

UNIVERSIDADE FEDERAL DE SÃO CARLOS

CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**GERAÇÃO GENÉTICA MULTIOBJETIVO DE
SISTEMAS FUZZY USANDO A ABORDAGEM
ITERATIVA**

EDWARD HINOJOSA CÁRDENAS

ORIENTADORA: PROFA. DRA. HELOISA DE ARRUDA CAMARGO

São Carlos - SP
Junho/2011

UNIVERSIDADE FEDERAL DE SÃO CARLOS

CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**GERAÇÃO GENÉTICA MULTIOBJETIVO DE
SISTEMAS FUZZY USANDO A ABORDAGEM
ITERATIVA**

EDWARD HINOJOSA CÁRDENAS

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação, área de concentração: Inteligência Artificial.
Orientadora: Dra. Heloisa de Arruda Camargo

São Carlos - SP
Junho/2011

**Ficha catalográfica elaborada pelo DePT da
Biblioteca Comunitária da UFSCar**

H663gg

Hinojosa Cárdenas, Edward.

Geração genética multiobjetivo de sistemas fuzzy usando a abordagem iterativa / Edward Hinojosa Cárdenas. -- São Carlos : UFSCar, 2011.

115 f.

Dissertação (Mestrado) -- Universidade Federal de São Carlos, 2011.

1. Inteligência artificial. 2. Sistemas Fuzzy. 3. Algoritmos genéticos. 4. Sistemas Fuzzy-genético. I. Título.

CDD: 006.3 (20^a)

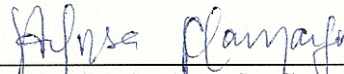
Universidade Federal de São Carlos
Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Ciência da Computação

“Geração Genética Multiobjetivo de Sistemas Fuzzy usando a Abordagem Iterativa”

EDWARD HINOJOSA CARDENAS

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação

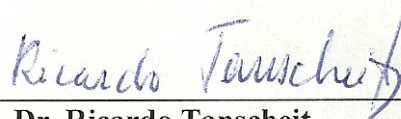
Membros da Banca:



Profa. Dra. Heloisa de Arruda Camargo
(Orientadora - DC/UFSCar)



Prof. Dr. Estevam Rafael Hruschka Júnior
(DC/UFSCar)



Prof. Dr. Ricardo Tanscheit
(PUC-Rio – Dep. de Engenharia Elétrica)

São Carlos
Junho/2011

Gostaria dedicar este trabalho a toda minha família, principalmente aos meus pais (Edwin e Reina), meu irmão (Edwin), minha namorada (Vanessa), meus sobrinhos (Dianita e Edwincito), minha cunhada (Eveline), minha tia (Luz), meus avós (Alberto e Grimaldo) e minhas avós (Doris e Aurora).

AGRADECIMENTO

Agradeço a Deus por ter me concedido sabedoria, saúde, disposição, condições espirituais e materiais para que, por sua vontade, consegui terminar este trabalho.

Aos meus pais, Edwin Grimaldo Hinojosa Aybar e Reina Fernanda Cárdenas Torres, meu irmão Edwin, minha namorada Vanessa, minha cunhada Eveline, meus sobrinhos Dianita e Edwincito, minha tia Luz, meus avôs Alberto e Grimaldo e minhas avós Doris e Aurora, pela família, amor, compreensão, que atendem com atenção e carinho meu crescimento pessoal e profissional.

À Professora Dr.^a Heloisa de Arruda Camargo por acreditar em mim e me mostrar o caminho da ciência. Agradeço pela oportunidade, orientação, compreensão, amizade, confiança e todos os ensinamentos transmitidos.

Aos meus amigos no grupo de pesquisa CIG (Computational Intelligence Group), pela grande amizade e ajuda neste trabalho.

Aos meus amigos do mestrado em Ciências da Computação turma 2009 da UFSCar, pela amizade e apoio quando cheguei ao Brasil e durante todo o mestrado.

A todos os colegas e professores da pós-graduação em Ciências da Computação da UFSCar pelo convívio e aprendizado.

A todos meus amigos que conheci no Brasil, agradeço todos os bons momentos que agente passou.

A todos meus amigos do Peru, principalmente do distrito Alto Selva Alegre, agradeço muitos anos de risos e companhia.

RESUMO

O objetivo deste trabalho é estudar, expandir e avaliar o uso dos algoritmos genéticos multiobjetivo e a abordagem iterativa na geração de sistemas *fuzzy*, mais especificamente para sistemas *fuzzy* baseados em regras, tanto na geração automática da base de regras *fuzzy* a partir de conjuntos de dados, como a otimização dos conjuntos *fuzzy*. Esse trabalho investiga o uso dos algoritmos genéticos multiobjetivo com enfoque na questão de balanceamento entre precisão e interpretabilidade, ambos considerados contraditórios entre si na representação de sistemas *fuzzy*. Com este intuito, é proposto e implementado um modelo evolutivo multiobjetivo genético composto por três etapas. Na primeira etapa são criados os conjuntos *fuzzy* uniformemente distribuídos. Na segunda etapa é tratada a geração da base de regras usando a abordagem iterativa e um algoritmo genético multiobjetivo. Por fim, na terceira etapa os conjuntos *fuzzy* criados na primeira etapa são otimizados mediante um algoritmo genético multiobjetivo. O modelo desenvolvido foi avaliado em diversos conjuntos de dados benchmark e os resultados obtidos foram comparados com outros três métodos, que geram regras de classificação, encontrados na literatura. Os resultados obtidos após a otimização dos conjuntos *fuzzy* foram comparados com resultados de outro otimizador de conjuntos *fuzzy* encontrado na literatura. Métodos estatísticos de comparação usualmente aplicados em contextos semelhantes mostram uma melhor taxa de classificação e interpretabilidade do método proposto com relação a outros métodos.

Palavras-chave: Sistemas *Fuzzy*, Algoritmos Genéticos Multiobjetivo, Precisão, Interpretabilidade, Abordagem Iterativa, Sistemas *Fuzzy* Genéticos, Geração Automática de Regras *Fuzzy*, Otimização de Conjuntos *Fuzzy*.

ABSTRACT

The goal of this work is to study, expand and evaluate the use of multi-objective genetic algorithms and the iterative rule learning approach in fuzzy system generation, especially, in fuzzy rule-based systems, both in automatic fuzzy rule generation from datasets and in fuzzy sets optimization. This work investigates the use of multi-objective genetic algorithms with a focus on the trade-off between accuracy and interpretability, considered contradictory objectives in the representation of fuzzy systems. With this purpose, we propose and implement an evolutive multi-objective genetic model composed of three stages. In the first stage uniformly distributed fuzzy sets are created. In the second stage, the rule base is generated by using an iterative rule learning approach and a multi-objective genetic algorithm. Finally the fuzzy sets created in the first stage are optimized through a multi-objective genetic algorithm. The proposed model was evaluated with a number of benchmark datasets and the results were compared to three other methods found in the literature. The results obtained with the optimization of the fuzzy sets were compared to the result of another fuzzy set optimizer found in the literature. Statistical comparison methods usually applied in similar context show that the proposed method has an improved classification rate and interpretability in comparison with the other methods.

Keywords: Fuzzy Systems, Multiobjective Genetic Algorithms, Accuracy, Interpretability, Iterative Rule Learning, Genetic Fuzzy Systems, Automatic Fuzzy Rule Base Generation, Fuzzy Set Optimization.

LISTA DE FIGURAS

Figura 2.1:	Função triangular	7
Figura 2.2:	Função trapezoidal	7
Figura 2.3:	Conjuntos <i>fuzzy</i> da variável linguística Altura	8
Figura 2.4:	Suporte e núcleo do conjunto normal <i>fuzzy</i> F	9
Figura 2.5:	Exemplos de conjuntos <i>fuzzy</i>	9
Figura 2.6:	Interseção padrão de conjuntos <i>fuzzy</i>	10
Figura 2.7:	União padrão de conjuntos <i>fuzzy</i>	10
Figura 2.8:	Complemento padrão de um conjunto <i>fuzzy</i>	10
Figura 3.1:	Procedimento de um AE – Adaptada de (JONG, 2006)	18
Figura 3.2:	Procedimento de um AG – Adaptada de (JONG, 2006)	19
Figura 3.3:	Estratégia roleta para seleção – Adaptado (CORDON; HERRERA; HOFFMANN; MAGADALENA, 2001).....	21
Figura 3.4:	Estratégia torneio para seleção – Adaptado (SANCHES, 2008)	21
Figura 3.5:	Ranking das soluções	27
Figura 3.6:	Técnica de otimização multiobjetivo NSGA-II – Adaptado (CANCINO, 2003)	32
Figura 3.7:	Distância de multidão em NSGA-II – Adaptado (CANCINO, 2003)	33
Figura 3.8:	Cálculo da distância de multidão em NSGA-II – Adaptado (CANCINO, 2003)	33
Figura 3.9:	Técnica de otimização multiobjetivo SPEA2 – Adaptado (CANCINO, 2003)	34
Figura 3.10:	Algoritmo de corte do SPEA2 – Adaptado (CANCINO, 2003)	36
Figura 4.1:	Taxonomia dos sistemas <i>fuzzy</i> genéticos baseados em regras – Adaptado (HERRERA, 2008)	39
Figura 4.2:	Curva do equilíbrio entre precisão – interpretabilidade em sistemas fuzzy baseados em regras – Adaptado (ISHIBUCHI, 2007)	43

Figura 4.3:	Fluxo do método NSLV – Adaptado (GONZALEZ; RAUL, 2008)	47
Figura 4.4:	Intervalos de variação para conjuntos <i>fuzzy</i> – Adaptado (GACTO; ALCALA; HERRERA, 2010)	48
Figura 4.5:	Exemplos de conjuntos <i>fuzzy</i> transparentes Triangulares - Adaptado (PULKKIEN; KOIVISTO, 2010)	51
Figura 5.1:	Metodologia proposta para a Geração Multiobjetivo de um Sistema <i>Fuzzy</i> (GGMOSF).....	55
Figura 5.2:	Esquema do método genético multiobjetivo iterativo proposto para a geração da BR	57
Figura 5.3:	Algoritmo do método proposto para a geração da melhor regra utilizando NSGA-II	59
Figura 5.4:	Algoritmo do métodos proposto para a geração da melhor regra utilizando SPEA2	60
Figura 5.5:	Codificação das regras	61
Figura 5.6:	Codificação das regras com condições <i>don't care</i>	61
Figura 5.7:	Algoritmo do método proposto para a otimização da BD utilizando NSGA-II	67
Figura 5.8:	Algoritmo do método proposto para a otimização da BD utilizando SPEA2	68

LISTA DE TABELAS

Tabela 3.1: Valores para uma população de 6 soluções	26
Tabela 3.2: Ranking das soluções	26
Tabela 3.3: Aptidão média das soluções	28
Tabela 3.4: Contador do nicho e aptidão compartilhada	30
Tabela 3.5: Aptidão final	30
Tabela 3.6: Valores finais para o exemplo de um AGMO	31
Tabela 6.1: Conjuntos de dados utilizados nos experimentos	72
Tabela 6.2: Parâmetros dos AGMO	73
Tabela 6.3: Resultados obtidos com o método IRL-NSGA-II	74
Tabela 6.4: Resultados obtidos com o método IRL-SPEA2	74
Tabela 6.5: Resultados obtidos com o método OM-NSGA-II	75
Tabela 6.6: Resultados obtidos com o método OM-SPEA2	75
Tabela 6.7: Resultados finais com o método proposto com NSGA-II	75
Tabela 6.8: Resultados finais com o método proposto com SPEA2	76
Tabela 6.9: Média da taxa de erro dos métodos IRL-NSGA-II, SPEA2, NSLV, NMEEF e MESDIF	77
Tabela 6.10: Média da quantidade de regras dos métodos IRL-NSGA-II, SPEA2, NSLV, NMEEF e MESDIF	77
Tabela 6.11: Média da quantidade de condições dos métodos IRL-NSGA-II, SPEA2, NSLV, NMEEF e MESDIF	77
Tabela 6.12: Posto da comparação da média da taxa de erro do método IRL-NSGA-II com os métodos IRL-SPEA2, NSLV, NMEEF e MESDIF	78
Tabela 6.13: Posto da comparação da média da taxa de erro do método IRL-SPEA2 com os métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF	79

Tabela 6.14: Posto da comparação da média da quantidade de regras do método IRL-NSGA-II com os métodos IRL-SPEA2, NSLV, NMEEF e MESDIF	79
Tabela 6.15: Posto da comparação da média da quantidade de regras do método IRL-SPEA2 com os métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF	80
Tabela 6.16: Posto da comparação da média da quantidade de condições do método IRL-NSGA-II com os métodos IRL-SPEA2, NSLV, NMEEF e MESDIF	80
Tabela 6.17: Posto da comparação da média da quantidade de condições do método IRL-SPEA2 com os métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF	81
Tabela 6.18: Média da melhora da taxa de erro dos métodos OM-NSGA-II e AG-1	82
Tabela 6.19: Média da melhora da taxa de erro dos métodos OM-SPEA2 e AG-2	83
Tabela 6.20: Média do Índice de Interpretabilidade Semântica dos métodos OM-NSGA-II e AG-1	83
Tabela 6.21: Média do índice de interpretabilidade semântica dos métodos OM-SPEA2 e AG-2	83
Tabela 6.22: Média da quantidade de conjuntos transparentes dos métodos OM-NSGA-II e AG-1	84
Tabela 6.23: Média de quantidade de conjuntos transparentes dos métodos OM-SPEA2 e AG-2	84
Tabela 6.24: Posto da comparação entre o método OM-NSGA-II e AG-1	85
Tabela 6.25: Posto da comparação entre o método OM-SPEA2 e AG-2	85

LISTA DE ABREVIATURAS E SIGLAS

SFBR – *Sistemas Fuzzy Baseados em Regras*

MI – *Mecanismo de Inferência*

BC – *Base de Conhecimento*

BD – *Base de Dados*

BR – *Base de Regras*

CD – *Conjunto de Dados*

SFGBR – *Sistemas Fuzzy Genético Baseado em Regras*

AG – *Algoritmos Genéticos*

AGMO – *Algoritmos Genéticos Multiobjetivo*

IRL – *Aprendizado Iterativo de Regras Fuzzy (Iterative Rule Learning)*

t-norma – *Normas Triangulares*

s-norma – *Co-normas Triangulares*

SFCBR – *Sistemas Fuzzy de Classificação Baseados em Regras*

NSLV – *New Structural Learning algorithm in Vague environment*

SLAVE - *Structural Learning Algorithm in Vague Environment*

AE – *Algoritmos Evolutivos*

PE – *Programação Evolutiva*

CE – *Computação Evolutiva*

EE – *Estratégias Evolutivas*

VEGA – *Vector Evaluated Genetic Algorithm*

AEMO – *Algoritmo Evolutivo Multiobjetivo*

NSGA-II – *Algoritmo Genético com Ordenação não Dominada II (Non-dominated Sorting Genetic Algorithm II)*

SPEA2 – *Algoritmo Evolutivo de Pareto Robusto 2 (Strength Pareto Evolutionary Algorithm 2)*

SF – *Sistemas Fuzzy*

SFG – *Sistemas Fuzzy Genéticos*

CF – *Conjuntos Fuzzy*

IIS – *Índice de Interpretabilidade Semântica*

GM3M – *Índice de Interpretabilidade Semântica Baseado em três medidas (Semantics-Based Interpretability Index Based on Aggregation of the Three Measures)*

MESDIF-SD – *Algoritmo evolutivo multiobjetivo para a geração de regras fuzzy descritivas de subgrupos (Multiobjective Evolutionary Subgroup Discovery Fuzzy rules)*

NMEEF-SD – *Algoritmo evolutivo multiobjetivo não dominado para a geração de regras fuzzy descritivas de subgrupos (Non-dominated Multiobjective Evolutionary Algorithm for Extracting Fuzzy Rules in Subgroup Discovery)*

IRL-NSGA-II – *Algoritmo que utiliza a abordagem IRL e o NSGA-II para gerar regras fuzzy*

IRL-SPEA2 – *Algoritmo que utiliza a abordagem IRL e o SPEA2 para gerar regras fuzzy*

OM-NSGA-II – *Otimização Multiobjetivo de conjuntos fuzzy utilizando o NSGA-II*

OM-SPEA2 – *Otimização Multiobjetivo de conjuntos fuzzy utilizando o SPEA2*

AG-1 – *Resultados da otimização genética dos conjuntos fuzzy utilizando as regras geradas pelo IRL-NSGA-II*

AG-2 – *Resultados da otimização genética dos conjuntos fuzzy utilizando as regras geradas pelo IRL-SPEA2*

SUMÁRIO

1. Capítulo 1: Introdução	1
1.1. Motivação e objetivos	2
1.2. Metodologia de desenvolvimento do trabalho	4
2. Capítulo 2: Sistemas <i>Fuzzy</i>	5
2.1. Considerações iniciais	5
2.2. Teoria de conjuntos <i>fuzzy</i>	6
2.2.1. Conceitos básicos	6
2.2.2. Conceitos relacionados aos conjuntos <i>fuzzy</i>	8
2.2.3. Operações básicas entre conjuntos <i>fuzzy</i>	9
2.2.4. Operações generalizadas sobre conjuntos <i>fuzzy</i>	11
2.3. O raciocínio aproximado	12
2.4. Sistemas de classificação <i>fuzzy</i>	12
2.4.1. Método de raciocínio <i>fuzzy</i> clássico	13
2.4.2. Método de raciocínio <i>fuzzy</i> geral	14
2.5. Considerações finais	15
3. Capítulo 3: Computação Evolutiva	16
3.1. Considerações iniciais	16
3.2. Computação evolutiva	16
3.3. Algoritmos genéticos	18
3.3.1. População inicial	19
3.3.2. Função de aptidão	20
3.3.3. Operadores genéticos	20
3.4. Otimização multiobjetivo	23

3.4.1.	Fronteira de Pareto	24
3.4.2.	Otimização multiobjetivo usando algoritmos evolutivos	24
3.5.	Otimização multiobjetivo usando algoritmos genéticos	25
3.5.1.	Algoritmos genéticos multiobjetivo	25
3.5.2.	Algoritmo Genético com Ordenação não Dominada II (Non-dominated Sorting Genetic Algorithm II – NSGA-II)	31
3.5.3.	Algoritmo Evolutivo de Pareto Robusto 2 (Strength Pareto Evolutionary Algorithm 2 – SPEA2)	34
3.6.	Considerações finais	36
4.	Capítulo 4: Sistemas <i>Fuzzy</i> Genéticos	37
4.1.	Considerações iniciais	37
4.2.	Sistemas <i>fuzzy</i> genéticos	37
4.3.	Abordagens dos sistemas <i>fuzzy</i> genéticos	38
4.3.1.	Adaptação genética	40
4.3.2.	Aprendizado genético dos componentes dos SFBR	41
4.4.	Aprendizado genético multiobjetivo de SFBR	43
4.5.	Pesquisas sobre a Interpretabilidade em SFBR	45
4.6.	Pesquisas importantes para o método proposto de geração genética multiobjetivo utilizando a abordagem iterativa	46
4.6.1.	Novo algoritmo de aprendizagem estrutural em ambientes imprecisos (New Structural Learning algorithm in Vague environment – NSLV)	47
4.6.2.	Índice de interpretabilidade semântica	48
4.6.3.	Conjuntos <i>fuzzy</i> transparentes	50
4.6.4.	Algoritmo evolutivo multiobjetivo não dominado para a geração de regras <i>fuzzy</i> na descoberta de subgrupos (Non-dominated Multiobjective Evolutionary Algorithm for Extracting Fuzzy Rules in Subgroup Discovery – NMEEF-SD)	51
4.6.5.	Algoritmo evolutivo multiobjetivo para a geração de regras <i>fuzzy</i> na descoberta de subgrupos (Multiobjective Evolutionary	

Subgroup Discovery Fuzzy rules – MESDIF-SD).....	52
4.6.6. Método de otimização genética da BD	52
4.7. Considerações finais	53
5. Capítulo 5: Geração Genética Multiobjetivo de Sistemas <i>Fuzzy</i>	54
5.1. Metodologia proposta	55
5.2. Geração genética multiobjetivo da base de regras	56
5.3. Otimização genética multiobjetivo da base de dados	66
5.4. Considerações finais	71
6. Capítulo 6: Experimentos e Análises de Resultados	72
6.1. Descrição dos conjuntos de dados	72
6.2. Experimentos realizados	73
6.2.1. Resultados para os métodos propostos de geração genética multiobjetivo da base de regras	74
6.2.2. Resultados para os métodos propostos de otimização genética multiobjetivo da base de dados	74
6.2.3. Resultados para o método genético multiobjetivo proposto para a geração de sistemas <i>fuzzy</i>	75
6.2.4. Comparação do método proposto para a geração genética multiobjetivo da base de regras	76
6.2.5. Comparação do método proposto para a otimização genética multiobjetivo da base de dados	82
6.3. Considerações finais	86
7. Capítulo 7: Conclusões	87
Apêndice A	
Apêndice B	

Capítulo 1

INTRODUÇÃO

A teoria de conjuntos *fuzzy* desenvolvida por Zadeh (1965) é aplicada em diferentes áreas como engenharia (BARDOSSY; DUKSTEIN, 1995), diagnósticos (HUDSON; COHEN, 1994), economia (FIESTRAS-JANEIRO; TABOADA; ORDOÑEZ; SAAVEDRA, 2006), manufatura (YEUNG; SHIU; TSANG, 1999), entre outras.

Uma das mais importantes aplicações da teoria de conjuntos *fuzzy* são os Sistemas *Fuzzy* Baseados em Regras (SFBR). Esses sistemas constituem uma extensão dos sistemas baseados em regras, porque operam com regras *fuzzy* em vez de regras da lógica clássica.

Os SFBR têm dois componentes: 1) O Mecanismo de Inferência (MI), que executa o processo de inferência *fuzzy* necessário para gerar uma saída do SFBR para uma entrada específica, e 2) A Base de Conhecimento (BC), que abrange uma Base de Dados (BD) onde se encontram os conjuntos *fuzzy*, e uma Base de Regras (BR) onde se encontra a coleção de regras *fuzzy*.

Uma das tarefas mais importantes e difíceis no desenvolvimento de um SFBR é a geração da BC, que pode ser realizada de forma manual, através de especialistas humanos (GAO; ZHANG, 2001) e de forma automática, fazendo uso dos dados coletados e de interesse neste trabalho.

Entre as técnicas mais bem sucedidas para tal propósito, encontram-se o uso de algoritmos de agrupamento (LIAO; CELMINS; HAMMELL, 2003), as redes neurais (NAUCK; CRUSE, 1997), os algoritmos genéticos (AG) (CORDON; HERRERA; GOMIDE; HOFFMANN; MAGDALENA, 2004) e métodos baseados em gradientes (NOMURA; HAYASHI; WAKAMI, 1992).

Neste trabalho é utilizada a técnica de AG para gerar a BC de forma automática. Este campo de pesquisa é chamado de Sistemas *Fuzzy* Genéticos Baseados em Regras (SFGBR).

Os SFGBR têm como objetivo transformar a geração ou a otimização dos componentes dos sistemas *fuzzy* em um problema de otimização que pode ser solucionado mediante a utilização de AG. Existem diferentes métodos que utilizam os AG para gerar ou otimizar diferentes componentes dos sistemas *fuzzy* (HERRERA, 2008). Este trabalho estuda dois desses métodos para gerar a BC: o método para gerar a BR (ABADEH; HABIBI, 2007) (EVSUKOFF, 2007) (GONZALEZ; RAUL, 1999) (GONZALEZ; RAUL, 2008); e o método para otimizar a BD (PIRES; CAMARGO, 2004) (CASILLAS; CORDON; JESUS; HERRERA, 2005) (ALONSO; CORDON; GUILLAUME; MAGDALENA, 2007).

A motivação para propor esse método e o objetivo deste trabalho são apresentados na seção seguinte.

1.1. Motivação e objetivos

No âmbito da geração automática da BC em SFBR, duas características são geralmente levadas em conta: a precisão, que pode ser medida, por exemplo, pela taxa de acerto, e a interpretabilidade, que pode ser medida, por exemplo, pela quantidade de regras na BR ou quantidade de condições dessas regras.

Segundo (HERRERA, 2008), obter um alto grau de interpretabilidade e uma alta precisão são objetivos contraditórios em SFBR, e, na prática, um dos objetivos prevalece sobre o outro. No entanto, uma nova tendência na comunidade científica de modelagem *fuzzy* é procurar um bom equilíbrio entre interpretabilidade e precisão, sendo esse equilíbrio, a principal motivação deste trabalho de pesquisa.

Uma das formas estudadas para tratar o problema de objetivos contraditórios na geração de SFBR é mediante os Algoritmos Genéticos Multiobjetivo (AGMO). Por meio desses algoritmos é possível representar a precisão e a interpretabilidade em duas funções de aptidão, para encontrar um equilíbrio entre eles. Essa forma de representação multiobjetivo apresenta vantagens com relação aos trabalhos que utilizam AG clássicos para geração da BC, que também tentam encontrar esse equilíbrio, combinando os objetivos contraditórios em uma única função aptidão.

Essa tarefa pode, entretanto, ser uma tarefa de difícil solução, dependendo do tipo de problema.

Têm sido desenvolvidos diferentes trabalhos que utilizam os AGMO na geração de SFBR, como, por exemplo, (MARQUEZ, MARQUEZ, PEREGRIN, 2008) (ZUSUKI; FURUHASHI; MATSUSHINA; TUSUTSUI, 1999) (YAOCHU; SENDHOFF, 2008) (GACTO; ALCALA; HERRERA, 2009) (BOTTA; LAZZERINE; MARCELLONI; STEFANESCUE, 2009) (ALCALA; DUCANGE; HERRERA; LAZZERINI; MARCELLONI, 2009) (DI NUOVO; CATANIA, 2009) (ANTONELLI; DUCANGE; MARCELLONI, 2010).

Na geração da BR mediante AGMO, este trabalho utilizada a abordagem Iterativa (*Iterative Learning Rule IRL*) para reduzir o espaço de busca (GONZALEZ; PEREZ, 2008) do AGMO.

O objetivo deste trabalho é estudar, expandir e avaliar o uso de AGMO com a abordagem IRL na geração de sistemas *fuzzy*, mais especificamente para os SFBR.

Esse trabalho investiga o uso dos AGMO com enfoque na questão do equilíbrio entre precisão e interpretabilidade, ambos considerados como objetivos contraditórios na geração de sistemas *fuzzy*.

Para tanto, foi criado um método de Geração Genética Multiobjetivo de Sistemas *Fuzzy* (GGMOSF), composto por três etapas:

1. Predefinição da Base de Dados: Criação dos conjuntos *fuzzy* uniformemente distribuídos.
2. Geração Genética Multiobjetivo da Base de Regras: Geração da BR mediante a abordagem IRL e um AGMO, utilizando a BD criada na etapa anterior.
3. Otimização Genética Multiobjetivo da Base de Dados: Otimização da BD criada na primeira etapa mediante um AGMO, utilizando a BR criada na etapa anterior.

Para avaliar o desempenho do método genético multiobjetivo proposto para a geração de sistemas *fuzzy* foram realizados diferentes testes utilizando alguns conjuntos de dados do tipo classificação obtidos no *UCI Repository of Machine Learning Databases* (FRANK; ASUNCION, 2010). Os resultados obtidos com o método proposto foram comparados com métodos de geração da BR e otimização da BD encontrados na literatura.

1.2. Organização do trabalho

Esta dissertação está organizada em 7 capítulos. O capítulo 2 trata de sistemas *fuzzy*; o capítulo 3 aborda os principais conceitos relacionados à computação evolutiva, entre eles os algoritmos genéticos, os algoritmos genéticos multiobjetivo e as técnicas de otimização multiobjetivo; o capítulo 4 trata dos sistemas *fuzzy* genéticos e apresenta um estudo sobre as principais abordagens propostas recentemente na literatura; o capítulo 5 apresenta o método proposto para a geração genética multiobjetivo de sistemas *fuzzy*; o capítulo 6 apresenta os experimentos, os resultados obtidos e a comparação do método proposto com outros métodos encontrados na literatura; o capítulo 7 apresenta as conclusões e trabalhos futuros.

Capítulo 2

SISTEMAS *FUZZY*

2.1. Considerações iniciais

Os sistemas *fuzzy* são sistemas baseados na lógica *fuzzy* e na teoria de conjuntos *fuzzy*, propostos por Lotfi A. Zadeh (1965), para resolver problemas com informação incerta e imprecisa, abundantes no mundo real.

Para resolver tais tipos de problemas, Zadeh desenvolveu uma extensão da teoria dos conjuntos clássica, definida como a teoria de conjuntos *fuzzy*, para tornar possível as operações entre conjuntos com valores lógicos entre 0 e 1 introduzindo graus de pertinência de elementos a conjuntos.

A lógica *fuzzy* é baseada na teoria dos conjuntos *fuzzy* e permite inferir conclusões a partir de informações vagas ou imprecisas definidas por regras *fuzzy*, mediante o uso de um mecanismo de raciocínio baseado no raciocínio aproximado.

Nas próximas seções são apresentados os conceitos da teoria de conjuntos *fuzzy*, raciocínio aproximado e sistemas de classificação *fuzzy*, importantes para o desenvolvimento deste trabalho. Todos esses conceitos podem ser encontrados em forma integral em (LEON, 2005), (NICOLETTI; CAMARGO, 2004) (KLIR; YUAN, 1995) e (PEDRYCZ; GOMIDE, 1998).

2.2. Teoria de conjuntos *fuzzy*

Os conjuntos *fuzzy* são usados para modelar informação vaga. De maneira simplista, a noção de conjunto *fuzzy* pode ser abordada como uma generalização da noção clássica de conjunto, que objetiva representar conjuntos cujas fronteiras não estão claras (NICOLETTI; CAMARGO, 2004).

Segundo a teoria clássica de conjuntos, um objeto pertence ou não a um conjunto, não há situação intermédia, ou seja, não existem membros parciais a um conjunto. Por exemplo, para determinar a altura de pessoas, segundo a teoria de conjuntos clássica, pode-se considerar que uma pessoa é alta a partir de uma altura, por exemplo, 1,80m, então, segundo esta classificação pode-se considerar que uma pessoa de 1,79m como não alta, classificando-a em outro conjunto, por exemplo, no conjunto das pessoas medianas.

Ao contrario da teoria de conjuntos clássica, na teoria de conjuntos *fuzzy* os elementos podem pertencer parcialmente a um ou mais conjuntos. No exemplo anterior da altura, uma pessoa com 1,79m pode pertencer com maior grau ao conjunto dos altos e com menor grau ao conjunto dos medianos, que revela melhor a realidade.

2.2.1. Conceitos básicos

Para entender melhor a teoria de conjuntos *fuzzy*, são necessários os seguintes conceitos (PEDRYCZ; GOMIDE, 1998) (NICOLETTI; CAMARGO, 2004):

- Conjunto *fuzzy*: Coleção de elementos com grau de pertinência entre 0, rejeição total do conjunto, e 1, membro pleno do conjunto (ZADEH, 1965).

O grau de pertinência é determinado por uma função de pertinência. Uma função de pertinência define um conjunto *fuzzy* e pode ser denotada por μ_F :

$$\mu_F: U \rightarrow [0,1] \quad 2.3$$

Onde U representa o universo de discurso, o qual contém todos os possíveis elementos de determinado contexto ou aplicação.

Existem diversas formas de funções de pertinência para definir os conjuntos *fuzzy*. Duas das formas mais conhecidas são a função triangular e a função

trapezoidal. Os gráficos e a expressão algébrica dessas duas funções são apresentadas a seguir:

a) Função triangular: Definida pelos parâmetros (a, m, b) , tal que $a \leq m \leq b$.

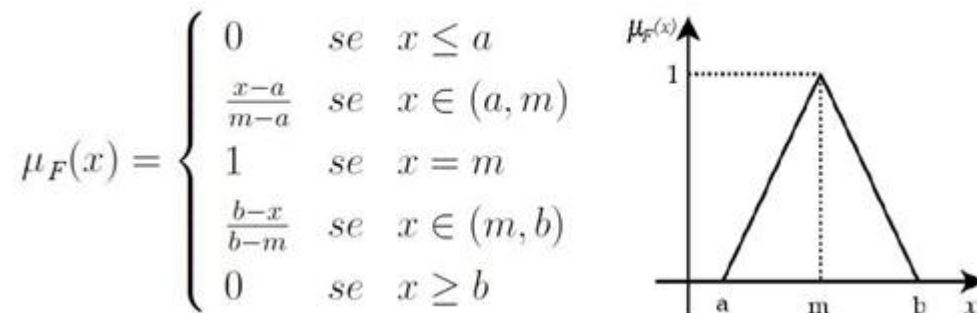


Figura 2.1: Função triangular

b) Função trapezoidal: Definida pelos parâmetros (a, m, n, b) , tal que $a \leq m < n \leq b$.

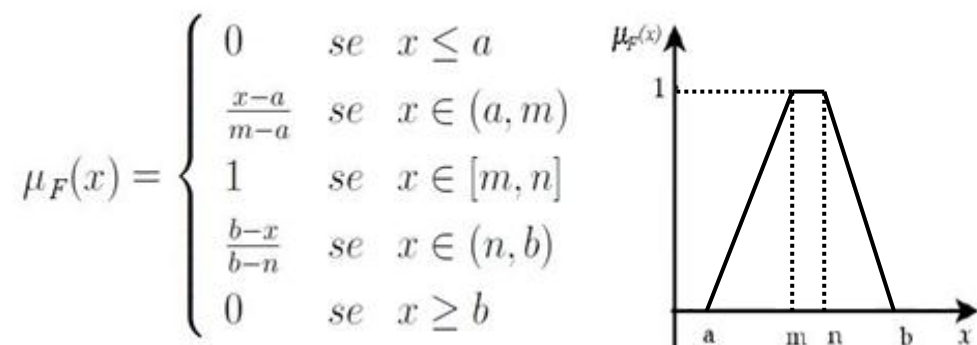


Figura 2.2: Função trapezoidal

- Variável Linguística: Uma variável linguística é uma variável cujos valores são sentenças na forma de linguagem natural, por exemplo, temperatura, altura, velocidade, distancia, etc.
- Termo Linguístico: Um termo linguístico é um rótulo ou valor de uma variável linguística o qual está associado a um conjunto *fuzzy*, por exemplo, frio, alto, rápido, longe, etc.

A Figura 2.3 mostra um exemplo dos conceitos explicados anteriormente.

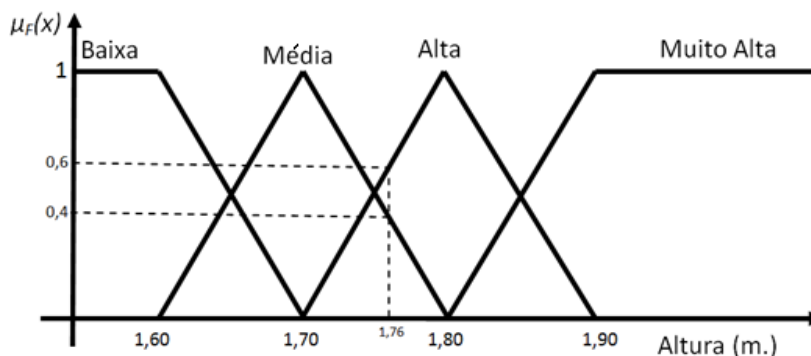


Figura 2.3: Conjuntos *fuzzy* da variável linguística *Altura*

Essa figura mostra a variável linguística *Altura* com os seus valores ou termos linguísticos (*Baixa*, *Média*, *Alta* e *Muito Alta*), cada um associado a um conjunto *fuzzy* triangular ou trapezoidal. Mostra também os graus de pertinência de uma altura de 1,76m nos conjuntos *fuzzy* triangulares *Alta* (0,6), e *Média* (0,4). O universo de discurso é definido por todos os valores reais que a variável linguística *Altura* pode assumir.

2.2.2. Conceitos relacionados aos conjuntos *fuzzy*

Abordam-se a seguir os seguintes conceitos (KLIR; YUAN, 1995):

- *Altura*: A altura, $h(F)$, de um conjunto *fuzzy* F é o maior grau de pertinência obtido por qualquer elemento que pertence a F .

$$h(F) = \sup_{x \in U} \{\mu_F(x)\} \quad 2.4$$

- *Conjunto normal*: Um conjunto *fuzzy* F é chamado normal quando $h(F) = 1$. Os conjuntos *fuzzy* não normais são chamados subnormais.
- *Suporte*: O suporte de um conjunto *fuzzy* F no conjunto universo U é o conjunto clássico que contém todos os elementos de U que têm grau de pertinência diferente de zero em F . Um exemplo é mostrado na Figura 2.4.
- *Núcleo*: O núcleo de um conjunto *fuzzy* F no conjunto universo U é o conjunto clássico que contém todos os elementos de U que têm grau de pertinência igual a 1 em F . Um exemplo é mostrado na Figura 2.4.

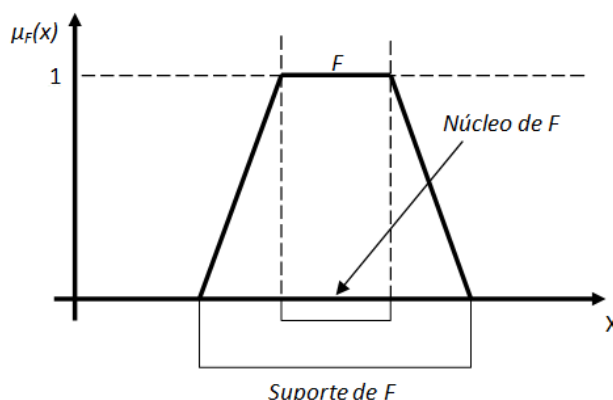


Figura 2.4: Suporte e núcleo do conjunto normal *fuzzy* F

2.2.3. Operações básicas entre conjuntos *fuzzy*

As operações básicas entre conjuntos *fuzzy* são a interseção, a união e o complemento, analogamente ao observado na teoria de conjuntos clássicos. Enquanto que o resultado de uma operação entre dois conjuntos clássicos é um novo conjunto clássico, as mesmas operações com conjuntos *fuzzy* têm como resultado um novo conjunto *fuzzy* (KLIR; YUAN, 1995).

Sejam os conjuntos *fuzzy* F e G , definidos no domínio X e mostrados na Figura 2.5:

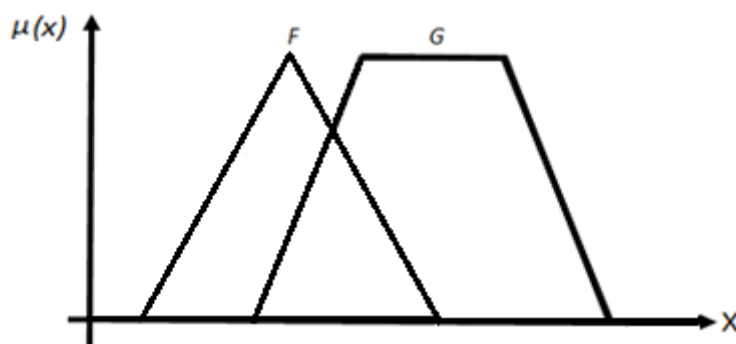


Figura 2.5: Exemplos de conjuntos *fuzzy*

A interseção padrão entre F e G é o conjunto *fuzzy* E , denotado como $E = F \cap G$, tal que para todo $x \in X$:

$$\mu_{F \cap G}(x) = \min(\mu_F(x), \mu_G(x)) \tag{2.5}$$

Um exemplo é apresentado na Figura 2.6:

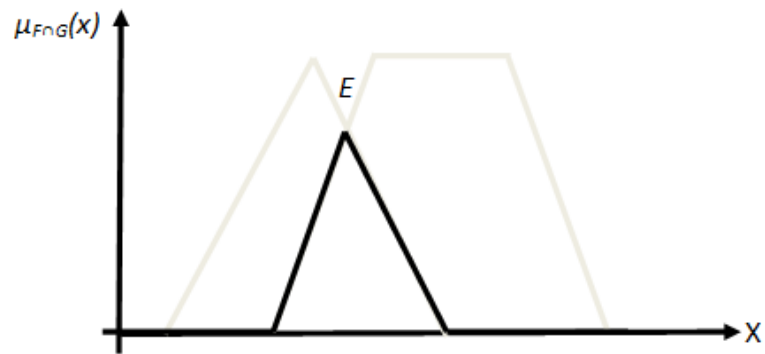


Figura 2.6: Interseção padrão de conjuntos *fuzzy*

A união padrão entre F e G é um conjunto *fuzzy* E , denotado como $E = F \cup G$, tal que para todo $x \in X$:

$$\mu_{F \cup G}(x) = \max(\mu_F(x), \mu_G(x)) \tag{2.6}$$

Um exemplo é apresentado na Figura 2.7:

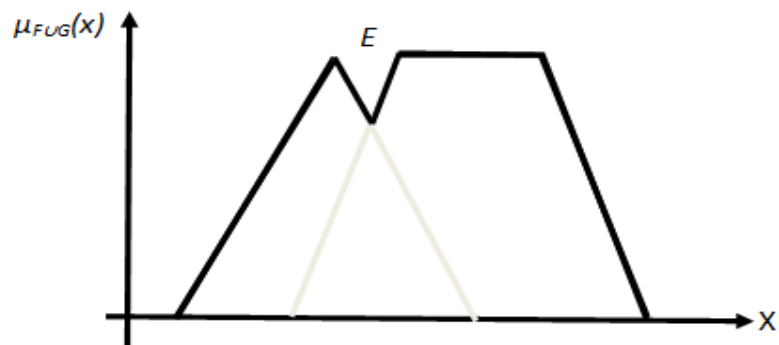


Figura 2.7: União padrão de conjuntos *fuzzy*

O complemento padrão do conjunto F é o conjunto *fuzzy* $\sim F$, tal que para todo $x \in X$:

$$\mu_{\sim F}(x) = 1 - \mu_F(x) \tag{2.7}$$

Um exemplo é apresentado na Figura 2.8:

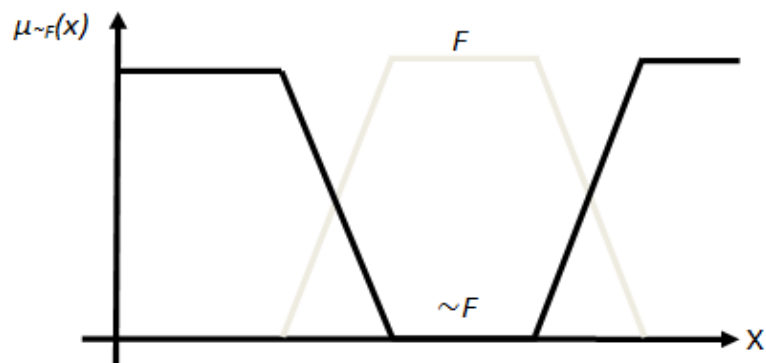


Figura 2.8: Complemento padrão de um conjunto *fuzzy*

2.2.4. Operações generalizadas sobre conjuntos fuzzy

As operações generalizadas são as operações entre conjuntos *fuzzy* (união, interseção e união) que assumem formas diferentes das operações padrão. No caso da interseção e união, são operações que utilizam outros operadores em substituição ao mínimo e máximo respectivamente. Esses operadores pertencem a categorias denominadas genericamente por normas triangulares, as quais garantem que propriedades de operações entre conjuntos serão satisfeitas. Formalmente essas operações podem ser definidas como (PEDRYCZ; GOMIDE, 1998):

- Norma triangular ou t-norma (utilizada para a interseção de conjuntos *fuzzy*): Uma t-norma t é uma operação binária $t: [0,1]^2 \rightarrow [0,1]$, que satisfaz os seguintes axiomas, para toda $a, b, c \in [0,1]$.

$$t(a, 1) = a \quad (\text{condição limite}) \quad 2.8$$

$$b \leq c \text{ implica que } t(a, b) \leq t(a, c) \quad (\text{mono tonicidade}) \quad 2.9$$

$$t(a, b) = t(b, a) \quad (\text{comutatividade}) \quad 2.10$$

$$t(a, t(b, c)) = t(t(a, b), c) \quad (\text{associatividade}) \quad 2.11$$

Exemplos:

$$1. \text{ Intersecção padrão: } a \mathbf{t}_1 b = \min \{a, b\} \quad 2.12$$

$$2. \text{ Produto algébrico: } a \mathbf{t}_2 b = ab \quad 2.13$$

$$3. \text{ Diferença limitada: } a \mathbf{t}_3 b = \max \{0, a + b - 1\} \quad 2.14$$

- Co-norma triangular ou s-norma (utilizada para a união de conjuntos *fuzzy*): Uma s-norma s é uma operação binária $s: [0,1]^2 \rightarrow [0,1]$, que satisfaz os seguintes axiomas, para toda $a, b, c \in [0,1]$.

$$s(a, 0) = a \quad (\text{condição limite}) \quad 2.15$$

$$b \leq c \text{ implica que } s(a, b) \leq s(a, c) \quad (\text{mono tonicidade}) \quad 2.16$$

$$s(a, b) = s(b, a) \quad (\text{comutatividade}) \quad 2.17$$

$$s(a, s(b, c)) = s(s(a, b), c) \quad (\text{associatividade}) \quad 2.18$$

Exemplos:

$$1. \text{ União padrão: } a \mathbf{s}_1 b = \max \{a, b\} \quad 2.19$$

$$2. \text{ Soma algébrica: } a \mathbf{s}_2 b = a + b - ab \quad 2.20$$

$$3. \text{ Soma limitada: } a \mathbf{s}_3 b = \min \{0, a + b\} \quad 2.21$$

2.3. O raciocínio aproximado

O raciocínio aproximado trata de inferir conclusões a partir de um conjunto formado de uma ou mais proposições condicionais vagas, de formato “se A então B”, e fatos conhecidos.

As proposições condicionais vagas são definidas como regras *fuzzy*, sendo A o antecedente e B o conseqüente. A e B podem ser qualquer proposição, simples ou composta. As regras *fuzzy* modelam sentenças em linguagem natural através dos conceitos de variáveis linguísticas e termos linguísticos; por exemplo, a regra “se o preço é baixo então o consumo é grande” ou a regra “se o preço é alto então o consumo é pequeno”.

O mecanismo utilizado para inferir uma conclusão é a implicação, que consiste em definir uma relação *fuzzy* entre os graus de verdade ou compatibilidade de cada regra do conjunto de regras *fuzzy* com um fato específico. Por exemplo, se são consideradas as duas regras *fuzzy* mencionadas anteriormente e um fato conhecido como preço Y, para a primeira regra é calculado o grau de pertinência do preço Y ao conjunto baixo e para segunda regra é calculado o grau de pertinência do preço Y ao conjunto alto. Esses dois graus de pertinência, neste exemplo, são considerados como o grau de verdade de cada regra e formam a relação *fuzzy* utilizada para inferir uma solução.

Existem diferentes métodos para inferir conclusões por meio de regras. Na seção seguinte são abordados dois métodos específicos para classificação, de interesse para este trabalho. O método de raciocínio *fuzzy* clássico e o método de raciocínio *fuzzy* geral.

2.4. Sistemas de classificação *fuzzy*

Classificação é uma importante tarefa encontrada nas áreas de reconhecimento de padrões, tomada de decisão, mineração de dados e modelagem. Dado um conjunto de objetos $E = \{e_1, e_2, \dots, e_p\}$, também chamados de exemplos, o objetivo da classificação é atribuir uma classe C_j de um conjunto de classes

$C = \{C_1, C_2, \dots, C_m\}$ a um objeto e_q , $e_q = (a_{q1}, a_{q2}, \dots, a_{qn})$, descrito por n atributos.

Muitos métodos têm sido utilizados para a tarefa de classificação; entre eles destacam-se os estatísticos (DUDA; HART, 1973), as redes neurais (BISHOP, 1995), o aprendizado de máquina (CASALE; RUSSO; SCEBBA; SERRANO, 2008) e os sistemas *fuzzy* (KECMAN, 2001). Neste trabalho é investigado o uso de Sistemas *Fuzzy* Baseados em Regras (SFBR) para resolver problemas de classificação, ou seja, Sistemas de Classificação *Fuzzy* Baseado em Regras (SCFBR).

Um SCFBR é um SFBR cujas regras foram projetadas para resolver um problema de classificação.

Uma típica regra *fuzzy* de classificação pode ser expressa por:

$$R_i: \mathbf{SE} X_1 \text{ é } A_{1l_1} \mathbf{E} X_2 \text{ é } A_{2l_2} \mathbf{E} \dots \mathbf{E} X_n \text{ é } A_{nl_n} \mathbf{ENTÃO} \text{ Classe} = C_j$$

no qual R_i é o identificador da regra, X_1, \dots, X_n são os atributos do exemplo considerado no problema, $A_{1l_1}, \dots, A_{nl_n}$ são os valores linguísticos usados para representar os valores de tais atributos e C_j é a classe a que pertence o exemplo.

O mecanismo de inferência aplica o conjunto de regras *fuzzy* no exemplo a ser classificado, determinando a classe a que ele pertence. A maioria dos SFCBR utiliza o método de raciocínio *fuzzy* clássico, também chamado de método da regra vencedora (CHI; YAN; PHAM, 1996) (GONZALEZ; PEREZ, 1999), que classifica um exemplo usando a regra que possui o maior grau de compatibilidade com esse exemplo. Já o método de raciocínio *fuzzy* geral agrega os graus de compatibilidade entre as regras da BR e o exemplo, considerando as regras que tenham a mesma classe consequente. Desta forma um exemplo é classificado de acordo com a classe com maior grau de compatibilidade com relação a ele, considerando todas as regras de mesma classe. A seguir são descritos os métodos de raciocínio *fuzzy* clássico e geral.

2.4.1. Método de raciocínio *fuzzy* clássico

Seja $e_q = (a_{q1}, a_{q2}, \dots, a_{qn})$ um exemplo a ser classificado e $\{R_1, R_2, \dots, R_s\}$, o conjunto de regras *fuzzy* para um SFCBR, cada uma com n antecedentes. Seja $A_{kl_k}(a_{qk})$, $k = 1, \dots, n$, o grau de pertinência do valor do antecedente a_{qk} no

k – ésimo conjunto fuzzy da regra fuzzy R_v . O método de raciocínio fuzzy clássico aplica os seguintes passos para classificar o exemplo e_q :

1. Calcular o grau de compatibilidade entre o exemplo e_q e cada regra R_v , para $v = 1, \dots, s$.

$$Compat(R_v, e_q) = t(A_{1l_1}(a_{q1}), A_{2l_2}(a_{q2}), \dots, A_{nl_n}(a_{qn})) \quad 2.22$$

na qual t denota uma t-norma.

2. Encontrar a regra R_{vmax} com o maior grau de compatibilidade com o exemplo:

$$R_{vmax} = \max\{Compat(R_v, e_q)\}, v = 1, 2, \dots, s \quad 2.23$$

3. Atribuir a classe C_j ao exemplo e_q , tal que C_j é a classe predita pela regra R_{vmax} encontrada no passo anterior.

2.4.2. Método de raciocínio fuzzy geral

Seja $e_q = (a_{q1}, a_{q2}, \dots, a_{qn})$ um exemplo a ser classificado e $\{R_1, R_2, \dots, R_s\}$, o conjunto de regras fuzzy para um SFCBR, cada uma com n antecedentes. Seja $A_{kl_k}(a_{qk}), k = 1, \dots, n$, o grau de pertinência do valor do antecedente a_{pi} no k – ésimo conjunto fuzzy da regra fuzzy R_v . O método de raciocínio fuzzy geral segue os seguintes passos para classificar o exemplo e_q :

1. Calcular o grau de compatibilidade entre o padrão e_q e cada regra R_v , para $v = 1, \dots, s$.

$$Compat(R_v, e_q) = t(A_{1l_1}(a_{q1}), A_{2l_2}(a_{q2}), \dots, A_{nl_n}(a_{qn})) \quad 2.24$$

na qual t denota uma t-norma.

2. Para cada classe C , calcular o valor de $Classe_C$, ou seja, o grau de classificação do exemplo na classe, agregando os graus de associação do passo anterior de todas as regras cuja classe predita é C :

$$Classe_C = f\{Compat(R_v, e_q) | C \text{ é a classe de } R_v\} \quad 2.25$$

Sendo f um operador de agregação tal que $min \leq f \leq max$.

3. A classe C_j será atribuída ao exemplo e_q , sendo C_j a classe de *Classe_C* que possui o maior valor de f encontrado no passo anterior.

2.5. Considerações finais

Neste capítulo foram abordados os conceitos básicos da teoria de conjuntos *fuzzy*, raciocínio aproximado e sistemas *fuzzy* de classificação. Como a teoria de sistemas *fuzzy* é muito ampla, este capítulo abordou apenas os principais conceitos necessários à compressão deste trabalho, assim como os sistemas *fuzzy* de classificação baseados em regras. O capítulo seguinte destaca os principais fundamentos da computação evolutiva.

Capítulo 3

COMPUTAÇÃO EVOLUTIVA

3.1. Considerações iniciais

Neste capítulo é apresentada a computação evolutiva e suas diferentes abordagens, já que uma dessas abordagens, a otimização genética multiobjetivo, é utilizada para realizar a geração de sistema *fuzzy* de classificação.

A seguir são descritos alguns conceitos sobre computação evolutiva, algoritmos genéticos, otimização multiobjetivo, algoritmos genéticos multiobjetivo, e duas técnicas de otimização genética multiobjetivo: o NSGA-II e o SPEA2.

3.2. Computação evolutiva

A Computação Evolutiva (CE) é um nome genérico para um conjunto de técnicas baseadas nos princípios da evolução biológica, como a seleção natural e herança genética, para resolver diferentes tipos de problemas. Essas técnicas estão sendo cada vez mais utilizadas para resolver uma grande variedade de problemas, que vão desde as aplicações práticas na indústria e comércio até a investigação científica (EIBEN; SMITH, 2007).

A base da CE foi o reconhecimento de que os seres vivos são capazes de se adaptar para sobreviver em ambientes diversos, ou seja, podem resolver problemas de forma versátil. Para essa adaptação, a natureza utiliza, entre outros, o mecanismo de seleção natural formulado por Darwin (1959), segundo o qual os indivíduos mais aptos tendem a sobreviver e reproduzir, enquanto os menos aptos tendem a desaparecer. Os Algoritmos Evolutivos (AE) são técnicas que se baseiam

nos princípios da computação evolutiva e podem ser utilizados como métodos de busca robustos para resolver problemas complexos.

Os algoritmos baseados em processo naturais surgiram no começo da década de cinquenta, quando se utilizaram computadores para simular sistemas biológicos. No entanto, foi o trabalho realizado na Universidade de Michigan sob a direção de John Holland (1975) que deu origem ao primeiro tipo de AE, os Algoritmos Genéticos (AG). Holland escreveu a principal monografia sobre o tema, *Adaptation in Natural and Artificial Systems*.

Segundo (EIBEN, SMITH, 2007) o princípio básico dos AE é: dada uma população de indivíduos, pressões do ambiente desencadeiam um processo de seleção natural, que favorece os indivíduos mais aptos e que causa uma melhoria na adequação da população. Os indivíduos de uma população representam soluções candidatas. A adequação de um indivíduo é medida por uma função de aptidão, que deve ser otimizada (maximizada ou minimizada) durante o processo evolutivo.

Com base no valor de aptidão de cada indivíduo, alguns dos melhores indivíduos (melhores soluções) são selecionados para dar origem a uma nova população pela aplicação de operadores de cruzamento e/ou mutação. O cruzamento é um operador aplicado a duas ou mais soluções candidatas (chamadas pais) e resulta em duas ou mais novas soluções (chamadas descendentes ou filhos). A mutação é aplicada em uma solução a fim de gerar outra. Ao final desse processo, as novas candidatas (descendentes) competem com as candidatas da geração anterior, com base novamente na aptidão, para assumir um lugar na nova população. Esse processo é repetido até que uma solução suficientemente qualificada seja encontrada ou até que um número máximo de iterações (chamadas de gerações) seja atingido (JONG, 2006). A Figura 3.1 descreve o procedimento de um AE.

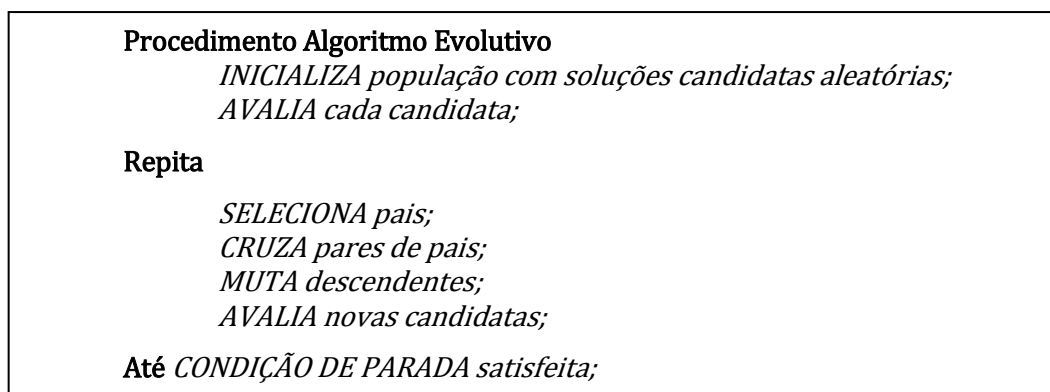


Figura 3.1: Procedimento de um AE – Adaptada de (JONG, 2006)

Segundo (JONG, 2006), três técnicas de AE foram desenvolvidas de forma independente: a Programação Evolutiva (PE), as Estratégias Evolutivas (EE) e os Algoritmos Genéticos (AG). As três utilizam o mesmo princípio básico dos AE explicado anteriormente, mas operam de forma distinta. A PE utiliza a predição de comportamento de máquinas de estado finitos para resolver qualquer tipo de problema; a seleção de indivíduos é baseada no elitismo. A ideia principal nas EE é a combinação dos indivíduos, mediante cruzamento, para gerar um descendente que substituirá o pior indivíduo da população. Os AG são algoritmos que modelam uma solução para um problema específico em uma estrutura de dados como a de um cromossomo e aplicam operadores que recombina esses cromossomos preservando informações críticas.

Dado a sua importância para a realização deste trabalho, os AG serão apresentados com mais detalhes na próxima seção. A PE e as EE não serão estudadas nesta proposta, mas informações mais detalhadas a seu respeito podem ser encontradas em (BACK; SCHWEFEL, 1993) (JONG, 2006) (EIBEN; SMITH, 2007).

3.3. Algoritmos genéticos

Os Algoritmos Genéticos (AG) são técnicas que simulam o comportamento genético da evolução das espécies e principalmente são utilizados para resolver problemas de otimização (HOLLAND, 1975).

Como mencionado anteriormente, os AG são uma técnica de AE que se diferencia das outras técnicas principalmente pelas seguintes características: as soluções são representadas por cromossomos ou vetores de tamanho fixo, que são divididos em genes; não possuem um mecanismo de auto adaptação nos

operadores de cruzamento e mutação; a seleção é probabilística e preservativa e não determinística; define o cruzamento como operador genético principal e a mutação como operador genético secundário (COELHO; COELHO, 1999).

O funcionamento de um AG é o mesmo de um AE levando em conta as características definidas anteriormente, ou seja, primeiramente um conjunto de soluções (cromossomos) é gerado de forma aleatória; Em seguida, as soluções são avaliadas por uma função de aptidão e, com base no valor de aptidão obtido para cada solução, algumas das melhores soluções são selecionadas mediante um operador de seleção para dar origem a uma nova população pela aplicação de operadores de cruzamento e mutação. Após o operador de mutação ser executado, o processo de geração de uma nova população é concluído. A avaliação de soluções e a aplicação dos operadores são repetidas até uma condição limite. No final, o melhor cromossomo representa a solução do problema. A Figura 3.2 mostra este processo.

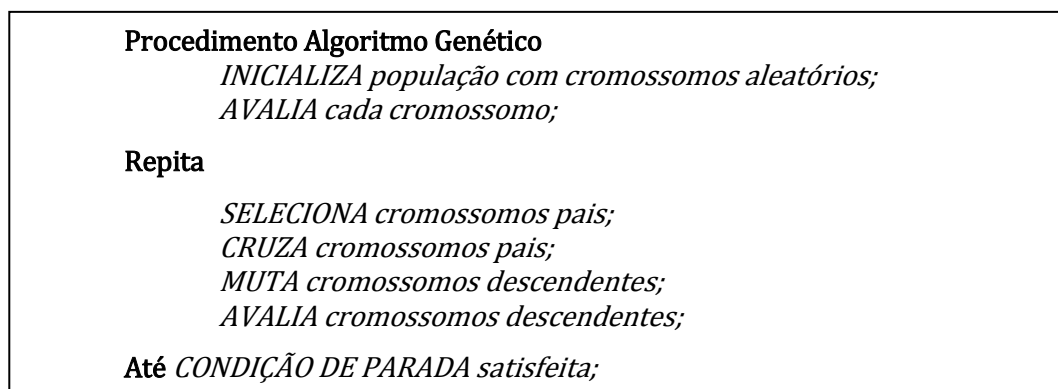


Figura 3.2: Procedimento de um AG – Adaptada de (JONG, 2006)

Para melhor entender este procedimento, a seguir são apresentados os conceitos de população inicial, função de aptidão e os operadores genéticos (seleção, cruzamento e mutação).

3.3.1. População inicial

Uma população em um AG é formada por um conjunto de m soluções ou cromossomos, onde o número m (tamanho da população) é um parâmetro de entrada do AG. Desta forma, uma geração determinada t pode representar uma população $P(t)$ como:

$$P(t) = \{X_1^t, \dots, X_m^t\}$$

onde X_m^t define a m -ésima solução na população t . Assim, uma população inicial pode ser representada como:

$$P(0) = \{X_1^0, \dots, X_m^0\}$$

3.3.2. Função de aptidão

O valor de aptidão de cada indivíduo da população é calculado mediante uma determinada função. Este é o componente mais importante de qualquer AG. É por meio desta função que se mede o quão próximo um indivíduo está da solução desejada, ou quão boa é a solução.

É essencial que esta função seja muito representativa e diferencie, na proporção correta, as más soluções das boas. Se houver pouca precisão na avaliação, uma solução ótima pode ser posta de lado durante a execução do algoritmo, além de se gastar mais tempo explorando soluções pouco promissoras (CINTRA; CAMARGO, 2004).

3.3.3. Operadores genéticos

Os operadores genéticos são utilizados para gerar uma nova população a partir da transformação da população anterior. O objetivo desta transformação é guiar a população de uma forma probabilística em direção a uma solução ótima do problema, de forma que as soluções da nova população se encontrem, em conjunto, mais 'perto' da solução ótima do problema que as soluções da população anterior.

A transformação se realiza no esquema básico de funcionamento, por meio da aplicação dos operadores genéticos de seleção, cruzamento e mutação, explicados a seguir:

- **Seleção:** A seleção está relacionada com o valor de aptidão de cada cromossomo ou solução. A estratégia da roleta foi a primeira estratégia para a seleção de cromossomos. A probabilidade de seleção de um cromossomo é diretamente proporcional ao seu valor de aptidão. A Figura 3.3 mostra um exemplo da seleção por roleta, onde o cromossomo ou solução C1 tem um valor de aptidão de 0,87 e uma probabilidade de seleção de 29% (0,29), o cromossomo C2 tem um valor de aptidão de 0,57 e uma probabilidade de seleção de 19% (0,19), o cromossomo C3 tem um valor de aptidão de 1,17 e

uma probabilidade de seleção de 39% (0,39) e o cromossomo C4 tem um valor de aptidão de 0,39 e uma probabilidade de seleção de 13% (0,13). Depois de ‘girar’ a roleta, a probabilidade de que o cromossomo C3 (39%) seja escolhido é maior do que a probabilidade de que o cromossomo C4 (13%) seja escolhido. Por isso, depois de ‘girar’ quatro vezes a roleta no exemplo da Figura 3.3, a população selecionada pode conter duas vezes o cromossomo C3 e não conter o cromossomo C4 .

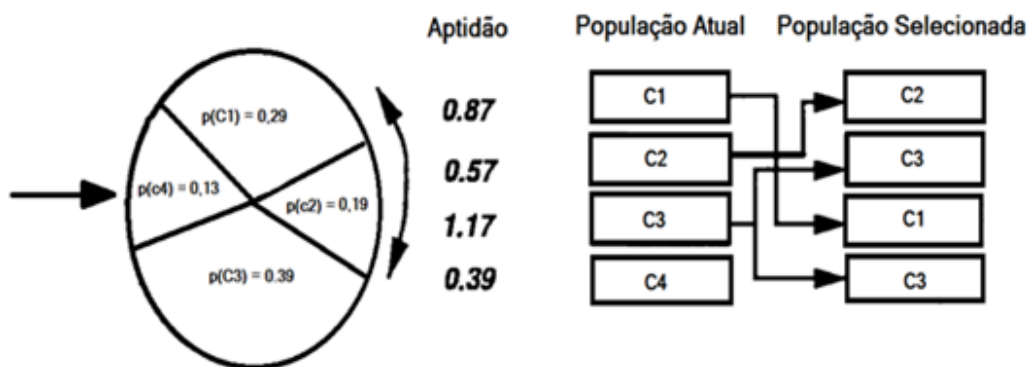


Figura 3.3: Estratégia roleta para seleção – Adaptado (CORDON; HERRERA; HOFFMANN; MAGADALENA, 2001)

Uma estratégia alternativa e também conhecida é chamada de torneio, onde um conjunto de m cromossomos é escolhido aleatoriamente e avaliado. O melhor cromossomo desse subgrupo é selecionado. A Figura 3.4 mostra um exemplo da seleção por torneio, onde de uma população de 5 cromossomos, C1, C2, C3, C4 e C5, são escolhidos aleatoriamente três cromossomos, sendo selecionado o cromossomo com maior valor de aptidão. Esse processo é repetido até se selecionar a quantidade de cromossomos que contém a nova população.

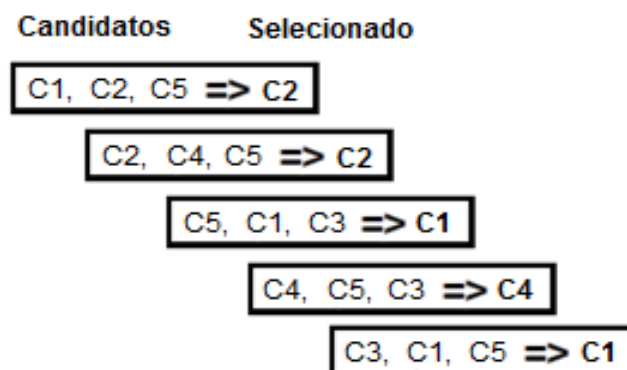


Figura 3.4: Estratégia torneio para seleção – Adaptado (SANCHES, 2008)

- Cruzamento: O cruzamento é o operador responsável pela recombinação de características dos cromossomos pais durante a reprodução, permitindo que as

próximas gerações herdem essas características. Ele é considerado o operador genético principal; por isso a taxa de cruzamento, parâmetro de entrada do AG, deve ser maior que a taxa de mutação (CORDON; HERRERA; HOFFMANN; MAGADALENA, 2001).

O cruzamento consiste em simplesmente substituir alguns dos genes de um cromossomo pai pelos genes correspondentes do outro cromossomo pai. Considere-se, por exemplo, os cromossomos (soluções) A e B com contém 7 variáveis (genes):

$$(A_1, A_2, A_3, A_4, A_5, A_6, A_7) \text{ e } (B_1, B_2, B_3, B_4, B_5, B_6, B_7)$$

Ao aplicar um tipo de cruzamento chamado de ponto único, um ponto de cruzamento é escolhido aleatoriamente a partir de 1 até 6, e duas novas soluções são produzidas pela combinação das duas anteriores. Por exemplo, se o ponto aleatório fosse 2, e o ponto único de cruzamento fosse aplicado em A e em B , o resultado seria:

$$(A_1, A_2, B_3, B_4, B_5, B_6, B_7) \text{ e } (B_1, B_2, A_3, A_4, A_5, A_6, A_7)$$

Um procedimento similar é realizado quando existe mais de um ponto de cruzamento.

Outra alternativa é o cruzamento uniforme, que consiste em gerar cada gene dos descendentes copiando o correspondente gene de um dos pais, onde este gene é escolhido de acordo com uma máscara de cruzamento gerada aleatoriamente. No processo de geração dos cromossomos, percorre-se todas as posições da máscara analisando seus valores. Quando o valor da posição corrente for 1, o gene do primeiro pai da mesma posição da máscara é copiado no primeiro descendente e o gene do segundo pai da mesma posição é copiado no segundo descendente, caso o valor da máscara for 0 a copia dos genes é contrária. Considere-se, por exemplo, os cromossomos A e B :

$$(A_1, A_2, A_3, A_4, A_5, A_6, A_7) \text{ e } (B_1, B_2, B_3, B_4, B_5, B_6, B_7)$$

e a máscara:

$$(0,1,1,0,0,0,1)$$

o resultado do cruzamento seria:

$$(B_1, A_2, A_3, B_4, B_5, B_6, A_7) \text{ e } (A_1, B_2, B_3, A_4, A_5, A_6, B_7)$$

- **Mutação:** O operador de mutação é necessário para a introdução e manutenção da diversidade genética da população (CORDON; HERRERA; HOFFMANN; MAGADALENA, 2001). A mutação consiste em selecionar um gene de um cromossomo e trocar o seu valor. Por exemplo, se uma mutação simples é aplicada no gene 2 do cromossomo $A = (0,0,1,1,0,1,0)$, um novo cromossomo $A' = (0,1,1,1,0,1,0)$ é criado.

A mutação também pode ser aplicada a mais de um gene. Se for aplicada nos genes 2 e 6 de A , um novo cromossomo $A'' = (0,1,1,1,0,0,0)$ é criado.

A mutação assegura que a probabilidade de se chegar a qualquer ponto do espaço de busca nunca será zero, além de contornar o problema de mínimos locais, pois com este mecanismo, altera-se levemente a direção da busca. O operador de mutação é aplicado aos indivíduos com uma probabilidade dada pela taxa de mutação, parâmetro de entrada do AG, que geralmente é pequena, menor do que a taxa cruzamento, pois é um operador genético secundário.

3.4. Otimização multiobjetivo

Segundo (ABRAHAM; JAIN; GOLDBERG, 2004), mesmo que alguns problemas do mundo real possam ser reduzidos a um único objetivo, muitas vezes é difícil resumir todas as características do problema em um único objetivo e resolvê-lo utilizando técnicas de otimização mono objetivo ou simples, como, por exemplo, os AG. Assim, definir múltiplos objetivos frequentemente é a melhor forma de resolver um problema. A otimização multiobjetivo tem sido utilizada satisfatoriamente por mais de duas décadas, e sua aplicação em problemas reais está aumentando.

Na otimização simples, o espaço de busca geralmente é bem definido. Mas quando o problema tem múltiplos objetivos, possivelmente contraditórios entre si, não existe mais uma única solução ótima, mas sim um conjunto de possíveis soluções de qualidade equivalente. Por exemplo, na fabricação de um produto tenta-se minimizar o custo e maximizar a qualidade, sendo que ambos os objetivos são contraditórios. Neste tipo de problema, para se obter uma solução ótima, é preciso encontrar uma solução ou um conjunto de soluções que apresentem o melhor equilíbrio entre os objetivos contraditórios. Essa solução ou conjunto de soluções é denominada fronteira de Pareto.

3.4.1. Fronteira de Pareto

Também chamada de ótimo de Pareto, foi desenvolvida por Vilfredo Pareto (PARETO, 1896) no século XIX e constitui a origem da pesquisa em otimização multiobjetivo. A fronteira de Pareto é baseada na dominância das soluções.

Considerando-se n funções objetivo f_1, \dots, f_n , uma solução x domina outra solução x' , no caso de minimização, se as duas condições abaixo forem satisfeitas:

$$f_i(x) \leq f_i(x') \quad \text{para todo } i = 1, \dots, n \quad 3.1$$

$$\text{e } f_i(x) < f_i(x') \quad \text{para algum } i = 1, \dots, n \quad 3.2$$

Considerando-se n funções objetivo f_1, \dots, f_n e X um conjunto de soluções, a solução $x \in X$ pertence à fronteira de Pareto de X , no caso de minimização, se, para toda solução $x' \in X$ as duas condições 3.1 e 3.2 forem satisfeitas, ou seja, se não existe outra solução $x' \in X$ que melhore ou se iguale à solução $x \in X$ em todas as funções objetivo e que melhore a solução $x \in X$ em alguma função objetivo.

As soluções da fronteira de Pareto também se denominam soluções não inferiores, admissíveis, eficientes ou não dominadas e constituem o conjunto de soluções para um problema multiobjetivo.

3.4.2. Otimização multiobjetivo usando algoritmos evolutivos

Os AE mono objetivos que resolvem problemas de otimização multiobjetivo transformam o problema de otimização multiobjetivo em um problema de otimização mono objetivo, utilizando algum critério de preferência para encontrar uma única solução. Essa solução deve possuir valores aceitáveis em todos os objetivos segundo o critério de preferências que se tem estabelecido, como, por exemplo, priorizar um ou mais objetivos. Se mais de uma solução for requerida, é necessário executar repetidamente o algoritmo modificando o critério de preferências.

Pode-se também obter múltiplas soluções em uma execução do AE para resolver problemas multiobjetivo, tentando encontrar as soluções que estejam na fronteira de Pareto. Estes AE são denominados Algoritmos Evolutivos Multiobjetivo (AEMO). Existem muitos AE que podem tentar encontrar a fronteira de Pareto em problemas

multiobjetivo, como, por exemplo, AG, colônia de formigas, entre outros. Os AG serão detalhados na seção seguinte para resolver problemas multiobjetivo, porque são de interesse neste trabalho.

3.5. Otimização multiobjetivo usando algoritmos genéticos

Como os AE mono objetivo, os AG mono objetivo também são utilizados para resolver problemas de otimização multiobjetivo. Estes AG agrupam todos os objetivos em um único objetivo segundo um critério de preferência, geralmente definindo-se pesos para cada objetivo.

Outra forma de resolver problemas de otimização multiobjetivo usando AG foi proposto por Schaffer (1985) pelo modelo *Vector Evaluated Genetic Algorithm* (VEGA). Nesse modelo, a população é dividida em n subpopulações, sendo n a quantidade de objetivos diferentes. As subpopulações evoluíam separadamente, otimizando um objetivo, e em dados instantes as subpopulações se misturavam e se dividiam novamente. Esse modelo pode obter um conjunto de boas soluções para o problema de otimização multiobjetivo, mas é difícil encontrar as soluções na fronteira de Pareto porque o VEGA não opera com o conceito de dominância das soluções.

Os AG que consideram a dominância das soluções utilizando um conjunto de funções objetivo são denominados Algoritmo Genético Multiobjetivo (AGMO), detalhados na subseção seguinte.

3.5.1. Algoritmos genéticos multiobjetivo

O AGMO proposto por Fonseca e Fleming (1993), foi o primeiro AG a considerar o conceito de dominância de soluções. O resultado de um AGMO é um conjunto de soluções equilibradas em todos os objetivos, ou seja, a fronteira de Pareto.

O funcionamento, a população e os operadores de um AGMO são os mesmos de um AG; a principal diferença entre eles é o cálculo do valor de aptidão de cada solução. O valor de aptidão para uma solução é proporcional ao número de soluções que domina. As soluções não dominadas são destacadas com um valor de aptidão maior.

O entendimento do cálculo do valor de aptidão em um AGMO é um ponto importante para se compreender o funcionamento dos algoritmos propostos neste trabalho.

Descreve-se brevemente a seguir o procedimento para calcular o valor aptidão das soluções em um AGMO mediante um exemplo extraído de (DEB, 2001).

Foram utilizados os valores da Tabela 3.1, onde são mostrados, para uma população de 6 soluções, os valores para os objetivos x e y , e as funções $f(x)$ e $f(y)$, que devem ser minimizadas.

Tabela 3.1: Valores para uma população de 6 soluções

Solução	x	y	$f(x)$	$f(y)$
1	0,31	0,89	0,31	6,10
2	0,43	1,92	0,43	6,79
3	0,22	0,56	0,22	7,09
4	0,59	3,63	0,59	7,85
5	0,66	1,41	0,66	3,65
6	0,83	2,51	0,83	4,23

- 1) Primeiramente a cada solução i , neste caso $i = 1, \dots, 6$, é associado um valor de ranking r_i que é igual ao número de soluções que a dominam (n_i) mais um:

$$r_i = 1 + n_i \quad 3.3$$

A última coluna da Tabela 3.2 mostra os rankings das soluções para os valores da Tabela 3.1:

Tabela 3.2: Ranking das soluções

Solução	x	y	$f(x)$	$f(y)$	Ranking
1	0,31	0,89	0,31	6,10	1
2	0,43	1,92	0,43	6,79	2
3	0,22	0,56	0,22	7,09	1
4	0,59	3,63	0,59	7,85	4
5	0,66	1,41	0,66	3,65	1
6	0,83	2,51	0,83	4,23	2

O valor de n_1 da solução 1 é 0 porque não existe uma solução, das 6 soluções, que a domine, ou seja, nenhuma solução é menor em uma função e menor ou igual na outra função. Assim, o valor para r_1 é 1, segundo a equação 3.3. Para a solução 2, o valor de n_2 é 1, pois há a solução 1 que a domina, já que, tanto o valor de $f(x)$ quanto de $f(y)$ da solução 2 são maiores que os respectivos valores para a solução 1. Portanto o valor de r_2 é 2. Não existe uma solução que domine a solução 3; por isso o valor de r_3 é 1. A solução 4 é dominada por 3 soluções (1, 2 e 3); por isso, o valor de r_4 é 4. Não existe uma solução que

domine a solução 5; por isso o valor de r_5 é 1. A solução 6 é dominada por uma solução (5); por isso, o valor de r_6 é 1.

As soluções de ranking 1 são as soluções não dominadas: 1, 3 e 5. A Figura 3.5 mostra as soluções não dominadas ou soluções de ranking 1 (+), as soluções de ranking 2 (X) e as soluções de ranking 4 (-).

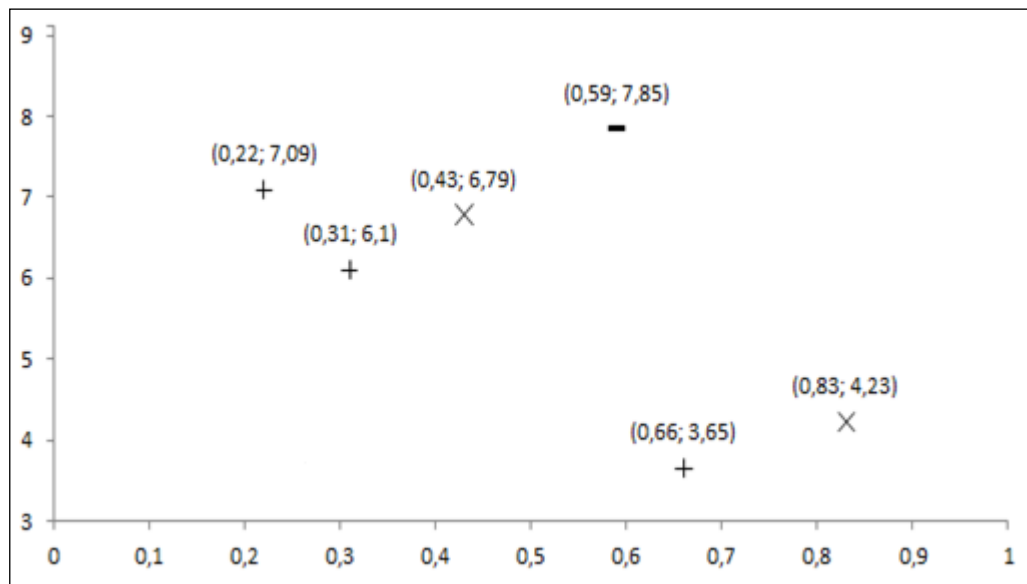


Figura 3.5: Ranking das soluções

- 2) A seguir, a população é ordenada conforme o valor do ranking, e se atribui um valor de aptidão preliminar (*raw fitness*). As soluções com mesmo ranking podem ser ordenadas de forma aleatória; neste exemplo a ordem é definida como (1, 3, 5), (2, 6) e (4). O valor de aptidão preliminar para a solução 1 é 6, para a solução 3 é 5, e assim por diante, até chegar à solução 4, que é 1.
- 3) Posteriormente, o valor de aptidão média é calculado para cada solução i mediante a equação 3.4, dependendo do ranking q dessa solução.

$$F_i = \frac{\sum raw_q}{p_q} \quad 3.4$$

onde:

F_i : aptidão média da solução i .

$\sum raw_q$: soma da aptidão preliminar das soluções do ranking q .

p_q : o número de soluções do ranking q .

A última coluna da Tabela 3.3 mostra os valores da aptidão média para cada solução:

Tabela 3.3: Aptidão média das soluções

Solução	x	y	$f(x)$	$f(y)$	Ranking	Apt. Méd.
1	0,31	0,89	0,31	6,10	1	5,0
2	0,43	1,92	0,43	6,79	2	2,5
3	0,22	0,56	0,22	7,09	1	5,0
4	0,59	3,63	0,59	7,85	4	1,0
5	0,66	1,41	0,66	3,65	1	5,0
6	0,83	2,51	0,83	4,23	2	2,5

O valor da aptidão média para as soluções do ranking 1 é 5,0, que é o resultado da divisão entre o somatório da aptidão preliminar das soluções desse ranking (15), pela quantidade de soluções desse ranking (3). O mesmo cálculo é realizado para as soluções dos rankings 2 e 4.

- 4) Para manter a diversidade entre as soluções de cada ranking, ou seja, que elas estejam mais bem espalhadas no universo das soluções, são utilizados os nichos (conjunto de soluções), sendo que um ranking pode ter 1 ou mais nichos. Quando a distância entre duas soluções do mesmo ranking é maior que o raio do nicho σ_{shared} , que define a vizinhança de uma solução, as soluções do mesmo ranking estão em nichos separados. Uma vez obtidos os nichos, é calculada a aptidão compartilhada F'_i de cada solução i , com base na distância das soluções no mesmo ranking, segundo a equação 3.5.

$$F'_i = \frac{F_i}{nc_{iq}} \quad 3.5$$

onde:

nc_{iq} : soma das distâncias da solução i com as soluções do ranking q no qual pertence i .

O valor para nc_{iq} é calculado segundo a equação 3.6.

$$nc_{iq} = \sum_{j=1}^{p_q} Sh(d_{ij}) \quad 3.6$$

onde:

d_{ij} : distância entre as soluções i e j que pertencem ao ranking q .

$Sh(d_{ij})$: função de compartilhamento entre d_{ij} .

O valor para $Sh(d_{ij})$ é calculado segundo a equação 3.7:

$$Sh(d_{ij}) = \begin{cases} 1 - \left(\frac{d_{ij}}{\sigma_{shared}}\right)^\alpha, & \text{se } d_{ij} \leq \sigma_{shared} \\ 0, & \text{caso contrario} \end{cases} \quad 3.7$$

onde:

α : Parâmetro que define o comportamento da função $Sh(d_{ij})$. Essa função pode ser linear com valor de $\alpha = 1$, quadrática com valor de $\alpha = 2$, cúbica com valor de $\alpha = 3$, ou qualquer outro valor.

No exemplo, o valor para α é definido como 1, e o valor para σ_{shared} é definido como 0,5. A distância d_{ij} é calculada segundo a equação 3.8.

$$d_{ij} = \sqrt{\sum_{k=1}^m \left(\frac{f_k^i - f_k^j}{f_k^{max} - f_k^{min}} \right)^2} \quad 3.8$$

onde:

m : quantidade de funções objetivo.

f_k^i : valor da função objetivo k da solução i .

f_k^j : valor da função objetivo k da solução j .

f_k^{max} : valor máximo para a k – ésima função objetivo.

f_k^{min} : valor mínimo para a k – ésima função objetivo.

No exemplo, $f_x^{max} = 1$, $f_y^{max} = 10$, $f_x^{min} = 0,1$ e $f_y^{min} = 1$.

Para as soluções de ranking 1 no exemplo, segundo as equações e os valores apresentados anteriormente, são obtidas as seguintes distâncias:

$$d_{13} = 0,149$$

$$d_{15} = 0,475$$

$$d_{35} = 0,621$$

Com os valores das distâncias para as soluções de ranking 1, é calculado o valor de Sh :

$$Sh(d_{13}) = 0,702$$

$$Sh(d_{15}) = 0,005$$

$$Sh(d_{35}) = 0 \text{ (0,621 é maior que } \sigma_{shared} = 0,5)$$

Os valores de Sh para as distâncias entre um ponto e ele mesmo é 1.

Em seguida, para cada solução do ranking 1, são calculado os valores de nc : $n_{11} = 1,752$; $d_{31} = 1,702$; $d_{51} = 1,050$ e os valores de F' : $F'_1 = 2,854$; $F'_3 = 2,983$; $F'_5 = 4,762$.

A penúltima coluna da Tabela 3.4 mostra o valor do contador do nicho, e a última coluna, o valor de aptidão compartilhada para as soluções do ranking 1.

Tabela 3.4: Contador do nicho e aptidão compartilhada

Solu.	x	y	$f(x)$	$f(y)$	Rank.	Apt. Méd.	Cont. do Nicho.	Apt. Comp.
1	0,31	0,89	0,31	6,10	1	5,0	1,752	2,854
2	0,43	1,92	0,43	6,79	2	2,5		
3	0,22	0,56	0,22	7,09	1	5,0	1,702	2,938
4	0,59	3,63	0,59	7,85	4	1,0		
5	0,66	1,41	0,66	3,65	1	5,0	1,050	4,762
6	0,83	2,51	0,83	4,23	2	2,5		

Finalmente a aptidão compartilhada é multiplicada por um fator de conversão escalar, resultando no valor da aptidão final da solução das soluções i , Ap_i , definida pela equação 3.16.

$$Ap_i = \frac{F_i \times p_q}{\sum_{j=1}^{p_q} F'_j} \times F'_i \tag{3.9}$$

As aptidões finais para as soluções de ranking 1 são mostradas na última coluna da Tabela 3.5.

Tabela 3.5: Aptidão final

Solu.	x	y	$f(x)$	$f(y)$	Rank.	Apt. Méd.	Cont. do Nicho.	Apt. Comp.	Apt. Final.
1	0,31	0,89	0,31	6,10	1	5,0	1,752	2,854	4,056
2	0,43	1,92	0,43	6,79	2	2,5			
3	0,22	0,56	0,22	7,09	1	5,0	1,702	2,938	4,176
4	0,59	3,63	0,59	7,85	4	1,0			
5	0,66	1,41	0,66	3,65	1	5,0	1,050	4,762	6,768
6	0,83	2,51	0,83	4,23	2	2,5			

Como mencionado anteriormente, os valores para o contador do nicho, a aptidão compartilhada e a aptidão final, são calculados para os ranking 2 e 4. A Tabela 3.6 mostra os valores finais para o exemplo descrito sobre AGMO.

Tabela 3.6: Valores finais para o exemplo de um AGMO

Sol.	x	Y	f(x)	f(y)	Rank.	Apt. Méd.	Cont. do Nicho.	Apt. Comp.	Apt. Final.
1	0,31	0,89	0,31	6,10	1	5,0	1,752	2,854	4,056
2	0,43	1,92	0,43	6,79	2	2,5	1,000	2,500	2,500
3	0,22	0,56	0,22	7,09	1	5,0	1,702	2,938	4,176
4	0,59	3,63	0,59	7,85	4	1,0	1,000	1,000	1,000
5	0,66	1,41	0,66	3,65	1	5,0	1,050	4,762	6,768
6	0,83	2,51	0,83	4,23	2	2,5	1,000	2,5000	2,500

Os valores da aptidão final mostram que as soluções com maior ranking tem uma aptidão maior do que outras soluções com ranking menor; e dentro do mesmo ranking, as soluções mais distantes têm maior aptidão que as soluções mais próximas.

Este exemplo mostra como o conceito de dominância das soluções mediante rankings e a diversidade das soluções são levados em conta para definir a aptidão das soluções, e conseqüentemente, após se aplicar o processo evolutivo do AGMO, encontrar a fronteira de Pareto para um problema multiobjetivo. No entanto, o AGMO não utiliza o conceito de elitismo das soluções, ou seja, não garante a permanência da melhor solução ou melhores soluções dentro do processo evolutivo.

Como solução desse problema foram propostos diferentes técnicas ou algoritmos de otimização multiobjetivo que consideram tanto o conceito de dominância das soluções mediante rankings e a diversidade das soluções, geralmente mudando a forma de encontrar uma boa diversidade, como o elitismo das soluções. Duas técnicas bastante utilizadas são o Algoritmo Genético com Ordenação não Dominada (*Non-dominated Sorting Genetic Algorithm II - NSGA-II*) e o Algoritmo Evolutivo de Pareto Robusto 2 (*Strength Pareto Evolutionary Algorithm – SPEA2*), apresentadas na próxima subseção.

3.5.2. Algoritmo Genético com Ordenação não Dominada II (Non-dominated Sorting Genetic Algorithm II – NSGA-II)

A técnica de otimização multiobjetivo NSGA-II foi apresentada em (DEB; PRATAB; AGRAWAL; MEYARIVAN; 2002) e é baseada em um ordenamento elitista por não dominância das soluções e a distância das soluções.

No NSGA-II, inicialmente existe uma população de soluções em que são aplicados operadores genéticos de acordo com a dominância das soluções para gerar uma

segunda população. Ambas as populações se unem, em uma única população, na qual as soluções são classificadas em fronteiras de acordo a sua dominância. As melhores soluções, ou seja, as soluções que pertençam às primeiras fronteiras e estejam mais distantes das outras soluções, formaram parte da população inicial seguinte; as piores soluções são rejeitadas. O processo continua até atingir um número máximo de gerações.

O NSGA-II é ilustrado na Figura 3.6:

Variáveis:

P : População pai

Q : População filha

T : Tamanho fixo para P e Q

F_i : Conjunto de soluções na fronteira i

n : Número de geração atual

N : Máximo número de gerações

1. Gerar a população inicial P_0 e $Q_0 = \{ \}$
2. Atribuir $n = 0$
3. Realizar a seleção, o cruzamento e a mutação em P_n para gerar a população filha Q_n
4. Fazer $R_n = P_n \cup Q_n$
5. Realizar a ordenação por não dominância em R_n , gerando $F_i, i = 1, \dots, v$ em R_n
6. Criar $P_{n+1} = \{ \}$
7. Enquanto $|P_{n+1}| + |F_i| \leq N$
8. Copiar as soluções de F_i em P_{n+1}
9. $i = i + 1$
10. Calcular as distâncias de multidão para cada solução em F_i
11. Ordenar F_i decrescentemente de acordo as distâncias de multidão de cada solução
12. Copiar as primeiras $N - |P_{n+1}|$ soluções ordenadas de F_i para P_{n+1}
13. Se $n \geq N$ então pare
14. Senão atribuir $n = n + 1$ e voltar ao passo 3.

Figura 3.6: Técnica de otimização multiobjetivo NSGA-II – Adaptado (CANCINO, 2003)

Na linha 10 do NSGA-II é necessário calcular a distância de multidão para cada solução em uma fronteira F_i . A distância de multidão de uma solução j , d_j , representa uma estimativa do perímetro formado pelo cuboide cujos vértices são os seus vizinhos mais próximos. A Figura 3.7 mostra a distância de multidão para a solução j .

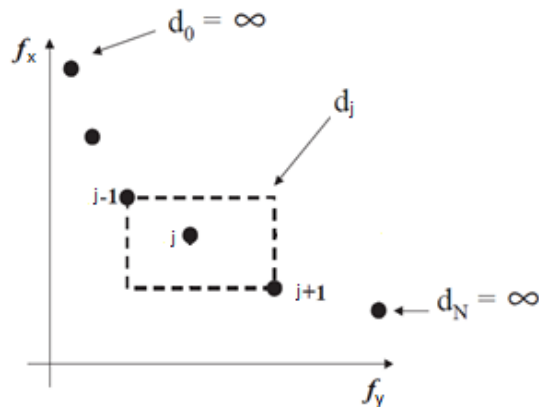


Figura 3.7: Distância de multidão em NSGA-II – Adaptado (CANCINO, 2003)

Quanto maior o cuboide de j , mais afastada se encontra j dos seus vizinhos. As soluções extremas em cada objetivo terão um cuboide infinito, ou seja, uma distância de multidão infinita. O procedimento para achar a distância de multidão de uma solução está descrito na Figura 3.8.

Variáveis:

F_i : Conjunto de soluções na fronteira i

v_i : Número de soluções na fronteira i

m : Número de funções objetivo

1. Para cada solução j em F_i atribui-se $d_j = 0$.
2. Para cada função objetivo $k = 1, 2, \dots, m$
3. Gerar a lista de soluções I^m igual ao conjunto soluções de F_i ordenadas decrescentemente por f_k
4. Fazer $d_{I_1^m} = d_{I_{v_i}^m} = \infty$
5. Para as soluções $t = 2, \dots, v_i - 1$
6.
$$d_{I_t^m} = d_{I_t^m} + \frac{f_k^{(I_{t+1}^m)} - f_k^{(I_{t-1}^m)}}{f_k^{max} - f_k^{min}}$$

Figura 3.8: Cálculo da distância de multidão em NSGA-II – Adaptado (CANCINO, 2003)

Onde:

f_k : valor da função objetivo k .

I^m : lista ordenada decrescentemente por f_k das soluções em F_i .

$d_{I_t^m}$: distância de multidão da t – ésima solução em I^m .

$f_k^{(I_{t+1}^m)}$ e $f_k^{(I_{t-1}^m)}$: f_k dos vizinhos da t – ésima solução em I^m .

f_k^{max} e f_k^{min} : máximo e mínimo global para o f_k .

3.5.3. Algoritmo Evolutivo de Pareto Robusto 2 (Strength Pareto Evolutionary Algorithm 2 – SPEA2)

O algoritmo SPEA2 foi apresentado em (ZITZLER; LAUMANNNS; THIELE; 2001) e usa elitismo através de uma população externa onde são armazenadas um conjunto de soluções não dominadas.

No SPEA2, inicialmente existem duas populações de soluções. Ambas as populações se unem, em uma única população, da qual são escolhidas as melhores soluções, ou seja, as soluções que dominem mais soluções, que sejam dominadas por menos soluções e estejam mais distantes das outras soluções, para formar parte da primeira população da iteração seguinte; as piores soluções são rejeitadas. Na primeira solução são aplicados operadores genéticos para gerar a segunda população. O processo continua até atingir um número máximo de gerações.

O SPEA2 é ilustrado na Figura 3.9:

Variáveis:

P : População interna

E : População externa

T : Tamanho fixo para E

N : Máximo número de gerações

n : Número de geração atual

1. Gerar a população inicial P_0 e $E_0 = \{ \}$
2. Atribuir $n = 0$
3. $Q_n = P_n \cup E_n$
4. Fazer $E_n = \{ \}$
5. Para cada solução j em Q_n
6. Calcular os valores s_j , r_j , d_j e F_j
7. Copiar as soluções não dominadas de Q_n em E_n
8. Se $|E_n| < T$
9. Ordenar Q_n conforme a F_j
10. Copiar as melhores $E_n - N$ soluções j de Q_n tal que $F_j \geq 1$
11. Se $|E_n| > T$
12. Reduzir E_n utilizando o algoritmo de corte
13. Se $n > N$ então pare
14. Aplicar seleção, cruzamento e mutação sobre E_n para gerar P_{n+1}
15. Fazer $E_{n+1} = E_n$
16. Fazer $n = n + 1$
17. Voltar ao passo 3.

Figura 3.9: Técnica de otimização multiobjetivo SPEA2 – Adaptado (CANCINO, 2003)

Na linha 6 do SPEA2 calcula-se quatro valores: s_j (aptidão robusta ou *strength fitness*), r_j (aptidão preliminar ou raw fitness), d_j (densidade) e F_j (aptidão ou fitness), explicados a seguir:

- Aptidão robusta: O valor s_j é o número de soluções que j domina, as soluções não dominadas têm $s_j = 0$. O valor s_j é calculado mediante a equação 3.10 onde $j \lesssim l$ expressa que a solução j domina a solução l .

$$s_j = |\{l, l \in Q_n, \text{ tal que } j \lesssim l\}| \quad 3.10$$

- Aptidão preliminar: O valor r_j é a soma dos s_l das soluções l que j domina e é calculado mediante a equação 3.11. As soluções não dominadas têm $r_j = 0$.

$$r_j = \sum_{l \in Q_n, j \lesssim l} s_l \quad 3.11$$

- Densidade: O valor d_j é calculado mediante a equação 3.12.

$$d_j = \frac{1}{\sigma_j^{k+2}} \quad 3.12$$

ou seja, é inversamente proporcional à distância ao k – éximo vizinho mais próximo σ_j^k , onde $k = \sqrt{|Q_n|}$. O valor da d_j está dentro do intervalo aberto $(0,1)$.

- Aptidão: O valor F_j é calculado mediante a equação 3.13, que representa a soma da aptidão preliminar e a densidade da solução j .

$$F_j = r_j + d_j \quad 3.13$$

Segundo a equação 3.13, as soluções não dominadas terão $F_j < 1$, e as demais soluções terão $F_j \geq 1$.

Na linha 12 do SPEA2 a população E_n é reduzida mediante um algoritmo de corte, onde em cada iteração escolhe-se uma solução tal que a sua distância para o vizinho mais próximo seja a menor possível. No caso de empate, calcula-se a segunda menor distância, e assim por diante. A Figura 3.10 ilustra um conjunto de soluções; as setas indicam o segundo vizinho mais próximo. Aplicado o algoritmo de corte do SPEA2, a solução com segunda menor distância é eliminada.

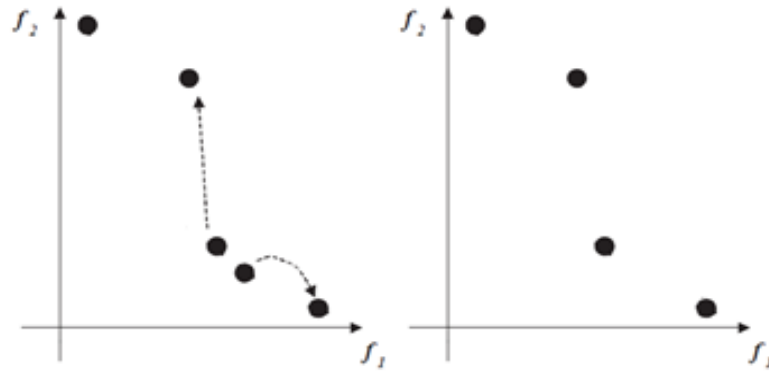


Figura 3.10: Algoritmo de corte do SPEA2 – Adaptado (CANCINO, 2003)

3.6. Considerações finais

Neste capítulo foi apresentado o conceito de computação evolutiva, assim como o conceito dos Algoritmos Genéticos (AG) como técnica de otimização. Esses AG podem ser utilizados para resolver problemas de otimização multiobjetivo, definidos como Algoritmos Genéticos Multiobjetivo (AGMO), mudando-se principalmente a forma de cálculo da aptidão de cada solução, baseada na dominância das soluções.

Foi apresentado o procedimento e um exemplo do cálculo de aptidão de um AGMO, e também explicadas duas das técnicas de otimização multiobjetivo mais conhecidas, o NSGA-II e o SPEA2.

Todos os conceitos sobre AG e otimização multiobjetivo, vistos neste capítulo, serão usados nos próximos, que aborda sistemas *fuzzy* genéticos e o método proposto neste trabalho para a geração genética multiobjetivo da BC em SFBR.

Capítulo 4

SISTEMAS *FUZZY* GENÉTICOS

4.1. Considerações iniciais

Com base nos conceitos de sistemas *fuzzy* e computação evolutiva, apresentados nos capítulos 2 e 3 respectivamente, neste capítulo são apresentados os sistemas *fuzzy* genéticos. Esses sistemas relacionam-se com a utilização das técnicas da computação evolutiva, especificamente os AG, para gerar ou otimizar componentes dos sistemas *fuzzy*.

Neste capítulo serão apresentados os conceitos de sistemas *fuzzy* genéticos, as diferentes abordagens dos sistemas *fuzzy* genéticos e mais especificamente, geração de SFBR que utilizam AGMO.

4.2. Sistemas *fuzzy* genéticos

O termo Sistemas *Fuzzy* Genéticos (SFG) tem sido utilizado para denominar um SF combinado com um processo de aprendizado baseado em AG. A classe dos SFG inclui as Redes Neurais *Fuzzy* Genéticas, os Algoritmos de Agrupamento *Fuzzy* Genéticos e os Sistemas *Fuzzy* Genéticos Baseados em Regras (SFGBR).

O tipo de SFG de interesse para este trabalho é o SFGBR, onde um AG é empregado para aprender ou otimizar diferentes componentes de um Sistema *Fuzzy* Baseado em Regras (SFBR) (CORDON; HERRERA; GOMIDE; HOFFMANN; MAGDALENA, 2004) (CASILLAS; JESUS; HERRERA; PEREZ; VILAR, 2007) (CORDON; ALCALA; ALCALA-FDEZ; ROJAS, 2007) (CARSE; PIPE, 2007) (HERRERA, 2008).

Nesse contexto, os AG têm sido utilizados com sucesso na geração ou na adaptação dos componentes da Base de Conhecimento (BC), ou seja, tanto da Base de Regras (BR), quanto da Base de Dados (BD), bem como na adaptação do Mecanismo de Inferência (MI) do SF.

A organização das abordagens de SFGBR existentes em categorias está fundamentada na definição de dois aspectos básicos: qual componente do SFBR é o objeto de otimização pelo AG e qual classe de problemas está sendo abordada: problemas de otimização ou adaptação dos componentes do SFBR previamente definidos ou problemas de geração ou projeto dos componentes do SFBR sem definição prévia.

A seguir são apresentadas as principais abordagens encontradas na literatura para geração ou otimização de SFBR, agrupadas em uma classificação extraída e adaptada de (CORDON; HERRERA; GOMIDE; HOFFMANN; MAGDALENA, 2004) e (HERRERA, 2008).

4.3. Abordagens de sistemas *fuzzy* genéticos

A ideia básica da utilização de AG no aprendizado automático de SFBR é que o processo de construção do sistema pode ser modelado como um problema de busca no espaço de soluções.

Do ponto de vista de otimização, encontrar um SF apropriado é equivalente a codificar esse SF como uma estrutura de parâmetros e então encontrar os valores dos parâmetros que dão uma boa solução de acordo com uma função de aptidão.

No contexto dos SFGBR, os métodos para a geração da BC, podem ser divididos em dois grupos principais:

- Adaptação genética: Se existe uma BC, é aplicado um processo de refinamento genético para melhorar o desempenho do SFBR.
- Aprendizado genético: Neste grupo, o objetivo é aprender um ou mais componentes da BC, onde se pode incluir um mecanismo de inferência adaptativo.

De acordo com esses dois grupos, (HERRERA, 2008) propõe a taxonomia mostrada na Figura 4.1 e apresentadas a seguir:

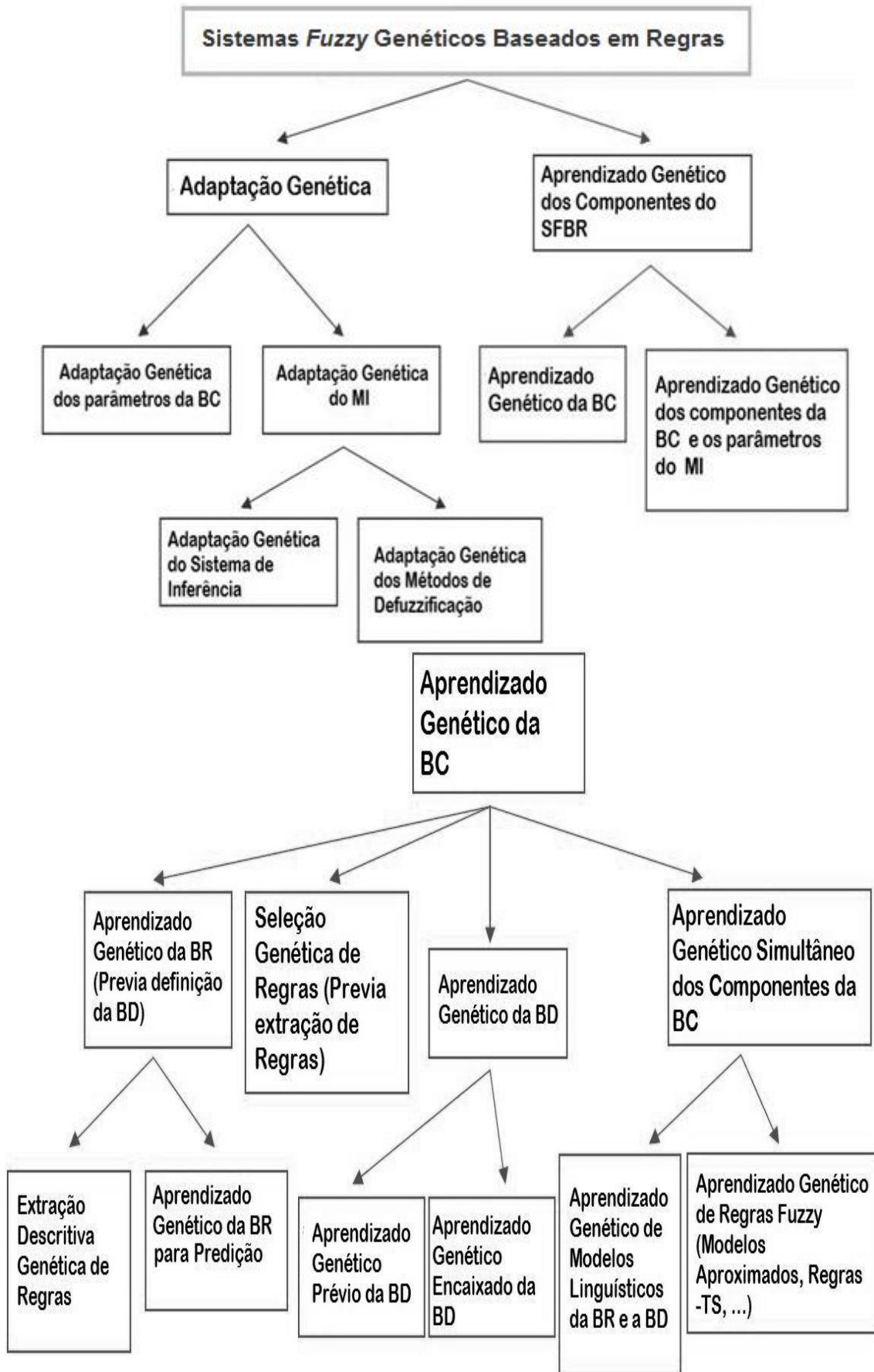


Figura 4.1: Taxonomia dos sistemas *fuzzy* genéticos baseados em regras– Adaptado (HERRERA, 2008)

Existem duas áreas principais na taxonomia apresentada na Figura 4.1: adaptação genética e o aprendizado genético dos componentes do SFBR. As subseções seguintes apresentam brevemente essas áreas.

4.3.1. Adaptação genética

Estão incluídos no grupo de adaptação genética os métodos que partem da BR ou BD existentes. Eles utilizam os AG para aperfeiçoar o desempenho do SFBR, adaptando os parâmetros da BC ou adaptando o Mecanismo de Inferência (MI). A seguir são brevemente apresentados esses métodos:

1) Adaptação genética dos parâmetros da BC

Neste grupo encontram-se os métodos que focalizam a sintonia de conjuntos *fuzzy* (CASILLAS; CORDON; JESUS; HERRERA, 2001) (PIRES; CAMARGO, 2004) (ALONSO; CORDON; GUILLAUME; MAGDALENA, 2007). Nesses métodos a BR e a BD são previamente definidas, manualmente ou por um processo automático qualquer. A BR permanece fixa e a BD é ajustada pelo AG, por meio da alteração dos parâmetros das funções de pertinência de todas as partições *fuzzy* envolvidas no problema. O número de termos linguísticos em cada partição permanece fixo desde o início do processo.

2) Adaptação genética do sistema de inferência

O principal alvo desta abordagem é obter uma alta cooperação entre as regras *fuzzy*, para aumentar a precisão nos modelos *fuzzy*, sem perder a interpretabilidade linguística, como buscar a melhor t-norma utilizada na inferência de um SFBR para aumentar a precisão. Em (CROCKETT; BANDAR; MCLEAN, 2007) (ALCALA-FDEZ; HERRERA; MARQUEZ; PEREGRIN, 2007) se pode encontrar propostas nesta área focadas em problemas de regressão e classificação.

3) Adaptação genética do método de defuzificação

A técnica de defuzificação mais usada na prática consiste em aplicar a função de defuzificação para cada conjunto *fuzzy* das regras disparadas e depois calcular o valor de defuzificação por um operador de média ponderada. Esta forma de cálculo introduz a possibilidade de usar funções de média baseadas em

parâmetros, e a utilização de AG pode permitir adaptar o método de defuzzificação. Em (KIM; CHOI; LEE, 2002) pode-se encontrar uma proposta nesta área.

4.3.2. Aprendizado genético dos componentes dos SFBR

Estão incluídos no grupo do aprendizado genético dos componentes dos SFBR os métodos de aprendizado genético da BC e os métodos baseados em um modelo híbrido entre o aprendizado genético da BC e a adaptação genética do MI.

Esse modelo híbrido tenta encontrar uma alta cooperação entre os parâmetros do MI e o aprendizado dos componentes da BC, incluindo ambos em um processo de aprendizado simultâneo. Uma proposta desse modelo pode ser encontrada em (MARQUEZ; PEREGRIN; HERRERA, 2007).

Os métodos de aprendizado genético da BC utilizam AG para construir, gerar ou projetar efetivamente um ou mais componentes da BC. Esse grupo de métodos foi o que gerou o maior número de pesquisas e pode ainda ser subdividido em quatro subgrupos ou abordagens, brevemente apresentados a seguir.

1) Aprendizado genético da BR

Essa abordagem estuda a construção das regras *fuzzy* considerando um conjunto de funções de pertinência já definido e fixo. Usualmente a definição da BD é feita escolhendo-se um número de valores linguísticos para as variáveis e distribuindo uniformemente os valores, que são rótulos dos conjuntos *fuzzy* no domínio de cada variável (ABADEH; HABIBI, 2007) (EVSUKOFF, 2007) (SANCHEZ; COUSO; CASILLAS, 2009) (XIONG; 2009).

No aprendizado genético de regras, quatro abordagens são comumente utilizadas (CORDON; HERRERA; GOMIDE; HOFFMANN; MAGDALENA, 2004): Pittsburgh onde cada cromossomo representa uma BR na qual a BR final é o melhor cromossomo obtido (SMITH, 1980); Michigan, onde uma regra Se→Então é codificada como um cromossomo e a BR final é a união de todos os cromossomos da melhor população (HOLLAND; REITMAN, 1978); iterativa ou *Iterative Rule Learning* (IRL), onde uma regra Se→Então é codificada como um cromossomo, mas a diferença da abordagem Michigan, uma nova regra é adaptada e adicionada na BR final de forma iterativa em cada execução do AG (ABADEH; HABIBI, 2007) (GONZALEZ; PEREZ, 2008); e Cooperativa–Competitiva, em que a população inteira ou um subconjunto dela codifica a BR e

os cromossomos competem e cooperam simultaneamente (GREENE; SMITH, 1993).

2) Seleção genética de regras

Ocasionalmente, pode-se ter um grande número de regras extraídas por meio de um método de mineração de dados. Uma BR com um número excessivo de regras dificulta o entendimento do comportamento do SFBR. Diferentes tipos de regras podem ser encontrados em um conjunto de regras *fuzzy*: regras irrelevantes, redundantes, erradas e conflitantes, que perturbam o desempenho do SFBR. Para tratar este problema, pode-se utilizar um processo de seleção de regras genético para obter um subconjunto de regras a partir de um conjunto prévio de regras *fuzzy* mediante a seleção de algumas dessas regras. (CINTRA; CAMARGO, 2007) (ISHIBUCHI; NOJIMA; KUWAJIMA, 2006) (ISHIBUCHI; NOJIMA; KUWAJIMA, 2008) (ISHIBUCHI; NOJIMA, 2009).

A seleção de regras também pode ser combinada com abordagens de refinamento, para tentar obter um bom conjunto de regras junto a um conjunto refinado de parâmetros. Em (CASILLAS; CORDON; JESUS; HERRERA, 2005) (ALCALA; GACTO; HERRERA; ALCALA-FDEZ, 2007) encontram-se duas propostas que combinam refinamento genético com seleção de regras.

3) Aprendizado genético da BD

Existem outras formas para gerar a BC que consideram dois diferentes processos que derivam ambos os componentes, a BD e a BR. Na primeira possibilidade, definida como aprendizado genético 'a priori' da BD, um processo de geração da BD permite aprender a forma da função de pertinência dos conjuntos *fuzzy*. Este processo de geração da BD pode usar uma medida para avaliar a qualidade da BD; depois, um método de geração de regras é utilizado. A segunda possibilidade é considerar um processo de aprendizado genético embutido, onde a geração da BD envolve um aprendizado da BR, ou seja, cada vez que uma BD é obtida pelo processo de geração da BD, o método de geração da BR é usado para derivar as regras; depois um tipo de medida de erro é utilizado para validar a BC.

Algumas propostas sobre o aprendizado genético da BD podem ser encontradas em (CORDON; JESUS; HERRERA; VILLAR, 2001) (CORDON; HERRERA; VILLAR, 2001) (PIRES; CAMARGO, 2004) (TZUNG-PEI; WEI-TEE; CHIH-PING; CHEN-SEM, 2009).

4) Aprendizado genético simultâneo dos componentes da BC

Esta abordagem tem como objetivo aprender os dois componentes da BC simultaneamente (ANTONELLI; DUCANGE; LAZZERINI; MARCELLONI, 2008) (ISHIBUCHI; NOJIMA; KUWAJIMA, 2009) (ALCALA; DUCANGE; HERRERA; LAZZERINI; MARCELLONI, 2009). Seguindo esta abordagem é possível gerar uma melhor definição do SFBR, mas a necessidade de lidar com um grande espaço de busca torna o processo difícil e lento.

4.4. Aprendizado genético multiobjetivo de SFBR

Segundo (HERRERA, 2008), os Algoritmos Evolutivos Multiobjetivo (AEMO), apresentados no capítulo 3, constituem-se em uma das áreas mais ativas de pesquisa no campo da Computação Evolutiva (CE). São capazes de encontrar um conjunto de soluções não dominadas em uma única execução do algoritmo.

Na geração de SFBR geralmente dois objetivos são levados em conta: aumentar a precisão (por exemplo, aumentar a taxa de acerto do SFBR) e aumentar a interpretabilidade (por exemplo, diminuir a quantidade de regras ou a quantidade de condições). Porém, obter um alto grau de precisão e de interpretabilidade são objetivos contraditórios em SFBR, e, na prática, uma das duas propriedades prevalece sobre a outra. A Figura 4.2 mostra o relacionamento entre os dois objetivos desse problema onde cada elipse é um SFBR.

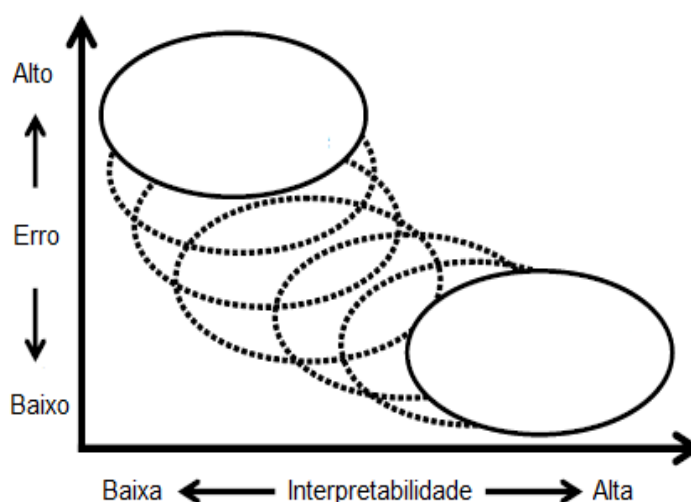


Figura 4.2: Curva do equilíbrio entre precisão – interpretabilidade em sistemas fuzzy baseados em regras – Adaptado (ISHIBUCHI, 2007)

Em SFGBR multiobjetivo é desejável projetar algoritmos para o aprendizado genético em que o mecanismo de aprendizagem encontre um equilíbrio adequado

entre precisão e interpretabilidade. Em (HERRERA, 2005) a utilização de AGMO foi considerada como uma das mais promissoras técnicas para a geração de SFG.

A grande quantidade de pesquisas atuais sobre a geração de SFBR utilizando AGMO revalida essa consideração apresentada em (HERRERA, 2008). A seguir são mencionados alguns métodos nesta área de pesquisa:

Na geração genética multiobjetivo da BR, o método *Non-dominated Multiobjective Evolutionary Algorithm for Extracting Fuzzy Rules in Subgroup Discovery* NMEEF-SD foi proposto em (CARMONA; GONZALEZ; JESUS; HERRERA, 2009) mediante a técnica de otimização multiobjetivo NSGA-II. Outro método utilizado para gerar regras *fuzzy*, mediante a técnica de otimização multiobjetivo SPEA2, é o *Multiobjective Evolutionary Subgroup Discovery Fuzzy rules* (MESDIF-SD) proposto por (BERLANGA; JESUS; GONZALEZ; HERRERA; MESONERO, 2006). Em (ANTONELLI; DUCANGE; MARCELLONI, 2010) é aplicado um AGMO baseado na técnica de otimização multiobjetivo (2+2)M-PAES para gerar um conjunto de BR não dominadas tipo Mamdani com equilíbrio entre precisão e interpretabilidade.

Algumas pesquisas focam a seleção de regras utilizando um AGMO, como em (ISHIBUCHI; YAMAMOTO, 2004) onde são utilizados três objetivos: erro de classificação para medir a precisão; o número de regras e a quantidade de premissas por regra para medir a interpretabilidade e complexidade do SFBR. Todos os objetivos são minimizados. Em (GACTO; ALCALA; HERRERA, 2009) foram apresentados e analisados seis diferentes AGMO para obter modelos linguísticos *fuzzy* simples e precisos que, além da seleção de regras, otimizam os conjuntos *fuzzy*.

Outras pesquisas geram a BR e otimizam a BD mediante um AGMO, como a pesquisa apresentada em (DI NUOVO; CATANIA, 2009), onde são utilizados quatro objetivos: precisão, interpretabilidade, complexidade e transparência. Em (PULKKIEN; KOIVISTO, 2010) um AGMO é utilizado para otimizar a BD e gerar a BR. Nesse trabalho é apresentado também o conceito de conjuntos *fuzzy* transparentes para aumentar a interpretabilidade da BD.

A geração simultânea da BR e da BD também pode ser realizada mediante AGMO, como na pesquisa apresentada em (COCOCCIONI; DUCANGE; LANZERINI; MARCELLONI, 2007), que especificamente gera sistemas *fuzzy* Mamdani. Em (ALCALA; DUCANGE; HERRERA; LAZZERINI; MARCELLONI, 2009)

propõe-se o uso de um AEMO para gerar um conjunto de SFBR com diferentes compensações entre precisão e interpretabilidade em problemas de regressão.

Assim como as pesquisas abordam a geração ou otimização dos componentes da BC, elas também abordam a otimização do MI. Por exemplo, no trabalho apresentado em (MARQUEZ; MARQUEZ; PEREGRIN, 2008), é proposto utilizar AGMO para a geração de SFBR com um bom equilíbrio entre precisão e interpretabilidade baseado na geração cooperativa entre os parâmetros dos operadores *fuzzy* adaptativos no sistema de inferência e as regras na BR através de um mecanismo de seleção. O mecanismo proposto em (MARQUEZ, MARQUEZ, PEREGRIN, 2010) melhora a interpretabilidade no sentido de complexidade linguística para SFBR com defuzzificação adaptativa.

4.5. Pesquisas sobre a interpretabilidade em SFBR

Como apresentado na seção anterior, o aprendizado genético multiobjetivo de SFBR geralmente têm dois objetivos: aumentar a precisão e aumentar a interpretabilidade. Enquanto que, determinar a precisão em um SFBR é relativamente simples, determinar a interpretabilidade é uma tarefa bastante complexa. A maioria dos pesquisadores concorda que a interpretabilidade pode ser medida envolvendo diferentes aspectos ou características dos SFBR, como: o número de regras deve ser suficiente para que o funcionamento do sistema seja compreensível; as premissas das regras devem ter uma estrutura simples e conter poucas variáveis de entrada; os termos linguísticos devem ser intuitivamente compreensíveis.

Embora muitos trabalhos adotem como medida de interpretabilidade do sistema parâmetros simples como o número de regras ou número de condições nos antecedentes, diversos outros aspectos, além dos já mencionados, interferem na avaliação da interpretabilidade.

Alguns trabalhos propõem novas métricas de interpretabilidade como por exemplo, a geração de índices de interpretabilidade para os conjuntos *fuzzy* da BD apresentada em (BOTTA; LAZZERINE; MARCELLONI; STEFANESCU, 2007), que avalia a ordenação, a distinguibilidade e a cobertura dos conjuntos *fuzzy*. o trabalho apresentado em (GACTO; ALCALA; HERRERA, 2010) propõe um índice que ajuda

a preservar a interpretabilidade semântica da BD enquanto um método de otimização genética multiobjetivo da BD é executado.

Um trabalho que categoriza a interpretabilidade de um modelo *fuzzy* em interpretabilidade de baixo nível (otimização da BD) e interpretabilidade de alto nível (diminuir a complexidade da BD) é apresentado em (ZHOU; GAN, 2008);

Em (MENCAR; CASTIELLO; FANELLI, 2009) descreve-se uma abordagem para avaliar a interpretabilidade de SFBR para classificação, mediante a avaliação da semântica da BR e da BD, criando uma nova BR a partir da BR original utilizando um operador conservador da verdade (“truth-preserving”). Depois a taxa de erro das duas BR é determinada, concluindo que, se as taxas de erro são similares, então o SFBR tem alta interpretabilidade ou, em caso contrário, o SFBR tem baixa interpretabilidade.

4.6. Pesquisas importantes para o método proposto de geração genética multiobjetivo utilizando a abordagem iterativa

Nas seções anteriores foram mencionadas algumas pesquisas importantes para o desenvolvimento deste trabalho. Essas pesquisas são: o método de geração genética iterativa da BR apresentado em (GONZALEZ; PEREZ, 2008), chamado de *New Structural Learning algorithm in Vague environment* (NSLV); o índice de interpretabilidade semântica para uma BD apresentado em (GACTO; ALCALA; HERRERA, 2010); os métodos de geração genética multiobjetivo de regras *fuzzy*, NMEEF-SD e MESDIF-SD apresentados em (CARMONA; GONZALEZ; JESUS; HERRERA, 2009) e (BERLANGA; JESUS; GONZALEZ; HERRERA; MESONERO, 2006) respectivamente; o conceito de conjuntos *fuzzy* transparentes apresentado em (PULKKIEN; KOIVISTO, 2010); e o método de otimização genética da BD apresentado em (PIMENTA, 2009). A seguir são detalhadas brevemente cada uma dessas pesquisas.

4.6.1. Novo algoritmo de aprendizagem estrutural em ambientes imprecisos (New Structural Learning algorithm in Vague environment – NSLV)

Esse método é uma melhora do método de aprendizado genético da BR baseado na abordagem IRL (GONZALEZ; RAUL, 1999), definido como *Structural Learning Algorithm in Vague Environment* (SLAVE).

A principal diferença entre o SLAVE e o NSLV é que no NSLV não é necessário definir a ordem das classes a aprender, para evitar que a ordem do aprendizado das classes influencie na geração das regras.

O método NSLV segue o fluxo ilustrado na Figura 4.3.

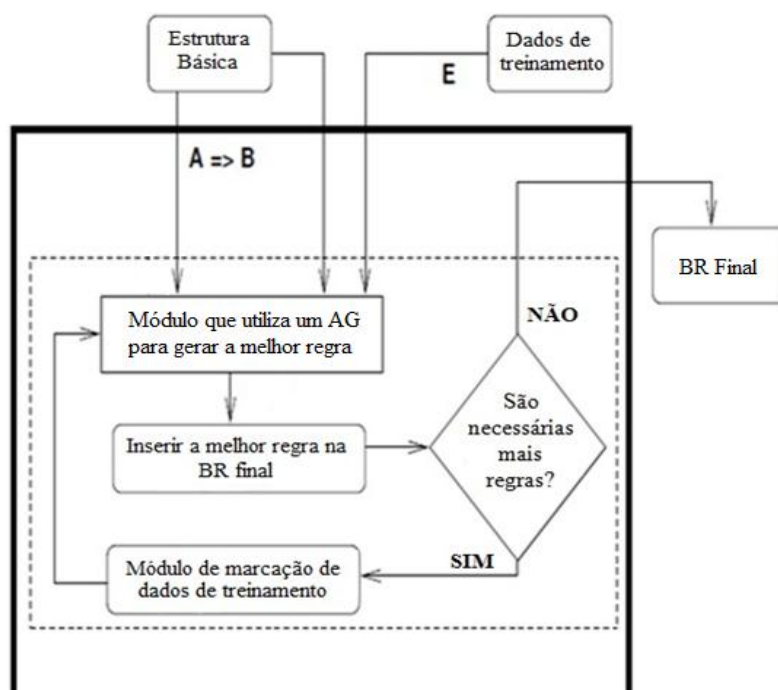


Figura 4.3: Fluxo do método NSLV – Adaptado (GONZALEZ; RAUL, 2008)

Primeiramente é definida a estrutura das regras *fuzzy*, neste caso $A \Rightarrow B$, onde A é o antecedente que contém os atributos dos exemplos e os conjuntos *fuzzy* para cada atributo e B contém a classe da regra.

O processo iterativo começa com a utilização de um AG para gerar a melhor regra com os dados de treinamento iniciais. A função de aptidão está baseada nos conceitos de integridade e consistência de uma regra. A melhor regra da última população do AG é inserida na BR final.

Se não forem necessárias mais regras, o processo de geração de regras termina, e o resultado é a BR final. Se forem necessárias mais regras, é executado um módulo de marcação de dados de treinamento, marcando os exemplos que tenham um grau

de compatibilidade maior ou igual que um grau de compatibilidade previamente definido, por exemplo 0,8, com a última regra gerada. Esses exemplos marcados não influenciarão positivamente nenhuma regra gerada posteriormente, mas influenciarão negativamente as regras que não tenham a mesma classe que os exemplos marcados. Esse módulo de marcação é diferente do módulo utilizado no SLAVE, que era um módulo de exclusão, onde os exemplos com um grau de compatibilidade maior ou igual a um grau de compatibilidade previamente definido eram excluídos dos dados de treinamento do problema. Uma característica importante dos métodos SLAVE e NSLV, é que as regras geradas tem um peso no final, que é o número total de exemplos da classe da regra dividido entre a quantidade exemplos, da mesma classe que a regra, que têm um grau de compatibilidade maior que zero.

O método NSLV pode ser analisado com mais detalhes em (GONZALEZ; RAUL, 2008).

4.6.2. Índice de interpretabilidade semântica

O Índice de Interpretabilidade Semântica (IIS) é baseado na diferença entre o conjunto otimizado e o conjunto original. Para calculá-lo primeiro são definidos os intervalos de variação para cada ponto de cada conjunto j . no caso de Conjuntos Fuzzy (CF) com função triangular, esses intervalos são ilustrados na Figura 4.4 e calculados utilizando as equações 4.1, 4.2 e 4.3.

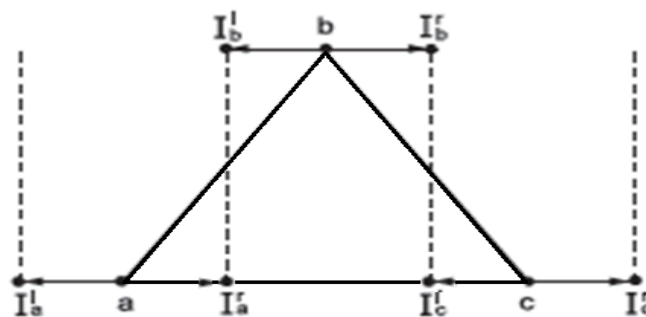


Figura 4.4: Intervalos de Variação para conjuntos fuzzy – Adaptado (GACTO; ALCALA; HERRERA, 2010)

$$\left[I_{a_j}^l, I_{a_j}^r \right] = \left[a_j - \left(\frac{b_j - a_j}{2} \right), a_j + \left(\frac{b_j - a_j}{2} \right) \right] \quad 4.1$$

$$\left[I_{b_j}^l, I_{b_j}^r \right] = \left[b_j - \left(\frac{b_j - a_j}{2} \right), b_j + \left(\frac{c_j - b_j}{2} \right) \right] \quad 4.2$$

$$\left[I_{c_j}^l, I_{c_j}^r \right] = \left[c_j - \left(\frac{c_j - b_j}{2} \right), c_j + \left(\frac{c_j - b_j}{2} \right) \right] \quad 4.3$$

ou seja, cada parâmetro pode variar a metade da distância entre o ponto mais próximo para a direita ou para a esquerda.

O IIS utiliza as seguintes métricas:

- a) Deslocamento dos CF (δ): mede a proximidade dos pontos centrais dos CF otimizados j' aos CF originais j . Esse deslocamento é calculado pelas equações 4.4 e 4.5.

$$\delta_j = \frac{|b_j - b'_j|}{I} \quad 4.4$$

$$I = \frac{(I_{b_j}^r - I_{b_j}^l)}{2} \quad 4.5$$

Esta métrica tenta encontrar o máximo valor de deslocamento do ponto central definido por δ^* . Esse valor é calculado pela equação 4.6.

$$\delta^* = \max_j \{\delta_j\} \quad 4.6$$

O Deslocamento será um sub objetivo para maximizar, definido pela equação 4.7.

$$\text{Maximizar } \delta = 1 - \delta^* \quad 4.7$$

- b) Amplitude Lateral (γ): mede a diferença entre os parâmetros, direito e esquerdo, dos CF otimizados j' e dos CF originais j . Essa amplitude é calculada pelas equações 4.8, 4.9, 4.10, 4.11 e 4.12.

$$\gamma_j = \frac{\min\left\{\frac{EsqS_j}{DirS_j}, \frac{EsqS'_j}{DirS'_j}\right\}}{\max\left\{\frac{EsqS_j}{DirS_j}, \frac{EsqS'_j}{DirS'_j}\right\}} \quad 4.8$$

$$EsqS_j = |a_j - b_j| \quad 4.9$$

$$DirS_j = |b_j - c_j| \quad 4.10$$

$$EsqS'_j = |a'_j - b'_j| \quad 4.11$$

$$DirS'_j = |b'_j - c'_j| \quad 4.12$$

A amplitude Lateral será um subobjetivo a maximizar, definido pela equação 4.13.

$$\text{Maximizar } \gamma = \min\{\gamma_j\} \quad 4.13$$

- c) Similaridade das Áreas (ρ): mede a diferença entre a área dos CF otimizados j' e dos CF originais j . Essa similaridade é calculada pela equação 4.14.

$$\rho_j = \frac{\min\{A_j, A'_j\}}{\max\{A_j, A'_j\}} \quad 4.14$$

onde A_j representa a área do CF original j e A'_j representa a área do CF otimizado j' . A similaridade das áreas será um subobjetivo a maximizar, definido pela equação 4.15.

$$\text{Maximizar } \rho = \min\{\rho_j\} \quad 4.15$$

As três métricas têm um valor entre 0 e 1 e o IIS as une em um objetivo a maximizar, definido como *Geometric Mean of Three Metrics* (GM3M) e calculado pela equação 4.16.

$$\text{Maximizar } GM3M = \sqrt[3]{\delta \times \gamma \times \rho} \quad 4.16$$

O IIS pode ser analisado com mais detalhes em (GACTO; ALCALA; HERRERA, 2010).

4.6.3. Conjuntos *fuzzy* transparentes

A transparência dos CF nas variáveis linguísticas de um SFBR é uma característica importante que ajuda a obter um alto grau de interpretabilidade na BD, além de reduzir o número de regras e condições em uma possível geração da BR a partir dessa BD.

Os CF de uma variável linguística devem possuir as seguintes características para serem considerados CF transparentes:

- 1) O número de CF por variável deve ser moderado.
- 2) Os CF devem ser distinguíveis, ou seja, dois CF não podem ter o mesmo ou quase o mesmo significado linguístico.
- 3) Cada CF deve ser normal.
- 4) O universo de discurso deve estar fortemente coberto.

A condição 1 pode ser satisfeita definindo um número máximo de CF para cada variável, por exemplo 7. A condição 3 pode ser satisfeita normalizando os conjuntos se tiver conjuntos não normais.

As condições 2 e 4 são mais difíceis de satisfazer. Para satisfazer essas condições os conjuntos devem cumprir as seguintes condições:

- a) Simetria: A forma de todos os CF deve ser simétrica. Por exemplo, a gaussiana e a *generalized bell* (gbell) são simétricas por definição. Também outras formas, como a triangular e trapezoidal podem ser desenhadas simetricamente.
- b) Condição α : Em qualquer ponto de interseção de dois CF, o grau de pertinência desse ponto deve ser menor ou igual a α .
- c) Condição γ : No centro de cada CF, nenhum outro CF pode ter um grau de pertinência maior ou igual a γ .
- d) Condição β : Em cada ponto do universo da variável linguística, pelo menos um CF deve ter um grau de pertinência maior ou igual a β .

A Figura 4.5 ilustra um exemplo de CF transparentes, com CF triangulares e simétricos, considerando $\alpha = 0,8$, $\gamma = 0,25$ e $\beta = 0,05$.

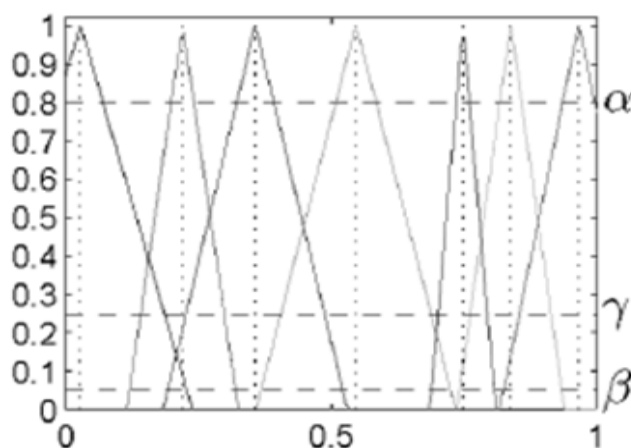


Figura 4.5: Exemplos de conjuntos *fuzzy* transparentes triangulares - Adaptado (PULKKIEN; KOIVISTO, 2010)

Os CF transparentes podem ser analisados com mais detalhes em (PULKKIEN; KOIVISTO, 2010), assim como um exemplo para as funções de pertinência trapezoidal, gaussiana e *gbell*.

4.6.4. Algoritmo evolutivo multiobjetivo não dominado para a geração de regras *fuzzy* na descoberta de subgrupos (Non-dominated Multiobjective Evolutionary Algorithm for Extracting Fuzzy Rules in Subgroup Discovery – NMEEF-SD)

Esse algoritmo é um AEMO que utiliza a técnica de otimização multiobjetivo NSGA-II, apresentado no capítulo 3, para gerar a BR em uma só iteração do NSGA-II. Esse algoritmo utiliza dois objetivos:

- a) Suporte da regra: É definido como a frequência de exemplos corretamente classificados cobertos pela regra.

- b) Originalidade da regra: Determina o equilíbrio entre a cobertura da regra, ou seja, a quantidade de exemplos que tem grau de compatibilidade maior que 0 com a regra, e a precisão da regra.

Depois de executado o AEMO utilizado no NMEEF-SD, as soluções ou regras não dominadas são o conjunto de regras que formam a BR final.

O algoritmo NMEEF-SD pode ser analisado com mais detalhes em (CARMONA; GONZALEZ; JESUS; HERRERA, 2009).

4.6.5. Algoritmo evolutivo multiobjetivo para a geração de regras fuzzy na descoberta de subgrupos (Multiobjective Evolutionary Subgroup Discovery Fuzzy rules – MESDIF-SD)

Este algoritmo é um AEMO que utiliza a técnica de otimização multiobjetivo SPEA2, apresentado no capítulo 3, para gerar a BR em uma só iteração do SPEA2. Esse algoritmo utiliza três objetivos:

- a) Suporte da regra: É definido como a frequência de exemplos corretamente classificados cobertos pela regra.
- b) Confiança da regra: Determina a frequência relativa dos exemplos que satisfazem a regra completa, ou seja, os exemplos que tem um grau de compatibilidade maior que 0 e tem a mesma classe que a regra, e aqueles exemplos que satisfazem apenas o antecedente, ou seja, os exemplos que tem grau de compatibilidade maior que 0 com a regra, sem importar a classe.
- c) Originalidade da regra: Determina o equilíbrio entre a cobertura da regra, ou seja, a quantidade de exemplos que tem grau de compatibilidade maior que 0 com a regra, e a precisão da regra.

Depois de executado o SPEA2, as soluções ou regras não dominadas são o conjunto de regras que formam a BR final.

O algoritmo MESDIF-SD pode ser analisado com mais detalhes em (BERLANGA; JESUS; GONZALEZ; HERRERA; MESONERO, 2006).

4.6.6. Método de otimização genética da BD

O algoritmo apresentado em (PIMENTA, 2009) é responsável por evoluir uma população de BD a fim de encontrar o melhor ajuste para cada CF de cada partição fuzzy, usando uma base de regras fixa definida previamente. O algoritmo obedece ao seguinte esquema:

1. Gerar uma população inicial com N indivíduos.
2. Classificar todos os exemplos do conjunto de treinamento usando as BD da população corrente e calcular o valor de aptidão para cada uma.
3. Aplicar as operações genéticas nas bases de dados.
4. Se atingir o critério de parada, retorna a base de dados com maior valor de aptidão. Caso contrário, volta ao passo 2.

Os cromossomos para esse método genético possuem tamanho fixo e é necessário estabelecer previamente a quantidade de conjuntos que as partições terão. Os conjuntos são codificados com números reais, os quais representam os parâmetros de cada função de pertinência de cada partição. Cada indivíduo da população representa uma base de dados completa.

A função de aptidão desse método genético é definida com base no desempenho da base de dados codificada em cada cromossomo, medido pelo número de exemplos classificados corretamente utilizando a base de regras definida anteriormente e a base de dados codificada no cromossomo. O melhor cromossomo será aquele que maximiza os exemplos classificados corretamente. O AG utiliza o método da roleta para seleção, o operador de cruzamento de um ponto para o cruzamento e a mutação padrão. O critério de parada adotado para este AG é o número máximo de gerações. Esse método pode ser analisado com mais detalhes em (PIMENTA, 2009).

4.7. Considerações finais

Neste capítulo foram mostrados diferentes métodos para o tratamento da questão de aprendizado e otimização de SF com o uso de AG. O sucesso obtido por esses métodos em muitos casos indica que este é um ramo promissor na pesquisa da geração automática da BC, ao mesmo tempo em que se verifica que diversos problemas permanecem em aberto, como o problema que trata este trabalho, de encontrar um equilíbrio entre precisão e interpretabilidade em SFBR.

Para encontrar esse equilíbrio, atualmente é utilizada uma variante dos AG baseados no conceito da dominância das soluções, ou seja, os AGMO. O trabalho aqui proposto dá continuidade a outros trabalhos desenvolvidos na área de SFGBR abordando a questão do uso de AGMO e a abordagem IRL para a geração da BC que será detalhado na próxima seção.

Capítulo 5

GERAÇÃO GENÉTICA

MULTIOBJETIVO DE SISTEMAS *FUZZY*

Este capítulo tem por objetivo apresentar o método proposto neste trabalho para a Geração Genética Multiobjetivo de Sistemas *Fuzzy* (GGMOSF), mais especificamente de um SFBR para resolver problemas de classificação usando Algoritmos Genéticos Multiobjetivo (AGMO) e a abordagem iterativa, *Iterative Rule Learning* (IRL), visando encontrar um equilíbrio entre precisão e interpretabilidade.

O uso dos AGMO neste trabalho foi motivado por esses algoritmos permitirem que o próprio processo de aprendizado busque um equilíbrio entre os objetivos de precisão e interpretabilidade, que vem a ser um dos pontos relevantes investigados nesta proposta.

A escolha da abordagem IRL na geração da BR foi motivada pelas seguintes vantagens (GONZALEZ; PEREZ; 2008):

- A codificação utilizada não exige um cromossomo muito longo, o que poderia deteriorar o desempenho do AG (neste projeto, do AGMO).
- Decompõe o problema em subproblemas mais simples, a serem resolvidos em cada execução do AG, em vez de resolver o problema inteiro em uma só execução do AG (neste projeto, do AGMO).

A metodologia proposta para a GGMOSF está baseada no processo de aprendizado de forma sequencial, ou seja, aprender a BR a partir de uma BD predefinida e depois otimizar a BD utilizando a BR obtida anteriormente. Essa forma sequencial foi escolhida como uma forma de reduzir a complexidade do problema, evitando a geração simultânea dos componentes da BC porque dessa forma é necessário lidar

com um grande espaço de busca que pode tornar o processo de geração difícil e lento.

Nas próximas seções é apresentada a metodologia proposta, assim como cada etapa da mesma.

5.1. Metodologia proposta

A metodologia proposta está baseada no processo de aprendizado de forma sequencial e é ilustrada na Figura 5.1:

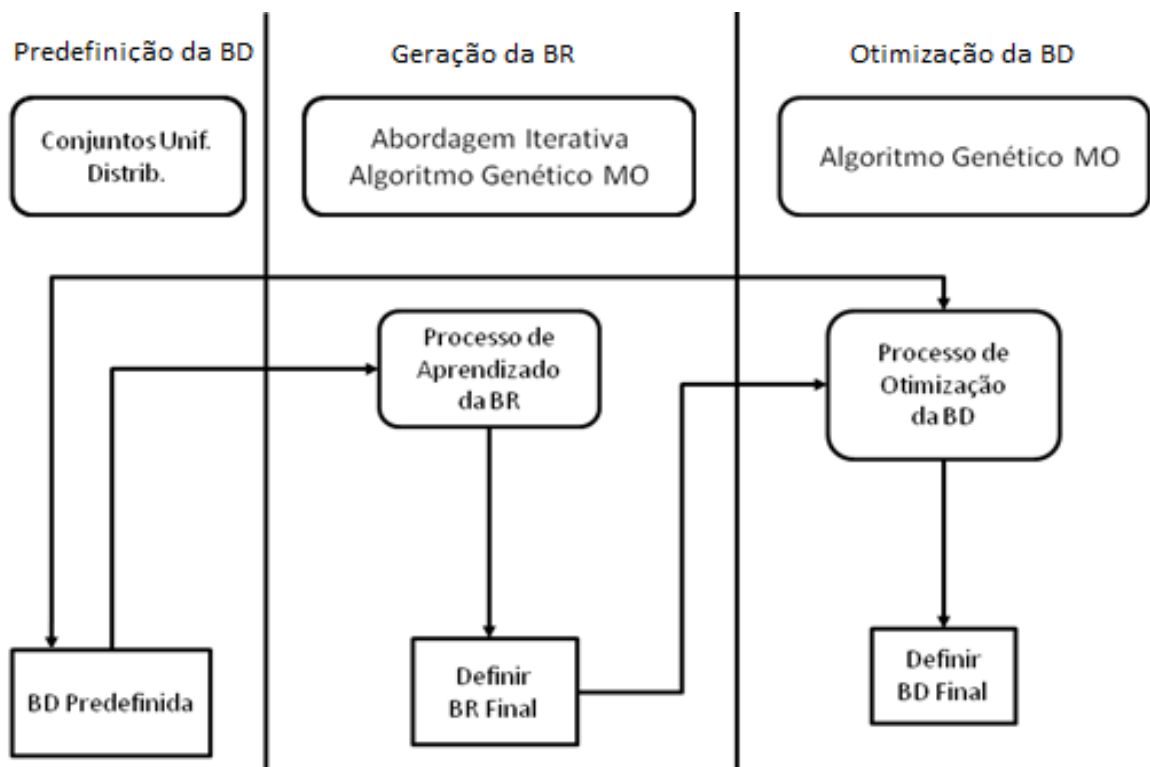


Figura 5.1: Metodologia proposta para a Geração Genética Multiobjetivo de Sistemas Fuzzy (GGMOSF)

Como mostra a Figura 5.1 a metodologia está dividida nas seguintes etapas:

1. Predefinir a Base de Dados (BD): Nesta etapa são criados conjuntos uniformemente distribuídos, segundo (PULKKINEN; KOIVISTO, 2010). Os conjuntos uniformemente distribuídos tem um alto grau de interpretabilidade e esta é uma das formas mais simples para criar conjuntos transparentes, desde que a quantidade de conjuntos seja moderada, considerando-se como moderada entre 3 e 9 conjuntos.
2. Geração multiobjetivo da Base de Regras (BR): Nesta etapa é utilizada a abordagem IRL baseada em um AGMO para gerar regras *fuzzy* capazes de

representar o conhecimento existente no conjunto de exemplos. O AGMO tem dois objetivos, sendo que o primeiro objetivo representa a precisão, que é calculada com base nos conceitos de integridade e consistência das regras, e o segundo representa a interpretabilidade, que é calculada com base na quantidade de condições da regra. Nesta etapa são utilizados os conjuntos uniformemente distribuídos criados na etapa anterior.

3. Otimização da Base de Dados (BD): Nesta etapa um AGMO é utilizado para otimizar os conjuntos *fuzzy* uniformemente distribuídos da primeira etapa. É usada a BR gerada na segunda etapa. O AGMO tem dois objetivos, sendo que o primeiro objetivo representa a precisão, que é calculada com base na taxa de classificação do classificador *fuzzy*, e o segundo objetivo representa a interpretabilidade, que é calculada com base no Índice de Interpretabilidade Semântica (IIS) proposto em (GACTO; ALCALA; HERRERA; 2010), e no conceito de BD transparente proposto em (PULKKINEN; KOIVISTO, 2010) que foram apresentados no capítulo 4.

Com essas três etapas o algoritmo é capaz de gerar um sistema classificador *fuzzy*, com equilíbrio entre precisão e interpretabilidade. A seguir são descritas as etapas 2 e 3 do método proposto. A etapa 1, devido à sua simplicidade, não requer maior detalhamento.

5.2. Geração genética multiobjetivo da base de regras

A partir dos conjuntos de dados que representa os exemplos do problema e a BD predefinida, a abordagem IRL baseada em um AGMO procura uma BR apropriada para classificar corretamente esses exemplos.

Como é utilizada a abordagem IRL, o AGMO desta etapa é responsável por evoluir e encontrar a melhor regra em cada iteração, ou seja, em cada execução do AGMO, que possa equilibrar dois objetivos: maximizar a precisão e maximizar a interpretabilidade. O esquema do método de geração genética multiobjetivo proposto para a geração da BR é ilustrado na Figura 5.2:

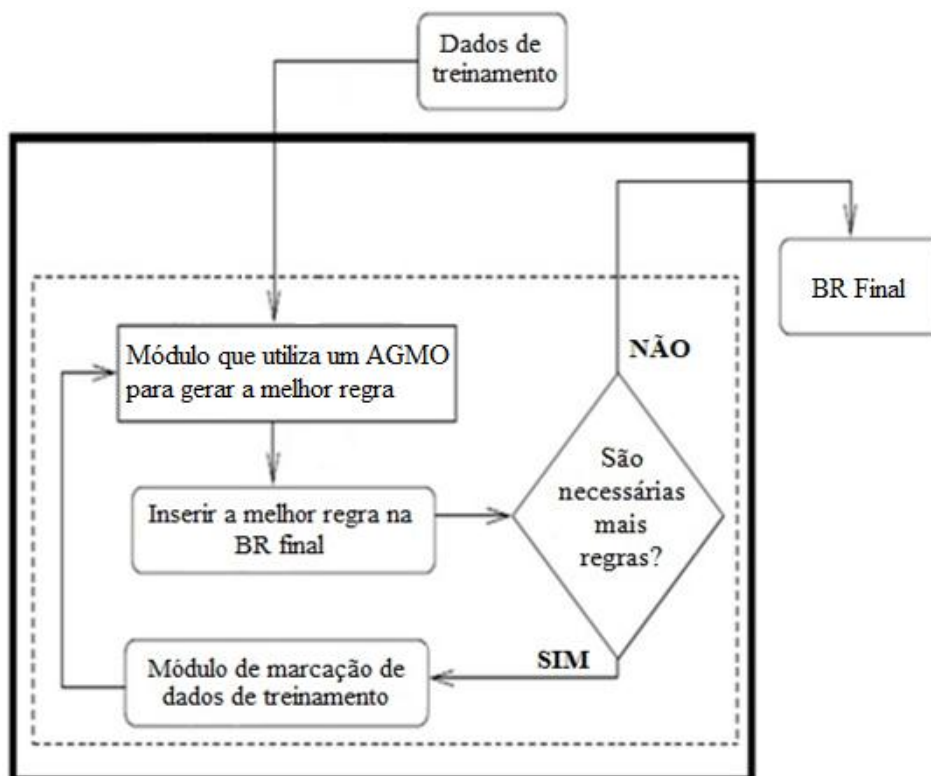


Figura 5.2 – Esquema do método genético multiobjetivo iterativo proposto para a geração da BR

Este método proposto é baseado no método NSLV (GONZALEZ; PEREZ, 2008), apresentado no capítulo 3. Seguindo esse esquema, são utilizados os dados de treinamento no módulo que utiliza um AGMO para gerar a melhor regra *fuzzy* de formato Se→Então. Para este módulo, foram implementados dois algoritmos, sendo que o primeiro algoritmo se baseia na técnica de otimização multiobjetivo NSGA-II e o segundo algoritmo se baseia na técnica de otimização multiobjetivo SPEA2. As técnicas de otimização multiobjetivo, NSGA-II e SPEA2 foram apresentadas no capítulo 4.

Uma vez gerada a melhor regra, ela é inserida na BR final. Se não são necessárias mais regras, o método termina, e o resultado obtido é a BR final. O critério adotado para determinar se não são necessárias mais regras é que todas as regras não dominadas da última iteração, candidatas a serem inseridas na BR, cumpram pelo menos uma das seguintes condições:

- A quantidade de exemplos corretamente classificados pela BR com a inserção da nova regra é menor ou igual à quantidade de exemplos erradamente classificados.

- A quantidade de exemplos corretamente classificados pela BR com a inserção da nova regra é menor ou igual à quantidade de exemplos corretamente classificados mais um, antes da inserção da nova regra.
- O antecedente da nova regra já está na BR final obtida anteriormente.

Se alguma das regras não dominadas não cumpre nenhuma dessas condições ela é inserida na BR final, desde que seja verificada a condição definida pela equação 5.5, que será definida posteriormente.

Caso ocorra a inserção de uma nova regra, é executado o módulo de marcação de dados de treinamento, onde todos os dados de treinamento classificados com um grau de compatibilidade maior que um valor previamente definido, por exemplo, 0,8, são marcados, e assim, não são usados como dados de treinamento na iteração seguinte.

Para obter a taxa de classificação da BR final, foi utilizado o método de raciocínio clássico, descrito no capítulo 2.

A seguir será apresentado com mais detalhes o módulo que utiliza um AGMO para gerar a melhor regra. Esse módulo foi implementado utilizando duas técnicas de otimização multiobjetivo, o NSGA-II e o SPEA2.

No NSGA-II, inicialmente existe uma população de regras onde são aplicados operadores genéticos (seleção por torneio, cruzamento uniforme e mutação simples) de acordo à dominância das soluções para gerar uma segunda população de regras. Ambas as populações se unem e, nessa população resultante, as regras são classificadas em fronteiras de acordo a sua dominância, entre dois objetivos, precisão e interpretabilidade, que são detalhados posteriormente. As melhores regras, ou seja, as regras que pertençam às primeiras fronteiras e estejam mais distantes das outras soluções, formarão parte da população seguinte; as piores soluções são rejeitadas. O processo continua até atingir um número máximo de gerações.

O NSGA-II utilizado no módulo de geração da melhor regra é ilustrado na Figura 5.3.

Variáveis: P : População pai Q : População filha T : Tamanho fixo para P e Q F_i : Conjunto de regras na fronteira i n : Número de geração atual N : Máximo número de gerações

1. Gerar a população inicial P_0 e fazer $Q_0 = \{ \}$
2. Atribuir $n = 0$
3. Calcular para cada regra em P_n o objetivo 1 (precisão) e o objetivo 2 (interpretabilidade)
4. Calcular para cada regra em P_n o ranking e a distância de multidão
5. Realizar a seleção por torneio, o cruzamento uniforme e a mutação simples em P_n para gerar a população filha Q_n
6. Fazer $R_n = P_n \cup Q_n$
7. Calcular, para cada regra, em R_n o objetivo 1 (precisão) e o objetivo 2 (interpretabilidade)
8. Calcular para cada regra em R_n o ranking
9. Realizar a ordenação por não dominância em R_n , gerando $F_i, i = 1, \dots, v$ em R_n
10. Criar $P_{n+1} = \{ \}$
11. Fazer $i = 0$
12. Enquanto $|P_{n+1}| + |F_i| \leq T$
13. Copiar as regras de F_i em P_{n+1}
14. Fazer $i = i + 1$
15. Calcular as distâncias de multidão para cada regra em F_i
16. Ordenar F_i decrescentemente de acordo as distâncias de multidão de cada regra
17. Copiar as primeiras $T - |P_{n+1}|$ regras ordenadas de F_i para P_{n+1}
18. Fazer $n = n + 1$
19. Se $n \geq N$ então pare
20. Senão voltar ao passo 3.

Figura 5.3: Algoritmo do método proposto para a geração da melhor regra utilizando NSGA-II

Para gerar a melhor regra utilizando o SPEA2, inicialmente existem duas populações de regras. Ambas as populações se unem e, dessa população resultante são escolhidas as melhores regras, ou seja, as regras que dominem mais regras, que sejam dominadas por menos regras e estejam mais distantes das outras regras. As melhores regras selecionadas formam parte da primeira população da iteração seguinte e as piores regras são rejeitadas. Na primeira população são aplicados operadores genéticos (seleção por torneio, cruzamento uniforme e mutação simples) para gerar a segunda população. O processo continua até atingir um número máximo de gerações. Os valores da aptidão robusta (s), aptidão preliminar (r), densidade (d) e aptidão final (F) foram detalhados no capítulo 3.

O SPEA2 utilizado no módulo de geração da melhor regra é ilustrado na Figura 5.4:

Variáveis:

P : População interna

E : População externa

T : Tamanho fixo para E

N : Máximo número de gerações

n : Número de geração atual

1. Gerar a população inicial P_0 e $E_0 = \{ \}$
2. Fazer $n = 0$
3. $Q_n = P_n \cup E_n$
4. Fazer $E_n = \{ \}$
5. Para cada regra j em Q_n
6. Calcular o objetivo 1 (precisão) e o objetivo 2 (interpretabilidade)
7. Para cada regra j em Q_n
8. Calcular os valores s_j , r_j , d_j e F_j
9. Copiar as regras não dominadas de Q_n em E_n
10. Se $|E_n| < T$
11. Ordenar Q_n conforme a F_j
12. Copiar as melhores $|E_n| - N$ regras j de Q_n tal que $F_j \geq 1$
13. Se $|E_n| > T$
14. Reduzir E_n utilizando o algoritmo de corte
15. Se $n > N$ então pare
16. Aplicar seleção por torneio, cruzamento uniforme e mutação simples sobre E_n para gerar P_{n+1}
17. Fazer $E_{n+1} = E_n$
18. Fazer $n = n + 1$
19. Voltar ao passo 3.

Figura 5.4: Algoritmo do método proposto para a geração da melhor regra utilizando SPEA2

Além dos componentes próprios de cada algoritmo, como a distância de multidão e o algoritmo de corte, já descritos no capítulo 4, nos algoritmos apresentados baseados em NSGA-II e SPEA2 respectivamente, esses algoritmos compartilham os seguintes componentes:

1) Codificação das regras

É preciso estabelecer a forma de codificar as possíveis regras ou soluções para que o AGMO possa manipulá-las. Foi utilizada a codificação apresentada em (ABADEH; HABIBI, 2007) onde os cromossomos possuem tamanho fixo e cada gene tem um valor numérico, que representa o índice dos conjuntos *fuzzy* que aparecem na parte antecedente e a classe que aparece no consequente. A

Figura 5.5 ilustra a codificação de uma regra com três antecedentes e a classe consequente.

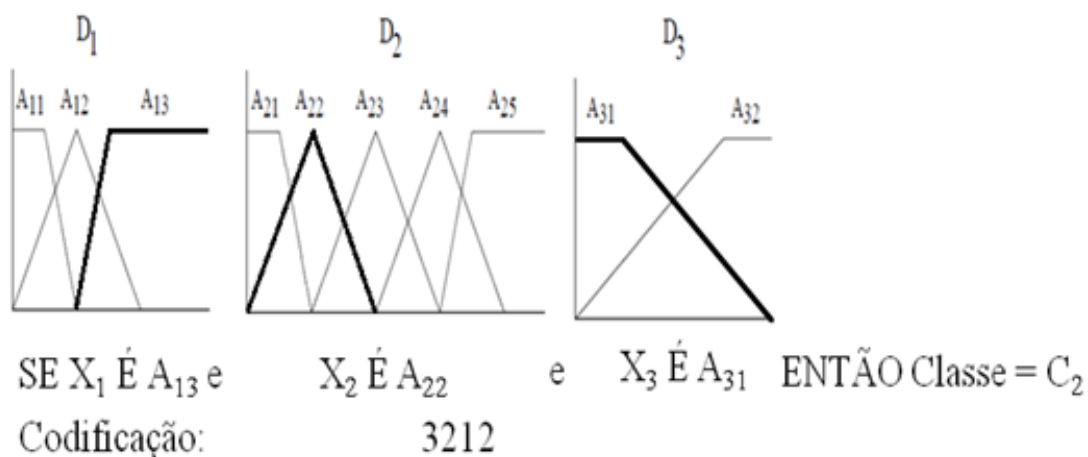


Figura 5.5: Codificação das regras – Adaptado (ABADEH; HABIBI, 2007)

No caso em que uma condição não aparece na regra, chamada de condição *don't care*, ela é codificada pelo valor 0 (zero), como mostra a Figura 5.6, onde a segunda condição é *don't care*.

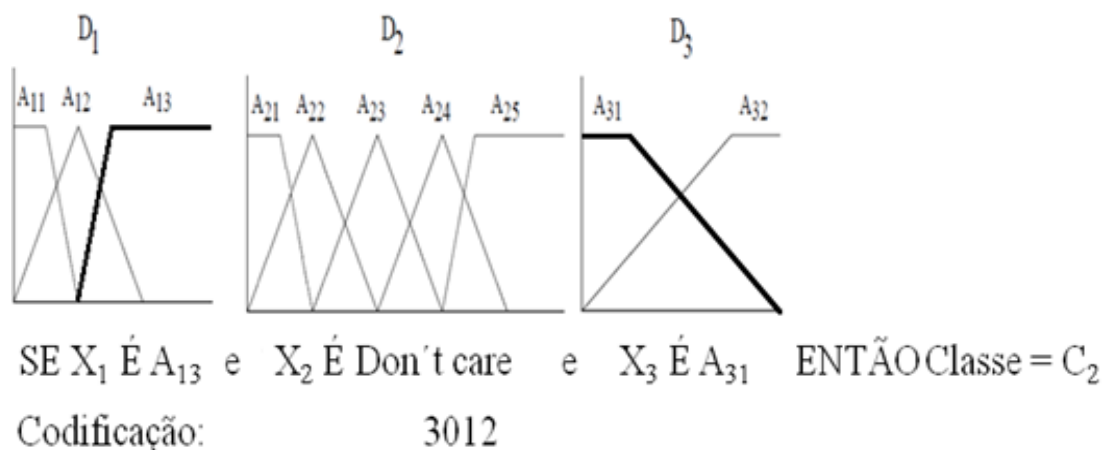


Figura 5.6: Codificação das regras com condições *don't care* -- Adaptado (ABADEH; HABIBI, 2007)

2) População inicial

A população inicial é gerada de forma aleatória, sendo a quantidade de regras para cada classe proporcional ao número de exemplos dessa classe nos dados de treinamento. Todas as regras são geradas aleatoriamente considerando a quantidade de conjuntos *fuzzy* para cada atributo, a condição *don't care* e a quantidade de classes, mas o antecedente de cada regra na população inicial deve ter um grau de compatibilidade maior que 0 com pelo menos um exemplo.

3) Função de aptidão

Os AGMO utilizados têm uma função de aptidão para cada um dos dois objetivos, sendo um representando a precisão e o outro a interpretabilidade. Neste caso um objetivo será maximizar a precisão e o outro objetivo será maximizar a interpretabilidade.

A função de aptidão que representa o objetivo da precisão da regra é baseada em duas condições que deve ter uma BR que é obtida por um algoritmo de aprendizado: condição de integridade e condição de consistência (GONZALEZ; PEREZ, 1999) que são definidas a seguir:

- a) Condição de Integridade: A condição de integridade define que cada exemplo de uma determinada classe deve ser verificado por uma regra da mesma classe.
- b) Condição de Consistência: A condição de consistência define que se um exemplo satisfaz a descrição de uma classe, então o exemplo não pode ser membro dos dados de treinamento para gerar uma regra de outra classe.

Essas definições estão originalmente associadas a uma BR ou conjunto de regras. Como o método proposto para geração de regras está baseado na abordagem IRL, em que cada cromossomo representa apenas uma regra, é necessário definir esses conceitos para cada regra. Por isso, é proposta a equação 5.1, que calcula a precisão de uma regra i pela multiplicação da integridade da regra i pela consistência da regra i . Assim, quanto maior for a integridade e a consistência da regra, maior será a precisão da regra.

$$Precisão(R_i) = Integridade(R_i) \times Consistência(R_i) \quad 5.1$$

$$Integridade(R_i) = \frac{1 + \sum_{e_j \in C_{R_i}} Compat(R_i, e_j)}{1 + \sum_{e_j \notin C_{R_i}} Compat(R_i, e_j)} \quad j = 1, \dots, p \quad 5.2$$

Onde:

p : Quantidade de exemplos

e_j : exemplo j

C_{R_i} : Classe da regra R_i

$Compat(R_i, e_j)$: Compatibilidade entre a regra R_i e o exemplo e_j calculado pela equação 2.22.

$$Consist\ência(R_i) = \frac{1 + QEm_{R_i}^+}{1 + QEm_{R_i}^-} \quad 5.3$$

Onde:

$QEm_{R_i}^+$: Quantidade de exemplos que têm a mesma classe que a regra R_i e um alto grau de compatibilidade com a regra R_i .

$QEm_{R_i}^-$: Quantidade de exemplos que não têm a mesma classe que a regra R_i e que têm um alto grau de compatibilidade com a regra R_i .

A integridade de uma regra é calculada pela divisão do somatório dos graus de compatibilidade dos exemplos que têm a mesma classe que a regra pelo somatório dos graus de compatibilidade dos exemplos que não têm a mesma classe que a regra. Então, quanto maior for a quantidade de exemplos verificados pela regra, ou seja, a quantidade de exemplos que tenham a mesma classe da regra com um grau de compatibilidade maior que zero, e quanto maior forem os graus de compatibilidade desses exemplos com a regra, maior será a integridade da regra. Assim como, quanto menor for a quantidade de exemplos não verificados pela regra, ou seja, exemplos que não tenham a mesma classe que a regra com um grau de compatibilidade maior que zero, e quanto menores forem os graus de compatibilidade desses exemplos com a regra, menor será a integridade da regra.

A consistência de uma regra é calculada pela divisão da quantidade de exemplos que têm a mesma classe da regra e um grau de compatibilidade alto com essa regra, pela quantidade de exemplos que não têm a mesma classe da regra e tem um alto grau de compatibilidade com essa regra. Assim, quanto maior for a quantidade de exemplos que têm a mesma classe da regra e têm um alto grau de compatibilidade com essa regra maior será a consistência, e quanto menor for a quantidade de exemplos que têm a mesma classe da regra e têm um alto grau de compatibilidade com essa regra menor será a consistência .

Como foi mencionado anteriormente, os exemplos que são classificados corretamente com um alto grau de compatibilidade, por alguma regra na BR, não serão considerados para os numeradores das equações para o cálculo da integridade e consistência da regra na iteração seguinte, ou seja, nas equações 5.2 e 5.3.

A função de aptidão que representa o objetivo da interpretabilidade é baseada na quantidade de condições *don't care* em cada regra. O valor máximo para esta função é a quantidade de variáveis ou condições menos um. A interpretabilidade também é um objetivo a ser maximizado no antecedente da regra, porque segundo (GONZALEZ; PEREZ, 2008) quanto menor for o número de condições nas regras e na BR final, mais interpretáveis serão as regras e a BR. A função da interpretabilidade é definida na equação 5.4.

$$\text{Interpretabilidade}(R_i) = \text{don't care}_i \quad 5.4$$

Onde:

R_i : Índice da regra i .

don't care_i : Quantidade de condições *don't care* na regra R_i , ou seja, a quantidade de 'zeros' na codificação das regras.

Na função de interpretabilidade, não é considerada a quantidade de regras porque, na abordagem IRL, as regras são avaliadas individualmente. A quantidade de regras da BR é definida fora do módulo da geração da melhor regra.

4) Operadores genéticos

Como foi explicado no capítulo anterior, os operadores genéticos dos AGMO são iguais aos operadores genéticos dos AG.

A seleção é a primeira operação a ser empregada e utiliza o método do torneio para escolher os indivíduos com base na dominância das soluções. No método que utiliza o NSGA-II são selecionadas aleatoriamente duas regras e é escolhida a regra com maior ranking. Se os rankings forem iguais, é escolhida a regra com maior distância de multidão. Se as distâncias de multidão forem também iguais, então é escolhida a regra com a precisão maior, e se as precisões forem iguais,

é escolhida a regra com a maior quantidade de condições *don't care*. Finalmente, se as quantidades de condições *don't care* também forem iguais, a regra é escolhida aleatoriamente.

No método que utiliza o SPEA2 também são selecionadas aleatoriamente duas regras, mas é escolhida a regra com menor valor de F , valor que representa a aptidão na técnica SPEA2. Se os valores de F forem iguais, a regra é escolhida aleatoriamente.

A segunda operação a ser realizada é o cruzamento. Neste método foi utilizado o cruzamento uniforme. A máscara utilizada é gerada aleatoriamente em cada cruzamento.

Por fim, a mutação simples é aplicada para alterar um gene, ou seja, o valor de uma condição ou classe da regra, considerando o intervalo de valores possíveis.

5) Condição de parada e seleção da melhor regra

O critério de parada adotado para os dois métodos genéticos multiobjetivo propostos é o número máximo de gerações. Ao atingir esse número, o método multiobjetivo retornará o conjunto de regras não dominadas. Uma vez encontrado esse conjunto de regras ou soluções, pode-se escolher aquela solução que melhor satisfaça algum critério de preferência (ABRAHAM; JAIN; GOLDBERG, 2004). Neste caso, como o módulo para gerar a melhor regra requer que seja escolhida só uma regra do conjunto de regras não dominadas, é escolhida a regra que, uma vez inserida na BR final, faça com que essa BR tenha a maior proporção entre a quantidade de exemplos classificados corretamente e a quantidade exemplos classificados erradamente. Esse valor é representado na equação 5.5.

$$\text{Máximo} \left(\frac{QEC_r^+}{1 + QEC_r^-} \right) \quad 5.5$$

Onde:

QEC_r^+ : Quantidade de exemplos classificados corretamente pela BR gerada adicionando a regra R_r .

QEC_r^- : Quantidade de exemplos classificados erradamente pela BR gerada adicionando a regra R_r .

Se duas ou mais soluções obtêm o máximo valor para a equação 5.5, uma dessas regras é escolhida aleatoriamente.

5.3. Otimização genética multiobjetivo da base de dados

Ao término da etapa da geração da BR, seguindo o método sequencial utilizado, a BD definida na etapa 1 é otimizada usando a BR gerada na etapa 2. Para realizar este processo é utilizado um AGMO, que tenta também encontrar um equilíbrio entre precisão e interpretabilidade.

O AGMO desta etapa é responsável por evoluir uma nova BD a fim de encontrar o melhor ajuste para cada conjunto de cada partição *fuzzy*. Esse AGMO tem também dois objetivos, a precisão representada pela taxa de classificação e a interpretabilidade representada pelo Índice de Interpretabilidade Semântica (IIS) apresentado em (GACTO; ALCALA; HERRERA; 2010) e explicado no capítulo 4.

Para esta etapa, foram implementados dois algoritmos para otimizar a BD, sendo que o primeiro algoritmo se baseia na técnica de otimização multiobjetivo NSGA-II, e o segundo algoritmo se baseia na técnica de otimização multiobjetivo SPEA2.

Para otimizar a BD utilizando o NSGA-II, é utilizado o mesmo algoritmo utilizado para gerar a BR. A diferença é que, nesta etapa, os cromossomos ou solução não representam uma regra, mas sim uma BD inteira. O NSGA-II utilizado na otimização da BD é ilustrado na Figura 5.7.

Variáveis: P : População pai Q : População filha T : Tamanho fixo para P e Q F_i : Conjunto de BD na fronteira i n : Número de geração atual N : Máximo número de gerações

1. Gerar a população inicial P_0 e fazer $Q_0 = \{ \}$
2. Atribuir $n = 0$
3. Calcular para cada BD em P_n o objetivo 1 (precisão) e o objetivo 2 (interpretabilidade)
4. Calcular para cada BD em P_n o ranking e a distância de multidão
5. Realizar a seleção por torneio, o cruzamento uniforme e a mutação simples em P_n para gerar a população filha Q_n
6. Fazer $R_n = P_n \cup Q_n$
7. Calcular para cada BD em R_n o objetivo 1 (precisão) e o objetivo 2 (interpretabilidade)
8. Calcular para cada BD em R_n o ranking
9. Realizar a ordenação por não dominância em R_n , gerando $F_i, i = 1, \dots, v$ em R_n
10. Criar $P_{n+1} = \{ \}$
11. Fazer $i = 0$
12. Enquanto $|P_{n+1}| + |F_i| \leq T$
13. Copiar as BD de F_i em P_{n+1}
14. Fazer $i = i + 1$
15. Calcular as distâncias de multidão para cada BD em F_i
16. Ordenar F_i decrescentemente de acordo as distâncias de multidão de cada BD
17. Copiar as primeiras $T - |P_{n+1}|$ BD ordenadas de F_i para P_{n+1}
18. Fazer $n = n + 1$
19. Se $n \geq N$ então pare
20. Senão voltar ao passo 3.

Figura 5.7: Algoritmo do método proposto para a otimização da BD utilizando NSGA-II

Para otimizar a BD utilizando o SPEA2, também é utilizado o mesmo algoritmo que foi utilizado para gerar a BR. A diferença é que, nesta etapa, os cromossomos ou solução não representam uma regra, mas sim uma BD inteira. O SPEA2 utilizado para otimizar a BD é ilustrado na Figura 5.8:

Variáveis: P : População interna E : População externa T : Tamanho fixo para N N : Máximo número de gerações n : Número de geração atual

1. Gerar a população inicial P_0 e $E_0 = \{ \}$
2. Atribuir $n = 0$
3. $Q_n = P_n \cup E_n$
4. Fazer $E_n = \{ \}$
5. Para cada BD_j em Q_n
6. Calcular o objetivo 1 (precisão) e o objetivo 2 (interpretabilidade)
7. Para cada BD_j em Q_n
8. Calcular os valores s_j , r_j , d_j e F_j
9. Copiar as BD não dominadas de Q_n em E_n
10. Se $|E_n| < T$
11. Ordenar Q_n conforme a F_j
12. Copiar as melhores $|E_n| - N$ BD_j de Q_n tal que $F_j \geq 1$
13. Se $|E_n| > T$
14. Reduzir E_n utilizando o algoritmo de corte
15. Se $n > N$ então pare
16. Aplicar seleção por torneio, cruzamento uniforme e mutação simples sobre E_n para gerar P_{n+1}
17. Fazer $E_{n+1} = E_n$
18. Fazer $n = n + 1$
19. Voltar ao passo 3.

Figura 5.8: Algoritmo do método proposto para a otimização da BD utilizando SPEA2

Além dos componentes próprios de cada algoritmo, como a distância de multidão e o algoritmo de corte, nos algoritmos apresentados baseados em NSGA-II e SPEA2 respectivamente, esses algoritmos compartilham os seguintes componentes:

1) Codificação da base de dados

Foi definido para o AGMO desta etapa que os cromossomos possuem tamanho fixo e é mantida a mesma quantidade de conjuntos das partições iniciais da distribuição uniforme. Os conjuntos são codificados com números reais, os quais representam os parâmetros de cada função de pertinência dos conjuntos para cada variável. Neste caso as funções de pertinência são representadas por três valores porque as funções de pertinência utilizadas são triangulares.

Cada indivíduo da população representa uma BD completa. Foi utilizada a representação adotada em (GACTO; ALCALA; HERRERA, 2009), ilustrada nas equações 5.6 e 5.7:

$$C_j = C_1 C_2 \dots C_n \quad 5.6$$

$$C_i = (a_1^i, b_1^i, c_1^i, \dots, a_{m^i}^i, a_{m^i}^i, a_{m^i}^i) \quad i = 1, \dots, n \quad 5.7$$

Onde:

n : quantidade de variáveis ou condições

j : Índice da BD.

m^i : Quantidade de conjuntos da variável i .

2) População inicial

O primeiro cromossomo da população inicial é representado pelas funções de pertinência uniformemente distribuídas geradas na etapa 1. Os outros cromossomos são gerados aleatoriamente, seguindo os limites para cada parâmetro dos conjuntos com base nos conjuntos uniformemente distribuídos, apresentados em (GACTO; ALCALA; HERRERA, 2010) e explicados no capítulo anterior.

Como a otimização dos conjuntos *fuzzy* também considera a interpretabilidade, são gerados conjuntos *fuzzy* transparentes para cada BD ou solução. A definição de conjuntos transparentes foi apresentada no capítulo anterior.

3) Função de aptidão

Os AGMO utilizados têm uma função de aptidão para cada um dos dois objetivos, um representa a precisão e o outro a interpretabilidade.

A função de aptidão que representa o objetivo da precisão é definido pela maximização da taxa de classificação obtida mediante o método de raciocínio clássico que utiliza da BR fixa gerada na etapa 2 e a BD que representa uma solução do AGMO.

A função de aptidão que representa o objetivo da interpretabilidade é definido pela maximização do IIS, apresentado em (GACTO; ALCALA; HERRERA, 2010) e explicado no capítulo 4, para cada solução ou BD no AGMO.

4) Operadores genéticos

Como explicado no capítulo 3, os operadores genéticos, seleção, cruzamento e mutação dos AGMO são iguais aos operadores genéticos dos AG.

A seleção é a primeira operação a ser empregada e utiliza o método do torneio para escolher os indivíduos com base na dominância das soluções. No método que utiliza o NSGA-II são selecionadas aleatoriamente duas BD e é escolhida a BD com maior ranking, se os rankings forem iguais, é escolhida a BD com maior distância de multidão, se as distâncias de multidão forem iguais, é escolhida a BD com a qual a BR obtém precisão maior. Se as precisões forem iguais, é escolhida a BD com IIS maior, e se o IIS das BD também forem iguais, a BD é escolhida aleatoriamente.

No método que utiliza SPEA2 também são selecionadas aleatoriamente duas BD, mas é escolhida a BD com menor valor de F , valor que representa a aptidão na técnica SPEA2. Se os valores de F forem iguais, a BD é escolhida aleatoriamente.

A segunda operação a ser realizada é o cruzamento. Foi utilizado o cruzamento de um 1 ponto. Tanto pares de cromossomos a serem cruzados como o ponto de cruzamento são escolhidos aleatoriamente. Para manter os conjuntos *fuzzy* transparentes, o ponto escolhido é um ponto entre as codificações dos conjuntos *fuzzy* de uma variável.

Por fim, a mutação simples é aplicada para alterar um conjunto de genes que definem os parâmetros de um conjunto *fuzzy* de uma variável, para um novo conjunto escolhido aleatoriamente dentro dos parâmetros possíveis e que mantenha a transparência dos conjuntos dessa BD.

5) Condição de parada e seleção da melhor BD

O critério de parada adotado para o AGMO de otimização da BD é o número máximo de gerações. Ao atingir esse número o AGMO retornará um conjunto de BD com objetivos não dominados. Desse conjunto de BD é escolhida a BD que tenha os valores dos objetivos mais próximos ao ponto médio da curva gerada pelos valores dos objetivos das BD não dominadas. Para encontrar esse ponto médio, são calculados o valor médio entre o valor máximo e o valor mínimo do objetivo precisão, e o valor médio entre o valor máximo e o valor mínimo do

objetivo interpretabilidade. Esses dois valores médios formam o ponto médio. Depois são calculadas as distâncias euclidianas entre esse ponto médio e os outros pontos determinados pelos valores dos dois objetivos de todas as BD não dominadas. A solução que estiver mais próxima do ponto médio dos objetivos será a BD escolhida. Se duas ou mais BD têm a mesma distância mínima, é escolhida aleatoriamente uma BD desse grupo.

5.4. Considerações finais

Neste capítulo foi apresentada a metodologia proposta neste trabalho para geração genética multiobjetivo de um sistema classificador *fuzzy* baseado em regras. O método proposto para a geração genética multiobjetivo da BR utilizando a abordagem IRL e o método proposto para a otimização genética multiobjetivo da BD foram descritos detalhadamente.

Para validar a metodologia e os métodos propostos, foram feitos vários experimentos e análises, que são mostrados no capítulo a seguir.

Capítulo 6

EXPERIMENTOS E ANÁLISES DE RESULTADOS

Neste capítulo são apresentados os experimentos realizados com o método proposto para a geração automática de sistemas *fuzzy* apresentado no capítulo 5 e são discutidos os resultados obtidos para a avaliação do método em comparação com outros métodos propostos na literatura.

6.1. Descrição dos conjuntos de dados

Com a finalidade de testar a metodologia e os métodos previamente apresentados de geração da BR e otimização da BD usando AGMO e a abordagem IRL, foram realizados diferentes experimentos utilizando alguns conjuntos de dados obtidos no *UCI Repository of Machine Learning Databases* (FRANK; ASUNCION, 2010).

Os conjuntos de dados utilizados durante os testes são apresentados na Tabela 6.1, onde são mostrados a quantidade de exemplos, o número de variáveis e o número de classes para cada conjunto de dados.

Tabela 6.1: Conjuntos de dados utilizados nos experimentos

Conjunto de Dados	Quantidade de Exemplos	Número de Variáveis	Número de Classes
iris	150	4	3
wine	178	13	3
thyroid	215	5	3
heart_s	270	13	2
sonar	208	60	2
bupa	345	6	2
breast_can	569	30	2

Os domínios de cada um dos atributos dos conjuntos de dados estão representados por valores numéricos contínuos e precisam ser granularizados em um número de conjuntos *fuzzy* que representem os valores linguísticos que cada atributo pode assumir, a fim de permitir a generalização do conhecimento. Nos testes, cada atributo foi granularizado em cinco conjuntos *fuzzy*, uniformemente distribuídos, conforme apresentado na seção 5.1.

6.2. Experimentos realizados

A Tabela 6.2 ilustra os valores dos parâmetros dos AGMO NSGA-II e SPEA2, utilizados no método proposto. Para o NSGA-II o tamanho da população representa o tamanho fixo para P e Q , e para o SPEA2 o tamanho da população representa o tamanho fixo para E e também o tamanho da população P . Os valores iniciais utilizados foram escolhidos com base em experimentos preliminares que variaram as taxas utilizadas em experimentos similares. Foi utilizado o método de validação cruzada de 10 partições (CHANG; LUO 1992) (KOHAVI, 1995).

Tabela 6.2: Parâmetros dos AGMO

Parâmetro	Valor
Tamanho da população	50
Taxa de cruzamento	1,0
Taxa de mutação	0,2
Número de gerações	500

Conforme explicado no capítulo anterior, foram implementados quatro algoritmos para a geração genética multiobjetivo de classificadores *fuzzy*. Dois dos algoritmos implementados fazem a geração da BR: o primeiro utiliza a abordagem IRL e a técnica de otimização multiobjetivo NSGA-II, chamado de IRL-NSGA-II, e o segundo utiliza a abordagem IRL e a técnica de otimização multiobjetivo SPEA2, chamado de IRL-SPEA2. Os outros algoritmos implementados fazem otimização da BD: o primeiro utiliza a técnica de otimização multiobjetivo NSGA-II, chamado de OM-NSGA-II, e o segundo utiliza a técnica de otimização multiobjetivo SPEA2, chamado de OM-SPEA2.

6.2.1. Resultados para os métodos propostos de geração genética multiobjetivo da base de regras

Para os métodos de geração da BR, IRL-NSGA-II e IRL-SPEA2, os resultados são apresentados nas Tabelas 6.3 e 6.4 respectivamente. Pode-se observar os Conjuntos de Dados (CD), o Erro Majoritário (EM), a taxa de erro média, o número médio de regras do classificador e o número médio de condições totais da BR, assim como o desvio padrão para cada um desses resultados, entre parêntesis.

Tabela 6.3: Resultados obtidos com o método IRL-NSGA-II

CD	EM	Taxa de Erro	Nº médio de regras	Condições
iris	66,7	2,7 (0,00107)	4,0 (0,00000)	6,0 (0,00000)
wine	60,1	7,3 (0,00223)	6,1 (0,00000)	9,1 (0,00000)
thyroid	30,2	3,8 (0,00127)	4,9 (0,00000)	7,4 (0,00000)
heart_s	44,4	25,7 (0,00560)	20,6 (0,38471)	64,5 (1,69988)
sonar	46,4	25,9 (0,01029)	8,2 (0,85182)	18,1 (1,61914)
bupa	42,0	40,4 (0,00465)	6,6 (0,11662)	11,6 (0,32000)
breast_can	37,2	4,8 (0,00080)	6,3 (0,04000)	8,5 (0,04000)

Tabela 6.4: Resultados obtidos com o método IRL-SPEA2

CD	EM	Taxa de Erro	Nº médio de regras	Condições
iris	66,7	2,7 (0,00107)	4,0 (0,00000)	6,0 (0,00000)
wine	60,1	7,4 (0,00197)	6,1 (0,00000)	9,1 (0,00000)
thyroid	30,2	3,8 (0,00127)	4,3 (0,00000)	6,6 (0,00000)
heart_s	44,4	23,2 (0,00437)	16,7 (0,25768)	49,8 (1,06395)
sonar	46,4	27,2 (0,01209)	7,8 (0,45782)	15,8 (0,89219)
bupa	42,0	38,3 (0,00574)	6,1 (0,04000)	10,6 (0,12000)
breast_can	37,2	5,2 (0,00077)	6,2 (0,09798)	9,2 (0,00160)

6.2.2. Resultados para os métodos propostos de otimização genética multiobjetivo da base de dados

Para a otimização da BD, os resultados obtidos são apresentados nas Tabelas 6.5 e 6.6, para os métodos multiobjetivo de otimização da BD, OM-NSGA-II e OM-SPEA2. O método OM-NSGA-II usa a BR gerada pelo método proposto IRL-NSGA-II, e o método OM-SPEA2 usa a BR gerada pelo método proposto IRL-SPEA2.

As tabelas 6.5 e 6.6 mostram a média da taxa de erro obtida antes e depois da etapa de otimização da BD (Erro-ant) e (Erro-apos), além da variação entre essas taxas (VE). Na última coluna das tabelas é mostrada a média do Índice de Interpretabilidade Semântica (IIS).

Esses resultados indicam que, em todos os conjuntos de dados, exceto no íris, a taxa de erro diminui com a otimização da BD.

Tabela 6.5: Resultados obtidos com o método OM-NSGA-II

CD	Erro-ant	Erro-apos	VE	IIS
iris	2,7 (0,00107)	2,7 (0,00107)	0,0 (0,0000)	1,00 (0,00000)
wine	7,3 (0,00223)	4,6 (0,00311)	2,7 (0,00553)	0,76 (0,05323)
thyroid	3,8 (0,00127)	3,2 (0,00222)	0,6 (0,00381)	0,94 (0,04183)
heart_s	25,7 (0,00560)	19,7 (0,00489)	6,0 (0,00502)	0,75 (0,08955)
sonar	25,9 (0,01029)	23,5 (0,00910)	2,4 (0,00830)	0,73 (0,14662)
bupa	40,4 (0,00465)	35,7 (0,00801)	4,7 (0,01239)	0,60 (0,08944)
breast_can	4,8 (0,00080)	4,3 (0,00107)	0,5 (0,00238)	0,79 (0,09203)

Tabela 6.6: Resultados obtidos com o método OM-SPEA2

CD	Erro-ant	Erro-apos	VE	IIS
iris	2,7 (0,00107)	2,7 (0,00107)	0,0 (0,00000)	1,00 (0,00000)
wine	7,4 (0,00197)	3,3 (0,00078)	4,1 (0,00272)	0,49 (0,03218)
thyroid	3,8 (0,00127)	2,8 (0,00020)	1,0 (0,00301)	0,88 (0,04195)
heart_s	23,2 (0,00437)	20,4 (0,00336)	2,8 (0,01133)	0,65 (0,09829)
sonar	27,2 (0,01209)	24,3 (0,00956)	2,9 (0,00830)	0,66 (0,14662)
bupa	38,3 (0,00574)	36,0 (0,00411)	2,3 (0,00660)	0,70 (0,06428)
breast_can	5,2 (0,00077)	4,6 (0,00055)	0,6 (0,00238)	0,71 (0,09203)

6.2.3. Resultados para o método genético multiobjetivo proposto para a geração de sistemas *fuzzy*

As Tabelas 6.7 e 6.8 sintetizam os resultados obtidos pela técnicas de otimização multiobjetivo NSGA-II e SPEA2 respectivamente. Nessas tabelas pode-se visualizar, para cada conjunto de dados, o Erro Majoritário (EM), os valores médios da taxa de erro, número de regras, número total de condições na BR e IIS.

Tabela 6.7: Resultados finais do método proposto com NSGA-II

CD	EM	Taxa de Erro	Regras	Condições	IIS
iris	66,7	2,7 (0,00107)	4,0 (0,00000)	6,0 (0,00000)	1,00 (0,00000)
wine	60,1	4,6 (0,00311)	6,1 (0,00000)	9,1 (0,00000)	0,76 (0,05323)
thyroid	30,2	3,2 (0,00222)	4,9 (0,00000)	7,4 (0,00000)	0,94 (0,04183)
heart_s	44,4	19,7 (0,00489)	20,6 (0,38471)	64,5 (1,69988)	0,75 (0,08955)
sonar	46,4	23,5 (0,00910)	8,2 (0,85182)	18,1 (1,61914)	0,73 (0,14662)
bupa	42,0	35,7 (0,00801)	6,6 (0,11662)	11,6 (0,32000)	0,60 (0,08944)
breast_can	37,2	4,3 (0,00107)	6,3 (0,04000)	8,5 (0,04000)	0,79 (0,09203)

Tabela 6.8: Resultados finais do método proposto com SPEA2

CD	EM	Taxa de Erro	Regras	Condições	IIS
iris	66,7	2,7 (0,00107)	4,0 (0,00000)	6,0 (0,00000)	1,00 (0,00000)
wine	60,1	3,3 (0,00078)	6,1 (0,00000)	9,1 (0,00000)	0,49 (0,03218)
thyroid	30,2	2,8 (0,00020)	4,3 (0,00000)	6,6 (0,00000)	0,88 (0,04195)
heart_s	44,4	20,4 (00336)	16,7 (0,25768)	49,8 (1,06395)	0,65 (0,09829)
sonar	46,4	24,3 (0,00956)	7,8 (0,45782)	15,8 (0,89219)	0,66 (0,14662)
bupa	42,0	36,0 (0,00411)	6,1 (0,04000)	10,6 (0,12000)	0,70 (0,06428)
breast_can	37,2	4,6 (0,00055)	6,2 (0,09798)	9,2 (0,00160)	0,71 (0,09203)

Para validar os resultados obtidos, foram realizadas comparações dos métodos propostos com métodos encontrados na literatura, tanto aqueles propostos para a geração da BR, quanto dos métodos propostos para a otimização da BD. Essas comparações são apresentadas a seguir.

6.2.4. Comparação do método proposto para a geração genética multiobjetivo da base de regras

Os métodos de geração genética multiobjetivo da BR foram comparados com três métodos de geração da BR encontrados na literatura e apresentados no capítulo 4. O primeiro foi o método genético NSLV, que utiliza a abordagem IRL (GONZALEZ; PEREZ, 2008). Este método foi considerado para mostrar a diferença na utilização de AGMO em vez de AG.

O segundo método utilizado para comparações foi o método genético NMEEF, que utiliza a técnica de otimização multiobjetivo NSGA-II (CARMONA; GONZALEZ, JESUS, HERRERA, 2009). Este método foi considerado para mostrar a diferença de usar a técnica NSGA-II para gerar a melhor regra em vez de gerar as melhores regras em uma só execução.

Por fim, foi utilizado o método genético MESDIF que utiliza a técnica de otimização multiobjetivo SPEA2 (BERLANGA; JESUS; GONZALEZ; HERRERA; MESONERO, 2006). Este método foi considerado pelo mesmo motivo que o método NMEEF.

Os resultados do método NSLV foram extraídos do mesmo artigo e os resultados dos métodos NMEEF e MESDIF foram extraídos da ferramenta KEEL (ALCALÁ-FDEZ, FERNANDEZ, LUENGO, DERRAC, GARCIA, SANCHEZ, HERRERA, 2011).

A Tabela 6.9 mostra a média da taxa de erro dos métodos IRL-NSGA-II, IRL-SPEA2 e dos três métodos utilizados para a comparação, respectivamente.

Tabela 6.9: Média da taxa de erro dos métodos IRL-NSGA-II, IRL-SPEA2, NSLV, NMEEF e MESDIF

CD	IRL-NSGA-II	IRL-SPEA2	NSLV	NMEEF	MESDIF
iris	2,7 (0,00107)	2,7 (0,00107)	6,1 (0,00376)	9,4 (0,01478)	4,0 (0,00040)
wine	7,3 (0,00223)	7,4 (0,00197)	6,2 (0,01363)	27,9 (0,0923)	15,3 (0,00139)
thyroid	3,8 (0,00127)	3,8 (0,00127)	9,3 (0,1328)	14,8 (0,00155)	13,3 (0,00155)
heart_s	25,7 (0,00560)	23,2 (0,00437)	24,5 (0,0354)	40,0 (0,1170)	27,8 (0,01381)
sonar	25,9 (0,01029)	27,2 (0,01209)	27,9 (0,00415)	34,7 (0,00678)	47,6 (0,00743)
bupa	40,4 (0,00465)	38,3 (0,00574)	42,1 (0,01723)	42,1 (0,00758)	48,4 (0,00448)
breast_can	4,8 (0,00080)	5,2 (0,00077)	6,2 (0,00329)	22,6 (0,00712)	36,8 (0,01248)
MÉDIA	15,8	15,4	17,5	27,4	27,6

A tabela 6.10 mostra a média da quantidade de regras na BR final dos métodos IRL-NSGA-II, IRL-SPEA2 e dos três métodos utilizados para comparação.

Tabela 6.10: Média da quantidade de regras dos métodos IRL-NSGA-II, IRL-SPEA2, NSLV, NMEEF e MESDIF

CD	IRL-NSGA-II	IRL-SPEA2	NSLV	NMEEF	MESDIF
iris	4,0 (0,00000)	4,0 (0,00000)	4,0 (0,00000)	3,8 (0,09318)	9,0 (0,00000)
wine	6,1 (0,00000)	6,1 (0,00000)	4,5 (0,00710)	3,1 (0,52410)	9,0 (0,00000)
thyroid	4,9 (0,00000)	4,3 (0,00000)	5,0 (0,00000)	7,5 (0,47988)	9,0 (0,00000)
heart_s	20,6 (0,38471)	16,7 (0,25768)	8,6 (0,97056)	3,3 (0,75296)	6,0 (0,00000)
sonar	8,2 (0,85182)	7,8 (0,45782)	8,3 (0,30640)	10,5 (0,21291)	15,0 (0,00000)
bupa	6,6 (0,11662)	6,1 (0,04000)	5,0 (0,00000)	2,3 (0,15360)	6,0 (0,00000)
breast_can	6,3 (0,04000)	6,2 (0,09798)	4,2 (0,88823)	9,9 (0,95796)	6,0 (0,00000)
MÉDIA	8,1	7,3	5,7	5,8	8,6

A tabela 6.11 mostra a média da quantidade total de condições na BR dos métodos IRL-NSGA-II, IRL-SPEA2 e dos três métodos utilizados para comparação.

Tabela 6.11: Média da quantidade de condições dos métodos IRL-NSGA-II, IRL-SPEA2, NSLV, NMEEF e MESDIF

CD	IRL-NSGA-II	IRL-SPEA2	NSLV	NMEEF	MESDIF
iris	6,0 (0,00000)	6,0 (0,00000)	3,5 (0,00000)	13,8 (0,92595)	13,3 (0,00044)
wine	9,1 (0,00000)	9,1 (0,00000)	14,5 (0,00555)	20,1 (0,41000)	15,3 (0,00021)
thyroid	7,4 (0,00000)	6,6 (0,00000)	11,5 (0,00922)	15,8 (0,64792)	19,2 (0,00721)
heart_s	64,5 (1,69988)	49,8 (1,06395)	23,5 (0,00310)	12,3 (0,46515)	12,4 (0,00453)
sonar	18,1 (1,61914)	15,8 (0,89219)	49,7 (0,08782)	76,7 (1,00765)	60,5 (0,07730)
bupa	11,6 (0,32000)	10,6 (0,12000)	16,1 (0,00927)	13,6 (0,91897)	11,0 (0,05543)
breast_can	8,5 (0,04000)	9,2 (0,00160)	18,6 (0,00304)	76,0 (0,33681)	49,2 (0,10043)
MÉDIA	17,9	15,3	19,6	32,6	25,8

Realizada uma comparação superficial dos métodos de geração da BR, os resultados são satisfatórios, demonstrando que os métodos genéticos multiobjetivo para a geração da BR propostos neste trabalho apresentam taxa média de erro e quantidade de condições média menores que os três métodos de geração da BR usados para comparação. Entretanto, a quantidade média de regras foi maior do que a dos três métodos de geração da BR.

Com o objetivo de fazer uma comparação mais precisa dos métodos de geração da BR, foi utilizado o teste de postos com sinal de Wilcoxon (DEMSAR, 2006), cujo funcionamento é detalhado no Anexo A.

A primeira comparação realizada foi entre IRL-NSGA-II e os métodos: IRL-SPEA2, NSLV, NMEEF e MESDIF. Os postos obtidos para cada comparação são mostrados na Tabela 6.12.

Tabela 6.12: Posto da comparação da média da taxa de erro do método IRL-NSGA-II com os métodos IRL-SPEA2, NSLV, NMEEF e MESDIF

CD	IRL-SPEA2	NSLV	NMEEF	MESDIF
iris	-	6	2	1
wine	1	-1	7	3,5
thyroid	-	7	4	5
heart_s	-5	-2	5	2
sonar	3	5	3	6
bupa	-4	4	1	3,5
breast_can	2	3	6	7
Soma T⁺	6	25	28	28
Soma T⁻	-9	-3	0	0

Na comparação realizada pelo teste de postos com sinal de Wilcoxon se conclui que a média da taxa de erro do método IRL-NSGA-II é significativamente menor que as médias das taxas de erro dos métodos: NSLV, NMEEF e MESDIF. Nada pode ser concluído da comparação entre os métodos IRL-NSGA-II e IRL-SPEA2.

O mesmo teste de comparação foi realizado entre o método proposto IRL-SPEA2 e os métodos: IRL-NSGA-II, NSLV, NMEEF e MESDIF. Os postos obtidos para cada comparação são mostrados na Tabela 6.13.

Tabela 6.13: Posto da comparação da média da taxa de erro do método IRL-SPEA2 com os métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF

CD	IRL-NSGA-II	NSLV	NMEEF	MESDIF
iris	-	5	2	1
wine	-1	-3	7	3
thyroid	-	7	4	4
heart_s	5	4	5	2
sonar	-3	1	3	6
bupa	4	6	1	5
breast_can	-2	2	6	7
Soma T⁺	9	25	28	28
Soma T⁻	-6	-3	0	0

Nessa comparação se conclui que a média da taxa de erro do método IRL-SPEA2 é significativamente menor que as médias das taxas de erro dos métodos NSLV, NMEEF e MESDIF. Enquanto na comparação entre IRL-SPEA2 e IRL-NSGA-II se conclui que a média da taxa de erro entre IRL-SPEA2 e IRL-NSGA-II não tem diferença significativa.

O mesmo tipo de comparação foi realizado focando o desempenho dos métodos de geração da BR em relação ao número de regras. A comparação da média da quantidade de regras na BR final foi realizada entre o método IRL-NSGA-II e os métodos: IRL-SPEA2, NSLV, NMEEF e MESDIF. Os postos obtidos para cada comparação são mostrados na Tabela 6.14.

Tabela 6.14: Posto da comparação da média da quantidade de regras do método IRL-NSGA-II com os métodos IRL-SPEA2, NSLV, NMEEF e MESDIF

CD	IRL-SPEA2	NSLV	NMEEF	MESDIF
iris	-	-	-1	5
wine	-3,5	-2,5	-4	3
thyroid	1,5	1,5	3	4
heart_s	-6	-6	-7	-7
sonar	1,5	1,5	2	6
bupa	-3,5	-2,5	-6	-2
breast_can	-5	-5	5	-1
Soma T⁺	3	3	10	18
Soma T⁻	-18	-18	-18	-10

Nessa comparação se conclui que a média da quantidade de regras do método IRL-NSGA-II não tem diferença significativa com as médias da quantidade de regras dos

métodos NSLV, NMEEF e MESDIF. Nada pode ser concluído da comparação entre os métodos IRL-NSGA-II e IRL-SPEA2.

O mesmo teste de comparação para a média da quantidade de regras foi realizado entre o método proposto IRL-SPEA2 e os métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF. Os postos obtidos para cada comparação são mostrados na Tabela 6.15.

Tabela 6.15: Posto da comparação da média da quantidade de regras do método IRL-SPEA2 com os métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF

CD	IRL-NSGA-II	NSLV	NMEEF	MESDIF
iris	-	-	-1	5
wine	4	-4	-3	3
thyroid	-1,5	2	4	4
heart_s	6	-6	-7	-7
sonar	-1,5	1	2	6
bupa	3	-3	-6	-1
breast_can	5	-5	5	-2
Soma T⁺	18	3	11	18
Soma T⁻	-3	-18	-17	-10

Nessa comparação se conclui que a média da quantidade de regras do método IRL-SPEA2 não tem diferença significativa com as médias da quantidade de regras dos métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF.

A última comparação dos métodos para a geração da BR foi realizada para comparar a quantidade de condições da BR final. Foi realizada entre o método proposto IRL-NSGA-II e os métodos: IRL-SPEA2, NSLV, NMEEF e MESDIF. Os postos obtidos para cada comparação são mostrados na Tabela 6.16.

Tabela 6.16: Posto da comparação da média da quantidade de condições do método IRL-NSGA-II com os métodos IRL-SPEA2, NSLV, NMEEF e MESDIF

CD	IRL-SPEA2	NSLV	NMEEF	MESDIF
iris	-	-1	2	3
wine	-	4	4	2
thyroid	-2	2	3	4
heart_s	-5	-7	-5	-7
sonar	-4	6	6	6
bupa	-3	3	1	-1
breast_can	1	5	7	5
Soma T⁺	1	20	23	20
Soma T⁻	-14	-8	-5	-8

Nessa comparação se conclui que a média da quantidade de condições do método IRL-NSGA-II não tem diferença significativa com as médias da quantidade de regras dos métodos NSLV, NMEEF e MESDIF. Nada pode ser concluído da comparação entre os métodos IRL-NSGA-II e IRL-SPEA2.

O mesmo teste de comparação foi realizado entre o método IRL-SPEA2 e os métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF. Os postos obtidos para cada comparação são mostrados na Tabela 6.17.

Tabela 6.17: Posto da comparação da média da quantidade de condições do método IRL-SPEA2 com os métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF

CD	IRL-NSGA-II	NSLV	NMEEF	MESDIF
iris	-	-1	2	3
wine	-	3	4	2
thyroid	2	2	3	4
heart_s	5	-6	-5	-5
sonar	4	7	6	7
bupa	3	4	1	1
breast_can	-1	5	7	6
Soma T⁺	14	21	23	23
Soma T⁻	-1	-7	-5	-5

Nessa comparação se conclui que a média da quantidade de condições do método IRL-SPEA2 não tem diferença significativa com as médias da quantidade de condições dos métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF.

Entre os métodos IRL-NSGA-II e IRL-SPEA2 não se encontram diferenças significativas porque esses métodos utilizam os mesmos objetivos e as duas técnicas, NSGA-II e SPEA2, tentam encontrar a fronteira de Pareto. Devido ao alto número de iterações, geralmente os dois métodos encontram a mesma melhor regra em cada execução.

Os métodos propostos IRL-NSGA-II e IRL-SPEA2, foram melhor significativamente na precisão e não tiveram diferença significativa na quantidade de regra e de condições que o método NSLV. Isso ocorre porque ao unir a precisão e interpretabilidade em um objetivo, o método NSLV pode considerar em alguns casos como melhor regra, uma regra com altíssima precisão e baixa interpretabilidade ou altíssima interpretabilidade e baixa precisão, apesar da definição de pesos diferentes para cada objetivo. Isso faz com que, na inserção da melhor regra na BR final um objetivo predomine sobre o outro. Esta situação é mais difícil de ocorrer

utilizando os métodos propostos, porque o resultado é um conjunto de regras candidatas, que são equilibradas nos dois objetivos, sendo que é inserida na B R final a regra que melhore o seu desempenho.

Todas essas comparações utilizando o teste de postos com sinal de Wilcoxon podem ser analisadas com mais detalhes no Apêndice B.

6.2.5. Comparação do método proposto para a otimização genética multiobjetivo da base de dados

Os métodos de otimização genética multiobjetivo da BD foram comparados com o método de otimização genética da BD utilizado em (PIMENTA, 2009) e apresentado no capítulo 4. Esse método foi executado utilizando as BR obtidas pelos métodos IRL-NSGA-II e IRL-SPEA2 e resultados da otimização são chamados neste trabalho de AG-1 e AG-2 respectivamente. O método genético de (PIMENTA, 2009) foi considerado para mostrar a diferença na utilização de AGMO em vez de AG.

Foram comparados três índices gerados pelos métodos de otimização da BD: a melhora da média da taxa de erro, a média do Índice de Interpretabilidade Semântica (IIS) e a média da quantidade de BD Transparentes.

As tabelas 6.18 e 6.19 mostram a média da melhora ou Variação do Erro (VE) dos métodos OM-NSGA-II e OM-SPEA2 com o método genético de otimização da BD, AG-1 e AG-2.

Tabela 6.18: Média da melhora da taxa de erro dos métodos OM-NSGA-II e AG-1

CD	IRL-NSGA-II	OM-NSGA-II	VE(NSGA-II)	AG-1	VE(AG-1)
iris	2,7 (0,00107)	2,7 (0,00107)	0,0 (0,00000)	2,7 (0,00000)	0,0 (0,00000)
wine	7,3 (0,00223)	4,6 (0,00311)	2,7 (0,00553)	3,4 (0,00548)	3,9 (0,00555)
thyroid	3,8 (0,00127)	3,2 (0,00222)	0,6 (0,00381)	3,0 (0,00481)	0,8 (0,00812)
heart_s	25,7 (0,00560)	19,7 (0,00489)	6,0 (0,00502)	20,0 (0,00519)	5,7 (0,00129)
sonar	25,9 (0,01029)	23,5 (0,00910)	2,4 (0,00830)	22,9(0,00865)	3,0 (0,00235)
bupa	40,4 (0,00465)	35,7 (0,00801)	4,7 (0,01239)	35,3 (0,00112)	5,1 (0,00284)
brt_can	4,8 (0,00080)	4,3 (0,00107)	0,5 (0,00238)	4,1 (0,00566)	0,7 (0,00109)
MÉD.	15,8	13,4	2,4	13,1	2,7

Tabela 6.19: Média da melhora da taxa de erro dos métodos OM-SPEA2 e AG-2

CD	IRL-SPEA2	OM-SPEA2	VE(SPEA2)	AG-2	VE(AG-2)
iris	2,7 (0,00107)	2,7 (0,00107)	0,0 (0,00000)	2,7 (0,00000)	0,0 (0,00000)
wine	7,4 (0,00197)	3,3 (0,00078)	4,1 (0,00272)	3,4 (0,00365)	4,0 (0,00771)
thyroid	3,8 (0,00127)	2,8 (0,00020)	1,0 (0,00301)	2,8 (0,00305)	1,0 (0,00564)
heart_s	23,2 (0,00437)	20,4 (0,00336)	2,8 (0,01133)	21,1 (0,00654)	2,1 (0,00010)
sonar	27,2 (0,01209)	24,3 (0,00956)	2,9 (0,00830)	24,1 (0,00077)	3,1 (0,00136)
bupa	38,3 (0,00574)	36,0 (0,00411)	2,3 (0,00660)	36,2 (0,02001)	2,0 (0,00249)
brt_can	5,2 (0,00077)	4,6 (0,00055)	0,6 (0,00238)	4,4 (0,00333)	0,8 (0,00110)
MÉD.	15,4	13,4	2,0	13,5	1,8

As tabelas 6.20 e 6.21 mostram a média do IIS da BD após a otimização dos métodos OM-NSGA-II e OM-SPEA2 com o método genético de otimização da BD, AG-1 e AG-2.

Tabela 6.20: Média do Índice de Interpretabilidade Semântica dos métodos OM-NSGA-II e AG-1

CD	OM-NSGA-II	AG-1
iris	1,00 (0,00000)	1,00 (0,00000)
wine	0,76 (0,05323)	0,50 (0,04808)
thyroid	0,94 (0,04183)	0,82 (0,00632)
heart_s	0,75 (0,08955)	0,72 (0,08052)
sonar	0,73 (0,14662)	0,70 (0,06187)
bupa	0,60 (0,08944)	0,50 (0,01663)
breast_can	0,79 (0,09203)	0,74 (0,09976)
MÉDIA	0,80	0,71

Tabela 6.21: Média do índice de interpretabilidade semântica dos métodos OM-SPEA2 e AG-2

CD	OM-SPEA2	AG-2
iris	1,00 (0,00000)	1,00 (0,00000)
wine	0,49 (0,03218)	0,46 (0,08669)
thyroid	0,88 (0,04195)	0,80 (0,04086)
heart_s	0,65 (0,09829)	0,64 (0,04358)
sonar	0,66 (0,14662)	0,64 (0,04898)
bupa	0,70 (0,06428)	0,58 (0,00121)
breast_can	0,71 (0,09203)	0,68 (0,00652)
MÉDIA	0,80	0,69

As tabelas 6.22 e 6.23 mostram a comparação da média da quantidade de BD transparentes após a otimização dos métodos OM-NSGA-II e OM-SPEA2 com o método genético de otimização da BD, AG-1 e AG-2.

Tabela 6.22: Média da quantidade de BD transparentes dos métodos OM-NSGA-II e AG-1

CD	OM-NSGA-II	AG-1
iris	1,00 (0,00000)	1,00 (0,00000)
wine	1,00 (0,00000)	0,46 (0,04658)
thyroid	1,00 (0,00000)	0,78 (0,02531)
heart_s	1,00 (0,00000)	0,60 (0,09723)
sonar	1,00 (0,00000)	0,58 (0,08018)
bupa	1,00 (0,00000)	0,45 (0,04552)
breast_can	1,00 (0,00000)	0,70 (0,00629)
MÉDIA	1,00	0,65

Tabela 6.23: Média de quantidade de BD transparentes dos métodos OM-SPEA2 e AG-2

CD	OM-SPEA2	AG-2
iris	1,00 (0,00000)	1,00 (0,00000)
wine	1,00 (0,00000)	0,43 (0,03211)
thyroid	1,00 (0,00000)	0,81 (0,04806)
heart_s	1,00 (0,00000)	0,42 (0,02631)
sonar	1,00 (0,00000)	0,57 (0,04582)
bupa	1,00 (0,00000)	0,39 (0,08764)
breast_can	1,00 (0,00000)	0,69 (0,08731)
MÉDIA	1,00	0,61

Realizada uma comparação superficial dos métodos de otimização da BD, os resultados são considerados satisfatórios e demonstram que os métodos genéticos multiobjetivo para a otimização BD possuem uma média da melhora de taxa de erro, média de IIS e média da porcentagem de BD transparentes maiores que as médias dos métodos genéticos para a otimização da BD, AG-1 e AG-2.

Da mesma forma que na comparação dos métodos de geração da BR, com o objetivo de fazer uma comparação mais precisa dos métodos de otimização da BD, foi utilizado o teste de postos com sinal de Wilcoxon (DEMSAR, 2006).

A primeira comparação foi entre o método OM-NSGA-II e o método AG-1. Os postos obtidos para cada comparação da melhora da média da taxa de erro (VE), a média do IIS e a média da quantidade de BD Transparentes (BDT) são mostrados na Tabela 6.24.

Tabela 6.24: Posto da comparação entre o método OM-NSGA-II e AG-1

	VE	IIS	BDT
iris	-	-	-
wine	-6	6	5
thyroid	-2,5	5	1
heart_s	2,5	1,5	3
sonar	-5	1,5	4
bupa	-4	4	6
breast_can	-1	3	2
Soma T⁺	2,5	21	21
Soma T⁻	-18,5	0	0

Nessa comparação se conclui que a melhora ou variação da taxa de erro não tem diferença significativa entre OM-NSGA-II e AG-1. Conclui-se também que a média do IIS e a quantidade de BD transparentes do método OM-NSGA-II são significativamente maiores que essas médias do método AG-1.

A segunda e última comparação foi entre o método proposto OM-SPEA2 e o método AG-2. Os postos obtidos para cada comparação da média da melhora da taxa de erro (VE), a média do IIS e a média da quantidade de BD Transparentes (BDT) são mostrados na Tabela 6.25.

Tabela 6.25: Posto da comparação entre o método OM-SPEA2 e AG-2

	VE	IIS	BDT
iris	-	-	-
wine	1	3,5	4
thyroid	-	5	1
heart_s	5	1	5
sonar	-2,5	2	3
bupa	4	6	6
breast_can	-2,5	3,5	2
Soma T⁺	10	21	21
Soma T⁻	-5	0	0

Nessa comparação conclui-se que a melhora ou variação da taxa de erro não tem diferença significativa entre OM-SPEA2 e AG-2. Conclui-se também que a média do IIS e a quantidade de BD transparentes do método OM-SPEA2 são significativamente maiores que essas médias do método AG-2. Nesta comparação, o método proposto utilizando AGMO melhorou a interpretabilidade da BD otimizada, porque diferentemente do método genético, é considerado o objetivo de IIS, além do objetivo de gerar BD transparentes, sem perder o objetivo de precisão.

Todas essas comparações utilizando o teste de postos com sinal de Wilcoxon podem ser analisadas com mais detalhes no Apêndice B.

6.3. Considerações finais

Neste capítulo foram discutidos os experimentos realizados com os quatro métodos proposto que são parte do método geral para a geração genética multiobjetivo de sistemas *fuzzy*, sendo dois métodos para a geração da BR e dois métodos para a otimização da BD. Foram apresentadas também comparações com métodos encontrados na literatura.

No capítulo seguinte são apresentadas as conclusões sobre os resultados obtidos e propostas para trabalhos futuros.

Capítulo 7

CONCLUSÕES

Este trabalho apresentou um método para geração de sistemas *fuzzy* a partir de um conjunto de exemplos usando algoritmos genéticos multiobjetivo, especificamente baseados nas técnicas de otimização multiobjetivo NSGA-II e SPEA2, usando a abordagem iterativa na geração de regras *fuzzy*. O processo de geração é composto por três etapas. Na primeira etapa são gerados os conjuntos *fuzzy* uniformemente distribuídos para cada atributo dos exemplos. Esses conjuntos são utilizados na segunda etapa, onde um algoritmo genético multiobjetivo, usando a abordagem iterativa, é utilizado para gerar as regras *fuzzy*. Por fim, na terceira etapa, outro algoritmo genético multiobjetivo é utilizado para otimizar o formato das funções de pertinência dos conjuntos *fuzzy*.

A terceira etapa pode ser aplicada separadamente como pós-processamento para outros métodos que geram regras *fuzzy*. Pode-se, assim, analisar o seu comportamento quando as regras são criadas por qualquer método de aprendizado.

Cada algoritmo genético multiobjetivo utilizado no método de geração de sistemas *fuzzy* utiliza dois objetivos. O primeiro representa a precisão e o segundo, a interpretabilidade.

Os resultados dos experimentos evidenciam um bom desempenho do método de geração automática de sistemas *fuzzy* mediante algoritmos genéticos multiobjetivo. A eficiência desses sistemas foi verificada nos experimentos apresentados para resolver problemas de classificação. Esses experimentos mostraram, na geração de regras, uma taxa média de erro menor em relação à geração de regras efetuada pelo método genético NSLV e por dois métodos genéticos multiobjetivos, NMEEF e MESDIF. Na otimização de conjuntos, os

métodos propostos mostraram uma média de interpretabilidade semântica e transparência melhores em relação à otimização de conjuntos obtida por um método genético proposto anteriormente.

Para estudos futuros podem-se incluir as seguintes modificações no método proposto e possíveis comparações:

1. Avaliar o uso de novos métodos para escolher uma única regra do conjunto de regras não dominadas na geração da BR.
2. Avaliar a utilização de outros índices ou características de interpretabilidade como objetivos em AGMO, na geração de regras *fuzzy* e otimização de conjuntos *fuzzy*.
3. Comparar os resultados na geração de regras com outros métodos que utilizem algoritmos genéticos multiobjetivos usando a abordagem Michigan e Pittsburgh.
4. Comparar os resultados da otimização de conjuntos com outros métodos que utilizem algoritmos genéticos multiobjetivos.

REFERÊNCIAS

- ABADEH, M. & HABIBI, J. Computer Intrusion Detection Using an Iterative Fuzzy Rule Learning Approach. **IEEE International Fuzzy Systems Conference**. London, pp. 1-6. 2007.
- ABRAHAM, A.; JAIN, L. & GOLDBERG, R. Evolutionary Multiobjective Optimization. Theoretical Advances and Applications. **Springer**. 2004.
- ALCALA, R.; ALCALA-FDEZ, J. & HERRERA, F. A Proposal for the Genetic Lateral Tuning of Linguistic Fuzzy Systems and Its Interaction With Rule Selection. **IEEE Transactions on Fuzzy Systems**. v. 15, n. 4, pp. 616-635. 2007.
- ALCALA, R.; DUCANGE, P.; HERRERA, F.; LAZZERINI B. & MARCELLONI, F. A Multiobjective Evolutionary Approach to Concurrently Learn Rule and Data Bases of Linguistic Fuzzy-Rule-Based Systems. **IEEE Transactions on Fuzzy Systems**. v. 17, n. 5, pp. 1106–1122, 2009.
- ALCALA, R.; GACTO, M.; HERRERA, F. & ALCALA-FDEZ, J. A multi-objective genetic algorithm for tuning and rule selection to obtain accurate and compact linguistic fuzzy rule-based systems. **International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems**. v. 15, n. 5, pp. 521–537, 2007.
- ALCALA-FDEZ, J.; HERRERA, F.; MARQUEZ, F. & PEREGRIN, A. Increasing fuzzy rules cooperation based on evolutionary adaptive inference systems. **International Journal of Intelligent Systems**. v. 22, n. 9, pp. 1035–1064, 2007.
- ALCALA-FDEZ, J.; FERNANDEZ, A.; LUENGO, J.; DERRAC, J.; GARCIA, S.; SANCHEZ, L. & HERRERA, F. KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework. *Journal of Multiple-Valued Logic and Soft Computing* v. 17, n. 2, pp. 255-287, 2011.
- ALONSO, J.; CORDON, O.; GUILLAUME, S. & MAGDALENA, L. Highly Interpretable Linguistic Knowledge Bases Optimization: Genetic Tuning versus Solis-Wetts. Looking for a good interpretability-accuracy trade-off. London, pp. 901-90. **IEEE International Conference on Fuzzy Systems**. 2007.
- ANTONELLI, M.; DUCANGE, P.; LAZZERINI, B. & MARCELLONI, F. A Multi-Objective Genetic Approach to Concurrently Learn Partition Granularity and Rule Bases of Mamdani Fuzzy Systems. **Eighth International Conference on Hybrid Intelligent Systems**. Washington, DC, USA, pp. 278-283. 2008.
- ANTONELLI, M.; DUCANGE, P. & MARCELLONI, F. Exploiting a coevolutionary approach to concurrently select training instances and learn rule bases of Mamdani fuzzy systems. **IEEE International Conference on Fuzzy Systems**. Barcelona, pp. 1-7. 2010.

- BACK, T. & SCHWEFEL, H. An overview of evolutionary algorithms for parameter optimization. **Evolutionary Computation**. v. 1, n. 1, pp. 1-23, 1993.
- BARDOSSY, A. & DUKSTEIN, L. Fuzzy-Rule-Based Modeling With Application to Geophysical, **Biological and Engineering Systems**, 1995.
- BERLANGA, F.; JESUS, M.; GONZALEZ, P.; HERRERA, F. & MESONERO, M. Multiobjective evolutionary induction of subgroup discovery fuzzy rules: a case study in marketing. **Industrial Conference on Data Mining – ICDM**. Germany, pp. 337-349. 2006.
- FRANK, A. & ASUNCION, A. UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science. 2010.
- BOTTA, A.; LAZZERINE, B.; MARCELLONI, F. & STEFANESCU, D. Exploiting Fuzzy Ordering Relations to Preserve Interpretability in Context Adaptation of Fuzzy Systems. **IEEE International Fuzzy Systems Conference**. London, pp. 1-6. 2007.
- BOTTA, A.; LAZZERINE, B.; MARCELLONI, F. & STEFANESCU, D. Context adaptation of fuzzy systems through a multi-objective evolutionary approach based on a novel interpretability index. **Soft Computing**. v. 13, n. 5, pp. 437–449, 2009.
- CANCINO, W. Aplicação de Algoritmos Genéticos Multi-Objetivo para Alinhamento de Sequências Biológicas Dissertação (Mestrado). **Instituto de Ciências Matemáticas e de Computação, USP**. 2003.
- CARMONA, C.; GONZALEZ, P.; JESUS, M. & HERRERA, F. Non-dominated Multi-objective Evolutionary algorithm based on Fuzzy rules extraction for Subgroup Discovery. 4th International Conference on Hybrid Artificial Intelligence Systems (HAIS09). LNCS 5572, Springer 2009, Salamanca, pp. 573-580, 2009.
- CARSE, B. & PIPE, A. Special Issue on Genetic Fuzzy Systems. **International Journal of Intelligent Systems**. v 22, n. 9, pp. 905–907, 2007.
- CASALE; S.; RUSSO, A.; SCEBBA, G. & SERRANO, S. Speech Emotion Classification Using Machine Learning Algorithms. **IEEE International Conference on Semantic Computing**. pp. 158-165. 2008
- CASILLAS, J.; CORDON, O.; JESUS, J. & HERRERA, F. Genetic feature selection in a fuzzy rule-based classification system learning process for high-dimensional problems. **Information Sciences**. v. 136, n. 1–4, pp. 135–157, 2001.
- CASILLAS, J.; CORDON, O.; JESUS, M. & HERRERA, F. Genetic tuning of fuzzy rule deep structures preserving interpretability for linguistic modeling. **IEEE Transactions on Fuzzy Systems**. v. 13, n. 1, pp. 13–29, 2005.
- CASILLAS, J.; JESUS, M.; HERRERA, F.; PEREZ, P. & VILLAR, R. Special Issue on Genetic Fuzzy Systems and the Interpretability-Accuracy Trade-off. **International Journal of Approximate Reasoning**. v. 44, n. 1, 2007.

- CHANG, J. & LUO, Y. GPSM: a Generalized Probabilistic Semantic Model for ambiguity resolution. In Proceedings of the 30th Annual Meeting on Association For Computational Linguistics. 1992.
- CHI, Z.; YAN, H. & PHAM, T. Fuzzy Algorithms with Applications to Image Processing and Pattern Recognition. **World Scientific**. 1996.
- CINTRA, M. & CAMARGO, H. Geração Genética de Regras Fuzzy com Pré-Seleção de Regras Candidatas. **Encontro Nacional de Inteligência Artificial – ENIA**. Rio de Janeiro. pp. 1341-1350. 2007.
- COCOCCIONI, M.; DUCANGE P. LANZERINI B. & MARCELLONI, F. A pareto based multi-objective evolutionary approach to the identification of mamdani fuzzy systems. **Soft Computing Journal**. v. 11, n. 11, pp. 1013–1031, 2007.
- COELHO, L. & COELHO, A. Algoritmos Evolutivos em Identificação e Controle de Processos: Uma Visão Integrada e Perspectivas. **SBA Controle & Automação**. 1999.
- CORDON, O.; ALCALA, R.; ALCALA-FDEZ, J. & ROJAS, I. Genetic Fuzzy Systems. Special Section on Genetic Fuzzy Systems: What's Next?. **IEEE Transactions on Fuzzy Systems**. v. 15, n. 4, 2007.
- CORDON, O.; HERRERA, F.; HOFFMANN, F. & MAGDALENA ,L. Genetic fuzzy systems – evolutionary tuning and learning of fuzzy knowledge bases. **Advances in Fuzzy Systems-Applications and Theory**, 2001.
- CORDON, O.; HERRERA, F; GOMIDE, F.; HOFFMANN, F. & MAGDALENA, L. Ten years of genetic fuzzy systems current framework and new trends. **Fuzzy Sets and Systems**. v. 141, n. 1, pp. 5–31, 2004.
- CORDON, O.; HERRERA, F. & VILLAR, P. Generating the knowledge base of a fuzzy-based system by the genetic learning of the data base. **IEEE Transactions on Fuzzy Systems**. v. 9, n. 4, pp. 667–674, 2001.
- CORDON, O; JESUS, M.; HERRERA, F. & VILLAR, P. A Multiobjective Genetic Algorithm for Feature Selection and Granularity Learning in Fuzzy Rule-Based Classification Systems. **Nineth International Conference IFSA-NAFIPS**. Vancouver. pp. 1253 – 1258. 2001.
- CROCKETT, K.; BANDAR, Z. & MCLEAN, D. On the optimization of T-norm parameters within fuzzy decision trees. **IEEE International Conference on Fuzzy Systems**. London. pp. 1-6. 2007.
- DARWIN, C. On the Origin of Species by Means of Natural Selection. **London: John Murray**. 1859.
- DEB, K. Multi-Objective Optimization using Evolutionary Algorithms. **John Wiley & Sons**. 2001.

- DEB, K.; PRATAP, A.; AGARWAL, S. & MEYARIVAN, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. **IEEE Transactions on Evolutionary Computation**. v. 6, n. 2, pp.182–197, 2002.
- DEMSAR J. Statistical comparisons of classifiers over multiple data sets. **The Journal of Machine Learning Research**. vol. 7, p.1. 2006.
- DI NUOVO, A. & CATANIA, V. Linguistic Modifiers to improve the Accuracy-Interpretability Trade-off in Multi-Objective Genetic Design of Fuzzy Rule Based Classifier Systems. **Ninth International Conference on Intelligent Systems Design and Applications**. Pisa. pp. 128 - 133.2009.
- DUDA, R. & HART P. Pattern classification and Scene Analysis. **John Wiley & Sons**. 1973.
- EIBEN, A. & SMITH, J. Introduction to Evolutionary Computing. **Springer**. 2007.
- EVSUKOFF, A. Learning Fuzzy Rule Based Classifier with Rule Weights Optimization and Structure Selection by a Genetic Algorithm. **IEEE International Fuzzy Systems Conference**. pp. 1. 2007.
- FIESTRAS-JANEIRO, G.; TABOADA, J.; ORDOÑEZ, C. & SAAVEDRA, A. Fuzzy expert system for economic zonation of an ornamental slate deposit. **Engineering geology: An international journal**. v. 84, n. 3–4, pp. 220–228, 2006.
- FONSECA, C. & FLEMING, P. Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. **International Conference on Genetic Algorithms**. San Francisco . pp. 416—423. 1993.
- FRANK, A. & ASUNCION, A. UCI machine learning repository, 2010. [Online]. Available: <http://archive.ics.uci.edu/ml>
- GACTO, M.; ALCALA, R. & HERRERA, F. Adaptation and application of multi-objective evolutionary algorithms for rule reduction and parameter tuning of fuzzy rule-based systems. **Soft Computing**. v. 13, n. 5, pp. 419–436 , 2009.
- GACTO, M.; ALCALA, R. & HERRERA, F. Integration of an Index to Preserve the Semantic Interpretability in the Multiobjective Evolutionary Rule Selection and Tuning of Linguistic Fuzzy Systems. **IEEE Transactions on Fuzzy Systems**. v. 18, n. 3, pp. 515-531. 2010
- GAO, Y. & ZHANG, W. A genetic-based method for training fuzzy systems. **IEEE International Fuzzy Systems Conference**. pp. 123. 2001.
- GOLDBERG, D. Genetic Algorithms in Search, Optimization, and Machine Learning. **Addison-Wesley publishing Company**. 1989.
- GONZALEZ, A. & PEREZ, R. SLAVE: A genetic learning system based on the iterative approach. **IEEE Transaction on Fuzzy Systems**. v. 7, n. 2, pp. 176–191, 1999.

- GONZALEZ, A. & PEREZ, R. Una mejora del modelo genético iterativo de SLAVE. **XIV Congreso Español sobre Tecnologías y Lógica Fuzzy – ESTYLF**. España. pp. 155-161. 2008.
- GREENE, D. & SMITH S. Competition-based induction of decision models from examples. **Machine Learning**. 1993.
- HERRERA, F. Genetic fuzzy systems: Status, critical considerations and future directions. **International Journal of Computational Intelligence Research**. v. 1, n. 1, pp. 59–67, 2005.
- HERRERA, F. Genetic fuzzy systems: taxonomy, current research trends and prospects. **Evolutionary Intelligence**. v. 1, n. 1, pp. 27–46, 2008.
- HOLLAND, J. *Adaptation in Natural and Artificial Systems*. University of Michigan. **MIT**. 1975.
- HOLLAND, J. & REITMAN, J. *Cognitive Systems Based on Adaptive Algorithms*. **ACM SIGART Bulletin**. 1978.
- HUDSON, L. & COHEN, M. *Fuzzy Logic in Medical Expert Systems*, **IEEE Engineering Medicine and Biology**, 1994.
- ISHIBUCHI, H. Multiobjective genetic fuzzy systems: review and future research directions. **IEEE International Conference on Fuzzy Systems**. London. pp. 913-918. 2007.
- ISHIBUCHI, H.; KAISHO, Y. & NOJIMA, Y. Complexity, interpretability and explanation capability of fuzzy rule-based classifiers. **IEEE International Conference on Fuzzy Systems**. Korea. pp. 1730-1735. 2009.
- ISHIBUCHI, H. & NOJIMA, Y. Interactive genetic fuzzy rule selection through evolutionary multiobjective optimization with user preference. **IEEE Symposium on Computational intelligence in multi-criteria decision-making**. Nashville. pp. 141 – 148. 2009.
- ISHIBUCHI, H.; NOJIMA, Y. & KUWAJIMA, I. Fuzzy data mining by heuristic rule extraction and multiobjective genetic rule selection. **IEEE International Conference on Fuzzy Systems**. 2006.
- ISHIBUCHI, H.; NOJIMA, Y. & KUWAJIMA, I. Effectiveness of designing fuzzy rule-based classifiers from Pareto-optimal rules. **IEEE International Conference on Fuzzy Systems**. Hong Kong. pp. 1185 – 1192. 2008.
- ISHIBUCHI, H.; NOJIMA, Y. & NAKASHIMA, Y. Effects of the Use of Multiple Fuzzy Partitions on the Search Ability of Multiobjective Fuzzy Genetics-Based Machine Learning. **International Conference of Soft Computing and Pattern Recognition**. Washington. pp. 341-346. 2009.

- ISHIBUCHI, H. & YAMAMOTO T. Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining. **Fuzzy Sets and Systems**. v. 141, n.1, pp. 59–88, 2004.
- JONG, K. Evolutionary Computation: A Unified Approach. **Springer Netherlands**. 2006.
- KAYA, M. Multi-objective genetic algorithm based approaches for mining optimized fuzzy association rules. **Soft Computing**. v. 10, n. 7, pp. 578–586. 2006.
- KECMAN, V. Learning and Soft Computing. **MIT**. 2001.
- KIM, D; CHOI, Y. & LEE, S. An accurate COG defuzzifier design using Lamarckian co-adaptation of learning and evolution. **Fuzzy Sets and Systems**. v. 130, n. 2, pp. 207–225, 2002.
- KLIR, G. & YUAN, B. Fuzzy sets and fuzzy logic: theory and applications. **Upper Saddle River**, NJ: Prentice Hall PTR. 1995.
- KOHAVI, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence v.2, n.12, pp.1137–1143. 1995.
- LEON, J. Metodología para la detección de requerimientos subjetivos en el diseño de producto. **Universidad Politécnica de Catalunya**. 2005.
- LIAO, T. W.; CELMINS, A. K. & HAMMELL, R. J. A fuzzy C-Means variant for the generation of fuzzy term sets. **Fuzzy Sets and Fuzzy Systems**. v. 135, n. 2, pp. 241–257, 2003.
- MARQUEZ, A. A.; MARQUEZ, F. A. & PEREGRIN, A. A multi-objective evolutionary algorithm with an interpretability improvement mechanism for linguistic fuzzy systems with adaptive defuzzification. **IEEE International Fuzzy Systems Conference**. Barcelona. pp. 1 - 7.2010
- MARQUEZ, F; MARQUEZ, A. & PEREGRIN, A. Aprendizaje Cooperativo de la Base de Reglas y el Sistema de Inferencia mediante Algoritmos Genéticos Multi – Objetivo. **XIV Congreso Español sobre Tecnologías y Lógica Fuzzy – ESTYLF**. España. pp. 163-170. 2008.
- MARQUEZ, F.; PEREGRIN, A. & HERRERA, F. Cooperative evolutionary learning of linguistic fuzzy rules and parametric aggregation connectors for Mamdani fuzzy systems. **IEEE Transactions on Fuzzy Systems**. v. 15, n. 6, pp. 1168–1178, 2007.
- MENCAR, C.; CASTIELLO, C. & FANELLI, A. A logic-based approach for evaluating interpretability of fuzzy rule-based classifiers. **International Fuzzy Systems Association World Congress – European Society for Fuzzy Logic and Technology Conference, IFSA – EUSFLAT**. Portugal. pp. 339-344. 2009.
- NAUCK, D. & CRUSE, R. A neuro-fuzzy method to learn fuzzy classification rules from data. **Fuzzy Sets and Systems**. v. 89, n. 1, p. 277–288, 1997.

- NICOLETTI, M. & CAMARGO, H. Fundamentos da Teoria de Conjuntos Fuzzy. **EdUFScar**. 2004.
- NOMURA, H.; HAYASHI, L. & WAKAMI, N. A learning method of fuzzy inference rules by descent method. **IEEE International Conference on Fuzzy Systems**. San Diego. pp. 203 – 210. 1992.
- PARETO, V. Cours D'Economie Politique. à l'**Université de Lausanne**. 1896.
- PEDRYCZ, W. & GOMIDE, F. An Introduction to Fuzzy Sets. **MIT**. 1998.
- PIMENTA, A. Geração Genética de Classificador Fuzzy Intervalar do Tipo-2. Dissertação Mestrado. **Universidade Federal de São Carlos**. 2009.
- PIRES, M. G. & CAMARGO, H. C. Genetic learning and optimization of fuzzy sets in fuzzy rule-based systems. **IEEE International Conference on Information Reuse and Integration**. Piscataway. pp. 623 – 628. 2004.
- PULKKIEN, P & KOIVISTO, H. A Dynamically Constrained Multiobjective Genetic Fuzzy System for Regression Problems. **IEEE Transactions on Fuzzy Systems**. v. 18, n. 1, pp. 161-177. 2010
- SANCHES, D. Estratégia de Modelagem por Algoritmo Genético Adaptativo para Programação Reativa da Produção de Produtos com uso Simultâneo de Máquinas e Sistemas de Transporte em Sistemas de Manufatura, Dissertação Mestrado. **Universidade Federal de São Carlos**. 2008.
- SANCHEZ, L.; COUSO, I. & CASILLAS, J. Genetic learning of fuzzy rules based on low quality data. **Fuzzy Sets and Systems**. v. 160, n. 17, pp. 2524–2552, 2009.
- SCHAFFER, J. Multiple objective optimization with vector evaluated genetic algorithms. Genetic Algorithms and their Applications. **First International Conference on Genetic Algorithms**. Hillsdale. pp. 93-100. 1985.
- SIDNEY, S. & CASTELLAN, N. Estatística Não – Paramétrica para Ciências do Comportamento. **Artmed**. Publicação em língua portuguesa. 2006.
- SMITH, S. F. A learning System Based on Genetic Adaptive Algorithms. PhD thesis. **University of Pittsburgh**. 1980.
- TZUNG-PEI, H.; WEI-TEE, L.; CHIH-PING, C. & CHEN-SEN, O. Learning Membership Functions in Takagi-Sugeno Fuzzy Systems by Genetic Algorithms. **First Asian Conference on Intelligent Information and Database Systems**. Dong Hoi. pp. 301 - 306. 2009.
- XIONG, N. Learning flexible structured linguistic fuzzy rules for mamdani fuzzy systems. **IEEE International Conference on Fuzzy Systems**. Piscataway. pp. 86. 2009.

YAOCHU, J & SENDHOFF, B. Pareto-Based Multiobjective Machine Learning: An Overview and Case Studies. **IEEE Transaction on Systems, Man and Cybernetics—PART C: Applications and Reviews**. v. 38, n. 3, pp. 397–415, 2008.

YEUNG, D.; SHIU, S. & TSANG, E. Modelling flexible manufacturing systems using weighted Fuzzy Coloured Petri Nets. **Journal of Intelligent and Fuzzy Systems**. v. 7, n. 2, pp. 137–149, 1999.

ZADEH, L. Fuzzy sets, **Information and Control**, v. 8, n. 3, p. 338–353, 1965.

ZHOU, S. & GAN J. Low-level interpretability and high-level interpretability: a unified view of data-driven interpretable fuzzy system modeling. **Fuzzy Sets and Systems**. v. 159, n. 23, pp. 3091–3131, 2008.

ZITZLER, E.; LAUMANN, M. & THIELE, L. SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. **Evolutionary methods for design, optimization and control with applications to industrial problems—EUROGEN2001**, Greece. pp. 95 - 100. 2001.

ZUSUKI, T.; FURUHASHI, T; MATSUSHINA, S. & TUSUTSUI, H. Efficient fuzzy modeling under multiple criteria by using genetic algorithms. **IEEE International Conference on Systems, Man and Cybernetics**. Tokyo. pp. 314 – 319. 1999.

Apêndice A

TESTE DE POSTOS E SINAL DE WILCOXON

O teste de Postos com Sinal de Wilcoxon é um teste não paramétrico e baseado em hipóteses indicado no caso de tratamento simultâneo de duas amostras, eventualmente pequenas. Esse teste dá mais peso a um par que mostra uma diferença grande entre as duas condições do que a um par que mostra uma diferença pequena, além de utilizar a direção das diferenças, ou seja, se a diferença entre o par de amostras é positiva ou negativa.

Esse teste é recomendado para comparar dois classificadores sobre múltiplos conjuntos de dados (DEMSAR, 2006). Esse teste foi utilizado para comparar tanto o método proposto de geração genética multiobjetivo de regras *fuzzy* como o método proposto de otimização de conjuntos *fuzzy* com outros métodos encontrados na literatura. Foi utilizado para comparar uma por uma, algumas das características dos classificadores *fuzzy*.

Focando a comparação realizada neste trabalho e assumindo-se que são conhecidos os valores para as duas características a serem comparadas dos dois métodos, a comparação do teste do postos com sinal de Wilcoxon é realizado através dos seguintes passos (SIDNEY; CASTELLAN, 2006):

1. Para cada par de valores, X_i e Y_i , dos métodos X e Y para a característica i , determine a diferença com sinal $d_i = X_i - Y_i$.
2. Atribua postos a estes d_i 's sem considerar o sinal. Para os d_i 's empatados, atribua a média dos postos empatados.
3. Afixe em cada posto o sinal, + ou -, do d que ele representa.

4. Determine N , o número de d_i 's não nulos.
5. Determine T^+ , a soma dos pontos que tem um sinal positivo, e T^- , a soma dos pontos que tem um sinal negativo.
6. O procedimento para determinar a significância do valor observado de T^+ depende do tamanho de N :
 - a) Se N é 15 ou menos, a tabela A-1 dá probabilidades associadas com vários valores de T^+ . Se a probabilidade associada como o valor de T^+ for menor ou igual ao nível de significância escolhido, rejeite a hipótese H_0 , ou seja, a hipótese que determina que os valores dos dois métodos, X e Y , para essa característica são quase iguais. Valide a hipóteses H_1 que determina que os valores do método X para essa características são maiores que os valores do método Y . Caso contrário valide H_0 .
 - b) Se N é maior do que 15, calcule o valor de z usando a seguinte equação:

$$z = \frac{T^+ - Média}{Variância} = \frac{T^+ - \frac{N(N+1)}{4}}{\sqrt{\frac{N(N+1)(2N+1)}{24}}}$$

e, se existem postos empatados, corrija a variância usando seguinte equação:

$$Variância = \sqrt{\frac{N(N+1)(2N+1)}{24} - \frac{1}{2} \sum_{j=1}^g t_j(t_j-1)(t_j+1)}$$

Onde:

g = número de grupos de postos empatados diferentes.

t_j = número de postos empatados no grupos j .

Determine sua probabilidade associada utilizando a tabela A-2. Se a probabilidade assim obtida for menor ou igual a α , rejeite a hipótese H_0 , ou seja, a hipótese que determina que os valores dos dois métodos, X e Y , para essa característica são quase iguais. Valide a hipóteses H_1 que determina que os valores do método X para essa características são maiores que os valores do método Y . Caso contrário valide H_0 .

Tabela A-1: Valores críticos de T^+ para o teste de postos com sinal de Wilcoxon

e	3	4	5	6	7	8	9	10	11	12	13	14	15
3	0,6250												
4	0,3750												
5	0,2500	0,5625											
6	0,1250	0,4375											
7		0,3125											
8		0,1875	0,5000										
9		0,1250	0,4063										
10		0,0625	0,3125										
11			0,2188	0,5000									
12			0,1563	0,4219									
13			0,0938	0,3438									
14			0,0625	0,2813	0,5313								
15			0,0313	0,2188	0,4688								
16			0,1563	0,4063	0,4063								
17			0,1094	0,3438	0,3438								
18			0,0781	0,2891	0,2891	0,5273							
19			0,0469	0,2344	0,4727	0,4727							
20			0,0313	0,1875	0,4219	0,4219							
21			0,0156	0,1484	0,3711	0,3711							
22				0,1094	0,3203	0,3203							
23				0,0781	0,2734	0,5000							
24				0,0547	0,2305	0,4551							
25				0,0391	0,1914	0,4102							
26				0,0234	0,1563	0,3672							
27				0,0156	0,1250	0,3262							
28				0,0078	0,0977	0,2852	0,5000						
29					0,0742	0,2480	0,4609						
30					0,0547	0,2129	0,4229						
31					0,0391	0,1797	0,3848						
32					0,0273	0,1504	0,3477						
33					0,0195	0,1250	0,3125						
34					0,0117	0,1016	0,2783	0,4829	0,5171				
35					0,0078	0,0820	0,2461	0,4492	0,4492				
36					0,0039	0,0645	0,2158	0,4155	0,4155				
37						0,0488	0,1875	0,3823					
38						0,0371	0,1611	0,3501					
39						0,0273	0,1377	0,3188	0,5151				
40						0,0195	0,1162	0,2886	0,4849				
41						0,0137	0,0967	0,2598	0,4548				
42						0,0098	0,0801	0,2324	0,4250				
43						0,0059	0,0654	0,2065	0,3955				
44						0,0039	0,0527	0,1826	0,3667				
45						0,0020	0,0420	0,1602	0,3386				
46							0,0322	0,1392	0,3110	0,5000			
47							0,0244	0,1201	0,2847	0,4730			
48							0,0186	0,1030	0,2593	0,4463			
49							0,0137	0,0874	0,2349	0,4197			
50							0,0098	0,0737	0,2119	0,3934			
51							0,0068	0,0615	0,1902	0,3677			
52							0,0049	0,0508	0,1697	0,3424			
53							0,0029	0,0415	0,1506	0,3177	0,5000		
54							0,0020	0,0337	0,1331	0,2939	0,4758		
55								0,0269	0,1167	0,2709	0,4516		
56								0,0210	0,1018	0,2487	0,4276		
57								0,0161	0,0881	0,2274	0,4039		
58								0,0122	0,0757	0,2072	0,3804		
59								0,0093	0,0647	0,1879	0,3574		
60								0,0068	0,0549	0,1698	0,3349	0,5110	
61								0,0049	0,0461	0,1527	0,3129	0,4890	
62								0,0034	0,0386	0,1367	0,2915	0,4670	
63								0,0024	0,0320	0,1219	0,2708	0,4452	
64								0,0015	0,0261	0,1082	0,2508	0,4235	
65								0,0008	0,0212	0,0955	0,2316	0,4020	
66								0,0005	0,0171	0,0839	0,2131	0,3808	
67									0,0134	0,0732	0,1955	0,3599	
68									0,0105	0,0636	0,1788	0,3394	
69									0,0081	0,0549	0,1629	0,3193	
70									0,0061	0,0471	0,1479	0,2997	
71									0,0046	0,0402	0,1338	0,2807	
72									0,0034	0,0341	0,1206	0,2622	
73									0,0024	0,0287	0,1083	0,2444	
74									0,0017	0,0239	0,0969	0,2271	
75									0,0012	0,0199	0,0863	0,2106	
76									0,0007	0,0164	0,0765	0,1947	
77									0,0005	0,0133	0,0676	0,1796	
78									0,0002	0,0107	0,0594	0,1651	
79										0,0085	0,0520	0,1514	
80										0,0067	0,0453	0,1384	
81										0,0052	0,0392	0,1262	
82										0,0040	0,0338	0,1147	
83										0,0031	0,0290	0,1039	
84										0,0023	0,0247	0,0938	
85										0,0017	0,0209	0,0844	
86										0,0012	0,0176	0,0757	
87										0,0009	0,0148	0,0677	
88										0,0006	0,0123	0,0603	
89										0,0004	0,0101	0,0535	
90										0,0002	0,0083	0,0473	
91										0,0001	0,0067	0,0416	
92											0,0054	0,0365	
93											0,0043	0,0319	
94											0,0034	0,0277	
95											0,0026	0,0240	
96											0,0020	0,0206	
97											0,0015	0,0177	
98											0,0012	0,0151	
99											0,0009	0,0128	
100											0,0006	0,0108	
101											0,0004	0,0090	
102											0,0003	0,0075	
103											0,0002	0,0062	
104											0,0001	0,0051	
105												0,0042	
106												0,0034	
107												0,0027	
108												0,0021	
109												0,0017	
110												0,0013	
111												0,0010	
112												0,0008	
113												0,0006	
114												0,0004	
115												0,0003	
116												0,0002	
117												0,0002	
118												0,0001	
119												0,0001	
120												0,0000	

Tabela A-2: Probabilidades associadas com a cauda superior da distribuição normal

O corpo da tabela dá as probabilidades unilaterais sob H_0 de z . A coluna marginal do lado esquerdo dá vários valores de z com uma casa decimal. A linha do topo dá vários valores para a segunda casa decimal. Então, por exemplo, o p unilateral de $z \geq 0,11$ ou $z \leq -0,11$ é $p = 0,4562$.

z	0,00	0,01	0,02	0,03	0,04	0,05	0,06	0,07	0,08	0,09
0,0	0,5000	0,4960	0,4920	0,4880	0,4840	0,4801	0,4761	0,4721	0,4681	0,4641
0,1	0,4602	0,4562	0,4522	0,4483	0,4443	0,4404	0,4364	0,4325	0,4286	0,4247
0,2	0,4207	0,4168	0,4129	0,4090	0,4052	0,4013	0,3974	0,3936	0,3897	0,3859
0,3	0,3821	0,3783	0,3745	0,3707	0,3669	0,3632	0,3594	0,3557	0,3520	0,3483
0,4	0,3446	0,3409	0,3372	0,3336	0,3300	0,3264	0,3228	0,3192	0,3156	0,3121
0,5	0,3085	0,3050	0,3015	0,2981	0,2946	0,2912	0,2877	0,2843	0,2810	0,2776
0,6	0,2743	0,2709	0,2676	0,2643	0,2611	0,2578	0,2546	0,2514	0,2483	0,2451
0,7	0,2420	0,2389	0,2358	0,2327	0,2296	0,2266	0,2236	0,2206	0,2177	0,2148
0,8	0,2119	0,2090	0,2061	0,2033	0,2005	0,1977	0,1949	0,1922	0,1894	0,1867
0,9	0,1841	0,1814	0,1788	0,1762	0,1736	0,1711	0,1685	0,1660	0,1635	0,1611
1,0	0,1587	0,1562	0,1539	0,1515	0,1492	0,1469	0,1446	0,1423	0,1401	0,1379
1,1	0,1357	0,1335	0,1314	0,1292	0,1271	0,1251	0,1230	0,1210	0,1190	0,1170
1,2	0,1151	0,1131	0,1112	0,1093	0,1075	0,1056	0,1038	0,1020	0,1003	0,0985
1,3	0,0968	0,0951	0,0934	0,0918	0,0901	0,0885	0,0869	0,0853	0,0838	0,0823
1,4	0,0808	0,0793	0,0778	0,0764	0,0749	0,0735	0,0721	0,0708	0,0694	0,0681
1,5	0,0668	0,0655	0,0643	0,0630	0,0618	0,0606	0,0594	0,0582	0,0571	0,0559
1,6	0,0548	0,0537	0,0526	0,0516	0,0505	0,0495	0,0485	0,0475	0,0465	0,0455
1,7	0,0446	0,0436	0,0427	0,0418	0,0409	0,0401	0,0392	0,0384	0,0375	0,0367
1,8	0,0359	0,0351	0,0344	0,0336	0,0329	0,0322	0,0314	0,0307	0,0301	0,0294
1,9	0,0287	0,0281	0,0274	0,0268	0,0262	0,0256	0,0250	0,0244	0,0239	0,0233
2,0	0,0228	0,0222	0,0217	0,0212	0,0207	0,0202	0,0197	0,0192	0,0188	0,0183
2,1	0,0179	0,0174	0,0170	0,0166	0,0162	0,0158	0,0154	0,0150	0,0146	0,0143
2,2	0,0139	0,0136	0,0132	0,0129	0,0125	0,0122	0,0119	0,0116	0,0113	0,0110
2,3	0,0107	0,0104	0,0102	0,0099	0,0096	0,0094	0,0091	0,0089	0,0087	0,0084
2,4	0,0082	0,0080	0,0078	0,0075	0,0073	0,0071	0,0069	0,0068	0,0066	0,0064
2,5	0,0062	0,0060	0,0059	0,0057	0,0055	0,0054	0,0052	0,0051	0,0049	0,0048
2,6	0,0047	0,0045	0,0044	0,0043	0,0041	0,0040	0,0039	0,0038	0,0037	0,0036
2,7	0,0035	0,0034	0,0033	0,0032	0,0031	0,0030	0,0029	0,0028	0,0027	0,0026
2,8	0,0026	0,0025	0,0024	0,0023	0,0023	0,0022	0,0021	0,0021	0,0020	0,0019
2,9	0,0019	0,0018	0,0018	0,0017	0,0016	0,0016	0,0015	0,0015	0,0014	0,0014
3,0	0,0013	0,0013	0,0013	0,0012	0,0012	0,0011	0,0011	0,0011	0,0010	0,0010
3,1	0,0010	0,0009	0,0009	0,0009	0,0008	0,0008	0,0008	0,0008	0,0007	0,0007
3,2	0,0007									
3,3	0,0005									
3,4	0,0003									
3,5	0,00023									
3,6	0,00016									
3,7	0,00011									
3,8	0,00007									
3,9	0,00005									
4,0	0,00003									

Níveis de significância selecionados para a distribuição normal.

Bilateral α	0,20	0,10	0,05	0,02	0,01	0,002	0,001	0,0001	0,00001
Unilateral α	0,10	0,05	0,025	0,01	0,005	0,001	0,0005	0,00005	0,000005
z	1,282	1,645	1,960	2,326	2,576	3,090	3,291	3,891	4,417

Apêndice B

COMPARAÇÃO DOS MÉTODOS PROPOSTOS

Como foi mencionado no capítulo 6, foram implementados quatro algoritmos para a geração genética multiobjetivo de classificadores *fuzzy*. Dois dos algoritmos implementados fazem a geração da BR: o primeiro utiliza a abordagem IRL e a técnica de otimização multiobjetivo NSGA-II, chamado de IRL-NSGA-II, e o segundo utiliza a abordagem IRL e a técnica de otimização multiobjetivo SPEA2, chamado de IRL-SPEA2. Os outros algoritmos implementados fazem otimização da BD: o primeiro utiliza a técnica de otimização multiobjetivo NSGA-II, chamado de OM-NSGA-II, e o segundo utiliza a técnica de otimização multiobjetivo SPEA2, chamado de OM-SPEA2.

Este anexo mostra com mais detalhes a comparação dos métodos propostos para a geração da BR, IRL-NSGA-II e IRL-SPEA2, com os métodos NSLV, NEEMF e MESDIF. E os métodos propostos para a otimização da BD, OM-NSGA-II e OM-SPEA2, com os métodos que utilizam uma otimização genética, definidos como AG-1 e AG-2. Para todas as comparações foi utilizado o teste de postos com sinal de Wilcoxon apresentado no Anexo A.

A comparação dos métodos de geração de regras é detalhada na seção 1, e a comparação dos métodos de otimização da BD é detalhada na seção 2.

1. Comparação dos métodos para a geração da BR

Os métodos de geração genética multiobjetivo propostos da BR, IRL-NSGA-II e IRL-SPEA2, foram comparados com três métodos de geração da BR encontrados na literatura, NSLV, NMEEF e MESDIF. É utilizado o teste de postos com sinal de Wilcoxon para fazer a comparação do método proposto IRL-NSGA-II com os outros

quatro métodos e a comparação do método proposto IRL-SPEA2 com os outros quatro métodos.

Foram comparadas três características dos métodos de geração da BR: a média da taxa de erro, a média da quantidade de regras na BR final e a média da quantidade de condições na BR final. O Conjunto de Dados (CD) e os valores para as três métricas são os mesmos apresentados no capítulo 6.

A primeira comparação feita foi entre o método de geração da BR proposto IRL-NSGA-II com os outros quatro métodos: IRL-SPEA2, NSLV, NMEEF e MESDIF. Para esta comparação foi calculada a diferença (d) entre os resultados, assim como o posto das diferenças, ou seja, o posto 1 ao menor valor absoluto de d , o posto 2 ao segundo valor absoluto, e assim por diante. Seguidamente é realizada a soma dos postos d positivos (T^+) e a soma dos pontos d negativos (T^-). As tabelas B.1, B.2, B.3 e B.4 mostram a comparação da média da taxa de erro entre o método IRL-NSGA-II com os métodos IRL-SPEA2, NSLV, NMEEF e MESDIF respectivamente.

Tabela B.1: Comparação da média da taxa de erro IRL-NSGA-II e IRL-SPEA2

CD	IRL NSGA-II	IRL-SPEA2	d	Posto d
iris	2,7	2,7	0,0	-
wine	7,3	7,4	0,1	1
thyroid	3,8	3,8	0,0	-
heart_s	25,7	23,2	-2,5	-5
sonar	25,9	27,2	1,3	3
bupa	40,4	38,3	-2,1	-4
w_diag_breast_c	4,8	5,2	0,4	2
Soma T^+				6
Soma T^-				-9

Tabela B.2: Comparação da média da taxa de erro IRL-NSGA-II e NSLV

CD	IRL NSGA-II	NSLV	d	Posto d
iris	2,7	6,1	3,4	6
wine	7,3	6,2	-1,1	-1
thyroid	3,8	9,3	5,5	7
heart_s	25,7	24,5	-1,2	-2
sonar	25,9	27,9	2,0	5
bupa	40,4	42,1	1,7	4
w_diag_breast_c	4,8	6,2	1,4	3
Soma T^+				25
Soma T^-				-3

Tabela B.3 – Comparação da média da taxa de erro IRL-NSGA-II e NMEEF

CD	IRL NSGA-II	NMEEF	d	Posto d
iris	2,7	9,4	6,7	2
wine	7,3	27,9	20,6	7
thyroid	3,8	14,8	11,0	4
heart_s	25,7	40,0	14,3	5
sonar	25,9	34,7	8,8	3
bupa	40,4	42,1	1,7	1
w_diag_breast_c	4,8	22,6	17,8	6
Soma T⁺				28
Soma T⁻				0

Tabela B.4: Comparação da média da taxa de erro IRL-NSGA-II e MESDIF

CD	IRL NSGA-II	MESDIF	d	Posto d
iris	2,7	4,0	1,3	1
wine	7,3	15,3	8,0	3,5
thyroid	3,8	13,3	9,5	5
heart_s	25,7	27,8	2,1	2
sonar	25,9	47,6	21,7	6
bupa	40,4	48,4	8,0	3,5
w_diag_breast_c	4,8	36,8	32,0	7
Soma T⁺				28
Soma T⁻				0

Ocasionalmente duas diferenças d podem ser iguais, para esse caso esses bancos de dados são retirados da análise e o tamanho das amostras N é reduzido, o que aconteceu na tabela B.1 nos conjuntos de dados *iris* e *thyroid*. Outra espécie de empate pode ocorrer, quando dois ou mais d 's podem ser de mesma magnitude, então é atribuído o mesmo posto para esses conjuntos de dados. Esse posto atribuído é a média dos postos que teriam sido atribuídos se os d 's tivessem diferido levemente, o que aconteceu com os conjuntos de dados *wine* e *bupa* na tabela B.4, onde a média dos postos 3 e 4, ou seja 3,5, foi atribuída a cada conjunto de dados. Nesse caso foi atribuído o posto 5 ao próximo d maior, ou seja *thyroid*, porque os postos 3 e 4 já tinham sido atribuídos.

Na comparação feita pelo teste de postos com sinal de Wilcoxon é considerada como hipótese nula H_0 que não há diferença significativa entre as taxas médias de erro dos métodos comparados, ou seja, entre IRL-NSGA-II com IRL-SPEA2, NSLV, NMEEF e/ou MESDIF. Como hipótese alternativa H_1 que a taxa média de erro do método IRL-NSGA-II é menor que a taxa média de erro do método IRL-SPEA2, NSLV, NMEEF e/ou MESDIF.

Para rejeitar H_0 em favor de H_1 , o teste estatístico deve dar um valor cuja probabilidade associada de ocorrência sob H_0 é menor ou igual a uma pequena probabilidade, usualmente denotada por α . Esta probabilidade é chamada nível de significância. Um valor comum para α é 0,05 (SIDNEY; CASTELLAN, 2006), que será utilizado neste projeto.

Seja $\alpha = 0,05$ e N o número de conjuntos de dados. Considerando a Tabela A.1 do Apêndice A, que contém valores de probabilidades da distribuição amostral de T^+ (c) para $N \leq 15$, e os valores das tabelas B.1, B.2, B.3 e B.4, na primeira comparação, entre IRL-NSGA-II e IRL-SPEA2 não podemos concluir nada porque o valor de T^+ é menor que T^- . Assim, é preciso fazer a comparação contrária. Nas três comparações seguintes a hipótese H_0 é rejeitada em favor de H_1 no nível utilizado $\alpha = 0,05$, pois as probabilidades tabeladas 0,0391, 0,0078 e 0,0078 são menores que α . Então é concluído que a média da taxa de erro do método IRL-NSGA-II é menor que as médias das taxas de erro dos métodos NSLV, NMEEF e MESDIF.

O mesmo teste de comparação foi realizado entre o método proposto IRL-SPEA2 e os métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF. Os valores das diferenças e os postos das diferenças para cada comparação são mostrados nas Tabelas B.5, B.6, B.7 e B.8 respectivamente.

Tabela B.5: Comparação da média da taxa de erro IRL-SPEA2 e IRL-NSGA-II

CD	IRL-SPEA2	IRL-NSGA-II	d	Posto d
iris	2,7	2,7	0,0	-
wine	7,4	7,3	-0,1	-1
thyroid	3,8	3,8	0,0	-
heart_s	23,2	25,7	2,5	5
sonar	27,2	25,9	-1,3	-3
bupa	38,3	40,4	2,1	4
w_diag_breast_c	5,2	4,8	-0,4	-2
Soma T^+				9
Soma T^-				-6

Tabela B.6: Comparação da média da taxa de erro IRL-SPEA2 e NSLV

CD	IRL-SPEA2	NSLV	d	Posto d
iris	2,7	6,1	3,4	5
wine	7,4	6,2	-1,2	-3
thyroid	3,8	9,3	5,5	7
heart_s	23,2	24,5	1,3	4
sonar	27,2	27,9	0,7	1
bupa	38,3	42,1	3,8	6
w_diag_breast_c	5,2	6,2	1,0	2
Soma T^+				25
Soma T^-				-3

Tabela B.7: Comparação da média da taxa de erro IRL- SPEA2 e NMEEF

CD	IRL-SPEA2	NMEEF	d	Posto d
iris	2,7	9,4	6,7	2
wine	7,4	27,9	20,5	7
thyroid	3,8	14,8	11,0	4
heart_s	23,2	40,0	16,8	5
sonar	27,2	34,7	7,5	3
bupa	38,3	42,1	3,8	1
w_diag_breast_c	5,2	22,6	17,4	6
Soma T⁺				28
Soma T⁻				0

Tabela B.8: Comparação da média da taxa de erro IRL- SPEA2 e MESDIF

CD	IRL-SPEA2	MESDIF	d	Posto d
iris	2,7	4,0	1,3	1
wine	7,4	15,3	7,9	3
thyroid	3,8	13,3	9,5	4
heart_s	23,2	27,8	4,6	2
sonar	27,2	47,6	20,4	6
bupa	38,3	48,4	10,1	5
w_diag_breast_c	5,2	36,8	31,6	7
Soma T⁺				28
Soma T⁻				0

Como na comparação anterior, seja $\alpha = 0,05$ e N o número de conjuntos de dados. Usando a Tabela A.1 do Apêndice A e os valores das Tabelas B.5, B.6, B.7 e B.8, na primeira comparação, entre IRL-SPEA2 e IRL-NSGA-II a hipótese H_0 não é rejeitada porque a probabilidade tabelada é 0,4063 com $N = 5$, que é maior que $\alpha = 0,05$. Concluímos que as médias das taxas de erro dos métodos IRL-SPEA2 e IRL-NSGA-II não tem diferença significativa. Nas três comparações seguintes a hipótese H_0 é rejeitada em favor de H_1 no nível utilizado $\alpha = 0,05$, pois as probabilidades tabeladas 0,0391, 0,0078 e 0,0078 são menores que α . Então é concluído que a média da taxa de erro do método IRL-SPEA2 é menor que as médias das taxas de erro dos métodos NSLV, NMEEF e MESDIF.

O mesmo tipo de comparação foi realizado focando o desempenho dos métodos de geração da BR em relação ao número de regras. A comparação da média da quantidade de regras na BR final foi feita entre o método IRL-NSGA-II com os métodos IRL-SPEA2, NSLV, NMEEF e MESDIF. Os valores das diferenças e os

postos das diferenças para cada comparação são mostrados nas Tabelas B.9, B.10, B.11 e B.12 respectivamente.

Tabela B.10: Comparação da média da quantidade de regras IRL-NSGA-II e IRL-SPEA2

CD	IRL-NSGA-II	IRL-SPEA2	d	Posto d
iris	4,0	4,0	0,0	-
wine	6,1	4,5	-1,6	-3,5
thyroid	4,9	5,0	0,1	1,5
heart_s	20,6	8,6	-12,0	-6
sonar	8,2	8,3	0,1	1,5
bupa	6,6	5,0	-1,6	-3,5
w_diag_breast_c	6,3	4,2	-2,1	-5
Soma T⁺				3
Soma T⁻				-18

Tabela B.11: Comparação da média da quantidade de regras IRL-NSGA-II e NSLV

CD	IRL-NSGA-II	NSLV	d	Posto d
iris	4,0	4,0	0,0	-
wine	6,1	4,5	-1,6	-2,5
thyroid	4,9	5,0	0,1	1,5
heart_s	20,6	8,6	-12,0	-6
sonar	8,2	8,3	0,1	1,5
bupa	6,6	5,0	-1,6	-2,5
w_diag_breast_c	6,3	4,2	-2,1	-5
Soma T⁺				3
Soma T⁻				-18

Tabela B.12: Comparação da média da quantidade de regras IRL-NSGA-II e NMEEF

CD	IRL-NSGA-II	NMEEF	d	Posto d
iris	4,0	3,8	-0,2	-1
wine	6,1	3,1	-3,0	-4
thyroid	4,9	7,5	2,6	3
heart_s	20,6	3,3	-17,3	-7
sonar	8,2	10,5	2,3	2
bupa	6,6	2,3	-4,3	-6
w_diag_breast_c	6,3	9,9	3,6	5
Soma T⁺				10
Soma T⁻				-18

Tabela B.13: Comparação da média da quantidade de regras IRL-NSGA-II e MESDIF

CD	IRL-NSGA-II	MESDIF	d	Posto d
iris	4,0	9,0	5,0	5
wine	6,1	9,0	2,9	3
thyroid	4,9	9,0	4,1	4
heart_s	20,6	6,0	-14,6	-7
sonar	8,2	15,0	6,8	6
bupa	6,6	6,0	-0,6	-2
w_diag_breast_c	6,3	6,0	-0,3	-1
Soma T⁺				18
Soma T⁻				-10

Na comparação feita com a média da quantidade de regras pelo teste de postos com sinal de Wilcoxon é considerada como hipótese nula H_0 que não há diferença entre as médias das quantidades de regras dos métodos comparados, ou seja, entre IRL-NSGA-II com IRL-SPEA2, NSLV, NMEEF e/ou MESDIF. Como hipótese alternativa H_1 é considerado que a média da quantidade de regras do método IRL-NSGA-II é menor que a média da quantidade de regras dos métodos IRL-SPEA2, NSLV, NMEEF e/ou MESDIF.

Seja $\alpha = 0,05$ e N o número de conjuntos de dados, Usando a Tabela A.1 do Apêndice A e os valores das Tabelas B.10, B.11, B.12 e B.13, na primeira comparação, entre IRL-NSGA-II e IRL-SPEA2 não se pode concluir nada porque o valor de T^+ é menor que T^- . Assim, é preciso fazer a comparação contrária, que será realizada na segunda comparação da média da quantidade de regras. Nas duas comparações seguintes também o valor de T^+ é menor que T^- . As tabelas da comparação contrária, ou seja, comparar NSLV com IRL-NSGA-II e NMEEF com IRL-NSGA-II, não será mostradas. Para essas duas comparações o valor de T^+ é 18, logo a hipóteses H_0 não é rejeitada porque a probabilidade tabelada é 0,0781 e 0,2891 que não são menores que $\alpha = 0,05$, sendo que $N=6$ e $N=7$ para cada comparação respectivamente. Para a comparação entre IRL-NSGA-II e MESDIF, a hipóteses H_0 também não é rejeitada, pois a probabilidade tabelada 0,2891 não é menor que $\alpha = 0,05$. Então é concluído que a média da quantidade de regras do método IRL-NSGA-II não tem diferença significativa com relação aos métodos NSLV, NMEEF e MESDIF.

O mesmo teste de comparação para a média da quantidade de regras foi realizado entre o método proposto IRL-SPEA2 e os métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF. Os valores das diferenças e os postos das diferenças para cada comparação são mostrados nas Tabelas B.14, B.15, B.16 e B.17 respectivamente.

Tabela B.14: Comparação da média da quantidade de regras IRL-SPEA2 e IRL-NSGA-II

CD	IRL-SPEA2	IRL-NSGA-II	d	Posto d
iris	4,0	4,0	0,0	-
wine	4,5	6,1	1,6	4
thyroid	5,0	4,9	-0,1	-1,5
heart_s	8,6	20,6	12,0	6
sonar	8,3	8,2	-0,1	-1,5
bupa	5,0	6,6	1,6	3
w_diag_breast_c	4,2	6,3	2,1	5
Soma T⁺				18
Soma T⁻				-3

Tabela B.15: Comparação da média da quantidade de regras IRL-SPEA2 e NSLV

CD	IRL-SPEA2	NSLV	d	Posto d
iris	4,0	4,0	0,0	-
wine	6,1	4,5	-1,6	-4
thyroid	4,3	5,0	0,7	2
heart_s	16,7	8,6	-8,1	-6
sonar	7,8	8,3	0,5	1
bupa	6,1	5,0	-1,1	-3
w_diag_breast_c	6,2	4,2	-2,0	-5
Soma T⁺				3
Soma T⁻				-18

Tabela B.16: Comparação da média da quantidade de regras IRL- SPEA2 e NMEEF

CD	IRL-SPEA2	NMEEF	d	Posto d
iris	4,0	3,8	-0,2	-1
wine	6,1	3,1	-3,0	-3
thyroid	4,3	7,5	3,2	4
heart_s	16,7	3,3	-13,4	-7
sonar	7,8	10,5	2,7	2
bupa	6,1	2,3	-3,8	-6
w_diag_breast_c	6,2	9,9	3,7	5
Soma T⁺				11
Soma T⁻				-17

Tabela B.17: Comparação da média da quantidade de regras IRL- SPEA2 e MESDIF

CD	IRL-SPEA2	MESDIF	d	Posto d
iris	4,0	9,0	5,0	5
wine	6,1	9,0	2,9	3
thyroid	4,3	9,0	4,7	4
heart_s	16,7	6,0	-10,7	-7
sonar	7,8	15,0	7,2	6
bupa	6,1	6,0	-0,1	-1
w_diag_breast_c	6,2	6,0	-0,2	-2
Soma T⁺				18
Soma T⁻				-10

Como na comparação anterior, seja $\alpha = 0,05$ e N o número de conjuntos de dados. Usando a Tabela A.1 do Apêndice A e os valores das Tabelas B.14, B.15, B.16 e B.17, na primeira comparação, entre IRL-SPEA2 e IRL-NSGA-II a hipótese H_0 não é rejeitada porque a probabilidade tabelada é 0,0781 com $N = 6$, é maior que $\alpha = 0,05$. Nas duas comparações seguintes o valor de T^+ é menor que T^- . As tabelas de comparação contrárias, ou seja, comparação de NSLV com IRL-SPEA2 e NMEEF com IRL-SPEA2, não serão mostradas. Para essas comparações o valor de T^+ é 18 e 17 com $N = 6$ e $N = 7$ respectivamente. Assim, a hipótese H_0 para as duas comparações não é rejeitada porque as probabilidades tabeladas são 0,0781 e 0,3438 e não são menores que $\alpha = 0,05$. Para a comparação entre IRL-SPEA2 e MESDIF, a hipótese H_0 também não é rejeitada, pois a probabilidade tabelada 0,2891 não é menor que $\alpha = 0,05$. Então é concluído que a média da quantidade de regras do método IRL-SPEA2 não tem diferença significativa com relação aos métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF.

A última comparação dos métodos para a geração da BR foi realizado focando o desempenho dos métodos em relação à quantidade de condições da BR final. A comparação da média da quantidade de condições na BR final foi feita entre o método proposto IRL-NSGA-II com os métodos IRL-SPEA2, NSLV, NMEEF e MESDIF. Os valores das diferenças e os postos das diferenças para cada comparação são mostrados nas Tabelas B.18, B.19, B.20 e B.21 respectivamente.

Tabela B.18: Comparação da média da quantidade de condições IRL-NSGA-II e IRL-SPEA2

CD	IRL-NSGA-II	IRL-SPEA2	d	Posto d
iris	6,0	6,0	0,0	-
wine	9,1	9,1	0,0	-
thyroid	7,4	6,6	-0,8	-2
heart_s	64,5	49,8	-14,7	-5
sonar	18,1	15,8	-2,3	-4
bupa	11,6	10,6	-1,0	-3
w_diag_breast_c	8,5	9,2	0,7	1
Soma T^+				1
Soma T^-				-14

Tabela B.19: Comparação da média da quantidade de condições IRL-NSGA-II e NSLV

CD	IRL-NSGA -II	NSLV	d	Posto d
iris	6,0	3,5	-2,5	-1
wine	9,1	14,5	5,4	4
thyroid	7,4	11,5	4,1	2
heart_s	64,5	23,5	-41,0	-7
sonar	18,1	49,7	31,6	6
bupa	11,6	16,1	4,5	3
w_diag_breast_c	8,5	18,6	10,1	5
Soma T⁺				20
Soma T⁻				-8

Tabela B.20: Comparação da média da quantidade de condições IRL-NSGA-II e NMEEF

CD	IRL-NSGA -II	NMEEF	d	Posto d
iris	6,0	13,8	7,8	2
wine	9,1	20,1	11,0	4
thyroid	7,4	15,8	8,4	3
heart_s	64,5	12,3	-52,2	-5
sonar	18,1	76,7	58,6	6
bupa	11,6	13,6	2,0	1
w_diag_breast_c	8,5	76,0	67,5	7
Soma T⁺				23
Soma T⁻				-5

Tabela B.21: Comparação da média da quantidade de condições IRL-NSGA-II e MESDIF

CD	IRL-NSGA -II	MESDIF	d	Posto d
iris	6,0	13,3	7,3	3
wine	9,1	15,3	6,2	2
thyroid	7,4	19,2	11,8	4
heart_s	64,5	12,4	-52,1	-7
sonar	18,1	60,5	42,4	6
bupa	11,6	11,0	-0,6	-1
w_diag_breast_c	8,5	49,2	40,7	5
Soma T⁺				20
Soma T⁻				-8

Na comparação feita para a média da quantidade de condições pelo teste de postos com sinal de Wilcoxon é considerada como hipótese nula H_0 que não há diferença entre as médias das quantidades de condições dos métodos comparados, ou seja, entre IRL-NSGA-II com IRL-SPEA2, NSLV, NMEEF e/ou MESDIF. Como hipótese alternativa H_1 é considerado que a média da quantidade de regras do método IRL-NSGA-II é menor que a média da quantidade de regras dos métodos IRL-SPEA2, NSLV, NMEEF e/ou MESDIF.

Seja $\alpha = 0,05$ e N o número de conjuntos de dados. Usando a Tabela A.1 do Apêndice A e os valores das Tabelas B.18, B.19, B.20 e B.21, na primeira comparação, entre IRL-NSGA-II e IRL-SPEA2 não podemos concluir nada porque o valor de T^+ é menor que T^- . Logo, é preciso fazer a comparação contrária, que será realizada na segunda comparação da média da quantidade de condições. Nas três comparações seguintes a hipóteses H_0 não é rejeitada, pois as probabilidades tabeladas 0,1875, 0,0781 e 0,1875 não são menores que $\alpha = 0,05$. Então é concluído que a média da quantidade de condições do método IRL-NSGA-II não tem diferença significativa com relação aos métodos NSLV, NMEEF e MESDIF.

O mesmo teste de comparação para a média da quantidade de condições foi realizado entre o método proposto IRL-SPEA2 e os métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF. Os valores das diferenças e os postos das diferenças para cada comparação são mostrados nas Tabelas B.22, B.23, B.24 e B.25 respectivamente.

Tabela B.22: Comparação da média da quantidade de condições IRL-SPEA2 e IRL-NSGA-II

CD	IRL-SPEA2	IRL-NSGA -II	d	Posto d
iris	6,0	6,0	0,0	-
wine	9,1	9,1	0,0	-
thyroid	6,6	7,4	0,8	2
heart_s	49,8	64,5	14,7	5
sonar	15,8	18,1	2,3	4
bupa	10,6	11,6	1,0	3
w_diag_breast_c	9,2	8,5	-0,7	-1
Soma T^+				14
Soma T^-				-1

Tabela B.23: Comparação da média quantidade de condições IRL-SPEA2 e NSLV

CD	IRL-SPEA2	NSLV	d	Posto d
iris	6,0	3,5	-2,5	-1
wine	9,1	14,5	5,4	3
thyroid	6,6	11,5	4,9	2
heart_s	49,8	23,5	-26,3	-6
sonar	15,8	49,7	33,9	7
bupa	10,6	16,1	5,5	4
w_diag_breast_c	9,2	18,6	9,4	5
Soma T^+				21
Soma T^-				-7

Tabela B.24: Comparação da média quantidade de condições IRL-SPEA2 e NMEEF

CD	IRL-SPEA2	NMEEF	d	Posto d
iris	6,0	13,8	7,8	2
wine	9,1	20,1	11,0	4
thyroid	6,6	15,8	9,2	3
heart_s	49,8	12,3	-37,5	-5
sonar	15,8	76,7	60,9	6
bupa	10,6	13,6	3,0	1
w_diag_breast_c	9,2	76,0	66,8	7
Soma T⁺				23
Soma T⁻				-5

Tabela B.25: Comparação da média quantidade de condições IRL-SPEA2 e MESDIF

CD	IRL-SPEA2	MESDIF	d	Posto d
iris	6,0	13,3	7,3	3
wine	9,1	15,3	6,2	2
thyroid	6,6	19,2	12,6	4
heart_s	49,8	12,4	-37,4	-5
sonar	15,8	60,5	44,7	7
bupa	10,6	11,0	0,4	1
w_diag_breast_c	9,2	49,2	40,0	6
Soma T⁺				23
Soma T⁻				-5

Como na comparação anterior, seja $\alpha = 0,05$ e N o número de conjuntos de dados. Usando a Tabela A.1 do Apêndice e os valores das Tabelas B.22, B.23, B.24 e B.25, na primeira comparação, entre IRL-SPEA2 e IRL-NSGA-II foi usado o valor de $N = 5$, e nas outras comparações, o valor de $N = 7$. Nas quatro comparações a hipóteses H_0 não é rejeitada, pois as probabilidades tabeladas 0,0625, 0,1484, 0,0781 e 0,0781 não são menores que $\alpha = 0,05$. Então é concluído que a média da quantidade de condições do método IRL-SPEA2 não tem diferença significativa com relação aos métodos IRL-NSGA-II, NSLV, NMEEF e MESDIF.

2. Comparação dos métodos para a geração da BR

Os métodos propostos de otimização genética multiobjetivo da BD, OM-NSGA-II e OM-SPEA2, foram comparados com o método de otimização genética da BD apresentado em (PIMENTA, 2009). Esse método foi rodado utilizando duas BR: uma obtida pelo método proposto IRL-NSGA-II e a outra obtida pelo método proposto IRL-SPEA2. Os resultados para esses métodos de otimização são chamados AG-1 e AG-2 respectivamente. É utilizado o teste de postos com sinal de Wilcoxon para fazer a comparação do método proposto OM-NSGA-II com o método

genético AG-1 e a comparação do método proposto OM-SPEA2 com o método genético AG-2.

Foram comparadas três características dos métodos de otimização da BD: a melhora da média da taxa de erro, a média do Índice de Interpretabilidade Semântica (IIS) e a média da quantidade de BD Transparentes. Os Conjuntos de Dados (CD) e os valores para as três métricas são os mesmos que foram apresentados no capítulo 6.

A primeira comparação foi entre o método proposto OM-NSGA-II e o método AG-1. Os valores das diferenças e os postos das diferenças para cada característica mencionada anteriormente são mostrados nas tabelas B.26, B.27 e B.28.

Tabela B.26: Comparação da média da melhora da taxa de erro OM-NSGA-II e AG-1

CD	OM-NSGA-II	AG-1	d	Posto d
iris	0,0	0,0	0,0	-
wine	2,7	3,9	-1,2	-6
thyroid	0,6	0,8	-0,3	-2,5
heart_s	6,0	5,7	0,3	2,5
sonar	2,4	3,0	-0,6	-5
bupa	4,7	5,1	-0,4	-4
w_diag_breast_c	0,5	0,7	-0,2	-1
Soma T⁺				2,5
Soma T⁻				-18,5

Tabela B.27: Comparação da média do IIS OM-NSGA-II e AG-1

CD	OM-NSGA-II	AG-1	d	Posto d
iris	1,00	1,00	0,00	-
wine	0,76	0,50	0,26	6
thyroid	0,94	0,82	0,12	5
heart_s	0,75	0,72	0,03	1,5
sonar	0,73	0,70	0,03	1,5
bupa	0,60	0,50	0,10	4
w_diag_breast_c	0,79	0,74	0,05	3
Soma T⁺				21
Soma T⁻				0

Tabela B.28: Comparação da média da quantidade de BD transparentes OM-NSGA-II e AG-1

CD	OM-NSGA-II	AG-1	d	Posto d
iris	1,00	1,00	0,00	-
wine	1,00	0,46	0,54	5
thyroid	1,00	0,78	0,22	1
heart_s	1,00	0,60	0,40	3
sonar	1,00	0,58	0,42	4
bupa	1,00	0,45	0,55	6
w_diag_breast_c	1,00	0,70	0,30	2
Soma T⁺				21
Soma T⁻				0

Seja $\alpha = 0,05$ e N o número de conjuntos de dados. Usando a Tabela A.1 do Apêndice A e os valores das Tabelas B.26, B.27 e B.28 Na primeira comparação, entre OM-NSGA-II e AG-1 não é possível concluir nada porque o valor de T^- é maior que T^+ . Se for realizada a comparação contrária, ou seja, entre AG-1 e OM-NSGA-II, o valor de T^+ seria de 18,5 e o valor de T^- seria 2,5. Nesse caso com $N=6$, a hipótese H_0 não é rejeitada porque a probabilidade tabelada é aproximadamente 0,0625, portanto maior que $\alpha = 0,05$. Assim concluímos que as médias das melhoras taxas de erro dos métodos OM-NSGA-II e AG-1 não tem diferença significativa. Nas duas comparações seguintes a hipótese H_0 é rejeitada em favor de H_1 no nível utilizado $\alpha = 0,05$, pois com $N = 6$ as probabilidades tabeladas 0,0156, para as duas comparações, são menores que $\alpha = 0,05$. Então é concluído que a média do índice de interpretabilidade semântica e a média da quantidade de BD Transparentes do método OM-NSGA-II é maior que essas médias do índice de interpretabilidade semântica e BD transparentes do método AG-1.

A segunda comparação foi entre o método proposto OM-SPEA2 e o método AG-2. Os valores das diferenças e os postos das diferenças para cada característica mencionada anteriormente são mostrados nas Tabelas B.29, B.30 e B.31.

Tabela B.29: Comparação da média da melhora da taxa de erro OM-SPEA2 e AG-2

CD	OM-SPEA2	AG-2	d	Posto d
iris	0,0	0,0	0,0	-
wine	4,1	4,0	0,1	1
thyroid	1,0	1,0	0,0	-
heart_s	2,8	2,1	0,7	5
sonar	2,9	3,1	-0,2	-2,5
bupa	2,3	2,0	0,3	4
w_diag_breast_c	0,6	0,8	-0,2	-2,5
Soma T^+				10
Soma T^-				-5

Tabela B.30: Comparação da média do IIS OM-SPEA2 e AG-2

CD	OM-SPEA2	AG-2	d	Posto d
iris	1,00	1,00	0,00	-
wine	0,49	0,46	0,03	3,5
thyroid	0,88	0,80	0,08	5
heart_s	0,65	0,64	0,01	1
sonar	0,66	0,64	0,02	2
bupa	0,70	0,58	0,12	6
w_diag_breast_c	0,71	0,68	0,03	3,5
Soma T^+				21
Soma T^-				0

Tabela B.31: Comparação da média da quantidade de BD Transparentes OM-SPEA2 e AG-2

CD	OM-SPEA2	AG-2	d	Posto d
iris	1,00	1,00	0,00	-
wine	1,00	0,43	0,57	4
thyroid	1,00	0,81	0,19	1
heart_s	1,00	0,42	0,58	5
sonar	1,00	0,57	0,43	3
bupa	1,00	0,39	0,61	6
w_diag_breast_c	1,00	0,69	0,31	2
Soma T⁺				21
Soma T⁻				0

Seja $\alpha = 0,05$ e N o número de conjuntos de dados. Usando a Tabela A.1 do Apêndice A e os valores das Tabelas B.29, B.30 e B.31, na primeira comparação, entre OM-SPEA2 e AG-2, com $N = 6$, a hipótese H_0 não é rejeitada porque a probabilidade tabelada é 1 que é maior que $\alpha = 0,05$. Assim, concluímos que as médias das melhores taxas de erro dos métodos OM-SPEA2 e AG-2 não tem diferença significativa. Nas duas comparações seguintes a hipótese H_0 é rejeitada em favor de H_1 no nível utilizado $\alpha = 0,05$ com $N = 6$, pois as probabilidades tabeladas 0,0156 para as duas comparações são menores que $\alpha = 0,05$. Então é concluído que a média do IIS e a média da quantidade de BD Transparentes do método OM-SPEA2 são maiores que as médias do método AG-2.