UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS
DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

Nádia Junqueira Martarelli

**Seleção de atributos em agrupamento
de dados utilizando algoritmos
evolutivos**

Nádia Junqueira Martarelli

**Seleção de atributos em agrupamento
de dados utilizando algoritmos
evolutivos**

Texto apresentado ao Programa de Engenharia de
Produção da Escola de Engenharia de São Carlos
como parte dos requisitos para a obtenção do título de
Mestre em Engenharia de Produção.

Área de concentração: Processos e Gestão de Operações

Orientador: Prof. Dr. Marcelo Seido Nagano

São Carlos - SP

2016

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

M375s Martarelli, Nádia Junqueira
Seleção de atributos em agrupamento de dados
utilizando algoritmos evolutivos / Nádia Junqueira
Martarelli; orientador Marcelo Seido Nagano. São
Carlos, 2016.

Dissertação (Mestrado) - Programa de Pós-Graduação
em Engenharia de Produção e Área de Concentração em
Processos e Gestão de Operações -- Escola de Engenharia
de São Carlos da Universidade de São Paulo, 2016.

1. Agrupamento de Dados. 2. Algoritmos Evolutivos.
3. Seleção de Atributos. 4. Algoritmos Genéticos. I.
Título.

FOLHA DE JULGAMENTO

Candidata: Engenheira **NÁDIA JUNQUEIRA MARTARELLI**.

Título da dissertação: "Seleção de atributos em agrupamento de dados utilizando algoritmos evolutivos".

Data da defesa: 03/08/2016.

Comissão Julgadora:

Prof. Dr. **Marcelo Seido Nagano (Orientador)**
(Escola de Engenharia de São Carlos/EESC)

Prof. Dr. **Roberto Fernandes Tavares Neto**
(Universidade Federal de São Carlos/UFSCar)

Prof. Dr. **Luis Aparecido Milan**
(Universidade Federal de São Carlos/UFSCar)

Resultado:

Aprovada

APROVADA

APROVADA

Coordenadora do Programa de Pós-Graduação em Engenharia de Produção:
Profa. Associada **Daisy Aparecida do Nascimento Rebelatto**

Presidente da Comissão de Pós-Graduação:
Prof. Associado **Luis Fernando Costa Alberto**

À minha família

Este documento foi formatado pelo editor de textos $\text{\LaTeX}$. O sistema de citações de referências bibliográficas utiliza o padrão ABNT do sistema BibTeX e segue normas definidas pela Universidade de São Paulo - USP.

Agradecimentos

Agradeço primeiramente a Deus, pela presença constante, condução, oportunidades oferecidas a mim. Ao meu orientador, Professor Doutor Marcelo Seido Nagano, pela confiança, paciência e incentivo durante esta jornada. A minha família, meus pais, Andrea e Antônio e aos meus irmãos, Thais e Rafael, pelo apoio incondicional durante toda minha vida e por sempre se posicionarem de forma positiva e motivadora. Ao meu noivo, Thiago, pelo apoio, paciência e dedicação, não só nesta etapa, mas em todas as outras. Aos amigos do Laboratório de Pesquisa Operacional Aplicada (LAOR), Fernando Rossi, Hugo Miyata, Thiago Soares e João Vítor Robazzi pela companhia, companheirismo e pelas discussões que enriqueceram este trabalho e minha experiência de vida. Aos funcionários do departamento de Engenharia de Produção da Escola de Engenharia de São Carlos (EESC) da Universidade de São Paulo, pela competência, disposição e dedicação. E, ao CNPq, pelo apoio financeiro concedido, o que tornou possível o desenvolvimento deste trabalho.

"Those that know, do. Those that understand, teach"
- *Aristotle*

Resumo

Martarelli, Nádia Junqueira **Seleção de atributos em agrupamento de dados utilizando algoritmos evolutivos**. 137 p. Dissertação de mestrado – Escola de Engenharia de São Carlos, Universidade de São Paulo, 2016.

Com o surgimento da tecnologia da informação, o processo de análise e interpretação de dados deixou de ser executado exclusivamente por seres humanos, passando a contar com auxílio computacional para a descoberta de conhecimento em grandes bancos de dados. Este auxílio exige uma organização e ordenação das atividades, antes manualmente exercidas, em um processo composto de três grandes etapas. A primeira etapa deste processo conta com uma tarefa de redução da dimensionalidade, que tem como objetivo a eliminação de atributos que não contribuem para a análise dos dados, resultando portanto, na seleção de um subconjunto dos atributos originais. A seleção de um subconjunto de atributos pode ser encarada como um problema de busca, já que há inúmeras possibilidades de combinação dos atributos originais em subconjuntos. Dessa forma, uma das estratégias de busca que pode ser adotada consiste na busca randômica, executada por um algoritmo genético ou pelas suas variações. Este trabalho propõe a aplicação de duas variações do algoritmo genético, Algoritmo Genético Construtivo e Algoritmo Genético Enviesado com Chave Aleatória, no problema de seleção de atributos em agrupamento de dados, já que estas duas variações ainda não foram aplicadas em tal problema. A fim de verificar o desempenho destas duas variações, comparou-se ambas com a abordagem tradicional do algoritmo genético. Efetuou-se também a comparação entre as duas variações. Para isto, foi utilizada três bases de dados retiradas do repositório UCI ¹ de aprendizado de máquinas. Os resultados obtidos mostraram que os desempenhos, em termos de qualidade da solução, dos algoritmos: genético construtivo e genético enviesado com chave aleatório foram melhores, de maneira geral, do que o desempenho da abordagem tradicional. Constatou-se também diferença significativa em termos de eficiência entre as duas

¹<http://archive.ics.uci.edu/ml/>

variações e a abordagem tradicional.

Palavras-chave: Agrupamento de Dados, Algoritmos Evolutivos, Seleção de Atributos, Algoritmos Genéticos.

Abstract

Martarelli, Nádia Junqueira **Feature subset selection in data clustering using evolutionary algorithm.** 137 p. Master Thesis – São Carlos School of Engineering, University of São Paulo, 2016.

With the advent of information technology, the process of analysis and interpretation of data left to be run exclusively by humans, going to rely on computational support for knowledge discovery in large databases. This aid requires an organization and sequencing of activities before manually performed in a compound of three major step process. The first step of this process has a reduced dimensionality task, which aims to eliminate attributes that do not contribute to the data analysis, resulting therefore, in selecting a subset of the original attributes. Selecting a subset of attributes can be viewed as a search problem, since there are numerous possible combinations of unique attributes into subsets. Thus, one search strategies that can be adopted is to randomly search, performed by a genetic algorithm or its variants. This paper proposes the application of two variations of the genetic algorithm, Constructive Genetic Algorithm and Biased Random Key Genetic Algorithm in the feature selection problem in data grouping, as these two variations have not been applied in such a problem. In order to verify the performance of the two variations, we compare them with the traditional algorithm , genetic algorithm. It was also executed the comparison between the two variations. For this, we used three databases removed from the UCI ² repository of machine learning.

The results showed that the performance, in term of quality solution, of algorithms: genetic constructive and genetic biased with random key are better than the performance of the traditional approach. It was also observed a significant difference in efficiency between of the two variations and the traditional approach.

Keywords: Clustering Data, Evolucionary Algorithm, Feature Subset Selection, Genetic

²<http://archive.ics.uci.edu/ml/>

Algorithms.

Lista de Ilustrações

Figura 1	Evolução da quantidade de atributos nos bancos de dados inseridos em três repositórios: (i) UCI KDD, em vermelho; (ii) UCI, em azul e (iii) LIBSVM, em verde, nas últimas duas décadas.	32
Figura 2	Diagramas de dispersão usando as variáveis (a) COL 1 e COL 2 e (b) COL 3 e COL 4.	33
Figura 3	Diagrama de dispersão usando as variáveis (a) COL 1, COL 2 e COL 4 e (b) COL 2, COL 3 e COL 4.	34
Figura 4	Diagrama de dispersão de duas variáveis que formam cinco grupos. . .	34
Figura 5	Dendrogramas resultante do agrupamento dos dados.	35
Figura 6	Evolução histórica do processo de análise e interpretação de dados para obtenção de conhecimento com foco no assunto abordado neste trabalho, seleção de subconjunto de atributos utilizando duas variações do algoritmo genético.	36
Figura 7	Hierarquia do tipo de variáveis.	42
Figura 8	A hierarquia do aprendizado.	43
Figura 9	Três exemplos gráficos da relação entre o número de grupos e o número de variáveis em agrupamento de dados.	44
Figura 10	Medidas de importância de atributos.	45
Figura 11	Etapas mais detalhadas do processo de KDD.	47
Figura 12	Relação entre o número de dimensões e o volume do espaço.	49
Figura 13	Hierarquia das operações com atributos.	50
Figura 14	Diagrama resumido de decisão da análise de agrupamento.	54
Figura 15	Dendrograma.	56
Figura 16	Todos os possíveis subconjuntos de atributos para $p_{co}=4$	61
Figura 17	Caracterização de um algoritmo de seleção de atributos.	66
Figura 18	Abordagem filtro para seleção de atributos em agrupamento de dados. .	67
Figura 19	Método envelope para seleção de atributos em agrupamento de dados. .	68

Figura 20	Esboço da estrutura de um cromossomo, gene e DNA.	72
Figura 21	Parâmetros e operações que compõe a estrutura lógica de um algoritmo genético.	74
Figura 22	Seleção por roleta como procedimento de seleção de indivíduos para reprodução e consequente construção da próxima geração.	76
Figura 23	Operador recombinação de um e dois pontos.	77
Figura 24	Operador de mutação em um ponto.	78
Figura 25	Influência dos operadores de seleção, recombinação e mutação no algoritmo genético tradicional	79
Figura 26	Exemplo de esquemas, estruturas e suas respectivas propriedades. . . .	80
Figura 27	Classificação de técnicas para ajuste de parâmetros.	81
Figura 28	Visão geral das operações do algoritmo genético tradicional.	82
Figura 29	Seleção dos indivíduos na população para a reprodução.	85
Figura 30	Visão geral das operações do algoritmo genético construtivo e as mudanças perante a abordagem tradicional.	87
Figura 31	Fluxograma do funcionamento do BRKGA	88
Figura 32	Método de seleção do BRKGA.	90
Figura 33	Visão geral das operações do algoritmo genético envisado com chave aleatória e as mudanças perante a abordagem tradicional.	91
Figura 34	Gráfico de caixa da base de dados autenticação de cédulas.	102
Figura 35	Gráfico de caixa da base de dados qualidade de vinho.	104
Figura 36	Gráfico de caixa da base de dados qualidade do ar.	105
Figura 37	Gráfico de barra de erro das médias dos índices CCC e tempos de execução da base de dados autenticidade de cédula a um nível de 95%.	111
Figura 38	Gráfico de barra de erro das médias dos índices CCC e tempos de execução da base de dados qualidade do vinho branco a um nível de 95%.	111
Figura 39	Gráfico de barra de erro das médias dos índices CCC e tempos de execução da base de dados qualidade do ar a um nível de 95%.	111
Figura 40	Gráfico resultante do teste de Turkey para a base de dados autenticidade de cédula um nível de 95%.	113
Figura 41	Gráfico resultante do teste de Turkey para a base de dados qualidade do vinho branco um nível de 95%.	114
Figura 42	Gráfico resultante do teste de Turkey para a base de dados qualidade do ar um nível de 95%.	114

Lista de Tabelas

Tabela 1	Formato atributo-valor	41
Tabela 2	Combinação das estratégias e direções de busca	65
Tabela 3	Vantagens e desvantagens da abordagem filtro e métodos mais utilizados	67
Tabela 4	Vantagens e desvantagens dos métodos envelope e métodos mais utilizados	69
Tabela 5	Parâmetros definidos para cada base de dados na modelagem do algoritmo genético tradicional	94
Tabela 6	Operações definidas para cada base de dados na modelagem do algoritmo genético tradicional	94
Tabela 7	Parâmetros definidos para cada base de dados na modelagem do algoritmo genético construtivo	97
Tabela 8	Operações definidas para cada base de dados na modelagem do algoritmo genético construtivo	97
Tabela 9	Parâmetros definidos para cada base de dados na modelagem do algoritmo genético enviesado com chave aleatória.	99
Tabela 10	Operações definidas para cada base de dados na modelagem do algoritmo genético enviesado com chave aleatória	99
Tabela 11	Estatísticas descritivas da base de dados: autenticidade de cédula . . .	102
Tabela 12	Estatísticas descritivas da base de dados: qualidade do vinho branco. .	103
Tabela 13	Estatísticas descritivas da base de dados: qualidade do ar	105
Tabela 14	Valores do índice CCC para o conjunto de atributos completos, sem seleção de variáveis	106
Tabela 15	Comparação dos resultados obtidos na execução dos três algoritmos para base de dados: autenticidade de cédula	107
Tabela 16	Comparação dos resultados obtidos na execução dos três algoritmos para base de dados: qualidade do vinho branco	108

Tabela 17	Comparação dos resultados obtidos na execução dos três algoritmos para base de dados: qualidade do ar	109
Tabela 18	Média e desvio padrão dos valores do índice CCC obtidos na execução dos três algoritmos para as três base de dados	110
Tabela 19	Média e desvio padrão dos tempos de execução obtidos pelos três algoritmos para as três base de dados	110
Tabela 20	Desvio relativo para valores do índice CCC correspondentes às três bases de dados analisadas	112
Tabela 21	Porcetagem de sucesso de cada algoritmo	112
Tabela 22	Teste T de amostras independentes para verificar a diferença de médias dos valores do índice CCC e o tempo de execução entre os algoritmos .	113
Tabela 23	Matriz de correlação da base de dados: autenticação de cédula	135
Tabela 24	Matriz de correlação da base de dados: qualidade do vinho branco . . .	136
Tabela 25	Matriz de correlação da base de dados: qualidade do ar	137

Lista de Abreviaturas

ACP Análise de Componentes Principais

AE Algoritmo Evolutivo

AG Algoritmo Genético

AGC Algoritmo Genético Construtivo

AGCA Algoritmo Genético com Chave Aleatória

AGECA Algoritmo Genético Enviesado com Chave Aleatória

A.C. Banco de dados autenticidade de cédula

BRKGA Biased Random-Key Genetic Algorithms

CAC Critério de Agrupamento Cúbico

CCC Cubic Criterion Clustering

DCDB Descoberta de Conhecimento em Banco de Dados

DM Data Mining

DNA Deoxyribonucleid Acid

HP Heurística de Permutação

KDD Knowledge Discovery in Databases

MD Mineração de Dados

PCA Principal Components Analysis

Q.V.B. Banco de dados qualidade do vinho branco

Q.A. Banco de dados qualidade do ar

RKGA Random-Key Genetic Algorithms

SA Seleção de Atributos

SSA Seleção de Subconjunto de Atributos

TA Transformação de Atributos

Lista de Símbolos

s_{novo}	novo indivíduo gerado através de reprodução ou mutação
α	parâmetro de evolução
$\mu(\bar{t})$	adaptação média de todos os esquemas no instante t
$\bar{x}_c$	centróide do grupo c
$\delta(s_i)$	valor associado a cada indivíduo, que posteriormente é comparado com α na avaliação da população
ϵ	incremento de α
$\mu(\hat{H}, t)$	adaptação média observada de H no instante t
ρ_s	porcentagem predefinida para a definição se o atributo será selecionado ou não na abordagem BRKGA.
$\varphi(H)$	diferença entre o primeiro e o último símbolo fixo em H
$ A ^b$	tamanho do espaço de indivíduos
$ A $	número de símbolos em A
A	alfabeto que irá compor a <i>string</i>
A^*	algoritmo de busca de caminho
A^b	espaço de indivíduos
b	tamanho da <i>string</i>
CCC	valor do índice CCC
d_i	desvio percentual de $g(s_i)$

$dist$ distância euclidiana ao quadrado

$E(M(H, t + 1))$ número esperado de estruturas do esquema H no instante da evolução $t + 1$

$E(R^2)$ esperança de R^2

$f(s_i)$ função de avaliação auxiliar do algoritmo genético construtivo, a qual determina o limite inferior para $g(s_i)$

$g(s_i)$ função de avaliação do algoritmo genético construtivo

g_{max} limite superior para $g(s_i)$

H esquema

k número de grupos

k_{max} número máximo de grupos

k_{min} número mínimo de grupos

$M(H)$ quantidade de estruturas que advém do esquema H

n número total de registros

n_{base} número de indivíduos pertencentes a região base, considerados os melhores da população

$o(H)$ número símbolos fixos no esquema H

P^+ população de indivíduos da geração subsequente a população P_{BRKGA}

p_c taxa de recombinação

p_j atributo j de uma matriz de dados no formato atributo-valor, em que $j \in \{1, 2, 3, \dots, p_{co}\}$

p_k taxa de recombinação para grupo

p_m taxa de mutação

P_α número de indivíduos na população cujo parâmetro de evolução é α

p_{base} porcentagem que define a região base para a seleção de indivíduos no AGC.

P_{BRKGA} população de indivíduos

p_{co} número de atributos do conjunto original

p_{exo} número de atributos de um subconjunto após a execução da operação específica

q	número de iterações ou gerações
R^2	representa proporção de variância nos grupos
r_i	linha do registo i em matriz de dados no formato atributo-valor, em que $i \in \{1,2,3,\dots,n\}$
s	<i>string</i> , indivíduo, solução
S_q	subconjunto q , sendo $q \in \{1,2,3,\dots,t\}$
s_{base}	indivíduo que é selecionado da região base para participar da reprodução
s_{guia}	indivíduo que é selecionado da população inteira para participar da reprodução
t	instante t da evolução
t	total de subconjuntos possíveis
t'	instante imediatamente posterior a ordenação decrescente dos indivíduos
u	número de iterações consecutivas de adição
w	número de iterações consecutivas de eliminação
$x_{i,j}$	valor do registo de linha i e de atributo j de uma matriz de dados no formato atributo-valor
y	número de atributos como critério de parada.
z	número de iterações, como critério de parada, sem ganho significativo na medida de avaliação.
c	número do grupo

Lista de Algoritmos

1	Algoritmo de agrupamento <i>k-means</i>	57
2	Algoritmo genético tradicional	83
3	Algoritmo genético construtivo	87
4	Algoritmo genético enviesado com chave aleatória	91
5	Heurística de permutação	97

Sumário

1	Introdução	31
1.1	Objetivo da pesquisa	37
1.2	Justificativa	37
1.3	Estrutura do trabalho	38
2	Terminologias e conceitos	41
2.1	Definições, notações e terminologias	41
2.2	Descoberta de conhecimento em base de dados	46
2.3	Maldição da dimensionalidade	49
2.4	Agrupamento de dados	51
3	Seleção de atributos	61
3.1	Seleção de atributos como problema de busca	61
3.2	Geração de sucessores	62
3.3	Estratégia de busca	64
3.4	Avaliação	65
3.5	Critério de parada	69
4	Algoritmos evolutivos	71
4.1	Evolução natural e a computação evolutiva	71
4.2	Algoritmo genético	73
4.3	Algoritmo genético construtivo	82
4.4	Algoritmo genético enviesado com chave aleatória	88
5	Modelagem	93
5.1	Considerações Gerais	93
5.2	Algoritmo genético	94
5.3	Algoritmo genético construtivo	95
5.4	Algoritmo genético enviesado com chave aleatória	98

6	Experimentação computacional	101
6.1	Bases de dados	101
6.2	Estrutura computacional	105
6.3	Resultados e discussão	106
7	Conclusão	119
8	Trabalhos Futuros	121
9	Limitações do Trabalho	123
	Referências Bibliográficas	125
A	Apêndice	135

Introdução

Até meados do século XX, os dados eram transformados em informação útil através de um processo de análise e interpretação exclusivamente manual. Pode-se citar como exemplo desse processo uma fábrica que registra as falhas de uma determinada máquina durante o mês. Ao final de um trimestre, o engenheiro analisa os dados registrados e emite um relatório que será utilizado em futuras decisões. O mesmo processo acontece em um laboratório de astronomia, quando pesquisadores analisam as características de objetos planetários e os categorizam para posterior utilização.

Não é incomum que este processo manual solicite mais de um especialista, já que a quantidade de dados a ser analisada pode ser grande e conhecimento prévio requerido, multidisciplinar. Estes especialistas se tornam profundos conhecedores dos dados e agem como interface entre estes e os usuários finais do conhecimento.

Com o surgimento da tecnologia da informação, o volume de dados cresceu de tal forma que impossibilitou a intervenção humana direta e exclusiva neste processo, exigindo assim sua automatização, mesmo que de forma parcial. Ainda são necessários especialistas no domínio dos dados para extrair o conhecimento, no entanto, ferramentas computacionais auxiliam grande parte deste processo.

Não apenas o volume dos dados cresceu, como também a velocidade em que eles são gerados e a variedade, visto que atualmente armazena-se números, imagens, vídeos, áudio, e informações genéticas (DNA). Este fenômeno, nomeado de *Big data*, representa as possibilidades e desafios de extrair conhecimento de grandes bancos de dados.

As atividades humanas que contribuem com o crescimento destes bancos vão desde atividades rotineiras, como o registro de um supermercado, os detalhes do uso de cartões de crédito e o detalhamento de chamadas telefônicas, até atividades mais sofisticadas, como o armazenamento de imagens de corpos astronômicos, banco de dados moleculares e registros médicos.

Em 2003, a humanidade já havia gerado um total de cinco exabytes de dados e, em 2008, em torno de quatorze e meio exabytes. Atualmente, cinco exabytes são gerados a cada dois anos e a produção de dados só tende a crescer (CANEDO; MARONO; BETANZOS,

2015).

Para ilustrar esta problemática, a Figura 1 mostra a evolução da quantidade de atributos nos bancos de dados de três repositórios (UCI KDD, UCI, LIBSVM), nas últimas duas décadas.

Figura 1 – Evolução da quantidade de atributos nos bancos de dados inseridos em três repositórios: (i) UCI KDD, em vermelho; (ii) UCI, em azul e (iii) LIBSVM, em verde, nas últimas duas décadas.

Fonte: Zhai, Ong e Tsang (2014). Adaptado pela autora.

Como é possível visualizar, ao passar dos anos a dimensionalidade dos bancos de dados aumentou. O primeiro registro do gráfico, a base de dados SOYBEAN do repositório UCI, inserida em 1988 contém 35 atributos. Já um dos últimos registros, a base de dados KDD2010, inserida em 2010, contém a maior dimensionalidade do gráfico, 29.890.095 atributos. Mais informações sobre as bases de dados do Gráfico 1 podem ser obtidas em Zhai, Ong e Tsang (2014).

Analisar esta quantidade de dados se tornou desafiador, estratégico e fundamental para o avanço de algumas áreas mercadológicas e para a ciência. Para que isso fosse possível, houve o desenvolvimento de métodos e técnicas, além da criação de conceitos e metodologias, todos estes provenientes da combinação de áreas como computação, estatística e matemática, para dar mais independência ao ser humano no processo de descoberta do conhecimento.

Este aglomerado teórico e prático fundamentou o que atualmente se conhece por processo de Descoberta de Conhecimento em Banco de Dados (DCDB), no inglês, *Knowledge Discovery in Databases* (KDD).

Uma das primeiras aplicações do processo de KDD na ciência foi na astronomia, atra-

vés da análise de imagens e classificação de objetos no espaço astronômico. Nos negócios, o processo de KDD está amplamente difundido. Alguns exemplos de aplicação são em (1) Marketing: identificação de diferentes grupos de clientes e previsão de comportamento; (2) Investimentos: gerenciamento de portfólio; (3) Detecção de fraudes: monitoramento de contas a fim de identificar transações financeiras que podem indicar uma atividade fraudulenta e (4) Manufatura: diagnóstico e predição em sistemas de resolução de problemas (FAYYAD; PIATETSKY-SHAPIO; SMYTH, 1996).

O processo de KDD é composto por três grandes etapas. A primeira, chamada de pré-processamento, realiza a preparação e a redução da base de dados. A segunda, Mineração de Dados (MD), no inglês, *Data Mining* (DM), aplica um algoritmo que realiza a descoberta de padrões no conjunto de dados. E, a terceira, especialistas interpretam o padrão encontrado, gerando assim o conhecimento em si.

A tarefa que reduz o tamanho do banco de dados é realizada para facilitar e até viabilizar a descoberta de padrões. Os algoritmos que procuram por estes padrões solicitam apenas variáveis relevantes e não redundantes ao objetivo da análise. Por isso, reduzir a dimensionalidade é uma tarefa fundamental para o processo de KDD.

Um destes algoritmos procuram por agrupamentos formados pelos dados, com o objetivo de identificar grupos menores e mais homogêneos, assim como mostra a Figura 2. Uma aplicação deste tipo de abordagem é a segmentação de mercado, na área de *marketing*.

Figura 2 – Diagramas de dispersão usando as variáveis (a) COL 1 e COL 2 e (b) COL 3 e COL 4.

Fonte: Liu e Ong (2008)

Caso o algoritmo de agrupamento fosse alimentado pelas variáveis COL 3 e COL 4, da Figura 2 (b), não haveria descoberta de grupos, já que os dados estão dispersos, ao contrário do que aconteceria com a utilização das variáveis COL 1 e COL 2, que resultaria na descoberta de três grupos.

Para ilustrar a importância da redução da dimensionalidade, não só para a viabilização do algoritmo de descoberta de padrões, mas também para a qualidade e veracidade da

análise, a Figura 3 ilustra a dispersão dos dados quando as variáveis (a) COL 1, COL 2 e COL 4 e as variáveis (b) COL 2, COL 3 e COL 4 estão selecionadas.

Figura 3 – Diagrama de dispersão usando as variáveis (a) COL 1, COL 2 e COL 4 e (b) COL 2, COL 3 e COL 4.

Fonte: Liu e Ong (2008)

Como se pode observar, a dispersão dos dados não indica a formação de nenhum grupo tão bem definido quanto os apresentados na Figura 2. Desta forma, fica evidenciada a importância de reduzir a dimensionalidade do conjunto de dados, permitindo que apenas variáveis relevantes e não redundantes permaneçam no processo de KDD.

Um outro exemplo também pode ser visto no trabalho de Fowlkes, Gnanadesikan e Kettenring (1988). Neste caso, cinco variáveis são analisadas, sendo duas delas relevantes, as quais indicam a formação de cinco grupos, como mostrado na Figura 4.

Figura 4 – Diagrama de dispersão de duas variáveis que formam cinco grupos.

Fonte: Fowlkes, Gnanadesikan e Kettenring (1988)

Através de um método de agrupamento chamado de ligação completa, em inglês *complete linkage*, os dados das duas variáveis são agrupados, e como esperado, mostram a presença de cinco grupos na Figura 5 (a), que por sua vez é nomeada de dendrograma. No entanto, quando o agrupamento conta com as cinco variáveis, o dendrograma resultante apresenta ruídos evidenciando a influência negativa que as variáveis não relevantes exercem sobre o resultado final do agrupamento, Figura 5 (b).

Figura 5 – Dendrogramas resultante do agrupamento dos dados.

Fonte: Fowlkes, Gnanadesikan e Kettenring (1988). Adaptado pela autora.

A interpretação do dendrograma é feita analisando como as observações se combinam e em que altura (distância) esta ligação é feita. Cada ligação indica a formação de um grupo. Na Figura 5 (a), os números em vermelho indicam a posição em que houve a formação dos cinco grupos visualizados no diagrama de dispersão da Figura 4.

A influência de variáveis irrelevantes e/ou redundantes no agrupamento de dados é negativa, levando a conclusões errôneas com relação à quantidade de grupos e sua existência. Identificar as variáveis relevantes ao problema alvo é de grande importância para a validação e qualidade da análise.

Milligan (1980) também traz um exemplo parecido com os exibidos até este ponto.

Sendo assim, uma das formas de reduzir a dimensionalidade através do número de atributos é selecionar um subconjunto de atributos que idealmente é necessário e suficiente para descrever um conceito alvo. A seleção de um subconjunto de atributos é um problema de busca, visto que há inúmeras possíveis combinações dos atributos do conjunto original. Saber como encontrar o melhor deles no espaço de soluções é o desafio a ser estudado neste trabalho.

Uma das estratégias de busca consiste em utilizar meta-heurísticas, como é o caso do Algoritmo Genético (AG) e suas variações, a fim de percorrer grande parte do espaço de solução em um tempo computacional viável. Apesar de não ser garantida a descoberta da solução ótima, as soluções encontradas são subótimas.

O uso de algoritmo genético em Seleção de Atributos (SA) foi apresentado pela primeira vez por Siedlecki e Sklansky (1989). Nas últimas décadas sua aplicação continuou por meio de outros trabalhos científicos, como em Morita et al. (2003), Rostami e Moradi (2014), Saha et al. (2015) e Swapna, Kumar e Murthy (2015).

A proposta deste trabalho, portanto, consiste em realizar uma busca por um subconjunto de atributos em agrupamento de dados utilizando como algoritmo de busca duas variações do algoritmo genético: Algoritmo Genético Construtivo (AGC) e Algoritmo Genético Enviesado com Chave Aleatória (AGECA), no inglês, *Biased Random-Key Genetic Algorithms* (BRKGA), visto que esta meta-heurística assim como suas variações se apresentam como ferramentas eficientes para tal fim. Ademais, não é de nosso conhecimento a implementação destes algoritmos no problema de SA.

A fim de esboçar o que foi exposto até este ponto, a Figura 6 traz uma ilustração da evolução do processo de análise e interpretação de dados ao longo dos anos, e enfatiza a proposta deste trabalho.

Figura 6 – Evolução histórica do processo de análise e interpretação de dados para obtenção de conhecimento com foco no assunto abordado neste trabalho, seleção de subconjunto de atributos utilizando duas variações do algoritmo genético.

Inicialmente o processo de análise e interpretação de dados era realizado apenas por seres humanos. Com o avanço da tecnologia da informação, houve a geração de uma grande quantidade de dados, e por isso o processo de descoberta de conhecimento necessitou do auxílio de ferramentas computacionais, matemáticas e estatísticas para ser realizado. Diante desta nova demanda, estruturaram-se as atividades, antes realizadas manualmente, em um processo de descoberta de conhecimento. Uma das tarefas iniciais deste processo consiste em reduzir a dimensionalidade da base de dados a fim de que seja possível a aplicação de algoritmos de descoberta de padrões. Uma das formas de redução consiste na seleção de um subconjunto de atributos que são necessários e suficientes para

a descrição dos dados. A seleção deste subconjunto é caracterizado como um problema de busca, já que há inúmeras formas de selecionar um subconjunto de atributos das variáveis originais. Sendo assim, uma das estratégias de busca consiste em utilizar algoritmos evolutivos, como é o caso do algoritmo genético e suas variações, por exemplo BRKGA e AGC. A aplicação destas duas variações do algoritmo genético para o problema de seleção de atributos não foi realizada até então, apesar de elas apresentarem bom desempenho quando aplicadas em problemas semelhantes.

1.1 Objetivo da pesquisa

Este trabalho tem como objetivo principal modelar e avaliar o problema da seleção de atributos em agrupamento de dados utilizando como algoritmo de busca por subconjuntos de atributos duas variações do algoritmo genético, sendo elas: Algoritmo Genético Construtivo e o Algoritmo Genético Enviesado com Chave Aleatória.

Estas duas variações apresentaram bons resultados quando foram aplicadas em outros problemas semelhantes na literatura. O ganho em desempenho diante da abordagem tradicional resultou das seguintes modificações, respectivamente:

1. Avaliação direta de esquemas (soluções¹ codificadas) através de duas funções de avaliação, permitindo assim a criação de novos indivíduos (soluções) de alta qualidade, o que contribui para a produção de melhores soluções;
2. Forma diferenciada de seleção e recombinação, bem como de decodificação dos indivíduos em soluções.

1.2 Justificativa

Métodos meta-heurísticos tem sido frequentemente apresentados pela comunidade científica para solução de problemas de otimização combinatória, como é o caso da seleção de atributos em agrupamento de dados (KIM; STREET; MENCZER, 2000). Entre os métodos mais utilizados estão o algoritmo genético, busca tabu e recozimento simulado, em inglês, *simulated annealing* (FURTADO, 1998).

Morita et al. (2003), Siedlecki e Sklansky (1989) e Liu e Ong (2008) destacam que para solução do problema de seleção de atributos, o algoritmo genético se apresenta atraente pela sua rapidez e eficácia na busca no espaço de soluções, pelo fato de evitar (sem garantias de obter êxito) que a busca fique presa em mínimos ou máximos locais e de poder combinar diversos objetivos em uma única função objetivo. Ademais, os autores relatam que este algoritmo se mostra uma ferramenta poderosa na seleção de atributos, principalmente quando o número de variáveis é alto, maior do que 20.

¹subconjunto de atributos

Os autores supracitados se referem ao algoritmo genético em sua abordagem tradicional, no entanto, com o objetivo de melhorar a eficiência computacional e a qualidade da solução encontrada, pesquisadores têm desenvolvido variações.

Uma destas variações é apresentada por Furtado (1998). Em seu trabalho, o autor a nomeia de algoritmo genético construtivo. O algoritmo apresenta uma nova proposta para realizar a avaliação dos indivíduos, bem como a seleção destes para a reprodução. O AGC se mostrou competitivo perante outras meta-heurísticas clássicas. Maiores detalhes estão presentes na Capítulo 4.

Goncalves e Resende (2011) destacam que estas diferenças se mostraram contribuidoras de um melhor desempenho em alguns problemas de otimização combinatória, como é o caso de formação de células de manufatura, empacotamento bidimensional e recursos limitados em programação de projetos.

Uma outra variação foi apresentada por Goncalves e Resende (2011) e, nomeada por eles de Algoritmo Genético Enviesado com Chave Aleatória. As principais diferenças desta variação para a abordagem tradicional são: o modo como o cromossomo é representado, a seleção dos cromossomos pais para a reprodução, a forma de realizar a recombinação e o algoritmo de decodificação. Maiores detalhes também estão presentes na Capítulo 4.

Pelo fato destas duas variações do algoritmo genético apresentarem resultados competitivos perante outras meta-heurísticas, conclui-se que estas são, portanto, fortes candidatas a serem aplicadas no problema de seleção de atributos em agrupamento de dados.

A aplicação destas duas variações em um problema de agrupamento de dados também se apresenta como um diferencial. O agrupamento de dados é um problema de aprendizado não supervisionado (maiores detalhes deste conceito será apresentado no capítulo 2), sendo assim, a forma de avaliar uma boa solução não é trivial. O aprendizado não supervisionado requer conhecimentos matemáticos e estatísticos mais aprofundados para sua resolução.

Uma busca nas principais bases de dados foi realizada a fim de verificar a existência de trabalhos relacionados à aplicação destas duas variações do algoritmo genético para o problema de seleção de atributos. Como resultado, não foram encontrados trabalhos relacionando estas duas variações ao problema em questão. Sendo assim, não é do nosso conhecimento o desenvolvimento destes algoritmos no problema exposto. Justifica-se, portanto, o uso do ACG e BRKGA no problema de SA em agrupamento de dados.

1.3 Estrutura do trabalho

Este trabalho está estruturado em nove capítulos. O primeiro, abordou conteúdos introdutórios. O segundo, diz respeito a terminologias e conceitos. O terceiro traz o referencial teórico sobre seleção de atributos. O quarto fala sobre algoritmos evolutivos, detalhando o algoritmo genético e as duas variações utilizadas nesta pesquisa. O quinto capítulo traz a modelagem dos algoritmos genéticos para o problema estudado, seleção de

atributos em agrupamento de dados. No sexto capítulo, a experimentação computacional é detalhada, bem como a apresentação dos resultados e discussão sobre eles. Em seguida, o capítulo sete apresenta a conclusão e o capítulo oito que traz os trabalhos futuros. Finaliza-se com o capítulo nove, com as limitações do trabalho realizado.

Terminologias e conceitos

Neste capítulo são apresentadas as principais definições, notações e terminologias que serão utilizadas neste trabalho. Também são apresentados brevemente alguns conceitos sobre o processo de descoberta de conhecimento em base de dados, a maldição da dimensionalidade e sobre agrupamento de dados. Este capítulo tem como objetivo, portanto, localizar teoricamente o que será aplicado nesta pesquisa dentro dos principais temas abordados. Além disso, objetiva-se fornecer definições básicas sobre os assuntos supracitados para fundamentar o entendimento dos capítulos subsequentes.

2.1 Definições, notações e terminologias

Há inúmeras formas de representar um conjunto de dados. No entanto, uma delas é amplamente utilizada por algoritmos de aprendizado de máquina, reconhecimento de padrões e banco de dados. Este formato, nomeado de atributo-valor, consiste em uma matriz em que: cada linha representa um registro, r_i , em que $i \in \{1, 2, 3, \dots, n\}$ e n denota o número total de registros; cada coluna representa um valor de uma variável, p_j , em que $j \in \{1, 2, 3, \dots, p_{co}\}$ sendo p_{co} o número total de variáveis e cada valor do registro de linha i e variável j é representado por $x_{i,j}$ (BARANAUSKAS, 2001) (LEE, 2005) (CANEDO; MARONO; BETANZOS, 2015), como se pode observar na Tabela 1.

Tabela 1 – Formato atributo-valor

Registro	Atributos			
	p_1	p_2	$\dots$	$p_{p_{co}}$
r_1	$x_{1,1}$	$x_{1,2}$	$\dots$	$x_{1,p_{co}}$
r_2	$x_{2,1}$	$x_{2,2}$	$\dots$	$x_{2,p_{co}}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
r_n	$x_{n,1}$	$x_{n,2}$	$\dots$	$x_{n,p_{co}}$

Um registro, também denominado na literatura de *exemplo*, *caso*, *objeto*, *observação*,

sujeito ou *dado*, é um vetor de variáveis, que por sua vez, também é denominada de *característica* ou *atributo*. O uso dos diferentes termos encontrados na literatura para referenciar registro e variável são utilizados de forma sinonima neste trabalho.

Este vetor pode ser exemplificado através do registro de um produto no estoque de um supermercado. As possíveis variáveis que estariam presentes no vetor seriam o preço do produto, descrição, valor do imposto e uma ilustração (BARANAUSKAS, 2001).

Com o exemplo dado é possível observar que há diferentes tipos de variáveis, como o preço que é numérico, a descrição que é representada por um texto e a ilustração, por uma imagem. Para facilitar esta classificação, a literatura definiu uma hierarquia para o tipo de variáveis, como mostra a Figura 7.

Figura 7 – Hierarquia do tipo de variáveis.

Fonte: Lee (2005). Adaptada pela autora.

As variáveis cujos dados são do tipo categórica representam uma classe ou uma ordem. Por isso, há uma subclassificação deste tipo de variável em ordinais, dados que representam uma ordem, e nominais, dados que representam uma classe, nome.

As variáveis do tipo numérica expressam o significado do próprio número. Podem assumir inúmeros valores se forem subclassificadas como contínuas, pois apresentam domínio do conjunto dos reais ($\mathbb{R}$). Há também o caso da variável ser subclassificadas como discretas.

Já as variáveis do tipo complexa representam tipos não usuais de dados como áudio, imagens e DNA. E, por fim, o tipo de variável composta representa dados que são uma combinação dos tipos anteriores (LEE, 2005).

Este vetor de variáveis ainda pode receber uma classificação, também chamada de *rótulo*, armazenada em uma outra variável no vetor. No exemplo do supermercado, um produto pode ser classificado como "Básico", "Supérfluo" ou "Luxuoso" dependendo dos valores das outras variáveis. Ainda é possível classifica-lo como "Industrializado", "Natural" ou "Orgânico". Outras classificações são possíveis, e se elas existirem, a Tabela 1 con-

tará com mais uma coluna, que representará a classe de cada registro, vide Baranauskas (2001).

A presença de rótulos na base de dados norteia a escolha do algoritmo de aprendizado que é aplicado na descoberta de padrões. É dado o nome de aprendizado supervisionado a abordagem que trabalha com dados rotulados e de aprendizado não supervisionado àquelas que trabalham com dados não rotulados.

A Figura 8 traz a hierarquia do aprendizado, bem como as opções de aplicação em cada ramificação, destacando o que será utilizado neste trabalho.

Figura 8 – A hierarquia do aprendizado.

Como resultado do aprendizado supervisionado obtém-se a predição, através de algoritmos de classificação e regressão. Já com o aprendizado não supervisionado, tem-se como resultado a descrição dos dados, através da sumarização, associação e agrupamento.

É importante destacar que alguns algoritmos de aprendizado aceitam apenas um tipo de variável, sendo nestes casos, necessária a transformação dos dados. Ainda há o caso de algoritmos que aceitam todos os tipos de variáveis, como o utilizado neste trabalho, *k*-médias, em inglês, *k-means*, com a aplicação das devidas medidas de distância. No entanto, neste caso, ainda é necessário verificar a necessidade de padronizar os valores das variáveis para que haja um aumento no desempenho dos resultados.

Além do formato de representação do conjunto de dados, tipos de variáveis e as diversas nomenclaturas encontradas na literatura para referenciar registro e variável, é igualmente importante identificar a relação entre o número de variáveis e o número de grupos em agrupamento de dados, assim como declarar o que se entende por variáveis redundantes e irrelevantes neste trabalho.

Sendo assim, Hruschka e Covoos (2005) mostram a relação entre o número de variáveis e número de grupos na Figura 9, através de três exemplos gráficos.

Figura 9 – Três exemplos gráficos da relação entre o número de grupos e o número de variáveis em agrupamento de dados.

Fonte: Hruschka e Covoos (2005)

É possível observar que a Figura 9 (a) apresenta dois grupos bem definidos para quando as variáveis x e y são selecionadas. Se apenas a variável y for considerada, também tem-se dois grupos, o que não ocorre com a seleção exclusiva de x . Caso o critério numérico escolhido para definir o melhor agrupamento privilegiar a separação dos grupos, a variável x pode ser classificada como irrelevante.

A Figura 9 (b) também indica a formação de dois grupos bem definidos quando as variáveis x e y são selecionadas. Nota-se que se a variável x ou a variável y for selecionada, o número de grupos ainda será dois. Neste caso, define-se que as variáveis x e y são redundantes, ou seja, levam a mesma informação, sendo necessária apenas uma delas para esboçar a estrutura dos dados, no caso, dois grupos.

A Figura 9 (c) indica a formação de três grupos quando as variáveis x e y são selecionadas. O mesmo número de grupos é encontrado caso apenas a variável x seja selecionada. No entanto, apenas dois grupos se formam se for considerada apenas a variável y . Caso o critério numérico para medir a qualidade de agrupamento aferir a densidade dos grupos, o algoritmo de seleção indicaria que apenas a variável y contribui com a melhor descrição dos dados.

Não há um consenso na literatura sobre o melhor critério numérico para ser adotado e tampouco qual a medida de importância a ser considerada para os atributos. Apenas é usual considerar a densidade e separação dos grupos. É possível que o algoritmo de SA, ao ser submetido a dois critérios de avaliação de agrupamento diferentes indique dois subconjuntos ótimos ou subótimos diferentes.

O termo *importância* também é expresso como *relevância* ou *critério de avaliação* de atributos. Estes termos, que são equivalentes, exprimem a ideia de que cada atributo contribui, com intensidades e direções diferentes para a descrição dos dados. Liu e Motoda

(1998) traz uma definição para importância de um atributo. Outras definições podem ser encontradas em Blum e Langley (1997), Kohavi e John (1997) e Lee (2000).

Definição (*Importância de um Atributo*) *Um atributo é considerado importante quando, se removido, a medida de importância considerada em relação aos atributos restantes é deteriorada.*

Os mesmos autores que declaram a definição acima, elencam e hierarquizam cinco medidas de importância, exibidas na Figura 10.

Figura 10 – Medidas de importância de atributos.

Fonte: Liu e Motoda (1998)

O primeiro desdobramento que ocorre na hierarquia é referente à dependência da medida em relação ao algoritmo de indução. Apenas a medida de precisão é dependente do algoritmo de aprendizado, as outras quatro medidas são independentes do mesmo.

As medidas de importância que priorizam a separabilidade dos grupos ou das classes ainda podem ser divididas em duas categorias, sendo a primeira composta das medidas de informação, distância e dependência, e a segunda composta apenas pela medida de consistência.

As medidas que compõem a primeira categoria são chamadas de clássicas, pois estão intimamente relacionadas. Uma breve descrição de cada medida é realizada a seguir. Mais detalhes estão presentes no trabalho de Lee (2005).

❑ **Medida de informação:** esta medida verifica o quanto de informação é obtida com a inclusão de um atributo no conjunto de dados. Uma das maneiras de mensurar este ganho é pela teoria da entropia (FAST, 1962), que diz que a entropia é menor para configurações organizadas e maior para uma configuração desorganizada, pelo simples fato de que haver mais configurações desorganizadas do que organizadas. Sendo

assim, um atributo é dito relevante se sua inclusão no subconjunto de atributos diminuir a entropia.

- ❑ **Medida de distância:** esta medida considera relevante o atributo que contribui com a separabilidade dos grupos ou classes. Caso um atributo p_i seja adicionado ao subconjunto de atributos e como resultado forem obtidos grupos ou classes mais separadas, com maiores distâncias entre elas, do que no subconjunto anterior, então o atributo é selecionado para compor o subconjunto final. A distância euclidiana é uma das formas de calcular esta medida, que por sua vez também é nomeada de medida de *separabilidade*, *divergência* ou *discriminação*.
- ❑ **Medida de dependência:** são medidas que quantificam o quanto uma variável prediz a outra, ou seja, o quanto uma variável prevê a outra. Também são conhecidas pelos nomes de medida de *correlação* ou *associação*. A predição está relacionada com a correlação entre duas variáveis, por isso, uma forma de mensura-la é através do coeficiente de correlação. Em aprendizado supervisionado, é dito que um atributo p_i é mais relevante do que outro se o seu coeficiente de correlação em relação a uma classe (rótulo) é maior do que o coeficiente de correlação de outro atributo p_j . Também é possível mensurar o grau de redundância entre atributos, através do coeficiente de correlação entre eles.
- ❑ **Medida consistência:** esta medida se diferencia das outras por ser fortemente dependente do conjunto de treinamento. Um subconjunto mínimo de atributos é selecionado caso satisfaça a proporção de inconsistência aceita, geralmente definida pelo usuário. Entende-se por inconsistência dois casos iguais, ou seja, com valores de atributos iguais, mas que pertencem a classes diferentes. Um problema desta medida está no fato de que não é possível identificar atributos redundantes.
- ❑ **Medida de precisão:** esta medida se refere à tarefa de predição. O subconjunto de atributos que obtiver a maior precisão é selecionado. É usual utilizar o mesmo algoritmo que processa os dados para realizar a tarefa de seleção de atributos.

2.2 Descoberta de conhecimento em base de dados

Inicialmente, muitos nomes foram atribuídos à ideia de encontrar padrões utilizáveis nos dados, como mineração de dados, extração de conhecimento, descoberta de informações, entre outros. Em 1989, o conceito se solidificou com a frase "Descoberta de Conhecimento em Banco de Dados", no inglês, *Knowledge Discovery in Databases*, definindo assim que o conhecimento é o produto final de um processo de descoberta orientada por dados (FAYYAD; PIATETSKY-SHAPIO; SMYTH, 1996).

Sendo assim, entende-se que o processo de KDD está estruturado em fases bem definidas, como mostra a Figura 11.

Figura 11 – Etapas mais detalhadas do processo de KDD.

Fonte: Fayyad, Piatetsky-Shapiro e Smyth (1996). Traduzido pela autora.

Com o objetivo de identificar os padrões válidos, potencialmente úteis e compreensíveis, além da descrição de forma clara destes dados em alguma linguagem, o processo da Figura 11 é descrito por Fayyad, Piatetsky-Shapiro e Smyth (1996) em nove passos, divididos em três grandes etapas: pré-processamento, MD e pós-processamento.

A etapa de pré-processamento compreende o intervalo do primeiro ao quarto passo, sendo que o **primeiro** conta com o entendimento do domínio da aplicação, conhecimento dos dados através de estatística descritiva e técnicas de visualização e definição do objetivo a ser alcançado com a análise. O **segundo** passo representa a definição dos dados alvos. O **terceiro**, consiste no processamento destes dados, que envolve a limpeza de ruídos e o tratamento dos dados faltantes. E, por último, o **quarto** passo é representado pela transformação destes dados e atributos, onde objetiva-se a transformação do tipo de dado, normalização e a redução da dimensionalidade.

A redução tem como objetivo entregar ao algoritmo de MD apenas atributos que contribuirão com o descobrimento de padrões. Uma das formas de redução consiste em selecionar um subconjunto de variáveis que idealmente é necessário e suficiente para descrever os dados (KIRA; RENDELL, 1992) (BLUM; LANGLEY, 1997). Um dos métodos que realizam este tipo de redução é nomeado de Seleção de Atributos, cujo detalhes estão presentes no Capítulo 3.

Além de ter como objetivo o refino dos dados a fim de melhorar o desempenho do algoritmo de MD, eliminando o ruído e redundância, os métodos de SA também primam pelo simples funcionamento destes algoritmos quando a quantidade de dados a ser analisada é grande. Neste caso, a etapa de pré-processamento é ainda mais importante para

o funcionamento do algoritmo de aprendizagem, uma vez que excede-se a capacidade de processamento caso a dimensionalidade não seja reduzida.

Lee (2005) também destaca a importância que esta etapa tem na garantia de uma boa qualidade dos dados, realçando que esta etapa é uma das tarefas mais trabalhosas do processo de KDD e que aproximadamente 80% do tempo total do processo é dedicado a ela. Apesar da importância, Baranauskas (2001) relata que a literatura fornece pouca atenção para as tarefas desta etapa por necessitar de conhecimento prévio sobre domínio, importância de atributos e objetivo da aplicação.

A etapa de MD compreende o intervalo do quinto ao sétimo passo, sendo o **quinto** referente a combinação do objetivo traçado na primeira etapa com o objetivo do algoritmo de aprendizado, objetivo este que pode ser uma classificação, regressão, agrupamento, entre outros. O **sexto** passo envolve a escolha do algoritmo e seleção de parâmetros do método de busca de padrões. E, o **sétimo**, envolve a procura de padrões em si, ou seja, a aplicação do algoritmo no conjunto de dados.

A etapa de pós-processamento compreende o **oitavo** e o **nono** passo, que por sua vez exercem a interpretação dos padrões com a ajuda de técnicas de visualização e com o auxílio do especialista de domínio, profissional que apresenta profundo conhecimento da área de aplicação. O conhecimento descoberto é fornecido para alimentação de outros sistemas ou simplesmente para a documentação e divulgação para fontes interessadas.

As nove tarefas descritas e agrupadas em três grandes etapas são assim definidas no trabalho de Fayyad, Piatetsky-Shapiro e Smyth (1996). No entanto, outros autores dividem o processo de KDD de outras maneiras, como é o caso de Weiss e Indurkha (1998) que o descreve em 4 etapas e o caso de Baranauskas (2001) que o descreve em três.

É comum encontrar o termo MD sendo utilizado no lugar de KDD. Por isso, é importante destacar que conceitualmente a MD é uma etapa do processo de KDD, antecedida pelas tarefas de pré-processamento e seu objetivo é encontrar um padrão nos dados, contribuir para a extração de informação útil.

A MD é interdisciplinar, fazendo interface com o aprendizado de máquinas, reconhecimento de padrões e com a estatística para poder cumprir seu papel neste processo (HAND; SMYTH; MANNILA, 2001 apud COVOES, 2010).

Fayyad, Piatetsky-Shapiro e Smyth (1996) destacam que o processo conta com decisões do usuário em diversas fases, sendo por isso denominado um processo interativo. Também define-se que o processo é iterativo, pois há a repetição de alguns ou até de todos os passos. Além disso, o mesmo é classificado como não-trivial, já que buscas e inferências são realizadas nas diversas tarefas.

O núcleo deste trabalho está localizado na tarefa de transformação, uma vez que a SA compete a esta etapa. Além disso, o algoritmo de MD utilizado é o *k-means*, algoritmo de agrupamento de dados, que será detalhado ainda neste capítulo.

2.3 Maldição da dimensionalidade

Segundo Donoho (2000) o termo maldição da dimensionalidade foi expresso pela primeira vez por Bellman em 1957, para relatar a dificuldade em otimizar por enumeração exaustiva um espaço de soluções (BELLMAN, 1961). O termo também faz referência ao surgimento de diversos fenômenos nocivos que ocorrem na tentativa de organizar e analisar os dados em altas dimensões.

Por estas dificuldades, surgiu a necessidade de reduzir a dimensionalidade dos conjuntos de dados para que fosse possível a extração do conhecimento. São duas as principais motivações que levam à redução, ambas com o objetivo de se obter maior representatividade, diminuir o tempo de execução do algoritmo e até viabilizar a tarefa de MD.

A primeira é referente à quantidade de dados. Esta se baseia na relação entre a dimensionalidade e o volume do espaço. Quando se aumenta uma dimensão, o volume do espaço aumenta exponencialmente. Para manter a mesma densidade e por consequência a qualidade dos resultados, a amostra de dados deve aumentar na mesma proporção que o volume do espaço. A Figura 12 ilustra a relação entre o número de dimensões e o volume do espaço.

Figura 12 – Relação entre o número de dimensões e o volume do espaço.

Fonte: Bishop (2006). Adaptada pela autora.

A segunda motivação é referente à contribuição ou não de variáveis para a descrição dos dados. Manter variáveis não significativas pode implicar na descoberta de padrões de baixa qualidade. Já a exclusão ou transformação destas elimina a redundância e o ruído (HRUSCHKA; COVOES, 2005).

Esta segunda motivação requer que operações sejam realizadas nos atributos para que estes contribuam melhor para a descrição dos dados. Lee (2005) traz na Figura 13 uma hierarquia das operações que podem ser executadas, onde p_{exo} representa o número de atributos do subconjunto obtido após a execução da operação em questão e p_{co} representa o número de atributos do conjunto original.

Figura 13 – Hierarquia das operações com atributos.

Fonte: Lee (2005)

As duas grandes operações são a transformação e a seleção de atributos. A diferença entre elas reside na permanência do significado físico das variáveis originais no conjunto de variáveis geradas após a aplicação da operação em questão.

A operação de transformação muda o significado físico das variáveis, o que não ocorre nas operações de seleção. Tendo em vista que p_{exo} representa a quantidade de atributos no subconjunto de variáveis e p_{co} representa o número de atributos totais do conjunto de dados, as suboperações de Transformação de Atributos (TA) podem ser descritas como:

- ❑ **Construção:** criação de novos atributos para melhor representar o conceito presente nos dados. Há um aumento do número de atributos neste caso ($p_{exo} > p_{co}$). Como exemplo, pode-se citar a criação do atributo velocidade, representando a razão entre o atributo distância e o atributo tempo.
- ❑ **Extração:** redução do número de atributos através da combinação linear e não-linear das variáveis originais. Como exemplo de um método de extração, pode-se citar a Análise de Componentes Principais, no inglês, *Principal Components Analysis* (PCA), introduzida por Pearson (1901). O exemplo supracitado da velocidade também pode ser utilizado nesta suboperação, no entanto, os atributos originais, distância e/ou tempo, seriam desconsiderados no subconjunto final ($p_{exo} < p_{co}$).
- ❑ **Discretização:** neste caso os atributos de valores numéricos são transformados a fim de que só contenham valores categóricos. Esta transformação é feita definindo intervalos de valores e atribuindo a cada um deles um valor inteiro. Posteriormente, analisa-se o valor numérico do atributo e o substitui pelo respectivo valor discreto do intervalo ao qual ele pertence. O subconjunto final tem uma quantidade de variáveis igual ou maior do que o conjunto original ($p_{exo} \Rightarrow p_{co}$).

As suboperações de seleção de atributos podem ser descritas como:

- ❑ **Subconjunto (SSA):** operação que escolhe um subconjunto mínimo de variáveis entre as variáveis originais que melhor represente o padrão dos dados. Esta escolha é feita diante de um critério de avaliação de atributos. O subconjunto final, menor ou igual ao conjunto original ($p_{exo} \leq p_{co}$), pode ser definido apenas considerando a contribuição individual de cada atributo como também se pode considerar, além da contribuição individual, a contribuição ou não da interação entre eles.
- ❑ **Ranking:** operação que consiste em apenas ordenar os atributos em relação a relevância de cada um. Esta relevância é sinônimo do critério de avaliação escolhido. O subconjunto final tem a mesma quantidade de variáveis que o original ($p_{exo} = p_{co}$).

Lee (2005) destaca que o termo SA e SSA tem sido utilizado por muitos pesquisadores para descrever a mesma tarefa. Além disso, a autora também salienta que o termo SA já foi utilizado para representar tanto as operações de SSA como as de construção de atributos. Neste trabalho, os termos citados, SA e SSA, são utilizados com o mesmo significado, seleção de um subconjunto de atributos, onde $p_{exo} \leq p_{co}$, não compreendendo a construção de atributos, apenas a seleção dos mesmos do conjunto original.

2.4 Agrupamento de dados

O termo análise de agrupamento, em inglês, *clustering analysis*, é também conhecido como análise Q , construção de tipologia, análise de classificação, taxonomia numérica. Teve sua origem em uma das pesquisas de Sokal e Sneath (1963), quando os autores publicaram um trabalho sobre os princípios da taxonomia numérica. A partir disso, a comunidade científica se motivou a estudar técnicas de agrupamento de dados (BOCK, 2008).

Atualmente, o agrupamento de dados representa um conjunto de técnicas multivariadas que são responsáveis por realizar o agrupamento de dados com objetivo de representar a estrutura natural destes, agregando os objetos com base nas características que eles possuem.

Jain e Dubes (1988) elencam três definições para o termo *agrupamento*, sendo que a primeira define que um agrupamento é um conjunto de entidades que são semelhantes, e entidades de diferentes grupos que são não iguais. A segunda diz que um agrupamento é uma agregação de pontos em um espaço teste, de modo que a distância entre qualquer dois pontos em um grupo é menor do que a distância entre qualquer ponto do grupo e qualquer ponto fora do grupo. E, a terceira, relata que um agrupamento pode ser descrito como regiões conectadas de um espaço multidimensional, sendo que os grupos são caracterizados por um espaço cuja função densidade de probabilidade é alta, separada de outras regiões com valores da função densidade de probabilidade baixos.

A representação dos agrupamentos naturais intrínsecos nos dados é almejado por diversas ciências para resolução de inúmeros desafios.

O simples agrupamento de objetos, pessoas, empresas e até mesmo comportamentos semelhantes contribuem com as ciências biológicas, sociais, econômicas, áreas da engenharia e da administração para a solução de problemas como a segmentação de mercado, análise de perfis psiquiátricos, criação de taxonomias biológicas, entre outros.

O ser humano é capaz de identificar um agrupamento em gráficos de dispersão de duas, e no máximo, três dimensões. Esta identificação e descrição dos grupos naturais formados pelos dados é executada pelo cérebro humano de forma natural e automática. Porém, a limitação da capacidade humana não permite a mesma execução em dimensões maiores (HAIR; ANDERSON; TATHAM, 2005) (JAIN; DUBES, 1988).

Além disso, a identificação e descrição de grupos é subjetiva, visto que os relatos advindos de diferentes pessoas sobre a quantidade, o tamanho e a forma dos possíveis grupos de um conjunto de dados não são idênticos. Isto acontece porque esta análise depende de fatores como a educação e questões culturais dos analistas (JAIN; DUBES, 1988).

Através do auxílio computacional, torna-se possível realizar o agrupamento em dimensões maiores, além da validação numérica da existência e da quantidade k de grupos presentes nos dados. Um exemplo disto se dá através de métodos que buscam a maximização da homogeneidade entre os objetos do mesmo grupo e da heterogeneidade dos objetos de grupos diferentes.

Apesar deste auxílio se apresentar como facilitador do processo de agrupamento, ainda há desafios a serem contornados. Um deles é a definição do número de grupos, k , que é fornecido como parâmetro de entrada para a maioria dos algoritmos de agrupamento. Apesar de ser desafiador sua definição, (RICHARDSON; GREEN, 1997) apresenta um método de obter tal número estatisticamente.

Para estes casos, segundo Kaufman e Rousseeuw (1990), a maioria das abordagens elegem critérios numéricos que apontam o melhor número de grupos, fundamentando-se no valor de k para diversas partições. Pode-se citar como critérios utilizados pela literatura a silhueta (KAUFMAN; ROUSSEEUW, 1990), o índice de Dunn (DUNN, 1974) e o índice Critério de Agrupamento Cúbico (CAC), no inglês Cubic Clustering Criterion (CCC), (SARLE, 1993), o qual será apresentado com mais detalhes ainda neste capítulo.

A fim de explicitar as principais decisões do processo de agrupamento e de descrever as principais opções de escolha presentes em cada um, a Figura 14 traz uma visão geral do diagrama que resume as decisões primordiais deste processo.

Na primeira decisão, problema de pesquisa, define-se o objetivo almejado com a identificação de grupos. São de três tipos os objetivos que podem ser alcançados com a obtenção de agrupamentos: (i) a descrição taxonômica; (ii) a simplificação de dados e (iii) a identificação de relações. Realiza-se também a seleção de variáveis, que deve ser fundamentada

teoricamente, com base em pesquisas anteriores ou suposições.

A técnica de agrupamento de dados em si não tem meios para diferenciar variáveis relevantes das irrelevantes, ficando a cargo do pesquisador esta análise ou de métodos de apoio. A seleção de variáveis, tema central deste trabalho de pesquisa, pode ser realizada através de distintos métodos, que serão descritos com mais detalhes no Capítulo 3.

Na **segunda** decisão, seleciona-se a medida de similaridade que será empregada, ou seja, define-se como será a medição de proximidade entre os objetos. O conceito de similaridade entre objetos é de extrema importância para este tema, sendo definida como medida de correspondência ou semelhança entre objetos a serem agrupados.

Apesar da similaridade poder ser mensurada de diversas maneiras, três medidas dominam a aplicação, como segue:

- ❑ **Medidas correlacionais:** revelam a associação entre variáveis pela semelhança de padrões no espaço de características e não pela magnitude. Um grupo se define pela junção de variáveis altamente correlacionadas, positivamente ou negativamente. Correlações altas indicam associações e correlações baixas indicam a falta de associações, ou seja, indicam a dissimilaridade. Medidas correlacionais são raramente utilizadas, visto que a maioria das aplicações de análise de agrupamento enfatizam a magnitude dos dados e não os padrões que os mesmos apresentam (HAIR; ANDERSON; TATHAM, 2005).
- ❑ **Medidas de distância:** são medidas baseadas na distância entre os objetos. Diz-se que esta medida é uma medida de dissimilaridade, visto que quanto maior a distância, menor a similaridade dos dados. Para representá-la como uma medida de similaridade, a distância é transformada pela relação inversa. Esta medida é a mais frequentemente empregada pelos métodos de agrupamento de dados e se concentram na magnitude dos mesmos. Pode-se citar como medidas de distância a distância euclidiana, euclidiana quadrada ou absoluta, abordagem *city-block*, distância de Mahalanobis, entre outras (HAIR; ANDERSON; TATHAM, 2005).
- ❑ **Medidas de associação:** esta medida avalia o grau de concordância entre pares de respondentes. Este grau de concordância, por sua vez, pode ser medido pelo percentual de vezes que ocorre concordância entre os pares. Em um campo cujos dados são do tipo sim ou não, o grau de concordância indicaria a percentagem de respondentes em que ambos dizem sim ou em que ambos dizem não a uma mesma pergunta.

As medidas de distância e de correlação são empregadas quando os dados são métricos. Já as medidas de associação são utilizadas em dados que qualificam características ou categorias que identificam ou descrevem o objeto.

Figura 14 – Diagrama resumido de decisão da análise de agrupamento.

Fonte: Hair, Anderson e Tatham (2005). Adaptado pela autora.

Ainda na segunda decisão, há a padronização dos dados quando a medida de associação se apresenta sensível a diferentes escalas e magnitudes. A padronização pode ser realizada pela conversão de cada variável em valores padrão (escore Z), que ocorre pela subtração da média e divisão pelo desvio-padrão para cada variável. Também é possível a padronização pelas observações, vide Hair, Anderson e Tatham (2005).

A **terceira** decisão concentra em duas questões críticas para a análise de agrupamento: (i) representatividade da amostra e (ii) impacto da multicolinearidade. Quando a representatividade da amostra é garantida pela aleatorização, apenas é possível esperar que a estrutura de grupos presente na amostra reflita a mesma da população. Por esta razão, a importância da representatividade da amostra se mostra como um dos pontos chaves nesta etapa. A outra questão, impacto da multicolinearidade, se mostra crítica uma vez que variáveis redundantes distorcem a estrutura de agrupamento dos dados pelo processo de inversão da matriz. Uma das formas de solucionar este problema é realizar apenas a seleção de atributos relevantes.

Na decisão **quatro**, a escolha do algoritmo de agrupamento é realizada. Com as variáveis selecionadas e a medida de similaridade definida, o algoritmo de agrupamento inicia o processo de partição dos dados. Esta partição pode ocorrer de duas formas, hierarquicamente e não hierarquicamente.

A primeira, forma hierárquica, consiste em agrupar os dados através de uma construção do tipo árvore. Esta construção também pode ocorrer de duas formas. Na primeira, método aglomerativo, o particionamento acontece através da aglomeração dos dados, ou seja, da união de observações individuais, uma a uma, priorizando os pares que apresentem mais similaridades. Este procedimento recebe o nome de método aglomerativo. Na segunda forma, método divisivo, o particionamento consiste em começar a análise considerando o conjunto de dados como um único grupo, um grande aglomerado. Nos passos que sucedem, a criação de grupos menores é realizada através da separação de observações que apresentam mais dissimilaridades, ou seja, observações mais diferentes entre si. Nomeia-se de método divisivo o procedimento descrito.

Ambos os métodos da forma hierárquica recebem o auxílio visual de gráficos chamados de dendrogramas para descrever a estrutura do agrupamento. Na Figura 15 é possível observar onde reside a diferença das duas formas hierárquicas. O dendrograma dos métodos aglomerativos são construídos da esquerda para a direita, já o dendrograma dos métodos divisivos são construídos da direita para a esquerda.

A interpretação do dendrograma da Figura 15 pelo método aglomerativo se inicia no eixo vertical, onde se encontra as observações. É possível constatar que as observações 1 e 2, e as observações 5 e 6 foram as primeiras a serem unidas, visto que são mais similares entre si. A dissimilaridade entre elas é apresentada no eixo horizontal, indicando que o par 1 e 2 é mais similar do que o par 5 e 6, uma vez que o primeiro se une quando o valor do eixo horizontal é de 1 unidade e o segundo, quando o valor é de duas unidades. As

uniões continuam até a formação de um único grupo que contém todas as observações do conjunto de dados.

Figura 15 – Dendrograma.

Fonte: Hair, Anderson e Tatham (2005). Adaptado pela autora.

Também é possível realizar a interpretação da Figura 15 pelo método divisivo. Para este caso, o início se dá com apenas um grupo contendo todas as observações. Um passo a frente, há a separação do grupo que contém os registros 1, 2, 3 e 4, do grupo que contém os demais dados do conjunto, visto que a maior dissimilaridade residia na ligação entre os grupos citados. Os passos que sucedem também realizam a quebra das ligações mais dissimilares, convergindo para um número de grupos igual ao número de observações do conjunto.

A segunda forma de partição, a forma não hierárquica, envolve a definição *a priori* do número k de grupos desejados. Diante disso, o algoritmo cria k pontos que são chamados de centróides a partir dos quais os grupos serão gerados. A criação e posição destes pontos se apresenta como um desafio na obtenção de resultados consistentes, requerendo assim uma base prática, objetiva e teórica do pesquisador.

Um dos algoritmos mais conhecidos que realizam o agrupamento pela forma não hierárquica é conhecido como k -médiãs, e será descrito mais detalhadamente até o final deste capítulo.

Além das duas formas de partições supracitadas, há a possibilidade de realizar o agrupamento pela combinação das abordagens descritas. Normalmente, a contribuição do método hierárquico nesta combinação é fornecer os centróides para o algoritmo de agrupamento não hierárquico.

Ainda na decisão de número quatro da Figura 14, é analisada a quantidade de grupos k presente nos dados, das maneiras que seguem: (i) examinar o aumento ou diminuição

relativa do critério de qualidade de agrupamento para diferentes valores de k ; (ii) examinar o dendrograma no caso dos métodos hierárquicos; (iii) considerar questões conceituais para a validação do número de grupos.

Já nas decisões de número **cinco e seis**, o processo de agrupamento é finalizado realizando a interpretação dos grupos, que inclui a análise dos centróides e nomeação dos grupos diante das variáveis selecionadas, além da validação da presença de grupos naturais.

Após dessa breve descrição sobre as principais atividades realizadas no processo de análise de agrupamento, é importante detalhar o algoritmo k -médias e o índice CCC, visto que estes dois tópicos estão presentes no desenvolvimento da modelagem proposta.

K-médias

O algoritmo k -médias, um dos mais tradicionais da mineração de dados teve sua origem em diversos campos científicos de forma independente (JAIN, 2010). Os trabalhos elencados como descobridores do algoritmo são os de Steinhaus (1956), Ball e Hall (1967), MacQueen (1967) e Lloyd (1982), sendo o de MacQueen (1967) o mais aceito.

Classificado como não hierárquico, o algoritmo divide os n registros da base de dados em k grupos mutuamente exclusivos, sendo $k \leq n$. Este agrupamento acontece elegendo, de maneira aleatória, k centróides no domínio do conjunto com o objetivo de cada um ser o representante inicial do grupo. Os outros registros do conjunto são atribuídos aos grupos através da distância que os mesmos se encontram dos centróides (CAMILO; SILVA, 2009).

Um registro é unido a um centróide específico se a distância entre eles for a menor entre todas as demais distâncias entre o mesmo registro e os outros centróides. Esta análise da distância é feita para os demais registros e o valor do centróide é recalculado, através da média ponderada, a medida de uma nova observação passa a integrar o grupo. O algoritmo cessa a execução quando atinge um número máximo de iterações ou quando atinge o limiar mínimo de mudança nos centróides.

A complexidade computacional deste algoritmo é de $O(nkq)$, em que n representa o número de objetos a serem agrupados, k a quantidade de grupos e q representa o número de iterações. O pseudocódigo deste algoritmo segue abaixo.

Algoritmo 1 Algoritmo de agrupamento k -means

Definir o número de grupos, k

Associar cada registo para um grupo

enquanto algum registo mudar de grupo **faça**

 Calcular o centróide de cada grupo

 Reorganizar os registos de modo que cada um pertença ao grupo mais próximo

fim

Fonte: Linden (2008), com adaptações da autora.

É importante destacar que o algoritmo é iterativo e que, sob uma perspectiva de otimização, o algoritmo *k-means* preza por minimizar a soma da distância euclidiana ao quadrado entre os objetos dos grupo até seu respectivo centróide, como mostra a Equação 1 (LLETI et al., 2004).

$$J = \sum_{c=1}^k \sum_{x_j \in C_c} = dist(x_j, \bar{x}_c)^2 \quad (1)$$

Em que c representa o grupo em questão, x_j o objeto do conjunto de dados, $\bar{x}_c$ o centróide do grupo c e $dist$ representa a distância euclidiana ao quadrado.

O resultado, portanto, apresenta um conjunto de grupos compactos e mais separados possível um dos outros.

Apesar de muito utilizado em trabalhos científicos e práticos, o algoritmo k-médias apresenta alguns inconvenientes, reportados pela literatura. Alguns exemplos são: assumir que o número de grupos é conhecido e portanto, requisita-lo como entrada do algoritmo; ser muito sensível às condições iniciais, como definição dos centróides; convergir rapidamente para um mínimo local e aceitar apenas variáveis numéricas, quando a função de otimização lidar com a distância euclidiana.

Diante dessas particularidades, muitos autores propuseram modificações no algoritmo a fim de amenizar estas singularidades (RALAMBONDRAINY, 1995) (PENA; LOZANO; LAR-RANAGA, 1999) (LIKAS; VLASSIS; VERBEEK, 2003) (AHMAD; DEY, 2007) (HUNT; JORGENSEN, 2011) (JAIN, 2010)(AMORIM; MAKARENKOV, 2016).

Índice critério de agrupamento cúbico

A fim de vencer o desafio da definição do número de grupos, parâmetro de entrada para o algoritmo *k-means*, é usual analisar os valores de critérios numéricos emitidos para diferentes números de grupos. Um destes critérios é chamado de índice CCC, já citado neste capítulo. Este índice, proposto por Sarle (1993), penaliza o agrupamento que contiver variáveis irrelevantes e valores atípicos e indica um bom agrupamento quanto maior for o seu valor (MILLIGAN; COOPER, 1985) (NBCLUST..., 2014). Para valores negativos, o índice indica que há presença de valores atípicos nos dados, e por isso, deve-se eliminá-los antes de realizar o agrupamento novamente. Já os valores positivos, indicam que pode haver presença de grupos nos dados (LIU; ONG, 2008).

O índice CCC é calculado como segue, Equação 2.

$$CCC = \ln \left[\frac{1 - E(R^2)}{1 - R^2} \right] \frac{\sqrt{\frac{np}{2}}}{[0.001 + E(R^2)]^{1.2}} \quad (2)$$

Em que R^2 representa proporção de variância nos grupos, n o número de registros e p o número de variáveis selecionadas e, $E(R^2)$ representa a esperança de R^2 . Os cálculos detalhados para obtenção destes dois valores são mostrados nas Equações 3 e 4.

$$R^2 = 1 - \frac{\text{traço}(X^T X - \bar{X}^T Z^T Z \bar{X})}{\text{traço}(X^T X)} \quad (3)$$

A variância nos grupos é calculada através de operações matemáticas com as matrizes X e Z , que respetivamente representam: matriz de dados (n por k) e matriz indicadora de grupos (n por k) com o elemento $z_{ij} = 1$ se a i -ésima observação pertencer ao j -ésimo grupo e $z_{ij} = 0$, caso contrário.

$$E(R^2) = 1 - \left[\frac{\sum_{j=1}^{p^*} \frac{1}{n+u_j} + \sum_{j=p^*+1}^p \frac{u_j^2}{n+u_j}}{\sum_{j=1}^p u_j^2} \right] \left[\frac{(n-q)^2}{n} \right] \left[1 + \frac{4}{n} \right] \quad (4)$$

A esperança da variação de grupos é calculada considerando que dados distribuídos uniformemente estão em um *hiperbox* de p -dimensões e assumindo que os agrupamentos são aproximações de um paralelepípedo. Dessa forma, tem-se que as variáveis presentes na Equação 4 e, ainda não explicadas, significam:

- $u_j = \frac{s_j}{c}$,
- s_j é a raiz quadrada do j -ésimo autovalor de $\frac{X^T X}{(n-1)}$,
- $c = \left(\frac{v^*}{q} \right)^{\frac{1}{p^*}}$,
- $v^* = \prod_{j=1}^{p^*} s_j$,
- p^* é escolhido para ser o maior inteiro menor do que k sendo que u_{p^*} não pode ser menor que um.

Mais detalhas dos cálculos que envolvem o índice podem ser vistos em Sarle (1993).

Neste capítulo foram expostas as principais definições, nomenclaturas e terminologias sobre os temas que alimentam esta pesquisa. Espera-se que o entendimento dos próximos tópicos seja facilitado com o conteúdo abordado até então.

Seleção de atributos

Este capítulo fornece uma apresentação das principais abordagens de seleção de atributos presentes na literatura. Além disso, considerando esta atividade como um problema de busca, este capítulo ainda traz uma descrição dos quatro passos essenciais que fundamentam os algoritmos de seleção, independentemente da abordagem. O enfoque deste levamento teórico está pautado no aprendizado não supervisionado, visto que esta pesquisa trabalha com algoritmos de agrupamento de dados.

3.1 Seleção de atributos como problema de busca

A seleção de atributos se faz necessária por inúmeras razões no processo de descoberta do conhecimento, no entanto, segundo Saeys, Inza e Larranaga (2007), são três os principais objetivos desta tarefa: (a) evitar o sobreajuste, *overfitting*, e melhorar o desempenho dos algoritmos de aprendizado; (b) fornecer métodos mais rápidos e de menor custo, e (c) obter uma visão mais profunda dos processos subjacentes que geraram os dados.

Figura 16 – Todos os possíveis subconjuntos de atributos para $p_{co}=4$.

Fonte: Langley (1994)

Para alcançar os objetivos supracitados, a seleção de atributos pode ser encarada como um problema de busca, já que se procura por um subconjunto ótimo de variáveis dentre todas as possibilidades de subconjuntos existentes (LANGLEY, 1994). A fim de ilustrar todas estas as possibilidades quando o número de atributos originais é 4, a Figura 16 esboça o espaço de soluções para este caso.

Como pode-se observar, o espaço de soluções para $p_{co}=4$, sendo p_{co} o número de variáveis, conta com dezesseis subconjuntos de atributos, incluindo o subconjunto vazio, sendo que os círculos em preto representam os atributos selecionados e os círculos em branco, os atributos não selecionados.

Apesar de ser um problema de busca em ambos os aprendizados (supervisionado ou não), em aprendizado não supervisionado o problema é mais complexo. Neste caso, além de não se conhecer o subconjunto ótimo de atributos, também não se conhece o número ótimo de grupos, o que aumenta o espaço de soluções, que é representado por $2^{p_{co}}[(k_{max} - k_{min}) + 1] - 1$, k_{max} e k_{min} o número máximo e mínimo de grupos. Dy e Brodley (2004) trazem uma definição para a seleção de atributos em aprendizado não supervisionado.

"O objetivo da seleção de atributos em aprendizado não supervisionado é encontrar o menor subconjunto de atributos que melhor descobre grupos "naturalmente interessantes" nos dados de acordo com o critério escolhido." Traduzido pela autora.

Por sua complexidade, as pesquisas de seleção de atributos em aprendizado não supervisionado são recentes e ainda escassas na literatura (COVOES, 2010), (SAEYS; INZA; LARRANAGA, 2007), (HRUSCHKA; COVOES, 2005), (MORITA et al., 2003) e (DASH; LIU, 1999).

Em altas dimensões, é impraticável realizar uma busca em todo o espaço de solução a fim de encontrar a solução ótima devido ao tempo computacional (KOHAVI; JOHN, 1997) (SIEDLECKI; SKLANSKY, 1989). Por isso, compreender a natureza do processo de busca é fundamental para a seleção de um subconjunto ótimo global ou ótimo local.

Sendo assim, o processo referenciado está estruturado em quatro passos, que independem da abordagem e do algoritmo de aprendizado utilizado, a saber: (i) geração de sucessores; (ii) estratégia da busca; (iii) avaliação e (iv) critério de parada (LANGLEY, 1994). Estes passos serão detalhados nos próximos tópicos.

3.2 Geração de sucessores

O primeiro passo a ser desenvolvido envolve a escolha do ponto inicial (ou pontos iniciais), também chamado de ponto de partida (ou pontos de partida). Esta escolha gera uma interferência direta na direção que o processo de busca tomará e na geração dos pontos sucessores. De forma geral, quatro abordagens são usualmente utilizadas para a

definição destes componentes, como segue (MOLINA; BELANCHE; NEBOT, 2002) (BLUM; LANGLEY, 1997):

- ❑ **Busca para frente:** o conjunto vazio, sem atributos, representa o ponto inicial da busca nesta abordagem, que também é conhecida como seleção *forward*. A cada iteração, a próxima solução é gerada pela adição de um atributo no subconjunto analisado. O processo continua até que o critério de parada seja atingido. Frequentemente, elege-se uma quantidade de atributos para interromper o processo ou analisa-se o ganho obtido nas últimas iterações. O processo também pode ser cessado quando a medida de avaliação atinge um valor predefinido. A complexidade desta abordagem, introduzida por Whitney (1971), é da ordem de $O(q)$, onde q representa o número de iterações (gerações).
- ❑ **Busca para trás:** esta abordagem, que também é conhecida como seleção *backward*, introduzida por Marill e Green (1963), segue a mesma diretriz da anterior, no entanto, inicia-se a busca com o conjunto original, com todos os atributos. A cada iteração, a eliminação de um atributo é realizada a fim de encontrar a solução ótima. O processo de busca é interrompido utilizando as mesmas opções de critérios de parada do item anterior e a complexidade desta abordagem também é da mesma ordem que na busca para frente.
- ❑ **Seleção bidirecional:** introduzida por Stearns (1976) e por Pudil, Novovicova e Kittler (1994), e também conhecida por seleção *outward*, esta abordagem permite que o ponto inicial seja qualquer ponto do espaço de soluções. As buscas para frente e para trás são combinadas, tornando possível a adição ou a eliminação de um atributo em cada iteração. Considerando que u representa o número de iterações consecutivas de adição e w o número de iterações consecutivas de eliminação, e que $u \neq w$, tem-se que esta abordagem apresenta uma ordem de $O(q^{u+w+1})$. Um possível critério de parada consiste em analisar se o variável que foi adicionada é a mesma que foi eliminada. Segundo Lee (2005), quando não há indícios sobre o número de atributos do subconjunto ótimo há uma grande chance de o mesmo estar no centro do espaço de busca. Sendo assim, esta abordagem é a mais indicada.
- ❑ **Geração aleatória:** esta última abordagem é semelhante a anterior no que diz respeito ao ponto inicial, visto que este pode estar em qualquer local do espaço de soluções. A diferença reside na geração do sucessor. Neste caso a geração do novo subconjunto é aleatória, não limitando a direção da busca. Uma das vantagens desta abordagem consiste em evitar (sem garantias de obter êxito) que a busca fique presa em mínimos ou máximos locais, já que não há um direcionamento predefinido (BLUM; LANGLEY, 1997).

Até este passo foi definido como a direção da busca é efetuada a partir do ponto inicial (ou pontos iniciais). É possível a busca começar pelos extremos, conjunto vazio ou conjunto original, e a cada iteração ir adicionando ou eliminando atributos, respectivamente. Também é possível começar de qualquer ponto do espaço de soluções, e a cada iteração, adicionar ou eliminar um atributo, ou ainda, eleger o sucessor de forma aleatória.

O próximo passo consiste em definir como a busca será realizada, como o espaço de soluções será percorrido. Para este passo, dá-se o nome de estratégia da busca.

3.3 Estratégia de busca

Também chamado de organização da busca, este passo é responsável por definir de que maneira o espaço de soluções será percorrido a fim de encontrar uma solução ótima global ou local.

É possível percorre-lo de tal forma que se garante a solução ótima global. Também é possível percorre-lo de forma parcial, seguindo uma sequência ou não, neste caso não garantindo a solução ótima, mas ganhando em velocidade no processamento e até a viabilidade do procedimento.

A fim de detalhar e pontuar cada estratégia brevemente introduzida acima Molina, Belanche e Nebot (2002) especificam cada uma das três, além dos respectivos algoritmos:

□ **Busca exponencial:** esta estratégia consiste em realizar uma busca que garante como resultado a solução ótima através de um algoritmo cujo custo é da ordem de $O(2^q)$, em que q representa o número de iterações (gerações). Uma das abordagens é a busca exaustiva, também nomeada de busca completa ou busca por força bruta, a qual percorre todo o o espaço de busca. Por necessariamente passar por todos os pontos, o tempo computacional é o fator limitante desta opção, a tornando impraticável na maioria dos casos. Uma outra abordagem consiste em percorrer o espaço de soluções de forma parcial, através do método ramificação e corte, em inglês *branch and bound* (NARENDRA; FUKUNAGA, 1977). Nesta opção é necessário que o critério de avaliação dos subconjuntos obedeça a propriedade de monotocidade. Esta propriedade diz que, tendo dois subconjuntos quaisquer S_1 e S_2 e sendo $S_1 \subseteq S_2$, então a medida de avaliação de S_1 será sempre maior ou igual a medida de avaliação de S_2 . Uma última abordagem seria uma busca A^* com uma heurística admissível (MOLINA; BELANCHE; NEBOT, 2002).

□ **Busca sequencial:** esta estratégia consiste em realizar apenas uma mudança específica no subconjunto atual. Normalmente, técnicas gulosas, em inglês *greedy methods*, são utilizadas para realizar este tipo de busca. Entre elas pode-se citar: busca para frente, busca para trás e seleção bidirecional. Também há técnicas mais sofisticadas como a chamada "melhor primeiro", em inglês *best-first*, cujo custo

computacional é da ordem de $O(q^2)$, em que q representa o número de iterações (gerações) (LIU; MOTODA, 2007).

- **Busca aleatória:** esta estratégia compreende percorrer o espaço de soluções de forma parcial e aleatória, obtendo como vantagem maior abrangência em relação a localização dos pontos acessados. Dessa forma, não se fica aprisionado em ótimos locais. Os algoritmos genéticos são os métodos mais utilizados neste caso e serão detalhados no Capítulo 4.

A Tabela 2 mostra todas as possibilidades de combinação dos três tipos de estratégia da busca com as quatro abordagens apresentadas do Tópico Geração do Sucessor, indicando qual delas são possíveis através da palavra "sim" e através da palavra "não" quando a combinação não é possível.

Tabela 2 – Combinação das estratégias e direções de busca

Direção da Busca	Estratégia da Busca		
	Exponencial	Sequencial	Aleatória
Busca para frente	sim	sim	não
Busca para trás	sim	sim	não
Seleção bidirecional	sim	sim	não
Geração aleatória	não	sim	sim

Fonte: Liu e Motoda (1998) com adaptações na nomenclatura.

Como é possível observar, a direção e a estratégia da busca estão diretamente relacionadas. Observa-se que não é possível optar pela geração aleatória do sucessor em uma busca exponencial, assim como as buscas para frente e para trás e a seleção bidirecional não se relacionam com a estratégia de busca aleatória.

A Figura 17 ainda traz a relação entre a medida de importância do atributo, descritas no Capítulo 3, as opções de geração de sucessores e as estratégias da busca.

Como é possível observar, há três características que definem um algoritmo de seleção de atributos. Escolher a medida de importância do atributo, estratégia da busca e de que forma será gerado o sucesso é o alicerce para a obtenção de um bom subconjunto de atributos.

3.4 Avaliação

A terceira etapa de um processo de busca consiste em definir a forma de avaliação do subconjunto selecionado ou das variáveis. Durante muitas décadas, inúmeros métodos de avaliação foram propostos, no entanto, é possível dividi-los em três categorias, (i) Filtro,

em inglês *filter*, (ii) Envelope, em inglês *wrapper* e, (iii) Embutido, em inglês *embedded*. Neste capítulo serão abordadas as duas primeiras.

Figura 17 – Caracterização de um algoritmo de seleção de atributos.

Fonte: Molina, Belanche e Nebot (2002). Adaptado pela autora.

A grande diferença entre estas duas abordagens reside no modo como o processo de seleção do subconjunto e o algoritmo de aprendizado estão relacionados. Enquanto na primeira, filtro, os atributos são avaliados antes do algoritmo de aprendizado ser aplicado, na segunda, envelope, o desempenho do subconjunto de atributos é mensurado após o algoritmo de aprendizado se alimentar por tal subconjunto. Nas secções seguintes estas abordagens são descritas com mais detalhes.

Método filtro

Os métodos filtro utilizam propriedades intrínsecas dos dados, selecionando os atributos sem que o algoritmo de aprendizado seja selecionado. Por esta razão, torna-se um desafio para esta abordagem definir o que são atributos relevantes e redundantes. A teoria sobre as possíveis medidas de relevância e redundância de atributos foram apresentada no Capítulo 2, Terminologia e Conceitos. A Figura 18 esboça um diagrama da relação entre a seleção dos atributos e o algoritmo de agrupamento de dados.

Figura 18 – Abordagem filtro para seleção de atributos em agrupamento de dados.

Fonte: Liu e Motoda (1998) Traduzido pela autora.

Como é possível observar, a busca seleciona um subconjunto de atributos dentre todos os atributos disponíveis, realiza a avaliação deste, e verifica se o valor do critério de avaliação satisfaz a condição predefinida anteriormente. Caso não, um outro subconjunto é selecionado e o processo se repete até que o valor do critério de avaliação seja satisfeito. Apenas após estes passos serem executados que o algoritmo de agrupamento é alimentado pelo subconjunto selecionado.

Muitos autores fizeram uso desta abordagem de avaliação para realizar a redução da dimensionalidade em bases de dados, como ocorre no trabalho de Dash et al. (2002), onde se considerou a entropia entre dois pontos como medida de avaliação dos atributos. Outras técnicas de redução da dimensionalidade, como a técnica de transformação de atributos Análise de Componentes Principais (ACP), em inglês, *Principal Components Analysis* (PCA), também pode ser considerada uma abordagem filtro, já que realiza-se a redução antes de alimentar o algoritmo de aprendizado com as novas variáveis.

Tabela 3 – Vantagens e desvantagens da abordagem filtro e métodos mais utilizados

Vantagens	Desvantagens	Métodos
Univariado		
Rápido	Ignora as dependências dos atributos	χ^2
Escalável	Ignora a interação com o classificador	Distância euclidiana
Independente do classificador		<i>i</i> -teste
		Ganho de informação
		Taxa de ganho
Multivariado		
Modelos considera a dependência	Mais lento do que técnicas univariadas	Seleção de atributos baseado em correlação
Independente do classificador	Menos escalável do que técnicas univariadas	Markov
		<i>Blanket Filter</i>
Complexidade computacional melhor que o método envelope	Ignora (não em todos os casos) a interação com o classificador	Seleção de atributos baseado em correlação

Fonte: Saeys, Inza e Larranaga (2007). Adaptado pela autora.

Para finalizar, a Tabela 3 traz as vantagens e desvantagens desta abordagem, além dos

métodos mais utilizados, separando-os entre métodos univariados e multivariados, onde o primeiro avalia apenas cada atributo isoladamente, não considerando a possível relação existente entre eles, e o segundo, faz esta consideração, e por tanto, elimina também atributos redundantes além dos irrelevantes.

Método envelope

Esta abordagem de avaliação consiste em solicitar o algoritmo de agrupamento de dados em todas as iterações para que este emita o critério de avaliação do subconjunto em questão. A Figura 19 traz um diagrama da relação entre a seleção do subconjunto e o algoritmo de agrupamento de dados.

Figura 19 – Método envelope para seleção de atributos em agrupamento de dados.

Fonte: Liu e Motoda (1998). Traduzido pela autora.

Como é possível observar, a busca seleciona um subconjunto de atributos dentre todos os atributos disponíveis e alimenta o algoritmo de agrupamento com este subconjunto. A partir disso, o algoritmo emite um índice de avaliação de agrupamento, que será o critério de avaliação do subconjunto selecionado. Caso este critério não satisfaça um valor predefinido, a busca retorna ao ponto da seleção de um novo subconjunto e o processo continua até o critério ser satisfatório. Este método é mais custoso computacionalmente do que o método filtro, já que o algoritmo de aprendizado é solicitado para realizar cada avaliação dos subconjuntos acessados. Além disso, Dy e Brodley (2000), ao estudar problemas envolvendo a criação de métodos envelope gerais, os quais pudessem ser aplicados em qualquer seleção de atributos, agrupamento e seleção de critério, descobriram dois pontos de atenção nesta abordagem.

O primeiro, refere-se ao número de grupos no processo de avaliação do subconjunto. Sabe-se que o número de grupos e o subconjunto ótimo ou subótimo de variáveis estão relacionados. Portanto, os autores dizem que não é uma boa ideia utilizar o mesmo número de grupos para avaliar um subconjunto, sugerindo assim que é necessário percorrer um intervalo para o número de grupos em cada avaliação, o que torna a abordagem ainda mais custosa.

O segundo ponto consiste na descoberta de que alguns critérios de seleção são enviesados com relação a dimensionalidade. Os autores examinaram dois critérios, a máxima

verossimilhança e a dispersão de separabilidade. Eles mostraram que o último aumenta com o aumento da dimensionalidade e o primeiro, ao contrário, prefere dimensões baixas. Para anular o efeito do viés, os autores propuseram uma normalização para qualquer critério de avaliação, através de uma projeção cruzada, ver em Liu e Motoda (2007).

Para finalizar, a Tabela 4 traz as vantagens e desvantagens desta abordagem, além dos métodos mais utilizados, dividindo-os em determinísticos e não determinísticos.

Tabela 4 – Vantagens e desvantagens da abordagem *wrapper* e métodos mais utilizados

Vantagens	Desvantagens	Métodos
Determinístico		
Simples	Risco de sobreajuste dos atributos	Busca para frente
Interação com o classificador	Mais propenso do que algoritmo aleatório	Busca para trás
Modelo de atributos dependentes		<i>Plus q take-away r</i>
Menos intensivo computacionalmente do que algoritmo aleatório		<i>Beam search</i>
Não determinístico		
Menos propensos para ótimos locais	Intensivo computacionalmente	Recozimento simulado
Interação com o classificador	Seleção dependente do classificador	Subidas Aleatórias
Método de dependência de atributo	Maior risco de sobreajuste do que o determinístico	Algoritmo genético
		Estimação do algoritmo de distribuição

Fonte: Saeys, Inza e Larranaga (2007). Adaptado pela autora.

3.5 Critério de parada

Como último passo, a busca necessita de uma diretriz, nomeada de critério de parada, para finalizar o processo. Este critério está intimamente relacionado com o tipo de estratégia adotada e por consequência, com a direção da busca. Em uma busca exponencial, por exemplo, o critério de parada só é atingido quando todo o espaço de soluções é percorrido. Em uma busca sequencial, pode-se eleger como critério uma quantidade y de atributos, um ganho mínimo de desempenho nas últimas z iterações ou até mesmo um valor predefinido da medida de avaliação. Em uma busca aleatória, os critérios de avaliação elencados anteriormente também são válidos. Podemos considerar também o número de iterações, a distância relativa entre duas iterações e o tempo computacional.

Com a descrição deste último passo, finaliza-se o detalhamento das etapas de um processo de busca. Como já mencionado no início deste capítulo, a seleção de atributos pode ser considerada como um problema de busca. E, sendo assim, este trabalho elegerá um ponto (ou pontos) de partida, direção e estratégia da busca, medida de avaliação e critério de parada a fim de encontrar um subconjunto de atributos ótimo, não necessariamente global, dentre todos os subconjuntos do espaço de soluções.

Como direção e estratégia da busca esta pesquisa faz uso da geração e busca aleatória, respectivamente. É utilizado o algoritmo genético, um algoritmo evolutivo, como método nestes dois passos. Mais especificamente, utiliza-se duas variações deste, o AGC e o BKGA, que serão detalhados no Capítulo 4, assim como a teoria que fundamenta o próprio algoritmo genético em sua forma mais tradicional.

Algoritmos evolutivos

Este capítulo tem como objetivo expor brevemente a história e princípios dos algoritmos evolutivos. De forma mais detalhada, é descrita as principais características e elementos dos algoritmos genéticos, ainda em sua abordagem mais tradicional. Para finalizar o conteúdo teórico deste tema, é exposto as duas variações da abordagem tradicional, Algoritmo Genético Construtivo e Algoritmo Genético Enviesado com Chave Aleatória a fim de detalhar as principais diferenças que contribuem para a melhoria de desempenho destes algoritmos.

4.1 Evolução natural e a computação evolutiva

Charles Darwin, em 1859, após observar inúmeros indivíduos na natureza, concluiu que muitas formas de vida são suscetíveis a adaptação ao meio ambiente em que elas vivem para sobreviver.

Percebeu também que tal adaptação é relativa, e não absoluta, sendo correto afirmar que um pinguim está apto a ambientes frios, como a Antártica, mas não estaria a ambientes de calor intenso, como em uma região tropical.

Além do meio, a habilidade que este indivíduo apresenta para executar tarefas em seu ambiente e de se reproduzir são outras formas que traduzem sua adaptação (FURTADO, 1998).

Após observar a natureza e o processo pelo qual esta evolui, Darwin formalizou o conceito sobre seleção natural através da seguinte oração *os indivíduos mais adaptados ao meio apresentam maior possibilidade de sobrevivência* (MITCHELL, 1998) e concluiu que o processo evolutivo ocorre através de lentas transformações genéticas, de geração em geração (RIBEIRO, 2001).

Apesar de identificar que a adaptação é um dos componentes do processo evolutivo, o pesquisador destaca que apenas seleção natural pela aptidão não caracteriza este processo. Para tal, é necessário que se satisfaça quatro condições, sendo que a primeira diz que um indivíduo deve ser capaz de se reproduzir, a segunda relata que na natureza deve

haver uma população destes indivíduos, a terceira condição afirma que estes indivíduos devem ser variados dentro da população, e por fim, a quarta diz que as habilidades de sobrevivência destes indivíduos na natureza devem ser diferentes e estão relacionadas com a variedade da população.

Na natureza, a variedade da população é expressa pelas diferenças entre os cromossomos. Cromossomo nada mais é do que uma longa sequência de DNA, e este por sua vez, é definido como um composto orgânico que armazena instruções genéticas através da combinação dos elementos: adenina (A), citosina (C), guanina (G) e timina (T), como traz a Figura 20.

Figura 20 – Esboço da estrutura de um cromossomo, gene e DNA.

A informação genética está presente em estruturas chamadas genes, partes em verde e vermelho da Figura 20. O restante da estrutura, partes em preto, tem importância estrutural ou exerce papel regulador no uso da informação genética. Cada gene ainda pode assumir um valor, chamado de alelo (GONCALVES; RESENDE, 2011).

Estas diferenças nos cromossomos resultam em variabilidade tanto na estrutura quanto no comportamento dos indivíduos. Consequentemente, obtêm-se diferentes taxas de sobrevivência e reprodução.

Ao longo das gerações, a variedade da população é garantida pelo nascimento de novos indivíduos que apresentam carga genética diferente da carga genética dos pais. Esta variedade é resultado da combinação das informações genéticas de cada gerador ou por uma alteração gênica pontual, a mutação.

De tempos em tempos, os indivíduos menos aptos e/ou mais antigos são eliminados, renovando assim a população. A mutação contribui demasiadamente para a inovação genética entre os indivíduos. Quando estas mudanças nas gerações são visíveis e mensuráveis é possível dizer que a população evoluiu (RIBEIRO, 2001).

Nas décadas de 1950 e 1960, pesquisadores se inspiraram na estrutura lógica do processo de evolução e começaram a estudá-lo a fim de utilizar o mesmo raciocínio na resolução problemas de otimização combinatória.

Muitos trabalhos surgiram desde então, recebendo nomes como Estratégias Evolutivas (RECHENBERG, 1965), Programação Evolutiva (FOGEL, 1991) (FOGEL; OWENS; WALSH, 1966) e Algoritmos Genéticos (HOLLAND, 1975) (GOLDBERG; DEB; KORB, 1989).

O desenvolvimento destes trabalhos ocorreu de forma independente e muitas vezes de maneira paralela. A consequência desta situação se mostrou em resultados semelhantes e até redundantes (FOGEL; OWENS; WALSH, 1966), além de procedimentos similares entre as abordagens propostas na época. Posteriormente, surgiram trabalhos que exploravam técnicas mais especializadas, como é o caso da Programação Genética (KOZA, 1992).

Atualmente, nomeia-se de Algoritmos Evolutivos (AE) o conjunto teórico desenvolvido ao longo destas cinco décadas de estudo. Apesar das grandes simplificações que os procedimentos computacionais apresentam ao reproduzir o processo de evolução natural, boas soluções para problemas de otimização são encontradas.

Muitas áreas do conhecimento são beneficiadas por esta abordagem, como a Economia, Imunologia, Sistemas Sociais e Inteligência Artificial, entre outras.

4.2 Algoritmo genético

Criado por Holland (1975), e posteriormente aperfeiçoado pelos seus alunos (JONG, 1975) (GREFENSTETTE, 1986) (GREFENSTETTE, 1987) (GOLDBERG; DEB; KORB, 1989), o algoritmo genético teve como principal motivação utilizar os conceitos do processo de evolução natural para criar sistemas artificiais que seriam capazes de resolver uma gama de problemas de otimização, não apenas problemas particulares.

A generalização dos conceitos foi a contribuição que Holland deixou, uma vez que as abordagens anteriores, como a Programação Evolutiva e as Estratégias de Evolução, utilizavam como inspiração o processo evolutivo para a resolução de problemas específicos, ou seja, atingiram a resolução de problemas pontuais.

Em termos práticos e fazendo uma analogia com o que se encontra na natureza, o funcionamento do algoritmo genético ocorre com a definição de uma população inicial de indivíduos, também chamados de estruturas. Cada indivíduo representa uma solução no espaço de soluções do problema estudado.

Estes indivíduos são formados por uma sequência de símbolos, chamadas de *string*, de tamanho b , em que $b > 0$, de um determinado alfabeto A , de forma análoga à estrutura de um cromossomo. O espaço de estruturas é representado por A^b , sendo que seu tamanho é calculado por $|A|^b$, em que $|A|$ representa o número de símbolos em A .

A população pode ser iniciada de forma aleatória ou seguindo algum critério que expresse a estrutura do problema de forma mais eficaz.

O processo de evolução começa com a execução do procedimento de seleção dos indivíduos para a reprodução. Pode haver a combinação de material genético para a composição de novos indivíduos, procedimento este chamado de recombinação, ou em inglês, *crosso-*

ver, ou pode ocorrer a clonagem de um indivíduo de uma geração para outra, o que consiste em realizar uma cópia idêntica da estrutura.

Outro procedimento que as estruturas estão suscetíveis é a mutação, que introduz uma inovação na população, alterando a informação genética.

O processo termina quando atinge um critério de parada, que pode ser um número de gerações previamente estipulada, um valor de adaptação satisfatório, ou então, um nível de estagnação atingido pelo processo.

Na Figura 21 podemos observar os parâmetros e operações que compõe a essência básica de qualquer algoritmo genético, sendo este tradicional ou uma variação.

Figura 21 – Parâmetros e operações que compõe a estrutura lógica de um algoritmo genético.

O tamanho da população, a taxa com que ocorre a recombinação e a mutação, além do critério de parada, são parâmetro do algoritmo genético. Já a função de adaptação, seleção, reprodução, métodos de recombinação e mutação compõem o conjunto de operações destes algoritmos.

As variações do algoritmo genético tradicional surgem quando os componentes da Figura 21 apresentam um diferencial que gera ganho significativo nos resultados.

População inicial e codificação

O número de indivíduos da população inicial, n , é um parâmetro importante no algoritmo genético, uma vez que quanto mais indivíduos se tem em uma população, maior o espaço percorrido pela busca. No entanto, este número não pode ser muito grande, visto que o algoritmo desacelera, resultado do lento processamento.

Alguns algoritmos mantêm um número fixo de indivíduos na população, ao passo em que outros este número pode variar. Esta decisão é tomada de modo empírico e é considerada um parâmetro do algoritmo.

Para o problema de seleção de atributos, Scrucca (2010) sugere que o número de indivíduos na população inicial deve ser $n = \min\{2^{p_{co}}, 50\}$, sendo p_{co} o número de variáveis do conjunto original.

Já a maneira como os indivíduos são codificados depende diretamente da natureza do problema abordado, no entanto, há muitas formas de codificação possíveis.

Uma maneira usual de codificação consiste em utilizar o conjunto binário $\{0,1\}$ como alfabeto A (RIBEIRO, 2001) (FURTADO, 1998).

Além do alfabeto binário, é possível também inserir o símbolo $\#$ no corpo da *string*, significando que no lugar em que este símbolo está posicionado é possível substituir qualquer outro caractere do alfabeto A . Caso a *string* apresente o símbolo $\#$, então está já não é mais chamada de estrutura e sim de esquema.

Para exemplificar esse novo conceito, considera-se a estrutura $s = (0 \ 0 \ 1 \ 0 \ 1 \ 1)$ e o esquema $H = (0 \ 0 \ 1 \ 0 \ 1 \ \#)$. Diz-se, portanto, que a estrutura s é uma representante do esquema H , ou que s é uma instância do esquema H , isto porque s pode ser derivada de H substituindo o símbolo $\#$ de H por 1.

Algumas propriedades sobre o conceito de esquemas são destacadas, como segue:

- A quantidade de estruturas que podem ser derivadas de H é representado por $M(H)$. No caso do exemplo dado, $M(0 \ 0 \ 1 \ 0 \ 1 \ \#) = 2$
- Diz-se que a ordem de H representa o número de símbolos fixos do esquema. No exemplo dado, $o(0 \ 0 \ 1 \ 0 \ 1 \ \#) = 5$
- Diz-se que o comprimento de H , representado por $\varphi(H)$, corresponde a diferença entre o primeiro e o último símbolo fixo em H . No exemplo dado, $\varphi(0 \ 0 \ 1 \ 0 \ 1 \ \#) = 5 - 1 = 4$

Para o problema de seleção de atributos em agrupamento de dados, a representação de um indivíduo pode ser feita através da estrutura $(0 \ 0 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1)$, neste caso, 8 variáveis fazem parte do conjunto original de variáveis, sendo que a terceira, sexta e oitava estão selecionadas na *string* representada pela estrutura binária.

Liu e Ong (2008) abordam o problema de seleção de atributos em agrupamento de dados utilizando o algoritmo genético em sua forma mais tradicional. Em seu trabalho, os autores utilizam a estrutura descrita acima para indicar quais variáveis estão selecionadas naquele indivíduo. No entanto, um último carácter ainda é adicionado para indicar a quantidade de grupos que aquela estrutura trabalhará. Para este último carácter o número pode variar de 2 a 10 grupos.

O funcionamento do algoritmo genético está intimamente relacionado ao conceito de esquemas e a maneira com que estes são propagados pelas gerações. Considera-se os esquemas como blocos construtivos e sua propagação e combinação fundamentam hipóteses sobre o desempenho do algoritmo genético (GOLDBERG; DEB; KORB, 1989). No tópico

"O Funcionamento do Algoritmo Genético" será detalhado com maior profundidade o Teorema dos Esquemas e a Hipótese do Bloco Construtivo.

Função de avaliação

É a função de avaliação que traduz a adaptação do indivíduo em uma população. E, é na codificação de cada indivíduo e na função de avaliação que reside a relação direta do AG com o problema em si.

Em problemas de seleção de atributos em agrupamento de dados é usual eleger índices que medem a qualidade do agrupamento como fator de adaptação do indivíduo. Pode-se citar como exemplo o trabalho de Lleti et al. (2004), onde autores elencaram a largura da silhueta como função de avaliação. No trabalho de Morita et al. (2003) os autores também utilizaram um índice de qualidade, chamado de *DB Index*. Já no trabalho de Liu e Ong (2008), o índice CCC, já detalhado no capítulo 2, foi eleito o critério que representaria a função de avaliação.

Seleção de indivíduos

Os indivíduos mais aptos são selecionados para se reproduzir e contribuir com a composição da próxima geração. Há inúmeras maneiras para realizar a seleção destes indivíduos seguindo esta premissa, privilegiar os mais aptos.

Na abordagem tradicional do algoritmo genético, a seleção por roleta é usualmente sugerida (HOLLAND, 1975). Goldberg, em seu livro *Genetic Algorithms - in Search, Optimization and Machine Learning* (GOLDBERG, 1989) faz menção a este procedimento e explica que seu funcionamento consiste em construir uma roleta onde cada indivíduo da população atual tem um espaço proporcional a sua adaptação, resultando assim em maiores chances de ser selecionado para reprodutor àqueles que são mais adaptados.

Figura 22 – Seleção por roleta como procedimento de seleção de indivíduos para reprodução e consequente construção da próxima geração.

Fonte: Goldberg (1989). Adaptado pela autora.

A Figura 22 traz um exemplo da seleção por roleta para uma população inicial contendo 4 indivíduos.

Como é possível observar na Figura 22 o indivíduo número 2 apresenta 49,2% (576/1170) de chances de ser selecionado como reprodutor. Já o indivíduo número 3, apresenta apenas 5,5% de chances de ser escolhido para contribuir com a próxima geração.

A seleção por roleta é apenas um dos métodos de seleção que o algoritmo genético pode adotar. Há outros como seleção por torneio (GOLDBERG; DEB, 1991), seleção com ordenação (BAKER, 1985) e seleção de Boltzmann (GOLDBERG, 1990).

Recombinação e mutação

Como já mencionado neste tópico, a nova geração pode ser criada combinando diversas opções: simples cópia idêntica de indivíduos da geração atual para a próxima geração; recombinação de dois indivíduos pais, gerando dois filhos para a próxima geração ou ainda pela mutação, procedimento que altera aleatoriamente um indivíduo da população atual e adiciona-o na próxima geração.

Operador de recombinação

O operador de recombinação tem como objetivo introduzir novas estruturas e eliminar estruturas/esquemas com baixa adaptação na geração seguinte. Por isso, esta operação determina como será a combinação do material genético dos indivíduos pais nos filhos. A maneira com que esta combinação é realizada influencia diretamente na composição dos indivíduos da próxima geração, e por consequência, seu valor de adaptação.

Realizar a recombinação significa determinar qual(is) intervalo(s) das *strings* dos indivíduos pais irão compor as *strings* dos indivíduos filhos. Este(s) intervalo(s) é(são) definido(s) através de um ou mais pontos de ruptura, como mostra a Figura 23.

PAI	1 0 0 0 1 1 1 0 0 0 1 0 1
MÃE	0 0 1 0 1 1 0 0 1 0 1 0 0
RUPTURA	*
FILHO 1	1 0 0 0 1 1 0 0 1 0 1 0 0
FILHO 2	0 0 1 0 1 1 1 0 0 0 1 0 1

(a)

PAI	1 0 0 0 1 1 1 0 0 0 1 0 1
MÃE	0 0 1 0 1 1 0 0 1 0 1 0 0
RUPTURA	* *
FILHO 1	1 0 0 0 1 1 0 0 0 0 1 0 1
FILHO 2	0 0 1 0 1 1 1 0 1 0 1 0 0

(b)

Figura 23 – Operador recombinação de um e dois pontos.

Fonte: Furtado (1998).

Como é possível observar, a Figura 23 (a) exemplifica uma operação de recombinação de um ponto. Sendo assim, o primeiro filho receberá o material genético do pai até a

terceira posição da *string* deste. A partir da quarta posição, o material genético virá da mãe. O segundo filho, por sua vez, receberá uma cópia do material genético da mãe até a terceira posição, e a partir da quarta posição o material genético virá do pai.

A Figura 23 (b) traz um exemplo de uma operação de recombinação de dois pontos. Neste caso, o primeiro filho apresenta as informações genéticas do pai nos intervalos de posição um a três e nove a treze, e da mãe no intervalo de posição quatro a oito. Já o segundo filho apresenta informações genéticas do pai no intervalo de posição quatro a oito e da mãe nos intervalos de posição um a três e nove a treze.

O exemplo exposto na Figura 23 é apenas uma das possibilidades de se realizar a recombinação, outras maneiras podem ser vistas em Beasley, Bull e Martin (1993). Esta operação acontece com uma determinada taxa, chama taxa de recombinação (p_c) visto que não é em todas as gerações que se aplica a recombinação. A literatura sugere uma taxa no intervalo de 80% a 90%, no entanto, o valor deste parâmetro é definido pelo usuário e também é influenciado pelo tipo de problema em questão.

Operador de mutação

O operador de mutação introduz variedade na população e evita a homogeneidade ao longo das gerações, através de uma mudança aleatória na *string* de um ou mais indivíduos. Esta alteração também coopera para que o algoritmo não fique preso em máximos ou mínimos locais.

Na Figura 24 é possível observar um exemplo da alteração causada na *string* de um indivíduo quando ocorre a mutação em um ponto.

ESTRUTURA	0 0 1 0 1 1 0 0 1 0 1 0 0
MUTAÇÃO	*
NOVA ESTRUTURA	0 0 1 0 1 1 1 0 1 0 1 0 0

Figura 24 – Operador de mutação em um ponto.

Fonte: Furtado (1998).

A estrutura da Figura 24 sofreu uma alteração no valor da sétima posição, de zero passou para um. É importante ressaltar que procedimentos mais sofisticados de mutação podem ser desenvolvidos, mas sempre estarão atrelados ao tipo de codificação e as características do problema.

É sugerida pela literatura que a taxa de mutação (p_m) esteja entre 10% a 15%. Esta é apenas uma sugestão, não impede o usuário do algoritmo a optar por outro valor.

A Figura 25 tem como objetivo exemplificar a influência dos operadores de seleção, recombinação e mutação no valor de adaptação de quatro indivíduos durante quatro gerações.

População P1	
<i>String</i>	Valor de adaptação
0000011100	0.3
1000011111	0.6
0110101011	0.6
1111111011	0.9
População P2: depois da seleção	
<i>String</i>	Valor de adaptação
1000011111	0.6
0110101011	0.6
1111111011	0.9
1111111011	0.9
População P3: depois do <i>crossover</i>	
<i>String</i>	Valor de adaptação
1000011011	0.5
0110101011	0.6
1111111011	0.9
1111111111	1.0
População P4: depois da mutação	
<i>String</i>	Valor de adaptação
1000011111	0.5
0110111011	0.7
1111111011	0.9
0111111111	0.9

Figura 25 – Influência dos operadores de seleção, recombinação e mutação no algoritmo genético tradicional

Fonte: Goldberg (1989). Adaptado pela autora.

Cada indivíduo é representado por uma *string* de tamanho dez, composta de valores binários, $\{0,1\}$. Em cada geração, a Figura 25 mostra o valor normalizado de adaptação do respectivo indivíduo. Como é possível observar, a medida que as operações (seleção, recombinação e mutação) são realizadas o valor de adaptação médio da população aumenta.

Assim, o algoritmo genético possui como elementos básicos para sua estruturação lógica os parâmetros e operadores apresentados e descritos até este ponto do trabalho.

No entanto, em termos práticos, o algoritmo ainda conta com um teorema (Teorema do Esquema), uma hipótese (Hipótese de Blocos Construtivos) e com a decisão de como os parâmetros serão definidos. Dessa forma, para o seu funcionamento, é necessário que estes itens sejam detalhados com maior ênfase. Este detalhamento é feito no próximo tópico.

Funcionamento do algoritmo genético

O funcionamento do algoritmo genético está baseado no descobrimento, ênfase e recombinação de bons blocos construtivos, que nada mais são que esquemas, o conceito

introduzido no início deste capítulo. Compreende-se que as boas soluções são advindas de segmentos de bons esquemas.

O objetivo, portanto, é manter estes segmentos no decorrer das gerações, combinando-os com outros segmentos advindos de bons blocos construtivos. Acredita-se que assim uma boa solução será encontrada após o critério de parada ser atingido. Esta ideia é o que caracteriza a Hipótese dos Blocos Construtivos.

Identificar a quantidade de possíveis esquemas que uma população pode apresentar é fundamental para estimar como será a evolução desta população em relação ao número de bons blocos construtivos ao longo das gerações.

Dessa forma, considerando-se que o alfabeto que compõe as *strings* de tamanho b seja o alfabeto binário e que cada estrutura da população de tamanho n represente um total de 2^b esquemas, então nesta população estão presentes $n(2^b)$ esquemas diferentes, no caso de todas as estruturas serem diferentes.

A Figura 26 traz dois esquemas e uma estrutura para exemplificar os conceitos já introduzidos em tópico anterior.

A estrutura da Figura 26 representa os dois esquemas, H_1 e H_2 . Dessa forma, quando o algoritmo genético calcula o valor de adaptação de uma estrutura ele está, na verdade, calculando o valor de adaptação médio de um número muito maior de esquemas. Isto se dá pela razão entre o valor de adaptação da estrutura em questão e a soma dos valores de adaptação de todas as estruturas daquele(s) esquema(s) presentes na população no instante t da evolução. Mais detalhes em (GOLDBERG, 1989).

	$\#$: símbolo que significa que qualquer caractere do alfabeto A , neste caso $\{0,1\}$, pode ocupar a posição em que ele está inserido.				
Esquema 1 (H_1)	#	1	1	#	1
Esquema 2 (H_2)	#	1	1	#	#
Instância (s)	0	1	1	0	1
	tamanho da string : quantidade de caracteres. No caso, 5. $M(H)$: quantidade de instâncias que podem ser derivadas de H . No caso, $M(H_1) = 4$ e $M(H_2) = 4$. $o(H)$: número fixo de posições. No caso, $o(H_1) = 2$ e $o(H_2) = 2$. $\Phi(H)$: distância entre a primeira e a última posição de caractere fixo da <i>string</i> . No caso, $\Phi(H_1) = 3$ e $\Phi(H_2) = 1$.				

Figura 26 – Exemplo de esquemas, estruturas e suas respectivas propriedades.

Apesar dos esquemas não serem avaliados diretamente na abordagem tradicional, é possível estimar o número de estruturas que aquele esquema terá nas gerações que seguem. Lembrando que o interesse está nos bons segmentos dos esquemas, e por isso, é importante ressaltar que o tamanho deste segmento afeta diretamente a estimativa do número de estruturas em gerações futuras, visto que há maiores chances de propagação de segmentos pequenos, já que as operações de recombinação e mutação podem destruir com maior facilidade segmentos grandes.

O Teorema dos Esquemas surge diante da intenção de expressar a dinâmica de crescimento e diminuição das estruturas de um determinado esquema H no próximo instante da evolução, $t + 1$. A equação matemática, que exhibe o limite inferior da estimativa, por considerar os efeitos negativos das operações de recombinação e mutação na propagação dos esquemas, pode ser vista na Equação 13.

$$E(M(H, t + 1)) \geq \frac{\mu(\hat{H}, t)}{\mu(t)} M(H, t) \left(1 - p_c \frac{\varphi(H)}{b - 1} \right) [(1 - p_m)^{o(H)}] \quad (5)$$

A Equação 13 fornece o número esperado de estruturas do esquema H no instante da evolução $t + 1$, representado por $E(M(H, t + 1))$. São considerados para o cálculo a adaptação média observada de H no instante t , representada por $\mu(\hat{H}, t)$; a adaptação média de todos os esquemas no instante t , $\mu(t)$; o número de representantes de H , $M(H, t)$; taxa de recombinação, p_c ; tamanho da *string*, b ; taxa de mutação, p_m ; número de símbolos fixos em H , $o(H)$ e a diferença entre o primeiro e último símbolo fixo em H , $\varphi(H)$.

Maiores detalhes sobre os cálculos que levaram a composição da equação acima podem ser encontrados em Furtado (1998).

Através da expressão matemática, é possível constatar que esquemas curtos e de baixa ordem, que apresentam adaptação acima da média, recebem aumento exponencial do número de estruturas com o passar das gerações. Isto ocorre se as estruturas destes esquemas não sofrerem ruptura e o valor de adaptação permanecer acima da média. O aumento do número de estruturas de H , caso as condições forem cumpridas, acontece por um fator $\frac{\mu(\hat{H}, t)}{\mu(t)}$ a cada geração.

Uma outra frente que merece destaque no funcionamento do algoritmo genético é a definição de como os parâmetros serão determinados. Michalewicz e Fogel (2004) traz na Figura 27 a classificação de técnicas que são aplicadas para determinar os parâmetros.

Figura 27 – Classificação de técnicas para ajuste de parâmetros.

Fonte: Michalewicz e Fogel (2004).

Conforme ilustrado na Figura 27 os parâmetros podem ser ajustados antes da exe-

cução do algoritmo, e sendo assim, o ajuste é feito de maneira empírica. Também é possível controlar os parâmetros, durante a execução do algoritmo. Nesta situação, os parâmetros podem ser determinados de três maneiras: (i) de forma determinística, isto é, através de regras predeterminadas; (ii) de modo adaptativo, quando a determinação dos parâmetros está vinculada com o retorno do próprio processo evolutivo; ou, (iii) de modo auto-adaptativo, quando as informações dos parâmetros são codificadas nas *strings* de cada indivíduo, sofrendo reprodução e mutação. Dessa última maneira, os parâmetros se desenvolvem juntamente com a população.

Para finalizar, a Figura 28 traz uma visão geral deste algoritmo.

Figura 28 – Visão geral das operações do algoritmo genético tradicional.

Como é possível visualizar, a estrutura do algoritmo genético consiste em sete passos todos envolvidos pela Teoria da Esquemas e Hipótese dos Blocos Construtivos.

Algoritmo

Detalha-se os procedimentos lógicos do algoritmo genético através do pseudocódigo abaixo. Ressalta-se que este pseudocódigo representa apenas uma das possibilidades de estruturação lógica do algoritmo.

4.3 Algoritmo genético construtivo

Esta variação do algoritmo genético tradicional, proposta por João C. Furtado em sua tese de doutorado (FURTADO, 1998), é baseada na ideia de que boas soluções são obtidas através de bons blocos construtivos e nas ideias de um artigo de Lorena e Lopes (1996), onde os autores trabalharam com combinação horizontal e vertical de retângulos em cortes 2D.

Algoritmo 2 Algoritmo genético tradicional

Definir valores para os parâmetros: p_c , p_m

Inicializar a população

Avaliar a população

enquanto não atingir o critério de parada **faça** Gerar número aleatório a , no intervalo de 0 a 1, inclusive **se** p_c for maior ou igual a a **então**

Realizar a recombinação

fim Gerar número aleatório b , no intervalo de 0 a 1, inclusive **se** p_m for maior ou igual a b **então**

Realizar mutação

fim

Avaliar população

fim

Fonte: Srinivas e Patnaik (1994), com adaptações da autora.

O trabalho de Lorena e Lopes (1996) tinha como objetivo minimizar a perda de material. Para isso, utilizou uma heurística que resultava em uma lista dinâmica, onde soluções parciais eram avaliadas por duas funções que mediam a perda da área local e da área total.

Furtado observou que a avaliação individual dos esquemas poderia ser realizada de forma semelhante a executada no problema de corte 2D, isto é, através de duas funções de avaliação.

O pesquisador também notou que o funcionamento do algoritmo A^* era análogo ao que propunha a heurística utilizada por Lorena e Lopes (1996), no sentido de que as soluções candidatas com os melhores desempenhos eram exploradas primeiro.

E, por fim, Furtado relata que uma outra variação do algoritmo genético, chamada de *Messy Genetic Algorithm*, havia tentado avaliar os esquemas individualmente, no entanto, sem sucesso.

Baseado nestas constatações, o pesquisador propôs o Algoritmo Genético Construtivo, deixando claro em seu trabalho as principais semelhanças e diferenças entre esta variação e a abordagem tradicional.

Maiores detalhes do desenvolvimento do AGC estão presente no trabalho de Furtado (1998).

Nos tópicos que seguem, as principais diferenças entre esta variação e a abordagem tradicional são apontadas.

População inicial

Nesta variação, a forma de representar a solução de um problema é a mesma da abordagem tradicional, isto é, consiste em criar uma *string* composta de elementos do

alfabeto A . Também é possível a criação de esquemas, utilizado o caractere $\#$ para indicar um ponto curinga, que pode assumir qualquer valor do alfabeto A .

A diferença da abordagem tradicional está na composição da população, que deve ser apenas de esquemas. Estes esquemas podem ser gerados de forma aleatória, mas é necessário que eles possuam características estruturais do problema. Este requisito é tão importante que, no caso de necessidade, é justificável realizar uma interferência determinística que garanta uma população de esquemas com diferentes segmentos, cada qual carregando uma característica do problema em questão.

Tendo uma população rica em segmentos informativos, os processos de seleção, reprodução e mutação convergiram a população em uma população de estruturas completas e agregadoras de informações estruturais.

Funções de adaptação

É na forma de avaliação que reside a maior e mais significativa diferença que esta variação propõe frente a abordagem original. Isto porque, até o desenvolvimento desta variação não havia nenhum algoritmo que realizasse, com sucesso, a avaliação de esquemas individualmente.

A avaliação dos esquemas é feita através de duas funções de avaliação, onde a diferença entre elas expressa o objetivo, seja ele de maximização ou minimização, do problema.

Algumas premissas devem ser atendidas e alguns valores de entrada devem ser fornecidos para que este tipo de avaliação seja possível. As premissas são:

- Devem existir duas funções, $g(s_i)$ e $f(s_i)$, ambas definidas em $P\alpha \rightarrow R_+$, onde $P\alpha$ representa o número de indivíduos (estruturas ou esquemas) da população no instante α da evolução e s_i representa um indivíduo/solução/string.
- Para todo o domínio, deve-se ter $f(s_i) \leq g(s_i)$.
- Deve ser definido um limite superior para $g(s_i)$, denominado g_{max} , onde $g_{max} > \max_{s_i \in P_\alpha} g(s_i)$.

Esta dupla forma de avaliar s_i é nomeada por Furtado de avaliação-fg. O desafio do AGC está em definir estas funções, respeitando as premissas supracitadas para o problema a ser trabalhado.

O limite superior (g_{max}) pode ser definido de inúmeras formas, uma delas consiste em selecionar aleatoriamente uma *string* e calcular seu respectivo valor $g(s_i)$ no início do programa, adotando este como o g_{max} . Para garantir que este valor seja o maior valor da função $g(s_i)$, elimina-se todas as novas *strings* (s_{nova}) onde $g_{max} \leq s_{nova}$.

Seleção, recombinação e mutação

A seleção dos indivíduos se inicia ordenando a população de forma crescente de acordo com o valor da função $g(s_i)$ de cada indivíduo.

Define-se uma região de melhores indivíduos, nomeada de região base, através de uma porcentagem, p_{base} . A seleção ocorre escolhendo aleatoriamente um indivíduo (s_{base}) da região base e um outro indivíduo (s_{guia}) da população inteira. A Figura 29 traz um esboço do processo de seleção.

A recombinação do material genético resulta em novos indivíduos (s_{nova}). Objetiva-se que o valor de adaptação da solução s_{nova} seja melhor do que os valores dos pais. Para isso, os bons blocos construtivos devem ser conservados e a cada novo indivíduo novas informações devem ser agregadas.

Os indivíduos da região base, por serem de melhor qualidade, carregam informações estruturais do problema. Já nos indivíduos guia s_{guia} estão presentes informações complementares. Novamente, cada problema exige uma definição específica dos procedimentos de recombinação.

Figura 29 – Seleção dos indivíduos na população para a reprodução.

Após a recombinação, a mutação pode ser aplicada a fim de aumentar a variabilidade das soluções na população assim como na abordagem tradicional. No entanto, a forma de executar este procedimento deve ser analisado para cada tipo de problema.

Avaliação da população

A avaliação da população consiste em definir como o processo evolutivo ocorre a medida que as gerações avançam. Desta forma, a avaliação se inicia com o cálculo do desvio percentual, d_i , em relação a $g(s_i)$, para cada indivíduo, Equação 6.

$$d_i = \frac{g(s_i) - f(s_i)}{g(s_i)} \quad (6)$$

Obtém-se o desvio absoluto de $g(s_i)$ através da multiplicação $d(i)g(s_i)$. Admitindo-se que haja um desvio percentual global em relação a g_{max} , denominado de d , então o desvio absoluto em relação a g_{max} é obtido pelo produto dg_{max} . Assim, compara-se os desvios absolutos dg_{max} e $d(i)g(s_i)$, Inequação 7.

$$d_i g(s_i) + d[g_{max} - g(s_i)] \leq dg_{max} \quad (7)$$

Caso a Inequação 7 seja satisfeita, o individuo s_i é eliminado da população.

Ao observar que boas soluções seriam eliminadas pela Inequação acima em um momento t da evolução, mas em $t + 1$ iriam contribuir para a geração de soluções que atendessem a premissa do desvio, e conseqüentemente, contribuiriam para a resolução do problema, Furtado (1998) incluiu na Inequação 7 um parâmetro de evolução α , resultado na Inequação 8.

$$d_i g(s_i) + \alpha d[g_{max} - g(s_i)] \leq dg_{max} \quad (8)$$

O valor de α , que se inicia em 0, é sempre positivo e vai crescendo gradativamente à medida que as gerações avançam. A Inequação 9 expressa a comparação realizada a fim de decidir se a solução s_i fica ou é eliminada da população, isolando α .

$$\alpha \geq \frac{dg_{max} - d_i g(s_i)}{d[g_{max} - g(s_i)]} = \delta(s_i) \quad (9)$$

O valor $\delta(s_i)$ é atribuído a cada estrutura ou esquema no instante de sua criação, sendo posteriormente comparado com α . Se a Inequação 9 é satisfeita, a solução s_i não permanece na população. Quanto maior o valor de $\delta(s_i)$, mais tempo a estrutura ou esquema sobrevive e se recombina.

A Inequação 9 pode ser reescrita e expressa pela Equação 10.

$$\delta(s_i) = \frac{1 - \frac{d_i g(s_i)}{dg_{max}}}{1 - \frac{g(s_i)}{g_{max}}} \quad (10)$$

Dessa forma, torna-se mais simples perceber que quanto mais próximo $g(s_i)$ é de $g_{max}(s_i)$ maior é o tempo que este esquema tem para recombinar-se. O mesmo ocorre se d_i for pequeno.

Os parâmetros de entrada d e g_{max} devem ser definidos cuidadosamente, visto que um efeito indesejado pode ocorrer fazendo com que a população atinja um grande número de estruturas quando $g_{max} \gg g(s_i)$. Isto porque todas as estruturas receberão $\delta(s_i) \cong 1$, só sendo eliminadas na geração em que $\alpha = 1$.

O tamanho da população do AGC também apresenta um comportamento típico. Ela cresce inicialmente e depois diminui com o aumento progressivo (ϵ) do parâmetro de evolução α .

Para finalizar, a Figura 30 traz uma visão geral das mudanças que o AGC propõe na abordagem tradicional.

Figura 30 – Visão geral das operações do algoritmo genético construtivo e as mudanças perante a abordagem tradicional.

Algoritmo

O pseudocódigo do algoritmo genético construtivo segue abaixo.

Algoritmo 3 Algoritmo genético construtivo

Definir valores para os parâmetros: ϵ , d , g_{max} , p_{base}

$\alpha = 0$

Construir P_α

Avaliar P_α (cálculo dos valores de $g(s_i)$, $f(s_i)$ e δ_i)

Eliminar indivíduos em que $\alpha \geq \delta_i$

enquanto não atingir o critério de parada **fazer**

Dividir P_α em dois subconjuntos: P_{base} com os n_{base} melhores indivíduos e $P_{n\grave{a}obase}$ com os restantes

Realizar a recombinação e/ou mutação dos indivíduos de P_α

Avaliar P_α (cálculo dos valores de $g(s_i)$, $f(s_i)$ e δ_i)

Eliminar indivíduos em que $\alpha \geq \delta_i$

Incrementar α : $\alpha = \alpha + \epsilon$

fim

Fonte: Furtado (1998), com adaptações da autora.

Ao longo destas quase duas décadas, o AGC foi aplicado em diversos trabalhos que abordavam diferentes problemas, como é o caso da coloração de grafos de Ribeiro e Lorena (2000a), da formação de células de Ribeiro e Lorena (2000b), do arranjo em matriz de portas de Oliveira e Lorena (2002) e da programação *flowshop* permutacional de Nagano, Ruiz e Lorena (2008).

4.4 Algoritmo genético enviesado com chave aleatória

O Algoritmo Genético Enviesada com Chave Aleatória é uma variação de uma outra ramificação algoritmo genético tradicional, chamada de Algoritmo Genético com Chave Aleatória (AGCA), em inglês, Random-Key Genetic Algorithm (RKGA), criado por Bean (1994) com o objetivo de resolver problemas de otimização combinatória que envolvessem sequencia.

Uma das grandes diferenças encontradas nesta variação (BRKGA) está no fato de que os procedimentos para a execução do algoritmo são separados em procedimentos dependentes e procedimentos independentes do problema em questão. A Figura 31 ilustra esta estrutura. Esta independência permite a reutilização do *software*, e por consequência, maior dedicação ao conjunto de procedimentos que realmente dependem do problema.

Figura 31 – Fluxograma do funcionamento do BRKGA

Fonte: Goncalves e Resende (2011)

Comparações entre o algoritmo genético tradicional e a variação BRKGA podem ser encontradas nos trabalhos de Goncalves J.alves e Resende (2011), Goncalves, Resende e Mendes (2011), Goncalves (2007), Goncalves, Mendes e Resende (2005) e Goncalves e Resende (2004).

Como é possível observar, o decodificador é a única etapa que depende do conhecimento do problema. É neste ponto que a *string* do indivíduo será transformada em uma das soluções do espaço de soluções e alimentará a função de avaliação, permitindo assim a associação de um valor de adaptação a mesma.

População inicial

Nesta variação, o indivíduo é representado por um vetor composto de números reais entre 0 e 1 (inclusive), gerados aleatoriamente. Cada indivíduo é associado a uma solução do problema de otimização através de um algoritmo determinístico nomeado de decodificador.

Função de adaptação

Como função de adaptação é usual utilizar índice de qualidade de agrupamento de dados, já descritos neste capítulo.

Seleção, recombinação e mutação

Os indivíduos da população são classificados de forma decrescente, considerando o valor de adaptação de cada um. Os p_e melhores indivíduos são colocados em um grupo chamado de elite, já os $p - p_e$ indivíduos são colocados em outro grupo, chamado de não elite.

A cada nova geração uma cópia idêntica dos indivíduos do grupo elite é realizada para compor a nova geração. Também é gerado um pequeno grupo de p_m indivíduos mutantes a fim de evitar o aprisionamento do algoritmo em soluções locais. Por fim, $p - p_e - p_m$ indivíduos são gerados através da recombinação de dois indivíduos pais da população anterior. A Figura 32 traz a dinâmica desta evolução.

A população t' , onde t' representa o instante imediatamente posterior a ordenação decrescente dos indivíduos, apresenta dois grupos: elite e não elite, como já descrito. Já a população $t' + 1$ é composta por três grupos. O primeiro, chamado de TOP, contém a cópia idêntica dos indivíduos do grupo elite da população t' . O terceiro, chamado de BOT, contém os indivíduos resultantes de um procedimento de mutação. Já o segundo grupo, contendo $p - p_e - p_m$ indivíduos, apresenta os indivíduos resultantes do procedimento de recombinação.

A geração de indivíduos mutantes é realizada da mesma maneira que a geração de indivíduos não mutantes, isto é, através de números gerados aleatoriamente.

A seleção é realizada elegendo um dos pais do grupo elite, e o outro, do grupo não elite. Pode ocorrer também do segundo pai ser selecionado da população t' inteira. Além disso, também é permitido que um indivíduo da população t' , seja ele do grupo elite ou do grupo não elite, seja selecionado mais de uma vez para a reprodução. Sendo assim, um pai pode gerar mais de um filho.

Figura 32 – Método de seleção do BRKGA.

Fonte: Goldberg (1989). Adaptado pela autora.

Nota-se que a probabilidade de se selecionar um indivíduo do grupo elite ($\frac{1}{p_e}$) é maior do que a probabilidade de se eleger um indivíduo do grupo não elite ($\frac{1}{p-p_e}$), isto porque é definido que $p_e < p - p_e$. Desta maneira, os indivíduos do grupo elite apresentam maiores probabilidades de passarem suas características para gerações futuras.

Nesta variação o cruzamento uniforme padronizado é empregado, o que consiste em definir um parâmetro (ρ_e) que ao ser comparado com o valor do alelo dos indivíduos pais (valor real associado a cada atributo na *string* do indivíduo), definirá qual dos alelos contribuirá com o indivíduo filho.

O algoritmo BRKGA realiza os procedimentos descritos repetidamente em cada geração, cessando apenas quando atinge um critério de parada preestabelecido, como um número específico de gerações desde o início; o número fixo de gerações a partir do momento em que não há mais ganho de qualidade; ao atingir um tempo determinado ou ainda quando um determinado limite de qualidade é atingido.

Nesta variação da abordagem tradicional, a população se mantém estável em relação ao número de indivíduos. O algoritmo cessa apenas quando o(s) critério(s) de parada é(ão) satisfeito(s).

Para finalizar, a Figura 33 traz uma visão geral das mudanças que o BRKGA propõe na abordagem tradicional.

Figura 33 – Visão geral das operações do algoritmo genético envisado com chave aleatória e as mudanças perante a abordagem tradicional.

Algoritmo

Algoritmo 4 Algoritmo genético enviesado com chave aleatória

Definir valores para os parâmetros: n , p , p_e , p_m , ρ_c

Inicializar a população, P_{BRKGA}

enquanto não atingir o critério de parada **faça**

 Aplicar o decodificador para avaliar cada indivíduo da população

 Particionar a população em: P_e com os p_e melhores indivíduos e $\bar{P}_e$

 com os $n - p_e$ indivíduos restantes

 Inicializar a próxima geração: $P^+ \leftarrow P_e$

 Gerar conjunto P_m com p_m indivíduos mutantes,

 gerados aleatoriamente por distribuição uniforme

 Adicionar os mutantes na próxima geração: $P^+ \leftarrow P^+ \cup P_m$

para $i \leftarrow 1$ **até** $n - p_e - p_m$ **fazer**

 Selecionar o pai randomicamente de P_e

 Selecionar a mãe randomicamente de P_{BRKGA}

para $j \leftarrow 1$ **até** p **fazer**

 Gerar um número aleatório, a , no intervalo de 0 a 1, inclusive

se a for menor do que ρ_c **então**

 filho[j] $\leftarrow$ pai[j]

senão

 filho[j] $\leftarrow$ mãe[j]

fim

fim

fim

fim

Como é possível visualizar na Figura 33, as principais diferenças da abordagem tradicional residem em três dos setes passos gerais, sendo eles: função de avaliação, seleção de indivíduos e forma de recombinação.

Finaliza-se, portanto, a teoria sobre os algoritmos genéticos que serão utilizados neste trabalho de pesquisa. O conteúdo apresentado neste capítulo servirá como base para o entendimento do capítulo seguinte, Modelagem.

Modelagem

Este capítulo aborda como os algoritmos genéticos foram modelados para o problema de seleção de atributos em agrupamento de dados. Todos os elementos, operadores e parâmetros são definidos, bem como as populações iniciais e os critérios de parada.

5.1 Considerações Gerais

A modelagem dos algoritmos consiste em definir os parâmetros que serão utilizados, métodos de codificação, seleção, recombinação, mutação, além do(s) critério(s) de parada. Alguns parâmetros e métodos são iguais as três abordagens, como é o caso do número de indivíduos da população inicial, que definiu-se por $n = 50$.

A forma de codificação da abordagem tradicional e do algoritmo genético construtivo é a mesma, isto é, optou-se como alfabeto o conjunto binário $\{0,1\}$ para compor as p_{co} primeiras posições da *string*, com p_{co} representando o número de atributos do conjunto original. O número um representa o atributo selecionado e o número zero, o atributo não selecionado. A última posição da *string* ainda conta com um número inteiro, no intervalo de dois a dez, inclusive, que indica o número de grupos.

A função de avaliação também é a mesma para os três algoritmos, isto é, o índice de qualidade de agrupamento CCC, já detalhado no capítulo 2. Apenas no caso do algoritmo genético construtivo, onde são necessárias duas funções de avaliação, há uma particularidade que é detalhada mais a diante.

Como critério de parada, utilizou-se a mesma regra para a abordagem tradicional e a variação enviesado com chave aleatória, que determina que a execução se encerre quando o número de iterações (gerações) for maior do que 30 ou quando, após 10 gerações consecutivas, não houver ganho no valor da solução maior do que 0,01, em valores absolutos, o que acontecer primeiro.

A escolha dos valores dos parâmetros que foram apresentados e os que ainda serão introduzidos foi definido experimentalmente, em testes empíricos realizados em diversas

bases de dados e também nas considerações de trabalhos na área (GOLDBERG, 1989) (TSAI; EBERLE; CHU, 2013) (SCRUCCA, 2010) (MORITA et al., 2003) (LIU; ONG, 2008).

Com relação aos demais parâmetros e operações, cada base de dados necessita de um ajuste diferenciado, visto que são diferentes estruturalmente, principalmente a primeira, autenticidade de cédula, que contém quatro atributos, enquanto as outras, qualidade do vinho branco e qualidade do ar, que contém onze e treze atributos, respectivamente.

5.2 Algoritmo genético

A seleção de indivíduos para reprodução é feita através da seleção por roleta, já explicada no capítulo 4. A recombinação é feita através da seleção aleatória de duas posições da *string*, bem como da seleção aleatória de pares (pai e mãe), sem repetição, da população em questão.

Esta recombinação só é realizada apenas se, ao gerar um número real aleatório no intervalo de zero a um, este for menor ou igual a p_c (taxa de recombinação), que neste trabalho foi definida como $p_c = 0,95$.

A mutação, para ser realizada, também se baseia na comparação de um número real gerado aleatoriamente no intervalo de zero a um com uma taxa de mutação p_m , que neste caso é de 0,1. É realizada a mutação de um ponto.

As Tabelas 5 e 6 trazem um resumo dos parâmetros e operações definidas para cada base de dados, onde A.C. significa o banco de dados autenticidade de cédula, Q.V.B. significa o banco de dados qualidade do vinho branco e Q.A. significa o banco de dados qualidade do ar.

Tabela 5 – Parâmetros definidos para cada base de dados na modelagem do algoritmo genético tradicional

Parâmetros			
	População inicial, n	taxa de recombinação, p_c	taxa de mutação, p_m
A.C.	50	0,95	0,1
Q.V.B.	50	0,95	0,1
Q.A.	50	0,95	0,1

Tabela 6 – Operações definidas para cada base de dados na modelagem do algoritmo genético tradicional

Operações						
	Alfabeto	Seleção	F. Avaliação	Recombinação	Mutação	Critério de parada
A.C.	binário e número	por roleta	índice	de dois	de um	até 30 gerações ou até
Q.V.B.	inteiro de 2 a 10		CCC	pontos	ponto	10 gerações consecutivas
Q.A.	para grupos					sem ganho de desempenho

Como é possível observar, em ambas tabelas, são definidos os mesmos parâmetros e as mesmas operações para os três algoritmos.

5.3 Algoritmo genético construtivo

As funções de avaliação bi-objetiva $f-g$ foram definidas como segue, Equações 11 e 12:

$$f(s_i) = \frac{1}{CCC_{max}} \quad (11)$$

$$g(s_i) = \frac{1}{CCC} \quad (12)$$

Sendo CCC_{max} o valor do índice CCC (apresentado no capítulo 2) calculado para o caso em que todas as distâncias entre os pontos pertencentes a um determinado grupo até seu centro recebem o valor da menor distância. Assim, tem-se o maior valor do índice para aquele subconjunto.

Dessa forma, garante-se que $f(s_i) \leq g(s_i)$. A inversão do valor CCC tanto para a definição de $f(s_i)$ quanto de $g(s_i)$ deve-se ao fato de que o índice CCC expressa maior qualidade de agrupamento quanto maior este é, indo de encontro com o que se busca na função bi-objetiva, minimização entre $f(s_i)$ e $g(s_i)$, sendo f menor que $g(s_i)$.

No algoritmo genético construtivo, a população inicial é composta apenas por bons esquemas e esta é uma das grandes diferenças da abordagem tradicional. Para escolher apenas esquemas promissores, optou-se por construir a população deterministicamente, e por isso, foi analisado os coeficiente de correlação de cada variável (Apêndice A), de modo a auxiliar a decisão de quais iriam compor o subconjunto dos indivíduos.

Como limite superior g_{max} , definiu-se que o algoritmo adotará o maior valor entre: o maior valor do índice CCC para o conjunto de dados original (sem a seleção de atributos) considerando a variação de grupos de dois a dez (inclusive) ou, o valor de CCC_{max} para o grupo que resultou em maior índice CCC no conjunto completo de dados (sem a seleção de atributos) multiplicado por uma fator $\rho_{g_{max}}$.

O valor de $\rho_{g_{max}}$ foi definido conforme a base de dados para obtenção de maior convergência, a saber: (i) autenticidade de cédula: $\rho_{g_{max}} = 0,000$; (ii) qualidade do vinho branco: $\rho_{g_{max}} = 0,600$ e (iii) qualidade do ar: $\rho_{g_{max}} = 0,000$.

Definiu-se como desvio absoluto d em relação a g_{max} , os seguintes valores para cada base de dados: (i) autenticidade de cédula: $d = 0,900$; (ii) qualidade do vinho branco: $d = 0,600$ e (iii) qualidade do ar: $d = 0,500$.

O incremento ϵ de α também foi definido conforme a base de dados, como segue: (i) autenticidade de cédula: $\epsilon = 0,010$ até $\alpha < 0,050$ e $\epsilon = 0,080$ para $\alpha \geq 0,050$ e ; (ii) qualidade do vinho branco: $\epsilon = 0,100$ até $\alpha < 0,300$ e $\epsilon = 0,300$ para $\alpha \geq 0,300$ e (iii) qualidade do ar: $\epsilon = 0,100$ até $\alpha < 0,300$ e $\epsilon = 0,300$ para $\alpha \geq 0,300$.

A porcentagem que define a população base foi estabelecida em 0,3, ou seja, 30% dos melhores indivíduos constituirão a população base.

Os procedimentos de recombinação foram definidos como segue:

{recombinação}

1. se $s_{base_i} = \#$ e $s_{guide_i} = \#$ então $s_{novo_i} = 0$
2. se $s_{base_i} = 1$ e $s_{guide_i} = 1$ então $s_{novo_i} = 1$
3. se $s_{base_i} = 0$ e $s_{guide_i} = 0$ então $s_{novo_i} = 0$
4. se $s_{base_i} = 1$ e $s_{guide_i} = \#$ então $s_{novo_i} = 1$
5. se $s_{base_i} = 0$ e $s_{guide_i} = \#$ então $s_{novo_i} = 0$
6. se $s_{base_i} = \#$ e $s_{guide_i} = 0$ então $s_{novo_i} = 0$
7. se $s_{base_i} = \#$ or 0 e $s_{guide_i} = 1$ então há dois casos possíveis:
 - 7.1. atribua $s_{novo_j} = 1$, também se $|V_1(s_{novo})| = p$, atribua $s_{novo_l} = 0$ para $l \in 1, \dots, n$ apresentando $s_{novo_l} = 1$ (seleção aleatória)
 - 7.2. atribua $s_{novo_j} = 1$, também se $|V_1(s_{novo})| = p$, atribua $s_{novo_l} = 0$ para cada $l \in 1, \dots, n$ apresentando $s_{novo_l} = 1$ gerar p novas estruturas e/ou esquemas.
8. se $s_{base_j} = 1$ e $s_{guide_j} = 0$ então há dois casos possíveis:
 - 8.1. atribua $s_{novo_j} = 1$ and $s_{novo_l} = 0$ para $l \in 1, \dots, n$ apresentando $s_{novo_l} = 1$ (aleatoriamente selecionadas)
 - 8.2. atribua $s_{novo_j} = 1$ and $s_{novo_l} = 0$ para cada $l \in 1, \dots, n$ apresentando $s_{novo_l} = 1$ gerando $|V_0(s_{base})|$ novas estruturas e/ou estruturas
9. se $s_{base_j} = k$, em que $k \in 2, \dots, 10$ representar o número de grupo e, $s_{guide_i} = k^*$ então s_{novo_i} é igual a k se p_k for maior ou igual a um número real gerado aleatoriamente entre 0 e 1, inclusive, em que p_k representa uma taxa de recombinação para grupo. Caso contrário, o novo indivíduo recebe como número de grupo k .

Tem-se que, $|V_0(s_i)|$ representa a quantidade de zeros presentes na *string* do indivíduo e $|V_1(s_i)|$ representa a quantidade de uns presentes na *string* do indivíduo.

Quando s_{base} é uma estrutura ($|V_{\#}(s_i)| = 0$), em que $|V_{\#}(s_i)|$ representa a quantidade de caracter $\#$ presentes na *string* do indivíduo, então a mutação é feita através da execução da Heurística de Permutação (HP). O pseudocódigo deste algoritmo pode ser visto abaixo (LORENA; FURTADO, 2001).

{mutação}

Algoritmo 5 Heurística de permutação

para $j \leftarrow 1$ até $|V_1(s_{base})|$ **faça**
 para $l \leftarrow 1$ até $|V_0(s_{base})|$ **faça**
 Permutar j e l
 Gerar um novo indivíduo, s_{novo}
 Permutar l e j
 fim
fim

Dois critérios de parada são utilizados nesta abordagem para determinar quando o algoritmo deve parar, o que acontecer primeiro: a população ficar vazia ou o algoritmo atingir o número de iterações (gerações) predeterminado, no caso 30.

As Tabelas 7 e 8 trazem um resumo dos parâmetros e operações definidas para cada base de dados, onde A.C. significa o banco de dados autenticidade de cédula, Q.V.B. significa o banco de dados qualidade do vinho branco e Q.A. significa o banco de dados qualidade do ar.

Tabela 7 – Parâmetros definidos para cada base de dados na modelagem do algoritmo genético construtivo

		Parâmetros					
	População inicial, n	d	ρ_{gmax}	ϵ	p_{base}	p_k	
A.C.	50	0,9	0,0	0,01 para $\alpha < 0,05$ 0,08 para $\alpha \geq 0,05$	0,3	0,8	
Q.V.B.	50	0,6	0,5	0,1 para $\alpha < 0,3$ 0,3 para $\alpha \geq 0,3$	0,3	0,8	
Q.A.	50	0,5	0,0	0,1 para $\alpha < 0,3$ 0,3 para $\alpha \geq 0,3$	0,3	0,8	

Tabela 8 – Operações definidas para cada base de dados na modelagem do algoritmo genético construtivo

		Operações				
	Alfabeto	Seleção	F. Avaliação	Recombinação	Mutação	Critério de parada
A.C.	binário e número inteiro de 2 a 10 para grupos	sbase da	$g = \frac{1}{CCC}$ $f = \frac{1}{CCC_{max}}$	regras específicas	heurística de permutação	população vazia ou até 30 gerações
Q.V.B.		população base e sgua da população				
Q.A.		inteira				

Como é possível observar, na Tabela 8, são definidos as mesmas operações para os três algoritmos.

5.4 Algoritmo genético enviesado com chave aleatória

Esta variação do AG requer que a população seja constituída de valores reais gerados aleatoriamente no intervalo de zero a um, inclusive. Por isto, para o problema deste trabalho de pesquisa, a população é representada por uma matriz com n indivíduos (linhas) por $p + 1$ atributos (colunas). Adiciona-se um ao número p de atributos para informar o número de grupos.

A decodificação da *string* com relação a seleção ou não do atributo é feita comparando o número real gerado aleatoriamente na posição do atributo correspondente a um número predefinido ρ_s , no caso $\rho_s = 0,5$. Caso o número real seja maior ou igual a ρ_s então o atributo em questão é selecionado para compor o subconjunto a ser analisado pelo algoritmo. Caso o número real seja menor que ρ_s , o atributo não é selecionado.

Com relação a decodificação do número de grupos, definiu-se as seguintes regras, sendo que neste caso i representa a posição na *string* do número de grupos:

{decodificação - número de grupos}

1. se $0 \leq s_i < 0,111$ então $s_{novo_i} = 2$
2. se $0,111 \leq s_i < 0,222$ então $s_{novo_i} = 3$
3. se $0,222 \leq s_i < 0,333$ então $s_{novo_i} = 4$
4. se $0,333 \leq s_i < 0,444$ então $s_{novo_i} = 5$
5. se $0,444 \leq s_i < 0,555$ então $s_{novo_i} = 6$
6. se $0,555 \leq s_i < 0,666$ então $s_{novo_i} = 7$
7. se $0,666 \leq s_i < 0,777$ então $s_{novo_i} = 8$
8. se $0,777 \leq s_i < 0,888$ então $s_{novo_i} = 9$
9. se $0,888 \leq s_i < 0,100$ então $s_{novo_i} = 10$

Para a seleção de indivíduos, definiu-se como porcentagem para a população elite 20%. Para a porcentagem para população de mutantes 10% e como parâmetro de cruzamento: $\rho_e = 0,7$ para todas as bases de dados.

As Tabelas 9 e 10 trazem um resumo dos parâmetros e operações definidas para cada base de dados, onde A.C. significa o banco de dados autenticidade de cédula, Q.V.B. significa o banco de dados qualidade do vinho branco e Q.A. significa o banco de dados qualidade do ar.

Tabela 9 – Parâmetros definidos para cada base de dados na modelagem do algoritmo genético enviesado com chave aleatória.

		Parâmetros				
	População inicial, n	p_e	p_m	ρ_s	ρ_e	
A.C.	50	0,3	0,2	0,5	0,7	
Q.V.B.	50	0,3	0,2	0,5	0,7	
Q.A.	50	0,3	0,2	0,5	0,7	

Tabela 10 – Operações definidas para cada base de dados na modelagem do algoritmo genético enviesado com chave aleatória

		Operações					
	Alfabeto	Decodificador	Seleção	F. Avaliação	Recombinação	Mutação	Critério de parada
A.C.	números reais		um indivíduo da				até 30 gerações ou
			população elite	índice	comparação	geração	até 10 gerações
Q.V.B.	entre 0 e 1, inclusive	regras específicas	e o outro da população	CCC	alelo a alelo	aleatória	consecutivas sem ganho de
Q.A.			inteira				desempenho

Como é possível observar, em ambas tabelas, são definidos os mesmos parâmetros e as mesmas operações para os três algoritmos.

Finaliza-se, portanto, a modelagem dos algoritmos genéticos. As operações e parâmetros definidos neste capítulo para cada algoritmo foram utilizados para a emissão dos resultados experimentais do próximo capítulo, Experimentação computacional.

Experimentação computacional

Este capítulo traz a experimentação computacional realizada em três bases de dados reais, extraídas do repositório de aprendizado de máquinas UCI. Uma descrição estatística e qualitativa de cada base de dados são apresentadas. A estrutura computacional utilizada para a emissão dos resultados, bem como a discussão destes também estão presentes neste capítulo.

6.1 Bases de dados

As três bases de dados escolhidas para a emissão dos resultados práticos foram retiradas no repositório de dados de aprendizado de máquina UCI, cujo endereço é <https://archive.ics.uci.edu/ml/index.html>. Todas as bases suportam a execução de seleção de variáveis por algoritmos genéticos propostos neste trabalho.

Autenticação de cédula

Criada por Volker Lohweg (LOHWEG, 2012), esta base de dados se refere a informações de imagens escaneadas em escala de cinza de cédulas (notas de dinheiro). É composta por números contínuos que representam as estatísticas transformadas destas imagens. Foi idealizada com o objetivo de identificar notas falsas e verdadeiras.

Com 1.372 registros, a base apresenta cinco atributos, sendo que um deles não é utilizado neste trabalho. Os quatro atributos utilizados são referentes as características dos coeficientes das imagens digitalizadas (GILLICH; LOHWEG, 2010). Os dados foram inseridos no repositório no ano de 2013 e, se encontra no seguinte endereço <https://archive.ics.uci.edu/ml/datasets/banknote+authentication>.

A base de dados já foi utilizada em trabalhos acadêmicos (KUMAR; DUDYALA, 2015) e (GHAZVINI; AWWALU; BAKAR, 2014).

Na Tabela 11 é possível observar as estatísticas descritivas dos dados. Destaca-se que os valores que não apresentam unidades de medida são adimensionais.

Tabela 11 – Estatísticas descritivas da base de dados: autenticidade de cédula

Atributos	Média	Desvio padrão	Valor máximo	Valor mínimo
VWTI	0,434	2,843	6,825	-7,042
SWTI	1,922	5,869	12,952	-13,773
CWTI	1,398	4,310	17,927	-5,286
EWTI	-1,192	2,101	2,450	-8,548

As estatísticas descritivas da Tabela 11 revelam a média, desvio padrão, valor máximo e mínimo de cada variável. É possível observar que a variável AWTI apresenta o maior desvio padrão, seguido da variável CWT. As variáveis VWTI e EWTI seguem com os maiores desvios, respectivamente.

É possível observar também que o intervalo em que os valores das variáveis estão distribuídos são bem diferentes. Exemplo disto são as variáveis: SWTI, cujo valor mínimo é de -13,773 e o máximo é de 12,952 e VWTI com valor mínimo de -7,042 e valor máximo de 6,825.

Esta diferença dos valores absolutos de cada variável requer que os dados sejam normalizados para melhor desempenho dos algoritmos genéticos. Neste trabalho foi utilizado a função $\text{normc}(M)$ do Matlab, a qual normaliza as colunas da matriz M para o comprimento de 1.

A fim de visualizar graficamente as informações da Tabela 11, o Gráfico 34 de traz o gráfico de caixas.

Figura 34 – Gráfico de caixa da base de dados autenticação de cédulas.

No gráfico de caixas fica evidente graficamente a diferença entre as médias das variáveis, os desvios padrão de cada uma e os valores de mínimo e máximo.

Qualidade do vinho branco

Criada por Paulo Cortez (CORTEZ et al., 2009b), esta base de dados reúne informações sobre propriedades físico-químicas de vinhos brancos da região do Minho, norte de Portugal. Apresenta um total de 4.898 registros e 12 variáveis, sendo que uma delas representa o rótulo de classificação e não será utilizada neste trabalho (CORTEZ et al., 2009a).

Os dados foram inseridos no repositório no ano de 2009 e, se encontra no seguinte endereço <https://archive.ics.uci.edu/ml/datasets/Wine+Quality>. O autor declara que não foi verificado se todas as variáveis são relevantes, e que por isso, seria interessante realizar a seleção de atributos.

Esta base de dados já foi largamente utilizada na literatura para problemas de classificação, regressão e seleção de atributos (LEE; PARK; KANG, 2015), (APPALASAMY et al., 2012), (HU et al., 2016), (VACHKOV, 2016) e (CORTEZ et al., 2009c).

As estatísticas descritivas, bem como as unidades de medida de cada atributo, seguem na Tabela 12.

Tabela 12 – Estatísticas descritivas da base de dados: qualidade do vinho branco.

Atributo (unidade)	Média	Desvio padrão	Valor mínimo	Valor máximo
Acidez fixa ($g_{tartaricacid}/dm^3$)	6,855	0,844	3,800	14,200
Acidez volátil ($g_{aceticacid}/dm^3$)	0,278	0,101	0,080	1,100
Ácido cítrico (g/dm^3)	0,334	0,121	0,000	1,660
Açúcar residual (g/dm^3)	6,391	5,072	0,600	65,800
Cloretos ($g_{sodiumchloride}/dm^3$)	0,046	0,022	0,010	0,350
Dióxido de enxofre livre (mg/dm^3)	35,308	17,007	2,000	289,000
Dióxido de enxofre total (mg/dm^3)	138,361	42,498	9,000	440,000
densidade (g/dm^3)	0,994	0,005	0,990	1,040
pH	3,188	0,151	2,720	3,820
Sulfatos ($g_{potassiumsulphate}/dm^3$)	0,490	0,114	0,220	1,080
Álcool (%vol)	10,514	1,231	8,000	14,200

As estatísticas descritivas da Tabela 12 revelam a média, desvio padrão, valor máximo e mínimo de cada variável. É possível observar que a variável Dióxido de enxofre total apresenta o maior desvio padrão, seguido da variável Dióxido de enxofre livre. As variáveis Açúcar residual e Álcool seguem com os maiores desvios, respectivamente.

É possível observar também que o intervalo em que os valores das variáveis estão distribuídos são bem diferentes. Exemplo disto são as variáveis: Acidez volátil, cujo valor mínimo é de 0,080 e o máximo é de 1,100 e Dióxido de enxofre total com valor mínimo de 9,000 e valor máximo de 440,000.

Esta diferença dos valores absolutos de cada variável requer que os dados sejam normalizados para melhor desempenho dos algoritmos genéticos.

A fim de visualizar graficamente as informações da Tabela 12, o Gráfico 35 de traz o gráfico de caixas.

Figura 35 – Gráfico de caixa da base de dados qualidade de vinho.

No gráfico de caixas fica evidente graficamente a diferença entre as médias das variáveis, os desvios padrão de cada uma e os valores de mínimo e máximo.

Qualidade do ar

Criada por Saverio De Vito (VITO, 2008), esta base de dados reúne características do ar atmosférico de uma cidade da Italia, capturadas através de um dispositivo multisensor de gás. As informações encontradas na base de dados são referentes a média da resposta do sensor em uma hora juntamente com a referencia certificada de concentração de gases.

A base apresenta 9.358 registros e 15 variáveis, sendo que duas não serão utilizadas neste trabalho por não irem ao encontro do objetivo, seleção de variáveis (VITO E. MASSERA, 2008).

Os dados foram inseridos no repositório no ano de 2016 e, se encontra no seguinte endereço <https://archive.ics.uci.edu/ml/datasets/Air+Quality>.

A base de dados já foi utilizada em trabalhos acadêmicos (VITO et al., 2012), (VITO et al., 2009) e (VITO et al., 2008).

As estatísticas descritivas e as unidades de medida de cada atributo seguem na Tabela 13. Destaca-se que os atributos que não possuem unidade de medidas são adimensionais.

As estatísticas descritivas da Tabela 13 revelam a média, desvio padrão, valor máximo e mínimo de cada variável. É possível observar que a variável PT08.S5(O3) apresenta o maior desvio padrão, seguido da variável PT08.S4(NO2). As variáveis PT08.S2(NMHC) e PT08.S3(NOx) seguem com os maiores desvios, respectivamente.

É possível observar também que o intervalo em que os valores das variáveis estão distribuídos são bem diferentes. Exemplo disto são as duas primeiras variáveis: CO(GT), cujo valor mínimo é de 0,300 e o máximo é de 8,100 e PT08.S1(CO) com valor mínimo de 753,000 e valor máximo de 2.040,000.

Esta diferença dos valores absolutos de cada variável requer que os dados sejam normalizados para melhor desempenho dos algoritmos genéticos.

Tabela 13 – Estatísticas descritivas da base de dados: qualidade do ar

Atributo (unidade)	Média	Desvio padrão	Valor máximo	Valor mínimo
CO(GT) (mg/cm^3)	2,354	1,409	8,100	0,300
PT08.S1(CO)	1.207,879	241,817	2.040,000	753,000
NMHC(GT) (mg/cm^3)	231,025	208,462	1.189,000	7,000
C6H6(GT) (mg/cm^3)	10,771	7,418	39,200	0,500
PT08.S2(NMHC)	966,116	266,425	1.754,000	448,000
NOx(GT) (ppb)	143,502	81,830	478,000	12,000
PT08.S3(NOx)	963,297	265,914	1.935,000	461,000
NO2(GT)	100,260	31,494	196,000	19,000
PT08.S4(NO2) (mg/cm^3)	1.600,620	302,292	2.679,000	955,000
PT08.S5(O3)	1.045,813	400,135	2.359,000	263,000
T (°C)	15,601	4,825	30,000	6,300
RH (%)	49,050	15,267	83,200	14,900
AH	0,832	0,179	1,485	0,402

A fim de visualizar graficamente as informações da Tabela 13, o Gráfico 36 de traz o gráfico de caixas.

Figura 36 – Gráfico de caixa da base de dados qualidade do ar.

No gráfico de caixas fica evidente graficamente a diferença entre as médias das variáveis, os desvios padrão de cada uma e os valores de mínimo e máximo.

6.2 Estrutura computacional

Os algoritmos genéticos foram executados em computador portátil MacBook Pro 2011 de 13 polegadas, com memória RAM de dez gigabites e processador Intel(R) Core(TM)

i5-2435M (2,40GHz). Apesar do notebook ser da empresa Apple, o sistema operacional utilizado foi o Windows 7.0 Professional, de 64bits. Os resultados computacionais foram gerados pelos softwares: MATLAB 8.5.0 (MATLAB, 2015) para a programação dos algoritmos genéticos em conjunto com o R 3.2.5 (R Development Core Team, 2008) para a emissão dos valores do índice CCC através do pacote NbClust (CHARRAD et al., 2014); IBM SPSS Statistics 20 para a emissão das estatísticas descritivas das bases de dados, bem como para o teste T de amostras independentes (CORPORATION, 2011) e Minitab (INC, 2010) para a emissão dos resultados do teste estatístico de Tukey.

6.3 Resultados e discussão

Inicialmente, calculou-se o índice CCC para as três bases de dados com todos os respectivos atributos, isto é, sem que houvesse a seleção de um subconjunto de variáveis. Calculou-se o índice para um intervalo de grupos [2,10], a fim de identificar o maior valor do índice em cada base de dados, conforme observado na Tabela 14.

Tabela 14 – Valores do índice CCC para o conjunto de atributos completos, sem seleção de variáveis

	Número de grupos								
	2	3	4	5	6	7	8	9	10
autenticidade de cédula	5,884	6,412	8,863	15,262	16,491	14,968	16,770	16,163	16,360
qualidade do vinho	113,399	83,624	86,991	94,902	100,474	103,118	101,416	100,847	100,917
qualidade do ar	49,505	51,282	46,296	43,290	42,553	36,900	37,313	39,370	41,152

Nota-se que o maior valor do índice CCC para a base de dados autenticidade de cédula é de 16,770 quando agrupa-se todos os dados do conjunto original em 8 grupos. Para a base de dados qualidade do vinho, o maior valor do índice é 113,399, correspondente a 2 grupos. E, por fim, o maior valor do índice CCC para a base de dados qualidade do ar é de 51,282 correspondente a 3 grupos.

A identificação dos maiores valores do índice para cada base na Tabela 14 é importante pois, o algoritmo genético construtivo, tem como limite superior g_{max} o maior valor do índice CCC encontrado na Tabela 14 para a respectiva base de dados, ou então, g_{max} assume o valor da multiplicação de $\rho_{g_{max}}$ e o valor da função f com o número de grupos indicado como o melhor na Tabela 14 para a respectiva base de dados, o que resultar em maior valor.

Na Tabela 15 os algoritmos propostos são executados 10 vezes para a base de dados: autenticidade de cédula, a fim de identificar o valor do índice CCC, tempo de execução em segundos, os atributos selecionados e o número de grupos.

Tabela 15 – Comparação dos resultados obtidos na execução dos três algoritmos para base de dados: autenticidade de cédula

repetição	Índice CCC			Tempo de execução (segundos)			Atributos selecionados			Número de grupos		
	AG	AGC	BRKGA	AG	AGC	BRKGA	AG	AGC	BRKGA	AG	AGC	BRKGA
1	21,346	21,345	21,109	12,875	28,928	5,392	2, 3, 4	2, 3, 4	2, 3, 4	9	9	10
2	21,346	21,345	21,346	9,517	31,196	6,183	2, 3, 4	2, 3, 4	2, 3, 4	9	9	9
3	21,1009	21,345	21,346	13,577	31,485	7,179	2, 3, 4	2, 3, 4	2, 3, 4	10	9	9
4	21,346	21,345	21,346	14,271	69,075	7,630	2, 3, 4	2, 3, 4	2, 3, 4	9	9	9
5	21,346	17,394	21,346	9,857	5,816	6,586	2, 3, 4	3, 4	2, 3, 4	9	9	9
6	21,346	21,110	21,346	18,414	75,726	8,9	2, 3, 4	2, 3, 4	2, 3, 4	9	10	9
7	21,346	21,345	21,346	12,861	21,576	4,383	2, 3, 4	2, 3, 4	2, 3, 4	9	9	9
8	21,346	20,640	21,346	5,941	17,454	9,036	2, 3, 4	3, 4	2, 3, 4	9	8	9
9	21,346	21,110	21,346	8,292	33,284	4,907	2, 3, 4	2, 3, 4	2, 3, 4	9	10	9
10	21,346	21,345	21,110	9,41	34,091	5,036	2, 3, 4	2, 3, 4	2, 3, 4	9	9	10
Média	21,321	20,832	21,299	11,501	34,863	6,526						
Desvio padrão	0,077	1,229	0,1	3,586	21,671	1,652						

Como é possível observar, a média do índice CCC dos três algoritmos foram próximas uma das outras. No entanto, o desvio padrão do algoritmo genético construtivo foi maior do que as duas outras abordagens, 1,229, contra 0,077 da abordagem tradicional e 0,1 do algoritmo genético enviesado com chave aleatória.

Já com relação ao tempo de execução, constata-se que o algoritmo genético construtivo necessitou de um tempo maior do que os outros dois algoritmos, sendo este tempo de 34,863 segundos, contra 11,501 da abordagem tradicional e 6,526 do algoritmo genético enviesado com chave aleatória.

Os atributos selecionados foram, em sua maioria: SWTI (2), CWT (3) e EWT (4). Apenas em duas repetições (5^a e 8^a) do algoritmo genético construtivo, selecionou-se somente os atributos CWT (3) e EWT (4). Os índices CCC destas repetições cujo subconjunto apresentava apenas duas variáveis não foram altos, ou seja, não foram bons. Analisando as repetições como um todo, acredita-se que o algoritmo ao verificar que o subconjunto de três atributos apresentava bons resultados, decidiu verificar se o índice CCC melhorava com a retirada de um dos atributos. Identificando que não, pois o valor do índice caiu, voltou a selecionar um subgrupo com três atributos.

É interessante observar que em nenhum caso a variável VWTI (1) foi selecionada para compor o subconjunto. Ao retornar no gráfico de caixas, Gráfico 34, observa-se que apenas esta variável tem uma simetria bem definida em relação ao eixo vertical (média) e não apresenta nenhum ponto extremo.

Com relação ao número de grupos, observa-se que em todos os algoritmos e em todas as repetições o número de grupos variou entre 8, 9 e 10. Retornando a Tabela 14 é possível constatar que esta variação está próxima ao número de grupo indicado como bom para quando esta base de dados está completa, sem seleção de subconjunto de variáveis, que é

8.

Na Tabela 16 os algoritmos propostos são executados 10 vezes para a base de dados: qualidade do vinho branco, a fim de identificar o valor do índice CCC, tempo de execução em segundos, os atributos selecionados e o número de grupos.

Tabela 16 – Comparação dos resultados obtidos na execução dos três algoritmos para base de dados: qualidade do vinho branco

repetição	Índice CCC			Tempo de execução (segundos)			Atributos selecionados			Número de grupos		
	AG	AGC	BRKGA	AG	AGC	BRKGA	AG	AGC	BRKGA	AG	AGC	BRKGA
1	443,548	840,336	646,116	127,897	201,383	83,029	8, 9	8	8	2	3	2
2	443,548	840,336	842,571	124,252	230,305	87,063	8, 9	8	8	2	3	3
3	338,743	840,336	646,116	170,707	180,423	55,551	8, 9, 11	8	8	2	3	2
4	443,548	840,336	842,571	148,778	115,591	51,441	8, 9	8	8	2	3	3
5	443,548	645,161	842,571	123,265	329,364	64,343	8, 9	8	8	2	2	3
6	350,469	840,336	646,116	97,173	160,226	102,245	8, 9	8	8	2	3	2
7	443,548	840,336	646,116	165,270	139,770	92,262	8, 9	8	8	3	3	2
8	646,116	840,336	842,571	164,769	108,934	110,855	8	8	8	2	3	3
9	318,714	840,336	842,571	73,134	116,779	49,809	8, 9	8	8	2	3	3
10	342,903	840,336	646,116	76,691	107,883	82,474	8, 11	8	8	2	3	2
Média	421,468	820,819	744,343	127,194	169,066	77,907						
Desvio padrão	95,172	61,72	103,541	35,932	70,323	21,573						

Como é possível observar, a média do índice CCC dos três algoritmos foram bem diferentes, principalmente a do algoritmo genético construtivo (820,819) e a da abordagem tradicional (421,468), quase o dobro uma da outra, respectivamente. Já o desvio padrão, observa-se que o algoritmo genético construtivo obteve o menor desvio entre os três (61,72), indicando que os resultados do índice desta abordagem não apresentam muita variação ao longo das repetições.

Já com relação ao tempo de execução, constata-se que o algoritmo genético construtivo necessitou de um tempo maior do que os outros dois algoritmos, sendo este tempo de 169,066 segundos, contra 127,194 da abordagem tradicional e 77,907 do algoritmo genético enviesado com chave aleatória.

O atributo selecionado em todas as repetições do algoritmo genético construtivo e do algoritmo genético enviesado com chave aleatória o atributo Densidade (8). Algumas repetições do algoritmo genético tradicional selecionou, além do atributo Densidade (8), os atributos: pH (9) e Álcool (11). Os índices CCC das repetições cujo subconjunto apresentava mais de uma variável não foram altos, isto é, não foram bons.

Com relação ao número de grupos, observa-se que em todos os algoritmos e em todas as repetições o número de grupos variou entre 2 e 3. Retornando a Tabela 14 é possível constatar que esta variação está próxima ao número de grupo indicado como bom para

quando esta base de dados está completa, sem seleção de subconjunto de variáveis, que é 2.

Na Tabela 17 os algoritmos propostos são executados 10 vezes para a base de dados: qualidade do ar, a fim de identificar o valor do índice CCC, tempo de execução em segundos, os atributos selecionados e o número de grupos.

Tabela 17 – Comparação dos resultados obtidos na execução dos três algoritmos para base de dados: qualidade do ar

repetição	Índice CCC			Tempo de execução (segundos)			Atributos selecionados			Número de grupos		
	AG	AGC	BRKGA	AG	AGC	BRKGA	AG	AGC	BRKGA	AG	AGC	BRKGA
1	91,839	90,253	91,839	9,370	14,063	7,612	7, 9	7, 9	7, 9	7	10	7
2	92,536	92,507	89,818	9,704	11,158	8,121	7, 9	7, 9	7, 9	8	8	5
3	81,185	92,507	92,536	9,478	10,471	6,887	2, 5, 7, 9	7, 9	7, 9	4	8	8
4	89,334	92,507	90,045	9,605	10,759	5,381	5, 7	7, 9	5, 7	8	8	9
5	90,256	92,507	91,839	9,453	10,627	5,832	7, 9	7, 9	7, 9	10	8	7
6	87,836	92,507	89,818	10,028	10,563	9,224	5, 7, 9	7, 9	7, 9	3	8	5
7	84,082	92,421	92,536	9,892	12,175	5,693	2, 5, 7, 9	7, 9	7, 9	3	9	8
8	92,427	92,507	89,818	10,090	11,221	12,224	7, 9	7, 9	7, 9	9	8	5
9	92,427	92,507	91,839	9,813	9,354	8,780	7, 9	7, 9	7, 9	9	8	7
10	87,836	92,421	88,810	9,897	13,153	4,180	5, 7, 9	7, 9	7, 9	3	9	5
Média	88,976	92,264	90,889	9,732	11,354	7,393						
Desvio padrão	3,854	0,708	1,359	0,25	1,398	2,337						

Como é possível observar, a média do índice CCC dos três algoritmos variou de os valores 88,976 (algoritmo genético tradicional) a 92,264 (algoritmo genético construtivo). No entanto, o desvio padrão do algoritmo genético tradicional foi maior do que as duas outras abordagens, 3,854, contra 0,708 do algoritmo genético construtivo e 1,359 do algoritmo genético enviesado com chave aleatória.

Já com relação ao tempo de execução, constata-se que o algoritmo genético construtivo necessitou de um tempo maior do que os outros dois algoritmos, sendo este tempo de 11,354 segundos, contra 9,732 da abordagem tradicional e 7,393 do algoritmo genético enviesado com chave aleatória.

Os atributos selecionados foram, em sua maioria: PT08.S3(NOx) (7) e PT08.S4(NO2) (9). Apenas em algumas repetições dos algoritmos genéticos, selecionou-se o atributo PT08.S2(NMHC) (5) e PT08.S1(CO) (2).

Com relação ao número de grupos, observa-se este variou entre 8, 9 e 10. Houve também registros de número de grupos como 3, 4 e 5, mas em menor quantidade. Neste caso, retornando a Tabela 14, não é possível constatar que esta variação esteja próxima ao número de grupo indicado como bom para quando esta base de dados está completa, sem seleção de subconjunto de variáveis, que é 3.

As Tabelas 18 e 19 trazem resumidamente a média e o desvio padrão de cada base de dados e cada algoritmo genético em relação ao desempenho (valores do índice CCC) e do tempo de execução.

Tabela 18 – Média e desvio padrão dos valores do índice CCC obtidos na execução dos três algoritmos para as três base de dados

		AG	AGC	BRKGA
autenticidade de cédula	média	21,321	20,832	21,299
	desvio padrão	0,077	1,229	0,1
qualidade do vinho branco	média	421,468	820,819	744,343
	desvio padrão	95,172	61,72	103,541
qualidade do ar	média	88,976	92,264	90,889
	desvio padrão	3,854	0,708	1,359
	média	177,255	311,305	285,51
	desvio padrão	53,846	35,076	59,362

Tabela 19 – Média e desvio padrão dos tempos de execução obtidos pelos três algoritmos para as três base de dados

		AG	AGC	BRKGA
autenticidade de cédula	média	11,501	34,863	6,526
	desvio padrão	3,586	21,671	1,652
qualidade do vinho branco	média	127,194	169,066	77,907
	desvio padrão	35,932	70,323	21,573
qualidade do ar	média	9,732	11,354	7,393
	desvio padrão	0,25	1,398	2,337
	média	49,476	71,761	30,609
	desvio padrão	19,709	35,423	11,309

O Gráfico 37 traz as médias dos valores do índice CCC, os respectivos tempos de execução em segundos e o intervalo de segurança de cada valor a um nível de 95%.

É possível notar facilmente no Gráfico 37 que as médias do índice CCC estão próximas, ao contrário dos tempos de execução, onde o tempo do algoritmo genético construtivo se mostrou o maior em termos absolutos e o que apresentou a maior variância.

O Gráfico 38 traz as médias dos valores do índice CCC, os respectivos tempos de execução em segundos e o intervalo de segurança de cada valor a um nível de 95%.

Figura 37 – Gráfico de barra de erro das médias dos índices CCC e tempos de execução da base de dados autenticidade de cédula a um nível de 95%.

Figura 38 – Gráfico de barra de erro das médias dos índices CCC e tempos de execução da base de dados qualidade do vinho branco a um nível de 95%.

Figura 39 – Gráfico de barra de erro das médias dos índices CCC e tempos de execução da base de dados qualidade do ar a um nível de 95%.

É possível notar facilmente no Gráfico 38 que as médias do índice CCC estão distintas. Já os tempos de execução neste caso se apresentaram mais próximos um dos outros em comparação ao gráfico da base de dados anterior.

O Gráfico 39 traz as médias dos valores do índice CCC, os respectivos tempos de execução em segundos e o intervalo de segurança de cada valor a um nível de 95%.

Já no Gráfico 39 tanto as médias do índice CCC quanto o tempo de execução das três bases de dados estão próximos um dos outros, respectivamente.

Além dos gráficos e tabelas apresentadas acima, também calculou-se o desvio relativo para cada base de dados que foi executada nos três algoritmos genéticos. A Equação 13 traz a fórmula utilizada para o cálculo deste desvio. Já a Tabela 20 traz os valores de desvio relativo obtidos.

$$DR = \frac{(MelhorResultado - Algoritmo) * 100}{MelhorResultado} \quad (13)$$

Tabela 20 – Desvio relativo para valores do índice CCC correspondentes às três bases de dados analisadas

	AG	AGC	BRKGA
autenticidade de cédula	0,0%	2,3%	0,1%
qualidade do vinho branco	48,7%	0,0%	9,3%
qualidade do ar	3,6%	0,0%	1,5%

O desvio relativo tem como objetivo destacar a melhor solução encontrada (neste caso o desvio relativo é zero) e exibir o quão longe as outras soluções encontradas estão da melhor solução. Como exemplo, pode-se destacar a base de dados qualidade do vinho branco que apresentou melhor resultado quando foi executada no algoritmo genético construtivo (desvio relativo = 0%). Podemos perceber que a melhor solução encontrada pela abordagem tradicional está distante 48,7% da encontrada pela variação construtivo e 9,3% da variação enviesada com chave aleatória.

Um outra forma de analisar os resultados advindos das repetições feitas nas Tabelas 15, 16 e 17 é identificar qual dos algoritmos obteve a maior percentagem de sucesso. Para isso, conta-se quantas vezes um algoritmo foi melhor do que os outros no decorrer das repetições e divide-se essa quantidade pela quantidade total de repetições. A Tabela 21 traz os resultados desta análise.

Tabela 21 – Porcentagem de sucesso de cada algoritmo

	Porcentagem de sucesso
AG	23,33%
AGC	36,67%
BRKGA	40,00%

É possível observar que o algoritmo genético enviesado com chave aleatória obteve a maior percentagem de sucesso, sendo esta de 40%. Logo após, vem o algoritmo genético construtivo, com 36,67% de sucesso, seguido da abordagem tradicional com 23,33% de sucesso.

Afim de verificar se há uma real diferença entre as médias dos valores do índice CCC resultantes dos algoritmos executados foram realizados dois testes estatísticos: teste T de amostras independentes e o teste de Turkey, ambos com um nível de significância de 95%, conforme pode ser observado na Tabela 22 e nos Gráficos 40, 41 e 42. Para verificar a real diferença entre médias dos tempos de execução dos mesmos algoritmos foi executado apenas o teste T de amostras independentes, Tabela 22.

Tabela 22 – Teste T de amostras independentes para verificar a diferença de médias dos valores do índice CCC e o tempo de execução entre os algoritmos

	Índice CCC			Tempode Execução		
	AG - AGC	AG - BRKGA	AGC - BRKGA	AG - AGC	AG - BRKGA	AGC - BRKGA
Autenticidade de cédula	0,225	0,574*	0,247	0,003	0,001	0,001
Qualidade do vinho branco	0,000*	0,000*	0,060	0,111*	0,002*	0,001
Qualidade do ar	0,016	0,156	0,011	0,002	0,006	0,000*

Os valores em negrito realçam as diferenças significativas. Os valores que apresentam o asterisco não apresentam variâncias diferentes, segundo o teste de igualdade de variância de Levene, no entanto, os valores de significância apresentados na Tabela 22 levam em consideração este fato.

Figura 40 – Gráfico resultante do teste de Turkey para a base de dados autenticidade de cédula um nível de 95%.

Figura 41 – Gráfico resultante do teste de Turkey para a base de dados qualidade do vinho branco um nível de 95%.

Figura 42 – Gráfico resultante do teste de Turkey para a base de dados qualidade do ar um nível de 95%.

Na primeira base de dados, autenticidade da cédula, com quatro atributos, ambos os testes não constaram-se diferença significativa entre os valores dos índices CCC encontrados pelos três algoritmos. Já com relação ao tempo de execução, o teste T de amostras independentes apontou diferença entre todos os algoritmos, sendo que o que levou menor tempo para a execução foi a abordagem tradicional, seguida da variação enviesado com chave aleatória e do construtivo.

Nesta primeira base de dados, não observou-se o esperado, isto é, que as duas variações da abordagem tradicional (construtivo e enviesado com chave aleatória) resultasse em melhor desempenho que a abordagem original. Acredita-se que isto ocorreu pois, o tamanho do conjunto original de atributos, quatro, é pequeno, permitindo que todos os algoritmos chegassem rapidamente em um bom subconjunto de atributos, variando alguns resultados pelo fato do número de grupos ser diferente, pois é gerado aleatoriamente. Destaca-se que o termo pequeno é empregado levando em consideração apenas as bases trabalhadas, sendo assim, a convergência proposta como explicação do comportamento dos resultados pode não se apresentar em outras bases de dados do mesmo tamanho. A

discussão é baseada apenas nas bases de dados escolhidas neste trabalho. Com relação ao tempo de execução, acredita-se que a diferença entre os algoritmos se deve ao fato de que as duas variações são mais densas em operações do que a abordagem tradicional, ocasionando mais tempo despendido.

Na segunda base de dados, qualidade do vinho branco, com onze atributos, o teste T de amostras independentes constatou diferença significativa na média dos valores dos índices CCC entre a abordagem tradicional e o algoritmo genético construtivo e entre o algoritmo genético tradicional e o algoritmo genético enviesado com chave aleatória. Já o teste Turkey constatou diferença apenas entre os valores do índice do algoritmo genético enviesado com chave aleatória e o algoritmo genético construtivo. A discussão está baseada no resultado do teste de T de amostras independentes. Já com relação ao tempo de execução, o teste T de amostras independentes apontou diferença entre a abordagem tradicional e o construtivo, além do algoritmo genético construtivo e da variação enviesado com chave aleatória. O menor tempo de execução foi o do algoritmo genético com chave aleatória, 77,907 segundos, seguido da abordagem tradicional com 127,194 segundos. O algoritmo que levou mais tempo para executar os procedimentos, 169,066 segundos, foi o algoritmo genético construtivo.

Nesta segunda base de dados, observou-se o esperado, ou seja, as variações construtivo e enviesado com chave aleatória resultaram em melhor desempenho do que a abordagem tradicional. Como nesta base de dados o número de atributos é onze, maior que o da primeira, acredita-se que as duas variações conseguiram chegar em um bom subconjunto de atributos decorrentes das operações mais elaboradas que elas apresentam, a saber:

1. Algoritmo genético construtivo: população inicial definida deterministicamente, avaliação de esquemas, garantia de seleção de pelo menos um indivíduo bom para a reprodução, indivíduos compostos de bons blocos construtivos;
2. Algoritmo genético com chave aleatória: propagação de bons indivíduos ao longo das gerações, presença de indivíduos mutantes em todas as gerações em proporção predefinida, recombinação realizada alelo a alelo, operações independentes do problema e presença do decodificador.

Não houve diferença significativa entre a média dos valores do índice CCC do algoritmo genético construtivo e do algoritmo genético enviesado com chave aleatória, a um nível de significância de 95%. No entanto, caso o nível de significância fosse 90%, observaríamos a diferença significativa.

Com relação ao tempo de execução, o teste T de amostras independentes mostra que não há diferença entre a abordagem tradicional e o algoritmo genético construtivo. Apenas entre o construtivo e o algoritmo genético enviesado com chave aleatória e o algoritmo genético tradicional e o algoritmo genético enviesado com chave aleatória.

Isto se dá pelo fato de que pode acontecer na abordagem tradicional, que ele passe gerações sem modificação nos indivíduos, já que a recombinação e a mutação estão condicionadas a uma taxa, p_c e p_m , respectivamente. No algoritmo genético construtivo estas duas operações sempre ocorrem, no entanto, como a avaliação é dupla $f - g$, isto pode tomar mais tempo do algoritmo. Já no algoritmo genético enviesado com chave aleatória a avaliação é simples e as operações de recombinação e mutação ocorrem em todas as gerações, pois não estão condicionadas a nenhuma taxa.

Na terceira base de dados, qualidade do ar, com treze atributos, ambos os testes constataram diferença significativa na média dos valores dos índices CCC entre a abordagem tradicional e o algoritmo genético construtivo e entre o algoritmo genético construtivo e algoritmo genético enviesado com chave aleatória. Já com relação ao tempo de execução, o teste T de amostras independentes apontou diferença entre os três algoritmos. O menor tempo de execução foi o do algoritmo genético com chave aleatória, 7,393 segundos, seguido da abordagem tradicional com 9,732 segundos. O algoritmo que levou mais tempo para executar os procedimentos, 11,354 segundos, foi o algoritmo genético construtivo.

Nesta segunda e terceira base de dados, observou-se o esperado, ou seja, uma das variações (algoritmo genético construtivo ou algoritmo genético enviesado com chave aleatória), resultou em melhor desempenho do que a abordagem tradicional. Acredita-se que apenas o algoritmo genético construtivo alcançou tal resultado, pois em sua população inicial são definidos indivíduos com bons blocos construtivos de maneira determinística.

Além disso, a avaliação $f - g$ também contribuiu bastando no sentido de estabelecer limites de valores de índice para os novos indivíduos. Com relação ao tempo de execução, acredita-se que o algoritmo genético construtivo levou maior tempo para executar os procedimentos devido a maior sofisticação de suas operações. Também pelo fato de, muitas vezes, ao longo de inúmeras gerações apenas um indivíduo permanece na população com um d_i alto. Isto faz com que este indivíduo se arraste pelas gerações até que o α atinja um valor alto para elimina-lo.

A seleção de atributos em todos os algoritmos resultou em valores de índice CCC melhores do que os apresentados na Tabela 14. No entanto, entre os algoritmos genéticos, notou-se diferenças tanto em qualidade da solução (valor do índice CCC) quanto em eficiência (tempo de execução), dois critérios importantes ao se comparar algoritmos e utilizados neste trabalho.

Por fim, destaca-se a importância de variar o número de grupos na *string*, solução. Não é incomum trabalhos científicos fixar o número de grupos em problemas como o apresentado neste trabalho. O número fixado é o número de grupos cujo índice CCC é o de maior valor quando o conjunto de atributos é o original, ou seja, sem haver a seleção de atributos. Caso isto fosse aplicado aqui, os resultados encontrados neste trabalho não seriam aquém aos mostrados nas Tabelas 15, 16 e 17. Isto porque, a Tabela 14 aponta que o melhor grupo para a base: (i) autenticidade de cédula é 8, (ii) qualidade do vinho

branco, 2 e (iii) qualidade do ar, 3. No entanto, os melhores resultados, após a seleção de variáveis, apontam os seguintes número de grupos: (i) autenticidade de cédula é 9, (ii) qualidade do vinho branco, 3 e (iii) qualidade do ar, 8.

Conclusão

A seleção de atributos em agrupamento de dados é um procedimento presente e necessário no processo de descoberta do conhecimento. Pouca atenção é dada ao tratamento dos dados, apesar de ser uma tarefa imprescindível tanto para garantir a correta conclusão quanto para diminuir e, até viabilizar, o tempo de execução dos algoritmos de mineração de dados, como é o k-médias, utilizado neste trabalho.

A importância da seleção de atributos em agrupamento de dados ficou evidenciada neste trabalho após a emissão do índice de qualidade de agrupamento CCC para a base de dados com e sem a seleção de atributos relevantes.

Dessa forma, encontrar um algoritmo que realize a seleção de um subconjunto e encontre de forma rápida um bom subconjunto é essencial para emitir qualquer conclusão sobre os dados.

Este trabalho aplicou o algoritmo genético em sua forma tradicional, além de duas variações, algoritmo genético construtivo e algoritmo genético enviesado com chave aleatória, em três bancos de dados a fim de verificar a qualidade das soluções emitidas por eles e as respectivas eficiências.

Assim, após a modelagem dos algoritmos constatou-se que em duas das três bases de dados as variações algoritmo genético construtivo e algoritmo genético enviesado com chave aleatória apresentam melhores desempenhos em relação ao índice de qualidade em agrupamento CCC, do que a abordagem tradicional, sendo o algoritmo genético construtivo o que apresentou os melhores resultados. Notou-se que em uma base de dados, com poucos atributos (quatro), não houve diferença entre os algoritmos.

Constatou-se em todas as bases de dados que a média do tempo de execução do algoritmo genético construtivo é mais elevado, de maneira significativa, do que as outras duas abordagens. Também em todas as bases de dados, a média do tempo de execução do BRKGA foi menor, de maneira significativa, do que as outras duas abordagens.

Através da execução dos algoritmos nas bases de dados selecionadas, fica evidenciado a importância da seleção de atributos em agrupamento de dados. Verificou-se que o desempenho do algoritmo de agrupamento é superior quando há a seleção de atributos

relevantes ao problema.

Dessa forma, este trabalho compre com o objetivo proposto, isto é, a modelagem e avaliação do algoritmo genético tradicional, algoritmo genético construtivo e algoritmo genético enviesado com chave aleatória.

Trabalhos Futuros

A seleção de atributos em agrupamento de dados é um tema vasto e ainda pouco explorado na literatura. Por esta razão, ainda há muito o que se desenvolver.

Como trabalhos futuros, sugere-se que após o encontro do melhor subconjunto de atributos pelos algoritmos de busca, no caso deste trabalho, algoritmos genéticos, haja uma validação estatística do valor do índice, indicando se o índice encontrado realmente é um índice de valor alto o suficiente para apontar a presença de grupos. Uma das formas de realizar a validação estatística é através da simulação de Monte Carlo.

Sugere-se também a utilização de outros índices de qualidade de agrupamento, além do índice CCC, utilizado neste trabalho, como função de avaliação dos algoritmos genéticos.

Como trabalhos futuros também é interessante realizar a seleção de variáveis para base de dados com mais atributos visto que neste trabalho o maior número de atributos foi 13, com a base de dados qualidade do ar.

Limitações do Trabalho

Destaca-se como uma limitação o número pequeno de repetições, mas ainda sim válidas, dos algoritmos genéticos para as diferentes bases de dados.

Também destaca-se que uma limitação é o uso de bases de dados com número de atributos de no máximo 13 (base de dados qualidade do ar). Acredita-se que é importante a execução dos algoritmos genéticos em bases de dados com maior número de atributos, mesmo que em bases sintéticas, isto é, criadas por distribuição normal para atributos relevantes e distribuição uniforme para atributos não relevantes.

Apesar de nenhum trabalho de seleção de atributos em agrupamento de dados realizar a validação estatística do índice apontado como o melhor por cada algoritmo, considera-se que há uma limitação neste aspecto, ou seja, estatisticamente não é garantido que o índice CCC encontrado pelos algoritmos genéticos represente um grupo verdadeiro de dados. Para isso seria necessário realizar uma simulação de Monte Carlo para cada valor do índice e após isso apontar o valor do índice como um possível candidato encontrado pelos algoritmos genéticos. Esta limitação se dá em termos computacionais, pois a execução deste procedimento demandaria demasiado recurso computacional.

Referências Bibliográficas

AHMAD, A.; DEY, L. A k-mean clustering algorithm for mixed numeric and categorical data. **Data and Knowledge Engineering**, 2007. v. 63, n. 2, p. 503 – 527, 2007. ISSN 0169-023X. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0169023X0700050X>>.

AMORIM, R. C. de; MAKARENKOV, V. Applying subclustering and lp distance in weighted k-means with distributed centroids. **Neurocomputing**, 2016. v. 173, Part 3, p. 700 – 707, 2016. ISSN 0925-2312. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0925231215011650>>.

APPALASAMY, P. et al. Classification-based data mining approach for quality control in wine production. **Journal of Applied Sciences**, 2012. v. 47, n. 4, p. 598–601, 2012.

BAKER, J. E. Adaptive selection methods for genetic algorithms. In: **Proceedings of the 1st International Conference on Genetic Algorithms**. Hillsdale, NJ, USA: L. Erlbaum Associates Inc., 1985. p. 101–111. ISBN 0-8058-0426-9. Disponível em: <<http://dl.acm.org/citation.cfm?id=645511.657075>>.

BALL, G. H.; HALL, D. J. A clustering technique for summarizing multivariate data. **Behavioral Science**, 1967. John Wiley and Sons, Ltd., v. 12, n. 2, p. 153–155, 1967. ISSN 1099-1743. Disponível em: <<http://dx.doi.org/10.1002/bs.3830120210>>.

BARANAUSKAS, J. A. **Extração do Conhecimento por Múltiplos Indutores**. Tese (Doutorado) — Instituto de Ciências Matemáticas e Computação (ICMC), Universidade de São Paulo (USP), 2001.

BEAN, J. C. Genetic algorithms and random keys for sequencing and optimization. **Journal on Computing**, 1994. v. 6, n. 2, p. 154–160, 1994.

BEASLEY, D.; BULL, D. R.; MARTIN, R. R. An overview of genetic algorithms: Part 2, research topics. **University Computing**, 1993. v. 15, n. 4, p. 170–181, 1993.

BELLMAN, R. E. **Adaptive Control Processes**. Princeton, New Jersey, U.S.A.: Princeton University Press, 1961. 255 p.

BISHOP, C. **Pattern Recognition and Machine Learning**. [S.l.]: Springer, 2006.

BLUM, A. L.; LANGLEY, P. Selection of relevant features and examples in machine learning. **Artif. Intell.**, 1997. Elsevier Science Publishers Ltd., Essex, UK, v. 97, n. 1-2,

p. 245–271, dez. 1997. ISSN 0004-3702. Disponível em: <[http://dx.doi.org/10.1016/S0004-3702\(97\)00063-5](http://dx.doi.org/10.1016/S0004-3702(97)00063-5)>.

BOCK, H.-H. Origins and extensions of the k-means algorithm in cluster analysis. **Journal Électronique d'Histoire des Probabilités et de la Statistique [electronic only]**, 2008. Séminaire d'Histoire du Calcul des Probabilités et de la Statistique, EHESS, Paris; Laboratoire de Probabilités et Modèles Aléatoires, Université Paris VI et VII, v. 4, n. 2, p. Article 14, 18 p., electronic only—Article 14, 18 p., electronic only, 2008. Disponível em: <<http://eudml.org/doc/130880>>.

CAMILO, C. O.; SILVA, J. C. **Mineração de Dados: Conceitos, Tarefas, Métodos e Ferramentas**. [S.l.], 2009.

CANEDO, V. B.; MARONO, N. S.; BETANZOS, A. A. Recent advances and emerging challenges of feature selection in the context of big data. **Knowledge-Based Systems**, 2015. v. 86, p. 33 – 45, 2015. ISSN 0950-7051. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0950705115002002>>.

CHARRAD, M. et al. Nbclust: An r package for determining the relevant number of clusters in a data set. **Journal of Statistical Software**, 2014. v. 61, n. 6, 11 2014. ISSN 1548-7660. Disponível em: <<http://www.jstatsoft.org/v61/i06>>.

CORPORATION, I. **IBM SPSS Statistics for Windows, Version 20.0**. Armonk, NY: IBM Corp.: IBM Corporation, 2011. Disponível em: <<https://www-01.ibm.com/software/br/analytics/spss/>>.

CORTEZ, P. et al. Modeling wine preferences by data mining from physicochemical properties. **Decis. Support Syst.**, 2009. Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, v. 47, n. 4, p. 547–553, nov. 2009. ISSN 0167-9236. Disponível em: <<http://dx.doi.org/10.1016/j.dss.2009.05.016>>.

CORTEZ, P. et al. **UCI Machine Learning Repository**. 2009. Disponível em: <<http://archive.ics.uci.edu/ml>>.

CORTEZ, P. et al. Using data mining for wine quality assessment. **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**, 2009. v. 5808 LNAI, p. 66–79, 2009. Cited By 4. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-71049191103partnerID=40md5=26319d9b3274fef808770bd40a08e938>>.

COVOES, T. F. **Seleção de atributos via agrupamento**. Dissertação (Mestrado) — Instituto de Ciências Matemáticas e Computação (ICMC), Universidade de São Paulo (USP), 2010.

DASH, M. et al. Feature selection for clustering - a filter solution. In: **In Proceedings of the Second International Conference on Data Mining**. [S.l.: s.n.], 2002. p. 115–122.

DASH, M.; LIU, H. Handling large unsupervised data via dimensionality reduction. In: **1999 ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery**. [s.n.], 1999. Disponível em: <<http://www.almaden.ibm.com/cs/dmkd/papers/dash.ps>>.

DONOHO, D. L. **High-dimensional data analysis: the curses and blessings of dimensionality**. 2000. American Mathematical Society "Math Challenges of the 21st Century". Disponível em: <<http://www-stat.stanford.edu/~donoho/>>.

DUNN, J. C. Well-separated clusters and optimal fuzzy partitions. **Journal of Cybernetics**, 1974. v. 4, n. 1, p. 95–104, 1974. Disponível em: <<http://dx.doi.org/10.1080/01969727408546059>>.

DY, J. G.; BRODLEY, C. E. Visualization and interactive feature selection for unsupervised data. In: **Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. New York, NY, USA: ACM, 2000. (KDD '00), p. 360–364. ISBN 1-58113-233-6. Disponível em: <<http://doi.acm.org/10.1145/347090.347168>>.

DY, J. G.; BRODLEY, C. E. Feature selection for unsupervised learning. **J. Mach. Learn. Res.**, 2004. JMLR.org, v. 5, p. 845–889, dez. 2004. ISSN 1532-4435. Disponível em: <<http://dl.acm.org/citation.cfm?id=1005332.1016787>>.

FAST, J. **Entropy: the significance of the concept of entropy and its applications in science and technology**. McGraw-Hill, 1962. (Philips technical library). Disponível em: <<https://books.google.com.au/books?id=B5s8AAAAIAAJ>>.

FAYYAD, U. M.; PIATETSKY-SHAPIO, G.; SMYTH, P. Advances in knowledge discovery and data mining. In: FAYYAD, U. M. et al. (Ed.). Menlo Park, CA, USA: American Association for Artificial Intelligence, 1996. cap. From Data Mining to Knowledge Discovery: An Overview, p. 1–34. ISBN 0-262-56097-6. Disponível em: <<http://dl.acm.org/citation.cfm?id=257938.257942>>.

FOGEL, D. B. **System Identification through Simulated Evolution: A Machine Learning Approach to Modeling**. Ginn Press, 1991. ISBN 0536579431. Disponível em: <<http://portal.acm.org/citation.cfm?id=532104>>.

FOGEL, L.; OWENS, A.; WALSH, M. **Artificial intelligence through simulated evolution**. Chichester, WS, UK: Wiley, 1966.

FOWLKES, E.; GNANADESIKAN, R.; KETTENRING, J. Variable selection in clustering. **Journal of Classification**, 1988. Springer-Verlag, v. 5, n. 2, p. 205–228, 1988. ISSN 0176-4268. Disponível em: <<http://dx.doi.org/10.1007/BF01897164>>.

FURTADO, J. C. **Algoritmo Genético Construtivo na Otimização de Problemas Combinatoriais de Agrupamentos**. Tese (Doutorado) — Instituto Nacional De Pesquisas Espaciais (INPE), 1998.

GHAZVINI, A.; AWWALU, J.; BAKAR, A. A. Comparative analysis of algorithms in supervised classification: A case study of bank notes dataset. **International Journal of Computer Trends and Technology (IJCTT)**, 2014. v. 17, n. 1, 2014.

GILLICH, E.; LOHWEG, V. Banknote authentication. In: . [S.l.: s.n.], 2010.

GOLDBERG, D. A note on boltzmann tournament selection for genetic algorithms and population-oriented simulated annealing. **Complex Systems**, 1990. 1990.

GOLDBERG, D.; DEB, K.; KORB, B. Messy genetic algorithms: motivation, analysis, and first results. **Complex Systems**, 1989. n. 3, p. 493–530, 1989.

GOLDBERG, D. E. **Genetic Algorithms in Search, Optimization and Machine Learning**. [S.l.]: Addison-Wesley Longman Publishing Co., 1989.

GOLDBERG, D. E.; DEB, K. A comparative analysis of selection schemes used in genetic algorithms. In: **Foundations of Genetic Algorithms**. [S.l.]: Morgan Kaufmann, 1991. p. 69–93.

GONCALVES, J. A hybrid genetic algorithm-heuristic for a two-dimensional orthogonal packing problem. **European Journal of Operational Research**, 2007. v. 183, n. 3, p. 1212–1229, 2007. ISSN 03772217. Disponível em: <<http://dx.doi.org/10.1016/j.ejor.2005.11.062>>.

GONCALVES, J. A.; MENDES, J. J. M.; RESENDE, M. G. C. A hybrid genetic algorithm for the job shop scheduling problem. **European Journal of Operational Research**, 2005. v. 167, n. 1, p. 77 – 95, 2005. ISSN 0377-2217. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0377221704002656>>.

GONCALVES, J. A.; RESENDE, M. C.; MENDES, J. J. M. A biased random-key genetic algorithm with forward-backward improvement for the resource constrained project scheduling problem. **Journal of Heuristics**, 2011. Springer US, v. 17, n. 5, p. 467–486, 2011. ISSN 1381-1231. Disponível em: <<http://dx.doi.org/10.1007/s10732-010-9142-2>>.

GONCALVES, J. F.; RESENDE, M. G. An evolutionary algorithm for manufacturing cell formation. **Computer**, 2004. v. 47, p. 247 – 273, 2004. ISSN 0360-8352. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0360835204001172>>.

GONCALVES, J. F.; RESENDE, M. G. C. Biased random-key genetic algorithms for combinatorial optimization. **Journal of Heuristics**, 2011. Springer US, v. 17, n. 5, p. 487–525, 2011. ISSN 1381-1231. Disponível em: <<http://dx.doi.org/10.1007/s10732-010-9143-1>>.

GONCALVES J. ALVES, J. F.; RESENDE, M. G. A parallel multi-population genetic algorithm for a constrained two-dimensional orthogonal packing problem. **J. Comb. Optim.**, 2011. v. 22, n. 2, p. 180–201, 2011.

GREFENSTETTE, J. Optimization of control parameters for genetic algorithms. **IEEE Trans. Syst. Man Cybern.**, 1986. IEEE Press, Piscataway, NJ, USA, v. 16, n. 1, p. 122–128, jan. 1986. ISSN 0018-9472. Disponível em: <<http://dx.doi.org/10.1109/TSMC.1986.289288>>.

GREFENSTETTE, J. J. (Ed.). **Proceedings of the Second International Conference on Genetic Algorithms on Genetic Algorithms and Their Application**. Hillsdale, NJ, USA: L. Erlbaum Associates Inc., 1987. ISBN 0-8058-0158-8.

HAIR, J.; ANDERSON, R.; TATHAM, R. **Análise Multivariada de Dados**. Bookman, 2005. ISBN 9788536304823. Disponível em: <<https://books.google.com.br/books?id=LxFb5JzXdbUC>>.

HAND, D. J.; SMYTH, P.; MANNILA, H. **Principles of Data Mining**. Cambridge, MA, USA: MIT Press, 2001. ISBN 0-262-08290-X, 9780262082907.

HOLLAND, J. H. **Adaptation in Natural and Artificial Systems**. Ann Arbor, MI: University of Michigan Press, 1975. Second edition, 1992.

HRUSCHKA, E.; COVOES, T. Feature selection for cluster analysis: an approach based on the simplified silhouette criterion. In: **Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on**. [S.l.: s.n.], 2005. v. 1, p. 32–38.

HU, G. et al. Classification of wine quality with imbalanced data. In: . [s.n.], 2016. v. 2016-May, p. 1712–1717. Disponível em: <<https://www.scopus.com/inward/record-uri?eid=2-s2.0-84974625277partnerID=40md5=c5cb8186745d08e1d9db980667c8d06f>>.

HUNT, L.; JORGENSEN, M. Clustering mixed data. **Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery**, 2011. John Wiley and Sons, Inc., v. 1, n. 4, p. 352–361, 2011. ISSN 1942-4795. Disponível em: <<http://dx.doi.org/10.1002/widm.33>>.

INC, M. **Minitab 17 Statistical Software [Computer software]**. 2010. [Www.minitab.com](http://www.minitab.com).

JAIN, A. K. Data clustering: 50 years beyond k-means. **Pattern Recognition Letters**, 2010. v. 31, n. 8, p. 651 – 666, 2010. ISSN 0167-8655. Award winning papers from the 19th International Conference on Pattern Recognition (ICPR)19th International Conference in Pattern Recognition (ICPR). Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0167865509002323>>.

JAIN, A. K.; DUBES, R. C. **Algorithms for Clustering Data**. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1988. ISBN 0-13-022278-X.

JONG, K. A. D. **An Analysis of the Behavior of a Class of Genetic Adaptive Systems**. Tese (Doutorado), Ann Arbor, MI, USA, 1975. AAI7609381.

KAUFMAN, L.; ROUSSEEUW, P. J. **Finding Groups in Data: An Introduction to Cluster Analysis**. [S.l.]: John Wiley, 1990.

KIM, Y.; STREET, W. N.; MENCZER, F. Feature selection in unsupervised learning via evolutionary search. In: **Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. New York, NY, USA: ACM, 2000. (KDD '00), p. 365–369. ISBN 1-58113-233-6. Disponível em: <<http://doi.acm.org/10.1145/347090.347169>>.

KIRA, K.; RENDELL, L. A. A practical approach to feature selection. In: **Proceedings of the Ninth International Workshop on Machine Learning**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1992. (ML92), p. 249–256. ISBN 1-5586-247-X. Disponível em: <<http://dl.acm.org/citation.cfm?id=141975.142034>>.

KOHAVI, R.; JOHN, G. H. Wrappers for feature subset selection. **Artif. Intell.**, 1997. Elsevier Science Publishers Ltd., Essex, UK, v. 97, n. 1-2, p. 273–324, dez. 1997. ISSN 0004-3702. Disponível em: <[http://dx.doi.org/10.1016/S0004-3702\(97\)00043-X](http://dx.doi.org/10.1016/S0004-3702(97)00043-X)>.

KOZA, J. R. **Genetic Programming: On the Programming of Computers by Means of Natural Selection**. Cambridge, MA, USA: MIT Press, 1992. ISBN 0-262-11170-5.

- KUMAR, C.; DUDYALA, A. K. Bank note authentication using decision tree rules and machine learning techniques. In: **Computer Engineering and Applications (ICACEA), 2015 International Conference on Advances in**. [S.l.: s.n.], 2015. p. 310–314.
- LANGLEY, P. Selection of relevant features in machine learning. In: **In Proceedings of the AAAI Fall symposium on relevance**. [S.l.: AAAI Press, 1994. p. 140–144.
- LEE, H. D. **Seleção e construção de features relevantes para o aprendizado de máquina**. Dissertação (Mestrado) — Instituto de Ciências Matemáticas e Computação (ICMC), Universidade de São Paulo (USP), 2000.
- LEE, H. D. **Seleção de atributos importantes para a extração de conhecimento de bases de dados**. Tese (Doutorado) — Instituto de Ciências Matemáticas e Computação (ICMC), Universidade de São Paulo (USP), 2005.
- LEE, S.; PARK, J.; KANG, K. Assessing wine quality using a decision tree. In: **Systems Engineering (ISSE), 2015 IEEE International Symposium on**. [S.l.: s.n.], 2015. p. 176–178.
- LIKAS, A.; VLASSIS, N.; VERBEEK, J. J. The global k-means clustering algorithm. **Pattern Recognition**, 2003. v. 36, n. 2, p. 451 – 461, 2003. ISSN 0031-3203. Biometrics. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0031320302000602>>.
- LINDEN, R. **Algoritmos Genético**. 2 edição. ed. Brasport, 2008. ISBN 9788574523736. Disponível em: <<https://books.google.com.br/books?id=it0kv6UsEMEC>>.
- LIU, H.; MOTODA, H. **Feature Selection for Knowledge Discovery and Data Mining**. Norwell, MA, USA: Kluwer Academic Publishers, 1998. ISBN 079238198X.
- LIU, H.; MOTODA, H. **Computational Methods of Feature Selection**. [S.l.]: Chapman and Hall, 2007. ISBN 1584888784.
- LIU, H.-H.; ONG, C.-S. Variable selection in clustering for marketing segmentation using genetic algorithms. **Expert Systems with Applications**, 2008. v. 34, n. 1, p. 502 – 510, 2008. ISSN 0957-4174. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0957417406003010>>.
- LLETI, R. et al. Selecting variables for k-means cluster analysis by using a genetic algorithm that optimises the silhouettes. **Analytica Chimica Acta**, 2004. v. 515, n. 1, p. 87 – 100, 2004. ISSN 0003-2670. Papers presented at the 5th Colloquium Chemiometricum Mediterraneum. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0003267003016246>>.
- LLOYD, S. P. Least squares quantization in pcm. **IEEE Transactions on Information Theory**, 1982. v. 28, p. 129–137, 1982.
- LOHWEG, V. **UCI Machine Learning Repository**. 2012. Disponível em: <<http://archive.ics.uci.edu/ml>>.
- LORENA, L. A. N.; FURTADO, J. C. Constructive genetic algorithm for clustering problems. **Evolutionary Computation**, 2001. v. 9, n. 3, p. 309–328, 2001. Disponível em: <<http://dx.doi.org/10.1162/106365601750406019>>.

LORENA, L. A. N.; LOPES, F. B. A dynamic list heuristic for 2d-cutting. In: **System Modelling and Optimization**. Springer US, 1996, (The International Federation for Information Processing). p. 481–488. ISBN 978-1-4757-6671-4. Disponível em: <http://dx.doi.org/10.1007/978-0-387-34897-1_58>.

MACQUEEN, J. Some methods for classification and analysis of multivariate observations. In: **Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics**. Berkeley, Calif.: University of California Press, 1967. p. 281–297. Disponível em: <<http://projecteuclid.org/euclid.bsmmsp/1200512992>>.

MARILL, T.; GREEN, D. On the effectiveness of receptors in recognition systems. **Information Theory, IEEE Transactions on**, 1963. v. 9, n. 1, p. 11–17, Jan 1963. ISSN 0018-9448.

MATLAB. **version 8.5.0 (R2015a)**. Natick, Massachusetts: The MathWorks Inc., 2015. Disponível em: <www.mathworks.com>.

MICHALEWICZ, Z.; FOGEL, D. B. **How to Solve It: Modern Heuristics**. Enlarged 2nd. [S.l.]: Springer, 2004. Hardcover. ISBN 9783540224945.

MILLIGAN, G. An examination of the effect of six types of error perturbation on fifteen clustering algorithms. **Psychometrika**, 1980. Springer-Verlag, v. 45, n. 3, p. 325–342, 1980. ISSN 0033-3123. Disponível em: <<http://dx.doi.org/10.1007/BF02293907>>.

MILLIGAN, G.; COOPER, M. An examination of procedures for determining the number of clusters in a data set. **Psychometrika**, 1985. Springer-Verlag, v. 50, n. 2, p. 159–179, 1985. ISSN 0033-3123. Disponível em: <<http://dx.doi.org/10.1007/BF02294245>>.

MITCHELL, M. **An Introduction to Genetic Algorithms**. Cambridge, MA, USA: MIT Press, 1998. ISBN 0262631857.

MOLINA, L.; BELANCHE, L.; NEBOT, A. Feature selection algorithms: a survey and experimental evaluation. In: **Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on**. [S.l.: s.n.], 2002. p. 306–313.

MORITA, M. et al. Unsupervised feature selection using multi-objective genetic algorithms for handwritten word recognition. In: **Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on**. [S.l.: s.n.], 2003. p. 666–670.

NAGANO, M. S.; RUIZ, R.; LORENA, L. A. N. A constructive genetic algorithm for permutation flowshop scheduling. **Computers and Industrial Engineering**, 2008. v. 55, n. 1, p. 195 – 207, 2008. ISSN 0360-8352. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0360835207002811>>.

NARENDRA, P. M.; FUKUNAGA, K. A branch and bound algorithm for feature subset selection. **Computers, IEEE Transactions on**, 1977. C-26, n. 9, p. 917–922, set. 1977. ISSN 0018-9340.

NBCLUST: An R Package for Determining the Relevant Number of Clusters in a Data Set. **Journal of Statistical Software**, 2014. v. 61, n. 6, 11 2014. ISSN 1548-7660. Disponível em: <<http://www.jstatsoft.org/v61/i06>>.

OLIVEIRA, A.; LORENA, L. A constructive genetic algorithm for gate matrix layout problems. **Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on**, 2002. v. 21, n. 8, p. 969–974, Aug 2002. ISSN 0278-0070.

PEARSON, K. F. R. S. Liii. on lines and planes of closest fit to systems of points in space. **Philosophical Magazine Series 6**, 1901. v. 2, n. 11, p. 559–572, 1901. Disponível em: <<http://dx.doi.org/10.1080/14786440109462720>>.

PENA, J.; LOZANO, J.; LARRANAGA, P. An empirical comparison of four initialization methods for the k-means algorithm. **Pattern Recognition Letters**, 1999. v. 20, n. 10, p. 1027 – 1040, 1999. ISSN 0167-8655. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0167865599000690>>.

PUDIL, P.; NOVOVICOVA, J.; KITTLER, J. Floating search methods in feature selection. **Pattern Recognition Letters**, 1994. Elsevier Science Inc., New York, NY, USA, v. 15, n. 11, p. 1119–1125, nov. 1994. ISSN 0167-8655. Disponível em: <[http://dx.doi.org/10.1016/0167-8655\(94\)90127-9](http://dx.doi.org/10.1016/0167-8655(94)90127-9)>.

R Development Core Team. **R: A Language and Environment for Statistical Computing**. 3.1.1. ed. Vienna, Austria, 2008. ISBN 3-900051-07-0. Disponível em: <<http://www.R-project.org>>.

RALAMBONDRAIN, H. A conceptual version of the k-means algorithm. **Pattern Recognition Letters**, 1995. v. 16, n. 11, p. 1147 – 1157, 1995. ISSN 0167-8655. Disponível em: <<http://www.sciencedirect.com/science/article/pii/016786559500075R>>.

RECHENBERG, I. Cybernetic solution path of an experimental problem. **Royal Aircraft Establishment Library Translation.**, 1965. 1965.

RIBEIRO, G. **Melhoramentos no Algoritmo Genético Construtivo e Novas Aplicações em Problemas de Agrupamento**. Tese (Doutorado) — Instituto Nacional de Pesquisas Espaciais (INPE), 2001.

RIBEIRO, G.; LORENA, L. A. N. Constructive genetic algorithm and column generation: an application to graph coloring. In: **The fifth conference of the association of Asian-pacific Operations Research Societies**. [S.l.: s.n.], 2000.

RIBEIRO, G.; LORENA, L. A. N. Constructive genetic algorithm for machine-part cell formation. In: . [S.l.: s.n.], 2000.

RICHARDSON, S.; GREEN, P. On bayesian analysis of mixtures with an unknown number of components. **JOURNAL OF THE ROYAL STATISTICAL SOCIETY SERIES B-METHODOLOGICAL**, 1997. v. 59, p. 731–758, 1997. Disponível em: <<http://dx.doi.org/10.1111/1467-9868.00095>>.

ROSTAMI, M.; MORADI, P. A clustering based genetic algorithm for feature selection. In: **Information and Knowledge Technology (IKT), 2014 6th Conference on**. [S.l.: s.n.], 2014. p. 112–116.

RUIZ, E. et al. A biased random-key genetic algorithm for the capacitated minimum spanning tree problem. **Computers and Operations Research**, 2015. v. 57, p. 95 – 108, 2015. ISSN 0305-0548. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0305054814003153>>.

- SAEYS, Y.; INZA, L.; LARRANAGA, P. A review of feature selection techniques in bioinformatics. **Bioinformatics**, 2007. v. 23, n. 19, p. 2507–2517, 2007. Disponível em: <<http://bioinformatics.oxfordjournals.org/content/23/19/2507.abstract>>.
- SAHA, S. et al. Simultaneous feature selection and symmetry based clustering using multiobjective framework. **Applied Soft Computing**, 2015. v. 29, n. 0, p. 479 – 486, 2015. ISSN 1568-4946. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1568494614006413>>.
- SARLE, W. **SAS Technical Report A-108, Cubic Clustering Criterion**. [S.l.], 1993.
- SCRUCCA, L. Genetic algorithms for subset selection in model-based clustering. In: **Events and meetings of Italian statistical society, 45th scientific meeting of the Italian statistical society**. [S.l.: s.n.], 2010.
- SIEDLECKI, W.; SKLANSKY, J. A note on genetic algorithms for large-scale feature selection. **Pattern Recognition Letters**, 1989. v. 10, n. 5, p. 335 – 347, 1989. ISSN 0167-8655. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0167865589900378>>.
- SOKAL, R.; SNEATH, P. H. A. **Principles of Numerical Taxonomy**. London: W. H. Freeman, 1963.
- SRINIVAS, M.; PATNAIK, L. Genetic algorithms: a survey. **Computer**, 1994. v. 27, n. 6, p. 17–26, June 1994. ISSN 0018-9162.
- STEARNS, S. on selecting features for pattern classification. In: **Third International Conf. on Pattern Recognition**. [S.l.: s.n.], 1976. p. 71–75.
- STEINHAUS, H. Sur la division des corp materiels en parties. **Bull. Acad. Polon. Sci**, 1956. v. 1, p. 801–804, 1956.
- SWAPNA, C. S.; KUMAR, V.; MURTHY, J. A novel approach for feature selection. **Advances in Intelligent Systems and Computing**, 2015. v. 339, p. 877–885, 2015. Cited By 0. Disponível em: <<http://www.scopus.com/inward/record.url?eid=2-s2.0-84922021109partnerID=40md5=91b3e5f0f009ab22c3e1d844478f28b7>>.
- TSAI, C.-F.; EBERLE, W.; CHU, C.-Y. Genetic algorithms in feature and instance selection. **Knowledge-Based Systems**, 2013. v. 39, p. 240 – 247, 2013. ISSN 0950-7051. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0950705112003140>>.
- VACHKOV, G. Multistep modeling for approximation and classification by use of rbf network models. **Studies in Computational Intelligence**, 2016. v. 623, p. 325–353, 2016. Cited By 0. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-84957553094partnerID=40md5=2745532927ec8c7f09609a85b6aa52d4>>.
- VITO E. MASSERA, M. P. L. M. G. D. F. S. D. On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario. **Sensors and Actuators B: Chemical**, 2008. v. 129, n. 2, p. 750 – 757, 2008. ISSN 0925-4005. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0925400507007691>>.

VITO, S. D. **UCI Machine Learning Repository**. 2008. Disponível em: <<http://archive.ics.uci.edu/ml>>.

VITO, S. D. et al. Semi-supervised learning techniques in artificial olfaction: A novel approach to classification problems and drift counteraction. **IEEE Sensors Journal**, 2012. v. 12, n. 11, p. 3215–3224, Nov 2012. ISSN 1530-437X.

VITO, S. D. et al. On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario. **Sensors and Actuators B: Chemical**, 2008. v. 129, n. 2, p. 750 – 757, 2008. ISSN 0925-4005. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0925400507007691>>.

VITO, S. D. et al. Co, {NO₂} and {NO_x} urban pollution monitoring with on-field calibrated electronic nose by automatic bayesian regularization. **Sensors and Actuators B: Chemical**, 2009. v. 143, n. 1, p. 182 – 191, 2009. ISSN 0925-4005. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S092540050900673X>>.

WEISS, S.; INDURKHYA, N. **Predictive Data Mining: A Practical Guide**. Morgan Kaufmann Publishers, 1998. (The Morgan Kaufmann Series in Data Management Systems Series). ISBN 9781558604032. Disponível em: <<https://books.google.com.br/books?id=qslQAAAAMAAJ>>.

WHITNEY, A. W. A direct method of nonparametric measurement selection. **IEEE Trans. Comput.**, 1971. IEEE Computer Society, Washington, DC, USA, v. 20, n. 9, p. 1100–1103, set. 1971. ISSN 0018-9340. Disponível em: <<http://dx.doi.org/10.1109/T-C-1971.223410>>.

ZHAI, Y.; ONG, Y.-S.; TSANG, I. The emerging "big dimensionality". **Computational Intelligence Magazine, IEEE**, 2014. v. 9, n. 3, p. 14–26, Aug 2014. ISSN 1556-603X.

Matrizes de Correlação

Tabela 23 – Matriz de correlação da base de dados: autenticação de cédula

		VWTI	AWTI	CWTI	EWTI
VWTI	Correlação	1,000	0,264**	-0,381**	0,277**
	Significância	-	0,000	0,000	0,000
AWTI	Correlação	0,264**	1,000	-0,787**	-0,526**
	Significância	0,000	-	0,000	0,000
CWTI	Correlação	-0,381**	-0,787**	1,000	0,319**
	Significância	0,000	0,000	-	0,000
EWTI	Correlação	0,277**	-0,526**	0,319**	1,000
	Significância	0,000	0,000	0,000	-
** Correlação é significativa ao nível de 0.01 (2-caudas)					
* Correlação é significativa ao nível de 0.05 (2-caudas)					

Tabela 25 – Matriz de correlação da base de dados: qualidade do ar

	CO	PT08.S1	NMHC	C6H6	PT08.S2	NOx	PT08.S3	NO2	PT08.S4	PT08.S5	T	RH	AH			
	(GT)	(CO)	(GT)	(GT)	(NMHC)	(GT)	(NOx)	(GT)	(NO2)	(O3)						
CO	Correlação	1	0,936**	0,887**	0,973**	0,931**	0,936**	0,958**	0,951**	-0,824**	0,861**	0,940**	0,883**	0,318**	-0,105**	0,296**
(GT)	Significância	-	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,002	0,000
PT08.S1	Correlação	0,936**	1	0,782**	0,931**	0,936**	-0,830**	0,867**	0,945**	0,935**	0,325**	-0,040	0,407**			
(CO)	Significância	0,000	-	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,256	0,000			
NMHC	Correlação	0,887**	0,782**	1	0,898**	0,875**	-0,774**	0,728**	0,848**	0,762**	0,367**	-0,160**	0,282**			
(GT)	Significância	0,000	0,000	-	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000			
C6H6	Correlação	0,973**	0,931**	0,898**	1	0,985**	-0,849**	0,847**	0,961**	0,897**	0,418**	-0,178**	0,313**			
(GT)	Significância	0,000	0,000	0,000	-	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000			
PT08.S2	Correlação	0,958**	0,936**	0,875**	0,985**	1	0,927**	0,885**	0,958**	0,909**	0,446**	-0,193**	0,325**			
(NMHC)	Significância	0,000	0,000	0,000	0,000	-	0,000	0,000	0,000	0,000	0,000	0,000	0,000			
NOx	Correlação	0,951**	0,923**	0,811**	0,927**	1	-0,814**	0,857**	0,913**	0,893**	0,238**	-0,042	0,271**			
(GT)	Significância	0,000	0,000	0,000	0,000	-	0,000	0,000	0,000	0,000	0,000	0,228	0,000			
PT08.S3	Correlação	-0,824**	-0,830**	-0,774**	-0,849**	-0,911**	1	-0,815**	-0,880**	-0,858**	-0,424**	0,097**	-0,463**			
(NOx)	Significância	0,000	0,000	0,000	0,000	0,000	-	0,000	0,000	0,000	0,000	0,005	0,000			
NO2	Correlação	0,861**	0,867**	0,728**	0,847**	0,885**	-0,815**	1	0,808**	0,840**	0,407**	-0,223**	0,215**			
(GT)	Significância	0,000	0,000	0,000	0,000	0,000	0,000	-	0,000	0,000	0,000	0,000	0,000			
PT08.S4	Correlação	0,940**	0,945**	0,848**	0,961**	0,958**	-0,880**	0,808**	1	0,919**	0,344**	0,012	0,516**			
(NO2)	Significância	0,000	0,000	0,000	0,000	0,000	0,000	0,000	-	0,000	0,000	0,722	0,000			
PT08.S5	Correlação	0,883**	0,935**	0,762**	0,897**	0,909**	-0,858**	0,840**	0,919**	1	0,313**	-0,015	0,440**			
(O3)	Significância	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	-	0,000	0,677	0,000			
T	Correlação	0,318**	0,325**	0,367**	0,418**	0,446**	-0,424**	0,407**	0,344**	0,313**	1	-0,770**	0,160**			
	Significância	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	-	0,000	0,000			
RH	Correlação	-0,105**	-0,040	-0,160**	-0,178**	-0,193**	0,097**	-0,223**	0,012	-0,015	-0,770**	1	0,476**			
	Significância	0,002	0,256	0,000	0,000	0,000	0,228	0,005	0,000	0,722	0,000	-	0,000			
AH	Correlação	0,296**	0,407**	0,282**	0,313**	0,325**	0,271**	0,215**	0,516**	0,440**	0,160**	0,476**	1			
	Significância	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	-			
** Correlação é significante ao nível de 0,01 (2 caudas)																

** Correlação é significativa ao nível de 0,01 (2 caudas)