	75,5442	5,46

[13] Número de ensaios diferentes de geração de conjuntos de treinamento e estimação de erro.

A Figura 5.1 representa graficamente as acurácias médias da Tabela 5.1. Esta figura fornece uma noção geral da qualidade dos conjuntos de dados utilizados visando à tarefa de classificação de padrões, notando-se melhores resultados para o problema 7 (*Promoter*) e os piores resultados para o problema 3 (diagnóstico de rinoconjuntivite).

O objetivo de apresentar os resultados de classificação dos dados brutos não é o de comparar a acurácia preditiva entre os classificadores *NB* e *LDA*, mas somente situar o "ponto de partida" para a aplicação das simulações sobre os dados tratados (por transformação geométrica e seleção de atributos), as quais sim são objetos de análises comparativas com os resultados da classificação sobre os dados brutos nas demais seções do presente capítulo.

Neste sentido, uma inspeção visual na Figura 5.1 mostra grande semelhança de resultados entre os classificadores utilizados para cada um dos problemas (bases de dados) descritos no capítulo 4.

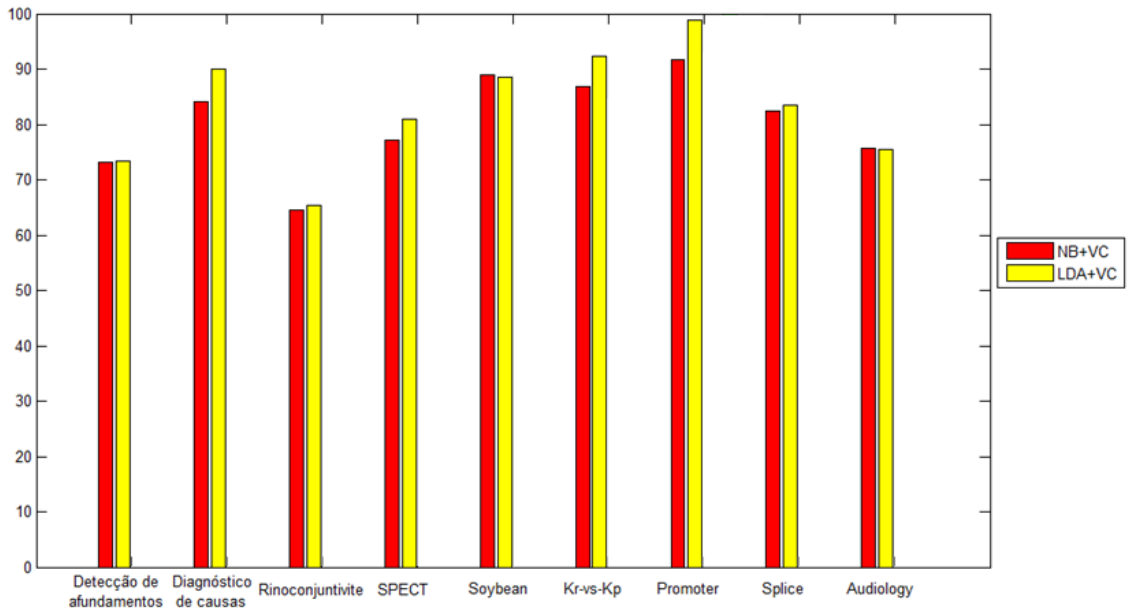

Figura 5.1 – Gráfico das acurácias médias da Tabela 5.1.

5.1 TRANSFORMAÇÃO POR *GDA*

Após a realização da classificação de padrões sobre os dados brutos, são aplicadas as transformações *GDA* (*PCA* e *MCA*) sobre estes mesmos

dados visando obter dados transformados, potencialmente, trazendo ganhos de desempenho em termos de acurácia preditiva em relação aos dados brutos. As transformações *GDA* são determinísticas, assim os algoritmos são aplicados somente uma vez para cada conjunto de dados.

A Figura 5.2 apresenta o gráfico dos dados transformados por *PCA* referente aos dois primeiros *PCs*[14] de todos os nove problemas (conjuntos de dados) analisados: (a) detecção de afundamentos; (b) diagnóstico de causas; (c) diagnóstico de rinoconjuntivite; (d) *SPECT*; (e) *Soybean*; (f) *Kr-vs-Kp*; (g) *Promoter*; (h) *Splice*; e (i) *Audiology*. Em geral, este gráfico bidimensional deve auxiliar a detecção de padrões nos dados ao fornecer uma representação visual de "como os dados se parecem", isto é, como as instâncias se relacionam entre si e as classes (representadas por cores) de cada uma delas.

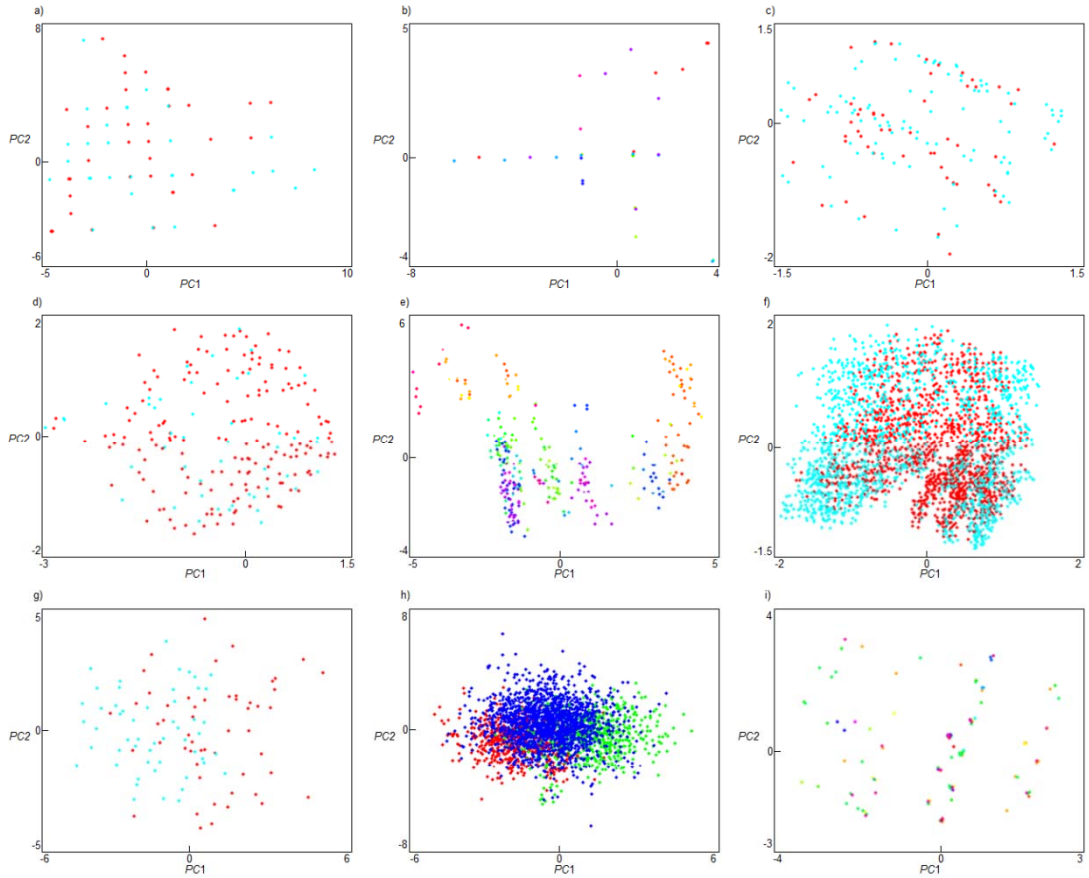

Figura 5.2 – Gráficos *PC*1 x *PC*2 para os nove problemas analisados.

[14] Os dois primeiros *PCs* devem captar a maior parte da variância dos dados, conforme apresentado na seção 3.2.1.1.

Contudo, devido ao fato dos dados analisados serem nominais, a transformação por *PCA* apresenta pouca (ou nenhuma) eficiência na separação dos dados (as classes dos dados transformados deveriam formar *clusters* com o mínimo de sobreposição entre si). A Figura 5.3 apresenta os *clusters* dos dois primeiros *factor scores* (*FS*1 e *FS*2) do *MCA*. Embora as instâncias da Figura 5.3 não apresentem tão evidente menor sobreposição que as da Figura 5.2, em termos de acurácia os resultados do *MCA* são consideravelmente superiores em comparação com o *PCA*.

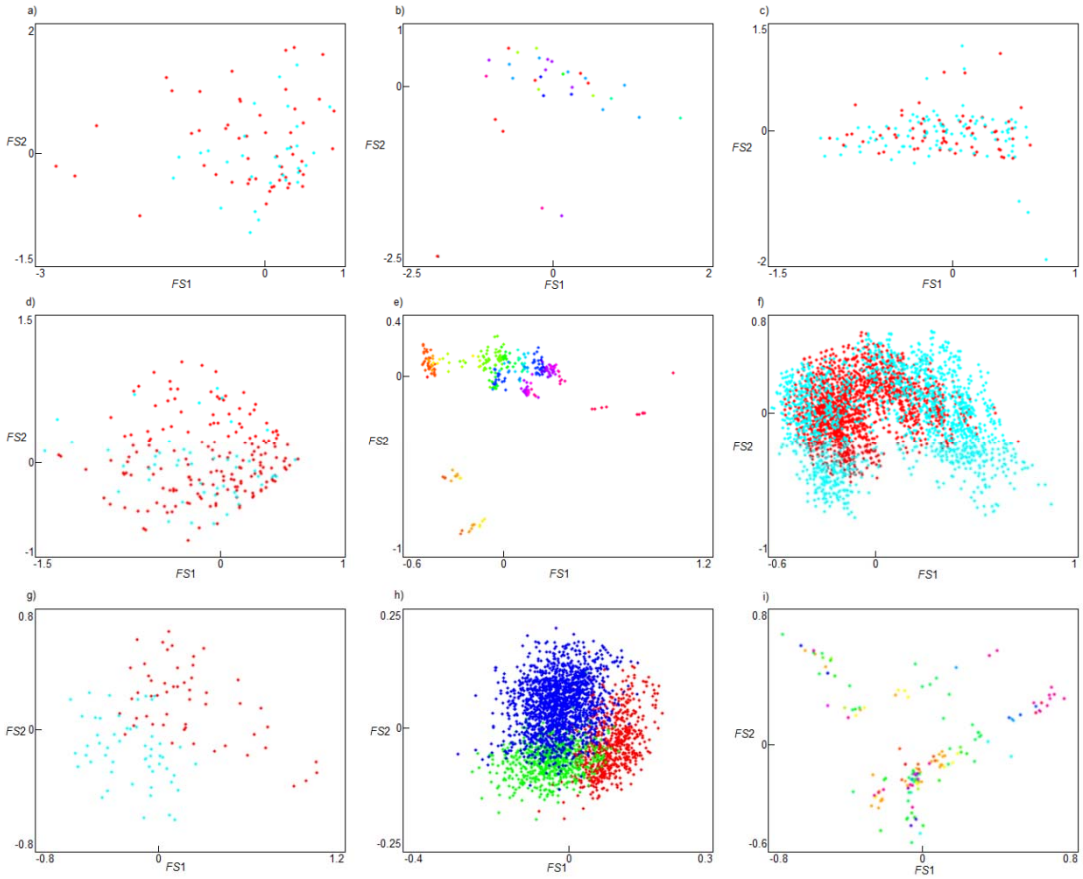

Figura 5.3 – Gráficos *FS1* x *FS2* para os nove problemas analisados.

Ainda na Figura 5.3, provavelmente, o melhor exemplo de boa separação é para o conjunto de dados *Promoter* (Figura 5.3g), o qual, como se verá a seguir, foi o conjunto de dados que obteve os melhores resultados utilizando o *MCA*, alcançando 100% de acurácia para os dois classificadores utilizados (*NB* e *LDA*).

As Tabelas 5.2 e 5.3 apresentam, respectivamente, a análise comparativa das acurácias médias do classificador *NB* e *LDA*, quando aplicados sobre os dados brutos e sobre os dados transformados por *PCA* e *MCA* sobre os problemas analisados.

Tabela 5.2 – Transformações *GDA* e classificação *NB*.

Problema	Dados brutos				Dados transformados por *PCA*					Dados transformados por *MCA*				
	Média (%)	Desvio Padrão (%)	Mediana (%)	Tempo (s)	Média (%)	Desvio Padrão (%)	Mediana (%)	Tempo (s)	Ganho*	Média (%)	Desvio Padrão (%)	Mediana (%)	Tempo (s)	Ganho*
Problema 1: Detec. afund.	73,2596	0,3720	73,2741	3,45	73,3754	0,3916	73,3528	8,82	+0,2% ▲	**81,9028**	0,7159	81,9910	58,72	+11,8% ▲
Problema 2: Diag. causas	84,2002	1,2961	84,2181	1,99	88,7352	1,1144	88,7752	9,82	+5,4% ▲	**98,4968**	0,4022	98,5391	39,60	+17,0% ▲
Problema 3: Rinoconjuntivite	64,5446	0,7056	64,4596	2,33	64,7566	0,5546	64,7852	25,88	+0,3% ▲	**67,8551**	0,4207	67,9514	28,66	+5,1% ▲
Problema 4: *SPECT*	77,2067	0,4413	77,1767	2,95	**81,6027**	0,6215	81,6343	47,38	+5,7% ▲	81,2748	0,49354	81,3211	49,82	+5,3% ▲
Problema 5: *Soybean*	89,0327	0,3120	89,0141	6,73	97,2652	0,1436	97,2909	223,11	+9,2% ▲	**97,7717**	0,0911	97,7669	222,49	+9,8% ▲
Problema 6: *Kr-vs-Kp*	86,8401	0,2502	86,8734	347,89	**93,0810**	0,1361	93,1252	6.036,63	+7,2% ▲	90,1410	0,0526	90,1567	7.443,81	+3,8% ▲
Problema 7: *Promoter*	91,6206	0,4450	91,6603	2,48	97,5688	0,5309	97,5786	110,50	+6,5% ▲	**100**	0,0000	100	226,44	+9,1% ▲
Problema 8: *Splice*	82,5188	0,1012	82,5305	913,85	80,0480	0,0548	80,0349	18.315,40	-3,1% ▼	**92,8417**	0,0638	92,8430	20.097,59	+12,5% ▲
Problema 9: *Audiology*	75,5837	1,1709	75,5679	5,83	94,5969	0,4315	94,6305	303,09	+25,1% ▲	**99,0107**	0,1627	98,9781	286,75	+31,0% ▲

* Ganho de desempenho de acurácia preditiva média em relação aos dados brutos.

Tabela 5.3 – Transformações *GDA* e classificação *LDA*.

Problema	Dados brutos				Dados transformados por *PCA*					Dados transformados por *MCA*				
	Média (%)	Desvio Padrão (%)	Mediana (%)	Tempo (s)	Média (%)	Desvio Padrão (%)	Mediana (%)	Tempo (s)	Ganho*	Média (%)	Desvio Padrão (%)	Mediana (%)	Tempo (s)	Ganho*
Problema 1: Detec. afund.	73,3859	0,3968	73,3341	2,19	73,3515	0,3634	73,3583	9,17	-0,1% ▼	**82,5157**	0,5051	82,4479	85,97	+12,4% ▲
Problema 2: Diag. causas	89,9855	0,9907	90,0681	1,89	90,1011	0,7482	90,0520	11,20	+0,1% ▲	**100**	0	100	55,76	+11,1% ▲
Problema 3: Rinoconjuntivite	65,3929	1,0655	65,6427	3,20	64,5948	0,9587	64,7778	33,70	-1,2% ▼	**67,9210**	0,5376	67,9449	41,44	+3,9% ▲
Problema 4: *SPECT*	80,8926	0,5555	80,8926	3,83	**82,5266**	0,6036	82,5256	60,42	+2,0% ▲	81,2539	0,5799	81,5451	84,72	+0,5% ▲
Problema 5: *Soybean*	88,4793	0,4627	88,3516	8,23	97,5755	0,1451	97,5848	260,77	+10,3% ▲	**98,4702**	0,0628	98,4585	274,91	+11,3% ▲
Problema 6: *Kr-vs-Kp*	92,3526	2,3317	92,3580	361,05	**93,5576**	0,0115	93,5590	195,52	+0,7% ▲	91,0560	0,0984	91,0478	215,88	-1,4% ▼
Problema 7: *Promoter*	98,8874	0,3412	98,9398	3,45	99,7513	0,1882	99,7895	151,18	+0,9% ▲	**100**	0	100	528,52	+1,1% ▲
Problema 8: *Splice*	83,4158	0,0939	83,4132	850,19	80,9287	0,1469	80,9080	19.659,99	-3,1% ▼	**98,9518**	0,0070	98,9482	22.638,17	+18,6% ▲
Problema 9: *Audiology*	75,5180	1,3024	75,5442	5,46	96,3829	1,0295	96,4011	481,14	+27,6% ▲	**99,0137**	2,9319	99,5748	451,78	+31,1% ▲

As Figuras 5.4 e 5.5 apresentam graficamente as acurácias médias das Tabelas 5.2 e 5.3. Nestas Figuras, as barras em cor vermelha representam as acurácias de classificação diretamente a partir dos dados brutos (sem a aplicação prévia de nenhum tipo de transformação *GDA*). As barras em cor amarela, as acurácias de classificação com os dados transformados pelo *PCA* e, em cor azul, com os dados transformados pelo *MCA*.

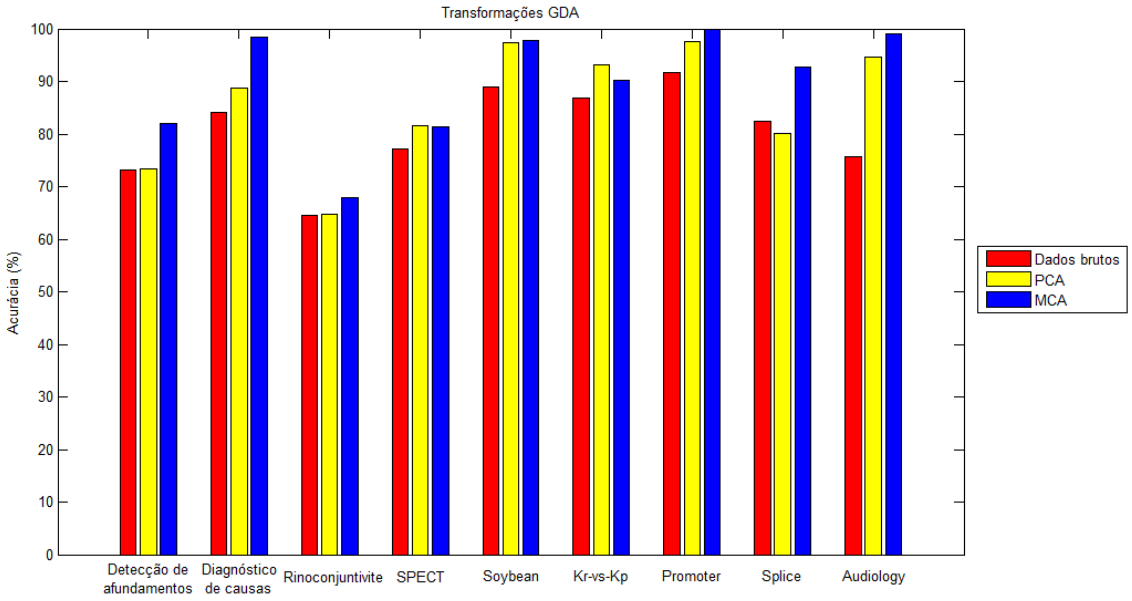

Figura 5.4 – Transformações *GDA* e classificação *NB*.

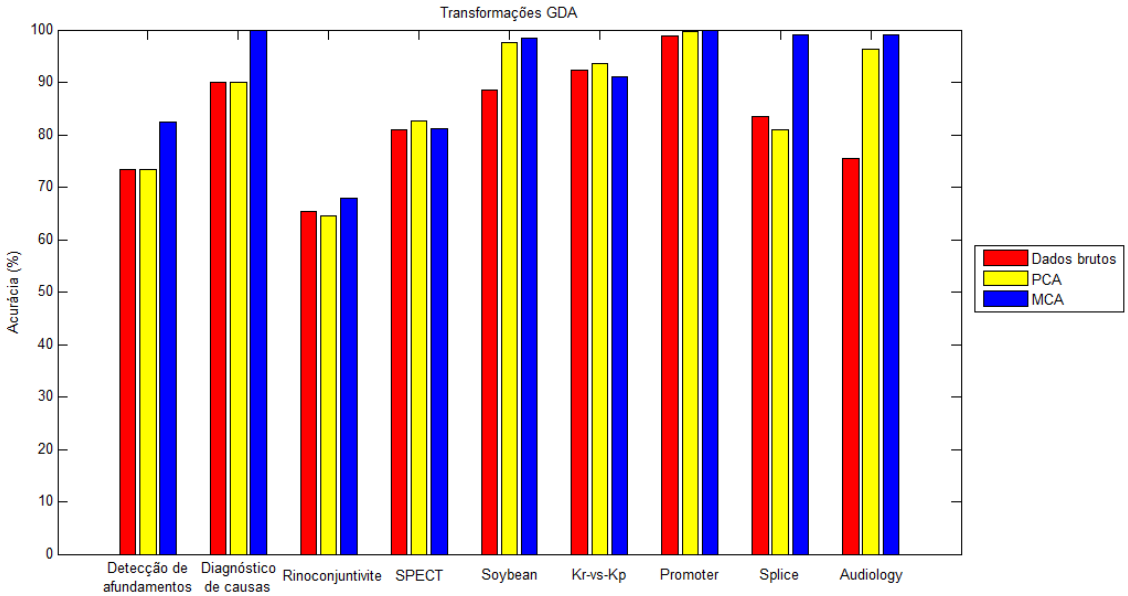

Figura 5.5 – Transformações *GDA* e classificação *LDA*.

O classificador *NB* obteve ganho de desempenho em todas as transformações *GDA*, seja por *PCA* ou por *MCA*, com exceção da base *Splice* usando a transformação *PCA*. A base de dados mais sensível à melhoria de desempenho proporcionada por ambas as transformações *GDA* foi a *Audiology Standardized*. A base de dados em que o *MCA* apresentou seu pior desempenho foi a *Kr-vs-Kp*, a qual inclusive o *MCA* chegou a piorar a acurácia em comparação com os dados brutos nas situações em que o classificador utilizado foi o *LDA*.

Embora o *PCA* tenha se apresentado melhor em 4 das 18 comparações (bases de dados *SPECT* e *Kr-vs-Kp*), na maioria dos casos, o *MCA* mostrou-se superior (venceu em todas as outras 14 comparações). Em nenhum caso os resultados de classificação pioraram ao aplicarem-se ambas as técnicas de *GDA*.

Com relação as base de dados *SPECT* e *Kr-vs-Kp*, o fato de estas serem as únicas bases de dados, simultaneamente, dicotômicas[15] e compostas exclusivamente por dados binários pode ter feito com que o *MCA* não tenha sido mais vantajoso que o *PCA*. Um fato curioso é que o *LDA* classificou corretamente 100% dos casos dos problemas *Promoter* e de diagnóstico de causas de afundamentos, quando combinado à transformação por *MCA*. Aliás, para o conjunto de dados *Promoter*, o *MCA* permitiu uma classificação correta de 100% dos casos.

Outra observação é que os piores resultados para o *PCA* foram para a base de dados do problema de *Splice*, à qual inclusive houve perda de acurácia em comparação com a classificação dos dados brutos em todas as simulações analisadas. Já para o *MCA*, os piores resultados ocorreram para a base de dados *Kr-vs-Kp*, à qual houve perda de acurácia em comparação com a classificação dos dados brutos no caso do classificador *LDA*.

Calculando-se a média (das médias de cada uma das 18 situações) para a melhoria de desempenho (ganho) alcançada pelo *PCA*, obtem-se que esta transformação melhorou a acurácia preditiva em 5,21% em média, enquanto que o *MCA* melhorou 10,78%, mais que o dobro do *PCA*.

---

[15] Conjuntos de dados que possuem somente duas classes.

5.2 TESTES DE PERMUTAÇÃO

Testes de permutação são uma classe de testes não paramétricos estatisticamente válidos em situações com amostragem pequena e baseiam-se no fato de que, sob a hipótese nula, todas as permutações possíveis de um conjunto de dados são igualmente possíveis de ocorrer (Good, 1994).

Em problemas de classificação de padrões, os testes de permutação correspondem à permutação das classes associadas a cada instância de um conjunto de dados (Witten e Frank, 2005). O teste é feito de tal modo que sejam permutadas apenas as classes de cada instância (sem alterar os valores dos demais atributos) e mantendo-se o número de instâncias existentes para cada classe no conjunto de dados original.

Os testes de permutação visam avaliar de forma isenta se o desempenho de um classificador é superior para o conjunto de dados não permutados em relação ao conjunto de dados permutados (situações reais, mas com as classes atribuídas propositalmente de forma incorreta).

Ao se assumir que as classes de cada instância estão erradas (sem que isto seja informado ao classificador) e, mesmo assim, o classificador apresentar desempenho superior (ou mesmo equivalente) ao seu resultado com o conjunto de dados com as classes corretas, isto deve implicar que os dados são meramente aleatórios ou espúrios e que não são um bom conjunto de dados para ser utilizado como problema de classificação. Um modelo de classificação utilizando dados permutados deve se comportar de maneira inferior a um modelo usando dados não permutados.

As Tabelas 5.4 e 5.5 apresentam a comparação dos resultados dos classificadores analisados utilizando-se os dados sem permutação e com permutação para os classificadores *NB* e *LDA*, respectivamente.

Tabela 5.4 – Teste de permutação do *NB*.

Problema	Dados permutados?	Dados brutos		Dados transformados por *PCA*		Dados transformados por *MCA*	
		Média (%)	Desvio Padrão (%)	Média (%)	Desvio Padrão (%)	Média (%)	Desvio Padrão (%)
Problema 1: Detec. afund.	Não	**73,2596**	0,3720	**73,3754**	0,3916	**82,5535**	0,5960
	Sim	62,6836	0,9903	53,6950	0,7032	66,7178	0,7175
Problema 2: Diag. causas	Não	**84,2002**	1,2961	**88,7352**	1,1144	**98,4968**	0,4022
	Sim	57,4406	1,6395	63,5209	1,2286	77,1975	1,2531
Problema 3: Rinoconjuntivite	Não	**64,5446**	0,7056	**64,7566**	0,5546	**68,0478**	0,6142
	Sim	60,4942	0,7284	57,3573	0,8559	61,4297	0,6526
Problema 4: *SPECT*	Não	**77,2067**	0,4413	**81,6027**	0,6215	**81,2748**	0,4935
	Sim	61,0098	0,6004	63,3307	0,7828	63,6170	0,5756
Problema 5: *Soybean*	Não	**89,0327**	0,3120	**97,2652**	0,1436	**97,7717**	0,0911
	Sim	70,2019	0,9102	72,6784	0,8147	72,5503	0,9599
Problema 6: *Kr-vs-Kp*	Não	**86,8401**	0,2502	**93,0810**	0,1361	**90,1410**	0,0526
	Sim	53,1989	0,2537	53,4822	0,1928	53,6183	0,1064
Problema 7: *Promoter*	Não	**91,6206**	0,4450	**97,5688**	0,5309	**100**	0
	Sim	81,2315	0,7992	85,5034	0,8747	91,2451	0,8010
Problema 8: *Splice*	Não	**82,5188**	0,1012	**80,0480**	0,0548	**92,8417**	0,0638
	Sim	73,1155	1,0498	75,8806	1,1197	74,2081	0,9154
Problema 9: *Audiology*	Não	**75,5837**	1,1709	**94,5969**	0,4315	**99,0254**	0,1541
	Sim	44,0111	1,3739	70,6134	0,7966	71,9507	0,8176

Tabela 5.5 – Teste de permutação do *LDA*.

Problema	Dados permutados?	Dados brutos		Dados transformados por *PCA*		Dados transformados por *MCA*	
		Média (%)	Desvio Padrão (%)	Média (%)	Desvio Padrão (%)	Média (%)	Desvio Padrão (%)
Problema 1: Detec. afund.	Não	**73,3859**	0,3968	**73,3515**	0,3634	**82,5157**	0,5051
	Sim	60,7218	0,7155	57,0578	0,4078	67,7974	0,7513
Problema 2: Diag. causas	Não	**89,9855**	0,9907	**90,1011**	0,7482	**100**	0
	Sim	56,9839	2,3124	51,1529	1,6875	87,8780	0,91834
Problema 3: Rinoconjuntivite	Não	**65,3929**	1,0655	**64,5948**	0,9587	**67,9210**	0,5376
	Sim	58,3710	0,9091	56,6630	0,8040	61,6868	0,7466
Problema 4: *SPECT*	Não	**80,8926**	0,5555	**82,5266**	0,6036	**81,2602**	0,51696
	Sim	64,9382	0,5826	68,9675	0,6592	67,0144	0,7876
Problema 5: *Soybean*	Não	**88,4793**	0,4627	**97,5755**	0,1451	**98,4702**	0,0628
	Sim	70,6791	0,7149	71,4816	0,9227	72,0391	1,0199
Problema 6: *Kr-vs-Kp*	Não	**92,3526**	2,3317	**93,5576**	0,0115	**91,0560**	0,0984
	Sim	55,2316	0,2446	54,7696	0,1861	55,4037	0,1008
Problema 7: *Promoter*	Não	**98,8874**	0,3412	**99,7513**	0,1882	**100**	0
	Sim	88,0408	1,0232	91,2497	0,9039	92,3918	1,0015
Problema 8: *Splice*	Não	**83,4158**	0,0939	**80,9287**	0,1469	**98,9518**	0,0070
	Sim	73,7243	1,1964	75,5024	1,3548	73,3982	0,9287
Problema 9: *Audiology*	Não	**75,5180**	1,3024	**96,3829**	1,0295	**99,0137**	2,9319
	Sim	44,0749	0,7996	75,9721	1,2621	79,3427	2,6207

As Figuras 5.6 e 5.7 exibem de forma gráfica os resultados das Tabelas 5.4 e 5.5, respectivamente. Para cada uma destas figuras, o eixo das ordenadas representa a variância da acurácia preditiva média de cada modelo.

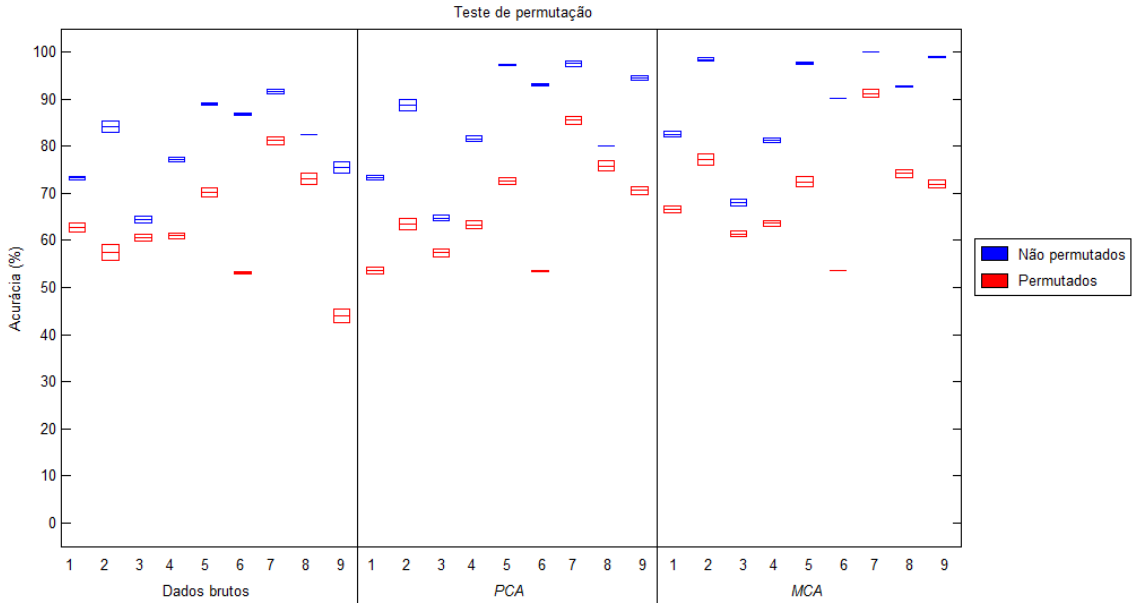

Figura 5.6 –Teste de permutação do *NB*.

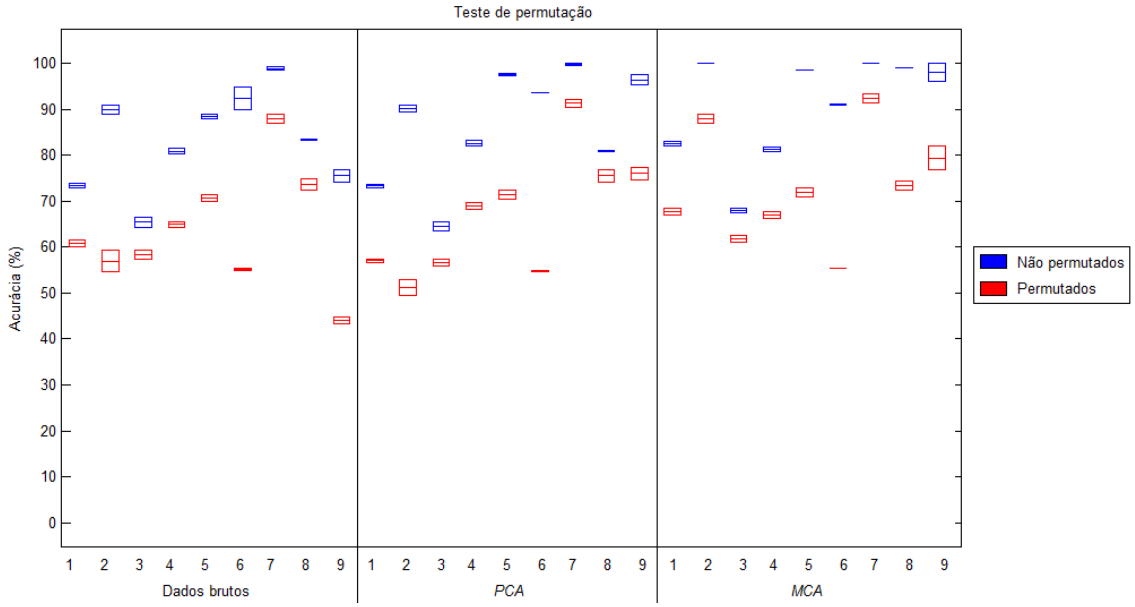

Figura 5.7 –Teste de permutação do *LDA*.

No eixo das abscissas são apresentados os resultados dos testes de permutação sobre os dados brutos dos problemas de 1 a 9 (detecção de afundamentos, diagnóstico de causas, diagnóstico de rinoconjuntivite, *SPECT*,

Soybean, *Kr-vs-Kp*, *Promoter*, *Splice* e *Audiology*, respectivamente), sobre os dados transformados por *PCA*, também dos problemas de 1 a 9, e sobre os dados transformados por *MCA* para os mesmos problemas.

Em todas as situações os resultados com dados não permutados (em cor azul) foram superiores aos resultados com dados permutados (em cor vermelha), ressaltando empiricamente o quão desafiadores são os problemas estudados e a relevância dos resultados obtidos. Em nenhum caso os resultados dos dados não permutados apresentaram menor superioridade em relação aos dados permutados.

5.3 SELEÇÃO DE ATRIBUTOS

O objetivo dos experimentos da presente seção é analisar as vantagens e limitações do *BBIL* em relação ao AG e ao *PBIL* padrão quando aplicados ao problema de seleção de atributos sobre dados nominais através da abordagem *wrapper* sob os seguintes critérios: acurácias de treino e teste (principais indicadores da maioria dos trabalhos de classificação de padrões), *overfitting*, custo computacional e número de atributos selecionados.

O presente trabalho utiliza como definição para *overfitting* a “piora” da acurácia obtida com os dados de teste em relação à acurácia obtida com os dados de treinamento com o mesmo classificador (Gütlein, 2006). O trabalho avalia se o melhor modelo (em termos de acurácia de treino) obtido na fase de treinamento tem perda de desempenho de acurácia ao ser aplicado aos dados de teste. O algoritmo de classificação utilizado nos *wrappers* é o *NB*.

Como se faz necessária para a realização dos experimentos a escolha de uma informação objetiva (mensurável quantitativamente), o critério de comparação adotado aqui para mensuração do *overfitting* é o quociente entre a diferença das acurácias de teste e de treinamento e a acurácia de treinamento. A ideia é avaliar se o modelo obtido na fase de treinamento está “superajustado” somente aos dados de treinamento (e o quanto seus resultados pioram ao aplicá-lo a dados que ele ainda não havia se deparado anteriormente) ou se o modelo tem boa capacidade de generalização, isto é, se

ele apresenta pouca ou nenhuma perda de desempenho ao ser aplicado os dados de teste.

Para tanto, após a execução do *wrapper* (que recebe como entrada os dados de treino e, apresenta como saída, os atributos selecionados e a acurácia de treino), o mesmo classificador "empacotado" no *wrapper* é executado com outros dados (dados de teste) utilizando somente os atributos selecionados. Espera-se que quanto menor essa acurácia de teste em relação à acurácia de treino (do melhor modelo de treinamento), pior a capacidade de generalização proporcionada pelos atributos selecionados. Mesmo que os atributos selecionados possam ser os mais adequados para aquele classificador com aqueles dados de treino, ao utilizá-lo para realizar classificações com dados diferentes (como é comum no mundo real), os atributos podem não se mostrar tão adequados assim.

O presente trabalho considera que há superajustamento sempre que o valor do *overfitting* for negativo, isto é, todas as vezes que a acurácia de teste (para o mesmo seletor de atributos, o mesmo classificador e os mesmos dados) piorar em relação à de treino e, quanto mais negativo for seu valor, maior o grau (intensidade) de *overfitting*.

Esta medida justifica-se, pois, usuários podem ser induzidos a pensar (equivocadamente) que, bons resultados com dados de treino são garantia de bons resultados com quaisquer dados diferentes. A medida tenta mostrar se os resultados com os dados de treino para um determinado problema podem, ou não, ser generalizados (e em que grau) para dados de teste diferentes dos dados de treino, como é comum no mundo real.

Obviamente, a importância deste critério e a sua melhor forma de utilização para mensuração do *overfitting* em uma situação real vão depender do objetivo do usuário. O usuário pode, por exemplo, usar a acurácia de teste como primeiro critério e o *overfitting* como critério de desempate. O importante é que o usuário esteja alertado de que bons resultados com dados de treino não são garantia de bons resultados com dados de teste e que disponha desta medida para o caso de precisar tomar uma decisão com base nesta informação.

O critério de avaliação adotado neste trabalho para medição do custo computacional é o número de vezes que a função de *fitness* foi chamada para avaliar cada nova solução gerada (independente de ter sido incluída ou não na população) até se alcançar a melhor solução obtida (subconjunto de atributos que forneceram a melhor acurácia de treinamento).

Além disso, o presente trabalho aproveita para mostrar a fração de atributos selecionados em cada caso, obtida pelo quociente entre o número de atributos selecionados e o total de atributos disponíveis no conjunto de dados original. Quanto menor esta fração, mais compreensíveis tornam-se os dados e mais rápida a classificação dos dados de teste.

Não houve a necessidade de execução da seleção de atributos através de meta-heurísticas para os estudos de caso reais, tendo em vista a pequena quantidade de atributos existente em cada um de seus conjuntos de dados. Para estes casos, uma busca exaustiva foi o suficiente. Assim, os experimentos de seleção de atributos por meio de meta-heurísticas de AE foram executados somente para os demais seis conjuntos de dados nominais (os *benchmarks*), contendo um mínimo de 22 e um máximo de 67 atributos de predição. O número de possibilidades de combinações de atributos para o menor conjunto é de 4.194.304 (2^{22}) e para o maior 1,47 x 10^{20} (2^{67}). Algumas características destes conjuntos de dados são mostradas na Tabela 5.6.

Tabela 5.6 – Conjuntos de dados de teste utilizados e suas características.

Conjunto de dados	# Atributos de predição	# Instâncias	# Classes	Tipos dos dados	Predomínio da classe mais prevalente (%)	Valores ausentes (%)	Valores distintos por atributo (média)
SPECT Heart	22	267	2	Nominais	79,40%	0,00%	2,0
Soybean Large	34	307	19	Nominais	13,03%	21,75%	2,9
Kr-vs-Kp	35	3.196	2	Nominais	52,22%	0,00%	2,0
Promoter	57	106	2	Nominais	50,00%	0,00%	4,0
Splice Junction	60	3.190	3	Nominais	50,00%	0,00%	4,0
Audiology Std.	67	226	23	Nominais	21,24%	2,09%	2,3

A Figura 5.8 apresenta a distribuição dos conjuntos de dados em termos do número de atributos de predição e do número de instâncias, quando comparados entre si. Os conjuntos *Soybean Large* e *SPECT Heart* encontram-se no quadrante inferior esquerdo, referente aos menores conjuntos de dados,

tanto em termos de número de instâncias como de número de atributos. Já o conjunto *Kr-vs-Kp* está no quadrante inferior direito por se tratar de um conjunto de dados com muitas instâncias e poucos atributos. No quadrante superior esquerdo, referente aos conjuntos de dados com poucas instâncias, mas com muitos atributos, encontram-se os conjuntos de dados *Audiology Standardized* e *Promoter*. Finalmente, o conjunto de dados *Splice* está situado no quadrante superior direito, pois se trata de um conjunto de dados com muitos atributos e muitas instâncias.

Figura 5.8 – Distribuição dos conjuntos de dados *benchmark* em termos do número de atributos de predição e do número de instâncias.

Quanto ao número de classes, os conjuntos *SPECT Heart*, *Kr-vs-Kp* e *Promoter* tratam-se das únicas bases de dados dicotômicas. O *Splice Junction* possui 3 classes e, os demais conjuntos, *Soybean Large* e *Audiology Standardized*, possuem respectivamente, 19 e 23 classes. O único conjunto de dados totalmente balanceado em termos do número de instâncias por classe é o *Promoter*, sendo que o *Kr-vs-Kp* é quase totalmente balanceado. O *SPECT Heart*, que possui uma classe com 79,40% das instâncias, é o conjunto de dados que apresenta a maior predominância de uma classe sobre as demais.

Os únicos conjuntos de dados que possuem uma quantidade considerável de valores ausentes (em inglês, *missing values*) são, justamente, os que possuem o maior número de classes: *Soybean Large* com 21,75%;

Audiology Standardized com 2,09%. Os demais conjuntos de dados não possuem valores ausentes. O conjunto de dados *SPECT Heart* é composto somente por atributos de predição binários. *Splice Junction* e *Promoter*, por serem conjuntos de dados de *DNA*, possuem uma média de quatro valores distintos (A, C, G e T) por atributo. Os demais conjuntos de dados possuem uma média entre dois e três valores distintos por atributo.

O AG[16] utilizado é de representação binária, o cruzamento é feito somente com um ponto de corte e o critério de parada adotado foi o número máximo de gerações[17], cujo valor foi fixado em 100 gerações para todas as simulações. A opção pelo número máximo de gerações como critério de parada para o presente trabalho tem a intenção de evitar que o algoritmo permaneça em execução indefinidamente por um tempo muito longo, além de tornar mais simples a comparação visual dos gráficos de resultados.

As simulações foram realizadas com diferentes tamanhos de população, tendo sido testadas populações formadas por 10, 20 e 40 indivíduos. As probabilidades de cruzamento e de mutação também foram variadas: 60% e 80% para cruzamento e 1% e 5% para mutação. A seleção é “sorteada” por roleta, sendo que as porções da roleta são distribuídas entre os indivíduos proporcionalmente ao *fitness* de cada um (os indivíduos com porções maiores devem ser selecionados mais vezes do que os com porções menores). Este AG usa estratégia elitista, isto é, é assegurado que o melhor indivíduo de uma geração permaneça na geração seguinte. O pseudocódigo do AG utilizado neste trabalho é mostrado na Figura 5.9.

---

[16] A implementação do AG utilizada é a de Potvin (1993).

[17] Geração é definida aqui como cada vez que uma população “atual” é substituída por uma nova população.

Gerar aleatoriamente uma população inicial P de N indivíduos usando distribuição uniforme (o número de dimensões de cada indivíduo é o número de atributos originais do conjunto de dados, n)

Repetir:

Para: cada geração (l) da população:

//seleção

Avaliar o *fitness* dos N indivíduos da população P

Selecionar N indivíduos (não necessariamente distintos) proporcionalmente ao seu *fitness* (seleção por roleta)

//cruzamento

Combinar aleatoriamente dois a dois os N indivíduos selecionados (obedecendo uma probabilidade de cruzamento predefinida) para geração da nova população P também formada por N indivíduos

//mutação

Trocar os valores dos bits dos N indivíduos da população P obedecendo uma probabilidade de mutação predefinida

Fim

Até que o critério de parada (número de gerações = 100) seja satisfeito

Figura 5.9 – Pseudocódigo do AG.

Para o *PBIL* e o *BBIL* também foram testadas populações de 10, 20 e 40 indivíduos por geração com um número máximo, também fixo, de 100 gerações. O tamanho de população e o número de gerações foram escolhidos os mesmos para todos os algoritmos.

Valores entre 0,1% e 10% para a taxa de aprendizagem do *PBIL* têm se mostrado satisfatórios para uma grande variedade de problemas da literatura (Coelho e Grebogi, 2010; Folly e Venayagamoorthy, 2009; Ventresca e Tizhoosh, 2008). Assim, para o presente trabalho, a taxa de aprendizagem do *PBIL* foi variada usando os valores de 0,1%, 1% e 10%, mesmos valores testados para a taxa de aprendizagem inicial do *BBIL*. Já a taxa de rejeição do *BBIL* foi testada com valores de 1% e 10%. Os critérios de parada são o número máximo de gerações e a convergência (quando todos os componentes do vetor de probabilidades forem menores que 0,05 ou maiores que 0,95). Os

pseudocódigos do *PBIL* e do *BBIL* utilizado neste trabalho são mostrados nas Figuras 5.10 e 5.11, respectivamente.

Criar um vetor de probabilidades p com n dimensões

Iniciar o vetor p atribuindo 0,5 para cada um dos seus n componentes

Gerar uma população inicial P de M indivíduos obedecendo a distribuição definida no vetor p

Repetir:

Para: cada geração (l) da população:

//seleção

Avaliar o *fitness* dos M indivíduos da população P

Selecionar os $N \leq M$ indivíduos com maior *fitness* da população P

//atualização do vetor de probabilidades

Atualizar cada componente do vetor p com base nos N indivíduos selecionados utilizando como tamanho do passo uma taxa de aprendizagem $\alpha \in (0,1]$ pré-definida:

$$p_l(x) = (1-\alpha)p_{l-1}(x) + \alpha \frac{1}{N} \sum_{k=1}^{N} x_{k:M}^{l-1}$$

Fim

Até que o critério de parada (número de gerações = 100 ou todos os n componentes do vetor p menores que 0,05 ou maiores que 0,95) seja satisfeito

Figura 5.10 – Pseudocódigo do *PBIL*.

Criar um vetor de probabilidades p com n dimensões

Iniciar o vetor p atribuindo 0,5 para cada um dos seus n componentes

Gerar uma população inicial P de M indivíduos obedecendo a distribuição definida no vetor p

Repetir:

Para: cada geração (l) da população:

//seleção

Avaliar o *fitness* dos M indivíduos da população P

Selecionar os $N \leq M$ indivíduos com maior *fitness* da população P

//atualização da taxa de aprendizagem

Se: $f(pbest_l) > f(gbest)$

$gbest = pbest_l$

$\alpha = \alpha_{inicial}$ // $\alpha_{inicial} \in (0,1]$

Senão: $f(pbest_l) \leq f(gbest)$

$\alpha = \alpha - \alpha.\beta$ //$\beta \in (0,1]$

Fim

//atualização do vetor de probabilidades

Atualizar cada componente do vetor p com base nos N indivíduos selecionados utilizando como tamanho do passo a taxa de aprendizagem α:

$$p_l(x) = (1-\alpha)p_{l-1}(x) + \alpha\frac{1}{N}\sum_{k=1}^{N} x_{k:M}^{l-1}$$

Fim

Até que o critério de parada (número de gerações = 100 ou todos os n componentes do vetor p menores que 0,05 ou maiores que 0,95) seja satisfeito

Figura 5.11 – Pseudocódigo do *BBIL*.

Como os algoritmos de seleção de atributos são mais computacionalmente intensivos, para efeitos de simplificação, esta seção compara os algoritmos de seleção de atributos apresentados utilizando somente o classificador *NB*. Os melhores resultados em termos de acurácia média de treinamento e de teste para cada conjunto de dados utilizando cada

algoritmo de seleção de atributos são apresentados na Tabela 5.7 e na Figura 5.12.

Nesta tabela, o melhor resultado de cada indicador para cada base de dados é evidenciado em negrito e as situações em que ocorreram *overfitting* são marcadas em cor vermelha (e as que não ocorreram, em cor azul).

Tabela 5.7 – Parametrização, acurácias e *overfitting* do modelo de seleção de atributos com melhor acurácia média de treinamento.

Conjunto de dados	Seletor de atributos	Parâmetros dos algoritmos de seleção de atributos	A_{TR} = Acurácia média de treinamento (%)	A_{TE} = Acurácia média de teste (%)	Aumento da acurácia média[18] $(A_{TE} - A_{TR}) / A_{TR}$
Detecção de afundamentos	Busca exaustiva		80,6993	**81,7021**	**+1,2%**
Diagnóstico de causas	Busca exaustiva		89,3011	**89,3882**	**+0,0%**
Diagnóstico de rinoconjuntivite	Busca exaustiva		74,3204	**80,4483**	**+8,2%**
SPECT	AG	*Tp*[19] = 10; *Pc*[20] = 0,8; *Pm*[21] = 0,01	88,1001	77,5350	-12,0%
	PBIL	*Tp* = 40; *Ta*[22] = 0,1	87,4061	72,3075	-17,3%
	BBIL	*Tp* = 40; *Tai*[23] = 0,1; *Tr* [24]= 0,01	**87,6775**	**84,1621**	**-4,0%**
Soybean	AG	*Tp* = 20; *Pc* = 0,6; *Pm* = 0,01	**95,0257**	83,0426	-12,6%
	PBIL	*Tp* = 40; *Ta* = 0,1	93,4542	92,8064	**-0,8%**
	BBIL	*Tp* = 40; *Tai* = 0,1; *Tr* = 0,1	93,8257	**92,9458**	-1,0%
Kr-vs-Kp	AG	*Tp* = 20; *Pc* = 0,6; *Pm* = 0,01	**96,6691**	85,9277	-11,2%
	PBIL	*Tp* = 40; *Ta* = 0,1	96,4437	90,7610	-5,8%
	BBIL	*Tp* = 40; *Tai* = 0,1; *Tr* = 0,1	96,6464	**92,1615**	**-4,6%**
Promoter	AG	*Tp* = 10; *Pc* = 0,6; *Pm* = 0,01	96,3928	**100**	**+3,7%**
	PBIL	*Tp* = 40; *Ta* = 0,1	97,4046	98,8889	+1,5%
	BBIL	*Tp* = 40; *Tai* = 0,1; *Tr* = 0,1	**97,5339**	**100**	+2,6%
Splice	AG	*Tp* = 10; *Pc* = 0,8; *Pm* = 0,01	**88,1001**	81,7016	-7,3%
	PBIL	*Tp* = 40; *Ta* = 0,1	87,3900	80,1128	-8,3%
	BBIL	*Tp* = 40; *Tai* = 0,1; *Tr* = 0,01	87,1410	**83,5686**	**-4,0%**
Audiology	AG	*Tp* = 40; *Pc* = 0,8; *Pm* = 0,05	83,9162	75,3126	-10,3%
	PBIL	*Tp* = 40; *Ta* = 0,1	92,3263	96,1829	**+4,2%**
	BBIL	*Tp* = 40; *Tai* = 0,1; *Tr* = 0,1	**93,7533**	**96,5179**	+2,9%

[18] O presente trabalho convenciona que valores negativos indicam a presença de *overfitting*.

[19] Tamanho da população

[20] Probabilidade de cruzamento

[21] Probabilidade de mutação

[22] Taxa de aprendizagem

[23] Taxa de aprendizagem inicial

[24] Taxa de rejeição

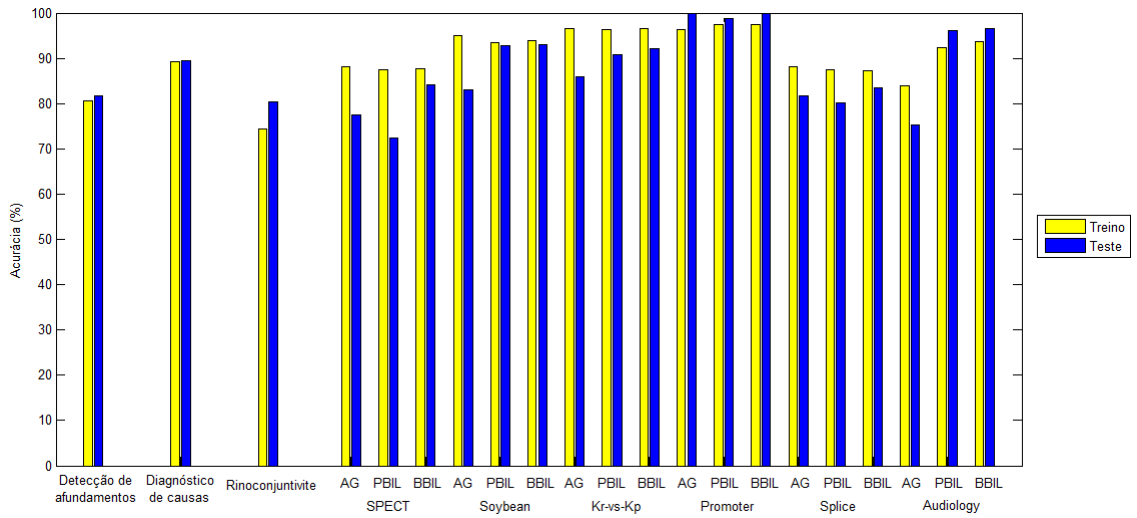

Figura 5.12 – Seleção de atributos.

Para efeitos de comparação e maior percepção das melhorias de desempenho proporcionadas pela seleção de atributos, a Tabela 5.8 e a Figura 5.13 apresentam os resultados das acurácias com todos os atributos originais de cada conjunto de dados e as acurácias de teste do melhor método de seleção de atributos para cada um dos conjuntos de dados analisados (obtidas da Tabela 5.7).

Tabela 5.8 – Comparação entre as acurácias de teste do classificador *NB* usando todos os atributos originais e usando somente os atributos selecionados.

Conjunto de dados	A_{TET} = Acurácia média com todos os atributos (%)	A_{TES} = Acurácia média somente com os atributos selecionados (%)	Método de seleção de atributos escolhido	Aumento da acurácia média de teste após a seleção de atributos (A_{TES} / A_{TET}) - 1
Detecção de afundamentos	73,2596	81,7021	Busca exaustiva	+11,5%
Diagnóstico de causas	84,2002	89,3882	Busca exaustiva	+6,2%
Diagnóstico de rinoconjuntivite	64,5446	80,4483	Busca exaustiva	+24,6%
SPECT	77,2067	84,1621	*BBIL*	+9,0%
Soybean	81,9495	92,9458	*BBIL*	+13,4%
Kr-vs-Kp	86,8401	92,1615	*BBIL*	+6,1%
Promoter	91,6206	100	AG e *BBIL*	+9,1%
Splice	80,0214	83,5686	*BBIL*	+4,4%
Audiology	75,5837	96,5179	*BBIL*	+27,7%

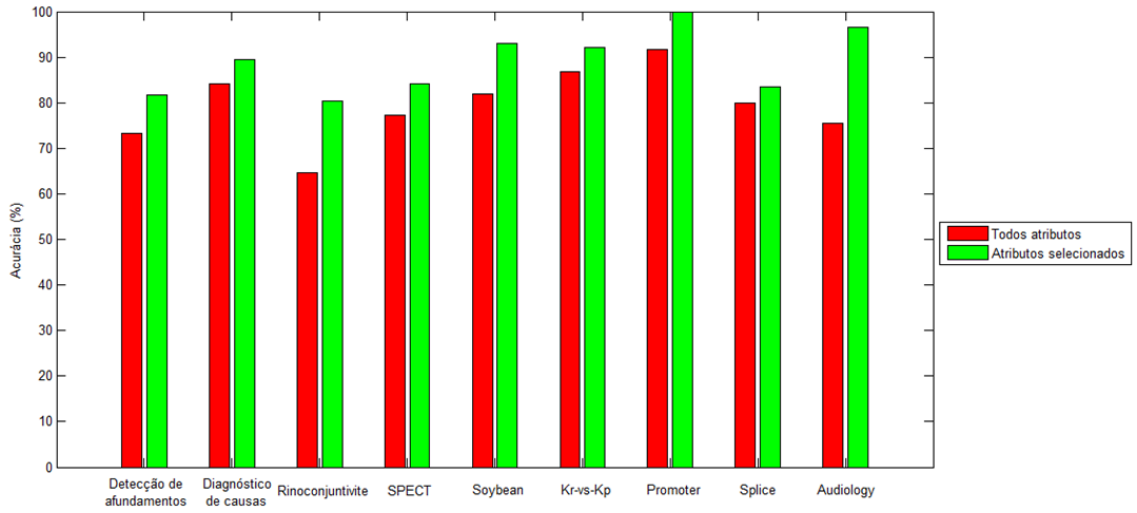

Figura 5.13 – Comparação entre as acurácias de teste do classificador *NB* usando todos os atributos originais e usando somente os atributos selecionados

A Tabela 5.8 mostra que os conjuntos de dados em que o classificador *NB* apresentou pior desempenho em termos de acurácia de teste utilizando todos os atributos originais são o diagnóstico de rinoconjuntivite, a detecção de afundamentos, o *Audiology Standardized* e o *SPECT Heart*, todos eles com acurácia inferior a 80%. O conjunto de dados para o qual o *NB* apresentou o melhor acurácia de teste usando todos os atributos foi o *Promoter* com 91,6206%.

Conforme esperado, ao compararem-se as Tabela 5.7 e 5.8, nota-se que a seleção de atributos por qualquer um métodos utilizados no presente trabalho (busca exaustiva, AG, *PBIL* e *BBIL*) melhorou o desempenho do classificador *NB* em termos de acurácia de treino e de teste para todos os conjuntos de dados em comparação com a aplicação direta do mesmo classificador sobre todos os atributos originais. Em termos comparativos, a maior melhora de acurácia de teste após a seleção de atributos ficou por conta do conjunto *Audiology Standardized*, com 27,7% de aumento da acurácia média de teste após a seleção de atributos por *BBIL*. Para os seis *benchmarks* analisados, o *BBIL* foi o melhor método de seleção de atributos, superando o *PBIL* em todas as comparações e empatando com o AG somente para o conjunto de dados *Promoter*.

Com relação à melhor configuração dos parâmetros de cada algoritmo meta-heurístico de seleção de atributos para cada conjunto de dados, para o

AG, o tamanho da população mais comum foi de 10 indivíduos por geração (em três dos seis casos), sendo que em dois conjuntos de dados (*Soybean Large* e *Kr-vs-Kp*) a melhor configuração exigiu 20 indivíduos por geração e, para o conjunto de dados *Audiology Standardized* foi necessária uma população de 40 indivíduos. Para dois dos três casos em que bastaram 10 indivíduos, o AG funcionou melhor com uma probabilidade de cruzamento de 80%, enquanto que nas situações com 20 indivíduos por geração o melhor desempenho foi para uma probabilidade de cruzamento de 60%. Somente para o conjunto de dados *Audiology Standardized* a melhor parametrização para a probabilidade de mutação foi de 5%, enquanto que para os demais conjuntos, foi melhor uma probabilidade de mutação de apenas 1%.

Para o *PBIL*, uma taxa de aprendizagem de 10% e uma população de 40 indivíduos apresentou o melhor desempenho em todos os casos. No caso do *BBIL*, uma população de 40 indivíduos e uma taxa de aprendizagem inicial de 10% também proporcionou melhores resultados para todos os casos. Quanto à taxa de rejeição, a melhor configuração foi 10% em cinco dos seis casos e 1% na outra situação (*SPECT Heart*).

Os resultados experimentais obtidos no presente trabalho confirmam Frölich *et al*. (2003) e Loughrey e Cunningham (2004) ao mostrar a grande propensão do AG em causar *overfitting*. Embora o AG tenha apresentado a melhor acurácia de treinamento em três dos seis conjuntos de dados analisados, ele obteve a melhor acurácia de teste somente em um único conjunto de dados e causou *overfitting* para cinco dos seis (só não apresentou para o conjunto de dados *Promoter*, justamente o único caso em que ele apresentou a melhor acurácia de teste).

Pela Tabela 5.7, o AG só não apresentou a pior acurácia de teste em três situações (*Promoter*, *SPECT* e *Splice Junction*). Embora talvez haja uma maior propensão a *overfitting* por parte do AG em conjuntos de dados nominais, em termos absolutos sua capacidade de generalização pode possuir também alguma relação com a quantidade de atributos disponíveis no conjunto de dados original, pois para os conjuntos de dados com menos atributos (com até 35 atributos) seu *overfitting* foi superior a 10%. Em termos relativos, a situação parece ser oposta, o AG causa *overfitting* para o conjunto de dados

que possui mais atributos (*Audiology Standardized*), enquanto que os demais algoritmos não causaram *overfitting* para estes mesmos conjuntos.

Em contrapartida, o *PBIL*, apesar de não ter obtido nem a maior acurácia de treinamento, nem a maior acurácia de teste, para nenhum dos conjuntos de dados analisados, apresentou *overfitting* em três dos seis conjuntos (só não apresentou para o *Audiology Standardized* e o *Promoter*). O *BBIL* também apresentou *overfitting* exatamente nos mesmos quatro conjuntos de dados que o *PBIL*, contudo obteve a melhor acurácia de teste para todos os conjuntos de dados. O *BBIL* obteve ainda a melhor acurácia de treino em quatro conjuntos de dados, tendo sido superior ao *PBIL* neste critério em todos, exceto para o conjunto *Splice Junction*.

Além do *BBIL* ter apresentado *overfitting* em menos situações que o AG, os *overfittings* daquele foram consideravelmente mais baixos se comparados com os apresentados por este e, mesmo nas situações em que o *BBIL* apresentou *overfitting*, sua acurácia de teste foi sempre superior se comparadas ao AG. O *PBIL* e o *BBIL* foram amplamente superiores ao AG para os problemas analisados em termos de acurácia de teste e *overfitting* (com ligeira vantagem para o *BBIL* em relação ao *PBIL* padrão, sobretudo em termos de acurácia). Esta superioridade só não foi verificada nos conjuntos de dados *Promoter* e *Splice Junction*.

As Figuras 5.14 a 5.19 apresentam graficamente a evolução do maior *fitness* (linhas em cor azul), do pior *fitness* (cor vermelha) e do *fitness* médio (cor preta) para os algoritmos AG, *PBIL* e *BBIL*, bem como a convergência dos algoritmos *PBIL* e *BBIL*. Os símbolos de “+” em cor verde correspondem a passos de aprendizagem normais e os símbolos de “x” em cor lilás a passos de aprendizagem reduzidos devido a rejeições ocorridas (ao conhecimento do indivíduo mais sábio até o momento). As Figuras 5.14 a 5.19 referem-se, respectivamente, aos conjuntos de dados *SPECT Heart*, *Soybean Large*, *Kr-vs-Kp*, *Promoter*, *Splice Junction* e *Audiology Standardized*.

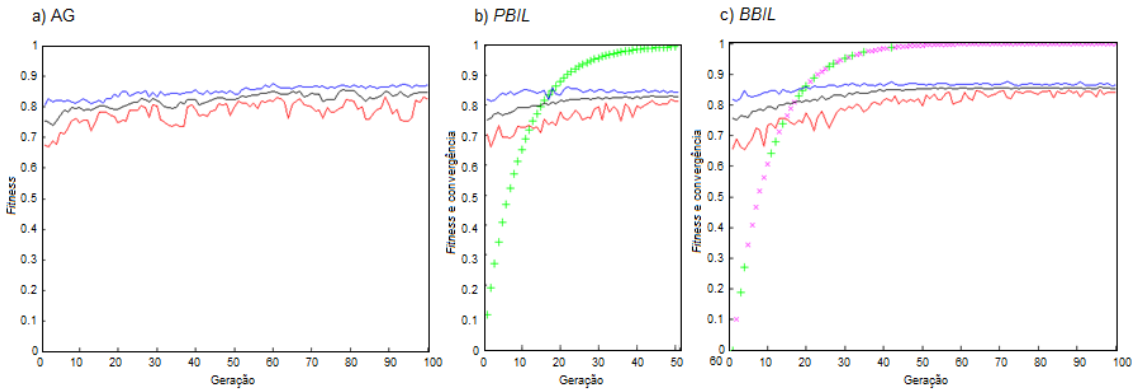

Figura 5.14 – Evolução do *fitness* e convergência da seleção de atributos para o conjunto de dados *SPECT Heart*.

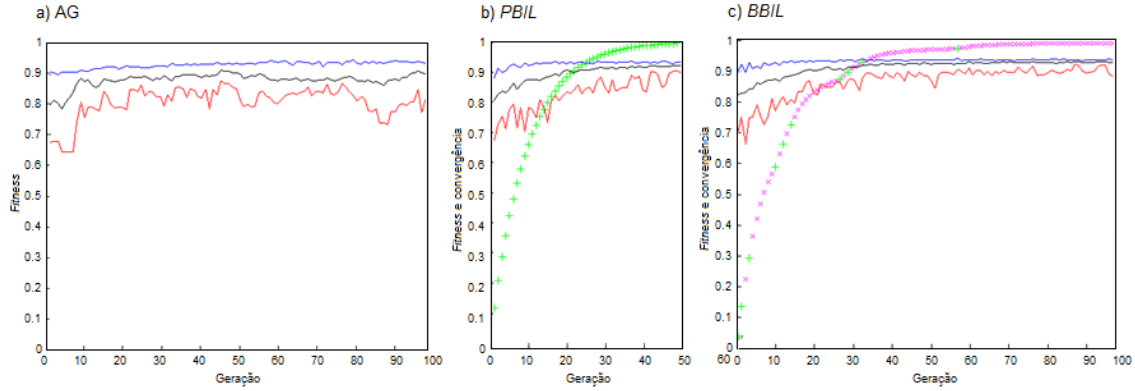

Figura 5.15 – Evolução do *fitness* e convergência da seleção de atributos para o conjunto de dados *Soybean Large*.

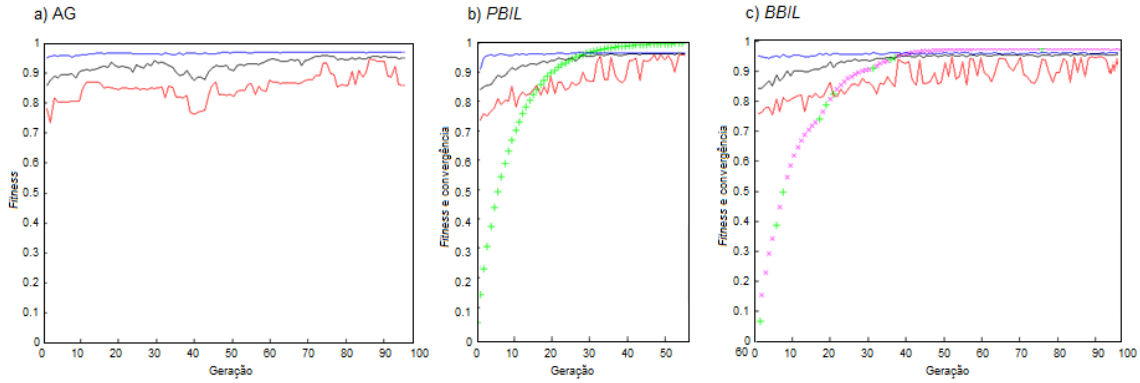

Figura 5.16 – Evolução do *fitness* e convergência da seleção de atributos para o conjunto de dados *Kr-vs-Kp*.

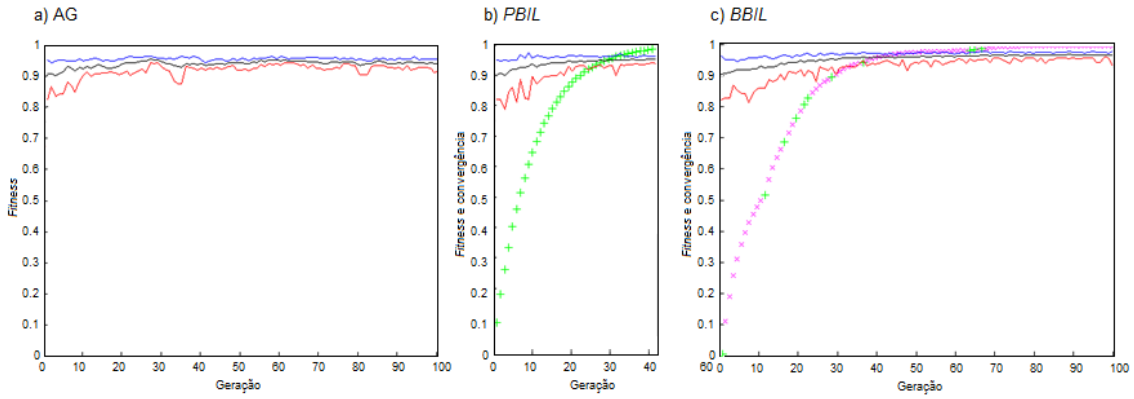

Figura 5.17 – Evolução do *fitness* e convergência da seleção de atributos para o conjunto de dados *Promoter*.

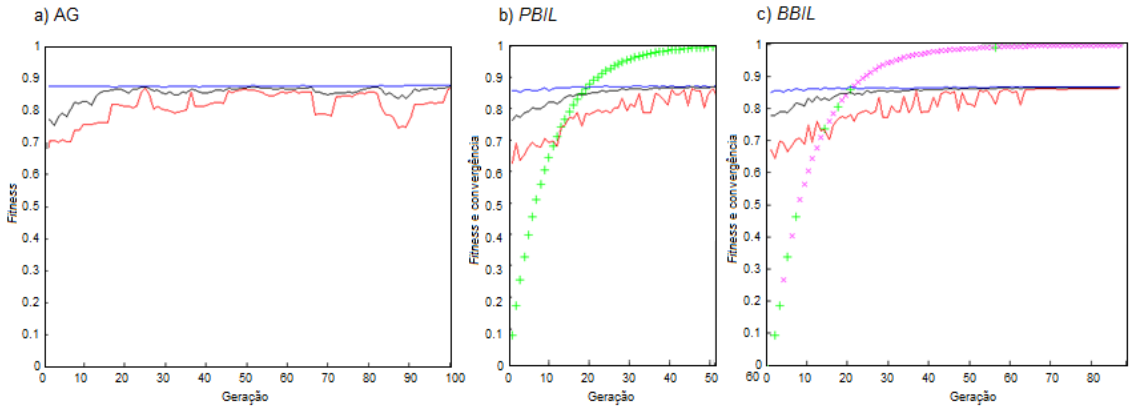

Figura 5.18 – Evolução do *fitness* e convergência da seleção de atributos para o conjunto de dados *Splice Junction*.

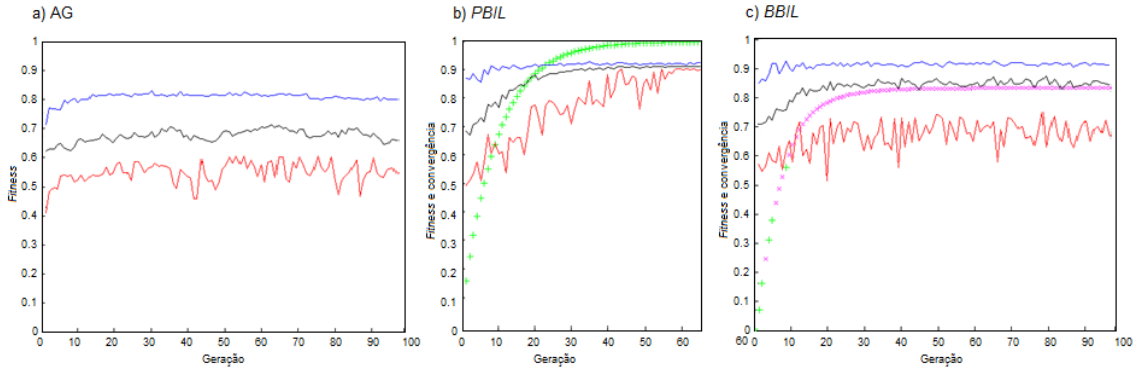

Figura 5.19 – Evolução do *fitness* e convergência da seleção de atributos para o conjunto de dados *Audiology Standardized*.

Para a maioria dos conjuntos de dados, as curvas de evolução do melhor *fitness* do AG tiveram mais crescimento nas gerações iniciais e pouco evoluíram (caso do *SPECT Heart*) ou, na maioria dos casos, praticamente se

estabilizaram nas gerações subsequentes. A maior variância de *fitness* para o AG ocorreu para o conjunto de dados *Audiology Standardized* e a menor para o *Promoter*. Na maior parte dos casos, a variância entre o melhor e o pior *fitness* do AG ao longo das gerações era relativamente estável com algumas poucas aproximações significativas entre ambas as curvas (curva azul e curva vermelha), como por exemplo, as gerações 25, 36, 48 e 98 do *Splice Junction*.

Já as curvas de *fitness* do *PBIL* e do *BBIL* mostraram um comportamento mais robusto de diminuição da variância ao longo das gerações. A convergência dos componentes do vetor de probabilidades ao longo das gerações torna os indivíduos das últimas gerações cada vez mais parecidos entre si, pois as populações de cada geração são criadas com base na distribuição de probabilidades determinada pelo vetor de probabilidades.

Com relação às curvas de convergência, taxas de rejeição maiores aumentam o impacto dos AC no resultado do *BBIL* e são mais facilmente visualizadas graficamente do que taxas de rejeição menores. Observando as Figuras 5.15, 5.16 e 5.17, é fácil notar a atuação do componente cultural através das “quebras no gráfico” nos pontos em que o conhecimento de um novo indivíduo mais sábio é aceito pela população (situações em que aparecem o símbolo de “+” em cor verde). Estes conjuntos de dados (*Soybean Large*, *Kr-vs-Kp* e *Promoter*) utilizaram taxas de rejeição superiores às dos demais conjuntos de dados (10% contra 1% dos outros conjuntos de dados). A única exceção ficou por conta do conjunto de dados *Audiology Standardized* que, mesmo utilizando uma taxa de rejeição igual a 10%, graficamente a atuação do componente cultural não pôde ser claramente percebida. Aparentemente, o *BBIL* encontrou uma boa solução prematuramente sem que todos os componentes do vetor de probabilidades tenham convergido antes do número máximo de gerações.

A Tabela 5.9 apresenta o número de avaliações de *fitness* até o atendimento ao critério de parada e a fração de atributos selecionados para cada conjunto de dados com cada algoritmo (usando a mesma configuração de parâmetros apresentada na Tabela 5.8). As Figuras 5.14 a 5.19 corroboram com a Tabela 5.9 ao mostrar que para todos os conjuntos de dados, como era de se esperar, o número de gerações até a convergência do *PBIL* foi sempre

menor que a do *BBIL*, já que o *PBIL* tende a convergir antes do *BBIL* sempre que adota uma taxa de aprendizagem fixa igual a dele.

Tabela 5.9 – Custo computacional e fração de atributos selecionados.

Conjunto de dados	Seletor de atributos	Número de avaliações de *fitness*[25]	N_S = Número de atributos selecionados	N_T = Número total de atributos	Fração de atributos selecionados (N_S/N_T)
SPECT	AG	**1.000**	**10**	22	**45,45%**
	PBIL	2.040	13	22	59,09%
	BBIL	4.000	**10**	22	**45,45%**
Soybean	AG	**2.000**	**19**	35	**54,29%**
	PBIL	**2.000**	21	35	60,00%
	BBIL	4.000	22	35	62,86%
Kr-vs-Kp	AG	**2.000**	19	36	52,78%
	PBIL	2.360	19	36	52,78%
	BBIL	4.000	**18**	36	**50,00%**
Promoter	AG	**1.000**	47	57	82,46%
	PBIL	1.640	**40**	57	**70,18%**
	BBIL	4.000	**40**	57	**70,18%**
Splice	AG	**1.000**	48	60	80,00%
	PBIL	2.040	**35**	60	**58,33%**
	BBIL	3.400	36	60	60,00%
Audiology	AG	4.000	51	67	76,12%
	PBIL	**2.640**	**33**	67	**49,25%**
	BBIL	4.000	41	67	61,19%

De maneira geral, com relação ao custo computacional, os experimentos conferem uma vantagem ao AG. Com relação à fração de atributos selecionados, o *PBIL* e o *BBIL* parecem obter melhores resultados que o AG para conjuntos de dados nominais com mais atributos, tais como *Audiology Standardized*, *Splice Junction*, *Promoter* e *Kr-vs-Kp*. A seleção de um subconjunto de atributos menor a partir de conjuntos com mais atributos permite a estes algoritmos gerar resultados mais compreensíveis ao usuário justamente em problemas de maior complexidade.

[25] Número de vezes que a função de *fitness* foi chamada para avaliar cada nova solução gerada (independente de ter sido incluída ou não na população).

5.4 TRANSFORMAÇÃO POR *GDA* PÓS SELEÇÃO DE ATRIBUTOS

Muito embora a metodologia proposta na seção 4.1 tenha sido a aplicação da seleção de atributos sobre os *factor scores* resultantes da transformação por *MCA*, isto é, após a aplicação deste tipo de transformação, a presente seção avalia também a aplicação das transformações *GDA* (*PCA* e *MCA*) após a aplicação de algoritmos de seleção de atributos.

A Tabela 5.10 e a Figura 5.20 apresentam uma análise comparativa da aplicação (ou não) de transformações *GDA* sobre os novos conjuntos de dados gerados após a seleção de atributos. Esta comparação também utiliza como critério de desempenho a acurácia de teste do classificador *NB*. Exceto em quatro situações (detecção de afundamentos de tensão, diagnóstico de rinoconjuntivite e *Audiology* com atributos selecionados por *PBIL* e por *BBIL*), nos demais conjuntos de dados o *MCA* foi vantajoso, mesmo após a seleção de atributos. A aplicação da transformação por *PCA* sobre os conjuntos de dados com os atributos previamente selecionados, por sua vez, não foi o melhor método em nenhuma situação como mostra a Tabela 5.10.

Tabela 5.10 – Acurácia de teste de classificação com e sem transformações *GDA* sobre os conjuntos de dados com atributos previamente selecionados.

Conjunto de dados	Seletor de atributos	Dados não transformados			Dados transformados por *PCA*			Dados transformados por *MCA*		
		Média (%)	Desvio Padrão (%)	Mediana (%)	Média (%)	Desvio Padrão (%)	Mediana (%)	Média (%)	Desvio Padrão (%)	Mediana (%)
Detecção de afundamentos	Exaustiva	**81,7021**	0,4402	81,4910	69,0192	0,5563	69,1092	66,3276	0,4822	66,2650
Diagnóstico de causas	Exaustiva	89,3882	0,4195	88,8011	84,3793	1,5612	84,5483	**99,4652**	0,4061	99,5556
Diagnóstico de rinoconjuntivite	Exaustiva	**80,4483**	1,281	79,3291	63,5093	0,6184	63,4738	64,9575	0,9098	64,4707
SPECT	AG	77,5350	0,9105	77,2754	77,5463	0,8298	77,8311	**78,3947**	0,8829	78,4131
	PBIL	72,3075	0,8714	72,9365	83,8245	0,7942	83,5428	**83,1590**	0,8801	83,1524
	BBIL	84,1621	0,8466	85,5433	84,4794	0,6222	83,9335	**84,4244**	0,5354	84,4568
Soybean	AG	83,0426	0,2717	83,1009	89,2830	0,2974	89,2978	**91,2019**	0,3018	91,0915
	PBIL	92,8064	0,2832	93,0192	95,8686	0,2870	95,8820	**96,8381**	0,3155	96,2211
	BBIL	92,9458	0,3391	92,7142	95,2659	0,2026	95,2640	**96,7168**	0,2701	96,9110
Kr-vs-Kp	AG	85,9277	0,1297	86,7254	88,3017	0,0689	88,2877	**89,0258**	0,0735	89,0395
	PBIL	90,7610	0,2010	90,2256	89,5462	0,2041	89,4977	**93,9938**	0,1002	94,0198
	BBIL	92,1615	0,1489	91,8588	91,1322	0,0843	91,0965	**93,3918**	0,1047	93,4015
Promoter	AG	**100**	0	100	95,2158	0,5799	95,2528	**100**	0	100
	PBIL	98,8889	0,0548	98,6288	93,3116	0,5040	93,3396	**100**	0	100
	BBIL	**100**	0	100	96,4793	0,4305	96,5347	**100**	0	100
Splice	AG	81,7016	0,0802	81,6281	81,1094	0,0794	81,1074	**82,0109**	0,0881	82,1416
	PBIL	80,1128	0,0769	80,1043	79,6521	0,0814	79,6804	**82,8744**	0,1001	82,9102
	BBIL	83,5686	0,0920	83,2092	79,8424	0,1219	79,8785	**83,7179**	0,0918	83,6710
Audiology	AG	75,3126	0,5150	75,0185	82,1098	0,6344	81,7845	**83,0861**	0,5224	82,9207
	PBIL	**96,1829**	0,6009	96,0106	88,9059	0,2934	89,0655	86,7217	0,4282	86,8785
	BBIL	**96,5179**	0,5918	95,9811	86,1108	1,0216	86,1906	83,5617	0,8088	83,9809

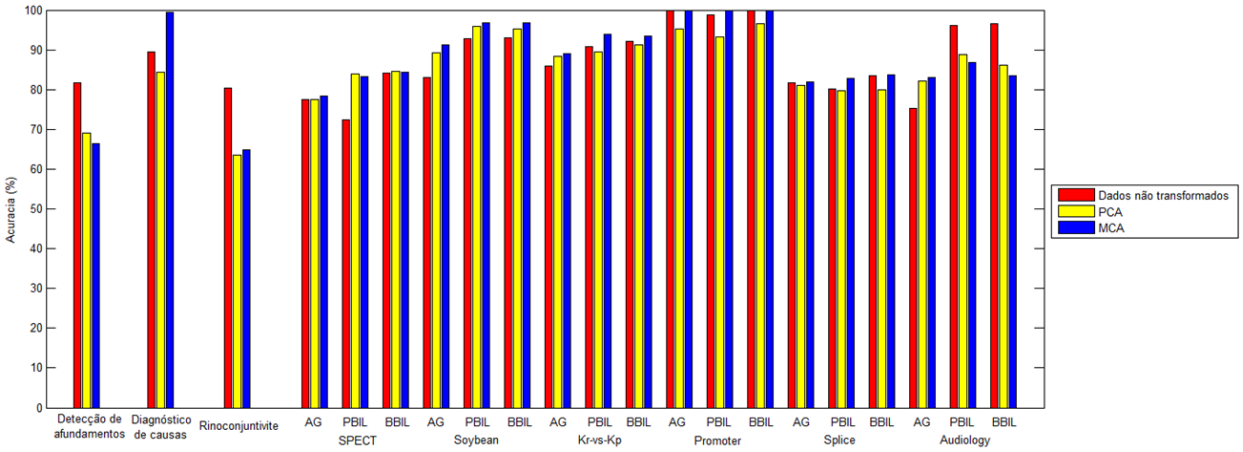

Figura 5.20 – Acurácia de teste de classificação com e sem transformações *GDA* sobre os conjuntos de dados com atributos previamente selecionados.

5.5 SELEÇÃO DE ATRIBUTOS SOBRE OS *FACTOR SCORES*

Na metodologia proposta no presente trabalho, o novo algoritmo, baseado em uma hibridização entre o *PBIL* e o AC, denominado *BBIL* (descrito em detalhes na seção 3.1.5) é utilizado para seleção de atributos sobre os dados resultantes (*factor scores*) da transformação por *MCA*. Assim como para as simulações das seções 5.3 e 5.4, a presente seção executa os testes para as situações contemplando todas as bases de dados descritas no capítulo 4 e o classificador *NB*. Além disso, o *BBIL*, como seletor de *factor scores*, é comparado com o AG e o *PBIL* com a mesma finalidade.

As parametrizações para cada seletor de atributos são as mesmas adotadas nas seções 5.3 e 5.4, com exceção dos problemas reais (detecção de afundamentos, diagnóstico de causas e diagnóstico de rinoconjuntivite alérgica) que nas ocasiões haviam utilizado busca exaustiva devido à simplicidade do problema. Como a binarização antes da obtenção dos *factor scores* do *MCA* aumentam o número de atributos do problema a ponto de tornar recomendável uma busca não exaustiva para estes três problemas, para estes conjuntos de dados foram comparados o AG, o *PBIL* e o *BBIL* com as melhores configurações obtidas durante testes de parametrização.

Os resultados da seleção de *factor scores* são apresentados na Tabela 5.11 e na Figura 5.21. Com exceção do conjunto de dados *SPECT*, os resultados para os três diferente seletores de *factor scores* (AG, *PBIL* e *BBIL*) foram todos muito próximos entre si. O *BBIL* foi o melhor em seis dos nove conjuntos de dados analisados. O conjunto de dados com o pior resultado absoluto para o *BBIL* foi o de diagnóstico de rinoconjuntivite alérgica, e os piores resultados relativos (em comparação com o AG e o *PBIL*) foram o de diagnóstico de rinoconjuntivite alérgica, o *Kr-vs-Kp* e o *Audiology Standardized*, as únicas situações em que o *BBIL* não obteve a melhor acurácia preditiva média. Os melhores resultados absolutos para o *BBIL* foram para o diagnóstico de causas de afundamentos de tensão e o *Audiology Standardized*, ambos com resultados próximos de 100% de acurácia.

Tabela 5.11 – Seleção de *factor scores* utilizando algoritmos de seleção de atributos.

Problema	AG			*PBIL*			*BBIL*		
	Média (%)	Desvio Padrão (%)	Mediana (%)	Média (%)	Desvio Padrão (%)	Mediana (%)	Média (%)	Desvio Padrão (%)	Mediana (%)
Detecção de afundamentos	82,4814	0,5660	82,3614	81,9237	0,3380	81,8979	**82,6623**	0,9632	82,3170
Diagnóstico de causas	98,4963	0,2895	98,5547	98,5967	0,3769	98,5648	**98,6546**	0,1887	98,6586
Diagnóstico de rinoconjuntivite	68,2240	0,1885	68,2497	**68,4246**	0,0539	68,4326	68,0079	0,3452	67,9319
SPECT	92,4767	0,5819	92,5026	95,1295	0,5297	95,1804	**98,5677**	0,4633	98,6032
Soybean	97,8818	0,1285	97,9012	97,8436	0,1049	97,7970	**97,9316**	0,1108	97,9092
Kr-vs-Kp	90,1646	0,0549	90,1590	**90,1889**	0,0750	90,2075	90,1002	0,0508	90,1047
Promoter	**100**	0	100	**100**	0	100	**100**	0	100
Splice	92,8799	0,0434	92,9022	92,8417	0,0638	92,8430	**92,9438**	0,0257	92,9471
Audiology	99,2307	0,0865	99,2200	**99,2309**	0,0942	99,2542	99,1277	0,1016	99,0929

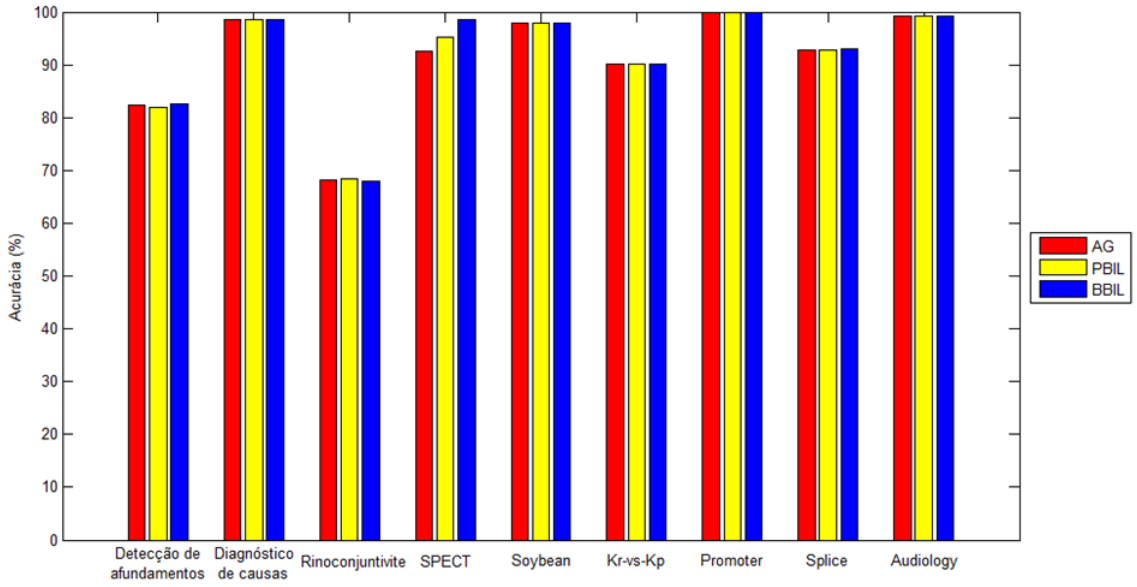

Figura 5.21 – Seleção de *factor scores* utilizando algoritmos de seleção de atributos.

Finalmente, a Tabela 5.12 apresenta uma comparação entre todas as técnicas de transformação e pré-processamento de dados aplicadas no presente trabalho para melhoria de desempenho do classificador *NB*. De maneira geral, a metodologia proposta de seleção de *factor scores* do *MCA* por *BBIL* mostrou-se a melhor técnica em cinco das nove situações à qual foi submetida. Estas cinco situações foram a detecção de afundamentos, o *SPECT Heart*, o *Soybean Large*, o *Promoter* e o *Splice Junction*. Mesmo para

as situações em que a metodologia proposta não foi a melhor, seus resultados ficaram muito próximos dos melhores métodos (exceto para o diagnóstico de rinoconjuntivite, o qual a seleção de atributos por busca exaustiva obteve um resultado consideravelmente superior à metodologia proposta).

Outro dado interessante é que a base de dados *Promoter*, que já obtinha 91,6206% para o *NB* sobre os dados brutos, obteve 100% de acurácia preditiva média em diversas situações em que foi submetida a tratamentos prévios (transformação *MCA*, seleção de atributos por AG e *BBIL*, transformação *MCA* sobre os atributos previamente selecionados por AG, *PBIL* e *BBIL* e seleção de factor scores do *MCA* com AG, *PBIL* e *BBIL*).

Tabela 5.12 – Comparação entre todas as técnicas utilizadas no presente trabalho.

Problemas	Dados brutos	*GDA*		Seleção de atributos				Transformação *GDA* sobre os atributos previamente selecionados								Seleção de *factor scores*		
		PCA	*MCA*	Exaustiva	AG	*PBIL*	*BBIL*	Exaustiva + *PCA*	Exaustiva + *MCA*	AG + *PCA*	AG + *MCA*	*PBIL* + *PCA*	*PBIL* + *MCA*	*BBIL* + *PCA*	*BBIL* + *MCA*	*MCA* + AG	*MCA* + *PBIL*	*MCA* + *BBIL*
Detec. afund.	73,2596	73,3754	81,9028	81,7021				69,0192	66,3276							82,4814	81,9237	**82,6623**
Diag. causas	84,2002	88,7352	98,4968	89,3882				84,3793	**99,4652**							98,4963	98,5967	98,6546
Rinoconjuntivite	64,5446	64,7566	67,8551	**80,4483**				63,5093	64,9575							68,2240	68,4246	68,0079
SPECT	77,2067	81,6027	81,2748		77,5350	72,3075	84,1621			77,5463	78,3947	83,8245	83,1590	84,4794	84,4244	92,4767	95,1295	**98,5677**
Soybean	89,0327	97,2652	97,7717		83,0426	92,8064	92,9458			89,2830	91,2019	95,8686	96,8381	95,2659	96,7168	97,8818	97,8436	**97,9316**
Kr-vs-Kp	86,8401	93,0810	90,1410		85,9277	90,7610	92,1615			88,3017	89,0258	89,5462	**93,9938**	91,1322	93,3918	90,1646	90,1889	90,1002
Promoter	91,6206	97,5688	**100**		**100**	98,8889	**100**			95,2158	**100**	93,3116	**100**	96,4793	**100**	**100**	**100**	**100**
Splice	82,5188	80,0480	92,8417		81,7016	80,1128	83,5686			81,1094	82,0109	79,6521	82,8744	79,8424	83,7179	92,8799	92,8417	**92,9438**
Audiology	75,5837	94,5969	99,0107		75,3126	96,1829	96,5179			82,1098	83,0861	88,9059	86,7217	86,1108	83,5617	99,2307	**99,2309**	99,1277

6 CONCLUSÕES

A maioria dos trabalhos sobre o processo *KDD* possui enfoque na etapa de mineração de dados. Entretanto, o presente trabalho mostra que tratamentos específicos sobre os dados, tais como limpeza e transformação de dados melhoram substancialmente o desempenho de algoritmos de classificação, concluindo que estas etapas são, pelo menos, tão importantes quanto a mineração de dados propriamente dita.

Mais especificamente, o presente trabalho propôs a transformação geométrica por *MCA* e a seleção de atributos por *BBIL* como técnicas de tratamentos prévios para a melhoria do desempenho de classificadores aplicados a nove diferentes bases de dados nominais e trazendo benefícios significativos ao resultado do processo *KDD* como um todo.

A transformação de dados nominais por *MCA* apresentou ganho de desempenho de classificação em relação à classificação “direta” dos dados brutos ou até mesmo quando comparado ao *PCA*. A opção pelo *MCA* em detrimento do *PCA* se justifica devido à sua habilidade em lidar com dados nominais em outros tipos de aplicações, tais como mapeamento perceptual (Wen e Chen, 2011) e análise indutiva de dados (Le Roux e Rouanet, 2005).

O *MCA* obtêm *factor scores* a serem utilizados como atributos pelos classificadores analisados. Comparativamente à seleção de atributos, o custo computacional do *MCA*, bem como das demais transformações *GDA*, é bastante baixo, isto porque são algoritmos determinísticos (algoritmos são aplicados somente uma vez para cada conjunto de dados). A representação dos dados como núvens de pontos em espaços geométricos também permite uma visualização do “comportamento” dos dados.

Em termos relativos, os melhores resultados do *MCA* foram para os conjuntos de dados de detecção de afundamentos, diagnóstico de causas e *Splice Junction* e, os piores resultados foram para o *Kr-vs-Kp* e o *SPECT Heart*. No *MCA*, o classificador utiliza os preditores numéricos obtidos pela transformação (*factor scores*), enquanto que a seleção de atributos permite que o classificador utilize de forma direta os atributos mais relevantes. Esta diferença torna a abordagem de transformação por *MCA* um pouco mais

“caixa-preta” do que a seleção de atributos *wrapper*, independente da técnica utilizada. Esta vantagem da seleção de atributos em apresentar os atributos mais relevantes por si só já serve de auxílio para tomadas de decisão.

O uso de bases de dados de domínio público foi feito em função de facilitar a reprodução dos resultados obtidos e, juntamente com os testes de permutação, pretenderam dar maior garantia de isenção aos experimentos realizados.

A aplicação das técnicas em nove diferentes bases de dados nominais (sendo três delas pertencentes a estudos de caso reais) e dois dos mais tradicionais e difundidos classificadores da área de RP (*NB* e *LDA*) evidenciaram a generalização dos resultados para diferentes situações, tendo em comum a questão de todas as bases serem compostas, exclusivamente, por dados nominais.

O presente trabalho também avaliou a acurácia preditiva, o *overfitting* de classificação, a fração de atributos selecionados e o custo computacional dos algoritmos de seleção de atributos (*wrappers*).

A extensão do *PBIL* proposta no presente trabalho, denominada *BBIL*, realiza uma busca mais parcimoniosa elevando o número de oportunidades de se deparar com soluções melhores em relação ao *PBIL* padrão e trazendo bons resultados em termos de equilíbrio no conflito entre *exploration* e *exploitation*. Embora uma maior diversidade na população, obtida por uma taxa de aprendizagem suficientemente pequena, seja fator determinante para o processo de adaptação da seleção natural, permitir que se continue explorando o espaço de soluções, mesmo após a obtenção da melhor solução, pode ser considerado um desperdício desnecessário de tempo.

Apesar do custo computacional mais elevado do *BBIL*, os experimentos mostraram que este algoritmo conseguiu obter os melhores resultados em termos acurácia de teste (e até mesmo de treino) e *overfitting*, principais critérios analisados no presente trabalho. O *BBIL* encontrou ainda subconjuntos de atributos relativamente pequenos e até mesmo sua acurácia de treinamento apresentou-se bastante elevada. Em termos relativos, os melhores resultados do *BBIL* foram para os conjuntos de dados *SPECT Heart*, *Promoter* e *Kr-vs-Kp*, bases com maior predominância da classe mais prevalente (vide Tabela 5.10).

O único conjunto de dados que o *BBIL* não apresentou bons resultados foi o *Soybean Large*, base com menor predomínio da classe mais prevalente.

Sob a perspectiva dos parâmetros, as melhores configurações para a taxa de aprendizagem do *PBIL* e a taxa de aprendizagem inicial do *BBIL* foram as mesmas para todos os conjuntos de dados. Todavia, a taxa de rejeição, único parâmetro que o *BBIL* tem a mais do que o *PBIL*, mostrou-se fortemente dependente do problema. Embora não tenha havido uma única configuração melhor para este parâmetro para todos os problemas, o *benchmark* com o menor número de atributos, *SPECT Heart*, se utilizou de uma taxa de rejeição menor (iguais a 1%), enquanto que para os demais conjuntos de dados foram necessárias taxas de rejeição maiores (iguais a 10%).

A simples diminuição da taxa de aprendizagem do *PBIL* padrão não mostrou bons resultados, tendo em vista que as taxas de aprendizagem em que o *PBIL* apresentou-se melhor foram iguais (e as mais altas simuladas) em todos os conjuntos de dados analisados. Embora o *PBIL* fique, aparentemente, mais “minucioso” durante a varredura, o aumento da quantidade de passos tem o inconveniente de elevar de modo indesejado o número de avaliações de *fitness*.

Com uma taxa de aprendizagem menor, a busca se torna mais lenta e sem a garantia de encontrar o ótimo, pois o *PBIL* padrão, tal qual o *BBIL*, não volta atrás em uma região já visitada, sendo, por esta razão, muito dependente dos passos tomados inicialmente. Embora configurar a taxa de rejeição do *BBIL* possa não ser uma tarefa totalmente trivial, a opção pelo *BBIL* com uma taxa de rejeição configurada de modo conveniente parece surtir mais efeito que utilizar o *PBIL* padrão com uma taxa de aprendizagem configurada, também, convenientemente. Outra desvantagem do *BBIL* é que, tal qual o *PBIL* padrão, ele continua muito dependente de boas escolhas nos passos iniciais, pois não volta a uma região já visitada. Em contrapartida, o *BBIL* é uma solução um pouco mais simples e com menos parâmetros que outras soluções disponíveis da literatura, o que também não deixa de ser uma vantagem do *BBIL*.

A taxa de rejeição do *BBIL* mostrou-se capaz de controlar melhor a convergência em relação ao *PBIL* ao aproveitar com maior intensidade as soluções que estão se mostrando eficientes ao longo das gerações e

atenuando o impacto das soluções que estão piorando os resultados. De maneira geral, sobre dados nominais, a seleção de atributos *wrapper* baseada em *BBIL* alia uma boa capacidade de melhorar o desempenho da tarefa de classificação em termos de acurácia e generalização proporcionada pelo *PBIL*, com uma convergência mais robusta (mas não necessariamente mais rápida) propiciada pela taxa de rejeição baseada em AC.

Com o objetivo de se beneficiar simultaneamente das vantagens de ambos os tratamentos de dados abordados no presente trabalho, foram avaliadas duas combinações entre a transformação *GDA* e a seleção de atributos. A primeira destas combinações trata-se da transformação *GDA* sobre os atributos previamente selecionados por alguma técnica de seleção de atributos e, a segunda, a seleção de *factor scores* do *MCA* utilizando algoritmos de seleção de atributos.

Os resultados dos experimentos referentes a ambas as combinações confirmam a melhoria de desempenho de classificação proporcionada pelos tratamentos realizados e atestam a superioridade da metodologia proposta, tendo sido esta superior a todas as demais técnicas de tratamento abordadas no trabalho.

Para a transformação por *MCA* sobre os conjuntos de dados com atributos previamente selecionados por *BBIL*, os resultados mostraram-se melhores nos conjuntos de dados *benchmark* com menor quantidade de atributos originais. Os melhores resultados relativos foram para os conjuntos de dados *Soybean Large* e *Kr-vs-Kp*.

A seleção de *factor scores* do *MCA* utilizando o *BBIL* “comprimiu” ao máximo a dimensionalidade dos resultados do *MCA*, sem perder a variação presente nos dados. A seção 5.5 comprovou empiricamente a superioridade do modelo proposto nesta metodologia em relação a todos os modelos mostrados no trabalho. Esta combinação obteve a melhor acurácia preditiva em cinco dos nove problemas avaliados em relação a todos os demais modelos analisados. Estes cinco problemas foram: detecção de afundamentos, *SPECT Heart*, *Soybean Large*, *Promoter* e *Splice Junction*. Com exceção do conjunto *Splice Junction*, a seleção de *factor scores* por *BBIL* mostrou-se menos suscetível ao problema da baixa quantidade de instâncias, enquanto que a aplicação do

MCA sobre os atributos previamente selecionados parece apresentar maior potencialidade para conjuntos de dados com baixa quantidade de atributos.

6.1 LIMITAÇÕES DA PESQUISA E TRABALHOS FUTUROS

Em contraste com as técnicas de seleção de atributos, as técnicas de *GDA* não escolhem um subconjunto de atributos, mas sim um conjunto de combinações lineares de atributos. Embora as transformações *GDA* possam ser muito eficientes em certos casos, os preditores extraídos por estas transformações sob a forma de combinações lineares dos atributos são muitas vezes difíceis de interpretar, de modo que os passos subsequentes a estas transformações resultem em um modelo de caixa preta.

Como aplicações futuras sugere-se avaliar as transformações *MCA* sobre dados numéricos. Com o objetivo de avaliar a eficiência do *MCA* em situações mais gerais, sugere-se, por exemplo, sua aplicação combinada a classificadores baseados em *SVM* (pertencente à abordagem das *Kernel Machines*) e em RNAs. Outra situação a se analisar é a comparação dos algoritmos propostos no presente trabalho com as mesmas bases de dados utilizadas, mas com outros algoritmos de tratamento de dados, específicos para dados nominais, como a Regressão Logística e o *Barycentric Discriminant Analysis*.

Uma limitação desta pesquisa diz respeito ao tamanho das bases de dados utilizadas, tanto em termos do número de instâncias, quanto do número de atributos. Apesar de não serem tão comuns bases de dados exclusivamente nominais com muitas instâncias e, principalmente, com muitos atributos, a submissão dos algoritmos utilizados a bases de dados maiores traria melhor aproveitamento do potencial de redução de dimensionalidade e do custo computacional dos algoritmos utilizados, características não enfatizadas no presente trabalho. Uma alternativa para viabilizar testes com bases de dados maiores seria a adoção de bases com variáveis mistas (nominais e contínuas).

Para tentar tornar o *BBIL* mais robusto, sugere-se futuramente implementar a “rejeição” por variável ao invés de implementá-la para o indivíduo como um todo. O componente cultural utilizado no presente trabalho

é apenas um exemplo de como os AC podem contribuir com a melhoria de um uma meta-heurística baseada em populações já existente. A hibridização pioneira entre *EDA* e AC, introduzida por meio do *BBIL*, pode ser estendida a novas formas de hibridização entre estas duas técnicas. Com o objetivo de analisar de forma mais ampla a capacidade de generalização do *BBIL*, sua aplicação a outros problemas de otimização, diferentes da seleção de atributos, também pode servir de objeto de estudo. A inclusão de novos componentes culturais (situacionais, históricos, normativos, topográficos e de domínio) certamente deve ser tema de investigações futuras. Outra possibilidade seria a inclusão do componente cultural em outras implementações do *EDA*, tais como *UMDA*, *BOA* e *CGA.*

Também é recomendado que no futuro as técnicas empregadas no presente trabalho, isoladamente ou de modo combinado, sejam avaliadas também para a melhoria de desempenho de outros algoritmos de classificação, bem como de outras tarefas de mineração de dados, como associação e agrupamento.

REFERÊNCIAS

Abdi, H., Williams, L. J. *Barycentric discriminant analysis*, Encyclopedia of Research Design, SAGE, Thousand Oaks, CA, USA, 2010a.

Abdi, H., Williams, L. J. *Correspondence analysis*, Encyclopedia of Research Design, SAGE, Thousand Oaks, CA, USA, 2010b.

Abegaz, T., Doizer, G., Bryant, K., Adams, J. Shelton, J., Ricanek, K., Woodard, D. L. SSGA & EDA based feature selection and weighting for face recognition. *IEEE Congress on Evolutionary Computation* (CEC), 2011.

Agência Nacional de Energia Elétrica – ANEEL. *Procedimentos de distribuição – módulo 8*, 2009.

Agresti, A. *Categorical data analysis*, 2nd ed. John Wiley & Sons, New York, 2002.

Ahmad, A. *Data transformation for decision tree ensembles*, Ph.D. thesis, The University of Manchester, Manchester, Reino Unido, 2009.

Alami, J., El Imrani, A., Bouroumi, A. A multipopulation cultural algorithm using fuzzy clustering. *Applied Soft Computing*, v. 7, n. 2, p. 506–519, 2007.

Amaldi, E., Kann, V. On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems. *Theoretical Computer Sciences*, v. 109, n. 1-2, p. 237-260, 1998.

Arinyo, R. J., Luzón, M. V., Yeguas, E. Parameter tuning of PBIL and CHC evolutionary algorithms applied to solve root identification problem. *Applied Soft Computing*, v. 11, n. 1, p. 754-767, 2011.

Associação Brasileira de Normas Técnicas – ABNT. *NBR 5460*, Sistemas Elétricos de Potência, 1992.

Athitsos, V., Sclaroff, S. Boosting nearest neighbor classifiers for multiclass recognition, *Boston University Computer Science Tech*. Report No, 2004-006, Boston, MA, USA, 2004.

Bache, K., Lichman, M. *UCI machine learning repository*. University of California, CA, USA, 2013.

Baluja, S. *Population-based incremental learning: a method for integrating genetic search based function optimization and competitive learning*. Technical Report CMU-CS-94-163, Carnegie Mellon University, Pittsburgh, PA, USA, 1994.

Baranauskas, J. A. Evaluation of feature selection by wrapping around the CN2 inducer. Encontro Nacional de Inteligência Artificial, Sociedade Brasileira de Computação, p. 315-326, 1999.

Baranauskas, J. A., Monard, M. C. Metodologias para seleção de atributos relevantes. *XIII Simpósio Brasileiro de Inteligência Artificial*, 1998.

Benzécri, J. P. *L' analyse des données*, Dunod, Paris, França, 1973.

Benzécri, J. P. *Correspondence analysis handbook*, Marcel Dekker, New York, NY, USA, 1992.

Berthold, M. R., Borgelt, C., Hoeppner, F., Klawonn, F. *Guide to intelligent data analysis: how to intelligently make sense of real data*, Series Texts in Computer Science, Springer-Verlag, 2010.

Bianchi, L., Dorigo, M., Gambardella, L. M., Gutjahr, W. J. A survey on metaheuristics for stochastic combinatorial optimization. *Natural Computing*, v. 8, n. 2, p. 239-287, 2009.

Billinton, R., Oteng-Adjei, J. Utilization of interrupted energy assessment rates in generation and transmission system planning. *IEEE Transactions on Power Systems*, v. 6, n. 3, p. 1245-1253, 1991.

Boussaïd, I., Lepagnot, J., Siarry, P. A survey on optimization metaheuristics. *Information Sciences*, v. 237, n. 10, p. 82–117, 2013.

Bougeard, S., Qannari, E. M., Fablet, C., Multiblock method for categorical variables: application to air quality in pig farms. *International Conference on Correspondence Analysis and Related Methods*, Rennes, França, 2011.

Brown, R. E. *Electric power distribution reliability*. Marcel Dekker, New York, NY, USA, 2002.

Cantú-Paz, E. Feature subset selection by estimation of distribution algorithms. *Proceedings of Genetic and Evolutionary Computation Conference*, p. 303-310, New York, NY, USA, 2002.

Caruana, R., Freitag, D. Greedy attribute selection. *11th International Conference on Machine Learning*, p. 75-83, New Brunswick, NJ, USA, 1994.

Castro, L. N. Fundamentals of natural computing: an overview. *Physics of Life Reviews*, v. 4, n. 1, p. 1-36, 2007.

Catal, C., Sevim, U., Diri, B. Practical development of an Eclipse-based software fault prediction tool using Naive Bayes algorithm. *Expert Systems with Applications*, v. 38, n. 3, p. 2347-2353, 2011.

Chamroukhi, F., Glotin, H., Sam, A. Model-based functional mixture discriminant analysis with hidden process regression for curve classification. *Neurocomputing*, v. 112, n. 15, p. 153-163, 2013.

Chen, H. L., Huang, C. C., Yu, X. G., Xu, X., Sun, X., Wang, G., Wang, S. J. An efficient diagnosis system for detection of Parkinson's disease using fuzzy k-nearest neighbor approach. *Expert Systems with Applications*, v. 40, n. 1, p. 263-271, 2013.

Chen, S. M., Sue, P. J. Constructing concept maps for adaptive learning systems based on data mining techniques. *Expert Systems with Applications*, v. 40, n. 7, p. 2746-2755, 2013.

Coelho, L. S., Thom de Souza, R. C., Mariani, V. C. Improved differential evolution approach based on cultural algorithm and diversity measure applied to solve economic load dispatch problems. *Mathematics and Computers in Simulation*, v. 79, n. 10, p. 3136-3147, 2009.

Coelho, L. S., Grebogi, R. B. Chaotic synchronization using PID control combined with population based incremental learning algorithm. *Expert Systems with Applications*, v. 37, n. 7, p. 5347-5352, 2010.

Csirik, J., Bunke, H. Feature selection and ranking for pattern classification in wireless sensor networks. In: Wang, P. S. P. *Pattern recognition, machine intelligence and biometrics*, Springer, Heidelberg, 2011.

Dart, J. K. G., Wilkins, M. *External eye disease and the oculocutaneos disorders*. In: Taylor, D., Hoyt, C. S. (eds). Pediatric ophthalmology and strabismus. Philadelphia: Elsevier Saunders, p. 163-186, 2005.

Das, S., Maity, S., Qu, B. Y., Suganthan, P. N. Real-parameter evolutionary multimodal optimization: A survey of the state-of-the-art. *Swarm and Evolutionary Computation*, v. 1, n. 2, p. 71-88, 2011.

Dombal, F., Leaper, D., Staniland, J., McCann, A., Harrocks, J. Computer-aided diagnosis of acute abdominal pain. *British Medical Journal*, v. 2, n. 5804, p. 9-13, 1972.

Domingos, P., Pazzani, M. J. On the optimality of the simple Bayesian classifier under zero-one loss. *Machine Learning*, v. 29, n. 2, p. 213-244, 1997.

Dorigo, M., Stützle, T. *Ant colony optimization*, A Bradford Book. Cambridge, MA: MIT, 2004.

Duda, R. O., Hart, P. E. *Pattern classification and scene analysis*. John Wiley & Sons, New York, NY, USA, 1973.

Dugan, R. C., McGranaghan, M. F., Santoso, S., Beaty, H. W. *Electrical power systems quality*, 2nd Edition, McGraw-Hill, London, UK, 2003.

Escalante, H. J., Gómez, M. M., González, J. A., Gil, P. G., Altamirano, L., Reyes, C. A., Reta, C., Rosales, A. Acute leukemia classification by ensemble particle swarm model selection. *Artificial Intelligence in Medicine*, v. 55, n. 3, p. 163-175, 2012.

Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., Uthurusamy, R. *Advances in knowledge discovery e data mining*. Association for the Advancement of

Artificial Intelligence, Massachusetts Institute of Technology, Cambridge, MA, USA, 1996.

Ferrari, F. P., Rosário, N. A., Ribas, L. F. O., Callefe, L. G. Prevalência de asma em escolares de Curitiba – projeto ISAAC (International Study of Asthma and Allergies in Childhood). *Jornal de Pediatria* (Rio), 74, 299-305, 1998.

Fisher, D. H. Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, v. 2, n. 2, p. 139-172, 1987.

Folly K., Venayagamoorthy, G. K. Effects of learning rate on the performance of the population based incremental learning algorithm. *Proceedings of International Joint Conference on Neural Networks*, p. 861-868, Atlanta, GA, USA, 2009.

Freitas, A. A. *Data mining and knowledge discovery with evolutionary algorithms*. Natural Computing Series, Springer-Verlag, Berlim, Alemanha, 2002.

Friedman, N., Geiger, D., Goldszmidt, M. Bayesian network classifiers. *Machine Learning*, v. 29, n. 2, p. 131-163, 1997.

Frölich, H, Chapelle, O., Schölkopf, B. Feature selection for support vector machines by means of genetic algorithms. *International Conference on Tools with Artificial Intelligence*, p. 142-148, 2003.

Gao, J., Li, Q. Z., Wang, J. Method for voltage sag disturbance source location by the real current component. *Power and Energy Engineering Conference*, Wuhan, Hubei, China, 2011.

Gifi, A. *Nonlinear multivariate analysis*. Wiley Series in Probability and Mathematical Statistics, 1989.

Global Initiative for Asthma – GINA. *Bethesda: Global Initiative for Asthma. Global strategy for asthma management and prevention*, 2006.

Góes, A. R. T. Uma metodologia para criação de etiqueta de qualidade no contexto de descoberta de conhecimento em bases de dados: aplicação nas áreas elétrica e educacional. Tese (Doutorado em Métodos Numéricos em Engenharia) – Universidade Federal do Paraná - UFPR, Curitiba, Paraná, 2012.

González, S., Robles, V., Peña, J. M., Cubo, O. EDA-based logistic Regression applied to biomarkers selection in breast cancer. In: Omatu, S., Rocha, M. P., Bravo, J., Fdez Riverola, F., Corchado, E., Bustillo, A., Corchado Rodríguez, J. M. Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living Lecture Notes in Computer Science, v. 5518, pp 979-987, Springer, New York, NY, 2009.

Good, P. *Permutation tests: A practical guide to resampling methods for testing hypotheses*. Springer-Verlag, New York, NY, 1994.

Gunal, S., Gerek, O. N., Ece, D. G., Edizkan, R. The search for optimal feature set in power quality event classification. *Expert Systems with Applications*, v. 36, n. 7, p. 10266-10273, 2009.

Gütlein, M. *Large scale attribute selection using wrappers*, Ph.D. thesis, Albert Ludwigs Universität, Freiburg, Alemanha, 2006.

Guyon, I., Elisseff, A. An introduction to variable and feature selection. *Journal of Machine Learning Research*, v. 3, n. 45, p. 1157-1182, 2003.

Hair, J. F., Black, W. C., Babin, B. J., Anderson, R. E., Tatham, R. L. *Multivariate data analysis*, 7th edition, Pearson, New Jersey, NJ, USA, 2009.

Hall, M., Holmes, G. Benchmarking attribute seleciton techniques for discrete class data mining. *IEEE Transactions on Knowledge Data Engineering*, v. 15, n. 6, p. 1437-1447, 2003.

Hauschild, M., Pelikan, M. A survey of estimation of distribution algorithms. *Swarm and Evolutionary Computation*, v. 1, n. 3, p. 111-128, 2011.

He, X., Sun, N., Zhang, Q., Dong, Y. Feature selection with discrete binary differential evolution. *International Conference on Artificial Intelligence and Computational Intelligence*, Shanghai, China, 2009.

Holland J. H. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*, The MIT Press, Cambridge, MA, USA, 1992.

Hong, Y., Kwong, S., Chang, Y., Ren, Q. Unsupervised feature selection using clustering ensembles and population based incremental learning algorithm. *Pattern Recognition*, v. 41, n. 9, p. 2742-2756, 2008.

Huang, H., Xie, H. B., Guo, J. Y., Chen, H. J. Ant colony optimization-based feature selection method for surface electromyography signals classification. *Computers in Biology and Medicine*, v. 42, n. 2, p. 30-38, 2012.

Huang, N. Lin, L. Power quality disturbances recognition based on PCA and BP neural network. *Power and Energy Engineering Conference*, Chengdu, China, 2010.

Hutter, M., Zaffalon, M. Distribution of mutual information from complete and incomplete data, *Center of Computation and Language Studies*, Report No 0403025, Trinity College Dublin, Dublin, Ireland, 2004.

Iacoban, R., Reynolds, R. & Brewster, J. Cultural swarms: modeling the impact of culture on social interaction and problem solving. *IEEE Swarm Intelligence Symposium*, p. 205-211, Indianapolis, IN, USA, 2003.

Institute of Electrical and Electronics Engineers – IEEE, IEEE Standard 1159-1995 - *IEEE Recommended Practice for Monitoring Electric Power Quality*, 1995.

International Electrotechnical Commission - IEC, IEC 1000-2-1-1990 - Part 2: Environment, Section 1 - Description of the environment. *Electromagnetic environment for low-frequency conducted disturbances and signaling in public low-voltage power supply systems*, 1990.

Inza, P., Larrañaga, B. S. Feature subset selection by bayesian networks: a comparison with genetic and sequential algorithms. *International Journal of Approximate Reasoning*, v. 27, n. 2, p. 143-164, 2001.

Jain, A., Zongker, D. Feature selection: evaluation, application and small sample performance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 19, n. 2, p. 153-158, 1997.

Jensen, F. V., Nielsen, T. D. *Bayesian networks and decision graphs*, 1st Edition, Springer, New York, NY, USA, 2001.

John, G. H., Kohavi, R., Pfleger, K. Irrelevant features and the subset selection problem. *Proceedings of XI International Conference on Machine Learning*, p. 121-129, New Brunswick, NJ, USA, 1994.

Jolliffe, I. T. *Principal Component Analysis*, 2nd edition, Springer, New York, NY, USA, 2002.

Kagan, N., Robba, E. J., Schmidt, H. P. *Estimação de indicadores de qualidade da energia elétrica*, 1ª Edição, Edgard Blücher, São Paulo, Brasil, 2009.

Kang, N., Liao, Y. Fault location estimation for transmission lines using voltage sag data. *IEEE Power and Energy Society General Meeting*, p. 1-6, Minneapolis, MN, USA, 2010.

Kano, M., Nagao, K., Hasebe, S., Hashimoto, I., Ohno, H., Strauss, R., Bakshi, B. Comparison of multivariate statistical process monitoring methods with applications to the Eastman challenge problem. *Computers and Chemical Engineering*, v. 26, n. 2, p. 161-174, 2002.

Karimi, M., Banejad, M., Hassanpour, H., Moeini, A. Classification of power system faults using ANN classifiers. *Proceedings on International Power and Energy Conference*, Shahrood, Irã, 2010.

Katragadda, S. *Multivariate mixed data mining with gifi system using genetic algorithm and information complexity*. Ph.D. thesis, University of Tennessee, Knoxville, TN, USA, 2008.

Keramati, A., Darzi, M., Hosseini, M., Liaei, A. A. Cultural algorithm for feature selection. *3rd International Conference on Data Mining and Intelligent Information Technology Applications*, p. 71-76, Macau, China, 2011.

Khodabakhshian, A., Hemmati, R. Multi-machine power system stabilizer design by using cultural algorithms. *International Journal of Electrical Power & Energy Systems*, v. 44, n. 1, p. 571-580, 2013.

Kohavi, R., John, G. H. Wrappers for feature subset selection. *Artificial Intelligence*, v. 97, n. 1-2, p. 273-324, 1997.

Kusiak, A. Feature transformation methods in data mining. *IEEE Transactions on Electronics Packaging Manufacturing*, v. 24, n. 3, p. 214-221, 2001.

Langley, P., Iba, W.,Thompson, K. An analysis of bayesian classifiers. *National Conference on Artificial Intelligence*, San Jose, CA, USA, 1992.

Le Roux, B., Rouanet, H. *Geometric data analysis: from correspondence analysis to structured data analysis*, 1st Edition, Kluwer Academic Publishers, New York, 2005.

Le Roux, B., Rouanet, H. *Multiple correspondence analysis*, SAGE, Thousand Oaks, CA, USA, 2010.

Ledesma, R. D., Mora, P. V. Determining the number of factors to retain in EFA: an easy-touse computer program for carrying out parallel analysis. *Practical Assessment, Research and Evaluation*, v. 12, n. 2, p. 1-11, 2007.

Li, C., Tayjasanant, T., Xu, W., Liu, X. Method for voltage-sag-source detection by investigating slope of the system trajectory. *IEE Proceedings in Generation, Transmission and Distribution*, v. 150, n. 3, p. 367-372, 2003.

Li, Y., Davis, C. H. Pixel-based invariant feature extraction and its application to radiometric co-registration for multi-temporal high-resolution satellite imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, v. 4, n. 2, p. 348-360, 2011.

Liu, H., Setiono, R. Feature selection and classification – a probabilistic wrapper approach. *Proceedings of 9th International Conference on Industrial and Engineering Applications of AI and ES*, p. 419-424, 1996.

Lopes, H. S., Rodrigues, L. C. A., Steiner, M. T. A. *Meta-Heurísticas em Pesquisa Operacional* - 1a. Edição, Editora Omnipax, Curitiba, PR, 2013.

Loughrey, J., Cunningham, P. Overfitting in wrapper-based feature subset selection: the harder you try the worse it gets. *24th International Conference on Artificial Intelligence*, p. 33-43, Cambridge, UK, 2004.

Ma, H., Zhang, Q. Research on cultural-based multi-objective particle swarm optimization in image compression quality assessment. *Optik - International Journal for Light and Electron Optics*, v. 124, n. 10, p. 957-961, 2013.

Maimon, O. Z., Rokach, L. *Decomposition methodology for knowledge discovery and data mining: theory and applications*, World Scientific Publishing Co. Pte. Ltd., Singapura, Singapura, 2005.

Martin-Bautista, M. J., Vila, M. A. A survey of genetic feature selection in mining issues. *Proceedings of 2nd Conference on Evolutionary Computation*, p. 1314-1321, Washington, DC, USA, 1999.

Matsumoto, K., Sakaguchi, T., Wake, T. Fault diagnosis of a power system based on a description of the structure and function of the relay system. *Expert Systems*, v. 2, n. 3, p. 134-138, 1985.

Michailidis, G., de Leeuw, J. The Gifi system of descriptive multivariate analysis. *Statistical Science*, v. 13, n. 4, p. 307-336, 1996.

Michie, D., Spiegelhalter, D., Taylor, C. *Machine Learning, Neural and Statistical Classification*. Ellis Horwood, New York, NY, USA, 1994.

Minsky, M. Steps toward artificial intelligence. In: Feigenbaum, E. A., Feldman, J. *Computers and Thoughts.* McGraw-Hill, London, UK, 1963.

Mitchell, T. M. *Machine learning*. McGraw-Hill, London, UK, 1997.

Mori, H. State-of-the-art overview on data mining in power systems. Proceedings in *IEEE Power Engineering Society Power Systems Conference and Exposition*, Atlanta, GA, USA, 2006.

Mourão E. M. M., Rosário Filho, N. A. Teste de provocação conjuntival com alérgenos no diagnóstico de conjuntivite alérgica. *Revista Brasileira de Alergia e Imunopatologia*, v. 34, n. 3, p. 90-96, 2011.

Nettleton, D. F. Data mining of social networks represented as graphs. *Computer Science Review*, v. 7, n. 1, p. 1-34, 2013.

Palade, V., Bocaniala, C. D., Jain, L. *Computational Intelligence in Fault Diagnosis*, Springer-Verlag, London, UK, 2006.

Patton, R.J., Uppal, F.J., Lopez-Toribio, C.J. Soft computing approaches to fault diagnosis for dynamic systems: a survey. *IFAC Symposium SAFEPROCESS*, Budapest, Hungria, 2000.

Pearl, J. Fusion, propagation, and structuring in belief networks. *Artificial Intelligence*, v. 29, n. 3, p. 241-288, 1986.

Pearl, J. *Probabilistic reasoning in intelligent systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, San Francisco, CA, USA, 1988.

Pedrycz, W., Ahmad, S. S. S. Evolutionary feature selection via structure retention. *Expert Systems with Applications*, v. 39, n. 15, p. 11801-11807, 2012.

Perez, A., Larranaga, P., Inza, I. Supervised classification with conditional gaussian networks: Increasing the structure complexity from Naïve Bayes. *International Journal of Approximate Reasoning*, v. 43, n. 1, p. 1-25, 2006.

Potvin, A. F. *Nonlinear control design toolbox for use with Matlab*, the MathWorks, Inc., 1993.

Pudil, P., Novovicová, J., Kittler, J. Floating search methods in feature selection. *Pattern Recognition Letters*, v. 15, n. 10, p. 1119-1125, 1994.

Quinlan, J. R. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

Raizman, M. B.: Atualização em alergia ocular. *American Academy of Ophtahlmology*, Focal Points, 1994.

Reunanen, J. *Overfitting in feature selection: pitfalls and solutions*, Ph.D. thesis, Aalto University, Espoo, Finlândia, 2012.

Reynolds, R. G. An introduction to cultural algorithms, *Proceedings of 3rd Annual Conference on Evolutionary Programming*, A. V. Sebald and L. J. Fogel (eds.), World Scientific, p. 131-139, River Edge, NJ, USA, 1994.

Reynolds, R. G., Ali, M. Computing with the social fabric: the evolution of social intelligence within a cultural framework. *IEEE Computational Intelligence Magazine*, v. 3, n. 1, p. 18-30, 2008.

Riedi, C. A., Rosário Filho, N. A., Ribas, L. F. O., Backes, A. S., Kleiniibing, G. F., Popija, M. Increase in prevalence of rhinoconjunctivitis but not asthma and atopic eczema in teenagers. *Journal of Investigational Allergology and Clinical Immunology*, v. 15, n. 1, p. 183-8, 2005.

Riella, R. J., Ferrari, V. P., Paulillo, G., Ortega, M. R., Pereira, J. G. Desenvolvimento de um sistema de monitoramento contínuo da qualidade da energia elétrica para subestações de distribuição. *Seminário Nacional de Distribuição de Energia Elétrica*, Olinda, PE, Brasil, 2008.

Sá, L. C. F., Bechara, S. J. *Conjuntivite alérgica*. In: Grumach, A. S. Alergia e Imunologia na Infância e na Adolescência. São Paulo: Atheneu, p. 321-324, 2009.

Sacha, J. P., Goodenday, L. S., Cios, K. J. Bayesian learning for cardiac SPECT image interpretation. *Journal of Artificial Intelligence in Medicine*, v. 26, n. 1, p. 109-143, 2002.

Saeys, Y., Degroeve, S., Van der Peer, Y. Feature ranking using an EDA-based wrapper approach. In: Lozano, J. A., Larrañaga, P., Inza, I., Bengotxea, E. *Towards a new evolutionary computation: advances in the estimation of distribution algorithms*, v. 192, p. 243-257, Springer, New York, NY, USA, 2006.

Sahami, M. Learning limited dependence bayesian classifiers. *International Conference on Knowledge Discovery in Databases*, Portland, OR, USA, 1996.

Saporta, G., Niang, N. Correspondence analysis and classification, In: Greenacre, M., Blasius, J. *Multiple correspondence analysis and related methods*, p. 371-392, Chapman & Hall/CRC, London, UK, 2006.

Schneider, K., Techniques for improving the performance of Naive Bayes for text classification. *Lecture Notes in Computer Science*, v. 3406, p. 682-693, 2005.

Shapiro, J. L. The sensitivity of PBIL to its learning rate, and how detailed balance can remove it. In: Cotta, C., Jong, K., Poli, R., Rowe, J. *Foundations of Genetic Algorithms VII*. Morgan Kaufmann Publishers, 1st edition, San Francisco, CA, USA 2002.

Shi, C., Wang, Y., Zhang, H. Fault diagnosis based on support vector machines and particle swarm optimization. *International Journal of Advancements in Computing Technology*, v. 3, n. 5, p. 161-169, 2011.

Smith, M. R., Martinez, T. Improving classification accuracy by identifying and removing instances that should be misclassified. *Proceedings of International Joint Conference on Neural Networks*, p. 2690-2697, San Jose, CA, USA, 2011.

Soza, C., Becerra, R. L., Riff, M. C., Coello, C. A. C. Solving timetabling problems using a cultural algorithm. *Applied Soft Computing*, v. 11, n. 1, p. 337-344, 2011.

Steiner, M. T. A. *Uma metodologia para o reconhecimento de padrões multivariados com resposta dicotômica*. Tese (Doutorado em Engenharia de Produção) – Universidade Federal de Santa Catarina - UFSC, Florianópolis, Santa Catarina, SC, 1995.

Steiner, M. T. A., Soma, N. Y., Shimizu, T., Nievola, J. C., Steiner Neto, P. J. Abordagem de um problema médico por meio do processo de KDD com ênfase à análise exploratória dos dados. *Gestão e Produção*, v. 13, n. 2, p. 325-337, 2006.

Steiner, M. T. A., Nievola, J. C., Soma, N. Y., Shimizu, T., Steiner Neto, P. J. Extração de regras de classificação a partir de redes neurais para auxílio à tomada de decisão na concessão de crédito bancário. *Pesquisa Operacional*, v. 27, n. 3, p. 407-426, 2007.

Storn, R., Price, K. Differential evolution – a simple and efficiente heuristic for global optimization over continuous spaces*. Journal of Global Optimization*, v. 11, n. 4, p. 341-359, 1997.

Strohmeier, S., Piazza, F. Domain driven data mining in human resource management: A review of current research. *Expert Systems with Applications*, v. 40, n. 7, p. 2410-2420, 2013.

Sun, Y., Zhang, L., Gu, X. A hybrid co-evolutionary cultural algorithm based on particle swarm optimization for solving global optimization problems. *Neurocomputing*, v. 98, n. 3, p. 76-89, 2012.

Tan, F., Fu, X., Wang, H., Zhang, Y., Bourgeois, A. G. A hybrid feature selection approach for microarray gene expression data. *2nd International Conference on Computational Science*, p. 678-685, Reading, UK, 2006.

Tay, W. L., Chui, C. K., Ong, S. H., Ng, A. C. M. Ensemble-based regression analysis of multimodal medical data for osteopenia diagnosis. *Expert Systems with Applications*, v. 40, n. 2, p. 811-819, 2013.

Thom de Souza, R. C., Coelho, L. S., Mariani, V. C., Steiner, M. T. A. Coffee price forecasting through RBF neural network trained by Kalman filter optimized by genetic algorithms. *International Congress of Mathematics, Engineering and Society*, Curitiba, PR, Brasil, 2009.

Tiplica, T., Kobi, A., Barreau, A. Synthese et comparaison des methodes pour la maitrise statistique des processus multivaries. *Actes du congres QUALITA*, Annecy, France, 2001.

Tsai, C. F., Eberle, W., Chu, C. Y. Genetic algorithms in feature and instance selection. *Knowledge-Based Systems*, v. 39, n. 24, p. 240-247, 2013.

Uyar, A., Benner, A., Ciray, H. N., Bahceci, M. A frequency based encoding technique for transformation of categorical variables in mixed IVF dataset. *31st Annual International Conference of the IEEE EMBS*, Minneapolis, MN, USA, 2009.

Vafaie, H. De Jong, K. Robust feature selection algorithms. *Proceedings of the 5th International Conference on Tools with Artificial Intelligence*, p. 356-363, Boston, MA, USA, 1993.

Van Cauwenberge, P., De Belder, T., Vermeiren, J., Kaplan, A. Global resources in Allergy: allergic rhinitis and allergic conjunctivitis. *Clinical & Experimental Allergy Reviews*, v. 3, n. 1, p. 46-50, 2003.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. N., Yin, K. A review of process fault detection and diagnosis – Part III: Process history based methods, *Computers and Chemical Engineering*, v. 27, n. 3, p. 327-346, 2003.

Ventresca, M., Tizhoosh, H. R. A diversity maintaining population-based incremental learning algorithm. *Information Sciences*, v. 178, n. 21, p. 4038-4056, 2008.

Visconti, I. F., Ross, R. P. D., Souza, L. F. W., Leitão, J. J. A. L. Validação de programa de identificação de VTCD oriundas de aplicações de curtos-circuitos. *Simpósio de Especialistas em Planejamento da Operação e Expansão Elétrica*, Belém, PA, Brasil, 2009.

Wang, L., Wang, S. Y., Xu, Y. An effective hybrid EDA-based algorithm for solving multidimensional knapsack problem. *Expert Systems with Applications*, v. 39, n. 5, p. 5593-5599, 2012.

Wen, C. H., Chen, W. Y. Using multiple correspondence cluster analysis to map the competitive position of airlines. *Journal of Air Transport Management*, v. 17, n. 5, p. 302-304, 2011.

Westphal, G. L. C., Rosário Filho, N. A., Riedi, C. A., Santos, H. L. B. S., Takizawa, K., Souza, R. V. S., Aguilera, C. D. Allergic conjunctivitis is underdiagnosed in asthmatic patients. *Journal of Allergy and Clinical Immunology*, v. 123, n. 2, p. 129-130, 2009.

Witten, L. H., Frank, E. *Data mining: practical machine learning tools and techniques with Java Implementations*. Morgan Kaufmann Publishers, 2nd edition, San Francisco, CA, USA, 2005.

Woolley, N. C., Avendano-Mora, J. M., Milanovic, J. V. A comparison of voltage sag estimation algorithms using optimal monitorring locations. *International Conference on Harmonics and Quality of Power*, v. 1, Bergamo, BG, Itália, 2010.

Yang, L., Yu, J., Lai, Y. Disturbance source identification of voltage sags based on Hilbert-Huang transform. *Power and Energy Engineering Conference*, p. 1-4, Chengdu, China, 2010.

Yu, L., Liu. H. Feature selection for high-dimensional data: a fast correlation-based filter solution. *20th International Conference on Machine Leaning*, p. 856-863, 2003.

Yule, G. U. On the association of attributes in statistics. *Philosophical Transactions of the Royal Society London*, A194, p. 257-319, 1900.

Zollanvari, A. Hua, J., Dougherty, E. R. Analytical study of performance of linear discriminant analysis in stochastic settings. *Pattern Recognition*, v. 46, n. 11, p. 3017-3029, 2013.

ANEXOS

ANEXO 1 - AFUNDAMENTOS DE TENSÃO

Segundo a norma brasileira NBR 5460 (ABNT, 1992), Sistemas Elétricos de Potência (SEP) podem ser definidos como o conjunto de todas as instalações e equipamentos destinados à geração, transmissão e distribuição de energia elétrica.

A QEE de um SEP é a condição de compatibilidade entre sistema supridor e carga atendendo critérios de conformidade senoidal. Segundo Dugan *et al.* (2003), um problema de QEE é “qualquer problema manifestado na tensão, corrente ou na frequência que resulte em falha ou má operação de equipamento do consumidor”. Segundo Kagan *et al.* (2009), os principais fenômenos estudados na QEE são as Variações de Tensão de Curta Duração (VTCD).

As VTCD são variações nos níveis de tensão acarretadas principalmente por faltas no sistema elétrico ou por outros tipos de eventos, como é o caso, por exemplo, da partida de grandes motores ligados ao sistema de distribuição. As VTCD podem ser classificadas em afundamentos de tensão (em inglês, *voltage sags*) e elevações de tensão (*voltage swells*). O efeito maior destes fenômenos leva ao mau funcionamento de equipamentos sensíveis, principalmente no caso de afundamentos de tensão.

Valores de magnitude e de duração dos afundamentos de tensão medem a sua severidade e devem ser confrontados com o nível de sensibilidade (ou susceptibilidade) dos equipamentos. Determinados tipos de processos industriais podem sofrer sérias consequências pela ocorrência de afundamentos de tensão quando, o mau funcionamento de equipamentos sensíveis, provoca a parada de processos, perda de matéria-prima, longo tempo para reinicialização do processo, *etc*., que em suma podem gerar prejuízos para as empresas produtoras.

Embora haja um entendimento generalizado de que um afundamento de tensão é uma redução do valor eficaz da tensão por um período de curta

duração, seguido de sua restauração, há divergências nas normas quanto à metodologia para sua quantificação.

Em 1995, foi publicada pelo *Institute of Electrical and Electronics Engineers* (*IEEE*) uma recomendação para o monitoramento da QEE, onde podem ser encontrados a definição de afundamento de tensão, os objetivos de seu monitoramento, os principais instrumentos e técnicas de medição, bem como técnicas para a interpretação dos resultados de medições.

O *IEEE* (1995) define um afundamento de tensão como sendo uma diminuição do valor eficaz da tensão para 0,1 a 0,9 pu, durante um intervalo de tempo entre 0,5 ciclo (~8 ms) da frequência fundamental até 1 minuto. Quando a tensão eficaz cai abaixo de 0,1 pu, considera-se o evento como interrupção de curta duração.

A norma *IEC* 1000-2-1 (1990) define afundamento de tensão como “uma redução súbita da tensão em um ponto do sistema elétrico, seguido de seu restabelecimento após um curto período de tempo, de 0,5 ciclo até poucos segundos”.

No Brasil, a ANEEL classifica os afundamentos de tensão em afundamentos momentâneos de tensão, com durações de até 3 segundos, e em afundamentos temporários de tensão, com durações entre 3 segundos e 3 minutos. Em seu procedimento de distribuição, denominado PRODIST (ANEEL, 2009), a ANEEL define afundamento momentâneo de tensão como sendo um “evento em que o valor eficaz da tensão do sistema se reduz, momentaneamente, para valores abaixo de 90% da tensão nominal de operação, durante um intervalo inferior a 3 segundos”.

Os afundamentos momentâneos de tensão ocorrem devido a duas causas (eventos básicos) principais: (i) conexão de cargas de grande porte no sistema e (ii) curtos-circuitos.

A conexão de cargas de grande porte no sistema como, por exemplo, a partida de um grande motor industrial, pode causar afundamentos de tensão durante o período inicial da alimentação. Este problema é dependente de um projeto adequado e compatível da rede de suprimento e do método de partida ou ligação da carga, sendo, em geral, convenientemente tratado e resolvido por engenheiros na fase de projeto.

Já os curtos-circuitos geralmente provocam afundamentos de tensão até que a proteção do sistema isole o defeito. Este tipo de falha é a principal causa de afundamentos. Os diversos dispositivos e equipamentos que compõem os sistemas elétricos estão sujeitos a diferentes níveis de tensão e corrente. Por diversas vezes, estes sistemas também se encontram expostos a fatores climáticos (no caso de uma subestação aberta, por exemplo). Devido às suas condições, os sistemas elétricos estão muito vulneráveis quando se trata da possibilidade de ocorrência de curtos-circuitos.

Um curto-circuito é caracterizado pela passagem de corrente elétrica acima dos valores nominais, o que ocorre devido à redução da impedância como consequência de um defeito. Curtos-circuitos podem ocorrer em qualquer parte do sistema, sem qualquer restrição e podem envolver de uma a três fases de um único barramento (monofásico, bifásico ou trifásico), além das possibilidades monofásico-terra e bifásico-terra.

Na realidade, o curto-circuito é consequência de algum evento que ocorreu na rede. A correta identificação da causa-raiz é tão ou mais importante, pois muitas vezes permite identificar um conjunto de causas que estão originando a degradação da rede e por consequência à baixa QEE.

Um exemplo disto é a identificação do padrão de construção e manutenção da rede que pode estar contribuindo para os curtos-circuitos, tais como postes desalinhados, postes podres, estais frouxos, condutores com problemas de tracionamento, entre outros.

Segundo Kagan *et al.* (2009), os afundamentos momentâneos de tensão podem provocar diferentes impactos sobre a carga. Dispositivos que utilizam componentes eletrônicos ou de eletrônica de potência no controle de diversos processos, tais como Controladores Lógicos Programáveis (CLP), quando submetidos a afundamentos, podem apresentar problemas no funcionamento, como a perda da programação de seus microprocessadores.

Tais problemas afetam o ambiente no qual estão inseridos interrompendo processos industriais automatizados (Visconti *et al.*, 2009), causando perdas de produção, atrasos devido a reinicialização do processo, dentre outros prejuízos.

A magnitude e a duração dos afundamentos momentâneos de tensão são suas principais características causadoras de impactos sobre equipamentos. Alguns equipamentos são sensíveis e apresentam atuações indevidas tão somente à magnitude dos afundamentos, casos de relés, robôs, contatores eletromecânicos, acionamentos de velocidade variável em corrente alternada, certos controles de processo e muitos tipos de máquinas automatizadas.

Quando a tensão cai abaixo de um dado valor mínimo, estes equipamentos podem apresentar atuações indesejadas e outros tipos de problemas. Outros equipamentos são sensíveis à magnitude e à duração, ou seja, nestes casos é importante saber a duração na qual o equipamento apresenta mau funcionamento quando a tensão está abaixo de um determinado valor.

Alguns equipamentos são afetados por outras características dos afundamentos, como o desequilíbrio entre fases, o ponto na forma de onda da tensão onde o afundamento se inicia, entre outras características. Estes parâmetros são mais difíceis para generalizar. Como resultado, os indicadores utilizados para análise de desempenho concentram-se principalmente nos parâmetros de magnitude e duração dos afundamentos.

O diagnóstico eficiente das características (caracterização) dos afundamentos de tensão é de fundamental importância para a determinação das prioridades de atuação das concessionárias nos afundamentos com maior potencial de danos aos consumidores.

As principais causas de afundamentos de tensão são os curtos-circuitos em qualquer ponto de fornecimento de energia. O curto-circuito provoca uma grande elevação da corrente, e esta, por sua vez, ocasiona grandes quedas de tensão nas impedâncias do sistema. Curtos-circuitos são ocorrências inevitáveis nos sistemas elétricos. Suas causas são diversas, mas basicamente envolvem um rompimento do dielétrico entre dois pontos que deveriam ser isolados entre si e que, em condições normais, estão em potenciais diferentes.

Deve-se considerar como a causa de um afundamento de tensão, o motivo primário que levou a variação da tensão oferecida, uma vez que causas

secundárias originadas pela causa-raiz podem “mascarar” a verdadeira causa e originar uma ação de manutenção e operação inadequada.

Na literatura existem algumas classificações de causas bem definidas. Em Brown (2002), as causas de curtos-circuitos são classificadas em função de sua natureza, tais como falhas em equipamento, animais e árvores em contato com a rede, efeitos do clima adverso, descargas atmosféricas e interferência humana. Identificando esses problemas como causa-raiz, é possível então investir recursos financeiros de forma eficaz no intuito de reduzir afundamentos momentâneos de tensão e melhorar a QEE.

No Brasil, a ANEEL já estabelece às concessionárias, indicadores de confiabilidade, que medem a frequência e a duração de interrupções no fornecimento de energia elétrica. Como forma de minimizar os prejuízos aos consumidores e estimular a melhora de serviço, a ANEEL exige uma compensação direta na fatura do cliente, caso as interrupções apuradas no mês anterior ultrapassem determinados limites (ANEEL, 2009).

É de se esperar que em um futuro não muito distante, a maioria dos países, incluindo o Brasil, também exija o cumprimento de metas de QEE, em especial, em termos de magnitude da tensão contratada. Se antecipando a esta possibilidade, torna-se de suma importância que as concessionárias atuem, desde já, de forma a diminuir (e se possível evitar) o fornecimento de energia elétrica fora dos limites adequados de tensão.

Para realizar um controle a respeito de curtos-circuitos, muitas empresas de energia adotam o registro desses eventos em bancos de dados para posterior análise. No momento de um reparo, o eletricista responsável registra uma série de informações a respeito do sistema elétrico, tais como as condições climáticas observadas no ambiente, atuações de sistemas de proteção (fusíveis, disjuntores, *etc*.), elementos defeituosos, bem como a possível causa do defeito.

Entretanto, a sistemática de identificação da causa por parte do eletricista e o fluxo dessa informação para a base de dados, se tornam muitas vezes ineficientes. Idealmente essa identificação deveria ser realizada por um eletricista experiente, tendo recebido um treinamento adequado para uma análise minuciosa da falta. Na prática, tais condições nem sempre são

possíveis e o resultado final do armazenamento desses dados em estado bruto proporciona registros com diversas falhas humanas, como o não preenchimento de determinados campos, defeitos pouco coerentes com as condições locais observadas e atribuição de consequências de falhas que não são as causas-raiz da falha.

Além do estudo pioneiro de Matsumoto *et al.* (1985) a respeito de diagnóstico de falhas baseado em conhecimento em SEP, poucos outros trabalhos sobre o tema foram relatados antes dos anos 1990. A preocupação com o tema foi retomada somente com os esforços de Billinton *et al.* (1991), quando estes propuseram uma metodologia para avaliação das consequências causadas aos consumidores pela falta de energia. A partir daí, o interesse foi crescente, ampliando-se também rapidamente a literatura relacionada ao assunto.

Nos últimos anos, diversos estudos no campo do diagnóstico de falhas em SEP baseados em classificação de padrões têm sido realizados. Revisões neste sentido podem ser obtidas em Patton *et al.* (2000), Venkatasubramanian *et al.* (2003), Mori (2006) e Palade *et al.* (2006).

O recente aumento do interesse em se diagnosticar causas de afundamentos de tensão se deve, principalmente, pela maior exigência dos governos em relação à qualidade da tensão fornecida aos consumidores (em especial, afundamentos de tensão). Um dos primeiros trabalhos neste sentido foi o de Li *et al.* (2003). Nos anos que se seguiram, as pesquisas se intensificaram ainda mais com trabalhos como os de Gunal *et al.* (2009), Huang e Lin (2010), Kang e Liao (2010), Woolley *et al.* (2010), Yang *et al.* (2010), Karimi *et al.* (2010), Gao *et al.* (2011) e Góes (2012).

A maioria destes trabalhos realizaram suas análises por métodos híbridos envolvendo dados de impedância, ondas de tensão, dispositivos distribuídos e, até mesmo, de classificação de padrões. Contudo, tais trabalhos não se utilizam do histórico de curtos-circuitos correlacionados aos dados de tensão para detecção de afundamentos de tensão e diagnóstico de suas causas-raiz, como é proposto no presente trabalho.

ANEXO 2 - RINOCONJUNTIVITE ALÉRGICA

A rinoconjuntivite alérgica é uma manifestação inflamatória de hipersensibilidade tipo I[26] de Gell e Coombs (1963) *apud* Sá e Bechara (2009). Nos indivíduos predispostos, os aeroalérgenos (ácaros da poeira, baratas, polens de gramíneas e epitélios de animais) entram em contato com as mucosas nasal e/ou conjuntival desencadeando esta resposta alérgica. Estima-se que entre 10% e 30% da população mundial apresente sintomas alérgicos, e destes, cerca de um terço apresentam sintomas oculares (Sá e Bechara, 2009).

A gravidade da reação alérgica se relaciona à intensidade da resposta inflamatória, à idade do paciente e aos fatores genéticos e geográficos. De maneira geral, crianças são mais predispostas a esta condição e a presença de história familiar de doenças alérgicas também parece ser um fator predisponente para a doença. Outras doenças alérgicas como asma, rinite e dermatite atópica encontram-se frequentemente associadas. As condições geográficas também estão associadas aos tipos de alérgenos desencadeadores dos sintomas de acordo com o perfil climático de cada local.

As manifestações clínicas da doença dividem-se entre sintomas nasais (rinite) e oculares (conjuntivite). Os sintomas nasais caracterizam-se por coriza (secreção nasal clara), obstrução nasal, espirros e prurido (coceira) nasal. Dentre os sintomas oculares encontra-se prurido (coceira) ocular, lacrimejamento, hiperemia conjuntival (vermelhidão dos olhos) e edema (inchaço) palpebral (Mourão e Rosário Filho, 2011). A alergia ocular pode ser classificada em: conjuntivite alérgica sazonal ou perene, ceratoconjuntivite primaveril ou vernal, ceratoconjuntivite atópica, conjuntivite papilar gigante e ceratoconjuntivite tóxica (Dart e Wilkins, 2005; Raizman, 1994).

A conjuntivite alérgica sazonal ou perene é a forma mais frequente, sendo responsável por 50% dos casos de rinoconjuntivite alérgica. Apresenta como manifestação inicial e mais frequente o prurido nasal e ocular, seguido

---

[26] Tipo I significa resposta alérgica imediata.

por edema e lacrimejamento. As formas sazonais são relacionadas à alergia aos polens e as formas perenes aos ácaros da poeira doméstica.

A ceratoconjuntivite primaveril ou vernal é uma forma rara, porém grave da doença, pois frequentemente ocorre acometimento não só conjuntival, mas de córnea. Associada principalmente a regiões de clima mais árido, com pico de sintomas entre a primavera e verão. Predomina na população pediátrica e em adolescentes com história prévia de asma, rinite alérgica e dermatite atópica. Assim como a conjuntivite alérgica sazonal ou perene, manifesta-se com prurido ocular intenso, porém com sintomas de fotofobia (aversão à luz) e ardência, estes últimos relacionados ao comprometimento corneano.

A ceratoconjuntivite atópica assemelha-se em sintomas a ceratoconjuntivite primaveril, contudo esta última tende a melhorar com idade (por volta dos 5 a 10 anos) ao contrário da primeira que pode ter longa duração.

A ceratoconjuntivite papilar gigante e ceratoconjuntivite tóxica são formas de alergias oculares, contudo não associadas aos sintomas alérgicos nasais, nem mesmo aos aeroalérgenos. A ceratoconjuntivite papilar gigante é classicamente associada à intolerância às lentes de contato ou produtos utilizados para conservação e esterelização das mesmas. A ceratoconjuntivite tóxica ocorre por alergia a drogas (colírios), cosméticos ou fatores ambientais como poluição, ar condicionado e vapores.

O diagnóstico da rinoconjuntivite alérgica é essencialmente clínico, ou seja, realizado pelos sintomas relatados pelo paciente e sinais vistos no exame físico. Os exames laboratoriais podem ser úteis para identificar os tipos de alérgenos desencadeadores da doença. Contudo, o diagnóstico da rinoconjuntivite pode não ser uma tarefa simples, pois alguns sintomas desta doença podem também estar presentes em outras patologias oculares, gerando dúvidas diagnósticas. Além disso, o fato destes sintomas serem, na maioria das vezes, brandos e autolimitados faz com que a rinoconjuntivite alérgica seja frequentemente ignorada por médicos e pacientes. Apesar da maior parte dos casos serem benignos, em algumas situações esta doença pode levar a sintomatologia grave, com risco de evoluir para complicações sérias e irreversíveis, incluindo perda visual.

Em decorrência disto, existem poucos dados epidemiológicos sobre a prevalência da rinoconjuntivite alérgica, mas estima-se que entre 5% e 20% da população mundial apresentem a doença (Van Cauwenberge *et al.*, 2003). No Brasil, apesar da escassez de dados epidemiológicos, o número de casos parece estar em ascensão. Riedi *et al.* (2005) encontraram 17,2% de prevalência de rinoconjuntivite em crianças de 13 a 14 anos em 2001, um relativo aumento do número de casos em relação aos 13,9% obtidos em um levantamento anterior (Ferrari *et al.*, 1995). Esses achados sugerem que, assim como as demais doenças alérgicas, a rinoconjuntivite alérgica apresenta-se em franca ascensão em nosso meio.